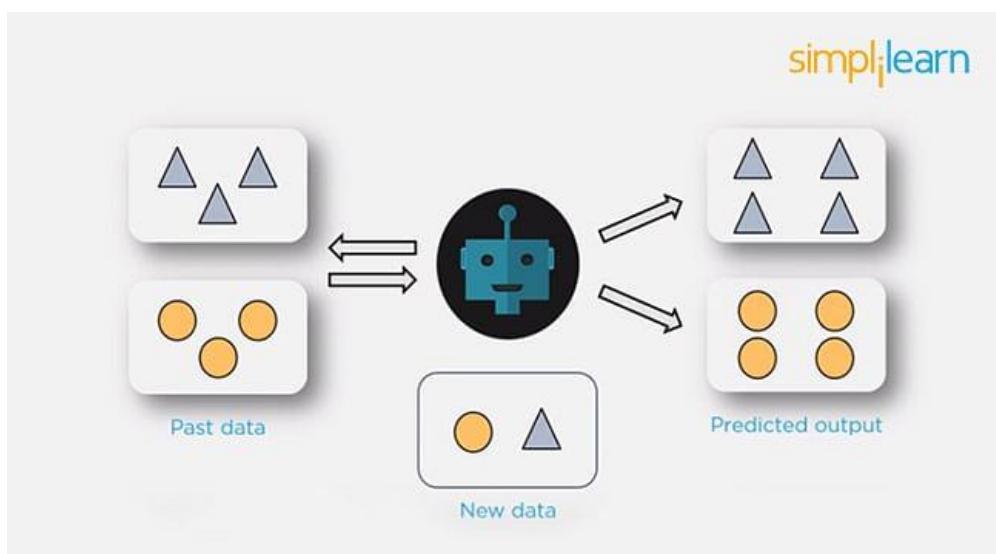


## 1. What Are the Different Types of Machine Learning?

There are three types of machine learning:

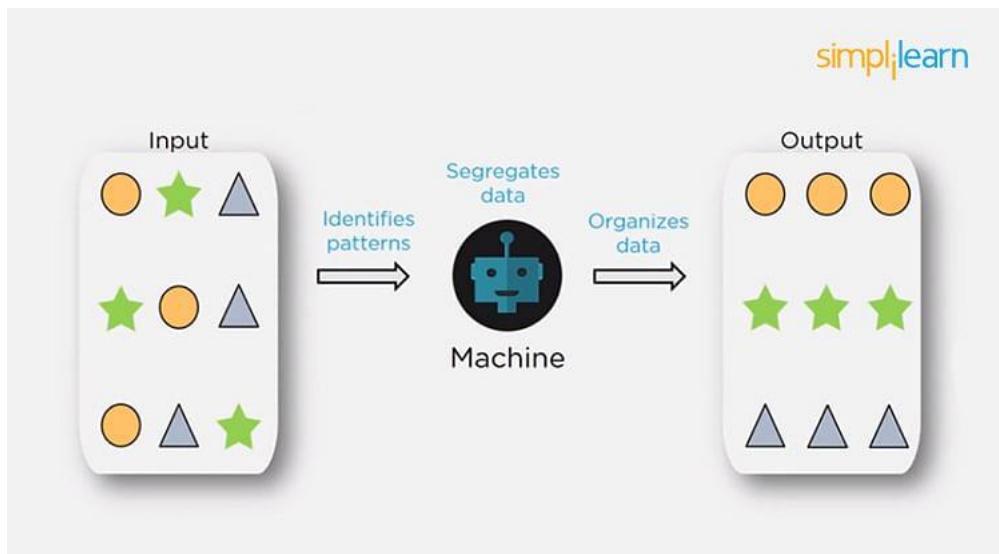
### Supervised Learning

In supervised machine learning, a model makes predictions or decisions based on past or labeled data. Labeled data refers to sets of data that are given tags or labels, and thus made more meaningful.



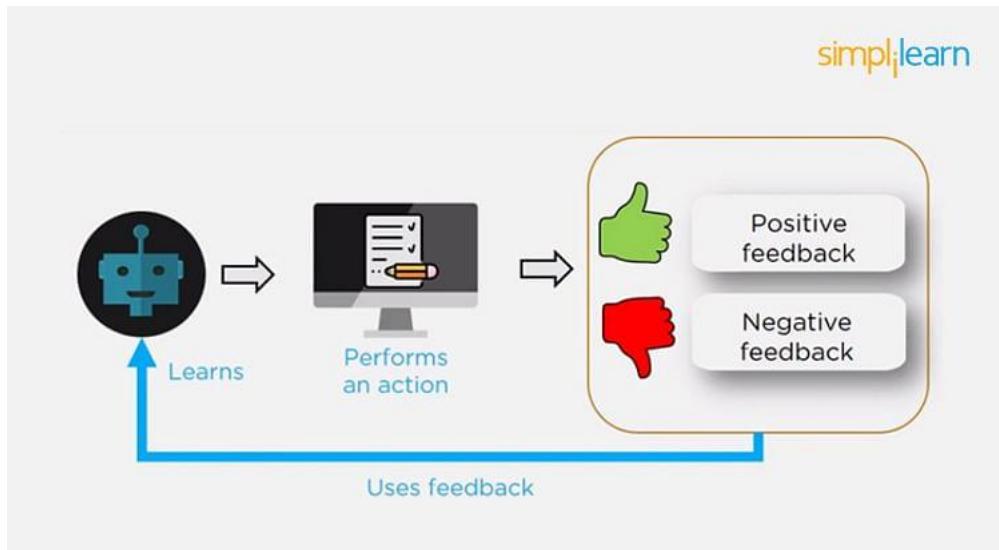
### Unsupervised Learning

In unsupervised learning, we don't have labeled data. A model can identify patterns, anomalies, and relationships in the input data.



## Reinforcement Learning

Using reinforcement learning, the model can learn based on the rewards it received for its previous action.



Consider an environment where an agent is working. The agent is given a target to achieve. Every time the agent takes some action toward the target, it is given positive feedback. And, if the action taken is going away from the goal, the agent is given negative feedback.

## 2. What is Overfitting, and How Can You Avoid It?

The Overfitting is a situation that occurs when a model learns the training set too well, taking up random fluctuations in the training data as concepts. These impact the model's ability to generalize and don't apply to new data.

When a model is given the training data, it shows 100 percent accuracy—technically a slight loss. But, when we use the test data, there may be an error and low efficiency. This condition is known as overfitting.

There are multiple ways of avoiding overfitting, such as:

- Regularization. It involves a cost term for the features involved with the objective function
- Making a simple model. With lesser variables and parameters, the variance can be reduced
- Cross-validation methods like k-folds can also be used
- If some model parameters are likely to cause overfitting, techniques for regularization like LASSO can be used that penalize these parameters

### 3. What is 'training Set' and 'test Set' in a Machine Learning Model? How Much Data Will You Allocate for Your Training, Validation, and Test Sets?

There is a three-step process followed to create a model:

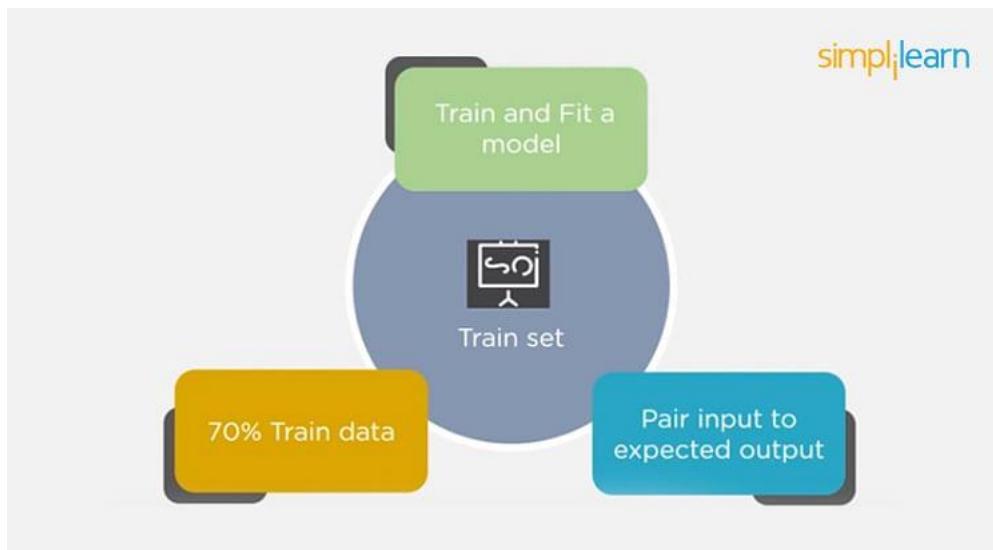
1. Train the model
2. Test the model
3. Deploy the model

Training Set	Test Set
<ul style="list-style-type: none"><li>• The training set is examples given to the model to analyze and learn</li><li>• 70% of the total data is typically taken as the training dataset</li><li>• This is labeled data used to train the model</li></ul>	<ul style="list-style-type: none"><li>• The test set is used to test the accuracy of the hypothesis generated by the model</li></ul>

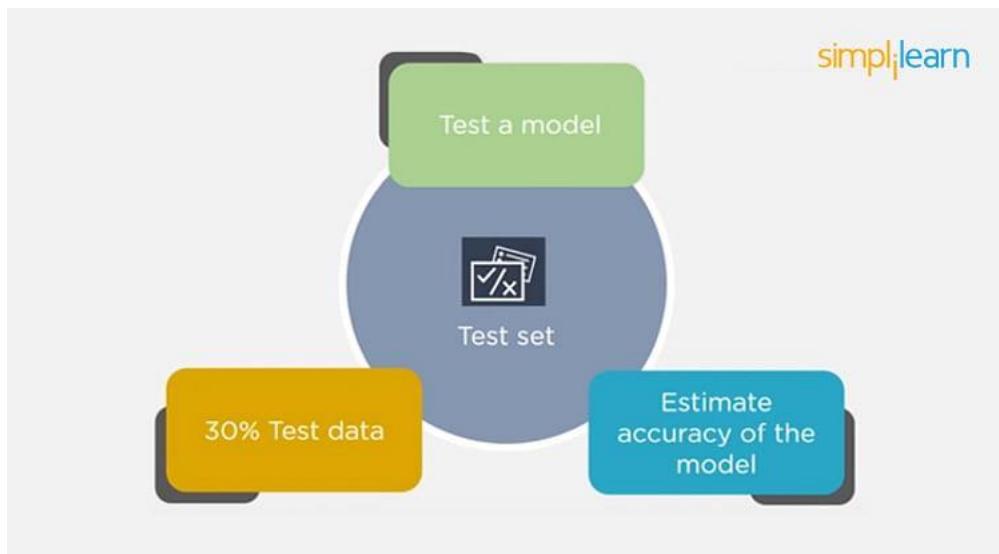
- Remaining 30% is taken as testing dataset
- We test without labeled data and then verify results with labels

Consider a case where you have labeled data for 1,000 records. One way to train the model is to expose all 1,000 records during the training process. Then you take a small set of the same data to test the model, which would give good results in this case.

But, this is not an accurate way of testing. So, we set aside a portion of that data called the ‘test set’ before starting the training process. The remaining data is called the ‘training set’ that we use for training the model. The training set passes through the model multiple times until the accuracy is high, and errors are minimized.



Now, we pass the test data to check if the model can accurately predict the values and determine if training is effective. If you get errors, you either need to change your model or retrain it with more data.



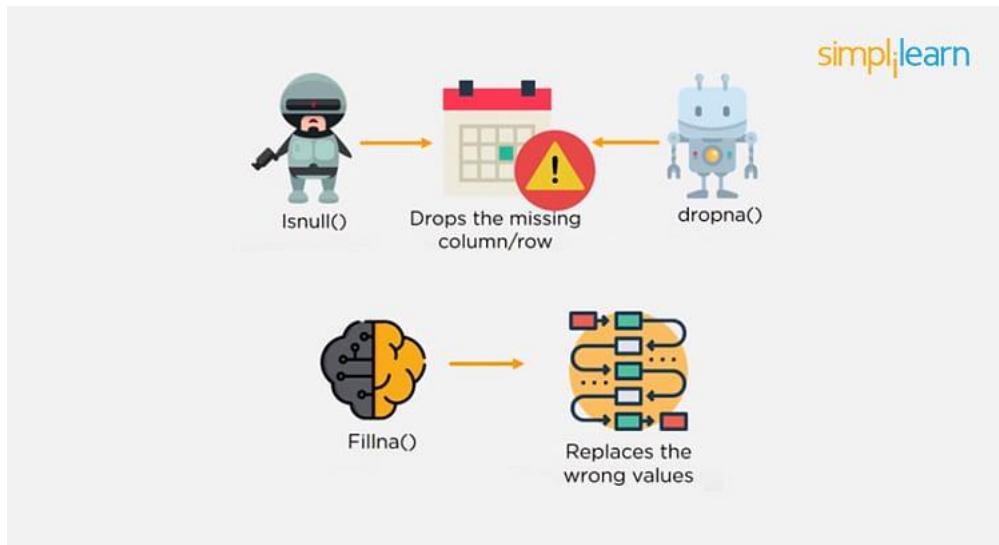
Regarding the question of how to split the data into a training set and test set, there is no fixed rule, and the ratio can vary based on individual preferences.

#### 4. How Do You Handle Missing or Corrupted Data in a Dataset?

One of the easiest ways to handle missing or corrupted data is to drop those rows or columns or replace them entirely with some other value.

There are two useful methods in Pandas:

- `Isnull()` and `dropna()` will help to find the columns/rows with missing data and drop them
- `Fillna()` will replace the wrong values with a placeholder value



## 5. How Can You Choose a Classifier Based on a Training Set Data Size?

When the training set is small, a model that has a right bias and low variance seems to work better because they are less likely to overfit.

For example, Naive Bayes works best when the training set is large. Models with low bias and high variance tend to perform better as they work fine with complex relationships.

## 6. Explain the Confusion Matrix with Respect to Machine Learning Algorithms.

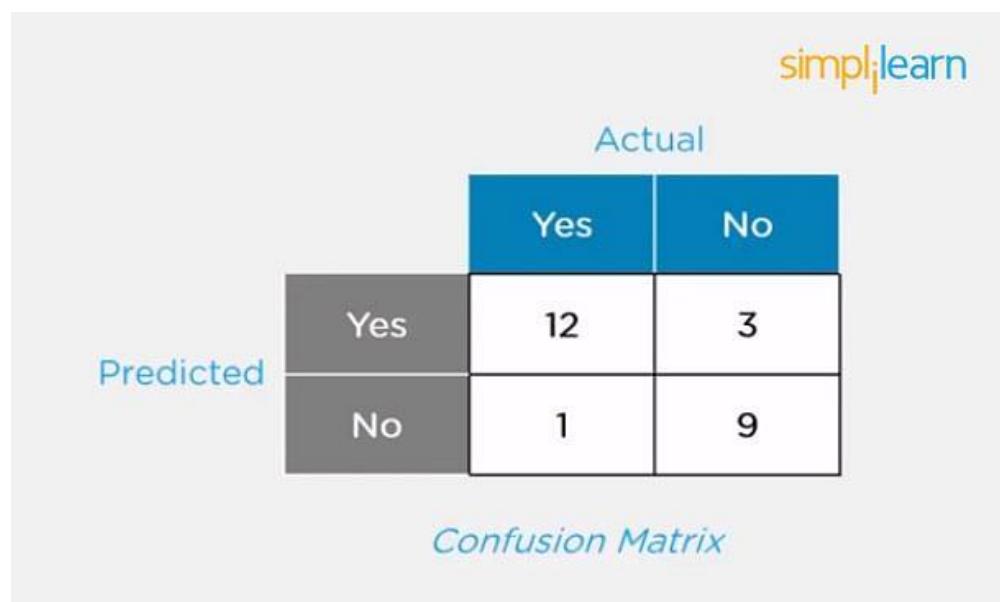
A confusion matrix (or error matrix) is a specific table that is used to measure the performance of an algorithm. It is mostly used in supervised learning; in unsupervised learning, it's called the matching matrix.

The confusion matrix has two parameters:

- Actual
- Predicted

It also has identical sets of features in both of these dimensions.

Consider a confusion matrix (binary matrix) shown below:



The image shows a 2x2 confusion matrix from simplilearn.com. The columns are labeled 'Yes' and 'No'. The rows are labeled 'Actual' (top) and 'Predicted' (left). The matrix values are: Actual Yes | Predicted Yes: 12, Predicted No: 3. Actual No | Predicted Yes: 1, Predicted No: 9. The 'simplilearn' logo is in the top right corner, and the caption 'Confusion Matrix' is at the bottom.

		Actual	
		Yes	No
Predicted	Yes	12	3
	No	1	9

Here,

For actual values:

$$\text{Total Yes} = 12+1 = 13$$

$$\text{Total No} = 3+9 = 12$$

Similarly, for predicted values:

$$\text{Total Yes} = 12+3 = 15$$

$$\text{Total No} = 1+9 = 10$$

For a model to be accurate, the values across the diagonals should be high. The total sum of all the values in the matrix equals the total observations in the test data set.

For the above matrix, total observations =  $12+3+1+9 = 25$

Now, accuracy = sum of the values across the diagonal/total dataset

$$= (12+9) / 25$$

$$= 21 / 25$$

$$= 84\%$$

## 7. What Is a False Positive and False Negative and How Are They Significant?

False positives are those cases that wrongly get classified as True but are False.

False negatives are those cases that wrongly get classified as False but are True.

In the term ‘False Positive,’ the word ‘Positive’ refers to the ‘Yes’ row of the predicted value in the confusion matrix. The complete term indicates that the system has predicted it as a positive, but the actual value is negative.

		Actual		
		Yes	No	
Predicted	Yes	12	3	False Positive
	No	1	9	False Negative
		Confusion Matrix		

So, looking at the confusion matrix, we get:

False-positive = 3

True positive = 12

Similarly, in the term ‘False Negative,’ the word ‘Negative’ refers to the ‘No’ row of the predicted value in the confusion matrix. And the complete term indicates that the system has predicted it as negative, but the actual value is positive.

So, looking at the confusion matrix, we get:

False Negative = 1

True Negative = 9

## 8. What Are the Three Stages of Building a Model in Machine Learning?

The three stages of building a machine learning model are:

- Model Building

Choose a suitable algorithm for the model and train it according to the requirement

- Model Testing

- Check the accuracy of the model through the test data
- Applying the Model

Make the required changes after testing and use the final model for real-time projects

Here, it's important to remember that once in a while, the model needs to be checked to make sure it's working correctly. It should be modified to make sure that it is up-to-date.

## 9. What is Deep Learning?

The Deep learning is a subset of machine learning that involves systems that think and learn like humans using artificial neural networks. The term ‘deep’ comes from the fact that you can have several layers of neural networks.

One of the primary differences between machine learning and deep learning is that feature engineering is done manually in machine learning. In the case of deep learning, the model consisting of neural networks will automatically determine which features to use (and which not to use).

This is a commonly asked question asked in both Machine Learning Interviews as well as Deep Learning Interview Questions

## 10. What Are the Differences Between Machine Learning and Deep Learning?

Machine Learning	Deep Learning
<ul style="list-style-type: none"><li>• Enables machines to take decisions on their own, based on past data</li><li>• It needs only a small amount of data for training</li><li>• Works well on the low-end system, so you don't need large machines</li></ul>	<ul style="list-style-type: none"><li>• Enables machines to take decisions with the help of artificial neural networks</li><li>• It needs a large amount of training data</li></ul>

- |  |   |
|--|---|
| <ul style="list-style-type: none"> <li>• Most features need to be identified in advance and manually coded</li> <li>• The problem is divided into two parts and solved individually and then combined</li> </ul> | <ul style="list-style-type: none"> <li>• Needs high-end machines because it requires a lot of computing power</li> <li>• The machine learns the features from the data it is provided</li> <li>• The problem is solved in an end-to-end manner</li> </ul> |
|--|---|

## 11. What Are the Applications of Supervised Machine Learning in Modern Businesses?

Applications of supervised machine learning include:

- Email Spam Detection

Here we train the model using historical data that consists of emails categorized as spam or not spam. This labeled information is fed as input to the model.

- Healthcare Diagnosis

By providing images regarding a disease, a model can be trained to detect if a person is suffering from the disease or not.

- Sentiment Analysis

This refers to the process of using algorithms to mine documents and determine whether they're positive, neutral, or negative in sentiment.

- Fraud Detection

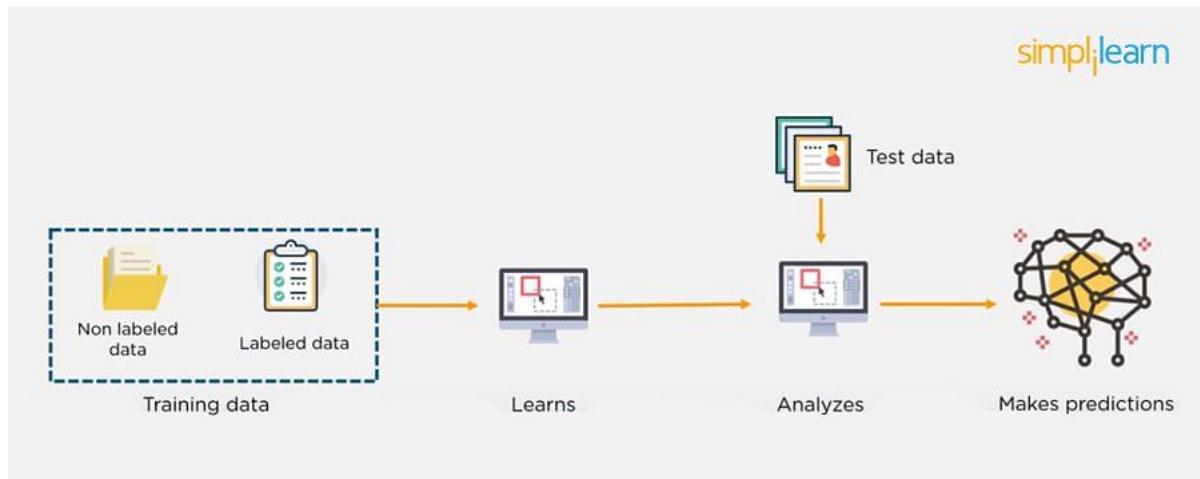
By training the model to identify suspicious patterns, we can detect instances of possible fraud.

Related Interview Questions and Answers  
AI | Data Science

## 12. What is Semi-supervised Machine Learning?

Supervised learning uses data that is completely labeled, whereas unsupervised learning uses no training data.

In the case of semi-supervised learning, the training data contains a small amount of labeled data and a large amount of unlabeled data.

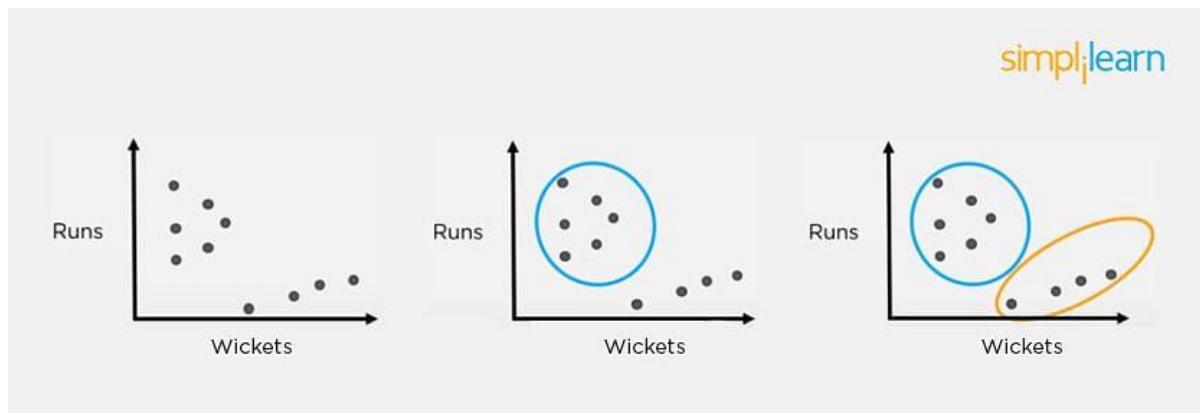


### 13. What Are Unsupervised Machine Learning Techniques?

There are two techniques used in unsupervised learning: clustering and association.

#### Clustering

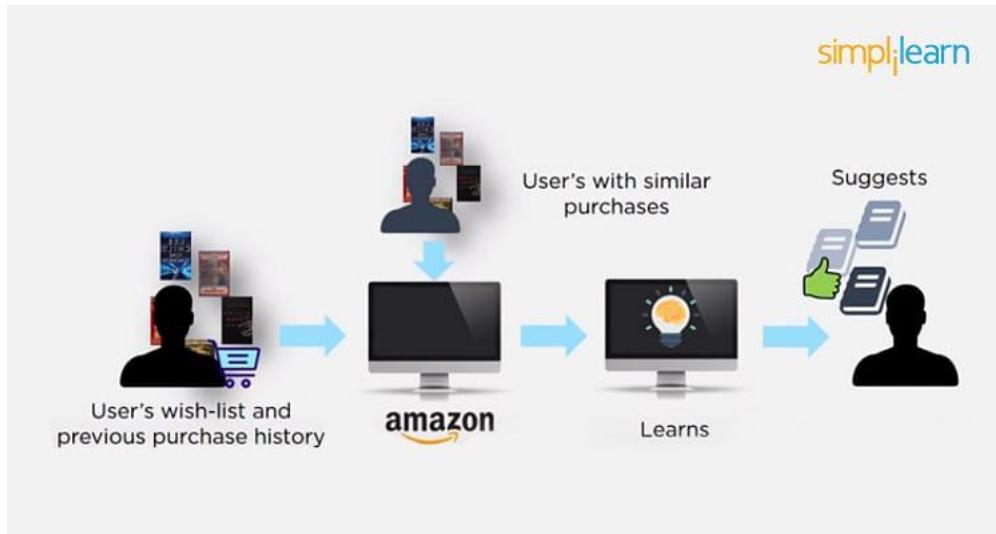
Clustering problems involve data to be divided into subsets. These subsets, also called clusters, contain data that are similar to each other. Different clusters reveal different details about the objects, unlike classification or regression.



#### Association

In an association problem, we identify patterns of associations between different variables or items.

For example, an e-commerce website can suggest other items for you to buy, based on the prior purchases that you have made, spending habits, items in your wishlist, other customers' purchase habits, and so on.



#### 14. What is the Difference Between Supervised and Unsupervised Machine Learning?

- Supervised learning - This model learns from the labeled data and makes a future prediction as output
- Unsupervised learning - This model uses unlabeled input data and allows the algorithm to act on that information without guidance.

#### 15. What is the Difference Between Inductive Machine Learning and Deductive Machine Learning?

Inductive Learning	Deductive Learning
<ul style="list-style-type: none"><li>• It observes instances based on defined principles to draw a conclusion</li></ul>	<ul style="list-style-type: none"><li>• It concludes experiences</li><li>• Example: Allow the child to play with</li></ul>

- Example: Explaining to a child to keep away from the fire by showing a video where fire causes damage

fire. If he or she gets burned, they will learn that it is dangerous and will refrain from making the same mistake again

## 16. Compare K-means and KNN Algorithms.

K-means	KNN
<ul style="list-style-type: none"> <li>• K-Means is unsupervised</li> <li>• K-Means is a clustering algorithm</li> <li>• The points in each cluster are similar to each other, and each cluster is different from its neighboring clusters</li> </ul>	<ul style="list-style-type: none"> <li>• KNN is supervised in nature</li> <li>• KNN is a classification algorithm</li> <li>• It classifies an unlabeled observation based on its K (can be any number) surrounding neighbors</li> </ul>

## 17. What Is ‘naive’ in the Naive Bayes Classifier?

The classifier is called ‘naive’ because it makes assumptions that may or may not turn out to be correct.

The algorithm assumes that the presence of one feature of a class is not related to the presence of any other feature (absolute independence of features), given the class variable.

For instance, a fruit may be considered to be a cherry if it is red in color and round in shape, regardless of other features. This assumption may or may not be right (as an apple also matches the description).

## 18. Explain How a System Can Play a Game of Chess Using Reinforcement Learning.

Reinforcement learning has an environment and an agent. The agent performs some actions to achieve a specific goal. Every time the agent performs a task that is taking it towards the goal, it is rewarded. And, every time it takes a step that goes against that goal or in the reverse direction, it is penalized.

Earlier, chess programs had to determine the best moves after much research on numerous factors. Building a machine designed to play such games would require many rules to be specified.

With reinforced learning, we don't have to deal with this problem as the learning agent learns by playing the game. It will make a move (decision), check if it's the right move (feedback), and keep the outcomes in memory for the next step it takes (learning). There is a reward for every correct decision the system takes and punishment for the wrong one.

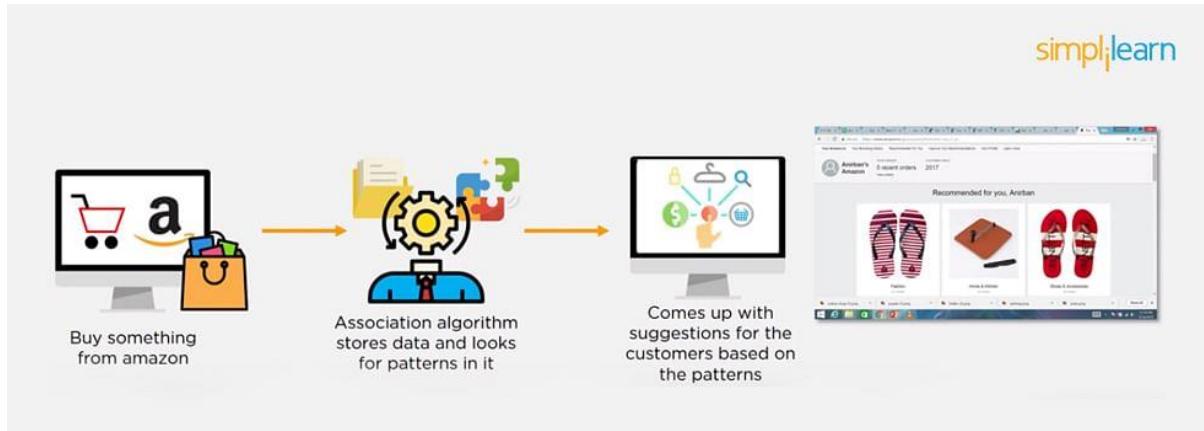
## 19. How Will You Know Which Machine Learning Algorithm to Choose for Your Classification Problem?

While there is no fixed rule to choose an algorithm for a classification problem, you can follow these guidelines:

- If accuracy is a concern, test different algorithms and cross-validate them
- If the training dataset is small, use models that have low variance and high bias
- If the training dataset is large, use models that have high variance and little bias

## 20. How is Amazon Able to Recommend Other Things to Buy? How Does the Recommendation Engine Work?

Once a user buys something from Amazon, Amazon stores that purchase data for future reference and finds products that are most likely also to be bought, it is possible because of the Association algorithm, which can identify patterns in a given dataset.



## 21. When Will You Use Classification over Regression?

Classification is used when your target is categorical, while regression is used when your target variable is continuous. Both classification and regression belong to the category of supervised machine learning algorithms.

Examples of classification problems include:

- Predicting yes or no
- Estimating gender
- Breed of an animal
- Type of color

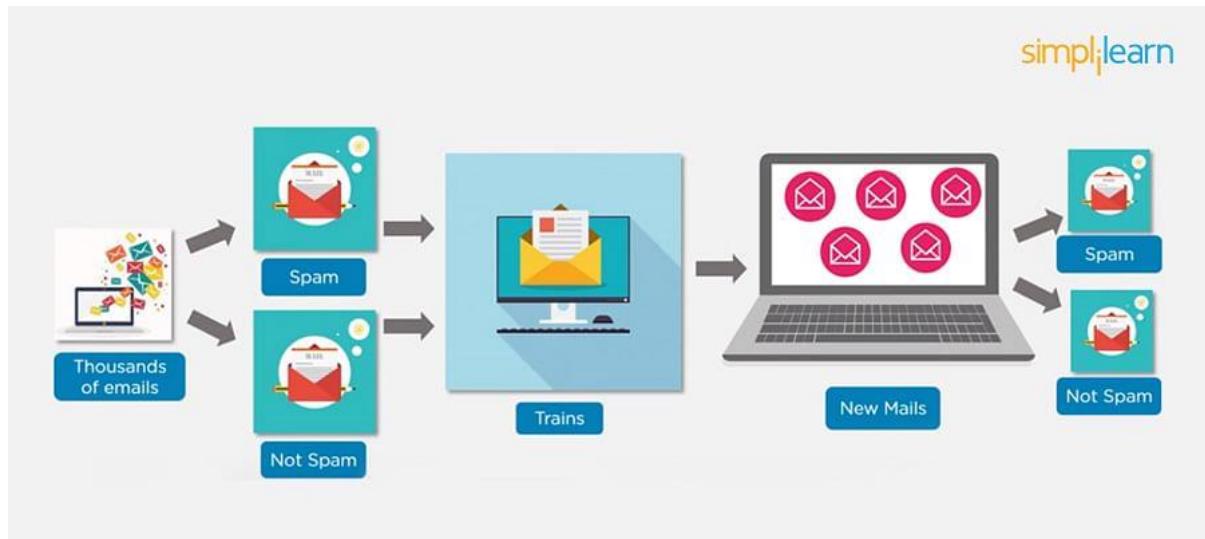
Examples of regression problems include:

- Estimating sales and price of a product
- Predicting the score of a team
- Predicting the amount of rainfall

## 22. How Do You Design an Email Spam Filter?

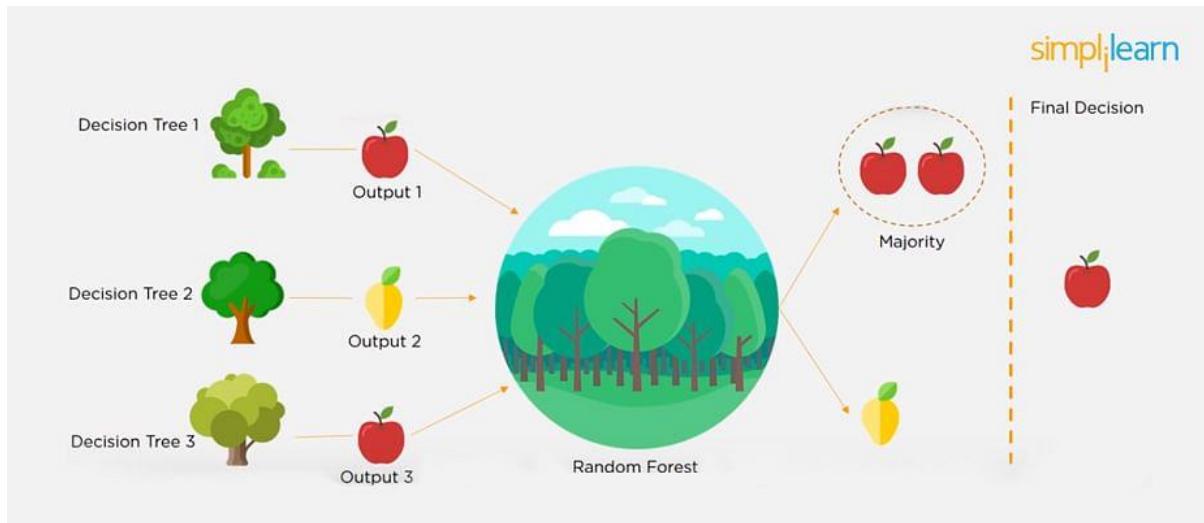
Building a spam filter involves the following process:

- The email spam filter will be fed with thousands of emails
- Each of these emails already has a label: ‘spam’ or ‘not spam.’
- The supervised machine learning algorithm will then determine which type of emails are being marked as spam based on spam words like the lottery, free offer, no money, full refund, etc.
- The next time an email is about to hit your inbox, the spam filter will use statistical analysis and algorithms like Decision Trees and SVM to determine how likely the email is spam
- If the likelihood is high, it will label it as spam, and the email won’t hit your inbox
- Based on the accuracy of each model, we will use the algorithm with the highest accuracy after testing all the models



### 23. What is a Random Forest?

A ‘random forest’ is a supervised machine learning algorithm that is generally used for classification problems. It operates by constructing multiple decision trees during the training phase. The random forest chooses the decision of the majority of the trees as the final decision.

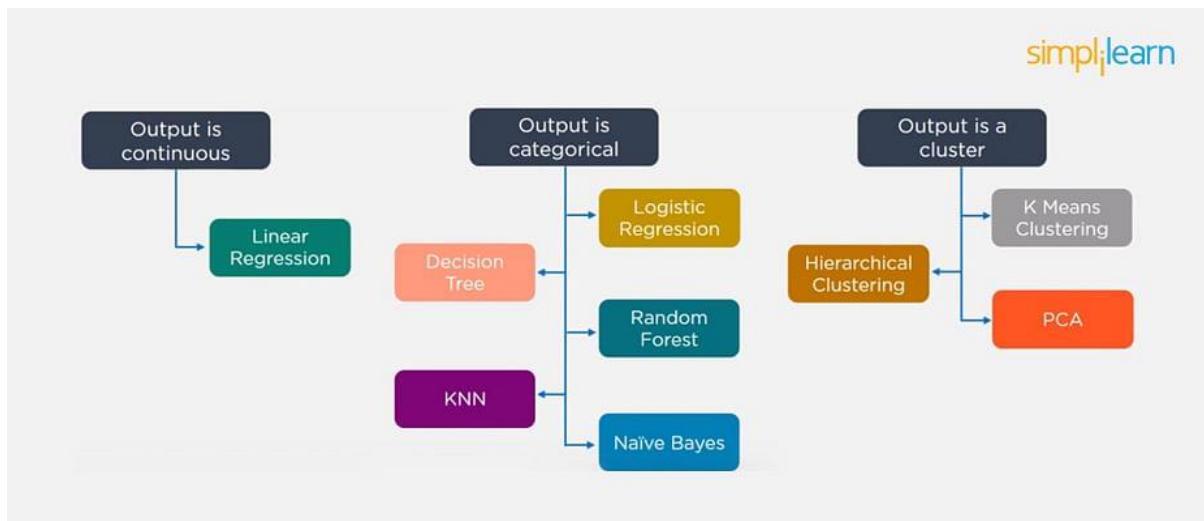


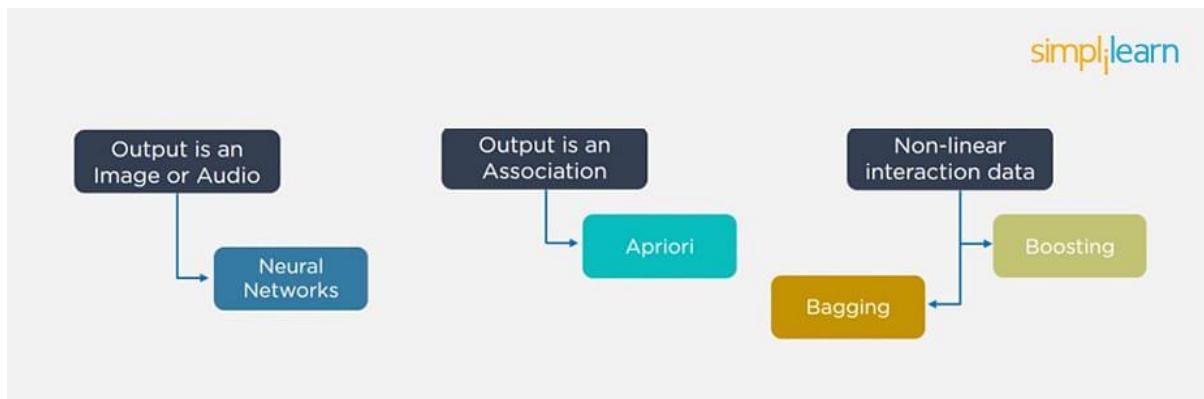
## 24. Considering a Long List of Machine Learning Algorithms, given a Data Set, How Do You Decide Which One to Use?

There is no master algorithm for all situations. Choosing an algorithm depends on the following questions:

- How much data do you have, and is it continuous or categorical?
- Is the problem related to classification, association, clustering, or regression?
- Predefined variables (labeled), unlabeled, or mix?
- What is the goal?

Based on the above questions, the following algorithms can be used:





## 25. What is Bias and Variance in a Machine Learning Model?

### Bias

Bias in a machine learning model occurs when the predicted values are further from the actual values. Low bias indicates a model where the prediction values are very close to the actual ones.

**Underfitting:** High bias can cause an algorithm to miss the relevant relations between features and target outputs.

### Variance

Variance refers to the amount the target model will change when trained with different training data. For a good model, the variance should be minimized.

**Overfitting:** High variance can cause an algorithm to model the random noise in the training data rather than the intended outputs.

## 26. What is the Trade-off Between Bias and Variance?

The bias-variance decomposition essentially decomposes the learning error from any algorithm by adding the bias, variance, and a bit of irreducible error due to noise in the underlying dataset.

Necessarily, if you make the model more complex and add more variables, you'll lose bias but gain variance. To get the optimally-reduced amount of error, you'll have to trade off bias and variance. Neither high bias nor high variance is desired.

High bias and low variance algorithms train models that are consistent, but inaccurate on average.

High variance and low bias algorithms train models that are accurate but inconsistent.

## 27. Define Precision and Recall.

### Precision

Precision is the ratio of several events you can correctly recall to the total number of events you recall (mix of correct and wrong recalls).

$$\text{Precision} = (\text{True Positive}) / (\text{True Positive} + \text{False Positive})$$

### Recall

A recall is the ratio of the number of events you can recall to the number of total events.

$$\text{Recall} = (\text{True Positive}) / (\text{True Positive} + \text{False Negative})$$

## 28. What is a Decision Tree Classification?

A decision tree builds classification (or regression) models as a tree structure, with datasets broken up into ever-smaller subsets while developing the decision tree, literally in a tree-like way with branches and nodes. Decision trees can handle both categorical and numerical data.

## 29. What is Pruning in Decision Trees, and How Is It Done?

Pruning is a technique in machine learning that reduces the size of decision trees. It reduces the complexity of the final classifier, and hence improves predictive accuracy by the reduction of overfitting.

Pruning can occur in:

- Top-down fashion. It will traverse nodes and trim subtrees starting at the root
- Bottom-up fashion. It will begin at the leaf nodes

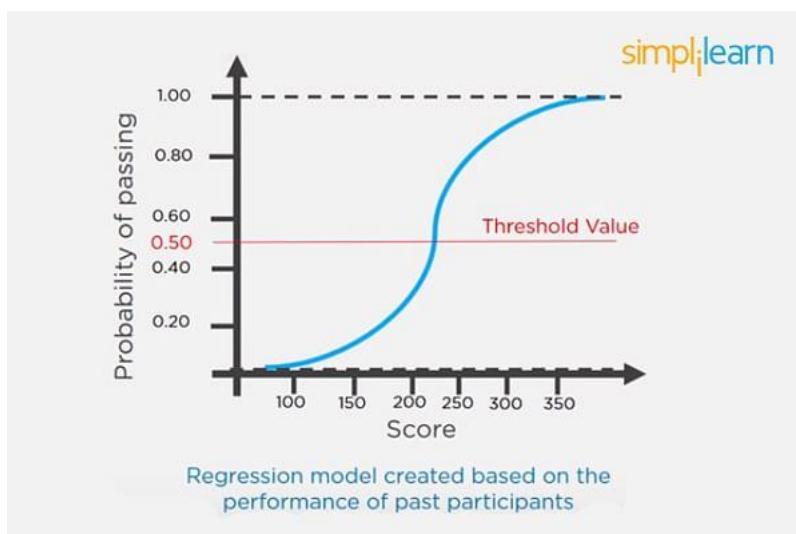
There is a popular pruning algorithm called reduced error pruning, in which:

- Starting at the leaves, each node is replaced with its most popular class
- If the prediction accuracy is not affected, the change is kept
- There is an advantage of simplicity and speed

### 30. Briefly Explain Logistic Regression.

Logistic regression is a classification algorithm used to predict a binary outcome for a given set of independent variables.

The output of logistic regression is either a 0 or 1 with a threshold value of generally 0.5. Any value above 0.5 is considered as 1, and any point below 0.5 is considered as 0.



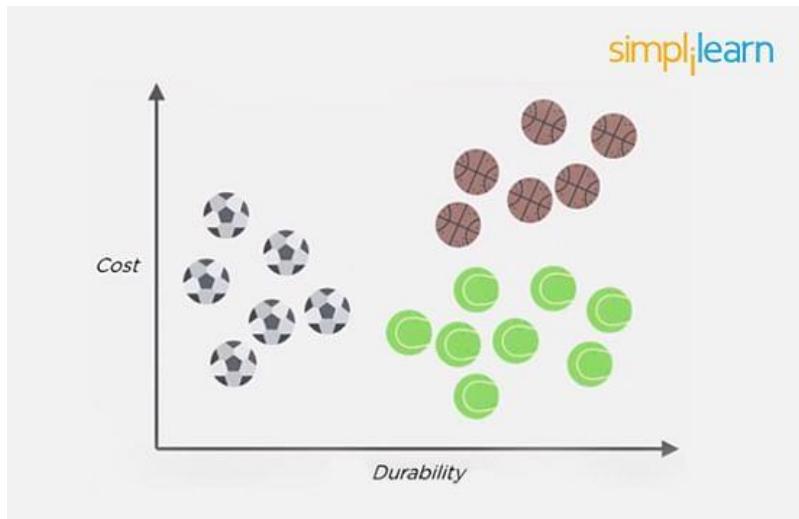
### 31. Explain the K Nearest Neighbor Algorithm.

K nearest neighbor algorithm is a classification algorithm that works in a way that a new data point is assigned to a neighboring group to which it is most similar.

In K nearest neighbors, K can be an integer greater than 1. So, for every new data point, we want to classify, we compute to which neighboring group it is closest.

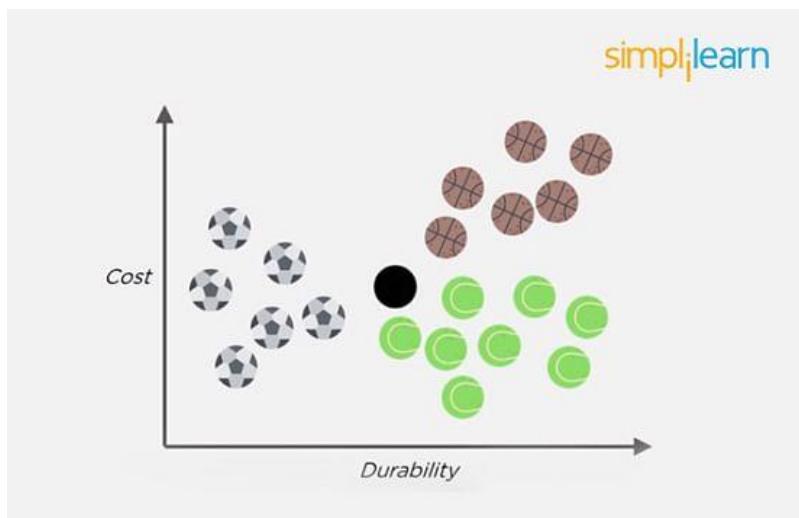
Let us classify an object using the following example. Consider there are three clusters:

- Football
- Basketball
- Tennis ball



Let the new data point to be classified is a black ball. We use KNN to classify it. Assume  $K = 5$  (initially).

Next, we find the  $K$  (five) nearest data points, as shown.



Observe that all five selected points do not belong to the same cluster. There are three tennis balls and one each of basketball and football.

When multiple classes are involved, we prefer the majority. Here the majority is with the tennis ball, so the new data point is assigned to this cluster.

### 32. What is a Recommendation System?

Anyone who has used Spotify or shopped at Amazon will recognize a recommendation system: It's an information filtering system that predicts what a user might want to hear or see based on choice patterns provided by the user.

### 33. What is Kernel SVM?

Kernel SVM is the abbreviated version of the kernel support vector machine. Kernel methods are a class of algorithms for pattern analysis, and the most common one is the kernel SVM.

### 34. What Are Some Methods of Reducing Dimensionality?

You can reduce dimensionality by combining features with feature engineering, removing collinear features, or using algorithmic dimensionality reduction.

Now that you have gone through these machine learning interview questions, you must have got an idea of your strengths and weaknesses in this domain.

### 35. What is Principal Component Analysis?

Principal Component Analysis or PCA is a multivariate statistical technique that is used for analyzing quantitative data. The objective of PCA is to reduce higher dimensional data to lower dimensions, remove noise, and extract crucial information such as features and attributes from large amounts of data.

### 36. What do you understand by the F1 score?

The F1 score is a metric that combines both Precision and Recall. It is also the weighted average of precision and recall.

The F1 score can be calculated using the below formula:

$$F1 = 2 * (P * R) / (P + R)$$

The F1 score is one when both Precision and Recall scores are one.

37. What do you understand by Type I vs Type II error?

Type I Error: Type I error occurs when the null hypothesis is true and we reject it.

Type II Error: Type II error occurs when the null hypothesis is false and we accept it.

		reality	
		$H_0 = \text{True}$	$H_0 = \text{False}$
Conclusion	$H_0$ is not rejected	OK	Type II error
	$H_0$ is rejected	Type I error	OK

38. Explain Correlation and Covariance?

Correlation: Correlation tells us how strongly two random variables are related to each other. It takes values between -1 to +1.

Formula to calculate Correlation:

$$\text{Correlation} = \frac{\text{Cov}(x, y)}{s_x * s_y}$$

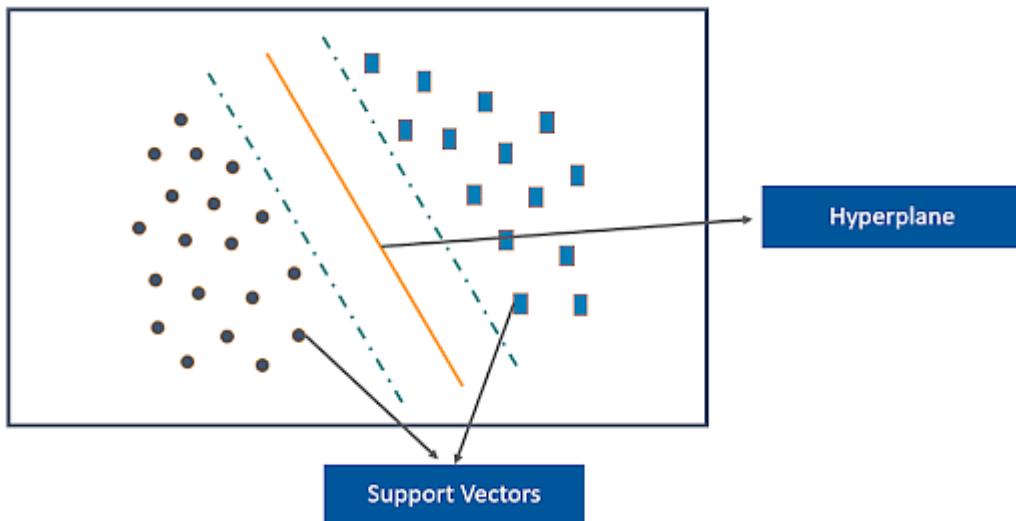
Covariance: Covariance tells us the direction of the linear relationship between two random variables. It can take any value between  $-\infty$  and  $+\infty$ .

Formula to calculate Covariance:

$$\text{Cov}(x, y) = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{N}$$

### 39. What are Support Vectors in SVM?

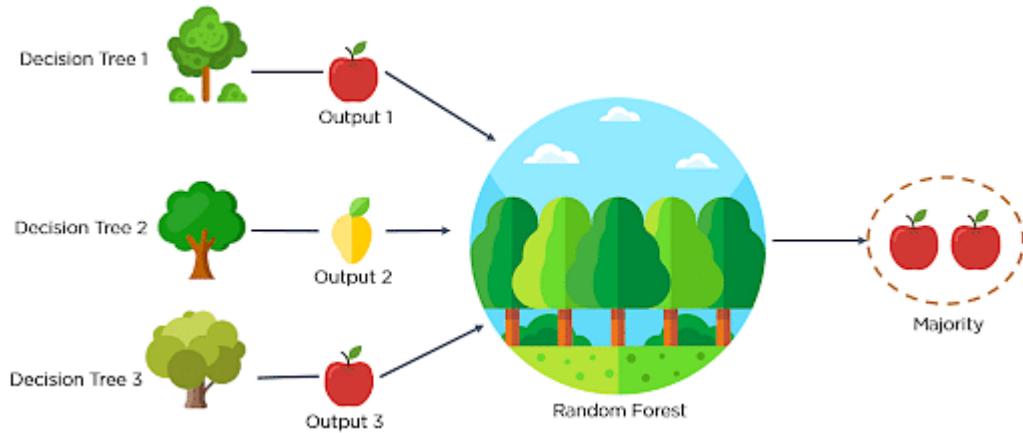
Support Vectors are data points that are nearest to the hyperplane. It influences the position and orientation of the hyperplane. Removing the support vectors will alter the position of the hyperplane. The support vectors help us build our support vector machine model.



### 40. What is Ensemble learning?

Ensemble learning is a combination of the results obtained from multiple machine learning models to increase the accuracy for improved decision-making.

Example: A Random Forest with 100 trees can provide much better results than using just one decision tree.



#### 41. What is Cross-Validation?

Cross-Validation in Machine Learning is a statistical resampling technique that uses different parts of the dataset to train and test a machine learning algorithm on different iterations. The aim of cross-validation is to test the model's ability to predict a new set of data that was not used to train the model. Cross-validation avoids the overfitting of data.

K-Fold Cross Validation is the most popular resampling technique that divides the whole dataset into K sets of equal sizes.

#### 42. What are the different methods to split a tree in a decision tree algorithm?

Variance: Splitting the nodes of a decision tree using the variance is done when the target variable is continuous.

$$\text{Variance} = \frac{\sum (x - \bar{x})^2}{N}$$

Information Gain: Splitting the nodes of a decision tree using Information Gain is preferred when the target variable is categorical.

**IG = 1 - Entropy**

**Entropy = -  $\sum p_i \log_2 p_i$**

Gini Impurity: Splitting the nodes of a decision tree using Gini Impurity is followed when the target variable is categorical.

$$I_G(n) = 1 - \sum_{i=1}^n (p_i)^2$$

43. How does the Support Vector Machine algorithm handle self-learning?

The SVM algorithm has a learning rate and expansion rate which takes care of self-learning. The learning rate compensates or penalizes the hyperplanes for making all the incorrect moves while the expansion rate handles finding the maximum separation area between different classes.

44. What are the assumptions you need to take before starting with linear regression?

There are primarily 5 assumptions for a Linear Regression model:

- Multivariate normality
- No auto-correlation
- Homoscedasticity
- Linear relationship
- No or little multicollinearity

45. What is the difference between Lasso and Ridge regression?

Lasso(also known as L1) and Ridge(also known as L2) regression are two popular regularization techniques that are used to avoid overfitting of data. These methods are used to penalize the coefficients to find the optimum solution and reduce complexity. The Lasso regression works by penalizing the sum of the absolute values of the coefficients. In Ridge or L2 regression, the penalty function is determined by the sum of the squares of the coefficients.

## 1. Why was Machine Learning Introduced?

The simplest answer is to make our lives easier. In the early days of “intelligent” applications, many systems used hardcoded rules of “if” and “else” decisions to process data or adjust the user input. Think of a spam filter whose job is to move the appropriate incoming email messages to a spam folder.

But with the machine learning algorithms, we are given ample information for the data to learn and identify the patterns from the data.

Unlike the normal problems we don’t need to write the new rules for each problem in machine learning, we just need to use the same workflow but with a different dataset.

Let’s talk about Alan Turing, in his 1950 paper, “Computing Machinery and Intelligence”, Alan asked, “Can machines think?”

Full paper [here](#)

The paper describes the “Imitation Game”, which includes three participants -

- Human acting as a judge,
- Another human, and
- A computer is an attempt to convince the judge that it is human.

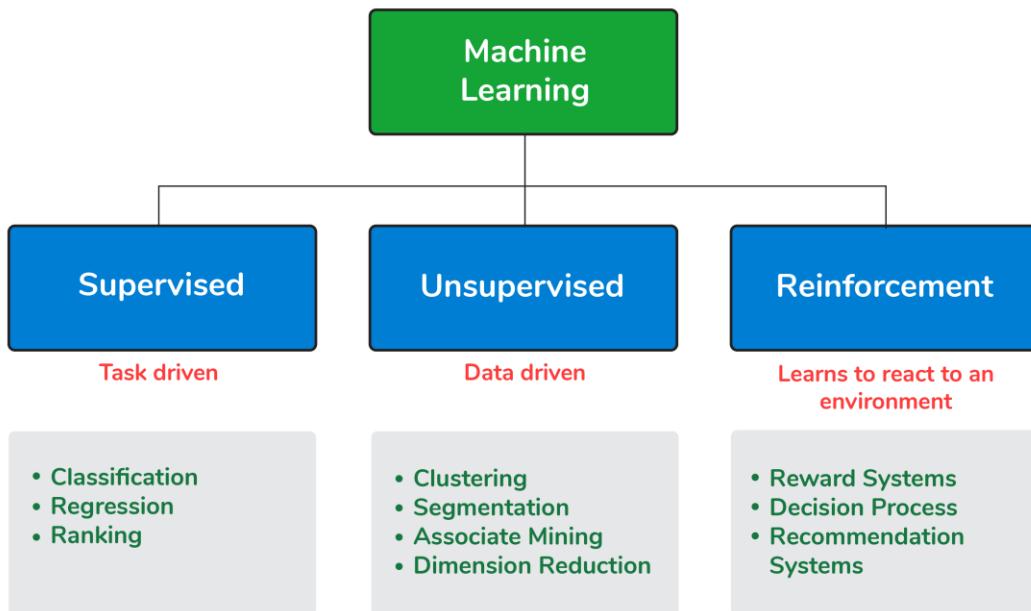
The judge asks the other two participants to talk. While they respond the judge needs to decide which response came from the computer. If the judge could not tell the difference the computer won the game.

The test continues today as an annual competition in artificial intelligence. The aim is simple enough: convince the judge that they are chatting to a human instead of a computer chatbot program.

## 2. What are Different Types of Machine Learning algorithms?

There are various types of machine learning algorithms. Here is the list of them in a broad category based on:

- Whether they are trained with human supervision (Supervised, unsupervised, reinforcement learning)
- The criteria in the below diagram are not exclusive, we can combine them any way we like.



## Types of Machine Learning algorithms

### 3. What is Supervised Learning?

Supervised learning is a machine learning algorithm of inferring a function from labeled training data. The training data consists of a set of training examples.

#### Example: 01

Knowing the height and weight identifying the gender of the person. Below are the popular supervised learning algorithms.

- Support Vector Machines
- Regression
- Naive Bayes
- Decision Trees
- K-nearest Neighbour Algorithm and Neural Networks.

## **Example: 02**

If you build a T-shirt classifier, the labels will be “this is an S, this is an M and this is L”, based on showing the classifier examples of S, M, and L.

## **4. What is Unsupervised Learning?**

Unsupervised learning is also a type of machine learning algorithm used to find patterns on the set of data given. In this, we don't have any dependent variable or label to predict. Unsupervised Learning Algorithms:

- Clustering,
- Anomaly Detection,
- Neural Networks and Latent Variable Models.

## **Example:**

In the same example, a T-shirt clustering will categorize as “collar style and V neck style”, “crew neck style” and “sleeve types”.

## **5. What is ‘Naive’ in a Naive Bayes?**

The Naive Bayes method is a supervised learning algorithm, it is naive since it makes assumptions by applying Bayes' theorem that all attributes are independent of each other.

Bayes' theorem states the following relationship, given class variable y and dependent vector  $x_1$  through  $x_n$ :

$$P(y_i | x_1, \dots, x_n) = P(y_i)P(x_1, \dots, x_n | y_i)(P(x_1, \dots, x_n))$$

Using the naive conditional independence assumption that each  $x_i$  is independent: for all  $i$  this relationship is simplified to:

$$P(x_i | y_i, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n) = P(x_i | y_i)$$

Since,  $P(x_1, \dots, x_n)$  is a constant given the input, we can use the following classification rule:

$P(y_i | x_1, \dots, x_n) = P(y) \prod_{i=1}^n P(x_i | y_i) P(x_1, \dots, x_n)$  and we can also use Maximum A Posteriori (MAP) estimation to estimate  $P(y_i)$  and  $P(y_i | x_i)$  the former is then the relative frequency of class  $y$  in the training set.

$$P(y_i | x_1, \dots, x_n) = P(y) \prod_{i=1}^n P(x_i | y_i)$$

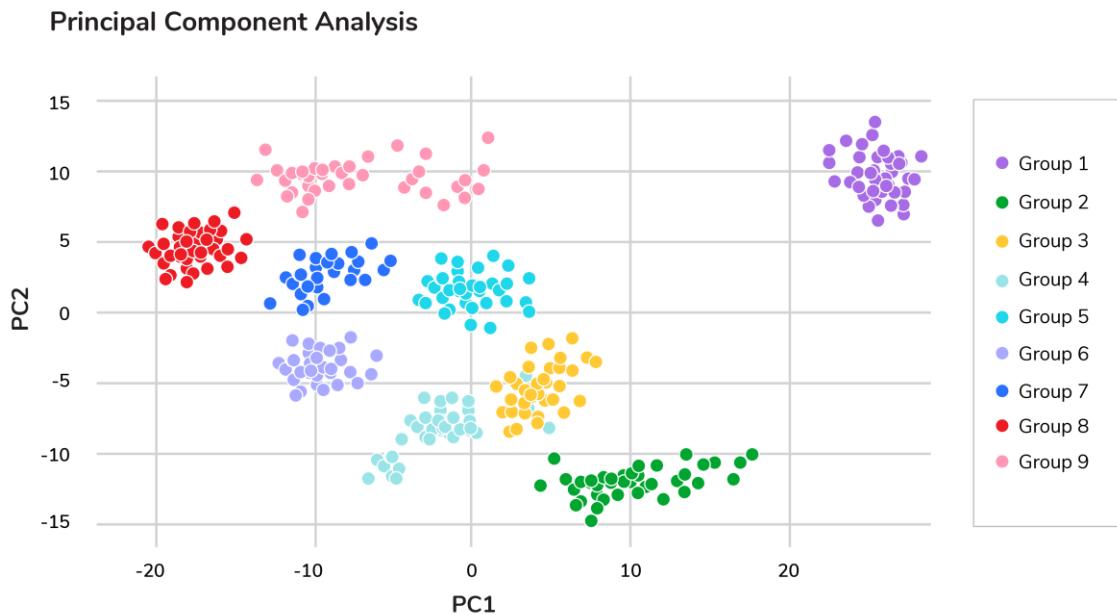
$$y = \arg \max P(y) \prod_{i=1}^n P(x_i | y_i)$$

The different naive Bayes classifiers mainly differ by the assumptions they make regarding the distribution of  $P(y_i | x_i)$ : can be Bernoulli, binomial, Gaussian, and so on.

## 6. What is PCA? When do you use it?

Principal component analysis (PCA) is most commonly used for dimension reduction.

In this case, PCA measures the variation in each variable (or column in the table). If there is little variation, it throws the variable out, as illustrated in the figure below:



Principal component analysis (PCA)

Thus making the dataset easier to visualize. PCA is used in finance, neuroscience, and pharmacology.

It is very useful as a preprocessing step, especially when there are linear correlations between features.

## 7. Explain SVM Algorithm in Detail

A Support Vector Machine (SVM) is a very powerful and versatile supervised machine learning model, capable of performing linear or non-linear classification, regression, and even outlier detection.

Suppose we have given some data points that each belong to one of two classes, and the goal is to separate two classes based on a set of examples.

In SVM, a data point is viewed as a p-dimensional vector (a list of p numbers), and we wanted to know whether we can separate such points with a (p-1)-dimensional hyperplane. This is called a linear classifier.

There are many hyperplanes that classify the data. To choose the best hyperplane that represents the largest separation or margin between the two classes.

If such a hyperplane exists, it is known as a maximum-margin hyperplane and the linear classifier it defines is known as a maximum margin classifier. The best hyperplane that divides the data in H3

We have data  $(x_1, y_1), \dots, (x_n, y_n)$ , and different features  $(x_{i1}, \dots, x_{ip})$ , and  $y_i$  is either 1 or -1.

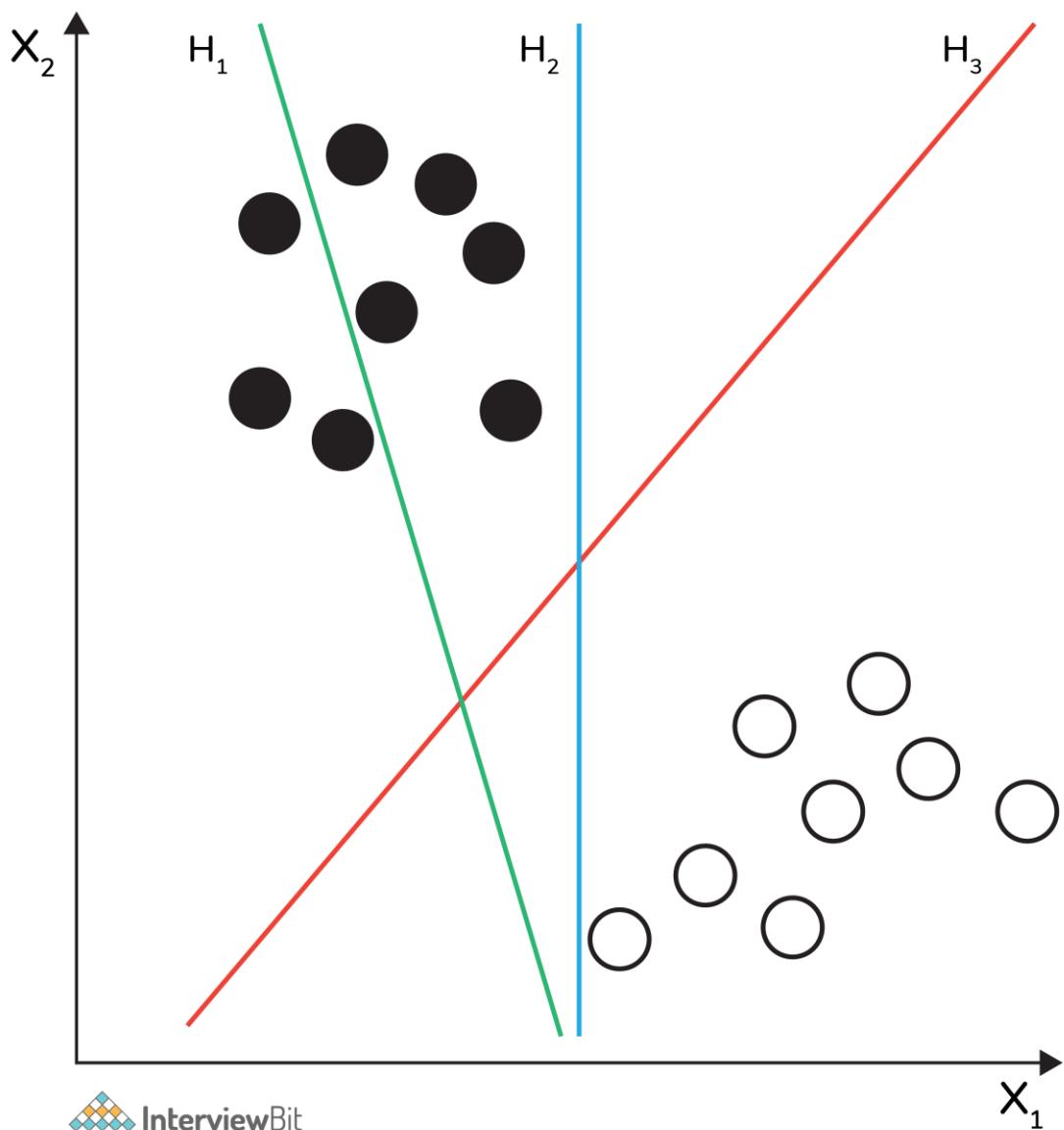
The equation of the hyperplane H3 is the set of points satisfying:

$$w \cdot x - b = 0$$

Where  $w$  is the normal vector of the hyperplane. The parameter  $b/\|w\|$  determines the offset of the hyperplane from the origin along the normal vector  $w$

So for each  $i$ , either  $x_i$  is in the hyperplane of 1 or -1. Basically,  $x_i$  satisfies:

$$w \cdot x_i - b = 1 \text{ or } w \cdot x_i - b = -1$$

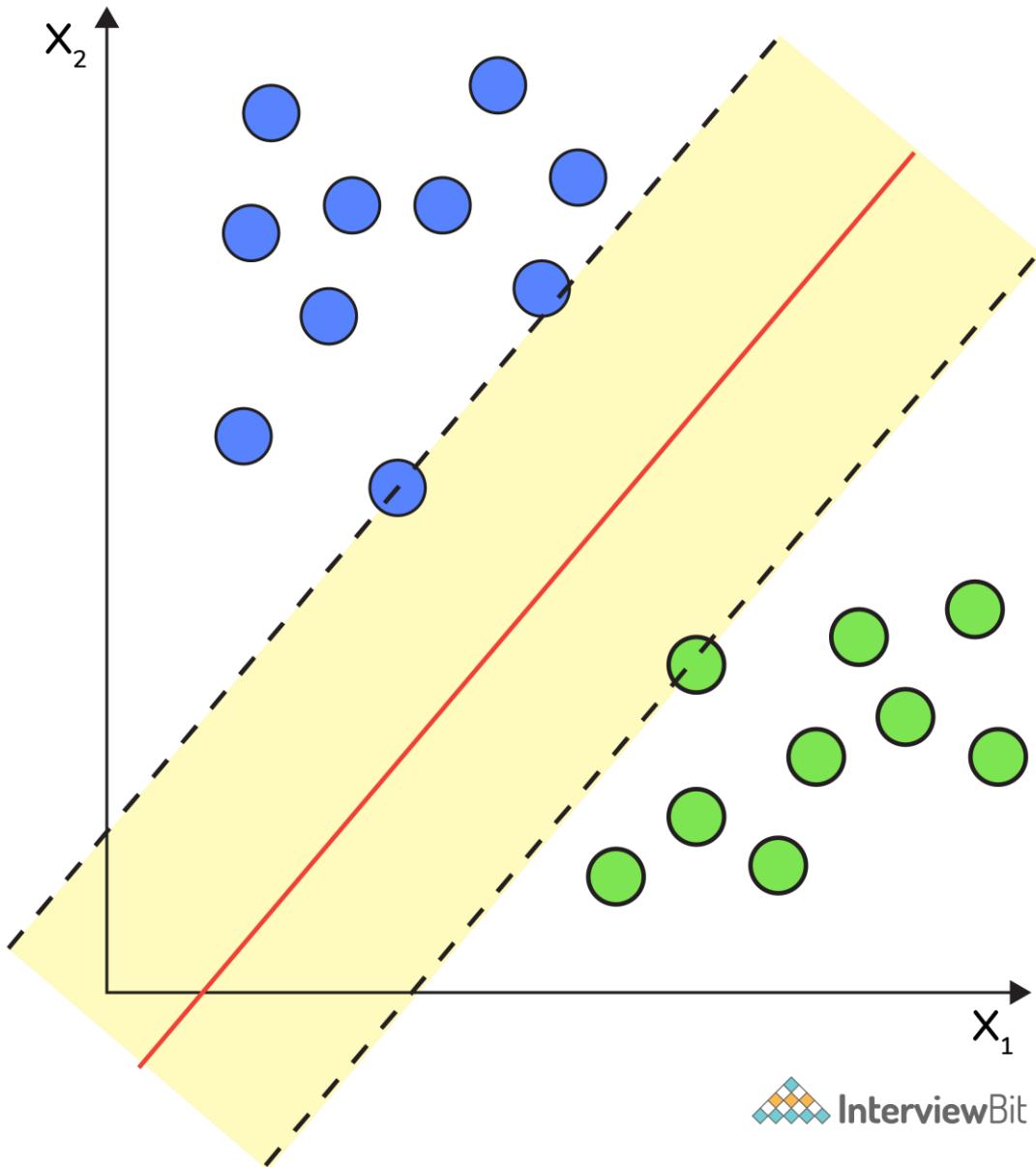


## Support Vector Machine (SVM)

### 8. What are Support Vectors in SVM?

A Support Vector Machine (SVM) is an algorithm that tries to fit a line (or plane or hyperplane) between the different classes that maximizes the distance from the line to the points of the classes.

In this way, it tries to find a robust separation between the classes. The Support Vectors are the points of the edge of the dividing hyperplane as in the below figure.



Support Vector Machine (SVM)

## 9. What are Different Kernels in SVM?

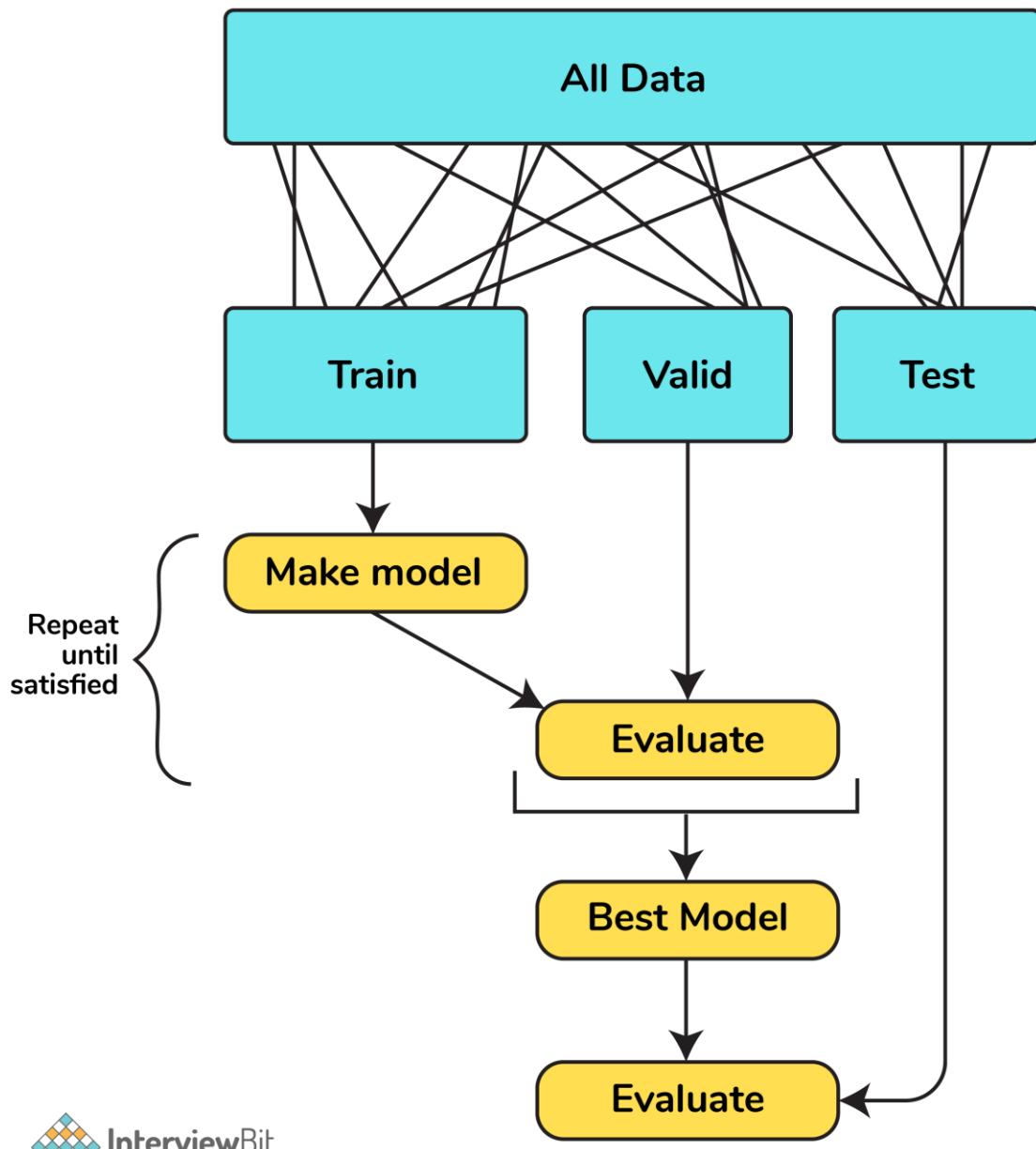
There are six types of kernels in SVM:

- Linear kernel - used when data is linearly separable.
- Polynomial kernel - When you have discrete data that has no natural notion of smoothness.
- Radial basis kernel - Create a decision boundary able to do a much better job of separating two classes than the linear kernel.
- Sigmoid kernel - used as an activation function for neural networks.

## 10. What is Cross-Validation?

Cross-validation is a method of splitting all your data into three parts: training, testing, and validation data. Data is split into  $k$  subsets, and the model has trained on  $k-1$  of those datasets.

The last subset is held for testing. This is done for each of the subsets. This is  $k$ -fold cross-validation. Finally, the scores from all the  $k$ -folds are averaged to produce the final score.



Cross-validation

## **11. What is Bias in Machine Learning?**

Bias in data tells us there is inconsistency in data. The inconsistency may occur for several reasons which are not mutually exclusive.

For example, a tech giant like Amazon to speed the hiring process they build one engine where they are going to give 100 resumes, it will spit out the top five, and hire those.

When the company realized the software was not producing gender-neutral results it was tweaked to remove this bias.

## **12. Explain the Difference Between Classification and Regression?**

Classification is used to produce discrete results, classification is used to classify data into some specific categories.

For example, classifying emails into spam and non-spam categories.

Whereas, regression deals with continuous data.

For example, predicting stock prices at a certain point in time.

Classification is used to predict the output into a group of classes.

For example, Is it Hot or Cold tomorrow?

Whereas, regression is used to predict the relationship that data represents.

For example, What is the temperature tomorrow?

Advanced Machine Learning Questions

## **13. What is F1 score? How would you use it?**

Let's have a look at this table before directly jumping into the F1 score.

Prediction	Predicted Yes	Predicted No
Actual Yes	True Positive (TP)	False Negative (FN)
Actual No	False Positive (FP)	True Negative (TN)

In binary classification we consider the F1 score to be a measure of the model's accuracy. The F1 score is a weighted average of precision and recall scores.

$$F1 = \frac{2TP}{2TP + FP + FN}$$

We see scores for F1 between 0 and 1, where 0 is the worst score and 1 is the best score.

The F1 score is typically used in information retrieval to see how well a model retrieves relevant results and our model is performing.

## **14. Define Precision and Recall?**

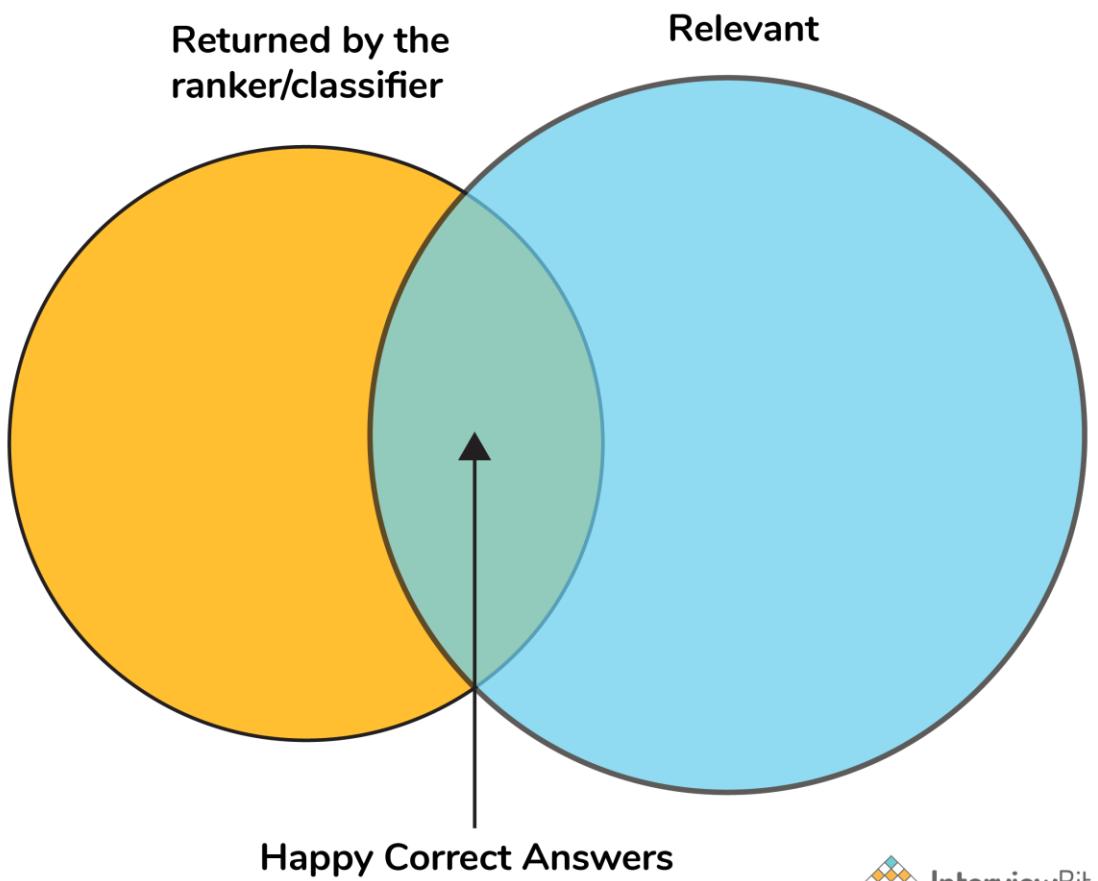
Precision and recall are ways of monitoring the power of machine learning implementation. But they often used at the same time.

Precision answers the question, “Out of the items that the classifier predicted to be relevant, how many are truly relevant?”

Whereas, recall answers the question, “Out of all the items that are truly relevant, how many are found by the classifier?

In general, the meaning of precision is the fact of being exact and accurate. So the same will go in our machine learning model as well. If you have a set of items that your model needs to predict to be relevant. How many items are truly relevant?

The below figure shows the Venn diagram that precision and recall.



Precision and recall

Mathematically, precision and recall can be defined as the following:

precision = # happy correct answers/# total items returned by ranker

recall = # happy correct answers/# total relevant answers

## 15. How to Tackle Overfitting and Underfitting?

Overfitting means the model fitted to training **data too well**, in this case, we need to resample the data and estimate the model accuracy using techniques like k-fold cross-validation.

Whereas for the Underfitting case we are **not able to understand** or capture the patterns from the data, in this case, we need to change the algorithms, or we need to feed more data points to the model.

## 16. What is a Neural Network?

It is a simplified model of the human brain. Much like the brain, it has neurons that activate when encountering something similar.

The different neurons are connected via connections that help information flow from one neuron to another.

## **17. What are Loss Function and Cost Functions? Explain the key Difference Between them?**

When calculating loss we consider only a single data point, then we use the term loss function.

Whereas, when calculating the sum of error for multiple data then we use the cost function. There is no major difference.

In other words, the loss function is to capture the difference between the actual and predicted values for a single record whereas cost functions aggregate the difference for the entire training dataset.

The Most commonly used loss functions are Mean-squared error and Hinge loss.

**Mean-Squared Error(MSE):** In simple words, we can say how our model predicted values against the actual values.

$$\text{MSE} = \sqrt{(\text{predicted value} - \text{actual value})^2}$$

**Hinge loss:** It is used to train the machine learning classifier, which is

$$L(y) = \max(0, 1 - yy)$$

Where  $y = -1$  or  $1$  indicating two classes and  $y$  represents the output form of the classifier. The most common cost function represents the total cost as the sum of the fixed costs and the variable costs in the equation  $y = mx + b$

## **18. What is Ensemble learning?**

Ensemble learning is a method that combines multiple machine learning models to create more powerful models.

There are many reasons for a model to be different. Few reasons are:

- Different Population
- Different Hypothesis
- Different modeling techniques

When working with the model's training and testing data, we will experience an error. This error might be bias, variance, and irreducible error.

Now the model should always have a balance between bias and variance, which we call a bias-variance trade-off.

This ensemble learning is a way to perform this trade-off.

There are many ensemble techniques available but when aggregating multiple models there are two general methods:

- Bagging, a native method: take the training set and generate new training sets off of it.
- Boosting, a more elegant method: similar to bagging, boosting is used to optimize the best weighting scheme for a training set.

## **19. How do you make sure which Machine Learning Algorithm to use?**

It completely depends on the dataset we have. If the data is discrete we use SVM. If the dataset is continuous we use linear regression.

So there is no specific way that lets us know which ML algorithm to use, it all depends on the exploratory data analysis (EDA).

EDA is like “interviewing” the dataset; As part of our interview we do the following:

- Classify our variables as continuous, categorical, and so forth.
- Summarize our variables using descriptive statistics.
- Visualize our variables using charts.

Based on the above observations select one best-fit algorithm for a particular dataset.

## **20. How to Handle Outlier Values?**

An Outlier is an observation in the dataset that is far away from other observations in the dataset. Tools used to discover outliers are

- Box plot
- Z-score
- Scatter plot, etc.

Typically, we need to follow three simple strategies to handle outliers:

- We can drop them.
- We can mark them as outliers and include them as a feature.
- Likewise, we can transform the feature to reduce the effect of the outlier.

## **21. What is a Random Forest? How does it work?**

Random forest is a versatile machine learning method capable of performing both regression and classification tasks.

Like bagging and boosting, random forest works by combining a set of other tree models. Random forest builds a tree from a random sample of the columns in the test data.

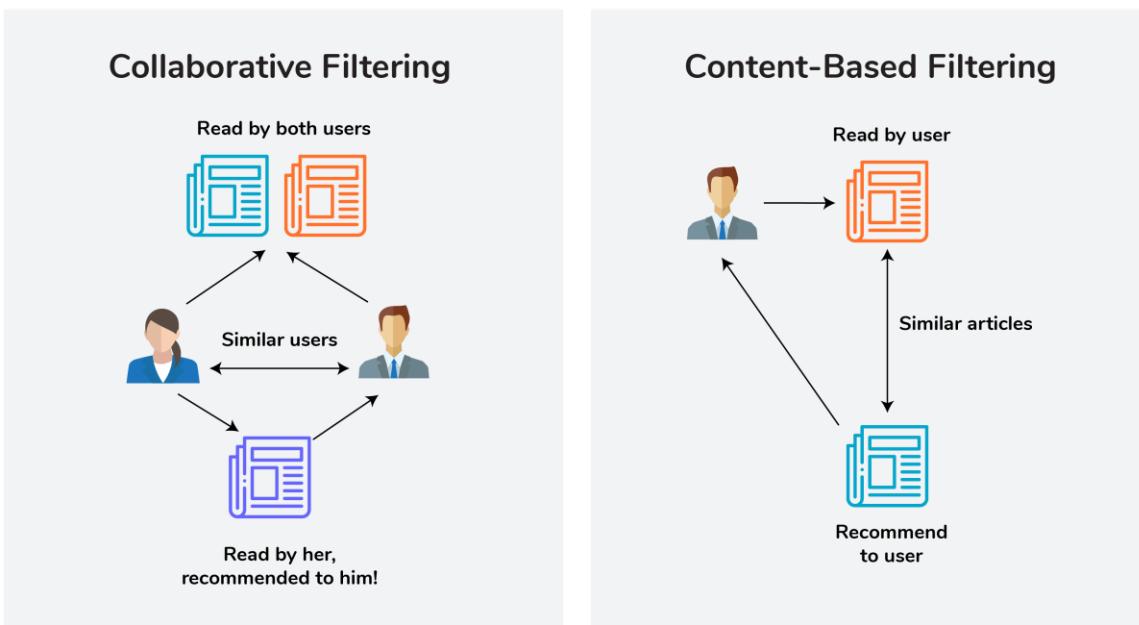
Here's are the steps how a random forest creates the trees:

- Take a sample size from the training data.
- Begin with a single node.
- Run the following algorithm, from the start node:
  - If the number of observations is less than node size then stop.
  - Select random variables.
  - Find the variable that does the “best” job of splitting the observations.
  - Split the observations into two nodes.
  - Call step `a` on each of these nodes.

## **22. What is Collaborative Filtering? And Content-Based Filtering?**

Collaborative filtering is a proven technique for personalized content recommendations. Collaborative filtering is a type of recommendation system that predicts new content by matching the interests of the individual user with the preferences of many users.

Content-based recommender systems are focused only on the preferences of the user. New recommendations are made to the user from similar content according to the user’s previous choices.



## Collaborative Filtering and Content-Based Filtering

### 23. What is Clustering?

Clustering is the process of grouping a set of objects into a number of groups. Objects should be similar to one another within the same cluster and dissimilar to those in other clusters.

A few types of clustering are:

- Hierarchical clustering
- K means clustering
- Density-based clustering
- Fuzzy clustering, etc.

### 24. How can you select K for K-means Clustering?

There are two kinds of methods that include direct methods and statistical testing methods:

- Direct methods: It contains elbow and silhouette
- Statistical testing methods: It has gap statistics.

The silhouette is the most frequently used while determining the optimal value of k.

### 25. What are Recommender Systems?

A recommendation engine is a system used to predict users' interests and recommend products that are quite likely interesting for them.

Data required for recommender systems stems from explicit user ratings after watching a film or listening to a song, from implicit search engine queries and purchase histories, or from other knowledge about the users/items themselves.

## **26. How do check the Normality of a dataset?**

Visually, we can use plots. A few of the normality checks are as follows:

- Shapiro-Wilk Test
- Anderson-Darling Test
- Martinez-Iglewicz Test
- Kolmogorov-Smirnov Test
- D'Agostino Skewness Test

## **27. Can logistic regression use for more than 2 classes?**

No, by default logistic regression is a binary classifier, so it cannot be applied to more than 2 classes. However, it can be extended for solving multi-class classification problems (**multinomial logistic regression**)

## **28. Explain Correlation and Covariance?**

Correlation is used for measuring and also for estimating the quantitative relationship between two variables. Correlation measures how strongly two variables are related. Examples like, income and expenditure, demand and supply, etc.

Covariance is a simple way to measure the correlation between two variables. The problem with covariance is that they are hard to compare without normalization.

## **29. What is P-value?**

P-values are used to make a decision about a hypothesis test. P-value is the minimum significant level at which you can reject the null hypothesis. The lower the p-value, the more likely you reject the null hypothesis.

## **30. What are Parametric and Non-Parametric Models?**

Parametric models will have limited parameters and to predict new data, you only need to know the parameter of the model.

Non-Parametric models have no limits in taking a number of parameters, allowing for more flexibility and to predict new data. You need to know the state of the data and model parameters.

### **31. What is Reinforcement Learning?**

Reinforcement learning is different from the other types of learning like supervised and unsupervised. In reinforcement learning, we are given neither data nor labels. Our learning is based on the rewards given to the agent by the environment.

### **32. Difference Between Sigmoid and Softmax functions?**

The sigmoid function is used for binary classification. The probabilities sum needs to be 1. Whereas, Softmax function is used for multi-classification. The probabilities sum will be 1.

#### 1) What do you understand by Machine learning?

Machine learning is the form of Artificial Intelligence that deals with system programming and automates data analysis to enable computers to learn and act through experiences without being explicitly programmed.

**For example,** Robots are coded in such a way that they can perform the tasks based on data they collect from sensors. They automatically learn programs from data and improve with experiences.

---

#### 2) Differentiate between inductive learning and deductive learning?

In inductive learning, the model learns by examples from a set of observed instances to draw a generalized conclusion. On the other side, in deductive learning, the model first applies the conclusion, and then the conclusion is drawn.

- Inductive learning is the method of using observations to draw conclusions.
- Deductive learning is the method of using conclusions to form observations.

**For example,** if we have to explain to a kid that playing with fire can cause burns. There are two ways we can explain this to a kid; we can show training examples of various fire accidents or images of burnt people and label them as

"Hazardous". In this case, a kid will understand with the help of examples and not play with the fire. It is the form of Inductive machine learning. The other way to teach the same thing is to let the kid play with the fire and wait to see what happens. If the kid gets a burn, it will teach the kid not to play with fire and avoid going near it. It is the form of deductive learning.

---

### 3) What is the difference between Data Mining and Machine Learning?

**Data mining** can be described as the process in which the structured data tries to abstract knowledge or interesting unknown patterns. During this process, machine learning algorithms are used.

**Machine learning** represents the study, design, and development of the algorithms which provide the ability to the processors to learn without being explicitly programmed.

---

### 4) What is the meaning of Overfitting in Machine learning?

Overfitting can be seen in machine learning when a statistical model describes random error or noise instead of the underlying relationship. Overfitting is usually observed when a model is excessively complex. It happens because of having too many parameters concerning the number of training data types. The model displays poor performance, which has been overfitted.

---

### 5) Why overfitting occurs?

The possibility of overfitting occurs when the criteria used for training the model is not as per the criteria used to judge the efficiency of a model.

---

### 6) What is the method to avoid overfitting?

Overfitting occurs when we have a small dataset, and a model is trying to learn from it. By using a large amount of data, overfitting can be avoided. But if we have a small database and are forced to build a model based on that, then we can use a technique known as **cross-validation**. In this method, a model is usually given a dataset of a known data on which training data set is run and dataset of unknown data against which the model is tested. The primary aim of

cross-validation is to define a dataset to "test" the model in the training phase. If there is sufficient data, '**Isotonic Regression**' is used to prevent overfitting.

---

## 7) Differentiate supervised and unsupervised machine learning.

- In supervised machine learning, the machine is trained using labeled data. Then a new dataset is given into the learning model so that the algorithm provides a positive outcome by analyzing the labeled data. For example, we first require to label the data which is necessary to train the model while performing classification.
  - In the unsupervised machine learning, the machine is not trained using labeled data and let the algorithms make the decisions without any corresponding output variables.
- 

## 8) How does Machine Learning differ from Deep Learning?

- Machine learning is all about algorithms which are used to parse data, learn from that data, and then apply whatever they have learned to make informed decisions.
  - Deep learning is a part of machine learning, which is inspired by the structure of the human brain and is particularly useful in feature detection.
- 

## 9) How is KNN different from k-means?

KNN or K nearest neighbors is a supervised algorithm which is used for classification purpose. In KNN, a test sample is given as the class of the majority of its nearest neighbors. On the other side, K-means is an unsupervised algorithm which is mainly used for clustering. In k-means clustering, it needs a set of unlabeled points and a threshold only. The algorithm further takes unlabeled data and learns how to cluster it into groups by computing the mean of the distance between different unlabeled points.

---

## 10) What are the different types of Algorithm methods in Machine Learning?

The different types of algorithm methods in machine learning are:

- Supervised Learning
  - Semi-supervised Learning
  - Unsupervised Learning
  - Transduction
  - Reinforcement Learning
- 

11) What do you understand by Reinforcement Learning technique?

Reinforcement learning is an algorithm technique used in Machine Learning. It involves an agent that interacts with its environment by producing actions & discovering errors or rewards. Reinforcement learning is employed by different software and machines to search for the best suitable behavior or path it should follow in a specific situation. It usually learns on the basis of reward or penalty given for every action it performs.

---

12) What is the trade-off between bias and variance?

Both bias and variance are errors. Bias is an error due to erroneous or overly simplistic assumptions in the learning algorithm. It can lead to the model underfitting the data, making it hard to have high predictive accuracy and generalize the knowledge from the training set to the test set.

Variance is an error due to too much complexity in the learning algorithm. It leads to the algorithm being highly sensitive to high degrees of variation in the training data, which can lead the model to overfit the data.

To optimally reduce the number of errors, we will need to tradeoff bias and variance.

---

13) How do classification and regression differ?

**Classification**

**Regression**

<ul style="list-style-type: none"> <li>○ Classification is the task to predict a discrete class label.</li> </ul>	<ul style="list-style-type: none"> <li>○ Regression is the task to predict continuous quantity.</li> </ul>
<ul style="list-style-type: none"> <li>○ In a classification problem, data is labeled into one of two or more classes.</li> </ul>	<ul style="list-style-type: none"> <li>○ A regression problem needs the prediction of a quantity.</li> </ul>
<ul style="list-style-type: none"> <li>○ A classification having problem with two classes is called binary classification, and more than two classes is called multi-class classification</li> </ul>	<ul style="list-style-type: none"> <li>○ A regression problem containing multiple input variables is called multivariate regression problem</li> </ul>
<ul style="list-style-type: none"> <li>○ Classifying an email as spam or non-spam is an example of a classification problem.</li> </ul>	<ul style="list-style-type: none"> <li>○ Predicting the price of a stock over a period of time is a regression problem.</li> </ul>

---

14) What are the five popular algorithms we use in Machine Learning?

Five popular algorithms are:

- Decision Trees
  - Probabilistic Networks
  - Neural Networks
  - Support Vector Machines
  - Nearest Neighbor
- 

15) What do you mean by ensemble learning?

Numerous models, such as classifiers are strategically made and combined to solve a specific computational program which is known as ensemble learning. The ensemble methods are also known as committee-based learning or learning multiple classifier systems. It trains various hypotheses to fix the same issue. One of the most suitable examples of ensemble modeling is the random forest

trees where several decision trees are used to predict outcomes. It is used to improve the classification, function approximation, prediction, etc. of a model.

---

## 16) What is a model selection in Machine Learning?

The process of choosing models among diverse mathematical models, which are used to define the same data is known as **Model Selection**. Model learning is applied to the fields of **statistics**, **data mining**, and **machine learning**.

---

## 17) What are the three stages of building the hypotheses or model in machine learning?

There are three stages to build hypotheses or model in machine learning:

- **Model building**  
It chooses a suitable algorithm for the model and trains it according to the requirement of the problem.
  - **Applying the model**  
It is responsible for checking the accuracy of the model through the test data.
  - **Model testing**  
It performs the required changes after testing and apply the final model.
- 

## 18) What according to you, is the standard approach to supervised learning?

In supervised learning, the standard approach is to split the set of example into the training set and the test.

---

## 19) Describe 'Training set' and 'training Test'.

In various areas of information of machine learning, a set of data is used to discover the potentially predictive relationship, which is known as 'Training Set'. The training set is an example that is given to the learner. Besides, the 'Test set' is used to test the accuracy of the hypotheses generated by the learner. It is

the set of instances held back from the learner. Thus, the training set is distinct from the test set.

---

20) What are the common ways to handle missing data in a dataset?

Missing data is one of the standard factors while working with data and handling. It is considered as one of the greatest challenges faced by the data analysts. There are many ways one can impute the missing values. Some of the common methods to handle missing data in datasets can be defined as **deleting the rows, replacing with mean/median/mode, predicting the missing values, assigning a unique category, using algorithms that support missing values**, etc.

---

21) What do you understand by ILP?

ILP stands for **Inductive Logic Programming**. It is a part of machine learning which uses logic programming. It aims at searching patterns in data which can be used to build predictive models. In this process, the logic programs are assumed as a hypothesis.

---

22) What are the necessary steps involved in Machine Learning Project?

There are several essential steps we must follow to achieve a good working model while doing a Machine Learning Project. Those steps may include **parameter tuning, data preparation, data collection, training the model, model evaluation, and prediction**, etc.

---

23) Describe Precision and Recall?

Precision and Recall both are the measures which are used in the information retrieval domain to measure how good an information retrieval system reclaims the related data as requested by the user.

**Precision** can be said as a positive predictive value. It is the fraction of relevant instances among the received instances.

On the other side, **recall** is the fraction of relevant instances that have been retrieved over the total amount of relevant instances. The recall is also known as **sensitivity**.

---

#### 24) What do you understand by Decision Tree in Machine Learning?

Decision Trees can be defined as the Supervised Machine Learning, where the data is continuously split according to a certain parameter. It builds classification or regression models as similar as a tree structure, with datasets broken up into ever smaller subsets while developing the decision tree. The tree can be defined by two entities, namely **decision nodes**, and **leaves**. The leaves are the decisions or the outcomes, and the decision nodes are where the data is split. Decision trees can manage both categorical and numerical data.

---

#### 25) What are the functions of Supervised Learning?

- Classification
  - Speech Recognition
  - Regression
  - Predict Time Series
  - Annotate Strings
- 

#### 26) What are the functions of Unsupervised Learning?

- Finding clusters of the data
  - Finding low-dimensional representations of the data
  - Finding interesting directions in data
  - Finding novel observations/ database cleaning
  - Finding interesting coordinates and correlations
- 

#### 27) What do you understand by algorithm independent machine learning?

Algorithm independent machine learning can be defined as machine learning, where mathematical foundations are independent of any particular classifier or learning algorithm.

---

28) Describe the classifier in machine learning.

A classifier is a case of a hypothesis or discrete-valued function which is used to assign class labels to particular data points. It is a system that inputs a vector of discrete or continuous feature values and outputs a single discrete value, the class.

---

29) What do you mean by Genetic Programming?

**Genetic Programming (GP)** is almost similar to an **Evolutionary Algorithm**, a subset of machine learning. Genetic programming software systems implement an algorithm that uses random mutation, a fitness function, crossover, and multiple generations of evolution to resolve a user-defined task. The genetic programming model is based on testing and choosing the best option among a set of results.

---

30) What is SVM in machine learning? What are the classification methods that SVM can handle?

SVM stands for **Support Vector Machine**. SVM are supervised learning models with an associated learning algorithm which analyze the data used for classification and regression analysis.

The classification methods that SVM can handle are:

- Combining binary classifiers
  - Modifying binary to incorporate multiclass learning
- 

31) How will you explain a linked list and an array?

An array is a datatype which is widely implemented as a default type, in almost all the modern programming languages. It is used to store data of a similar type.

But there are many use-cases where we don't know the quantity of data to be stored. For such cases, advanced data structures are required, and one such data structure is **linked list**.

There are some points which explain how the linked list is different from an array:

<b>ARRAY</b>	<b>LINKED LIST</b>
<ul style="list-style-type: none"> <li>An array is a group of elements of a similar data type.</li> </ul>	<ul style="list-style-type: none"> <li>Linked List is an ordered group of elements of the same type, which are connected using pointers.</li> </ul>
<ul style="list-style-type: none"> <li>Elements are stored consecutively in the memory.</li> </ul>	<ul style="list-style-type: none"> <li>New elements can be stored anywhere in memory.</li> </ul>
<ul style="list-style-type: none"> <li>An Array supports <b>Random Access</b>. It means that the elements can be accessed directly using their index value, like arr[0] for 1st element, arr[5] for 6th element, etc. As a result, accessing elements in an array is fast with constant time complexity of O(1).</li> </ul>	<ul style="list-style-type: none"> <li>Linked List supports <b>Sequential Access</b>. It means that we have to traverse the complete linked list, up to that element/note we want to access in a linked list. To access the nth element of a linked list, the time complexity is O(n).</li> </ul>
<ul style="list-style-type: none"> <li>Memory is allocated at <b>compile time</b> as soon as the array is declared. It is known as <b>Static Memory Allocation</b>.</li> </ul>	<ul style="list-style-type: none"> <li>Memory is allocated at <b>runtime</b>, whenever a new node is added. It is known as <b>Dynamic Memory Allocation</b>.</li> </ul>
<ul style="list-style-type: none"> <li><b>Insertion and Deletion</b> operation takes more time in the array, as the memory locations are consecutive and fixed.</li> </ul>	<ul style="list-style-type: none"> <li>In case of a linked list, a new element is stored at the first free available memory location.</li> </ul>

	Thus, Insertion and Deletion operations are fast in the linked list.
<ul style="list-style-type: none"> <li>○ Size of the array must be declared at the time of array declaration.</li> </ul>	<ul style="list-style-type: none"> <li>○ Size of a Linked list is variable. It grows at runtime whenever nodes are added to it.</li> </ul>

---

### 32) What do you understand by the Confusion Matrix?

A confusion matrix is a table which is used for summarizing the performance of a classification algorithm. It is also known as the **error matrix**.

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Where,

**TN**= True Negative

**TP**= True Positive

**FN**= False Negative

**FP**= False Positive

---

### 33) Explain True Positive, True Negative, False Positive, and False Negative in Confusion Matrix with an example.

- **True Positive**

When a model correctly predicts the positive class, it is said to be a true positive.

For example, Umpire gives a Batsman NOT OUT when he is NOT OUT.

- **True Negative**

When a model correctly predicts the negative class, it is said to be a true negative.

For example, Umpire gives a Batsman OUT when he is OUT.

- **False Positive**

When a model incorrectly predicts the positive class, it is said to be a false positive. It is also known as '**Type I**' error.

For example, Umpire gives a Batsman NOT OUT when he is OUT.

- **False Negative**

When a model incorrectly predicts the negative class, it is said to be a false negative. It is also known as '**Type II**' error.

For example, Umpire gives a Batsman OUT when he is NOT OUT.

---

34) What according to you, is more important between model accuracy and model performance?

Model accuracy is a subset of model performance. The accuracy of the model is directly proportional to the performance of the model. Thus, better the performance of the model, more accurate are the predictions.

---

35) What is Bagging and Boosting?

- Bagging is a process in ensemble learning which is used for improving unstable estimation or classification schemes.
  - Boosting methods are used sequentially to reduce the bias of the combined model.
- 

36) What are the similarities and differences between bagging and boosting in Machine Learning?

### **Similarities of Bagging and Boosting**

- Both are the ensemble methods to get N learns from 1 learner.
- Both generate several training data sets with random sampling.

- Both generate the final result by taking the average of N learners.
- Both reduce variance and provide higher scalability.

## Differences between Bagging and Boosting

- Although they are built independently, but for Bagging, Boosting tries to add new models which perform well where previous models fail.
  - Only Boosting determines the weight for the data to tip the scales in favor of the most challenging cases.
  - Only Boosting tries to reduce bias. Instead, Bagging may solve the problem of over-fitting while boosting can increase it.
- 

37) What do you understand by Cluster Sampling?

Cluster Sampling is a process of randomly selecting intact groups within a defined population, sharing similar characteristics. Cluster sample is a probability where each sampling unit is a collection or cluster of elements.

**For example**, if we are clustering the total number of managers in a set of companies, in that case, managers (sample) will represent elements and companies will represent clusters.

---

38) What do you know about Bayesian Networks?

Bayesian Networks also referred to as '**belief networks**' or '**casual networks**', are used to represent the graphical model for probability relationship among a set of variables.

**For example**, a Bayesian network can be used to represent the probabilistic relationships between diseases and symptoms. As per the symptoms, the network can also compute the probabilities of the presence of various diseases.

Efficient algorithms can perform inference or learning in Bayesian networks. Bayesian networks which relate the variables (e.g., speech signals or protein sequences) are called dynamic Bayesian networks.

---

39) Which are the two components of Bayesian logic program?

A Bayesian logic program consists of two components:

- **Logical**

It contains a set of Bayesian Clauses, which capture the qualitative structure of the domain.

- **Quantitative**

It is used to encode quantitative information about the domain.

---

40) Describe dimension reduction in machine learning.

Dimension reduction is the process which is used to reduce the number of random variables under considerations.

Dimension reduction can be divided into feature selection and extraction.

---

41) Why instance-based learning algorithm sometimes referred to as Lazy learning algorithm?

In machine learning, **lazy learning** can be described as a method where induction and generalization processes are delayed until classification is performed. Because of the same property, an instance-based learning algorithm is sometimes called lazy learning algorithm.

---

42) What do you understand by the F1 score?

The F1 score represents the measurement of a model's performance. It is referred to as a weighted average of the precision and recall of a model. The results tending to **1** are considered as the best, and those tending to **0** are the worst. It could be used in classification tests, where true negatives don't matter much.

---

43) How is a decision tree pruned?

Pruning is said to occur in decision trees when the branches which may consist of weak predictive power are removed to reduce the complexity of the model and increase the predictive accuracy of a decision tree model. Pruning can occur

bottom-up and top-down, with approaches such as **reduced error pruning** and **cost complexity pruning**.

Reduced error pruning is the simplest version, and it replaces each node. If it is unable to decrease predictive accuracy, one should keep it pruned. But, it usually comes pretty close to an approach that would optimize for maximum accuracy.

---

#### 44) What are the Recommended Systems?

Recommended System is a sub-directory of information filtering systems. It predicts the preferences or rankings offered by a user to a product. According to the preferences, it provides similar recommendations to a user.

Recommendation systems are widely used in **movies, news, research articles, products, social tips, music**, etc.

---

#### 45) What do you understand by Underfitting?

Underfitting is an issue when we have a low error in both the training set and the testing set. Few algorithms work better for interpretations but fail for better predictions.

---

#### 46) When does regularization become necessary in Machine Learning?

Regularization is necessary whenever the model begins to overfit/ underfit. It is a cost term for bringing in more features with the objective function. Hence, it tries to push the coefficients for many variables to zero and reduce cost term. It helps to reduce model complexity so that the model can become better at predicting (generalizing).

---

#### 47) What is Regularization? What kind of problems does regularization solve?

A regularization is a form of regression, which constrains/ regularizes or shrinks the coefficient estimates towards zero. In other words, it discourages learning a more complex or flexible model to avoid the risk of overfitting. It reduces the variance of the model, without a substantial increase in its bias.

Regularization is used to address overfitting problems as it penalizes the loss function by adding a multiple of an L1 (LASSO) or an L2 (Ridge) norm of weights vector  $w$ .

---

48) Why do we need to convert categorical variables into factor? Which functions are used to perform the conversion?

Most Machine learning algorithms require number as input. That is why we convert categorical values into factors to get numerical values. We also don't have to deal with dummy variables.

The functions **factor()** and **as.factor()** are used to convert variables into factors.

---

49) Do you think that treating a categorical variable as a continuous variable would result in a better predictive model?

For a better predictive model, the categorical variable can be considered as a continuous variable only when the variable is ordinal in nature.

---

50) How is machine learning used in day-to-day life?

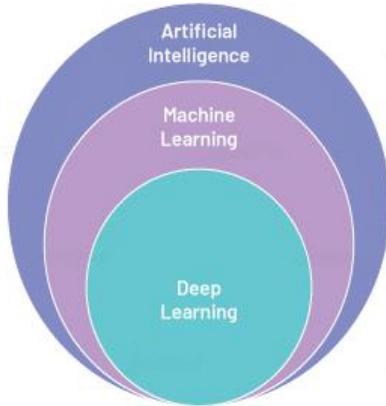
Most of the people are already using machine learning in their everyday life. Assume that you are engaging with the internet, you are actually expressing your preferences, likes, dislikes through your searches. All these things are picked up by cookies coming on your computer, from this, the behavior of a user is evaluated. It helps to increase the progress of a user through the internet and provide similar suggestions.

The navigation system can also be considered as one of the examples where we are using machine learning to calculate a distance between two places using optimization techniques. Surely, people are going to more engage with machine learning in the near future.

## **1. Explain Machine Learning, Artificial Intelligence, and Deep Learning**

It is common to get confused between the three in-demand technologies, Machine Learning, Artificial Intelligence, and Deep Learning. These three

technologies, though a little different from one another, are interrelated. While Deep Learning is a subset of Machine Learning, Machine Learning is a subset of Artificial Intelligence. Since some terms and techniques may overlap in these technologies, it is easy to get confused among them.



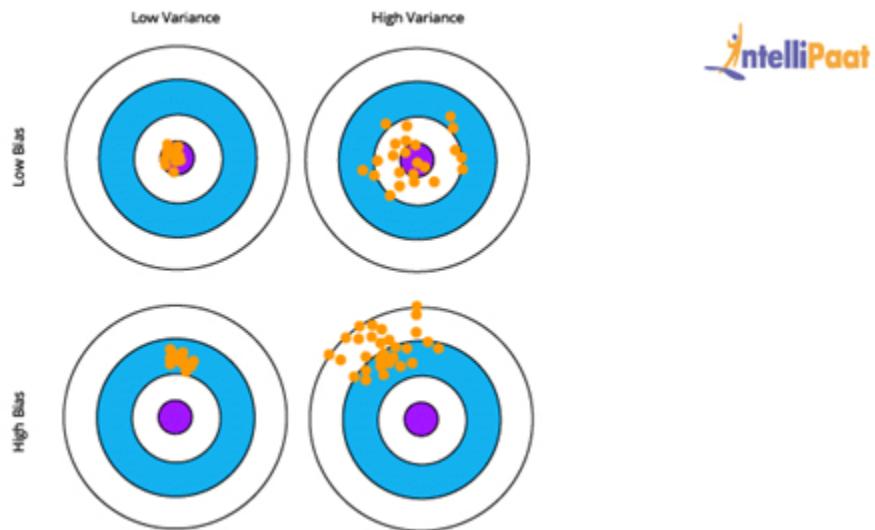
So, let us learn about these technologies in detail:

- Machine Learning: Machine Learning involves various statistical and Deep Learning techniques that allow machines to use their past experiences and get better at performing specific tasks without having to be monitored.
- Artificial Intelligence: Artificial Intelligence uses numerous Machine Learning and Deep Learning techniques that enable computer systems to perform tasks using human-like intelligence with logic and rules. Artificial intelligence is used in every sector hence it is necessary to pursue Artificial Intelligence Course to make your career in AI.
- Deep Learning: Deep Learning comprises several algorithms that enable software to learn from themselves and perform various business tasks including image and speech recognition. Deep Learning is possible when systems expose their multilayered neural networks to large volumes of data for learning.

## 2. What is Bias and Variance in Machine Learning?

- Bias is the difference between the average prediction of a model and the correct value of the model. If the bias value is high, then the prediction of the model is not accurate. Hence, the bias value should be as low as possible to make the desired predictions.
- Variance is the number that gives the difference of prediction over a training set and the anticipated value of other training sets. High variance may lead to large fluctuation in the output. Therefore, a model's output should have low variance.

The following diagram shows the bias-variance trade-off:



Here, the desired result is the blue circle at the center. If we get off from the blue section, then the prediction goes wrong.

### 3. What is Clustering in Machine Learning?

Clustering is a technique used in unsupervised learning that involves grouping data points. The clustering algorithm can be used with a set of data points. This technique will allow you to classify all data points into their particular groups. The data points that are thrown into the same category have similar features and properties, while the data points that belong to different groups have distinct features and properties. Statistical data analysis can be performed by this method. Let us take a look at three of the most popular and useful clustering algorithms.

- K-means clustering: This algorithm is commonly used when there is data with no specific group or category. K-means clustering allows you to find the hidden patterns in the data, which can be used to classify the data into various groups. The variable  $k$  is used to represent the number of groups the data is divided into, and the data points are clustered using the similarity of features. Here, the centroids of the clusters are used for labeling new data.
- Mean-shift clustering: The main aim of this algorithm is to update the center-point candidates to be mean and find the center points of all groups. In mean-shift clustering, unlike k-means clustering, the possible number of clusters need not be selected as it can automatically be discovered by the mean shift.
- Density-based spatial clustering of applications with noise (DBSCAN): This clustering algorithm is based on density and has similarities with mean-shift clustering. There is no need to preset the number of clusters, but unlike mean-shift clustering, DBSCAN identifies outliers and treats them like noise. Moreover, it can identify arbitrarily-sized and -shaped clusters without much effort.

#### 4. What is Linear Regression in Machine Learning?

Linear Regression is a supervised Machine Learning algorithm. It is used to find the linear relationship between the dependent and independent variables for predictive analysis.

The equation for Linear Regression:

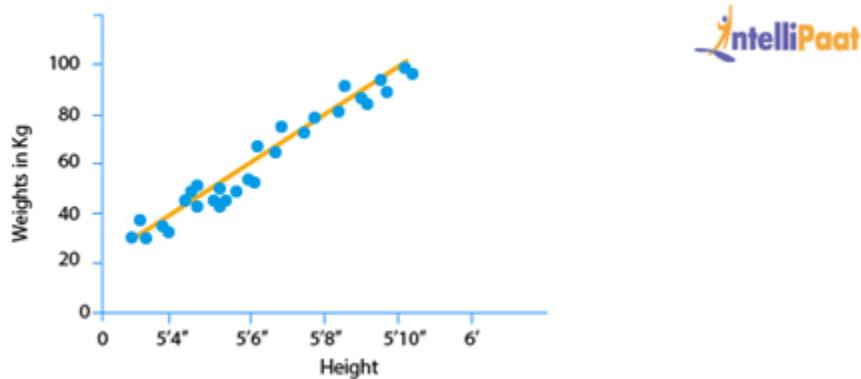
$$Y = A + B.X$$



where:

- X is the input or independent variable
- Y is the output or dependent variable
- a is the intercept, and b is the coefficient of X

Below is the best-fit line that shows the data of weight, Y or the dependent variable, and the



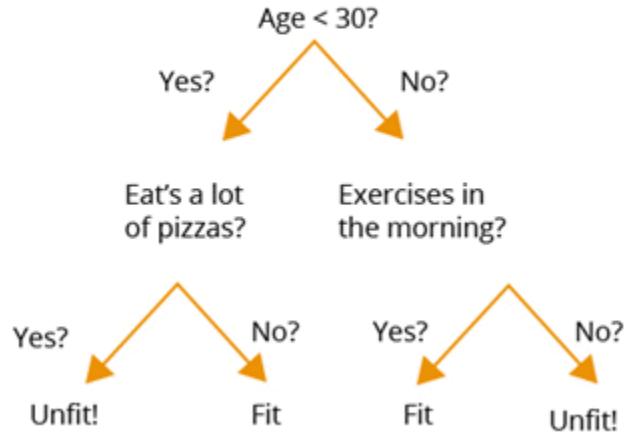
ata of height, X or the independent variable, of 21-year-old candidates scattered over the plot. The straight line shows the best linear relationship that would help in predicting the weight of candidates according to their height.

To get this best-fit line, the best values of a and b should be found. By adjusting the values of a and b, the errors in the prediction of Y can be reduced.

This is how linear regression helps in finding the linear relationship and predicting the output.

## 5. What is a Decision Tree in Machine Learning?

A decision tree is used to explain the sequence of actions that must be performed to get the desired output. It is a hierarchical diagram that shows the actions.



An algorithm can be created for a decision tree on the basis of the set hierarchy of actions.

In the above decision-tree diagram, a sequence of actions has been made for driving a vehicle with or without a license.

## 6. What is Overfitting in Machine Learning and how can it be avoided?

Overfitting happens when a machine has an inadequate dataset and tries to learn from it. So, overfitting is inversely proportional to the amount of data.

For small databases, overfitting can be bypassed by the cross-validation method. In this approach, a dataset is divided into two sections. These two sections will comprise the testing and training dataset. To train a model, the training dataset is used, and for testing the model for new inputs, the testing dataset is used. This is how to avoid overfitting.

## 7. What is Hypothesis in Machine Learning?

Machine Learning allows the use of available dataset to understand a specific function that maps input to output in the best possible way. This problem is known as function approximation. Here, approximation needs to be used for the unknown target function that maps all plausible observations based on the given

problem in the best manner. Hypothesis in Machine learning is a model that helps in approximating the target function and performing the necessary input-to-output mappings. The choice and configuration of algorithms allow defining the space of plausible hypotheses that may be represented by a model.

In the hypothesis, lowercase h ( $h$ ) is used for a specific hypothesis, while uppercase h ( $H$ ) is used for the hypothesis space that is being searched. Let us briefly understand these notations:

- Hypothesis ( $h$ ): A hypothesis is a specific model that helps in mapping input to output; the mapping can further be used for evaluation and prediction.
- Hypothesis set ( $H$ ): Hypothesis set consists of a space of hypotheses that can be used to map inputs to outputs, which can be searched. The general constraints include the choice of problem framing, the model, and the model configuration.

## **8. What are the differences between Deep Learning and Machine Learning?**

- Deep Learning: Deep Learning allows machines to make various business-related decisions using artificial neural networks, which is one of the reasons why it needs a vast amount of data for training. Since there is a lot of computing power required, Deep Learning requires high-end systems as well. The systems acquire various properties and features with the help of the given data, and the problem is solved using an end-to-end method.
- Machine Learning: Machine Learning gives machines the ability to make business decisions without any external help, using the knowledge gained from past data. Machine Learning systems require relatively small amounts of data to train themselves, and most of the features need to be manually coded and understood in advance. In Machine Learning, a given business problem is dissected into two

and then solved individually. Once the solutions of both have been acquired, they are then combined.

## **9. What are the differences between Supervised and Unsupervised Machine Learning?**

- Supervised learning: The algorithms of supervised learning use labeled data to get trained. The models take direct feedback to confirm whether the output that is being predicted is, indeed, correct. Moreover, both the input data and the output data are provided to the model, and the main aim here is to train the model to predict the output upon receiving new data. Supervised learning offers accurate results and can largely be divided into two parts, classification and regression.
- Unsupervised learning: The algorithms of unsupervised learning use unlabeled data for training purposes. In unsupervised learning, the models identify hidden data trends and do not take any feedback. The unsupervised learning model is only provided with input data. Unsupervised learning's main aim is to identify hidden patterns to extract information from unknown sets of data. It can also be classified into two parts, clustering, and associations. Unfortunately, unsupervised learning offers results that are comparatively less accurate.

## **10. What is Bayes's Theorem in Machine Learning?**

Bayes's theorem offers the probability of any given event to occur using prior knowledge. In mathematical terms, it can be defined as the true positive rate of the given sample condition divided by the sum of the true positive rate of the said condition and the false positive rate of the entire population.

Two of the most significant applications of Bayes's theorem in Machine Learning are Bayesian optimization and Bayesian belief networks. This theorem

is also the foundation behind the Machine Learning brand that involves the Naive Bayes classifier.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

The diagram illustrates the Naive Bayes formula. At the center is the formula  $P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$ . Four arrows point to different parts of the formula:

- An arrow points to  $P(B|A)$  from the text "Probability of B occurring given evidence A has already occurred".
- An arrow points to  $P(A)$  from the text "Probability of A occurring".
- An arrow points to  $P(B)$  from the text "Probability of B occurring".
- An arrow points to  $P(B|A)$  from the text "Probability of B occurring given evidence B has already occurred".

## 11. What is PCA in Machine Learning?

Multidimensional data is at play in the real world. Data visualization and computation become more challenging with the increase in dimensions. In such a scenario, the dimensions of data might have to be reduced to analyze and visualize it easily. This is done by:

- Removing irrelevant dimensions
- Keeping only the most relevant dimensions

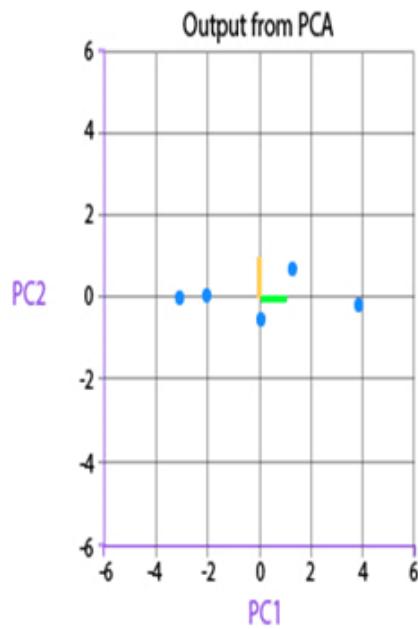
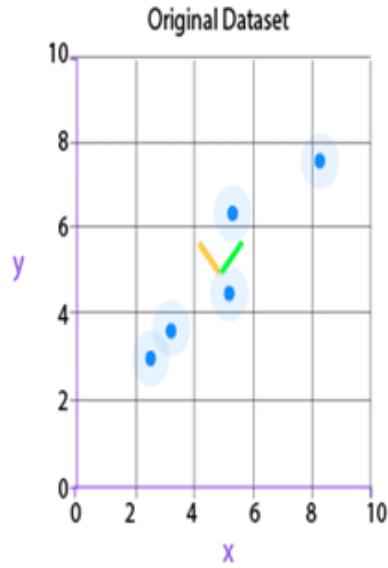
This is where Principal Component Analysis (PCA) is used.

The goal of PCA is to find a fresh collection of uncorrelated dimensions (orthogonal) and rank them on the basis of variance.

Mechanism of PCA:

- Compute the covariance matrix for data objects
- Compute eigenvectors and eigenvalues in descending order
- Select the initial  $N$  eigenvectors to get new dimensions
- Finally, change the initial n-dimensional data objects into N-dimensions

**Example:** Below are two graphs showing data points or objects and two directions, one is green and the other is yellow. Graph 2 is arrived at by rotating Graph 1 so that the x-axis and y-axis represent the green and yellow direction respectively.



After the rotation of data points, it can be inferred that the green direction, the x-axis, gives the line that best fits the data points.

Here, two-dimensional data is being represented; but in real life, the data would be multidimensional and complex. So, after recognizing the importance of each

direction, the area of dimensional analysis can be reduced by cutting off the less-significant directions.

Now, we will go through another important Machine Learning interview question on PCA.

### 12. What is Support Vector Machine (SVM) in Machine Learning?

SVM is a Machine Learning algorithm that is majorly used for classification. It is used on top of the high dimensionality of the characteristic vector.

The following is the code for SVM classifier:

```
# Introducing required libraries
from sklearn import datasets
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import train_test_split
# Stacking the Iris dataset
iris = datasets.load_iris()
# A -> features and B -> label
A = iris.data
B = iris.target
# Breaking A and B into train and test data
A_train, A_test, B_train, B_test = train_test_split(A, B, random_state = 0)
# Training a linear SVM classifier
from sklearn.svm import SVC
svm_model_linear = SVC(kernel = 'linear', C = 1).fit(A_train, B_train)
svm_predictions = svm_model_linear.predict(A_test)
# Model accuracy for A_test
accuracy = svm_model_linear.score(A_test, B_test)
# Creating a confusion matrix
cm = confusion_matrix(B_test, svm_predictions)
```

### 13. What is Cross-validation in Machine Learning?

Cross-validation allows a system to increase the performance of the given Machine Learning algorithm, which is fed a number of sample data from the

dataset. This sampling process is done to break the dataset into smaller parts that have the same number of rows, out of which a random part is selected as a test set and the rest of the parts are kept as train sets. Cross-validation consists of the following techniques:

- Holdout method
- K-fold cross-validation
- Stratified k-fold cross-validation
- Leave p-out cross-validation

## **14. What is Entropy in Machine Learning?**

Entropy in Machine Learning measures the randomness in the data that needs to be processed. The more entropy in the given data, the more difficult it becomes to draw any useful conclusion from the data. For example, let us take the flipping of a coin. The result of this act is random as it does not favor heads or tails. Here, the result for any number of tosses cannot be predicted easily as there is no definite relationship between the action of flipping and the possible outcomes.

## **15. What is Epoch in Machine Learning?**

Epoch in Machine Learning is used to indicate the count of passes in a given training dataset where the Machine Learning algorithm has done its job. Generally, when there is a large chunk of data, it is grouped into several batches. All these batches go through the given model, and this process is referred to as iteration. Now, if the batch size comprises the complete training dataset, then the count of iterations is the same as that of epochs.

In case there is more than one batch,  $d \times e = i \times b$  is the formula used, wherein d is the dataset, e is the number of epochs, i is the number of iterations, and b is the batch size.

## **16. What are the types of Machine Learning?**

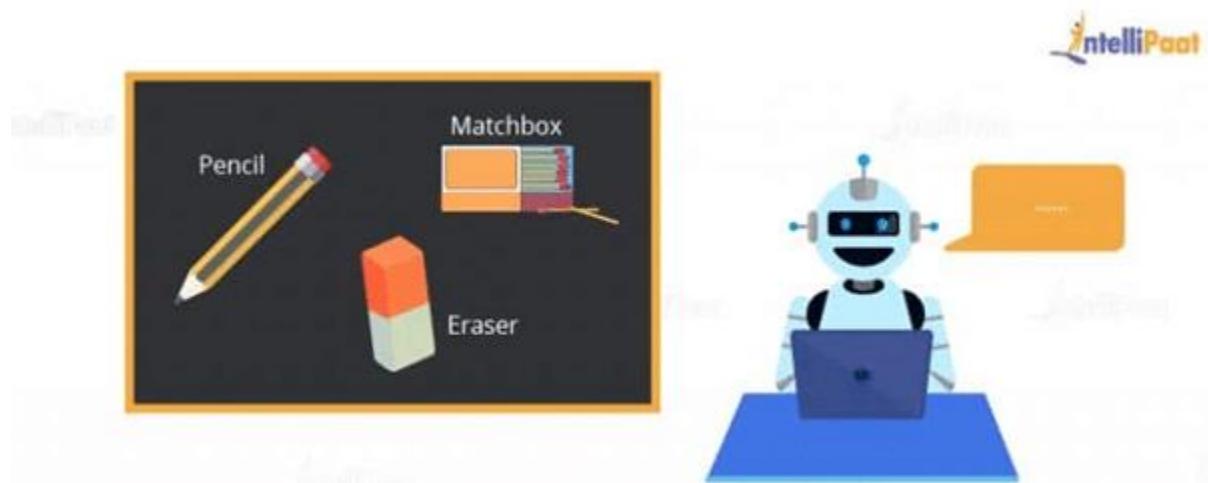
This is one of the most basic interview questions that everyone must know.

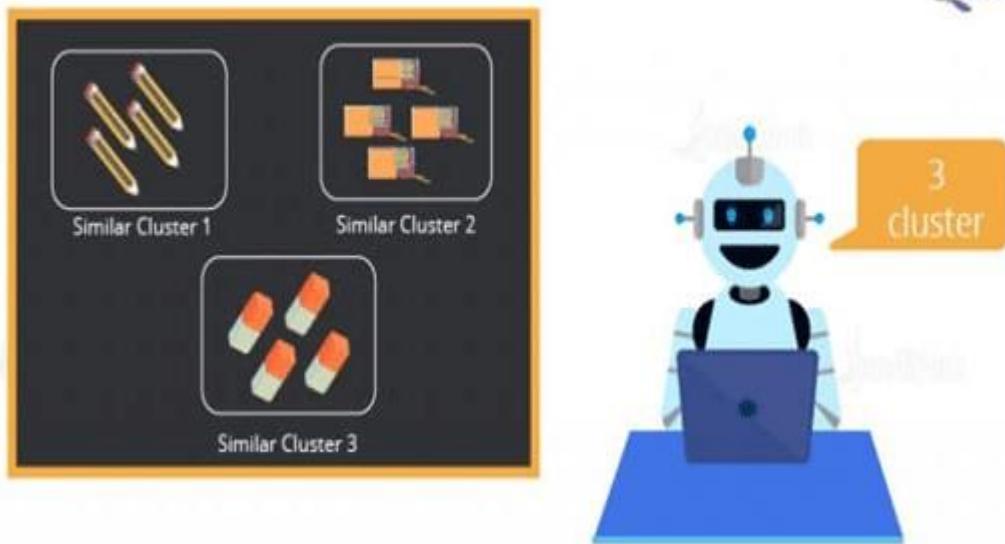
So, basically, there are three types of Machine Learning. They are described as follows:

Supervised learning: In this type of Machine Learning, machines learn under the supervision of labeled data. There is a training dataset on which a machine is trained, and it gives the output according to its training.

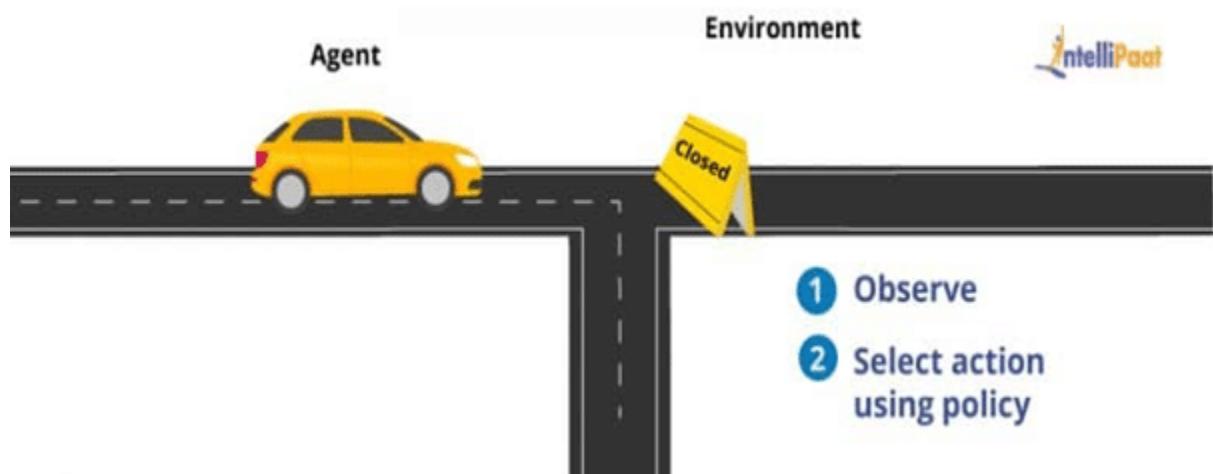


Unsupervised learning: This type of Machine Learning has unlabeled data unlike supervised learning. Unsupervised learning works on data under absolutely no supervision. Unsupervised learning tries to identify patterns in data and makes clusters of similar entities. After that, when a new input data is fed into the model, it does not identify the entity; rather, it puts the entity in a cluster of similar objects.





**Reinforcement learning:** Reinforcement learning includes models that learn and traverse to find the best possible move. The algorithms for reinforcement learning are constructed in a way that they try to find the best possible suite of action on the basis of the reward and punishment theory.





For next time....



## 17. Differentiate between Classification and Regression in Machine Learning

In Machine Learning, there are various types of prediction problems based on supervised and unsupervised learning. They are classification, regression, clustering, and association. Here, we will discuss classification and regression.

**Classification:** In classification, a Machine Learning model is created that assists in differentiating data into separate categories. The data is labeled and categorized based on the input parameters.

For example, predictions have to be made on the churning out customers for a particular product based on some recorded data. Either the customers will churn out or they will not. So, the labels for this would be “Yes” and “No.”

**Regression:** It is the process of creating a model for distinguishing data into continuous real values, instead of using classes or discrete values. It can also identify the distribution movement depending on historical data. It is used for predicting the occurrence of an event depending on the degree of association of variables.

For example, the prediction of weather conditions depends on factors such as temperature, air pressure, solar radiation, elevation, and distance from the sea. The relation among these factors assists in predicting the weather condition.

## **18. How is the suitability of a Machine Learning Algorithm determined for a particular problem?**

To identify a Machine Learning Algorithm for a particular problem, the following steps should be followed:

**Step 1:** Problem classification: Classification of the problem depends on the classification of input and output:

- Classifying the input: Classification of the input depends on whether there is data labeled (supervised learning) or unlabeled (unsupervised learning), or whether a model has to be created that interacts with the environment and improves itself (reinforcement learning.)
- Classifying the output: If the output of a model is required as a class, then some classification techniques need to be used.

If the output is a number, then regression techniques must be used; if the output is a different cluster of inputs, then clustering techniques should be used.

**Step 2:** Checking the algorithms in hand: After classifying the problem, the available algorithms that can be deployed for solving the classified problem should be considered.

**Step 3:** Implementing the algorithms: If there are multiple algorithms available, then all of them are to be implemented. Finally, the algorithm that gives the best performance is selected.

## 19. What is the Variance Inflation Factor?

Variance inflation factor (VIF) is the estimate of the volume of multicollinearity in a collection of many regression variables.

$VIF = \text{Variance of the model} / \text{Variance of the model with a single independent variable}$

This ratio has to be calculated for every independent variable. If VIF is high, then it shows the high collinearity of the independent variables.

## 20. What is a Confusion Matrix?

Confusion matrix is used to explain a model's performance and gives a summary of predictions of the classification problems. It assists in identifying the uncertainty between classes.

Confusion matrix gives the count of correct and incorrect values and error types. Accuracy of the model:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$



For example, consider the following confusion matrix. It consists of values as true positive, true negative, false positive, and false negative for a classification model. Now, the accuracy of the model can be calculated as follows:

	Predicted: No	Predicted: Yes	
Actual: No	TP = 200	FN = 60	260
Actual: Yes	FP = 10	TN = 50	60
210		110	

So, in the example:

$$\text{Accuracy} = (200 + 50) / (200 + 50 + 10 + 60) = 0.78$$

This means that the model's accuracy is 0.78, corresponding to its True Positive, True Negative, False Positive, and False Negative values.

## 21. What are Type I and Type II Errors?

**Type I Error:** Type I Error, false positive, is an error where the outcome of a test shows the nonacceptance of a true condition.

For example, suppose a person gets diagnosed with depression even when they are not suffering from the same, it is a case of false positive.

**Type II Error:** Type II Error, false negative, is an error where the outcome of a test shows the acceptance of a false condition.

For example, the CT scan of a person shows that they do not have a disease but in fact they do have the disease. Here, the test accepts the false condition that the person does not have the disease. This is a case of false negative.

## 22. When should Classification be used over Regression?

Both classification and regression are associated with prediction. Classification involves the identification of values or entities that lie in a specific group. Regression entails predicting a response value from consecutive sets of outcomes.

Classification is chosen over regression when the output of the model needs to yield the belongingness of data points in a dataset to a particular category.

For example, If you want to predict the price of a house, you should use regression since it is a numerical variable. However, if you are trying to predict whether a house situated in a particular area is going to be high-, medium-, or low-priced, then a classification model should be used.

### **23. Explain Logistic Regression**

Logistic regression is the proper regression analysis used when the dependent variable is categorical or binary. Like all regression analyses, logistic regression is a technique for predictive analysis. Logistic regression is used to explain data and the relationship between one dependent binary variable and one or more independent variables. Logistic regression is also employed to predict the probability of categorical dependent variables.

Logistic regression can be used in the following scenarios:

- To predict whether a citizen is a Senior Citizen (1) or not (0)
- To check whether a person has a disease (Yes) or not (No)

There are three types of logistic regression:

- Binary logistic regression: In this type of logistic regression, there are only two outcomes possible.

Example: To predict whether it will rain (1) or not (0)

- Multinomial logistic regression: In this type of logistic regression, the output consists of three or more unordered categories.

Example: Predicting whether the prize of the house is high, medium, or low.

- Ordinal logistic regression: In this type of logistic regression, the output consists of three or more ordered categories.

Example: Rating an Android application from one to five stars.

## **24. How to handle Missing or Corrupted Data in a Dataset?**

In Python pandas, there are two methods to locate lost or corrupted data and discard those values:

- `isNull()`: It can be used for detecting the missing values.
- `dropna()`: It can be used for removing columns or rows with null values.

`fillna()` can be used to fill the void values with placeholder values.

## **25. Why is rotation required in PCA? What will happen if the components are not rotated?**

Rotation is a significant step in principal component analysis (PCA.) Rotation maximizes the separation within the variance obtained by the components. This makes the interpretation of the components easier.

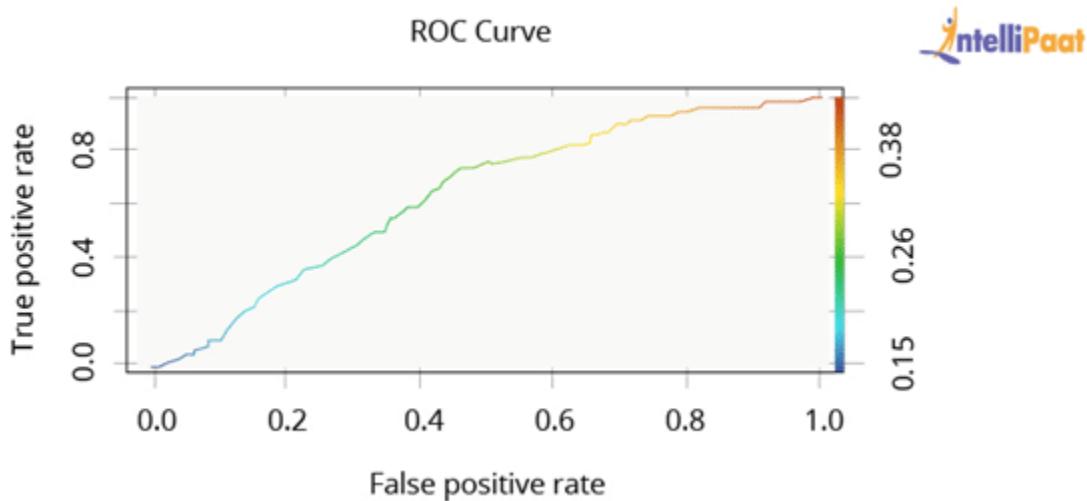
The motive behind conducting PCA is to choose fewer components that can explain the greatest variance in a dataset. When rotation is performed, the original coordinates of the points get changed. However, there is no change in the relative position of the components.

If the components are not rotated, then there needs to be more extended components to describe the variance.

## **26. What is ROC Curve and what does it represent?**

ROC stands for receiver operating characteristic. ROC Curve is used to graphically represent the trade-off between true and false-positive rates.

In ROC, the area under the curve (AUC) gives an idea about the accuracy of the model.



The above graph shows a ROC curve. The greater the AUC, the better the performance of the model.

Next, we will be taking a look at Machine Learning interview questions on rescaling, binarizing, and standardizing.

## 27. Why are Validation and Test Datasets Needed?

Data is split into three different categories while creating a model:

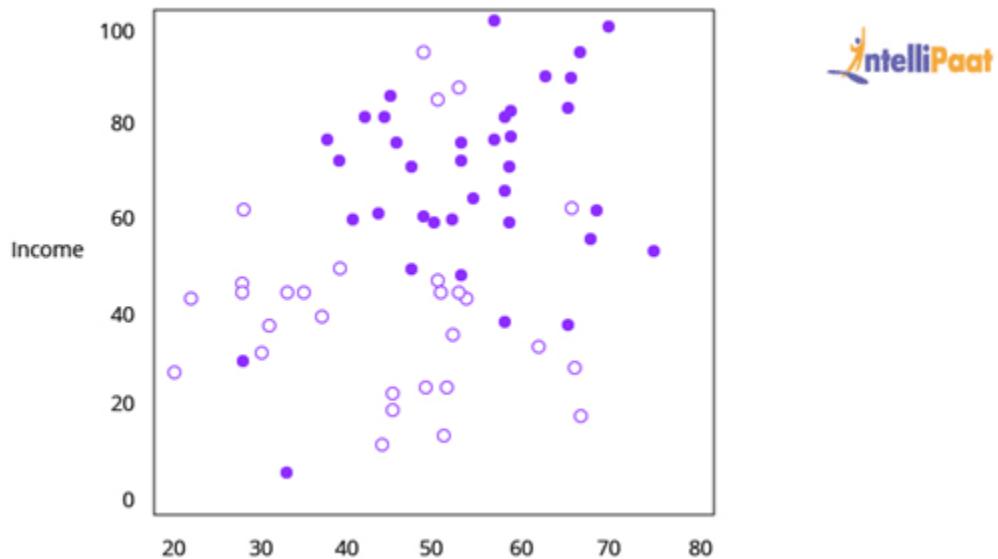
- **Training dataset:** Training dataset is used for building a model and adjusting its variables. The correctness of the model built on the training dataset cannot be relied on as the model might give incorrect outputs after being fed new inputs.
- **Validation dataset:** Validation dataset is used to look into a model's response. After this, the hyperparameters on the basis of the estimated benchmark of the validation dataset data are tuned. When a model's response is evaluated by using the validation dataset, the model is indirectly trained with the validation set. This may lead to the overfitting of the model to specific data. So, this model will not be strong enough to give the desired response to real-world data.
- **Test dataset:** Test dataset is the subset of the actual dataset, which is not yet used to train the model. The model is unaware of this dataset. So, by using the test dataset, the response of the created model can be

computed on hidden data. The model's performance is tested on the basis of the test dataset. Note: The model is always exposed to the test dataset after tuning the hyperparameters on top of the validation dataset.

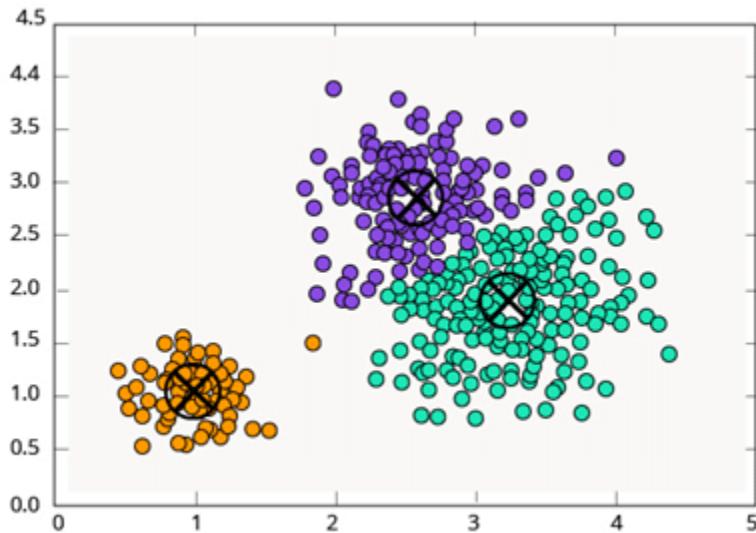
As we know, the evaluation of the model on the basis of the validation dataset would not be enough. Thus, the test dataset is used for computing the efficiency of the model.

## 28. Explain the difference between KNN and K-means Clustering

**K-nearest neighbors (KNN):** It is a supervised Machine Learning algorithm. In KNN, identified or labeled data is given to the model. The model then matches the points based on the distance from the closest points.



**K-means clustering:** It is an unsupervised Machine Learning algorithm. In K-means clustering, unidentified or unlabeled data is given to the model. The algorithm then creates batches of points based on the average of the distances between distinct points.



## 29. What is Dimensionality Reduction?

In the real world, Machine Learning models are built on top of features and parameters. These features can be multidimensional and large in number. Sometimes, the features may be irrelevant and it becomes a difficult task to visualize them.

This is where dimensionality reduction is used to cut down irrelevant and redundant features with the help of principal variables. These principal variables conserve the features, and are a subgroup, of the parent variables.

## 30. Both being Tree-based Algorithms, how is Random Forest different from Gradient Boosting Machine (GBM)?

The main difference between a random forest and GBM is the use of techniques. Random forest advances predictions using a technique called bagging. On the other hand, GBM advances predictions with the help of a technique called boosting.

- **Bagging:** In bagging, we apply arbitrary sampling and we divide the dataset into  $N$ . After that, we build a model by employing a single training algorithm. Following that, we combine the final predictions

by polling. Bagging helps to increase the efficiency of a model by decreasing the variance to eschew overfitting.

- **Boosting:** In boosting, the algorithm tries to review and correct the inadmissible predictions at the initial iteration. After that, the algorithm's sequence of iterations for correction continues until we get the desired prediction. Boosting assists in reducing bias and variance for strengthening the weak learners.

### **31. What is meant by Parametric and Non-parametric Models?**

Parametric models refer to the models having a limited number of parameters. In case of parametric models, only the parameter of a model is needed to be known to make predictions regarding the new data.

Non-parametric models do not have any restrictions on the number of parameters, which makes new data predictions more flexible. In case of non-parametric models, the knowledge of model parameters and the state of the data needs to be known to make predictions.

### **32. Differentiate between Sigmoid and Softmax Functions**

Sigmoid and Softmax functions differ based on their usage in Machine Learning task classification. Sigmoid function is used in the case of binary classification, while Softmax function is used in case of multi-classification.

### **33. In Machine Learning, for how many classes can Logistic Regression be used?**

Logistic regression cannot be used for more than two classes. Logistic regression is, by default, a binary classifier. However, in cases where multi-class classification problems need to be solved, the default number of classes can be extended, i.e., multinomial logistic regression.

### **34. What do you understand about the P-value?**

P-value is used in decision-making while testing a hypothesis. The null hypothesis is rejected at the minimum significance level of the P-value. A lower P-value indicates that the null hypothesis is to be rejected.

### **35. What is meant by Correlation and Covariance?**

Correlation is a mathematical concept used in statistics and probability theory to measure, estimate, and compare data samples taken from different populations. In simpler terms, correlation helps in establishing a quantitative relationship between two variables.

Covariance is also a mathematical concept; it is a simpler way to arrive at a correlation between two variables. Covariance basically helps in determining what change or affect does one variable has on another.

### **36. What are the Various Tests for Checking the Normality of a Dataset?**

In Machine Learning, checking the normality of a dataset is very important. Hence, certain tests are performed on a dataset to check its normality. Some of them are:

- D'Agostino Skewness Test
- Shapiro-Wilk Test
- Anderson-Darling Test
- Jarque-Bera Test
- Kolmogorov-Smirnov Test

### **37. What are the Two Main Types of Filtering in Machine Learning?**

**Explain.**

The two types of filtering are:

- Collaborative filtering

- Content-based filtering

Collaborative filtering refers to a recommender system where the interests of the individual user are matched with preferences of multiple users to predict new content.

Content-based filtering is a recommender system where the focus is only on the preferences of the individual user and not on multiple users.

### **38. Outlier Values can be Discovered from which Tools?**

The various tools that can be used to discover outlier values are scatterplots, boxplots, Z-score, etc.

### **39. What is meant by Ensemble Learning?**

Ensemble learning refers to the combination of multiple Machine Learning models to create more powerful models. The primary techniques involved in ensemble learning are bagging and boosting.

### **40. What are the Various Kernels that are present in SVM?**

The various kernels that are present in SVM are:

- Linear
- Polynomial
- Radial Basis
- Sigmoid

### **41. Suppose you found that your model is suffering from high variance. Which algorithm do you think could handle this situation and why?**

Handling High Variance

- For handling issues of high variance, we should use the bagging algorithm.
- The bagging algorithm would split data into subgroups with a replicated sampling of random data.
- Once the algorithm splits the data, we can use random data to create rules using a particular training algorithm.
- After that, we can use polling for combining the predictions of the model.

## 42. What is Rescaling of Data and how is it done?

In real-world scenarios, the attributes present in data are in a varying pattern. So, rescaling the characteristics to a common scale is beneficial for algorithms to process data efficiently.

We can rescale data using Scikit-learn. The code for rescaling the data using MinMaxScaler is as follows:

```
#Rescaling data
import pandas
import scipy
import numpy
from sklearn.preprocessing import MinMaxScaler
names = ['Abhi', 'Piyush', 'Pranay', 'Sourav', 'Sid', 'Mike', 'pedi', 'Jack', 'Tim']
Dataframe = pandas.read_csv(url, names=names)
Array = dataframe.values
# Splitting the array into input and output
X = array[:,0:8]
Y = array[:,8]
Scaler = MinMaxScaler(feature_range=(0, 1))
rescaledX = scaler.fit_transform(X)
# Summarizing the modified data
numpy.set_printoptions(precision=3)
print(rescaledX[0:5,:])
```

Apart from the theoretical concepts, some interviewers also focus on the implementation of Machine Learning topics. The following Interview Questions are related to the implementation of theoretical concepts.

### **43. What is Binarizing of Data? How to Binarize?**

Converting data into binary values on the basis of threshold values is known as binarizing of data. The values that are less than the threshold are set to 0 and the values that are greater than the threshold are set to 1. This process is useful when feature engineering has to be performed. This can also be used for adding unique features. Data can be binarized using Scikit-learn. The code for binarizing data using Binarizer is as follows:

```
from sklearn.preprocessing import Binarizer
import pandas
import numpy
names = ['Abhi', 'Piyush', 'Pranay', 'Sourav', 'Sid', 'Mike', 'pedi', 'Jack', 'Tim']
dataframe = pandas.read_csv(url, names=names)
array = dataframe.values
# Splitting the array into input and output
X = array[:,0:8]
Y = array[:,8]
binarizer = Binarizer(threshold=0.0).fit(X)
binaryX = binarizer.transform(X)
# Summarizing the modified data
numpy.set_printoptions(precision=3)
print(binaryX[0:5,:])
```

### **44. How to Standardize Data?**

Standardization is the method that is used for rescaling data attributes. The attributes are likely to have a mean value of 0 and a value of the standard deviation of 1. The main objective of standardization is to prompt the mean and standard deviation for the attributes.

Data can be standardized using Scikit-learn. The code for standardizing the data using StandardScaler is as follows:

```
# Python code to Standardize data (0 mean, 1 stdev)
from sklearn.preprocessing import StandardScaler
import pandas
import numpy
names = ['Abhi', 'Piyush', 'Pranay', 'Sourav', 'Sid', 'Mike', 'pedi', 'Jack', 'Tim']
dataframe = pandas.read_csv(url, names=names)
array = dataframe.values
# Separate the array into input and output components
X = array[:,0:8]
Y = array[:,8]
scaler = StandardScaler().fit(X)
rescaledX = scaler.transform(X)
# Summarize the transformed data
numpy.set_printoptions(precision=3)
print(rescaledX[0:5,:])
```

#### **45. We know that one-hot encoding increases the dimensionality of a dataset, but label encoding doesn't. How?**

When one-hot encoding is used, there is an increase in the dimensionality of a dataset. The reason for the increase in dimensionality is that every class in categorical variables, forms a different variable.

Example: Suppose there is a variable “Color.” It has three sublevels, “Yellow,” “Purple,” and “Orange.” So, one-hot encoding “Color” will create three different variables as Color.Yellow, Color.Purple, and Color.Orange.

In label encoding, the subclasses of a certain variable get the value 0 and 1. So, label encoding is only used for binary variables.

This is why one-hot encoding increases the dimensionality of data and label encoding does not.

*Now, if you are interested in doing an end-to-end certification course in Machine Learning, you can check out Intellipaat's Machine Learning Course with Python.*

**46. Executing a binary classification tree algorithm is a simple task. But how does tree splitting take place? How does the tree determine which variable to break at the root node and which at its child nodes?**

Gini index and Node Entropy assist the binary classification tree to make decisions. Basically, the tree algorithm determines the feasible feature that is used to distribute data into the most genuine child nodes.

According to the Gini index, if we arbitrarily pick a pair of objects from a group, then they should be of identical class and the probability for this event should be 1.

The following are the steps to compute the Gini index:

1. Compute Gini for sub-nodes with the formula: The sum of the square of probability for success and failure ( $p^2 + q^2$ )
2. Compute Gini for split by weighted Gini rate of every node of the split

Now, Entropy is the degree of indecency that is given by the following:

Where  $a$  and  $b$  are the probabilities of success and failure of the node

When Entropy = 0, the node is homogenous

When Entropy is high, both groups are present at 50–50 percent in the node.

Finally, to determine the suitability of the node as a root node, the entropy should be very low.

**47. Imagine you are given a dataset consisting of variables having more than 30% missing values. Let's say, out of 50 variables, 16 variables have missing values, which is higher than 30%. How will you deal with them?**

To deal with the missing values, we will do the following:

- We will specify a different class for the missing values.
- Now, we will check the distribution of values, and we will hold those missing values that are defining a pattern.
- Then, we will charge these values into yet another class while eliminating others.

**48. Explain False Negative, False Positive, True Negative, and True Positive with a simple example.**

**True Positive (TP):** When the Machine Learning model correctly predicts the condition, it is said to have a True Positive value.

**True Negative (TN):** When the Machine Learning model correctly predicts the negative condition or class, then it is said to have a True Negative value.

**False Positive (FP):** When the Machine Learning model incorrectly predicts a negative class or condition, then it is said to have a False Positive value.

**False Negative (FN):** When the Machine Learning model incorrectly predicts a positive class or condition, then it is said to have a False Negative value.

**49. What is F1-score and How Is It Used?**

F-score or F1-score is a measure of overall accuracy of a binary classification model. Before understanding F1-score, it is crucial to understand two more measures of accuracy, i.e., precision and recall.

Precision is defined as the percentage of True Positives to the total number of positive classifications predicted by the model. In other words,

Precision = (No. of True Positives / No. True Positives + No. of False Positives)

Recall is defined as the percentage of True Positives to the total number of actual positive labeled data passed to the model. In other words,

Precision = (No. of True Positives / No. True Positives + No. of False Negatives)

Both precision and recall are partial measures of accuracy of a model. F1-score combines precision and recall and provides an overall score to measure a model's accuracy.

F1-score =  $2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$

This is why, F1-score is the most popular measure of accuracy in any Machine-Learning-based binary classification model.

## 50. How to Implement the KNN Classification Algorithm?

Iris dataset is used for implementing the KNN classification algorithm.

```
# KNN classification algorithm
from sklearn.datasets import load_iris
from sklearn.neighbors import KNeighborsClassifier
import numpy as np
from sklearn.model_selection import train_test_split
iris_dataset=load_iris()
A_train, A_test, B_train, B_test = train_test_split(iris_dataset["data"],
iris_dataset["target"], random_state=0)
kn = KNeighborsClassifier(n_neighbors=1)
kn.fit(A_train, B_train)
A_new = np.array([[8, 2.5, 1, 1.2]])
prediction = kn.predict(A_new)
print("Predicted target value: {} \n".format(prediction))
print("Predicted feature name: {} \n".format
(iris_dataset["target_names"][prediction]))
print("Test score: {:.2f} ".format(kn.score(A_test, B_test)))
```

Output:

Predicted Target Name: [0]

Predicted Feature Name: [' Setosa']

Test Score: 0.92

## 1. Explain the terms Artificial Intelligence (AI), Machine Learning (ML) and Deep Learning?

Artificial Intelligence (AI) is the domain of producing intelligent machines. ML refers to systems that can assimilate from experience (training data) and Deep Learning (DL) states to systems that learn from experience on large data sets. ML can be considered as a subset of AI. Deep Learning (DL) is ML but useful to large data sets. The figure below roughly encapsulates the relation between AI, ML, and DL:

In summary, DL is a subset of ML & both were the subsets of AI.

Additional Information: ASR (Automatic Speech Recognition) & NLP (Natural Language Processing) fall under AI and overlay with ML & DL as ML is often utilized for NLP and ASR tasks.

## 2. What are the different types of Learning/ Training models in ML?

ML algorithms can be primarily classified depending on the presence/absence of target variables.

### A. Supervised learning: [Target is present]

The machine learns using labelled data. The model is trained on an existing data set before it starts making decisions with the new data.

*The target variable is continuous:* Linear Regression, polynomial Regression, and quadratic Regression.

*The target variable is categorical:* Logistic regression, Naive Bayes, KNN, SVM, Decision Tree, Gradient Boosting, ADA boosting, Bagging, Random forest etc.

### B. Unsupervised learning: [Target is absent]

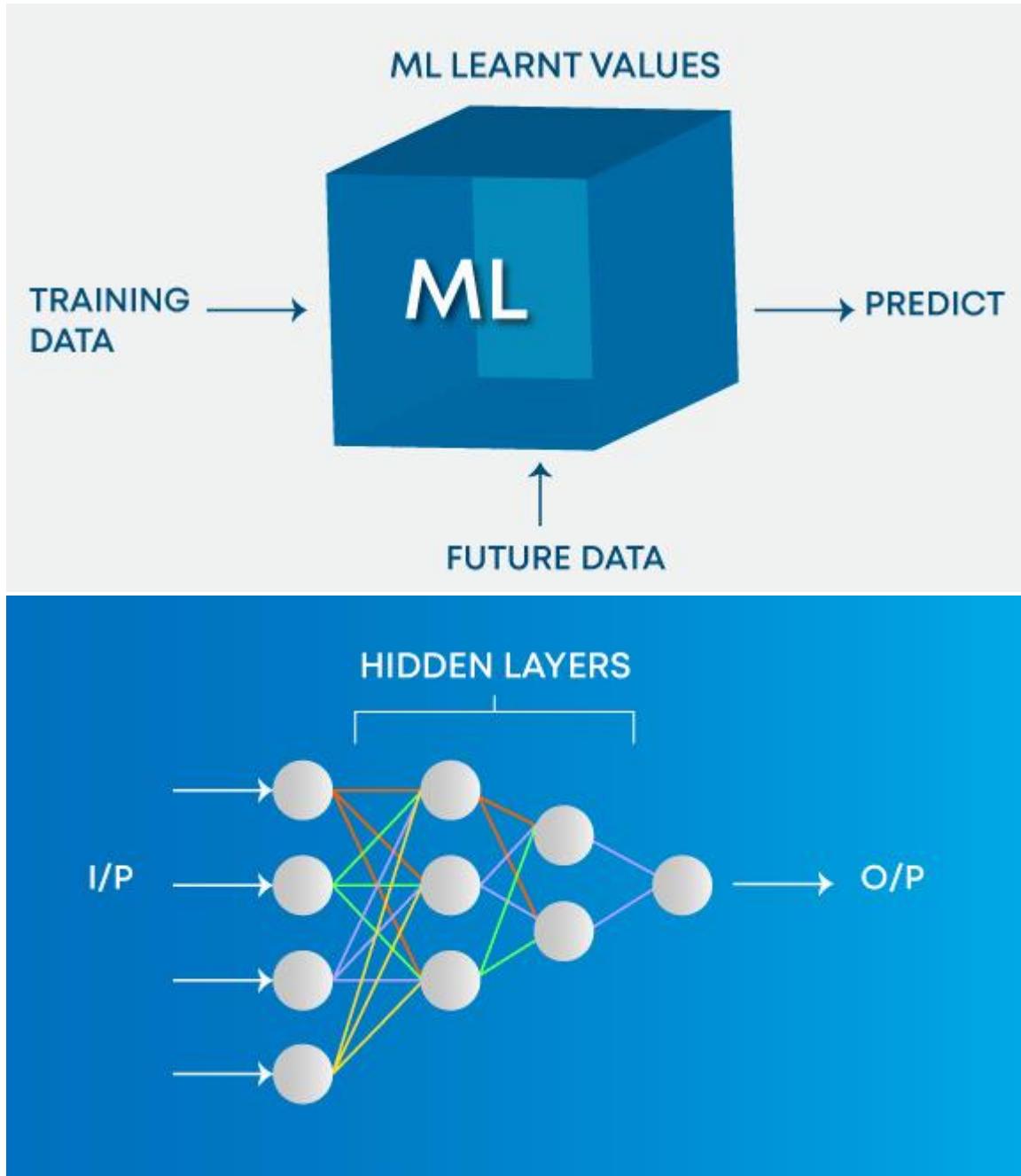
The machine is trained on unlabelled data and without any proper guidance. It automatically infers patterns and relationships in the data by creating clusters. The model learns through observations and deduced structures in the data.

Principal component Analysis, Factor analysis, Singular Value Decomposition etc.

### C. Reinforcement Learning:

The model learns through a trial and error method. This kind of learning involves an agent that will interact with the environment to create actions and then discover errors or rewards of that action.

### 3. What is the difference between deep learning and machine learning?



Machine Learning involves algorithms that learn from patterns of data and then apply it to decision making. Deep Learning, on the other hand, is able to learn through processing data on its own and is quite similar to the human brain

where it identifies something, analyse it, and makes a decision.

The key differences are as follows:

- The manner in which data is presented to the system.
- Machine learning algorithms always require structured data and deep learning networks rely on layers of artificial neural networks.

#### **4. What is the main key difference between supervised and unsupervised machine learning?**

##### **Supervised learning**

The supervised learning technique needs labelled data to train the model. For example, to solve a classification problem (a supervised learning task), you need to have label data to train the model and to classify the data into your labelled groups.

##### **Unsupervised learning**

Unsupervised learning does not need any labelled dataset. This is the main key difference between supervised learning and unsupervised learning.

Learn Machine Learning

#### **5. How do you select important variables while working on a data set?**

There are various means to select important variables from a data set that include the following:

- Identify and discard correlated variables before finalizing on important variables
- The variables could be selected based on ‘p’ values from Linear Regression
- Forward, Backward, and Stepwise selection
- Lasso Regression
- Random Forest and plot variable chart
- Top features can be selected based on information gain for the available set of features.

#### **6. There are many machine learning algorithms till now. If given a data set, how can one determine which algorithm to be used for that?**

Machine Learning algorithm to be used purely depends on the type of data in a given dataset. If data is linear then, we use linear regression. If data shows non-linearity then, the bagging algorithm would do better. If the data is to be analyzed/interpreted for some business purposes then we can use decision

trees or SVM. If the dataset consists of images, videos, audios then, neural networks would be helpful to get the solution accurately.

So, there is no certain metric to decide which algorithm to be used for a given situation or a data set. We need to explore the data using EDA (Exploratory Data Analysis) and understand the purpose of using the dataset to come up with the best fit algorithm. So, it is important to study all the algorithms in detail.

## 7. How are covariance and correlation different from one another?

### Covariance

### Correlation

Covariance measures how two variables are related to each other and how one would vary with respect to changes in the other variable. If the value is positive it means there is a direct relationship between the variables and one would increase or decrease with an increase or decrease in the base variable respectively, given that all other conditions remain constant.

Correlation quantifies the relationship between two random variables and has only three specific values, i.e., 1, 0, and -1.

1 denotes a positive relationship, -1 denotes a negative relationship, and 0 denotes that the two variables are independent of each other.

## 8. State the differences between causality and correlation?

Causality applies to situations where one action, say X, causes an outcome, say Y, whereas Correlation is just relating one action (X) to another action(Y) but X does not necessarily cause Y.

## 9. We look at machine learning software almost all the time. How do we apply Machine Learning to Hardware?

We have to build ML algorithms in System Verilog which is a Hardware development Language and then program it onto an FPGA to apply Machine Learning to hardware.

## 10. Explain One-hot encoding and Label Encoding. How do they affect the dimensionality of the given dataset?

One-hot encoding is the representation of categorical variables as binary vectors. Label Encoding is converting labels/words into numeric form. Using one-hot encoding increases the dimensionality of the data set. Label encoding doesn't affect the dimensionality of the data set. One-hot encoding creates a new variable for each level in the variable whereas, in Label encoding, the levels of a variable get encoded as 1 and 0.



**LABEL ENCODING**

Food Name	Categorical#	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

▼

**ONE HOT ENCODING**

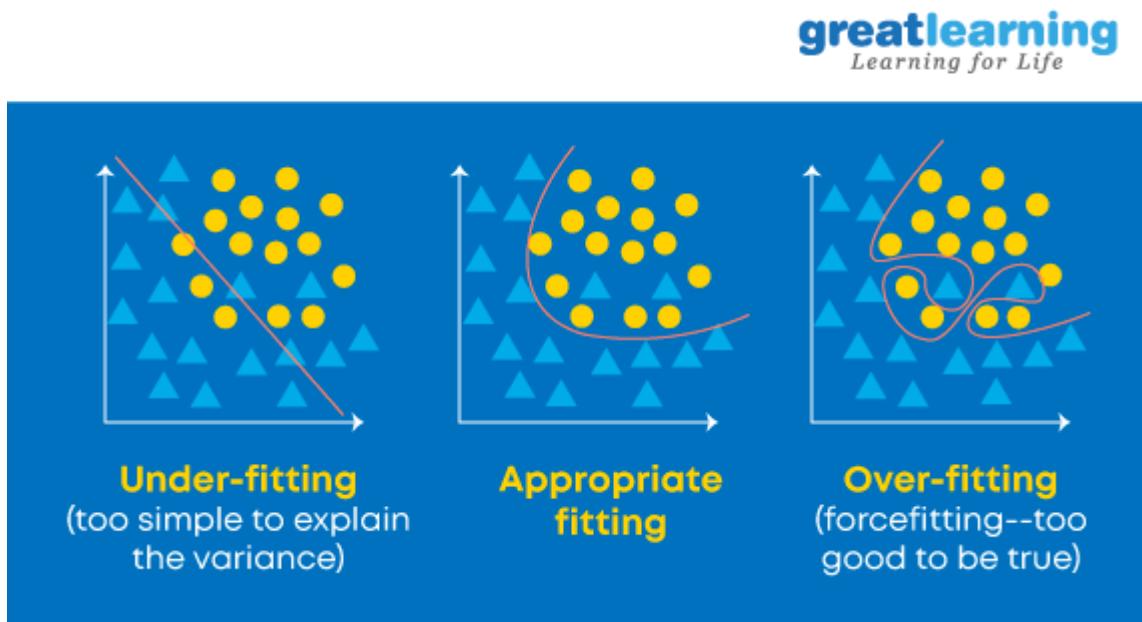
Apple	Chicken	Broccoli	Calories
1	0	0	95
0	1	0	231
0	0	1	50

## Deep Learning Interview Questions

Deep Learning is a part of machine learning that works with neural networks. It involves a hierarchical structure of networks that set up a process to help machines learn the human logic behind any action. We have compiled a list of the frequently asked deep learning interview questions to help you prepare.

## 11. When does regularization come into play in Machine Learning?

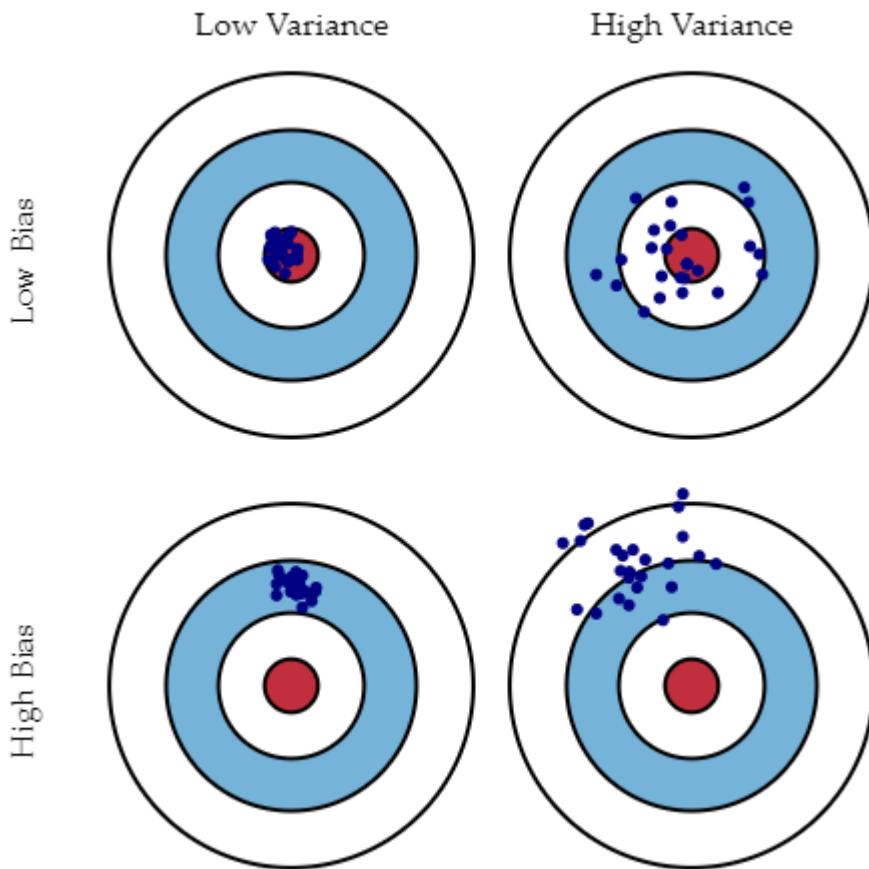
At times when the model begins to underfit or overfit, regularization becomes necessary. It is a regression that diverts or regularizes the coefficient estimates towards zero. It reduces flexibility and discourages learning in a model to avoid the risk of overfitting. The model complexity is reduced and it becomes better at predicting.



## 12. What is Bias, Variance and what do you mean by Bias-Variance Tradeoff?

Both are errors in Machine Learning Algorithms. When the algorithm has limited flexibility to deduce the correct observation from the dataset, it results in bias. On the other hand, variance occurs when the model is extremely sensitive to small fluctuations.

If one adds more features while building a model, it will add more complexity and we will lose bias but gain some variance. In order to maintain the optimal amount of error, we perform a tradeoff between bias and variance based on the needs of a business.



Source:

Understanding the Bias-Variance Tradeoff: Scott Fortmann – Roe

Bias stands for the error because of the erroneous or overly simplistic assumptions in the learning algorithm . This assumption can lead to the model underfitting the data, making it hard for it to have high predictive accuracy and for you to generalize your knowledge from the training set to the test set.

Variance is also an error because of too much complexity in the learning algorithm. This can be the reason for the algorithm being highly sensitive to high degrees of variation in training data, which can lead your model to overfit the data. Carrying too much noise from the training data for your model to be very useful for your test data.

The bias-variance decomposition essentially decomposes the learning error from any algorithm by adding the bias, the variance and a bit of irreducible error due to noise in the underlying dataset. Essentially, if you make the model more complex and add more variables, you'll lose bias but gain some variance — in order to get the optimally reduced amount of error, you'll have to trade off bias and variance. You don't want either high bias or high variance in your model.

### 13. How can we relate standard deviation and variance?

*Standard deviation* refers to the spread of your data from the mean. *Variance* is the average degree to which each point differs from the mean i.e. the average of all data points. We can relate Standard deviation and Variance because it is the square root of Variance.

**14. A data set is given to you and it has missing values which spread along 1 standard deviation from the mean. How much of the data would remain untouched?**

It is given that the data is spread across mean that is the data is spread across an average. So, we can presume that it is a normal distribution. In a normal distribution, about 68% of data lies in 1 standard deviation from averages like mean, mode or median. That means about 32% of the data remains uninfluenced by missing values.

**15. Is a high variance in data good or bad?**

Higher variance directly means that the data spread is big and the feature has a variety of data. Usually, high variance in a feature is seen as not so good quality.

**16. If your dataset is suffering from high variance, how would you handle it?**

For datasets with high variance, we could use the bagging algorithm to handle it. Bagging algorithm splits the data into subgroups with sampling replicated from random data. After the data is split, random data is used to create rules using a training algorithm. Then we use polling technique to combine all the predicted outcomes of the model.

**17. A data set is given to you about utilities fraud detection. You have built a classifier model and achieved a performance score of 98.5%. Is this a good model? If yes, justify. If not, what can you do about it?**

Data set about utilities fraud detection is not balanced enough i.e. imbalanced. In such a data set, accuracy score cannot be the measure of performance as it may only be predict the majority class label correctly but in this case our point of interest is to predict the minority label. But often minorities are treated as noise and ignored. So, there is a high probability of misclassification of the minority label as compared to the majority label. For evaluating the model performance in case of imbalanced data sets, we should use Sensitivity (True Positive rate) or Specificity (True Negative rate) to determine class label wise

performance of the classification model. If the minority class label's performance is not so good, we could do the following:

- We can use under sampling or over sampling to balance the data.
- We can change the prediction threshold value.
- We can assign weights to labels such that the minority class labels get larger weights.
- We could detect anomalies.

## **18. Explain the handling of missing or corrupted values in the given dataset.**

An easy way to handle missing values or corrupted values is to drop the corresponding rows or columns. If there are too many rows or columns to drop then we consider replacing the missing or corrupted values with some new value.

Identifying missing values and dropping the rows or columns can be done by using IsNull() and dropna( ) functions in Pandas. Also, the Fillna() function in Pandas replaces the incorrect values with the placeholder value.

## **19. What is Time series?**

A Time series is a sequence of numerical data points in successive order. It tracks the movement of the chosen data points, over a specified period of time and records the data points at regular intervals. Time series doesn't require any minimum or maximum time input. Analysts often use Time series to examine data according to their specific requirement.

## **20. What is a Box-Cox transformation?**

Box-Cox transformation is a power transform which transforms non-normal dependent variables into normal variables as normality is the most common assumption made while using many statistical techniques. It has a lambda parameter which when set to 0 implies that this transform is equivalent to log-transform. It is used for variance stabilization and also to normalize the distribution.

## **21. What is the difference between stochastic gradient descent (SGD) and gradient descent (GD)?**

Gradient Descent and Stochastic Gradient Descent are the algorithms that find the set of parameters that will minimize a loss function.

The difference is that in Gradient Descend, all training samples are evaluated for each set of parameters. While in Stochastic Gradient Descent only one training sample is evaluated for the set of parameters identified.

## **22. What is the exploding gradient problem while using the back propagation technique?**

When large error gradients accumulate and result in large changes in the neural network weights during training, it is called the exploding gradient problem. The values of weights can become so large as to overflow and result in NaN values. This makes the model unstable and the learning of the model to stall just like the vanishing gradient problem. This is one of the most commonly asked interview questions on machine learning.

## **23. Can you mention some advantages and disadvantages of decision trees?**

The advantages of decision trees are that they are easier to interpret, are nonparametric and hence robust to outliers, and have relatively few parameters to tune.

On the other hand, the disadvantage is that they are prone to overfitting.

## **24. Explain the differences between Random Forest and Gradient Boosting machines.**

### **Random Forests**

Random forests are a significant number of decision trees pooled using averages or majority rules at the end.

The random forest creates each tree independent of the others while gradient boosting develops one tree at a time.

### **Gradient Boosting**

Gradient boosting machines also combine decision trees but at the beginning of the process, unlike Random forests.

Gradient boosting yields better outcomes than random forests if parameters are carefully tuned but it's not a good option if the data set contains a lot of outliers/anomalies/noise as it can result in overfitting of the model.

Random forests perform well for multiclass object detection.

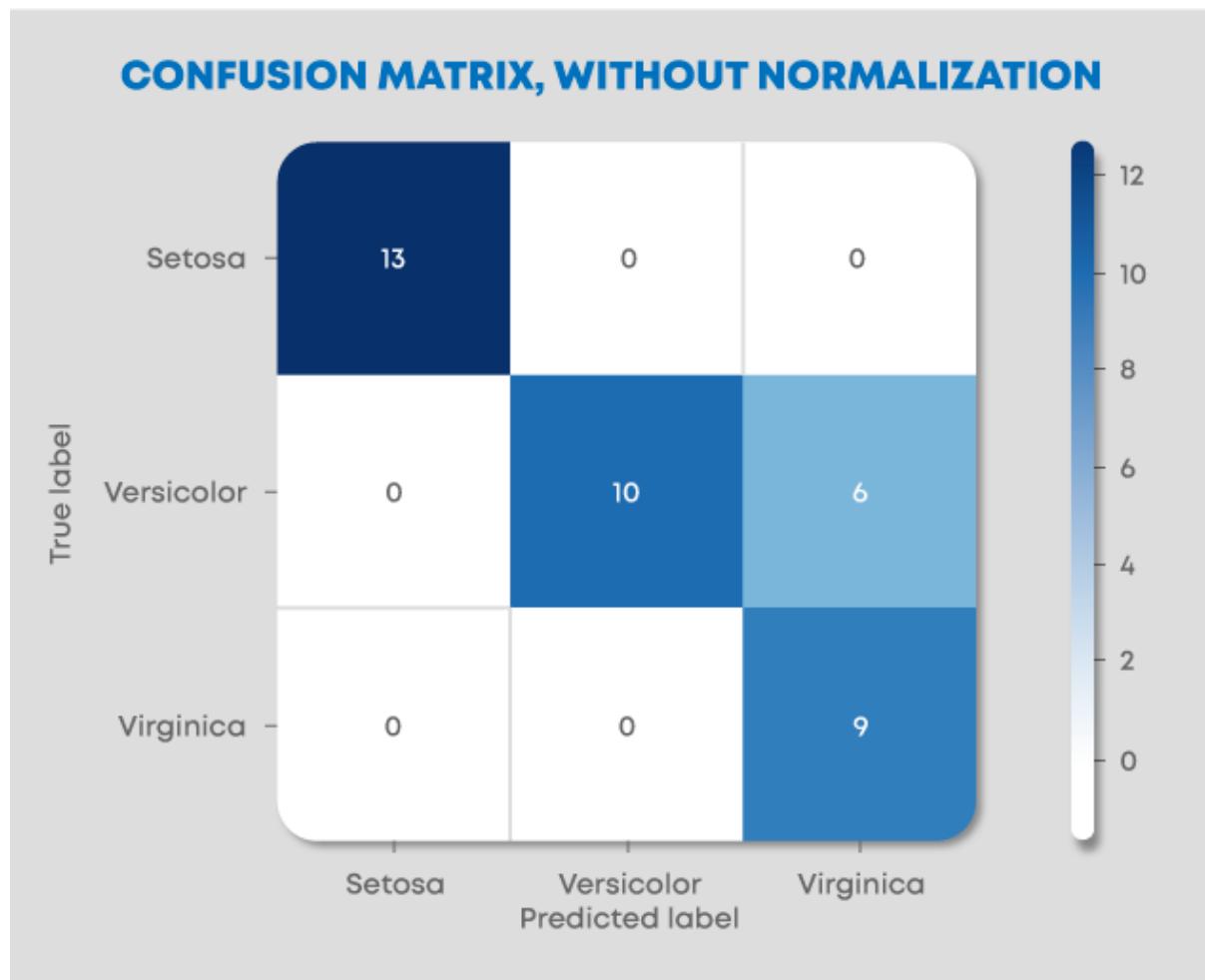
Gradient Boosting performs well when there is data which is not balanced such as in real-time risk assessment.

## 25. What is a confusion matrix and why do you need it?

Confusion matrix (also called the error matrix) is a table that is frequently used to illustrate the performance of a classification model i.e. classifier on a set of test data for which the true values are well-known.

It allows us to visualize the performance of an algorithm/model. It allows us to easily identify the confusion between different classes. It is used as a performance measure of a model/algorith.

A confusion matrix is known as a summary of predictions on a classification model. The number of right and wrong predictions were summarized with count values and broken down by each class label. It gives us information about the errors made through the classifier and also the types of errors made by a classifier.



Build the Best Machine Learning Resume and Stand out from the crowd

Resume Building Make Linkedin Profile

## 26. What's a Fourier transform?

Fourier Transform is a mathematical technique that transforms any function of time to a function of frequency. Fourier transform is closely related to Fourier series. It takes any time-based pattern for input and calculates the overall cycle offset, rotation speed and strength for all possible cycles. Fourier transform is best applied to waveforms since it has functions of time and space. Once a Fourier transform applied on a waveform, it gets decomposed into a sinusoid.

## 27. What do you mean by Associative Rule Mining (ARM)?

Associative Rule Mining is one of the techniques to discover patterns in data like features (dimensions) which occur together and features (dimensions)

which are correlated. It is mostly used in Market-based Analysis to find how frequently an itemset occurs in a transaction. Association rules have to satisfy minimum support and minimum confidence at the very same time. Association rule generation generally comprised of two different steps:

- “A min support threshold is given to obtain all frequent item-sets in a database.”
- “A min confidence constraint is given to these frequent item-sets in order to form the association rules.”

Support is a measure of how often the “item set” appears in the data set and Confidence is a measure of how often a particular rule has been found to be true.

## **28. What is Marginalisation? Explain the process.**

Marginalisation is summing the probability of a random variable X given joint probability distribution of X with other variables. It is an application of the law of total probability.

$$P(X=x) = \sum_Y P(X=x, Y)$$

Given the joint probability  $P(X=x, Y)$ , we can use marginalization to find  $P(X=x)$ . So, it is to find distribution of one random variable by exhausting cases on other random variables.

## **29. Explain the phrase “Curse of Dimensionality”.**

The Curse of Dimensionality refers to the situation when your data has too many features.

The phrase is used to express the difficulty of using brute force or grid search to optimize a function with too many inputs.

It can also refer to several other issues like:

- If we have more features than observations, we have a risk of overfitting the model.
- When we have too many features, observations become harder to cluster. Too many dimensions cause every observation in the dataset to appear equidistant from all others and no meaningful clusters can be formed.

Dimensionality reduction techniques like PCA come to the rescue in such cases.

### **30. What is the Principle Component Analysis?**

The idea here is to reduce the dimensionality of the data set by reducing the number of variables that are correlated with each other. Although the variation needs to be retained to the maximum extent.

The variables are transformed into a new set of variables that are known as ‘Principal Components’. These PCs are the eigenvectors of a covariance matrix and therefore are orthogonal.

### **31. Why is rotation of components so important in Principle Component Analysis (PCA)?**

Rotation in PCA is very important as it maximizes the separation within the variance obtained by all the components because of which interpretation of components would become easier. If the components are not rotated, then we need extended components to describe variance of the components.

### **32. What are outliers? Mention three methods to deal with outliers.**

A data point that is considerably distant from the other similar data points is known as an outlier. They may occur due to experimental errors or variability in measurement. They are problematic and can mislead a training process, which eventually results in longer training time, inaccurate models, and poor results.

The three methods to deal with outliers are:

**Univariate method** – looks for data points having extreme values on a single variable

**Multivariate method** – looks for unusual combinations on all the variables

**Minkowski error** – reduces the contribution of potential outliers in the training process

### **33. What is the difference between regularization and normalisation?**

#### **Normalisation**

Normalisation adjusts the data; . If your data is on very different scales (especially low to high), you

#### **Regularisation**

Regularisation adjusts the prediction function. Regularization

would want to normalise the data. Alter each column to have compatible basic statistics. This can be helpful to make sure there is no loss of accuracy. One of the goals of model training is to identify the signal and ignore the noise if the model is given free rein to minimize error, there is a possibility of suffering from overfitting.

imposes some control on this by providing simpler fitting functions over complex ones.

### **34. Explain the difference between Normalization and Standardization.**

Normalization and Standardization are the two very popular methods used for feature scaling.

#### **Normalisation**

Normalization refers to re-scaling the values to fit into a range of [0,1].

Normalization is useful when all parameters need to have an identical positive scale however the outliers from the data set are lost.

#### **Standardization**

Standardization refers to re-scaling data to have a mean of 0 and a standard deviation of 1 (Unit variance)

### **35. List the most popular distribution curves along with scenarios where you will use them in an algorithm.**

The most popular distribution curves are as follows- Bernoulli Distribution, Uniform Distribution, Binomial Distribution, Normal Distribution, Poisson Distribution, and Exponential Distribution. Check out the free Probability for Machine Learning course to enhance your knowledge on Probability Distributions for Machine Learning.

Each of these distribution curves is used in various scenarios.

Bernoulli Distribution can be used to check if a team will win a championship or not, a newborn child is either male or female, you either pass an exam or not, etc.

***Uniform distribution*** is a probability distribution that has a constant probability. Rolling a single dice is one example because it has a fixed number of outcomes.

**Binomial distribution** is a probability with only two possible outcomes, the prefix ‘bi’ means two or twice. An example of this would be a coin toss. The outcome will either be heads or tails.

**Normal distribution** describes how the values of a variable are distributed. It is typically a symmetric distribution where most of the observations cluster around the central peak. The values further away from the mean taper off equally in both directions. An example would be the height of students in a classroom.

**Poisson distribution** helps predict the probability of certain events happening when you know how often that event has occurred. It can be used by businessmen to make forecasts about the number of customers on certain days and allows them to adjust supply according to the demand.

**Exponential distribution** is concerned with the amount of time until a specific event occurs. For example, how long a car battery would last, in months.

### **36. How do we check the normality of a data set or a feature?**

Visually, we can check it using plots. There is a list of Normality checks, they are as follow:

- Shapiro-Wilk W Test
- Anderson-Darling Test
- Martinez-Iglewicz Test
- Kolmogorov-Smirnov Test
- D'Agostino Skewness Test

### **37. What is Linear Regression?**

Linear Function can be defined as a Mathematical function on a 2D plane as,  $Y = Mx + C$ , where Y is a dependent variable and X is Independent Variable, C is Intercept and M is slope and same can be expressed as Y is a Function of X or  $Y = F(x)$ .

At any given value of X, one can compute the value of Y, using the equation of Line. This relation between Y and X, with a degree of the polynomial as 1 is called Linear Regression.

In Predictive Modeling, LR is represented as  $Y = B_0 + B_1x_1 + B_2x_2$   
The value of B1 and B2 determines the strength of the correlation between features and the dependent variable.

Example: Stock Value in \$ = Intercept + (+/-B1)\*(Opening value of Stock) + (+/-B2)\*(Previous Day Highest value of Stock)

### **38. Differentiate between regression and classification.**

Regression and classification are categorized under the same umbrella of supervised machine learning. The main difference between them is that the output variable in the regression is numerical (or continuous) while that for classification is categorical (or discrete).

Example: To predict the definite Temperature of a place is Regression problem whereas predicting whether the day will be Sunny cloudy or there will be rain is a case of classification.

### **39. What is target imbalance? How do we fix it? A scenario where you have performed target imbalance on data. Which metrics and algorithms do you find suitable to input this data onto?**

If you have categorical variables as the target when you cluster them together or perform a frequency count on them if there are certain categories which are more in number as compared to others by a very significant number. This is known as the target imbalance.

Example: Target column – 0,0,0,1,0,2,0,0,1,1 [0s: 60%, 1: 30%, 2:10%] 0 are in majority. To fix this, we can perform up-sampling or down-sampling. Before fixing this problem let's assume that the performance metrics used was confusion metrics. After fixing this problem we can shift the metric system to AUC: ROC. Since we added/deleted data [up sampling or downsampling], we can go ahead with a stricter algorithm like SVM, Gradient boosting or ADA boosting.

### **40. List all assumptions for data to be met before starting with linear regression.**

Before starting linear regression, the assumptions to be met are as follow:

- Linear relationship
- Multivariate normality
- No or little multicollinearity
- No auto-correlation
- Homoscedasticity

**41. When does the linear regression line stop rotating or finds an optimal spot where it is fitted on data?**

A place where the highest RSquared value is found, is the place where the line comes to rest. RSquared represents the amount of variance captured by the virtual linear regression line with respect to the total variance captured by the dataset.

**42. Why is logistic regression a type of classification technique and not a regression? Name the function it is derived from?**

Since the target column is categorical, it uses linear regression to create an odd function that is wrapped with a log function to use regression as a classifier. Hence, it is a type of classification technique and not a regression. It is derived from cost function.

**43. What could be the issue when the beta value for a certain variable varies way too much in each subset when regression is run on different subsets of the given dataset?**

Variations in the beta values in every subset implies that the dataset is heterogeneous. To overcome this problem, we can use a different model for each of the dataset's clustered subsets or a non-parametric model such as decision trees.

**44. What does the term Variance Inflation Factor mean?**

Variation Inflation Factor (VIF) is the ratio of the model's variance to the model's variance with only one independent variable. VIF gives the estimate of the volume of multicollinearity in a set of many regression variables.

VIF = Variance of the model with one independent variable

**45. Which machine learning algorithm is known as the lazy learner, and why is it called so?**

KNN is a Machine Learning algorithm known as a lazy learner. K-NN is a lazy learner because it doesn't learn any machine-learned values or variables from the training data but dynamically calculates distance every time it wants to classify, hence memorizing the training dataset instead.

We know what the companies are looking for, and with that in mind, we have prepared the set of Machine Learning interview questions an experienced professional may be asked. So, prepare accordingly if you wish to ace the interview in one go.

#### 46. Is it possible to use KNN for image processing?



Yes, it is possible to use KNN for image processing. It can be done by converting the 3-dimensional image into a single-dimensional vector and using the same as input to KNN.

#### 47. Differentiate between K-Means and KNN algorithms?

##### KNN algorithms

KNN algorithms is Supervised Learning where-as K-Means is Unsupervised Learning. With KNN, we predict the label of the unidentified element based on its nearest neighbour and further extend this approach for solving classification/regression-based problems.

##### K-Means

K-Means is Unsupervised Learning, where we don't have any Labels present, in other words, no Target Variables and thus we try to cluster the data based upon their coord

#### NLP Interview Questions

NLP or Natural Language Processing helps machines analyse natural languages with the intention of learning them. It extracts information from data by applying machine learning algorithms. Apart from learning the basics of NLP, it is important to prepare specifically for the interviews. Check out the top NLP Interview Questions

#### 48. How does the SVM algorithm deal with self-learning?

SVM has a learning rate and expansion rate which takes care of this. The learning rate compensates or penalises the hyperplanes for making all the wrong moves and expansion rate deals with finding the maximum separation area between classes.

#### **49. What are Kernels in SVM? List popular kernels used in SVM along with a scenario of their applications.**

The function of the kernel is to take data as input and transform it into the required form. A few popular Kernels used in SVM are as follows: RBF, Linear, Sigmoid, Polynomial, Hyperbolic, Laplace, etc.

#### **50. What is Kernel Trick in an SVM Algorithm?**

Kernel Trick is a mathematical function which when applied on data points, can find the region of classification between two different classes. Based on the choice of function, be it linear or radial, which purely depends upon the distribution of data, one can build a classifier.

#### **51. What are ensemble models? Explain how ensemble techniques yield better learning as compared to traditional classification ML algorithms.**

An ensemble is a group of models that are used together for prediction both in classification and regression classes. Ensemble learning helps improve ML results because it combines several models. By doing so, it allows for a better predictive performance compared to a single model.

They are superior to individual models as they reduce variance, average out biases, and have lesser chances of overfitting.

#### **52. What are overfitting and underfitting? Why does the decision tree algorithm suffer often with overfitting problems?**

Overfitting is a statistical model or machine learning algorithm that captures the data's noise. Underfitting is a model or machine learning algorithm which does not fit the data well enough and occurs if the model or algorithm shows low variance but high bias.

In decision trees, overfitting occurs when the tree is designed to fit all samples in the training data set perfectly. This results in branches with strict rules or sparse data and affects the accuracy when predicting samples that aren't part of the training set.

### **53. What is OOB error and how does it occur?**

For each bootstrap sample, there is one-third of the data that was not used in the creation of the tree, i.e., it was out of the sample. This data is referred to as out of bag data. In order to get an unbiased measure of the accuracy of the model over test data, out of bag error is used. The out of bag data is passed for each tree is passed through that tree and the outputs are aggregated to give out of bag error. This percentage error is quite effective in estimating the error in the testing set and does not require further cross-validation.

### **54. Why boosting is a more stable algorithm as compared to other ensemble algorithms?**

Boosting focuses on errors found in previous iterations until they become obsolete. Whereas in bagging there is no corrective loop. This is why boosting is a more stable algorithm compared to other ensemble algorithms.

### **55. How do you handle outliers in the data?**

Outlier is an observation in the data set that is far away from other observations in the data set. We can discover outliers using tools and functions like box plot, scatter plot, Z-Score, IQR score etc. and then handle them based on the visualization we have got. To handle outliers, we can cap at some threshold, use transformations to reduce skewness of the data and remove outliers if they are anomalies or errors.

### **56. List popular cross validation techniques.**

There are mainly six types of cross validation techniques. They are as follow:

- K fold
- Stratified k fold
- Leave one out
- Bootstrapping
- Random search cv
- Grid search cv

### **57. Is it possible to test for the probability of improving model accuracy without cross-validation techniques? If yes, please explain.**

Yes, it is possible to test for the probability of improving model accuracy without cross-validation techniques. We can do so by running the ML model for say **n** number of iterations, recording the accuracy. Plot all the accuracies and

remove the 5% of low probability values. Measure the left [low] cut off and right [high] cut off. With the remaining 95% confidence, we can say that the model can go as low or as high [as mentioned within cut off points].

**58. Name a popular dimensionality reduction algorithm.**

Popular dimensionality reduction algorithms are Principal Component Analysis and Factor Analysis.

Principal Component Analysis creates one or more index variables from a larger set of measured variables. Factor Analysis is a model of the measurement of a latent variable. This latent variable cannot be measured with a single variable and is seen through a relationship it causes in a set of y variables.

**59. How can we use a dataset without the target variable into supervised learning algorithms?**

Input the data set into a clustering algorithm, generate optimal clusters, label the cluster numbers as the new target variable. Now, the dataset has independent and target variables present. This ensures that the dataset is ready to be used in supervised learning algorithms.

**60. List all types of popular recommendation systems? Name and explain two personalized recommendation systems along with their ease of implementation.**

Popularity based recommendation, content-based recommendation, user-based collaborative filter, and item-based recommendation are the popular types of recommendation systems.

Personalized Recommendation systems are- Content-based recommendations, user-based collaborative filter, and item-based recommendations. User-based collaborative filter and item-based recommendations are more personalized. Easy to maintain: Similarity matrix can be maintained easily with Item-based recommendations.

**61. How do we deal with sparsity issues in recommendation systems? How do we measure its effectiveness? Explain.**

Singular value decomposition can be used to generate the prediction matrix. RMSE is the measure that helps us understand how close the prediction matrix is to the original matrix.

**62. Name and define techniques used to find similarities in the recommendation system.**

Pearson correlation and Cosine correlation are techniques used to find similarities in recommendation systems.

**63. State the limitations of Fixed Basis Function.**

Linear separability in feature space doesn't imply linear separability in input space. So, Inputs are non-linearly transformed using vectors of basic functions with increased dimensionality. Limitations of Fixed basis functions are:

- Non-Linear transformations cannot remove overlap between two classes but they can increase overlap.
- Often it is not clear which basis functions are the best fit for a given task. So, learning the basic functions can be useful over using fixed basis functions.
- If we want to use only fixed ones, we can use a lot of them and let the model figure out the best fit but that would lead to overfitting the model thereby making it unstable.

**64. Define and explain the concept of Inductive Bias with some examples.**

Inductive Bias is a set of assumptions that humans use to predict outputs given inputs that the learning algorithm has not encountered yet. When we are trying to learn Y from X and the hypothesis space for Y is infinite, we need to reduce the scope by our beliefs/assumptions about the hypothesis space which is also called inductive bias. Through these assumptions, we constrain our hypothesis space and also get the capability to incrementally test and improve on the data using hyper-parameters. Examples:

1. We assume that Y varies linearly with X while applying Linear regression.
2. We assume that there exists a hyperplane separating negative and positive examples.

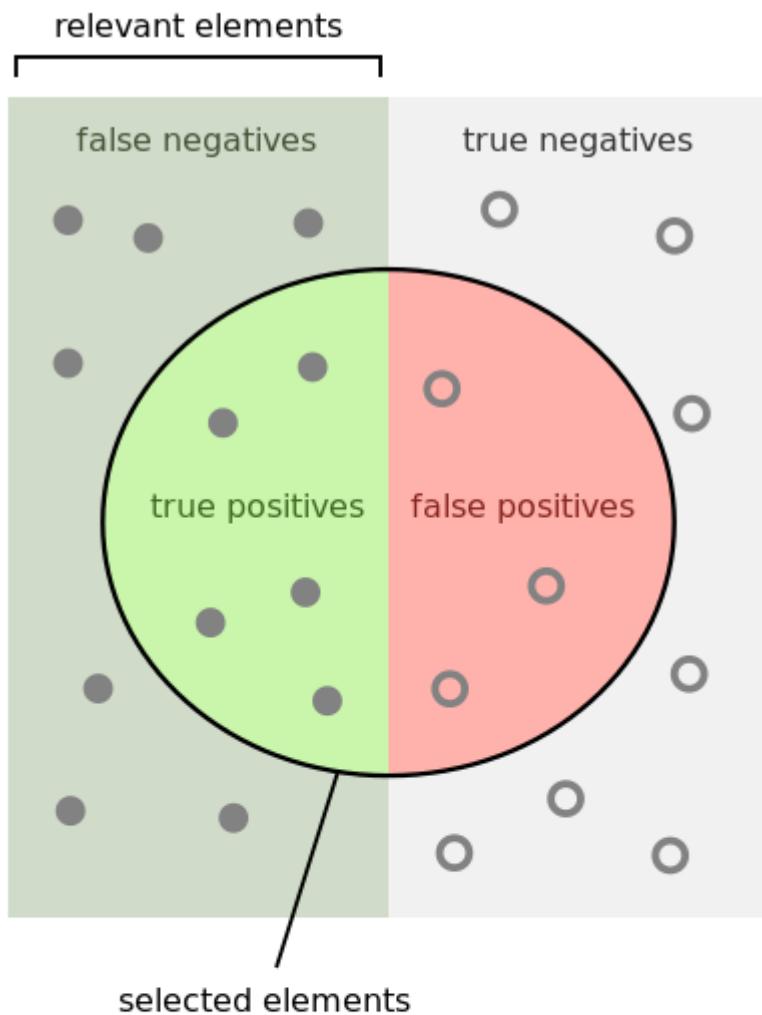
**65. Explain the term instance-based learning.**

Instance Based Learning is a set of procedures for regression and classification which produce a class label prediction based on resemblance to its nearest neighbors in the training data set. These algorithms just collects all the data and get an answer when required or queried. In simple words they are a set of procedures for solving new problems based on the solutions of already solved problems in the past which are similar to the current problem.

**66. Keeping train and test split criteria in mind, is it good to perform scaling before the split or after the split?**

Scaling should be done post-train and test split ideally. If the data is closely packed, then scaling post or pre-split should not make much difference.

**67. Define precision, recall and F1 Score?**



How many selected items are relevant?

$$\text{Precision} = \frac{\text{Selected True Positives}}{\text{Total Selected Items}}$$

How many relevant items are selected?

$$\text{Recall} = \frac{\text{Selected True Positives}}{\text{Total Relevant Items}}$$

The metric used to access the performance of the classification model is Confusion Metric. Confusion Metric can be further interpreted with the following terms:-

**True Positives (TP)** – These are the correctly predicted positive values. It implies that the value of the actual class is yes and the value of the predicted class is also yes.

**True Negatives (TN)** – These are the correctly predicted negative values. It implies that the value of the actual class is no and the value of the predicted class is also no.

**False positives and false negatives**, these values occur when your actual class contradicts with the predicted class.

Now,

**Recall**, also known as Sensitivity is the ratio of true positive rate (TP), to all observations in actual class – yes

$$\text{Recall} = \text{TP}/(\text{TP}+\text{FN})$$

**Precision** is the ratio of positive predictive value, which measures the amount of accurate positives model predicted viz a viz number of positives it claims.

$$\text{Precision} = \text{TP}/(\text{TP}+\text{FP})$$

**Accuracy** is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations.

$$\text{Accuracy} = (\text{TP}+\text{TN})/(\text{TP}+\text{FP}+\text{FN}+\text{TN})$$

**F1 Score** is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. Intuitively it is not as easy to understand as accuracy, but F1 is usually more useful than accuracy, especially if you have an uneven class distribution. Accuracy works best if false positives and false negatives have a similar cost. If the cost of false positives and false negatives are very different, it's better to look at both Precision and Recall.

## **68. Plot validation score and training score with data set size on the x-axis and another plot with model complexity on the x-axis.**

For high bias in the models, the performance of the model on the validation data set is similar to the performance on the training data set. For high variance in the models, the performance of the model on the validation set is worse than the performance on the training set.

## **69. What is Bayes' Theorem? State at least 1 use case with respect to the machine learning context?**

Bayes' Theorem describes the probability of an event, based on prior knowledge of conditions that might be related to the event. For example, if cancer is related to age, then, using Bayes' theorem, a person's age can be used to more accurately assess the probability that they have cancer than can be done without the knowledge of the person's age.

Chain rule for Bayesian probability can be used to predict the likelihood of the next word in the sentence.

## **70. What is Naive Bayes? Why is it Naive?**

Naive Bayes classifiers are a series of classification algorithms that are based on the Bayes theorem. This family of algorithm shares a common principle which treats every pair of features independently while being classified.

Naive Bayes is considered Naive because the attributes in it (for the class) is independent of others in the same class. This lack of dependence between two attributes of the same class creates the quality of naiveness.

## **71. Explain how a Naive Bayes Classifier works.**

Naive Bayes classifiers are a family of algorithms which are derived from the Bayes theorem of probability. It works on the fundamental assumption that every set of two features that is being classified is independent of each other and every feature makes an equal and independent contribution to the outcome.

## **72. What do the terms prior probability and marginal likelihood in context of Naive Bayes theorem mean?**

Prior probability is the percentage of dependent binary variables in the data set. If you are given a dataset and dependent variable is either 1 or 0 and percentage of 1 is 65% and percentage of 0 is 35%. Then, the probability that any new input for that variable of being 1 would be 65%.

Marginal likelihood is the denominator of the Bayes equation and it makes sure that the posterior probability is valid by making its area 1.

## **73. Explain the difference between Lasso and Ridge?**

Lasso(L1) and Ridge(L2) are the regularization techniques where we penalize the coefficients to find the optimum solution. In ridge, the penalty function is defined by the sum of the squares of the coefficients and for the Lasso, we penalize the sum of the absolute values of the coefficients. Another type of regularization method is ElasticNet, it is a hybrid penalizing function of both lasso and ridge.

#### **74. What's the difference between probability and likelihood?**

Probability is the measure of the likelihood that an event will occur that is, what is the certainty that a specific event will occur? Whereas a likelihood function is a function of parameters within the parameter space that describes the probability of obtaining the observed data.

So the fundamental difference is, Probability attaches to possible results; likelihood attaches to hypotheses.

#### **75. Why would you Prune your tree?**

In the context of data science or AIML, pruning refers to the process of reducing redundant branches of a decision tree. Decision Trees are prone to overfitting, pruning the tree helps to reduce the size and minimizes the chances of overfitting. Pruning involves turning branches of a decision tree into leaf nodes and removing the leaf nodes from the original branch. It serves as a tool to perform the tradeoff.

#### **76. Model accuracy or Model performance? Which one will you prefer and why?**

This is a trick question, one should first get a clear idea, what is Model Performance? If Performance means speed, then it depends upon the nature of the application, any application related to the real-time scenario will need high speed as an important feature. Example: The best of Search Results will lose its virtue if the Query results do not appear fast.

If Performance is hinted at Why Accuracy is not the most important virtue – For any imbalanced data set, more than Accuracy, it will be an F1 score than will explain the business case and in case data is imbalanced, then Precision and Recall will be more important than rest.

#### **77. List the advantages and limitations of the Temporal Difference Learning Method.**

Temporal Difference Learning Method is a mix of Monte Carlo method and Dynamic programming method. Some of the advantages of this method include:

- It can learn in every step online or offline.
- It can learn from a sequence which is not complete as well.
- It can work in continuous environments.
- It has lower variance compared to MC method and is more efficient than MC method.

*Limitations of TD method are:*

- It is a biased estimation.
- It is more sensitive to initialization.

## 78. How would you handle an imbalanced dataset?

Sampling Techniques can help with an imbalanced dataset. There are two ways to perform sampling, Under Sample or Over Sampling.

In Under Sampling, we reduce the size of the majority class to match minority class thus help by improving performance w.r.t storage and run-time execution, but it potentially discards useful information.

For Over Sampling, we upsample the Minority class and thus solve the problem of information loss, however, we get into the trouble of having Overfitting.

There are other techniques as well –

**Cluster-Based Over Sampling** – In this case, the K-means clustering algorithm is independently applied to minority and majority class instances. This is to identify clusters in the dataset. Subsequently, each cluster is oversampled such that all clusters of the same class have an equal number of instances and all classes have the same size

**Synthetic Minority Over-sampling Technique (SMOTE)** – A subset of data is taken from the minority class as an example and then new synthetic similar instances are created which are then added to the original dataset. This technique is good for Numerical data points.

## 79. Mention some of the EDA Techniques?

Exploratory Data Analysis (EDA) helps analysts to understand the data better and forms the foundation of better models.

## Visualization

- Univariate visualization
- Bivariate visualization
- Multivariate visualization

**Missing Value Treatment** – Replace missing values with Either Mean/Median

**Outlier Detection** – Use Boxplot to identify the distribution of Outliers, then Apply IQR to set the boundary for IQR

**Transformation** – Based on the distribution, apply a transformation on the features

**Scaling the Dataset** – Apply MinMax, Standard Scaler or Z Score Scaling mechanism to scale the data.

**Feature Engineering** – Need of the domain, and SME knowledge helps Analyst find derivative fields which can fetch more information about the nature of the data

**Dimensionality reduction** — Helps in reducing the volume of data without losing much information

## 80. Mention why feature engineering is important in model building and list out some of the techniques used for feature engineering.

Algorithms necessitate features with some specific characteristics to work appropriately. The data is initially in a raw form. You need to extract features from this data before supplying it to the algorithm. This process is called feature engineering. When you have relevant features, the complexity of the algorithms reduces. Then, even if a non-ideal algorithm is used, results come out to be accurate.

Feature engineering primarily has two goals:

- Prepare the suitable input data set to be compatible with the machine learning algorithm constraints.
- Enhance the performance of machine learning models.

Some of the techniques used for feature engineering include Imputation, Binning, Outliers Handling, Log transform, grouping operations, One-Hot encoding, Feature split, Scaling, Extracting date.

## **81. Differentiate between Statistical Modeling and Machine Learning?**

Machine learning models are about making accurate predictions about the situations, like Foot Fall in restaurants, Stock-Price, etc. where-as, Statistical models are designed for inference about the relationships between variables, as What drives the sales in a restaurant, is it food or Ambience.

## **82. Differentiate between Boosting and Bagging?**

Bagging and Boosting are variants of Ensemble Techniques.

**Bootstrap Aggregation or bagging** is a method that is used to reduce the variance for algorithms having very high variance. Decision trees are a particular family of classifiers which are susceptible to having high bias.

Decision trees have a lot of sensitiveness to the type of data they are trained on. Hence generalization of results is often much more complex to achieve in them despite very high fine-tuning. The results vary greatly if the training data is changed in decision trees.

Hence bagging is utilised where multiple decision trees are made which are trained on samples of the original data and the final result is the average of all these individual models.

**Boosting** is the process of using an n-weak classifier system for prediction such that every weak classifier compensates for the weaknesses of its classifiers. By weak classifier, we imply a classifier which performs poorly on a given data set.

It's evident that boosting is not an algorithm rather it's a process. Weak classifiers used are generally logistic regression, shallow decision trees etc.

There are many algorithms which make use of boosting processes but two of them are mainly used: Adaboost and Gradient Boosting and XGBoost.

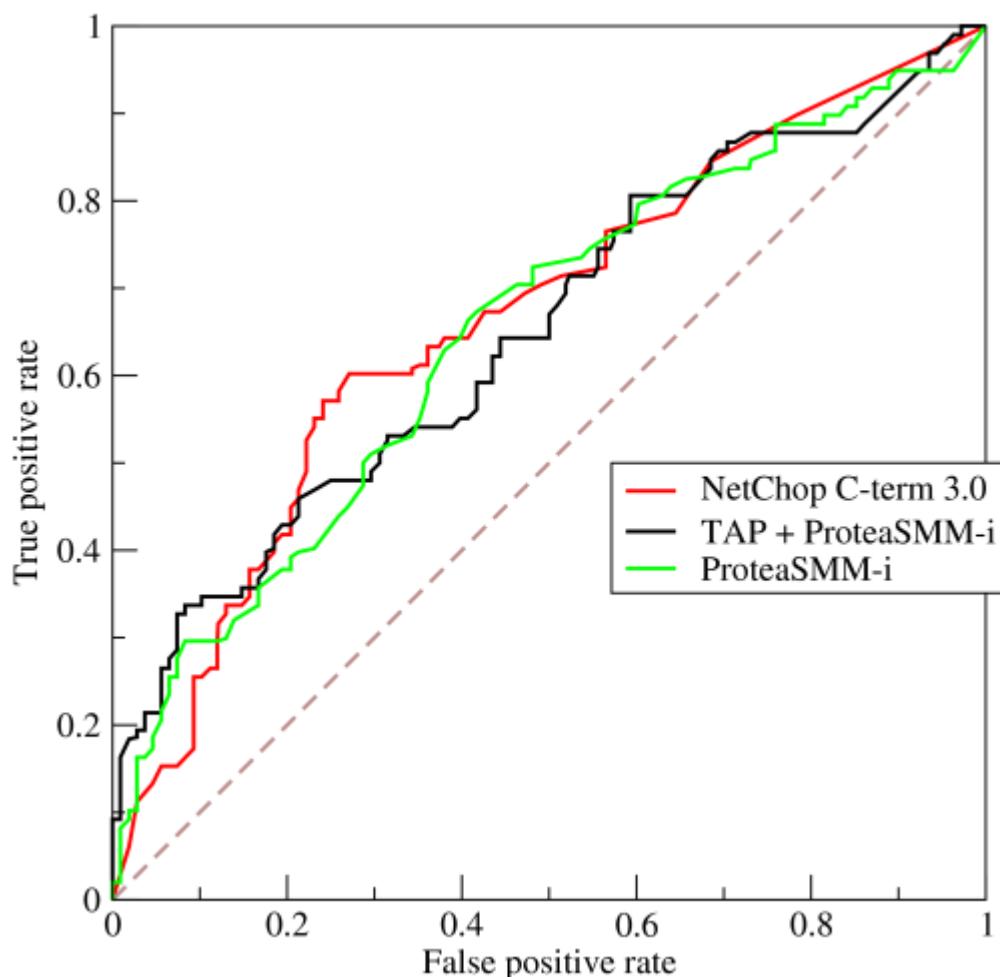
## **83. What is the significance of Gamma and Regularization in SVM?**

The gamma defines influence. Low values meaning ‘far’ and high values meaning ‘close’. If gamma is too large, the radius of the area of influence of the support vectors only includes the support vector itself and no amount of regularization with C will be able to prevent overfitting. If gamma is very small, the model is too constrained and cannot capture the complexity of the data.

The regularization parameter (lambda) serves as a degree of importance that is given to miss-classifications. This can be used to draw the tradeoff with OverFitting.

#### 84. Define ROC curve work

The graphical representation of the contrast between true positive rates and the false positive rate at various thresholds is known as the ROC curve. It is used as a proxy for the trade-off between true positives vs the false positives.



#### 85. What is the difference between a generative and discriminative model?

A generative model learns the different categories of data. On the other hand, a discriminative model will only learn the distinctions between different categories of data. Discriminative models perform much better than the generative models when it comes to classification tasks.

## **86. What are hyperparameters and how are they different from parameters?**

A parameter is a variable that is internal to the model and whose value is estimated from the training data. They are often saved as part of the learned model. Examples include weights, biases etc.

A hyperparameter is a variable that is external to the model whose value cannot be estimated from the data. They are often used to estimate model parameters. The choice of parameters is sensitive to implementation. Examples include learning rate, hidden layers etc.

## **87. What is shattering a set of points? Explain VC dimension.**

In order to shatter a given configuration of points, a classifier must be able to, for all possible assignments of positive and negative for the points, perfectly partition the plane such that positive points are separated from negative points. For a configuration of  $n$  points, there are  $2^n$  possible assignments of positive or negative.

When choosing a classifier, we need to consider the type of data to be classified and this can be known by VC dimension of a classifier. It is defined as cardinality of the largest set of points that the classification algorithm i.e. the classifier can shatter. In order to have a VC dimension of *at least n*, a classifier must be able to shatter a single given configuration of  $n$  points.

## **88. What are some differences between a linked list and an array?**

Arrays and Linked lists are both used to store linear data of similar types. However, there are a few difference between them.

**Array**

**Linked List**

Elements are well-indexed, making specific element accessing easier

Elements need to be accessed in a cumulative manner

Operations (insertion, deletion) are faster in array

Linked list takes linear time, making operations a bit slower

Arrays are of fixed size

Linked lists are dynamic and flexible

Memory is assigned during compile time in an array

Memory is allocated during execution or runtime in Linked list.

Elements are stored consecutively in arrays.

Elements are stored randomly in Linked list

Memory utilization is inefficient in the array

Memory utilization is efficient in the linked list.

## **89. What is the meshgrid () method and the contourf () method? State some uses of both.**

The meshgrid( ) function in numpy takes two arguments as input : range of x-values in the grid, range of y-values in the grid whereas meshgrid needs to be built before the contourf( ) function in matplotlib is used which takes in many inputs : x-values, y-values, fitting curve (contour line) to be plotted in grid, colours etc.

Meshgrid () function is used to create a grid using 1-D arrays of x-axis inputs and y-axis inputs to represent the matrix indexing. Contourf () is used to draw filled contours using the given x-axis inputs, y-axis inputs, contour line, colours etc.

## **90. Describe a hash table.**

Hashing is a technique for identifying unique objects from a group of similar objects. Hash functions are large keys converted into small keys in hashing techniques. The values of hash functions are stored in data structures which are known hash table.

## **91. List the advantages and disadvantages of using Neural Networks.**

Advantages:

We can store information on the entire network instead of storing it in a database. It has the ability to work and give a good accuracy even with inadequate information. A neural network has parallel processing ability and distributed memory.

Disadvantages:

Neural Networks requires processors which are capable of parallel processing. It's unexplained functioning of the network is also quite an issue as it reduces the trust in the network in some situations like when we have to show the problem we noticed to the network. Duration of the network is mostly unknown. We can only know that the training is finished by looking at the error value but it doesn't give us optimal results.

## **92. You have to train a 12GB dataset using a neural network with a machine which has only 3GB RAM. How would you go about it?**

We can use NumPy arrays to solve this issue. Load all the data into an array. In NumPy, arrays have a property to map the complete dataset without loading it completely in memory. We can pass the index of the array, dividing data into batches, to get the data required and then pass the data into the neural networks. But be careful about keeping the batch size normal.

## **93. Write a simple code to binarize data.**

Conversion of data into binary values on the basis of certain threshold is known as binarizing of data. Values below the threshold are set to 0 and those above the threshold are set to 1 which is useful for feature engineering.

Code:

```
from sklearn.preprocessing import Binarizer  
  
import pandas  
  
import numpy
```

```
names_list = ['Alaska', 'Pratyush', 'Pierce', 'Sandra', 'Soundarya', 'Meredith',
'Richard', 'Jackson', 'Tom', 'Joe']

data_frame = pandas.read_csv(url, names=names_list)

array = dataframe.values

# Splitting the array into input and output

A = array [: 0:7]

B = array [:7]

binarizer = Binarizer(threshold=0.0). fit(X)

binaryA = binarizer.transform(A)

numpy.set_printoptions(precision=5)

print (binaryA [0:7:])
```

## 94. What is an Array?

The array is defined as a collection of similar items, stored in a contiguous manner. Arrays is an intuitive concept as the need to group similar objects together arises in our day to day lives. Arrays satisfy the same need. How are they stored in the memory? Arrays consume blocks of data, where each element in the array consumes one unit of memory. The size of the unit depends on the type of data being used. For example, if the data type of elements of the array is int, then 4 bytes of data will be used to store each element. For character data type, 1 byte will be used. This is implementation specific, and the above units may change from computer to computer.

Example:

```
fruits = ['apple', 'banana', 'pineapple']
```

In the above case, fruits is a list that comprises of three fruits. To access them individually, we use their indexes. Python and C are 0- indexed languages, that

is, the first index is 0. MATLAB on the contrary starts from 1, and thus is a 1-indexed language.

## 95. What are the advantages and disadvantages of using an Array?

1. Advantages:
  - Random access is enabled
  - Saves memory
  - Cache friendly
  - Predictable compile timing
  - Helps in re-usability of code
- Disadvantages:
  1. Addition and deletion of records is time consuming even though we get the element of interest immediately through random access. This is due to the fact that the elements need to be reordered after insertion or deletion.
  2. If contiguous blocks of memory are not available in the memory, then there is an overhead on the CPU to search for the most optimal contiguous location available for the requirement.

Now that we know what arrays are, we shall understand them in detail by solving some interview questions. Before that, let us see the functions that Python as a language provides for arrays, also known as, lists.

append() – Adds an element at the end of the list

copy() – returns a copy of a list.

reverse() – reverses the elements of the list

sort() – sorts the elements in ascending order by default.

## 96. What is Lists in Python?

Lists is an effective data structure provided in python. There are various functionalities associated with the same. Let us consider the scenario where we want to copy a list to another list. If the same operation had to be done in C programming language, we would have to write our own function to implement the same.

On the contrary, Python provides us with a function called copy. We can copy a list to another just by calling the copy function.

```
new_list = old_list.copy()
```

We need to be careful while using the function. `copy()` is a shallow copy function, that is, it only stores the references of the original list in the new list. If the given argument is a compound data structure like a list then python creates another object of the same type (in this case, a new list) but for everything inside old list, only their reference is copied. Essentially, the new list consists of references to the elements of the older list.

Hence, upon changing the original list, the new list values also change. This can be dangerous in many applications. Therefore, Python provides us with another functionality called as `deepcopy`. Intuitively, we may consider that `deepcopy()` would follow the same paradigm, and the only difference would be that for each element we will recursively call `deepcopy`. Practically, this is not the case.

`deepcopy()` preserves the graphical structure of the original compound data. Let us understand this better with the help of an example:

```
import copy.deepcopy  
  
a = [1,2]  
  
b = [a,a] # there's only 1 object a  
  
c = deepcopy(b)  
  
  
  
# check the result by executing these lines  
  
c[0] is a # return False, a new object a' is created  
  
c[0] is c[1] # return True, c is [a',a'] not [a',a"]
```

This is the tricky part, during the process of `deepcopy()` a hashtable implemented as a dictionary in python is used to map: `old_object` reference onto `new_object` reference.

Therefore, this prevents unnecessary duplicates and thus preserves the structure of the copied compound data structure. Thus, in this case, `c[0]` is not equal to `a`, as internally their addresses are different.

### Normal copy

```
>>> a = [[1, 2, 3], [4, 5, 6]]
```

```
>>> b = list(a)
```

```
>>> a
```

```
[[1, 2, 3], [4, 5, 6]]
```

```
>>> b
```

```
[[1, 2, 3], [4, 5, 6]]
```

```
>>> a[0][1] = 10
```

```
>>> a
```

```
[[1, 10, 3], [4, 5, 6]]
```

```
>>> b # b changes too -> Not a deepcopy.
```

```
[[1, 10, 3], [4, 5, 6]]
```

### Deep copy

```
>>> import copy
```

```
>>> b = copy.deepcopy(a)
```

```
>>> a
```

```
[[1, 10, 3], [4, 5, 6]]
```

```
>>> b
```

```
[[1, 10, 3], [4, 5, 6]]
```

```
>>> a[0][1] = 9
```

```
>>> a
```

```
[[1, 9, 3], [4, 5, 6]]
```

```
>>> b # b doesn't change -> Deep Copy
```

```
[[1, 10, 3], [4, 5, 6]]
```

Now that we have understood the concept of lists, let us solve interview questions to get better exposure on the same.

**97. Given an array of integers where each element represents the max number of steps that can be made forward from that element. The task is to find the minimum number of jumps to reach the end of the array (starting from the first element). If an element is 0, then cannot move through that element.**

Solution: This problem is famously called as end of array problem. We want to determine the minimum number of jumps required in order to reach the end. The element in the array represents the maximum number of jumps that, that particular element can take.

Let us understand how to approach the problem initially.

We need to reach the end. Therefore, let us have a count that tells us how near we are to the end. Consider the array A=[1,2,3,1,1]

In the above example we can go from

> 2 - > 3 - > 1 - > 1 - 4 jumps

1 - > 2 - > 1 - > 1 - 3 jumps

1 - > 2 - > 3 - > 1 - 3 jumps

Hence, we have a fair idea of the problem. Let us come up with a logic for the same.

Let us start from the end and move backwards as that makes more sense intuitively. We will use variables right and prev\_r denoting previous right to keep track of the jumps.

Initially, right = prev\_r = the last but one element. We consider the distance of an element to the end, and the number of jumps possible by that element. Therefore, if the sum of the number of jumps possible and the distance is greater than the previous element, then we will discard the previous element and use the second element's value to jump. Try it out using a pen and paper first. The logic will seem very straight forward to implement. Later, implement it on your own and then verify with the result.

```
def min_jmp(arr):
```

```
n = len(arr)
```

```
right = prev_r = n-1
```

```
count = 0
```

```
# We start from rightmost index and traverse array to find the leftmost index
```

```
# from which we can reach index 'right'
```

```
while True:
```

```
for j in (range(prev_r-1,-1,-1)):
```

```
if j + arr[j] >= prev_r:
```

```
    right = j
```

```
if prev_r != right:
```

```
    prev_r = right
```

```
else:
```

```
    break
```

```
count += 1
```

```
return count if right == 0 else -1
```

```
# Enter the elements separated by a space
```

```
arr = list(map(int, input().split()))
```

```
print(min_jmp(n, arr))
```

## **98. Given a string S consisting only ‘a’s and ‘b’s, print the last index of the ‘b’ present in it.**

When we have are given a string of a’s and b’s, we can immediately find out the first location of a character occurring. Therefore, to find the last occurrence of a character, we reverse the string and find the first occurrence, which is equivalent to the last occurrence in the original string.

Here, we are given input as a string. Therefore, we begin by splitting the characters element wise using the function split. Later, we reverse the array, find the first occurrence position value, and get the index by finding the value len – position -1, where position is the index value.

```
def split(word):
```

```
    return [(char) for char in word]
```

```
a = input()
```

```
a= split(a)
```

```
a_rev = a[::-1]
```

```
pos = -1
```

```
for i in range(len(a_rev)):
```

```
    if a_rev[i] == ‘b’:
```

```
        pos = len(a_rev)- i -1
```

```
        print(pos)
```

```
        break
```

```
else:
```

continue

```
if pos== -1:
```

```
    print(-1)
```

**99. Rotate the elements of an array by d positions to the left. Let us initially look at an example.**

```
A = [1,2,3,4,5]
```

```
A <<2
```

```
[3,4,5,1,2]
```

```
A<<3
```

```
[4,5,1,2,3]
```

There exists a pattern here, that is, the first d elements are being interchanged with last n-d +1 elements. Therefore we can just swap the elements. Correct? What if the size of the array is huge, say 10000 elements. There are chances of memory error, run-time error etc. Therefore, we do it more carefully. We rotate the elements one by one in order to prevent the above errors, in case of large arrays.

```
# Rotate all the elements left by 1 position
```

```
def rot_left_once ( arr):
```

```
    n = len( arr)
```

```
    tmp = arr [0]
```

```
    for i in range ( n-1): #[0,n-2]
```

```
        arr[i] = arr[i + 1]
```

```
    arr[n-1] = tmp
```

```
# Use the above function to repeat the process for d times.
```

```
def rot_left (arr, d):
```

```
    n = len (arr)
```

```
    for i in range (d):
```

```
        rot_left_once ( arr, n)
```

```
arr = list( map( int, input().split()))
```

```
rot =int( input())
```

```
leftRotate ( arr, rot)
```

```
for i in range( len(arr)):
```

```
    print( arr[i], end=' ')
```

## 100. Water Trapping Problem

Given an array arr[] of N non-negative integers which represents the height of blocks at index I, where the width of each block is 1. Compute how much water can be trapped in between blocks after raining.

```
# Structure is like below:
```

```
# ||
```

```
# |_|
```

```
# answer is we can trap two units of water.
```

Solution: We are given an array, where each element denotes the height of the block. One unit of height is equal to one unit of water, given there exists space between the 2 elements to store it. Therefore, we need to find out all such pairs that exist which can store water. We need to take care of the possible cases:

- There should be no overlap of water saved
- Water should not overflow

Therefore, let us start with the extreme elements, and move towards the centre.

```
n = int(input())
```

```
arr = [int(i) for i in input().split()]
```

```
left, right = [arr[0]], [0] * n
```

```
# left =[arr[0]]
```

```
#right = [ 0 0 0 0...0] n terms
```

```
right[n-1] = arr[-1] # right most element
```

```
# we use two arrays left[ ] and right[ ], which keep track of elements greater than all
```

```
# elements the order of traversal respectively.
```

```
for elem in arr[1 :]:
```

```
    left.append(max(left[-1], elem))
```

```
for i in range( len( arr)-2, -1, -1):
```

```
    right[i] = max( arr[i] , right[i+1] )
```

```
water = 0
```

```
# once we have the arrays left, and right, we can find the water capacity between these arrays.
```

```
for i in range( 1, n - 1):
```

```
    add_water = min( left[i - 1], right[i]) - arr[i]
```

```
    if add_water > 0:
```

```
        water += add_water
```

```
    print(water)
```

## 101. Explain Eigenvectors and Eigenvalues.

**Ans.** Linear transformations are helpful to understand using eigenvectors. They find their prime usage in the creation of covariance and correlation matrices in data science.

Simply put, eigenvectors are directional entities along which linear transformation features like compression, flip etc. can be applied.

Eigenvalues are the magnitude of the linear transformation features along each direction of an Eigenvector.

## 102. How would you define the number of clusters in a clustering algorithm?

**Ans.** The number of clusters can be determined by finding the silhouette score. Often we aim to get some inferences from data using clustering techniques so that we can have a broader picture of a number of classes being represented by the data. In this case, the silhouette score helps us determine the number of cluster centres to cluster our data along.

Another technique that can be used is the elbow method.

## 103. What are the performance metrics that can be used to estimate the efficiency of a linear regression model?

**Ans.** The performance metric that is used in this case is:

- Mean Squared Error

- R<sup>2</sup> score
- Adjusted R<sup>2</sup> score
- Mean Absolute score

#### **104. What is the default method of splitting in decision trees?**

The default method of splitting in decision trees is the Gini Index. Gini Index is the measure of impurity of a particular node.

This can be changed by making changes to classifier parameters.

#### **105. How is p-value useful?**

**Ans.** The p-value gives the probability of the null hypothesis is true. It gives us the statistical significance of our results. In other words, p-value determines the confidence of a model in a particular output.

#### **106. Can logistic regression be used for classes more than 2?**

**Ans.** No, logistic regression cannot be used for classes more than 2 as it is a binary classifier. For multi-class classification algorithms like Decision Trees, Naïve Bayes' Classifiers are better suited.

#### **107. What are the hyperparameters of a logistic regression model?**

**Ans.** Classifier penalty, classifier solver and classifier C are the trainable hyperparameters of a Logistic Regression Classifier. These can be specified exclusively with values in Grid Search to hyper tune a Logistic Classifier.

#### **108. Name a few hyper-parameters of decision trees?**

**Ans.** The most important features which one can tune in decision trees are:

- Splitting criteria
- Min\_leaves
- Min\_samples
- Max\_depth

#### **109. How to deal with multicollinearity?**

**Ans.** Multi collinearity can be dealt with by the following steps:

- Remove highly correlated predictors from the model.

- Use Partial Least Squares Regression (PLS) or Principal Components Analysis

## **110. What is Heteroscedasticity?**

**Ans.** It is a situation in which the variance of a variable is unequal across the range of values of the predictor variable.

It should be avoided in regression as it introduces unnecessary variance.

## **111. Is ARIMA model a good fit for every time series problem?**

**Ans.** No, ARIMA model is not suitable for every type of time series problem. There are situations where ARMA model and others also come in handy.

ARIMA is best when different standard temporal structures require to be captured for time series data.

## **112. How do you deal with the class imbalance in a classification problem?**

**Ans.** Class imbalance can be dealt with in the following ways:

- Using class weights
- Using Sampling
- Using SMOTE
- Choosing loss functions like Focal Loss

## **113. What is the role of cross-validation?**

**Ans.** Cross-validation is a technique which is used to increase the performance of a machine learning algorithm, where the machine is fed sampled data out of the same data for a few times. The sampling is done so that the dataset is broken into small parts of the equal number of rows, and a random part is chosen as the test set, while all other parts are chosen as train sets.

## **114. What is a voting model?**

**Ans.** A voting model is an ensemble model which combines several classifiers but to produce the final result, in case of a classification-based model, takes into account, the classification of a certain data point of all the models and picks the most vouched/voted/generated option from all the given classes in the target column.

**115. How to deal with very few data samples? Is it possible to make a model out of it?**

**Ans.** If very few data samples are there, we can make use of oversampling to produce new data points. In this way, we can have new data points.

**116. What are the hyperparameters of an SVM?**

**Ans.** The gamma value, c value and the type of kernel are the hyperparameters of an SVM model.

**117. What is Pandas Profiling?**

**Ans.** Pandas profiling is a step to find the effective number of usable data. It gives us the statistics of NULL values and the usable values and thus makes variable selection and data selection for building models in the preprocessing phase very effective.

**118. What impact does correlation have on PCA?**

**Ans.** If data is correlated PCA does not work well. Because of the correlation of variables the effective variance of variables decreases. Hence correlated data when used for PCA does not work well.

**119. How is PCA different from LDA?**

**Ans.** PCA is unsupervised. LDA is unsupervised.

PCA takes into consideration the variance. LDA takes into account the distribution of classes.

**120. What distance metrics can be used in KNN?**

**Ans.** Following distance metrics can be used in KNN.

- Manhattan
- Minkowski
- Tanimoto
- Jaccard
- Mahalanobis

**121. Which metrics can be used to measure correlation of categorical data?**

**Ans.** Chi square test can be used for doing so. It gives the measure of correlation between categorical predictors.

**122. Which algorithm can be used in value imputation in both categorical and continuous categories of data?**

**Ans.** KNN is the only algorithm that can be used for imputation of both categorical and continuous variables.

**123. When should ridge regression be preferred over lasso?**

**Ans.** We should use ridge regression when we want to use all predictors and not remove any as it reduces the coefficient values but does not nullify them.

**124. Which algorithms can be used for important variable selection?**

**Ans.** Random Forest, Xgboost and plot variable importance charts can be used for variable selection.

**125. What ensemble technique is used by Random forests?**

**Ans.** Bagging is the technique used by Random Forests. Random forests are a collection of trees which work on sampled data from the original dataset with the final prediction being a voted average of all trees.

**126. What ensemble technique is used by gradient boosting trees?**

**Ans.** Boosting is the technique used by GBM.

**127. If we have a high bias error what does it mean? How to treat it?**

**Ans.** High bias error means that that model we are using is ignoring all the important trends in the model and the model is underfitting.

To reduce underfitting:

- We need to increase the complexity of the model
- Number of features need to be increased

Sometimes it also gives the impression that the data is noisy. Hence noise from data should be removed so that most important signals are found by the model to make effective predictions.

Increasing the number of epochs results in increasing the duration of training of the model. It's helpful in reducing the error.

## **128. Which type of sampling is better for a classification model and why?**

**Ans.** Stratified sampling is better in case of classification problems because it takes into account the balance of classes in train and test sets. The proportion of classes is maintained and hence the model performs better. In case of random sampling of data, the data is divided into two parts without taking into consideration the balance classes in the train and test sets. Hence some classes might be present only in training sets or validation sets. Hence the results of the resulting model are poor in this case.

## **129. What is a good metric for measuring the level of multicollinearity?**

**Ans.** VIF or 1/tolerance is a good measure of measuring multicollinearity in models. VIF is the percentage of the variance of a predictor which remains unaffected by other predictors. So higher the VIF value, greater is the multicollinearity amongst the predictors.

A rule of thumb for interpreting the variance inflation factor:

- 1 = not correlated.
- Between 1 and 5 = moderately correlated.
- Greater than 5 = highly correlated.

## **130. When can be a categorical value treated as a continuous variable and what effect does it have when done so?**

**Ans.** A categorical predictor can be treated as a continuous one when the nature of data points it represents is ordinal. If the predictor variable is having ordinal data then it can be treated as continuous and its inclusion in the model increases the performance of the model.

## **131. What is the role of maximum likelihood in logistic regression.**

**Ans.** Maximum likelihood equation helps in estimation of most probable values of the estimator's predictor variable coefficients which produces results which are the most likely or most probable and are quite close to the truth values.

## **132. Which distance do we measure in the case of KNN?**

**Ans.** The hamming distance is measured in case of KNN for the determination of nearest neighbours. Kmeans uses euclidean distance.

### **133. What is a pipeline?**

**Ans.** A pipeline is a sophisticated way of writing software such that each intended action while building a model can be serialized and the process calls the individual functions for the individual tasks. The tasks are carried out in sequence for a given sequence of data points and the entire process can be run onto n threads by use of composite estimators in scikit learn.

### **134. Which sampling technique is most suitable when working with time-series data?**

**Ans.** We can use a custom iterative sampling such that we continuously add samples to the train set. We only should keep in mind that the sample used for validation should be added to the next train sets and a new sample is used for validation.

### **135. What are the benefits of pruning?**

**Ans.** Pruning helps in the following:

- Reduces overfitting
- Shortens the size of the tree
- Reduces complexity of the model
- Increases bias

### **136. What is normal distribution?**

**Ans.** The distribution having the below properties is called normal distribution.

- The mean, mode and median are all equal.
- The curve is symmetric at the center (i.e. around the mean,  $\mu$ ).
- Exactly half of the values are to the left of center and exactly half the values are to the right.
- The total area under the curve is 1.

### **137. What is the 68 per cent rule in normal distribution?**

**Ans.** The normal distribution is a bell-shaped curve. Most of the data points are around the median. Hence approximately 68 per cent of the data is around the median. Since there is no skewness and its bell-shaped.

### **138. What is a chi-square test?**

**Ans.** A chi-square determines if a sample data matches a population.

A chi-square test for independence compares two variables in a contingency table to see if they are related.

A very small chi-square test statistics implies observed data fits the expected data extremely well.

### **139. What is a random variable?**

**Ans.** A Random Variable is a set of possible values from a random experiment. Example: Tossing a coin: we could get Heads or Tails. Rolling of a dice: we get 6 values

### **140. What is the degree of freedom?**

**Ans.** It is the number of independent values or quantities which can be assigned to a statistical distribution. It is used in Hypothesis testing and chi-square test.

### **141. Which kind of recommendation system is used by amazon to recommend similar items?**

**Ans.** Amazon uses a collaborative filtering algorithm for the recommendation of similar items. It's a user to user similarity based mapping of user likeness and susceptibility to buy.

### **142. What is a false positive?**

**Ans.** It is a test result which wrongly indicates that a particular condition or attribute is present.

Example – “Stress testing, a routine diagnostic tool used in detecting heart disease, results in a significant number of false positives in women”

### **143. What is a false negative?**

**Ans.** A test result which wrongly indicates that a particular condition or attribute is absent.

Example – “it’s possible to have a false negative—the test says you aren’t pregnant when you are”

#### **144. What is the error term composed of in regression?**

**Ans.** Error is a sum of bias error+variance error+ irreducible error in regression. Bias and variance error can be reduced but not the irreducible error.

#### **145. Which performance metric is better R2 or adjusted R2?**

**Ans.** Adjusted R2 because the performance of predictors impacts it. R2 is independent of predictors and shows performance improvement through increase if the number of predictors is increased.

#### **146. What's the difference between Type I and Type II error?**

Type I and Type II error in machine learning refers to false values. Type I is equivalent to a False positive while Type II is equivalent to a False negative. In Type I error, a hypothesis which ought to be accepted doesn't get accepted. Similarly, for Type II error, the hypothesis gets rejected which should have been accepted in the first place.

#### **147. What do you understand by L1 and L2 regularization?**

L2 regularization: It tries to spread error among all the terms. L2 corresponds to a Gaussian prior.

L1 regularization: It is more binary/sparse, with many variables either being assigned a 1 or 0 in weighting. L1 corresponds to setting a Laplacean prior on the terms.

#### **148. Which one is better, Naive Bayes Algorithm or Decision Trees?**

Although it depends on the problem you are solving, but some general advantages are following:

##### **Naive Bayes:**

- Work well with small dataset compared to DT which need more data
- Lesser overfitting
- Smaller in size and faster in processing

## **Decision Trees:**

- Decision Trees are very flexible, easy to understand, and easy to debug
- No preprocessing or transformation of features required
- Prone to overfitting but you can use pruning or Random forests to avoid that.

### **149. What do you mean by the ROC curve?**

Receiver operating characteristics (ROC curve): ROC curve illustrates the diagnostic ability of a binary classifier. It is calculated/created by plotting True Positive against False Positive at various threshold settings. The performance metric of ROC curve is AUC (area under curve). Higher the area under the curve, better the prediction power of the model.

### **150. What do you mean by AUC curve?**

AUC (area under curve). Higher the area under the curve, better the prediction power of the model.

### **151. What is log likelihood in logistic regression?**

It is the sum of the likelihood residuals. At record level, the natural log of the error (residual) is calculated for each record, multiplied by minus one, and those values are totaled. That total is then used as the basis for deviance ( $2 \times ll$ ) and likelihood ( $\exp(ll)$ ).

The same calculation can be applied to a naive model that assumes absolutely no predictive power, and a saturated model assuming perfect predictions.

The likelihood values are used to compare different models, while the deviances (test, naive, and saturated) can be used to determine the predictive power and accuracy. Logistic regression accuracy of the model will always be 100 percent for the development data set, but that is not the case once a model is applied to another data set.

### **152. How would you evaluate a logistic regression model?**

Model Evaluation is a very important part in any analysis to answer the following questions,

How well does the model fit the data?, Which predictors are most important?, Are the predictions accurate?

So the following are the criterion to access the model performance,

- **Akaike Information Criteria (AIC):** In simple terms, AIC estimates the relative amount of information lost by a given model. So the less information lost the higher the quality of the model. Therefore, we always prefer models with minimum AIC.
- **Receiver operating characteristics (ROC curve):** ROC curve illustrates the diagnostic ability of a binary classifier. It is calculated/ created by plotting True Positive against False Positive at various threshold settings. The performance metric of ROC curve is AUC (area under curve). Higher the area under the curve, better the prediction power of the model.
- **Confusion Matrix:** In order to find out how well the model does in predicting the target variable, we use a confusion matrix/ classification rate. It is nothing but a tabular representation of actual Vs predicted values which helps us to find the accuracy of the model.

### 153. What are the advantages of SVM algorithms?

SVM algorithms have basically advantages in terms of complexity. First I would like to clear that both Logistic regression as well as SVM can form non linear decision surfaces and can be coupled with the kernel trick. If Logistic regression can be coupled with kernel then why use SVM?

- SVM is found to have better performance practically in most cases.
- SVM is computationally cheaper  $O(N^2 * K)$  where K is no of support vectors (support vectors are those points that lie on the class margin) where as logistic regression is  $O(N^3)$
- Classifier in SVM depends only on a subset of points . Since we need to maximize distance between closest points of two classes (aka margin) we need to care about only a subset of points unlike logistic regression.

### 154. Why does XGBoost perform better than SVM?

First reason is that XGBoos is an ensemble method that uses many trees to make a decision so it gains power by repeating itself.

SVM is a linear separator, when data is not linearly separable SVM needs a Kernel to project the data into a space where it can separate it, there lies its greatest strength and weakness, by being able to project data into a high dimensional space SVM can find a linear separation for almost any data but at the same time it needs to use a Kernel and we can argue that there's not a perfect kernel for every dataset.

### **155. What is the difference between SVM Rank and SVR (Support Vector Regression)?**

One is used for ranking and the other is used for regression.

There is a crucial difference between *regression* and *ranking*. In regression, the absolute value is crucial. A real number is predicted.

In ranking, the only thing of concern is the ordering of a set of examples. We only want to know which example has the highest rank, which one has the second-highest, and so on. From the data, we only know that example 1 should be ranked higher than example 2, which in turn should be ranked higher than example 3, and so on. We do not know by *how much* example 1 is ranked higher than example 2, or whether this difference is bigger than the difference between examples 2 and 3.

### **156. What is the difference between the normal soft margin SVM and SVM with a linear kernel?**

#### **Hard-margin**

You have the basic SVM – hard margin. This assumes that data is very well behaved, and you can find a perfect classifier – which will have 0 error on train data.

#### **Soft-margin**

Data is usually not well behaved, so SVM hard margins may not have a solution at all. So we allow for a little bit of error on some points. So the training error will not be 0, but average error over all points is minimized.

#### **Kernels**

The above assume that the best classifier is a straight line. But what if it is not a straight line. (e.g. it is a circle, inside a circle is one class, outside is another class). If we are able to map the data into higher dimensions – the higher dimension may give us a straight line.

### **157. How is linear classifier relevant to SVM?**

An svm is a type of linear classifier. If you don't mess with kernels, it's arguably the most simple type of linear classifier.

Linear classifiers (all?) learn linear functions from your data that map your input to scores like so:  $\text{scores} = \mathbf{W}\mathbf{x} + b$ . Where  $\mathbf{W}$  is a matrix of learned weights,  $b$  is a learned bias vector that shifts your scores, and  $\mathbf{x}$  is your input data. This type of function may look familiar to you if you remember  $y = mx + b$  from high school.

A typical svm loss function (the function that tells you how good your calculated scores are in relation to the correct labels) would be hinge loss. It takes the form:  $\text{Loss} = \text{sum over all scores except the correct score of } \max(0, \text{scores} - \text{scores(correct class)} + 1)$ .

### **158. What are the advantages of using a naive Bayes for classification?**

- Very simple, easy to implement and fast.
- If the NB conditional independence assumption holds, then it will converge quicker than discriminative models like logistic regression.
- Even if the NB assumption doesn't hold, it works great in practice.
- Need less training data.
- Highly scalable. It scales linearly with the number of predictors and data points.
- Can be used for both binary and multiclass classification problems.
- Can make probabilistic predictions.
- Handles continuous and discrete data.
- Not sensitive to irrelevant features.

### **159. Are Gaussian Naive Bayes the same as binomial Naive Bayes?**

Binomial Naive Bayes: It assumes that all our features are binary such that they take only two values. Means 0s can represent “word does not occur in the document” and 1s as “word occurs in the document”.

Gaussian Naive Bayes: Because of the assumption of the normal distribution, Gaussian Naive Bayes is used in cases when all our features are continuous. For example in Iris dataset features are sepal width, petal width, sepal length, petal length. So its features can have different values in the data set as width and length can vary. We can't represent features in terms of their occurrences. This means data is continuous. Hence we use Gaussian Naive Bayes here.

## **160. What is the difference between the Naive Bayes Classifier and the Bayes classifier?**

Naive Bayes assumes conditional independence,  $P(X|Y, Z)=P(X|Z)$

$$P(X|Y,Z)=P(X|Z)$$

$P(X|Y,Z)=P(X|Z)$ , Whereas more general Bayes Nets (sometimes called Bayesian Belief Networks), will allow the user to specify which attributes are, in fact, conditionally independent.

For the Bayesian network as a classifier, the features are selected based on some scoring functions like Bayesian scoring function and minimal description length(the two are equivalent in theory to each other given that there is enough training data). The scoring functions mainly restrict the structure (connections and directions) and the parameters(likelihood) using the data. After the structure has been learned the class is only determined by the nodes in the Markov blanket(its parents, its children, and the parents of its children), and all variables given the Markov blanket are discarded.

## **161. In what real world applications is Naive Bayes classifier used?**

Some of real world examples are as given below

- To mark an email as spam, or not spam?
- Classify a news article about technology, politics, or sports?
- Check a piece of text expressing positive emotions, or negative emotions?
- Also used for face recognition software

## **162. Is naive Bayes supervised or unsupervised?**

First, Naive Bayes is not one algorithm but a family of Algorithms that inherits the following attributes:

- Discriminant Functions

- Probabilistic Generative Models
- Bayesian Theorem
- Naive Assumptions of Independence and Equal Importance of feature vectors.

Moreover, it is a special type of Supervised Learning algorithm that could do simultaneous multi-class predictions (as depicted by standing topics in many news apps).

Since these are generative models, so based upon the assumptions of the random variable mapping of each feature vector these may even be classified as Gaussian Naive Bayes, Multinomial Naive Bayes, Bernoulli Naive Bayes, etc.

### **163. What do you understand by selection bias in Machine Learning?**

Selection bias stands for the bias which was introduced by the selection of individuals, groups or data for doing analysis in a way that the proper randomization is not achieved. It ensures that the sample obtained is not representative of the population intended to be analyzed and sometimes it is referred to as the selection effect. This is the part of distortion of a statistical analysis which results from the method of collecting samples. If you don't take the selection bias into the account then some conclusions of the study may not be accurate.

The types of selection bias includes:

- **Sampling bias:** It is a systematic error due to a non-random sample of a population causing some members of the population to be less likely to be included than others resulting in a biased sample.
- **Time interval:** A trial may be terminated early at an extreme value (often for ethical reasons), but the extreme value is likely to be reached by the variable with the largest variance, even if all variables have a similar mean.
- **Data:** When specific subsets of data are chosen to support a conclusion or rejection of bad data on arbitrary grounds, instead of according to previously stated or generally agreed criteria.
- **Attrition:** Attrition bias is a kind of selection bias caused by attrition (loss of participants) discounting trial subjects/tests that did not run to completion.

### **164. What do you understand by Precision and Recall?**

In pattern recognition, The information retrieval and classification in machine learning are part of **precision**. It is also called as positive predictive value which is the fraction of relevant instances among the retrieved instances.

**Recall** is also known as sensitivity and the fraction of the total amount of relevant instances which were actually retrieved.

Both precision and recall are therefore based on an understanding and measure of relevance.

## **165. What Are the Three Stages of Building a Model in Machine Learning?**

To build a model in machine learning, you need to follow few steps:

- Understand the business model
- Data acquisitions
- Data cleaning
- Exploratory data analysis
- Use machine learning algorithms to make a model
- Use unknown dataset to check the accuracy of the model

## **166. How Do You Design an Email Spam Filter in Machine Learning?**

- Understand the business model: Try to understand the related attributes for the spam mail
- Data acquisitions: Collect the spam mail to read the hidden pattern from them
- Data cleaning: Clean the unstructured or semi structured data
- Exploratory data analysis: Use statistical concepts to understand the data like spread, outlier, etc.
- Use machine learning algorithms to make a model: can use naive bayes or some other algorithms as well
- Use unknown dataset to check the accuracy of the model

## **167. What is the difference between Entropy and Information Gain?**

The **information gain** is based on the decrease in **entropy** after a dataset is split on an attribute. Constructing a decision tree is all about finding the attribute that returns the highest **information gain** (i.e., the most homogeneous branches).

Step 1: Calculate **entropy** of the target.

## **168. What are collinearity and multicollinearity?**

**Collinearity** is a linear association **between** two predictors. **Multicollinearity** is a situation where two or more predictors are highly linearly related.

## 169. What is Kernel SVM?

SVM algorithms have basically advantages in terms of complexity. First I would like to clear that both Logistic regression as well as SVM can form non linear decision surfaces and can be coupled with the kernel trick. If Logistic regression can be coupled with kernel then why use SVM?

- SVM is found to have better performance practically in most cases.
- SVM is computationally cheaper  $O(N^2*K)$  where K is no of support vectors (support vectors are those points that lie on the class margin) where as logistic regression is  $O(N^3)$
- Classifier in SVM depends only on a subset of points . Since we need to maximize distance between closest points of two classes (aka margin) we need to care about only a subset of points unlike logistic regression.

## 170. What is the process of carrying out a linear regression?

**Linear Regression** Analysis consists of more than just fitting a **linear** line through a cloud of data points. It consists of 3 stages—

- analyzing the correlation and directionality of the data,
- estimating the **model**, i.e., fitting the line,
- evaluating the validity and usefulness of the **model**.

Q1: What's the trade-off between bias and variance?

**Answer:** Bias is error due to erroneous or overly simplistic assumptions in the learning algorithm you're using. This can lead to the model underfitting your data, making it hard for it to have high predictive accuracy and for you to generalize your knowledge from the training set to the test set.

Variance is error due to too much complexity in the learning algorithm you're using. This leads to the algorithm being highly sensitive to high degrees of variation in your training data, which can lead your model to overfit the data. You'll be carrying too much noise from your training data for your model to be very useful for your test data.

The bias-variance decomposition essentially decomposes the learning error from any algorithm by adding the bias, the variance and a bit of irreducible error due to noise in the underlying dataset. Essentially, if you make the model more complex and add more variables, you'll lose bias but gain some variance — in order to get the optimally reduced amount of error, you'll have to tradeoff bias and variance. You don't want either high bias or high variance in your model.

Q2: What is the difference between supervised and unsupervised machine learning?

**Answer:** Supervised learning requires training labeled data. For example, in order to do classification (a supervised learning task), you'll need to first label the data you'll use to train the model to classify data into your labeled groups. Unsupervised learning, in contrast, does not require labeling data explicitly.

Q3: How is KNN different from k-means clustering?

**Answer:** K-Nearest Neighbors is a supervised classification algorithm, while k-means clustering is an unsupervised clustering algorithm. While the mechanisms may seem similar at first, what this really means is that in order for K-Nearest Neighbors to work, you need labeled data you want to classify an unlabeled point into (thus the nearest neighbor part). K-means clustering requires only a set of unlabeled points and a threshold: the algorithm will take unlabeled points and gradually learn how to cluster them into groups by computing the mean of the distance between different points.

The critical difference here is that KNN needs labeled points and is thus supervised learning, while k-means doesn't—and is thus unsupervised learning.

Q4: Explain how a ROC curve works.

**Answer:** The ROC curve is a graphical representation of the contrast between true positive rates and the false positive rate at various thresholds. It's often used as a proxy for the trade-off between the sensitivity of the model (true positives) vs the fall-out or the probability it will trigger a false alarm (false positives).

Q5: Define precision and recall.

**Answer:** Recall is also known as the true positive rate: the amount of positives your model claims compared to the actual number of positives there are throughout the data. Precision is also known as the positive predictive value, and it is a measure of the amount of accurate positives your model claims

compared to the number of positives it actually claims. It can be easier to think of recall and precision in the context of a case where you've predicted that there were 10 apples and 5 oranges in a case of 10 apples. You'd have perfect recall (there are actually 10 apples, and you predicted there would be 10) but 66.7% precision because out of the 15 events you predicted, only 10 (the apples) are correct.

**Explanation:** Out of a sample size of 15 (10 apples + 5 oranges), you have identified 10 apples as apples BUT you have also incorrectly predicted 5 oranges as apples. This implies that the true positive figure is 10 (10 correctly identified apples), whereas the false positive figure is 5 (5 oranges incorrectly tagged as apples).

As per the formula of Precision = True Positive / (True Positive + False Positive), therefore the precision rate is 67%.

As per the Recall formula = True Positive / (True Positive + False Negative), hence the recall rate is 100%. This is because not a single apple was incorrectly predicted as an orange.

$$\text{Recall} = 10 / 10 + 0 = 100\%$$

$$\text{Precision} = 10 / 10 + 5 = 67\%$$

		ACTUAL CASE	
		APPLES	ORANGES
PREDICTED CASE	APPLES	TRUE POSITIVE 10	FALSE POSITIVE 5
	ORANGES	FALSE NEGATIVE 0	TRUE NEGATIVE  We don't particularly care for true negatives when calculating precision or recall as it isn't a part of the formula

## Q6: What is Bayes' Theorem? How is it useful in a machine learning context?

**Answer:** Bayes' Theorem gives you the posterior probability of an event given what is known as prior knowledge.

Mathematically, it's expressed as the true positive rate of a condition sample divided by the sum of the false positive rate of the population and the true

positive rate of a condition. Say you had a 60% chance of actually having the flu after a flu test, but out of people who had the flu, the test will be false 50% of the time, and the overall population only has a 5% chance of having the flu. Would you actually have a 60% chance of having the flu after having a positive test?

Bayes' Theorem says no. It says that you have a  $(.6 * 0.05) / (.6 * 0.05 + .5 * 0.95)$  = 0.0594 or 5.94% chance of getting a flu.

Bayes' Theorem is the basis behind a branch of machine learning that most notably includes the Naive Bayes classifier. That's something important to consider when you're faced with machine learning interview questions.

### **Q7: Why is “Naive” Bayes naive?**

**Answer:** Despite its practical applications, especially in text mining, Naive Bayes is considered “Naive” because it makes an assumption that is virtually impossible to see in real-life data: the conditional probability is calculated as the pure product of the individual probabilities of components. This implies the absolute independence of features — a condition probably never met in real life.

As a Quora commenter put it whimsically, a Naive Bayes classifier that figured out that you liked pickles and ice cream would probably naively recommend you a pickle ice cream.

### **Q8: Explain the difference between L1 and L2 regularization.**

**Answer:** L2 regularization tends to spread error among all the terms, while L1 is more binary/sparse, with many variables either being assigned a 1 or 0 in weighting. L1 corresponds to setting a Laplacean prior on the terms, while L2 corresponds to a Gaussian prior.

### **Q9: What’s your favorite algorithm, and can you explain it to me in less than a minute?**

**Answer:** Interviewers ask such machine learning interview questions to test your understanding of how to communicate complex and technical nuances with poise and the ability to summarize quickly and efficiently. While answering such questions, make sure you have a choice and ensure you can explain different algorithms so simply and effectively that a five-year-old could grasp the basics!

## **Q10: What's the difference between Type I and Type II error?**

**Answer:** Don't think that this is a trick question! Many machine learning interview questions will be an attempt to lob basic questions at you just to make sure you're on top of your game and you've prepared all of your bases.

Type I error is a false positive, while Type II error is a false negative. Briefly stated, Type I error means claiming something has happened when it hasn't, while Type II error means that you claim nothing is happening when in fact something is.

A clever way to think about this is to think of Type I error as telling a man he is pregnant, while Type II error means you tell a pregnant woman she isn't carrying a baby.

## **Q11: What's a Fourier transform?**

**Answer:** A Fourier transform is a generic method to decompose generic functions into a superposition of symmetric functions. Or as this more intuitive tutorial puts it, given a smoothie, it's how we find the recipe. The Fourier transform finds the set of cycle speeds, amplitudes, and phases to match any time signal. A Fourier transform converts a signal from time to frequency domain—it's a very common way to extract features from audio signals or other time series such as sensor data.

## **Q12: What's the difference between probability and likelihood?**

The term "probability" refers to the possibility of something happening. The term Likelihood refers to the process of determining the best data distribution given a specific situation in the data.

## **Q13: What is deep learning, and how does it contrast with other machine learning algorithms?**

**Answer:** Deep learning is a subset of machine learning that is concerned with neural networks: how to use backpropagation and certain principles from neuroscience to more accurately model large sets of unlabelled or semi-structured data. In that sense, deep learning represents an unsupervised learning algorithm that learns representations of data through the use of neural nets.

## **Q14: What's the difference between a generative and discriminative model?**

**Answer:** A generative model will learn categories of data while a discriminative model will simply learn the distinction between different categories of data. Discriminative models will generally outperform generative models on classification tasks.

### **Q15: What cross-validation technique would you use on a time series dataset?**

**Answer:** Instead of using standard k-folds cross-validation, you have to pay attention to the fact that a time series is not randomly distributed data—it is inherently ordered by chronological order. If a pattern emerges in later time periods, for example, your model may still pick up on it even if that effect doesn't hold in earlier years!

You'll want to do something like forward chaining where you'll be able to model on past data then look at forward-facing data.

- Fold 1 : training [1], test [2]
- Fold 2 : training [1 2], test [3]
- Fold 3 : training [1 2 3], test [4]
- Fold 4 : training [1 2 3 4], test [5]
- Fold 5 : training [1 2 3 4 5], test [6]

### **Q16: How is a decision tree pruned?**

**Answer:** Pruning is what happens in decision trees when branches that have weak predictive power are removed in order to reduce the complexity of the model and increase the predictive accuracy of a decision tree model. Pruning can happen bottom-up and top-down, with approaches such as reduced error pruning and cost complexity pruning.

Reduced error pruning is perhaps the simplest version: replace each node. If it doesn't decrease predictive accuracy, keep it pruned. While simple, this heuristic actually comes pretty close to an approach that would optimize for maximum accuracy.

### **Q17: Which is more important to you: model accuracy or model performance?**

**Answer:** Such machine learning interview questions tests your grasp of the nuances of machine learning model performance! Machine learning interview questions often look towards the details. There are models with higher accuracy that can perform worse in predictive power—how does that make sense?

Well, it has everything to do with how model accuracy is only a subset of model performance, and at that, a sometimes misleading one. For example, if you wanted to detect fraud in a massive dataset with a sample of millions, a more accurate model would most likely predict no fraud at all if only a vast minority of cases were fraud. However, this would be useless for a predictive model—a model designed to find fraud that asserted there was no fraud at all! Questions like this help you demonstrate that you understand model accuracy isn't the be-all and end-all of model performance.

### **Q18: What's the F1 score? How would you use it?**

**Answer:** The F1 score is a measure of a model's performance. It is a weighted average of the precision and recall of a model, with results tending to 1 being the best, and those tending to 0 being the worst. You would use it in classification tests where true negatives don't matter much.

### **Q19: How would you handle an imbalanced dataset?**

**Answer:** An imbalanced dataset is when you have, for example, a classification test and 90% of the data is in one class. That leads to problems: an accuracy of 90% can be skewed if you have no predictive power on the other category of data! Here are a few tactics to get over the hump:

1. Collect more data to even the imbalances in the dataset.
2. Resample the dataset to correct for imbalances.
3. Try a different algorithm altogether on your dataset.

What's important here is that you have a keen sense for what damage an unbalanced dataset can cause, and how to balance that.

### **Q20: When should you use classification over regression?**

**Answer:** Classification produces discrete values and dataset to strict categories, while regression gives you continuous results that allow you to better distinguish differences between individual points. You would use classification over regression if you wanted your results to reflect the belongingness of data points in your dataset to certain explicit categories (ex: If you wanted to know whether a name was male or female rather than just how correlated they were with male and female names.)

### **Q21: Name an example where ensemble techniques might be useful.**

**Answer:** Ensemble techniques use a combination of learning algorithms to optimize better predictive performance. They typically reduce overfitting in

models and make the model more robust (unlikely to be influenced by small changes in the training data).

You could list some examples of ensemble methods (bagging, boosting, the “bucket of models” method) and demonstrate how they could increase predictive power.

## **Q22: How do you ensure you’re not overfitting with a model?**

**Answer:** This is a simple restatement of a fundamental problem in machine learning: the possibility of overfitting training data and carrying the noise of that data through to the test set, thereby providing inaccurate generalizations.

There are three main methods to avoid overfitting:

1. Keep the model simpler: reduce variance by taking into account fewer variables and parameters, thereby removing some of the noise in the training data.
2. Use cross-validation techniques such as k-folds cross-validation.
3. Use regularization techniques such as LASSO that penalize certain model parameters if they’re likely to cause overfitting.

## **Q23: What evaluation approaches would you work to gauge the effectiveness of a machine learning model?**

**Answer:** You would first split the dataset into training and test sets, or perhaps use cross-validation techniques to further segment the dataset into composite sets of training and test sets within the data. You should then implement a choice selection of performance metrics: here is a fairly comprehensive list. You could use measures such as the F1 score, the accuracy, and the confusion matrix. What’s important here is to demonstrate that you understand the nuances of how a model is measured and how to choose the right performance measures for the right situations.

## **Q24: How would you evaluate a logistic regression model?**

**Answer:** A subsection of the question above. You have to demonstrate an understanding of what the typical goals of a logistic regression are (classification, prediction, etc.) and bring up a few examples and use cases.

## **Q25: What’s the “kernel trick” and how is it useful?**

**Answer:** The Kernel trick involves kernel functions that can enable in higher-dimension spaces without explicitly calculating the coordinates of points within

that dimension; instead, kernel functions compute the inner products between the images of all pairs of data in a feature space. This allows them the very useful attribute of calculating the coordinates of higher dimensions while being computationally cheaper than the explicit calculation of said coordinates. Many algorithms can be expressed in terms of inner products. Using the kernel trick enables us effectively run algorithms in a high-dimensional space with lower-dimensional data.

## **Q26: How do you handle missing or corrupted data in a dataset?**

**Answer:** You could find missing/corrupted data in a dataset and either drop those rows or columns, or decide to replace them with another value.

In Pandas, there are two very useful methods: `isnull()` and `dropna()` that will help you find columns of data with missing or corrupted data and drop those values. If you want to fill the invalid values with a placeholder value (for example, 0), you could use the `fillna()` method.

## **Q27: Do you have experience with Spark or big data tools for machine learning?**

**Answer:** You'll want to get familiar with the meaning of big data for different companies and the different tools they'll want. Spark is the big data tool most in demand now, able to handle immense datasets with speed. Be honest if you don't have experience with the tools demanded, but also take a look at job descriptions and see what tools pop up: you'll want to invest in familiarizing yourself with them.

## **Q28: Pick an algorithm. Write the pseudo-code for a parallel implementation.**

**Answer:** This kind of question demonstrates your ability to think in parallelism and how you could handle concurrency in programming implementations dealing with big data. Take a look at pseudocode frameworks such as Peril-L and visualization tools such as Web Sequence Diagrams to help you demonstrate your ability to write code that reflects parallelism.

## **Q29: What are some differences between a linked list and an array?**

**Answer:** An array is an ordered collection of objects. A linked list is a series of objects with pointers that direct how to process them sequentially. An array assumes that every element has the same size, unlike the linked list. A linked list can more easily grow organically: an array has to be pre-defined or re-defined for organic growth. Shuffling a linked list involves changing which

points direct where—meanwhile, shuffling an array is more complex and takes more memory.

### **Q30: Describe a hash table.**

**Answer:** A hash table is a data structure that produces an associative array. A key is mapped to certain values through the use of a hash function. They are often used for tasks such as database indexing.

### **Q31: Which data visualization libraries do you use? What are your thoughts on the best data visualization tools?**

**Answer:** What's important here is to define your views on how to properly visualize data and your personal preferences when it comes to tools. Popular tools include R's ggplot, Python's seaborn and matplotlib, and tools such as Plot.ly and Tableau.

### **Q32: Given two strings, A and B, of the same length n, find whether it is possible to cut both strings at a common point such that the first part of A and the second part of B form a palindrome.**

**Answer:** You'll often get standard algorithms and data structures questions as part of your interview process as a machine learning engineer that might feel akin to a software engineering interview. In this case, this comes from Google's interview process. There are multiple ways to check for palindromes—one way of doing so if you're using a programming language such as Python is to reverse the string and check to see if it still equals the original string, for example. The thing to look out for here is the category of questions you can expect, which will be akin to software engineering questions that drill down to your knowledge of algorithms and data structures. Make sure that you're totally comfortable with the language of your choice to express that logic.

### **Q33: How are primary and foreign keys related in SQL?**

**Answer:** Most machine learning engineers are going to have to be conversant with a lot of different data formats. SQL is still one of the key ones used. Your ability to understand how to manipulate SQL databases will be something you'll most likely need to demonstrate. In this example, you can talk about how foreign keys allow you to match up and join tables together on the primary key of the corresponding table—but just as useful is to talk through how you would think about setting up SQL tables and querying them.

### **Q34: How does XML and CSVs compare in terms of size?**

**Answer:** In practice, XML is much more verbose than CSVs are and takes up a lot more space. CSVs use some separators to categorize and organize data into neat columns. XML uses tags to delineate a tree-like structure for key-value pairs. You'll often get XML back as a way to semi-structure data from APIs or HTTP responses. In practice, you'll want to ingest XML data and try to process it into a usable CSV. This sort of question tests your familiarity with data wrangling sometimes messy data formats.

### **Q35: What are the data types supported by JSON?**

**Answer:** This tests your knowledge of JSON, another popular file format that wraps with JavaScript. There are six basic JSON datatypes you can manipulate: strings, numbers, objects, arrays, booleans, and null values.

### **Q36: How would you build a data pipeline?**

**Answer:** Data pipelines are the bread and butter of machine learning engineers, who take data science models and find ways to automate and scale them. Make sure you're familiar with the tools to build data pipelines (such as Apache Airflow) and the platforms where you can host models and pipelines (such as Google Cloud or AWS or Azure). Explain the steps required in a functioning data pipeline and talk through your actual experience building and scaling them in production

#### **Q1. What are the different types of Machine Learning?**

	Supervised Learning	Unsupervised Learning	Reinforcement Learning
Definition	The machine learns by using labelled data	The machine is trained on labelled data without any guidance	An agent interacts with its environment by producing actions & discovers errors or rewards
Types of Problems	Regression or Classification	Association or Classification	Reward Based
Types of Data	Labelled Data	Unlabelled Data	No pre-defined data
Training	External Supervision	No Supervision	No Supervision
Approach	Map Labelled input to known output	Understand pattern and discover output	Follow trail and error method

Popular Algorithms	Linear regression, Logistic regression, SVM, KNN, etc	K-means, C-means, etc	Q-Learning, SARSA, etc
--------------------	--	-----------------------	------------------------

## *Types of Machine Learning*

There are three ways in which machines learn:

1. Supervised Learning
2. Unsupervised Learning
3. Reinforcement Learning

### **Supervised Learning:**

Supervised learning is a method in which the machine learns using labeled data.

- It is like learning under the guidance of a teacher
- Training dataset is like a teacher which is used to train the machine
- Model is trained on a pre-defined dataset before it starts making decisions when given new data

### **Unsupervised Learning:**

Unsupervised learning is a method in which the machine is trained on unlabelled data or without any guidance

- It is like learning without a teacher.
- Model learns through observation & finds structures in data.
- Model is given a dataset and is left to automatically find patterns and relationships in that dataset by creating clusters.

### **Reinforcement Learning:**

Reinforcement learning involves an agent that interacts with its environment by producing actions & discovers errors or rewards.

- It is like being stuck in an isolated island, where you must explore the environment and learn how to live and adapt to the living conditions on your own.
- Model learns through the hit and trial method
- It learns on the basis of reward or penalty given for every action it performs

## **Q2. How would you explain Machine Learning to a school-going kid?**

- Suppose your friend invites you to his party where you meet total strangers. Since you have no idea about them, you will mentally classify them on the basis of gender, age group, dressing, etc.
- In this scenario, the strangers represent unlabeled data and the process of classifying unlabeled data points is nothing but unsupervised learning.
- Since you didn't use any prior knowledge about people and classified them on-the-go, this becomes an unsupervised learning problem.

## **Q3. How does Deep Learning differ from Machine Learning?**

Deep Learning	Machine Learning
 <p><i>Deep Learning is a form of machine learning that is inspired by the structure of the human brain and is particularly effective in feature detection.</i></p>	 <p><i>Machine Learning is all about algorithms that take in data, learn from that data, and then apply what they've learned to make informed decisions.</i></p>

### *Deep Learning vs Machine Learning*

## **Q4. Explain Classification and Regression**

Classification	Regression
<ul style="list-style-type: none"><li>• Classification is the task of predicting a discrete class label</li><li>• In a classification problem data is labelled into one of two or more classes</li><li>• A classification problem with two classes is called binary, more than two classes is called a multi-class classification</li><li>• Classifying an email as spam or non-spam is an example of a classification problem</li></ul>	<ul style="list-style-type: none"><li>• Regression is the task of predicting a continuous quantity</li><li>• A regression problem requires the prediction of a quantity</li><li>• A regression problem with multiple input variables is called a multivariate regression problem</li><li>• Predicting the price of a stock over a period of time is a regression problem</li></ul>

## **Q5. What do you understand by selection bias?**

- It is a statistical error that causes a bias in the sampling portion of an experiment.

- The error causes one sampling group to be selected more often than other groups included in the experiment.
- Selection bias may produce an inaccurate conclusion if the selection bias is not identified.

## Q6. What do you understand by Precision and Recall?

Let me explain you this with an analogy:

- Imagine that, your girlfriend gave you a birthday surprise every year for the last 10 years. One day, your girlfriend asks you: ‘Sweetie, do you remember all the birthday surprises from me?’
- To stay on good terms with your girlfriend, you need to recall all the 10 events from your memory. Therefore, **recall** is the ratio of the number of events you can correctly recall, to the total number of events.
- If you can recall all 10 events correctly, then, your recall ratio is 1.0 (100%) and if you can recall 7 events correctly, your recall ratio is 0.7 (70%)

However, you might be wrong in some answers.

- For example, let's assume that you took 15 guesses out of which 10 were correct and 5 were wrong. This means that you can recall all events but not so precisely
- Therefore, **precision** is the ratio of a number of events you can correctly recall, to the total number of events you can recall (mix of correct and wrong recalls).
- From the above example (10 real events, 15 answers: 10 correct, 5 wrong), you get 100% recall but your precision is only 66.67% (10 / 15)

## Q7. Explain false negative, false positive, true negative and true positive with a simple example.

Let's consider a scenario of a fire emergency:

- **True Positive:** If the alarm goes on in case of a fire.  
*Fire is positive and prediction made by the system is true.*
- **False Positive:** If the alarm goes on, and there is no fire.  
*System predicted fire to be positive which is a wrong prediction, hence the prediction is false.*
- **False Negative:** If the alarm does not ring but there was a fire.  
*System predicted fire to be negative which was false since there was fire.*
- **True Negative:** If the alarm does not ring and there was no fire.  
*The fire is negative and this prediction was true.*

## Q8. What is a Confusion Matrix?

A confusion matrix or an error matrix is a table which is used for summarizing the performance of a classification algorithm.

n=165	Predicted:		
	NO	YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

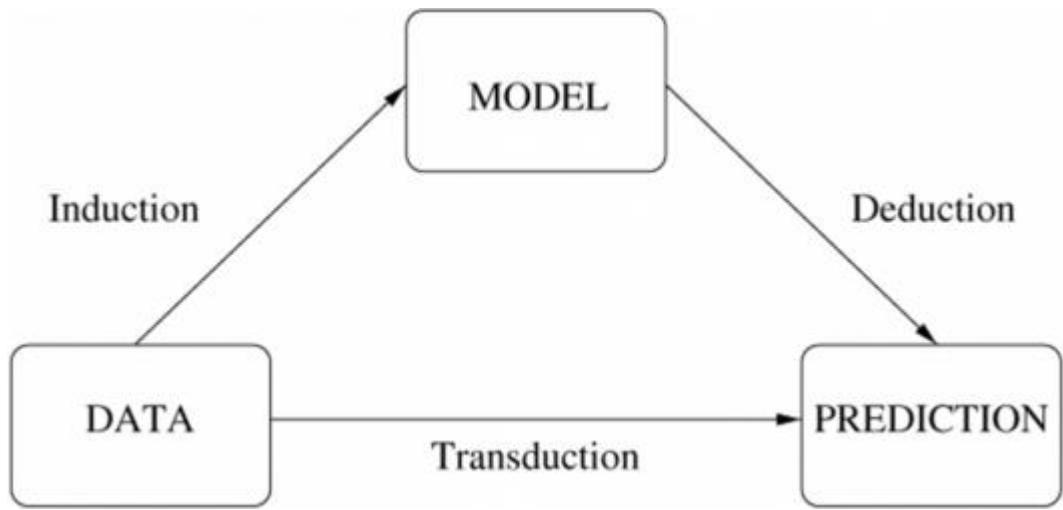
Confusion Matrix –

Consider the above table where:

- TN = True Negative
- TP = True Positive
- FN = False Negative
- FP = False Positive

## Q9. What is the difference between inductive and deductive learning?

- *Inductive learning is the process of using observations to draw conclusions*
- *Deductive learning is the process of using conclusions to form observations*



*Inductive vs Deductive*

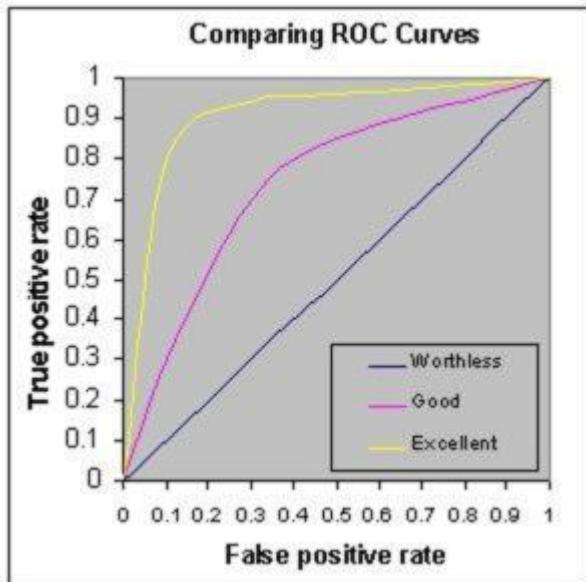
#### **Q10. How is KNN different from K-means clustering?**

K-Nearest Neighbour	K-Means Clustering
▪ Supervised Technique	▪ Unsupervised Technique
▪ Used for Classification or Regression	▪ Used for Clustering
▪ 'K' in KNN represents the number of nearest neighbours used to classify or predict in case of continuous variable/regression	▪ 'K' in K-Means represents the number of clusters the algorithm is trying to identify or learn from the data

*K-means vs KNN*

#### **Q11. What is ROC curve and what does it represent?**

*Receiver Operating Characteristic curve (or ROC curve) is a fundamental tool for diagnostic test evaluation and is a plot of the true positive rate (Sensitivity) against the false positive rate (Specificity) for the different possible cut-off points of a diagnostic test.*



## *ROC*

It shows the tradeoff between sensitivity and specificity (any increase in sensitivity will be accompanied by a decrease in specificity).

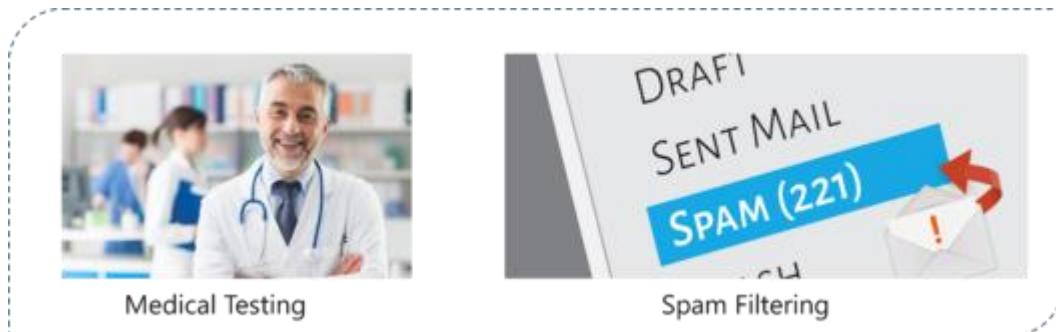
- The closer the curve follows the left-hand border and then the top border of the ROC space, the more accurate the test.
- The closer the curve comes to the 45-degree diagonal of the ROC space, the less accurate the test.
- The slope of the tangent line at a cutpoint gives the likelihood ratio (LR) for that value of the test.
- The area under the curve is a measure of test accuracy.

## **Q12. What's the difference between Type I and Type II error?**

Type I Error	Type II Error
<ul style="list-style-type: none"> <li>• Type I error is a false positive.</li> <li>• Type I error is claiming something has happened when it hasn't.</li> </ul>	<ul style="list-style-type: none"> <li>• Type II error is a false negative.</li> <li>• Type II error is claiming nothing when in fact something has happened.</li> </ul>

## *Type 1 vs Type 2 Error*

## **Q13. Is it better to have too many false positives or too many false negatives? Explain.**



### *False Negatives vs False Positives*

It depends on the question as well as on the domain for which we are trying to solve the problem. If you're using Machine Learning in the domain of medical testing, then a false negative is very risky, since the report will not show any health problem when a person is actually unwell. Similarly, if Machine Learning is used in spam detection, then a false positive is very risky because the algorithm may classify an important email as spam.

### **Q14. Which is more important to you – model accuracy or model performance?**



### *Model Accuracy vs Performance – Machine Learning Interview Questions – Edureka*

Well, you must know that model accuracy is only a subset of model performance. The accuracy of the model and performance of the model are directly proportional and hence better the performance of the model, more accurate are the predictions.

## **Q15. What is the difference between Gini Impurity and Entropy in a Decision Tree?**

- Gini Impurity and Entropy are the metrics used for deciding how to split a Decision Tree.
- Gini measurement is the probability of a random sample being classified correctly if you randomly pick a label according to the distribution in the branch.
- Entropy is a measurement to calculate the lack of information. You calculate the Information Gain (difference in entropies) by making a split. This measure helps to reduce the uncertainty about the output label.

## **Q16. What is the difference between Entropy and Information Gain?**

- Entropy is an indicator of how messy your data is. It decreases as you reach closer to the leaf node.
- The Information Gain is based on the decrease in entropy after a dataset is split on an attribute. It keeps on increasing as you reach closer to the leaf node.

## **Q17. What is Overfitting? And how do you ensure you're not overfitting with a model?**

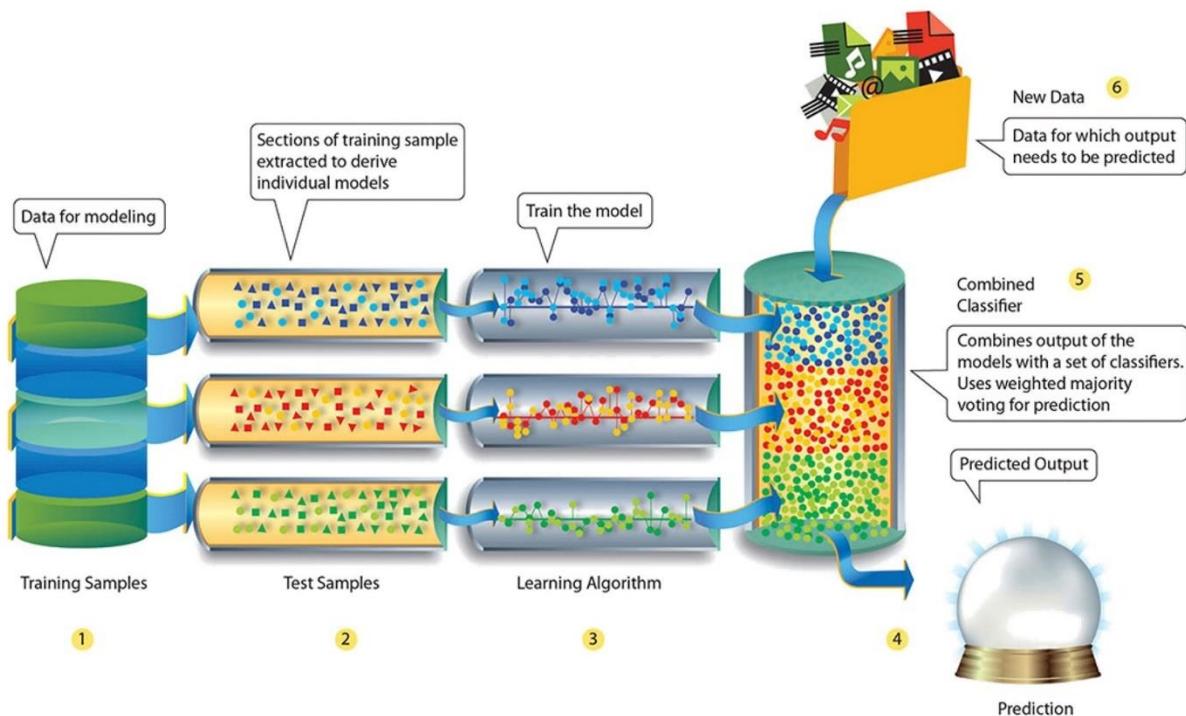
*Over-fitting occurs when a model studies the training data to such an extent that it negatively influences the performance of the model on new data.*

This means that the disturbance in the training data is recorded and learned as concepts by the model. But the problem here is that these concepts do not apply to the testing data and negatively impact the model's ability to classify the new data, hence reducing the accuracy on the testing data.

Three main methods to avoid overfitting:

- Collect more data so that the model can be trained with varied samples.
- Use ensembling methods, such as Random Forest. It is based on the idea of bagging, which is used to reduce the variation in the predictions by combining the result of multiple Decision trees on different samples of the data set.
- Choose the right algorithm.

## **Q18.Explain Ensemble learning technique in Machine Learning.**



Ensemble learning is a technique that is used to create multiple Machine Learning models, which are then combined to produce more accurate results. A general Machine Learning model is built by using the entire training data set. However, in Ensemble Learning the training data set is split into multiple subsets, wherein each subset is used to build a separate model. After the models are trained, they are then combined to predict an outcome in such a way that the variance in the output is reduced.

## Q19. What is bagging and boosting in Machine Learning?

Similarities	Difference
▪ Both are ensemble methods to get N learners from 1 learner	▪ While they are built independently for Bagging, Boosting tries to add new models that do well where previous models fall.
▪ Both generate several training data sets by random sampling	▪ Only Boosting determines weight for the data to tip the scales in favour of the most difficult cases
▪ Both make the final decision by taking the average of N learners	▪ Is an equally average for Bagging and a weighted average for Boosting more weight in those with better performance on training data
▪ Both are good at reducing variance and proving higher scalability	▪ Only Boosting tries to reduce bias. On the other hand, Bagging may solve the problem of over-fitting, while boosting can increase it

## Q20. How would you screen for outliers and what should you do if you find one?

The following methods can be used to screen outliers:

- Boxplot:** A box plot represents the distribution of the data and its variability. The box plot contains the upper and lower quartiles, so the box basically spans the Inter-Quartile Range (IQR). One of the main reasons why box plots are used is to detect outliers in the data. Since the box plot spans the IQR, it detects the data points that lie outside this range. These data points are nothing but outliers.
- Probabilistic and statistical models:** Statistical models such as normal distribution and exponential distribution can be used to detect any variations in the distribution of data points. If any data point is found outside the distribution range, it is rendered as an outlier.
- Linear models:** Linear models such as logistic regression can be trained to flag outliers. In this manner, the model picks up the next outlier it sees.
- Proximity-based models:** An example of this kind of model is the K-means clustering model wherein, data points form multiple or 'k' number of clusters based on features such as similarity or distance. Since similar data points form clusters, the outliers also form their own cluster. In this way, proximity-based models can easily help detect outliers.

*How do you handle these outliers?*

- If your data set is huge and rich then you can risk dropping the outliers.
- However, if your data set is small then you can cap the outliers, by setting a threshold percentile. For example, the data points that are above the 95th percentile can be used to cap the outliers.
- Lastly, based on the data exploration stage, you can narrow down some rules and impute the outliers based on those business rules.

## **Q21. What are collinearity and multicollinearity?**

- Collinearity occurs when two predictor variables (e.g., x1 and x2) in a multiple regression have some correlation.
- Multicollinearity occurs when more than two predictor variables (e.g., x1, x2, and x3) are inter-correlated.

## **Q22. What do you understand by Eigenvectors and Eigenvalues?**

- **Eigenvectors:** *Eigenvectors are those vectors whose direction remains unchanged even when a linear transformation is performed on them.*
- **Eigenvalues:** *Eigenvalue is the scalar that is used for the transformation of an Eigenvector.*

$$\begin{bmatrix} 3 & 4 & -2 \\ 1 & 4 & -1 \\ 2 & 6 & -1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \\ 6 \end{bmatrix}$$

$$\begin{bmatrix} 3 & 4 & -2 \\ 1 & 4 & -1 \\ 2 & 6 & -1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} = 3 \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}$$

*Eigenvalue & Eigenvectors – Machine Learning Interview Questions – Edureka*

In the above example, 3 is an Eigenvalue, with the original vector in the multiplication problem being an eigenvector.

The Eigenvector of a square matrix A is a nonzero vector x such that for some number  $\lambda$ , we have the following:

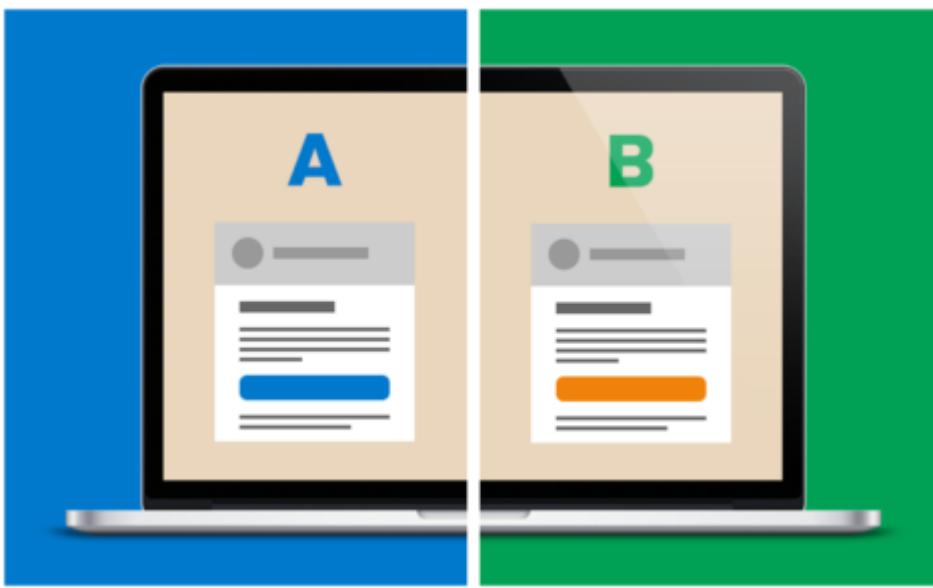
$$Ax = \lambda x,$$

where  $\lambda$  is an Eigenvalue

So, in our example,  $\lambda = 3$  and  $X = [1 \ 1 \ 2]$

### **Q23. What is A/B Testing?**

- A/B is Statistical hypothesis testing for randomized experiment with two variables A and B. It is used to compare two models that use different predictor variables in order to check which variable fits best for a given sample of data.
- Consider a scenario where you've created two models (using different predictor variables) that can be used to recommend products for an e-commerce platform.
- A/B Testing can be used to compare these two models to check which one best recommends products to a customer.



*A/B Testing – Machine Learning Interview Questions – Edureka*

#### **Q24. What is Cluster Sampling?**

- It is a process of randomly selecting intact groups within a defined population, sharing similar characteristics.
- Cluster Sample is a probability sample where each sampling unit is a collection or cluster of elements.
- For example, if you're clustering the total number of managers in a set of companies, in that case, managers (samples) will represent elements and companies will represent clusters.

#### **Q25. Running a binary classification tree algorithm is quite easy. But do you know how the tree decides on which variable to split at the root node and its succeeding child nodes?**

- Measures such as, Gini Index and Entropy can be used to decide which variable is best fitted for splitting the Decision Tree at the root node.
- We can calculate Gini as following:  
Calculate Gini for sub-nodes, using the formula – sum of square of probability for success and failure ( $p^2+q^2$ ).
- Calculate Gini for split using weighted Gini score of each node of that split
- Entropy is the measure of impurity or randomness in the data, (for binary class):

$$\text{Entropy} = -p \log_2 p - q \log_2 q$$

Here p and q is the probability of success and failure respectively in that node.

- Entropy is zero when a node is homogeneous and is maximum when both the classes are present in a node at 50% – 50%. To sum it up, the entropy must be as low as possible in order to decide whether or not a variable is suitable as the root node.

## **Q1. Name a few libraries in Python used for Data Analysis and Scientific Computations.**

Here is a list of Python libraries mainly used for Data Analysis:

- NumPy
- SciPy
- Pandas
- SciKit
- Matplotlib
- Seaborn
- Bokeh

## **Q2. Which library would you prefer for plotting in Python language: Seaborn or Matplotlib or Bokeh?**



*Python Libraries – Machine Learning Interview Questions – Edureka*

It depends on the visualization you're trying to achieve. Each of these libraries is used for a specific purpose:

- **Matplotlib:** Used for basic plotting like bars, pies, lines, scatter plots, etc
- **Seaborn:** Is built on top of Matplotlib and Pandas to ease data plotting. It is used for statistical visualizations like creating heatmaps or showing the distribution of your data

- **Bokeh:** Used for interactive visualization. In case your data is too complex and you haven't found any "message" in the data, then use Bokeh to create interactive visualizations that will allow your viewers to explore the data themselves

### Q3. How are NumPy and SciPy related?

- NumPy is part of SciPy.
- NumPy defines arrays along with some basic numerical functions like indexing, sorting, reshaping, etc.
- SciPy implements computations such as numerical integration, optimization and machine learning using NumPy's functionality.

### Q4. What is the main difference between a Pandas series and a single-column DataFrame in Python?

**Pandas Data Structures**

**Series**

A one-dimensional labeled array capable of holding any data type

```
>>> s = pd.Series([3, -5, 7, 4], index=['a', 'b', 'c', 'd'])
```

**DataFrame**

Columns

	Country	Capital	Population
1	Belgium	Brussels	11190846
2	India	New Delhi	1303171035
3	Brazil	Brasilia	207847528

Index

A two-dimensional labeled data structure with columns of potentially different types

```
>>> data = {'Country': ['Belgium', 'India', 'Brazil'],
   'Capital': ['Brussels', 'New Delhi', 'Brasilia'],
   'Population': [11190846, 1303171035, 207847528]}

>>> df = pd.DataFrame(data,
   columns=['Country', 'Capital', 'Population'])
```

*Pandas Series vs DataFrame – Machine Learning Interview Questions – Edureka*

## **Q5. How can you handle duplicate values in a dataset for a variable in Python?**

Consider the following Python code:

```
1 bill_data=pd.read_csv("datasetsTelecom Data AnalysisBill.csv")
2 bill_data.shape
3 #Identify duplicates records in the data
4 Dupes = bill_data.duplicated()
5 sum(dupes)
6 #Removing Duplicates
7 bill_data_uniq = bill_data.drop_duplicates()
```

## **Q6. Write a basic Machine Learning program to check the accuracy of a model, by importing any dataset using any classifier?**

```
1 #importing dataset
2 import sklearn
3 from sklearn import datasets
4 iris = datasets.load_iris()
5 X = iris.data
6 Y = iris.target
7
8 #splitting the dataset
9 from sklearn.cross_validation import train_test_split
10 X_train, Y_train, X_test, Y_test = train_test_split(X,Y, test_size = 0.5)
11
12 #Selecting Classifier
13 my_classifier = tree.DecisionTreeClassifier()
14 My_classifier.fit(X_train, Y_train)
15 predictions = my_classifier(X_test)
```

```
16 #check accuracy
17 From sklear.metrics import accuracy_score
18 print accuracy_score(y_test, predictions)
```

## Machine Learning Scenario Based Questions

This set of Machine Learning interview questions deal with scenario-based Machine Learning questions.

**Q1. You are given a data set consisting of variables having more than 30% missing values? Let's say, out of 50 variables, 8 variables have missing values higher than 30%. How will you deal with them?**

- Assign a unique category to the missing values, who knows the missing values might uncover some trend.
- We can remove them blatantly.
- Or, we can sensibly check their distribution with the target variable, and if found any pattern we'll keep those missing values and assign them a new category while removing others.

**Q2. Write an SQL query that makes recommendations using the pages that your friends liked. Assume you have two tables: a two-column table of users and their friends, and a two-column table of users and the pages they liked. It should not recommend pages you already like.**

```
1 SELECT f.user_id, l.page_id
2 FROM friend f JOIN like l
3 ON f.friend_id = l.user_id
4 WHERE l.page_id NOT IN (SELECT page_id FROM like
5 WHERE user_id = f.user_id)
```

**Q3. There's a game where you are asked to roll two fair six-sided dice. If the sum of the values on the dice equals seven, then you win \$21. However, you must pay \$5 to play each time you roll both dice. Do you play this game? And in the follow-up: If he plays 6 times what is the probability of making money from this game?**

- The first condition states that if the sum of the values on the 2 dices is equal to 7, then you win \$21. But for all the other cases you must pay \$5.
- First, let's calculate the number of possible cases. Since we have two 6-sided dices, the total number of cases =>  $6 \times 6 = 36$ .

- Out of 36 cases, we must calculate the number of cases that produces a sum of 7 (in such a way that the sum of the values on the 2 dices is equal to 7)
- Possible combinations that produce a sum of 7 is, (1,6), (2,5), (3,4), (4,3), (5,2) and (6,1). All these 6 combinations generate a sum of 7.
- This means that out of 36 chances, only 6 will produce a sum of 7. On taking the ratio, we get:  $6/36 = 1/6$
- So this suggests that we have a chance of winning \$21, once in 6 games.
- So to answer the question if a person plays 6 times, he will win one game of \$21, whereas for the other 5 games he will have to pay \$5 each, which is \$25 for all five games. Therefore, he will face a loss because he wins \$21 but ends up paying \$25.

**Q4. We have two options for serving ads within Newsfeed:**

**1 – out of every 25 stories, one will be an ad**

**2 – every story has a 4% chance of being an ad**

**For each option, what is the expected number of ads shown in 100 news stories?**

**If we go with option 2, what is the chance a user will be shown only a single ad in 100 stories? What about no ads at all?**

- The expected number of ads shown in 100 new stories for option 1 is equal to 4 ( $100/25 = 4$ ).
- Similarly, for option 2, the expected number of ads shown in 100 new stories is also equal to 4 ( $4/100 = 1/25$  which suggests that one out of every 25 stories will be an ad, therefore in 100 new stories there will be 4 ads)
- Therefore for each option, the total number of ads shown in 100 new stories is 4.
- The second part of the question can be solved by using Binomial distribution. Binomial distribution takes three parameters:
  - The probability of success and failure, which in our case is 4%.
  - The total number of cases, which is 100 in our case.
  - The probability of the outcome, which is a chance that a user will be shown only a single ad in 100 stories
- $p(\text{single ad}) = (0.96)^{99} \cdot (0.04)^1$

(note: here 0.96 denotes the chance of not seeing an ad in 100 stories, 99 denotes the possibility of seeing only 1 ad, 0.04 is the probability of seeing an ad once in 100 stories )

- In total, there are 100 positions for the ad. Therefore,  $100 * p(\text{single ad}) = 7.03\%$

**Q5. How would you predict who will renew their subscription next month? What data would you need to solve this? What analysis would you do? Would you build predictive models? If so, which algorithms?**

- Let's assume that we're trying to predict renewal rate for Netflix subscription. So our problem statement is to predict which users will renew their subscription plan for the next month.
- Next, we must understand the data that is needed to solve this problem. In this case, we need to check the number of hours the channel is active for each household, the number of adults in the household, number of kids, which channels are streamed the most, how much time is spent on each channel, how much has the watch rate varied from last month, etc. Such data is needed to predict whether or not a person will continue the subscription for the upcoming month.
- After collecting this data, it is important that you find patterns and correlations. For example, we know that if a household has kids, then they are more likely to subscribe. Similarly, by studying the watch rate of the previous month, you can predict whether a person is still interested in a subscription. Such trends must be studied.
- The next step is analysis. For this kind of problem statement, you must use a classification algorithm that classifies customers into 2 groups:
  - Customers who are likely to subscribe next month
  - Customers who are not likely to subscribe next month
- Would you build predictive models? Yes, in order to achieve this you must build a predictive model that classifies the customers into 2 classes like mentioned above.
- Which algorithms to choose? You can choose classification algorithms such as Logistic Regression, Random Forest, Support Vector Machine, etc.
- Once you've opted the right algorithm, you must perform model evaluation to calculate the efficiency of the algorithm. This is followed by deployment.

**Q6. How do you map nicknames (Pete, Andy, Nick, Rob, etc) to real names?**

- This problem can be solved in n number of ways. Let's assume that you're given a data set containing 1000s of twitter interactions. You will begin by studying the relationship between two people by carefully analyzing the words used in the tweets.
- This kind of problem statement can be solved by implementing Text Mining using Natural Language Processing techniques, wherein each

word in a sentence is broken down and co-relations between various words are found.

- NLP is actively used in understanding customer feedback, performing sentimental analysis on Twitter and Facebook. Thus, one of the ways to solve this problem is through Text Mining and Natural Language Processing techniques.

**Q7. A jar has 1000 coins, of which 999 are fair and 1 is double headed. Pick a coin at random, and toss it 10 times. Given that you see 10 heads, what is the probability that the next toss of that coin is also a head?**

- There are two ways of choosing a coin. One is to pick a fair coin and the other is to pick the one with two heads.
- $\text{Probability of selecting fair coin} = 999/1000 = 0.999$   
 $\text{Probability of selecting unfair coin} = 1/1000 = 0.001$
- Selecting 10 heads in a row = Selecting fair coin \* Getting 10 heads + Selecting an unfair coin
- $P(A) = 0.999 * (1/2)^{10} = 0.999 * (1/1024) = 0.000976$   
 $P(B) = 0.001 * 1 = 0.001$   
 $P(A/A+B) = 0.000976 / (0.000976 + 0.001) = 0.4939$   
 $P(B/A+B) = 0.001 / 0.001976 = 0.5061$
- $\text{Probability of selecting another head} = P(A/A+B) * 0.5 + P(B/A+B) * 1$   
 $= 0.4939 * 0.5 + 0.5061 = 0.7531$

**Q8. Suppose you are given a data set which has missing values spread along 1 standard deviation from the median. What percentage of data would remain unaffected and Why?**

Since the data is spread across the median, let's assume it's a normal distribution.

As you know, in a normal distribution, ~68% of the data lies in 1 standard deviation from mean (or mode, median), which leaves ~32% of the data unaffected. Therefore, ~32% of the data would remain unaffected by missing values.

**Q9. You are given a cancer detection data set. Let's suppose when you build a classification model you achieved an accuracy of 96%. Why shouldn't you be happy with your model performance? What can you do about it?**

You can do the following:

- Add more data
- Treat missing outlier values

- Feature Engineering
- Feature Selection
- Multiple Algorithms
- Algorithm Tuning
- Ensemble Method
- Cross-Validation

**Q10. You are working on a time series data set. Your manager has asked you to build a high accuracy model. You start with the decision tree algorithm since you know it works fairly well on all kinds of data. Later, you tried a time series regression model and got higher accuracy than the decision tree model. Can this happen? Why?**

- Time series data is based on linearity while a decision tree algorithm is known to work best to detect non-linear interactions
- Decision tree fails to provide robust predictions. Why?
  - The reason is that it couldn't map the linear relationship as good as a regression model did.
  - We also know that a linear regression model can provide a robust prediction only if the data set satisfies its linearity assumptions.

**Q11. Suppose you found that your model is suffering from low bias and high variance. Which algorithm you think could tackle this situation and Why?**

*Type 1: How to tackle high variance?*

- Low bias occurs when the model's predicted values are near to actual values.
- In this case, we can use the bagging algorithm (eg: Random Forest) to tackle high variance problem.
- Bagging algorithm will divide the data set into its subsets with repeated randomized sampling.
- Once divided, these samples can be used to generate a set of models using a single learning algorithm. Later, the model predictions are combined using voting (classification) or averaging (regression).

*Type 2: How to tackle high variance?*

- Lower the model complexity by using regularization technique, where higher model coefficients get penalized.
- You can also use top n features from variable importance chart. It might be possible that with all the variable in the data set, the algorithm is facing difficulty in finding the meaningful signal.

**Q12. You are given a data set. The data set contains many variables, some of which are highly correlated and you know about it. Your manager has asked you to run PCA. Would you remove correlated variables first? Why?**

*Possibly, you might get tempted to say no, but that would be incorrect.*

Discarding correlated variables will have a substantial effect on PCA because, in the presence of correlated variables, the variance explained by a particular component gets inflated.

**Q13. You are asked to build a multiple regression model but your model R<sup>2</sup> isn't as good as you wanted. For improvement, you remove the intercept term now your model R<sup>2</sup> becomes 0.8 from 0.3. Is it possible? How?**

*Yes, it is possible.*

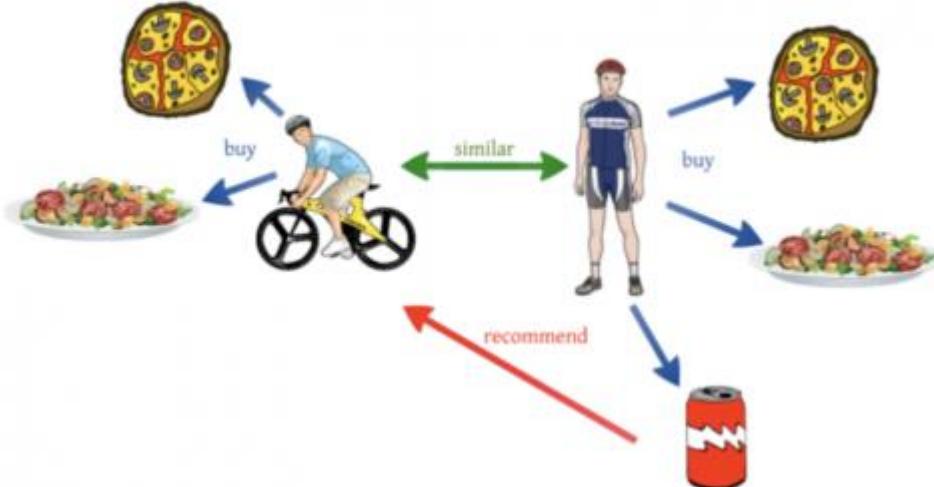
- The intercept term refers to model prediction without any independent variable or in other words, mean prediction  
 $R^2 = 1 - \sum(Y - Y')^2 / \sum(Y - Y_{\text{mean}})^2$  where  $Y'$  is the predicted value.
- In the presence of the intercept term,  $R^2$  value will evaluate your model with respect to the mean model.
- In the absence of the intercept term ( $Y_{\text{mean}}$ ), the model can make no such evaluation,
- With large denominator,  
Value of  $\sum(Y - Y')^2 / \sum(Y)^2$  equation becomes smaller than actual, thereby resulting in a higher value of  $R^2$ .

**Q14. You're asked to build a random forest model with 10000 trees. During its training, you got training error as 0.00. But, on testing the validation error was 34.23. What is going on? Haven't you trained your model perfectly?**

- The model is overfitting the data.
- Training error of 0.00 means that the classifier has mimicked the training data patterns to an extent.
- But when this classifier runs on the unseen sample, it was not able to find those patterns and returned the predictions with more number of errors.
- In Random Forest, it usually happens when we use a larger number of trees than necessary. Hence, to avoid such situations, we should tune the number of trees using cross-validation.

**Q15. 'People who bought this also bought...' recommendations seen on Amazon is based on which algorithm?**

E-commerce websites like Amazon make use of Machine Learning to recommend products to their customers. The basic idea of this kind of recommendation comes from collaborative filtering. Collaborative filtering is the process of comparing users with similar shopping behaviors in order to recommend products to a new user with similar shopping behavior.



### *Collaborative Filtering – Machine Learning Interview Questions – Edureka*

To better understand this, let's look at an example. Let's say a user A who is a sports enthusiast bought, pizza, pasta, and a coke. Now a couple of weeks later, another user B who rides a bicycle buys pizza and pasta. He does not buy the coke, but Amazon recommends a bottle of coke to user B since his shopping behaviors and his lifestyle is quite similar to user A. This is how collaborative filtering works.

### **1) What are the basic differences between Machine Learning and Deep Learning?**

Differences between Machine Learning and Deep Learning are:

	<b>Machine Learning</b>	<b>Deep Learning</b>
Definition	Sub-discipline of AI	A subset of machine learning
Data	Parses the data	Creates an artificial neural network

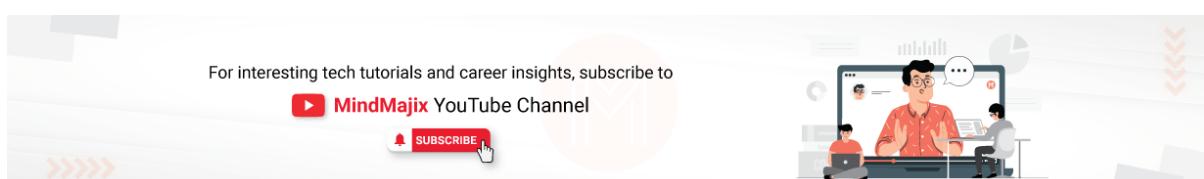
Accuracy	Requires manual intervention means decreased accuracy	Self-learning capability higher accuracy
Interpretability	Machine Learning is Faster	10 Times Faster than ML
Output	ML models produce a numerical output	DL algorithms can rank an image to text or even audio
Data dependencies	High	Low
Hardware dependencies	Can work on low-end machines.	Heavily depend on high machines
Future	Effective with image recognition and face recognition in mobiles	Not much effective due to processing limitations

## 2) What is the difference between Bias and Variance?

- **Bias:** Bias can be defined as a situation where an error has occurred due to the use of assumptions in the learning algorithm.
- **Variance:** Variance is an error caused because of the complexity of the algorithm that is been used to analyze the data.

## 3) What is the difference between supervised and unsupervised machine learning?

Supervised learning is a process where it requires training labeled data. When it comes to Unsupervised learning doesn't require data labeling.



#### **4) What are the three stages of model building in machine learning?**

Following are the three stages of model building:

- **Model Building:** In this stage, we will choose the ideal algorithm for the model, and we will train it based on our requirements.
- **Model Testing:** In this stage, we will check the model's accuracy by using test data.
- **Applying Model:** After testing, we have to make the changes, and then we can use the model for real-time projects.

#### **5) What are the applications of supervised machine learning?**

Following are the applications of machine learning:

1. **Fraud Identification:** Supervised learning trains the model for identifying the suspicious patterns; we can identify the feasible fraud instances.
2. **Healthcare:** By giving images about a disease, supervised machine learning can train the model for detecting whether a person is affected by illness or not.
3. **Email spam identification:** We train the model through historical data which contains emails that are classified as spam or not spam. This labeled data is supplied as the input to the model.
4. **Sentiment Analysis:** This relates to the process of using algorithms for mining the documents and determining if they are negative, neutral, positive in sentiment.

## *Related Article - Artificial Intelligence Vs Machine Learning*

### **6) What are the techniques of Unsupervised machine learning?**

Following are the different techniques of unsupervised machine learning:

1. **Clustering:** It includes the data that must be divided into subsets. These subsets are also known as clusters. Diverse clusters disclose details about objects, unlike regression or classification.
2. **Association:** In the association problem, we can recognize the association patterns between different items and variables. For instance, e-commerce can indicate other items for us to buy according to our previous purchases.

### **7) What are the different types of Machine Learning?**

Following are the different kinds of machine learning:

- **Unsupervised Learning:** In this kind of machine learning, we will not have labeled data. A model can recognize anomalies, relationships, and patterns in the input data.
- **Supervised Learning:** In this kind of machine learning, the model makes decisions or predictions according to the labeled or past data. Labeled data relates to data sets that provide labels or tags.
- **Reinforcement Learning:** In reinforcement learning, a model can learn according to the rewards it obtained from its past actions.

### **8) What is Deep Learning?**

Deep learning is a branch of machine learning which is relevant to neural networks. Deep learning tells us how to use the principles and backpropagation from neuroscience to the large sets of semi-structured or unlabelled data. Deep learning portrays the unsupervised learning algorithm which learns data representation by using neural nets

*Explore - Deep Learning Tools for more information*

## 9) How to Build a Data Pipeline?

Data pipelines are the core of the machine learning engineers, which take data science models and discover methods for scaling and automating them if you are accustomed to the tools for building the platforms and data pipelines where we can host pipelines and models.

## 10) How is KNN different from k-means clustering?

K-Nearest Neighbours is a supervised algorithm, and k-means clustering is an unsupervised algorithm. For the K-nearest neighbors to work, we require labeled data to classify the unlabeled point. K-means clustering needs only a threshold and a group of unlabeled points: the algorithm takes the unlabeled points and slowly learns how to divide them into groups by calculating the mean of distance between the points.

## 11) Comparision between Machine Learning and Big Data

Machine Learning Vs Big Data		
Feature	Machine Learning	Big Data
Data Use	Technology that helps in reducing human intervention.	Data research, especially when work huge data.
Operations	Existing data helps to teach machine what can be done further	Design patterns with analytics on existing data in terms of decision making.
Pattern Recognition	Similar to Big Data, existing data helps in pattern recognition.	Sequence and classification analysis for pattern recognition.
Data Volume	Best performance, while working with small datasets.	Datasets help in understanding and solving problems associated with large data.

## Machine Learning Vs Big Data

Application	Read existing data to predict future information.	Storing and analyzing patterns within data volumes.
-------------	---	---

### 12) Explain what is precision and Recall?

- **Recall:** It is known as a true positive rate. The number of positives that your model has claimed compared to the actual defined number of positives available throughout the data.
- **Precision:** It is also known as a positive predicted value. This is more based on the prediction. It is a measure of the number of accurate positives that the model claims when compared to the number of positives it actually claims.

### 13) What is your favorite algorithm and also explain the algorithm briefly in a minute?

This type of question is very common and asked by the interviewers to understand the candidate's skills and assess how well he can communicate complex theories in the simplest language.

This one is a tough question and usually, individuals are not at all prepared for this situation so please be prepared and have a choice of algorithms and make sure you practice a lot before going into any sort of interviews.

*Related Article - Machine Learning Applications*

### 14) What is the difference between Type1 and Type2 errors?

Type 1 error is classified as a false positive. I.e. This error claims that something has happened but the fact is nothing has happened. It is like a false fire alarm. The alarm rings but there is no fire.

Type 2 error is classified as a false negative. I.e. This error claims that nothing has happened but the fact is that actually, something happened at the instance.

The best way to differentiate a type 1 vs type 2 error is:

- Calling a man to be pregnant- This is a Type 1 example
- Calling pregnant women and telling them that she isn't carrying any baby- This is a type 2 example

### **15) Define what is Fourier Transform in a single sentence?**

A process of decomposing generic functions into a superposition of symmetric functions is considered to be a Fourier Transform.

### **16) What is deep learning?**

Deep learning is a process where it is considered to be a subset of the machine learning process.

### **17) What is the F1 score?**

The F1 score is defined as a measure of a model's performance.

### **18) How is the F1 score is used?**

The average Precision and Recall of a model is nothing but an F1 score measure. Based on the results, the F1 score is 1 then it is classified as best and 0 being the worst.

### **19) How can you ensure that you are not overfitting with a particular model?**

In Machine Learning concepts, there are three main methods or processes to avoid overfitting:

Firstly, keep the model simple

Must and should use cross-validation techniques

It is mandatory to use regularization techniques, for example, LASSO.

### **20) How to handle or missing data in a dataset?**

An individual can easily find missing or corrupted data in a data set either by dropping the rows or columns. On contrary, they can decide to replace the data with another value.

In Pandas there are two ways to identify the missing data, these two methods are very useful.

isnull() and dropna().

## **21) Do you have any relevant experience on Spark or any of the big data tools that are used for Machine Learning?**

Well, this sort of question is tricky to answer and the best way to respond back is, to be honest. Make sure you are familiar with Big data and the different tools that are available. If you know about Spark then it is always good to talk about it and if you are unsure then it is best, to be honest, and let the interviewer know about it.

So for this, you have to prepare what is Spark and it's good to prepare other available Big data tools that are used for Machine learning.

*Related Article - Machine Learning with Python*

## **22) Pick an algorithm and write a Pseudocode for the same?**

This question depicts your understanding of the algorithm. This is something that one has to be very creative and also should have in-depth knowledge about the algorithms and first and foremost the individual should have a good understanding of the algorithms. The best way to answer this question would be to start off with Web Sequence Diagrams.

## **23) What is the difference between an array and a Linked list?**

An array is an ordered fashion of collection of objects while a linked list is a series of objects that are processed in sequential order.

## **24): Define a hash table?**

They are generally used for database indexing. A hash table is nothing but a data structure that produces an associative array.

## **25) Mention any one of the data visualization tools that you are familiar with?**

This is another question where one has to be completely honest and also giving out your personal experience with these type of tools are really important. Some of the data visualization tools are Tableau, Plot.ly, and matplotlib.

**26) What is your opinion on our current data process?**

This type of question is asked and the individuals have to carefully listen to their use case and at the same time, the reply should be in a constructive and insightful manner. Based on your responses, the interviewer will have a chance to review and understand whether you are a value add to their team or not.

**27) Please let us know what was your last read book or learning paper on Machine Learning?**

This type of question is asked to see whether the individual has a keen interest in learning and also he is up to the latest market standards. This is something that every candidate should be looking out for and it is vital for individuals to read through the latest publishings.

**28) What is your favorite use case for machine learning models?**

The decision tree is one of my favorite use cases for machine learning models.

**29) Is rotation necessary in PCA?**

Yes, rotation is definitely necessary because it maximizes the differences between the variance captured by the components.

**30) What happens if the components are not rotated in PCA?**

It is a straight effect. If the components are not rotated then it will diminish eventually and one has to use a lot of various components to explain the data set variance.

**31) Explain why Naive Bayes is so Naive?**

It is based on an assumption that all of the features in the data set are important, equal, and independent.

**32) How Recall and True positive rates are related?**

The relation is True Positive Rate = Recall.

**33) Assume that you are working on a data set, explain how would you select important variables?**

The following are a few methods that can be used to select important variables:

1. Use of Lasso Regression method.
2. Using Random Forest, plot variable importance chart.
3. Using Linear regression.

**34) Explain how we can capture the correlation between continuous and categorical variables?**

Yes, it is possible by using the ANCOVA technique. It stands for Analysis of Covariance. It is used to calculate the association between continuous and categorical variables.

**35) Explain the concept of machine learning and assume that you are explaining this to a 5-year-old baby?**

Yes, the question itself is the answer.

Machine learning is exactly the same way how babies do their day-to-day activities, the way they walk or sleep, etc. It is a common fact that babies cannot walk straight away and they fall and then they get up again and then try. This is the same thing when it comes to machine learning, it is all about how the algorithm is working and at the same time redefining every time to make sure the end result is as perfect as possible.

One has to take real-time examples while explaining these questions.

**36) What is the difference between Machine learning and Data Mining?**

Data mining is about working on unstructured data and then extract it to a level where interesting and unknown patterns are identified. Machine learning is a process or a study whether it closely relates to the design, development of the algorithms that provide an ability to the machines to capacity to learn.

**37) What is inductive machine learning?**

Inductive machine learning is all about a process of learning by live examples.

**38) Please state a few popular Machine Learning algorithms?**

Few popular Machine Learning algorithms are:

1. Nearest Neighbour
2. Neural Networks
3. Decision Trees etc
4. Support vector machines

**39) What are the different types of algorithm techniques are available in machine learning?**

Some of them are :

1. Supervised learning
2. Unsupervised learning
3. Semi-supervised learning
4. Transduction
5. Learning to learn

**40) What are the three stages to build the model in machine learning?**

The three stages to build the model in machine learning is:

1. Model building
2. Model testing
3. Applying the model

**41) Explain how the ROC curve works?**

A ROC curve (receiver operating characteristic) is a graph that shows the performance of a classification model at all classification thresholds. It plots two parameters -

- True positive rate
- False-positive rate

True Positive Rate (TPR) is defined as follows:

$$TPR = TP/(TP+FN)$$

False Positive Rate (FPR) is defined as follows:

$$FPR = FP/(FP+TN)$$

## 42) What is the difference between L1 and L2 regularization?

Regularization is a process of introducing some information in order to prevent overfitting.

### L1 Regularization

It is more binary/sparse

L1 regularization corresponds to setting a Laplacean prior on the terms

### L2 Regularization

Tends to spread error among all the terms

It corresponds to a Gaussian prior

## 43) What is a type 1 and type 2 error?

- Type 1 Error - Type 1 error also called false positive, is asserting something true when it is actually false.
- Type 2 Error - Type 2 error also called false negative, is a test result indicating that a condition is failed but in actuality it is successful.

## 44) What is the difference between machine learning and deep learning?

Deep learning is a subset of machine learning and is called so because it makes use of deep neural networks. Let's find out machine learning Vs Deep learning

### Machine learning

Data dependencies

Performs better on small and medium datasets

### Deep Learning

Hardware dependencies

Work on low-end machines

Works better for big data

Requires powerful machines preferably with GPU

Interpretability	Algorithms are easy to interpret	Difficult to interpret
Execution time	From a few minutes to hours	It May take up to a week
Feature Engineering	Need to understand the features that represent the data	No need to understand feature that represents

#### **45) What is Bayes Theorem and how it is used in machine learning?**

Bayes theorem is a way of calculating conditional probability ie. finding the probability of an event occurring based on the given probability of other events that have already occurred. Mathematically, it is stated as -

$$P(A|B) = \{P(B|A).P(A)\}/P(B)$$

Bayes theorem has become a very useful tool in applied machine learning. It provides a way of thinking about the relationship shared by data and the models.

A machine learning model is a specific way of thinking about the structured relationship in the data such as relationships shared by input (x) and output (y).

If we have some prior domain knowledge about the hypothesis, Then the Bayes theorem can help in solving machine learning problems.

#### **46) What is cross-validation techniques would you be using on a time series dataset?**

Cross-validation is used for tuning the hyperparameters and producing measurements of model performance. With the time series data, we can't use the traditional cross-validation technique due to two main reasons which are as follows -

- Temporal dependencies
- Arbitrary Choice of Test Set

For time-series data, we use nested cross-validation that provides an almost unbiased estimate of the true error. A nested CV consists of an inner loop for parameter tuning and an outer loop for error estimation.

#### **47) What are the Discriminative and generative models?**

To understand these terms better, let us consider an example. Suppose a person has two kids - kid A and kid B. Kid A learns and understands everything in depth whereas Kid B can only learn the differences between what he sees. One day, that person took them to the zoo where they saw a deer and a lion. After coming from the zoo, the person showed them an animal and asked them what it was. Kid A drew the images of both the animals he saw in the zoo. He compared the images and answered "the animal is deer" based on the closest match of the image. As Kid B learns things based on only differences, therefore, he easily answered: "the animal is deer.".

In ML, we call Kid A a Generative Model and Kid B a Discriminative Model. To make it more clear, the Generative Model learns the joint probability distribution  $p(x,y)$ . It predicts the conditional probability using Bayes Theorem. Whereas a Discriminative model predicts the conditional probability distribution  $p(y|x)$ . Both of these models are used in supervised learning problems.

#### **48) How is a decision tree pruned?**

In Machine learning, pruning means simplifying and optimizing a decision tree by cutting nodes of the tree that causes overfitting. The pruning process can be divided into two types -

- **Bottom-up pruning** - procedure starts at the last node
- **Top-down pruning** - procedure starts at the root node

Pruning is done to increase the predictive accuracy of a decision tree model.

#### **49) What is more important - model accuracy or model performance?**

Accuracy is more important in machine learning models. We can improve model performance by using distributed computing and parallelizing over the scored assets. But accuracy should be built during the model training process.

#### **50) How would you handle an imbalanced dataset?**

Imbalanced data set is a classification problem where the number of observations per class is not distributed equally. For some classes, there will be a large number of observations whereas for others fewer observations are present. We can fix this issue by -

- Collecting more data to even the imbalances in the dataset.
- Resample the dataset to correct for imbalances.
- Try a different algorithm altogether on your dataset.

## **51) When should you use classification over regression?**

In supervised learning, we have datasets and a list of outcomes. Types of outcomes that we have helped us categorize the problem into classification and regression. For regression problems, the outcomes are typically in real numbers whereas for classification problems outcomes are classes or categories. Therefore, we can say that we would use regression if the outputs are in real numbers and we would go with classifications if the outputs are in the form of classes or categories.

## **52) Tell me a situation where ensemble techniques might be useful?**

Ensemble learning combines several models into one predictive model to decrease the variance and improve results. The ensemble method is divided into two groups - the sequential method and the parallel method.

Sequential method - base learners are generated sequentially

Parallel method - base learners are generated parallelly

Ensembles techniques are -

- Bagging
- Stacking
- Boosting

Scenario - suppose you want to buy a new pair of headphones. What will you do? Being an aware consumer, first, you will do research on which company offers the best headphones and also take some suggestions from your friends. In short, you will be making informed decisions after thoroughly researching work.

## **53) What are the data types supported by JSON?**

Here, the interviewer wants to test your knowledge of JSON. There are six basic data types supported by JSON: strings, numbers, objects, arrays, booleans, and null values.

## **54) According to you, what is the most valuable data in our business?**

Through this question, the interviewer tries to test you on two dimensions: your knowledge and understanding about business models and how you correlated data and apply that thinking about the company. To answer this question, you'll

have to research the business model, learn their business problems, and solve most with their data.

### **55) Tell us about machine learning papers you've read lately?**

To answer this question, you need to keep yourself updated with the latest scientific literature on machine learning to demonstrate your interest in a machine learning position.

### **56) What GPU/hardware do you use and what models do you train for?**

This question tests if you have handled machine learning projects outside of a corporate role and understand how to resource projects and allocate GPU time efficiently. These kinds of questions are usually asked by hiring managers as they want to know what you've done independently.

There are some general questions that an interviewer may ask you depending upon your working experiences and awareness. Some of them are as follows -

1. Do you have research experience in machine learning?
2. What uses cases do you like the most in machine learning?
3. What are your views on GPT-3 and OpenAI's model?

### **57) What is a Fourier Transform?**

Fourier Transform is a general method for decomposing the general functions into asymmetric functions superposition. The Fourier transform discovers the cycle set amplitudes, phases, and speeds for matching any time signal. Fourier transform converts the signal from the time to frequency domain.

### **58) What is the use of the Kernel trick?**

Kernel trick includes kernel functions that can allow higher-dimension spaces without externally computing the dimension's points coordinates. Kernel functions calculate the inner products among the images of all the data pairs in the feature space. This enables them to attribute computing coordinates of higher dimensions when the computation of the said coordinates' external calculation is cheaper.

### **59) Why Naive Bayes is Naive?**

Regardless of its practical applications, particularly in text mining, we consider Naive Bayes naive because it makes superposition which is practically

impossible to see in real-time data. We calculate the conditional probability in the form of the product of the separate probabilities of the components.

## **60) Define Overfitting? How do we assure that we are not overfitting a model?**

Overfitting happens when the model researches the training data to affect the model performance on the latest data significantly. This indicates that we record the disruption in the training data, and we learn the concepts by model. The problem is that concepts that do not employ the testing data negatively affect the ability of the model for classifying the new data; therefore, it decreases the testing data accuracy.

To avoid Overfitting, we have to apply the following methods:

- We collect more data so that we can train the model with diverse samples.
- We can avoid overfitting by using the ensembling methods, like Random Forest. According to the bagging idea, we use them to minimize the change in the projections by joining the result of the multiple decision trees on various samples of the data set.
- By Selecting the correct algorithm, we can avoid overfitting.

## **61) Explain Cluster sampling?**

Cluster sampling is a process of arbitrarily choosing the integral algorithms inside a specified population, and distributing the same characteristics. Cluster sampling is the likelihood sample where a single sampling unit is a cluster or collection of the elements. For instance, if we are clustering the cumulative number of managers in a group of companies, in such a case, managers will depict employees and companies will depict the clusters.

## **62) What are the methods available to screen the Outliers?**

We can use the following methods to screen the outliers:

**Linear models:** Linear models like logistic regression can be trained to screen the outliers. In this way, the model collects the subsequent outlier it meets.

**Boxplot:** The box plot depicts the allocation of the data and its changeability. Box plot includes lower and upper quartiles; therefore, the box fundamentally stretches the Inter-Quartile Range(IQR). The main reason for using the box plot is to identify the outliers in the data.

**Proximity-based models:** K-means clustering is the example of this kind of model, where data points form various or “k” clusters based on the features like distance or similarity.

**Probabilistic and Statistical models:** We can use statistical models like exponential distribution and normal distribution for identifying the variations in the allocation of the data points. If we found any data point outside the distribution scope, then we can render it an outlier.

## **Q. What are L1 and L2 Regularization?**

L1 and L2 are the regularization techniques used to reduce or avoid overfitting in machine-learning models. These regularization techniques add penalties as model complexity increases.

- L1 regularization is called Lasso Regression, while L2 regularization is called Ridge Regression.
- Regularization parameters penalize all the parameters except the intercept.

### **L1 Regularization: Lasso Regression**

- It stands for Least Absolute Shrinkage and Selection Operator.
- In this technique, the data points shrink towards the central point, like the mean.
- L1 regularization has built-in feature selection as it shrinks the less important features coefficient to zero.
- Robust to outliers.

### **L2 Regularization: Ridge Regression**

- Used to analyze Multi-linear Regression.
- It is not used for feature selection, as weights are only reduced to approximately zero.
- Not Robust to outliers

## **Q. What is Central Limit Theorem? Explain the importance.**

The Central Limit Theorem states for a given population mean and standard deviation, if you take a large random sample from the population with replacement, then the distribution of the sample mean will be approximately normally distributed regardless of whether the population is normal or skewed.

Note: The sample size for CLT must be greater than 30.

Importance:

- Allows using standard statistical techniques to analyze the data even when the population data is not normal, which makes it easy to make decisions about the population.
- It allows us to assume the sampling distribution of the mean will be normal in most cases.

## Q. Explain the concept of Precision and Recall?

Precision and recall are the evaluation matrices that are used to evaluate the model performance.

The values of precision and recall come from the confusion matrix.

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

### Precision:

It is a measure of relevant data points. In simple terms, it is the ratio of True Positive and all the Positives.

$$\text{Precision} = \text{True Positive (TP)} / (\text{True Positive (TP)} + \text{False Positive (FP)})$$

**Note:** Precision tries to answer, What proportion of positive identification was actually correct?

**Recall:**

Sensitivity measures how well a machine learning model can detect positive instances. In other words, it measures how likely you will get a positive result when you test for something.

**Recall = True Positive (TP) / True Positive (TP) + False Negative (FN)**

**Note:** Recall tries to answer; what proportion of actual positives was identified correctly.

**Q. How to deal with outliers in the dataset?**

An outlier is a value in the dataset that is extremely different from most of the other values. It can be identified using:

- Box-Plot
- Z-score
- Normal Distribution Curve
- Inter-Quartile Range

There are different methods to handle the outlier in the dataset:

- Replacing the outlier value with the mean and median value.
- Dropping the outliers – to prevent the skewness
- Deleting the outliers if they are due to human error or data processing error
- Change the Scale using Normalization
- Quantile-based flooring and capping: Outliers are capped at a certain value above the 90th percentile or floored below the 10th percentile.

**Q. What are some of the most commonly used Machine Learning algorithms?**

Ans. Most commonly used machine learning algorithms based on supervised and unsupervised machine learning are:

- Linear Regression
- Logistic Regression

- Decision Tree
- SVM
- Naive Bayes
- KNN
- K-Means
- Random Forest
- Dimensionality Reduction Algorithms
- Boosting algorithms

## Q2. How Do You Handle Missing or Corrupted Data in a Dataset?

**Ans:** Method of dealing with missing data is completely scenario based. Same method cannot be applied to all the datasets. One of the easiest ways to deal with handling missing or corrupted data in a dataset is by simply dropping the row. But simply dropping the multiple rows of data might result in error if the size of data is low.

There are two useful methods in Pandas:

- **IsNull()** and **dropna()** will help to find the columns/rows with missing data and drop them
- **Fillna()** will replace the wrong values with a fixed value. You can even replace the dataset with the mean value or simply 0

## Q3. What are different types of Machine Learning?

**Ans:** Machine Learning is broadly categorized in four types:

- Supervised Learning
- Unsupervised Learning
- Semi-Supervised Learning
- Reinforcement Learning

**Supervised Learning:** In this type of machine learning the model is trained on the labelled dataset for classification and regression based problem. Some of the algorithm which are a part of supervised learning are Linear Regression, Logistic Regression, Decision Tree, Random Forest, Naive Bayes.

**Unsupervised Learning:** In this type of machine learning the model is trained for finding patterns, anomalies and clusters in unlabeled dataset. Some of the algorithm which are a part of unsupervised learning are K-Means, C-Means, Hierarchical Clustering.

**Semi Supervised Learning:** In this type of machine learning the model is trained using both labelled and unlabeled datasets.

**Reinforcement Learning:** In this type of machine learning the model is left to train on their own using the concept of rewards and penalty. In simple words, there's an agent with a task to completed with rewards, penalties and many hurdles in between.

#### Q4. What is Overfitting and how to avoid it?

Overfitting is a situation which occurs when the model learns the training dataset too well. An overfit model will give high accuracy in case of training dataset even 100% accuracy, but the same model will result in low accuracy when it is applied to a new dataset. High accuracy on training data and low accuracy on validation data or new data is the case of a overfit model.

#### Why does Overfitting occurs?

- **Size of Training data is too low**

Low size of training dataset results in overfitting condition. Model learns each and every data points when the size of training data is too low. In this case the error will be negligible when the model is trained on training data, but when tested on a new data the error rate of the model will be high and the predictions made would be incorrect.

- **Model tries to make predictions on Noisy Data**

Overfitting also occurs when the model tries to make predictions on data that is very noisy, which is caused due to an overly complex model having too many parameters. So, due to this, the overfitted model is inaccurate as the trend does not reflect the reality present in the data.

#### Different ways to deal with the overfitting condition:

- **Train with more data:** Increase the data that you are using to train your model. Low data will mostly result in an overfitting condition.
- **Data Augmentation:** Data augmentation makes a sample data look slightly different every time the model processes it.
- **Cross Validation:** Cross-validation is a powerful measure which can help us deal with overfitting. The initial training data is used to generate multiple mini train-test splits. Use these splits to tune your model.

**Q5.** What do you understand by ensemble learning?

**Ans:** Ensemble learning is a machine learning technique that is used to combine different machine learning base models using the concept of bagging and boosting to improve the accuracy of the model.

**Q6.** What are the different stages of building a model in Machine Learning?

**Ans:** Different phases of machine learning are:

S.No	Stages of Machine Learning	Python Libraries for each stage
1	Data Acquisition	Beautiful Soup, Selenium, Scrapy, Tweepy, PySQL
2	Data Cleaning	Pandas, Dora, Arrow, Scrubadub, Missingno, Dabl, spacy, NLTK
3	Data Manipulation	Modin, Pandas, Pandas-Profilng, Dask, Polars,

		Pyspark, featuretools. AutoFeat
4	Data Visualization	Matplotlib, Plotly, Seaborn, Sweetviz, Autoviz
5	Building Machine Learning Models	Scikit-Learn, Pytorch, Tensorflow, Pyspark, MLlib, Weka, Knime, Prophet, MLflow, H2O, Autosklearn, OpenCV, spacy, NLTK, detectron, yolo
6	Model Optimisation	HyperOpt, Optuna
7	Model Deployment	Heroku, Streamlit, Flask, Django, AWS Sagemaker

## Q7. What is Cost Function?

**Ans:** Cost function or loss function is an important parameter which tells us how well a model is performing. The main agenda while training a model is to optimize the cost function. It is the measure of how wrong the model is in estimating the relationship between X(input) and Y(output) Parameter.

**Q8. What is Linear Regression in Machine Learning**

**Ans:** Linear Regression is a supervised machine learning algorithm which is trained on labelled dataset and is used to predict continuous data.

Linear Regression models finds a linear relationship between continuous independent variable (x) and dependent variables (Y). The relationship between the dependent and independent variable is figured out using a straight line equation of  $Y = mx + c$ , where m is the slope of line and c is the intercept.

**Q9. Name the paradigms of ensemble methods.**

**Ans:** There are two paradigms of ensemble methods, which are –

- Bagging
- Boosting

**Q10. What is Regularization?**

**Ans:** Regularization is a technique to improve the validation score. Most of the time, it is achieved by reducing the training score.

**Q11. What are the full forms of PCA, KPCA, and ICA, and what is their use?**

**Ans:** PCA – Principal Components Analysis

KPCA – Kernel-based Principal Component Analysis

ICA – Independent Component Analysis

These are important feature extraction techniques, which are majorly used for dimensionality reduction.

**Q12. Name the components of relational evaluation techniques.**

**Ans:** The main components of relational evaluation techniques are –

- Data Acquisition
- Ground Truth Acquisition
- Cross-Validation Technique
- Query Type
- Scoring Metric
- Significance Test

### **Q13. What is a Confusion Matrix?**

**Ans:** A confusion matrix is a summary of correct and incorrect predictions and helps visualize the outcomes. Confusion Matrix is a simple technique for checking the performance of a classification model for a given set of test data.

Confusion matrix has two most important parameters: **Actual and Predicted values**

Predicted values	Actual Values	
	Corona +ve	Corona -ve
Corona +ve	TP=560	FP=60
Corona -ve	FN=50	TN=330

*Let's understand TP, FP, FN, TN in terms of Coronavirus affected people analogy.*

- **True Positive:** Actual values is positive and is correctly predicted.
  - You predicted that a person is Corona positive and he actually is having Corona.
- **True Negative:** Actual value is negative and it is correctly predicted.
  - You predicted that person is Corona negative and he actually is NOT having Corona.
- **False Negative:** Actual value is negative and is incorrectly predicted (Type 2 Error)
  - You predicted that person is Corona negative but actually he was Corona positive.
- **False Positive:** Actual value is positive and is incorrectly predicted

- You predicted the person is Corona positive but actually he is not having Corona.

#### **Q14. What is a ROC curve?**

**Ans:** It is a Receiver Operating Characteristic (ROC curve), a fundamental tool for diagnostic test evaluation. ROC curve is a plot of Sensitivity against Specificity for probable cut-off points of a diagnostic test. It is the graphical representation of the contrast between true positive rates and the false positive rate at different thresholds.

#### **Q15. Can you name some libraries in Python used for Data Analysis and Scientific Computations?**

**Ans:** Python is among the most discussed topics in machine learning interview questions.

Some of the key Python libraries used in Data Analysis include –

- Bokeh
- Matplotlib
- NumPy
- Pandas
- SciKit
- SciPy
- Seaborn

#### **Q16. What is the difference between supervised and unsupervised machine learning.**

**Ans:** Supervised learning is all about training labeled data for tasks like data classification, while unsupervised learning does not require explicitly labeling data.

Supervised Learning	Unsupervised Learning
The input data is labeled.	The input data is not labeled.

The data is classified based on the training dataset.	Assigns properties of the given data to categorize it.
Supervised algorithms have a training phase to learn the mapping between input and output.	Unsupervised algorithms have no training phase.
Used for prediction problems.	Used for detecting anomalies and clusters in the dataset.
Supervised algorithms include Classification and Regression.	Unsupervised algorithms include Clustering and Association.
Algorithms used – Linear Regression, Logistic Regression, Decision Tree, Random Forest, etc.	Algorithms used – K-Means, C- Means, Hierarchical Clustering, etc.

### **Q17. Name different methods to solve Sequential Supervised Learning problems.**

**Ans:** Some of the most popular methods to solve Sequential Supervised Learning problems include –

- Sliding-window methods
- Recurrent sliding windows
- Hidden Markov models
- Maximum entropy Markov models
- Conditional random fields
- Graph transformer networks

### **Q18. What is the use of Box-Cox transformation?**

**Ans:** The Box-Cox transformation is a generalized “power transformation” that ensures normal data transformation and distribution. It is used to eliminate heteroscedasticity.

### **Q19. What is a Fourier transform?**

**Ans:** It is a generic method to breaks a waveform into an alternate representation, mainly characterized by sine and cosines.

### **Q20. What is PAC Learning?**

**Ans:** It is an abbreviation for Probably Approximately Correct. This learning framework analyzes learning algorithms and statistical efficiency.

### **Q21. What are the different machine learning approaches?**

Ans. The different machine learning approaches are –

- Concept Vs. Classification Learning
- Symbolic Vs. Statistical Learning
- Inductive Vs. Analytical Learning

### **Q22. What is Gradient Descent?**

**Ans:** Gradient Descent is a popular algorithm used for training Machine Learning models. It is also used to find the values of parameters of a function ( $f$ ) to minimize a cost function.

### **Q23. What is a Hash Table?**

**Ans:** A Hash Table is a data structure that produces an associative array, and is used for database indexing.

### **Q24. What is the difference between Causation and Correlation?**

**Ans:** Causation denotes any causal relationship between two events and represents its cause and effects.

Correlation determines the relationship between two or more variables. Causation necessarily denotes the presence of correlation, but correlation does not necessarily denote causation.

### **Q25. What is the difference between a Validation Set and a Test Set?**

**Ans:** The validation set is used to minimize overfitting. This is used in parameter selection, which means that it helps to verify any accuracy improvement over the training data set. Test Set is used to test and evaluate the performance of a trained Machine Learning model.

## **Q26. What is a Boltzmann Machine?**

**Ans:** Boltzmann Machines have a simple learning algorithm that helps to discover exciting features in training data. These were among the first neural networks to learn internal representations and are capable of solving severe combinatorial problems.

## **Q27. What are Recommender Systems?**

**Ans:** Recommender systems are information filtering systems that predict which products will attract customers, but these systems are not ideal for every business situation. These systems are used in movies, news, research articles, products, etc. These systems are content and collaborative filtering-based.

## **Q28. What is Deep Learning?**

**Ans:** Deep Learning is an artificial intelligence function used in decision-making. It is among the most important functions of machine learning and among the most commonly asked machine learning interview questions.

Deep Learning imitates the human brain's functioning to process the data and create the patterns used in decision-making. Deep learning is a key technology behind automated driving, automated machine translation, automated game playing, object classification in photographs, and automated handwriting generation, among others.

## **Q29. What are imbalanced datasets?**

**Ans:** Imbalanced datasets refer to the different numbers of data points available for different classes.

## **Q30. How would you handle imbalanced datasets?**

**Ans:** We can handle imbalanced datasets in the following ways –

Oversampling/Undersampling – We can use oversampling or undersampling instead of sampling with a uniform distribution from the training dataset. This will help to see a more balanced dataset.

**Data augmentation** – We can modify the existing data in a controlled way by adding data in the less frequent categories.

**Use of appropriate metrics** – Usage of metrics like precision, recall, and F-score can help to describe the model accuracy in a better way if an imbalanced dataset is being used.

### **Q31. What is Pattern Recognition?**

**Ans:** Pattern recognition is the process of data classification by recognizing patterns and data regularities. This methodology involves the use of machine learning algorithms.

### **Q32. Where can you use Pattern Recognition?**

**Ans:** Pattern Recognition can be used in

- Bio-Informatics
- Computer Vision
- Data Mining
- Informal Retrieval
- Statistics
- Speech Recognition

### **Q33. What is Data augmentation? Can you give an example?**

**Ans:** Data augmentation is a machine learning strategy that enables the users to increase the data diversity for training models remarkably from internal and external sources within an enterprise. This does not require any new data collection.

Modification in images is one of the most helpful examples of data augmentation. We can easily perform the following activities on an image and modify it –

- Resizing the image
- Flipping it horizontally or vertically
- Adding noise
- Deforming
- Modifying colors

### **Q34. Mention the differences between Type I and Type II errors.**

**Ans:** The most significant differences between Type I and Type II errors are –

Type I Error	Type II Error
False-positive error	False-negative error
Claims something when nothing has happened	Claims nothing when something has happened
It is the probability of rejecting a true null hypothesis	It is the probability of failing to reject a false null hypothesis

### **Q35. How will you perform static analysis in a Python application?**

**Ans:** PyChecker can be helpful as a static analyzer to identify the bugs in the Python project. This also helps to find out the complexity-related bugs. Pylint is another tool that is helpful in checking if the Python module is at par with the coding standards.

### **Q36. What is Genetic Programming?**

**Ans:** Genetic Programming is a type of Evolutionary Algorithm (EA). It can be used to solve problems across different fields, including optimization, automatic programming, and machine learning. Genetic Programming is inspired by biological evolution. This system implements algorithms that use random mutation, crossover, fitness functions, and multiple generations of evolution, which altogether contribute to solving user-defined tasks.

### **Q37. What are the different types of Genetic Programming?**

**Ans:** Different types of Genetic Programming are –

- Cartesian Genetic Programming (CGP)
- Extended Compact Genetic Programming (ECGP)
- Genetic Improvement of Software for Multiple Objectives (GISMO)
- Grammatical Evolution
- Linear Genetic Programming (LGP)

- Probabilistic Incremental Program Evolution (PIPE)
- Stack-based Genetic Programming
- Strongly Typed Genetic Programming (STGP)
- Tree-based Genetic Programming

### **Q38. What is the Model Selection?**

**Ans:** It is one of the most important machine learning interview questions.

Model Selection refers to a process of selecting models from different mathematical models for describing the same data set. The model selection has its applications across various fields, including statistics, machine learning as well as data mining.

### **Q39. Which classification methods can be handled by Support Vector Machines?**

**Ans:** SVMs can handle two classification methods –

- Combining binary classifiers
- Modifying binary to incorporate multiclass learning

### **Q40. In how many groups can SVM models be classified?**

**Ans:** SVM models are classified into four distinct groups:

- Classification SVM Type 1 (also called C-SVM classification)
- Classification SVM Type 2 (also called nu-SVM classification)
- Regression SVM Type 1 (also called epsilon-SVM regression)
- Regression SVM Type 2 (also called nu-SVM regression)

### **Q41. High variance in data – is it good or bad?**

**Ans:** It is bad. Higher variance in the data suggests that the spread of data is bigger and the dataset is not presenting a very accurate or representative picture of the relationship between the inputs and predicted output.

### **Q42. If your dataset has the issue of high variance, how would you handle it?**

**Ans:** We can use a bagging algorithm to handle the high variance in datasets. These algorithms split the data into subgroups with sampling replicated from random data. After the data is split, we can use random data to create rules

using a training algorithm. We can then use the polling technique to combine all the predicted outcomes of the dataset.

**Q43. What knowledge do you need to have to extract the predicted information from the raw data?**

Ans. To extract the predicted information from the raw data, one must have a good understanding of mathematics, statistics, computer science, machine learning, data visualization, cluster analysis, and data modeling.

**Q44. What is logistic regression?**

Ans. Logistic regression is a statistical technique used to predict a binary result that is zero or one, or a yes or a no.

**Q45. Why is data cleansing important in data analysis?**

Ans. Data is accumulated from a variety of sources. It is important to ensure that the data collected is good enough for analysis. Data cleaning or erasure ensures that data is complete and accurate, and does not contain redundant or irrelevant components.

**Q45. What does the A/B test aim to accomplish?**

Ans. It is a statistical hypothesis test used to detect any changes to the website so that measures can be taken to maximize the possibility of the desired result.

**Q46. Python or R – Which is the best for machine learning?**

Ans. In machine learning projects, both R and Python come with their own advantages. However, Python is more useful in data manipulation and repetitive tasks, making it the right choice if you plan to build a digital product based on machine learning. Moreover, to develop a tool for ad-hoc analysis at an early stage of the project, R is more suitable.

**Q47. What is TF / IDF vectorization?**

Ans. TF-IDF stands for Reverse Document Frequency. It is a numerical statistic is used to determine the importance of a word in a document of a collection or corpus.

**Q48. What are tensioners?**

**Ans.** Tensors are similar to matrices in programming languages, but here they are larger. Tensors can be considered as a generalization of matrices that form a matrix of n dimensions. TensorFlow provides methods that can be used to easily create tensor functions and calculate their derivatives. This is what distinguishes tensors from NumPy matrices.

#### **Q49. What are the benefits of using TensorFlow?**

**Ans.** TensorFlow has numerous advantages, which is why it is the most widely used framework for machine learning. Some of which include –

- Platform independence
- GPU use for distributed computing
- Self-differentiation capacity
- Open source and a great community
- Highly customizable according to requirements
- Support for asynchronous calculations

#### **Q50. Are there any limitations to using TensorFlow?**

**Ans.** Although TensorFlow offers numerous benefits, it has a caveat or two in current versions:

- No support for OpenCL (Open Computing Language) yet
- GPU memory conflicts when used with Theano
- It can be overwhelming for beginners to start

#### **Q51. Can we capture the correlation between continuous and categorical variables?**

**Ans:** Yes, we can establish the correlation between continuous and categorical variables by using the Analysis of Covariance or ANCOVA technique. ANCOVA controls the effects of selected other continuous variables, which covary with the dependent.

#### **Q52. What is selection bias?**

**Ans:** A statistical error that leads to a bias in the sampling portion of an experiment is called selection bias. If the selection bias remains unidentified, it may lead to a wrong conclusion.

#### **Q53. What is PCA? Why is it used?**

**Ans:** Principal component analysis (PCA) is one of the most popular statistical analysis methods used in dimension reduction. PCA is mainly used to summarize the data structure while acquiring factors that are not correlated with each other.

#### **Q54. Explain Features vs. Labels.**

**Ans.** Features are the input information and are independent variables. Labels are the output information for a mode and are dependent variables. Features are one column of the data in your input set and are used in prediction. Labels are the information that gets predicted.

#### **Q55. What is Bias in Machine Learning?**

**Ans.** Data bias in machine learning is a type of error and suggests that there is some inconsistency in data. This error is usually an indication that certain elements of a dataset are more heavily weighted than others. The inconsistencies are not mutually exclusive.

#### **Q56. What is an OOB error?**

Ans. OOB or Out Of Bag (OOB) error is the average error for each calculated sample using predictions from the trees that do not contain in their respective bootstrap sample. OOB error is calculated to get an unbiased measure of the accuracy of the model over test data.

#### **Explain the difference between supervised and unsupervised machine learning?**

In supervised machine learning algorithms, we have to provide labeled data, for example, prediction of stock market prices, whereas in unsupervised we do not have labeled data where we group the unlabeled data, for example, conducting market segmentation.

#### **Explain the difference between KNN and K-Means clustering?**

K-Nearest Neighbours is a supervised machine learning algorithm where we need to provide the labeled data to the model it then classifies the points based on the distance of the point from the nearest points. Whereas, on the other hand, K-Means clustering is an unsupervised machine learning algorithm thus we need to provide the model with unlabelled data and this algorithm classifies points into clusters based on the mean of the distances between different points.

## **What is the difference between classification and regression?**

Classification is used to produce discrete results, classification is used to classify data into some specific categories .for example classifying e-mails into spam and non-spam categories. Whereas, We use regression analysis when we are dealing with continuous data, for example predicting stock prices at a certain point of time.

## **How to ensure that your model is not overfitting?**

Keep the design of the model simple. Try to reduce the noise in the model by considering fewer variables and parameters. Cross-validation techniques such as K-folds cross-validation help us keep overfitting under control. Regularization techniques such as LASSO help in avoiding overfitting by penalizing certain parameters if they are likely to cause overfitting.

## **What are the different sets in which we divide any dataset for Machine Learning?**

For any ML application, we divide our dataset into three segments namely ‘Training Set’, ‘Validation Set’ & ‘Testing Set’. Training Set is used for training the ML model, Validation Set is used for Hyperparameter tuning and Testing Set is used for testing the model to see how well it is performing.

## **List the main advantage of Naive Bayes?**

A Naive Bayes classifier converges very quickly as compared to other models like logistic regression. As a result, we need less training data in the case of a naive Bayes classifier.

## **Explain Ensemble learning?**

In ensemble learning, many base models like classifiers and regressors are generated and combined together so that they give better results. It is used when we build component classifiers that are accurate and independent. There are sequential as well as parallel ensemble methods.

## **Explain Dimensionality Reduction in machine learning.**

Dimensionality Reduction is the method of reducing the number of dimensions of any dataset by reducing the number of features. It is important because as we move into higher dimensions, the datapoints start becoming equidistant from each other which can affect the performance of unsupervised ML algorithms which use euclidean distance as the similarity function to classify datapoints. This is known as the Curse of Dimensionality. Also, it is difficult to visualize data beyond 4 dimensions.

## **What should you do when your model is suffering from low bias and high variance?**

When the model is suffering from low bias and high variance, it is essentially overfitting, where the accuracy of train dataset is much higher than the accuracy of the test dataset. In such a situation, techniques such as Regularization can be used or the model can be simplified by reducing the number of features in the dataset.

## **Explain the differences between random forest and gradient boosting algorithms.**

Random forest uses bagging techniques whereas GBM uses boosting techniques. Random forests mainly try to reduce variance and GBM reduces both bias and variance of a model.

### **1) What is Machine learning?**

Machine learning is a branch of computer science which deals with system programming in order to automatically learn and improve with experience. For example: Robots are programmed so that they can perform the task based on data they gather from sensors. It automatically learns programs from data.

### **2) Mention the difference between Data Mining and Machine learning?**

Machine learning relates with the study, design and development of the algorithms that give computers the capability to learn without being explicitly programmed. While, data mining can be defined as the process in which the unstructured data tries to extract knowledge or unknown interesting patterns. During this process machine, learning algorithms are used.

### **3) What is ‘Overfitting’ in Machine learning?**

In machine learning, when a statistical model describes random error or noise instead of underlying relationship ‘overfitting’ occurs. When a model is excessively complex, overfitting is normally observed, because of having too many parameters with respect to the number of training data types. The model exhibits poor performance which has been overfit.

### **4) Why overfitting happens?**

The possibility of overfitting exists as the criteria used for training the model is not the same as the criteria used to judge the efficacy of a model.

## **5) How can you avoid overfitting?**

By using a lot of data overfitting can be avoided, overfitting happens relatively as you have a small dataset, and you try to learn from it. But if you have a small database and you are forced to come with a model based on that. In such situation, you can use a technique known as **cross validation**. In this method the dataset splits into two section, testing and training datasets, the testing dataset will only test the model while, in training dataset, the datapoints will come up with the model.

In this technique, a model is usually given a dataset of a known data on which training (training data set) is run and a dataset of unknown data against which the model is tested. The idea of cross validation is to define a dataset to “test” the model in the training phase.

## **6) What is inductive machine learning?**

The inductive machine learning involves the process of learning by examples, where a system, from a set of observed instances tries to induce a general rule.

## **7) What are the five popular algorithms of Machine Learning?**

- Decision Trees
- Neural Networks (back propagation)
- Probabilistic networks
- Nearest Neighbor
- Support vector machines

## **8) What are the different Algorithm techniques in Machine Learning?**

The different types of techniques in Machine Learning are

- Supervised Learning
- Unsupervised Learning
- Semi-supervised Learning
- Reinforcement Learning
- Transduction
- Learning to Learn

**9) What are the three stages to build the hypotheses or model in machine learning?**

- Model building
- Model testing
- Applying the model

**10) What is the standard approach to supervised learning?**

The standard approach to supervised learning is to split the set of example into the training set and the test.

**11) What is ‘Training set’ and ‘Test set’?**

In various areas of information science like machine learning, a set of data is used to discover the potentially predictive relationship known as ‘Training Set’. Training set is an examples given to the learner, while Test set is used to test the accuracy of the hypotheses generated by the learner, and it is the set of example held back from the learner. Training set are distinct from Test set.

**12) List down various approaches for machine learning?**

The different approaches in Machine Learning are

- Concept Vs Classification Learning
- Symbolic Vs Statistical Learning
- Inductive Vs Analytical Learning

**13) What is not Machine Learning?**

- Artificial Intelligence
- Rule based inference

**14) Explain what is the function of ‘Unsupervised Learning’?**

- Find clusters of the data
- Find low-dimensional representations of the data

- Find interesting directions in data
- Interesting coordinates and correlations
- Find novel observations/ database cleaning

### **15) Explain what is the function of ‘Supervised Learning’?**

- Classifications
- Speech recognition
- Regression
- Predict time series
- Annotate strings

### **16) What is algorithm independent machine learning?**

Machine learning in where mathematical foundations is independent of any particular classifier or learning algorithm is referred as algorithm independent machine learning?

### **17) What is the difference between artificial learning and machine learning?**

Designing and developing algorithms according to the behaviours based on empirical data are known as Machine Learning. While artificial intelligence in addition to machine learning, it also covers other aspects like knowledge representation, natural language processing, planning, robotics etc.

### **18) What is classifier in machine learning?**

A classifier in a Machine Learning is a system that inputs a vector of discrete or continuous feature values and outputs a single discrete value, the class.

### **19) What are the advantages of Naive Bayes?**

In Naïve Bayes classifier will converge quicker than discriminative models like logistic regression, so you need less training data. The main advantage is that it can't learn interactions between features.

## **20) In what areas Pattern Recognition is used?**

Pattern Recognition can be used in

- Computer Vision
- Speech Recognition
- Data Mining
- Statistics
- Informal Retrieval
- Bio-Informatics

## **21) What is Genetic Programming?**

Genetic programming is one of the two techniques used in machine learning. The model is based on the testing and selecting the best choice among a set of results.

## **22) What is Inductive Logic Programming in Machine Learning?**

Inductive Logic Programming (ILP) is a subfield of machine learning which uses logical programming representing background knowledge and examples.

## **23) What is Model Selection in Machine Learning?**

The process of selecting models among different mathematical models, which are used to describe the same data set is known as Model Selection. Model selection is applied to the fields of statistics, machine learning and data mining.

## **24) What are the two methods used for the calibration in Supervised Learning?**

The two methods used for predicting good probabilities in Supervised Learning are

- Platt Calibration
- Isotonic Regression

These methods are designed for binary classification, and it is not trivial.

**25) Which method is frequently used to prevent overfitting?**

When there is sufficient data ‘Isotonic Regression’ is used to prevent an overfitting issue.

**26) What is the difference between heuristic for rule learning and heuristics for decision trees?**

The difference is that the heuristics for decision trees evaluate the average quality of a number of disjointed sets while rule learners only evaluate the quality of the set of instances that is covered with the candidate rule.

**27) What is Perceptron in Machine Learning?**

In Machine Learning, Perceptron is a supervised learning algorithm for binary classifiers where a binary classifier is a deciding function of whether an input represents a vector or a number.

**28) Explain the two components of Bayesian logic program?**

Bayesian logic program consists of two components. The first component is a logical one ; it consists of a set of Bayesian Clauses, which captures the qualitative structure of the domain. The second component is a quantitative one, it encodes the quantitative information about the domain.

**29) What are Bayesian Networks (BN)?**

Bayesian Network is used to represent the graphical model for probability relationship among a set of variables.

**30) Why instance based learning algorithm sometimes referred as Lazy learning algorithm?**

Instance based learning algorithm is also referred as Lazy learning algorithm as they delay the induction or generalization process until classification is performed.

**31) What are the two classification methods that SVM ( Support Vector Machine) can handle?**

- Combining binary classifiers
- Modifying binary to incorporate multiclass learning

**32) What is ensemble learning?**

To solve a particular computational program, multiple models such as classifiers or experts are strategically generated and combined. This process is known as ensemble learning.

**33) Why ensemble learning is used?**

Ensemble learning is used to improve the classification, prediction, function approximation etc of a model.

**34) When to use ensemble learning?**

Ensemble learning is used when you build component classifiers that are more accurate and independent from each other.

**35) What are the two paradigms of ensemble methods?**

The two paradigms of ensemble methods are

- Sequential ensemble methods
- Parallel ensemble methods

**36) What is the general principle of an ensemble method and what is bagging and boosting in ensemble method?**

The general principle of an ensemble method is to combine the predictions of several models built with a given learning algorithm in order to improve robustness over a single model. Bagging is a method in ensemble for improving unstable estimation or classification schemes. While boosting method are used sequentially to reduce the bias of the combined model. Boosting and Bagging both can reduce errors by reducing the variance term.

**37) What is bias-variance decomposition of classification error in ensemble method?**

The expected error of a learning algorithm can be decomposed into bias and variance. A bias term measures how closely the average classifier produced by the learning algorithm matches the target function. The variance term measures how much the learning algorithm's prediction fluctuates for different training sets.

**38) What is an Incremental Learning algorithm in ensemble?**

Incremental learning method is the ability of an algorithm to learn from new data that may be available after classifier has already been generated from already available dataset.

**39) What is PCA, KPCA and ICA used for?**

PCA (Principal Components Analysis), KPCA ( Kernel based Principal Component Analysis) and ICA ( Independent Component Analysis) are important feature extraction techniques used for dimensionality reduction.

**40) What is dimension reduction in Machine Learning?**

In Machine Learning and statistics, dimension reduction is the process of reducing the number of random variables under considerations and can be divided into feature selection and feature extraction.

**41) What are support vector machines?**

Support vector machines are supervised learning algorithms used for classification and regression analysis.

**42) What are the components of relational evaluation techniques?**

The important components of relational evaluation techniques are

- Data Acquisition

- Ground Truth Acquisition
- Cross Validation Technique
- Query Type
- Scoring Metric
- Significance Test

### **43) What are the different methods for Sequential Supervised Learning?**

The different methods to solve Sequential Supervised Learning problems are

- Sliding-window methods
- Recurrent sliding windows
- Hidden Markow models
- Maximum entropy Markow models
- Conditional random fields
- Graph transformer networks

### **44) What are the areas in robotics and information processing where sequential prediction problem arises?**

The areas in robotics and information processing where sequential prediction problem arises are

- Imitation Learning
- Structured prediction
- Model based reinforcement learning

### **45) What is batch statistical learning?**

Statistical learning techniques allow learning a function or predictor from a set of observed data that can make predictions about unseen or future data. These techniques provide guarantees on the performance of the learned predictor on the future unseen data based on a statistical assumption on the data generating process.

### **46) What is PAC Learning?**

PAC (Probably Approximately Correct) learning is a learning framework that has been introduced to analyze learning algorithms and their statistical efficiency.

**47) What are the different categories you can categorize the sequence learning process?**

- Sequence prediction
- Sequence generation
- Sequence recognition
- Sequential decision

**48) What is sequence learning?**

Sequence learning is a method of teaching and learning in a logical manner.

**49) What are two techniques of Machine Learning?**

The two techniques of Machine Learning are

- Genetic Programming
- Inductive Learning

**50) Give a popular application of machine learning that you see on day to day basis?**

The recommendation engine implemented by major ecommerce websites uses Machine Learning.

**What is Semi-supervised Machine Learning?**

Semi-supervised learning is the blend of supervised and unsupervised learning. The algorithm is trained on a mix of labeled and unlabeled data. Generally, it is utilized when we have a very small labeled dataset and a large unlabeled dataset.

In simple terms, the unsupervised algorithm is used to create clusters and by using existing labeled data to label the rest of the unlabelled data. A Semi-supervised algorithm assumes continuity assumption, cluster assumption, and manifold assumption.

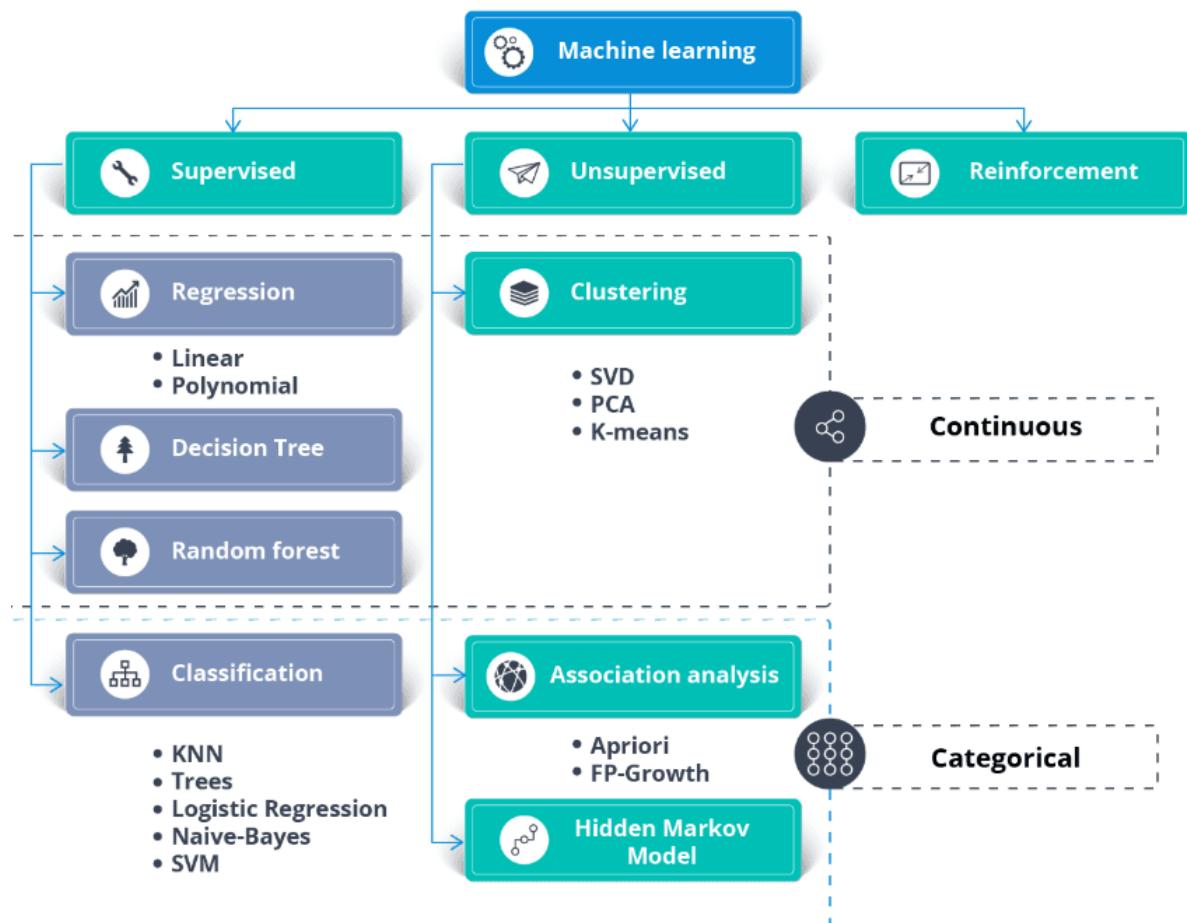
It is generally used to save the cost of acquiring labeled data. For example, protein sequence classification, automatic speech recognition, and self-driving cars.

## How do you choose which algorithm to use for a dataset?

Apart from the dataset, you need a business use case or application requirements. You can apply supervised and unsupervised learning to the same data.

Generally:

- Supervised learning algorithms require labeled data.
  - Regression algorithms require continuous numerical targets
  - Classification algorithms require categorical targets
- Unsupervised learning algorithms require unlabeled data.
- Semi-supervised learning requires the combination of labeled and unlabeled datasets.
- Reinforcement learning algorithms require environment, agent, state, and reward data.



## Explain the K Nearest Neighbor Algorithm.

The K Nearest Neighbor (KNN) is a supervised learning classifier. It uses proximity to classify labels or predict the grouping of individual data points. We can use it for regression and classification. KNN algorithm is non-parametric, meaning it doesn't make an underlying assumption of data distribution.

In the KNN classifier:

- We find K-neighbors nearest to the white point. In the example below, we chose k=5.
- To find the five nearest neighbors, we calculate the euclidean distance between the white point and the others. Then, we chose the 5 points closest to the white point.
- There are three red and two green points at K=5. Since the red has a majority, we assign a red label to it.

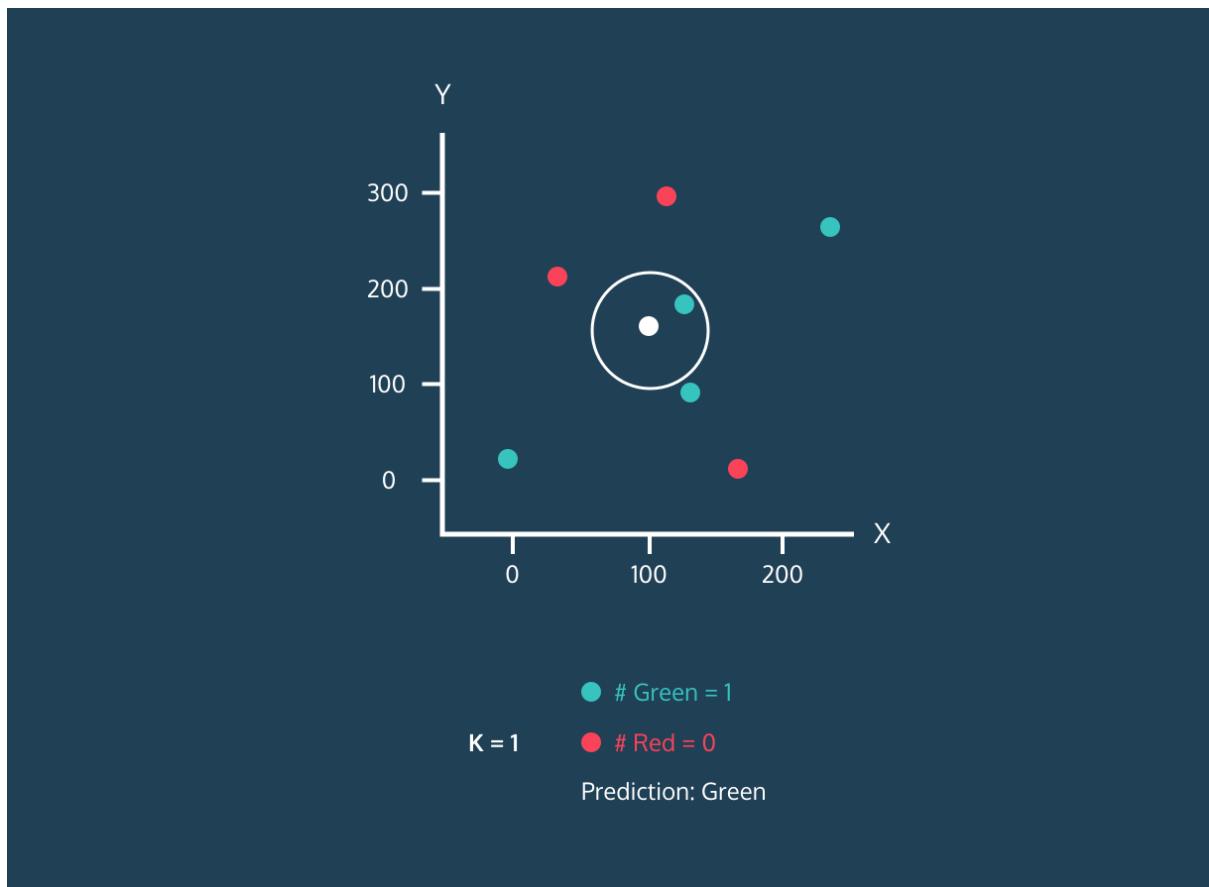
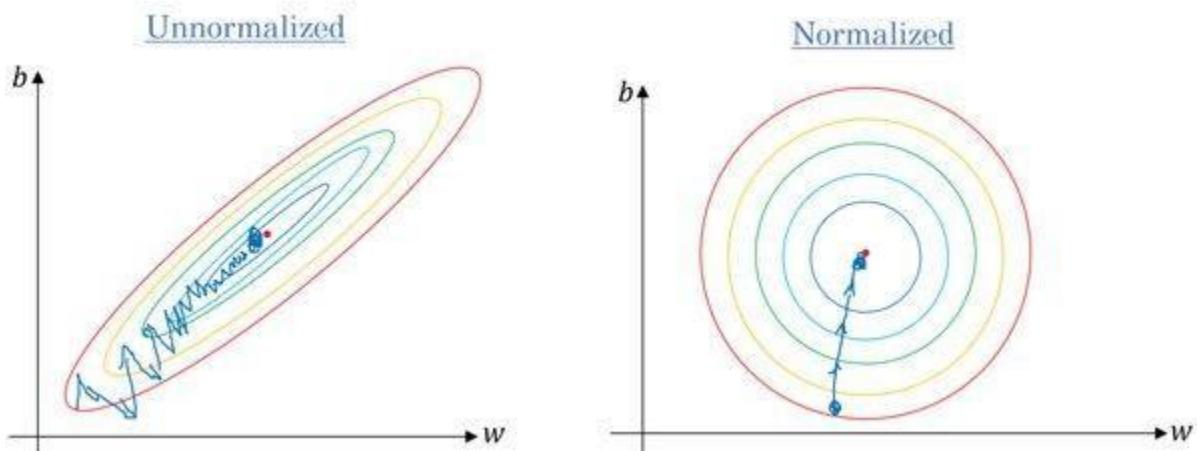


Image from [Codesigner's Dev Story](#)

**Is it true that we need to scale our feature values when they vary greatly?**

Yes. Most of the algorithms use Euclidean distance between data points, and if the feature value varies greatly, the results will be quite different. In most cases, outliers cause machine learning models to perform worse on the test dataset.

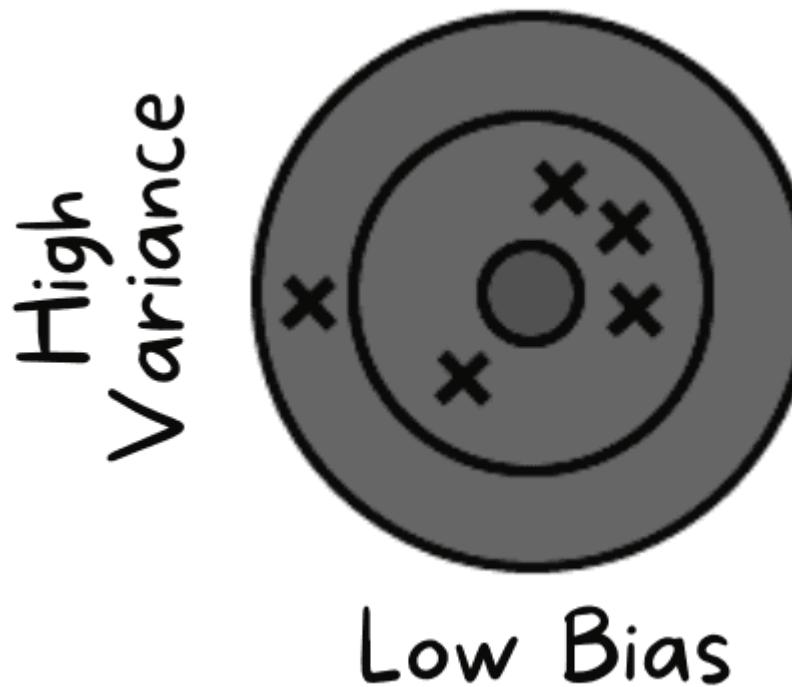
We also use feature scaling to reduce convergence time. It will take longer for gradient descent to reach local minima when features are not normalized.



Gradient without and with scaling | [Quora](#)

**The model you have trained has a low bias and high variance. How would you deal with it?**

Low bias occurs when the model is predicting values close to the actual value. It is mimicking the training dataset. The model has no generalization which means if the model is tested on unseen data, it will give poor results.



Low bias and high variance | Author

To fix these issues, we will use bagging algorithms as it divides a data set into subsets using randomized sampling. Then, we generate sets of models using these samples with a single algorithm. After that, we combine the model prediction using voting classification or averaging.

For high variance, we can use regularization techniques. It penalized higher model coefficients to lower model complexity. Furthermore, we can select the top features from the feature importance graph and train the model.

### **Which cross-validation technique would you suggest for a time-series dataset and why?**

Cross-validation is used to evaluate model performance robustly and prevent overfitting. Generally, cross-validation techniques randomly pick samples from the data and split them into train and test data sets. The number of splits is based on the K value.

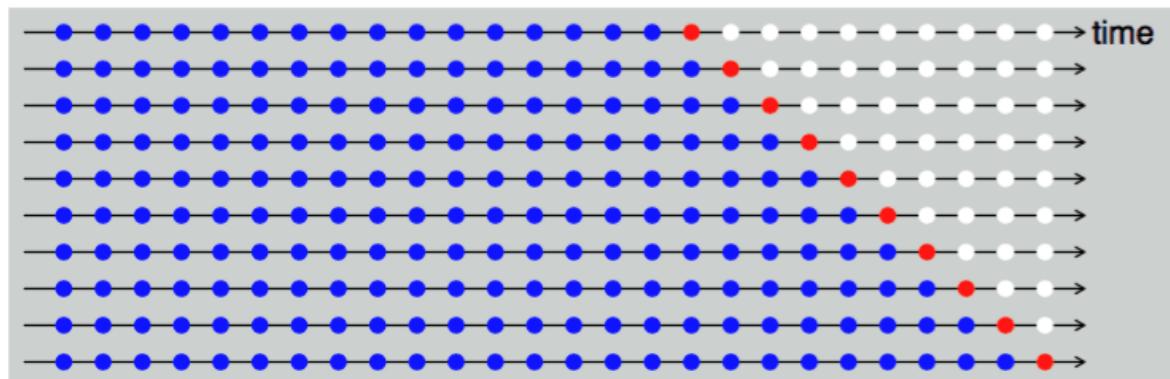
For example, if the K = 5, there will be four folds for the train and one for the test. It will repeat five times to measure the model performed on separate folds.

We cannot do it with a time series dataset because it doesn't make sense to use the value from the future to forecast the value of the past. There is a temporal

dependency between observations, and we can only split the data in one direction so that the values of the test dataset are after the training set.

The diagram shows that time series data k fold split is unidirectional. The blue points are the training set, the red point is the test set, and the white is unused data. As we can observe with every iteration, we are moving forward with the training set while the test set remains in front of the training set, not randomly selected.

## Time series cross-validation



Time series cross validation | UC Business Analytics R Programming Guide

**Why can the inputs in computer vision problems get huge? Explain it with an example.**

Imagine an image of 250 X 250 and a fully connected hidden first layer with 1000 hidden units. For this image, the input features are  $250 \times 250 \times 3 = 187,500$ , and the weight matrix at the first hidden layer will be  $187,500 \times 1000$  dimensional matrix. These numbers are huge for storage and computation, and to combat this problem, we use convolution operations.

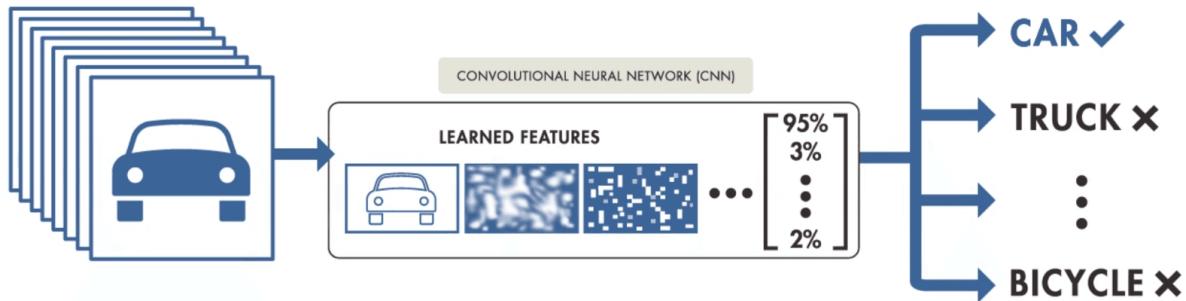
**When you have a small dataset, suggest a way to train a convolutional neural network.**

If you do not have enough data to train a convolutional neural network, you can use transfer learning to train your model and get state-of-the-art results. You need a pre-trained model which was trained on a general but larger dataset. After that, you will fine-tune it on newer data by training the last layers of the models.

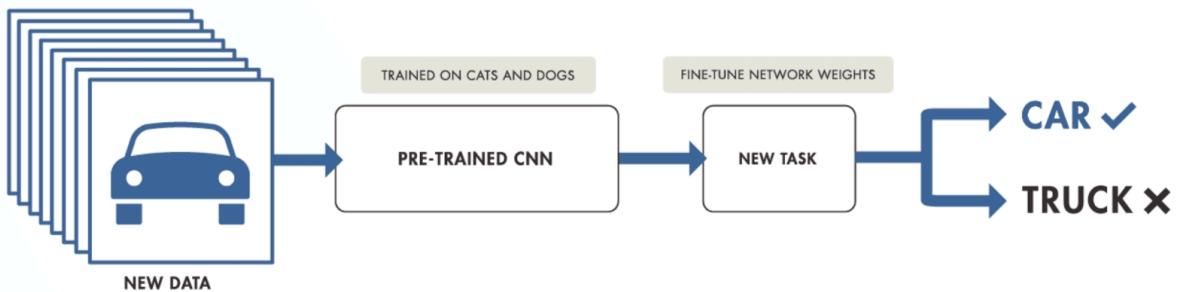
Transfer learning allows data scientists to train models on smaller data by using fewer resources, computing, and storage. You can find open-source pre-trained

models for various use cases easily, and most of them have a commercial license which means you can use them to create your application.

## TRAINING FROM SCRATCH



## TRANSFER LEARNING

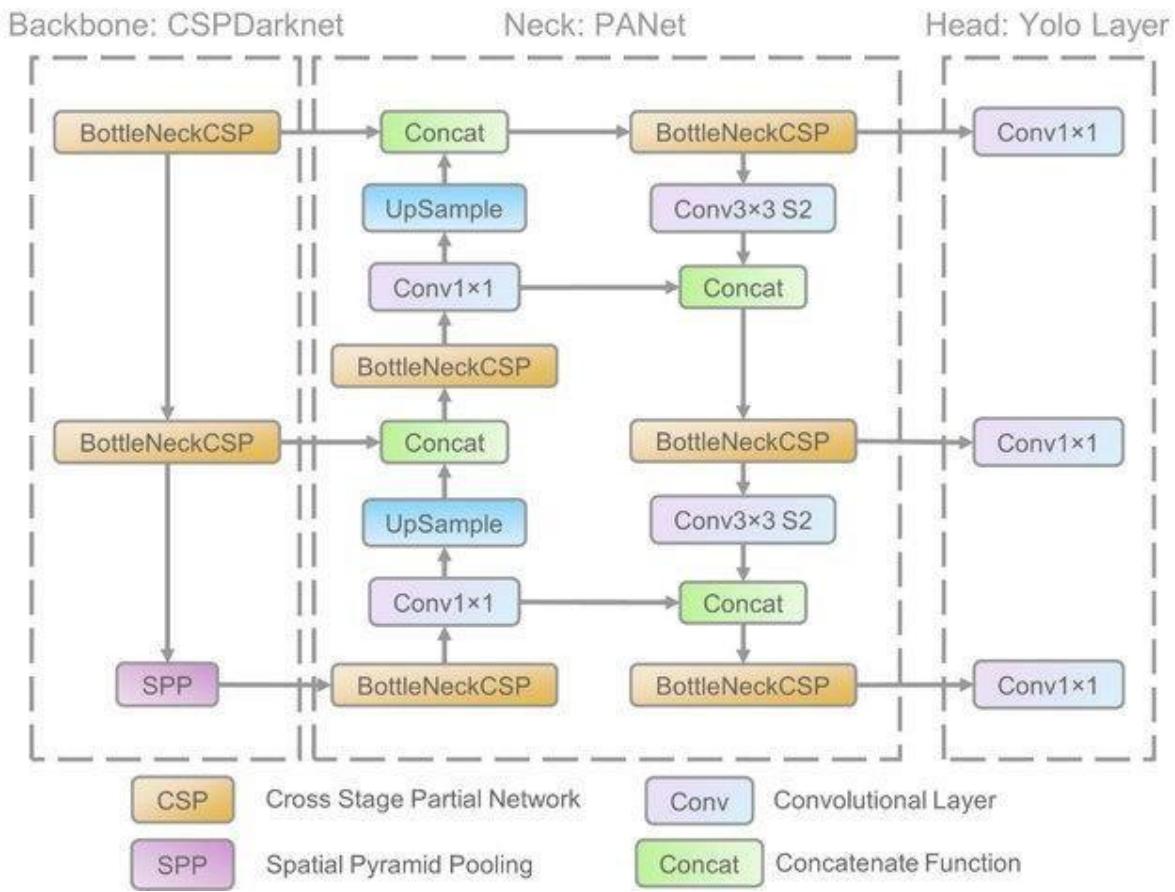


Transfer Learning by **purnasai gudikandula**

### What is the state-of-the-art object detection algorithm YOLO?

YOLO is an object detection algorithm based on convolutional neural networks, and it can provide real-time results. The YOLO algorithm requires a single forward pass through CNN to recognize the object. It predicts both various class probabilities and boundary boxes.

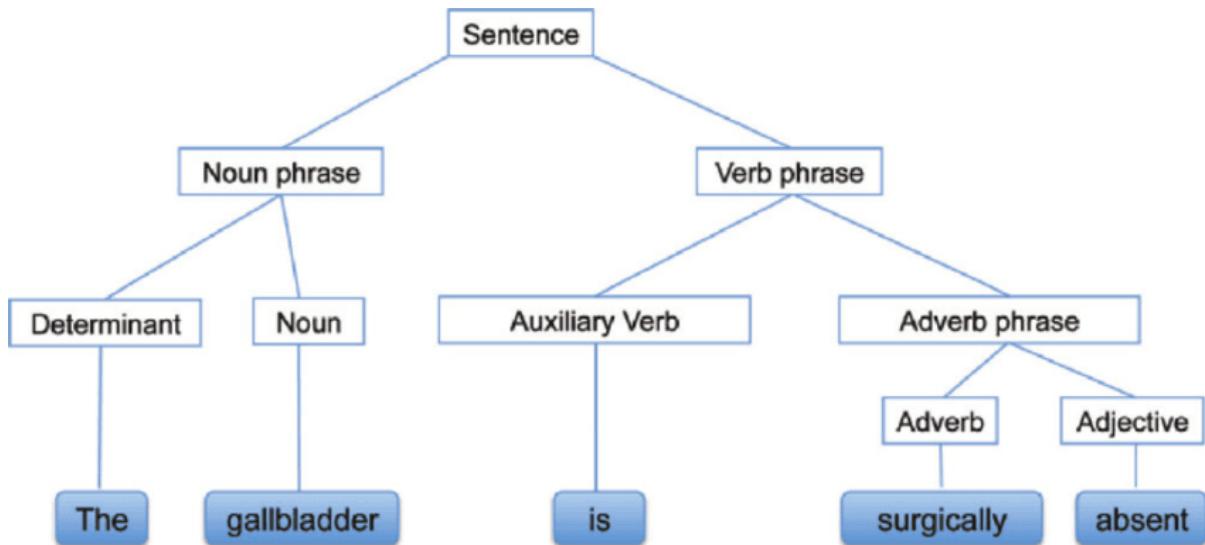
The model was trained to detect various objects, and companies are using transfer learning to fine-tune it on new data for modern applications such as autonomous driving, wildlife preservation, and security.



YOLO V5 model architecture | [researchgate](#)

## What is Syntactic Analysis?

Syntactic Analysis, also known as Syntax analysis or Parsing, is a text analysis that tells us the logical meaning behind the sentence or part of the sentence. It focuses on the relationship between words and the grammatical structure of sentences. You can also say that it is the processing of analyzing the natural language by using grammatical rules.



Syntactic Analysis | [researchgate](#)

## What are Stemming and Lemmatization?

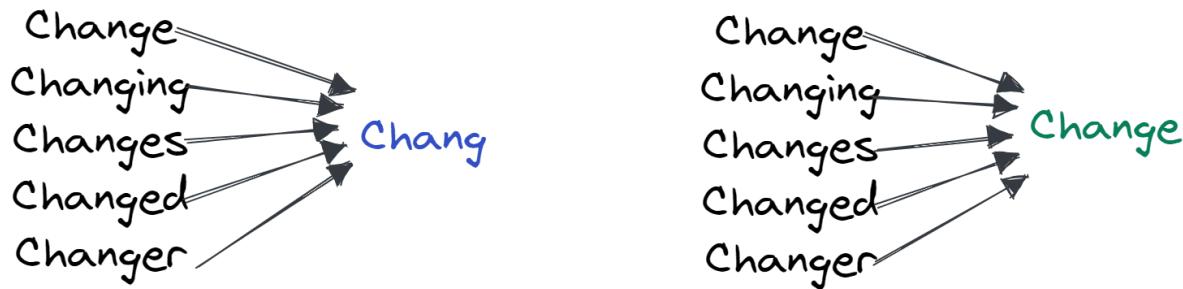
Stemming and lemmatization is a normalizing technique used to minimize the structural variation of words in a sentence.

Stemming removes the affixes added to the word and leaves it in base form. For example, Changing to Chang.

It is widely used by search engines for storage optimization. Instead of storing all the forms of the words, it only stores the stems.

Lemmatization converts the word into its lemma form. The output is the root word instead of the stem word. After lemmatization, we get the valid word that means something. For example, Changing to Change.

## Stemming v/s Lemmatization



Stemming vs. Lemmatization | Author

**How would you reduce the inference time of a trained transformer model?**

It is the responsibility of machine learning engineers to optimize the model inference. Due to large language models, it has become more difficult to deploy models in production and reduce inference time to microseconds.

To improve inference time, we can use:

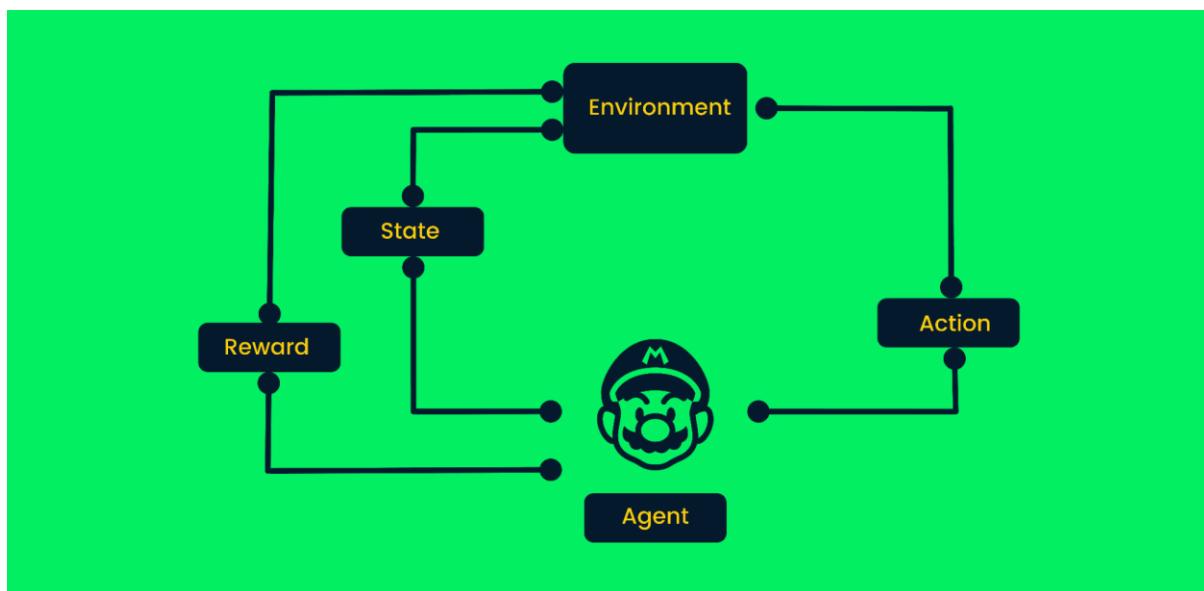
- GPU, TPU, or FPGA for acceleration.
- GPU with fp16 support
- Pruning to reduce parameters
- Knowledge distillation
- Hierarchical softmax or adaptive softmax
- Cache predictions
- Parallel/batch computing
- Reduce the model size

**What are the steps involved in a typical Reinforcement Learning algorithm?**

Reinforcement learning uses trial and error to reach goals. It is a goal-oriented algorithm and it learns from the environment by taking correct steps to maximize the cumulative reward.

In typical reinforcement learning:

1. At the start, the agent receives state zero from the environment
2. Based on the state, the agent will take an action
3. The state has changed, and the agent is at a new place in the environment.
4. The agent receives the reward if it has made the correct move.
5. The process will repeat until the agent has learned the best possible path to reach the goal by maximizing the cumulative rewards.



Reinforcement Learning Framework | Author

## What is the difference between Off-Policy and On-Policy Learning?

On-Policy learning algorithms evaluate and improve the same policy to act and update it. In other words, the policy that is used for updating and the policy that is used to take action are the same.

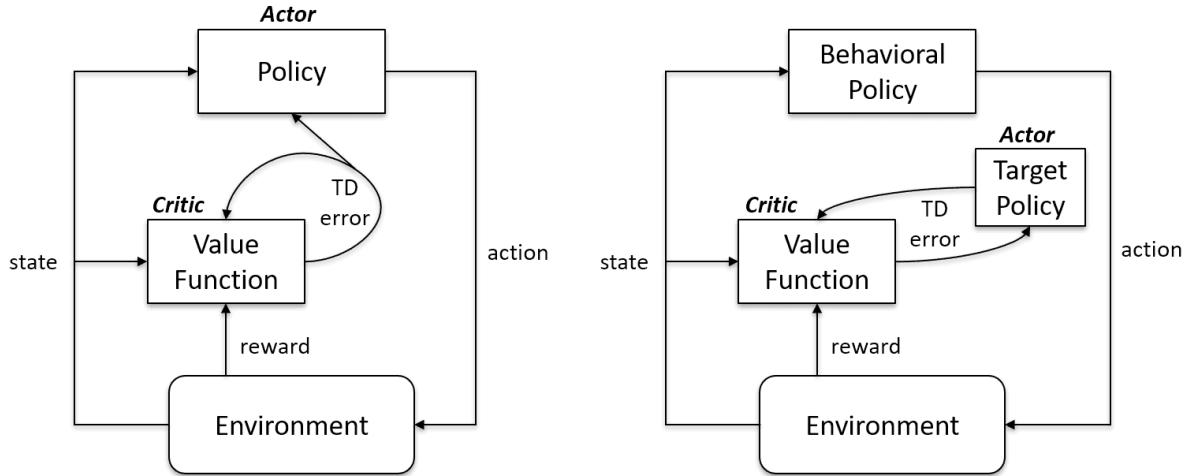
Target Policy == Behavior Policy

On-policy algorithms are Sarsa, Monte Carlo for On-Policy, Value Iteration, and Policy Iteration

Off-Policy Learning algorithms are completely different as the updated policy is different from the behavior policy. For example, in Q-learning, the agent learns

from an optimal policy with the help of a greedy policy and takes action using other policies.

Target Policy  $\neq$  Behavior Policy



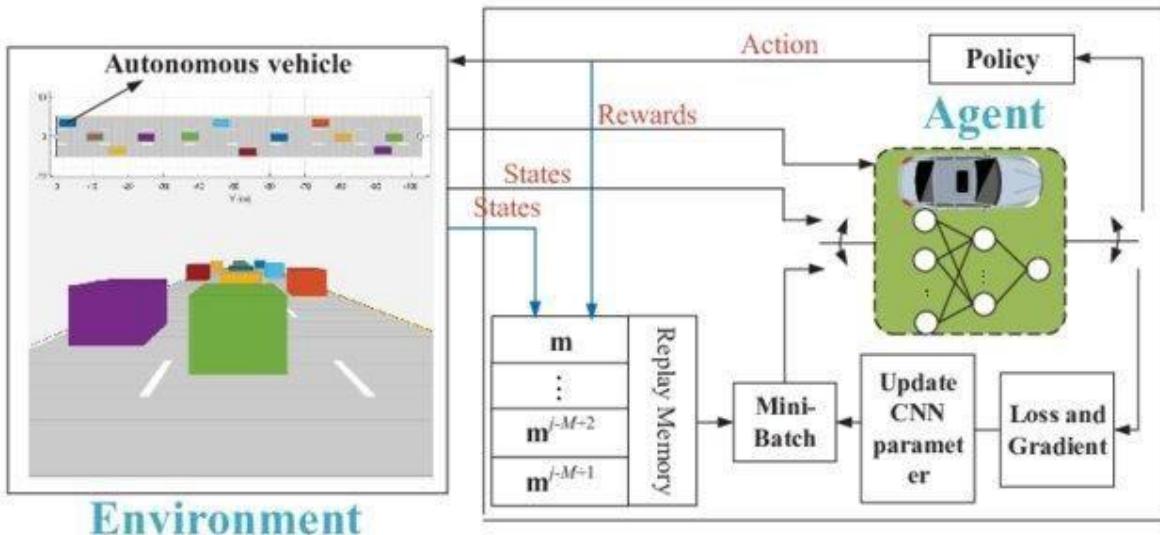
On-policy vs. Off-policy case | [Artificial Intelligence Stack Exchange](#)

### Why do we need “Deep” Q learning?

Simple Q learning is great. It solves the problem on a smaller scale, but on a larger scale, it fails.

Imagine if the environment has 1000 states and 1000 actions per state. We will require a Q table of millions of cells. The game of chess and Go will require an even bigger table. This is where Deep Q-learning comes for the rescue.

It utilizes a neural network to approximate the Q value function. The neural networks recipe states as an input and outputs the Q-value of all possible actions.



Deep Q-network for autonomous driving | [researchgate](#)

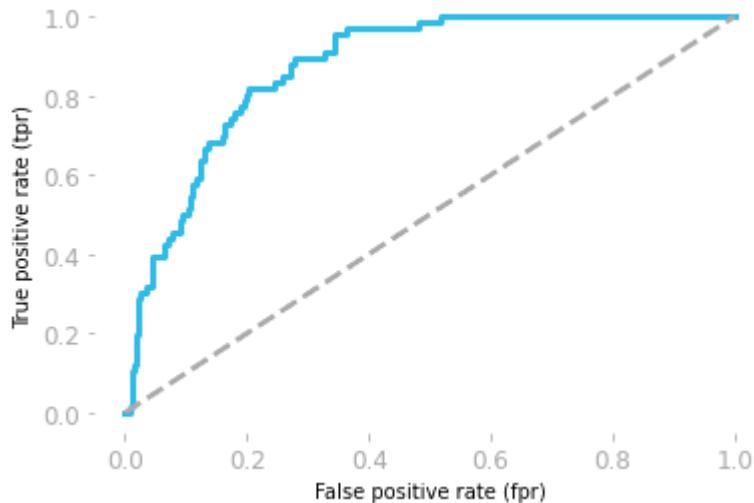
### What is the interpretation of a ROC area under the curve?

Receiver operating characteristics (ROC) shows the trade-off between sensitivity and specificity.

- Sensitivity: it is the probability that the model predicts a positive outcome when the actual value is also positive.
- Specificity: it is the probability that the model predicts a negative outcome when the actual value is also negative.

The curve is plotted using the False positive rate ( $FP/(TN + FP)$ ) and true positive rate ( $TP/(TP + FN)$ )

The area under the curve (AUC) shows the model performance. If the area under the ROC curve is 0.5, then our model is completely random. The model with AUC close to 1 is the better model.



## ROC curve by Hadrien Jean

### What are the methods of reducing dimensionality?

For dimensionality reduction, we can use feature selection or feature extraction methods.

Feature selection is a process of selecting optimal features and dropping irrelevant features. We use Filter, Wrapper, and Embedded methods to analyze feature importance and remove less important features to improve model performance.

Feature extraction transforms the space with multiple dimensions into fewer dimensions. No information is lost during the process, and it uses fewer resources to process the data. The most common extraction techniques are Linear discriminant analysis (LDA), Kernel PCA, and Quadratic discriminant analysis.

### How do you find thresholds for a classifier?

In the case of a spam classifier, a logistics regression model will return the probability. We either use the probability of 0.8999 or convert it into class (Spam/Not Spam) using a threshold.

Usually, the threshold of a classifier is 0.5, but in some cases, we need to fine-tune it to improve the accuracy. The 0.5 threshold means that if the probability is equal to or above 0.5, it is spam, and if it is lower, then it is not spam.

To find the threshold, we can use Precision-Recall curves and ROC curves, grid search, and by manually changing the value to get a better CV.

## **What are the assumptions of linear regression?**

Linear regression is used to understand the relation between features (X) and target (y). Before we train the model, we need to meet a few assumptions:

1. The residuals are independent
2. There is a linear relation between X independent variable and y dependent variable.
3. Constant residual variance at every level of X
4. The residuals are normally distributed.

Note: the residuals in linear regression are the difference between actual and predicted values.

## **Write a function `find_bigrams` to take a string and return a list of all bigrams.**

Creating a bigram function is quite easy. You need to use two loops with the zip function.

1. In bigram function, we are taking a list of the sentence as an input
2. Creating a loop to access a single sentence
3. Lowering and splitting the sentence into a list of words
4. Using `zip` to create a combination of the previous word and the next word
5. Appending the output to the result
6. Printing the results.

It is quite easy if you break down the problem and use zip functions.

```
def bigram(text_list:list):  
  
    result = []  
  
    for ls in text_list:  
  
        words = ls.lower().split()  
  
        for bi in zip(words, words[1:]):
```

```
result.append(bi)

return result

text = ["Data drives everything", "Get the skills you need for the future of
work"]

print(bigram(text))
```

Results:

```
[('Data', 'drives'), ('drives', 'everything'), ('Get', 'the'), ('the', 'skills'), ('skills',
'you'), ('you', 'need'), ('need', 'for'), ('for', 'the'), ('the', 'future'), ('future', 'of'), ('of',
'work')]
```

## What is the activation function in Machine Learning?

The activation function is a non-linear transformation in neural networks. We pass the input through the activation function before passing it to the next layer.

The net input value can be anything between -inf to +inf, and the neuron doesn't know how to bound the values, thus unable to decide the firing pattern. The activation function decides whether a neuron should be activated or not to bound the net input values.

Most common types of Activation Functions:

- Step Function
- Sigmoid Function
- ReLU
- Leaky ReLU

## How would you build a restaurant recommendation on Facebook?

The answer is completely up to you. But before answering, you need to consider what business goal you want to achieve to set a performance metric and how you are going to acquire the data.

In a typical machine learning system design, we:

- Collect, clean, and analyze the data.
- Perform feature engineering

- Select a methodology, algorithm, or machine learning model
- Train and evaluate the performance on test and validation datasets.
- Streamline the processes and deploy the model in production.

You need to make sure you are focusing on design rather than theory or model architecture. Make sure to talk about model inference and how improving it will increase the overall revenues.

Also, give an overview of why you selected a certain methodology over the other.

**Given two strings A and B, write a function can\_shift to return whether or not, A can be shifted some number of places to get B.**

You simply need to create a boolean function that will return True if by shifting the alphabets in String B, you get String A.

```
A = 'abid'  
  
B = 'bida'  
  
can_shift(A, B) == True
```

- Return false if the length of the string is not similar.
- Loop around the range of length of String A
- Create mut\_a to create various combinations of characters using the String A
- During the loop, if mut\_a is equal to String B returns True, else returns false.

```
def can_shift(a, b):  
  
    if len(a) != len(b):  
  
        return False  
  
    for i in range(len(a)):
```

```
mut_a = a[i:] + a[:i]
```

```
if mut_a == b:
```

```
    return True
```

```
return False
```

```
A = 'abid'
```

```
B = 'bida'
```

```
print(can_shift(A, B))
```

```
>>> True
```

## What is Ensemble learning?

Ensemble learning is used to combine the insights of multiple machine learning models to improve the accuracy and performance metrics.

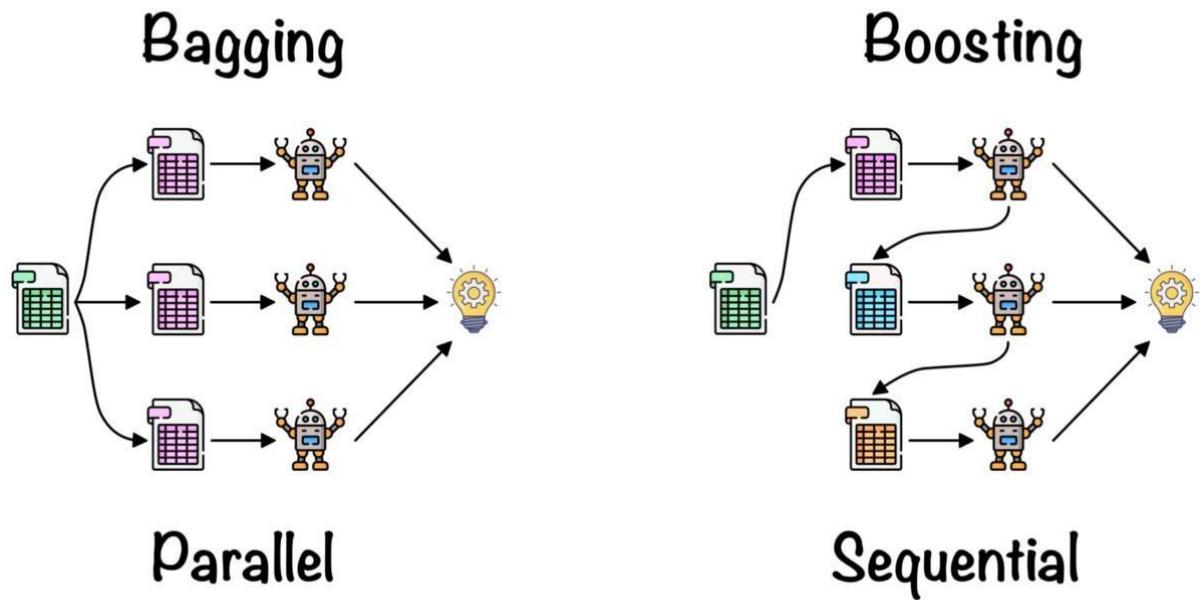
Simple ensemble methods:

- Mean/average: we average the predictions from multiple high-performing models.
- Weighted average: we assign different weights to machine learning models based on the performance and then combine them.

Advance ensemble methods:

- Bagging is used to minimize variance errors. It randomly creates the subset of training data and trains it on the models. The combination of models reduces the variance and makes it more reliable compared to a single model.
- Boosting is used to reduce bias errors and produce superior predictive models. It is an iterative ensemble technique that adjusts the weights

based on the last classification. Boosting algorithms give more weight to observations that the previous model predicted inaccurately.



*Q1. What are different types of Machine Learning and briefly explain them?*

The expected answer should mention supervised, unsupervised, and reinforcement learning.

**Supervised Learning** You give the algorithm labeled data and the algorithm has to learn from it and figure out how to solve future similar problems. Think of it as if you're giving the algorithm problems and answers, the algorithm has to learn how these problems were solved in order to solve future problems in a similar manner. This is like the example where the bank learns from your habits which credit card transactions are legit and which are fraudulent.

**Unsupervised Learning** You give the algorithm a problem without any labeled data or any prior knowledge of what the answer could be. Think of it as if you're giving the algorithm problems without any answers, the algorithm has to find the

best answer by driving insights from the data. This is similar to a bank clustering its customers according to various parameters and deciding who's eligible for a credit card offer, line of credit offer, and who isn't eligible for any offers. This is usually done using a Machine Learning method called **K-Means**.

**Reinforcement Learning** This is when the algorithm learns from its own experience using reward and punishment. The easiest example is self-driving cars where there is an agent that learns from each move it makes. A positive move toward the target earns the agent a reward while a negative move away from the target earns the agent a punishment.

***Q2. Give me an example of supervised learning and another for unsupervised learning?***

Here I usually expect to hear the 3 words: **Classification**, **Regression**, and **clustering**. These are some of the most popular and basic uses for Machine Learning.

Classification and Regression mainly use supervised learning and the candidate can give an example showing how historical data is used to train the model.

For example, if someone steals your credit card and makes an online transaction. You will probably get an email or text from your bank asking to verify this transaction otherwise the bank will consider it fraud. Your bank's algorithm learned your credit card purchasing habits through your purchase history and when an abnormal transaction was detected the bank suspected it's a fraud. This is a form of Machine Learning and probably it's decision tree **Classification**.

Another example is a car company trying to predict sales for next year based on this year's numbers and historical data, that's a form of Machine Learning and could be linear **Regression**.

**Clustering** mainly uses unsupervised learning where there is no historical data. A simple example is the spam email filter where the algorithm examines different parts of all incoming emails, group them together, then cluster the emails into spam and ham.

*Q3. You built a DL model and while training it you noticed that after a certain number of epochs the accuracy is decreasing. What's the problem and how to fix it?*

The answer should be around **overfitting**.

It seems the model is learning the exact dataset characteristics rather than capturing its features this is called overfitting the model. Probably the model is very complex in comparison to the dataset, the model is complex in terms of having many layers and neurons than needed.

Depending on the situation there are several ways to fix this overfitting model the most common are **early stopping** and **dropout regularization**.

Early stopping is what it sounds like, stop the training early once you start seeing the drop in the accuracy. Dropout regularization is dropping some outputs layers or nodes thus the remaining nodes have different weights and have to do extra work to capture the characteristics.

#### **Q4. What's the difference between Bias and Variance in DL models? How to achieve a balance between them?**

This is kinda related to the previous question. The answer should include simple models that underfit, complex models that overfit, and the fact that both Bias and Variance can't be minimized at the same time.

**High Bias** means the model is simple and can't capture many features during the training phase aka underfitting model. **High Variance** means the model is complex and is not only capturing features but also learning anything but those specific training set features, this is also referred to as overfitting.

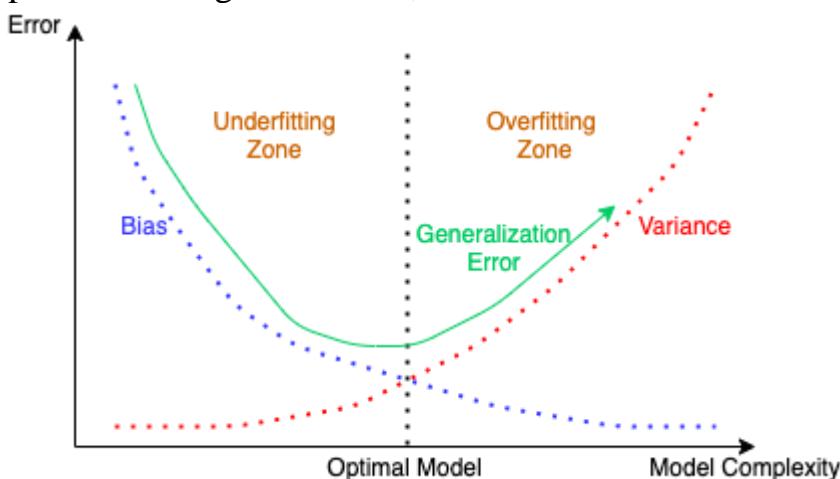


Image by Author

As you can see there is a sweet spot in the middle to balance both Bias and Variance. If your model shift to the right side then it's getting more complicated thus increasing variance and resulting in overfitting. If your model shifts to the left then it's getting too simple thus increasing bias and results in underfitting.

A good data scientist knows how to tradeoff bias and variance by tuning the model's hyperparameters thus achieving optimum model complexity.

A simple model means a small number of neurons and fewer layers while a complex model means a big number of neurons and several layers.

**Q5. What's the confusion matrix? Is it used for both supervised and unsupervised learning? What are Type 1 and Type 2 errors?**

Confusion Matrix is used to assess the performance of supervised learning models only and can't be used with unsupervised models.

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	True Positive (TP)	False Positive (FP)
	Negative (0)	Type 2 Error False Negative (FN)	Type 1 Error True Negative (TN)

Confusion Matrix is a way to present the 4 outcomes of the model: True Positive, False Positive, False Negative, and True Negative. Recall, Precision, Accuracy, and F1 can all be calculated from the Confusion Matrix.

**Type 1 error** is when your algorithm makes a positive prediction but in fact, it's negative. For example, your algorithm predicted a patient has cancer but in fact, he doesn't.

**Type 2 error** is when your algorithm makes a negative prediction but in fact, it's positive. For example, your algorithm predicted a patient doesn't have cancer but in fact, he does.

### ***Q6. What is a model learning rate? Is a high learning rate always good?***

The learning rate is a tuning parameter that determines the step size of each iteration (epoch) during model training. The step size is how fast (or slow) you update your neurons' weights in response to an estimated error. Model weights are updated using the backpropagation error method. So, the input will flow from the input nodes of your model through the neurons to the output nodes then the error is determined and backpropagated to update the neuron's (model) weights. How fast to update those neurons' weights is the learning rate.

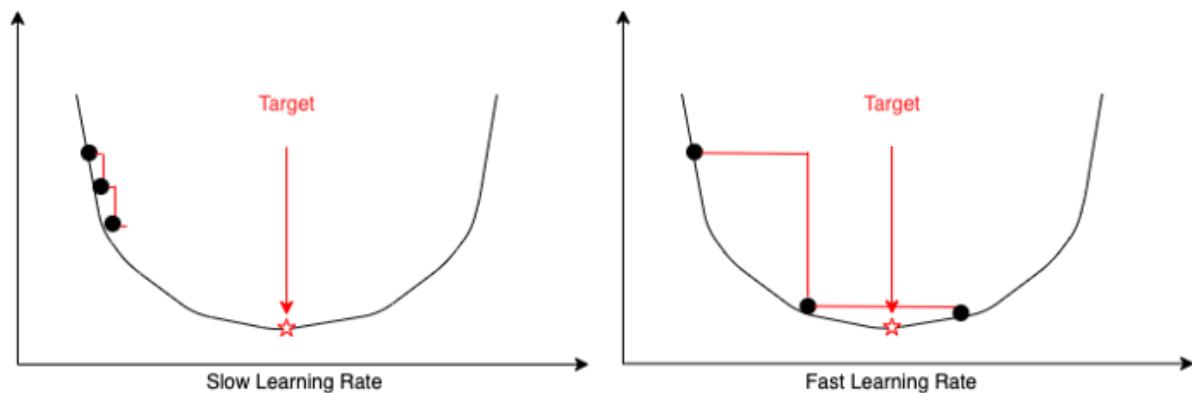


Image by Author

If the learning rate is high thus the model weights are updated fast and frequently your model will converge fast but it may overshoot the true error minima. This means a faster but erroneous model.

If the learning rate is low thus the model weights are updated slowly your model will take a long time to converge but will not overshoot the true error minima. This means a slower but more accurate model.

### ***Q7. What vanishing gradient descent?***

This question is related to the previous one. Here I expect a quick explanation of the gradient descent and how backpropagation affects it.

Think of gradient descent as the weights used to update your neural network during the backpropagation from output to input nodes. Think of Activation as the equation tied to each neuron in your model, this equation decides if this neuron should be activated or not depending on the neuron's input relevancy to the model prediction.

In some cases when you have a deep neural network with several layers and based on your choice of the activation function (along with other hyperparameters), the gradients will become very small and may vanish while backpropagating from the output to input nodes through the layers of the network. The problem here is the weights of the neurons in your model won't get updated (or get updated with very small values) thus your model won't learn (or will get minimal learning). This is a clear case of a vanishing gradient descent problem.

### ***Q8. What's the difference between KNN and K-means?***

I'm personally surprised by how many candidates confuse these two. The answer should state the fact that **KNN is a supervised** model used for classification and **K-means is an unsupervised** model used for clustering. Then the candidate should give an example of classification and another of clustering.

### ***Q9. What does it mean to cross-validate a machine learning model?***

This is another easy one where the answer should include testing the model on new data that the model never seen before. The best example is when you use Scikit Learn (or any other library) to split your data into training and test set. The test set data is used to cross-validate your model after it is trained so you can assess how well your model is performing.

***Q10. How to assess your supervised machine learning model? What's Recall and Precision?***

**Precision:** This is the answer for: out of all the times the model said positive, how many were really positive. You care about precision when False Positive is important to your output.

$$Precision = \frac{TP}{TP + FP}$$

Precision

Let's say you're a small company and you send samples to potential customers who might buy your product. You don't want to send samples to customers that will never buy your product no matter what. The customer who gets a sample but doesn't buy your product is false positive because you predicted they will buy your product (Predicted = 1) but actually, they never will (Actual = 0). In such cases, you want to decrease the FP as much as you can in order to have high precision.

$$Recall = \frac{TP}{TP + FN}$$

## Recall

**Recall:** This is the answer for: out of the actual positives, how many were classified correctly. You care about the recall when False Negative is important to your output. Let's take an example of your credit card, someone stole your credit card number and used it to purchase stuff online from a sketchy website that you never visit. That's clearly a fraudulent transaction but unfortunately, your banks' algorithm didn't catch it. What happened here is that your bank predicted it's not a fraud (predicted = 0) but it was actually a fraud (actual =1). In such a case, your bank should develop a fraud detection algorithm that decreases the FN thus increases the recall.

## *Q11. What's the Curse of Dimensionality and how to solve it?*

This is when your dataset has too many features thus it's hard for your model to learn and extract those features.

Two main things could happen

- More features than observations thus the risk of overfitting the model
- Too many features, observations become harder to cluster. Too many dimensions cause every observation in the dataset to appear equidistant from all others and no meaningful clusters can be formed

The main technique to solve this problem is **Principal Component Analysis (PCA)**.

PCA is an unsupervised machine learning algorithm that attempts to reduce the dimensionality (number of features) within a dataset while still retaining as much

information as possible. This is done by finding a new set of features called components, which are composites of the original features that are uncorrelated with one another. They are also constrained so that the first component accounts for the largest possible variability in the data, the second component the second most variability, and so on.

**Q1:**

**What is Reinforcement Learning? How does it compare with other ML techniques?**

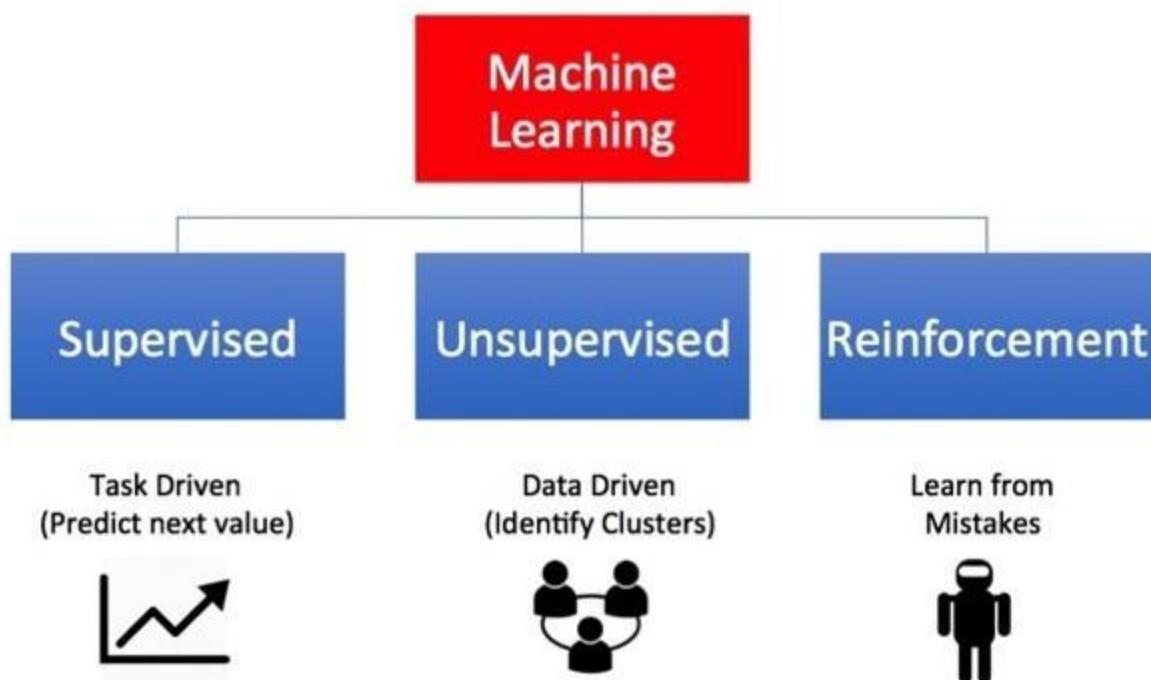
Answer

**Reinforcement learning (RL)** is a subset of machine learning that allows an AI-driven system (sometimes referred to as an agent) to learn through *trial* and *error* using *feedback* from its actions. This feedback is either negative or positive, signaled as punishment or reward with, of course, the aim of maximizing the reward function.

In terms of learning methods, RL is similar to **supervised learning** only in that it uses *mapping* between *input* and *output*, but that is the only thing they have in common. Whereas in **supervised learning**, the *feedback* contains the **correct set of actions** for the agent to follow. In RL there is **no such answer key**. The agent decides what to do itself to perform the task correctly.

Compared with **unsupervised learning**, RL has different goals. The goal of unsupervised learning is to find *similarities* or *differences* between *data points*. RL's goal is to find the *most suitable action model* to **maximize total cumulative reward** for the RL agent. With no training dataset, the RL problem is solved by the agent's own actions with input from the *environment*.

## Types of Machine Learning



*Q2:*

**How to define *States* in Reinforcement Learning?**

Answer

The problem of **State Representation** in Reinforcement Learning (RL) is similar to problems of feature representation, feature selection and feature engineering in *Supervised* or *Unsupervised Learning*.

A common approach to modelling complex problems is **Discretization**. At a basic level, this is splitting a *complex* and *continuous* space into a *grid*. Then you can use any of the classic RL techniques that are designed for discrete, linear, spaces.

Using **tabular learning algorithms** is another good approach to define states given that they have reasonable theoretical guarantees of convergence, which means if you can simplify your problem so that it has, say, less than a few million states, then this is worth trying.

Most interesting control problems will not fit into that number of states, even if you *discretize* them. This is due to the **curse of dimensionality**. For those problems, you will typically represent your *state* as a **vector of different features** - e.g. for a robot learning to walk, various positions, angles, velocities

of mechanical parts. As with supervised learning, you may want to treat these for use with a specific learning process. For instance, typically you will want them all to be numeric, and if you want to use a neural network you should also normalize them to a standard range (e.g. -1 to 1).

---

*Having Machine Learning, Data Science or Python Interview? Check ↗ 54 Reinforcement Learning Interview Questions*

*Source:* ai.stackexchange.com

**Q3:**

**Name some *approaches or algorithms* you know in to solve a problem in Reinforcement Learning**

**Junior**

★ Reinforcement Learning 54

Answer

- **Dynamic Programming (DP):** When the model is fully known, following Bellman equations, we can use **DP** to iteratively evaluate value functions and improve policy.
- **Monte-Carlo (MC)Methods:** It learns from *episodes* of raw experience without modeling the environmental dynamics and computes the observed mean return as an approximation of the expected return. One important thing here is that the episodes must be *complete*, which means that all the episodes must *eventually terminate*.
- **Temporal-Difference (TD) Learning:** Similar to *Monte-Carlo methods*, **TD** Learning is model-free and learns from episodes of experience. However, TD learning can learn from *incomplete episodes* and hence we don't need to track the episode up to termination.
- **Policy Gradient:** All previous methods aim to learn the *state/action-value function* and then to select actions accordingly. *Policy Gradient* methods instead learn the policy function directly with respect to some parameter  $\theta$ , so here we aim to find the best  $\theta$  that produces the *highest return*.

- **Evolution Strategies (ES):** It learns the *optimal solution* by imitating Darwin's theory of the evolution of species by *natural selection*. Two prerequisites for applying ES: (i) our solutions can *freely interact* with the *environment* and see whether they can solve the problem; (ii) we are able to compute a *fitness score* of how good each solution is. We don't have to know the *environment configuration* to solve the problem.
- 

*Having Machine Learning, Data Science or Python Interview? Check ↗ 54 Reinforcement Learning Interview Questions*

*Source:* [lilianweng.github.io](https://lilianweng.github.io)

**Q4:**

**Provide an intuitive explanation of what is a *Policy* in Reinforcement learning**

**Junior**

## ★ Reinforcement Learning 54

Answer

A typical **reinforcement learning (RL)** problem have some basics elements such as:

- An **Environment**: Physical world in which the agent operates.
- **State**: Current situation of the agent.
- **Reward**: Feedback from the environment.
- **Policy**: Method to map agent's state to actions.

But we can think the *policy* like an *agent's strategy*. For example, imagine a world where a robot (agent) moves across the room and the task is to get to the target point (x, y), where it gets a reward. Here:

- A *room* is an *environment*.
- Robot's *current position* is a *state*.
- A *policy* is what an *agent* (the robot) *does* to accomplish this task. The robots have a few options:

- Policy #1: dumb robots just wander around randomly until they accidentally end up in the right place.
- Policy #2: other robots may, for some reason, learn to go along the walls most of the route.
- Policy #3: smart robots plan the route in their "*head*" and go straight to the goal.

Obviously, some policies are better than others, and there are multiple ways to assess them, but the goal of RL is to learn the *best policy*. In the example, the best policy would be option 3. In such terms, the policy is then what defines the learning agent's way of behaving at a given time and is typically used by the agent to decide *what action should be performed*.

---

*Having Machine Learning, Data Science or Python Interview? Check ➤ 54 Reinforcement Learning Interview Questions*

*Source:* stackoverflow.com

**Q5:**

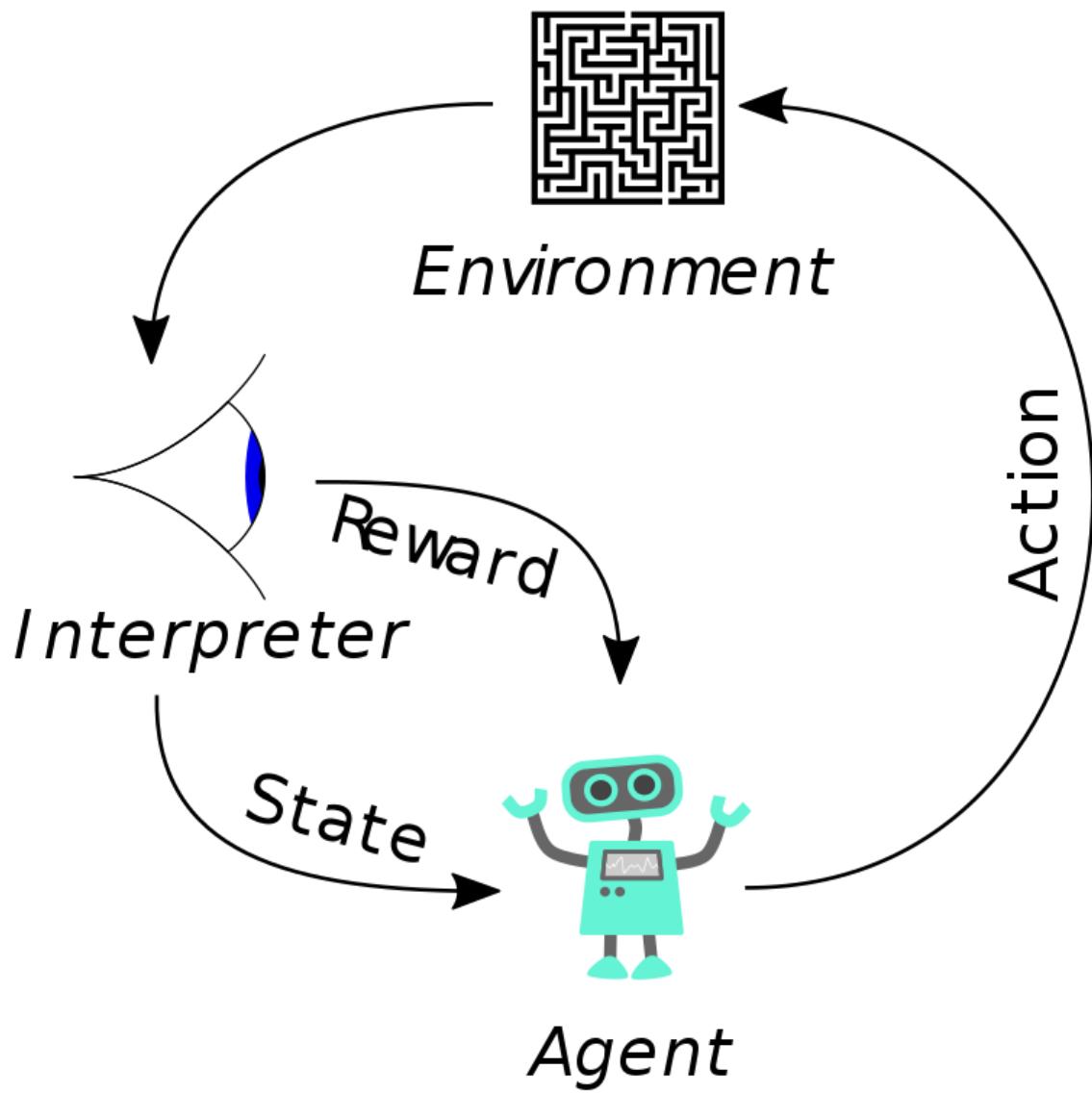
**What are the *steps* involved in a typical *Reinforcement Learning* algorithm?**

**Junior**

 **Reinforcement Learning** 54

**Answer**

1. First, the *agent* interacts with the *environment* by performing an *action*.
2. The *agent* performs an *action* and moves from one *state* to another.
3. Then the *agent* will receive a *reward* based on the *action* it performed.
4. Based on the *reward*, the agent will understand whether the *action* is *good* or *bad*.
5. If the action was *good*, that is, if the agent received a *positive reward*, then the agent will prefer performing *that action*, else the agent will try performing *other actions* that can result in a *positive reward*. So reinforcement learning is basically a *trial-and-error* learning process.



---

Having Machine Learning, Data Science or Python Interview? Check 54 Reinforcement Learning Interview Questions

Source: learning.oreilly.com



## 23 Time Series Interview Questions (ANSWERED) ML Devs Must Know

### ⌚Time Series 38

*Q6:*

**What is *Markov Decision Process*?**

**Junior**

### ⭐ Reinforcement Learning 54

Answer

- A **state** is *Markov* if and only if:

$$P[S_{t+1}|S_t] = P[S_{t+1}|S_1, \dots, S_t]$$

This equation means that the current *state* of the *agent* only depends on the previous *state* and not on any state prior to that.

- This **Markov Decision Process** is used in **Reinforcement Learning**.
- 

*Having Machine Learning, Data Science or Python Interview? Check ↗ 54 Reinforcement Learning Interview Questions*

*Source:* towardsdatascience.com

*Q7:*

**What is the difference between *Off-Policy* and *On-Policy* Learning?**

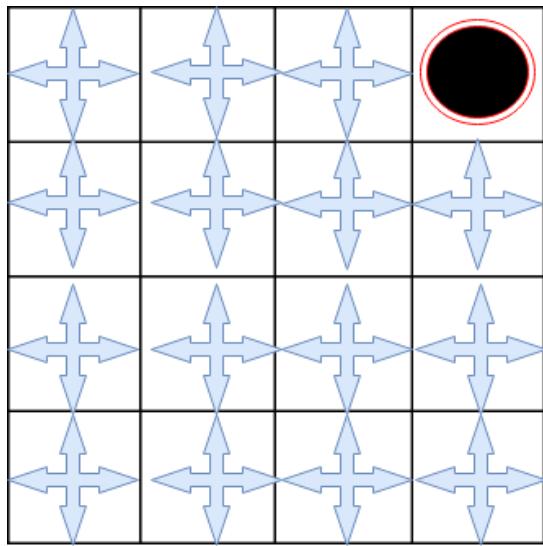
**Junior**

★ **Reinforcement Learning** 54

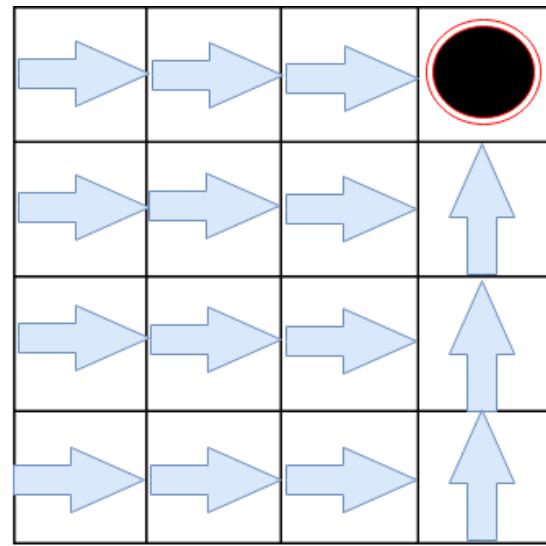
**Answer**

To understand the difference between **On-Policy Learning** and **Off-Policy Learning** let us first take a look at two terms before moving further.

- **Target Policy:** It is the policy that an agent is *trying to learn* i.e agent is learning *value function* for this policy.
- **Behavior Policy:** It is the policy that is being used by an agent for *action select* i.e agent follows this policy to *interact with the environment*.



**Behavior Policy**



**Target Policy**

Now, **On-Policy Learning :**

- We evaluate and improve the same policy which is being used to select actions. That means we will try to evaluate and improve the same policy that the agent is already using for action selection. In short , [Target Policy == Behavior Policy]. Some examples of On-Policy algorithms are *Policy Iteration*, *Value Iteration*, *Monte Carlo* for On-Policy, *Sarsa*, etc.

In **Off-Policy Learning:**

- We evaluate and improve a policy that is different from the policy that is used for action selection. In short, [Target Policy != Behavior Policy]. Some examples of Off-Policy learning algorithms are *Q learning*, *expected sarsa*(can act in both ways), etc.

*Having Machine Learning, Data Science or Python Interview? Check ↗ 54 Reinforcement Learning Interview Questions*

Source: towardsdatascience.com

**Q8:**

**What is the difference between a *Reward* and a *Value* for a given *State*?**

**Junior**

## Reinforcement Learning 54

Answer

- A **Reward** is a number returned at a *certain step* of the *Markov Decision Process*. If you arrange things in sequence over a whole time step s,a,r,s' for *state, action, reward, next state*, then the *reward r* is allowed to depend on all three of s,a,s', and it can also be from a random distribution of real numbers or just a single number.
  - **State values** are a way to measure *longer-term benefits* of being in a *state*, they are also called the *expected return* for an agent starting from that state and following a particular policy.
  - Therefore, we can see **state values** composed of many **rewards weighted by their probability** of occurring in the future.
- 

*Having Machine Learning, Data Science or Python Interview? Check  54 Reinforcement Learning Interview Questions*

*Source:* ai.stackexchange.com

ML Interview Questions

ML Jobs

## CNN 13

## NLP 32

## Big-O Notation 22

 Sorting 26

 Cost Function 13

 Pandas 68

 SQL 47

 Model Evaluation 32

 Gradient Descent 28

 Random Forest 41

*Q9:*

What is the role of the *Discount Factor* in Reinforcement Learning?

**Junior**

 Reinforcement Learning 54

## Answer

The discount factor,  $\gamma$ , is a real value  $\in [0, 1]$ , cares for *the rewards agent achieved in the past, present, and future*. In different words, it relates the *rewards* to the *time-domain*. Let's explore the two following cases:

- When we set the discount factor  $\gamma = 1$ , it implies that we consider all the **future rewards**. Then the agent will *learn forever*, looking for all the future rewards, which may lead to infinity.
- When we set the discount factor  $\gamma = 0$ , it implies that we consider only the **immediate reward** and not the reward obtained from the future time steps. Then the agent will never learn as it will consider only the immediate reward.

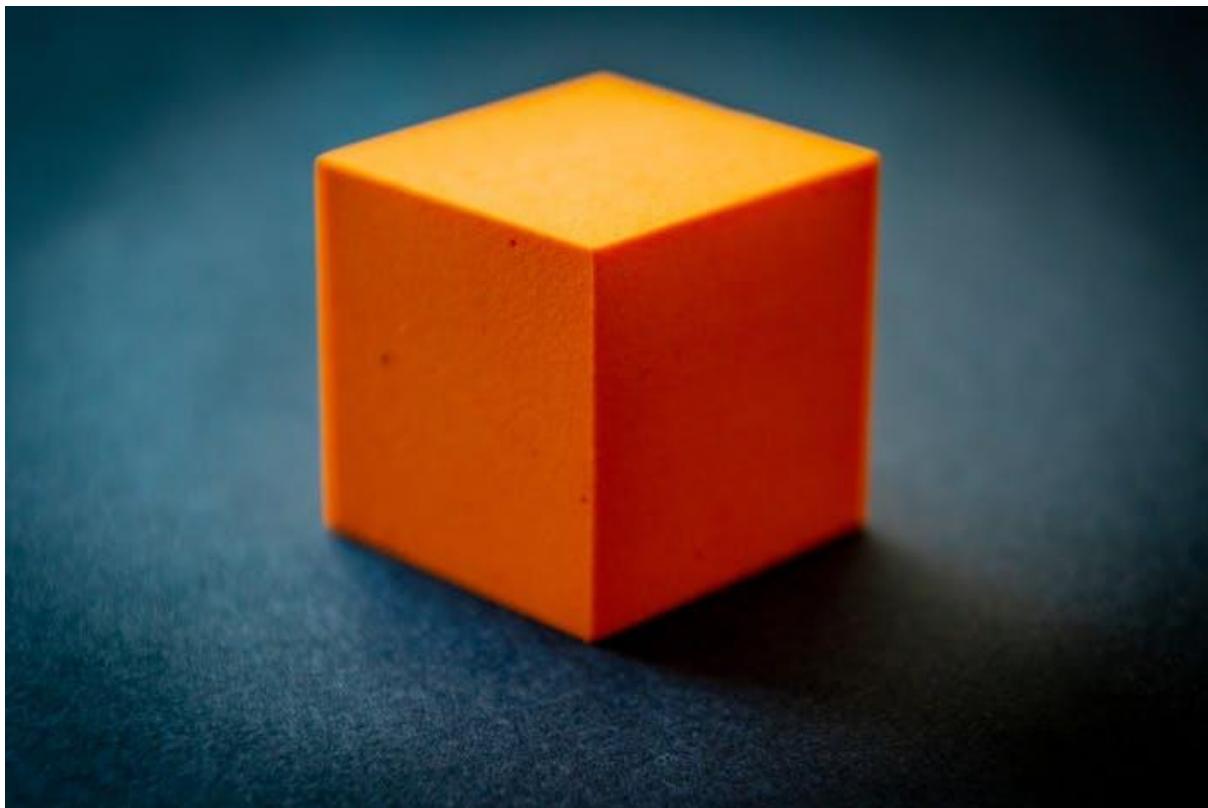
Therefore the optimal value of the discount factor lies between 0.2 and 0.8 it **set the importance to immediate and future rewards depending on the tasks**. In some tasks, *future rewards* are more desirable than *immediate rewards*, and vice versa.

For example, in a chess game, the goal is to defeat the opponent's king. If we give more importance to the *immediate reward*, which is acquired by actions such as our *pawn* defeating *any opposing chessman*, then the agent will learn to perform this sub-goal instead of learning the *actual goal*. So, in this case, we give greater importance to *future rewards* than the *immediate reward*, whereas in some cases, we prefer *immediate rewards* over *future rewards*. For example, you may prefer to eat a fresh chocolate chip today and not in 13 days later.

---

*Having Machine Learning, Data Science or Python Interview? Check ➡ 54 Reinforcement Learning Interview Questions*

*Source:* towardsdatascience.com



30 TensorFlow Interview Questions (ANSWERED) for ML Devs & Data Science



TensorFlow 30

*Q10:*

Are there any problems when using the *Epsilon-Greedy* method to find the *Optimal Policy*?

Mid



Reinforcement Learning 54

Answer

The  $\epsilon$ -greedy policy is a policy that chooses the *best action* (i.e. the action associated with the highest value) with probability  $1-\epsilon \in [0,1]$  and a *random action* with probability  $\epsilon$ . The problem with  $\epsilon$ -greedy is that when it chooses the *random actions* (i.e. with probability  $\epsilon$ ), it chooses them *uniformly* (i.e. it *considers all actions equally good*), even though certain actions (even excluding the currently best one) are better than others.

One solution is to this is to use **Softmax Action Selection Rules**, here we vary the *action probabilities* as a *graded function of estimated value*. In this way, the greedy action is still given the *highest selection probability*, but all the others are ranked and *weighted* according to their *value estimates*. The most common softmax method uses a *Gibbs*, or *Boltzmann*, distribution.

---

*Having Machine Learning, Data Science or Python Interview? Check ➤ 54 Reinforcement Learning Interview Questions*

*Source:* ai.stackexchange.com

**Q11:**

**Can the *Monte Carlo Method* be applicable to all tasks?**

**Mid**

★ **Reinforcement Learning** 54

**Answer**

Monte Carlo is a ***model-free*** method, and so it doesn't require the model dynamics of the *environment* to compute the *value* and *Q function* in order to find the *optimal policy*. Instead, the Monte Carlo method computes the value function and Q function by just taking the *average return of the state* and the *average return of the state-action pair*, respectively.

But one issue with the Monte Carlo method is that it is applicable only to ***episodic tasks***: In the Monte Carlo method, we compute the value of the state by taking the average return of the state and the return is the sum of rewards of the *episode*. But when there is no episode, that is, if our task is a ***continuous task*** (non-episodic task), then we cannot apply the Monte Carlo method.

---

*Having Machine Learning, Data Science or Python Interview? Check ↗ 54 Reinforcement Learning Interview Questions*

*Source:* subscription.packtpub.com

**Q12:**

**Can you think of an example of an *Epsilon-Greedy Policy* in real life?**

**Mid**

## ★ Reinforcement Learning 54

Answer

An **Epsilon-Greedy Policy** allows the *agent* to decide according to a certain *threshold*, between an action that *maximizes a Q-value* or over a *random action* that it may *maximize the Q-value*.

For example, say there are many routes from our work to home and we have explored only *two routes* so far. Thus, to reach home, we can select the *route that takes us home most quickly* out of the two routes we have explored (this is our *Q-value*). However, there are still many other routes that we have not explored yet that might be even better than our current *optimal route*. The question is whether we should explore new routes (**exploration**) or whether we should always use our current optimal route (**exploitation**).

In such context, we introduce a policy called the *epsilon-greedy policy*:

- With a probability *epsilon*, we *explore* different actions of ways to go home from work (*exploration*).
- With a probability  $1 - \text{epsilon}$ , we choose an action that has the maximum **Q** value, that is, the route that takes us to home in the quickest way (*exploitation*).

Now, before selecting an action, a random number *r* in the range of [0,1] is selected. If that *r* is larger than *epsilon*, we use the well-known route that will take us home more quickly; but if  $r < \text{epsilon}$ , a *random action* is selected and we explore other routes. If we follow these *rules* then we have implemented an *epsilon-greedy policy*.

*Q13:*

**Compare Reinforced Learning and Supervised Learning**

Answer

- **Supervised Learning** analyses the training data and produces a generalized formula.
- In **Reinforcement Learning** basic reinforcement is defined in the model **Markov's Decision process**.
- In **Supervised Learning**, each example will have *input objects* and *output* with desired values.
- In **Reinforcement Learning**, the **agent** interacts with the **environment** in discrete steps, and receives a reward for every observation. The goal is to collect as many rewards as possible to make more observations.

*Q14:*

**How does the Monte Carlo prediction method compute the *Value Function*?**

Answer

Let's recap the definition of the **Value Function**. The *value function* or the *value of the state s* can be defined as the *expected return* the agent would obtain *starting* from the state  $s$  and following the policy  $\pi$ . It can be expressed as:

$$V\pi(s) = \tau \sim \pi E[R(\tau) | s_0 = s]$$

In order to approximate the value of the state using the Monte Carlo method, we do the following: 1. Sample N **episodes** (*trajectories*) following the given policy  $\pi$ . Our approximation will be better when N is higher. 2. Compute *the value function* as the *average return of a state across the sample episodes*.

$$V(s) \approx \frac{1}{N} \sum_{i=1}^N R_i(s)$$

In a nutshell, in the Monte Carlo prediction method, we generate some N episodes using the given policy and then we compute the value function as the *average return* of the state across these N episodes, instead of taking the *expected return*.

*Q15:*

**How to choose the values of *Gamma* and *Lambda* in generalised temporal differencing algorithms?**

## Answer

- Typically the **gamma** ( $\gamma$ ) parameter is viewed as part of the *problem*, not of the *algorithm*. A reinforcement learning algorithm tries for each state to optimize the *cumulative discounted reward*:

$$r_1 + \gamma \cdot r_2 + \gamma^2 \cdot r_3 + \gamma^3 \cdot r_4 \dots$$

Where  $r_n$  is the *reward received* at time step  $n$  from the *current state*. So, for one choice of *gamma* the algorithm may optimize one thing, and for another choice, it will optimize something else. However, when you have defined a certain *high-level goal*, there still often remains a modeling choice, as many different gamma's might satisfy the requirements of the goal.

- In general, most algorithms learn *faster* when they don't have to look *too far into the future*. So, it sometimes helps the performance to set *gamma relatively low*. A *general rule of thumb* might be: determine the lowest *gamma min\_gamma* that still satisfies your *high-level goal*, and then set the *gamma* to  $\gamma = (\text{min\_gamma} + 1)/2$ . (You don't want to use  $\gamma = \text{min\_gamma}$  itself, since then some *suboptimal goal* will be deemed virtually as good as the *desired goal*.) Another useful rule of thumb: for many problems a *gamma* of 0.9 or 0.95 is fine. However, always think about what such a *gamma* means for the goal you are optimizing when combined with your reward function.
- The **lambda parameter** determines how much you *bootstrap on earlier learned value* versus *using the current Monte Carlo roll-out*. This implies a trade-off between more bias (*low lambda*) and more variance (*high lambda*). In many cases, setting *lambda* equal to zero is already a *fine algorithm*, but setting *lambda* somewhat higher helps speed up things. Here, you do not have to worry about what you are optimizing: the goal is unrelated to *lambda* and this parameter only helps to speed up learning. In other words, *lambda* is completely part of the *algorithm* and not of the *problem*.
- A *general rule of thumb* is to use a *lambda* equal to 0.9. However, it might be good just to try a few settings (e.g., 0, 0.5, 0.8, 0.9, 0.95 and 1.0) and plot the *learning curves*. Then, you can pick whichever seems to be learning the fastest.

## Q16:

Name some advantages of using *Temporal difference vs Monte Carlo methods for Reinforcement Learning*

## Answer

**Temporal Difference (TD)** is the combination of both *Monte Carlo* (*MC*) and *Dynamic Programming (DP)* ideas. In this sense, like Monte Carlo methods, TD methods can learn directly from the experiences without the model of the environment, but on other hand, there are inherent advantages of TD-learning over Monte Carlo methods.

- **In MC methods:**

- We must wait until the end of the episode before the return is known.
- We have high variance and low bias.
- We don't exploit the Markov property.

- **In Temporal Difference learning:**

- We can learn online after every step and do not need to wait until the end of the episode.
- We have low variance and some decent bias.
- We exploit the Markov property.

The Markov property state that the *future* is *independent of the past* given the *present*, so in this sense temporal difference could be applied in environments that satisfy such assumption.

**Q17:**

**What type of Neural Networks do Deep Reinforcement Learning use?**

Answer

- *Reinforcement learning* is a type of machine learning paradigm where the model take action to *maximize* the notion of *cumulative reward* much like living beings do.
- Learning how to play games, and self-driving cars are all modeled as a reinforcement learning problem.
- If the problem to be modeled is a game, then the screen is taken as input. The algorithm takes the pixels as input and passes it through multiple layers of ***convolutional neural networks*** to give an output for the next steps to take. The outcome of the steps taken by the model serves as the *positive* or *negative* reinforcement.

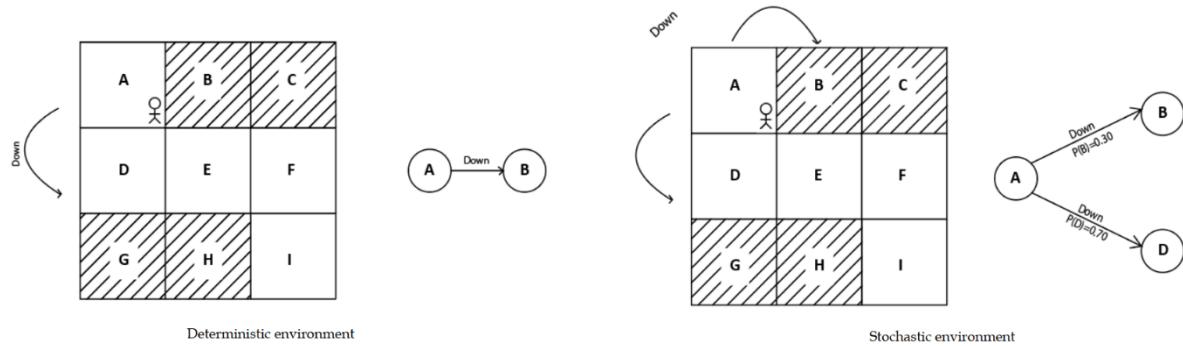
**Q18:**

**What types of Reinforcement Learning Environments do you know?**

Answer

We can categorize the environment into different types:

- **Deterministic** and **Stochastic** environments: In a *deterministic environment*, we are **certain** that when an *agent* performs action *a* in state *s*, then it always reaches state *s'*. For example, let's consider a grid world environment. Say the agent is in state **A**, and when it moves down from state **A**, it always reaches state **D**. Hence the environment is called a deterministic environment. On other hand, in a *stochastic environment* instead we don't have **certainly** but a **probability distribution** over the agent's actions. Taking the grid world environment example, let's say our agent is in state **A**; now if it moves *down* from state **A**, then the agent doesn't always reach state **D**. Instead, it reaches state **D** 70% of the time and state **B** 30% of the time.



- **Discrete** and **Continuous** environments: A *discrete environment* is one where the environment's action space is discrete. For instance, in the grid world environment, we have a *discrete action space*, which consists of the actions up, down, left, right. A *continuous environment* is one where the environment's action space is continuous. For instance, suppose we are training an agent to drive a car, then our action space will be continuous, with *several continuous actions* such as changing the car's speed, the number of degrees the agent needs to rotate the wheel, and so on.
- **Episodic** and **Non-episodic** environments: In an *episodic environment*, an agent's current action will *not affect future actions* and in a *non-episodic* (also called *sequential*) environment, an agent's current action will affect future actions. For example, a chessboard is a sequential environment since the agent's current action will affect future actions in a chess match.
- Finally, we have **Single** and **Multi-agent** environments: When our environment consists of only a *single-agent*, then it is called a *single-*

*agent environment*, and when we have *multiple agents*, then it is called a *multi-agent environment*.

**Q19:**

**What's the difference between *Q-Learning* and *Policy Gradients* methods?**

Answer

### 1. Objective Function

- In *Q-Learning* we learn a Q-function that satisfies the **Bellman (Optimality) Equation**. This is most often achieved by minimizing the **Mean Squared Bellman Error (MSBE)** as the *loss function*. The Q-function is then used to obtain a policy (e.g. by greedily selecting the action with maximum value).
- *Policy Gradient* methods directly try to maximize the expected return by taking small steps in the direction of the policy gradient. **The policy gradient is the derivative of the expected return with respect to the policy parameters.**

### 2. On vs. Off-Policy

- *The Policy Gradient* is derived as an *expectation over trajectories* ( $s_1, a_1, r_1, s_2, a_2, \dots, r_n$ ), which is estimated by a *sample mean*. To get an unbiased estimate of the gradient, the trajectories have to be sampled from the current policy. Thus, policy gradient methods are **on-policy methods**.
- *Q-Learning* only makes sure to satisfy the **Bellman-Equation**. This equation has to hold true for *all transitions*. Therefore, Q-learning can also use experiences collected from previous policies and is **off-policy**.

### 3. Stability and Sample Efficiency

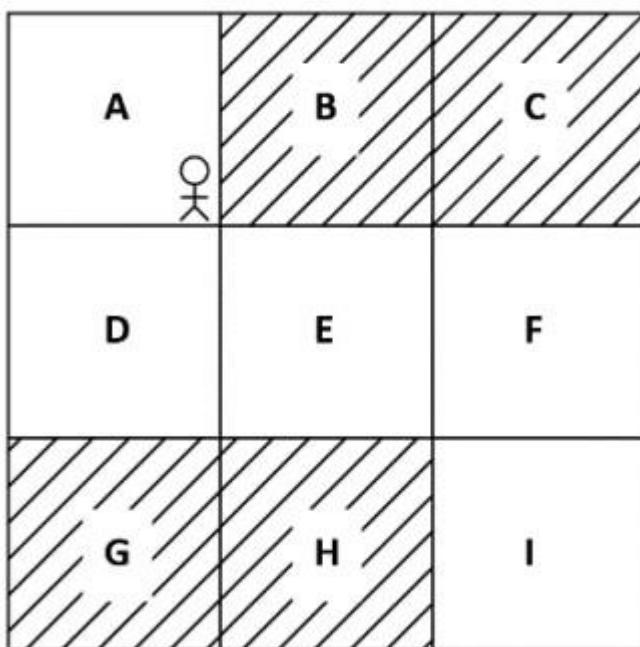
- By directly optimizing the return and thus the actual performance on a given task, *Policy Gradient* methods tend to more *stably converge* to a good behavior. Indeed being *on-policy*, makes them very **sample inefficient**.
- *Q-learning* find a function that is guaranteed to satisfy the *Bellman-Equation*, but this does not guarantee to result in *near-optimal behavior*. Several tricks are used to improve convergence and in this case, Q-learning is more **sample efficient**.

**Q20:**

**What's the difference between a *Deterministic* vs *Stochastic* policy?**

Answer

Let's illustrate the difference with an example. In the preceding **grid world environment**, the goal of the *agent* is to reach state **I** starting from state **A** without visiting the *shaded states*. In each of the states, the agent can perform any of the four actions: up, down, left, and right to achieve the goal.



- A **Deterministic Policy** tells the agent to perform *one particular action in a state*. Thus, the deterministic policy maps the state to one particular action and is often denoted by  $\mu$ . Formally, given a state  $s$  at a time  $t$ , a deterministic policy tells the agent to perform one particular action  $a$ . It can be expressed as:

$$\mu(s) = \mu(A) = \text{Down}$$

In the example, given state **A**, the deterministic policy  $\mu$  tells the agent to perform the action down. This can be expressed as:

$$\mu(A) = \text{Down}$$

Thus, according to the deterministic policy, whenever the agent visits state **A**, it performs the action down.

- A **Stochastic Policy** does not map a state directly to one particular action; instead, it maps the state to a *probability distribution* over an *action space*. So instead of performing the same action every time the agent visits the state, the agent performs *different actions* each time based on a *probability distribution* returned by the stochastic policy.

The stochastic policy is often denoted by  $\pi$ . Say we have a state  $s$  and action  $a$  at a time  $t$ , then we can express the stochastic policy as:

$$\text{P}(a_t = a | s_t = s) \sim \pi(s, a)$$

In the example, given a state **A**, suppose the stochastic policy returns the probability distribution over the action space as [0.10, 0.70, 0.10, 0.10]. Now, whenever the agent visits state **A**, the agent selects up 10% of the time, down 70% of the time, left 10% of the time, and right 10% of the time.

Deterministic policy	Stochastic policy
Maps states → Action	Maps states → Probability distribution over action space
Example :	Example :
A → Down	A → [0.10, 0.70, 0.10, 0.10] up    down    left    right

## Q21:

**Why would you use a Deep Q-Network?**

Answer

When the environment consists of a *large number of states and actions*, it will be very expensive to compute the *Q value* of *all possible state-action pairs* in an *exhaustive fashion*. So, instead of computing Q values in this way, can we approximate them using any *function approximator*, such as a neural network.

In a **Deep Q-Network** we can parameterize our *Q function* by a parameter  $\theta$  and compute the *Q value* where the parameter  $\theta$  is just the parameter of our neural network. So, we just feed the *state* of the environment to a neural network and it will return the *Q value* of all possible actions in that state.

Once we obtain the Q values, then we can select the best action as the one that has the *maximum Q value*.

## Q22:

**Why would you use a Policy-based method instead of a Value-based method?**

Answer

In **Value-based methods**, we improve the *Q function* iteratively, and once we have the optimal *Q function*, then we *extract* optimal policy by *selecting the action* in each state that has the *maximum Q value*.

One of the disadvantages of the *value-based method* is that it is suitable only for **discrete environments** (environments with a *discrete action space*), and we cannot apply value-based methods in **continuous environments** (environments with a *continuous action space*).

For example, say we are training an agent to drive a car and say we have one continuous action in our action space. Let the action be the speed of the car and the value of the speed of the car ranges from 0 to 150 kmph.

In this case, we can discretize the continuous actions into speed (0 to 10) as action 1, speed (10 to 20) as action 2, and so on. After discretization, we can compute the *Q value* of all possible *state-action pairs*. However, discretization is not always desirable. We might lose several important features and we might end up in an action space with a huge set of actions.

Most real-world problems have continuous action space, say, a self-driving car, or a robot learning to walk and more. Apart from having a continuous action space they also have a *high dimension*. Thus, using the **Deep Q Network** and *other value-based methods* cannot deal with the continuous action space effectively.

So, we use the **Policy-based methods**. With policy-based methods, we don't need to compute the *Q function* (*Q values*) to find the optimal policy; instead, it finds the optimal policy by *parameterizing the policy* using some parameter  $\theta$ . The basic idea is to find the best  $\theta$  that produces the **highest return**.

In addition to that, Most policy-based methods use a **stochastic policy**. We know that with a stochastic policy, we select actions based on the *probability distribution* over the action space, which allows the agent to explore different actions instead of performing the same action every time. Thus, *policy-based* methods take care of the **exploration-exploitation** trade-off implicitly by using a stochastic policy.

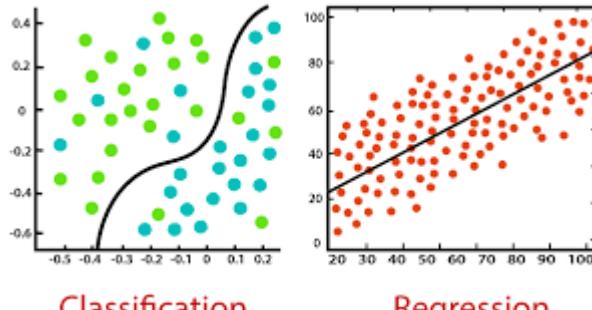
1. What is the difference between Supervised and Unsupervised machine learning?

Supervised learning requires training labeled data. For example, in order to do classification (a supervised learning task), you'll need to first label the data you'll use to train the model to classify data into your labeled groups. Unsupervised learning, in contrast, does not require labeling data explicitly.

2. What is the difference between classification and regression?

Classification is used to produce discrete results, classification is used to classify data into some specific categories .for example classifying e-mails into spam and non-spam categories.

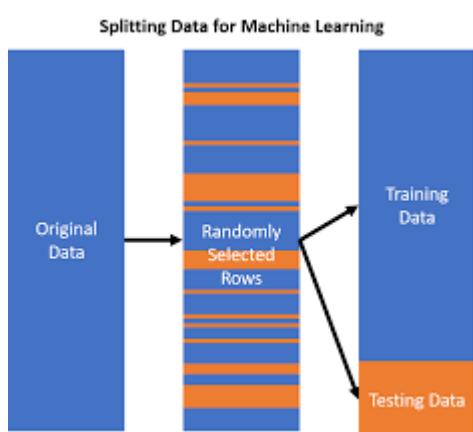
Whereas, We use regression analysis when we are dealing with continuous data, for example predicting stock prices at a certain point of time.



3. What is meant by ‘Training set’ and ‘Test Set’?

‘**Training set**’ is the portion of the dataset used to train the model.

‘**Testing set**’ is the portion of the dataset used to test the trained model.

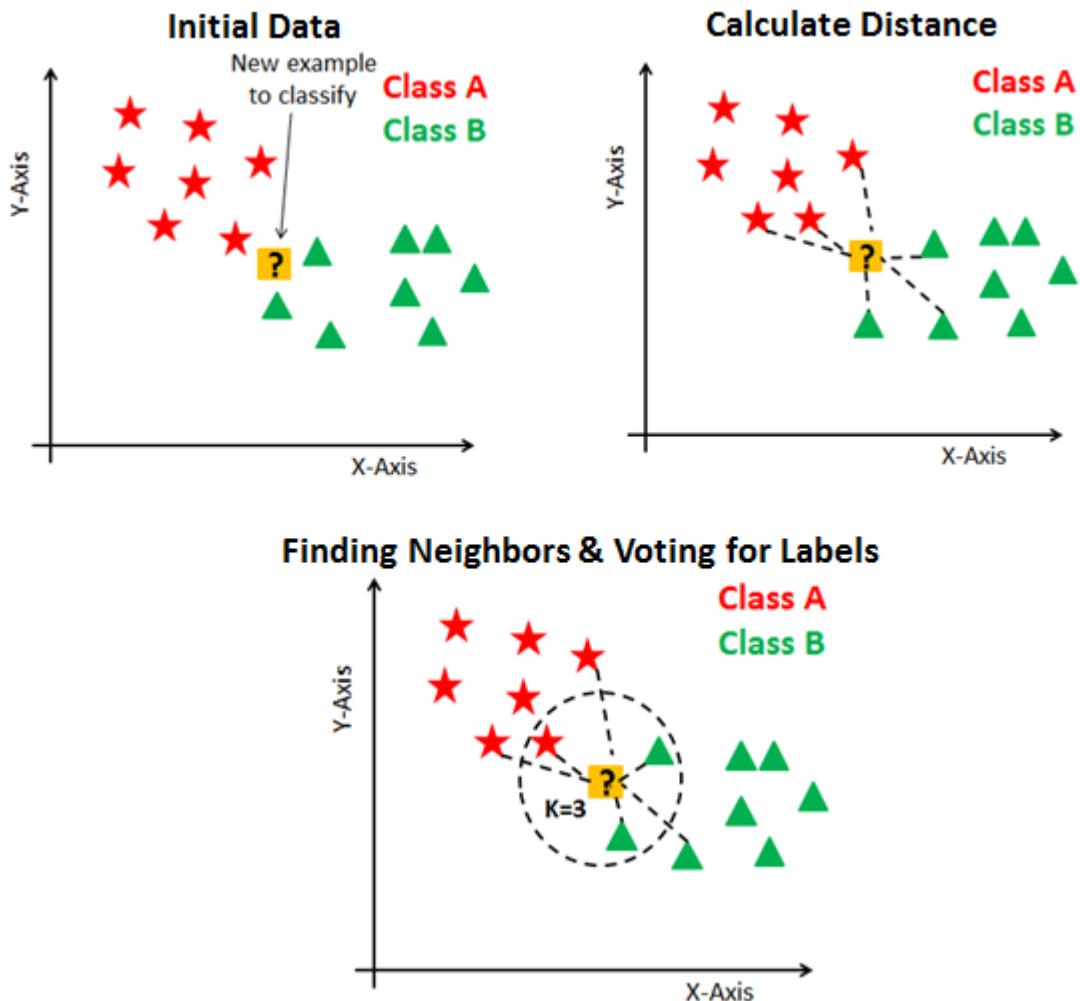


4. How do you handle missing or corrupted data in a dataset?

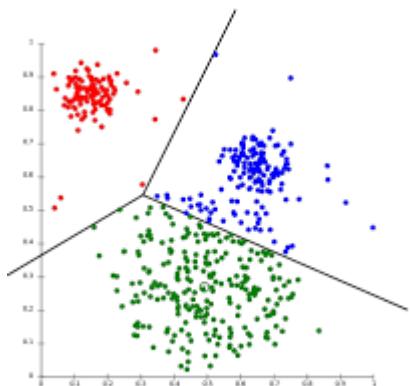
You could find missing/corrupted data in a dataset and either drop those rows or columns, or decide to replace them with another value.

In Pandas, there are two very useful methods: `isnull()` and `dropna()` that will help you find columns of data with missing or corrupted data and drop those values. If you want to fill the invalid values with a placeholder value (for example, 0), you could use the `fillna()` method.

## 5. How is KNN different from k-means clustering



K-Nearest Neighbors is a supervised classification algorithm, while k-means clustering is an unsupervised clustering algorithm. While the mechanisms may seem similar at first, what this really means is that in order for K-Nearest Neighbors to work, you need labeled data you want to classify an unlabeled point into (thus the nearest neighbor part). K-means clustering requires only a set of unlabeled points and a threshold: the algorithm will take unlabeled points and gradually learn how to cluster them into groups by computing the mean of the distance between different points.



The critical difference here is that KNN needs labeled points and is thus supervised learning, while k-means doesn't — and is thus unsupervised learning. KNN algorithm tries to classify an unlabeled observation based on its k (can be any number) surrounding neighbors. It is also known as a lazy learner because it involves minimal training of the model. Hence, it doesn't use training data to make generalizations on the unseen data set.

6. What is the main advantage of Naive Bayes?

A Naive Bayes classifier converges very quickly as compared to other models like logistic regression. As a result, we need less training data in the case of naive Bayes classifier.

7. What's the difference between Type I and Type II error?

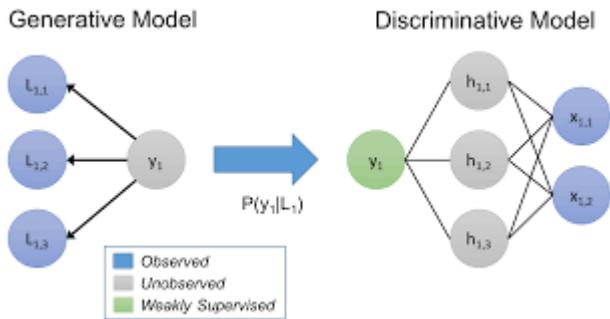
Don't think that this is a trick question! Many machine learning interview questions will be an attempt to lob basic questions at you just to make sure you're on top of your game and you've prepared all of your bases.

Type I error is a false positive, while Type II error is a false negative. Briefly stated, Type I error means claiming something has happened when it hasn't, while Type II error means that you claim nothing is happening when in fact something is.

A clever way to think about this is to think of Type I error as telling a man he is pregnant, while Type II error means you tell a pregnant woman she isn't carrying a baby.

8. What's the difference between a generative and discriminative model?

A generative model will learn categories of data while a discriminative model will simply learn the distinction between different categories of data.



Discriminative models will generally outperform generative models on classification tasks.

## 9. What are Parametric models?

Parametric models are those with a finite number of parameters. To predict new data, you only need to know the parameters of the model. Examples include linear regression, logistic regression, and linear SVMs.

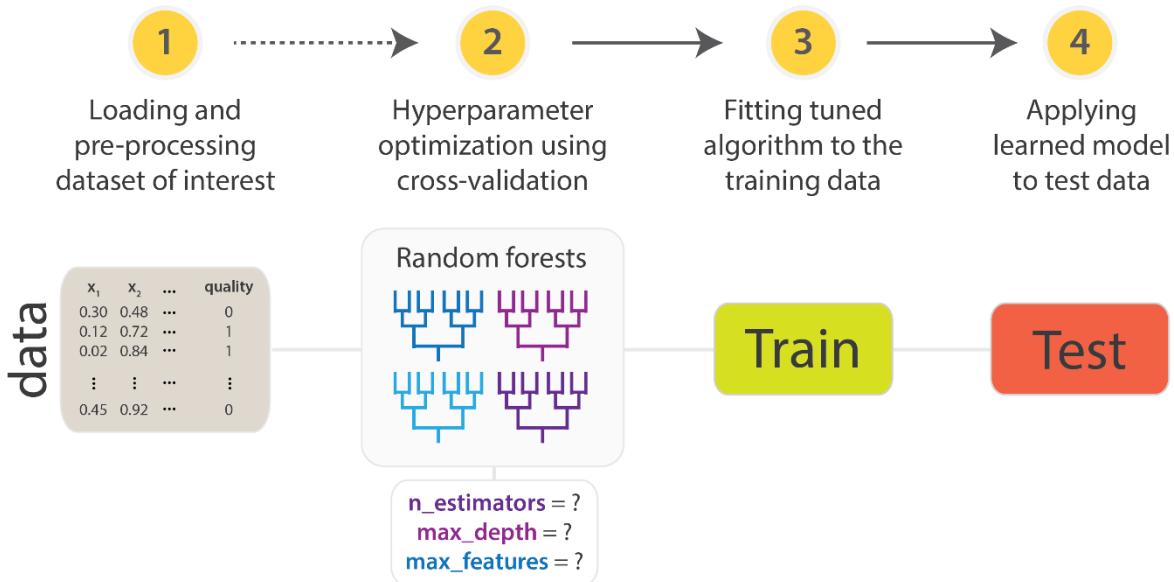
Non-parametric models are those with an unbounded number of parameters, allowing for more flexibility. To predict new data, you need to know the parameters of the model and the state of the data that has been observed. Examples include decision trees, k-nearest neighbors, and topic models using latent Dirichlet analysis.

## 10. How to ensure that your model is not overfitting?

Keep the design of the model simple. Try to reduce the noise in the model by considering fewer variables and parameters. Cross-validation techniques such as K-folds cross-validation help us keep overfitting under control. Regularization techniques such as LASSO help in avoiding overfitting by penalizing certain parameters if they are likely to cause overfitting.

## 11. How Much Data You should have to use For Training and Testing your Model?

You have to find a balance, and there's no right answer for every problem.



If your test set is too small, you'll have an unreliable estimation of model performance (performance statistic will have high variance). If your training set is too small, your actual model parameters will have a high variance.

A good rule of thumb is to use an 80/20 train/test split. Then, your train set can be further split into train/validation or into partitions for cross-validation.

12. What should you do when your model is suffering from low bias and high variance?

When the model's predicted value is very close to the actual value the condition is known as low bias. In this condition, we can use bagging algorithms like random forest regressor

13. What Is Bagging Algorithm?

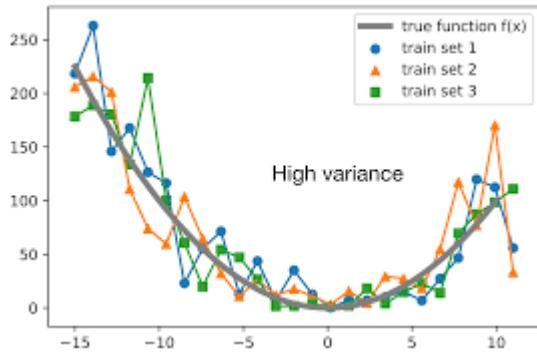
Bagging, or Bootstrap Aggregating, is an ensemble method in which the dataset is first divided into multiple subsets through resampling.

Then, each subset is used to train a model, and the final predictions are made through voting or averaging the component models.

Bagging is performed in parallel.

14. You came to know that your model is suffering from low bias and high variance. Which algorithm should you use to tackle it? Why?

Low bias occurs when the model's predicted values are near to actual values. In other words, the model becomes flexible enough to mimic the training data distribution. While it sounds like a great achievement, but not to forget, a flexible model has no generalization capabilities. It means, when this model is tested on unseen data, it gives disappointing results.



In such situations, we can use a bagging algorithm (like random forest) to tackle high variance problems. Bagging algorithms divide a data set into subsets made with repeated randomized sampling. Then, these samples are used to generate a set of models using a single learning algorithm. Later, the model predictions are combined using voting (classification) or averaging (regression).

Also, to combat high variance, we can:

1. Use the regularization techniques, where higher model coefficients get penalized, hence lowering model complexity.
2. Use top n features from the variable importance chart. Maybe, with all the variables in the data set, the algorithm is having difficulty in finding a meaningful signal.

## 15. List Down Advantages and Disadvantages of Neural Network

**Advantages:** Neural networks (specifically deep NNs) have led to performance breakthroughs for unstructured datasets such as images, audio, and video. Their incredible flexibility allows them to learn patterns that no other ML algorithm can learn.

**Disadvantages:** However, they require a large amount of training data to converge. It's also difficult to pick the right architecture, and the internal "hidden" layers are incomprehensible.

## 16. How do you think Google is training data for self-driving cars?



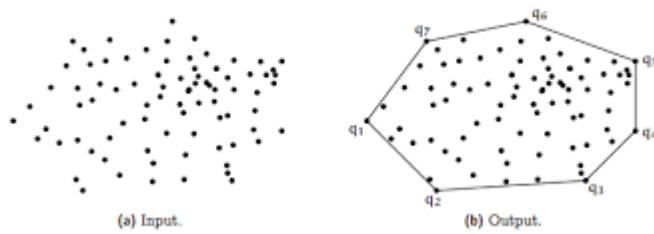
Google is currently using Recaptcha to source labeled data on storefronts and traffic signs. They are also building on training data collected by Sebastian Thrun at GoogleX — some of which was obtained by his grad students driving buggies on desert dunes!

## 17. How would you evaluate a logistic regression model?

A subsection of the question above. You have to demonstrate an understanding of what the typical goals of a logistic regression are (classification, prediction, etc.) and bring up a few examples and use cases.

## 18. What is Convex Hull?

In the case of linearly separable data, the convex hull represents the outer boundaries of the two groups of data points. Once the convex hull is created, we get maximum margin hyperplane (MMH) as a perpendicular bisector between two convex hulls.



MMH is the line which attempts to create the greatest separation between two groups.

## 1. What is the trade-off between bias and variance?

**Bias** (how well a model fits data) refers to errors due to inaccurate or simplistic assumptions in your ML algorithm, which leads to overfitting.

**Variance** (how much a model changes based on inputs) refers to errors due to complexity in your ML algorithm, which generates sensitivity to high levels of variation in training data and overfitting.

In other words, simple models are stable (low variance) but highly biased. Complex models are prone to overfitting but express the truth of the model (low bias). The optimal reduction of error requires a tradeoff of bias and variance to avoid both high variance and high bias.

## **2. Explain the difference between supervised and unsupervised machine learning.**

**Supervised learning** requires training labeled data. In other words, supervised learning uses a ground truth, meaning we have existing knowledge of our outputs and samples. The goal here is to learn a function that approximates a relationship between inputs and outputs.

**Unsupervised learning**, on the other hand, does not use labeled outputs. The goal here is to infer the natural structure in a dataset.

## **3. What are the most common algorithms for supervised learning and unsupervised learning?**

### **Supervised learning algorithms:**

- Linear regression
- Logistic regression
- Decision trees
- Random forests
- Naive Bayes
- Neural networks

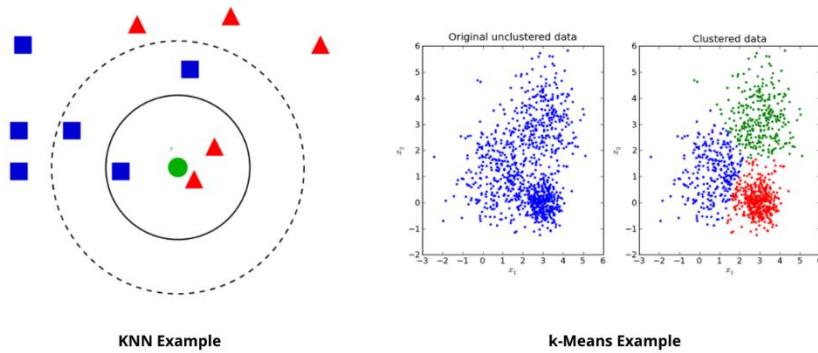
### **Examples of unsupervised algorithms:**

- Clustering: k-Means
- Visualization and dimensionality reduction
- Principal component analysis (PCA), t-distributed
- Stochastic neighbor embedding (t-SNE)
- Association rule learning (Apriori)

#### 4. Explain the difference between KNN and k-means clustering.

The main difference is that **KNN requires labeled points** (classification algorithm, supervised learning), but **k-means does not** (clustering algorithm, unsupervised learning).

To use K-Nearest Neighbors, you use labeled data that you want to classify into an unlabeled point. K-means clustering takes unlabeled points and learns how to group them using the mean of the distance between points.



#### 5. What is the Bayes' Theorem? Why do we use it?

Bayes' Theorem is how we find a probability when we know other probabilities. In other words, it provides the **posterior probability** of a prior knowledge event. This theorem is a principled way of calculating conditional probabilities.

In ML, Bayes' theorem is used in a probability framework that fits a model to a training dataset and for building classification predictive modeling problems (i.e. Naive Bayes, Bayes Optimal Classifier).

#### 6. What are Naive Bayes classifiers? Why do we use them?

Naive Bayes classifiers are a **collection of classification algorithms**. These classifiers are a family of algorithms that share a common principle. Naive Bayes classifiers assume that the occurrence or absence of a feature does not influence the presence or absence of another feature.

In other words, we call this “naive”, as it assumes that all dataset features are equally important and independent.

Naive Bayes classifiers are used for classification. When the assumption of independence holds, they are easy to implement and yield better results than

other sophisticated predictors. They are used in spam filtering, text analysis, and recommendation systems.

## 7. Explain difference between Type I and Type II error.

A Type I error is a **false positive** (claiming something has happened when it hasn't), and a Type II error is a **false negative** (claiming nothing has happened when it actually has).

## 8. What is the difference between a discriminative and a generative model?

A discriminative model learns **distinctions between different categories** of data. A generative model learns **categories of data**. Discriminative models generally perform better on classification tasks.

## 9. What are parametric models? Provide an example.

Parametric models have a **finite number of parameters**. You only need to know the parameters of the model to make a data prediction. Common examples are as follows: linear SVMs, linear regression, and logistic regression.

Non-parametric models have an **unbounded number of parameters** to offer flexibility. For data predictions, you need the parameters of the model and the state of the observed data. Common examples are as follows: k-nearest neighbors, decision trees, and topic models.

## 10. Explain the difference between an array and a linked list.

An array is an **ordered collection** of objects. It assumes that every element has the same size, since the entire array is stored in a contiguous block of memory. The size of an array is specified at the time of declaration and cannot be changed afterward.

Search options for an array are Linear search and Binary search (if it's sorted).

A linked list is a **series of objects** with pointers. Different elements are stored at different memory locations, and data items can be added or removed when desired.

The only search option for a linked list is Linear.

### Additional beginner questions may include:

- Which is more important: model performance or accuracy? Why?
- What's the F1 score? How is it used?
- What is the Curse of Dimensionality?

- When should we use classification rather than regression?
- Explain Deep Learning. How does it differ from other techniques?
- Explain the difference between likelihood and probability.

.1. Which cross-validation technique would you choose for a time series dataset?

A time series is not randomly distributed but has a chronological ordering. You want to use something like **forward chaining** so you can model based on past data before looking at future data. For example:

- Fold 1 : training [1], test [2]
- Fold 2 : training [1 2], test [3]
- Fold 3 : training [1 2 3], test [4]
- Fold 4 : training [1 2 3 4], test [5]
- Fold 5 : training [1 2 3 4 5], test [6]

## **2. How do you choose a classifier based on a training set size?**

For a small training set, a model with high bias and low variance models is better, as it is less likely overfit. An example is Naive Bayes.

For a large training set, a model with low bias and high variance models is better, as it expresses more complex relationships. An example is Logistic Regression.

## **3. Explain the ROC Curve and AUC.**

The ROC curve is a **graphical representation** of the performance of a classification model at all thresholds. It has two thresholds: true positive rate and false positive rate.

AUC (Area Under the ROC Curve) is, simply, the area under the ROC curve. AUC measures the two-dimensional area underneath the ROC curve from (0,0) to (1,1). It used as a performance metric for evaluating binary classification models.

#### **4. Explain LDA for unsupervised learning.**

Latent Dirichlet Allocation (LDA) is a common method for **topic modeling**. It is a generative model for representing documents as a combination of topics, each with their own probability distribution.

LDA aims to project the features of higher dimensional space onto a lower-dimensional space. This helps to avoid the curse of dimensionality.

#### **5. How do you ensure you are not overfitting a model?**

There are three methods we can use to prevent overfitting:

1. Use **cross-validation** techniques (like k-folds cross-validation)
2. Keep the model **simple** (i.e. take in fewer variables) to reduce variance
3. Use **regularization techniques** (like LASSO) that penalize model parameters likely to cause overfitting

#### **6. In SQL, how are primary and foreign keys related?**

SQL is one of the most popular data formats used in ML, so you need to demonstrate your ability to manipulate SQL databases.

Foreign keys allow you to **match and join tables** on the primary key of the corresponding table.

If you encounter this question, answer the basic concept, and then explain how you would set up SQL tables and query them.

#### **7. What evaluation approaches would you use to gauge the effectiveness of an ML model?**

First, you would split the dataset into training and test sets. You could also use a cross-validation technique to segment the dataset. Then, you would select and implement performance metrics. For example, you could use the confusion matrix, the F1 score and accuracy.

You'll want to explain the nuances of how a model is measured based on different parameters. Interviewees that stand out take questions like these one step further.

## **8. Explain how to handle missing or corrupted data in a dataset.**

You need to identify the find data and drop the rows/columns, or replace them with other values.

Pandas provides useful methods for doing this: `isnull()` and `dropna()`. These allow you to identify and drop corrupted data. The `fillna()` method can be used to fill invalid values with placeholders.

## **9. Explain how you would develop a data pipeline.**

Data pipelines enable us to take a data science model and automate or scale it. A common data pipeline tool is Apache Airflow, and Google Cloud, Azure, and AWS are used to host them.

For a question like this, you want to explain the required steps and discuss real experience you have building data pipelines.

**The basic steps are as follows for a Google Cloud host:**

1. Sign into Google Cloud Platform
2. Create a compute instance
3. Pull tutorial contents from GitHub
4. Use AirFlow for an overview of the pipeline
5. Use Docker to set up virtual hosts
6. Develop a Docker container
7. Open Airflow UI and run the ML pipeline
8. Run the deployed web app

## **10. How do you fix high variance in a model?**

If the model has low variance and high bias, we use a bagging algorithm, which divides a data set into subsets using randomized sampling. We use those samples to generate a set of models with a single learning algorithm.

Additionally, we can use the regularization technique, in which higher model coefficients are penalized to lower the complexity overall.

## **11. What are hyperparameters? How do they differ from model parameters?**

A model parameter is a variable that is **internal to the model**. The value of a parameter is estimated from training data.

A hyperparameter is a variable that is **external to the model**. The value cannot be estimated from data, and they are commonly used to estimate model parameters.

## **12. You are working on a dataset. How do you select important variables?**

- Remove correlated variables before selecting important variables
- Use Random Forest and a plot variable importance chart
- Use Lasso Regression
- Use linear regression to select variables based on p values
- Use Forward Selection, Stepwise Selection, and Backward Selection

## **13. How do you choose which algorithm to use for a dataset?**

Choosing an ML algorithm depends of the **type of data** in question. Business requirements are necessary for choosing an algorithm and building a model as well, so when answering this question, explain that you need more information.

For example, if your data organizes in a linear fashion, linear regression would be a good algorithm to use. Or, if the data is made up of non-linear interactions, a bagging or boosting algorithm is best. Or, if you're working with images, a neural network would be best.

## **14. What are advantages and disadvantages of using neural networks?**

### **Advantages:**

- Store data on the entire network rather than a database
- Parallel processing
- Distributed memory
- Provides great accuracy even with limited information

### **Disadvantages:**

- Requires complex processors
- Duration of a network is somewhat unknown
- We rely on error value too heavily
- Black-box nature

## **15. What is the default method for splitting in decision trees?**

The default method is the **Gini Index**, which is the measure of impurity of a particular node. Essentially, it calculates the probability of a specific feature that is classified incorrectly. When the elements are linked by a single class, we call this “pure”.

You could also use Random Forest, but the Gini Index is preferred because it isn't computationally intensive and doesn't involve logarithm functions.

### **Additional intermediate questions may include:**

- What is a Box-Cox transformation?
- Water Tapping problem
- Explain the advantages and disadvantages of decision trees.
- What is the exploding gradient problem when using back propagation technique?
- What is a confusion matrix? Why do you need it

### **1. You are given a data set with missing values that spread along 1 standard deviation from the median. What percentage of data would remain unaffected?**

The data is spread across median, so we can assume we're working with **normal distribution**. This means that approximately 68% of the data lies at 1 standard deviation from the mean. So, around 32% of the data unaffected.

### **2. You are told that your regression model is suffering from multicollinearity. How do verify this is true and build a better model?**

You should create a correlation matrix to identify and remove variables with a correlation above 75%. Keep in mind that our threshold here is subjective.

You could also calculate **VIF (variance inflation factor)** to check for the presence of multicollinearity. A VIF value greater than or equal to 4 suggests that there is no multicollinearity. A value less than or equal to 10 tells us there are serious multicollinearity issues.

You can't just remove variables, so you should use a penalized regression model or add random noise in the correlated variables, but this approach is less ideal.

### **3. Why does XGBoost perform better than SVM?**

XGBoos is an **ensemble method** that uses many trees. This means it improves as it repeats itself.

SVM is a **linear separator**. So, if our data is not linearly separable, SVM requires a Kernel to get the data to a state where it can be separated. This can limit us, as there is not a perfect Kernel for every given dataset.

#### **4. You build a random forest model with 10,000 trees. Training error as at 0.00, but validation error is 34.23. Explain what went wrong.**

Your model is likely **overfitted**. A training error of 0.00 means that the classifier has mimicked training data patterns. This means that they aren't available for our unseen data, returning a higher error.

When using random forest, this will occur if we use a large amount of trees.

#### **5. Explain the stages for building an ML model.**

This will largely depend on the model at hand, so you could ask clarifying questions. But generally, the process is as follows:

1. Understand the business model and end goal
2. Gather data acquisitions
3. Do data cleaning
4. Basic exploratory data analysis
5. Use machine learning algorithms to develop a model
6. Use an unknown dataset to check accuracy

#### **6. What is the recall, specificity and precision of the confusion matrix below?**

- TP / True Positive: the case was positive, and it was predicted as positive
- TN / True Negative: the case was negative, and it was predicted as negative
- FN / False Negative: the case was positive, but it was predicted as negative
- FP / False Positive: the case was negative, but it was predicted as positive

		Predicted	
		Apples	Oranges
Actual		Apples	TP = 10
		Oranges	FP = 35
		Apples	FN = 40
		Oranges	TN = 15

- Recall = 20%
- Specificity = 30%
- Precision = 22%

### Explanation:

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}) = 10 / 50 = 0.2 = 20\%$$

$$\text{Specificity} = \text{TN} / (\text{TN} + \text{FP}) = 15 / 50 = 0.3 = 30\%$$

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}) = 10 / 45 = 0.22 = 22\%$$

## 7. For NLP, what's the main purpose of using an encoder-decoder model?

We use the encoder-decoder model to generate an output sequence based on an input sequence.

What makes an encoder-decoder model so powerful is that the decoder uses the final state of the encoder as its initial state. This gives the decoder access to the information that the encoder extracted from the input sequence.

## 8. For Deep Learning with TensorFlow, which value is required as an input to an evaluation EstimatorSpec?

The loss metric is required. In model execution with TensorFlow, we use the `EstimatorSpec` object to organize training, evaluation, and prediction.

The `EstimatorSpec` object is initialized with a single required argument, called `mode`. The mode can take one of three values:

- `tf.estimator.ModeKeys.TRAIN`

- `tf.estimator.ModeKeys.EVAL`
- `tf.estimator.ModeKeys.PREDICT`

The keyword arguments required to initialize the `EstimatorSpec` will differ depending on the mode.

## **9. When using scikit-learn, is it true that we need to scale our feature values when they vary greatly?**

Yes. Most of the machine learning algorithms use Euclidean distance as the metrics to measure the distance between two data points. If the range of values is different greatly, the result of the same change in the different features will be very different.

## **10. Your dataset has 50 variables, but 8 variables have missing values higher than 30%. How do you address this?**

There are three general approaches you could take:

1. Just remove them (not ideal)
2. Assign a unique category to the missing values to see if there is a trend generating this issue
3. Check distribution with the target variable. If a pattern is found, keep the missing values, assign them to a new category, and remove the others.

### **Additional advanced questions may include:**

- You must evaluate a regression model based on  $R^2$ , adjusted  $R^2$  and tolerance. What are your criteria?
- For k-means or kNN, why do we use Euclidean distance over Manhattan distance?
- Linear regression models are usually evaluated using Adjusted  $R^2$  or an F value. How would you evaluate a logistic regression model?
- Explain the difference between the normal soft margin SVM and SVM with a linear kernel.

## **1. How would you implement a recommendation system for our users?**

Many ML interview questions like this involve implementing models to an organization's specific problems. To answer this question well, you need to research the company in advance. Read about revenue drivers and user base.

**Important:** Use questions like these to demonstrate your system design skills! You need to sketch out a solution with requirements, metrics, training data generation, and ranking.

Grokking the Machine Learning Interview goes over this question in detail using Netflix's recommendation system.

**The general steps for setting up a recommendation system are as follows:**

- Set up the problem by asking questions
- Understand scale and latency requirements
- Define the metrics for both online and offline testing
- Discuss the architecture of the system (how the data will flow)
- Discuss training data generation
- Outline feature engineering (what actors are involved)
- Discuss model training and algorithms
- Suggest how you'd scale and improve once it is deployed (i.e. issues you can predict)

## **2. What do you think is the most valuable data in our business?**

This tests your knowledge of the business/industry. It also tests for how you correlated data to business outcomes and applies it to a particular company's needs. You need to research an organization's business model. Be sure to ask questions to clarify the question further before jumping in.

**Some general answers could be:**

- Quality data that is understood by ML teams is useful for scaling and making correct predictions
- Data that tells us what the customer wants is essential for all business decisions
- Better data management can increase their annual revenue
- The types of data most valuable to a company is customer data, IT data, and internal financial data

## **3. How would you structure the ad selection process for an ad prediction system?**

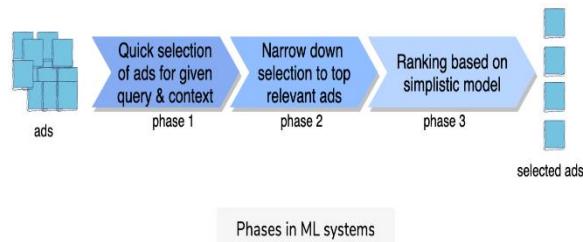
The main goal of an ads selection component is to narrow down the set of ads that are relevant for a given query. In a search-based system, the ads selection

component is responsible for retrieving the top relevant ads from the ads database according to the user and query context.

In a feed-based system, the ads selection component will select the top k relevant ads based more on user interests than search terms.

Here is a general solution to this question. Say we use a funnel-based approach for modeling. It would make sense to structure the ad selection process in these three phases:

- **Phase 1:** Quick selection of ads for the given query and user context according to selection criteria
- **Phase 2:** Rank these selected ads based on a simple and fast algorithm to trim ads.
- **Phase 3:** Apply the machine learning model on the trimmed ads to select the top ones.

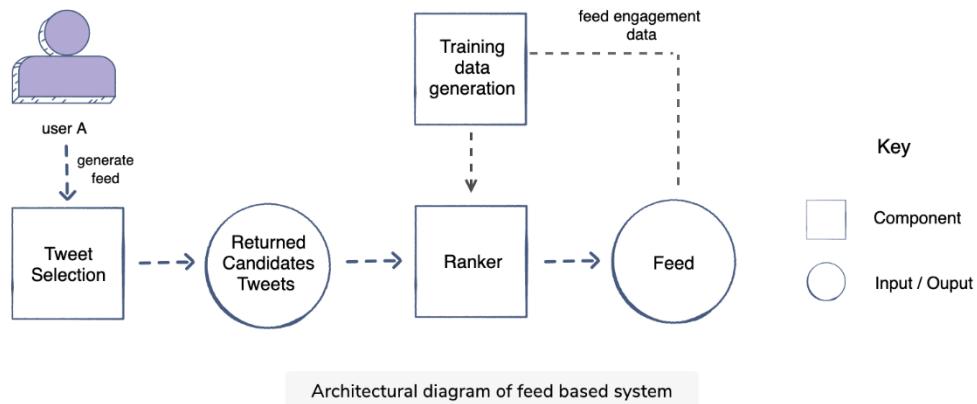


#### 4. What are the architectural components for a feed based system?

Again, this question largely depends on the organization in question. You'll first want to ask clarifying questions about the system to make sure you meet all its needs. You can speak in hypotheticals to leave room for inaccuracy.

I will explain it using Twitter's feed system to give you a sense of how to approach a problem like this. It will include:

- **Tweet selection:** a user's pool of Tweets is forwarded to ranker components
- **Training data generation:** positive and negative training examples
- **Ranker:** For predicting probability of engagement



## 5. What do you think about GPT-3? How do you think we can use it?

This question gauges your investment in the industry and your vision for how to apply new technologies. GPT-3 is a new language generation model that can generate human-like text.

There are many perspectives on GPT-3, so do some reading on how it's being used to demonstrate next-generation critical thinking.

**Some general answers could be:**

- Improving chatbots and customer service automation
- Improving search engines with NLP
- Job training and presentations for ongoing learning
- Improving JSX code
- Simplifying UI/UX design

### What is the difference between supervised learning and unsupervised learning?

The biggest difference is that unsupervised learning does not require explicitly labeled data, while supervised learning does – before you can do a classification, you must label the data to train the model to classify data into the correct groups.

- What are the different types of machine learning?

- What is deep learning, and how does it contrast with other machine learning algorithms?
- What are the differences between machine learning and deep learning?
- Explain the confusion matrix with respect to machine learning algorithms.
- What is the difference between artificial intelligence and machine learning?
- What's the trade-off between bias and variance?
- Explain the difference between L1 and L2 regularization.
- What's your favorite algorithm, and can you explain it to me in less than a minute?
- How is KNN different from k-means clustering?
- What is cross validation and what are different methods of using it?
- Explain how a ROC curve works.
- What's the difference between probability and likelihood?
- What's the difference between a generative and discriminative model?
- How is a decision tree pruned?
- How can you choose a classifier based on a training set size?
- What methods for dimensionality reduction do you know and how do they compare with each other?
- Define precision and recall.
- What's a Fourier transform?

- What's the difference between Type I and Type II error?
- When should you use classification over regression?
- How would you evaluate a logistic regression model?
- What is Bayes' Theorem? How is it useful in a machine learning context?
- Describe a hash table.

## Machine Learning Engineer Interviews Questions: Technical Skills

The company will want to make sure you have the hard skills needed to excel in the Machine Learning Engineer position. For technical questions, remember that interviewers are usually more interested in your thought process than the final solution.

Technical machine learning interview questions may include:

### **What's the difference between a Type I and II error?**

This is the type of basic question that could trip someone up in an interview, just because the wording of your answer could be a bit confusing. A Type I error is of course a false positive – when you think something has happened and it really hasn't – while a Type II is a false negative, or a situation where something is happening and it's missed.

- How would you handle an imbalanced dataset?
- How do you handle missing or corrupted data in a dataset?

- Do you have experience with Spark or big data tools for machine learning?
- Pick an algorithm. Write the pseudo-code for a parallel implementation.
- Which data visualization libraries do you use? What are your thoughts on the best data visualization tools?
- Given two strings, A and B, of the same length n, find whether it is possible to cut both strings at a common point such that the first part of A and the second part of B form a palindrome.
- How would you build a data pipeline?
- How would you implement a recommendation system for our company's users?
- Can you explain your approach to optimizing auto-tagging?
- Suppose you are given a data set that has missing values spread along 1 standard deviation from the median. What percentage of data would remain unaffected and why?
- Suppose you found that your model is suffering from low bias and high variance. Which algorithm do you think could tackle this situation and why?
- You are given a data set. The data set contains many variables, some of which are highly correlated and you know about it. Your manager has asked you to run PCA.
- Would you remove correlated variables first? Why?

- What are the advantages and disadvantages of neural networks?
- How would you go about understanding the sorts of mistakes an algorithm makes?
- Explain the steps involved in making decision trees.

### **Q1. Define P-value.**

When a decision has to be made about a hypothesis test, P-values are used. The P-value is the minimum level at which a null hypothesis is rejected. The lower the p-value, the likelier you'll reject the null hypothesis.

### **Q2. What do you mean by Reinforcement Learning?**

Unlike the other kinds of learning, such as supervised and unsupervised, neither data nor labels are provided in reinforcement learning. Our learning depends on the rewards provided to the agent by the environment.

### **Q3. How does one check the Normality of a dataset?**

When looking at it visually, certain plots can be used. Some normality checks have been given below:

- Shapiro-Wilk Test
- Anderson-Darling Test
- Martinez-Iglewicz Test
- Kolmogorov-Smirnov Test
- D'Agostino Skewness Test

### **Q4. Explain a Random Forest and its functioning.**

A versatile machine learning method that can perform both — regression and classification tasks — is known as a random forest. Like bagging and boosting, this method combines a set of other tree models.

It creates a tree using a random sample from the columns in the test data. The steps involved in the creation of trees in a random forest are:

- Procure a sample size from the training data.

- Start with a single node.
- Using the start node, run the algorithm given below:
  1. Stop if the number of observations in total is less than the node size.
  2. Pick random variables.
  3. Figure out the variable responsible for doing the ‘best’ job of splitting the observations.
  4. Divide the observations into two nodes.
  5. Implement step ‘a’ on both these nodes.

## **Q5. How would you define a neural network?**

A simplified model of the human brain is known as a neural network. Like the brain, the model has neurons that activate when they encounter something similar. The different neurons are connected through the connections that provide information flow from neuron to neuron.

## **Q6. How to deal with overfitting and underfitting?**

Overfitting refers to the model that is fitted to training data well. When it comes to this case, the data needs to be resampled, and the model accuracy needs to be estimated using techniques such as k-fold cross-validation.

On the other hand, in the case of underfitting, we can’t understand or gather the patterns from the data. We either have to change the algorithms or feed more data points to the model when this happens.

## **Q7. Define ensemble learning.**

The process of combining various machine learning models to develop more powerful models is known as ensemble learning. Now, a model can be different for many reasons. Some of these are:

- Different Population
- Different Hypothesis
- Different modeling techniques

As one works with the model’s training and testing data, an error occurs. It might just be a bias, variance, and irreducible error. The model always needs to strike a balance between bias and variance. This is called a bias-variance trade-off. Essentially, ensemble learning is a way that’s used to perform this trade-off.

Many ensemble techniques are available. However, when aggregating multiple models, usually there are only two methods:

- Bagging (native method): In this method, you take the training set and then generate new ones from it.
- Boosting (more elegant method): This method is similar to bagging. It is used to optimize the best weighing schemes for a training set.

## **Q8. How to know which machine learning algorithm to use?**

The answer to this question is dependent on the dataset you have. Linear progression is used whenever the dataset is continuous. There isn't any particular way that determines which ML algorithm should be used. It varies based on the exploratory data analysis (EDA).

You can think of EDA as something that ‘interviews’ the dataset. Now, as a part of this interview, the following things are done:

- Segregation of variables as continuous, categorical, and so on.
- Summarization of variables using descriptive statistics.
- Visualization of variables using charts.

Depending on the above observations, you choose the algorithm that best fits the particular dataset.

## **Q9. How should outlier values be handled?**

An observation in the dataset that is pretty far from the others in the dataset is known as an outlier. The following tools can be used to discover outliers:

- Box plot
- Z-score
- Scatter plot, etc.

Usually, three simple strategies can be followed to handle outliers:

- Drop them.
- Mark them as outliers and then include them as a feature.
- Similarly, the feature can be transformed to decrease the effect of the outlier.

## **Q10. How can you select K for K-means Clustering?**

To select K, two methods can be used. These are:

- Direct methods: These contain elbow and silhouette.
- Statistical testing methods: These have gap statistics.

Most often, the silhouette is used when the optimal value of k has to be determined.

1. If there is a data set with missing values spread along 1 standard deviation from the median, define the percentage of data that'll remain unaffected.
2. Explain why XGBoost performs better than SVM.
3. List the various stages involved in the development of an ML model.
4. When using scikit-learn, do we need to scale our feature values when they vary greatly?
5. Suppose a dataset has 50 variables. But out of these, 8 have values higher than 30%. How can this be addressed?
6. What is the difference between the normal soft margin SVM and SVM with a linear kernel?
7. Define loss and cost functions. What is the primary difference between them?
8. What do you mean by a generative model?
9. Explain the primary differences between classical and Bayesian statistics.
10. What is Bayes' theorem, and how does it work?
11. How would you explain the functioning of a recommendation system?
12. What would your criteria be if you had to evaluate a regression model based on  $R^2$ , adjusted  $R^2$ , and tolerance?
13. Define PCA and its function.
14. How does unsupervised learning differ from supervised learning?
15. Linear regression models are usually evaluated using Adjusted  $R^2$  or an F value. How is a logistic regression model evaluated?

## **Q3. What is bias in machine learning?**

The phenomenon that changes the result of an algorithm in favor of or against a particular idea is known as a bias. It is considered a systematic error in the machine learning model because of incorrect assumptions in the machine learning process.

### **1. How would you explain Machine Learning to a school-going kid?**

Machine learning is an application of Artificial Intelligence where we give machines access to data and let them use that data to learn for themselves. Then, you can input new conditions and it will predict the outcome. It's basically getting a computer to perform a task without explicitly being programmed to do so.

### **2. How does Deep Learning differ from Machine Learning?**

ML refers to an AI system that can self-learn based on the algorithm. Systems that get smarter and smarter over time without the human intervention is ML. Deep Learning is a machine learning applied to large data sets. Most AI work involves ML because intelligent behaviour requires considerable knowledge.

### **3. Explain Classification and Regression**

Classification is a process of categorizing a given set of data into classes, It can be performed on both structured or unstructured data. Regression in machine learning consists of mathematical methods that allow data scientists to predict a continuous outcome (y) based on the value of one or more predictor variables (x). Linear regression is probably the most popular form of regression analysis because of its ease-of-use in predicting and forecasting.

### **4. What do you understand by selection bias?**

Selection bias is a kind of error that occurs when the researcher decides who is going to be studied. It is usually associated with the research where the selection of participants isn't random.

### **5. What do you understand by Precision and Recall?**

Recall is the number of relevant documents retrieved by a search divided by the total number of the existing relevant documents, while precision is the number of relevant documents retrieved by a search divided by the total number of documents retrieved by that search.

### **6. What is a Confusion Matrix?**

The confusion is a 26 by 26 matrix with the probability of each reaction to each stimulus. This explains the name and matches the use in machine learning today.

## **7. What is the difference between inductive and deductive learning?**

The main difference between inductive and deductive reasoning is that inductive reasoning aims at developing a theory while deductive reasoning aims at testing an existing theory. Inductive reasoning moves from the specific observations to broad generalizations, and deductive reasoning the other way around.

## **8. How is KNN different from K-means clustering?**

K-means is an unsupervised learning algorithm used for the clustering problem whereas KNN is a supervised learning algorithm used for classification and regression problem. This is the basic difference between K-means and KNN algorithm. It makes predictions by learning from the past available data.

## **9. What is ROC curve and what does it represent?**

An ROC curve is a graph showing the performance of a classification model at all the classification thresholds. This curve plots two parameters: True Positive Rate. False Positive Rate.

## **10. What's the difference between Type I and Type II error?**

Type 1 error, in statistical hypothesis testing, is the error caused by rejecting a null hypothesis when it is true. Type II error is the error that occurs when the null hypothesis is accepted when it is not true. Type I error is equivalent to a false positive. Type II error is equivalent to a false negative.

## **11. Is it better to have too many false positives or too many false negatives? Explain.**

In medical testing, false negatives may provide a falsely reassuring message to patients and physicians that the disease is absent, when it is actually present. This sometimes leads to inappropriate or inadequate treatment of both the patient and their disease. So, it is desired to have too many false positive.

## **12. Which is more important to you – model accuracy or model performance?**

The accuracy extremely critical, even if the models would take minutes or hours to make a prediction. Other applications require the real time performance, even if this comes at a cost of accuracy.

### **13. What is the difference between Entropy and Information Gain?**

The information gain is the amount of information gained about a random variable or signal from observing another random variable. Entropy is that the average rate at which information is produced by a stochastic source of data, Or, it is a measure of the uncertainty associated with a random variable.

### **14. Explain Ensemble learning technique in Machine Learning.**

Ensemble methods are meta-algorithms that combine several machine learning techniques into the one predictive model in order to decrease variance (bagging), bias (boosting), or improve predictions (stacking).

### **15. What is bagging and boosting in Machine Learning?**

Bagging is a method of merging the same type of predictions. Boosting is a method of the merging different types of predictions. Bagging decreases variance, not bias, and solves over-fitting issues in a model. Boosting decreases bias, not variance.

### **16. What are collinearity and multicollinearity?**

Collinearity is a linear association between the two predictors. Multicollinearity is a situation where two or more predictors are highly linearly related. In general, an absolute correlation coefficient of  $>0.7$  among two or more predictors indicates the presence of multicollinearity.

Q1. You are given a train data set having 1000 columns and 1 million rows. The data set is based on a classification problem. Your manager has asked you to reduce the dimension of this data so that model computation time can be reduced. Your machine has memory constraints. What would you do? (You are free to make practical assumptions.)

Answer: Processing a high dimensional data on a limited memory machine is a strenuous task, your interviewer would be fully aware of that. Following are the methods you can use to tackle such situation:

Since we have lower RAM, we should close all other applications in our machine, including the web browser, so that most of the memory can be put to use.

We can randomly sample the data set. This means, we can create a smaller data set, let's say, having 1000 variables and 300000 rows and do the computations. To reduce dimensionality, we can separate the numerical and categorical variables and remove the correlated variables. For numerical variables, we'll use correlation. For categorical variables, we'll use chi-square test.

Also, we can use PCA and pick the components which can explain the maximum variance in the data set.

Using online learning algorithms like Vowpal Wabbit (available in Python) is a possible option.

Building a linear model using Stochastic Gradient Descent is also helpful.

We can also apply our business understanding to estimate which all predictors can impact the response variable. But, this is an intuitive approach, failing to identify useful predictors might result in significant loss of information.

Note: For point 4 & 5, make sure you read about online learning algorithms & Stochastic Gradient Descent. These are advanced methods.

Q2. Is rotation necessary in PCA? If yes, Why? What will happen if you don't rotate the components?

Answer: Yes, rotation (orthogonal) is necessary because it maximizes the difference between variance captured by the component. This makes the components easier to interpret. Not to forget, that's the motive of doing PCA where, we aim to select fewer components (than features) which can explain the maximum variance in the data set. By doing rotation, the relative location of the components doesn't change, it only changes the actual coordinates of the points.

If we don't rotate the components, the effect of PCA will diminish and we'll have to select more number of components to explain variance in the data set.

Q3. You are given a data set. The data set has missing values which spread along 1 standard deviation from the median. What percentage of data would remain unaffected? Why?

Answer: This question has enough hints for you to start thinking! Since, the data is spread across median, let's assume it's a normal distribution. We know, in a

normal distribution, ~68% of the data lies in 1 standard deviation from mean (or mode, median), which leaves ~32% of the data unaffected. Therefore, ~32% of the data would remain unaffected by missing values.

Q4. You are given a data set on cancer detection. You've build a classification model and achieved an accuracy of 96%. Why shouldn't you be happy with your model performance? What can you do about it?

Answer: If you have worked on enough data sets, you should deduce that cancer detection results in imbalanced data. In an imbalanced data set, accuracy should not be used as a measure of performance because 96% (as given) might only be predicting majority class correctly, but our class of interest is minority class (4%) which is the people who actually got diagnosed with cancer. Hence, in order to evaluate model performance, we should use Sensitivity (True Positive Rate), Specificity (True Negative Rate), F measure to determine class wise performance of the classifier. If the minority class performance is found to be poor, we can undertake the following steps:

We can use undersampling, oversampling or SMOTE to make the data balanced.

We can alter the prediction threshold value by doing probability calibration and finding a optimal threshold using AUC-ROC curve.

We can assign weight to classes such that the minority classes gets larger weight.

We can also use anomaly detection.

Q5. Why is naive Bayes so ‘naive’ ?

Answer: naive Bayes is so ‘naive’ because it assumes that all of the features in a data set are equally important and independent. As we know, these assumption are rarely true in real world scenario.

Q6. Explain prior probability, likelihood and marginal likelihood in context of naiveBayes algorithm?

Answer: Prior probability is nothing but, the proportion of dependent (binary) variable in the data set. It is the closest guess you can make about a class, without any further information. For example: In a data set, the dependent

variable is binary (1 and 0). The proportion of 1 (spam) is 70% and 0 (not spam) is 30%. Hence, we can estimate that there are 70% chances that any new email would be classified as spam.

Likelihood is the probability of classifying a given observation as 1 in presence of some other variable. For example: The probability that the word ‘FREE’ is used in previous spam message is likelihood. Marginal likelihood is, the probability that the word ‘FREE’ is used in any message.

Q7. You are working on a time series data set. Your manager has asked you to build a high accuracy model. You start with the decision tree algorithm, since you know it works fairly well on all kinds of data. Later, you tried a time series regression model and got higher accuracy than decision tree model. Can this happen? Why?

Answer: Time series data is known to possess linearity. On the other hand, a decision tree algorithm is known to work best to detect non – linear interactions. The reason why decision tree failed to provide robust predictions because it couldn’t map the linear relationship as good as a regression model did. Therefore, we learned that, a linear regression model can provide robust prediction given the data set satisfies its linearity assumptions.

Q8. You are assigned a new project which involves helping a food delivery company save more money. The problem is, company’s delivery team aren’t able to deliver food on time. As a result, their customers get unhappy. And, to keep them happy, they end up delivering food for free. Which machine learning algorithm can save them?

Answer: You might have started hopping through the list of ML algorithms in your mind. But, wait! Such questions are asked to test your machine learning fundamentals.

This is not a machine learning problem. This is a route optimization problem. A machine learning problem consists of three things:

There exist a pattern.

You cannot solve it mathematically (even by writing exponential equations).

You have data on it.

Always look for these three factors to decide if machine learning is a tool to solve a particular problem.

Q9. You came to know that your model is suffering from low bias and high variance. Which algorithm should you use to tackle it? Why?

Answer: Low bias occurs when the model's predicted values are near to actual values. In other words, the model becomes flexible enough to mimic the training data distribution. While it sounds like great achievement, but not to forget, a flexible model has no generalization capabilities. It means, when this model is tested on an unseen data, it gives disappointing results.

In such situations, we can use bagging algorithm (like random forest) to tackle high variance problem. Bagging algorithms divides a data set into subsets made with repeated randomized sampling. Then, these samples are used to generate a set of models using a single learning algorithm. Later, the model predictions are combined using voting (classification) or averaging (regression).

Also, to combat high variance, we can:

Use regularization technique, where higher model coefficients get penalized, hence lowering model complexity.

Use top n features from variable importance chart. May be, with all the variable in the data set, the algorithm is having difficulty in finding the meaningful signal.

Q10. You are given a data set. The data set contains many variables, some of which are highly correlated and you know about it. Your manager has asked you to run PCA. Would you remove correlated variables first? Why?

Answer: Chances are, you might be tempted to say No, but that would be incorrect. Discarding correlated variables have a substantial effect on PCA because, in presence of correlated variables, the variance explained by a particular component gets inflated.

For example: You have 3 variables in a data set, of which 2 are correlated. If you run PCA on this data set, the first principal component would exhibit twice the variance than it would exhibit with uncorrelated variables. Also, adding correlated variables lets PCA put more importance on those variable, which is misleading.

Q11. After spending several hours, you are now anxious to build a high accuracy model. As a result, you build 5 GBM models, thinking a boosting algorithm would do the magic. Unfortunately, neither of models could perform better than benchmark score. Finally, you decided to combine those models. Though, ensembled models are known to return high accuracy, but you are unfortunate. Where did you miss?

Answer: As we know, ensemble learners are based on the idea of combining weak learners to create strong learners. But, these learners provide superior result when the combined models are uncorrelated. Since, we have used 5 GBM models and got no accuracy improvement, suggests that the models are correlated. The problem with correlated models is, all the models provide same information.

For example: If model 1 has classified User1122 as 1, there are high chances model 2 and model 3 would have done the same, even if its actual value is 0. Therefore, ensemble learners are built on the premise of combining weak uncorrelated models to obtain better predictions.

Q12. How is kNN different from kmeans clustering?

Answer: Don't get mislead by 'k' in their names. You should know that the fundamental difference between both these algorithms is, kmeans is unsupervised in nature and kNN is supervised in nature. kmeans is a clustering algorithm. kNN is a classification (or regression) algorithm.

kmeans algorithm partitions a data set into clusters such that a cluster formed is homogeneous and the points in each cluster are close to each other. The algorithm tries to maintain enough separability between these clusters. Due to unsupervised nature, the clusters have no labels.

kNN algorithm tries to classify an unlabeled observation based on its k (can be any number ) surrounding neighbors. It is also known as lazy learner because it

involves minimal training of model. Hence, it doesn't use training data to make generalization on unseen data set.

Q13. How is True Positive Rate and Recall related? Write the equation.

Answer: True Positive Rate = Recall. Yes, they are equal having the formula  $(TP/TP + FN)$ .

Q14. You have built a multiple regression model. Your model  $R^2$  isn't as good as you wanted. For improvement, if you remove the intercept term, your model  $R^2$  becomes 0.8 from 0.3. Is it possible? How?

Answer: Yes, it is possible. We need to understand the significance of intercept term in a regression model. The intercept term shows model prediction without any independent variable i.e. mean prediction. The formula of  $R^2 = 1 - \sum(y - y')^2 / \sum(y - \text{ymean})^2$  where  $y'$  is predicted value.

When intercept term is present,  $R^2$  value evaluates your model wrt. to the mean model. In absence of intercept term ( $\text{ymean}$ ), the model can make no such evaluation, with large denominator,  $\sum(y - y')^2 / \sum(y)^2$  equation's value becomes smaller than actual, resulting in higher  $R^2$ .

Q15. After analyzing the model, your manager has informed that your regression model is suffering from multicollinearity. How would you check if he's true? Without losing any information, can you still build a better model?

Answer: To check multicollinearity, we can create a correlation matrix to identify & remove variables having correlation above 75% (deciding a threshold is subjective). In addition, we can use calculate VIF (variance inflation factor) to check the presence of multicollinearity. VIF value  $<= 4$  suggests no multicollinearity whereas a value of  $>= 10$  implies serious multicollinearity. Also, we can use tolerance as an indicator of multicollinearity.

But, removing correlated variables might lead to loss of information. In order to retain those variables, we can use penalized regression models like ridge or lasso regression. Also, we can add some random noise in correlated variable so that the variables become different from each other. But, adding noise might affect the prediction accuracy, hence this approach should be carefully used.

**Q16. When is Ridge regression favorable over Lasso regression?**

Answer: You can quote ISLR's authors Hastie, Tibshirani who asserted that, in presence of few variables with medium / large sized effect, use lasso regression. In presence of many variables with small / medium sized effect, use ridge regression.

Conceptually, we can say, lasso regression (L1) does both variable selection and parameter shrinkage, whereas Ridge regression only does parameter shrinkage and end up including all the coefficients in the model. In presence of correlated variables, ridge regression might be the preferred choice. Also, ridge regression works best in situations where the least square estimates have higher variance. Therefore, it depends on our model objective.

**Q17. Rise in global average temperature led to decrease in number of pirates around the world. Does that mean that decrease in number of pirates caused the climate change?**

Answer: After reading this question, you should have understood that this is a classic case of "causation and correlation". No, we can't conclude that decrease in number of pirates caused the climate change because there might be other factors (lurking or confounding variables) influencing this phenomenon.

Therefore, there might be a correlation between global average temperature and number of pirates, but based on this information we can't say that pirates died because of rise in global average temperature.

**Q18. While working on a data set, how do you select important variables?**

Explain your methods.

Answer: Following are the methods of variable selection you can use:

Remove the correlated variables prior to selecting important variables  
Use linear regression and select variables based on p values  
Use Forward Selection, Backward Selection, Stepwise Selection  
Use Random Forest, Xgboost and plot variable importance chart  
Use Lasso Regression

Measure information gain for the available set of features and select top n features accordingly.

Q19. What is the difference between covariance and correlation?

Answer: Correlation is the standardized form of covariance.

Covariances are difficult to compare. For example: if we calculate the covariances of salary (\$) and age (years), we'll get different covariances which can't be compared because of having unequal scales. To combat such situation, we calculate correlation to get a value between -1 and 1, irrespective of their respective scale.

Q20. Is it possible capture the correlation between continuous and categorical variable? If yes, how?

Answer: Yes, we can use ANCOVA (analysis of covariance) technique to capture association between continuous and categorical variables.

Q21. Both being tree based algorithm, how is random forest different from Gradient boosting algorithm (GBM)?

Answer: The fundamental difference is, random forest uses bagging technique to make predictions. GBM uses boosting techniques to make predictions.

In bagging technique, a data set is divided into n samples using randomized sampling. Then, using a single learning algorithm a model is build on all samples. Later, the resultant predictions are combined using voting or averaging. Bagging is done in parallel. In boosting, after the first round of predictions, the algorithm weighs misclassified predictions higher, such that they can be corrected in the succeeding round. This sequential process of giving higher weights to misclassified predictions continue until a stopping criterion is reached.

Random forest improves model accuracy by reducing variance (mainly). The trees grown are uncorrelated to maximize the decrease in variance. On the other hand, GBM improves accuracy by reducing both bias and variance in a model.

Q22. Running a binary classification tree algorithm is the easy part. Do you know how does a tree splitting takes place i.e. how does the tree decide which variable to split at the root node and succeeding nodes?

Answer: A classification trees makes decision based on Gini Index and Node Entropy. In simple words, the tree algorithm find the best possible feature which can divide the data set into purest possible children nodes.

Gini index says, if we select two items from a population at random then they must be of same class and probability for this is 1 if population is pure. We can calculate Gini as following:

Calculate Gini for sub-nodes, using formula sum of square of probability for success and failure ( $p^2+q^2$ ).  
Calculate Gini for split using weighted Gini score of each node of that split

Entropy is the measure of impurity as given by (for binary class):

Entropy, Decision Tree

Here p and q is probability of success and failure respectively in that node. Entropy is zero when a node is homogeneous. It is maximum when both the classes are present in a node at 50% – 50%. Lower entropy is desirable.

Q23. You've built a random forest model with 10000 trees. You got delighted after getting training error as 0.00. But, the validation error is 34.23. What is going on? Haven't you trained your model perfectly?

Answer: The model has overfitted. Training error 0.00 means the classifier has mimiced the training data patterns to an extent, that they are not available in the unseen data. Hence, when this classifier was run on unseen sample, it couldn't find those patterns and returned prediction with higher error. In random forest, it happens when we use larger number of trees than necessary. Hence, to avoid these situation, we should tune number of trees using cross validation.

Q24. You've got a data set to work having  $p$  (no. of variable)  $> n$  (no. of observation). Why is OLS as bad option to work with? Which techniques would be best to use? Why?

Answer: In such high dimensional data sets, we can't use classical regression techniques, since their assumptions tend to fail. When  $p > n$ , we can no longer calculate a unique least square coefficient estimate, the variances become infinite, so OLS cannot be used at all.

To combat this situation, we can use penalized regression methods like lasso, LARS, ridge which can shrink the coefficients to reduce variance. Precisely, ridge regression works best in situations where the least square estimates have higher variance.

Among other methods include subset regression, forward stepwise regression.

Q25. What is convex hull ? (Hint: Think SVM)

Answer: In case of linearly separable data, convex hull represents the outer boundaries of the two group of data points. Once convex hull is created, we get maximum margin hyperplane (MMH) as a perpendicular bisector between two convex hulls. MMH is the line which attempts to create greatest separation between two groups.

Q26. We know that one hot encoding increasing the dimensionality of a data set. But, label encoding doesn't. How ?

Answer: Don't get baffled at this question. It's a simple question asking the difference between the two.

Using one hot encoding, the dimensionality (a.k.a features) in a data set get increased because it creates a new variable for each level present in categorical variables. For example: let's say we have a variable 'color'. The variable has 3 levels namely Red, Blue and Green. One hot encoding 'color' variable will generate three new variables as Color.Red, Color.Blue and Color.Green containing 0 and 1 value.

In label encoding, the levels of a categorical variables gets encoded as 0 and 1, so no new variable is created. Label encoding is majorly used for binary variables.

Q27. What cross validation technique would you use on time series data set? Is it k-fold or LOOCV?

Answer: Neither.

In time series problem, k fold can be troublesome because there might be some pattern in year 4 or 5 which is not in year 3. Resampling the data set will separate these trends, and we might end up validation on past years, which is incorrect. Instead, we can use forward chaining strategy with 5 fold as shown below:

```
fold 1 : training [1], test [2]
fold 2 : training [1 2], test [3]
fold 3 : training [1 2 3], test [4]
fold 4 : training [1 2 3 4], test [5]
fold 5 : training [1 2 3 4 5], test [6]
```

where 1,2,3,4,5,6 represents “year”.

Q28. You are given a data set consisting of variables having more than 30% missing values? Let's say, out of 50 variables, 8 variables have missing values higher than 30%. How will you deal with them?

Answer: We can deal with them in the following ways:

Assign a unique category to missing values, who knows the missing values might decipher some trend

We can remove them blatantly.

Or, we can sensibly check their distribution with the target variable, and if found any pattern we'll keep those missing values and assign them a new category while removing others.

29. ‘People who bought this, also bought...’ recommendations seen on amazon is a result of which algorithm?

Answer: The basic idea for this kind of recommendation engine comes from collaborative filtering.

Collaborative Filtering algorithm considers “User Behavior” for recommending items. They exploit behavior of other users and items in terms of transaction history, ratings, selection and purchase information. Other users behaviour and preferences over the items are used to recommend items to the new users. In this case, features of the items are not known.

Know more: Recommender System

Q30. What do you understand by Type I vs Type II error ?

Answer: Type I error is committed when the null hypothesis is true and we reject it, also known as a ‘False Positive’. Type II error is committed when the null hypothesis is false and we accept it, also known as ‘False Negative’.

In the context of confusion matrix, we can say Type I error occurs when we classify a value as positive (1) when it is actually negative (0). Type II error occurs when we classify a value as negative (0) when it is actually positive(1).

Q31. You are working on a classification problem. For validation purposes, you've randomly sampled the training data set into train and validation. You are confident that your model will work incredibly well on unseen data since your validation accuracy is high. However, you get shocked after getting poor test accuracy. What went wrong?

Answer: In case of classification problem, we should always use stratified sampling instead of random sampling. A random sampling doesn't takes into consideration the proportion of target classes. On the contrary, stratified sampling helps to maintain the distribution of target variable in the resultant distributed samples also.

Q32. You have been asked to evaluate a regression model based on  $R^2$ , adjusted  $R^2$  and tolerance. What will be your criteria?

Answer: Tolerance (1 / VIF) is used as an indicator of multicollinearity. It is an indicator of percent of variance in a predictor which cannot be accounted by other predictors. Large values of tolerance is desirable.

We will consider adjusted R<sup>2</sup> as opposed to R<sup>2</sup> to evaluate model fit because R<sup>2</sup> increases irrespective of improvement in prediction accuracy as we add more variables. But, adjusted R<sup>2</sup> would only increase if an additional variable improves the accuracy of model, otherwise stays same. It is difficult to commit a general threshold value for adjusted R<sup>2</sup> because it varies between data sets. For example: a gene mutation data set might result in lower adjusted R<sup>2</sup> and still provide fairly good predictions, as compared to a stock market data where lower adjusted R<sup>2</sup> implies that model is not good.

Q33. In k-means or kNN, we use euclidean distance to calculate the distance between nearest neighbors. Why not manhattan distance ?

Answer: We don't use manhattan distance because it calculates distance horizontally or vertically only. It has dimension restrictions. On the other hand, euclidean metric can be used in any space to calculate distance. Since, the data points can be present in any dimension, euclidean distance is a more viable option.

Example: Think of a chess board, the movement made by a bishop or a rook is calculated by manhattan distance because of their respective vertical & horizontal movements.

Q34. Explain machine learning to me like a 5 year old.

Answer: It's simple. It's just like how babies learn to walk. Every time they fall down, they learn (unconsciously) & realize that their legs should be straight and not in a bend position. The next time they fall down, they feel pain. They cry. But, they learn 'not to stand like that again'. In order to avoid that pain, they try harder. To succeed, they even seek support from the door or wall or anything near them, which helps them stand firm.

This is how a machine works & develops intuition from its environment.

Note: The interview is only trying to test if have the ability of explain complex concepts in simple terms.

Q35. I know that a linear regression model is generally evaluated using Adjusted R<sup>2</sup> or F value. How would you evaluate a logistic regression model?

Answer: We can use the following methods:

Since logistic regression is used to predict probabilities, we can use AUC-ROC curve along with confusion matrix to determine its performance.

Also, the analogous metric of adjusted R<sup>2</sup> in logistic regression is AIC. AIC is the measure of fit which penalizes model for the number of model coefficients. Therefore, we always prefer model with minimum AIC value.

Null Deviance indicates the response predicted by a model with nothing but an intercept. Lower the value, better the model. Residual deviance indicates the response predicted by a model on adding independent variables. Lower the value, better the model.

**Q36.** Considering the long list of machine learning algorithm, given a data set, how do you decide which one to use?

Answer: You should say, the choice of machine learning algorithm solely depends of the type of data. If you are given a data set which is exhibits linearity, then linear regression would be the best algorithm to use. If you given to work on images, audios, then neural network would help you to build a robust model.

If the data comprises of non linear interactions, then a boosting or bagging algorithm should be the choice. If the business requirement is to build a model which can be deployed, then we'll use regression or a decision tree model (easy to interpret and explain) instead of black box algorithms like SVM, GBM etc.

In short, there is no one master algorithm for all situations. We must be scrupulous enough to understand which algorithm to use.

**Q37.** Do you suggest that treating a categorical variable as continuous variable would result in a better predictive model?

Answer: For better predictions, categorical variable can be considered as a continuous variable only when the variable is ordinal in nature.

**Q38.** When does regularization becomes necessary in Machine Learning?

Answer: Regularization becomes necessary when the model begins to overfit / underfit. This technique introduces a cost term for bringing in more features with the objective function. Hence, it tries to push the coefficients for many

variables to zero and hence reduce cost term. This helps to reduce model complexity so that the model can become better at predicting (generalizing).

**Q39.** What do you understand by Bias Variance trade off?

Answer: The error emerging from any model can be broken down into three components mathematically. Following are these component :

error of a model

Bias error is useful to quantify how much on an average are the predicted values different from the actual value. A high bias error means we have a under-performing model which keeps on missing important trends. Variance on the other side quantifies how are the prediction made on same observation different from each other. A high variance model will over-fit on your training population and perform badly on any observation beyond training.

**Q40.** OLS is to linear regression. Maximum likelihood is to logistic regression.

Explain the statement.

Answer: OLS and Maximum likelihood are the methods used by the respective regression methods to approximate the unknown parameter (coefficient) value. In simple words,

Ordinary least square(OLS) is a method used in linear regression which approximates the parameters resulting in minimum distance between actual and predicted values. Maximum Likelihood helps in choosing the the values of parameters which maximizes the likelihood that the parameters are most likely to produce observed data.

What do you mean by deep learning? How does it contrast with other traditional machine learnings?

What is EM algorithm and its applications?

What is a Fourier Transform?

What does ‘linear’ mean in a generalized linear model?

State an example of an application of non-negative matrix factorization.

Describe probabilistic graphical model.

How Bayesian networks differ from Markov networks?

What are the effective tactics for performing feature selection that does not involve exhaustive search?

What are the methods for dimensionality reduction?

Can you explain the concept of overfitting in a model?

Overfitting occurs when a model is too complex and can fit the noise in the data rather than the underlying patterns. It results in a model that performs well on the training data but poorly on new, unseen data.

Tips:

- Explain the concept of overfitting in simple terms, with an example.
- Mention techniques to prevent over fittings, such as regularisation and early stopping

How do you handle missing data in a dataset?

There are several ways to handle missing data, such as:

- Removing observations with missing data (listwise deletion)
- Imputing the missing values using statistical methods such as mean, median or mode.
- Using a machine learning algorithm that can handle missing data, such as decision trees or random forests.

Tips:

- Explain the pros and cons of each method and when to use them.
- Mention any specific libraries or packages that you have used to handle missing data in the past

Can you explain the bias-variance tradeoff in a model?

The bias-variance tradeoff is the balance between a model being too simple (high bias) and too complex (high variance). A model with high bias will have a low accuracy on both the training and test data, while a model with high variance will have a high precision on the training data but a low accuracy on the test data.

Tips:

- Use examples to explain bias-variance tradeoff
- Mention techniques to balance bias and variance, such as regularisation and ensemble methods.

How do you evaluate the performance of a model?

The performance of a model can be evaluated using various metrics such as accuracy, precision, recall, and F1-score for classification problems and mean squared error (MSE), and mean absolute error (MAE) for regression problems.

Additionally, k-fold cross-validation can estimate the model's performance on unseen data.

Tips:

- Explain what each metric measures and when to use them
- Mention any specific libraries or packages that you have used to evaluate the performance of a model in the past.

Can you walk us through a recent machine-learning project you worked on?

Provide a brief overview of the project, including the problem you were trying to solve, the dataset you used, and the techniques and models you employed. Highlight the key challenges you faced and how you overcame them, and discuss any interesting insights or results you obtained.

Tips:

- Prepare a clear and concise overview of the project before the interview
- Emphasise what you have learned and achieved from the project.
- Bring any relevant code, visualisations, or report you have prepared for the project to support your explanation.

## **1) What is the difference between inductive machine learning and deductive machine learning?**

In inductive machine learning, the model learns by examples from a set of observed instances to draw a generalized conclusion whereas in deductive learning the model first draws the conclusion and then the conclusion is drawn. Let's understand this with an example, for instance, if you have to explain to a kid that playing with fire can cause burns. There are two ways you can explain this to kids, you can show them training examples of various fire accidents or images with burnt people and label them as "Hazardous". In this case the kid will learn with the help of examples and not play with fire. This is referred to as Inductive machine learning. The other way is to let your kid play with fire and wait to see what happens. If the kid gets a burn they will learn not to play with fire and whenever they come across fire, they will avoid going near it. This is referred to as deductive learning.

## **2) How will you know which machine learning algorithm to choose for your classification problem?**

If accuracy is a major concern for you when deciding on a machine learning algorithm then the best way to go about it is test a couple of different ones (by

trying different parameters within each algorithm ) and choose the best one by cross-validation. A general rule of thumb to choose a good enough machine learning algorithm for your classification problem is based on how large your training set is. If the training set is small then using low variance/high bias classifiers like Naïve Bayes is advantageous over high variance/low bias classifiers like k-nearest neighbour algorithms as it might overfit the model. High variance/low bias classifiers tend to win when the training set grows in size.

### **3) Why is Naïve Bayes machine learning algorithm naïve?**

Naïve Bayes machine learning algorithm is considered Naïve because the assumptions the algorithm makes are virtually impossible to find in real-life data. Conditional probability is calculated as a pure product of individual probabilities of components. This means that the algorithm assumes the presence or absence of a specific feature of a class is not related to the presence or absence of any other feature (absolute independence of features), given the class variable. For instance, a fruit may be considered to be a banana if it is yellow, long and about 5 inches in length. However, if these features depend on each other or are based on the existence of other features, a naïve Bayes classifier will assume all these properties to contribute independently to the probability that this fruit is a banana. Assuming that all features in a given dataset are equally important and independent rarely exists in the real-world scenario.

### **4) How will you explain machine learning in to a layperson?**

Machine learning is all about making decisions based on previous experience with a task with the intent of improving its performance. There are multiple examples that can be given to explain machine learning to a layperson –

- Imagine a curious kid who sticks his palm
- You have observed from your connections that obese people often tend to get heart diseases thus you make the decision that you will try to remain thin otherwise you might suffer from a heart disease. You have observed a ton of data and come up with a general rule of classification.
- You are playing blackjack and based on the sequence of cards you see, you decide whether to hit or to stay. In this case based on the previous information you have and by looking at what happens, you make a decision quickly.

### **5) List out some important methods of reducing dimensionality.**

- Combine features with feature engineering.

- Use some form of algorithmic dimensionality reduction like ICA or PCA.
- Remove collinear features to reduce dimensionality.

**6) You are given a dataset where the number of variables (p) is greater than the number of observations (n) ( $p>n$ ). Which is the best technique to use and why?**

When the number of variables is greater than the number of observations, it represents a high dimensional dataset. In such cases, it is not possible to calculate a unique least square coefficient estimate. Penalized regression methods like LARS, Lasso or Ridge seem work well under these circumstances as they tend to shrink the coefficients to reduce variance. Whenever the least square estimates have higher variance, Ridge regression technique seems to work best.

**7) “People who bought this, also bought....” recommendations on Amazon are a result of which machine learning algorithm?**

Recommender systems usually implement the collaborative filtering machine learning algorithm that considers user behaviour for recommending products to users. Collaborative filtering machine learning algorithms exploit the behaviour of users and products through ratings, reviews, transaction history, browsing history, selection and purchase information.

**.8) Name some feature extraction techniques used for dimensionality reduction.**

- Independent Component Analysis
- Principal Component Analysis
- Kernel Based Principal Component Analysis

**9) List some use cases where classification machine learning algorithms can be used.**

- Natural language processing (Best example for this is Spoken Language Understanding )
- Market Segmentation
- Text Categorization (Spam Filtering )
- Bioinformatics (Classifying proteins according to their function)
- Fraud Detection
- Face detection

**10) What kind of problems does regularization solve?**

Regularization is used to address overfitting problems as it penalizes the loss function by adding a multiple of an L1 (LASSO) or an L2 (Ridge) norm of your weights vector  $w$ .

**11) How much data will you allocate for your training, validation and test sets?**

There is no to the point answer to this question but there needs to be a balance/equilibrium when allocating data for training, validation and test sets.

If you make the training set too small, then the actual model parameters might have high variance. Also, if the test set is too small, there are chances of unreliable estimation of model performance. A general thumb rule to follow is to use 80: 20 train/test split. After this the training set can be further split into validation sets.

**12) Which one would you prefer to choose – model accuracy or model performance?**

Model accuracy is just a subset of model performance but is not the be-all and end-all of model performance. This question is asked to test your knowledge on how well you can make a perfect balance between model accuracy and model performance.

**13) What is the most frequent metric to assess model accuracy for classification problems?**

Percent Correct Classification (PCC) measures the overall accuracy irrespective of the kind of errors that are made, all errors are considered to have same weight.

**14) When will you use classification over regression?**

Classification is about identifying group membership while regression technique involves predicting a response. Both techniques are related to prediction, where classification predicts the belonging to a class whereas regression predicts the value from a continuous set. Classification technique is preferred over regression when the results of the model need to return the belongingness of data points in a dataset to specific explicit categories. (For instance, when you want to find out whether a name is male or female instead of just finding it how correlated they are with male and female names).

**15) Why is Manhattan distance not used in kNN machine learning algorithm to calculate the distance between nearest neighbours?**

Manhattan distance has restrictions on dimensions and calculates the distance either vertically or horizontally. Euclidean distance is better option in kNN to calculate the distance between nearest neighbours because the data points can be represented in any space without any dimension restriction.

**16) What is a Receiver Operating Characteristic (ROC) curve? What does it convey?**

A ROC curve is a graph that plots True Positive Rate vs False Positive Rate. It displays the performance of a classification algorithm at all classification thresholds.

It highlights the trade-off between sensitivity and specificity where

Sensitivity = True Positive Rate and Specificity = 1 - False Positive Rate.

Curves that are pointed towards the left corner of the plot belong to good classifiers.

**17) List a few distances used for the K-means clustering algorithm.**

1. Euclidean Distance
2. Minkowski Distance
3. Hamming Distance
4. Manhattan Distance
5. Chebychev Distance

**18) Consider a dataset that contains features belonging to four classes. You are asked to use Logistic Regression as a classification algorithm to train your dataset. Do you think it is a good idea? If not, which algorithm would you suggest for the task?**

No, it is not advisable to use Logistic Regression for multi-class classification as even when the estimated coefficients for the Logistic Regression model are evaluated for well-separated classes, the model is unstable.

A simple alternative to this would be the Linear Discriminant Analysis algorithm which is more stable than the Logistic Regression for multi-class classification.

**19) What is the major difference between Quadratic Discriminant Analysis (QDA) and Linear Discriminant Analysis (LDA)?**

LDA presumes that features within each class form a part of a multivariate Gaussian distribution with a class-specific mean vector and a covariance matrix that is shared by all N classes. However, QDA assumes that each class has its own covariance matrix.

**20) Consider two classes A1 and A2 whose features have been generated using the Gaussian function. The correlation for feature variables in the first class is 0.5 and that for the second class is -0.5. Which of the following- KNN, Linear Regression, QDA, LDA; would you preferentially use for classifying the dataset?**

Quadratic Discriminant Analysis would be the perfect choice as it is best suited for cases where the decision boundaries between the classes are moderately non-linear.

**21) What are regularization techniques in machine learning?**

In Machine learning, when we are trying to fit a set of ‘f’ feature variables to a model, we shrink the coefficients of the model corresponding to a few of the f feature variables to zero. This shrinkage is known as regularization. It is a method to perform feature variable selection.

**22) What are the two best-known regularization techniques?**

The two most widely used regularization techniques are ridge regression and lasso regression.

**23) What is supervised learning?**

In supervised learning, we train our machine learning model using a dataset where the target values corresponding to each set of feature variables are known.

**24) Differentiate between Ridge and Lasso regression.**

Ridge Regression	Lasso Regression
In ridge regression, an extra term that is proportional to the square of the coefficients is added to the RSS (Residual sum-of-squares) to penalize the estimation of the coefficients of the linear regression model.	In Lasso regression, an extra term that is proportional to the coefficients is added to RSS (Residual sum-of-squares) to penalize estimation of the coefficients of the linear regression model.

It is also known as $l_2$ regularization.	It is also known as $l_1$ regularization.
It performs better for cases where the target to predict is a function of a large number of feature variables, all with coefficients of roughly the same size.	It is preferred for cases where a relatively number of feature variables have substantial coefficients, and the remaining features have coefficients that are small in value or zero value.

## 25) What is unsupervised learning?

In unsupervised learning, we train our machine learning model using a dataset where the target values corresponding to each set of feature variables are not known. We thus look for patterns in the feature variable space and cluster similar ones together.

## 26) Which algorithms can be used for both classification and regression problems?

Following machine learning algorithms can be used for both classification and regression:

Decision Trees, Random Forests, Neural Networks.

## 27) Why is recursive binary splitting in decision trees known as a *top-down greedy approach*?

The splitting technique is labelled top-down because it starts from the top of the tree (the point at which all the feature variables haven't been split) and then successively divides the target variable space, each division is indicated by two new branches further down on the tree.

The technique is also labelled as greedy. That is because, at every step of the tree-generating process, the best division (or split) is ensured at that specific step, instead of moving ahead and choosing a split that generates a better tree in some later step.

## 28) List a few supervised and unsupervised machine learning algorithms.

Unsupervised Machine Learning Algorithms	Supervised Machine Learning Algorithms
K-Nearest Neighbour	Naive Bayes

K-Means Clustering	Linear Regression
Hierarchical Clustering	Logistic Regression
	Neural Networks
	Decision Tree
	Random Forests

**29) Explain a few different types of classification algorithms.**

**30) What is the difference between probability and likelihood?**

Consider a random experiment whose all possible outcomes are finite. Let C denote the sample space of all possible outcomes. Then, the probability for an event A (which is a subset of C) is given by,

$$P(A) = \frac{\text{Number of elements in A}}{\text{Total number of elements in sample space, C}}$$

However, if a hypothesis H is given, then the probability of the event A is given by:

$$P(A|H) = \frac{P(A \cap H)}{P(H)}$$

Where A and H are both subsets of C. Thus, given anyone hypothesis, this defines the probability for any member of the set of possible results. It may be regarded as a function of both A and H, but is usually used as a function of A alone, for some specific H. On the other hand, the likelihood, L(H|A) of the hypothesis H given event of interest, A, is proportional to P(A|H), the constant of proportionality being arbitrary. The key here is that with probability, A is the variable and H is constant while with likelihood, H is the variable for constant A.

## **1) How will you find the middle element of a linked list in a single pass?**

Finding the middle element of a linked list in a single pass means we should traverse the complete linked list twice as we do in a two-pass case. To achieve this task, we can use two pointers: slw\_ptr (slow pointer) and fst\_ptr (fast pointer). If the slow pointer reads each element of the list and the fast pointer is made to run twice as fast as the slow pointer, then when the fast pointer is at the end of the linked list, the slow pointer will be at the middle element.

Steps:

1. Create two pointers slw\_ptr and fst\_ptr that point to the first element of the list.
2. Move the pointer fast\_ptr two steps and move the pointer slow\_ptr one step ahead until the fast\_ptr has reached the end of the string.
3. Return the value to which the slow\_ptr pointer is pointing to.

## **2) Write code to print the InOrder traversal of a tree.**

The following function will print the InOrder traversal of a tree in C++:

```
void printInorder(struct Node* node)

{
    if (node == NULL)
        return;

    printInorder(node->left);

    cout << node->data << " ";
    printInorder(node->right);

}
```

- 1) Given a particular machine learning model, what type of problems does it solve, what are the assumptions the model makes about the data and why it is best fit for a particular kind of problem?
- 2) Is the given machine learning model prone to overfitting? If so, what can you do about this?
- 3) What type of data does the machine learning model handle –categorical, numerical, etc.?

- 4) How interpretable is the given machine learning model?
- 5) What will you do if training results in very low accuracy?
- 6) Does the developed machine learning model have convergence problems?
- 7) Which is your favourite machine learning algorithm? Why it is your favourite and how will you explain about that machine learning algorithm to a layperson?
- 8) What approach will you follow to suggest followers on Twitter?
- 9) How will generate related searches on Bing?
- 10) Which tools and environments have you used to train and assess machine learning models?
- 11) If you claim in your resume that you know about recommender systems, then you might be asked to explain about PLSI and SVD models in detail.
- 12) How will you apply machine learning to images?

**1) How will you weigh 9 marbles 3 times on a balance scale to find the heaviest one?**

**Assume that all the marbles look the same and weigh the same except the one which is slightly heavier than each one of them.**

This can be achieved in the following way:

1. Divide the 9 marbles into groups of three.
2. The first time you'll use the balance scale will be to check which group contains the heaviest marble.  
If the two groups weigh the same, then the balance scale will stay balanced. This means that the group that was not placed on the scale has the heaviest element. However, if the balance scale is inclined toward either of the two chosen groups, then also the group with the heaviest element will be identified easily.
3. Now that you have a group of the three marbles that contains the heaviest element, you are left with the task of identifying the heaviest marble. For that, you can use the same logic as in the step above to determine which is the heaviest marble.

**2) Why is gradient checking important?**

In the neural network machine learning algorithm, we use the backpropagation algorithm to identify the correct weights for a given dataset. When the backpropagation algorithm is used with gradient descent, there are chances that the code will display that the loss function is decreasing with each iteration even though there is a bug in the code. Hence, it is important to implement gradient checking to be sure that the computer is evaluating the correct derivatives after each iteration.

### **3) What is loss function in a Neural Network?**

The loss function in the Neural Network machine learning algorithm is the function that the gradient descent algorithm uses to compute the gradient. It plays an important role in configuring a machine learning model.

### **4) Which one is better – random weight assignment or assigning the same weights to the units in the hidden layer?**

For hidden layers of a neural network, it is better to assign random weights to each unit of the layer than assigning the same weights to it. That is because if we use the same weights for each unit, then all the units will generate the same output and lower the entropy. Thus, we should always use random weights that are able to break the symmetry and to quickly reach the cost function minima.

### **5) How will you design a spam filter?**

A spam filter can be designed by training a neural network with emails that are spam and emails that are not spam. Of course, before feeding the network with emails, one must convert the textual data into numbers through text processing techniques.

6) Explain the difference between MLE and MAP inference.

7) Given a number, how will you find the closest number in a series of floating-point data?

### **8) What is boosting?**

Boosting is a technique that is commonly used to improve the performance of a decision tree machine learning algorithm. In Boosting, each tree is created using information from previously evaluated trees. Boosting involves learning the dataset slowly instead of creating a single large decision tree by rigorously fitting the dataset. In Boosting, we fit a decision tree using the current residuals in place of the target variable, Y. We then add this new decision tree into the fitted functions for updating the residuals.

## **1) What are the reasons for gradient descent to converge slow or not converge in various machine learning algorithms?**

A few of the reasons for the gradient descent algorithm showing slow convergence or no convergence at all are:

1. The cost function may not be a convex function.
2. The improper value is chosen for initializing the learning rate. If the learning rate is too high, the step oscillates and the global minimum is not reached. And, if the learning rate is too less, the gradient descent algorithm might take forever to reach the global minimum.

## **2) Given an objective function, calculate the range of its learning rate.**

A good way of calculating the range of an objective function's learning rate is by training a network beginning with a low learning rate and increasing the learning rate exponentially for every batch. One should then store the values for loss corresponding to each learning rate value and then plot it to visualize which range of learning rate corresponds to a fast decrease in the loss function.

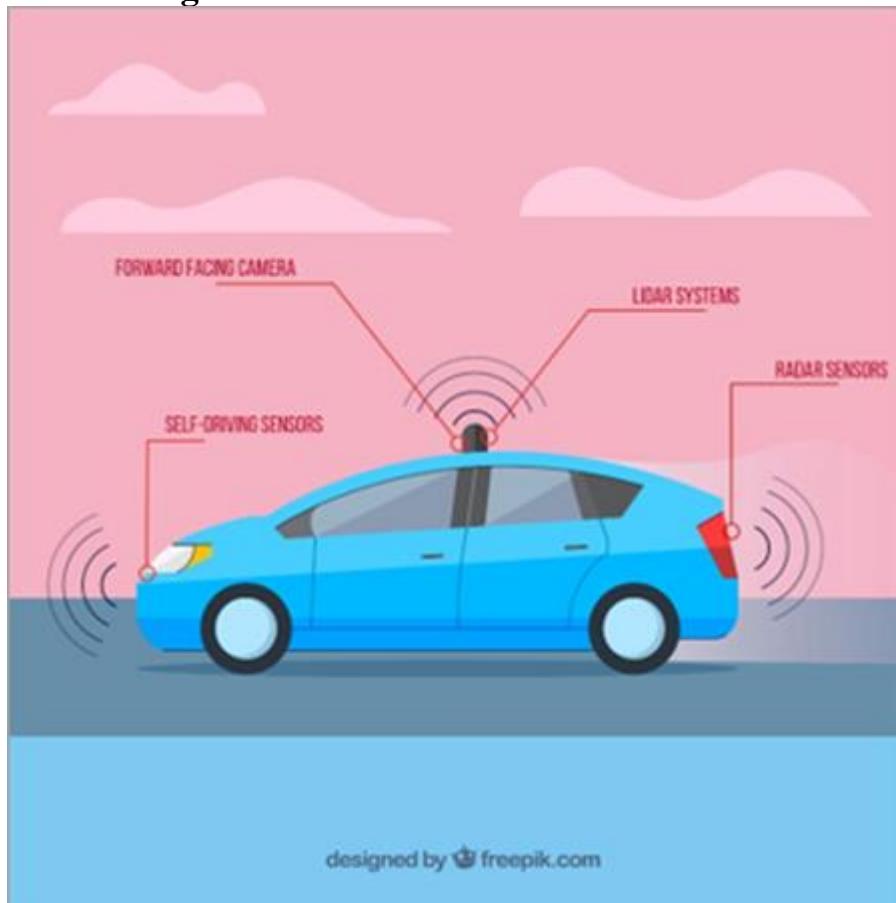
- 3) If the gradient descent does not converge, what could be the problem?
- 4) How will you check for a valid binary search tree?
- 1) Explain BFS (Breadth First Search algorithm)
- 2) How will you tell if a song in our catalogue is a duplicate or not?
- 3) What is your favourite machine learning algorithm and why?
- 4) Given the song list and metadata, disambiguate artists having same names.
- 5) How will you sample a stream of data to match the distribution with real data?

## **Q #1) What is machine learning?**

**Answer:** Machine Learning is a study in computer science which deals with making machines intelligent. A machine is called intelligent if it can make its own decisions.

The process of making machines learn is by providing a machine learning algorithm with training data. The output of this learning process is a trained ML model. This model artifact makes predictions on new data for which output is not known.

## Let us see a real-life example of ML: Self Driving Cars



A real-life example of machine learning is self-driving cars. With machine learning, self-driving cars exist. How does ML help Self Driven Cars?

So, the data of all the self-driving cars on the road is collected from the sensors and cameras attached to the cars which are been driven. Now, with machine learning algorithms and the collected data, the cars can learn themselves. Thus, by such training, they can perform tasks like humans.

### Q #2) What is machine learning system design?

**Answer:** It is a step-by-step process to define hardware and software requirements for machine learning model design. **The aim of machine learning design is:**

- **Adaptability:** The system should be flexible enough to adapt to new changes, such as new data or changes in business features.
- **Maintainability:** The performance of the system should not degrade with time. The system should have optimal performance with any data distribution changes that occur with time.
- **Scalability:** As the system grows, it should be able to accommodate the growth. Such changes are increases in complexity, data, or traffic.

- **Reliability:** The system should provide correct results or show errors (not show garbage output) for uncertain input data and environments.

### **Q #3) What are the steps involved in Machine Learning system design?**

**Answer:**



**A) Gather Requirements:** The system designer gathers the knowledge about designing the system, such as what size of datasets will be used? Does the system need to be more accurate or faster? What is the type of hardware requirements for the model? Would there be any need to retrain the model?

**B) Identify the Metrics:** Metrics are used to measure the outcome of the model. Functional metrics measure how beneficial the model will be like click-through rate, time spent watching the video, etc.

Some non-functional metrics could be scalability, flexibility, ease to train, etc. While the model is being developed, the dataset is broken into 3 sets- training, evaluation, and test. Some offline methods, such as Mean Squared Error, F1 score, Area under ROC Curve, are also employed to measure the outcome of the model.

### **C) Architecture:**

**When planning architecture:**

- Identify the target variable. **For example**, to design a system that recommends products to users, the target variable would be product.
- Finalize a few features of the variable. In our example, some features may be user age, user hobbies.
- Machine Learning operations such as storing data, data transformation, to be performed.
- Choose a baseline model. A model which does not need to be trained and acts as a baseline for other models.
- Start working on the model. This step deals with activities such as storing logs, using analytics tools that are performed in the production.

**D) Serve the model to the users.**

### **Q #4) What are the different types of Machine Learning Algorithms?**

**Answer:** These are classified as below:

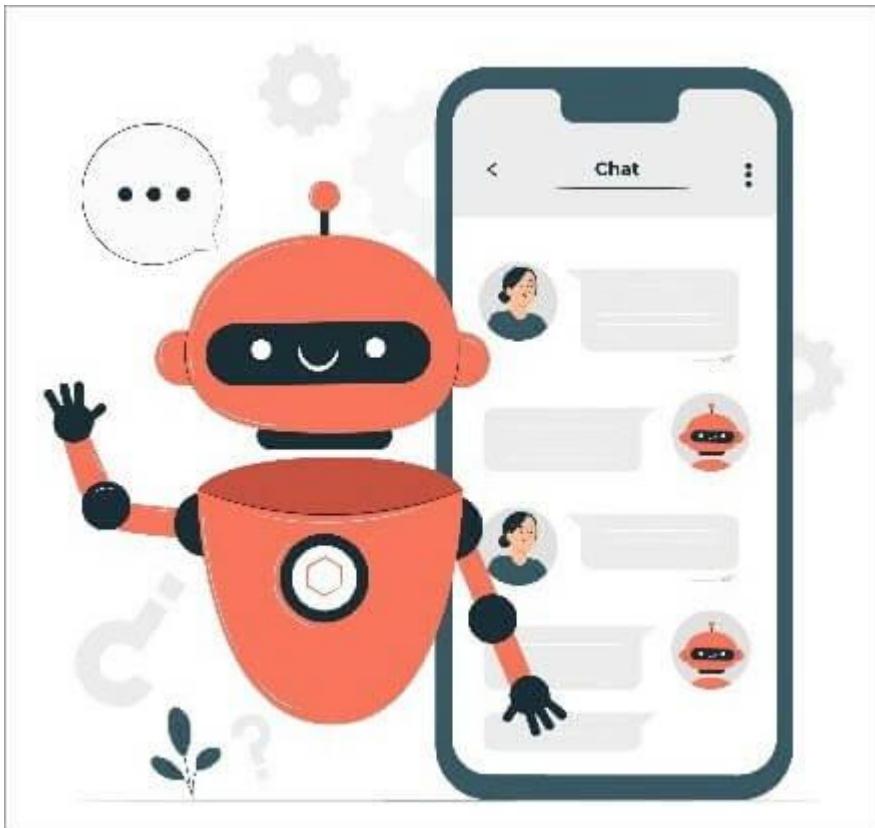
- **Supervised Learning Algorithm:** Supervised learning uses labeled data to predict outcomes. The learning happens in the presence of a supervisor, just like learning performed by a small child with the help of his teacher. By using labeled data, the machines can find out their accuracy and learn by themselves.

- **Unsupervised Learning Algorithms:** Unsupervised learning happens without the help of a supervisor. The machine learning algorithms were used to cluster the unlabelled data. These algorithms find out the hidden patterns in the data without any human help.
- **Reinforcement Learning:** The algorithm learns by the feedback mechanism and past experiences. This type of learning takes the feedback from the previous step and learns from experience to decide what the best next step would be. It is an iterative process, also called Markov Decision Process. In Reinforcement Learning, the more the number of feedbacks the more accurate the system would be.

### **Q #5) What are the applications of Machine Learning?**

**Answer:** Some of the most seen applications are listed as below:

#### **Chatbots:**



#### **Ecommerce:**



1. **Chatbot:** These days majority of the websites have a virtual customer service assistant which provides automated answers to your queries based on the information present on the website. With the help of machine learning algorithms, chatbots can train themselves with the inputs and provide better answers with time.
2. **Search Engine Results:** In any Web Search Engines, say Google, as we query, it provides some results. As we click on any of the results displayed and spend some time visiting the webpage, Google can find out whether or not the query results are appropriate? With the machine learning algorithms at the backend, the search engines can refine their results.
3. **Ecommerce Shopping:** Whenever user shops online, he/she is presented with product recommendations, some options such as “Customers also bought”, “Products Bought Together”, “Other similar Products” etc. These are nothing but the recommendations provided by machine learning algorithms running behind the website, which try to make the customer experience easy and friendly.
4. **Facial Recognition:** Nowadays the mobile phones, social media platforms such as Instagram, Facebook can automatically identify and suggest tagging the person in the uploaded pic. In such cases, these platforms have ML algorithms that extract the features of the picture and match them with the profile picture of people in your friend list.
5. **Personalized Virtual Assistants such as Siri, Alexa, Bixby:** These assistants run over voice and provide appropriate information. With such assistants, we can also create personalized tasks such as “Creating To-Do List”, “Listing Grocery Items”, “Setting Up Alarm”, “Play Music or Videos”. The machine learning algorithms here capture our previous inputs and refine

their output. Each time, the machine learns by itself to provide a personalized experience.

There are numerous other applications where machine learning is used, like Email Filtering, Security Systems, Fraud Detection, etc. From the above applications, we can see how it plays a vital role in our day-to-day lives.

## **Q #6) Is there a difference between Artificial Intelligence and Machine Learning?**

**Answer:** Artificial Intelligence and Machine Learning terms are used interchangeably always, but it is not so. There is a difference between both. Before going to the difference, let us understand what Artificial Intelligence is. Artificial Intelligence is the ability of a computer machine to show human-like intelligence and perform tasks like humans. A machine competent to think, learn on its own, and make its own decisions is nothing but an artificially intelligent machine.

## **Let us compare and differentiate them along with some real-life examples:**

### **Artificial Intelligence**

The art of making machines intelligent is AI

AI robots perform tasks to make the system successful rather than training and retraining

AI computers are programmed extensively

### **Machine Learning**

ML is a part of AI. It is a process of learning input data without any help of programming. The machines retrain themselves for accurate reduction of error.

ML mechanism does not involve programming; it learns from data

## **Q #7) Give an example to compare Artificial Intelligence and Machine Learning?**

**Answer:**

### **Example of Artificial Intelligence:**

The most seen example of AI is Tesla Car. All the cars are connected, so if one car learns about an unnoticed sharp turn, it is updated for all cars.

Another example is Drones, nowadays used by Tech Giant, Amazon for Logistics and Transportation. The drones use programming and technology, such as navigation systems, for automated flying. Sensors and cameras are attached to drones to capture data which is used by Machine Learning algorithms.

Some uses of AI-enabled drones are agriculture, smart cities, etc.

### **Example of Machine Learning: Drone**

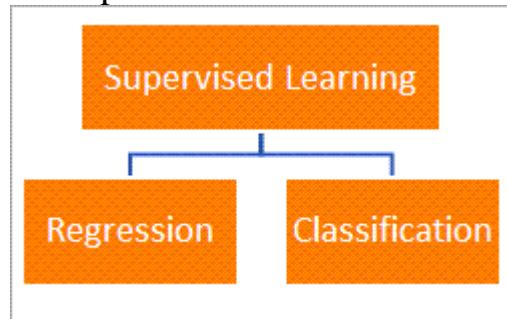
As we read above about Self-Flying Drones, the cameras and sensors attached to the drones capture images that are processed using Computer Vision. The computer vision marks objects for drones to recognize, which helps the drones to go in the right direction without colliding with obstacles.

The machine learning algorithms also learn from the captured images of the objects. The self-flying drones are also enabled by GPS navigation, due to which the destination coordinates are already fed in them. But the GPS system is not enough to avoid a collision, leading to droes crashing with the mountains or walls or trees.

Thus, there is a need to train drones. With the machine learning algorithms, the drones are fed with a large amount of data. The datasets train the drones to detect the objects and avoid such objects which may lead to a collision.

### **Q #8) What is the Classification and Regression in Machine Learning?**

**Answer:** Supervised Learning Methods are classified into Classification and Regression. Both these methods work with labeled data set and are used to make predictions.



**Classification Methods:** These methods categorize the input data into different output classes. In the Classification algorithm, the machine learns and gives the output in form of classes.

In other words, these classification methods provide an output function that maps the input data to an output class. The learned machine will categorize input data into generated output classes. As new data is fed to the machine, it will move to one of the output classes. The output classes are discrete, such as Yes/No, Long/Short.

As we know, the training sets (input data) for classification machine learning algorithms are labeled. By labeled data, we mean the input data is pre-categorized. Such as an image of fruit is labeled with fruit name or fruit description.

**Classification Methods are divided into binary classifiers and multi-class classifiers. Let us see each of them:**

- **Binary Classifier:** This type of classification has the outcome as only 2 classes.
- **Multi-Class Classifier:** In this type of classification, the outcome is more than 2 classes.

**Regression Methods:** Regression methods give the predicted output as a continuous variable like Cost, Price, Age, Salary, etc. In Regression, the machine learning algorithms predict output as continuous variables. The regression problems predict a mapping function based on the input and output variables.

**Q #9) What are the classification and regression methods?**

**Answer:**

**Classification algorithms are as below:**

1. Decision Tree Classification
2. K Nearest Neighbours
3. Naïve Bayes
4. Support Vector Machine
5. Random Forest
6. Stochastic Gradient Descent

**Some of the Regression Methods are:**

1. Linear Regression
2. Support Vector Regression
3. Regression Tree

**Q #10) Give an example of Classification and Regression in machine learning?**

**Answer:** Let us see a simple example to understand classification and regression.

Speed is a continuous variable, so if we must determine what is the speed of the car? – It is a Regression Problem

If the speed of the car is given, we can predict if the speed of the car moving at high speed or low speed? – It is a classification problem

**Q #11) How to build a Machine Learning Model?**

**Answer:** The ML model is built primarily using 3 steps:

1. Choose an algorithm for the model and train it.
2. Test the model by using test data.
3. Retrain the model if there are any changes and use the model for real-time projects

**Q #12) How to choose an appropriate algorithm to create a Machine Learning Model?**

**Answer:** To choose the most appropriate algorithm to train your machine, some steps to be followed are:

**a) Categorise the problem based on input and output:**

- **Based on Input:** If the data is labeled, we use supervised learning methods while for data that is not labeled unsupervised learning techniques are used. Reinforcement learning is used where

feedback from the previous step determines the next best step to follow. Each step takes the model to reach its goal.

- **Based on output:** If the output of the problem is continuous, such as a number, regression methods are used, while if the output is a class, classification techniques are applied.

#### **b) Prepare the data**

The data play an important role in determining the type of algorithm to be used. Some algorithms use small sets of data while other algorithms may need tons of data. The next step would be to analyze, process, and transform the data to use for modeling.

#### **c) Check out the available algorithms:**

To choose an appropriate algorithm based on the availability, focus on:

- Time is taken to build the model.
- The complexity of the algorithm.
- Accuracy of the model.
- Scalability of the model.
- How much time does it take to Predict the output?
- Is the model fulfilling the business requirements?

#### **d) Implement the ML algorithms:**

To choose the appropriate algorithm, run the available ML algorithms on different sets of data and evaluate their performance based on set criteria. Also, we can run a single algorithm on different datasets and find out the best algorithm.

### **Q #13) What are test data and training data?**

**Answer:** Training Data in Machine Learning is as important as a Machine Algorithm itself. As the name says, a training dataset is data to train the machine. The machine learns from the training data. The training data is labeled dataset. It means the output variable is mapped to one or more input variables. Test data is data used to check the accuracy of the machine. The machine output should have minimal error.

Now, how do we find out the training data and test data?

The training data and test data may be taken out from the same dataset. While training the machine, we may take out a portion of the data (training data) and pass through the model multiple times to reduce the error. After successful training, we feed the model with the remaining data (test data) to get the output.

If the predicted output variable is equal to the actual labeled output value, the model passes otherwise, we may need to retrain the machine or change the model.

#### **Q #14) What is deep learning? How is it different from Machine Learning?**

**Answer:** Deep learning is a part of the Machine learning process which uses Artificial Neural Networks (ANN) for making machines learn and have decision-making capabilities. The ANN corresponds to the neural system of the human brain, where all nerves are interconnected.

The neurons in the human brain correspond to the nodes in ANN. The Artificial Neural Network consists of many layers and intermediate layers between the input and output layers are called hidden layers. The Deep Learning Algorithms are like Machine Learning Algorithms except that the former contains many more layers (hidden layers) than the latter.

**Some differences between deep learning and machine learning are:**

##### **Deep Learning**

Deep Learning pass the data through multiple processing layers to predict the relation between input and output variables

The output data could be of any form such as shape, sound, or image

Deep Learning uses far more data than ML

The Deep Learning Algorithms does not need human intervention

##### **Q #15) What are the most popular algorithms used in machine learning?**

**Answer: The most common algorithms are:**

1. **K-Nearest Neighbour:** It is a supervised algorithm used for classification and regression problems. This algorithm assumes similar points are near to each other. It works by choosing an appropriate number of examples (k) as the query. By query, we mean the item in question. **For example,** songs recommended of 5 similar songs by the system. So, k here is 5.
2. **Decision Tree:** It is a supervised learning technique mostly used for classification problems. The decision tree is structured like a tree where the nodes represent the dataset, branches show rules on data and the leaf denotes the outcome.
3. **Neural Network Algorithms:** The artificial neural network learns by both supervised, unsupervised learning. An artificial neural network consists of multiple layers, namely input, output, and hidden layers. Two of the neural network training algorithms are Gradient Descent and Back-Propagation Algorithm.

4. **Support Vector Machine:** It is a supervised learning algorithm used for classification and regression problems. In this algorithm, we divide the data points with a hyperplane. The n-dimensional data points are divided into classes where new data points can be classified. Some applications of SVM are image categorization, facial recognition.

#### **Q #16) What do you mean by Genetic Programming?**

**Answer:** Genetic Programming is a form of artificial intelligence. It copies the process of natural selection to find out the optimal result.

This process is iterative in nature where at each step of the algorithm there might be randomly mutating offspring. Only the fittest offspring are chosen to cross and reproduce in the next generation. Thus, the fitness of the algorithm improves with generations. This algorithm terminates once it reaches a pre-defined fitness value.

#### **Q #17) What is Logistic Regression?**

**Answer:** Logistic Regression is an algorithm that comes under classification type. It predicts a binary outcome that is either 0 or 1 for given input variables. The output of Logistic Regression is 0/1. The threshold value is generally taken as 0.5. By threshold value, we mean any input below 0.5 has output 0, and any value more than threshold has output 1.

#### **Q #18) What is Lazy Learning?**

**Answer:** Lazy Learning is a machine learning method where the data is not generalized until the query is made to it. In other words, such learning defers the processing until the request for information is received. An example of a Lazy learning technique is KNN, where the data is just stored. It is processed only when the query is made to it.

#### **Q #19) What is a Perceptron? How does it work?**

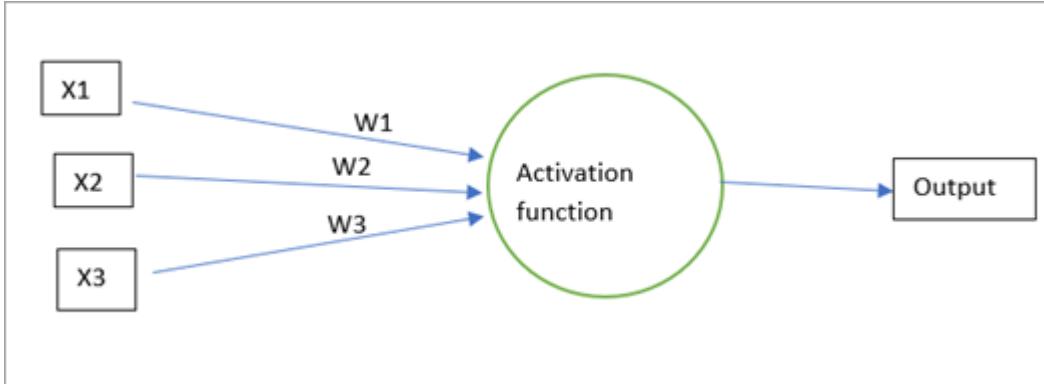
**Answer:** A Perceptron is the simplest ML algorithm for linear classification. A single-layer neural network is called a Perceptron. A perceptron model consists of the input layer, hidden layer, and output layer.

The input layer is connected to the hidden layer through weights and the weights are +1,0 or -1. The activation function for a single layer model is a binary step function.

The perceptron learning model is a binary classifier that classifies the inputs to output classes. The net input is fed to the activation function. If the output of the activation function is greater than the threshold value, it will return 1 otherwise, if the output is less than the threshold value, it will return 0.

**The output for the below model will be**

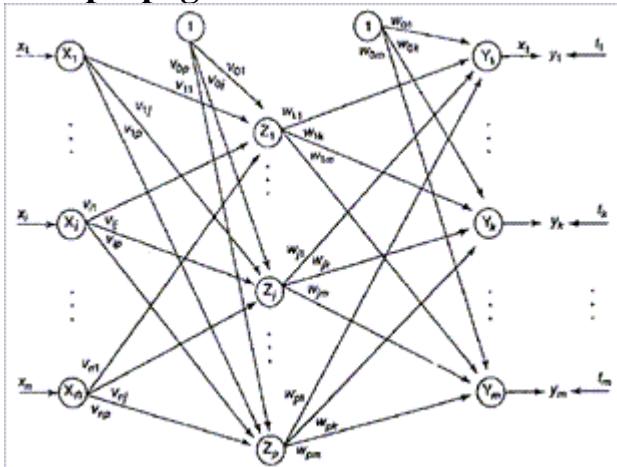
$$O = w_1 * x_1 + w_2 * x_2 + w_3 * x_3$$



### Q #20) What is Backpropagation Technique?

**Answer:** The backpropagation Method is an artificial neural network training method for machine learning. It is an iterative process for the reduction of error and makes the artificial neural network model more reliable and accurate. The error is calculated from the previous epoch output and input. The weights of the hidden and input layers are updated. Since the error travels back towards the hidden layer, that is why it is called backpropagation of error.

### Backpropagation Network:



Backpropagation Network is a multilayer perceptron network. It works in 2 phases: Feed Forward and Reverse Phase.

In the first phase, the network is fed with an input set of neurons, and the output is calculated. It is a supervised learning algorithm, therefore the target value is known. The output of the training model is compared with the target. The error is calculated and sent back for updating weight at the input and hidden layers.

1. Can you explain what a decision tree is?
2. How does a random forest work?
3. Can you explain the bias-variance tradeoff?
4. What is overfitting and how can it be avoided?

5. Can you explain how boosting algorithms work?
  6. Can you explain how a support vector machine (SVM) works?
  7. Can you explain how a neural network is trained?
  8. Can you explain how k-means clustering works?
  9. Can you explain how a recommendation system works?
  10. Can you explain how a Gaussian mixture model works?
11. Explain Linear and Logistic Regression. List their assumptions. Why cannot we use Linear Regression on categorical output?
12. Explain Bias-Variance Tradeoff. Explain underfitting and overfitting. What is the need for regularization?
13. Explain variants of Gradient Descent and the pros and cons of each variant.
14. Difference between Bagging and Boosting. Explain Random Forest.
15. Explain Precision and Recall measures and give examples of use cases where each is measured.
16. Briefly discuss some dimension reduction techniques. Difference between PCA and SVD.
17. Explain ROC Curve. What do the axes of the ROC Curve represent? Elaborate on the two extreme points of the ROC Curve – (0, 0) and (1, 1).
18. Explain AUC and its physical interpretation? Is it possible to get AUC below 0.5? What is the worst AUC that you can possibly achieve?
19. Explain the problem of vanishing and exploding gradients. Briefly describe some methods to solve these.
20. What is the need for a pooling layer in CNNs? Difference between max pooling and average pooling.
21. Explain how will you forecast a time series? Can we perform Linear Regression on time-series data?
22. What is Central Limit Theorem? Give an example where it is used.
23. Why is hyperparameter tuning required? Elaborate on some common hyperparameters for tree-based models.
24. Briefly discuss some clustering methods. What are the drawbacks of K-Means Clustering?
25. Differentiate between generative and discriminative models and give examples of each.
26. *Question:* What sort of optimization problem would you be solving to train a support vector machine?

*Answers:* maximize margin (best answer), quadratic program, quadratic with linear constraints, reference to solving the primal or dual form.

27. *Question:* Tell me about positives and negatives of using Gaussian processes / general kernel methods approach to learning.

*Answer:* Positives - non-linear, non-parametric. Negatives - bad scaling with instances, need to do hyper-parameter tuning

28. *Question:* How does a kernel method scale with the number of instances (e.g. with a Gaussian rbf kernel)?

*Answer:* Quadratic (referring to construction of the gram (kernel) matrix), cubic (referring to the matrix inversion)

29. *Question:* Describe ways to overcome scaling issues.

*Answers:* nystrom methods/low-rank kernel matrix approximations, random features, local by query/near neighbors

30. *Question:* What are some tools for parallelizing machine learning algorithms?

*Answers:* GPUs, Matlab parfor, write your own using low level primitives/RPC/MPI, mapreduce, spark, vowpal, graphlab, giraph, petuum, parameterserver

31. *Question:* In Python, do you have a favorite/least favorite PEP?

*Answer:* Peps are python enhancement proposal. If you have a favorite or least favorite, it means they have knowledge of Python.

1. What is bias and variance?
2. Difference between unsupervised and supervised learning?
3. Define accuracy and validation loss?
4. Define optimizers? Name some of them?
5. Difference between L1 and L2 regularization?
6. What does F1 score imply?
7. Difference between overfitting and underfitting?
8. Talk about a ML project you have recently worked on
9. How do you eliminate overfitting/underfitting?
10. Which mathematical operation is mainly used when image is passed through the DNN layers?
11. Explain one tactic to increase accuracy and one tactic to reduce the loss?

12. How do you reduce model size without affecting the accuracy much?

Explain one method to do so (Pruning/Deep compression/Design Space Exploration)?

13. What is model speed?

14. Name an activation function better than ReLU?

**15. Q1 Define Recall and Precision**

16. Recall or true positive rate refers to the total number of positive claimed by a model as compared to the actual number of positives which are present throughout the data

17. Precision or positive predictive value precisely predicts the number of true positives claimed by a model compared to the number of positives it actually claims.

**18. Q2) Why is Bayes referred to as “Naive Bayes”?**

19. Naive Bayes is referred to as “naive” because despite having many practical applications it is based on the assumption that it is impossible to find real-life data. All the features in a data set are independent, equal, and crucial. In the naive Bayes approach, conditional probability is computed as a pure product of the probabilities of individual components thus implying the complete independence of features. Sadly, this assumption can never occur in a real-life situation.

20. Q3) What are the two methods for calibration in supervised learning?

21. The two calibration methods are isotonic regression and Platt calibration. Both these methods are designed especially for binary classification.

**22. Q4) What is F1 Score?**

23. F1 score is a measure of a model’s performance that is an average of the recall of a model and precision which results nearing to 1 being the best and those nearing 0 being the worst. The F1 score can be used in classification tests that do not give priority to true negatives.

**24. Q5) What is Ensemble learning?**

25. It uses a combination of learning algorithms to optimize the predictive performance of models. In this, the multiple models like classifiers or experts are both strategically generated and combined to prevent Overfitting in models. It is used to enhance the classification, prediction, performance, function approximation etc. of a model.

**26. Q6) Difference between machine learning and deep learning?**

27. Machine learning involves the application and usage of advanced algorithms to uncover the hidden patterns within the data, parse data, and learn from it, and finally apply those learned insights to make informed business decisions. Meanwhile, Deep learning is a subset of Machine Learning that involves the use of Artificial Neural Nets which draw inspiration from the neural net structure of the human brain. Also, deep learning is used in feature detection.

**28. Q7) What is the difference between Type 1 and Type 2 errors?**

29. Type 1 error is a false positive error that ‘claims’ that an incident has occurred when actually nothing has happened. An example of a false positive error is a false fire alarm. The alarm starts ringing even though there’s no fire. Type 2 error is a false negative error that ‘claims’ nothing has occurred when something has surely happened. Type 2 error would be to tell a pregnant lady that she isn’t carrying a baby.

**30. Q8) Explain the variance and bias of the term.**

31. The error of a learning algorithm during the process of training is decomposed into two parts – bias and variance. Variance denotes an error caused because of the complexity of that learning algorithm in data analysis while bias is an error situation caused due to the use of simple assumptions in the learning algorithm. Variance is measured by how much the learning algorithm’s prediction varies for different training data sets and bias measures the proximity of the average classifier which is created by the learning algorithm to the target function.

**32. Q9) Why do you prune a Decision Tree?**

33. They are pruned to get rid of the branches with weak predictive abilities. Pruning can be either done from the top-down or bottom-up. It helps to minimize the complexity quotient of the Decision Tree model and optimize its predictive accuracy. Cost-complexity pruning, minimum error pruning, reduced error pruning, error complexity pruning are some of the used Decision Tree pruning methods.

**34. Q10) How would you handle corrupted or missing data in a dataset?**

35. You must drop the rows and columns or replace them with other values. Pandas library has two methods to find missing or corrupted data - dropna() and isnull(). These functions help find the rows/columns of data with corrupted data and drop those values.

1. What is the Central Limit Theorem and why is it important?

2. What is the difference between type I vs type II error?
3. What is linear regression?
4. What do the terms p-value, coefficient, and r-squared value mean? What is the significance of each of these components?
5. What is selection bias?
6. What is the Binomial Probability Formula?
7. What do you understand by the term Normal Distribution?
8. What is correlation and covariance in statistics?
9. What is the goal of A/B Testing?
  1. What is your understanding of Data Science?
  2. List the differences between supervised and unsupervised learning?
  3. What is the bias-variance trade-off?
  4. How is KNN different from k-means clustering?
  5. What is a confusion matrix?
  6. Explain how a ROC curve works.
  7. What is Bayes' Theorem? How is it useful in a machine learning context?
  8. What is Naive Bayes's theorem?
  9. What are the differences between over-fitting and under-fitting?
10. What is Cluster Sampling?
  1. What is Machine Learning?
  2. What are the various classification algorithms?
  3. What is 'Naive' in a Naive Bayes?
  4. Explain SVM algorithm in detail.
  5. What are the different kernels in SVM?
  6. What is Decision Tree?
  7. What are Entropy and Information gain in the Decision tree algorithm?
  8. What is logistic regression? State an example when you have used logistic regression recently

9. What Are the Drawbacks of the Linear Model?
10. What is the difference between Regression and classification of ML techniques?
11. What are Recommender Systems?
12. How can outlier values be treated?
13. What is a Random Forest? How does it work? What is ‘Naive’ in a Naive Bayes?
14. What is Cross-Validation?
15. How Do You Handle Missing or Corrupted Data in a Dataset?
16. Explain the Confusion Matrix with Respect to Machine Learning Algorithms.

An image processing company may ask you “how can you find all the images which are a photo of a landscape?”. A video processing company may ask you “In a video of a soccer match, how can you mark all the times that a certain player is in the view?”. A speech processing company may ask you “Among a large number of voicemails, how can you detect the ones that an old woman is talking?”. An NLP company might ask you “How would you provide suggestions for the next word in an incomplete sentence?”. An online shopping company may ask you “how would you store products in fixed-size bins where people are most likely to buy them together?”.

There are so many use cases and many of these types of questions which may be asked. I suggest that before the interview you read about a few case studies which relates to the company’s business. For example if it’s an image processing company, read about popular approaches for image detection, identification, segmentation, tracking, clustering, etc.

Also, there are a few things which you are expected to know and are shared among almost all use cases. You should have ideas how to approach scenarios like this:

- How to use labeled and unlabeled data?
- 17. What if you don’t have any labeled data?
- 18. What if your data set is skewed (e.g. 99.99 % positive and 0.01 % negative labels)?
- 19. How to test and know whether or not we have overfitting problem?
- 20. How to avoid overfitting?
- 21. How to make training faster?
- 22. How to make predictions faster?
- 23. What Does mAP mean?
- 24. Describe ROC curve for measuring accuracy.
- 25. What are forward, backward and step sampling.

- 26.what is the difference between image processing and computer vision?
- 27.How can we classify in the case of existing more outlier in the data?
- 28.How dose training iteration related to over-fitting?
- 29.Describe/differentiate between the terms: machine learning, artificial intelligence, and deep learning
- 30.How are bias and variance related?
- 31.How are Type I and Type II errors different?
- 32.Can you describe what “overfitting” is?
- 33.Describe your favorite machine learning algorithm
- 34.What’s the difference between supervised learning and unsupervised learning?
- 35.How are generative and discriminative models the same? How are they different?
- 36.How do you prune a decision tree?
- 37.How would you evaluate the effectiveness of your machine learning model?
- 38.Have you ever worked with a missing or corrupted dataset? How did you handle it?
- 39.What is a hash table?
- 40.How do you prefer to visualize your results? What tools do you use?
- 41.Name three machine learning algorithms
- 42.What data types does JSON support?
- 43.In SQL, how are primary and foreign keys related?

## 1. What is machine learning?

In answering this question, try to show you understand of the broad applications of machine learning, as well as how it fits into AI. Put it into your own words, but convey your understanding that machine learning is a form of AI that automates data analysis to enable computers to learn and adapt through experience to do specific tasks without explicit programming

## 2. What is candidate sampling in machine learning?

A training-time optimization in which a probability is calculated for all the positive labels, using, for example, softmax, but only for a random sample of negative labels. For example, if we have an example labeled beagle and dog

candidate sampling computes the predicted probabilities and corresponding loss terms for the beagle and dog class outputs in addition to a random subset of the remaining classes (cat, lollipop, fence).

### 3. Mention the difference between Data Mining and Machine learning?

Machine learning relates to the study, design, and development of the algorithms that give computers the capability to learn without being explicitly programmed. While data mining can be defined as the process by which the unstructured data tries to extract knowledge or unknown interesting patterns. During this processing machine, learning algorithms are used.

### 4. What is A/B testing in Machine Learning?

A statistical way of comparing two (or more) techniques, typically an incumbent against a new rival. A/B testing aims to determine not only which technique performs better but also to understand whether the difference is statistically significant. A/B testing usually considers only two techniques using one measurement, but it can be applied to any finite number of techniques and measures.

### 5. Explain How We Can Capture The Correlation Between Continuous And Categorical Variable?

Yes, it is possible by using ANCOVA technique. It stands for Analysis of Covariance.

It is used to calculate the association between continuous and categorical variables.

### 6. How does deductive and inductive machine learning differ?

Deductive machine learning starts with a conclusion, then learns by deducing what is right or wrong about that conclusion. Inductive machine learning starts with examples from which to draw conclusions.

### 7. What is inductive machine learning?

The inductive machine learning involves the process of learning by examples, where a system, from a set of observed instances, tries to induce a general rule.

### 8.What Is The Difference Between An Array And Linked List?

An array is an ordered fashion of collection of objects. A linked list is a series of objects that are processed in a sequential order.

### 9.Explain The Concept Of Machine Learning And Assume That You Are Explaining This To A 5-year-old Baby?

Yes, Machine learning is exactly the same way how babies do their day to day activities, the way they walk or sleep etc. It is a common fact that babies cannot walk straight away and they fall and then they get up again and then try. This is the same thing when it comes to machine learning, it is all about how the algorithm is working and at the same time redefining every time to make sure the end result is as perfect as possible.

#### 10. What is a sigmoid function in Machine learning?

A function that maps logistic or multinomial regression output (log odds) to probabilities, returning a value between 0 and 1

#### 11.What is bucketing in machine learning?

Converting a (usually continuous) feature into multiple binary features called buckets or bins, typically based on value range. For example, instead of representing temperature as a single continuous floating-point feature, you could chop ranges of temperatures into discrete bins. Given temperature data sensitive to a tenth of a degree, all temperatures between 0.0 and 15.0 degrees could be put into one bin, 15.1 to 30.0 degrees could be the second bin, and 30.1 to 50.0 degrees could be a third bin.

#### 12. What are some methods of reducing dimensionality?

You can reduce dimensionality by combining features with feature engineering, removing collinear features, or using algorithmic dimensionality reduction.

#### 13.What Is The Difference Between Machine Learning And Data Mining?

Data mining is about working on unstructured data and then extract it to a level where the interesting and unknown patterns are identified.

Machine learning is a process or a study whether it closely relates to design, development of the algorithms that provide an ability to the machines to capacity to learn.

#### 14.What is collaborative filtering in machine learning?

Making predictions about the interests of one user based on the interests of many other users. Collaborative filtering is often used in recommendation systems.

#### 15.What's your favorite algorithm, and can you explain it to me in less than a minute?

This type of question tests your understanding of how to communicate complex and technical nuances with poise and the ability to summarize quickly and

efficiently. Make sure you have a choice and make sure you can explain different algorithms so simply and effectively that a five-year-old could grasp the basics!

## 16. How is ML different from artificial intelligence?

AI involves machines that execute tasks which are programmed and based on human intelligence, whereas ML is a subset application of AI where machines are made to learn information. They gradually perform tasks and can automatically build models from the learnings.

## 17. Differentiate between statistics and ML.

In statistics, the relationships between relevant data (variables) are established; but in ML, the algorithms rely on data regardless of their statistical influence. In other words, statistics are concerned about inferences in the data whereas ML looks at optimization.

## 18. Mention key business metrics that help ML?

Identify the key services/products/functions that hold good for ML. For example, if you consider a commercial bank, metrics such as a number of new accounts, type of accounts, leads generated and so on, can be evaluated through ML methods.

## 19. What are the five popular algorithms of Machine Learning?

1. Decision Trees
2. Neural Networks (back propagation)
3. Probabilistic networks
4. Nearest Neighbor
5. Support vector machines

## 20. What are the different Algorithm techniques in Machine Learning?

The different types of techniques in Machine Learning are

1. Supervised Learning
2. Unsupervised Learning
3. Semi-supervised Learning
4. Reinforcement Learning
5. Transduction
6. Learning to Learn

## 21. What is algorithm independent machine learning?

Machine learning in where mathematical foundations are independent of any particular classifier or learning algorithm is referred to as algorithm independent machine learning?

22. What is the difference between artificial learning and machine learning?

Designing and developing algorithms according to the behaviors based on empirical data are known as Machine Learning. While artificial intelligence in addition to machine learning, it also covers other aspects like knowledge representation, natural language processing, planning, robotics etc.

23.What are the three stages to build the hypotheses or model in machine learning?

1. Model building
2. Model testing
3. Applying the model

24. What is the standard approach to supervised learning?

The standard approach to supervised learning is to split the set of example into the training set and the test.

25. What is ‘Training set’ and ‘Test set’?

In various areas of information science like machine learning, a set of data is used to discover the potentially predictive relationship known as ‘Training Set’. The training set is an example given to the learner, while Test set is used to test the accuracy of the hypotheses generated by the learner, and it is the set of example held back from the learner. The training set is distinct from the Test set.

26. List down various approaches to machine learning?

The different approaches in Machine Learning are

- i) Concept Vs Classification Learning
- ii) Symbolic Vs Statistical Learning
- iii) Inductive Vs Analytical Learning

26. What is not Machine Learning?

- i) Artificial Intelligence
- ii) Rule-based inference

27. What are the advantages of Naive Bayes?

In a Naïve Bayes classifier will converge quicker than discriminative models like logistic regression, so you need less training data. The main advantage is that it can't learn interactions between features.

28. In what areas Pattern Recognition is used?

1. Pattern Recognition can be used in
2. Computer Vision
3. Speech Recognition
4. Data Mining
5. Statistics
6. Informal Retrieval

#### 1. What is ‘Overfitting’ in Machine learning?

In machine learning, when a statistical model describes random error or noise instead of the underlying relationship ‘overfitting’ occurs. When a model is excessively complex, overfitting is normally observed, because of having too many parameters with respect to the number of training data types. The model exhibits poor performance which has been overfitted.

#### 2. Why does overfitting happen?

The possibility of overfitting exists as the criteria used for training the model is not the same as the criteria used to judge the efficacy of a model.

#### 3. How can you avoid overfitting?

By using a lot of data overfitting can be avoided, overfitting happens relatively as you have a small dataset, and you try to learn from it. But if you have a small database and you are forced to come with a model based on that. In such a situation, you can use a technique known as cross-validation. In this method the dataset splits into two section, testing and training datasets, the testing dataset will only test the model while, in the training dataset, the data points will come up with the model.

In this technique, a model is usually given a dataset of a known data on which training (training data set) is run and a dataset of unknown data against which the model is tested. The idea of cross-validation is to define a dataset to “test” the model in the training phase.

#### 4. How do bias and variance play out in machine learning?

Both bias and variance are errors. Bias is an error due to flawed assumptions in the learning algorithm. Variance is an error resulting from too much complexity in the learning algorithm.

#### 5. What is the calibration layer in machine learning?

A post-prediction adjustment, typically to account for prediction bias. The adjusted predictions and probabilities should match the distribution of an observed set of labels.

## 6.Explain How We Can Capture The Correlation Between Continuous And Categorical Variable?

Yes, it is possible by using ANCOVA technique. It stands for Analysis of Covariance. It is used to calculate the association between continuous and categorical variables.

## 7.What is the bias in machine learning?

An interceptor offset from an origin. Bias (also known as the bias term) is referred to as b or w<sub>0</sub> in machine learning models.

## 8.What Is The Difference Between Bias And Variance?

Bias: Bias can be defined as a situation where an error has occurred due to the use of assumptions in the learning algorithm.

Variance: Variance is an error caused because of the complexity of the algorithm that is been used to analyze the data.

## 9. What is the confusion matrix in machine learning?

An NxN table that summarizes how successful a classification model's predictions were; that is, the correlation between the label and the model's classification. One axis of a confusion matrix is the label that the model predicted, and the other axis is the actual label. N represents the number of classes.

## 10. What is the convex function in machine learning?

A function in which the region above the graph of the function is a convex set. The prototypical convex function is shaped something like the letter U.

A strictly convex function has exactly one local minimum point, which is also the global minimum point. The classic U-shaped functions are strictly convex functions. However, some convex functions (for example, straight lines) are not.

## 11. Explain the difference between L1 and L2 regularization.

L2 regularization tends to spread error among all the terms, while L1 is more binary/sparse, with many variables either being assigned a 1 or 0 in weighting. L1 corresponds to setting a Laplace a prior on the terms, while L2 corresponds to a Gaussian prior.

## 1. Explain Principal Component Analysis (PCA).

PCA is a dimensionality-reduction technique which mathematically transforms a set of correlated variables into a smaller set of uncorrelated variables called principal components.

2. What value do you optimize when using a support vector machine (SVM)?

For a linear function, SVM optimizes the product of input vectors as well as the coefficients. In other words, the algorithm with the linear function can be restructured into a dot-product.

3. Explain Principal Component Analysis (PCA).

PCA is a dimensionality-reduction technique which mathematically transforms a set of correlated variables into a smaller set of uncorrelated variables called principal components.

4. What value do you optimize when using a support vector machine (SVM)?

For a linear function, SVM optimizes the product of input vectors as well as the coefficients. In other words, the algorithm with the linear function can be restructured into a dot-product.

5. What is kernel SVM?

Kernel SVM is the abbreviated version of kernel support vector machine. Kernel methods are a class of algorithms for pattern analysis and the most common one is the kernel SVM.

6. What is the bias-variance decomposition of classification error in the ensemble method?

The expected error of a learning algorithm can be decomposed into bias and variance. A bias term measures how closely the average classifier produced by the learning algorithm matches the target function. The variance term measures how much the learning algorithm's prediction fluctuates for different training sets.

7. What is an Incremental Learning algorithm in the ensemble?

The incremental learning method is the ability of an algorithm to learn from new data that may be available after the classifier has already been generated from the already available dataset.

8. What are PCA, KPCA, and ICA used for?

PCA (Principal Components Analysis), KPCA (Kernel-based Principal Component Analysis) and ICA (Independent Component Analysis) are important feature extraction techniques used for dimensionality reduction.

9. What is dimension reduction in Machine Learning?

In Machine Learning and statistics, dimension reduction is the process of reducing the number of random variables under considerations and can be divided into feature selection and feature extraction

#### 10. What are support vector machines?

Support vector machines are supervised learning algorithms used for classification and regression analysis.

#### 11. What is ensemble learning?

To solve a particular computational program, multiple models such as classifiers or experts are strategically generated and combined. This process is known as ensemble learning.

#### 12. Why ensemble learning is used?

Ensemble learning is used to improve the classification, prediction, function approximation etc of a model.

#### 13 When to use ensemble learning?

Ensemble learning is used when you build component classifiers that are more accurate and independent from each other.

#### 14. What are the two paradigms of ensemble methods?

The two paradigms of ensemble methods are

- a) Sequential ensemble methods
- b) Parallel ensemble methods

#### 1. How do you choose an algorithm for a classification problem?

The answer depends on the degree of accuracy needed and the size of the training set. If you have a small training set, you can use a low variance/high bias classifier. If your training set is large, you will want to choose a high variance/low bias classifier

#### 2. What is a class in machine learning?

One of a set of enumerated target values for a label. For example, in a binary classification model that detects spam, the two classes are spam and not spam. In a multi-class classification model that identifies dog breeds, the classes would be a poodle, beagle, pug, and so on.

#### 3. What is the baseline for machine learning?

A simple model or heuristic used as reference point for comparing how well a model is performing. A baseline helps model developers quantify the minimal, expected performance on a particular problem.

#### 4.What is a checkpoint in machine learning?

Data that captures the state of the variables of a model at a particular time. Checkpoints enable exporting model weights, as well as performing training across multiple sessions. Checkpoints also enable training to continue past errors (for example, job preemption). Note that the graph itself is not included in a checkpoint.

#### 5. How To Handle Or Missing Data In A Dataset?

An individual can easily find missing or corrupted data in a data set either by dropping the rows or columns. On contrary, they can decide to replace the data with another value.

In Pandas there are two ways to identify the missing data, these two methods are very useful.

isnull() and dropna().

#### 6. How do classification and regression differ?

Classification predicts group or class membership. Regression involves predicting a response. Classification is the better technique when you need a more definite answer.

#### 7. What is supervised versus unsupervised learning?

Supervised learning is a process of machine learning in which outputs are fed back into a computer for the software to learn from for more accurate results the next time. With supervised learning, the “machine” receives initial training to start. In contrast, unsupervised learning means a computer will learn without initial training.

#### 8.Plese, State Few Popular Machine Learning Algorithms?

- Nearest Neighbour
- Neural Networks
- Decision Trees
- Support vector machines

#### 9. What is a binary classification in machine learning?

A type of classification task that outputs one of two mutually exclusive classes. For example, a machine learning model that evaluates email messages and outputs either “spam” or “not spam” is a binary classifier.

10.What is the difference between supervised and unsupervised machine learning?

Supervised learning requires training labeled data. For example, in order to do classification (a supervised learning task), you’ll need to first label the data you’ll use to train the model to classify data into your labeled groups. Unsupervised learning, in contrast, does not require labeling data explicitly.

11 What is batch in machine learning?

The set of examples used in one iteration (that is, one gradient update) of model training.

12. What is batch size machine learning?

The number of examples in a batch. For example, the batch size of SGD is 1, while the batch size of a mini-batch is usually between 10 and 1000. Batch size is usually fixed during training and inference.

13.What is class-imbalanced dataset in machine learning?

A binary classification problem in which the labels for the two classes have significantly different frequencies. For example, a disease data set in which 0.0001 of examples have positive labels and 0.9999 have negative labels is a class-imbalanced problem, but a football game predictor in which 0.51 of examples label one team winning and 0.49 label the other team winning is not a class- imbalanced problem.

14. What is the classification model in machine learning?

A type of machine learning model for distinguishing between two or more discrete classes. For example, a natural language processing classification model could determine whether an input sentence was in French, Spanish, or Italian. Compare with the regression model.

15.What is the classification threshold in machine learning?

A scalar-value criterion that is applied to a model’s predicted score in order to separate the positive class from the negative class. Used when mapping logistic regression results to binary classification.

16.What is class-imbalanced dataset in machine learning?

A binary classification problem in which the labels for the two classes have significantly different frequencies. For example, a disease data set in which 0.0001 of examples have positive labels and 0.9999 have negative labels is a class-imbalanced problem, but a football game predictor in which 0.51 of examples label one team winning and 0.49 label the other team winning is not a class- imbalanced problem.

17.What Are The Three Stages To Build The Model In Machine Learning?

- (a). Model building
- (b). Model testing
- (c). Applying the model

18.What is convergence in machine learning?

Informally, often refers to a state reached during training in which training loss and validation loss change very little or not at all with each iteration after a certain number of iterations.

In other words, a model reaches convergence when additional training on the current data will not improve the model. In deep learning, loss values sometimes stay constant or nearly so for many iterations before finally descending, temporarily producing a false sense of convergence.

19.Explain what is the function of ‘Unsupervised Learning’?

1. Find clusters of the data
2. Find low-dimensional representations of the data
3. Find interesting directions in data
4. Interesting coordinates and correlations
5. Find novel observations/ database cleaning

20. Explain what is the function of ‘Supervised Learning’?

1. Classifications
2. Speech recognition
3. Regression
4. Predict time series
5. Annotate strings

Linear regression

1. What is the use of gradient descent?

The use of gradient descent plainly lies with the fact that it is easy to implement and is compatible with most of the ML algorithms when it comes to optimization. This technique works on the principle of a cost function.

## 2. What is backpropagation in machine learning?

The primary algorithm for performing gradient descent on neural networks. First, the output values of each node are calculated (and cached) in a forward pass. Then, the partial derivative of the error with respect to each parameter is calculated in a backward pass through the graph.

The Area Under the ROC curve is the probability that a classifier will be more confident that a randomly chosen positive example is actually positive than that a randomly chosen negative example is positive.

## 3. What is a sigmoid function in Machine learning?

A function that maps logistic or multinomial regression output (log odds) to probabilities, returning a value between 0 and 1.

## 4. What is an AdaGrad algorithm in machine learning?

A sophisticated gradient descent algorithm that rescales the gradients of each parameter, effectively giving each parameter an independent learning rate.

## 5. What is batch statistical learning?

Statistical learning techniques allow learning a function or predictor from a set of observed data that can make predictions about unseen or future data. These techniques provide guarantees on the performance of the learned predictor on the future unseen data based on a statistical assumption on the data generating process.

## 6. What is PAC Learning?

PAC (Probably Approximately Correct) learning is a learning framework that has been introduced to analyze learning algorithms and their statistical efficiency.

## 7. What are the different categories you can categorize the sequence learning process?

- a) Sequence prediction
- b) Sequence generation
- c) Sequence recognition
- d) Sequential decision

## 8. What are the components of relational evaluation techniques?

The important components of relational evaluation techniques are

- a) Data Acquisition
- b) Ground Truth Acquisition
- c) Cross-Validation Technique

- d) Query Type
- e) Scoring Metric
- f) Significance Test

Deep learning neural network and decision tree

### 1. What Is Deep Learning?

Deep learning is a process where it is considered to be a subset of the machine learning process.

### 2. Define What Is Fourier Transform In A Single Sentence?

A process of decomposing generic functions into a superposition of symmetric functions is considered to be a Fourier Transform.

### 3.What is Rectified Linear Unit (ReLU) in Machine learning?

An activation function with the following rules:

- (a). If the input is negative or zero, the output is 0.
- (b). If the input is positive, the output is equal to input.

### 4. What is the activation function in Machine Learning?

A function (for example, ReLU or sigmoid) that takes in the weighted sum of all of the inputs from the previous layer and then generates and passes an output value (typically nonlinear) to the next layer.

### 5. What are neural networks and where do they find their application in ML? Elaborate.

Neural networks are information processing models that derive their functions based on biological neurons found in the human brain. The reason they are the choice of technique in ML is that they help discover patterns in data that are sometimes too complex to comprehend by humans.

### 6. Differentiate between a parameter and a hyperparameter?

Parameters are attributes in training data that can be estimated during ML. Hyperparameters are attributes that cannot be determined beforehand in the training data. Example: Learning rate in neural networks.

### 7. What is ‘tuning’ in ML?

Generally, the goal of ML is to automatically provide accurate output from the vast amounts of input data without human intervention. Tuning is a process

which makes this possible and it involves optimizing hyperparameters for an algorithm or an ML model to make them perform correctly.

#### 8. What is optimization in ML?

Optimisation, in general, refers to minimizing or maximizing an objective function (in linear programming). In the context of ML, optimization refers to the tuning of hyperparameters which result in minimizing the error function (or loss function).

#### 9.What is dimensionality reduction? Explain in detail.

The process of reducing variables in an ML classification scenario is called Dimensionality reduction. The process is segregated into sub-processes called feature extraction and feature selection. Dimensionality reduction is done to enhance visualization of training data. It finds the appropriate set of variables known as principal variables.

#### 10.On what basis do you choose a classifier?

Classifiers must be chosen based on the accuracy it provides on the trained data. Also, the size of the dataset sometimes affects accuracy. For example, Naive Bayes classifiers suit smaller datasets in terms of accuracy due to higher asymptotic errors.

#### 11. What is the decision tree classification?

A decision tree builds classification (or regression) models as a tree structure, with datasets broken up into ever smaller subsets while developing the decision tree, literally in a tree-like way with branches and nodes. Decision trees can handle both categorical and numerical data.

#### 12. What is a recommendation system?

Anyone who has used Spotify or shopped at Amazon will recognize a recommendation system: It's an information filtering system that predicts what a user might want to hear or see based on choice patterns provided by the user.

#### 13. What is the classifier in machine learning?

A classifier in a Machine Learning is a system that inputs a vector of discrete or continuous feature values and outputs a single discrete value, the class.

#### 14. What are the advantages of Naive Bayes?

In a Naïve Bayes classifier will converge quicker than discriminative models like logistic regression, so you need less training data. The main advantage is that it can't learn interactions between features.

15. In what areas Pattern Recognition is used?

1. Pattern Recognition can be used in
2. Computer Vision
3. Speech Recognition
4. Data Mining
5. Statistics
6. Informal Retrieval

16.What are the different methods of Sequential Supervised Learning?

The different methods to solve Sequential Supervised Learning problems are

- a) Sliding-window methods
- b) Recurrent sliding windows
- c) Hidden Markow models
- d) Maximum entropy Markow models
- e) Conditional random fields
- f) Graph transformer networks

17.What are the areas in robotics and information processing where the sequential prediction problem arises?

The areas in robotics and information processing where the sequential prediction problem arises are

- a) Imitation Learning
- b) Structured prediction
- c) Model-based reinforcement learning

### **1. How to use k-NN for classification and regression?**

A) For Classification apply majority vote of neighbors and for Regression, we do mean or median of all k neighbors.

**2)Why do we use the word ‘Regression’ in Logistic Regression even though we use it for Classification?**

A) We use the Logistic Regression for classification after the model predicting the continuous output between (0–1) which we can interpret as the probability of the point belonging to the class.

- such as the sigmoid( $W \cdot X_q$ ) > 0.5 we label it as positive else negative.

### 3) Explain intuition behind Boosting

A) Train the very base model  $h(0)$  with training data as the model that we fit this training data is set to be having a high bias, it makes more number of training errors.

- Then store the errors
- Train the next model on the errors got in the previous model
- If we keep on doing this.....each time we get residual errors and we try to predict them in the next model then our **final model**= $F_i(x) = a_0 h_0(x) + a_1 h_1(x) + \dots + a_i h_i(x)$

### 4) What does it mean by the precision of a model equal to zero is it possible to have precision equal to 0.

A) Precision represents out of all predicted positives of how many are actually positive.

$$\text{precision} = (\text{True positives}) / (\text{True positives} + \text{False positives})$$

- precision equal to 0 if every predicted point by the model is a false positive.

## **5 )Why we need Calibration?**

- Calibration is a must if probabilistic class-label is needed as output
- If the metric is log-loss and that needs the  $P(Y_i|X_i)$  values, then calibration is a must.
- The probabilities output by the models such as LR, naive Bayes are often NOT well calibrated which can be observed by plotting the calibration plot. Hence, we use calibration as a post-processing step to ensure that the final class-probabilities are well-calibrated.

## **6)Where is parameter sharing seen in deep learning?**

A) Parameter sharing is the sharing of weights by all neurons in a particular feature map. CNN uses the same weight vector to perform the convolution operation and RNN has the same weights at every time stamps.

## **7)How many parameters do we have in LSTM?**

A)  $4(mn+m^2+m)$  .For derivation checkout following blog.

Summing up DL -1: LSTM Parameters  
why  $4(nm+n^2+n)$ ?  
[medium.com](https://medium.com)

## **8)What is box cox transform? When can it be used?**

- Box-Cox transform helps us convert non-Gaussian distributed variables into Gaussian distributed variables.

- It is a good idea to perform it if your model expects features that are Gaussian distributed(Gaussian Naive Bayes).

**9)How to use the K-S test to find two random variables X1 and X2 follow the same distribution?**

- Plot CDF for both random variables
- Assume Null hypothesis: the two random variables come from the same distribution;
- Take Test statistic  $D = \text{supremum} (\text{CDF}(X1) — \text{CDF}(X2))$  throughout the CDF range
- Null hypothesis is rejected when  $D > c(\alpha) * \sqrt{\frac{(n+m)}{nm}}$
- where m and n are no of observations in CDF1 and CDF2 respectively
- .

**10) Explain the backpropagation mechanism in dropout layers?**

- While training Neural Network with dropout the output is calculated without considering those weights for chosen neurons that are selected to be dropped and they have the same value as they had in previous iterations. And that weight doesn't update while back-propagation.
- Note that weights will not become zero they just ignored for iteration.

**11)Find the output shape and parameters after following operations?**

$(7,7,512) \Rightarrow \text{flatern} \Rightarrow \text{Dense}(512)$

$(7,7,512) \Rightarrow Conv(512, (7,7))$

- **for (7,7,512)  $\Rightarrow$  flatern  $\Rightarrow$  Dense(512)** Trainable parameters =  $(7*7*512)*512=12845056$ , output shape = 512
- **For (7,7,512)  $\Rightarrow$  Conv (512,(7,7))** Trainable parameters =  $(7*7*512)*512=12845056$ , output shape = 1,1,512

**12)How will you calculate the  $P(x/y=0)$  in the case of x is continuous random variable?**

A) If x is a numerical feature then assume that the feature follows Normal distribution. Then we can obtain likelihood probabilities from (PDF) density function whereas the absolute likelihood value for any continuous variable is zero.

**13)Explain the Correlation and Covariance?**

A) Covariance shows the linear relationship between variables we cant interpret how strong they are related whereas with Correlation it gives the linear relationship strength and direction of the two variables.

**14)What is the problem with sigmoid during backpropagation?**

A) The derivative of the Sigmoid function lies between 0 and 0.25. During chain rule multiplication of the gradients, it will tend to zero which results in vanishing gradients problems and that impact on weight update.

## **15) Difference between micro average F1 and macro average F1 for a multiclass class classification?**

- $F1 \text{ Score} = 2 * \text{precision} * \text{Recall} / (\text{precision} + \text{recall})$
- For 3 classes Classification for each class, there will be respective True positives, False positives, True Negative, False Negative

### **a) Micro Avg F1**

- **Microaverage of precision**  
 $\text{precision} = \text{TP}_1 + \text{TP}_2 + \text{TP}_3 / (\text{TP}_1 + \text{TP}_2 + \text{TP}_3 + \text{FP}_1 + \text{FP}_2 + \text{FP}_3)$
- **Microaverage of Recall**  
 $\text{Recall} = \text{TP}_1 + \text{TP}_2 + \text{TP}_3 / (\text{TP}_1 + \text{TP}_2 + \text{TP}_3 + \text{FN}_1 + \text{FN}_2 + \text{FN}_3)$
- **Micro Avg F1** $= 2 * \text{precision} * \text{Recall} / (\text{precision} + \text{recall})$

### **b) Macro-average Method**

- **Macroaverage of precision** $= P_1 + P_2 + P_3 / 3$
- **Macroaverage of Recall** $= R_1 + R_2 + R_3 / 3$
- Where  $P_1 = \text{TP}_1 / (\text{TP}_1 + \text{FP}_1)$ ,  $R_1 = \text{TP}_1 / (\text{TP}_1 + \text{FN}_1)$  Same for  $P_2, P_3$
- **Macro Avg F1** $= 2 * \text{precision} * \text{Recall} / (\text{precision} + \text{recall})$

## **16) Why Image augmentation help in Image classification tasks?**

A) Image data augmentation used to create or expand data by artificially generating new images from changing input images, such as translation, scaling,

mirror, steering, Zoom, etc. Such that we can make our model be robust to input image change.

## **1. Differentiate between supervised learning and unsupervised learning**

These are some notable differences between the two:

Supervise Learning	Unsupervised Learning
Trained on labeled dataset	Trained on unlabeled dataset
Algorithms used: regression and classification	Algorithms used: clustering, association and density estimation
Suited for predictions	Suited for analysis
Maps input to the known output labels	Finds hidden patterns and discovers the output

## **2. Define logistic regression with example**

Also known as the Logit model, it's used for predicting a binary outcome from predictor variables having a linear combination. For instance, predicting a politician's victory or defeat in an election is binary. The predictor variables would be time spent in the camp and total money used for the camp.

## **3. How do classification machine learning techniques and regression differ?**

These are the key differences

Classification	Regression
Target variables can have discrete values	Target variables can have continuous values, usually real numbers
Evaluated by measuring accuracy	Evaluated by measuring root mean square error

## **4. What is meant by collaborative filtering?**

These are the steps taken in an analytics project:-

1. Comprehending the business problems.
2. Transforming the variables, outlier detection for Data preparation for modeling, checking missing values.
3. Analyzing the outcome, using tweaked approaches after running the model, this is done for achieving a good outcome.

4. Validation of the model via a few data sets. Further, implementing the model and analyzing its performance over a specific duration.

## **6. Explain in brief a few types of ensemble learning**

There are several types of ensemble learning, below are some of the more common types.

### **Boosting**

An iterative technique that helps in weight adjustment of a particular observation based on previous classification. In case, the classification is incorrect, then observation weight is increased. This helps in building reliable predictive models, as it reduces the bias error, but there's also a possibility of overfitting into the training data.

### **Bagging**

It attempts to implement learners on a particular sample bunch, further taking a mean of the productions. One can implement other learners on varying bunches in generalized bagging, this prevents some of the variance errors.

## **7. Describe box-cox transformation**

In a **regression analysis**, the dependent variable might not be able to satisfy ordinary least square regression assumptions. The residuals could be following the distribution (skewed) or curve, in case the prediction increases. In such scenarios, the transformation of response variable becomes a necessity in order for data to meet specific assumptions.

The **box-cox transformation** relates to Statistical techniques for transforming dependent, non-normal variables to a conventional shape. When the available data is unconventional, then many statistical techniques assume normality.

Numerous tests can be run when box-cox transformation is applied, it's a method for transforming unconventional, dependent variables into a more conventional shape.

## **8. What's Random Forest, and how does it function? Also, explain it's working.**

It's a versatile method for machine learning that can do classification and regression both. It gets used in outlier values, dimensionality reduction, treating

missing values. It's a kind of ensemble learning method, wherein clusters of weak models integrate to build a powerful model.

Numerous decision trees are created instead of a single tree in a random forest. For classifying new attribute-based objects, every tree provides a classification, and the one that has maximum votes (total trees in the forest) gets selected by the forest, as for regression average output of varying trees gets considered.

## **Working of Random Forest**

This technique's main principle is that various weak learners combine to make a strong learner. The steps include:-

### **9. If you were to train a model using 10 GB of data set and had only 4 GB RAM, then how would you approach this problem?**

To start, it's best to ask about the type of ML model that requires training.

#### **For SVM (partial fit will suit best)**

Follow these steps

1. Start by division of a large data set into smaller size sets.
2. Implement SVM's partial fit method, it will need the full data set's subset.
3. Repeat the second step for different subsets.

#### **For neural networks (NumPy array plus batch size will do)**

Follow these measures

In NumPy array, load the full data, NumPy array has a tendency to make mapping of the full data set. It doesn't load into the memory, the full data set.

For attaining required data, pass index into the NumPy array.

Make use of this data for passing to neural networks. Maintain a smaller batch size.

### **10. In an analysis, how do missing values get treated?**

Once the variables having missing values get identified, the extent of the values that are absent also gets discovered. In case any patterns are picked, it becomes necessary for the analyst to pay attention as these could bring about a couple of significant and valuable business-related insights.

And, if no patterns are discovered, then median or mean values can take place of the missing values, or it can be ignored. A default value can be allotted as maximum, minimum, or mean value. In case, the variable is categorical, the default value is assigned to the missing value.

If data distribution is incoming, then a mean value is assigned for normal distribution. Also, if a variable's 80% values seem missing, then it's reasonable to drop the variable than treat the missing values.

## **11. How to treat outlier values?**

For detection of outlier values, some graphical analysis or univariate method can be used. If the outliers are large, then either the 1st percentile value or the 99th percentile can replace the values. Also if the outliers are fewer, then the individual assessment can be done.

It should be noted that all outlier values are not necessarily extreme values. For treating outlier values, the values can either be modified and brought within range or they can be discarded.

## **12. Which cross-validation technique can be used on a time-series dataset?**

Rather than Implementing the K-Fold technique, one should know that time-series have an inherent chronological order, and is not some randomly distributed data. As far as time-series data is concerned, one can implement the forward-chaining technique, where one has to model previous data, then consider data that is forward-facing.

fold 1: training[1], test[2]

fold 1: training[1 2], test[3]

fold 1: training[1 2 3], test[4]

fold 1: training[1 2 3 4], test[5]

## **13. How often an algorithm requires updating?**

If these requirements call, then it's suitable for an algorithm to be updated:-

1. Model evolution is a must as data runs through infrastructure.
2. Data source (underlying) is not constant.
3. A non-stationary case shows up.
4. Results don't have good precision and accuracy as the algorithm doesn't perform well.

## **14. List some drawbacks of linear model**

These are a few drawbacks of the linear model

1. For binary and count outcomes, it is not usable.
2. Error linearity assumptions occur too often.
3. Over-fitting problems that cannot be solved.

## **15. Describe SVM algorithm**

SVM (Support Vector Machine) is an algorithm (supervised machine learning) implemented for classification and regression. If one's training data set has n features, then SVM does their plotting in a space that is n-dimensional, where every feature's value is a specific coordinate's value. SVM implements hyperplanes for the segregation of distinct classes.

When are deep learning algorithms more appropriate compared to traditional machine learning algorithms?

- Deep learning algorithms are capable of learning arbitrarily complex non-linear functions by using a deep enough and a wide enough network with the appropriate non-linear activation function.
- Traditional ML algorithms often require feature engineering of finding the subset of meaningful features to use. Deep learning algorithms often avoid the need for the feature engineering step.
- Deep Learning algorithms do well when there is a lot of data to work with.

How do you design a system that reads a natural language question and retrieves the closest FAQ answer?

There are multiple approaches for FAQ based question answering

1. Keyword based search (Information retrieval approach): Tag each question with keywords. Extract keywords from query and retrieve all relevant questions answers. Easy to scale with appropriate indexes reverse indexing.

2. Lexical matching approach : word level overlap between query and question. These approaches might be harder to scale to do real time matching based on the scale of the question-answer dataset.
3. Embedding of the query and of each FAQ question and pick the closest matching FAQ question based on the embedding distance.
  1. Could use common technique such as word2vec/glove and average word level embeddings to get sentence embedding
  2. Can find phrasal, document level embeddings.
4. Intent based retrieval : Understand the intent of the question and attributes of the intent – works well if there are a specific set of intents and the problem is to classify the query into one of the appropriate intents. Tag questions with appropriate intents and attributes to retrieve the appropriate answer.

How can you increase the recall of a search query (on search engine or e-commerce site) result without changing the underlying algorithm ?

Since we are not allowed to change the underlying algorithm, we can only play with the search query itself. Here are some ways we can modify the search query to get better recall:

- We want to modify the query in a way that we get results relevant to the original query. If the query is “dark pants”, results would still be relevant if it contained “black pants” as black is dark. This means we need to find results for a synonymous query too. “black pants”, “black trousers”, “dark trousers” are synonymous to “dark pants”. We don’t need to change the algorithm. So one way of increasing the recall is to also search for synonymous query by replacing words with their synonyms.
- You could apply the same principle as above to the result of the original query. Instead of changing the query, you get first set of results from original query, then get results which are synonymous to first set of results.

What is negative sampling when training the skip-gram model ?

*Skip-Gram Recap:* model tries to represent each word in a large text as a lower dimensional vector in a space of K dimensions making similar words also be

close to each other. This is achieved by training a feed-forward network where we try to *predict the context words given a specific words*.

*Why is it slow:* In this architecture, a soft-max is used to predict each context word. In practice, soft-max function is very slow in computation, specially for large vocabulary size.

*Resolution :*

- The objective function is reconstructed to treat the problem as classification problem where pairs of words : a given word and a corresponding context word are positive examples and a given word with non-context words are negative examples.
- While there can be a limited number of positive examples, there are many negative examples. Hence a randomly sampled set of negative examples are taken for each word when crafting the objective function.

This algorithm/model is called Skip Gram Negative Sampling(SGNS)

How will you build an auto suggestion feature for a messaging app or google search?

- Auto Suggestion feature involves recommending the next word in a sentence or a phrase. This is possible if we have built a language model on large enough “relevant” data.
- There are 2 caveats here –
  1. large corpus because we need to cover almost every case. This is important for recall.
  2. relevant data is useful for higher precision. As language model learnt on movie reviews may not be useful for an application like gmail which might have formal mails too, assuming movie reviews will be mostly written in natural and informal language.
- The data could be from google search queries or a user’s own chat. The language model could be built using probabilistic language modeling or neural language modeling.

What are the different ways of preventing over-fitting in a deep neural network ? Explain the intuition behind each

1. L2 norm regularization : Make the weights closer to zero prevent overfitting.
2. L1 Norm regularization : Make the weights closer to zero and also induce sparsity in weights. Less common form of regularization
3. Dropout regularization : Ensure some of the hidden units are dropped out at random to ensure the network does not overfit by becoming too reliant on a neuron by letting it overfit
4. Early stopping : Stop the training before weights are adjusted to overfit to the training data.

### **1) How would you explain machine learning to a kid?**

This question is to test if you can explain complex things simply and clearly for non-technical people. Prepare an explanation like this before the interview, with some examples within a context familiar to your interviewer.

### **2) What is the difference between a Type I and Type II error?**

Type I error is a false positive (if there's an alert, and there's no incident), and Type II error is a false negative (no alert, but there was an incident).

### **3) What's the difference between an array and a linked list?**

The crucial difference between an array and a linked list is that an **array is an ordered collection of objects**. The size of an array is specified at the time of declaration and can't be changed afterward. The **linked list** is a series of objects with pointers. New elements can be stored anywhere, and a reference is created for each new element using pointers.

### **4) How do you prevent overfitting?**

Detecting overfitting is useful, but the most important is to ensure you're not overfitting the model. Here are a few of the most popular solutions:

- Collect more data to train the model with more varied samples.
- Use cross-validation techniques
- Keep the model simple to reduce variance
- Use regularization techniques

### **5) What's the difference between Entropy and Information Gain?**

**Entropy** is the average rate at which information is produced by a stochastic source of data. It's an indicator of how dirty your data is. It decreases as you reach closer to the leaf node.

The **information gain** is the amount of information gained about a random variable or signal from observing another random variable. It's based on the decrease in entropy after a dataset is split on an attribute. It keeps on increasing as you get closer to the leaf node.

For a more detailed explanation, you could check this link.

## 6) What's an imbalanced dataset? Can you list some ways to deal with it?

Any dataset with an unequal class distribution is technically imbalanced.

Here are some techniques to handle imbalanced data:

- **Resample the training set:** There are two approaches to make a balanced dataset out of an imbalanced one are **under-sampling** and **over-sampling**.
- **Generate synthetic samples:** Using SMOTE (Synthetic Minority Oversampling Technique) to generate new and synthetic data to train the model.

## Read also

How to Deal With Imbalanced Classification and Regression Data

## 7) Why does XGBoost perform better than SVM?

XGBoost is an ensemble method that uses many trees, so it improves by repeating itself.

SVM is a linear separator. When data is not linearly separable, SVM needs a Kernel to project the data into a high-dimensional space. SVM can find a linear separation for almost any data.

## 8) What evaluation approaches would you use to gauge the effectiveness of an ML model?

- Split the dataset into training and test sets
- Use a cross-validation technique to segment the dataset
- Implement performance metrics like accuracy and the F1 score

## 9) What are dropouts?

Dropout is a straightforward implementation to halt neural network overfitting by terminating some of its units. Repeating this for every training example gives us different models for each one, improves processing, and reduces time.

## 10) What is GPT-3 (or other bleeding-edge technology)? How do you think we can use it?

This question tests if you're following new technology hype and research. GPT-3, as you probably know, is the newest (at least at the time of writing this article) language generation model that can generate human-like text. There are many perspectives on GPT-3. It can improve chatbots, automate customer service, and boost search engines with NLP.

What is data wrangling? Mention three points to consider in the process.

Data wrangling is a process by which we convert and map data. This changes data from its raw form to a format that is a lot more valuable.

Data wrangling is the first step for machine learning and deep learning. The end goal is to provide data that is actionable and to provide it as fast as possible.

There are three major things to focus on while talking about data wrangling –

### 1. Acquiring data

The first and probably the most important step in data science is the acquiring, sorting and cleaning of data. This is an extremely tedious process and requires the most amount of time.

One needs to:

- Check if the data is valid and up-to-date
- Check if the data acquired is relevant for the problem at hand

Sources for data collection Data is publicly available on various websites like kaggle.com, data.gov, World Bank, Five Thirty Eight Datasets, AWS Datasets, Google Datasets.

### 2. Data cleaning

Data cleaning is an essential component of data wrangling and requires a lot of patience. To make the job easier it is first essential to format the data make the data readable for humans at first.

The essentials involved are:

- Format the data to make it more readable

- Find outliers (data points that do not match the rest of the dataset) in data
- Find missing values and remove them from the data set (without this, any model being trained becomes incomplete and useless)

### 3. Data Computation

At times, your machine not have enough resources to run your algorithm e.g. you might not have a GPU. In these cases, you can use publicly available APIs to run your algorithm. These are standard end points found on the web which allow you to use computing power over the web and process data without having to rely on your own system. An example would be the Google Colab Platform.

Why is normalisation required before applying any machine learning model?  
What module can you use to perform normalisation?

Normalisation is a process that is required when an algorithm uses something like distance measures. Examples would be clustering data, finding cosine similarities, creating recommender systems.

Normalisation is not always required and is done to prevent variables that are on higher scale from affecting outcomes that are on lower levels. For example, consider a dataset of employees' income. This data won't be on the same scale if you try to cluster it. Hence, we would have to normalise the data to prevent incorrect clustering.

A key point to note is that normalisation does not distort the differences in the range of values.

A problem we might face if we don't normalise data is that gradients would take a very long time to descend and reach the global maxima/ minima.

For numerical data, normalisation is generally done between the range of 0 to 1.

The general formula is:

$$X_{\text{new}} = (x - \text{xmin}) / (\text{xmax} - \text{xmin})$$

Performing Normalisation in Python

In python, this can be easily done with the scikit-learn module (this can be installed through a pip command or installed from anaconda).

**From sklearn import preprocessing**

```
X = data #your data
```

```
Normalized_x_value = preprocessing.normalize(x)
```

What is a sobel filter? How would you implement it in Python?

The sobel filter performs a two-dimensional spatial gradient measurement on a given image which then emphasizes regions which have high spatial frequency. In effect, this means finding edges.

In most cases, sobel filters are used to find the approximate absolute gradient magnitude for every point in a grayscale image. The operator consists of a pair of  $3 \times 3$  convolution kernels. One of these kernels is rotated by 90 degrees.

These kernels respond to edges that run horizontal or vertical with respect to the pixel grid, one kernel for each orientation. A point to note is that these kernels can be applied either separately or can be combined together to find the absolute magnitude of the gradient at every point.

The sobel operator has a large convolution kernel which ends up smoothing the image to a greater extent and thus the operator becomes less sensitive to noise. It also produces higher output values for similar edges compared to other methods.

To overcome the problem of output values from the operator overflowing the maximum allowed pixel value per image type, avoid using image types that support pixel values.

### Implementation in Python

To implement it in Python, we can use the OpenCV module (can be installed from pip):

```
import cv2
```

```
import numpy as np
```

```
img = cv2.imread('your image.jpg',0)
```

```
laplacian = cv2.Laplacian(img,cv2.CV_64F)
```

```
sobelx = cv2.Sobel(img,cv2.CV_64F,1,0,ksize=5)
```

```
sobely = cv2.Sobel(img,cv2.CV_64F,0,1,ksize=5)
```

What is the curse of dimensionality?

The curse of dimensionality states that if the number of features is very large relative to the number of observations in a certain data set, many algorithms will fail to be able to train an effective model. This is extremely relevant to many of the commonly used algorithms, especially those that rely on distance measures.

What is the difference between feature selection and feature extraction?

Feature selection and feature extraction are two major ways of fixing the curse of dimensionality

### **1. Feature selection:**

Feature selection is used to filter a subset of input variables on which the attention should focus. Every other variable is ignored. This is something which we, as humans, tend to do subconsciously.

Many domains have tens of thousands of variables out of which most are irrelevant and redundant. Feature selection limits the training data and reduces the amount of computational resources used. It can significantly improve a learning algorithms performance.

In summary, we can say that the goal of feature selection is to find out an optimal feature subset. This might not be entirely accurate, however, methods of understanding the importance of features also exist. Some modules in python such as xgboost help achieve the same.

### **2. Feature extraction**

Feature extraction involves transformation of features so that we can extract features to improve the process of feature selection. For example, in an unsupervised learning problem, the extraction of bigrams from a text, or the extraction of contours from an image are examples of feature extraction.

The general workflow involves applying feature extraction on given data to extract features and then apply feature selection with respect to the target variable to select a subset of data. In effect, this helps improve the accuracy of a model.

What is Support Vector Regression?

Support Vector Machines (SVM) are used for regression. SVMs not only maintain the features of an algorithm but they can also be modified slightly to perform regression. Since the output is a real number, it can become difficult to predict information due to infinite number of possibilities for regions. Hence, in

the case of regression, a margin tolerance needs to be set and is an approximation to SVMs.

The main idea behind Support Vector Regression stays the same as SVMs i.e. to minimise error and individualise the hyperplane. It also maximises the margin. Form of error tolerance also plays a key role here.

In simple regression, we try to minimise the error rate. On the other hand, in support vector regression, we try to fit the error within a certain threshold.

What is sklearn? Why and when would you need to use it?

sklearn or scikit-learn is a python library which allows using a huge range of supervised and unsupervised learning algorithms.

The prerequisites for installing sklearn are:

- Scipy which is a library for scientific computing in python
- Matplotlib which allows plotting of 2d or 3d figures
- Numpy which allows for n dimensional array calculations
- Ipython which is an interactive console for python
- Pandas which is used for data analysis and working with data
- Sympy which is used for symbolic mathematics

Most of the features of scikit-learn fall under the following categories:

1. Supervised Learning – All of the common types of supervised learning models can be implemented in scikit-learn. Examples include svms, decision trees etc.
2. Feature extraction methods which help to define attributes in text and images.
3. Clustering techniques such as k-medians.
4. Dimensionality reduction techniques which help to reduce attributes from high dimensional spaces.
5. Feature selection which helps in choosing attributes and in turn, reduces computation.
6. Manifold learning which helps to summarise and depict complex multidimensional models.
7. Scikit-learn also provides a model for ensemble learning methods which are combinations of multiple supervised models.
8. Parameter tuning is also provided which allows for maximum efficiency of any algorithm.

How would you perform thresholding and canny detection using opencv?

Thresholding

Thresholding is one of the simplest methods which can be used for image segmentation. Its input is a grayscale image which can then be used to create binary images.

The most common use case of thresholding would be to extract a specific colour from an image. Mathematically, thresholding replaces each pixel value in an image with black if it is less than a particular constant i.e. the required color or replaces it with white if it is greater than the constant.

#### OpenCV Implementation

```
import cv2 as cv
import numpy as np
from matplotlib import pyplot as plt
```

```
# Then, load the image
img = cv.imread('image.jpeg', 0)
```

```
# Finally, apply the threshold method
ret, thresh1 = cv.threshold(img, 127, 255, cv.THRESH_BINARY)
```

#Note - THRESH\_BINARY is a variable and can be replaced by BINARY, BINARY\_INV, TRUNC, TOZERO, TOZERO\_INV. Each of these is a different type of threshold.

#### Edge Detection

Edge detection is an important principle in image processing. Canny edge detection is one of the most popular algorithms for edge detection.

The algorithm works as follows:

1. Reduce noise in the image
2. Find the intensity gradient
3. Perform non-maximum suppression
4. Perform thresholding.

#### OpenCV Implementation

```
import cv2 as cv
```

```
import numpy as np
```

```
from matplotlib import pyplot as plt
```

```
img = cv.imread('image.jpeg', 0)
```

```
edges = cv2.Canny(img, 100, 200)
```

## When would you use ARIMA?

ARIMA is a widely used statistical method which stands for Auto Regressive Integrated Moving Average. It is generally used for analysing time series data and time series forecasting. Let's take a quick look at the terms involved.

Auto Regression is a model that uses the relationship between the observation and some numbers of lagging observations.

Integrated means use of differences in raw observations which help make the time series stationary.

Moving Averages is a model that uses the relationship and dependency between the observation and residual error from the models being applied to the lagging observations.

Note that each of these components are used as parameters. After the construction of the model, a linear regression model is constructed.

Data is prepared by:

- Finding out the differences
- Removing trends and structures that will negatively affect the model
- Finally, making the model stationary.

Why is polarity and subjectivity an issue?

Polarity and subjectivity are terms which are generally used in sentiment analysis.

Polarity is the variation of emotions in a sentence. Since sentiment analysis is widely dependent on emotions and their intensity, polarity turns out to be an extremely important factor.

In most cases, opinions and sentiment analysis are evaluations. They fall under the categories of emotional and rational evaluations.

Rational evaluations, as the name suggests, are based on facts and rationality while emotional evaluations are based on non-tangible responses, which are not always easy to detect.

Subjectivity in sentiment analysis, is a matter of personal feelings and beliefs which may or may not be based on any fact. When there is a lot of subjectivity in a text, it must be explained and analysed in context. On the contrary, if there was a lot of polarity in the text, it could be expressed as a positive, negative or neutral emotion.

Where is the confusion matrix used? Which module would you use to show it?

In machine learning, confusion matrix is one of the easiest ways to summarise the performance of your algorithm.

At times, it is difficult to judge the accuracy of a model by just looking at the accuracy because of problems like unequal distribution. So, a better way to check how good your model is, is to use a confusion matrix.

First, let's look at some key terms.

Classification accuracy – This is the ratio of the number of correct predictions to the number of predictions made

True positives – Correct predictions of true events

False positives – Incorrect predictions of true events

True negatives – Correct predictions of false events

False negatives – Incorrect predictions of false events.

The confusion matrix is now simply a matrix containing true positives, false positives, true negatives, false negatives.

Example,

Let's take the Iris Flower dataset as an example.

The confusion matrix using a Linear Support Vector Classifier with parameter(C) as 0.01 is as follows:

Confusion matrix, without normalization

$\begin{bmatrix} 13 & 0 & 0 \end{bmatrix}$

$\begin{bmatrix} 0 & 10 & 6 \end{bmatrix}$

$\begin{bmatrix} 0 & 0 & 9 \end{bmatrix}$

Normalized confusion matrix

$\begin{bmatrix} 1.0 & 0.0 & 0.0 \end{bmatrix}$

$\begin{bmatrix} 0.0 & 0.62 & 0.38 \end{bmatrix}$

$\begin{bmatrix} 0.0 & 0.0 & 1.0 \end{bmatrix}$

## Python Implementation

Implementing it in python is extremely easy:

```
from sklearn.metrics import confusion_matrix
```

```
# expected and predicted are the initial and predicted arrays.  
result = confusion_matrix(expected, predicted)
```

```
print(results)
```

A fantastic way to start and build up a technical conversation is to have a candidate describe how a model with which they are familiar works. Technical interviews can often be very stressful for candidates and this is one way to allow candidates to relax slightly and talk about something in which they have more experience. It doesn't matter if they choose something very simple because the goal is to see if the candidate really understands the model and doesn't just know the basics. Going into substantial depth on something as simple as k-nearest neighbors or linear regression can be quite revealing about a candidate.

- What type of problem does the model try to solve?
- Is it prone to over-fitting? If so – what can be done about this?
- Does the model make any important assumptions about the data? When might these be unrealistic? How do we examine the data to test whether these assumptions are satisfied?
- Does the model have convergence problems? Does it have a random component or will the same training data always generate the same model? How do we deal with random effects in training?
- What types of data (numerical, categorical etc...) can the model handle?
- Can the model handle missing data? What could we do if we find missing fields in our data?
- How interpretable is the model?
- What alternative models might we use for the same type of problem that this one attempts to solve, and how does it compare to those?
- Can we update the model without retraining it from the beginning?

How fast is prediction compared to other models? How fast is training compared to other models?

Does the model have any meta-parameters and thus require tuning? How do we do this?

What is the EM algorithm? Give a couple of applications?

What is deep learning and what are some of the main characteristics that distinguish it from traditional machine learning?

What is linear in a generalized linear model?

What is a probabilistic graphical model? What is the difference between Markov networks and Bayesian networks?

Give an example of an application of non-negative matrix factorization

On what type of ensemble technique is a random forest based? What particular limitation does it try to address?

What methods for dimensionality reduction do you know and how do they compare with each other?

What are some good ways for performing feature selection that do not involve exhaustive search?

How would you evaluate the quality of the clusters that are generated by a run of K-means?

1. What do you mean by Machine learning?

- Machine learning is an IT field that focuses on machine programming to understand and develop knowledge automatically. For instance: bots are designed to accomplish the mission on the basis of data they obtain from detectors. It develops programming from user information.

2. Name a few machine learning algorithms

- Decision Tree, Neural Networks , Nearest Neighbor, Probabilistic networks, Support vector machines.

3. Explain the types of machine learning

- Supervised Learning: Machines train under the guidance of designated data in this form of machine learning technique. The machine is focused on a testing dataset and provides its performance according to its preparation.
- Reinforcing Learning: reinforcement learning requires models which learn and cross to create the best step possible. In order to seek to

determine the next possible course of action, algorithms for reinforcement learning are built on the basis of reward to punishment theory.

- Unsupervised learning: it does not have unlabeled results, unlike supervised learning. There is therefore no monitoring under which the data are being processed. Uncontrolled learning essentially tries to identify data patterns and create similar entities clusters. Once the model is reached with new input details, it does not identify the entity; rather, it positions the entity in a community of related objects.

#### 4. Explain inductive logic programming

- Genetic programming is among the multiple computer learning methods. The algorithm is designed to evaluate and pick from a range of outcomes the best alternative.

#### 5. What do you mean by model selection?

- The selection method between features of various computational frameworks that are used to represent the same dataset is referred to as the model selection. The application of models is implemented in analytics , machine learning and data analysis areas.

#### 6. Give one advantage and disadvantage of decision trees

- Benefits: Decision trees (which ensures that they're resilient to outliers) are simple to interpret, non parametric, and relative parameters can be modified.
- Disadvantage: Decision trees are unable to overfit. Yet ensemble approaches such as random forests or enhanced trees may fix this.

#### 7. Explain cross validation

- Cross-validation is primarily a method used to test the efficiency of a concept on a different and independent basis. The easiest method for cross-validation is by splitting the data into two groups: training data and test data, using the training data for model creation and testing data for model research ..

## 8. What is a classifier?

- A Machine Learning classifier is a program that inputs a matrix of distinct or cumulative significance of the function and outputs a single discrete class value.

## 9. What is the advantage of a Neural network?

- Neural networks, in particular, deep NNs, have contributed to breakthroughs in output for unorganized databases including images, sound / visual. The unbelievable simplicity helps them know patterns, which no other ML algorithm can do.

## 10. What is Principle Component Analysis?

- PCA is a tool of integrating features in unrelated linear combinations for turning features into a data collection. These new features or main components sequentially optimize the defined variation (i.e. the first main component is most variant, the second most significant version, etc.). It implies that PCA is valuable for raising dimensionality, since an adjustable variance limit is feasible.

## 11. What are Random forests?

- Random forests are an array of decision-making processes. Random forests require the development of several judgment gems by bootstrapping original data datasets and the random collection of a subset of variables at each stage. Afterwards the algorithm selects the mode of a Decision Tree prediction. This reduces the chance of a person tree mistake by utilizing a formula "vote wins."

## 12. When do you use Random forests?

- Random forests will decide the value of your app. It can not be achieved by SVM. Random forests are simpler and easier than an SVM to build. SVMs need a one-vs-rest approach for numerous classification issues, which is less sized and more resource consuming.

### 13. Explain kernel

- A kernel is a way to measure the point product in any (possibly very high-dimensional) field of two vectors  $xx$  and  $yy$ , which is why kernel functions are often named "generalized point product." The network model is a way to address a highly nonlinear question by converting linear regression data into linearly segregated data in higher dimensions.

### 14. Explain overfitting

- In computer analysis, a random error or noise is represented by a mathematical model instead of an underlying 'overfitting' relationship. When a model becomes too complicated, overfitting may typically arise because the amount of training data types contains so many parameters. The layout is badly implemented and overcrowded.

### 15. How is data mining different from machine learning?

- The research, product development of the algorithms allowing computers to work without specific programmatization contribute to machine learning. Data mining may, however, be described as the method in which data attempts to obtain information of unknown trends. Training algorithms are used in this method computer.

### 16. Explain parametric models

- The structures with the minimal amount of parameters are parametric structures. You just have to learn the performance of the network to predict new results. Examples cover linear regression, functional and linear SVM regression.

### 17. What is a REgression?

- It is the method of constructing the model to separate data from groups or distinct values into continuous actual values. Based on the historical evidence, it may also describe the propagation motion. This is used to forecast an occurrence based on the degree to which the factors are mixed. Of eg, the weather prediction relies on variables including temperature , air currents, solar radiation, region elevation and sea distance. The interaction between these variables allows one to forecast the environment.

## 18. What do you mean by confusion matrix?

- The confusion matrix is used to describe the success of a process and includes an overview of the category issues forecasts. It helps in the determination of class unpredictability.

## 19. Explain variance and bias

- Bias is the discrepancy between our model's average and the correct one. The model's forecast is not reliable when the bias value is strong. The bias factor would also be as small as practicable in order to produce the required predictions. Variance is the sum that shows the disparity between a forecast and the predicted value of certain sets of instruction. High variance may contribute to major production variability. The performance of the model will therefore be small.

## 20. What do you mean by linear regression?

- Linear regression is a guided algorithm for machine learning. This is used for statistical modeling to locate causal connections between the additive and the independent variables.

## 21. What is the importance of rotation in PCA?

- Rotation is a major step in PCA since it maximizes the separation of components within the variance. It makes it easy to understand the elements. The explanation why PCA is used is to pick fewer components that can describe the largest variation in a data collection. The initial positions of the points are modified when rotation is done. The relative location of the elements, however, is not modified. If the components are not rotating, the variation needs to be represented with extended components.

## 22. Explain k-means cluster

- It is an unattended algorithm in machine learning. Here we provide the model with unidentified (unlabeled) details. The algorithm then produces loads of points dependent on the average distances of various points.

### 23. Explain Bagging

- We use random sampling and then divide the data set into n. Afterwards, we construct a model of one training algorithm. They instead merge actual survey forecasts. Bagging aims to boost the model's efficacy by raising the variation from overriding.

### 24. What do you mean by Standardization?

- The approach used for rescaling device attributes is standardisation. The attributes will have a mean value of 0 and a minimum value of 1. The primary goal of standardization is to speed up the composite and standard attributes variance.

### 25. What are Support Vector Machines?

- SVM is an algorithm used primarily for classification of machine learning. The signature function is placed over the strong dimensionality.

### 26. What is Logistics regression?

- Logistic regression describes the right application of regression where a categorical or conditional dependent variable is used. Logistic regression is, like other regression analyzes, a statistical modeling method. The data and the relation from one contingent random vector and one or more variables is clarified using a logistic regression. The expectation of a categorically contingent variable is often used to forecast.

### 27. What are different types of Logistic regression?

- Three kinds of logistic regression emerge:
- Binary logistic regression: only two results are probable in this sense. Example: To determine whether (1) or not (0) is going to rain
- Multinomial logistics regression: For this the performance consists of three or four unordered groups. Example: Local language prediction (Kannada, Telugu, Marathi, etc.).

- Ordinary logistic regression: the output comprises of three or more organized groups, in the ordinary logistic regression.Example: ranking an app from 1 to 5 stars for Android.

## Tell me about Machine Learning.

**Machine learning** is a branch of computer science that deals with system programming to learn and develop automatically over time. For instance, robots are programmed to perform a task based on data collected from sensors. It learns programs from data on its own.

### 2. What do you mean by Inductive machine learning?

**Inductive machine learning** is the method of learning by example, in which a system attempts to infer a general rule from a collection of observed instances.

### 3. Tell the difference between Data Mining and Machine Learning?

The research, design, and creation of algorithms that enable computers to learn without being specifically programmed is referred to as machine learning. Data mining, on the other hand, is the method of extracting information or unknown interesting patterns from unstructured data. **Machine learning** algorithms are used in this process.

### 4. Explain Overfitting in Machine Learning?

Overfitting occurs in **machine learning** when a mathematical model defines random error or noise rather than the underlying relationship. Overfitting is common when a model is too complex, as a result of providing too many parameters concerning the number of training data types. The model has been overfitted, resulting in poor results.

**5. Do You know the reason why Overfitting happens?** Overfitting happens and is a risk because the parameters used to train the model are not the same as the criteria used to assess the model's efficacy.

### 6. List the 5 Popular Algorithms of Machine learning.

- Decision Trees
- Neural Networks (backpropagation)
- Probabilistic networks
- Support vector machines
- Nearest Neighbor

## **7. What are the various Machine Learning Algorithm**

**Techniques?** Machine Learning techniques come in a variety of forms. They are

- Supervised Learning
- Unsupervised Learning
- Semi-supervised Learning
- Reinforcement Learning
- Transduction

**8. What standard approach does supervised learning use?** The traditional method for supervised learning is to divide the example set into two parts: **the training set and the test set.**

**9. Explain the Training set and the Test set?** A collection of data is used in various fields of information technology, such as machine learning, to discover the potentially predictive relationship known as the ‘Training Set.’ A training set is a set of examples given to the learner, while a Test set is a set of examples withheld from the learner to test the accuracy of the hypotheses developed by the learner. Training and Test sets are not the same thing.

**10. In Machine learning, what are the three phases of developing hypotheses or models?**

- Model building
- Model testing
- Applying the model

**11. Explain Algorithm Independent Machine learning?**

Algorithm-independent **machine learning** refers to machine learning in which the **mathematical foundations** are independent of any single classifier or learning algorithm.

**2. Tell me the difference between Artificial Intelligence(AI) and Machine Learning(ML)?**

Machine Learning is the process of designing and creating algorithms based on **empirical data and behavior**. Artificial intelligence encompasses a variety of topics in addition to machine learning, such as information representation, natural language processing, planning, robotics, etc.

**13. Tell me about Classifiers in Machine Learning?**

In Machine Learning, a classifier is a method that takes a vector of discrete or continuous feature values as input and outputs a single discrete value, the class.

#### **14. What do you know about Genetic Programming?**

Genetic Programming (GP) is a subset of machine learning that uses **Evolutionary Algorithms (EA)**. EAs are used to find immediate solutions to problems that humans are unable to solve.

#### **15. Tell us the real-time applications of Pattern Recognition?**

- Computer Vision
- Speech Recognition
- Data Mining
- Statistics
- Informal Retrieval
- Bio-Informatics

#### **16. Explain Model Selection in Machine Learning?**

Model selection is the method of choosing a model from among many mathematical models that are used to represent the same data set. In the fields of statistics, **machine learning**, and data mining, model selection is used.

#### **17. What is Machine Learning's Inductive Logic Programming? ILP**

(Inductive Logic Programming) is a machine learning subfield that employs logical programming to reflect context information and examples.

#### **18. What are Bayesian Networks?**

The Bayesian Network is a graphical model for the **probability relationship** between a series of variables.

#### **19. What are the two components of the Bayesian Logic Program?**

There are two sections to the Bayesian logic program. The first is a logical part, which consists of a collection of Bayesian Clauses that capture the domain's qualitative structure. The second part is a quantitative one that encodes the domain's quantitative data.

#### **20. Which technique is most commonly used to avoid overfitting?**

To avoid an overfitting problem, 'Isotonic Regression' is used when there is enough data.

## **21. Define Perceptron.**

Perceptron is a machine learning algorithm for the supervised classification of input into one of several non-binary outputs.

## **22. Explain Ensemble Learning and why is it used?**

Multiple models, such as classifiers or experts, are strategically created and combined to solve a specific computational program. Ensemble learning is the name for this process. Ensemble learning is a technique for improving a model's classification, prediction, and function approximation, among other things.

## **23. In the ensemble process, what is the bias-variance decomposition of classification error?**

A learning algorithm's predicted error can be broken down into bias and variance. A bias term indicates how closely the learning algorithm's average classifier fits the target function. The variance term expresses how much the learning algorithm's prediction varies across training sets.

## **24. Why is it that an instance-based learning algorithm is also known as a lazy learning algorithm?**

Instance-based learning algorithms are also known as Lazy learning algorithms because they postpone induction or generalization until classification is completed.

## **25. Explain Dimension Reduction in Machine learning.**

Dimension reduction, which can be split into feature selection and feature extraction in Machine Learning and statistics, is the method of reducing the number of random variables under consideration.

## **26. Tell us the difference between supervised and unsupervised machine learning?**

Training labeled data is needed for supervised learning. To do classification (a supervised learning task), for example, you must first mark the data that will be used to train the model to classify data into your labeled classes. Unsupervised learning, on the other hand, does not necessitate clear data marking.

## **27. Explain the working of the ROC curve?**

At different thresholds, the ROC curve is a graphical representation of the contrast between **true positive rates** and **false-positive rates**. It's sometimes used as a proxy for the trade-off between the model's sensitivity (true positives) and the fall-out, or the likelihood of a false alarm (false positives).

## **28. How is KNN different from k-means clustering?**

K-means clustering is an unsupervised clustering algorithm, while K-Nearest Neighbors is a supervised classification algorithm. Although the mechanisms can appear to be identical at first glance, the truth is that for K-Nearest Neighbors to function, you must have labeled data to classify an unlabeled point into (thus the nearest neighbor part). Only a set of unlabeled points and a threshold are required for K-means clustering: the algorithm will take unlabeled points and eventually learn how to cluster them into categories by computing the mean of the distance between various points.

## **29. Do you know the reason why the Naive Bayes algorithm is said to be naive?**

Despite its practical applications, especially in text mining, Naive Bayes is labeled “Naive” because it relies on an assumption that is nearly impossible to verify in real-world data: the conditional probability is calculated as the pure product of the individual probabilities of components. This necessitates complete feature independence, which is a requirement that is unlikely to be fulfilled in real life.

## **30. What is the difference between L1 regularisation and L2 regularisation?**

L2 regularisation spreads error over all items, while L1 regularisation is more binary/sparse, with several variables assigned a 1 or 0 in weighting. Setting a Laplacean prior on the terms corresponds to L1, thus setting a Gaussian prior corresponds to L2.

## **31. Tell the difference between Type I and Type II error.**

A false positive is a Type I error, while a false negative is a Type II error. Type I error is defined as believing something has occurred when it hasn't, while Type II error is defined as claiming nothing is occurring when something is.

## **32. What is your favorite algorithm and explain it?**

## **33. Explain Decision tree pruning?**

Pruning is the process of removing branches from a decision tree model that have low predictive power to reduce the model's complexity and improve its predictive accuracy. Pruning can be done from the bottom up or from the top

down, using techniques including reduced error pruning and cost complexity pruning. The simplest version of reduced error pruning is to replace each node. Keep it pruned if it doesn't reduce predictive accuracy. Despite its simplicity, this heuristic is quite similar to a method that would optimize for full accuracy.

### **34. How do you make sure you are not overfitting a model?**

This is a straightforward restatement of a fundamental problem in machine learning: the risk of overfitting training data and bringing the noise into the test set, resulting in incorrect generalizations. You can avoid overfitting by following these methods.

- Reduce uncertainty by incorporating fewer variables and parameters into the model, eliminating some of the noise from the training results.
- Use techniques like k-folds cross-validation for cross-validation.
- Using regularisation techniques like LASSO to penalize those model parameters that are prone to overfitting.

### **35. When Ensemble Techniques will be useful?**

To improve predictive efficiency, ensemble techniques combine many learning algorithms. They usually help models become more stable by reducing overfitting (unlikely to be influenced by small changes in the training data).

### **36. Explain the Kernel trick and its uses?**

Kernel functions enable in higher-dimension spaces without directly computing the coordinates of points within that dimension: instead, kernel functions calculate the inner products between the images of all pairs of data in a feature space, which is known as the Kernel trick. This gives them the very useful property of being able to calculate the coordinates of higher dimensions while being computationally cheaper than doing so explicitly. Inner products can be used to express a variety of algorithms. We can run algorithms in a high-dimensional space with lower-dimensional data by employing the kernel trick.

### **37. Tell us how you use the F1 score?**

The F1 score is a metric for how well a model performs. It's a weighted average of a model's precision and recall, with scores ranging from 1 to 0, with 1 being the best and 0 being the worst. It would be used in classification tests where true negatives aren't as relevant.

### **38. List the components for relational evaluation techniques.**

- Data Acquisition
- Ground Truth Acquisition
- Cross-Validation Technique
- Scoring Metric
- Significance Test
- Query Type

### **39. Tell me what you know about Batch Statistical Learning?**

Statistical learning techniques allow you to learn a feature or indicator from a collection of observed data that can be used to forecast data that hasn't been seen before. Based on a statistical assumption about the data generation process, these techniques provide guarantees on the performance of the learned predictor on future unseen data.

### **40. List out the Supervised learning functions?**

- Classifications
- Regression
- Forecast time series
- Annotate strings
- Speech recognition

### **41. What are the functions of Unsupervised learning?**

- Look for data clusters.
- Discover low-dimensional data representations.
- Look for interesting directions in data.
- Correlations and coordinates of interest
- Find observations/clean up the database

### **42. List the two methods used in the Calibration of Supervised learning?**

In Supervised Learning, there are two methods for estimating successful probabilities.

- Platt Calibration
- Isotonic Regression

These approaches are intended for binary classification, which is an important task.

### **43. Why are PCA, KPCA, and ICA used?**

PCA (Principal Components Analysis), KPCA (Kernel-based Principal Component Analysis), and ICA (Independent Component Analysis) are common dimensionality reduction techniques

#### **44. Explain Reinforcement learning?**

Reinforcement learning is a type of learning in which an agent communicates with its surroundings by performing actions and discovering errors or rewards. It's like being stranded on a deserted island, where you must discover the environment on your own and learn to survive and adapt to the extreme conditions. The hit-and-trial approach is used by the model to understand. It learns by receiving a reward or a penalty for each action it takes.

#### **45. Which is better? Too many False Positives or Too many False Negatives? Explain.**

It depends on the query as well as the situation for which the problem is being solved. If you're using Machine Learning in the field of medical research, a false negative is a big risk, since the report won't reveal any health issues even if the individual is sick. Similarly, if Machine Learning is used to detect spam, a false positive is extremely dangerous because the algorithm can mistakenly identify a critical email as spam.

#### **46. Model Accuracy or Model Performance. What do you think is more important?**

You should be aware, however, that model accuracy is just one aspect of model efficiency. The model's accuracy and efficiency are directly proportional, so the higher the model's performance, the more accurate the predictions are.

#### **47. Explain A/B testing?**

**Statistical hypothesis testing** for a **randomized experiment** with two variables A and B are known as A/B. It's used to compare two models that use different predictor variables to see which one matches a given set of data the best. Consider the following scenario: you've built two models (each with its own set of predictor variables) that can be used to suggest goods on an e-commerce platform. These two models can be compared using A/B Testing to see which one better recommends services to a consumer.

#### **48. Tell us in detail about Cluster Sampling?**

It's a method of selecting unified groups with similar characteristics at random from a given population. A cluster sample is a probability sample in which each

sampling unit is a group of elements. Managers (samples) will represent elements, and companies will represent clusters if you're clustering the total number of managers in a group of companies.

## 49. How can imbalanced datasets be handled?

If you have a classification test, for example, and 90% of the data is in one class, you have an imbalanced dataset. This causes issues: a 90% accuracy can be skewed if you don't have any predictive capacity on the other type of data! Here are several strategies for getting over the imbalance:

- Collect more data to even out the dataset's imbalances.
- To compensate for imbalances, resample the dataset.
- On your dataset, try a different algorithm entirely.

## 50. What are the various categories in which the sequence learning process can be classified?

- Sequence prediction
- Sequence generation
- Sequential decision
- Sequence recognition

Why can't we use Mean Square Error (MSE) as a cost function for logistic regression?

In logistic regression, we use the sigmoid function and perform a non-linear transformation to obtain the probabilities. Squaring this non-linear transformation will lead to non-convexity with local minimums. Finding the global minimum in such cases using gradient descent is not possible. Due to this reason, MSE is not suitable for logistic regression. Cross-entropy or log loss is used as a cost function for logistic regression. In the cost function for logistic regression, the confident wrong predictions are penalised heavily. The confident right predictions are rewarded less. By optimising this cost function, convergence is achieved.

What is the output of a standard MLE program?

The output of a standard MLE program is as follows:

**Maximised likelihood value:** This is the numerical value obtained by replacing the unknown parameter values in the likelihood function with the MLE parameter estimator.

**Estimated variance-covariance matrix:** The diagonal of this matrix consists of the estimated variances of the ML estimates. The off-diagonal consists of the covariances of the pairs of the ML estimates

What are the advantages and disadvantages of conditional and unconditional methods of MLE?

Conditional methods do not estimate unwanted parameters. Unconditional methods estimate the values of unwanted parameters also. Unconditional formulas can directly be developed with joint probabilities. This cannot be done with conditional probability. If the number of parameters is high relative to the number of instances, then the unconditional method will give biased results. Conditional results will be unbiased in such cases.

What are the different methods of MLE and when is each method preferred?

In the case of logistic regression, there are two approaches to MLE. They are conditional and unconditional methods. Conditional and unconditional methods are algorithms that use different likelihood functions. The unconditional formula employs the joint probability of positives (for example, churn) and negatives (for example, non-churn). The conditional formula is the ratio of the probability of observed data to the probability of all possible configurations.

The unconditional method is preferred if the number of parameters is lower compared to the number of instances. If the number of parameters is high compared to the number of instances, then conditional MLE is to be preferred. Statisticians suggest that conditional MLE is to be used when in doubt. Conditional MLE will always provide unbiased results.

What is the Maximum Likelihood Estimator (MLE)?

The MLE chooses those sets of unknown parameters (estimator) that maximise the likelihood function. The method to find the MLE is to use calculus and setting the derivative of the logistic function with respect to an unknown parameter to zero, and solving it will give the MLE. For a binomial model, this will be easy, but for a logistic model, the calculations are complex. Computer programs are used for deriving MLE for logistic models.

*(Here's another approach to answering the question.)*

MLE is a statistical approach to estimate the parameters of a mathematical model. MLE and ordinary square estimation give the same results for linear regression if the dependent variable is assumed to be normally distributed. MLE does not assume anything about independent variables.

What is the formula for calculating odds ratio?

The formula can be given as:

$$OR_{X_1, X_2} = e^{\sum_{i=1}^k \beta_i (X_{1i} - X_{0i})}$$

In the formula above,  $X_1$  and  $X_0$  stand for two different groups for which the odds ratio needs to be calculated.  $X_{1i}$  stands for the instance ‘ $i$ ’ in group  $X_1$ .  $X_{0i}$  stands for the instance ‘ $i$ ’ in group  $X_0$ .  $\beta_0$  stands for the coefficient of the logistic regression model. Note that the baseline is not included in this formula.

What is odds ratio?

Odds ratio is the ratio of odds between two groups. For example, let’s assume that you are trying to ascertain the effectiveness of a medicine. You administered this medicine to the ‘intervention’ group and a placebo to the ‘control’ group.

$$\text{Odds Ratio (OR)} = \frac{\text{Odds of the Intervention Group}}{\text{Odds of the Control Group}}$$

## Interpretation

- If odds ratio = 1, then there is no difference between the intervention group and the control group.
- If the odds ratio is greater than 1, then the control group is better than the intervention group.
- If the odds ratio is less than 1, then the intervention group is better than the control group.

How to interpret the results of a logistic regression model? Or, what are the meanings of the different betas in a logistic regression model?

$\beta_0$  is the baseline in a logistic regression model. It is the log odds for an instance when all the attributes ( $X_1, X_2, X_3, \dots, X_n$ ) are zero. In practical scenarios, the probability of all the attributes being zero is very low. In another interpretation,  $\beta_0$  is the log odds for an instance when none of the attributes is taken into consideration.

All the other Betas are the values by which the log odds change by a unit change in a particular attribute by keeping all other attributes fixed or unchanged (control variables).

What are the outputs of the logistic model and the logistic function?

The logistic model outputs the logits, i.e. log odds; and the logistic function outputs the probabilities.

Logistic model =

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \dots + \beta_k X_k$$

The output of the same will be logits.

$$\text{Logistic function} = f(z) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \dots + \beta_n X_n)}}$$

The output, in this case, will be the probabilities

Why is logistic regression very popular/widely used?

Logistic regression is famous because it can convert the values of logits (log-odds), which can range from  $-\infty$  to  $+\infty$  to a range between 0 and 1. As logistic functions output the probability of occurrence of an event, it can be applied to many real-life scenarios. It is for this reason that the logistic regression model is very popular. Another reason why logistic fares in comparison to linear regression is that it is able to handle the categorical variables.

What are the differences between logistic regression and linear regression?

The main important differences between logistic and linear regression are:

1. Dependent/response variable in linear regression is continuous whereas, in logistic regression, it is the discrete type.
2. Cost function in linear regression minimise the error term  $\text{Sum}(\text{Actual}(Y) - \text{Predicted}(Y))^2$  but logistic regression uses maximum likelihood method for maximising probabilities.

**Problem Statement:** Given the data of individuals and their healthcare charges billed by their insurance provider ([Click here to download data](#)), following are the columns in the data set:

- sex: Gender of the individual (female, male)
- bmi: Body mass index (You can read more about BMI [here](#).)
- children: Number of children covered under health insurance / number of dependants
- smoker: Indicates whether the person smokes or not
- region: The individual's residential area in the US
- charges: Medical costs borne by the health insurance provider for the individual

Here, "charges" will be the dependent variable and all the other variables are independent variables.

**Question 1: Following are some questions that require you to do some EDA, data preparation and finally perform linear regression on the data set to predict the healthcare charges for the individuals.**

**Create another feature based called BMI\_group which groups people based on their BMI. The groups should be as follows:**

- Underweight: **BMI** is less than 18.5.
- Normal: **BMI** is 18.5 to 24.9.
- Overweight: **BMI** is 25 to 29.9.
- Obese: **BMI** is 30 or more.

The grouping is based on WHO standards.

The output should have first five rows of the resulting dataframe.

```
import pandas as pd
pd.set_option('display.max_columns', 500)
df=pd.read_csv("")
def bmi_group(val):
    if val<18.5:
        return "Underweight"
    if (val>=18.5) & (val<24.9):
        return "Normal"
    if (val>=24.9) & (val<=29.9):
        return "Overweight"
    if val>=30:
        return "Obese"

df["BMI_group"] = df.bmi.apply(bmi_group)
print(df.head())
```

**Question 2: Encode all categorical features such that they can be used in a regression model. i.e.sex, BMI\_group, smoker and region should be labelled properly. Use the label encoder for all features.**

```
import pandas as pd
from sklearn.preprocessing import LabelEncoder
pd.set_option('display.max_columns', 500)
df=pd.read_csv("")

le = LabelEncoder()
#sex
le.fit(df.sex.drop_duplicates())
df.sex = le.transform(df.sex)
```

```
# smoker or not  
le.fit(df.smoker.drop_duplicates())  
df.smoker = le.transform(df.smoker)  
#region  
le.fit(df.region.drop_duplicates())  
df.region = le.transform(df.region)  
#changing data type  
df.BMI_group=df.BMI_group.astype(str)  
le.fit(df.BMI_group.drop_duplicates())  
df.BMI_group = le.transform(df.BMI_group)  
print(df.head())
```

**Question 3:** As everyone knows, smoking is a major cause of bad health. Here, try to find if smoking is affecting health of people. Print the correlation value of "smoker" columns with "bmi", "age" and "charges" columns in three lines respectively. Note: You should round off all three values till four decimal places using the round() function.

```
import pandas as pd  
df=pd.read_csv("")  
  
print(round(df.smoker.corr(df.bmi),4))  
print(round(df.smoker.corr(df.age),4))  
print(round(df.smoker.corr(df.charges),4))
```

**Question 4:** We have divided the dataset now into test and train sets. Since you already saw that being a smoker and healthcare charges are highly correlated. Try to create a linear regression model using only the "smoker" variable as the independent variable and "charges" as dependent variable.

**Note:** All operations you performed in the previous questions have already been performed on the dataset here.

Click [here](#) to download train data

You can take any other measures to ensure a better outcome if you want. The dataset has been divided into train and test sets and both have been loaded in the coding console. You have to write the predictions in the file: /code/output/predictions.csv. You have to add the predictions in a column titled "predicted\_charges" in the test dataset. Make sure you use the same column name otherwise your score won't be evaluated.

Your model's R-squared will be evaluated on an unseen test dataset. The R-squared of your model should be greater than 0.6.

```
import numpy as np
```

```

import pandas as pd

# Read training data
train = pd.read_csv("insurance_training.csv")

# Read test data
test = pd.read_csv("insurance_test.csv")

# Linear regression
from sklearn.linear_model import LinearRegression
lr = LinearRegression()
lr.fit(np.array(train['smoker']).reshape(-1,1),train['charges'])
y_test_pred=lr.predict(np.array(test['smoker']).reshape(-1,1))

# Write the output
test["predicted_charges"]=y_test_pred
test.to_csv("/code/output/predictions.csv")

```

Question 5: You saw that by using only the "smoker" variable, you can get an r-squared of 0.66 easily. Now your task is to perform linear regression using the entire dataset.

**Note: All operations you performed in the questions 1-3 have already been performed on the dataset here.**

You can take any other measures to ensure a better outcome if you want.(for example: normalising or standardising any values or adding any other columns).

[Click here to download train data](#)

You have to write the predictions in the file: /code/output/predictions.csv. You have to add the predictions in a column titled "predicted\_charges" in the test dataset. Make sure you use the same column name otherwise your score won't be evaluated.

Your model's R-squared-adjusted will be evaluated on an unseen test dataset. The R-squared of your model should be greater than 0.72.

```

import numpy as np
import pandas as pd

# Read training data
train = pd.read_csv("insurance_training.csv")

# Read test data

```

```

test = pd.read_csv("insurance_test.csv")

# Linear regression
from sklearn.linear_model import LinearRegression
lr = LinearRegression()
lr.fit(train.drop(["region","charges"],axis=1),train['charges'])
y_test_predicted=lr.predict(test.drop("region",axis=1))

# Write the output
#Do not edit the last two lines here
#reload test set before this step if you have made any changes to the test set
test["predicted_charges"]=y_test_predicted
test.to_csv("/code/output/predictions.csv")

```

What is the likelihood function?

The likelihood function is the joint probability of observing the data. For example, let's assume that a coin is tossed 100 times and you want to know the probability of getting 60 heads from the tosses. This example follows the binomial distribution formula.

- **p = Probability of heads from a single coin toss**
- **n = 100 (the number of coin tosses)**
- **x = 60 (the number of heads – success)**
- **n - x = 40 (the number of tails)**
- **Pr (X=60 | n = 100, p)**

The likelihood function is the probability that the number of heads received is 60 in a trail of 100 coin tosses, where the probability of heads received in each coin toss is p. Here the coin toss result follows a binomial distribution.

This can be reframed as follows:

- **Pr(X=60|n=100, p) = c × p<sup>60</sup> × (1-p)<sup>100-60</sup>**
- **c = constant**
- **p = unknown parameter**

The likelihood function gives the probability of observing the results using unknown parameters.

Why can't linear regression be used in place of logistic regression for binary classification?

The reasons why linear regressions cannot be used in case of binary classification are as follows:

**Distribution of error terms:** The distribution of data in the case of linear and logistic regression is different. Linear regression assumes that error terms are normally distributed. In the case of binary classification, this assumption does not hold true.

**Model output:** In linear regression, the output is continuous. In the case of binary classification, an output of a continuous value does not make sense. For binary classification problems, linear regression may predict values that can go beyond 0 and 1. If we want the output in the form of probabilities, which can be mapped to two different classes, then its range should be restricted to 0 and 1. As the logistic regression model can output probabilities with logistic/sigmoid function, it is preferred over linear regression.

**Variance of Residual errors:** Linear regression assumes that the variance of random errors is constant. This assumption is also violated in the case of logistic regression.

What are odds?

It is the ratio of the probability of an event occurring to the probability of the event not occurring. For example, let's assume that the probability of winning a lottery is 0.01. Then, the probability of not winning is  $1 - 0.01 = 0.99$ .

Now, as per the definition,

The odds of winning the lottery = (Probability of winning) / (Probability of not winning)

The odds of winning the lottery =  $0.01/0.99$

Hence, the odds of winning the lottery is 1 to 99, and the odds of not winning the lottery is 99 to 1

How can the probability of a logistic regression model be expressed as a conditional probability?

The conditional probability can be given as:

$P(\text{Discrete value of target variable} | X_1, X_2, X_3 \dots X_N)$

It is the probability of the target variable to take up a discrete value (either 0 or 1 in case of binary classification problems) when the values of independent variables are given. For example, the probability an employee will attrite (target variable) given his attributes such as his age, salary, KRA's, etc.

What is the formula for the logistic regression function?

In general, the formula for logistic regression is given by the following expression:

$$f(z) = \frac{1}{(1 + e^{-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k)})}$$

What are the differences between logistic regression and linear regression?

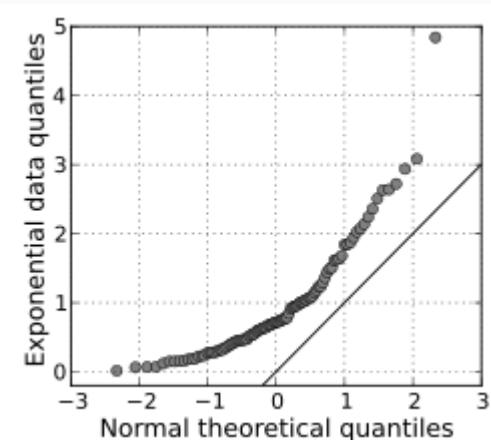
The main important differences between logistic and linear regression are:

1. Dependent/response variable in linear regression is continuous whereas, in logistic regression, it is the discrete type.
2. Cost function in linear regression minimise the error term  $\text{Sum}(\text{Actual}(Y) - \text{Predicted}(Y))^2$  but logistic regression uses maximum likelihood method for maximising probabilities.

What is a Q-Q plot? Explain the use and importance of a Q-Q plot in linear regression

Q-Q Plots (Quantile-Quantile plots) are plots of two quantiles against each other. A quantile is a fraction where certain values fall below that quantile. For example, the median is a quantile where 50% of the data fall below that point and 50% lie above it. The purpose of Q-Q plots is to find out if two sets of data come from the same distribution. A 45 degree angle is plotted on the Q-Q plot; if the two data sets come from a common distribution, the points will fall on that reference line.

A Q-Q plot showing the 45 degree reference line:



If the two distributions being compared are similar, the points in the Q-Q plot will approximately lie on the line  $y = x$ . If the distributions are linearly related, the points in the Q-Q plot will approximately lie on a line, but not necessarily on

the line  $y = x$ . Q–Q plots can also be used as a graphical means of estimating parameters in a location-scale family of distributions.

A Q–Q plot is used to compare the shapes of distributions, providing a graphical view of how properties such as location, scale, and skewness are similar or different in the two distributions.

You might have observed that sometimes the value of VIF is infinite. Why does this happen?

If there is perfect correlation, then  $VIF = \infty$ . This shows a perfect correlation between two independent variables. In the case of perfect correlation, we get  $R^2 = 1$ , which lead to  $1/(1-R^2) = \infty$ . To solve this problem we need to drop one of the variables from the dataset which is causing this perfect multicollinearity.

An infinite VIF value indicates that the corresponding variable may be expressed exactly by a linear combination of other variables (which show an infinite VIF as well).

What is the difference between normalized scaling and standardized scaling?

Normalization typically means rescales the values into a range of  $[0,1]$ . Standardization typically means rescales data to have a mean of 0 and a standard deviation of 1 (unit variance).

S.NO.	Normalisation	Standardisation
1.	Minimum and maximum value of features are used for scaling	Mean and standard deviation is used for scaling.
2.	It is used when features are of different scales.	It is used when we want to ensure zero mean and unit standard deviation.
3.	Scales values between $[0, 1]$ or $[-1, 1]$ .	It is not bounded to a certain range.
4.	It is really affected by outliers.	It is much less affected by outliers.

S.NO.	<b>Normalisation</b>	<b>Standardisation</b>
5.	Scikit-Learn provides a transformer called MinMaxScaler for Normalization.	Scikit-Learn provides a transformer called StandardScaler for standardization.
6.	This transformation squishes the n-dimensional data into an n-dimensional unit hypercube.	It translates the data to the mean vector of the original data to the origin and squishes or expands.
7.	It is useful when we don't know about the distribution	It is useful when the feature distribution is Normal or Gaussian.
8.	It is often called as Scaling Normalization	It is often called as Z-Score Normalization

### **What is scaling? Why is scaling performed?**

It is a step of data Pre-Processing which is applied to independent variables to normalize the data within a particular range. It also helps in speeding up the calculations in an algorithm.

Most of the times, collected data set contains features highly varying in magnitudes, units and range. If scaling is not done then algorithm only takes magnitude in account and not units hence incorrect modelling. To solve this issue, we have to do scaling to bring all the variables to the same level of magnitude.

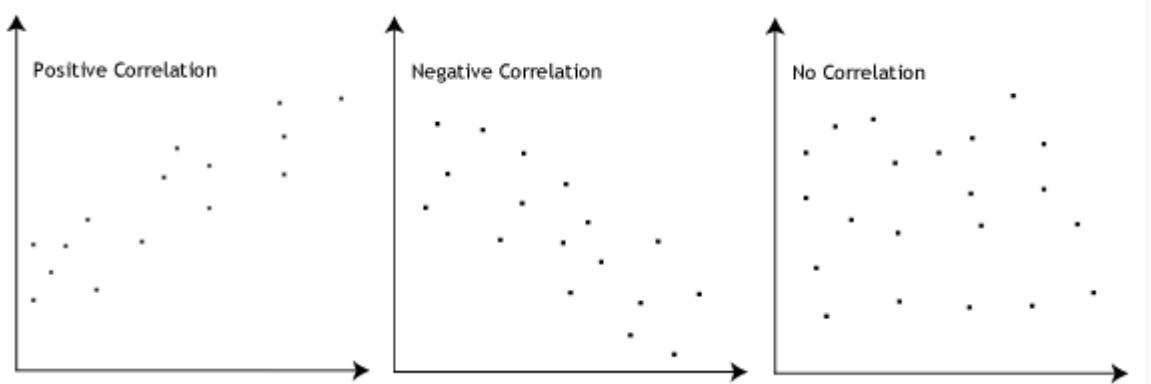
It is important to note that scaling just affects the coefficients and none of the other parameters like t-statistic, F-statistic, p-values, R-squared, etc.

### **What is Pearson's R?**

In statistics, the Pearson correlation coefficient (PCC), also referred to as Pearson's  $r$ , the Pearson product-moment correlation coefficient (PPMCC), or the bivariate correlation, is a measure of linear correlation between two sets of data. It is the covariance of two variables, divided by the product of their standard deviations; thus it is essentially a normalised measurement of the covariance, such that the result always has a value between  $-1$  and  $1$ .

The Pearson's correlation coefficient varies between  $-1$  and  $+1$  where:

- $r = 1$  means the data is perfectly linear with a positive slope ( i.e., both variables tend to change in the same direction)
- $r = -1$  means the data is perfectly linear with a negative slope ( i.e., both variables tend to change in different directions)
- $r = 0$  means there is no linear association
- $r > 0 < 5$  means there is a weak association
- $r > 5 < 8$  means there is a moderate association
- $r > 8$  means there is a strong association



Pearson r Formula

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

Here,

- $r$  =correlation coefficient
- $x_i$  =values of the x-variable in a sample
- $\bar{x}$  =mean of the values of the x-variable
- $y_i$  =values of the y-variable in a sample
- $\bar{y}$  =mean of the values of the y-variable

### Explain the Anscombe's quartet in detail.?

Anscombe's quartet comprises four datasets that have nearly identical simple statistical properties, yet appear very different when graphed. Each dataset consists of eleven (x,y) points. They were constructed in 1973 by the statistician

Francis Anscombe to demonstrate both the importance of graphing data before analyzing it and the effect of outliers on statistical properties.

### Simple understanding:

Once Francis John “Frank” Anscombe who was a statistician of great repute found 4 sets of 11 data-points in his dream and requested the council as his last wish to plot those points. Those 4 sets of 11 data-points are given below.

I		II		III		IV	
x	y	x	y	x	y	x	y
10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89

After that, the council analyzed them using only descriptive statistics and found the mean, standard deviation, and correlation between x and y.

### Explain the linear regression algorithm in detail?

Linear regression is one of the very basic forms of machine learning where we train a model to predict the behaviour of your data based on some variables. In the case of linear regression as you can see the name suggests linear that means the two variables which are on the x-axis and y-axis should be linearly correlated.

Mathematically, we can write a linear regression equation as:

$$y = a + bx$$

Where a and b given by the formulas:

$$b(\text{slope}) = \frac{n \sum xy - (\sum x)(\sum y)}{n \sum x^2 - (\sum x)^2}$$

$$a(\text{intercept}) = \frac{n \sum y - b(\sum x)}{n}$$

Here, x and y are two variables on the regression line.

b = Slope of the line

a = y-intercept of the line

x = Independent variable from dataset

y = Dependent variable from dataset

### **Explain the bias-variance trade-off.?**

Bias refers to the difference between the values predicted by the model and the real values. It is an error. One of the goals of an ML algorithm is to have a low bias.

Variance refers to the sensitivity of the model to small fluctuations in the training data set. Another goal of an ML algorithm is to have low variance.

For a data set that is not exactly linear, it is not possible to have both bias and variance low at the same time. A straight line model will have low variance but high bias, whereas a high-degree polynomial will have low bias but high variance.

There is no escaping the relationship between bias and variance in machine learning.

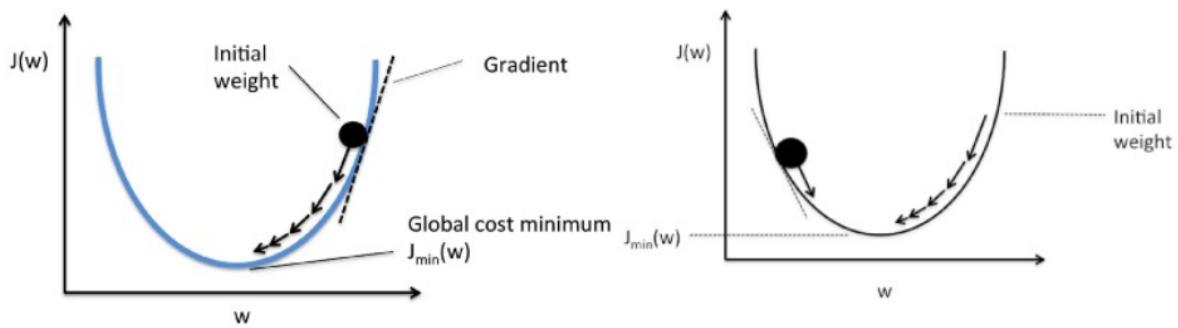
1. Decreasing the bias increases the variance.
2. Decreasing the variance increases the bias.

So, there is a trade-off between the two; the ML specialist has to decide, based on the assigned problem, how much bias and variance can be tolerated. Based on this, the final model is built.

### **Explain gradient descent with respect to linear regression.?**

Gradient descent is an optimisation algorithm. In linear regression, it is used to optimise the cost function and find the values of the  $\beta$ s (estimators) corresponding to the optimised value of the cost function.

Gradient descent works like a ball rolling down a graph (ignoring the inertia). The ball moves along the direction of the greatest gradient and comes to rest at the flat surface (minima).



Mathematically, the aim of gradient descent for linear regression is to find the solution of  $\text{ArgMin } J(\Theta_0, \Theta_1)$ , where  $J(\Theta_0, \Theta_1)$  is the cost function of the linear regression. It is given by:

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)})^2$$

Here,  $h$  is the linear hypothesis model,  $h = \Theta_0 + \Theta_1 x$ ,  $y$  is the true output, and  $m$  is the number of datapoints in the training set.

Gradient descent starts with a random solution, and then, based on the direction of the gradient, the solution is updated to the new value, where the cost function has a lower value.

The update is:

Repeat until convergence:

$$\theta_j = \theta_j - \alpha \frac{1}{m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_j^{(i)} \text{ for } j = 1, 2, \dots, n$$

**What is the major difference between R-squared and adjusted R-squared? Or, why is it advised to use adjusted R-squared in case of multiple linear regression?**

The major difference between R-squared and adjusted R-squared is that R-squared does not penalise the model for having a higher number of variables. Thus, if you keep on adding variables to the model, the R-squared value will

always increase (or remain the same in case the value of the correlation between that variable and the dependent variable is zero). Thus, R-squared assumes that any variable added to the model will increase the predictive power.

Adjusted R-squared, on the other hand, penalises models based on the number of variables present in it. Its formula is given as:

$$\text{Adj. } R^2 = \frac{1 - (1 - R^2)(N-1)}{N-k-1}$$

where 'N' is the number of datapoints and 'k' is the number of features.

So, if you add a variable and the adjusted R-squared drops, you can be certain that that variable is insignificant to the model and, hence, should not be used. Thus, in the case of multiple linear regression, you should always look at the adjusted R-squared value in order to keep redundant variables out of your regression model.

### How can you handle categorical variables present in the data set?

Many a time, your data set may have categorical variables that are potentially good predictors for the response variable. So, handling them right is quite crucial.

One of the ways to handle categorical data with just two levels is to do a binary mapping of the variables, wherein one of the levels will correspond to zero and the other to 1.

Another way of handling categorical variables with a few levels is to perform dummy encoding. The key idea behind dummy encoding is that for a variable with, say, 'N' levels, you create 'N-1' new indicator variables for each of these levels. So for a variable say, 'Relationship' with three levels, namely, 'Single', 'In a Relationship', and 'Married', you would create a dummy table like the following:

Relationship Status	Single	In a Relationship	Married
Single	1	0	0
In a Relationship	0	1	0
Married	0	0	1

But you can clearly see that there is no need to define **three** different levels. If you drop a level, say 'Single', you would still be able to explain the three levels.

Let's drop the dummy variable 'Single' from the columns and see what the table looks like:

Relationship Status	In a Relationship	Married
Single	0	0
In a Relationship	1	0
Married	0	1

If both the dummy variables, namely, 'In a Relationship' and 'Married', are equal to zero, that means that the person is single. If 'In a relationship' is one and 'Married' is zero, that means that the person is in a relationship, and finally, if 'In a relationship' is zero and 'Married' is 1, that means that the person is married.

Now, creating dummy variables may be useful when the number of levels in a categorical variable is small, but if a categorical variable has a hundred levels, it is clearly impossible to create 99 new variables. In such cases, grouping the variables could be useful. For example, for the variable 'Cities in India', you can use geographical grouping, such as follows:

- Keep the 'n' largest cities, group the rest.
- Geographical hierarchy:
  - City -> District -> State -> Zone
- Group cities with similar values for the outcome variable.
- Cluster cities with similar values for the predictor variables.

### **What is multicollinearity? How does it affect the linear regression? How can you deal with it?**

Multicollinearity occurs when some of the independent variables are highly correlated (positively or negatively) with each other. This causes a problem, as it is against the basic assumption of linear regression. The presence of multicollinearity does not affect the predictive capability of the model. So, if you just want predictions, the presence of multicollinearity does not affect your

output. However, if you want to draw some insights from the model and apply them in, let's say, some business model, it may cause problems.

One of the major problems caused by multicollinearity is that it leads to incorrect interpretations and offers wrong insights. The coefficients of linear regression suggest the mean change in the target value if a feature is changed by one unit. So, if multicollinearity exists, this does not hold true, as changing one feature will lead to changes in the correlated variable and consequent changes in the target variable. This leads to wrong insights and can produce hazardous results for a business.

A highly effective way of dealing with multicollinearity is the use of VIF (variance inflation factor). The higher the value of VIF for a feature, the more linearly correlated that feature is. Simply remove the feature with a very high VIF value and retrain the model on the remaining data set.

### **What is the difference between least squares error and mean squared error?**

Least squares error is the method used to find the best-fit line through a set of datapoints. The idea behind the least squares error method is to minimise the square of errors between the actual datapoints and the line fitted.

Mean squared error, on the other hand, is used once you have fitted the model and want to evaluate it. So, the mean squared error finds out the average of the difference between the actual and predicted values and, hence, is a good parameter to compare various models on the same data set.

Thus, LSE is a method used during model fitting to minimise the sum of squares, and MSE is a metric used to evaluate the model after fitting the model, based on the average squared errors.

### **If two variables are correlated, do they necessarily have a linear relationship?**

No, not necessarily. If two variables are correlated, they can possibly have any relationship and not just a linear one.

But the important point to note here is that there are two correlation coefficients that are widely used in regression. One is Pearson's R correlation coefficient, which is the correlation coefficient that you learnt about in the linear regression model. This correlation coefficient is designed for linear relationships, and it might not be a good measure for a non-linear relationship between the variables.

The other correlation coefficient is Spearman's R, which is used to determine the correlation if the relationship between the variables is not linear. So, even though Pearson's R may give a correlation coefficient for non-linear relationships, it might not be reliable. For example, the correlation coefficients, as given by both the techniques for the relationship  $y=X^3$  for 100 equally separated values between 1 and 100, were found out to be:

$$\text{Pearson's R} \approx 0.91$$

$$\text{Spearman's R} \approx 1$$

And as we keep on increasing the power, the Pearson's R value consistently drops, while the Spearman's R remains robust at 1. For example, for the relationship  $y=X^{10}$  for the same datapoints, the coefficients were:

$$\text{Pearson's R} \approx 0.66$$

$$\text{Spearman's R} \approx 1$$

So, the takeaway here is that if you have some sense of the relationship being non-linear, you should look at Spearman's R instead of Pearson's R. It might happen that even for a non-linear relationship, the Pearson's R value might be high, but it is simply not reliable.

### **What parameters are used to check the significance of the model and the goodness of fit?**

To check whether the overall model fit is significant or not, the primary parameter to be looked at is the **F-statistic**. While the t-test, along with the p-values for betas, tests whether each coefficient is significant or not individually, the F-statistic is a measure to determine whether the overall model fit with all the coefficients is significant or not.

The basic idea behind the F-test is that it is a relative comparison between the model that you have built and the model without any of the coefficients, except for  $\beta_0$ . If the value of the F-statistic is high, it would mean that the Prob(F) would be low and, hence, you can conclude that the model is significant. On the other hand, if the value of F-statistic is low, it might lead to the Prob(F) being higher than the significance level (taken as 0.05 usually), which, in turn, would conclude that the overall model fit is insignificant and the intercept-only model can provide a better fit.

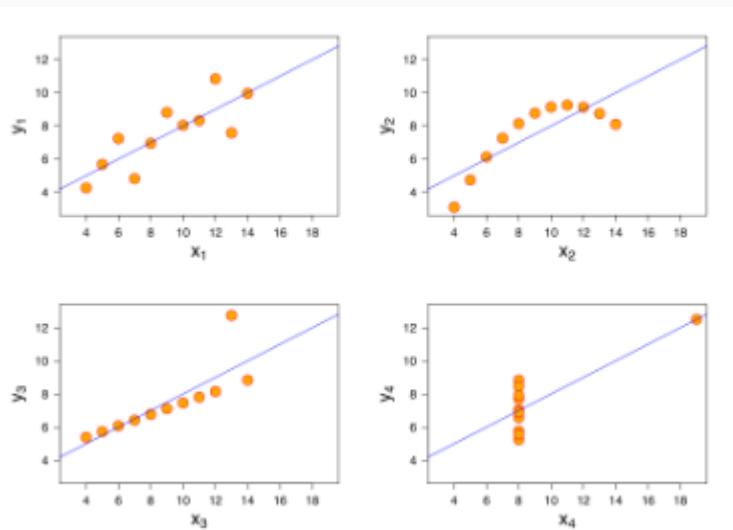
Apart from that, to test the goodness or the extent of fit, we look at a parameter called **R-squared** (for simple linear regression models) or **adjusted R-squared** (for multiple linear regression models). If your overall model fit is deemed to be significant by the F-test, you can go ahead and look at the value of R-squared. This value lies between 0 and 1, with 1 meaning a perfect fit. A higher value of R-squared is indicative of the model being good with much of the variance in the data being explained by the straight line fitted. For example, an R-squared value of 0.75 means that 75% of the variance in the data is being explained by the model. But it is important to remember that R-squared only tells the extent of the fit and should not be used to determine whether the model fit is significant or not.

### What are the shortcomings of linear regression?

You should never just run a regression without having a good look at your data because simple linear regression has quite a few shortcomings:

1. It is sensitive to outliers.
2. It models linear relationships only.
3. A few assumptions are required to make the inference.

These phenomena can be best explained by the Anscombe's Quartet, shown below:



As we can see, all the four linear regression are exactly the same. But there are some peculiarities in the data sets that have fooled the regression line. While the first one seems to be doing a decent job, the second one clearly shows that linear regression can only model linear relationships and is incapable of handling any other kind of data. The third and fourth images showcase the linear regression

model's sensitivity to outliers. Had the outlier not been present, we could have got a great line fitted through the data points. So, we should never run a regression without having a good look at our data.

## How do you interpret a linear regression model?

A linear regression model is quite easy to interpret. The model is of the following form:

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n$$

The significance of this model lies in the fact that one can easily interpret and understand the marginal changes and their consequences. For example, if the value of  $x_0$  increases by 1 unit, keeping other variables constant, the total increase in the value of  $y$  will be  $\beta_i$ . Mathematically, the intercept term ( $\beta_0$ ) is the response when all the predictor terms are set to zero or not considered.

## How is hypothesis testing used in linear regression?

Hypothesis testing can be carried out in linear regression for the following purposes:

1. To check whether a predictor is significant for the prediction of the target variable. Two common methods for this are as follows:
  1. By the use of p-values: If the p-value of a variable is greater than a certain limit (usually 0.05), the variable is insignificant in the prediction of the target variable.
  2. By checking the values of the regression coefficient: If the value of the regression coefficient corresponding to a predictor is zero, that variable is insignificant in the prediction of the target variable and has no linear relationship with it.
2. To check whether the calculated regression coefficients are good estimators of the actual coefficients.

The null and alternative hypotheses used in the case of linear regression, respectively, are:

- $\beta_1 = 0$
- $\beta_1 \neq 0$

Thus, if we reject the null hypothesis, we can say that the coefficient  $\beta_1$  is not equal to zero and, hence, is significant for the model. On the other hand, if we fail

to reject the null hypothesis, we can conclude that the coefficient is insignificant and should be dropped from the model.

### **How do you know that linear regression is suitable for any given data?**

To see whether linear regression is suitable for any given data, a scatter plot can be used. If the relationship looks linear, we can go for a linear model. But if it is not the case, we have to apply some transformations to make the relationship linear. Plotting the scatter plots is easy in case of simple or univariate linear regression. But in the case of multivariate linear regression, two-dimensional pair-wise scatter plots, rotating plots, and dynamic graphs can be plotted.

### **What is heteroscedasticity? What are the consequences and how can you overcome it?**

A random variable is said to be heteroscedastic when different subpopulations have different variabilities (standard deviation).

The existence of heteroscedasticity gives rise to certain problems in the regression analysis as the assumption says that error terms are uncorrelated and, hence, the variance is constant. The presence of heteroscedasticity can often be seen in the form of a cone-like scatter plot for residual vs fitted values.

One of the basic assumptions of linear regression is that the data should be homoscedastic, i.e., heteroscedasticity is not present in the data. Due to the violation of assumptions, the ordinary least squares (OLS) estimators are not the Best Linear Unbiased Estimators (BLUE). Hence, they do not give a lesser variance than other Linear Unbiased Estimators (LUEs).

There is no fixed procedure to overcome heteroscedasticity. However, there are some ways that may lead to a reduction of heteroscedasticity. They are:

- 1. Logarithmising the data:** A series that is increasing exponentially often results in increased variability. This can be overcome using the log transformation.
- 2. Using weighted linear regression:** Here, the OLS method is applied to the weighted values of X and Y. One way is to attach weights directly related to the magnitude of the dependent variable.

### **What are the assumptions in a linear regression mode?**

The assumptions of linear regression are:

- Assumption about the form of the model:** It is assumed that there is a linear relationship between the dependent and independent variables. It is known as the ‘linearity assumption’.
- Assumptions about the residuals:**
  - Normality assumption:** It is assumed that the error terms,  $\varepsilon^{(i)}$ , are normally distributed.
  - Zero mean assumption:** It is assumed that the residuals have a mean value of zero, i.e., the error terms are normally distributed around zero.
  - Constant variance assumption:** It is assumed that the residual terms have the same (but unknown) variance,  $\sigma^2$ . This assumption is also known as the assumption of homogeneity or homoscedasticity.
  - Independent error assumption:** It is assumed that the residual terms are independent of each other, i.e., their pair-wise covariance is zero.
- Assumptions about the estimators:**
  - The independent variables are measured without error.
  - The independent variables are linearly independent of each other, i.e., there is no multicollinearity in the data.

If the residuals are not normally distributed, their randomness is lost, which implies that the model is not able to explain the relation in the data. Also, the mean of the residuals should be zero.

$$Y^{(i)i} = \beta_0 + \beta_1 X^{(i)} + \varepsilon^{(i)}$$

This is the assumed linear model, where  $\varepsilon$  is the residual term.

$$\begin{aligned} E(Y) &= E(\beta_0 + \beta_1 X^{(i)} + \varepsilon^{(i)}) \\ &= E(\beta_0 + \beta_1 X^{(i)} + \varepsilon^{(i)}) \end{aligned}$$

If the expectation(mean) of residuals,  $E(\varepsilon^{(i)})$ , is zero, the expectations of the target variable and the model become the same, which is one of the targets of the model. The residuals (also known as the error terms) should be independent, meaning there is no correlation between the residuals and the predicted values, or among the residuals. Any correlation implies that there is some relation that the regression model is not able to identify.

If the independent variables are not linearly independent of each other, the uniqueness of the least squares solution (or normal equation solution) is lost.

## What is Linear Regression?

In simple terms, linear regression is a method of finding the best straight line fitting to the given data, i.e., finding the best linear relationship between the independent and dependent variables.

In technical terms, linear regression is a machine learning algorithm that finds the best linear-fit relationship on any given data, between independent and dependent variables. It is mostly done by the Residual Sum of Squares Method.

**Suppose you performed encoding with the variable ‘BloodGroup’ having four levels ‘A’ ‘B’ ‘AB’ and ‘O’.?**

**To perform the encoding, you wish to drop two of the levels, ‘AB’ and ‘O’. Suggest a suitable encoding process that will now represent the four levels.**

A	-	10
B	-	01
AB	-	11
O - 00		

Note that this encoding is not exactly dummy encoding; it's just manual encoding that you performed.

**According to you how does the REF measure the importance of the variable ?**

Recursive feature elimination is based on the idea of repeatedly constructing a model (for example, an SVM or a regression model) and choosing either the best or the worst performing feature (for example, based on coefficients), setting the feature aside and then repeating the process with the rest of the features.

This process is applied until all the features in the data set are exhausted. Features are then ranked according to when they were eliminated. As such, it is a greedy optimisation for finding the best performing subset of features.

**Why according to you is it better to use adjusted R-squared in multiple linear regression?**

The major difference between R-squared and adjusted R-squared is that R-squared does not penalise the model for having more number of variables. Thus, if you keep on adding variables to the model, the R-squared will always increase (or remain the same when the value of the correlation between that variable and

the dependent variable is 0). Thus, R-squared assumes that any variable added to the model will increase the predictive power.

Adjusted R-squared, on the other hand, penalises models on the basis of the number of variables present in them. So, if you add a variable and the adjusted R-squared drops, you can be certain that that variable is insignificant to the model and should not be used. Thus, in the case of multiple linear regression, you should always look at the adjusted R-squared value in order to keep redundant variables out of your regression model.

### **Is validation required for clustering? If yes; then why is it required?**

Clustering algorithms have a tendency to cluster even when the data is random. It is essential to validate if a non-random structure is present in the data. It is also required to validate whether the number of clusters formed is appropriate or not.

Evaluation of clusters is done with or without external reference to check the fitness of the data. Evaluation is also done to compare clusters and decide the better among them.

### **What are the disadvantages of agglomerative hierarchical clustering?**

**Objective function:** SSE is the objective function for K-means. Likewise, there exists no global objective function for hierarchical clustering. It considers proximity locally before merging two clusters.

**Time and space complexity:** The time and space complexity of agglomerative clustering is more than K-means clustering, and in some cases, it is prohibitive.

**Final merging decisions:** The merging decisions, once given by the algorithm, cannot be undone at a later point in time. Due to this, a local optimisation criteria cannot become global criteria. Note that there are some advanced approaches available to overcome this problem.

### **What are the types of hierarchical clustering?**

There are two types of hierarchical clustering. They are agglomerative clustering and divisive clustering.

**Agglomerative clustering:** In this algorithm, initially every data object will be treated as a cluster. In each step, the nearest clusters will fuse together and form

a bigger cluster. Ultimately, all the clusters will merge together. Finally, a single cluster, which encompasses all the data points, will remain.

**Divisive clustering:** This is the opposite of the agglomerative clustering. In this type, all the data objects will be considered as single clusters. In each step, the algorithm will split the cluster. This will repeat until only single data points remain, which will be considered as singleton clusters.

### **Is K-means clustering suitable for all shapes and sizes of clusters?**

K-means is not suitable for all shapes, sizes, and densities of clusters. If the natural clusters of a dataset are vastly different from a spherical shape, then K-means will face great difficulties in detecting it. K-means will also fail if the sizes and densities of the clusters are different by a large margin. This is mostly due to using SSE as the objective function, which is more suited for spherical shapes. SSE is not suited for clusters with non-spherical shapes, varied cluster sizes, and densities.

### **What is the objective function for measuring the quality of clustering in case of the K-means algorithm with Euclidean distance?**

Sum of squared errors (SSE) is used as the objective function for K-means clustering with Euclidean distance. The Euclidean distance is calculated from each data point to its nearest centroid. These distances are squared and summed to obtain the SSE. The aim of the algorithm is to minimize the SSE. Note that SSE considers all the clusters formed using the K-means algorithm.

### **How are outliers handled by the K-means algorithm?**

Handling of outliers differs from case to case. In some cases, it will provide very useful information, and in some cases, it will severely affect the results of the analysis. Having said that, let's learn about some of the issues that arise due to outliers in the K-means algorithm below.

The centroids will not be a true representation of a cluster in the presence of outliers. The sum of squared errors (SSE) will also be very high in the case of outliers. Small clusters will bond with outliers, which may not be the true representation of the natural patterns of clusters in data. Due to these reasons, outliers need to be removed before proceeding with clustering on the data.

## **What are the issues with random initialization of centroids in K-means algorithm and how to overcome it?**

Initiation of the centroids in a cluster is one of the most important steps of the K-means algorithm. Many times, random selection of initial centroid does not lead to an optimal solution. In order to overcome this problem, the algorithm is run multiple times with different random initialisations. The sum of squared errors (SSE) are calculated for different initial centroids.

The set of centroids with the minimum SSE is selected. Even though this is a very simple method, it is not foolproof. The results of multiple random cluster initialisations will depend on the dataset and the number of clusters selected, however, that still will not give an optimum output every time.

The other method involves first selecting the centroid of all the data points. Then, for each successive initial centroid, select the point which is the farthest from the already selected centroid. This procedure will ensure that the selection is random, and the centroids are far apart. The disadvantage of this method is that calculating the farthest point will be expensive.

In order to avoid this problem, initialisation is carried out on a subset of the dataset.

## **What are the different proximity functions or distance metrics used for the K-means algorithm?**

Euclidean, Manhattan, Cosine, and Bregman divergence are some distance metrics used for the K-means algorithm. Euclidean is the squared distance from a data point to the centroid. Manhattan is the absolute distance from a data point to the centroid. Cosine is the cosine distance from a data point to the cluster centroid. Bregman divergence is a class of distance metrics that includes Euclidean, Mahalanobis, and Cosine. Basically, Bregman divergence includes all those distance metrics for which the mean is a centroid.

## **Explain the types of segmentation that can be considered while solving a business problem using Clustering.**

You need to mention the RFM analysis, Behavioral Segmentation and Psychographic Segmentation.

**Explain the steps of K-means Clustering algorithm. Mention the key steps that need to be followed and how the algorithm works.**

The algorithm for K-means algorithm is as follows:

- Select initial centroids. The input regarding the number of centroids should be given by the user.
- Assign the data points to the closest centroid
- Recalculate the centroid for each cluster and assign the data objects again
- Follow the same procedure until convergence. Convergence is achieved when there is no more assignment of data objects from one cluster to another, or when there is no change in the centroid of clusters.

### **K-Modes clustering: Bank Marketing Data**

Download the data set from here. Some pointers before you proceed:

- Use only the following columns 'job', 'marital', 'education', 'default', 'housing', 'loan', 'contact', 'month', 'day\_of\_week', 'poutcome', 'age', 'duration', 'euribor3m' where age, duration and euribor3m are the numerical columns.
- Convert all categorical columns to numeric by using LabelEncoder()
- Standardize all the columns before using K-Prototype clustering
- Remember that you also need to convert the final dataframe to a matrix for applying K-Prototype.
- First check K-prototype with the number of clusters as 5.
- Please keep in mind that the code may take some time to execute as there are so many categorical variables, so be patient.

**Q1: Check if your final data set has any missing values in it? Please remember that this question is needed to be answered after selecting the required columns as stated above in the pointers i.e. 21.**

- True
- False

**Explanation:** Use df.info() and check.

**Q2: What is the average "Duration" before standardising the data?**

**Answer:** 258.285. Check your answer by running df.describe()

**Q3: Run the loop to check the cost against the number of clusters ranging from 1 to 8 and identify the suitable number of clusters.( More than one answer may be correct)**

**Answer:** 4. The answer is subjective and based on the business problem we are trying to solve.

Download the solution from [here](#).

**Can you use the dendrogram to make meaningful clusters? (By looking at which elements leave and join at what height)**

Yes. It is a great tool. You can look at what stage an element is joining a cluster and hence see how similar or dissimilar it is to the rest of the cluster. If it joins at the higher height, it is quite different from the rest of the group. You can also see which elements are joining which cluster at what stage and can thus use business understanding to cut the dendrogram more accurately.

**What are the benefits of Hierarchical Clustering over K-Means clustering?  
What are the disadvantages?**

Hierarchical clustering generally produces better clusters, but is more computationally intensive.

**Q1. You are given a train data set having 1000 columns and 1 million rows. The data set is based on a classification problem. Your manager has asked you to reduce the dimension of this data so that model computation time can be reduced. Your machine has memory constraints. What would you do? (You are free to make practical assumptions.)**

**Answer:** Processing a high dimensional data on a limited memory machine is a strenuous task, your interviewer would be fully aware of that. Following are the methods you can use to tackle such situation:

1. Since we have lower RAM, we should close all other applications in our machine, including the web browser, so that most of the memory can be put to use.

2. We can randomly sample the data set. This means, we can create a smaller data set, let's say, having 1000 variables and 300000 rows and do the computations.
3. To reduce dimensionality, we can separate the numerical and categorical variables and remove the correlated variables. For numerical variables, we'll use correlation. For categorical variables, we'll use chi-square test.
4. Also, we can use PCA and pick the components which can explain the maximum variance in the data set.
5. Using online learning algorithms like Vowpal Wabbit (available in Python) is a possible option.
6. Building a linear model using Stochastic Gradient Descent is also helpful.
7. We can also apply our business understanding to estimate which all predictors can impact the response variable. But, this is an intuitive approach, failing to identify useful predictors might result in significant loss of information.

**Q2. Is rotation necessary in PCA? If yes, Why? What will happen if you don't rotate the components?**

**Answer:** Yes, rotation (orthogonal) is necessary because it maximizes the difference between variance captured by the component. This makes the components easier to interpret. Not to forget, that's the motive of doing PCA where, we aim to select fewer components (than features) which can explain the maximum variance in the data set. By doing rotation, the relative location of the components doesn't change, it only changes the actual coordinates of the points.

If we don't rotate the components, the effect of PCA will diminish and we'll have to select more number of components to explain variance in the data set.

**Q3. You are given a data set. The data set has missing values which spread along 1 standard deviation from the median. What percentage of data would remain unaffected? Why?**

**Answer:** This question has enough hints for you to start thinking! Since, the data is spread across median, let's assume it's a normal distribution. We know, in a

normal distribution, ~68% of the data lies in 1 standard deviation from mean (or mode, median), which leaves ~32% of the data unaffected. Therefore, ~32% of the data would remain unaffected by missing values.

**Q4. You are given a data set on cancer detection. You've build a classification model and achieved an accuracy of 96%. Why shouldn't you be happy with your model performance? What can you do about it?**

**Answer:** If you have worked on enough data sets, you should deduce that cancer detection results in imbalanced data. In an imbalanced data set, accuracy should not be used as a measure of performance because 96% (as given) might only be predicting majority class correctly, but our class of interest is minority class (4%) which is the people who actually got diagnosed with cancer. Hence, in order to evaluate model performance, we should use Sensitivity (True Positive Rate), Specificity (True Negative Rate), F measure to determine class wise performance of the classifier. If the minority class performance is found to be poor, we can undertake the following steps:

1. We can use undersampling, oversampling or SMOTE to make the data balanced.
2. We can alter the prediction threshold value by doing probability calibration and finding an optimal threshold using AUC-ROC curve.
3. We can assign weight to classes such that the minority classes get larger weight.
4. We can also use anomaly detection.

**Q5. Why is naive Bayes so ‘naive’ ?**

**Answer:** naive Bayes is so ‘naive’ because it assumes that all of the features in a data set are equally important and independent. As we know, these assumptions are rarely true in real world scenario.

**Q6. Explain prior probability, likelihood and marginal likelihood in context of naiveBayes algorithm?**

**Answer:** Prior probability is nothing but, the proportion of dependent (binary) variable in the data set. It is the closest guess you can make about a class, without any further information. For example: In a data set, the dependent variable is binary (1 and 0). The proportion of 1 (spam) is 70% and 0 (not spam) is 30%. Hence, we can estimate that there are 70% chances that any new email would be classified as spam.

Likelihood is the probability of classifying a given observation as 1 in presence of some other variable. For example: The probability that the word ‘FREE’ is used in previous spam message is likelihood. Marginal likelihood is, the probability that the word ‘FREE’ is used in any message.

**Q7. You are working on a time series data set. Your manager has asked you to build a high accuracy model. You start with the decision tree algorithm, since you know it works fairly well on all kinds of data. Later, you tried a time series regression model and got higher accuracy than decision tree model. Can this happen? Why?**

**Answer:** Time series data is known to possess linearity. On the other hand, a decision tree algorithm is known to work best to detect non – linear interactions. The reason why decision tree failed to provide robust predictions because it couldn’t map the linear relationship as good as a regression model did. Therefore, we learned that, a linear regression model can provide robust prediction given the data set satisfies its linearity assumptions.

**Q8. You are assigned a new project which involves helping a food delivery company save more money. The problem is, company’s delivery team aren’t**

**able to deliver food on time. As a result, their customers get unhappy. And, to keep them happy, they end up delivering food for free. Which machine learning algorithm can save them?**

**Answer:** You might have started hopping through the list of ML algorithms in your mind. But, wait! Such questions are asked to test your machine learning fundamentals.

This is not a machine learning problem. This is a route optimization problem. A machine learning problem consist of three things:

1. There exist a pattern.
2. You cannot solve it mathematically (even by writing exponential equations).
3. You have data on it.

Always look for these three factors to decide if machine learning is a tool to solve a particular problem.

**Q9. You came to know that your model is suffering from low bias and high variance. Which algorithm should you use to tackle it? Why?**

**Answer:** Low bias occurs when the model's predicted values are near to actual values. In other words, the model becomes flexible enough to mimic the training data distribution. While it sounds like great achievement, but not to forget, a flexible model has no generalization capabilities. It means, when this model is tested on an unseen data, it gives disappointing results.

In such situations, we can use bagging algorithm (like random forest) to tackle high variance problem. Bagging algorithms divides a data set into subsets made with repeated randomized sampling. Then, these samples are used to generate a

set of models using a single learning algorithm. Later, the model predictions are combined using voting (classification) or averaging (regression).

Also, to combat high variance, we can:

1. Use regularization technique, where higher model coefficients get penalized, hence lowering model complexity.
2. Use top n features from variable importance chart. May be, with all the variable in the data set, the algorithm is having difficulty in finding the meaningful signal.

**Q10. You are given a data set. The data set contains many variables, some of which are highly correlated and you know about it. Your manager has asked you to run PCA. Would you remove correlated variables first? Why?**

**Answer:** Chances are, you might be tempted to say No, but that would be incorrect. Discarding correlated variables have a substantial effect on PCA because, in presence of correlated variables, the variance explained by a particular component gets inflated.

For example: You have 3 variables in a data set, of which 2 are correlated. If you run PCA on this data set, the first principal component would exhibit twice the variance than it would exhibit with uncorrelated variables. Also, adding correlated variables lets PCA put more importance on those variable, which is misleading.

**Q11. After spending several hours, you are now anxious to build a high accuracy model. As a result, you build 5 GBM models, thinking a boosting algorithm would do the magic. Unfortunately, neither of models could perform better than benchmark score. Finally, you decided to combine those models. Though, ensembled models are known to return high accuracy, but you are unfortunate. Where did you miss?**

**Answer:** As we know, ensemble learners are based on the idea of combining weak learners to create strong learners. But, these learners provide superior results when the combined models are uncorrelated. Since, we have used 5 GBM models and got no accuracy improvement, suggests that the models are correlated. The problem with correlated models is, all the models provide same information.

For example: If model 1 has classified User1122 as 1, there are high chances model 2 and model 3 would have done the same, even if its actual value is 0. Therefore, ensemble learners are built on the premise of combining weak uncorrelated models to obtain better predictions.

## **Q12. How is kNN different from kmeans clustering?**

**Answer:** Don't get misled by 'k' in their names. You should know that the fundamental difference between both these algorithms is, kmeans is unsupervised in nature and kNN is supervised in nature. kmeans is a clustering algorithm. kNN is a classification (or regression) algorithm.

kmeans algorithm partitions a data set into clusters such that a cluster formed is homogeneous and the points in each cluster are close to each other. The algorithm tries to maintain enough separability between these clusters. Due to unsupervised nature, the clusters have no labels.

kNN algorithm tries to classify an unlabeled observation based on its k (can be any number ) surrounding neighbors. It is also known as lazy learner because it involves minimal training of model. Hence, it doesn't use training data to make generalization on unseen data set.

**Q13. How is True Positive Rate and Recall related? Write the equation.**

**Answer:** True Positive Rate = Recall. Yes, they are equal having the formula  $(TP/TP + FN)$ .

**Q14. You have built a multiple regression model. Your model  $R^2$  isn't as good as you wanted. For improvement, you remove the intercept term, your model  $R^2$  becomes 0.8 from 0.3. Is it possible? How?**

**Answer:** Yes, it is possible. We need to understand the significance of intercept term in a regression model. The intercept term shows model prediction without any independent variable i.e. mean prediction. The formula of  $R^2 = 1 - \sum(y - y')^2 / \sum(y - y_{mean})^2$  where  $y'$  is predicted value.

When intercept term is present,  $R^2$  value evaluates your model wrt. to the mean model. In absence of intercept term ( $y_{mean}$ ), the model can make no such evaluation, with large denominator,  $\sum(y - y')^2 / \sum(y)^2$  equation's value becomes smaller than actual, resulting in higher  $R^2$ .

**Q15. After analyzing the model, your manager has informed that your regression model is suffering from multicollinearity. How would you check if he's true? Without losing any information, can you still build a better model?**

**Answer:** To check multicollinearity, we can create a correlation matrix to identify & remove variables having correlation above 75% (deciding a threshold is subjective). In addition, we can use calculate VIF (variance inflation factor) to check the presence of multicollinearity. VIF value  $\leq 4$  suggests no multicollinearity whereas a value of  $\geq 10$  implies serious multicollinearity. Also, we can use tolerance as an indicator of multicollinearity.

But, removing correlated variables might lead to loss of information. In order to retain those variables, we can use penalized regression models like ridge or lasso regression. Also, we can add some random noise in correlated variable so that the variables become different from each other. But, adding noise might affect the prediction accuracy, hence this approach should be carefully used.

### **Q16. When is Ridge regression favorable over Lasso regression?**

**Answer:** You can quote ISLR's authors Hastie, Tibshirani who asserted that, in presence of few variables with medium / large sized effect, use lasso regression. In presence of many variables with small / medium sized effect, use ridge regression.

Conceptually, we can say, lasso regression (L1) does both variable selection and parameter shrinkage, whereas Ridge regression only does parameter shrinkage and end up including all the coefficients in the model. In presence of correlated variables, ridge regression might be the preferred choice. Also, ridge regression works best in situations where the least square estimates have higher variance. Therefore, it depends on our model objective.

### **Q17. Rise in global average temperature led to decrease in number of pirates around the world. Does that mean that decrease in number of pirates caused the climate change?**

**Answer:** After reading this question, you should have understood that this is a classic case of “causation and correlation”. No, we can't conclude that decrease in number of pirates caused the climate change because there might be other factors (lurking or confounding variables) influencing this phenomenon.

Therefore, there might be a correlation between global average temperature and number of pirates, but based on this information we can't say that pirates died because of rise in global average temperature.

**Q18. While working on a data set, how do you select important variables?**

**Explain your methods.**

**Answer:** Following are the methods of variable selection you can use:

1. Remove the correlated variables prior to selecting important variables
2. Use linear regression and select variables based on p values
3. Use Forward Selection, Backward Selection, Stepwise Selection
4. Use Random Forest, Xgboost and plot variable importance chart
5. Use Lasso Regression
6. Measure information gain for the available set of features and select top n features accordingly.

**Q19. What is the difference between covariance and correlation?**

**Answer:** Correlation is the standardized form of covariance.

Covariances are difficult to compare. For example: if we calculate the covariances of salary (\$) and age (years), we'll get different covariances which can't be compared because of having unequal scales. To combat such situation, we calculate correlation to get a value between -1 and 1, irrespective of their respective scale.

**Q20. Is it possible capture the correlation between continuous and categorical variable? If yes, how?**

Answer: Yes, we can use ANCOVA (analysis of covariance) technique to capture association between continuous and categorical variables.

**Q21. Both being tree based algorithm, how is random forest different from Gradient boosting algorithm (GBM)?**

**Answer:** The fundamental difference is, random forest uses bagging technique to make predictions. GBM uses boosting techniques to make predictions.

In bagging technique, a data set is divided into n samples using randomized sampling. Then, using a single learning algorithm a model is build on all samples. Later, the resultant predictions are combined using voting or averaging. Bagging is done in parallel. In boosting, after the first round of predictions, the algorithm weighs misclassified predictions higher, such that they can be corrected in the succeeding round. This sequential process of giving higher weights to misclassified predictions continue until a stopping criterion is reached.

Random forest improves model accuracy by reducing variance (mainly). The trees grown are uncorrelated to maximize the decrease in variance. On the other hand, GBM improves accuracy by reducing both bias and variance in a model.

**Q22. Running a binary classification tree algorithm is the easy part. Do you know how does a tree splitting takes place i.e. how does the tree decide which variable to split at the root node and succeeding nodes?**

**Answer:** A classification trees makes decision based on Gini Index and Node Entropy. In simple words, the tree algorithm find the best possible feature which can divide the data set into purest possible children nodes.

Gini index says, if we select two items from a population at random then they must be of same class and probability for this is 1 if population is pure. We can calculate Gini as following:

1. Calculate Gini for sub-nodes, using formula sum of square of probability for success and failure ( $p^2+q^2$ ).
2. Calculate Gini for split using weighted Gini score of each node of that split

Entropy is the measure of impurity as given by (for binary class):

$$\text{Entropy} = -p \log_2 p - q \log_2 q$$

Here p and q is probability of success and failure respectively in that node. Entropy is zero when a node is homogeneous. It is maximum when both the classes are present in a node at 50% – 50%. Lower entropy is desirable.

**Q23. You've built a random forest model with 10000 trees. You got delighted after getting training error as 0.00. But, the validation error is 34.23. What is going on? Haven't you trained your model perfectly?**

**Answer:** The model has overfitted. Training error 0.00 means the classifier has mimiced the training data patterns to an extent, that they are not available in the unseen data. Hence, when this classifier was run on unseen sample, it couldn't find those patterns and returned prediction with higher error. In random forest, it happens when we use larger number of trees than necessary. Hence, to avoid these situation, we should tune number of trees using cross validation.

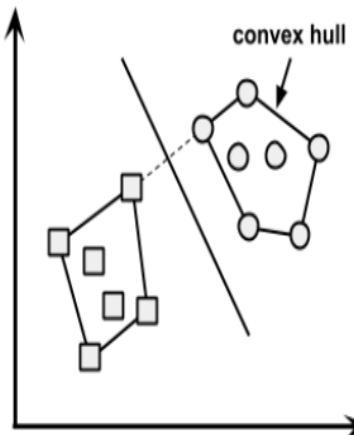
**Q24. You've got a data set to work having  $p$  (no. of variable)  $> n$  (no. of observation). Why is OLS as bad option to work with? Which techniques would be best to use? Why?**

**Answer:** In such high dimensional data sets, we can't use classical regression techniques, since their assumptions tend to fail. When  $p > n$ , we can no longer

calculate a unique least square coefficient estimate, the variances become infinite, so OLS cannot be used at all.

To combat this situation, we can use penalized regression methods like lasso, LARS, ridge which can shrink the coefficients to reduce variance. Precisely, ridge regression works best in situations where the least square estimates have higher variance.

Among other methods include subset regression, forward stepwise regression.



**Q25. What is convex hull ? (Hint: Think SVM)**

**Answer:** In case of linearly separable data, convex hull represents the outer boundaries of the two group of data points. Once convex hull is created, we get maximum margin hyperplane (MMH) as a perpendicular bisector between two convex hulls. MMH is the line which attempts to create greatest separation between two groups.

**Q26. We know that one hot encoding increasing the dimensionality of a data set. But, label encoding doesn't. How ?**

**Answer:** Don't get baffled at this question. It's a simple question asking the difference between the two.

Using one hot encoding, the dimensionality (a.k.a features) in a data set get increased because it creates a new variable for each level present in categorical variables. For example: let's say we have a variable 'color'. The variable has 3 levels namely Red, Blue and Green. One hot encoding 'color' variable will generate three new variables as Color.Red, Color.Blue and Color.Green containing 0 and 1 value.

In label encoding, the levels of a categorical variables gets encoded as 0 and 1, so no new variable is created. Label encoding is majorly used for binary variables.

**Q27. What cross validation technique would you use on time series data set? Is it k-fold or LOOCV?**

**Answer:** Neither.

In time series problem, k fold can be troublesome because there might be some pattern in year 4 or 5 which is not in year 3. Resampling the data set will separate these trends, and we might end up validation on past years, which is incorrect. Instead, we can use forward chaining strategy with 5 fold as shown below:

- fold 1 : training [1], test [2]
- fold 2 : training [1 2], test [3]
- fold 3 : training [1 2 3], test [4]
- fold 4 : training [1 2 3 4], test [5]
- fold 5 : training [1 2 3 4 5], test [6]

where 1,2,3,4,5,6 represents "year".

**Q28. You are given a data set consisting of variables having more than 30% missing values? Let's say, out of 50 variables, 8 variables have missing values higher than 30%. How will you deal with them?**

**Answer:** We can deal with them in the following ways:

1. Assign a unique category to missing values, who knows the missing values might decipher some trend
2. We can remove them blatantly.
3. Or, we can sensibly check their distribution with the target variable, and if found any pattern we'll keep those missing values and assign them a new category while removing others.

**29. ‘People who bought this, also bought...’ recommendations seen on amazon is a result of which algorithm?**

**Answer:** The basic idea for this kind of recommendation engine comes from collaborative filtering.

Collaborative Filtering algorithm considers “User Behavior” for recommending items. They exploit behavior of other users and items in terms of transaction history, ratings, selection and purchase information. Other users behaviour and preferences over the items are used to recommend items to the new users. In this case, features of the items are not known.

**Q30. What do you understand by Type I vs Type II error ?**

**Answer:** Type I error is committed when the null hypothesis is true and we reject it, also known as a ‘False Positive’. Type II error is committed when the null hypothesis is false and we accept it, also known as ‘False Negative’.

In the context of confusion matrix, we can say Type I error occurs when we classify a value as positive (1) when it is actually negative (0). Type II error occurs when we classify a value as negative (0) when it is actually positive(1).

**Q31. You are working on a classification problem. For validation purposes, you've randomly sampled the training data set into train and validation. You are confident that your model will work incredibly well on unseen data since your validation accuracy is high. However, you get shocked after getting poor test accuracy. What went wrong?**

**Answer:** In case of classification problem, we should always use stratified sampling instead of random sampling. A random sampling doesn't takes into consideration the proportion of target classes. On the contrary, stratified sampling helps to maintain the distribution of target variable in the resultant distributed samples also.

**Q32. You have been asked to evaluate a regression model based on  $R^2$ , adjusted  $R^2$  and tolerance. What will be your criteria?**

**Answer:** Tolerance ( $1 / VIF$ ) is used as an indicator of multicollinearity. It is an indicator of percent of variance in a predictor which cannot be accounted by other predictors. Large values of tolerance is desirable.

We will consider adjusted  $R^2$  as opposed to  $R^2$  to evaluate model fit because  $R^2$  increases irrespective of improvement in prediction accuracy as we add more variables. But, adjusted  $R^2$  would only increase if an additional variable improves the accuracy of model, otherwise stays same. It is difficult to commit a general threshold value for adjusted  $R^2$  because it varies between data sets. For example: a gene mutation data set might result in lower adjusted  $R^2$  and still provide fairly good predictions, as compared to a stock market data where lower adjusted  $R^2$  implies that model is not good.

**Q33. In k-means or kNN, we use euclidean distance to calculate the distance between nearest neighbors. Why not manhattan distance ?**

**Answer:** We don't use manhattan distance because it calculates distance horizontally or vertically only. It has dimension restrictions. On the other hand, euclidean metric can be used in any space to calculate distance. Since, the data points can be present in any dimension, euclidean distance is a more viable option.

Example: Think of a chess board, the movement made by a bishop or a rook is calculated by manhattan distance because of their respective vertical & horizontal movements.

**Q34. Explain machine learning to me like a 5 year old.**

**Answer:** It's simple. It's just like how babies learn to walk. Every time they fall down, they learn (unconsciously) & realize that their legs should be straight and not in a bend position. The next time they fall down, they feel pain. They cry.

But, they learn ‘not to stand like that again’. In order to avoid that pain, they try harder. To succeed, they even seek support from the door or wall or anything near them, which helps them stand firm.

This is how a machine works & develops intuition from its environment.

**Q35. I know that a linear regression model is generally evaluated using Adjusted R<sup>2</sup> or F value. How would you evaluate a logistic regression model?**

**Answer:** We can use the following methods:

1. Since logistic regression is used to predict probabilities, we can use AUC-ROC curve along with confusion matrix to determine its performance.
2. Also, the analogous metric of adjusted R<sup>2</sup> in logistic regression is AIC. AIC is the measure of fit which penalizes model for the number of model coefficients. Therefore, we always prefer model with minimum AIC value.
3. Null Deviance indicates the response predicted by a model with nothing but an intercept. Lower the value, better the model. Residual deviance indicates the response predicted by a model on adding independent variables. Lower the value, better the model.

**Q36. Considering the long list of machine learning algorithm, given a data set, how do you decide which one to use?**

**Answer:** You should say, the choice of machine learning algorithm solely depends of the type of data. If you are given a data set which is exhibits linearity, then linear regression would be the best algorithm to use. If you given to work on images, audios, then neural network would help you to build a robust model.

If the data comprises of non linear interactions, then a boosting or bagging algorithm should be the choice. If the business requirement is to build a model which can be deployed, then we’ll use regression or a decision tree model (easy to interpret and explain) instead of black box algorithms like SVM, GBM etc.

In short, there is no one master algorithm for all situations. We must be scrupulous enough to understand which algorithm to use.

**Q37. Do you suggest that treating a categorical variable as continuous variable would result in a better predictive model?**

**Answer:** For better predictions, categorical variable can be considered as a continuous variable only when the variable is ordinal in nature.

**Q38. When does regularization becomes necessary in Machine Learning?**

**Answer:** Regularization becomes necessary when the model begins to overfit / underfit. This technique introduces a cost term for bringing in more features with the objective function. Hence, it tries to push the coefficients for many variables to zero and hence reduce cost term. This helps to reduce model complexity so that the model can become better at predicting (generalizing).

**Q39. What do you understand by Bias Variance trade off?**

**Answer:** The error emerging from any model can be broken down into three components mathematically. Following are these component :

$$Err(x) = \left( E[\hat{f}(x)] - f(x) \right)^2 + E \left[ \hat{f}(x) - E[\hat{f}(x)] \right]^2 + \sigma_e^2$$

$$Err(x) = \text{Bias}^2 + \text{Variance} + \text{Irreducible Error}$$

**Bias error** is useful to quantify how much on an average are the predicted values different from the actual value. A high bias error means we have a under-performing model which keeps on missing important trends. **Variance** on the other side quantifies how are the prediction made on same observation different from each other. A high variance model will over-fit on your training population and perform badly on any observation beyond training.

**Q40. OLS is to linear regression. Maximum likelihood is to logistic regression. Explain the statement.**

**Answer:** OLS and Maximum likelihood are the methods used by the respective regression methods to approximate the unknown parameter (coefficient) value. In simple words,

Ordinary least square(OLS) is a method used in linear regression which approximates the parameters resulting in minimum distance between actual and predicted values. Maximum Likelihood helps in choosing the the values of parameters which maximizes the likelihood that the parameters are most likely to produce observed data.

- What type of data can it deal with?
- Does the model have convergence problems?
- What to do if the model is missing data?
- How interpretable and quick is the model?
- What do you know about EM algorithm?
- Tell us about the common algorithms for supervised and unsupervised machine learning.
- What do you know about KNN and k-means clustering? What's the difference between the two?
- Where is Bayes' theorem applicable?
- Explain Type 1 and Type 2 error.
- Explain ROC and AUC curve.

- How are primary and foreign keys related to SQL?
- How would you gauge the efficacy of an ML model?
- How do you develop a data pipeline?
- How do you deal with missing or corrupted data in a dataset?

## **1. Describe overfitting**

**Answer:** When a model learns the training set too well, it can overfit and start to interpret random oscillations in the training data as concepts. These have an effect on how well the model generalises and don't apply to fresh data.

A model displays 100% accuracy when it is given the training data, which is technically a small loss. However, there could be a mistake and poor performance if we use the test data. This condition is known as overfitting.

Several measures can be taken to avoid overfitting:

- Making a less complex model
- Apply cross-validation methods
- Regularization

## **2. How do you pick a classifier considering the size of a training set?**

**Answer:** A model with a right bias and low variance appears to perform better when the training set is small because they are less prone to overfit.

## **3. How Are Machine Learning Models Built? What Are the Three Steps?**

**Answer:** Building a machine learning model involves the following three steps:

- Building a model

Choosing a suitable algorithm for the model, then developing it to conform to the specifications.

- Model Validation

The model's accuracy can be evaluated using the test data.

- Use of the Model

Apply the resulting model to tasks in the actual world after making the necessary adjustments during testing.

It's important to remember that the model needs to be evaluated frequently to ensure that it is functioning effectively in this situation. To ensure that it is current, it should be changed.

## 4. Explain Deep Learning

**Answer:** Deep learning is a kind of machine learning that uses artificial neural networks to create systems that think and learn like people. The term "deep" refers to neural networks that might include more than one layer.

The manual process of feature engineering in machine learning is one of the key distinctions between both. The neural network model for deep learning will select the right features on its own (and which not to use).

### 1. Exactly how is a decision tree pruned?

**Answer:** Pruning is the process of removing branches from decision trees that have poor predictive power, which lowers the complexity of the model and improves forecast accuracy. You replace each node when you prune it, and you keep pruning until the predicted accuracy decreases.

### 2. What distinguishes the training set from the test set? Why do we just divide based on the dependent variable?

**Answer:** A subset of your data called the "training set" is used by your model to practise predicting the dependent variable using the independent variables. The test set, which is a complementary subset of the training set, is the basis for assessing your model's ability to accurately predict the dependent variable given the independent variables.

We split on the dependent variable because we want the values of the dependent variable to be evenly distributed between the training set and the test set. For

instance, our model wouldn't be able to predict the future if the dependant variable in the training set had only the same value.

### **3. Describe some of the pre-processing methods used in Python to get the data ready**

**Answer:** Mean removal: This feature entails taking the mean out of each feature and centering it on zero. The bias from the features is removed using mean removal.

Feature scaling: Each feature's value within a data point may range between two random values. Scaling them is crucial to ensure that they comply with the established rules.

Normalization is the process of altering the feature vector's values to put them on a similar scale. Here, a feature vector's values are changed so that they add up to 1.

A numerical feature vector is binarized into a Boolean vector using this technique.

### **4. What is PCA used for?**

**Answer:** A dimensionality-reduction approach called PCA breaks down data into primary components (PC) using transforms. A set of observations of potentially correlated variables (entities that each take on different numerical values) are transformed using an orthogonal transformation into a set of values of linearly uncorrelated variables known as principal components.

### **5. Tell us about the purposes of variable selection**

**Answer:** Three things are the three goals of variable selection:

**Answer:** Enhancing the predictors' ability to make accurate predictions, offering quicker and more affordable predictors, and offering a clearer understanding of the underlying process that produced the data.

## **6. Share your ideas about recall and precision**

**Answer:** Recall, also referred to as the true positive rate, is the ratio between the number of positives your model predicts and the actual number of positives present over the entire set of data.

Precision is a measurement of the number of precise positives your model claims versus the number of positives it actually claims. It is also referred to as the positive predictive value.

1. **(Robinhood)** What is user churn, and how can you build a model to predict whether a user will churn? What features would you include in the model and how do you assess importance?

2. **(Affirm)** Assume we have a classifier that produces a score between 0 and 1 for the probability of a particular loan application being fraudulent. In this scenario: a) what are false positives, b) what are false negatives, and c) what are the trade-offs between them in terms of dollars and how should the model be weighted accordingly?

3. **(Uber)** Say you need to produce a binary classifier for fraud detection. What metrics would you look at, how is each defined, and what is the interpretation of each one?

4. **(Google)** Say you are given a very large corpus of words. How would you identify synonyms?

5. **(Airbnb)** Say you are running a simple logistic regression on a problem but find the results to be unsatisfactory. What are some ways you might improve your model or what other models might you look into?

6. (**Amazon**) Describe both generative and discriminative models and give an example of each?

7. (**Affirm**) Assume we have some credit model, which has accurately calibrated (up to some error) score of how credit-worthy any individual person is. For example, if the model's estimate is 92% then we can assume the actual score is between 91 and 93. If we take 92 as a score cutoff and deem everyone above that score as credit-worthy, are we over-estimating or underestimating the actual population's credit score?

8. (**Microsoft**) What is the bias-variance tradeoff? How is it expressed using an equation?

9. (**Uber**) Define the cross validation process. What is the motivation behind using it?

10. (**Airbnb**) Say you are modeling the yearly revenue of new listings. What kinds of features would you use? What data processing steps need to be taken, and what kind of model would run?

11. (**Stitch Fix**) How would you build a model to calculate a customer's propensity to buy a particular item? What are some pros and cons of your approach?

12. (**Uber**) What is L1 and L2 regularization? What are the differences between the two?

13. (**Amazon**) Define what it means for a function to be convex. What is an example of a machine learning algorithm that is not convex and describe why that is so?

14. (**Affirm**) Assume we have a classifier that produces a score between 0 and 1 for the probability of a particular loan application being a fraud. Say that for

each application's score, we take the square root of that score. How would the ROC curve change? If it doesn't change, what kinds of functions would change the curve?

15. (**Amazon**) Describe gradient descent and the motivations behind stochastic gradient descent?

16. (**Microsoft**) Explain what Information Gain and Entropy are in a Decision Tree?

17. (**Airbnb**) Say you are tasked with producing a model that can recommend similar listings to an Airbnb user when they are looking at any given listing. What kind of model would you use, what data is needed for that model, and how would you evaluate the model?

## 12 Hard ML Interview Questions

18. (**Microsoft**) Describe the idea behind boosting. Give an example of one method and describe one advantage and disadvantage it has?

19. (**Google**) Say we are running a probabilistic linear regression which does a good job modeling the underlying relationship between some  $y$  and  $x$ . Now assume all inputs have some noise  $\epsilon$  added, which is independent of the training data. What is the new objective function? How do you compute it?

20. (**Netflix**) What is the loss function used in k-means clustering for  $k$  clusters and  $n$  sample points? Compute the update formula using 1) batch gradient descent, 2) stochastic gradient descent for the cluster mean for cluster  $k$  using a learning rate  $\epsilon$ .

21. (**Tesla**) You're working with several sensors that are designed to predict a particular energy consumption metric on a vehicle. Using the outputs of the sensors, you build a linear regression model to make the prediction. There are many sensors, and several of the sensors are prone to complete failure. What are

some cost functions you might consider, and which would you decide to minimize in this scenario?

22. (**Stripe**) Say we are using a Gaussian Mixture Model (GMM) for anomaly detection on fraudulent transactions to classify incoming transactions into K classes. Describe the model setup formulaically and how to evaluate the posterior probabilities and log likelihood. How can we determine if a new transaction should be deemed fraudulent?

23. (**Netflix**) What is Expectation-Maximization and when is it useful? Describe the setup algorithmically with formulas.

24. (**Opendoor**) Describe the setup and assumptions of using linear discriminant analysis (LDA). Show mathematically that the decision boundaries are linear.

25. (**Microsoft**) Formulate the background behind an SVM, and show the optimization problem it aims to solve.

26. (**Netflix**) Describe entropy in the context of machine learning, and show mathematically how to maximize it assuming N states.

27. (**Airbnb**) Suppose you are running a linear regression and model the error terms as being normally distributed. Show that in this setup, maximizing the likelihood of the data is equivalent to minimizing the sum of squared residuals.

28. (**Stripe**) Describe the model formulation behind logistic regression. How do you maximize the log-likelihood of a given model (using the two-class case)?

29. (**Netflix**) Say X is a univariate Gaussian random variable. What is the entropy of X?

30. (**Uber**) Describe the idea behind PCA and go through its formulation and

derivation in matrix form. Go through the procedural description and solve the constrained maximization.

### **Problem #3 Solution:**

Some main important ones are precision, recall, and the AUC of the ROC curve. Let us define TP as a true positive, FP as a false positive, TN as a true negative, and FN as a false negative.

Precision is defined by  $TP / (TP + FP)$ . It answers the question “what percent of fraudulent predictions were correct?” and is important to maximize since you want your classifier to be as correct as possible when it identifies a transaction as fraudulent.

Recall is defined by  $TP / (TP + FN)$ . It answers the question “what percent of fraudulent cases were caught?” and is important to maximize since you want your classifier to have caught as many of the fraudulent cases as possible.

AUC of the ROC curve is defined by the area under an ROC curve, which is constructed by plotting the true positive rate,  $TP / (TP + FN)$  versus the false positive rate,  $FP / (FP + TN)$  for various thresholds, which determines the label of fraudulent or not fraudulent. The area under this curve, or AUC, is a measure of separability of the model, and the closer to 1 it is, the higher the measure. It answers the question “is my classifier able to discriminate between fraud and not-fraud effectively” and is important to maximize since you want your classifier to separate the two classes accordingly.

### **Problem #8 Solution:**

The equation this tradeoff is expressed in is given by the following: Total model error = Bias + Variance + Irreducible error. Flexible models have low bias and high variance, whereas more rigid models have high bias and low variance.

The bias term comes from the error when a model is under-fitting data. Having a high bias means that the machine learning model is too simple and does not capture the relationship between the features and the target. An example would be using linear regression when the underlying relationship is nonlinear.

The variance term comes from the error when a model is overfitting data. Having a high variance means that a model is susceptible to changes in training data, which means it is capturing too much noise. An example would be using a

very complex neural network where the true underlying relationship between the features and the target is linear.

The irreducible term comes from the error that cannot be addressed directly by the model, such as from the noise in measurements of data.

### **Problem #12 Solution:**

L1 and L2 regularization are both methods of regularization that attempt to prevent overfitting in machine learning. For a regular regression model assume the loss function is given by  $L$ . L1 adds the absolute value of the coefficients as a penalty term, whereas L2 adds the squared magnitude of the coefficients as a penalty term.

The loss function for the two are:

$$Loss(L_1) = L + \lambda|w_i|$$

$$Loss(L_2) = L + \lambda|w_i^2|$$

Where the loss function  $L$  is the sum of errors squared, given by the following, where  $f(x)$  is the model of interest, for example, linear regression with  $p$  predictors:

$$L = \sum_{i=1}^n (y_i - f(x_i))^2 = \sum_{i=1}^n (y_i - \sum_{j=1}^p (x_{ij}w_j))^2 \text{ for linear regression}$$

If we run gradient descent on the weights  $w$ , we find that L1 regularization will force any weight closer to 0, irrespective of its magnitude, whereas, for the L2 regularization, the rate at which the weight goes towards 0 becomes slower as the rate goes towards 0. Because of this, L1 is more likely to “zero” out particular weights, and hence removing certain features from the model completely, leading to more sparse models.

### **Problem #15 Solution:**

Gradient descent is an algorithm that takes small steps in the direction of steepest descent for a particular objective function. Say we have the function  $f()$  and are currently at some point  $x$  at time  $t$ . Then gradient descent will update  $x$  as follows until convergence:

$$x^{t+1} = x^t - \alpha_t \nabla f(x^t)$$

that is, we calculate the negative of the gradient of  $f$  and scale that by some constant, and move in that direction each iteration.

Since many loss functions are decomposable into the sum of individual functions, then the gradient step overall can be broken down into adding separate gradients. However, for very large datasets this can be computationally intensive, and the algorithm might get stuck at local minima or saddle points.

Therefore, we can use stochastic gradient descent (SGD) in which we get an unbiased estimate of the true gradient without going through all data points by uniformly selecting a point at random and doing a gradient update there.

The estimate is unbiased since we have:

$$\nabla f(x) = \frac{1}{n} \sum_{i=1}^n \nabla f_i(x)$$

and since data is assumed to be i.i.d, then for the SGD  $g(x)$  in expectation:

$$E[g(x)] = \nabla f(x)$$

19. Recall the objective function for linear regression where  $x$  is set of input vectors and  $w$  are the weights:

$$L(w) = E[(w^T x - y)^2]$$

Assume that the noise added is Gaussian as follows:

$$\epsilon \sim N(0, \lambda I)$$

Then the new objective function is given by:

$$L(w) = E[(w^T (x + \epsilon) - y)^2]$$

To compute it, we simplify:

$$L'(w) = E[(w^T x - y + w^T \epsilon)^2]$$

$$L'(w) = E[(w^T x - y)^2 + 2(w^T x - y)w^T \epsilon + w^T \epsilon \epsilon^T w]$$

$$L'(w) = E[(w^T x - y)^2] + E[2(w^T x - y)w^T \epsilon] + E[w^T \epsilon \epsilon^T w]$$

We know that the expectation for  $\epsilon$  is 0 so the middle term becomes 0 and we are left with:

$$L'(w) = L(w) + 0 + w^T E[\epsilon \epsilon^T]w$$

The last term can be simplified as:

$$L'(w) = L(w) + w^T \lambda I w$$

And therefore the objective function simplifies to that of L2-regularization:

$$L'(w) = L(w) + \lambda ||w||^2$$

### Problem #21 Solution:

There are two potential cost functions here, one using the L1 norm and the other using the L2 norm. Below are two basic cost functions using an L1 and L2 norm respectively:

$$J(w) = ||Xw - y||$$

$$J(w) = |Xw - y|^2$$

It would be more sensible to use an L1 norm in this case since the L1 norm penalizes the outliers harder and thus gives less weight to the complete failures than the L2 norm does.

Additionally, it would be prudent to involve a regularization term to account for noise. If we assume that the noise added to each sensor uniformly as follows:

$$\epsilon \sim N(0, \lambda I)$$

then using traditional L2 regularization, we would have the cost function:

$$J(w) = ||Xw - y|| + \lambda ||w||^2$$

However, given the fact that there are many sensors (and a broad range of how useful they are), we could instead assume that noise is added by:

$$\epsilon \sim N(0, \lambda D)$$

where each diagonal term in the matrix D represents the error term used for each sensor (and hence penalizing certain sensors more than others). Then our final cost function is given by:

$$J(w) = ||Xw - y|| + \lambda w^T D w$$

### Problem #25 Solution:

The goal of an SVM is to form a hyperplane that linearly separates the given training data. Specifically it aims to maximize the margin, which is the minimum distance from the decision boundary to any training point. The points closest to the hyperplane are called the support vectors.

Mathematically, the hyperplane is given by the following, for some constant c:

$$H = \{h: w^T h = c\}$$

Now consider some arbitrary point  $x_i$  that is not on the hyperplane. The distance from  $x_i$  to H is the length of the projection from  $x_i - h$  to the vector perpendicular to H:

$$d = \frac{|w^T(x_i - h)|}{\|w\|_2} = \frac{|w^T x_i - c|}{\|w\|_2}$$

To get the actual classification signs (positive vs. negative), we can multiply this distance by the sign on each  $y_i$ :

$$y_i * \frac{(w^T x_i - c)}{\|w\|_2}$$

Assuming a margin of size m, then the optimization problem is to:

$$\max m \text{ s.t. } y_i * \frac{(w^T x_i - c)}{\|w\|_2} \geq m$$

To ensure uniqueness we can set a constraint on m:

$$m = \frac{1}{\|w\|_2}$$

And therefore the final optimization problem is:

$$\max \frac{1}{\|w\|_2} \text{ s.t. } y_i * (w^T x_i - c) \geq 1$$

### Problem #29 Solution:

We have:

$$X \sim N(\mu, \sigma^2)$$

and entropy for a continuous random variable is given by:

$$\text{H}(x) = -\int_{-\infty}^{\infty} p(x) \log p(x) dx$$

For a Gaussian we have:

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Plugging into the above yields:

$$H(x) = - \int_{-\infty}^{\infty} p(x) \log \sigma\sqrt{2\pi} dx - \int_{-\infty}^{\infty} p(x) \left(-\frac{(x-\mu)^2}{2\sigma^2}\right) dx$$

The first term is equal to

$$-\log \sigma\sqrt{2\pi} \int_{-\infty}^{\infty} p(x) dx = -\log \sigma\sqrt{2\pi}$$

since the integral evaluates to 1 (by definition of probability density). The second term is given by:

$$\frac{1}{2\sigma^2} \int_{-\infty}^{\infty} p(x)(x-\mu)^2 dx = \frac{\sigma^2}{2\sigma^2} = \frac{1}{2}$$

since the inner term is the expression for the variance. Therefore the entropy is:

$$H(x) = \frac{1}{2} - \log \sigma\sqrt{2\pi}$$

**Q7.** What is the difference between supervised learning and unsupervised learning?

**Answer:** “Supervised learning is a type of machine learning algorithm that uses labeled datasets. It trains models to predict the output of new data. Typically, the labels for these data are provided by a human supervisor. The data are then used to teach the model what the correct output should be for each input.”

Meanwhile, unsupervised learning is a type of machine learning algorithm that does not use any labels (or supervision) to train the model. Instead, it uses only unlabelled input data. The model then learns the underlying structure of the data to make predictions about new inputs.”

Q8. Can you explain the difference between KNN and k-means clustering?

**Answer:** “KNN (or K Nearest Neighbors) is a non-parametric algorithm used for classification and regression. KNN is a supervised learning model. KNN works by finding the distance between a new point and all labeled training points. The training point with the smallest distance is then used to predict the label of the new input. This point is then assigned to the class that has the most training points within the vicinity.

Meanwhile, k-means clustering is an unsupervised learning model that groups data points together based on their similarity. In this case, the algorithm measures similarity by the distance between data points, rather than by using labeled training data. Finally, the data points are clustered together such that the points within a cluster are more similar to each other than those in other clusters.

While these two algorithms are similar, the key difference is that KNN uses labeled data and k-means clustering does not.”

Q9. What is a decision tree?

**Answer:** “In machine learning, a decision tree is an algorithm that splits datasets into smaller and smaller subsets. It is commonly used to help choose between alternative options or to determine the optimal path. The decision tree has both decision nodes and leaf nodes. The decision nodes are where the tree splits and the leaf nodes are the final decisions. Decision trees are particularly useful for handling non-linear data sets.”

Q10. What is a random forest?

**Answer:** When answering a question like this (or the previous questions) the obvious priority is explaining the concept clearly. However, when appropriate, you can also expand on the different terminology. Doing so is a great way of showing that you know how to explain things in clear, non-technical terms. For instance:

“A random forest is a type of ensemble model composed of numerous decision trees. An ensemble model like this runs several related but different analytical models and then synthesizes the results. In this case, the individual decision trees in the forest are trained on different subsets of the data. The subsequent predictions of individual trees are then combined to produce the final prediction. An advantage of using a random forest over a single decision tree is that it is much less likely to overfit the data.”

Q11. Can you explain logistic regression?

**Answer:** “Logistic regression is a type of statistical analysis used to predict the probability of an outcome occurring. The outcome is always binary, meaning it can only be one of two things, such as yes or no, success or failure, etc. Logistic regression models use a formula to calculate the probability that the outcome will occur based on certain input variables.

The model then uses this probability to predict whether the outcome will happen or not. As a binary model, the output is either a 0 or 1. Values above 0.5 are considered 1, and values below are considered 0

Q12. What is Bayes Theorem and how does it apply to machine learning?

**Answer:** “Bayes theorem is a way of calculating the probability of something happening, given that something else has already occurred. In short, it provides the conditional probability of an event based on the values of specific, related, known probabilities.

For instance, if you know that there is an 85% chance of rain in the morning, and you also know that when it rains, there is a 95% chance that the sun will not be out, you can use Bayes Theorem to calculate the probability that the sun will not be out tomorrow morning. Within machine learning, a classification algorithm known as a Naïve Bayes classifier (which is a simplified version of the Theorem) can be used to classify data into various classes, quickly and with high accuracy.

That said, Naïve Bayes classifiers tend to make the strong assumption that the features in a dataset are independent of each other, which is not usually true in real-world datasets.”

Q13. What is the F1 score?

**Answer:** The ability to measure the success of a machine learning algorithm is as important as it is for any programming task. As such, you will likely get questions about the different evaluation metrics you can use (including recall, precision, and the F1 score). Before your interview, you should aim to read up on all success metrics, but in this case, your answer might be:

“The F1 score measures how well a machine learning classifier (or class labeling algorithm) performs. It takes into account both the precision and recall of the classifier. The score is then calculated by taking the harmonic mean of the precision and recall. The precision is the number of true positives divided by the sum of the true positives and the false positives. The recall is the number of true positives divided by the sum of the true positives and the false negatives.”

Q14. What big data tools have you used?

**Answer:** While all data analysts use data management tools at some point, make sure you consider the context here. Specifically, what tools have you used—or are at least familiar with—that are common in a machine learning

setting? Common tools used for machine learning include big data tools, like Apache Hadoop, Apache Spark, and NoSQL databases. These tools, used for distributed computing, are necessary for managing big data and real-time web applications. Apache Spark is arguably the most popular right now.

Spark is a powerful open-source processing engine built for speed, ease of use, and sophisticated analytics. It's used for various machine learning tasks, such as classification, regression, clustering, and dimensionality reduction. If you've never used any of these tools, be honest. But try to familiarize yourself with them before the interview, so at least you don't have to give the interviewer a blank expression if they ask you!

Q15. Which programming language would you favor for machine learning?

**Answer:** This could either be a trick question (answer: depends on the task!) or a genuine query to see which programming language you're most comfortable using. Either way, a frank assessment of the options might be your best bet. You could say something like:

“Both Python and R have advantages and disadvantages when used on machine learning tasks. Some people may prefer Python because it is a more general-purpose programming language and has countless libraries that make these tasks easier. However, others prefer R for its power in statistical computing and because it is a lower-level language. It's also more widely used by statisticians and data scientists.

My personal preference, however, is Python, although Java is also very robust and has better error-checking than either of the other two. Like Python, Java also has a large and active community, which makes it easy to find help and resources.”

Q16. How would you compare CSVs with XML and JSON?

**Answer:** CSV, XML, and JSON are all common file formats used by data scientists, analysts, and machine learning engineers. Each has different features and this question is testing your knowledge of these. Your answer might be:

“Generally speaking, CSV is much simpler than XML, both in terms of its syntax and its structure, using commas to separate data into columns.

Programmatically, this makes CSV files far easier to work with. It’s also worth noting that they’re usually smaller than XML files, which makes CSVs easier to download and parse.

However, XML can be used to preserve data formatting in ways that CSVs cannot. XML also supports hierarchical data. Meanwhile, JSON combines the best of both CSV and XML: it remains compact like CSV (typically JSON files are only about twice as large as similar CSVs) while also supporting hierarchical data like XML. On the downside, JSON’s data structure is not as robust as XML.”

Q17. What would be your approach to developing a data pipeline?

**Answer:** All data analysts and machine learning engineers need to produce data pipelines. In this question, you should talk the interviewer through the process, including which tools you might use. These might include things like Apache NiFi, Apache Kafka, and Apache Flume.

You should also consider the data sources you might be working with, identify the data transformations that need to be performed, how you would design the architecture of the pipeline, and how you would test and deploy it. Cover each of these bases and you shouldn’t go too far wrong.

Q18. Which data visualization libraries and tools do you use most?

**Answer:** This is another question that will depend on your preferences. Python, in particular, has a wide array of fantastic, open-source data viz libraries available on the Python Package Index. Check the job description before the

interview, though, to see if they mention any specific tools that they use. Otherwise, play around with Python libraries like Matplotlib or Seaborn to get a feel for them.

Meanwhile, if you're an R user, ggplot2 is popular. Finally, there are also lots of proprietary **data visualization software** out there. This includes Tableau, Power BI, and Qlikview.

**Q19.** How would you manage the issue of missing data in a dataset?

**Answer:** The answer to this question might benefit from an explanation of the tools and commands you'd use to fix data corruption issues. Broadly speaking, though, you might want to start by talking the interviewer through the different options:

"There are a few ways to manage missing data, depending on the amount and type. If only a small amount of data is missing, you can simply delete the relevant rows or columns. Of course, this is only an option if the amount of missing data is small and if the data is unimportant for analysis.

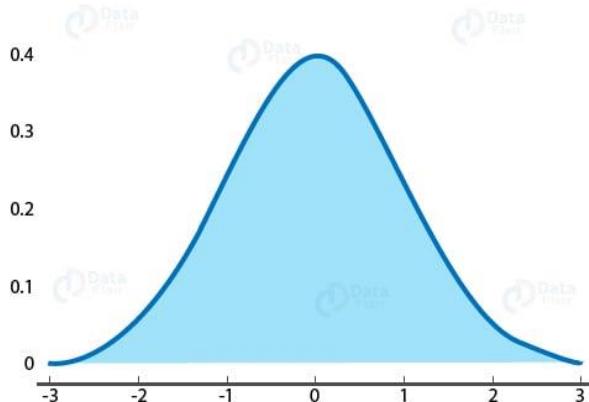
If a large amount of data is missing—which may be more common in vast machine learning datasets—another option is to impute the missing data by substituting the missing values with estimated ones. I might do this by using the mean or median of the data, or by applying a regression model to predict the missing values.

If the data is missing completely at random, I might use multiple imputations to estimate the missing values. This is a more sophisticated method that uses statistical techniques. There is also the option of creating an 'unknown' category for missing values.

**Q.1 What do you understand by the term Normal Distribution?**



## Normal Distribution



**Normal Distribution** is also known as Gaussian Distribution. It is a type of probability distribution that is symmetric about the mean. It shows that the data is closer to the mean and the frequency of occurrences in data are far from the mean.

### Q.2 How will you explain linear regression to a non-tech person?

Linear Regression is a statistical technique of measuring the linear relationship between the two variables. By linear relationship, we mean that an increase in a variable would lead to increase in the other variable and a decrease in one variable would lead to attenuation in the second variable as well. Based on this linear relationship, we establish a model that predicts the future outcomes based on an increase in one variable.

### Q.3 How will you handle missing values in data?

There are several ways to handle missing values in the given data-

- Dropping the values
- Deleting the observation (not always recommended).
- Replacing value with the mean, median and mode of the observation.
- Predicting value with regression
- Finding appropriate value with clustering

### Q.4 How will you verify if the items present in list A are present in series B?

We will use the `isin()` function. For this, we create two series `s1` and `s2` –

```
s1 = pd.Series([1, 2, 3, 4, 5])
s2 = pd.Series([4, 5, 6, 7, 8])
s1[s1.isin(s2)]
```

### Q.5 How to find the positions of numbers that are multiples of 4 from a series?

For finding the multiples of 4, we will use the `argwhere()` function. First, we will create a list of 10 numbers –

```
s1 = pd.Series([1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
np.argwhere(ser % 4==0)
```

### Output > [3], [7]

### Q.6 How are KNN and K-means clustering different?

Firstly, KNN is a supervised learning algorithm. In order to train this algorithm, we require labeled data. K-means is an unsupervised learning algorithm that looks for patterns that are intrinsic to the data. The K in KNN is the number of nearest data points. On the contrary, the K in K-means specify the number of centroids.

### Q.7 Can you stack two series horizontally? If so, how?

Yes, we can stack the two series horizontally using concat() function and setting axis = 1.

```
df = pd.concat([s1, s2], axis=1)
```

### Q.8 How can you convert date-strings to timeseries in a series?

#### Input:

```
s = pd.Series(['02 Feb 2011', '02-02-2013', '20160104', '2011/01/04', '2014-12-05', '2010-06-06T12:05'])
```

To solve this, we will use the to\_datetime() function.

```
pd.to_datetime(s)
```

### Q.9 Python or R – Which one would you prefer for text analytics?

Difference Between R and Python		
Features	R	Python
Scope	Used mainly for statistical modeling	Used for a variety of purposes like web-application development and data analysis
Used By	Statisticians, Analyst & Data Scientist	Developer, Data Engineers & Data Scientist
Suitable For	People with no prior experience in programming	Newbies to experienced IT professionals
Package Distribution	CRAN	PyPi
Visualization Tools	ggplot2, plotly, ggiraph	Matplotlib, bokkeh, seaborn

Both Python and R provide robust functionalities for working with text data. R provides extensive text analytics libraries but its data mining libraries are still in a nascent stage. Python is best suited for enterprise level and for increasing software productivity. For handling unstructured data, R provides a vast variety

of support packages. Python is best apt at handling colossal data while R has memory constraints and is slower in response to large data. Therefore, the preference for using Python or R depends on the area of functionality and usage.

### **Q.10 Explain ROC curve.**

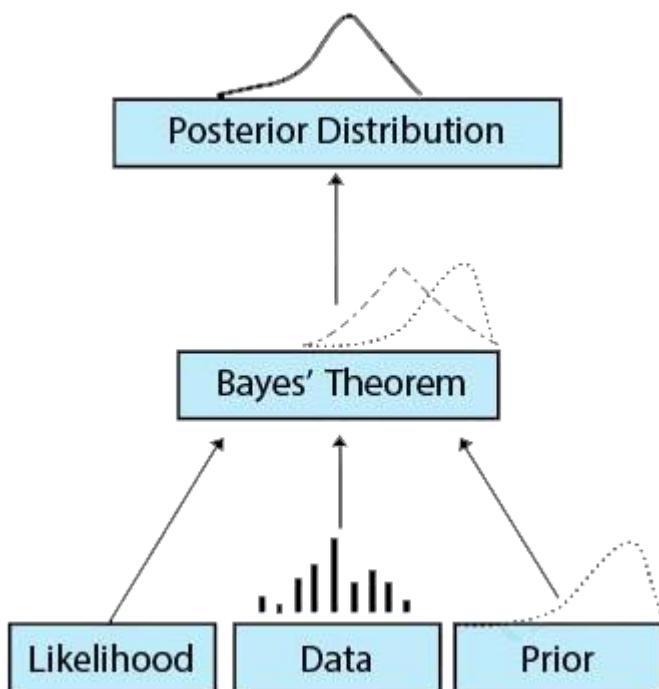
Receiver Operating Characteristic is a measurement of the True Positive Rate (TPR) against False Positive Rate (FPR). We calculate True Positive (TP) as  $TPR = TP / (TP + FN)$ . On the contrary, false positive rate is determined as  $FPR = FP / (FP + TN)$  where where TP = true positive, TN = true negative, FP = false positive, FN = false negative.

### **Q.11 How is AUC different from ROC?**

AUC curve is a measurement of precision against the recall. Precision =  $TP / (TP + FP)$  and  $TP / (TP + FN)$ . This is in contrast with ROC that measures and plots True Positive against False positive rate.

### **Q.12 Why is Naive Bayes referred to as Naive?**

Ans. In *Naive Bayes*, the assumptions and probabilities that are computed of the features are independent of each other. It is the assumption of feature independence that makes Naive Bayes, “Naive”.



### **Q.13 How will you create a series from a given list in Pandas?**

We will the list to the Series() function.

```
ser1 = pd.Series(mylist)
```

#### **Q.14 Explain bias, variance tradeoff.**

Bias leads to a phenomenon called underfitting. This is caused by the introduction of error due to the oversimplification of the model. On the contrary, variance occurs due to complexity in the machine learning algorithm. In variance, the model also learns noise and other distortions that affect the overall performance of it. If you increase the complexity of your model, then the error will go down due to reduction in bias. However, after a certain point, the error will increase due to increasing complexity and addition of noise. This is known as bias-variance tradeoff. A good machine learning algorithm should possess low bias and low variance.

#### **Q.15 What is a confusion matrix?**

A confusion matrix is a table that delineates the performance of a supervised learning algorithm. It provides a summary of prediction results on a classification problem. With the help of confusion matrix, you can not only find the errors made by the predictor but also the type of errors.



### Type I and Type II Errors

	Actually Pregnant	Actually Not Pregnant
Predicted Pregnant	 True Positive(TP)	 False Positive(FP)
Predicted Not Pregnant	 False Negative(FN)	 True Negative(TN)

### Confusion Matrix

#### **Q.16 What is SVM? Can you name some kernels used in SVM?**

**SVM stands for support vector machine.** They are used for classification and prediction tasks. SVM consists of a separating plane that discriminates between the two classes of variables. This separating plane is known as hyperplane.

Some of the kernels used in SVM are –

- Polynomial Kernel

- Gaussian Kernel
- Laplace RBF Kernel
- Sigmoid Kernel
- Hyperbolic Kernel

### **Q.17 How is Deep Learning different from Machine Learning?**

Deep Learning is an extension of Machine Learning. It is a special area within ML that is about developing algorithms that simulate human nervous system. Deep Learning involves neural networks which are trained over large datasets to understand the patterns and then perform classification and prediction.

<b>Deep Learning Vs Machine Learning</b>		
<b>Factors</b>	<b>Deep Learning</b>	<b>Machine Learning</b>
Data Requirement	Requires large data	Can train on lesser data
Accuracy	Provides high accuracy	Gives lesser accuracy
Training Time	Takes longer to train	Takes less time to train
Hardware Dependency	Requires GPU to train properly	Trains on CPU
Hyperparameter Tuning	Can be tuned in various different ways.	Limited tuning capabilities

### **Q.18 How can you compute significance using p-value?**

After a hypothesis test is conducted, we compute the significance of the results. The p-value is present between 0 and 1. If the p-value is less than 0.05, then it means that we cannot reject the null hypothesis. However, if it is greater than 0.05, then we reject the null hypothesis.

### **Q.19 Why don't gradient descent methods always converge to the same point?**

This is because, in some cases, they reach to local or local optima point. The methods don't always achieve global minima. This is also dependent on the data, the descent rate and origin point of descent.

### **Q.20 Explain A/B testing.**

To perform a hypothesis testing of a randomized experiment with two variables A and B, we make use of A/B testing. A/B testing is used to optimize web-pages based on user preferences where small changes are added to web-pages that are delivered to a sample of users. Based on their reaction to the web-page and reaction of the rest of the audience to the original page, we can carry out this statistical experiment.

### **Q.21 What is box cox transformation?**

In order to transform the response variable so that the data meets its required assumptions, we make use of Box Cox Transformation. With the help of this technique, we can transform non-normal dependent variables into normal shapes. We can apply a broader number of tests with the help of this transformation.

**Q.22 What is meant by ‘curse of dimensionality’? How can we solve it?**

While analyzing the dataset, there are instances where the number of variables or columns are in excess. However, we are required to only extract significant variables from the group. For example, consider that there are a thousand features. However, we only need to extract handful of significant features. This problem of having numerous features where we only need a few is called ‘curse of dimensionality’.

There are various algorithms for dimensionality reduction like PCA (Principal Component Analysis).

**Q.23 What is the difference between recall and precision?**

Recall is the fraction of instances that have been classified as true. On the contrary, precision is a measure of weighing instances that are actually true. While recall is an approximation, precision is a true value that represents factual knowledge.

**Q.24 What is pickle module in Python?**

For serializing and de-serializing an object in Python, we make use of pickle module. In order to save this object on drive, we make use of pickle. It converts an object structure into character stream.

**Q.25 What are the different forms of joins in a table?**

Some of the different joins in a table are –

- Inner Join
- Left Join
- Outer Join
- Full Join
- Self Join
- Cartesian Join

**Q.26 List differences between DELETE and TRUNCATE commands.**

DELETE command is used in conjunction with WHERE clause to delete some rows from the table. This action can be rolled back.

However, TRUNCATE is used to delete all the rows of a table and this action cannot be rolled back.

**Q.27 Can you tell some clauses used in SQL?**

Some of the commonly used *clauses in SQL* are –

- WHERE
- GROUP BY
- ORDER BY
- USING

### **Q.28 How will you get second highest salary of an employee emp from employee\_table?**

In order to get the second highest salary of an employee, we will use the following query –

```
SELECT TOP 1 salary
FROM(
SELECT TOP 2 salary
FROM employee_table
ORDER BY salary DESC) AS emp
ORDER BY salary ASC;
```

### **Q.29 What is a foreign key?**

A foreign key is a special key that belongs to one table and can be used as a primary key of another table. In order to create a relationship between the two tables, we reference the foreign key with the primary key of the other table.

### **Q.30 What do you mean by Data Integrity?**

With data integrity, we can define the accuracy as well as the consistency of the data. This integrity is to be ensured over the entire life-cycle.

### **Q.31 How is SQL different from NoSQL?**

SQL deals with *Relational Database Management Systems* or RDBMS. This type of database stores structured data that is organized in rows and columns, that is, in a table. However, NoSQL is a query language that deals with Non-Relational Database Management Systems. The data present here is unstructured. Structured data is mostly generated from services, gadgets and software systems. However, unstructured data, which is increasing day by day, is generated from users directly.

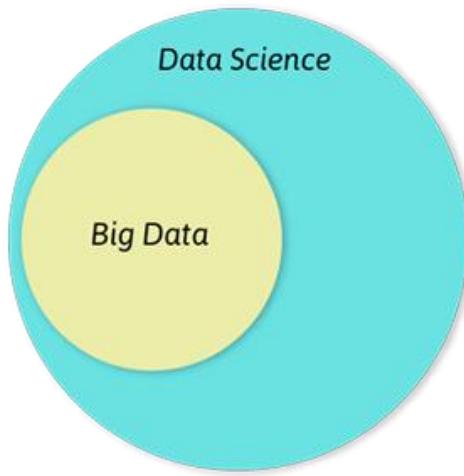
### **Q.32 Can you tell me about some NoSQL databases?**

Some of the popular NoSQL databases are Redis, MongoDB, Cassandra, HBase, Neo4j etc.

### **Q.33 How is Hadoop used in Data Science?**

Hadoop provides the data scientists the ability to deal with large scale unstructured data. Furthermore, various new extensions of Hadoop like Mahout and PIG provide various features to analyze and implement machine learning algorithms on large scale data. This makes Hadoop a comprehensive system that

is capable of handling all forms of data, making it an ideal suite for data scientists.



#### **Q.34 How can you select an ideal value of K for K-means clustering?**

There are several methods like the elbow method and kernel method to find the number of centroids in the given cluster. However, to ascertain an approximate number of centroids quickly, we can also take the square root of the number of data points divided by two. While this technique is not entirely accurate but is fast as compared to the previously mentioned techniques.

#### **Q.35 Define underfitting and overfitting.**

Most statistics and ML projects need to fit a model on training data to be able to create predictions. There can be two problems while fitting a model- overfitting and underfitting.

- Overfitting is when a model has random error/noise and not the expected relationship. If a model has a large number of parameters or is too complex, there can be overfitting. This leads to bad performance because minor changes to training data highly changes the model's result.
- Underfitting is when a model is not able to understand the trends in the data. This can happen if you try to fit a linear model to non-linear data. This also results in bad performance.

#### **Q.36 What are univariate, bivariate and multivariate analysis?**

Three types of analysis are univariate, bivariate and multivariate.

- Univariate analysis includes descriptive statistical analysis techniques which you can differentiate on the basis of how many variables are involved. Some pie charts can have a single variable.
- Bivariate analysis explains the difference between two variables at one time. This can be analyzing sale volume and spending volume using a scatterplot.

- Multivariate analysis has more than two variables and explains effects of variables on responses.

### **1. What is linear regression?**

In simple terms, linear regression is a method of finding the best straight line fitting to the given data, i.e. finding the best linear relationship between the independent and dependent variables.

In technical terms, linear regression is a machine learning algorithm that finds the best linear-fit relationship on any given data, between independent and dependent variables. It is mostly done by the Sum of Squared Residuals Method.

### **2. State the assumptions in a linear regression model.**

**There are three main assumptions in a linear regression model:**

1. The assumption about the form of the model:  
It is assumed that there is a linear relationship between the dependent and independent variables. It is known as the ‘linearity assumption’.
2. Assumptions about the residuals:
  1. Normality assumption: It is assumed that the error terms,  $\varepsilon^{(i)}$ , are normally distributed.
  2. Zero mean assumption: It is assumed that the residuals have a mean value of zero.
  3. Constant variance assumption: It is assumed that the residual terms have the same (but unknown) variance,  $\sigma^2$ . This assumption is also known as the assumption of homogeneity or homoscedasticity.
  4. Independent error assumption: It is assumed that the residual terms are independent of each other, i.e. their pair-wise covariance is zero.
3. Assumptions about the estimators:
  1. The independent variables are measured without error.
  2. The independent variables are linearly independent of each other, i.e. there is no multicollinearity in the data.

**Explanation:**

1. This is self-explanatory.
2. If the residuals are not normally distributed, their randomness is lost, which implies that the model is not able to explain the relation in the data. Also, the mean of the residuals should be zero.

$$Y^{(i)} = \beta_0 + \beta_1 X^{(i)} + \varepsilon^{(i)}$$

This is the assumed linear model, where  $\varepsilon$  is the residual term.

$$\begin{aligned} E(Y) &= E(\beta_0 + \beta_1 X^{(i)} + \varepsilon^{(i)}) \\ &= E(\beta_0 + \beta_1 X^{(i)}) + E(\varepsilon^{(i)}) \end{aligned}$$

If the expectation(mean) of residuals,  $E(\varepsilon^{(i)})$ , is zero, the expectations of the target variable and the model become the same, which is one of the

targets of the model.

The residuals (also known as error terms) should be independent. This means that there is no correlation between the residuals and the predicted values, or among the residuals themselves. If some correlation is present, it implies that there is some relation that the regression model is not able to identify.

3. If the independent variables are not linearly independent of each other, the uniqueness of the least squares solution (or normal equation solution) is lost.

### **3. What is feature engineering? How do you apply it in the process of modelling?**

Feature engineering is the process of transforming raw data into features that better represent the underlying problem to the predictive models , resulting in improved model accuracy on unseen data.

In layman terms, feature engineering means the development of new features that may help you understand and model the problem in a better way. Feature engineering is of two kinds — business driven and data-driven. Business-driven feature engineering revolves around the inclusion of features from a business point of view. The job here is to transform the business variables into features of the problem. In case of data-driven feature engineering, the features you add do not have any significant physical interpretation, but they help the model in the prediction of the target variable.

To apply feature engineering, one must be fully acquainted with the dataset. This involves knowing what the given data is, what it signifies, what the raw features are, etc. You must also have a crystal clear idea of the problem, such as what factors affect the target variable, what the physical interpretation of the variable is, etc.

## 5 Breakthrough Applications of Machine Learning

### **4. What is the use of regularisation? Explain L1 and L2 regularisations.**

Regularisation is a technique that is used to tackle the problem of overfitting of the model. When a very complex model is implemented on the training data, it overfits. At times, the simple model might not be able to generalise the data and the complex model overfits. To address this problem, regularisation is used. Regularisation is nothing but adding the coefficient terms (betas) to the cost function so that the terms are penalised and are small in magnitude. This essentially helps in capturing the trends in the data and at the same time prevents overfitting by not letting the model become too complex.

- L1 or LASSO regularisation: Here, the absolute values of the coefficients are added to the cost function. This can be seen in the following equation; the highlighted part corresponds to the L1 or LASSO regularisation. This regularisation technique gives sparse results, which lead to feature selection as well.

$$\sum_{i=1}^n (Y_i - \sum_{j=1}^p X_{ij}\beta_j)^2 + \lambda \sum_{j=1}^p |\beta_j|$$

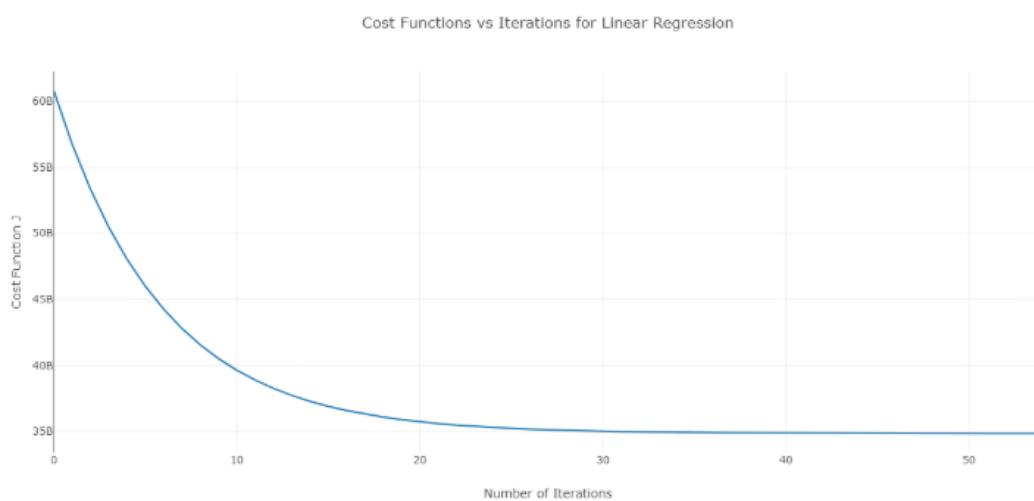
- L2 or Ridge regularisation: Here, the squares of the coefficients are added to the cost function. This can be seen in the following equation, where the highlighted part corresponds to the L2 or Ridge regularisation.

$$\sum_{i=1}^n (y_i - \sum_{j=1}^p x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^p \beta_j^2$$

### 5. How to choose the value of the parameter learning rate ( $\alpha$ )?

Selecting the value of learning rate is a tricky business. If the value is too small, the gradient descent algorithm takes ages to converge to the optimal solution. On the other hand, if the value of the learning rate is high, the gradient descent will overshoot the optimal solution and most likely never converge to the optimal solution.

To overcome this problem, you can try different values of alpha over a range of values and plot the cost vs the number of iterations. Then, based on the graphs, the value corresponding to the graph showing the rapid decrease can be chosen.



The aforementioned graph is an ideal cost vs the number of iterations curve. Note that the cost initially decreases as the number of iterations increases, but after certain iterations, the gradient descent converges and the cost does not

decrease anymore.

If you see that the cost is increasing with the number of iterations, your learning rate parameter is high and it needs to be decreased.

### **6. How to choose the value of the regularisation parameter ( $\lambda$ )?**

Selecting the regularisation parameter is a tricky business. If the value of  $\lambda$  is too high, it will lead to extremely small values of the regression coefficient  $\beta$ , which will lead to the model underfitting (high bias – low variance). On the other hand, if the value of  $\lambda$  is 0 (very small), the model will tend to overfit the training data (low bias – high variance).

There is no proper way to select the value of  $\lambda$ . What you can do is have a subsample of data and run the algorithm multiple times on different sets. Here, the person has to decide how much variance can be tolerated. Once the user is satisfied with the variance, that value of  $\lambda$  can be chosen for the full dataset. One thing to be noted is that the value of  $\lambda$  selected here was optimal for that subset, not for the entire training data.

### **7. Can we use linear regression for time series analysis?**

One can use linear regression for time series analysis, but the results are not promising. So, it is generally not advisable to do so. The reasons behind this are

- 
1. Time series data is mostly used for the prediction of the future, but linear regression seldom gives good results for future prediction as it is not meant for extrapolation.
  2. Mostly, time series data have a pattern, such as during peak hours, festive seasons, etc., which would most likely be treated as outliers in the linear regression analysis.

### **8. What value is the sum of the residuals of a linear regression close to? Justify.**

**Ans** The sum of the residuals of a linear regression is 0. Linear regression works on the assumption that the errors (residuals) are normally distributed with a mean of 0, i.e.

$$Y = \beta^T X + \varepsilon$$

Here,  $Y$  is the target or dependent variable,

$\beta$  is the vector of the regression coefficient,

$X$  is the feature matrix containing all the features as the columns,

$\varepsilon$  is the residual term such that  $\varepsilon \sim N(0, \sigma^2)$ .

So, the sum of all the residuals is the expected value of the residuals times the total number of data points. Since the expectation of residuals is 0, the sum of all the residual terms is zero.

**Note:**  $N(\mu, \sigma^2)$  is the standard notation for a normal distribution having mean  $\mu$  and standard deviation  $\sigma^2$ .

### **9. How does multicollinearity affect the linear regression?**

**Ans** Multicollinearity occurs when some of the independent variables are highly correlated (positively or negatively) with each other. This multicollinearity causes a problem as it is against the basic assumption of linear regression. The

presence of multicollinearity does not affect the predictive capability of the model. So, if you just want predictions, the presence of multicollinearity does not affect your output. However, if you want to draw some insights from the model and apply them in, let's say, some business model, it may cause problems.

One of the major problems caused by multicollinearity is that it leads to incorrect interpretations and provides wrong insights. The coefficients of linear regression suggest the mean change in the target value if a feature is changed by one unit. So, if multicollinearity exists, this does not hold true as changing one feature will lead to changes in the correlated variable and consequent changes in the target variable. This leads to wrong insights and can produce hazardous results for a business.

A highly effective way of dealing with multicollinearity is the use of VIF (Variance Inflation Factor). Higher the value of VIF for a feature, more linearly correlated is that feature. Simply remove the feature with very high VIF value and re-train the model on the remaining dataset.

**10. What is the normal form (equation) of linear regression? When should it be preferred to the gradient descent method?**

**The normal equation for linear regression is —**

$$\beta = (X^T X)^{-1} \cdot X^T Y$$

Here,  $Y = \beta^T X$  is the model for the linear regression,

$Y$  is the target or dependent variable,

$\beta$  is the vector of the regression coefficient, which is arrived at using the normal equation,

$X$  is the feature matrix containing all the features as the columns.

Note here that the first column in the  $X$  matrix consists of all 1s. This is to incorporate the offset value for the regression line.

Comparison between gradient descent and normal equation:

Gradient Descent	Normal Equation
Needs hyper-parameter tuning for alpha (learning parameter)	No such need
It is an iterative process	It is a non-iterative process
$O(kn^2)$ time complexity	$O(n^3)$ time complexity due to evaluation of $X^T X$
Preferred when $n$ is extremely large	Becomes quite slow for large values

Here, ‘ $k$ ’ is the maximum number of iterations for gradient descent, and ‘ $n$ ’ is the total number of data points in the training set.

Clearly, if we have large training data, normal equation is not preferred for use. For small values of ‘ $n$ ’, normal equation is faster than gradient descent.

What is Machine Learning and Why it matters

**11. You run your regression on different subsets of your data, and in each subset, the beta value for a certain variable varies wildly. What could be the issue here?**

This case implies that the dataset is heterogeneous. So, to overcome this problem, the dataset should be clustered into different subsets, and then separate models should be built for each cluster. Another way to deal with this problem is to use non-parametric models, such as decision trees, which can deal with heterogeneous data quite efficiently.

**12. Your linear regression doesn't run and communicates that there is an infinite number of best estimates for the regression coefficients. What could be wrong?**

This condition arises when there is a perfect correlation (positive or negative) between some variables. In this case, there is no unique value for the coefficients, and hence, the given condition arises.

**13. What do you mean by adjusted R<sup>2</sup>? How is it different from R<sup>2</sup>?**

Adjusted R<sup>2</sup>, just like R<sup>2</sup>, is a representative of the number of points lying around the regression line. That is, it shows how well the model is fitting the training data. The formula for adjusted R<sup>2</sup> is —

$$R_{adj}^2 = 1 - \left[ \frac{(1-R^2)(n-1)}{n-k-1} \right]$$

Here,  $n$  is the number of data points, and  $k$  is the number of features.

One drawback of R<sup>2</sup> is that it will always increase with the addition of a new feature, whether the new feature is useful or not. The adjusted R<sup>2</sup> overcomes this drawback. The value of the adjusted R<sup>2</sup> increases only if the newly added feature plays a significant role in the model.

**14. How do you interpret the residual vs fitted value curve?**

The residual vs fitted value plot is used to see whether the predicted values and residuals have a correlation or not. If the residuals are distributed normally, with a mean around the fitted value and a constant variance, our model is working fine; otherwise, there is some issue with the model.

The most common problem that can be found when training the model over a large range of a dataset is heteroscedasticity (this is explained in the answer below). The presence of heteroscedasticity can be easily seen by plotting the residual vs fitted value curve.

**15. What is heteroscedasticity? What are the consequences, and how can you overcome it?**

A random variable is said to be heteroscedastic when different subpopulations have different variabilities (standard deviation).

The existence of heteroscedasticity gives rise to certain problems in the regression analysis as the assumption says that error terms are uncorrelated and, hence, the variance is constant. The presence of heteroscedasticity can often be seen in the form of a cone-like scatter plot for residual vs fitted values.

One of the basic assumptions of linear regression is that heteroscedasticity is not present in the data. Due to the violation of assumptions, the Ordinary Least Squares (OLS) estimators are not the Best Linear Unbiased Estimators (BLUE). Hence, they do not give the least variance than other Linear Unbiased Estimators (LUEs).

There is no fixed procedure to overcome heteroscedasticity. However, there are some ways that may lead to a reduction of heteroscedasticity. They are —

1. Logarithmising the data: A series that is increasing exponentially often results in increased variability. This can be overcome using the log transformation.
2. Using weighted linear regression: Here, the OLS method is applied to the weighted values of X and Y. One way is to attach weights directly related to the magnitude of the dependent variable.

How does Unsupervised Machine Learning Work?

#### **16. What is VIF? How do you calculate it?**

Variance Inflation Factor (VIF) is used to check the presence of multicollinearity in a dataset. It is calculated as—

Here,  $VIF_j$  is the value of VIF for the  $j^{th}$  variable,

$R_j^2$  is the  $R^2$  value of the model when that variable is regressed against all the other independent variables.

If the value of VIF is high for a variable, it implies that the  $R^2$  value of the corresponding model is high, i.e. other independent variables are able to explain that variable. In simple terms, the variable is linearly dependent on some other variables.

#### **17. How do you know that linear regression is suitable for any given data?**

To see if linear regression is suitable for any given data, a scatter plot can be used. If the relationship looks linear, we can go for a linear model. But if it is not the case, we have to apply some transformations to make the relationship linear. Plotting the scatter plots is easy in case of simple or univariate linear regression. But in case of multivariate linear regression, two-dimensional pairwise scatter plots, rotating plots, and dynamic graphs can be plotted.

#### **18. How is hypothesis testing used in linear regression?**

Hypothesis testing can be carried out in linear regression for the following purposes:

1. To check whether a predictor is significant for the prediction of the target variable. Two common methods for this are —
  1. By the use of p-values:  
If the p-value of a variable is greater than a certain limit (usually

0.05), the variable is insignificant in the prediction of the target variable.

2. By checking the values of the regression coefficient:

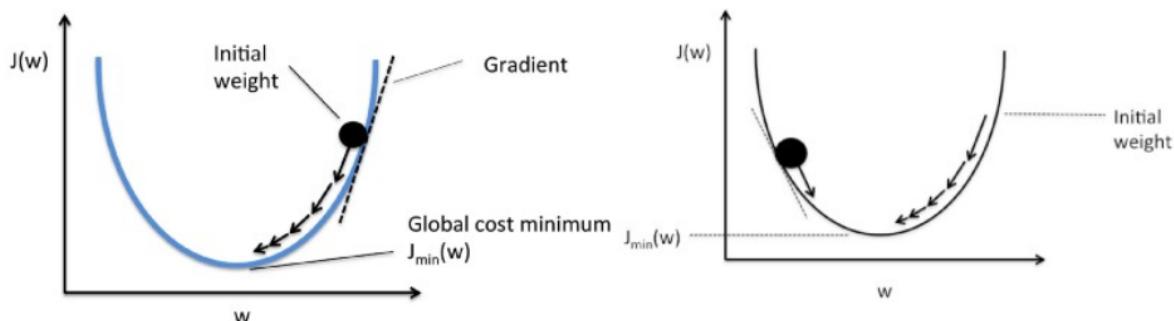
If the value of regression coefficient corresponding to a predictor is zero, that variable is insignificant in the prediction of the target variable and has no linear relationship with it.

2. To check whether the calculated regression coefficients are good estimators of the actual coefficients.

### **19. Explain gradient descent with respect to linear regression.**

Gradient descent is an optimisation algorithm. In linear regression, it is used to optimise the cost function and find the values of the  $\beta$ s (estimators) corresponding to the optimised value of the cost function.

Gradient descent works like a ball rolling down a graph (ignoring the inertia). The ball moves along the direction of the greatest gradient and comes to rest at the flat surface (minima).



Mathematically, the aim of gradient descent for linear regression is to find the solution of

$\text{ArgMin } J(\Theta_0, \Theta_1)$ , where  $J(\Theta_0, \Theta_1)$  is the cost function of the linear regression. It is given by —

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)})^2$$

Here,  $h$  is the linear hypothesis model,  $h = \Theta_0 + \Theta_1 x$ ,  $y$  is the true output, and  $m$  is the number of the data points in the training set.

Gradient Descent starts with a random solution, and then based on the direction of the gradient, the solution is updated to the new value where the cost function has a lower value.

The update is:

Repeat until convergence

$$\theta_j = \theta_j - \alpha \frac{1}{m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_j^{(i)} \text{ for } j = 1, 2, \dots, n$$

## **20. How do you interpret a linear regression model?**

A linear regression model is quite easy to interpret. The model is of the following form:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n$$

The significance of this model lies in the fact that one can easily interpret and understand the marginal changes and their consequences. For example, if the value of  $x_0$  increases by 1 unit, keeping other variables constant, the total increase in the value of  $y$  will be  $\beta_i$ . Mathematically, the intercept term ( $\beta_0$ ) is the response when all the predictor terms are set to zero or not considered.

These 6 Machine Learning Techniques are Improving Healthcare

## **21. What is robust regression?**

A regression model should be robust in nature. This means that with changes in a few observations, the model should not change drastically. Also, it should not be much affected by the outliers.

A regression model with OLS (Ordinary Least Squares) is quite sensitive to the outliers. To overcome this problem, we can use the WLS (Weighted Least Squares) method to determine the estimators of the regression coefficients. Here, less weights are given to the outliers or high leverage points in the fitting, making these points less impactful.

## **22. Which graphs are suggested to be observed before model fitting?**

Before fitting the model, one must be well aware of the data, such as what the trends, distribution, skewness, etc. in the variables are. Graphs such as histograms, box plots, and dot plots can be used to observe the distribution of the variables. Apart from this, one must also analyse what the relationship between dependent and independent variables is. This can be done by scatter plots (in case of univariate problems), rotating plots, dynamic plots, etc.

## **23. What is the generalized linear model?**

The generalized linear model is the derivative of the ordinary linear regression model. GLM is more flexible in terms of residuals and can be used where linear regression does not seem appropriate. GLM allows the distribution of residuals

to be other than a normal distribution. It generalizes the linear regression by allowing the linear model to link to the target variable using the linking function. Model estimation is done using the method of maximum likelihood estimation.

#### ***24. Explain the bias-variance trade-off.***

Bias refers to the difference between the values predicted by the model and the real values. It is an error. One of the goals of an ML algorithm is to have a low bias.

Variance refers to the sensitivity of the model to small fluctuations in the training dataset. Another goal of an ML algorithm is to have low variance. For a dataset that is not exactly linear, it is not possible to have both bias and variance low at the same time. A straight line model will have low variance but high bias, whereas a high-degree polynomial will have low bias but high variance.

There is no escaping the relationship between bias and variance in machine learning.

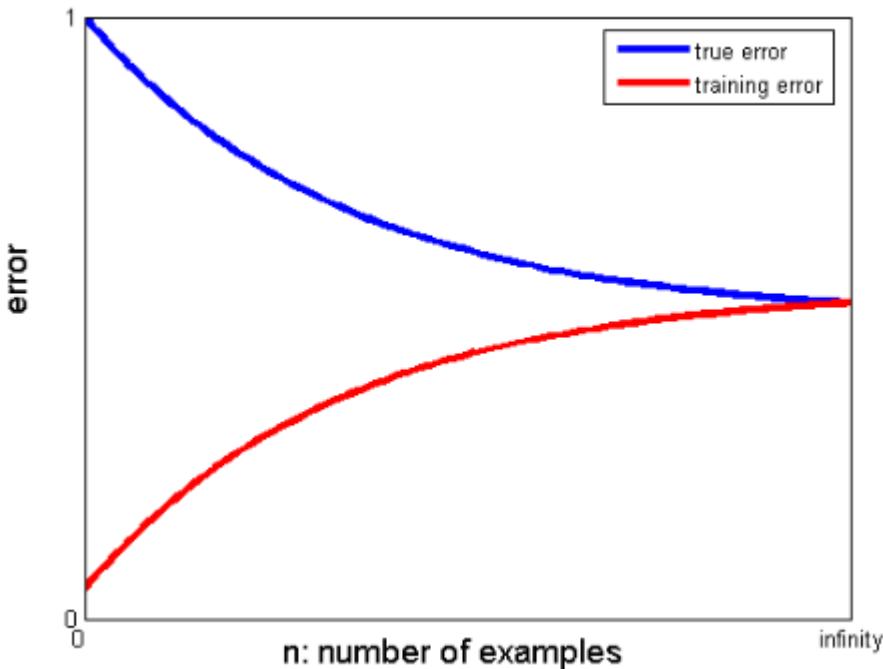
1. Decreasing the bias increases the variance.
2. Decreasing the variance increases the bias.

So, there is a trade-off between the two; the ML specialist has to decide, based on the assigned problem, how much bias and variance can be tolerated. Based on this, the final model is built.

#### ***25. How can learning curves help create a better model?***

Learning curves give the indication of the presence of overfitting or underfitting.

In a learning curve, the training error and cross-validating error are plotted against the number of training data points. A typical learning curve looks like this:



If the training error and true error (cross-validating error) converge to the same value and the corresponding value of the error is high, it indicates that the model is underfitting and is suffering from high bias.

What do you understand by regularization?

Regularization is a strategy for dealing with the problem of model overfitting. Overfitting occurs when a complicated model is applied to training data. The basic model may not be able to generalize the data at times, and the complicated model may overfit the data. Regularization is used to alleviate this issue.

Regularization is the process of adding coefficient terms (betas) to the minimization problem in such a way that the terms are penalized and have a modest magnitude. This essentially aids in identifying data patterns while also preventing overfitting by preventing the model from becoming too complex.

What do you understand about feature engineering?

The process of changing original data into features that better describe the underlying problem to predictive models, resulting in enhanced model accuracy on unseen data, is known as feature engineering. In layman's terms, feature engineering refers to the creation of additional features that may aid in the better understanding and modelling of an issue. There are two types of feature engineering: business-driven and data-driven. The incorporation of features

from a commercial standpoint is the focus of business-driven feature engineering.

What is the bias-variance tradeoff?

The gap between the model - predicted values and the actual values is referred to as bias. It's a mistake. A low bias is one of the objectives of an ML algorithm. The vulnerability of the model to tiny changes in the training dataset is referred to as variance. Low variance is another goal of an ML algorithm. It is impossible to have both low bias and low variance in a dataset that is not perfectly linear. The variance of a straight line model is low, but the bias is large, whereas the variance of a high-degree polynomial is low, but the bias is high. In machine learning, the link between bias and variation is unavoidable.

## **1. What are Overfitting and Underfitting in Machine Learning?**

**Ans:** Two main problems occurring in machine learning that degrade the ML model's performance are Overfitting and Underfitting.

Overfitting occurs mainly in supervised learning. When a machine learning model tries to cover more data points than required in a dataset, the model begins caching noise and incorrect values in the dataset. This decreases the model's accuracy and efficiency.

Underfitting occurs because of overfitting in a model. The training data being fed may be stopped due to overfitting, thereby leaving insufficient training data to learn from. The ML model may not discover the best match for the prevailing trend in the data. This reduces accuracy and results in irregular outcomes.

## **2. How to handle categorical variables in KNN?**

**Ans:** By generating dummy variables from categorical variables and using them rather than the actual category variable, they can be handled. Don't create  $(k-1)$  instead of  $k$  dummy variables.

Let's take "Age Group" as a categorical variable name that has 5 distinct categories/levels. Therefore, we generate 5 dummy variables. '1' as a value is allocated to every dummy variable that belongs to a department, and in another case, '0' is allocated as a value.

## **3. Is KNN suitable for regression? How to apply KNN to Regression?**

**Ans:** Yes, the K-nearest neighbor (KNN) is suitable for regression. We can employ the KNN algorithm when a continuous dependent variable is present. Taking the mean of the values from k to its closest neighbors will give us the predicted value.

#### **4. Compare KNN and K Means Algorithms.**

**Ans:** In contrast to K-means, a form of unsupervised machine learning, KNN would be supervised machine learning. While a regression or classification ML algorithm is employed by KNN, K-means employs a clustering ML algorithm. When every data presents the same scale, this situation is suitable for KNN but not for K-means.

#### **5. Besides altering k how can the model's higher variance be reduced?**

**Ans:** If the number of samples drawn from the original dataset is not limited, a sample variance reduction strategy would be to sample numerous times and then use the major votes in the KNN models to fit each of these samples and categorize all the test data points. This method, known as **bagging**, is a way of reducing variation.

#### **6. How does sampling affect KNN?**

**Ans:** Because KNN works in such a point-by-point manner, sampling accomplishes numerous tasks from the viewpoint of a single data point.

- As the dataset gets scarce, the average distance toward the k-nearest neighbors rises.
- As a result, the region covered by k-nearest neighbors grows in size, thus covering a bigger portion of the feature space.
- And sample variance increases.

The rise in variance is a result of such a change in input. Variance refers to the difference seen in projections made from various population samples. Why might the immediate impacts of sampling raise the model's variance?

Notice how the same k data points now represent a greater region of the feature space. The population space it depicts has grown even while the sample size has not grown. As a result, the proportion of classes within k closest data points will have a bigger variance, as will the categorization of each data point.

#### **7. What will happen if we change the K's value in KNN?**

**Ans:** As k rises, the class limits of the predictions get more smooth.

Lowering the k value renders the KNN model more "sensitive." In other words, it has a higher sensitivity to local variations in the dataset. The model's "sensitivity" immediately corresponds to its variance.

The examples demonstrate the inverse connection between variance and k. Consider what happens to KNN when k attains its maximum value,  $k=n$  ( $n$  refers to points present inside the practice set). In this scenario, the projections are led by the major class present in the practice set. It will select the most prevalent class inside the data and never stray, resulting in virtually zero variance. As a result, to minimize variance, k should be raised.

**Final Verdict:** To counterbalance the increased variance caused by sampling, k can be adjusted to reduce model variance.

## 8. What is the thumb rule for approaching the KNN problem?

**Ans:**

**Step 1:** Load the data

**Step 2:** Initialize the value of k

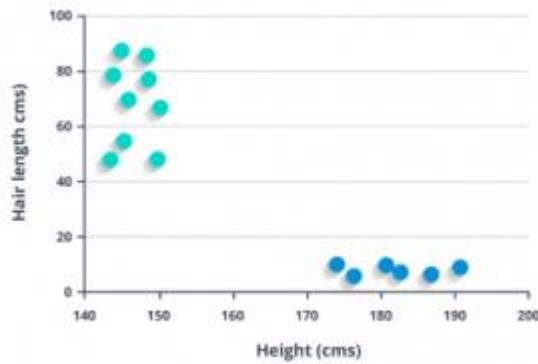
- Find the distance between each training data row and the test data. Because it is the most often used approach, we shall utilize Euclidean distance as our distance metric. Chebyshev, cosine, and other metrics can also be employed.
- Sort the computed distances by distance value in ascending order.
- Get the first k rows of the sorted array.
- Identify the most prominent class of these rows.
- After receiving the projected class, return it and repeat the process up to the amount of training information points that were received.

**KNN Code Snippet:**

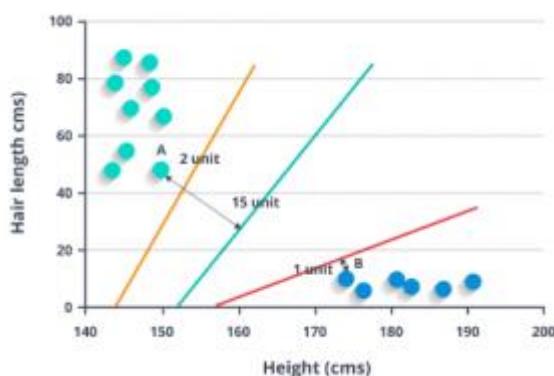
```

1 #KNN Algorithm for Classification / Author: Sarwar Ghousie Mastan
2 #Creating our own features
3 #Assigning features and Label variables
4 # First Feature
5 weather = ['Sunny','Sunny','Overcast','Rainy','Rainy','Rainy','Overcast','Sunny','Sunny',
6 'Rainy','Sunny','Overcast','Overcast','Rainy']
7
8 # Second Feature
9 temp=['Hot','Hot','Hot','Mild','Cool','Cool','Mild','Cool','Mild','Mild','Mild','Hot','Mild']
10
11 #Label or Target Variable
12 play = ['No','No','Yes','Yes','No','Yes','No','Yes','Yes','Yes','Yes','Yes','Yes','No']
13
14 #Importing LabelEncoder
15 from sklearn import preprocessing
16 #creating LabelEncoder
17 le = preprocessing.LabelEncoder()
18 weather_encoded = le.fit_transform(weather)
19 print(weather_encoded)
20
21 #Converting string Labels into numbers
22 temp_encoded = le.fit_transform(temp)
23 label= le.fit_transform(play)
24
25 #Combining weather and temp into single List of tuples
26 features = list(zip(weather_encoded,temp_encoded))
27
28 from sklearn.neighbors import KNeighborsClassifier
29 model = KNeighborsClassifier(n_neighbors=3)
30
31 #Train the model using the training sets
32 model.fit(features,label)
33
34 #Predict Output
35 predicted = model.predict([(0,2)]) # 0-Overcast ,2-Mild
36 print(predicted)

```



Now we'll look for some lines dividing the data into two distinct data groups. Along this line, each group's closest points will have the most distance.



The line dividing the dataset into two distinctly categorized groups is known as the classifier line because the two closest endpoints are at the farthest separation from the line. The new data may then be classified on the basis of where on each side of the line the testing data fits.

## **9. What is an SVM Algorithm?**

**Ans:** Machine Learning is one of the most astounding technological advancements in the modern era. This recent development and the popularity of machine learning algorithms have greatly changed the organization's focus on data-driven decision-making. As a result, the job vacancies in machine learning are at their peak. But as the number of vacancies is increasing and skill criteria, as well as the competition, are also getting high. This makes cracking a machine-learning interview is becoming a tough job.

## **10. Explain Support Vectors.**

**Ans:** The line having the largest difference within 2 classes, called the "best" line, is found using Support Vector Machines. The points on this margin are called Support Vectors.

## **11. What purpose does a Support Vector serve in SVM?**

**Ans:** An SVM classifies data by locating the hyperplane that optimizes the separation gap within two classes. In the data sets composing the hyperplane, support vectors are the largest spots.

## **12. What do you mean by kernels?**

**Ans:** Kernels are a group of mathematical operations that SVM algorithms rely on. The kernel's function changes the received input info into a suitable form. Various sorts of kernel functions are employed by several SVM algorithms. There are four different kernel types in SVM.

- Polynomial kernel
- Linear Kernel
- Sigmoid kernel
- Radial basis kernel

## **13. What is meant by Kernel Trick?**

**Ans:**

**Short**

**Answer:**

It enables us to interact in the initial feature space without having to compute data coordinates in a greater space.

### **Long Answer:**

1. SVMs find the n-1 higher dimensional space to separate a dataset having n features (n-dimensional).
2. Non-linearly distinct sets of data are not suitable for SVMs.
3. Currently, SVM can handle these datasets.
4. However, it is frequently feasible to turn our non-linearly separable dataset into a higher-dimensional dataset that is linearly separable, allowing SVMs to perform well.
5. Unfortunately, the number of dimensions you must add (through transformations) frequently relies on how many dimensions you currently have (and not linearly). It becomes practically challenging for datasets with several characteristics to test out all of the interesting modifications.
6. Here comes the Kernel Trick.
7. Fortunately, SVMs only need to compute the pair-wise dot products in the (higher-dimensional) feature space (during training).
8. The kernel function is presented as aimed at a specific vector set in a low-dimensional subspace. Also, a conversion into a high-dimensional space is possible. The dot product can be analyzed without explicitly transforming the vectors inside the high-dimensional space.
9. We are saved!

## **14. Why is SVM called a Large Margin Classifier?**

**Ans:**

**Short**

It positions the decision border so that the separation between the two clusters is maximized.

**Answer:**

**Long**

The hyperplane where the gap from the data sets is maximum should be chosen, which the geometric margin formalizes. This is how it's characterized:

**Answer:**

$$\gamma^{(i)} = \frac{y^{(i)}(w^T x^{(i)} + b)}{\|w\|}$$

It is just the standardized functional margin. As a result, the maximum margin classifier-a forerunner of the SVM-was created.

## **15. What are the variations in SVM and Logistics Regression?**

**Ans:**

- SVM seeks the best gap between the support vectors and the line that divides the classes, lowering the chance of data inaccuracy, whereas Logistic Regression does not. Instead, it has many decision limits with varying weights that are close to the optimum point.
- SVM works best with semi-structured and unstructured data, such as text and pictures, whereas Logistic Regression fits with independent variables that have previously been defined.
- SVM is based on data geometry, whereas Logistic Regression is dependent on statistical techniques.
- SVM may be used to analyze sets of data not separable linearly, but Logistic Regression cannot.
- Overfitting occurs in Logistic Regression but is less likely in SVM.

## **16. When Should Logistic Regression be used over Support Vector Machine?**

**Ans:** You can use a support vector machine or logistic regression depending on the number of training datasets/features you have.

As an example, consider the following:

*n is the number of features, while m denotes the number of training samples.*

- n is large and m is small: SVM alongside a linear kernel/Logistic Regression.
- n is small and m is moderate: SVM using a kernel (eg. Polynomial, Gaussian, etc.).
- n is a small number, m is a huge number: Use SVM alongside a linear kernel/Logistic Regression, after linking more characteristics yourself.

## **17. What does the C and Gamma parameter in SVM signify?**

**Ans:**

**Short Answer:**

The high energy boundaries determining how well an SVM model operates are gamma and cost. An ML model must carefully balance its bias and variance.

High Gamma values for SVM provide greater precision but produce false outcomes, and vice versa. Similarly, a high-cost parameter (C) value suggests poor accuracy but less bias, and vice versa.

Choose a model with the least bias and variance. As a result, you must select the appropriate values for C and Gamma.

Using approaches such as Grid search, optimal C values and Gamma may be identified.

### **Long Answer:**

By how much the training examples should be stopped from being incorrectly classified by the SVM operator can be referred to by the C parameter. If the optimizer correctly classifies each of the training points for high C values, it may choose a smaller-margin hyperplane. A rather small C value will cause the optimizer to seek a larger gap between the hyperplane and the points, even though the hyperplane misclassifies numerous points. Misclassified cases should be expected for very small values of C, even if your training data can be separated linearly.

The gamma parameter characterizes the impact of one training sample, typically low values indicating "far" and large values indicating "near."

The gamma parameters may be considered the opposite of the radius of effect of samples chosen as support vectors by the model. The impact sphere's area comprises the support vector itself when Gamma is extremely big. Overfitting can't be stopped using C regularization in this case.

When Gamma is very tiny, the model is too limited and cannot describe the data's complexity or "shape." The whole training set would be included in the zone of influence of any chosen support vector. The resultant model will operate in the same way as a linear model containing a series of hyperplanes separating any two classes' high-density centers.

## **18. Discuss SVM's plus points and downsides.**

**Ans:**

### **SVM Advantages**

- When we don't know anything about the data, SVMs come in handy.

- Even unstructured & semi-structured data such as text, images, & trees work well.
- SVM's kernel trick seems to be a real strength. We can solve any difficult issue with a suitable kernel function.
- SVM, unlike neural networks, does not solve for local optima.
- It scales reasonably well to high-dimensional data sets.
- In reality, SVM models have generality; the danger of over-fitting is lower in SVM.
- The SVM has always been compared to the ANN. In regards to performance, SVMs beat ANN models.

## **SVM Disadvantages**

- It is difficult to select a "good" kernel function.
- Long training time for large datasets.
- The individual impact, variable weights, and final model are difficult to grasp and analyze.
- We can't conduct modest calibrations to the model since the final model isn't easily visible, making it difficult to include our business logic.
- Gamma and Cost C are the high energy boundaries of SVM. These hyper-parameters are difficult to fine-tune. It is difficult to imagine their influence.

## **SVM code snippet:**

```

1 #SVM Algorithm for Classification | Author: Sarwar Ghousie Mastan
2 #Import scikit_learn dataset library
3 from sklearn import datasets
4 from sklearn.model_selection import train_test_split
5 from sklearn import svm
6 from sklearn import metrics
7
8 #Load dataset
9 cancer = datasets.load_breast_cancer()
10
11 #splitting dataset into training set and test set
12 x_train , x_test, y_train, y_test =train_test_split(cancer.data,cancer.target, test_size=0.1,random_state=101)
13
14 #Creating a SVM classifier
15 slf = svm.SVC(kernel='linear') # Linear Kernel
16
17 #training the model using training data
18 clf.fit(x_train , y_train)
19
20 #Predicting the response for test dataset
21 y_pred = clf.predict(x_test)
22
23 #model Accuracy: How often is the classifier correct?
24 print("Accuracy: ", metrics.accuracy_score(y_test,y_pred))
25
26 # Model Precesion
27 print("Precesion: ", metrics.precesion_score(y_test,y_pred))
28
29 #Model Recall
30 print("Recall: ", metrics.recall_score(y_test,y_pred))
31
32

```

## 19. What is the Reinforcement Learning technique?

**Ans:** Reinforcement Learning is a response-based Machine Learning technique in which an agent understands how to act in a specific environment by performing actions and watching the results. In general, a reinforcement learning agent can detect and grasp its environment, act, and improve via trial and error. Positive feedback is given to the agent for each positive activity; negative feedback or a penalty is given to the agent for each negative behavior.

## 20. What is Naïve Bayes Algorithm?

**Ans:** It is a classification method that estimates the likelihood of each data point relating to a group and afterward classifies the point as belonging to the category with the greatest likelihood.

### Discussion on Bayes' Theorem.

By combining the conditional probability provided a result and prior knowledge of an event occurring, Bayes' Theorem calculates the likelihood of an event occurring.

The chance that an event will occur, given that a certain event occurred, is known as conditional probability. Conditional probability, or  $P(X|Test)$ , is the possibility of X for a test outcome. For instance, what is the likelihood that an email is a spam if my spam filter flagged it as spam?

The previous likelihood is determined using preceding knowledge or the percentage of prior samples. For instance, what is the likelihood that any email is a spam?

### **Formally**

- Posterior probability =  $P(A|B)$  = The likelihood that A occurred if B occurred.
- Conditional probability =  $P(B|A)$  = The likelihood of B occurring if A is true.
- Prior probability =  $P(A)$  = Chance of A occurring in general
- Evidence Probability =  $P(B)$  = Chance of a Positive Test

### **21. Why is Naïve Bayes (NB) called Naïve?**

**Ans:** Naive Bayes is called naïve since it believes that the characteristics that go into the model are unrelated. A change in one variable has no direct effect on the other.

The term "Naive Bayes" refers to the assumption that measurement characteristics are independent of one another. Since it's mostly wrong, it is known as naive. Even so, here's how it works: NB is a pretty intuitive categorization method. It asks, "Which class (A or B) does this measurement belong to given certain characteristics?". It calculates the result by multiplying the proportion of all A-class samples by the sum of all previous observations that have the same attributes. If this value is greater than the comparable calculation of class B, the measurement is said to belong to class A.

### **22. What are the feature matrix and response vectors?**

**Ans:**

**Feature matrix:-** The vector or rows in the set of data composes a feature matrix. Every vector has a dependent feature value.

**Response vectors:-** For every row of a feature matrix, the response vector includes the value of a class variable (output or prediction).

### **23. What are the uses of Naive Bayes classification techniques?**

**Ans:** The uses of Naive Bayes classification techniques are as follow:

- Is a mail junk or not?

- What is the best way to categorize a news piece concerning technology, politics, or sports?
- Is a written line expressing positive or negative emotions?
- Face recognition software also makes use of it.

## 24. What are the Naïve Bayes Algorithm's Benefits and Drawbacks?

**Ans:**

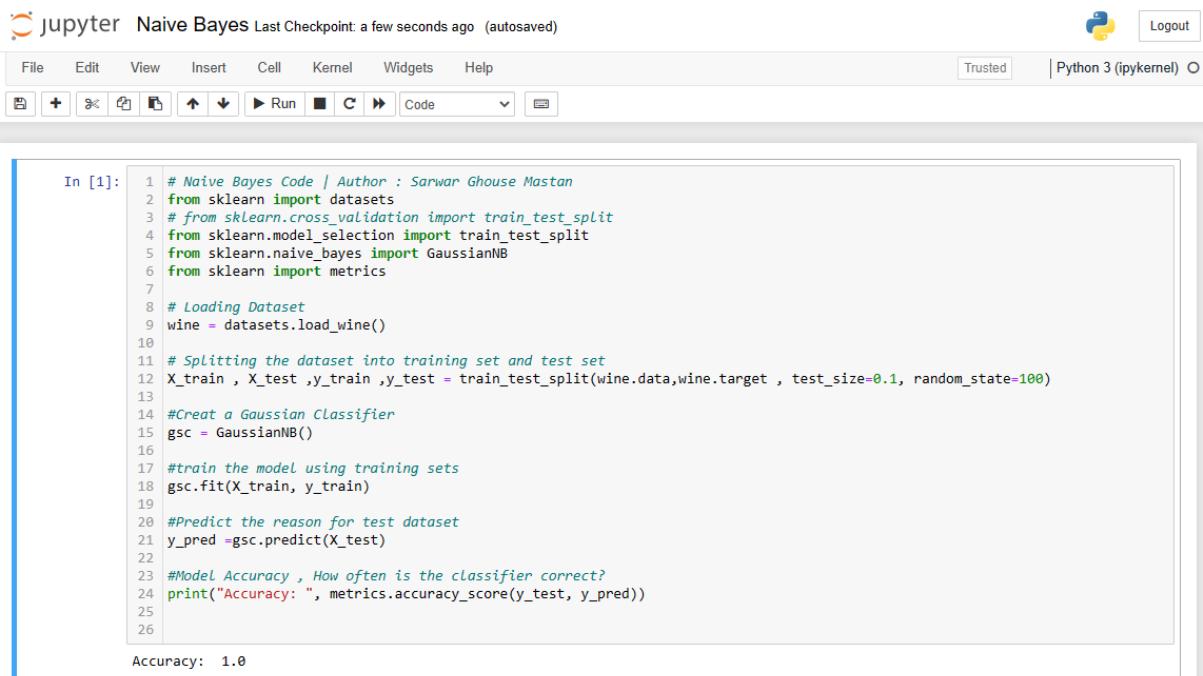
### Advantages

1. Fast
2. Highly scalable.
3. Used for binary and Multiclass Classification.
4. An excellent option for classifying text.
5. Shorter data sets may be taught with ease.

### Disadvantages

According to Naive Bayes, the characteristics are independent of one another. In the actual world, however, characteristics are interdependent.

### Naïve Bayes Code Snippet:



The screenshot shows a Jupyter Notebook interface with the following details:

- Title Bar:** jupyter Naive Bayes Last Checkpoint: a few seconds ago (autosaved)
- Toolbar:** File, Edit, View, Insert, Cell, Kernel, Widgets, Help, Trusted, Python 3 (ipykernel) ○
- Code Cell (In [1]):**

```

1 # Naive Bayes Code | Author : Sarwar Ghouse Mastan
2 from sklearn import datasets
3 # from sklearn.cross_validation import train_test_split
4 from sklearn.model_selection import train_test_split
5 from sklearn.naive_bayes import GaussianNB
6 from sklearn import metrics
7
8 # Loading Dataset
9 wine = datasets.load_wine()
10
11 # Splitting the dataset into training set and test set
12 X_train , X_test ,y_train ,y_test = train_test_split(wine.data,wine.target , test_size=0.1, random_state=100)
13
14 #Create a Gaussian Classifier
15 gsc = GaussianNB()
16
17 #train the model using training sets
18 gsc.fit(X_train, y_train)
19
20 #Predict the reason for test dataset
21 y_pred =gsc.predict(X_test)
22
23 #Model Accuracy , How often is the classifier correct?
24 print("Accuracy: ", metrics.accuracy_score(y_test, y_pred))
25
26

```
- Output:** Accuracy: 1.0

## **25. Explain K-means Clustering and the steps for achieving K-means Clustering.**

**Ans:** K-means (Macqueen, 1967) is among the most basic unsupervised learning methods for dealing with the well-known clustering challenge. K-means clustering is a signal processing-derived vector quantization method that is commonly utilized in data mining clustering algorithms.

The K-means algorithm can be applied as below if k is given:

- Creating k subsets (non-vacant) by dividing the objects.
- Finding the centroids of the current division.
- Allocating every point to a certain cluster.
- Determine the separations among each point, then allot points to the cluster that is closest to the centroid.
- Finding the newly formed cluster's core after reallocating the points.

## **26. In the K-means algorithm what does "means" signify?**

**Ans:** The 'means' signifies data averaging or determining the centroid.

There are k-medoids and k-medians algorithms as well.

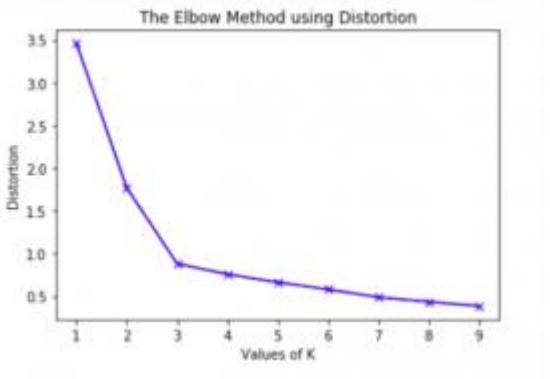
K-medoids reduce the total of dissimilarities among points tagged as belonging to a cluster as well as a point defined as the cluster's center. k-medoids select data points as centers (exemplars or medoids) unlike the K-means algorithms.

A variation of the K-means clustering is K-medians. In it, each cluster's centroid is determined by calculating the median rather than the mean.

## **27. How can I determine 'K-means' ideal amount of clusters? Describe the elbow technique and the elbow curve.**

**Ans:** The key premise underlying partitioning methods like K-means clustering is constructing clusters in a way that the total WSS (within-cluster sum of a square) [or total intra-cluster variation] is minimal. The closeness of the clustering is calculated by the total WSS and should be as low as possible.

The Elbow technique calculates total WSS as a function of cluster count: One should select a variety of clusters such that adding an additional cluster does not significantly increase the total WSS.



Notice the elbow at  $k = 3$ .

### **The ideal number of clusters are as follows:**

- Calculate the clustering algorithm (K-means clustering, for example) for various K values. Let's say, changing K's value from 10 to 100 clusters.
- Calculate the addition of squares in the cluster, WSS, for each value of K.
- Plot the WSS curve based on the number of clusters K.
- The position of the knee (bend) in the graph is frequently used to determine the number of clusters.

### **28. What is the difference between K-means and Hierarchical Clustering?**

**Ans:** K-means Clustering and Hierarchical Clustering work well together. The researcher is uninformed of the number of clusters to be created in hierarchical Clustering, but how many clusters are to be created in K-means Clustering is stated beforehand.

**Advice-** If you don't know how many clusters to build, apply hierarchical Clustering to figure it out. Then you may use K-means Clustering to create more suitable clusters. Hierarchical Clustering is just a one-time process whereas K-means is an iterative method.

### **29. What are the advantages and disadvantages of using K-means Algorithms?**

**Ans:**

#### **K-means Advantages:**

- If the variables are large and k is kept modest, K-means are usually computationally quicker than hierarchical clustering.

- When the clusters are round, K-means generates more compact clusters in comparison to hierarchical clustering.

### K-means Disadvantages:

- The value of K is difficult to anticipate.
- It doesn't perform properly when the cluster is global.
- Different beginning partitions might lead to various end clusters.
- Difficult to deal with distinct volume and size clusters (in the real sample).

### K-means code snippet:



The screenshot shows a Jupyter Notebook interface with a cell titled "K\_Means.py". The cell contains Python code for performing K-Means clustering on a dataset. The code includes importing libraries (numpy, pandas, matplotlib), preparing the data (a 2D array of points), visualizing the data, creating clusters (using KMeans from sklearn), and plotting the data points with their assigned clusters and centroids. The code is numbered from 1 to 35.

```

1 # K-Means Clustering | # Author : Sarwar Ghousie Mastan
2 #Importing Libraries
3 import numpy as np
4 import pandas as pd
5 import matplotlib.pyplot as plt
6 %matplotlib inline
7 from sklearn.cluster import KMeans
8
9 #Preparing the Data
10 X =np.array([[5,3],
11             [20,15],
12             [15,12],
13             [24,10],
14             [30,45],
15             [85,70],
16             [71,80],
17             [60,78],
18             [55,53],
19             [80,91]])
20 #Visualize data
21 plt.scatter(X[:,0],X[:,1],label='True Position')
22
23 #Creating Clusters
24 kmeans =KMeans(n_clusters=2)
25 kmeans.fit(X)
26 #To see what centroind values the algorithm generated for thr final clusters
27 print(kmeans.labels_)
28
29 #Lets plot the data points again on the graph and visualize how the data has been clustered
30 #Plotting along with labels
31 plt.scatter(X[:,0], X[:,1],c=kmeans.labels_ , cmap='rainbow')
32
33 #Plotting with 3 clusters and centroid
34 plt.scatter(X[:,0], X[:,1],c=kmeans.labels_ , cmap='rainbow')
35 plt.scatter(kmeans.cluster_centers_[:,0], kmeans.cluster_centers_[:,1], color='black')

```

### 30. What does Hierarchical Clustering mean?

**Ans:** A type of unsupervised learning approach used to combine unlabeled data points with comparable features is hierarchical clustering. There are two categories for hierarchical clustering techniques.

In hierarchical agglomerative algorithms, each piece of data is regarded as a distinct cluster, and the pairings of clusters are subsequently progressively agglomerated (bottom-up technique) or amalgamated.

Hierarchical dividing algorithms in divisive hierarchical algorithms, on the other hand, all data points are viewed as one huge cluster, and the clustering process entails splitting (Top-down method) the one big cluster into several tiny clusters.

### **31. What is the Procedure for Performing Agglomerative Hierarchical Clustering?**

**Ans:** The most used and important Hierarchical clustering, i.e., agglomerative. Below are the steps-

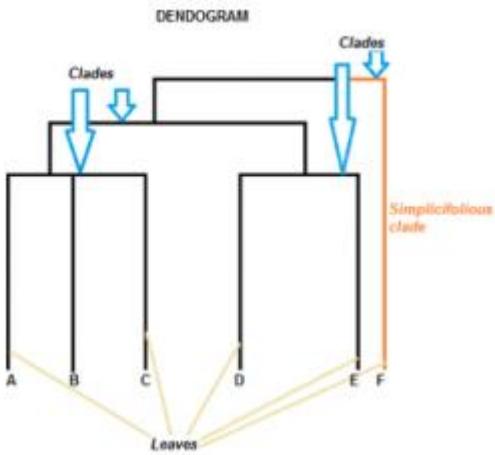
- Step 1 – Assess every point of information as a segregated cluster. As a result, we'll have, let's say, K clusters at first. The initial data points also will be K.
- Step 2 – In this step, we will create a large cluster by connecting two nearby data points. This yields K-1 clusters.
- Step 3 – Now, by linking 2 closest clusters we can construct additional clusters. We'll have a total of K-2 clusters.
- Step 4 – To build a single large cluster, repeat the preceding three stages until K equals zero, i.e., there are no more data points to connect.
- Step 5 – Finally, after creating a single large cluster, dendograms will be utilized to split the issue into numerous clusters.

### **32. What is Dendrogram and what is its importance in Hierarchical Clustering?**

**Ans:** A dendrogram is a Tree Diagram that displays hierarchical groupings-linkages between comparable groups of data. They are commonly used in biology to demonstrate the clustering of genes or samples, although they may represent any form of clustered data.

Once the large cluster is created, the dendrogram's duty begins. Depending on the situation, a dendrogram would be employed to divide the clusters into various clusters of similar data points.

## Parts of Dendrogram:



## Hierarchical Clustering Code Snippet:

jupyter Hierachial\_Clustering.py a few seconds ago

```
File Edit View Language
```

```
1 #Author - Sarwar Ghouse Mastan
2 #Hierachial Clustering (With Dendograms)
3 #Importing libraries
4 import pandas as pd
5 import numpy as np
6 import seaborn as sns
7 import matplotlib.pyplot as plt
8 %matplotlib inline
9
10 #importing data
11 customer_data = pd.read_csv('data/ta/file.csv')
12
13 #To View the results in 2-D feature space
14 #we will retain only two of these five columns(If more than 2 available)
15 # data = customer_data.iloc[:, 3:5].values
16
17 ''' We need to know the clusters thar we want our data to be split to,
18 We will again use the scipy library to create the dendograms for the dataset.
19 Execute the following Script do so: '''
20
21 import scipy.cluster.hierarchy as shc
22 plt.figure(figsize=(50,7))
23 plt.title("Customer Dendograms")
24 dend = shc.dendrogram(shc.linkage(data,method ='ward'))
25
26 #Now we know the number of clusters for our dataset,
27 #the next step is to group the data points into these five clusters
28 from sklearn.cluster import AgglomerativeClustering
29
30 cluster = AgglomerativeClustering(n_clusters=5,affinity='euclidean', linkage='ward')
31 cluster.fit_predict(data)
32
33 # As a final step, let's plot the clusters to see how actually our data has been clustered
34 plt.figure(figsize=18,7)
35 plt.scatter(data[:,0],data[:,1],c=cluster.labels_, cmap='rainbow')
```

### 33. What is Boosting?

**Ans:** Boosting is a technique for turning inferior learners into superiors. Each new tree in boosting fits an improved version of the initial data set.

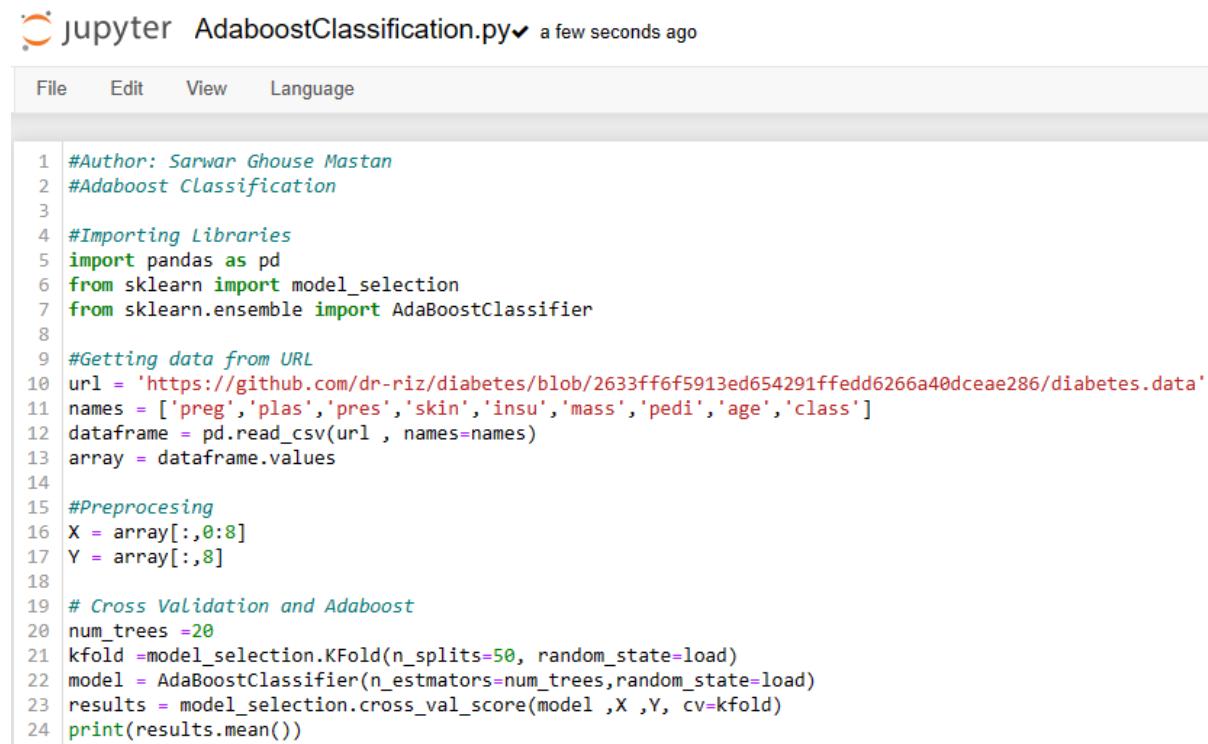
**Purpose of Boosting:** It helps the weak learner be modified to improve.

**How it evolved:** AdaBoost, or Adaptive Boosting, was the first Boosting algorithm that gained popularity. Further, it evolved and generalized as Gradient Boosting.

### 34. What is Adaboost?

**Ans:** Adaboost creates one strong learner by transforming a group of weak learners. The decision trees consisting of a single division, known as decision stumps, are weak learners. When AdaBoost generates its initial decision stump, all observations are equally weighted. The wrongly categorized observations now have greater weight than correctly classified ones, which helps in remedying the preceding issue. AdaBoost techniques apply to both classification and regression applications.

#### Adaboost Code Snippet:



The screenshot shows a Jupyter Notebook cell with the following code:

```
1 #Author: Sarwar Ghousie Mastan
2 #Adaboost Classification
3
4 #Importing Libraries
5 import pandas as pd
6 from sklearn import model_selection
7 from sklearn.ensemble import AdaBoostClassifier
8
9 #Getting data from URL
10 url = 'https://github.com/dr-riz/diabetes/blob/2633ff6f5913ed654291ffedd6266a40dceae286/diabetes.data'
11 names = ['preg', 'plas', 'pres', 'skin', 'insu', 'mass', 'pedi', 'age', 'class']
12 dataframe = pd.read_csv(url, names=names)
13 array = dataframe.values
14
15 #Preprocessing
16 X = array[:,0:8]
17 Y = array[:,8]
18
19 # Cross Validation and Adaboost
20 num_trees = 20
21 kfold = model_selection.KFold(n_splits=50, random_state=load)
22 model = AdaBoostClassifier(n_estimators=num_trees,random_state=load)
23 results = model_selection.cross_val_score(model ,X ,Y, cv=kfold)
24 print(results.mean())
```

### 35. What is Gradient Boosting Method (GBM)?

**Ans:** Gradient Boosting occurs when in an ensemble you keep adding predictors one at a time, each one correcting the one before it. However, unlike AdaBoost, which changes the weights for each incorrectly categorized observation at each iteration, the Gradient Boosting approach attempts to match the new predictor to the previous predictor's residual mistakes.

Gradient Descent is applied to identify flaws in the last learner's predictions by GBM. The GBM algorithm can be given in the following steps.

$F_1(x) = y$  to match a model with the information.

Make a new model with  $F_2(x) = F_1(x) + h_1(x)$ .

By consistently adding weak learners, we can account for a large portion of the mistake in the original model and minimize it over time.

### **36. What is XGBoost?**

**Ans:** XGBoost is known as eXtreme Gradient Boosting. It is a decision tree solution that is gradient-boosted and tailored for performance and speed. Gradient boosting machines are notoriously slow to construct due to sequential model training. These really aren't particularly scalable. Hence, XGBoost is focused on model performance and computational speed. XGBoost provides

- During training, you may parallelize tree construction by employing your CPU cores.
- Distributed computing trains huge models on a cluster of devices.
- Out-of-Core Computing is used for exceedingly big datasets that cannot be stored in memory.
- Data structure and algorithm cache optimization to maximize hardware utilization.

### **XGBoost Code Snippet:**

jupyter XGB.py 17 minutes ago

Logout Python

```
File Edit View Language
```

```
9 df.rename(columns={  
10     'preg': 'Pregnancies', 'plas': 'Glucose', 'pres': 'BloodPressure', 'skin': 'SkinThickness', 'insu': 'Insulin', 'pedi': 'DiabetesPedigreeFunction'},  
11     inplace=True)  
12 df.head()  
13  
14 # Encoding with Label Encoding  
15 df['class'] = df['class'].astype('category')  
16 df['class'] = df['class'].cat.codes  
17 df.head()  
18  
19 #Plotting Heatmap for Correlation  
20 # plt.figure(figsize=(10,10))  
21 sns.heatmap(df.corr(), annot=True)  
22  
23 #Dividing the Data into Dependent and Independent features  
24 #Dependent Variables  
25 X=df.iloc[:,0:8]  
26 X.head()  
27 # Dependent Variable  
28 Y=df['class']  
29 Y.head()  
30  
31 #Dividing the Data into Training and Test data  
32 from sklearn.model_selection import train_test_split  
33 x_train ,x_test,y_train, y_test =train_test_split(X, Y, train_size=0.75, random_state=101)  
34  
35 #Building Model  
36 # fit model to training data  
37 from xgboost import XGBClassifier  
38 xgb_model =XGBClassifier()  
39 xgb_model.fit(x_train , y_train)  
40  
41 #Predicting the model  
42 y_pred_train_xgb = xgb_model.predict(x_train)  
43 y_pred_test_xgb = xgb_model.predict(x_test)  
44  
45 #Checking the Accuracy of the model  
46 accuracy_train_xgb = accuracy_score(y_train , y_pred_train_xgb)  
47 accuracy_test_xgb = accuracy_score(y_test , y_pred_test_xgb)  
48 print('accuracy Score of train XGB: ',accuracy_train_xgb )  
49 print('accuracy Score of test XGB : ',accuracy_test_xgb )
```

## 37. What are the basic enhancements to Gradient Boosting?

**Ans:** Gradient boosting seems to be a strategy that rapidly overfits a trained model because it is greedy. It can profit from regularisation approaches that persecute different system sections and enhance overall algorithm performance by eliminating overfitting.

**Take a look at these 4 improvements to simple gradient boosting:**

1. Penalized Learning
2. Tree Constraints
3. Random sampling
4. Shrinkage

**Tree Constraints:** A nice general heuristic is that the greater constricted the single trees are, the more trees you'll require in the model; conversely, the less constricted the single trees are, the lesser trees you'll require.

**Some limits that can be imposed on decision tree building are as follows:**

- Overfitting may be exceedingly sluggish when increasing the tree numbers in the model. The recommendation is to maintain adding trees until no more improvement is shown.
- Deeper trees are more complicated, whereas shorter trees are desired. In general, 4-8 levels produce better outcomes.
- The number of leaves or nodes, such as depth, can limit the tree size but does not limit it to a symmetrical form if additional restrictions are utilized.
- The quantity of observations for each split establishes a minimal restriction on the training data number available at a training node prior to considering a split.
- The minimal improvement to lose is a constraint on the efficiency of every split introduced to a tree.

1. **Penalized Gradient Boosting:** The parametric trees and their design can be subjected to additional limitations. As weak learners, traditional decision trees such as CART are not utilized. Instead, a regression tree is utilized, which has numerical values inside the leaf nodes (also known as terminal nodes). In some literature, the values in the tree leaves are referred to as weights. L1 and L2 weights regularization may be used to regularize the trees' leaf weight values. The additional regularization factor softens the complete learning values to prevent over-fitting. Intuitively, the regularised goal will favor models with simple and predictive functions.
2. **Weighted Updates:** Each tree's predictions are combined together successively. Every tree's contribution is calculated to the sum which slows down the algorithm's learning. This weighting is referred to as a learning rate or a shrinkage.
3. **Stochastic Gradient Boosting:** Enabling trees to really be greedily constructed from subsets of said training dataset was a significant breakthrough in bagging groups and the random forest. In gradient-boosting models, the same advantage may be employed to lower the connection between the trees inside the sequence. This type of boosting is known as stochastic gradient boosting. A randomized subsample of said training data (without replacing) is obtained from the entire training dataset at each iteration. The randomly chosen subsample is then utilized to adapt the base learner rather than the entire sample.

### **38. What is Dimensionality Reduction? Why is it used?**

**Ans:** The process of transforming a dataset. That data must be transformed from data with large dimensions to data with smaller dimensions. It must also guarantee that comparable information is conveyed concisely.

Despite the fact that we employ these strategies to handle Machine Learning challenges, the challenge is to improve the regression or classification task's characteristics.

### **39. What are the commonly used Dimensionality Reduction Techniques?**

**Ans: Dimensionality reduction approaches include the following:**

- Linear Discriminant Analysis (LDA)
- Principal Component Analysis (PCA)
- Generalized Discriminant Analysis (GDA)

### **40. How does PCA work? When to apply it?**

**Ans:**

Principal Component Analysis (PCA) is a non-parametric, unstructured quantitative tool that is largely employed in ML to reduce dimensionality.

A high-dimensional dataset has a significant amount of features. In the Machine Learning area, the fundamental issue associated with large dimensionality is model overfitting, which lowers the capacity to generalize beyond the instances in the training set.

PCA in Layman's Term: Consider the 2D XY plane.

Let's think of variances like the information scatter, or the separation between the two distant locations, for the purpose of intuition.

#### **Assumption:**

In general, it is thought that datasets with a wide variation offer greater information than those with a slight variation. (This might be true or false.) This is the presumption that PCA wants to take advantages.

I give you 4 points –  $\{(1,1), (2,2), (3,3), (4,4)\}$

(all lie on the line  $X=Y$ )

What is the variance on X-axis?

$\text{Variance}(X) = 4 - 1 = 3$

What is the variance on Y-axis?

$\text{Variance}(Y) = 4 - 1 = 3$

**Can we somehow get new data with a bigger variance?**

Rotate your XY system by 45 degrees anticlockwise. What happens? The line  $X=Y$  has now become the X(new)-axis. And,  $X = -Y$  is now the Y(new)-axis. Let's compute the variance again (in the form of distance)

$\text{Variance}(X(\text{new})) = \text{distance } ((4,4), (1,1)) = \sqrt{18} = 4.24$

$\text{Variance}(Y(\text{new}))$  involves a computation.

**41. What benefit did such rotation provide for us?**

**Ans:** Original data had the highest variance on any axis as 3. This rotation gave us a variance of 4.24

That was a brief description of how PCA works. For the sake of completeness,

Eigenvalues are the variances of data along a certain axis inside the new coordinate frame.  $\text{Eigenvalue}(X(\text{new})) = 4.24$ .

Eigenvectors = the vectors that serve as the new coordinate program's representation. From previous sample, vector  $[1,1]$ = eigenvector for X(new),  $[1,-1]$ = eigenvector for Y(new). Since they are just directions – solvers typically give us unit vectors.

## Data transformation

After obtaining the eigenvectors, one may create a new position inside the current coordinate by doing a dot product of the eigenvector and the initial position..

## Steps of PCA:

1. Determine the covariance matrix  $X$  of the data points.
2. Calculate eigenvectors and corresponding eigenvalues.
3. Sort eigenvectors in decreasing order according to their provided value.
4. Initial  $k$  eigenvectors = The recent  $k$  dimensions.

5. Convert the original n-dimensional data points to k-dimensional data points.

### PCA code snippet:

The screenshot shows a Jupyter Notebook interface with a file named 'PCA.py'. The code is a script for performing PCA on the Iris dataset. It includes importing libraries, loading the dataset, handling null values, encoding labels, checking correlations, splitting data into independent and dependent variables, building a PCA model, fitting it to the training data, and finally testing the performance using a Random Forest classifier. The code is annotated with comments explaining each step.

```
1 # Principal Component Analysis Code | Author: Sarwar Ghouse Mastan
2 #Importing Libraries
3 import numpy as np
4 import pandas as pd
5 from sklearn.model_selection import train_test_split
6 from sklearn.preprocessing import StandardScaler
7 from sklearn.decomposition import PCA
8 from sklearn.ensemble import RandomForestClassifier
9 from sklearn.metrics import confusion_matrix
10 from sklearn.metrics import accuracy_score
11
12 #Loading the Dataset using pandas
13 dataset =pd.read_csv('Iris.csv')
14 dataset.head()
15
16 #Checking for null values
17 dataset.isnull().sum()
18 dataset.info()
19
20 #Removing the Index column and assignining the feaure as index
21 dataset.set_index("Id", inplace = True)
22 dataset.head()
23
24 # Encoding with Label Encoding
25 dataset['Species'] = dataset['Species'].astype('category')
26 dataset['Species'] = dataset['Species'].cat.codes
27
28 #Checking for Correlation
29 sns.heatmap(dataset.corr(), annot=True)
30 dataset = dataset.drop(['SepalWidthCm'], axis=1)
31 #Splitting the data into independent and dependent variables
32 X = dataset.iloc[:, :3]
33 Y = dataset.iloc[:, 3:4]
34
35 #Splitting the data into independent and dependent variables
36 from sklearn.model_selection import train_test_split
37 x_train ,x_test ,y_train ,y_test =train_test_split(X,Y, train_size=0.80 ,random_state=200)
38 #Building Principal Component Analysis
39 pca = PCA(n_components=2)
40 x_train = pca.fit_transform(x_train)
41 x_test = pca.transform(x_test)
42
43 #RandomForest Classifier Model
44 classifier = RandomForestClassifier(max_depth=2, random_state=9)
45 classifier.fit(x_train , y_train)
46 y_pred = classifier.predict(x_test)
47
48 #Testing performance
49 cm =confusion_matrix(y_test , y_pred)
50 print(cm, '\n\n' , 'Accuracy: ' , accuracy_score(y_test ,y_pred))
```

## 42. How does LDA work? When to use it?

**Ans:** LDA is a technique for reducing 'dimensionality' while retaining as much class-discriminating information as feasible.

### How does it work?

LDA assists you in determining the 'boundaries' of class clusters. It places your points on a plane, as much as possible separating your clusters, with every cluster possessing a relative (near) proximity to a centroid.

**Dimensionality:** Consider that you wish to divide a collection of information points into 2 dimensions into 2 categories. LDA reduces the dimensionality of your settings like so:  $K(\text{Groups}) = 2$ .  $2-1 = 1$ .

### **But why so?**

Because "at most  $K-1$ -dimensional affine subspace contains the  $K$  centroids."

### **What is the affine subspace?**

The phrase "I'm going to generalize the affine qualities of Euclidean space" is expressed in a geometric notion or architecture.

### **Now, what are those affine properties of Euclidean space?**

In three-dimensional space, a location could be represented by three coordinates. A location with two values in a two-dimensional sphere and a position with one value in a one-dimensional space ought to be capable to be represented. This two-dimensional problem's dimensionality was reduced via LDA to only one dimension. We may now begin the important task of responding to the information. We have two groups, as well as a line may connect any two places in either dimension.

Thus, a number of such data points are given to us, each with a 2D representation  $(x,y)$ . These values will be divided into categories 1 or 2 using LDA.

### **43. Explain LDA steps?**

#### **Ans: LDA steps include:**

1. For every class calculate its  $d$ -dimensional mean vector in the dataset.
2. Construct the dispersion matrix.
3. In a  $d * k$  directional matrix, select the  $k$  eigenvector having the highest eigenvalue by sorting the Eigen Vector into declining Eigen Value.
4. Through  $d * k$  eigenvector matrices, the data were projected into the resulting domain.

Multiplication of the matrix can be used to condense this.

$Y = X * W$ , where  $X$  represents the  $n$  samples as  $n * d$  dimension matrices and  $W$  represents the modified  $n * k$  dimensional samples inside the current subspace.

#### **LDA code snippet:**

---

## jupyter LDA.py 4 minutes ago

```
File Edit View Language

1 #Linear Discriminant Analysis Code / Author: Sarwar Ghouse Mastan
2 # Importing the Libraries
3 import pandas as pd
4 from sklearn.model_selection import train_test_split
5 from sklearn.preprocessing import StandardScaler
6 from sklearn.discriminant_analysis import LinearDiscriminantAnalysis as LDA
7 from sklearn.ensemble import RandomForestClassifier
8 from sklearn.metrics import confusion_matrix
9 from sklearn.metrics import accuracy_score
10
11 #importing Iris Dataset
12 dataset = pd.read_csv('path/to/data/file.csv')
13
14 X =dataset.iloc[:,0:4].values
15 Y =dataset.iloc[:,4].values
16
17 #Train Test Split
18 X_train , X_test ,y_train , y_test =train_test_split(X, y,test_size=0.2, random_state=101)
19
20 # Feature Scaling
21 sc =StandardScaler()
22 X_train = sc.fit_transform(X_train)
23 X_test = sc.transorm(X_test)
24
25 #Performing LDA
26 lda = LDA(n_components=1)
27 X_train = lda.fit_transform(X_train ,y_train)
28 X_test =lda.transorm(X_test)
29
30 #Training and making Predictions
31 classifier = RandomForestClassifier(max_depth=2 ,random_state=0)
32 classifier.fit(X_trin , y_train)
33 y_pred =classifier.predict(X_test)
34
35 #Evaluating Performance
36 cm = confusion_matrix(y_test ,y_pred)
37 print(cm ,'\n\n', 'Accuracy: ',accuracy_score(y_test ,y_pred))
```

## 44. What is GDA?

**Ans:** When we face a classification issue with constant random variables for input features, we may apply GDA, a generative learning technique that assumes  $p(x|y)$  is dispersed using a multivariate normal distribution and  $p(y)$  is dispersed as per Bernoulli.

GDA (Gaussian discriminant analysis) is a classification generative model in which each class distribution is treated as a multivariate Gaussian.

## 45. What pluses and minuses do Dimensionality Reduction offer?

**Ans:**

**Advantages:**

- Data compression is aided by dimension reduction, which results in little storage space.
- The calculation time is shortened.
- Additionally, it helps to get rid of any extraneous features.
- Dimensionality reduction aids in data compression and storage space reduction.
- It reduces the time needed to execute the same computations.
- There is less computation when there are fewer dimensions. Dimensions can also enable the use of unsuitable methods for a high number of dimensions.
- It handles multicollinearity, which enhances model performance. It gets rid of unnecessary features. For instance, it serves no purpose to hold a value in two different measurement types (in and m).
- Data reduction to 2D or 3D dimensions may allow us to plot and display it more precisely. Patterns can thus be seen more vividly.

### **Disadvantages:**

- It may result in some data loss.
- However, PCA finds linear relationships between variables, which is not always desired.
- Furthermore, PCA fails to characterize datasets when covariance and mean are inadequate.
- In addition, we may not know how many main components to keep-in practice, broad rules are followed.

## **46. What is Time series?**

**Ans:** A time series is a series of numerical data points presented in ascending order. It follows the movement of the selected data points over a predetermined time period and captures the datasets at regular intervals. There aren't any minimum or maximum time requirements for time series. Analysts frequently use time series to conduct analysis based on their individual needs.

### **Time Series Use Cases:**

- Retail demand forecasting, sourcing, and dynamic pricing

- Used in Healthcare to forecast pandemic spread, diagnostics, and prescription planning
- The price projection for customer-centric apps, as well as improved user experience

#### **47. What is a Box-Cox transformation?**

**Ans:** A Box-Cox transformation converts non-normal dependent variables into normal shapes. Normality is a key assumption for several statistical approaches; if your data isn't normal, using a Box-Cox means you can perform more tests.

#### **48. What role does maximum likelihood play in logistic regression?**

**Ans:** The maximum likelihood equation aids in estimating the most likely values of the estimator's predictor variable coefficients, yielding the most probable results and relatively near to the true values.

#### **49. Explain the chi-square test.**

**Ans:** A chi-square test examines if a data sample set corresponds to a population. A chi-square test of independence examines whether two factors in a contingency table are connected.

A relatively tiny chi-square test score shows that the observed data fit the predicted data very well.

#### **50. What exactly is the ROC curve?**

**Ans:** ROC curve (receiver operating characteristics): The ROC curve depicts a binary classifier's diagnostic capabilities. It is estimated or constructed by plotting True Positive versus False Positive at various threshold values. AUC is the ROC curve's performance statistic (area under the curve). The model's predictive power increases with the size of the area beneath the arc.

#### **1. What are the differences between Supervised and Unsupervised Learning?**

Supervised learning is a type of machine learning where a function is inferred from labeled training data. The training data contains a set of training examples.

Unsupervised learning, on the other hand, is when inferences are drawn from datasets containing input data without labeled responses.

The following are the various other differences between the two types of machine learning:

<b>Supervised Learning</b>	<b>Unsupervised Learning</b>	
Algorithms Used	Decision Trees, K-nearest Neighbor algorithm, Neural Networks, Regression, and Support Vector Machines	Anomaly Detection, Clustering, Latent Variable Models, and Neural Networks
Problems used for	Classification and regression	Classification, dimension reduction, and density estimation
Uses	Prediction	Analysis

## 2. What is Selection Bias and what are the various types?

Selection bias is typically associated with research that doesn't have a random selection of participants. It is a type of error that occurs when a researcher decides who is going to be studied. On some occasions, selection bias is also referred to as the selection effect.

In other words, selection bias is a distortion of statistical analysis that results from the sample collecting method. When selection bias is not taken into account, some conclusions made by a research study might not be accurate.

The following are the various types of selection bias:

- Sampling Bias: A systematic error resulting due to a non-random sample of a populace causing certain members of the same to be less likely to be included than others results in a biased sample
- Time Interval: A trial might end at an extreme value, usually due to ethical reasons, but the extreme value is most likely to be reached by the variable with the most variance, even though all variables have a similar mean
- Data: Results when specific data subsets are selected for supporting a conclusion or rejection of bad data arbitrarily
- Attrition: Caused due to attrition, i.e. loss of participants, discounting trial subjects or tests that didn't run to completion

## 3. What is the goal of A/B Testing?

A/B Testing is a statistical hypothesis testing meant for a randomized experiment with two variables, A and B. The goal of A/B Testing is to maximize the likelihood of an outcome of some interest by identifying any changes to a webpage.

A highly reliable method for finding out the best online marketing and promotional strategies for a business, A/B Testing can be employed for testing everything, ranging from sales emails to search ads and website copy.

#### **4. Between Python and R, which one would you pick for text analytics, and why?**

For text analytics, Python will gain an upper hand over R due for the following reasons:

- The Pandas library in Python offers easy-to-use data structures as well as high-performance data analysis tools
- Python has a faster performance for all types of text analytics

#### **5. What is the purpose of data cleaning in data analysis?**

Data cleaning can be a daunting task due to the fact that as the number of data sources grows, the time required for cleaning the data increases at an exponential rate.

This is due to the vast volume of data generated by additional sources. Data cleaning can solely take up to 80% of the total time required for carrying out a data analysis task.

Nevertheless, there are several reasons for using data cleaning in data analysis. Two of the most important ones are:

- Cleaning data from different sources helps transform the data into a format that is easy to work with
- Data cleaning increases the accuracy of a machine learning model

#### **6. Can you compare the validation set with the test set?**

A validation set is part of the training set used for parameter selection. It helps avoid overfitting the machine learning model being developed.

A test set is meant for evaluating or testing the performance of a trained machine learning model.

#### **7. What are linear regression and logistic regression?**

Linear regression is a form of statistical technique in which the score of some variable Y is predicted on the basis of the score of a second variable X, referred to as the predictor variable. The Y variable is known as the criterion variable.

Also known as the logit model, logistic regression is a statistical technique for predicting the binary outcome from a linear combination of predictor variables.

#### **8. Explain Recommender Systems and state an application.**

Recommender Systems is a subclass of information filtering systems, meant for predicting the preferences or ratings awarded by a user to some product.

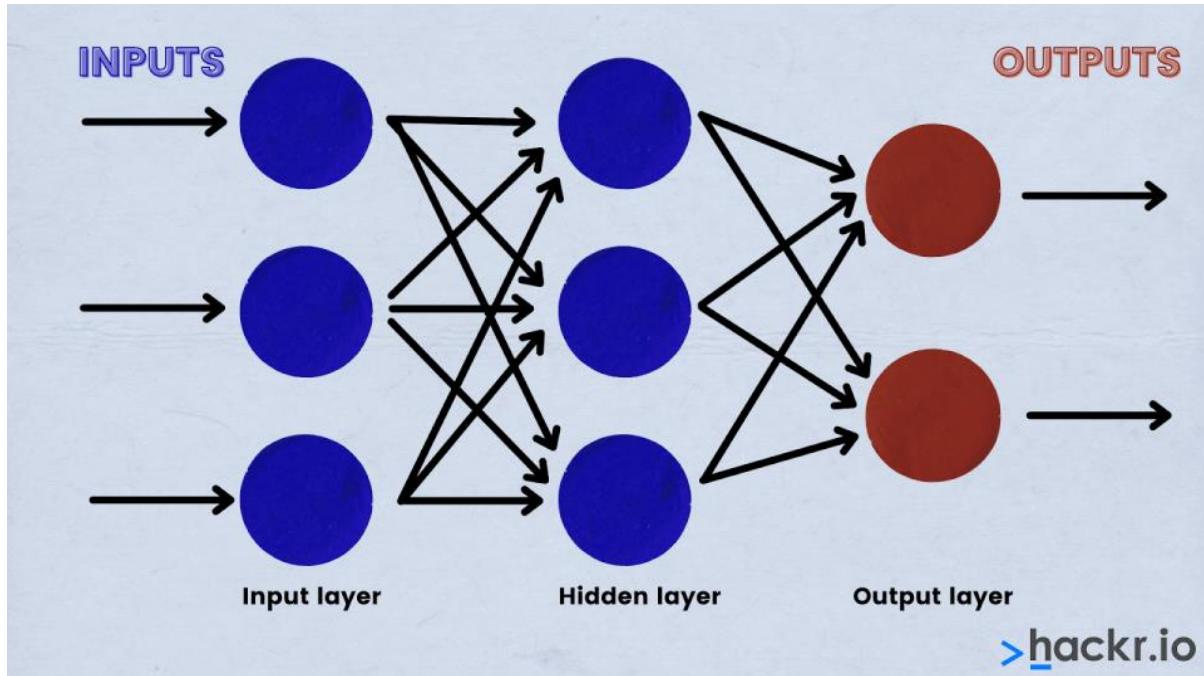
An application of a recommender system is the product recommendations section in Amazon. This section contains items based on the user's search history and past orders.

## 9. What are the steps involved in an analytics project?

The following are the numerous steps involved in an analytics project:

- Understanding the business problem
- Exploring the data and understanding it
- Preparing the data for modeling by means of detecting outlier values, transforming variables, treating missing values, et cetera
- Running the model and analyzing the result for making appropriate changes or modifications to the model (an iterative step that repeats until the best possible outcome is reached)
- Validating the model using a new dataset
- Implementing the model and tracking the result for analyzing the performance of the same

## 10. What is Deep Learning?



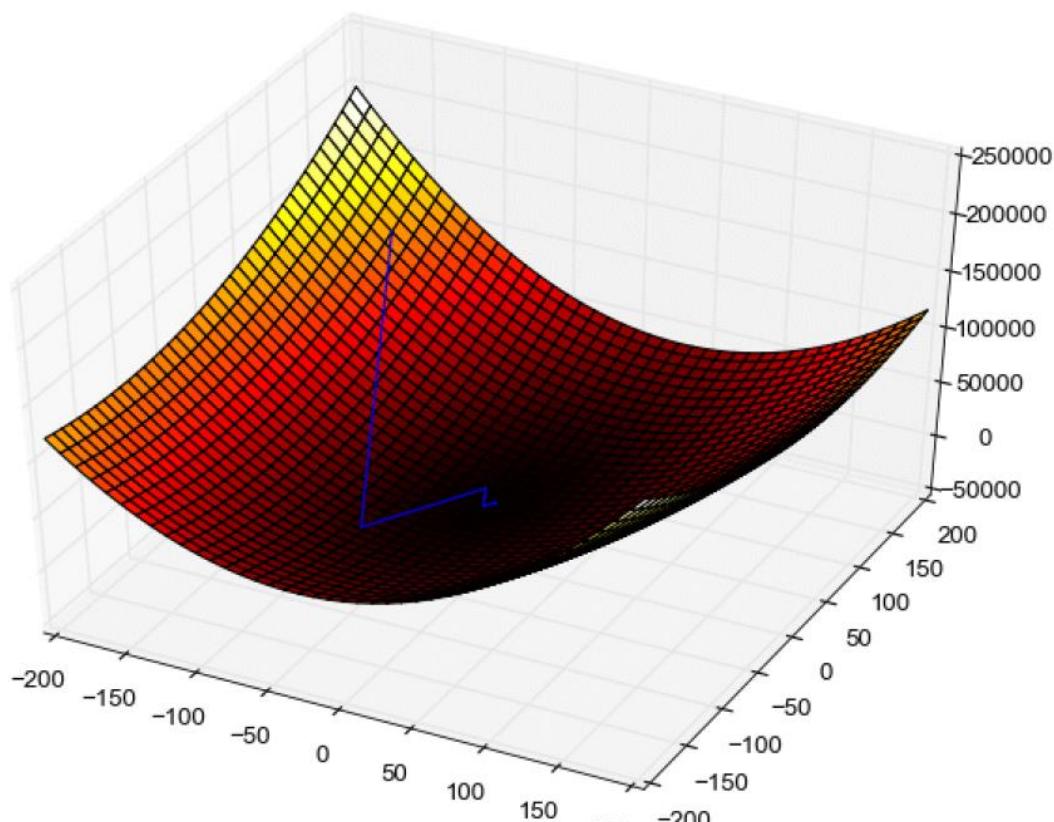
Deep Learning is a paradigm of machine learning that resembles, to a certain extent, the functioning of the human brain. It is a neural network method based on convolutional neural networks (CNN).

Deep learning has a wide array of uses, ranging from social network filtering to medical image analysis and speech recognition. Although Deep Learning has existed for a long time, it's only recently gained worldwide exposure. This is mainly due to:

- An increase in the amount of data generation
- The growth in hardware resources required for running Deep Learning models

Caffe, Chainer, Keras, Microsoft Cognitive Toolkit, Pytorch, and TensorFlow are some of the most popular Deep Learning frameworks.

## 11. What is Gradient Descent?



The gradient descent algorithm is represented by the blue line

In simple terms, gradient descent is a mathematical function that makes its way down to the bottom of a valley. It is a minimization algorithm meant for minimizing a given activation function.

The degree of change in the output of a function with respect to the changes made to the inputs is known as a gradient. It measures the change in all weights with respect to the change in error. A gradient can also be comprehended as the slope of a function.

## **12. What skills are important to become a Data Scientist?**

The skills required to become a certified Data Scientist include:

1. Knowledge of built-in data types including lists, tuples, sets, and related.
2. Expertise in N-dimensional NumPy Arrays.
3. Ability to apply Pandas Dataframes.
4. Strong holdover performance in element-wise vectors.
5. Knowledge of matrix operations on NumPy arrays.

## **13. What are the skills a Data Scientist requires, with respect to Python data analysis?**

The skills required as a Data Scientist that would help in using Python for data analysis purposes are:

- Understanding Pandas Dataframes, Scikit-learn, and N-dimensional NumPy Arrays.
- Knowing how to apply element-wise vector and matrix operations on NumPy arrays.
- Understanding built-in data types, including tuples, sets, dictionaries, and so on
- Knowing Anaconda distribution and the Conda package manager
- Writing efficient list comprehensions, small, clean functions, and avoiding traditional for loops
- Knowledge of Python script and optimizing bottlenecks

## **14. Why is TensorFlow considered important in Data Science?**

TensorFlow is considered a high priority when learning Data Science because it provides support for languages such as C++ and Python. As such, several data science processes benefit from faster compilation and completion, compared to the Keras and Torch libraries. TensorFlow also supports the CPU and GPU for faster inputs, editing, and analysis of the data.

## **15. What is Dropout?**

Dropout is a tool in data science, which is used for dropping out hidden and visible units of a network on a random basis. They prevent overfitting of the data by dropping as much as 20% of the nodes so that the required space can be arranged for iterations needed to converge the network.

## **16. What are the various Machine Learning Libraries and their benefits?**

The various machine learning libraries and their benefits are as follows.

- Numpy: Used for scientific computation

- Statsmodels:Used for time-series analysis
- Pandas:Used for tubular data analysis
- Scikit learns:Used for data modeling and pre-processing
- TensorFlow:Used for deep learning
- Regular Expressions:Used for text processing
- Pytorch:Used for deep learning
- NLTK:Used for text processing

## **17. State some Deep Learning Frameworks.**

Some Deep Learning frameworks are:

- Caffe
- Keras
- TensorFlow
- Pytorch
- Chainer
- Microsoft Cognitive Toolkit

## **18. What is an Epoch?**

Epoch in data science represents one iteration over the entire dataset. It includes everything that is applied to the learning model.

## **19. What is a Batch?**

A batch is a series of broken-down collections of the data set, which help pass the information into the system. It is used when the developer cannot pass the entire dataset into the neural network at once.

## **20. What is an iteration? State an example.**

An iteration is a classification of the data into different groups, applied within an epoch.

For example, when there are 50,000 images, and the batch size is 100, the Epoch will run about 500 iterations.

## **21. What is the cost function?**

Cost functions are a tool to evaluate how good the model's performance is. It takes into consideration the errors and losses made in the output layer during the backpropagation process. In such a case, the errors are moved backward in the neural network, and various other training functions are applied.

## **22. What are hyperparameters?**

Hyperparameters are a kind of parameter whose value is set before the learning process so that the network training requirements can be identified and the

structure of the network improved. This process includes recognizing hidden units, learning rate, and epochs, among other things.

### **23. What are the differences between Deep Learning and Machine Learning?**

Yes, there are differences between Deep Learning and Machine Learning. These are:

Deep Learning	Machine Learning
<p>It gives computers the ability to learn without being explicitly programmed</p>	<p>It gives computers a limited to unlimited ability wherein nothing major can be done without getting programmed, and many things can be done without prior programming. It includes supervised, unsupervised, and reinforcement machine learning processes.</p>
<p>It is a subcomponent of machine learning that is concerned with algorithms that are inspired by the structure and functions of human brains, called the Artificial Neural Networks</p>	<p>It includes Deep Learning as one of its components</p>

## Technical Data Science Interview Questions

### **24. Explain overfitting and underfitting.**

In order to make reliable predictions on untrained data in machine learning and statistics, it is required to fit a model to a set of training data. Overfitting and underfitting are two of the most common modeling errors that occur while doing so.

A statistical model suffering from overfitting relates to some random error or noise in place of the underlying relationship. When a statistical model or machine learning algorithm is excessively complex, it can result in overfitting. An example of a complex model is one having too many parameters when compared to the total number of observations.

When underfitting occurs, a statistical model or machine learning algorithm fails in capturing the underlying trend of the data. Underfitting occurs when trying to fit a linear model to non-linear data.

Although both overfitting and underfitting yield poor predictive performance, the way in which each one of them does so is different. While the overfitted model overreacts to minor fluctuations in the training data, the underfit model under-reacts to even bigger fluctuations.

## **25. What is batch normalization?**

Batch normalization is a technique through which attempts could be made to improve the performance and stability of the neural network. This can be done by normalizing the inputs in each layer so that the mean output activation remains 0 with the standard deviation at 1.

## **26. What do you mean by cluster sampling and systematic sampling?**

Studying the target population spread throughout a wide area can become difficult. Applying simple random sampling becomes ineffective, the technique of cluster sampling is used. A cluster sample is a probability sample, in which each of the sampling units is a collection or cluster of elements.

Following the technique of systematic sampling, elements are chosen from an ordered sampling frame. The list is advanced in a circular fashion. This is done in such a way so that once the end of the list is reached, the same is progressed from the start, or top, again.

## **27. What are Eigenvectors and Eigenvalues?**

Eigenvectors help in understanding linear transformations. They are calculated typically for a correlation or covariance matrix in data analysis. In other words, eigenvectors are those directions along which some particular linear transformation acts by compressing, flipping, or stretching.

Eigenvalues can be understood either as the strengths of the transformation in the direction of the eigenvectors or the factors by which the compressions happen.

## **28. What are outlier values and how do you treat them?**

Outlier values, or simply outliers, are data points in statistics that don't belong to a certain population. An outlier value is an abnormal observation that is very much different from other values belonging to the set. Not all extreme values are outlier values.

Identification of outlier values can be done by using univariate analysis, or some other graphical analysis method. Few outlier values can be assessed individually but assessing a large set of outlier values requires the substitution of the same with either the 99th or the 1st percentile values.

There are two popular ways of treating outlier values:

1. To change the value so that it can be brought within a range
2. To simply remove the value

## 29. How do you define the number of clusters in a clustering algorithm?

The primary objective of clustering is to group together similar identities in such a way that while entities within a group are similar to each other, the groups remain different from one another.

Generally, the Within Sum of Squares is used for explaining the homogeneity within a cluster. For defining the number of clusters in a clustering algorithm, WSS is plotted for a range pertaining to a number of clusters. The resultant graph is known as the Elbow Curve.

The Elbow Curve graph contains a point that represents the point post in which there aren't any decrements in the WSS. This is known as the bending point and represents K in K-Means.

Although the aforementioned is the widely-used approach, another important approach is hierarchical clustering. In this approach, dendograms are created first and then distinct groups are identified from there.

30. How does backpropagation work? States the variants.

Backpropagation refers to a training algorithm used for multilayer neural networks. Following the backpropagation algorithm, the error is moved from an end of the network to all weights inside the network. Doing so allows for efficient computation of the gradient.

Backpropagation works in the following way:

- Forward propagation of training data
- Output and target is used for computing derivatives
- Back propagate for computing the derivative of the error with respect to the output activation
- Using previously calculated derivatives for output generation
- Updating the weights

The following are the various variants of backpropagation:

- Batch Gradient Descent: The gradient is calculated for the complete dataset and an update is performed on each iteration
- Mini-batch Gradient Descent: Mini-batch samples are used for calculating gradient and updating parameters (a variant of the Stochastic Gradient Descent approach)
- Stochastic Gradient Descent: Only a single training example is used to calculate gradient and update parameters

### **31. What do you know about Autoencoders?**

Autoencoders are simplistic learning networks used for transforming inputs into outputs with minimal possible error. It means that the output's results are very close to the inputs.

A couple of layers are added between the input and the output with the size of each layer smaller than the size pertaining to the input layer. An autoencoder receives unlabeled input that is encoded for reconstructing the output.

### **32. Please explain the concept of a Boltzmann Machine.**

A Boltzmann Machine features a simple learning algorithm that enables it to discover fascinating features representing complex regularities present in the training data. It is basically used for optimizing the quantity and weight for some given problem.

The simple learning algorithm involved in a Boltzmann Machine is very slow in networks that have many layers of feature detectors.

### **33. What is GAN?**

The Generative Adversarial Network takes inputs from the noise vector and sends them forward to the Generator, and then to Discriminator, to identify and differentiate unique and fake inputs.

### **34. What are the components of GAN?**

There are two vital components of GAN. These are:

1. Generator: The Generator acts as a Forger, which creates fake copies
2. Discriminator: The Discriminator acts as a recognizer for fake and unique (real) copies

### **35. What is the Computational Graph?**

A computational graph is a graphical presentation that is based on TensorFlow. It has a wide network of different kinds of nodes wherein each node represents a particular mathematical operation. The edges in these nodes are called tensors. This is the reason the computational graph is called a TensorFlow of inputs. The computational graph is characterized by data flows in the form of a graph; therefore, it is also called the DataFlow Graph.

### **36. What are tensors?**

Tensors are mathematical objects that represent the collection of higher dimensions of data inputs in the form of alphabets, numerals, and rank fed as inputs to the neural network.

### 37. What is the difference between Batch and Stochastic Gradient Descent?

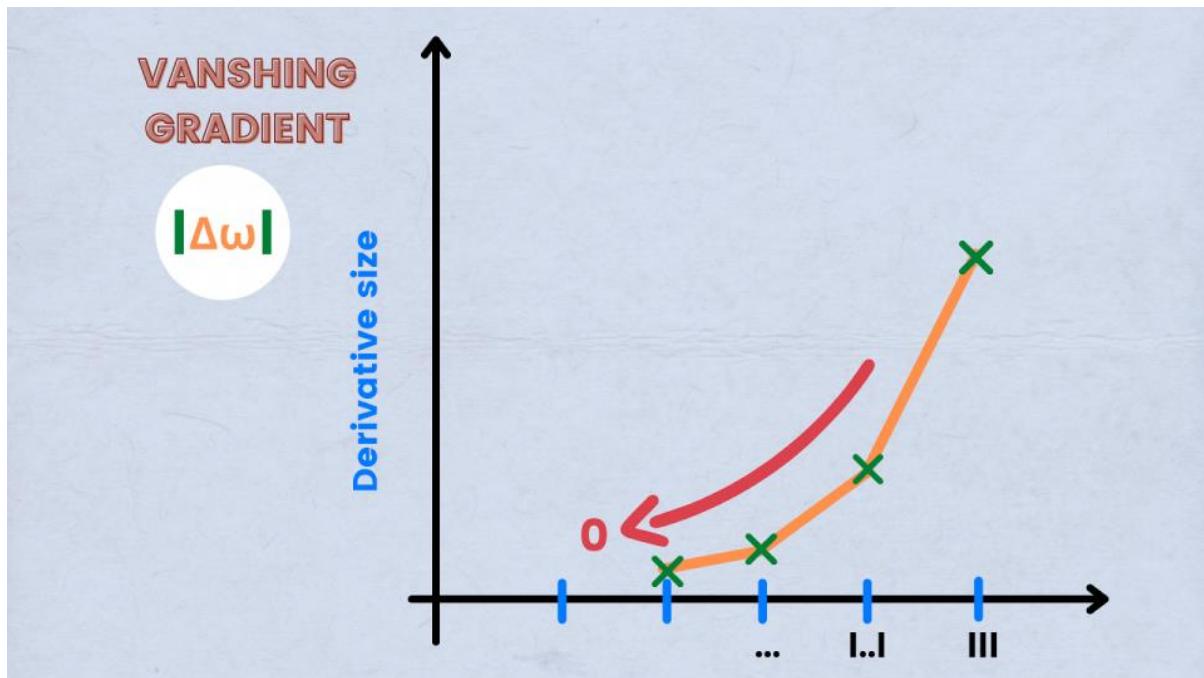
The difference between Batch and Stochastic Gradient Descent are as follows:

Batch Gradient Descent	Stochastic Gradient Descent
Helps in computing the gradient using the complete data set	Helps in computing the gradient using only single sample
Takes time to converge	Takes less time to converge
The volume is large for analysis purposes	The volume is lower for analysis purposes
Updates the weight comparatively infrequently	Updates the weight more frequently

### 38. What is an Activation function?

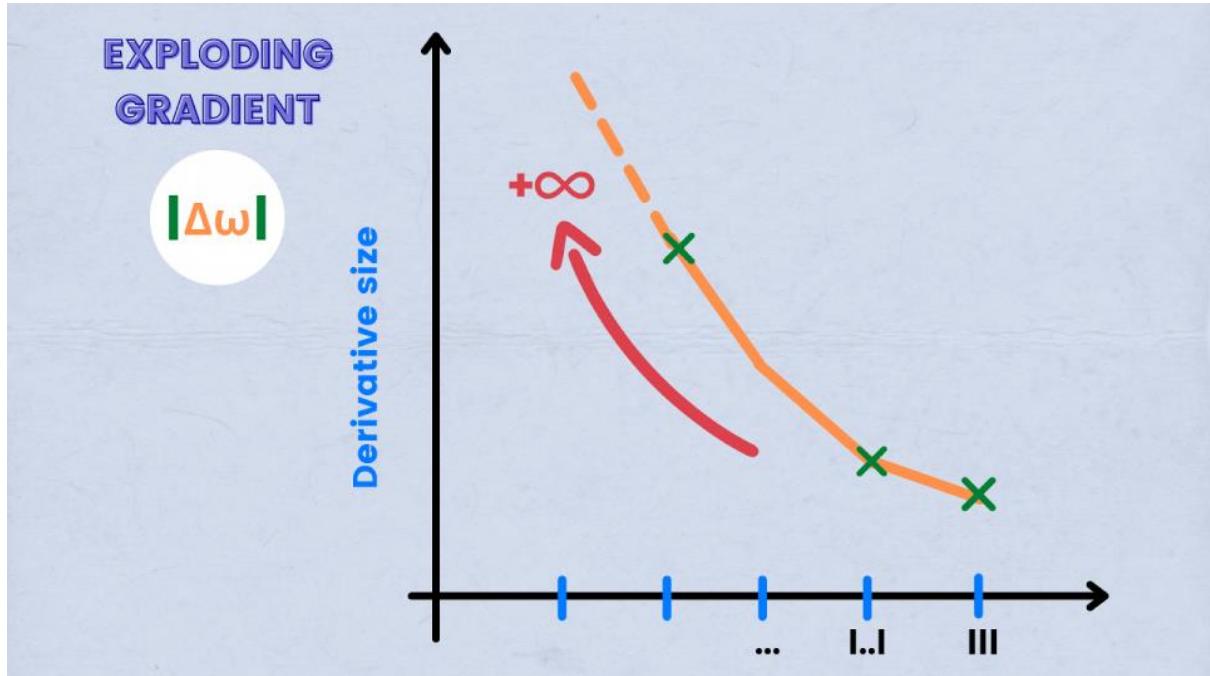
An Activation function helps introduce non-linearity in the neural network. This is done to help the learning process when it comes to complex functions. Without the activation function, the neural network will be unable to perform only the linear function and apply linear combinations. Activation function, therefore, offers complex functions and combinations by applying artificial neurons, which helps in delivering output based on the inputs.

### 39. What are vanishing gradients?



A vanishing gradient is a condition when the slope is too small during the training of Recurrent Neural Networks. The result of vanishing gradients is poor performance outcomes, low accuracy, and long-term training processes.

#### 40. What are exploding gradients?



The exploding gradient is a condition when the errors grow at an exponential rate or high rate during the training of Recurrent Neural Networks. This error gradient accumulates and results in applying large updates to the neural network, causes an overflow, and results in NaN values.

#### 41. What is the full form of LSTM? What is its function?

LSTM stands for Long Short Term Memory. It is a recurrent neural network that is capable of learning long term dependencies and recalling information for a longer period as part of its default behavior.

#### 42. What are the different steps in LSTM?

The different steps in LSTM include the following.

- Step 1: The network helps decide what needs to be remembered and forgotten
- Step 2: The selection is made for cell state values that can be updated
- Step 3: The network decides as to what can be made as part of the current output

#### 43. What is Polling in CNN?

Pooling is a method that is used to reduce the spatial dimensions of a CNN. It helps downsample operations for reducing dimensionality and creating pooled feature maps. Pooling in CNN helps in sliding the filter matrix over the input matrix.

#### **44. What is RNN?**

Recurrent Neural Networks are an artificial neural network that is a sequence of data, including stock markets, sequence of data including stock markets, time series, and various others. The main idea behind the RNN application is to understand the basics of the feedforward nets.

#### **45. What are the different layers on CNN?**

There are four different layers on CNN. These are:

1. Convolutional Layer: In this layer, several small picture windows are created to go over the data
2. ReLU Layer: This layer helps in bringing non-linearity to the network and converts the negative pixels to zero so that the output becomes a rectified feature map
3. Pooling Layer: This layer reduces the dimensionality of the feature map
4. Fully Connected Layer: This layer recognizes and classifies the objects in the image

#### **46. What is an Artificial Neural Network?**

Artificial Neural Networks is a specific set of algorithms that are inspired by the biological neural network meant to adapt the changes in the input so that the best output can be achieved. It helps in generating the best possible results without the need to redesign the output methods.

#### **47. What is Ensemble learning?**

Ensemble learning is a process of combining the diverse set of learners that are the individual models. It helps in improving the stability and predictive power of the model.

#### **48. What are the different kinds of Ensemble learning?**

The different kinds of ensemble learning are:

1. Bagging: It implements simple learners on one small population and takes mean for estimation purposes
2. Boosting: It adjusts the weight of the observation and thereby classifies the population in different sets before the outcome prediction is made

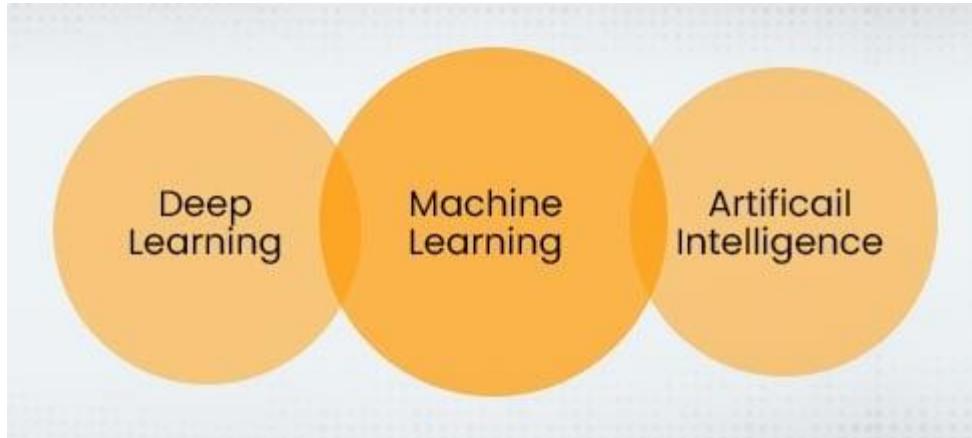
- What is Bias Error in machine learning algorithm?
- What do you understand about Variance Error in machine learning algorithms?
- What is the bias-variance trade-off?
- How will you differentiate supervised and unsupervised machine learning?
- How is the k-nearest algorithm different from the KNN clustering?
- What is ROC (Receiver operating characteristic) Curve? Explain the working of ROC.
- What do you mean by precision and recall?
- What is the significance of Bayes' theorem in the context of the machine learning algorithm?
- What is Naïve Bayes in machine learning?
- How will you differentiate the L1 and L2 regularization?
- What is your favorite algorithm? Explain in less than a minute based on your past experiences.
- Have you ever worked on type 1 or Type 2 errors?
- How will you explain the Fourier Transformation in Machine Learning?
- How will you differentiate machine learning and deep learning algorithms?
- How will you differentiate the generic model from the discriminative model?
- What seems more important is either model accuracy or performance of a model?
- What is the F1 score and explain its uses too?
- Is it possible to manage imbalanced datasets in machine learning?
- Why is classification better than regression for machine learning experts?
- How would you check the effectiveness of a machine learning model?

Q1). Explain Machine Learning, Artificial learning, and Deep learning in brief?

It is very common to get confused between the three in-demand technologies: Machine Learning, Artificial Intelligence, and Deep Learning. It is because these three technologies, though they are a little different from one another and are interrelated to each other.

While Deep Learning is a subset of Machine Learning and Machine Learning is a subset of Artificial Intelligence which you can clearly understand in the below-

mentioned image. Since some terms and techniques may overlap with each other while dealing with these technologies, it is easy to get confused between them.



Therefore, let's go through about these technologies in detail so that you become capable of differentiating between them:

- **Machine Learning:** Machine Learning includes multiple statistical and Deep Learning techniques that allow machines to use their past exposures and get better at performing particular tasks without being monitored.
- **Artificial Intelligence:** Artificial Intelligence uses multiple Machine Learning and Deep Learning techniques that enable computer systems to perform tasks using human intelligence, with logic and rules.
- **Deep Learning:** Deep Learning consists of multiple algorithms that enable software to learn from themselves and perform multiple business tasks, including image and speech recognition. Moreover, it is possible when the systems expose their multi-layered neural networks to large volumes of data for learning.

Q2). What is Bias Error in machine learning algorithm?

Bias is the common error in the machine learning algorithm due to simplistic assumptions. It may undermine your data and does not allow you to achieve maximum accuracy. Further generalizing the knowledge from the training set to the test sets would be highly difficult for you.

Q3). What do you understand about Variance Error in machine learning algorithms?

Variance error is common in machine learning when the algorithm is highly complex and difficult to understand as well. It may lead to a high degree of variation to your training data that can lead the model to overfit the data. Also,

there could be so much noise for the training data that is not necessary in case of the test data.

#### Q4). What is the bias-variance trade-off?

The bias-variance trade-off is able to handle the learning errors effectively and manages noise too that happens due to underlying data. Essentially, this trade-off will make the model more complex than usual but errors are reduced optimally.

#### Q5). How will you differentiate supervised and unsupervised machine learning?

Here is the difference between supervised and unsupervised machine learning that you can consider before going on a Machine Learning Interview:

- Supervised learning: Algorithms of supervised learning use labeled data to get trained and the models take direct feedback to confirm whether the output is, indeed, correct. Moreover, both the input data and the output data are provided to the model, and the main aim here is to train the model efficiently to predict the output when it receives new data. However, it can largely be divided into two parts, classification and regression which help a person to offer accurate results.
- Unsupervised learning: Unsupervised learning algorithms use unlabeled data for training purposes. In this, the models do not take any feedback unlike the case of supervised learning. However, these models identify hidden data trends from the models. The unsupervised learning model is usually provided with the input data, and its main aim is to identify hidden patterns to extract information from the unknown sets of data. It can also be classified into two main parts, namely, clustering and associations. Unfortunately, unsupervised learning offers outcomes that are comparatively less accurate.

#### Q6). How is the k-nearest algorithm different from the KNN clustering?

K-nearest algorithm is the supervised learning while the k-means algorithm is assigned under the unsupervised learning. While these two techniques look similar initially, still there is a lot of difference between the two Supervised learning requirements data in the labeled form.

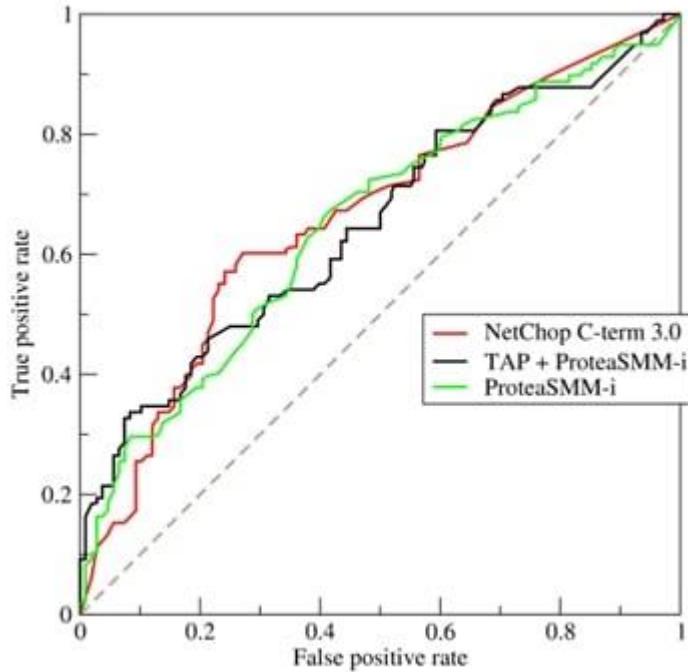
K-Nearest Neighbour	K-Means Clustering
Supervised Technique	Unsupervised Technique
Used for Classification or regression	Used for Clustering
'K' in KNN represents the number of nearest neighbours used to classify or predict in case of continuous variable/ regression	'K' in K-Means represents the number of cluster the algorithm is trying to identify of learn from the data.



For example, if you wanted to classify the data then you should first label the data then further classify it into different groups. On the other hand, unsupervised does not require any data labeling explicitly. The application of both the techniques also depends on project requirements.

Q7). What is ROC (Receiver operating characteristic) Curve? Explain the working of ROC.

Receiver Operating Characteristic curve (or ROC curve) is a fundamental tool used for diagnostic test evaluation and pictorial representation of the contrast between true positive rates and the false positive rates calculated at multiple thresholds. It is used as the proxy to measure the trade-offs and sensitivity of the model. Based on the observation, it will trigger false alarms.



- It shows the tradeoff between sensitivity and specificity (any increase in sensitivity will be accompanied by a decrease in specificity).
- The closer the curve follows the left-hand border and then the top border of the ROC space, the more accurate the test.
- The closer the curve comes to the 45-degree diagonal of the ROC space, the less accurate the test.
- The slope of the tangent line at a cut point gives the likelihood ratio (LR) for that value of the test.
- The area under the curve is a measure of test accuracy.

Q8). What do you mean by precision and recall?

The Recall is the measure of true positive rates claimed against the total number of datasets. Precision is the prediction of positive values that your model claims compared to the number of positives it actually claims. It can be taken as a special case of probability as well in the case of mathematics.

Q9). What is the significance of Bayes' theorem in the context of the machine learning algorithm?

With the Bayes' Theorem, you could measure the posterior probability of an event based on your prior knowledge. In mathematical terms, it will tell you the exact positive rate of a condition i.e. divided by the sum of total false rates of the entire population.

$$P(A|B) = \frac{P(A|B) = P(B|A) \cdot P(A)}{P(B)}$$

Probability of A occurring

Probability of B occurring given evidence a has already occurred.

Probability of B occurring given evidence a has already occurred.

Probability of A occurring



Bayes Theorem is also known as the Bayes Rule in mathematics, and it is popular for calculating the conditional probability. The name of the theorem was given after a popular mathematician Thomas Bayes. The two of the most significant applications of the Bayes' theorem in Machine Learning are Bayesian optimization and Bayesian belief networks. This theorem is also considered as the foundation behind the Machine Learning brand that includes the Naive Bayes classifier.

Q10). What is Naïve Bayes in machine learning?

Naïve is the word used to define the things that are virtually impossible in the real-life. Here, also you require to calculate the conditional probability as the product of individual probabilities of different components.

The Naive Bayes method is a supervised learning algorithm, it is naive since it makes assumptions by applying Bayes' theorem that all attributes are independent of each other. Bayes' theorem states the following relationship, given class variable y and dependent vector x<sub>1</sub> through x<sub>n</sub>:

$$P(y_i / x_1, \dots, x_n) = P(y_i)P(x_1, \dots, x_n / y_i)(P(x_1, \dots, x_n))$$

Using the naive conditional independence assumption that each x<sub>i</sub> is independent: for all i this relationship is simplified to:

$$P(x_i / y_i, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n) = P(x_i / y_i)$$

Since, P(x<sub>1</sub>, ..., x<sub>n</sub>) is a constant given the input, we can use the following classification rule:

$P(y_i / x_1, \dots, x_n) = P(y) \prod_{i=1}^n P(x_i | y_i) P(x_1, \dots, x_n)$  and we can also use Maximum A Posteriori (MAP) estimation to estimate  $P(y_i)$  and  $P(y_i | x_i)$  the former is then the relative frequency of class  $y$  in the training set.

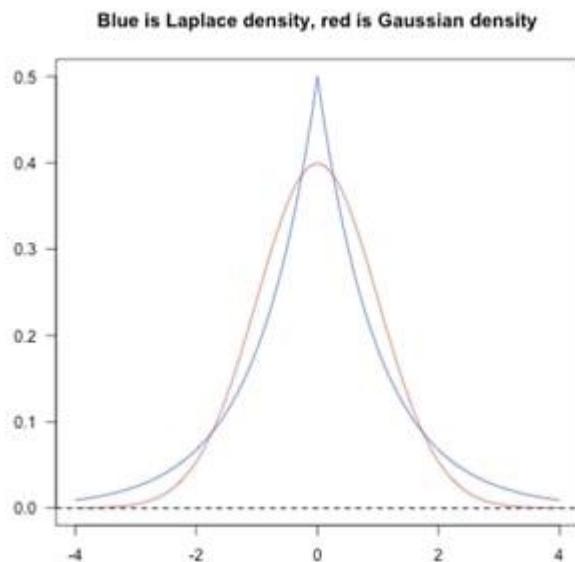
$$P(y_i / x_1, \dots, x_n) = P(y) \prod_{i=1}^n P(x_i | y_i)$$

$$y = \arg \max P(y_i) \prod_{i=1}^n P(x_i | y_i)$$

The different naive Bayes classifiers mainly differ by the assumptions they make regarding the distribution of  $P(y_i | x_i)$ : can be Bernoulli, binomial, Gaussian, and so on.

Q11). How will you differentiate the L1 and L2 regularization?

L2 regularization tends to spread error among multiple terms while L1 is more specific to binary variables where either 0 or 1 is assigned based on requirements. L1 tends to set a Laplacian prior on terms, but L2 tends to set a Gaussian prior on terms.



Q12). What is your favorite algorithm? Explain in less than a minute based on your past experiences.

The answer to this question will vary based on the projects you worked on earlier. Also, which algorithm assured better outcomes as compared to others?

Q13). Have you ever worked on type 1 or Type 2 errors?

This is a tricky question usually asked by experienced candidates only. If you would be able to answer this question then make sure that you are at the top of

the game. Type 1 error is the false positive and Type 2 error is a false negative. Type 1 error signifies something has happened even if it does not exist in real life while Type 2 error means you claim something is happening in real life. Here is a small difference between Type 1 and Type 2 error:

 Type I Error	Type II Error
<ul style="list-style-type: none"> <li>• Type I error is a false positive.</li> <li>• Type I error is claiming something has happened when it hasn't.</li> </ul>	<ul style="list-style-type: none"> <li>• Type II error is a false negative.</li> <li>• Type II error is claiming nothing when in fact something has happened.</li> </ul>

**Q14). How will you explain the Fourier Transformation in Machine Learning?**

A Fourier Transformation is the generic method that helps in decomposing functions into a series of symmetric functions. It helps you in finding the set of cycle speeds, phases, and amplitude to match the particular time signal. It has the capability to convert the signal into frequency domain like sensor data or more.

**Q15. What is bagging and boosting in Machine Learning?**

similarities between Bagging and Boosting in Machine learning	Difference between Bagging and Boosting in Machine learning
• Both are ensemble methods to get N learners from 1 learner.	• While they are built independently for Bagging, Boosting tries to add new models that do well where previous models fall.
• Both generate several training data sets by random sampling	• Only Boosting determines weight for the data to tip the scales in favour of the most difficult cases.
• Both make the final decision by taking the average of N learners	• Is an equally average for Bagging and a weighted average for Boosting, giving more weight to those with better performance on training data.
• Both are good at reducing variance and proving higher scalability	• Only Boosting tries to reduce bias. On the other hand, Bagging may solve the problem of over-fitting, while boosting can increase it.

**Q16). How will you differentiate machine learning and deep learning algorithms?**

Deep learning is a part of machine learning that is usually connected with the neural networks. This is a popular technique from neuroscience to model a set of labeled and structured data more precisely. In brief, deep learning is an unsupervised learning algorithm that represents data with the help of neural nets.

Q17). How will you differentiate the generic model from the discriminative model?

A generic model will explain the multiple categories of data while the discriminative model simply tells the difference between data categories. They are used in classification tasks and need to be studied deeply before you actually implement them.

Q18) What is cross-validation in Machine Learning?

The cross-validation method in Machine Learning allows a system to enhance the performance of the given Machine Learning algorithm to which you feed various sample data from the dataset. This sampling process is done to break the dataset into smaller parts that have the same number of rows, out of which a random part is selected as a test set, and the rest of the parts are kept as train sets. Cross-validation includes the following techniques:

- Holdout method
- K-fold cross-validation
- Stratified k-fold cross-validation
- Leave p-out cross-validation

Q19). What seems more important is either model accuracy or performance of a model?

Well, model accuracy is just a subset of the model performance parameter. For a model who is performing excellent, there are chances of more accuracy than others.

Q20). What is the F1 score and explain its uses too?

Let's go through the below-mentioned model before directly jumping onto F1 score:

Prediction	Predicted Yes	Pr
Actual Yes	True Positive (TP)	False Negati
Actual No	False Positive (FP)	True Negati

In binary classification we consider the F1 score to be a measure of the model's accuracy. The F1 score is a weighted average of precision and recall scores.

$$F1 = \frac{2TP}{2TP + FP + FN}$$

Now, let's learn about the F1 score, which is used to check the performance of a model or this is the average of precision and recall of a model where 1 means the best and 0 means the worst.

Q21). Is it possible to manage imbalanced datasets in machine learning?

Collect more data, manage the imbalanced data, try a different algorithm to work on imbalanced datasets in machine learning.

Q22). Why is classification better than regression for machine learning experts?

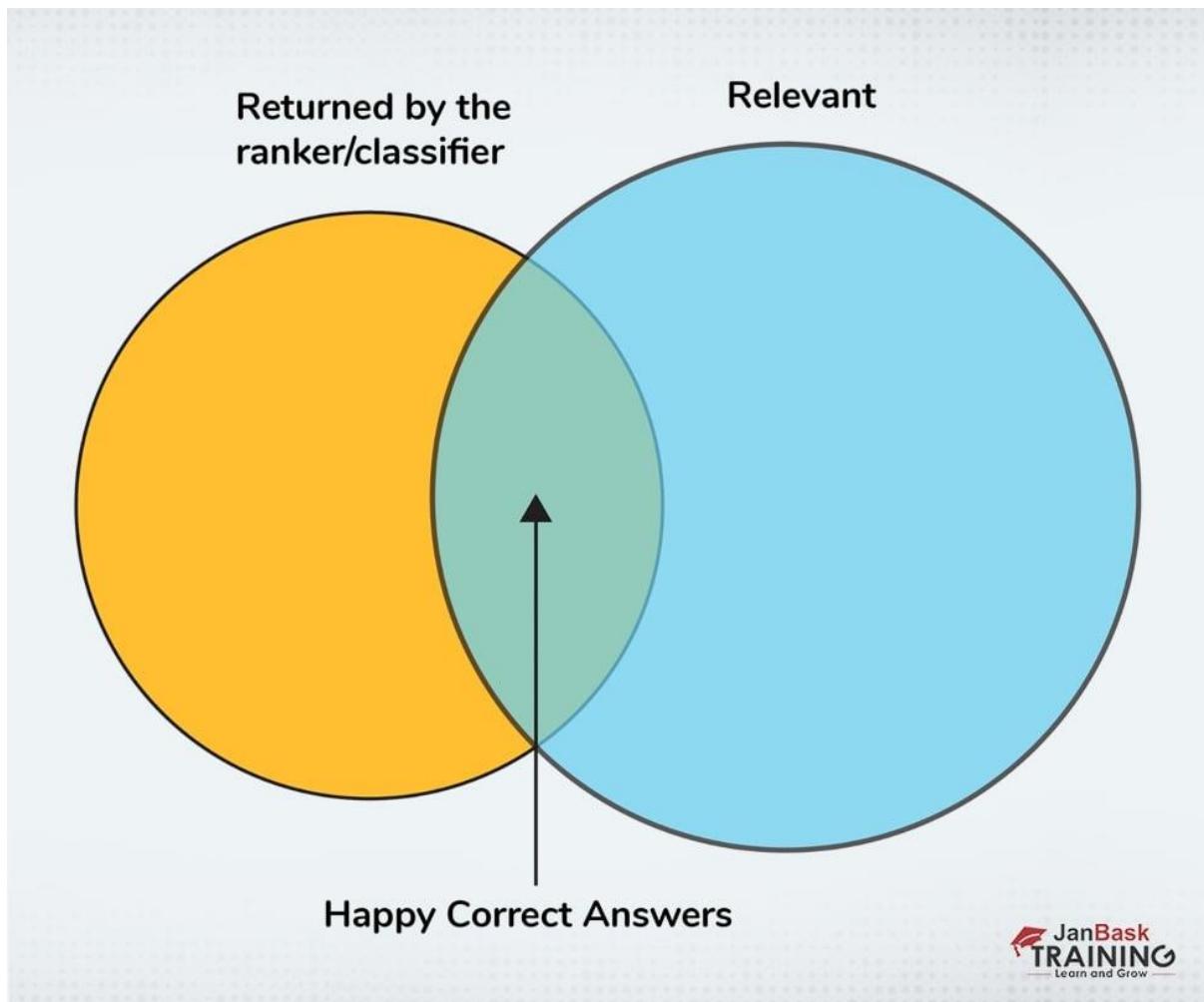
Classification gives you discrete results while regression works on continuous results more. To become more specific with data points, you are always recommended using classification over regression in machine learning.

Q23). How would you check the effectiveness of a machine learning model?

For this purpose, you can always check the F1 score to make sure either machine learning model is working effectively or needs improvement. All the best and Happy job hunting!

Precision and recall are the two different ways of monitoring the power of machine learning implementation. They are mostly used at the same time. Precision answers the question, "Out of the items that the classifier predicted to be relevant, how many are truly relevant?" Whereas, recall answers the question, "Out of all the items that are truly relevant, how many are discovered by the classifier? The basic meaning of precision is the fact of being exact and accurate. So the same will be followed in the machine learning model as well. If you have a set of items that your model requires to predict to be relevant.

The below figure shows the Venn diagram with precision and recall.



Precision and recall

**Mathematically, precision and recall can be defined as the following:**

- precision = # happy correct answers/# total items returned by ranker
- recall = # happy correct answers/# total relevant answers

Q25). How do you ensure which Machine Learning Algorithm to use?

It fully depends on the dataset you have and if the data is discrete then you may use SVM. In case the dataset is continuous then you can use linear regression. So there is no particular way that lets us know which Machine Learning algorithm to use, it all depends on the exploratory data analysis (EDA).

EDA is like “interviewing” the dataset; As part of our interview you may do the following:

- Classify the variables as continuous, categorical, and so forth.
- Summarize the variables utilizing descriptive statistics.

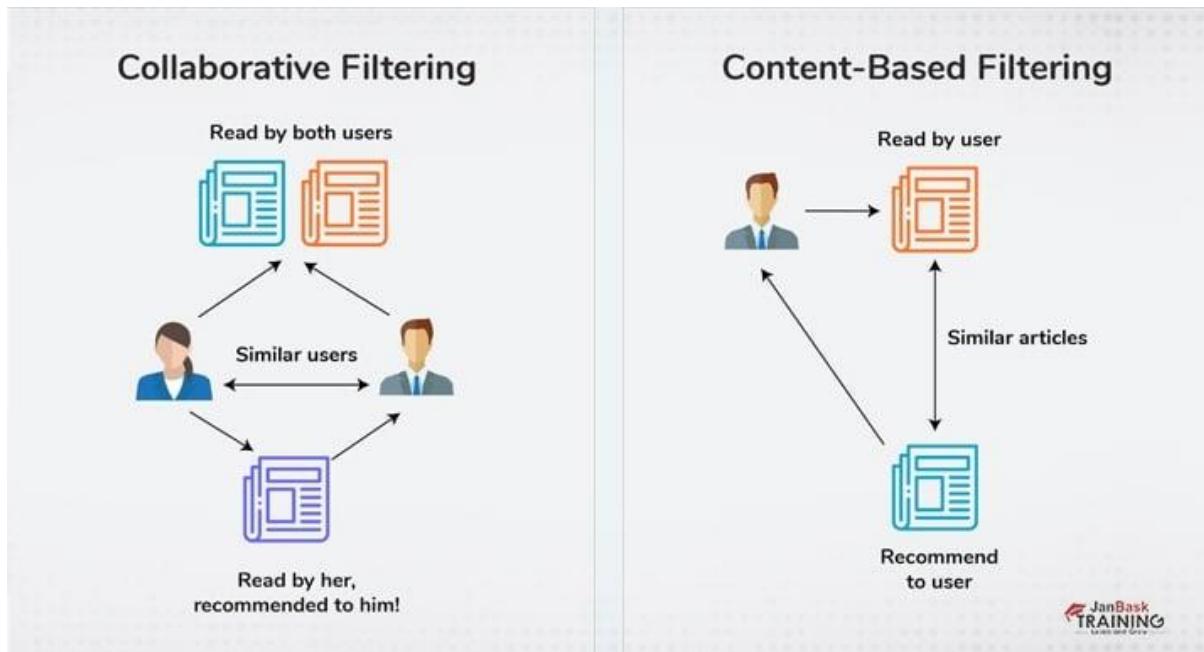
- Visualize the variables utilizing charts.

Based on the above observations, choose the best-fit algorithm for a particular dataset.

**Q26).What is Collaborative Filtering and Content-Based Filtering in Machine Learning?**

Collaborative filtering is considered to be a proven technique that is used for personalized content recommendations. It is a type of filtering system that predicts new content by matching an individual's interest with other user preferences.

However, the content-based filtering is focused only on the user preferences. Also, new recommendations are made to the user from similar content based on the user's previous choice.



**Q27). Explain Correlation and Covariance?**

Correlation is used for measuring and also for evaluating the quantitative relationship between two variables. Correlation measures the relationship of two variables such as Income and expenditure etc. Moreover, Covariance is a simple way to measure the correlation between two variables but there is a problem with covariance is that they are hard to compare without normalization.

**Q28). What are Parametric and Non-Parametric Models in Machine Learning?**

Parametric models have limited parameters and to predict new data, you only require to know the parameters of the model. However, Non-parametric models have no limits in taking a huge number of parameters that allow more flexibility to predict new data. You can efficiently know the state of the data and model parameters via Parametric and Non-parametric models.

Q29). What do you know about Reinforcement Learning?

Reinforcement learning varies from the other types of learning such as supervised and unsupervised learning. However, in reinforcement learning, we are given nothing neither the data nor the labels. Our learning is basically, based on the rewards given to the agent by the environment.

Q30). Differentiate Sigmoid and Softmax functions?

The sigmoid function is used for binary classification and the probabilities sum required to be 1. Whereas, Softmax function is used for multi-classification and its probability sum will be 1.

- Q1: What do you mean by cross-validation?
- Q2: How do you choose the metrics?
- Q3: What are the false positives and false negatives?  
Q4: Explain the terms ‘Recall’ and ‘Precision’:
- Q5: Distinguish between supervised learning and unsupervised learning.
- Q6: How do you validate a predictive model based on multiple regression?
- Q7: What is the full form of NLP?
- Q8: What is a random forest?
- Q9: Which model is better, Random forests or Support Vector Machine?  
Justify your answer.
- Q10: Explain PCA and its uses:
- Q11: What are the drawbacks of Naive Bayes? How can you improve it?
- Q12: Explain the shortcomings of a linear model?
- Q13: Are multiple small decision trees better than a single large one?  
Justify.
- Q14: What makes mean square error a poor metric of model performance?
- Q15. What assumptions is linear regression based upon?
- Q16: What is multicollinearity?  
Q17: Why should, or shouldn’t, you perform dimensionality reduction before fitting an SVM?
- Q18: Distinguish between classification and regression?
- Q19: Explain the difference between KNN and k-means clustering.

- Q20: How to ensure that your model is not overfitting?
- Q21: Explain Ensemble learning.
- Q22: How is ML different from Deep Learning?
- Q23: What is selection bias?
- Q24: Explain inductive and deductive reasoning:
- Q25: Elaborate the difference between Gini Impurity and Entropy in a Decision Tree.
- Q26. What are outliers and how to detect them?
- Q27. What is A/B Testing?
- Q28. Explain Cluster Sampling:
- Q29. Which Python libraries are commonly used in Machine Learning?
- Q30. What experience do you have with big data tools like Spark that are used in ML?
- Q31. How would you handle missing data in a dataset?
- Q32. Write a pseudocode for any algorithm.
- Q33. What was the last book or research paper that you read on machine learning?
- Q34. What ML model do you like the most?
- Q35. How is Data Mining different from Machine Learning?
- Q36. Name the life stages of model development in a Machine Learning project.
- Q37. Name some real-life applications of ML algorithms:
- Q38. Explain neural networks.
- Q39. Is Machine Learning just another name for Artificial Intelligence?
- Q40. What is a Hash Table?
- Q41. What are the different ways to perform dimensionality reduction on a dataset?
- Q42. Define the F1 Score.
- Q43. How do you prune a decision tree?
- Q44: How would you explain Machine Learning to a layman?
- Q45. What interests you the most about ML?

Q1: What do you mean by cross-validation?

As the name suggests, cross-validation is a technique to test whether the given ML system can perform accurately over datasets other than the one used to train it. Typically, programmers split their dataset into two different sets for cross-validation:

1. Training data- Used to train the system.
2. Testing data- Used to test and verify the system.

**Q2:** How do you choose the metrics?

Metrics are parameters that help you evaluate an ML model/system. The selection of metrics depends on a variety of factors like:

- Is it a classification or regression model?
- How varied are the target variables?

MAE, MAPE, RMSE, MSE for regression and Accuracy, Recall, Precision, and f1 score for classification are some of the most commonly used metrics.

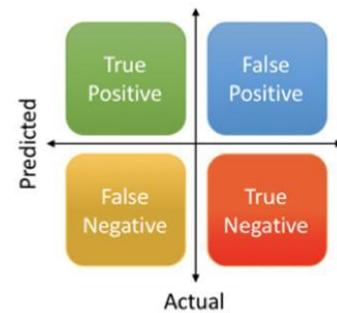
**Q3:** What are the false positives and false negatives?

A **false positive** is like a false alarm, where the model suggests the presence of a condition even when it doesn't exist. A **false negative** is the exact opposite of the above situation where the model suggests the absence of a condition when it is actually present.

**Q4:** Explain the terms 'Recall' and 'Precision':

Both **Recall** and **Precision** are accurate indicators of a model but have a distinct meaning. Where **recall** focuses on all the relevant results classified accurately by the model, **precision** helps you determine the percentage of the obtained results which are directly relevant to you.

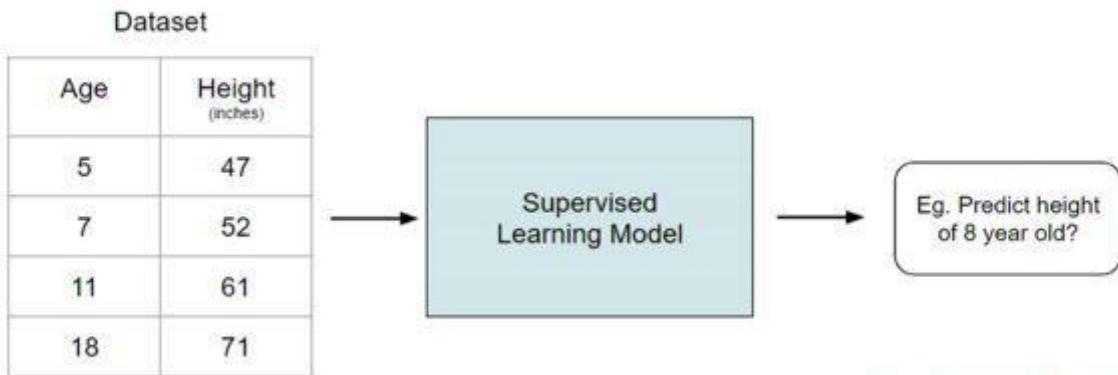
$$\text{Precision} = \frac{\text{True Positive}}{\text{Actual Results}} \quad \text{or} \quad \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$
$$\text{Recall} = \frac{\text{True Positive}}{\text{Predicted Results}} \quad \text{or} \quad \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$
$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{Total}}$$



**ANALYTIXLABS**

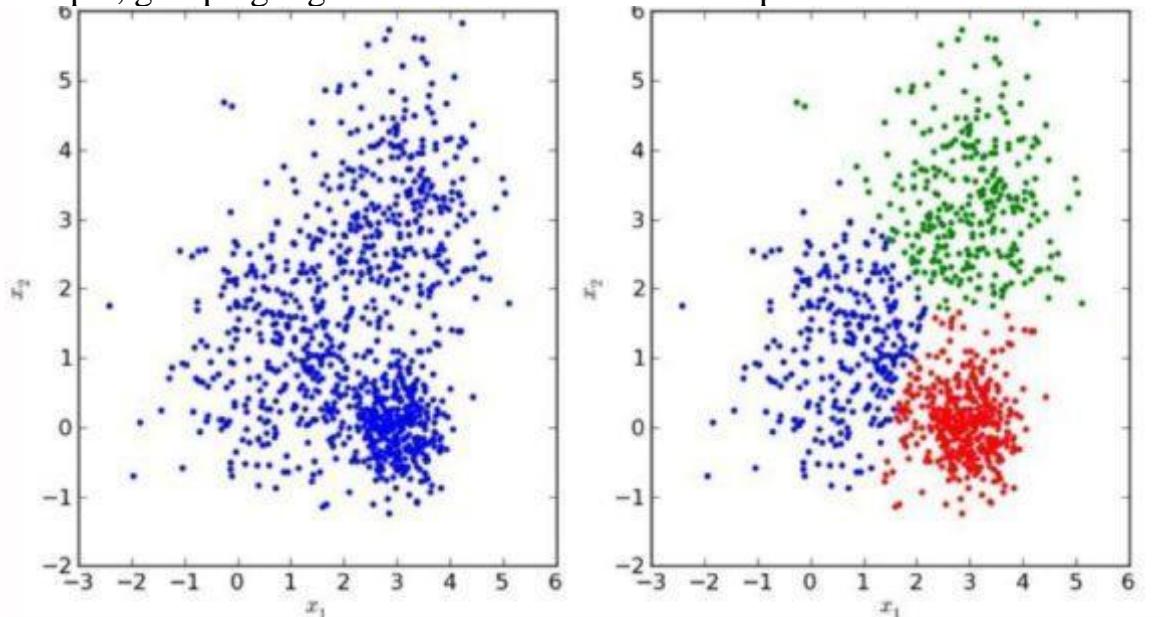
**Q5:** Distinguish between supervised learning and unsupervised learning.

In **Supervised learning**, you provide the model with an answer key to the questions it is supposed to solve so that the model can verify its results and improve its process accordingly—for example, the correlation between the age and height of a group of children.



**ANALYTIX LABS**

In the case of **Unsupervised learning**, the correct results are not known, so the model needs to draw inferences and find patterns from the given dataset. For example, grouping together customers with similar purchase histories.



**ANALYTIX LABS**

**Q6:** How do you validate a predictive model based on multiple regression?  
 The most commonly used method to do this would be through cross-validation, as explained in the previous question. But you can also choose to employ the **Adjusted R-squared** method. In this method, an r-squared value is generated, which determines the relation between the variance present in the dependent and independent variables of a dataset. So, the higher the r-squared value, the more accurate the model.

**Q7:** What is the full form of NLP?

**NLP** is short for Natural Language Processing. It is an AI discipline that focuses on helping machines understand and interact with humans in a more colloquial manner.

**Q8:** What is a random forest?

Random forests are a learning methodology based on the concept of decision trees. Multiple decision trees are created by a randomized selection of a subset of variables at each step of the decision tree which aggregates into a random forest. The mode of all predictions is then selected as a result with the least probability of errors.

**Q9:** Which model is better, Random forests or Support Vector Machine? Justify your answer.

When it comes to Machine Learning algorithms the theory of no free lunch comes into the picture. No single algorithm is superior to the other in absolute terms and comes with a set of some trade-offs. Depending on the use case we prefer one over the other.

But in general, Random forests are considered to be superior model to the SVM for the following reasons:

- You can determine the feature's importance using random forests, but not with SVM.
- It is easier to employ random forests than SVM, and the former is quicker too.
- Random forests prove to be more scalable and less memory intensive than SVMs for multi-class classifications.
- Lesser probability of over-fitting in general.
- Easy to tune the hyper-parameters.

**Q10:** Explain PCA and its uses:

PCA stands for Principal Component Analysis. It involves simplifying the data by reducing the dimensionality of the dataset — for example converting 3-D to 2-D — without changing the original variables of the model. PCA is a widely used compression technique utilized for better visualization and summarization of data, reducing the required memory, and speeding up the process

**Q11:** What are the drawbacks of Naive Bayes? How can you improve it?

The biggest drawback of Naive Bayes lies in its assumption that the features of a dataset are completely uncorrelated with one another, which is rarely the situation. The only way to improve the performance of Naive Bayes is to

actually remove the correlations between the features and make the process optimum for the Naive Bayes.

Q12: Explain the shortcomings of a linear model?

Following are the major downsides of a linear model:

- A linear model is based on too many theoretical assumptions which mostly don't hold true in reality.
- Discrete or binary outcomes cannot be obtained through a linear model.
- High inflexibility.

Q13: Are multiple small decision trees better than a single large one? Justify.

Having multiple small decision trees is the same as employing a random forest model, which is known to be more precise (low bias) and less prone to the problem of overfitting (high variance). So yes, having multiple small decision trees would be more preferable to having a single large one.

Q14: What makes mean square error a poor metric of model performance?

MSE or Mean square error is based on associating a considerably higher weight to large errors, thus putting greater emphasis on wider deviations. However, this works well in most of the algorithms to minimize the model error and cost.

Sometimes, a better option to MSE is MAE (mean absolute error) or MAPE (mean absolute percentage error), which does away with the above shortcoming and is easy to interpret.

Q15. What assumptions is linear regression based upon?

Linear regression is generally based on the following key assumption:

- The sample data must represent the entire population.
- The input and output variable must have a **linear relationship**.
- The input variable must show **homoscedasticity**.
- No **multicollinearity** among independent/ input variables.
- Normal distribution of the output variable for any value of the input variable.
- There is no serial or autocorrelation in the output/ dependent variable

Q16: What is multicollinearity?

When two independent variables display a high correlation to each other, multicollinearity is said to have occurred. Variance Inflation Factors (VIF) can

be used to detect multicollinearity between independent variables. Usually, a VIF of more than 4 is a sign of multicollinearity.

**Q17:** Why should, or shouldn't, you perform dimensionality reduction before fitting an SVM?

For an optimal model outcome, dimensionality reduction is highly recommended before fitting an SVM when the number of features is greater than the number of observations.

**Q18:** Distinguish between classification and regression?

**Classification**, as the name suggests, classifies or separates data into predetermined categories. The results obtained are discrete in nature. For example, classifying cricket players into bowler and batsman categories. Some business examples:

- Whether customers will **open an email or not?**
- Will a customer payback **credit card dues or default?**
- Is insurance claim **fraud or genuine claim?**

**Regression**, on the other hand, deals with continuous data like determining the temperature of an object at a certain point of the day. In this case we are predicting a numerical value/ a continuous number. Some business examples:

- Predicting **revenue of a company**
- **Footfall** in a mall
- **Total retail spend** by different customers

**Q19:** Explain the difference between KNN and k-means clustering.

**KNN** stands for K-Nearest Neighbours, which is a supervised learning method that requires labeled data that is then used to classify the points based on their distance from the nearest point.

**K-Means clustering** is an unsupervised ML algorithm where a model with unlabelled data is provided and the algorithm then groups observation/ data points based on the similarities, measured using the average of the distances between different points.

**Q20:** How to ensure that your model is not overfitting?

The primary reasons that cause overfitting in a model are the complexity of the model itself and the amount of noise in the variables used. **Cross-validation** methods like K-folds can be used to curb overfitting in a model. **Regularization** methods can be used to penalize parameters that might be causing overfitting.

**Q21:** Explain Ensemble learning.

Basically, ensemble learning is the collection and aggregation of multiple models using bootstrapped samples, usually decision trees (classifiers or regressors), to obtain more accurate results, with lower bias and variance. Ensemble learning models can be created sequentially or in parallel.

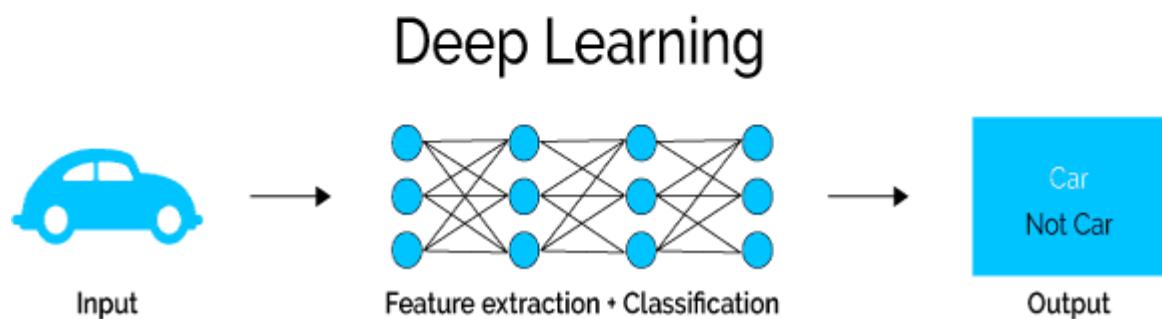
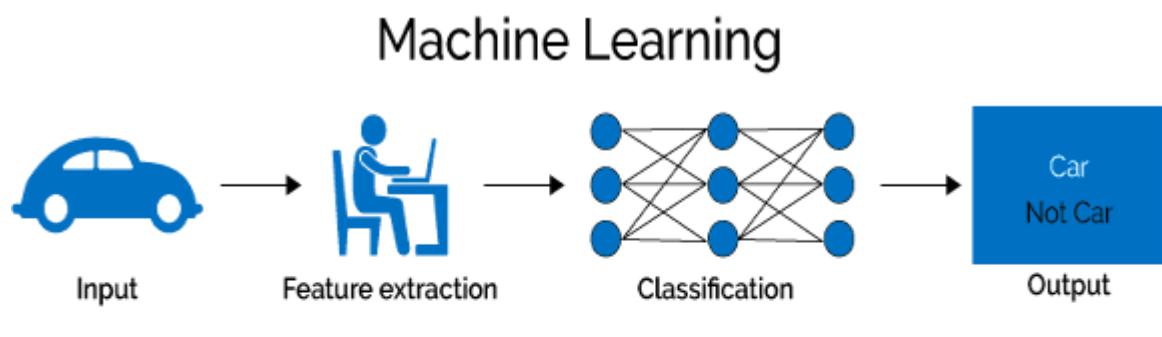
In **Bagging**, multiple models are created parallelly and the final results are aggregated outcomes of all these models, based on averages or majority voting. The most popular among such methods is Random Forest.

In **Boosting** a large number of sequential models are created parallelly and each subsequent model learns from the weakness of the previous model to improve the final accuracy. GBM (Gradient Boosting Method) and Xgboost are the two most popular boosting techniques.

Q22: How is ML different from Deep Learning?

Machine learning focuses on analyzing and learning from that data based on features fed into the model and using those insights to make better decisions.

Deep Learning, is basically a subset of ML that is inspired by the human brain. It focuses on feature extraction by deducing information from multiple layers, where each layer propagates the information to each layer for the final outcome.

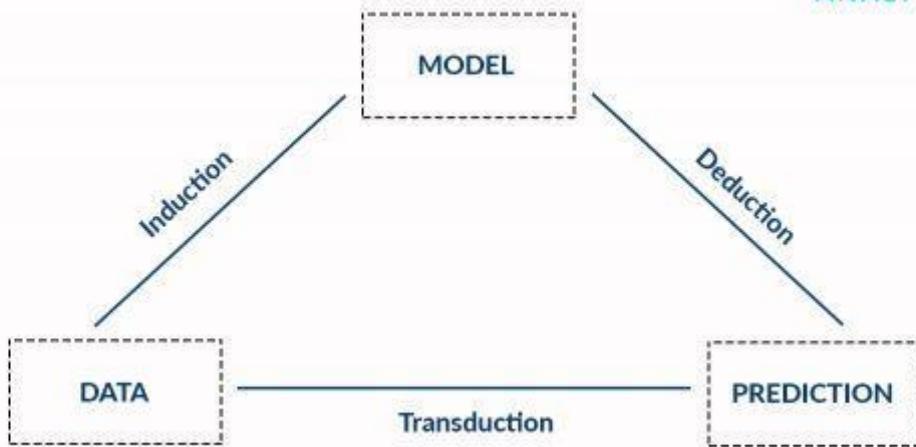


Q23: What is selection bias?

When a specific group or type of data is selected in a dataset more often, it leads to a statistical error called **selection bias**. Unless detected and resolved, selection bias can lead to inaccurate end results.

Q24: Explain inductive and deductive reasoning:

Inductive reasoning involves analyzing the available observations to draw a conclusion. Deductive reasoning, on the contrary, uses known conclusions or premises to form observations. Here is a good example.



Q25: Elaborate the difference between Gini Impurity and Entropy in a Decision Tree.

Gini Impurity and Entropy are both metrics that can help split a decision tree. The former measures the probability of correct classification of a random sample if you randomly pick a label in the branch.

Entropy is the measure of uncertainty in your model. Entropy is the lowest towards the leaf node. The Information Gain is the difference of entropies observed between a dataset before and after the split of an attribute. It has a maximum value near the leaf node. The difference between entropies can help understand the level of uncertainty in a decision tree.

# Impurity Criterion

## Gini Index

$$I_G = 1 - \sum_{j=1}^c p_j^2$$

$p_j$ : proportion of the samples that belongs to class c for a particular node

## Entropy

$$I_H = - \sum_{j=1}^c p_j \log_2(p_j)$$

$p_j$ : proportion of the samples that belongs to class c for a particular node.

\*This is the definition of entropy for all non-empty classes ( $p \neq 0$ ). The entropy is 0 if all samples at a node belong to the same class.

**Q26.** What are outliers and how to detect them?

Outliers are those data points that have a considerable difference of value from the average value of the dataset. Boxplot, linear models, and proximity-based models are often used for the screening of outliers in a dataset. For most of the models, it is highly recommended to treat outliers by either capping them or omitting them from the dataset.

**Q27.** What is A/B Testing?

A/B testing is dual-variable testing performed on randomized experiments for determining which of the two selected models is a better fit for the given dataset. Imagine having two movie recommendation models, A & B.

Performing A/B testing can then help us determine which of these two models will provide a better recommendation to the user.

**Q28.** Explain Cluster Sampling:

Cluster sampling is a grouping method used for a population that has separate subsets of homogeneous elements inside it. Commonly used for marketing research purposes, cluster sampling divides the given dataset into smaller groups and randomly selects a sample of the groups.

**Q29.** Which Python libraries are commonly used in Machine Learning?

Pandas, NumPy, SciPy, Seaborn, Sklearn, etc. are the top 5 most commonly used libraries for Data Analysis and Scientific Computations required for the ML models.

**Q30.** What experience do you have with big data tools like Spark that are used in ML?

At the enterprise level, Apache Spark plays a significant role to scale the machine models and enables real-time analytics on Big Data.

Spark is one of the most commonly used Big Data tools of ML and is probable to come up in at least some of the **machine learning interview questions** for job roles that involve handling Big Data. It is a common part of machine learning interview questions for professionals with some prior experience. Always be honest about such interview questions on machine learning. So ensure that you have some practical experience of using similar tools before attempting ML interview questions.

**Q31.** How would you handle missing data in a dataset?

Another hypothetical question that is a regular in a session of **machine learning interview questions and answers**. Most employers incorporate this situation in **machine learning interview questions** for freshers because they need to understand if the individual has enough practical knowledge to deal with such ubiquitous problems of everyday work.

Your answer to such an ML interview question should be that you could either replace the missing value with another value using the measure of central tendency, like mean or median or mode. Following approach is used most commonly:

- Continuous Variables: Replace missing with mean
- Ordinal Variables: Replace missing with the median
- Categorical Variables: Replace missing with the mode

In case, we have very small proportions of missing values in a large dataset then we can also drop them. `dropna()` method from the Pandas library.

**Q32.** Write a pseudocode for any algorithm.

The most important quality that interviewers try to ascertain through their interview questions on machine learning is the individual's understanding of the logic of ML. Writing the pseudocode of an algorithm requires an intuitive grasp of the fundamental concepts and strong logical reasoning skills. So always choose an algorithm that you have an in-depth understanding of.

One of the easiest algorithms is Decision Tree where we can split the data in each node in order to minimize MSE or GINI index.

**Q33. What was the last book or research paper that you read on machine learning?**

The interviewer will try to assess if you have a genuine interest in the field by asking such interview questions on machine learning. You must always be well-read and aware of the latest developments being made in ML by reading published research papers and scientific journals.

**Q34. What ML model do you like the most?**

Although the interviewer might only ask you to name your favorite machine learning model at first, there is a strong chance that he will have some follow-up questions on the model you choose. So bear in mind to name a simple enough ML model that you have good knowledge and understanding of.

And please remember the principle of no free lunch as explained in Q9! No single model is superior in every scenario. Every model has its pros and cons and we choose an appropriate model based on the business case and applicable trade-offs.

**Q35. How is Data Mining different from Machine Learning?**

Data mining is a discipline that deals with the extraction of data from unrefined sources so that it can be analyzed and studied to obtain meaningful patterns.

Machine Learning focuses on developing algorithms and methodologies that can help machines learn and evolve on their own.

**Q36. Name the life stages of model development in a Machine Learning project.**  
Development of an ML model progresses in the following stages:

1. **Define Business Problem:** Understand business objectives and convert it analytics problem
2. **Data Construct:** Identify the required data sources, extract and aggregate the data at the required level.
3. **Exploratory Analysis:** Understand the data, examine the variables for errors, outliers and missing values. Identify the relationship between different types of variables. Check for assumptions.
4. **Data Preparation:** Exclusions, type conversions, outlier treatment, missing value treatment. Create new, hypothetically relevant variables, e.g. max, min, sum, change, ratio. Binning variables, dummy variables creation, etc.

5. **Feature Engineering:** Avoid multicollinearity and optimize model complexity by reducing the number of input variables – variable cluster, correlation, factor analysis, RFE, etc.
6. **Data Split:** Split the data into training and testing samples.
7. **Model Building:** Fit, check accuracy, cross-validate, and tune the model with the help of parameters and hyperparameters.
8. **Model Testing:** Check the model on the testing sample, run diagnostics, and iterate the model if required.
9. **Model Implementation:** Prepare final model results—present the model. Identify the limitations of the model Implement the model (converting the ML solution into production).
10. **Performance Tracking:** Track model performance periodically and update it as and if required. With an evolving business environment, the performance of any ML model is likely to get impacted over a period of time.

Q37. Name some real-life applications of ML algorithms:

Machine learning algorithms are finding significant uses across the following sectors-

- Bioinformatics
- Robotics Process Automation
- Natural Language Processing
- Sentiment Analysis
- Fraud detection
- Facial & Vocal recognition systems
- Anti-money laundering

Q38. Explain neural networks.

You can expect a question on neural networks when the interviewer has moved on from basic and intermediate machine learning questions and answers. The neural network is an advanced discipline of ML that has shown some remarkable results through its increased adaptability and flexibility.

A **neural network** is a type of ML algorithm that identifies underlying hidden patterns & relationships in a dataset through a process that is inspired by the working of the human brain operates.

It is a nondeterministic algorithm without a strong mathematical foundation and can be roughly compared to large-scale trial and error computations. These models very well adapt to changes in the input data; hence generating highly

accurate results without explicit programming inputs. (You may also refer to Qn 22 again.)

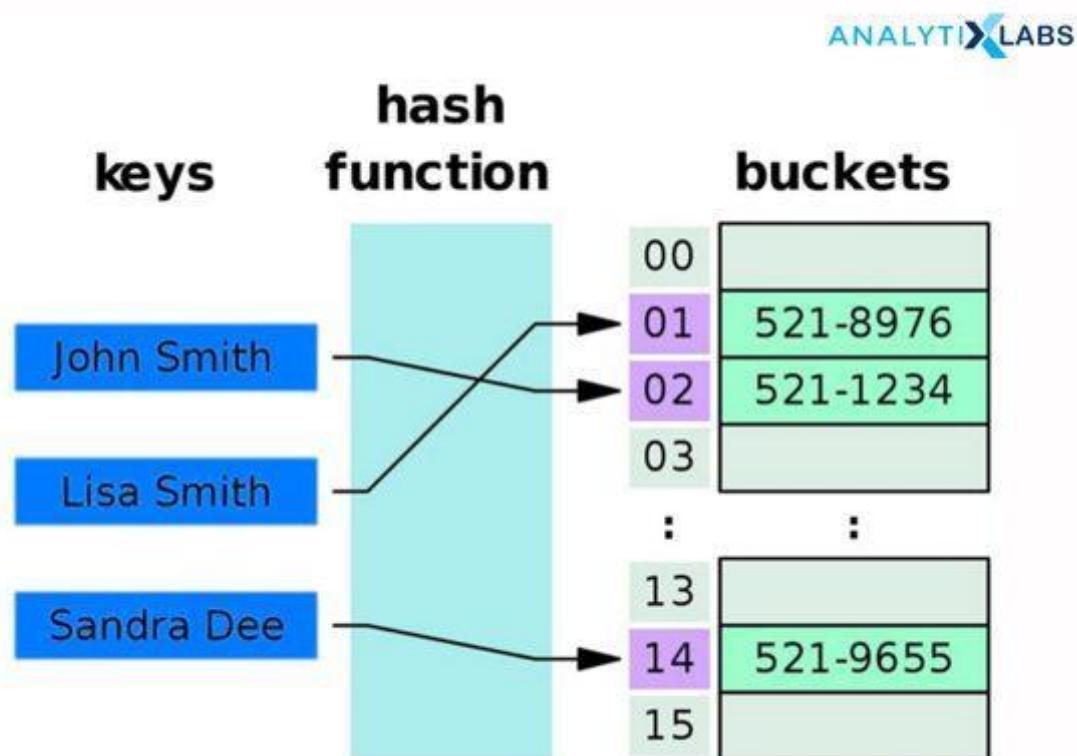
Q39. Is Machine Learning just another name for Artificial Intelligence?

This might seem like a trick question at first, but the simple answer is: No, ML and AI are not one and the same. Although both of them focus on making machines more intelligent and capable of doing what humans can, Machine Learning is actually a subset of AI that focuses specifically on the development of learning methodologies for machines.

Whereas, AI is broader and may all encompass other hardware and engineering elements to device the final solution. For example, Netflix's AI-enabled recommendation engine is predominantly a Machine Learning solution, while the same can not be said for an autonomous self-driving car.

Q40. What is a Hash Table?

A **Hash Table** is an organized listing of data elements where each element in the structure has a unique index value of its own. This allows hash tables to perform search and insert operations on data much more quickly as the data elements are stored in uniform association with one another.



Q41. What are the different ways to perform dimensionality reduction on a dataset?

Dimensionality reduction can be achieved through the following methods:

- Factor Analysis
- Principal Component Analysis
- Isomap
- Autoencoding
- Semidefinite Embedding

Q42. Define the F1 Score.

**F1 score** is a performance measuring metric-based statistic evaluation. It is the weighted average of the Recall and Precision values of a model. It is predominantly used to compare the performances of two ML algorithms across a common dataset.

$$\begin{aligned}
 precision &= \frac{TP}{TP + FP} \\
 recall &= \frac{TP}{TP + FN} \\
 F1 &= \frac{2 \times precision \times recall}{precision + recall} \\
 accuracy &= \frac{TP + TN}{TP + FN + TN + FP} \\
 specificity &= \frac{TN}{TN + FP}
 \end{aligned}$$

Q43. How do you prune a decision tree?

Pruning involves replacing nodes of a decision tree in a top-down or bottom-up manner. It is very helpful in increasing the accuracy of the decision tree while also reducing its complexity and overfitting.

Generally, a tree is grown until terminal nodes have a small sample then pruned to remove nodes that do not add additional accuracy or information. The objective is to reduce the size of a tree without affecting the accuracy as measured by cross-validation. There are the following 2 main approaches used for decision tree pruning:

- Error based
- Cost complexity based

1. What is Machine learning?

**Ans:** Machine learning is a field of computer science that deals with system programming to learn and improve with experience.

For example: Robots are coded so that they can perform the task based on data they collect from sensors. It robotically learns programs from data

Q2. What is the Box-Cox transformation used for?

**Ans:** The Box-Cox transformation is a generalized "power transformation" that transforms data to make the distribution more normal. For example, when its lambda parameter is 0, it's equivalent to the log-transformation.

It's used to stabilize the variance (eliminate heteroskedasticity) and normalize the distribution.

Q3. What is 'Overfitting' in Machine learning?

**Ans:** In machine learning, when a statistical model defines random error of underlying relationship 'overfitting' occurs. When a model is exceptionally complex, overfitting is generally observed, because of having too many factors with respect to the number of training data types. The model shows poor performance which has been overfit.

Q4. What are the different Algorithm techniques in Machine Learning?

**Ans: The different types of techniques in Machine Learning are:**

- Supervised Learning
- Semi-supervised Learning
- Unsupervised Learning
- Transduction
- Reinforcement Learning

Q5. How is KNN different from k-means clustering?

**Ans:** K-Nearest Neighbors is a supervised classification algorithm, while k-means clustering is an unsupervised clustering algorithm. While the mechanisms may seem similar at first, what this really means is that in order for K-Nearest Neighbors to work, you need labeled data you want to classify an unlabeled point into (thus the nearest neighbor part). K-means clustering requires only a set of unlabeled points and a threshold: the algorithm will take unlabeled points and gradually learn how to cluster them into groups by computing the mean of the distance between different points.

The critical difference here is that KNN needs labeled points and is thus supervised learning, while k-means doesn't — and is thus unsupervised learning.

**Q6.** Mention the difference between Data Mining and Machine learning?

**Ans:**

**Data mining:** It is defined as the process in which the unstructured data tries to abstract knowledge or unknown interesting patterns. During this machine process, learning algorithms are used.

**Machine learning:** It relates with the study, design and development of the algorithms that give processors the ability to learn without being openly programmed.

**Q7.** What are the five popular algorithms of Machine Learning?

**Ans: Five popular algorithms are:**

1. Decision Trees
2. Probabilistic networks
3. Neural Networks (back propagation)
4. Support vector machines
5. Nearest Neighbor

**Q8.** Define precision and recall.

**Ans:** Recall is also known as the true positive rate: the amount of positives your model claims compared to the actual number of positives there are throughout the data. Precision is also known as the positive predictive value, and it is a measure of the amount of accurate positives your model claims compared to the number of positives it actually claims. It can be easier to think of recall and precision in the context of a case where you've predicted that there were 10 apples and 5 oranges in a case of 10 apples. You'd have perfect recall (there are actually 10 apples, and you predicted there would be 10) but 66.7% precision because out of the 15 events you predicted, only 10 (the apples) are correct.

**Q9.** Why is “Naive” Bayes naive?

**Ans:** Despite its practical applications, especially in text mining, Naive Bayes is considered “Naive” because it makes an assumption that is virtually impossible to see in real-life data: the conditional probability is calculated as the pure product of the individual probabilities of components. This implies the absolute independence of features — a condition probably never met in real life. As a Quora commenter put it whimsically, a Naive Bayes classifier that figured

out that you liked pickles and ice cream would probably naively recommend you a pickle ice cream.

Q10. Why overfitting happens?

**Ans:** The possibility of overfitting happens as the criteria used for training the model is not the same as the criteria used to judge the efficiency of a model.

Q11. What is inductive machine learning?

**Ans:** The inductive machine learning implicates the process of learning by examples, where a system, from a set of observed instances tries to induce a general rule.

Q12. What is the standard approach to supervised learning?

**Ans:** Split the set of example into the training set and the test is the standard approach to supervised learning is.

Q13. What's your favorite algorithm, and can you explain it to me in less than a minute?

**Ans:** This type of question tests your understanding of how to communicate complex and technical nuances with poise and the ability to summarize quickly and efficiently. Make sure you have a choice and make sure you can explain different algorithms so simply and effectively that a five-year-old could grasp the basics!

Q14. What's the difference between Type I and Type II error?

**Ans:** Don't think that this is a trick question! Many machine learning interview questions will be an attempt to lob basic questions at you just to make sure you're on top of your game and you've prepared all of your bases. Type I error is a false positive, while Type II error is a false negative. Briefly stated, Type I error means claiming something has happened when it hasn't, while Type II error means that you claim nothing is happening when in fact something is.

A clever way to think about this is to think of Type I error as telling a man he is pregnant, while Type II error means you tell a pregnant woman she isn't carrying a baby.

**Q15.** In what areas Pattern Recognition is used?

**Ans:** Pattern Recognition can be used in the following areas:

- Computer Vision
- Data Mining
- Speech Recognition
- Informal Retrieval
- Statistics
- Bio-Informatics

**Q16.** What's a Fourier transform?

**Ans:** A Fourier transform is a generic method to decompose generic functions into a superposition of symmetric functions. Or as this more intuitive tutorial puts it, given a smoothie, it's how we find the recipe. The Fourier transform finds the set of cycle speeds, amplitudes and phases to match any time signal. A Fourier transform converts a signal from time to frequency domain — it's a very common way to extract features from audio signals or other time series such as sensor data.

**Q17.** How can you avoid overfitting?

**Ans:** By using a lot of data overfitting can be avoided, overfitting happens relatively as you have a small dataset, and you try to learn from it. But if you have a small database and you are forced to come with a model based on that. In such situation, you can use a technique known as cross validation. In this method the dataset splits into two section, testing and training datasets, the testing dataset will only test the model while, in training dataset, the data points will come up with the model.

In this technique, a model is usually given a dataset of a known data on which training (training data set) is run and a dataset of unknown data against which the model is tested. The idea of cross validation is to define a dataset to “test” the model in the training phase.

**Q18.** What are the three stages to build the hypotheses or model in machine learning?

**Ans:**

- Model building
- Applying the model
- Model testing.

**Q19.** What's the difference between a generative and discriminative model?

**Ans:** A generative model will learn categories of data while a discriminative model will simply learn the distinction between different categories of data. Discriminative models will generally outperform generative models on classification tasks.

**Q20.** How is a decision tree pruned?

**Ans:** Pruning is what happens in decision trees when branches that have weak predictive power are removed in order to reduce the complexity of the model and increase the predictive accuracy of a decision tree model. Pruning can happen bottom-up and top-down, with approaches such as reduced error pruning and cost complexity pruning.

Reduced error pruning is perhaps the simplest version: replace each node. If it doesn't decrease predictive accuracy, keep it pruned. While simple, this heuristic actually comes pretty close to an approach that would optimize for maximum accuracy.

**Q21.** How would you handle an imbalanced dataset?

**Ans:** An imbalanced dataset is when you have, for example, a classification test and 90% of the data is in one class. That leads to problems: an accuracy of 90% can be skewed if you have no predictive power on the other category of data! Here are a few tactics to get over the hump:

1. Collect more data to even the imbalances in the dataset.
2. Resample the dataset to correct for imbalances.
3. Try a different algorithm altogether on your dataset.

## **Q1. What are the different types of Machine Learning Algorithms?**

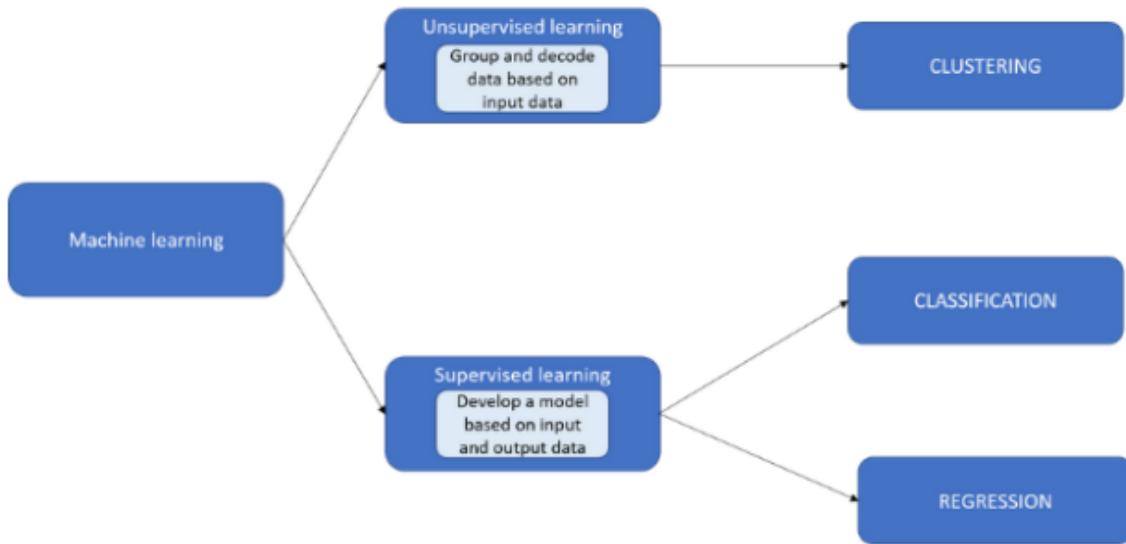
There are three different types of Machine Learning algorithms:

**i. Supervised Learning:** In this machine learns under the guidance of labelled data. In this, a model makes predictions and decisions based on past data.

Labelled data means sets of data that are numbered or labelled for reference. It can be further divided into two types: Classification and Regression.

**ii. Unsupervised Learning:** In unsupervised machine learning there is no such provision of labelled data. In this, the model input data needs to be given so that the machine can learn. It further consists of clustering algorithms.

**iii. Reinforcement Learning:** In this, the machine learns from a hit and trial method. The machine learns from the rewards or penalties it received from its previous actions.



## Q2. What is overfitting and how can you avoid it?

Overfitting is when the model learns too well. It happens when the model learns the details and noise in training data to an extent that it begins to negatively impact the performance of the model. The most popular solutions to prevent overfitting are:

- Cross-Validation
- Train with more data
- Remove features
- Early stopping of the machine when you find out something is going wrong
- Regularization of algorithms so that your model can be simpler



### Q3. What is the difference between classification and regression in Machine Learning?

Classification and regression are the two main prediction problems which are most commonly faced while using Machine Learning.

Classification	Regression
It is the process of finding or discovering a model or function which helps in separating the data into multiple categorical classes which is discrete values	It is the process of finding a model or function for distinguishing the data into continuous real values instead of using classes or discrete values.
Nature of predicted data is unordered	Nature of predicted data is ordered
Calculate using measuring accuracy	Calculated by measurement of root mean square error
Example: Decision tree algo, logistic regression etc.	Example: Regression tree, Linear regression

### Q4. What is “Training set” and “Test set” in Machine Learning?

<b>Training Set</b>	<b>Train Set</b>
It is the examples given to the models to analyze and learn.	It is used to test the accuracy of the hypothesis generated by the model.
70% of the total data is taken as the training dataset.	The rest 30% is taken as testing dataset
This is the labelled data we use to train the model.	We test without labelled data and verify results with labels.

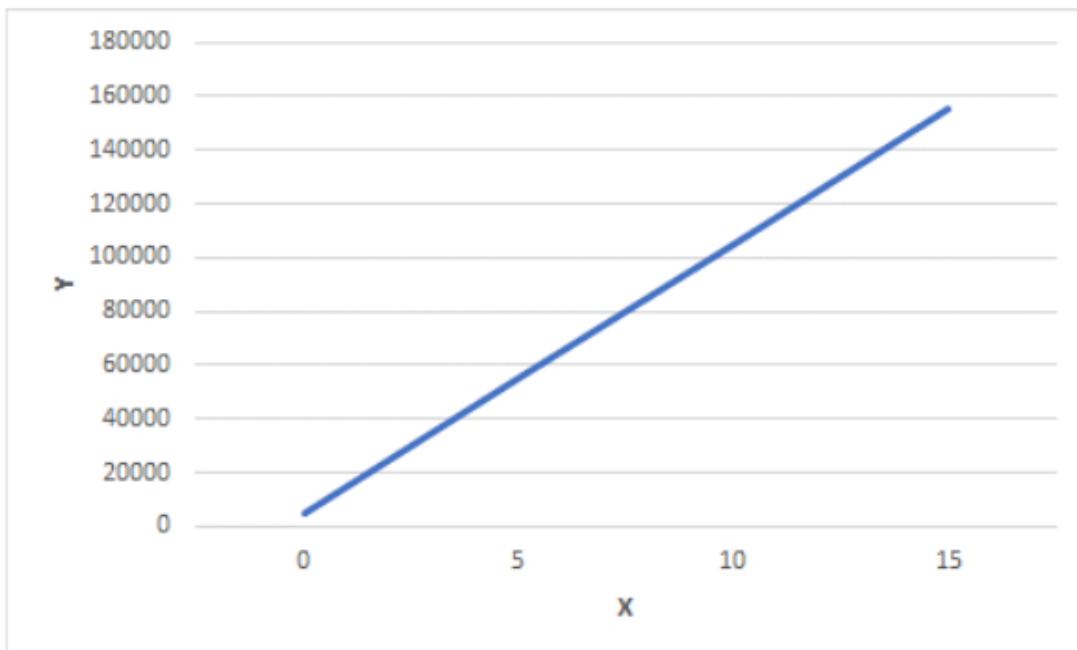
### **Q5. What is Linear Regression?**

Linear Regression is a supervised Machine Learning algorithm used to find out the linear relationship between the dependent and the independent variables for predictive analysis. Linear Regression equation is given as:

$Y = A + B.X$     Where :

- X is the input or the independent variable
- Y is the output or the dependent variable
- A is the intercept and,
- B is the coefficient of X

**VERZEO**



### **Q6. What are Bias and Variance?**

Bias is the accuracy of our predictions

“Bias is the algorithm’s tendency to consistently learn the wrong thing by not taking into account all the information in the data (underfitting).”

A high bias means that the prediction will be inaccurate. Hence the bias value should be as low as possible to make accurate desired predictions.

**Variance** is the change in prediction accuracy of Machine Learning model between training data and test data.

Simply put, if the ML model prediction accuracy is “X” on training data and its prediction accuracy on test data is “Y” then

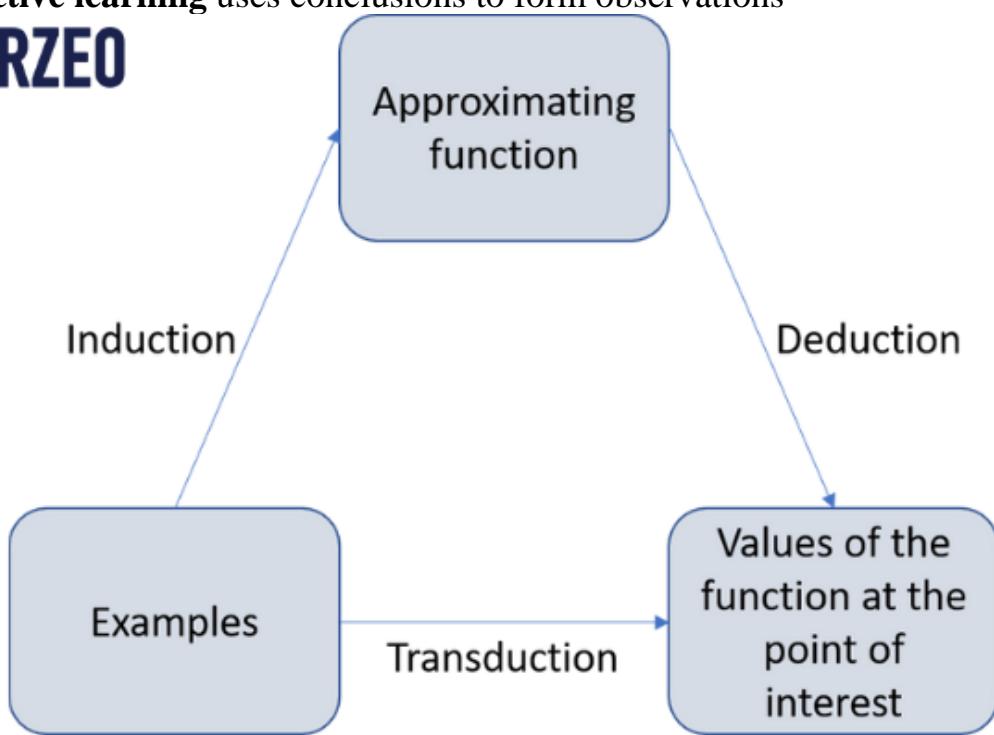
**Variance = X-Y**

**Q7. What is the difference between inductive and deductive learning?**

**Inductive learning** uses observations to draw conclusions

**Deductive learning** uses conclusions to form observations

**VERZE0**



**Q8. What is Variance Inflation Factor?**

Variance Inflation Factor (VIF) is an estimate of the volume of multicollinearity in the collection of regression variables.

It is given as;

VIF = Variance of model / Variance of the model with a single independent variable.

### **Q9. How do you handle missing data or corrupted data in the dataset?**

You can use the Pandas library in Python to handle the missing data. There are two methods to handle the missing data:

- **isNull()**: For detecting the missing values.
- **dropna()**: We use dropna() method for removing the columns/rows with null values.

### **Q10. Explain the Confusion Matrix with Respect to Machine Learning Algorithms.**

To measure the performance of an algorithm we use a confusion matrix. In supervised learning, it is called Confusion Matrix. In unsupervised learning, it is called matching matrix. The confusion matrix has two parameters:

- Actual
- Predicted

The confusion matrix visualizes the accuracy by comparing the actual and predicted classes. Below I have shown confusion table for a binary confusion matrix:

		PREDICTED		
		FALSE	TRUE	
ACTUAL	FALSE	True Negative(TN)	False Positive(FP)	PRECISION
	TRUE	False Negative(FN)	True Positive(TP)	
		RECAL		

- TP: True Positive: Predicted values correctly predicted as actual positive
- FP: Predicted values incorrectly predicted as actual positive. i.e., Negative values predicted as positive
- FN: False Negative: Positive values predicted as negative
- TN: True Negative: Predicted values correctly predicted as an actual negative

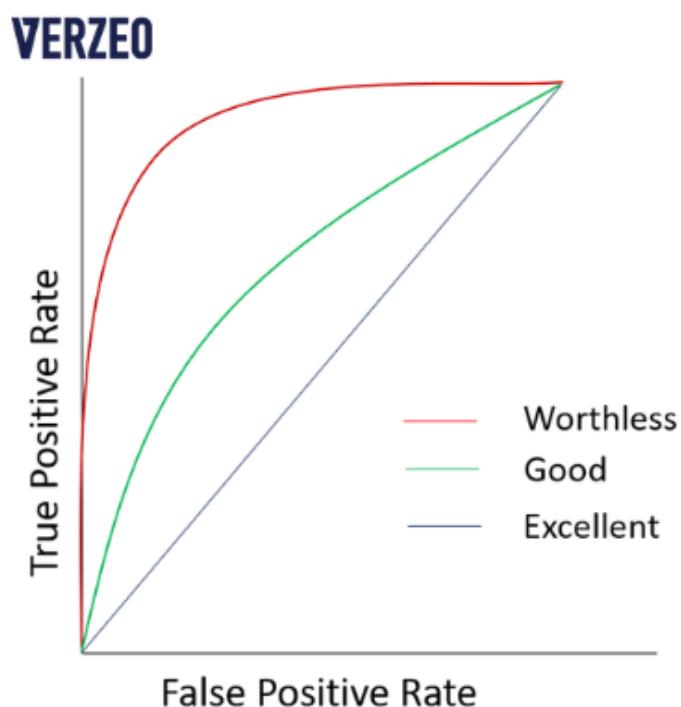
You can use the confusion matrix to compute the accuracy test.

### **Q11. Compare K-means and KNN algorithms.**

K-means	KNN
It is unsupervised in nature	It is supervised in nature
It is a clustering algorithm	KNN is a classification algorithm
It needs unlabelled data to train	It needs labelled data to train

### **Q12. What is ROC curve? What does it represent?**

Receiver Operating Characteristic curve (or ROC curve) is a fundamental tool which is used for diagnostic test evaluation. It is a plot of Sensitivity vs Specificity i.e. it is a plot of the true positive rate against the false-positive rate.



### **Q13. What is the difference between type I and type II error?**

**Type I** error is a false positive. Type I error is claiming something has happened when it hasn't.

**Type II** error is a false negative error. Type II error is claiming nothing when in fact something has happened.

### **Q14. What are collinearity and multicollinearity?**

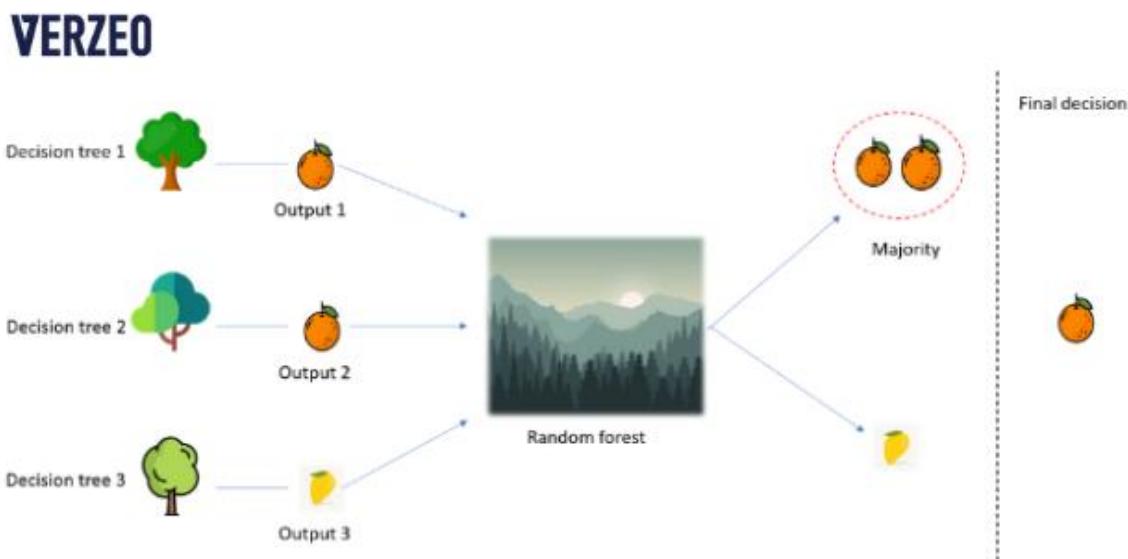
**Collinearity** is when two predictor variables in a multiple regression have some relation between them.

**Multicollinearity** occurs when more than two predictor variables are inter-correlated.

### **Q15. What Is a Random Forest?**

It is a supervised machine learning algorithm which is generally used for classification problems. It creates multiple decision trees during its training phase.

The **random forest** chooses the decision of the majority of trees and makes a final decision based on that.



### **Q16. When Will You Use Classification over Regression?**

When the target is categorical we will use classification, whereas when the target variable is continuous we will use regression. Both these belong to supervised machine learning algorithms.

Examples of Classification problem include predicting:

- Type of colour
- Breed of animal
- Gender of person
- A statement is true or false
- Yes or No
- Type of flower

Whereas examples of regression problems include predicting:

- Score of team
- Amount of rainfall
- Amount of revenue generated
- Price of a product

### **Q17. What are Eigenvectors and Eigenvalues?**

**Eigenvectors:** Their direction remains the same even when a linear transformation is performed on them

**Eigenvalues:** It is a scalar that is used for the transformation of an eigenvector. The Eigenvector of a square matrix B is a non zero vector such that for some numbers we have the following:

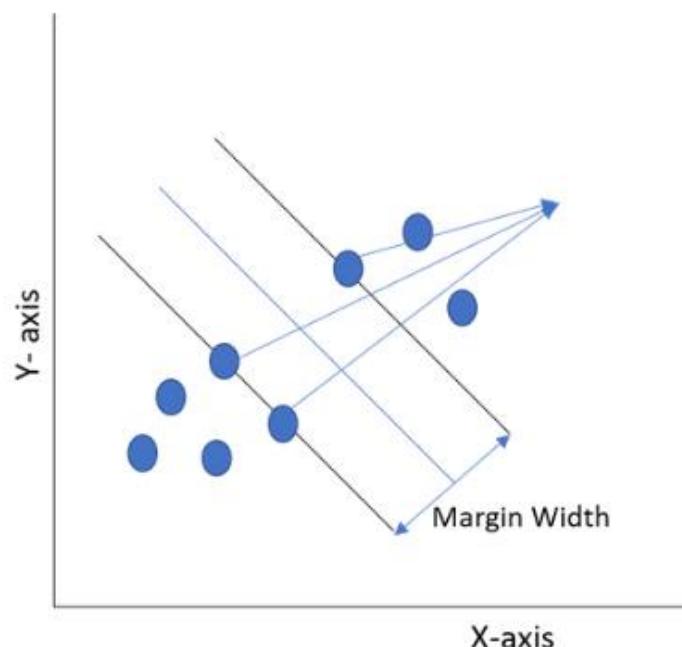
$$Ax = x$$

where  $x$  is an Eigenvalue.

### **Q18. What is SVM (Support Vector Machines)?**

SVM is a supervised machine learning algorithm that is used for classification. They can be used to analyse data for classification and regression analysis.

In SVM each data item is plotted in n-dimensional space with the value of each feature being the value of a particular coordinate. After this, we perform classification by finding the hyper-plane that differentiates the two classes. (Follow the below graph)



Support Vectors are the co-ordinates of individual observations.

Q19. Implement the KNN classification algorithm.

In the following code snippet, we are using Iris dataset to implement the KNN classification algorithm.

```
# KNN classification algorithm

from sklearn.datasets import load_iris

from sklearn.neighbors import KNeighborsClassifier

import numpy as np

from sklearn.model_selection import train_test_split

iris_dataset=load_iris()

X_train, X_test, Y_train, Y_test = train_test_split(iris_dataset["data"],
iris_dataset["target"], random_state=0)
```

```
kn = KNeighborsClassifier(n_neighbors=1)

kn.fit(X_train, Y_train)

X_new = np.array([[8, 2.5, 1, 1.2]])

prediction = kn.predict(X_new)

print("Predicted target value: {}".format(prediction))

print("Predicted feature name: {}".format

(iris_dataset["target_names"][prediction]))

print("Test score: {:.2f}".format(kn.score(X_test, Y_test)))
```

Output:

Predicted Target Name: [0]

Predicted Feature Name: [' Setosa']

Test Score: 0.92

## **Q20. What is cluster sampling?**

Cluster sampling is a process of randomly selecting intact groups within a defined population sharing similar characteristics. Cluster sampling is a probability sample where each unit is a cluster of elements.

In this, the total population is divided into groups known as clusters. The elements in these clusters are then sampled. If all the elements in these clusters are sampled then this is referred to as a “one-stage” cluster sampling plan. If in these clusters a random set of subgroups is selected, then it is called “two-stage” cluster sampling plan.

The common aim for the cluster sampling is to reduce the cost and attain a desired level of accuracy. Now that we have discussed various Machine learning

interview questions based on theory and algorithms, we will step up a bit and discuss certain machine learning questions based on real-life applications.

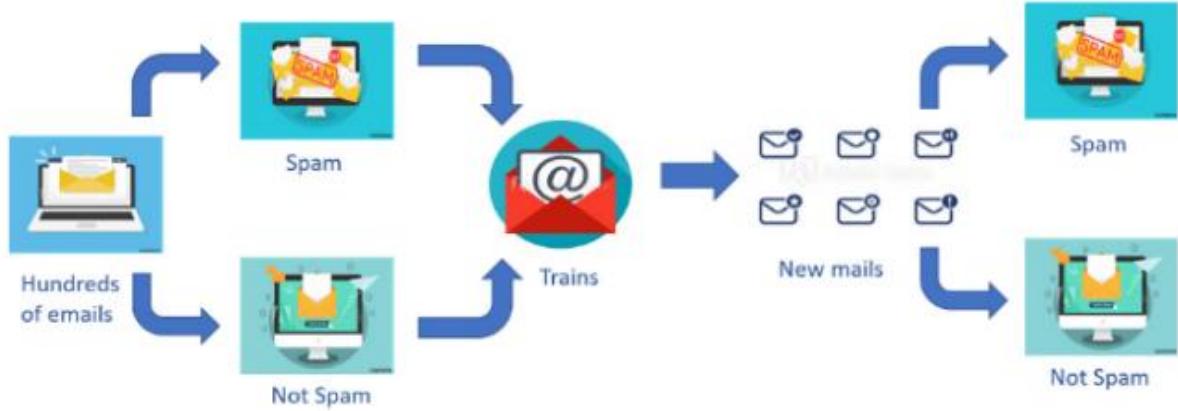
Read this section carefully because I am sure you will be asked most of the questions from this section. So, let's get started.

### **Q21. How will you design an Email spam filter?**

To build an **email spam filter** we follow the following steps:

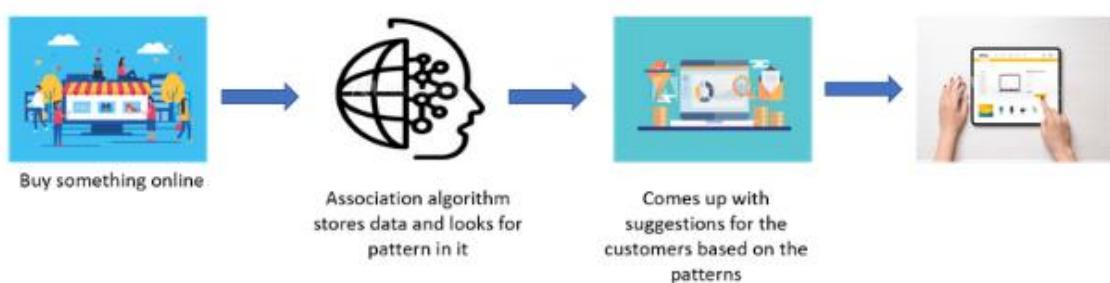
1. The email spam filter will be fed with hundreds of emails everyday.
2. Each of these emails after going through the filter will be labelled: 'spam' or 'not spam'.
3. The supervised machine learning algorithm will then find out which of these emails have been labelled spam based on spam words like the lottery, free offer, full refund, 100% off etc.
4. The next time the email is about to land in your mailbox, the spam filter will use algorithms like Decision Trees and SVM to determine that the mail is spam or not.
5. If the likelihood is high, it will be labelled as spam and the email won't land in your mailbox.
6. We will then send a certain number of test e-mails and check the accuracy of the model.
7. The accuracy should be as close to the desired level as possible.
8. We will test various models with different algorithms.
9. The model with the highest accuracy will be used.

## VERZEO



Q22. How does the recommendation engine work on e-commerce websites?  
Once a user buys something from an e-commerce website it stores the purchase data for future reference and finds products that are most likely to be bought by the user in future. This is possible because of a future algorithm, which can identify patterns in a given dataset.

## VERZEO



Consider this example:

1. Let us suppose you buy a cycle from an online store.
2. The bot will gather information about this and place your user id in a group who also bought items related to cycle.
3. Let us call this group a cluster.
4. Now suppose if any other user from this cluster bought a cycle seat cover.
5. The bot will then crawl the online store and show all the users in these cluster items related to the cycle seat cover in their recommendation list.

Suppose you buy an iPhone from an eCommerce website.

The bot will gather this information and place your user id in that group of users who also bought an iPhone. Let us call this group a cluster.

The bot will search the entire online store and accordingly show products related to iPhone, such as charger, earphone jack, screen guards etc. to this particular group in which you are placed.

This is how **recommendation engine** works in an e-commerce website.

### **Q23. How can you help our marketing team be more efficient?**

The answer to this question depends upon the type of company. The below-mentioned examples will help you:

1. Clustering algorithms to build custom customer segments for each type of marketing campaign.
2. Natural language processing for headlines to predict performance before running the ad spend.
3. Predicting conversion probability based on a user's website behaviour in order to create better re-targeting campaigns for potential customers.

This type of machine learning interview questions is very frequently asked. You can expect any kind of variation in this question.

**Q24. You've built a random forest model with 10000 trees. You got delighted after getting a training error of 0.00. But, the validation error is 34.23. What is going on? Haven't you trained your model perfectly?**  
It implies that the model has overfitted. Training error 0.00 implies that the classifier has copied the training data patterns to an extent, that they are not available in the unseen data. Hence when this classifier ran on an unseen sample, it couldn't find those patterns and returned prediction with a high error.

**Q25. Comment on the statement. Treating a categorical variable as a continuous variable would result in a better predictive model?**

When the variable is ordinal in nature, only then the categorical variable can be considered as a continuous variable.

**Q26. You are given a data set consisting of variables having more than 20% missing values? Let's say, out of 50 variables, 8 variables have missing values higher than 20%. How will you deal with them?**

The problem can be dealt with in the following ways:

1. Assign a unique category to the missing values.
2. We can remove them blatantly.
3. We can target variables to check their distribution. and if found any pattern we'll keep those missing values and assign them a new category while removing others.

**Q27. How would you approach the “Netflix Prize” competition?**

Netflix prize was a competition organised by Netflix which offered \$1,000,000 for a better collaborative filtering algorithm.

The problem statement says:

*Predict user rating for films based on previous ratings without any other information about the users of films, i.e. without the users or the films being identified except by numbers assigned for the contest.*

Its an open problem statement. Let me know what will be your solution to this problem

## **Q28. Explain How a System Can Play a Game of Chess Using Reinforcement Learning.**

Reinforcement learning consists of an environment and an agent. The agent performs certain actions to achieve a specific goal. Every time the agent performs a task that is in relation to the goal, it is rewarded. And, every time it takes a step which goes against that goal or in the reverse direction, it is penalized.

Earlier, chess programs had to determine the best moves after research on numerous factors. Building a machine designed to play such games would require many rules to be specified.

With reinforced learning, we don't have to deal with this problem as the learning agent learns by playing the game. It will make a move (decision), check if it's the right move (feedback), and keep the outcomes in memory for the next step it takes (learning). There is a reward for every correct decision the system takes and punishment for the wrong one.

Machine learning interview questions based on real-life scenarios can be asked at any point during the interview. So, you need to be updated with the various advancements in this industry.

## **Q29. How Will You decide which Machine Learning Algorithm to choose for your classification problem?**

You can follow the below-mentioned guidelines to choose an algorithm for your problem:

1. For accuracy, test the different algorithms and cross-validate them.
2. If the training dataset is small, you can use models that have low variance and high bias.
3. If the training dataset is large, you can use the models which have high variance and low bias.

## **Q30. How will you implement facebook's “people you may know” using machine learning?**

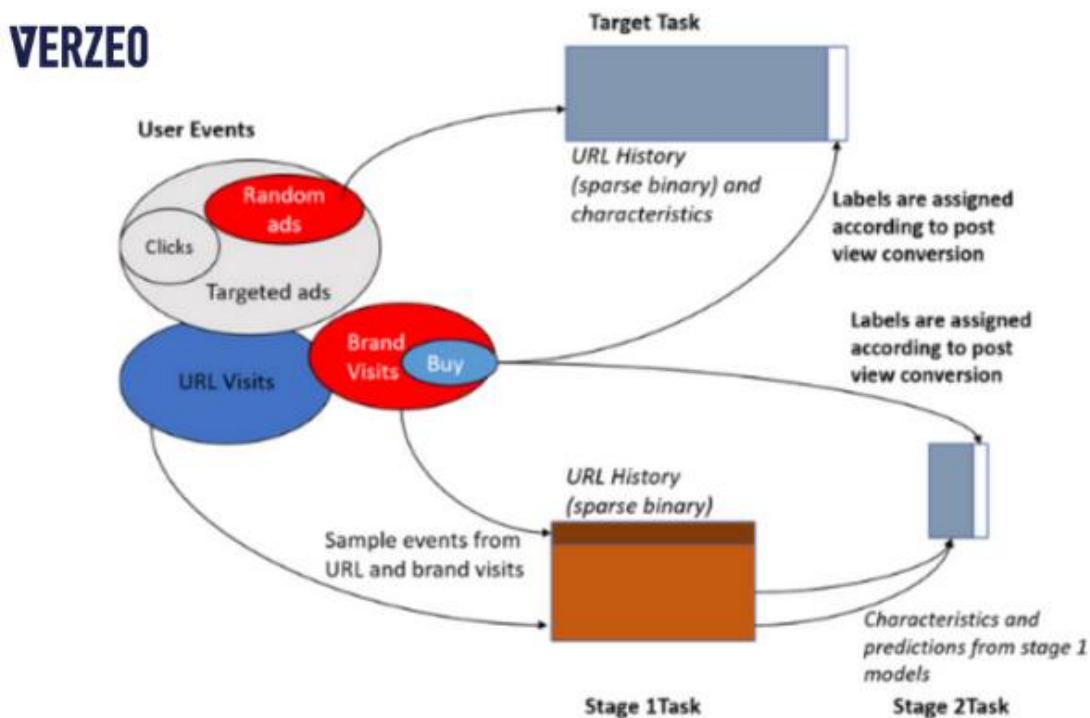
Facebook's “people you may know” works by inspecting the activity of users on its platform. The machine learning model evaluates the data and the internet activity of the users and keeps on storing it. Also, the model inspects the

existing friend list of the user, checks their friend list as well and gives suggestions based on it.

The unsupervised machine learning algorithm is used for this. Once started the model keeps learning on its own. A feedback system is used so that the model can evaluate the feedback and keep on learning from that to give more accurate and relevant results.

### **Q31. How machine learning powers targeted advertising?**

Targeted advertising is implemented using an unsupervised machine learning model. To do so the bot gathers certain information of the user, such as location, mouse movements, connected applications to show precisely targeted ads to the user. Google uses selective filtering features in which the Google search algorithm has a bias towards the client's products.



### **Q32. Give a brief overview of sentiment analysis using machine learning.**

Sentiment analysis involves the use of natural language processing to categorise opinions expressed usually in a piece of text, in order to determine whether the writer's attitude is positive, negative or neutral. Machine learning algorithms can be used to help ease the learning process of the bot. These learning algorithms are constantly fed with a massive amount of data so that it can adjust itself and continually improve.

**Q33. How many trigrams phrases can be generated from the given sentence, after performing the following text cleaning steps:**

**1. Stopword Removal**

**2. Replacing punctuations by a single space**

**“#Verzeo is a great source to learn @data\_science.”**

Ans. After performing stopword removal and punctuation replacement the text becomes: “Verzeo great source learn data science”.

Trigrams – Verzeo great, Verzeogreat source, great source learn, source learn data, learn data science.

**Q34. How would you implement a recommendation system for our company’s users?**

For this particular problem, you need to have complete knowledge about what the company does. You need to research the company thoroughly before you can answer this question. You need to know who all are the customers, what are their revenue channels etc. so that you can answer this question properly.

**Q35. How do you think Google is training data for self-driving cars?**

**Q1. What are the different types of machine learning?**

This is most important question in this Machine Learning Interview Questions answer series.

1. Supervised Learning
2. Unsupervised Learning
3. Reinforcement Learning

- **Supervised Learning**

In supervised learning used label data, it means to say we know the Output of our machine learning model. Train our machine learning model using training data and test on the testing.

Regression and classification type of problem solve by Supervised learning

- **Unsupervised learning**

Unsupervised learning uses unlabeled data, it means we don't know the exact output of the machine learning model and allow the model to act on that

information without guidance. Create a cluster of our data according to their feature similarities

Clustering types of problem solve by Unsupervised learning

- **Reinforcement Learning**

Reinforcement Learning is a part of Machine learning where an agent is put in an environment and he learns to behave in this environment by performing certain actions and observing the rewards which it gets from those actions.

Reward-based types of problems solved by Reinforcement Learning

## Q2. Explain Classification and Regression

- **Regression**:- In the regression type problem is solved continuous quality output, For example, if you want to predict years of experience VS Salary or Predict stock price over a period of time this type of problem is solved by Regression. This type of problem solved by supervised learning algorithms like linear regression and multi-liner regression.
- **Classification**:- In this type solve the categorical type of output, for example, to predict person has a disease or don does not have disease or mail is spam or not spam this type of problem statement is solved by supervise classification type of algorithm, like Logistic regression, k-nearest neighbors, etc.

## Q3. What is the confusion matrix?

		Predicted: NO	Predicted: YES	
n=165				
Actual:				
	NO	TN = 50	FP = 10	60
	YES	FN = 5	TP = 100	105
		55	110	

- **TN:- True Negative**
- **TP:- True Positive**
- **FP:- False Positive ( Type – I error )**
- **FN:- False Negative ( Type-II error )**

**Q4. Which is more important to you – model accuracy or model performance?**

Model Accuracy is the subset of model performance, hence the better the model performance means the better the model accuracy, so the accuracy and model performance is directly proportional.

**Q5. Explain false negative, false positive, true negative and true positive with a simple example.**

- **True Positives (TP):** These are cases in which we predicted yes (they have the disease), and they do have the disease.
- **True Negatives (TN):** We predicted no, and they don't have the disease.
- **False Positives (FP):** We predicted yes, but they don't actually have the disease. (Also known as a "Type I error.")
- **False Negatives (FN):** We predicted no, but they actually do have the disease. (Also known as a "Type II error.")

**Q6. Difference between K-Nearest neighbor and K Means**

- **K-Nearest neighbor:-** It is a type of supervised machine learning technique, Solved classification type od problem, k in the KNNis a number of nearest point.
- **K Means:-** It is Unsupervised machine learning technique, Solve clustering types of problem statement, K in the k means is number of clusters.

**Q7. What is the difference between Gini Impurity and Entropy in a Decision Tree?**

To find the best root note parameter in Decision tree used Gini Impurity and Entropy

Entropy is used to calculate randomness in data using this we find the information gain and whichever parameter is higher the importation is our root note a parameter.

Gini measurement is the probability of a random sample variable that also be used to select the root smallest value or Gini is considered as the root note.

### **Q8. What is the difference between Entropy and Information Gain?**

Entropy is just about the randomness in data. It decreases as you reach closer to the leaf node.

Information gain is used to reduce entropy of each attribute is also used to split out the dataset attribute, higher the values of information gain consider as the root node.

### **Q9. What is Overfitting? And how do you ensure you're not overfitting with a model?**

Overfitting occurs when a model highly train on training data but get a negative influence on testing data.

Methods to avoid overfitting

Use the ensemble methods such as random forest which reduces the variance in data by using a bagging method like uses multiple decision trees and combining their result.

Collect more data to train the varied sample data.

Use cross-validation technique.

Choose the right algorithm.

Widget not in any sidebars

### **Q10. Explain the Ensemble learning technique in Machine Learning.**

To create multiple Machine Learning models used Ensemble learning techniques, To get more accuracy combining these models, in ensemble learning split the training dataset into multiple subsets these multiple subsets build multiple separate models after the model is trained combining the all these model accuracies to get higher accuracy and predict an outcome in such a way that the variance in the output is reduced.

### **Q11. What are collinearity and multicollinearity?**

when two independent variables (e.g., x1 and x2) in a multiple regression have some correlation then there is collinearity occurs.

when more than two independent variables (e.g., x1, x2, and x3) are inter-correlated with each other then Multicollinearity occurs.

### **Q12. What is bagging and boosting in Machine Learning?**

Bagging tries to implement similar learners on small sample populations and then takes a mean of all the predictions.

In general, bagging you can use different learners on different populations. This helps us to reduce the variance error

Boosting is an iterative technique that adjusts the weight of an observation based on the last classification.

If an observation was classified incorrectly it tries to increase the weight of this observation and vice versa.

In general, decrease the bias error and builds strong predictive models.

### **Q13. What do you understand by Precision and Recall?**

**Precision:** When it predicts yes, how often is it correct?

$$\text{TP/predicted yes} = 100/110 = 0.91$$

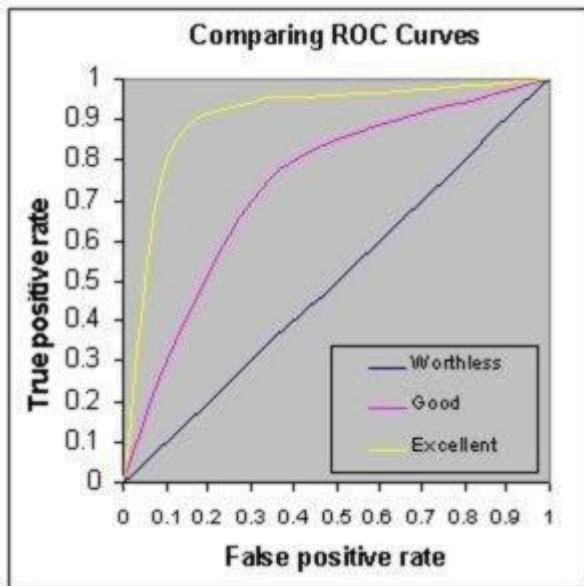
**True Positive Rate:** When it's actually yes, how often does it predict yes?

$$\text{TP/actual yes} = 100/105 = 0.95$$

Also known as "Sensitivity" or "Recall"

#### **Q14. What is ROC curve and what does it represent?**

*Receiver Operating Characteristic curve (or ROC curve) is a fundamental tool for diagnostic test evaluation and is a plot of the true positive rate (Sensitivity) against the false positive rate (Specificity) for the different possible cut-off points of a diagnostic test.*



#### **Q15. What is difference between R Square and R adjacent?**

**Coefficient of Determination (R Square)** It suggests the proportion of variation in Y which can be explained with the independent variables.  
Mathematically:

$$R^2 = \text{SSR/SST} \text{ or } R^2 = \text{Explained variation / Total variation}$$

**Adjusted R Square:-** Adding more independent variables or predictors to a regression model tends to increase the R-squared value, which tempts makers of the model to add even more. This is called overfitting and can return an unwarranted high R-squared value. Adjusted R-squared is used to determine how reliable the correlation is and how much is determined by the addition of independent variables.

$$R^2 \text{ adjusted} = 1 - \frac{(1-R^2)(N-1)}{N-p-1}$$

n = Number Of points in dataset

p = Number of independent variable in the model

#### **Q16. Assumption in linear regression?**

- Linear Relationship
- Autocorrelation
- Multivariate normality
- Homoscedasticity
- Multicollinearity

### **Q17. What Is autocorrelation? Which test is used to find autocorrelation?**

If the values of a column or feature is correlated with values of that same column then it is said to be autocorrelated, In other words, Correlation within a column.

Durbin-Watson(DW) Test is Generally used to check the Autocorrelation.

It has a range of 0 to 4

where 0-2 shows positive Autocorrelation

2 means NO Autocorrelation

and 2-4 means Negative Autocorrelation

Durbin Watson test works well with 1st order AutoCorrelation whereas Brusch-Godfrey test for(2,3,4 order)

### **Q18. Difference between homoscedasticity and heteroscedasticity?**

**Homoscedasticity:-** If the variance of the residual are symmetrically distributed across the residual line then data is said to be homoscedastic.

**Heteroscedasticity:-** If the variance is unequal for residual, across the residual line then the data is said to be Heteroscedasticity. In this case, the residual can form bow-tie, arrow, or any non-symmetric shape.

### **Q19. Difference Parameter and Hyperparameter?**

**Parameter:-** A model parameter is a configuration variable that is internal to the model and whose value can be estimated from the given data. They are required by the model when making predictions. Their values define the skill of

the model on your problem. They are estimated or learned from data. They are often not set manually by the practitioner. They are often saved as part of the learned model.

**Hyperparameter**:- A model hyperparameter is a configuration that is external to the model and whose value cannot be estimated from data. They are often used in processes to help estimate model parameters. They are often specified by the practitioner. They can often be set using heuristics. They are often tuned for a given predictive modeling problem

**Q20. Suppose you found that your model is suffering from low bias and high variance. Which algorithm you think could handle this situation and Why?**

The data is suffering from low bias and high variance means the model is Overfitting.

To handle this high variance parameter used random forest (Bagging method )

Bagging method has created a subset of data and build multiple decision trees to solve a particular problem, in classification type of problem get the prediction as majority of the class and the regression problem averaging the regression.

### **1.1 – What are parametric models? Give an example.**

*Parametric* models are those with a finite number of parameters. To predict new data, you only need to know the parameters of the model. Examples include linear regression, logistic regression, and linear SVMs.

*Non-parametric* models are those with an unbounded number of parameters, allowing for more flexibility. To predict new data, you need to know the parameters of the model and the state of the data that has been observed. Examples include decision trees, k-nearest neighbors, and topic models using latent dirichlet analysis.

### **1.2 – What is the “Curse of Dimensionality?”**

The difficulty of searching through a solution space becomes much harder as you have more features (dimensions).

Consider the analogy of looking for a penny in a line vs. a field vs. a building. The more dimensions you have, the higher volume of data you'll need.

### 1.3 – Explain the Bias-Variance Tradeoff.

Predictive models have a tradeoff between bias (how well the model fits the data) and variance (how much the model changes based on changes in the inputs).

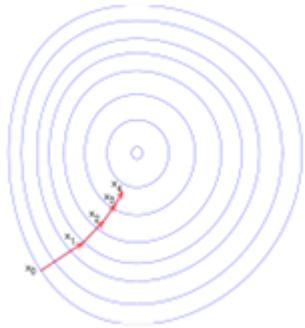
*Simpler models* are stable (low variance) but they don't get close to the truth (high bias).

More *complex models* are more prone to being overfit (high variance) but they are expressive enough to get close to the truth (low bias).

The best model for a given problem usually lies somewhere in the middle.

## 2. Optimization

*Algorithms for finding the best parameters for a model.*



### 2.1 – What is the difference between stochastic gradient descent (SGD) and gradient descent (GD)?

Both algorithms are methods for finding a set of parameters that minimize a loss function by evaluating parameters against data and then making adjustments.

In standard gradient descent, you'll evaluate all training samples for each set of parameters. This is akin to taking big, slow steps toward the solution.

In stochastic gradient descent, you'll evaluate only 1 training sample for the set of parameters before updating them. This is akin to taking small, quick steps toward the solution.

## 2.2 – When would you use GD over SDG, and vice-versa?

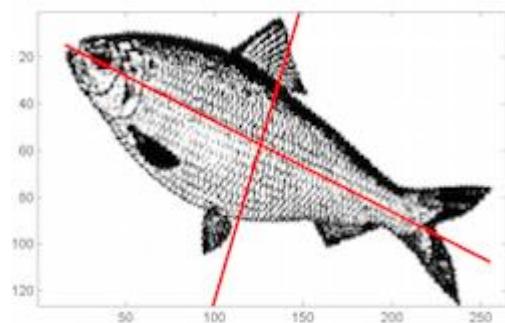
GD theoretically minimizes the error function better than SGD. However, SGD converges much faster once the dataset becomes large.

That means GD is preferable for small datasets while SGD is preferable for larger ones.

In practice, however, SGD is used for most applications because it minimizes the error function well enough while being much faster and more memory efficient for large datasets.

## 3. Data Preprocessing

*Dealing with missing data, skewed distributions, outliers, etc.*



## 3.1 – What is the Box-Cox transformation used for?

The Box-Cox transformation is a generalized “power transformation” that transforms data to make the distribution more normal.

For example, when its lambda parameter is 0, it's equivalent to the log-transformation.

It's used to stabilize the variance (eliminate heteroskedasticity) and normalize the distribution.

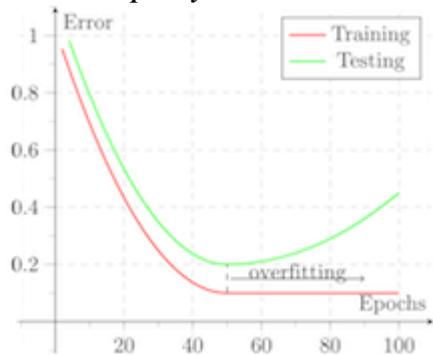
### 3.2 – What are 3 data preprocessing techniques to handle outliers?

1. Winsorize (cap at threshold).
2. Transform to reduce skew (using Box-Cox or similar).
3. Remove outliers if you're certain they are anomalies or measurement errors.

### 3.3 – What are 3 ways of reducing dimensionality?

1. Removing collinear features.
2. Performing PCA, ICA, or other forms of algorithmic dimensionality reduction.
3. Combining features with feature engineering.
4. Sampling & Splitting

*How to split your datasets to tune parameters and avoid overfitting.*



### 4.1 – How much data should you allocate for your training, validation, and test sets?

You have to find a balance, and there's no right answer for every problem.

If your test set is too small, you'll have an unreliable estimation of model performance (performance statistic will have high variance). If your training set is too small, your actual model parameters will have high variance.

A good rule of thumb is to use an 80/20 train/test split. Then, your train set can be further split into train/validation or into partitions for cross-validation.

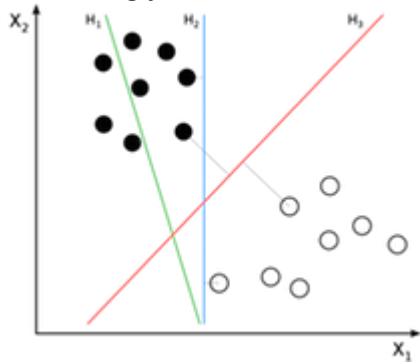
## 4.2 – If you split your data into train/test splits, is it still possible to overfit your model?

Yes, it's definitely possible. One common beginner mistake is re-tuning a model or training new models with different parameters after seeing its performance on the test set.

In this case, it's the model selection process that causes the overfitting. The test set should not be tainted until you're ready to make your final selection.

## 5. Supervised Learning

*Learning from labeled data using classification and regression models.*



### 5.1 – What are the advantages and disadvantages of decision trees?

*Advantages:* Decision trees are easy to interpret, nonparametric (which means they are robust to outliers), and there are relatively few parameters to tune.

*Disadvantages:* Decision trees are prone to be overfit. However, this can be addressed by ensemble methods like random forests or boosted trees.

### 5.2 – What are the advantages and disadvantages of neural networks?

*Advantages:* Neural networks (specifically deep NNs) have led to performance breakthroughs for unstructured datasets such as images, audio, and video. Their incredible flexibility allows them to learn patterns that no other ML algorithm can learn.

*Disadvantages:* However, they require a large amount of training data to converge. It's also difficult to pick the right architecture, and the internal “hidden” layers are incomprehensible.

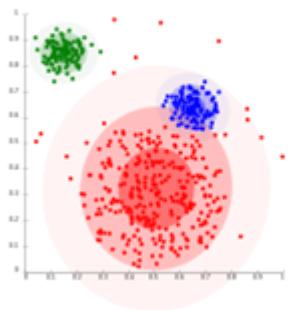
### 5.3 – How can you choose a classifier based on training set size?

If training set is small, high bias / low variance models (e.g. Naive Bayes) tend to perform better because they are less likely to be overfit.

If training set is large, low bias / high variance models (e.g. Logistic Regression) tend to perform better because they can reflect more complex relationships.

## 6. Unsupervised Learning

*Learning from unlabeled data using factor and cluster analysis models.*



### 6.1 – Explain Latent Dirichlet Allocation (LDA).

Latent Dirichlet Allocation (LDA) is a common method of topic modeling, or classifying documents by subject matter.

LDA is a generative model that represents documents as a mixture of topics that each have their own probability distribution of possible words.

The “Dirichlet” distribution is simply a distribution of distributions. In LDA, documents are distributions of topics that are distributions of words.

### 6.2 – Explain Principle Component Analysis (PCA).

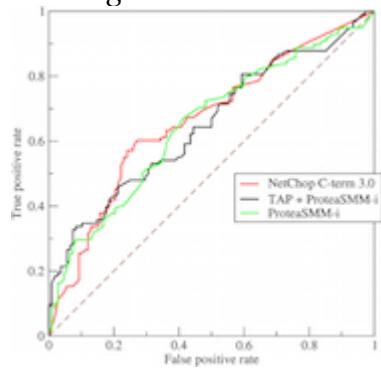
PCA is a method for transforming features in a dataset by combining them into uncorrelated linear combinations.

These new features, or principal components, sequentially maximize the variance represented (i.e. the first principal component has the most variance, the second principal component has the second most, and so on).

As a result, PCA is useful for dimensionality reduction because you can set an arbitrary variance cutoff.

## 7. Model Evaluation

*Making decisions based on various performance metrics.*



### 7.1 – What is the ROC Curve and what is AUC (a.k.a. AUROC)?

The ROC (receiver operating characteristic) is the performance plot for binary classifiers of True Positive Rate (y-axis) vs. False Positive Rate (x-axis).

AUC is area under the ROC curve, and it's a common performance metric for evaluating binary classification models.

It's equivalent to the expected probability that a uniformly drawn random positive is ranked before a uniformly drawn random negative.

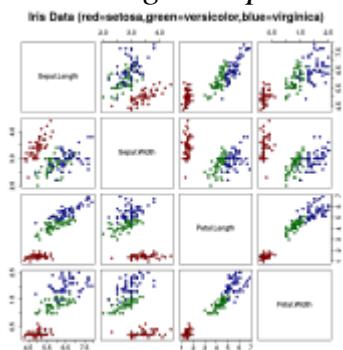
### 7.2 – Why is Area Under ROC Curve (AUROC) better than raw accuracy as an out-of-sample evaluation metric?

AUROC is robust to class imbalance, unlike raw accuracy.

For example, if you want to detect a type of cancer that's prevalent in only 1% of the population, you can build a model that achieves 99% accuracy by simply classifying everyone has cancer-free.

## 8. Ensemble Learning

*Combining multiple models for better performance.*



### 8.1 – Why are ensemble methods superior to individual models?

They average out biases, reduce variance, and are less likely to overfit.

There's a common line in machine learning which is: “ensemble and get 2%.”

This implies that you can build your models as usual and typically expect a small performance boost from ensembling.

### 8.2 – Explain bagging.

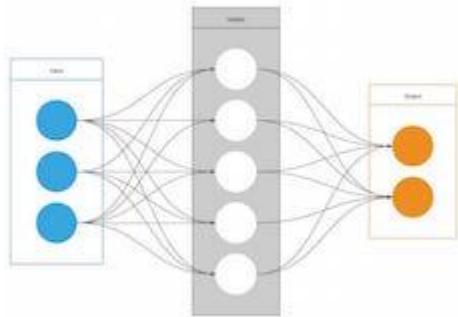
Bagging, or Bootstrap Aggregating, is an ensemble method in which the dataset is first divided into multiple subsets through resampling.

Then, each subset is used to train a model, and the final predictions are made through voting or averaging the component models.

Bagging is performed in parallel.

## 9. Business Applications

*How machine learning can help different types of businesses.*



### 9.1 – What are some key business metrics for (S-a-a-S startup | Retail bank | e-Commerce site)?

Thinking about key business metrics, often shortened as KPI's (Key Performance Indicators), is an essential part of a data scientist's job. Here are a few examples, but you should practice brainstorming your own.

- S-a-a-S startup: Customer lifetime value, new accounts, account lifetime, churn rate, usage rate, social share rate
- Retail bank: Offline leads, online leads, new accounts (segmented by account type), risk factors, product affinities
- e-Commerce: Product sales, average cart value, cart abandonment rate, email leads, conversion rate

### 9.2 – How can you help our marketing team be more efficient?

The answer will depend on the type of company. Here are some examples.

- Clustering algorithms to build custom customer segments for each type of marketing campaign.
- Natural language processing for headlines to predict performance before running ad spend.
- Predict conversion probability based on a user's website behavior in order to create better re-targeting campaigns.

1. What is the similarity between Hadoop and K?

2. If a linear regression model shows a 90% confidence interval, what does that mean?

3. A single-layer perceptron or a 2-layer decision tree, which one is superior in terms of expressiveness?
4. How can a neural network be used for dimensionality?
5. Name two utilities of the intercept term in linear regression?
6. Why do a majority of machine learning algorithms involve some kind of matrix manipulation?
7. Is time series really a simple linear regression problem with one response variable predictor?
8. Can it be mathematically proven that finding the optimal decision trees for a classification problem among all decisions trees is hard?
9. Which is easier, a deep neural network or a decision tree model?
10. Apart from back-propagation, what are some of the other alternative techniques to train a neural network?
11. How can one tackle the impact of correlation among predictors on principal component analysis?
12. Is there a way to work beyond the 99% accuracy mark on a classification model? 13. How can one capture the correlation between continuous and categorical variables?
14. Does k-fold cross-validation work well with time-series model?
15. Why can't simple random sampling of training data set and validation set work for a classification problem?
16. What should be a priority, a model accuracy or model performance?
17. What is your preferred approach for multiple CPU cores, boosted tree algorithm, or random forest?
18. What algorithm works best for tiny storage, logistic regression, or k-nearest neighbor?

19. What are the criteria to choose the right ML algorithm?.

20. Why can't logistic regression use more than 2 classes?

### **1) What's the trade-off between bias and variance?**

If our model is too simple and has very few parameters then it may have high bias and low variance. On the other hand if our model has large number of parameters then it's going to have high variance and low bias. So we need to find the right/good balance without overfitting and underfitting the data.

### **2) What is gradient descent?**

[Answer]

Gradient descent is an optimization algorithm used to find the values of parameters (coefficients) of a function ( $f$ ) that minimizes a cost function (cost).

Gradient descent is best used when the parameters cannot be calculated analytically (e.g. using linear algebra) and must be searched for by an optimization algorithm.

### **3) Explain over- and under-fitting and how to combat them?**

[Answer]

### **4) How do you combat the curse of dimensionality?**

- Manual Feature Selection
- Principal Component Analysis (PCA)
- Multidimensional Scaling
- Locally linear embedding

### **5) What is regularization, why do we use it, and give some examples of common methods?**

A technique that discourages learning a more complex or flexible model, so as to avoid the risk of overfitting. Examples

- Ridge (L2 norm)
- Lasso (L1 norm)

The obvious *disadvantage* of **ridge** regression, is model interpretability.

It will shrink the coefficients for least important predictors, very close to zero. But it will never make them exactly zero. In other words, the *final model will include all predictors*. However, in the case of the **lasso**, the L1 penalty has the effect of forcing some of the coefficient estimates to be *exactly equal* to zero when the tuning parameter  $\lambda$  is sufficiently large. Therefore, the lasso method also performs variable selection and is said to yield sparse models.

## 6) Explain Principal Component Analysis (PCA)?

[Answer]

## 7) Why is ReLU better and more often used than Sigmoid in Neural Networks? [

Imagine a network with random initialized weights ( or normalised ) and almost 50% of the network yields 0 activation because of the characteristic of ReLu ( output 0 for negative values of  $x$  ). This means a fewer neurons are firing ( sparse activation ) and the network is lighter.

## 8) Given stride S and kernel sizes for each layer of a (1-dimensional) CNN, create a function to compute the receptive field of a particular node in the network. This is just finding how many input nodes actually connect through to a neuron in a CNN.

The receptive field are defined portion of space within an inputs that will be used during an operation to generate an output.

Considering a CNN filter of size  $k$ , the receptive field of a peculiar layer is only the number of input used by the filter, in this case  $k$ , multiplied by the dimension of the input that is not being reduced by the convolutionnal filter  $a$ . This results in a receptive field of  $k*a$ .

More visually, in the case of an image of size  $32x32x3$ , with a CNN with a filter size of  $5x5$ , the corresponding receptive field will be the the filter size, 5 multiplied by the depth of the input volume (the RGB colors) which is the color dimensio. This thus gives us a recpetive field of dimension  $5x5x3$ .

## 9) Implement connected components on an image/matrix.

## 10) Implement a sparse matrix class in C++.

[Answer]

**11) Create a function to compute an integral image, and create another function to get area sums from the integral image.**

[Answer]

**12) How would you remove outliers when trying to estimate a flat plane from noisy samples?**

Random sample consensus (RANSAC) is an iterative method to estimate parameters of a mathematical model from a set of observed data that contains outliers, when outliers are to be accorded no influence on the values of the estimates.

**13) How does CBIR work?**

[Answer] Content-based image retrieval is the concept of using images to gather metadata on their content. Compared to the current image retrieval approach based on the keywords associated to the images, this technique generates its metadata from computer vision techniques to extract the relevant informations that will be used during the querying step. Many approach are possible from feature detection to retrieve keywords to the usage of CNN to extract dense features that will be associated to a known distribution of keywords.

With this last approach, we care less about what is shown on the image but more about the similarity between the metadata generated by a known image and a list of known label and or tags projected into this metadata space.

**14) How does image registration work? Sparse vs. dense optical flow and so on.**

**15) Describe how convolution works. What about if your inputs are grayscale vs RGB imagery? What determines the shape of the next layer?**

[Answer]

**16) Talk me through how you would create a 3D model of an object from imagery and depth sensor measurements taken at all angles around the object.**

**17) Implement SQRT(const double & x) without using any special functions, just fundamental arithmetic.**

The taylor series can be used for this step by providing an approximation of  $\sqrt{x}$ :

[Answer]

**18) Reverse a bitstring.**

If you are using python3 :

```
data = b'\xAD\xDE\xDE\xC0'  
my_data = bytearray(data)  
my_data.reverse()
```

**19) Implement non maximal suppression as efficiently as you can.**

**20) Reverse a linked list in place.**

[Answer]

**21) What is data normalization and why do we need it?**

Data normalization is very important preprocessing step, used to rescale values to fit in a specific range to assure better convergence during backpropagation. In general, it boils down to subtracting the mean of each data point and dividing by its standard deviation. If we don't do this then some of the features (those with high magnitude) will be weighted more in the cost function (if a higher-magnitude feature changes by 1%, then that change is pretty big, but for smaller features it's quite insignificant). The data normalization makes all features weighted equally.

**22) Why do we use convolutions for images rather than just FC layers?**

Firstly, convolutions preserve, encode, and actually use the spatial information from the image. If we used only FC layers we would have no relative spatial information. Secondly, Convolutional Neural Networks (CNNs) have a partially built-in translation in-variance, since each convolution kernel acts as its own filter/feature detector.

**23) What makes CNNs translation invariant?**

As explained above, each convolution kernel acts as its own filter/feature detector. So let's say you're doing object detection, it doesn't matter where in the image the object is since we're going to apply the convolution in a sliding window fashion across the entire image anyways.

**24) Why do we have max-pooling in classification CNNs?**

for a role in Computer Vision. Max-pooling in a CNN allows you to reduce computation since your feature maps are smaller after the pooling. You don't lose too much semantic information since you're taking the maximum activation. There's also a theory that max-pooling contributes a bit to giving CNNs more translation in-variance. Check out this great video from Andrew Ng on the benefits of max-pooling.

## **25) Why do segmentation CNNs typically have an encoder-decoder style / structure?**

The encoder CNN can basically be thought of as a feature extraction network, while the decoder uses that information to predict the image segments by "decoding" the features and upscaling to the original image size.

## **26) What is the significance of Residual Networks?**

The main thing that residual connections did was allow for direct feature access from previous layers. This makes information propagation throughout the network much easier. One very interesting paper about this shows how using local skip connections gives the network a type of ensemble multi-path structure, giving features multiple paths to propagate throughout the network.

## **27) What is batch normalization and why does it work?**

Training Deep Neural Networks is complicated by the fact that the distribution of each layer's inputs changes during training, as the parameters of the previous layers change. The idea is then to normalize the inputs of each layer in such a way that they have a mean output activation of zero and standard deviation of one. This is done for each individual mini-batch at each layer i.e compute the mean and variance of that mini-batch alone, then normalize. This is analogous to how the inputs to networks are standardized. How does this help? We know that normalizing the inputs to a network helps it learn. But a network is just a series of layers, where the output of one layer becomes the input to the next. That means we can think of any layer in a neural network as the first layer of a smaller subsequent network. Thought of as a series of neural networks feeding into each other, we normalize the output of one layer before applying the activation function, and then feed it into the following layer (sub-network).

## **28) Why would you use many small convolutional kernels such as 3x3 rather than a few large ones?**

This is very well explained in the VGGNet paper. There are 2 reasons: First, you can use several smaller kernels rather than few large ones to get the same

receptive field and capture more spatial context, but with the smaller kernels you are using less parameters and computations. Secondly, because with smaller kernels you will be using more filters, you'll be able to use more activation functions and thus have a more discriminative mapping function being learned by your CNN.

## **29) Why do we need a validation set and test set? What is the difference between them?**

When training a model, we divide the available data into three separate sets:

- The training dataset is used for fitting the model's parameters. However, the accuracy that we achieve on the training set is not reliable for predicting if the model will be accurate on new samples.
- The validation dataset is used to measure how well the model does on examples that weren't part of the training dataset. The metrics computed on the validation data can be used to tune the hyperparameters of the model. However, every time we evaluate the validation data and we make decisions based on those scores, we are leaking information from the validation data into our model. The more evaluations, the more information is leaked. So we can end up overfitting to the validation data, and once again the validation score won't be reliable for predicting the behaviour of the model in the real world.
- The test dataset is used to measure how well the model does on previously unseen examples. It should only be used once we have tuned the parameters using the validation set.

So if we omit the test set and only use a validation set, the validation score won't be a good estimate of the generalization of the model.

## **30) What is stratified cross-validation and when should we use it?**

Cross-validation is a technique for dividing data between training and validation sets. On typical cross-validation this split is done randomly. But in stratified cross-validation, the split preserves the ratio of the categories on both the training and validation datasets.

For example, if we have a dataset with 10% of category A and 90% of category B, and we use stratified cross-validation, we will have the same proportions in training and validation. In contrast, if we use simple cross-validation, in the worst case we may find that there are no samples of category A in the validation set.

Stratified cross-validation may be applied in the following scenarios:

- On a dataset with multiple categories. The smaller the dataset and the more imbalanced the categories, the more important it will be to use stratified cross-validation.
- On a dataset with data of different distributions. For example, in a dataset for autonomous driving, we may have images taken during the day and at night. If we do not ensure that both types are present in training and validation, we will have generalization problems.

### **31) Why do ensembles typically have higher scores than individual models?**

An ensemble is the combination of multiple models to create a single prediction. The key idea for making better predictions is that the models should make different errors. That way the errors of one model will be compensated by the right guesses of the other models and thus the score of the ensemble will be higher.

We need diverse models for creating an ensemble. Diversity can be achieved by:

- Using different ML algorithms. For example, you can combine logistic regression, k-nearest neighbors, and decision trees.
- Using different subsets of the data for training. This is called bagging.
- Giving a different weight to each of the samples of the training set. If this is done iteratively, weighting the samples according to the errors of the ensemble, it's called boosting. Many winning solutions to data science competitions are ensembles. However, in real-life machine learning projects, engineers need to find a balance between execution time and accuracy.

### **32) What is an imbalanced dataset? Can you list some ways to deal with it?**

An imbalanced dataset is one that has different proportions of target categories. For example, a dataset with medical images where we have to detect some illness will typically have many more negative samples than positive samples—say, 98% of images are without the illness and 2% of images are with the illness.

There are different options to deal with imbalanced datasets:

- Oversampling or undersampling. Instead of sampling with a uniform distribution from the training dataset, we can use other distributions so the model sees a more balanced dataset.

- Data augmentation. We can add data in the less frequent categories by modifying existing data in a controlled way. In the example dataset, we could flip the images with illnesses, or add noise to copies of the images in such a way that the illness remains visible.
- Using appropriate metrics. In the example dataset, if we had a model that always made negative predictions, it would achieve a precision of 98%. There are other metrics such as precision, recall, and F-score that describe the accuracy of the model better when using an imbalanced dataset.

### **33) Can you explain the differences between supervised, unsupervised, and reinforcement learning?**

In supervised learning, we train a model to learn the relationship between input data and output data. We need to have labeled data to be able to do supervised learning.

With unsupervised learning, we only have unlabeled data. The model learns a representation of the data. Unsupervised learning is frequently used to initialize the parameters of the model when we have a lot of unlabeled data and a small fraction of labeled data. We first train an unsupervised model and, after that, we use the weights of the model to train a supervised model.

In reinforcement learning, the model has some input data and a reward depending on the output of the model. The model learns a policy that maximizes the reward. Reinforcement learning has been applied successfully to strategic games such as Go and even classic Atari video games.

### **34) What is data augmentation? Can you give some examples?**

Data augmentation is a technique for synthesizing new data by modifying existing data in such a way that the target is not changed, or it is changed in a known way.

Computer vision is one of fields where data augmentation is very useful. There are many modifications that we can do to images:

- Resize
- Horizontal or vertical flip
- Rotate
- Add noise
- Deform

- Modify colors Each problem needs a customized data augmentation pipeline. For example, on OCR, doing flips will change the text and won't be beneficial; however, resizes and small rotations may help.

### **35) What is Turing test?**

The Turing test is a method to test the machine's ability to match the human level intelligence. A machine is used to challenge the human intelligence that when it passes the test, it is considered as intelligent. Yet a machine could be viewed as intelligent without sufficiently knowing about people to mimic a human.

### **36) What is Precision?**

Precision (also called positive predictive value) is the fraction of relevant instances among the retrieved instances

$$\text{Precision} = \text{true positive} / (\text{true positive} + \text{false positive})$$

[src]

### **37) What is Recall?**

Recall (also known as sensitivity) is the fraction of relevant instances that have been retrieved over the total amount of relevant instances. Recall = true positive / (true positive + false negative)

[src]

### **38) Define F1-score.**

It is the weighted average of precision and recall. It considers both false positive and false negative into account. It is used to measure the model's performance.

$$\text{F1-Score} = 2 * (\text{precision} * \text{recall}) / (\text{precision} + \text{recall})$$

### **39) What is cost function?**

Cost function is a scalar functions which Quantifies the error factor of the Neural Network. Lower the cost function better the Neural network. Eg:  
MNIST Data set to classify the image, input image is digit 2 and the Neural network wrongly predicts it to be 3

### **40) List different activation neurons or functions.**

- Linear Neuron
- Binary Threshold Neuron

- Stochastic Binary Neuron
- Sigmoid Neuron
- Tanh function
- Rectified Linear Unit (ReLU)

#### **41) Define Learning Rate.**

Learning rate is a hyper-parameter that controls how much we are adjusting the weights of our network with respect the loss gradient.

#### **42) What is Momentum (w.r.t NN optimization)?**

Momentum lets the optimization algorithm remembers its last step, and adds some proportion of it to the current step. This way, even if the algorithm is stuck in a flat region, or a small local minimum, it can get out and continue towards the true minimum.

#### **43) What is the difference between Batch Gradient Descent and Stochastic Gradient Descent?**

Batch gradient descent computes the gradient using the whole dataset. This is great for convex, or relatively smooth error manifolds. In this case, we move somewhat directly towards an optimum solution, either local or global. Additionally, batch gradient descent, given an annealed learning rate, will eventually find the minimum located in it's basin of attraction.

Stochastic gradient descent (SGD) computes the gradient using a single sample. SGD works well (Not well, I suppose, but better than batch gradient descent) for error manifolds that have lots of local maxima/minima. In this case, the somewhat noisier gradient calculated using the reduced number of samples tends to jerk the model out of local minima into a region that hopefully is more optimal.

#### **44) Epoch vs. Batch vs. Iteration.**

- **Epoch:** one forward pass and one backward pass of **all** the training examples
- **Batch:** examples processed together in one pass (forward and backward)
- **Iteration:** number of training examples / Batch size

#### **45) What is vanishing gradient?**

As we add more and more hidden layers, back propagation becomes less and less useful in passing information to the lower layers. In effect, as information is passed back, the gradients begin to vanish and become small relative to the weights of the networks.

#### **46) What are dropouts?**

Dropout is a simple way to prevent a neural network from overfitting. It is the dropping out of some of the units in a neural network. It is similar to the natural reproduction process, where the nature produces offsprings by combining distinct genes (dropping out others) rather than strengthening the co-adapting of them.

#### **47) Define LSTM.**

Long Short Term Memory – are explicitly designed to address the long term dependency problem, by maintaining a state what to remember and what to forget.

#### **48) List the key components of LSTM.**

- Gates (forget, Memory, update & Read)
- $\tanh(x)$  (values between -1 to 1)
- Sigmoid( $x$ ) (values between 0 to 1)

#### **49) List the variants of RNN.**

- LSTM: Long Short Term Memory
- GRU: Gated Recurrent Unit
- End to End Network
- Memory Network

#### **50) What is Autoencoder, name few applications.**

Auto encoder is basically used to learn a compressed form of given data. Few applications include

- Data denoising
- Dimensionality reduction
- Image reconstruction
- Image colorization

## **51) What are the components of GAN?**

- Generator
- Discriminator

## **52) What's the difference between boosting and bagging?**

Boosting and bagging are similar, in that they are both ensembling techniques, where a number of weak learners (classifiers/regressors that are barely better than guessing) combine (through averaging or max vote) to create a strong learner that can make accurate predictions. Bagging means that you take bootstrap samples (with replacement) of your data set and each sample trains a (potentially) weak learner. Boosting, on the other hand, uses all data to train each learner, but instances that were misclassified by the previous learners are given more weight so that subsequent learners give more focus to them during training.

## **53) Explain how a ROC curve works.**

The ROC curve is a graphical representation of the contrast between true positive rates and the false positive rate at various thresholds. It's often used as a proxy for the trade-off between the sensitivity of the model (true positives) vs the fall-out or the probability it will trigger a false alarm (false positives).

## **54) What's the difference between Type I and Type II error?**

Type I error is a false positive, while Type II error is a false negative. Briefly stated, Type I error means claiming something has happened when it hasn't, while Type II error means that you claim nothing is happening when in fact something is. A clever way to think about this is to think of Type I error as telling a man he is pregnant, while Type II error means you tell a pregnant woman she isn't carrying a baby.

## **55) What's the difference between a generative and discriminative model?**

A generative model will learn categories of data while a discriminative model will simply learn the distinction between different categories of data. Discriminative models will generally outperform generative models on classification tasks.

## **56) Instance-Based Versus Model-Based Learning.**

- **Instance-based Learning:** The system learns the examples by heart, then generalizes to new cases using a similarity measure.
- **Model-based Learning:** Another way to generalize from a set of examples is to build a model of these examples, then use that model to make predictions. This is called model-based learning.

### **57) When to use a Label Encoding vs. One Hot Encoding?**

This question generally depends on your dataset and the model which you wish to apply. But still, a few points to note before choosing the right encoding technique for your model:

We apply One-Hot Encoding when:

- The categorical feature is not ordinal (like the countries above)
  - The number of categorical features is less so one-hot encoding can be effectively applied
- We apply Label Encoding when:
- The categorical feature is ordinal (like Jr. kg, Sr. kg, Primary school, high school)
  - The number of categories is quite large as one-hot encoding can lead to high memory consumption

### **58) What is the difference between LDA and PCA for dimensionality reduction?**

Both LDA and PCA are linear transformation techniques: LDA is a supervised whereas PCA is unsupervised – PCA ignores class labels.

We can picture PCA as a technique that finds the directions of maximal variance. In contrast to PCA, LDA attempts to find a feature subspace that maximizes class separability.

### **59) What is t-SNE?**

t-Distributed Stochastic Neighbor Embedding (t-SNE) is an unsupervised, non-linear technique primarily used for data exploration and visualizing high-dimensional data. In simpler terms, t-SNE gives you a feel or intuition of how the data is arranged in a high-dimensional space.

## **60) What is the difference between t-SNE and PCA for dimensionality reduction?**

The first thing to note is that PCA was developed in 1933 while t-SNE was developed in 2008. A lot has changed in the world of data science since 1933 mainly in the realm of compute and size of data. Second, PCA is a linear dimension reduction technique that seeks to maximize variance and preserves large pairwise distances. In other words, things that are different end up far apart. This can lead to poor visualization especially when dealing with non-linear manifold structures. Think of a manifold structure as any geometric shape like: cylinder, ball, curve, etc.

t-SNE differs from PCA by preserving only small pairwise distances or local similarities whereas PCA is concerned with preserving large pairwise distances to maximize variance.

## **61) What is UMAP?**

UMAP (Uniform Manifold Approximation and Projection) is a novel manifold learning technique for dimension reduction. UMAP is constructed from a theoretical framework based in Riemannian geometry and algebraic topology. The result is a practical scalable algorithm that applies to real world data.

## **62) What is the difference between t-SNE and UMAP for dimensionality reduction?**

The biggest difference between the the output of UMAP when compared with t-SNE is this balance between local and global structure - UMAP is often better at preserving global structure in the final projection. This means that the inter-cluster relations are potentially more meaningful than in t-SNE. However, it's important to note that, because UMAP and t-SNE both necessarily warp the high-dimensional shape of the data when projecting to lower dimensions, any given axis or distance in lower dimensions still isn't directly interpretable in the way of techniques such as PCA.

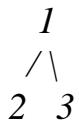
## **63) How Random Number Generator Works, e.g. `rand()` function in python works?**

It generates a pseudo random number based on the seed and there are some famous algorithm, please see below link for further information on this.

Given a binary tree, find the maximum path sum. The path may start and end at any node in the tree.

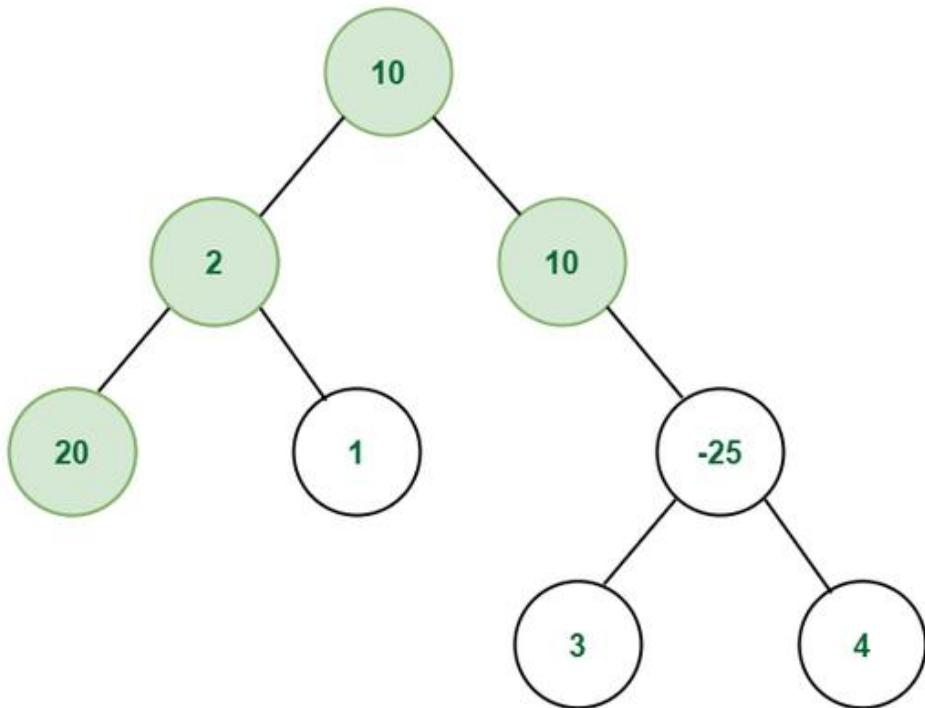
**Example:**

**Input:** Root of below tree



**Output:** 6

**Input:**



*Binary tree*

**Output:** 42

**Explanation:** Max path sum is represented using green color nodes in the above binary tree

**Approach:** To solve the problem follow the below idea:

For each node there can be four ways that the max path goes through the node:

- Node only

- *Max path through Left Child + Node*
- *Max path through Right Child + Node*
- *Max path through Left Child + Node + Max path through Right Child*

*The idea is to keep track of four paths and pick up the max one in the end. An important thing to note is, that the root of every subtree needs to return the maximum path sum such that at most one child of the root is involved. This is needed for the parent function call. In the below code, this sum is stored in 'max\_single' and returned by the recursive function.*

Follow the given steps to solve the problem:

- If the root is NULL, return 0(Base Case)
- Call the recursive function to find the max sum for the left and the right subtree
- In a variable store the maximum of (root->data, maximum of (leftSum, rightSum) + root->data)
- In another variable store the maximum of previous step and root->data + leftSum + rightSum
- Return the maximum of the previous step

Below is the implementation of the above approach:

class Node:

```
# Constructor to create a new node
def __init__(self, data):
    self.data = data
    self.left = None
    self.right = None

# This function returns overall maximum path sum in 'res'
# And returns max path sum going through root
```

def findMaxUtil(root):

```
# Base Case
if root is None:
    return 0

# l and r store maximum path sum going through left
# and right child of root respectively
l = findMaxUtil(root.left)
r = findMaxUtil(root.right)
```

```

# Max path for parent call of root. This path
# must include at most one child of root
max_single = max(max(l, r) + root.data, root.data)

# Max top represents the sum when the node under
# consideration is the root of the maxSum path and
# no ancestor of root are there in max sum path
max_top = max(max_single, l+r + root.data)

# Static variable to store the changes
# Store the maximum result
findMaxUtil.res = max(findMaxUtil.res, max_top)

return max_single

# Return maximum path sum in tree with given root

def findMaxSum(root):

    # Initialize result
    findMaxUtil.res = float("-inf")

    # Compute and return result
    findMaxUtil(root)
    return findMaxUtil.res

# Driver code
if __name__ == '__main__':
    root = Node(10)
    root.left = Node(2)
    root.right = Node(10)
    root.left.left = Node(20)
    root.left.right = Node(1)
    root.right.right = Node(-25)
    root.right.right.left = Node(3)
    root.right.right.right = Node(4)

    # Function call
    print "Max path sum is ", findMaxSum(root)

```

## Output

Max path sum is 42

"Given an encoded string, return its decoded string."

The encoding rule is:  $k[\text{encoded\_string}]$ , where the `encoded_string` inside the square brackets is being repeated exactly  $k$  times. Note that  $k$  is guaranteed to be a positive integer.

You may assume that the input string is always valid; there are no extra white spaces, square brackets are well-formed, etc. Furthermore, you may assume that the original data does not contain any digits and that digits are only for those repeat numbers,  $k$ . For example, there will not be input like `3a` or `2[4]`.

The test cases are generated so that the length of the output will never exceed  $10^5$ .

### Example 1:

**Input:** `s = "3[a]2[bc]"`

**Output:** "aaabcbc"

### Example 2:

**Input:** `s = "3[a2[c]]"`

**Output:** "accaccacc"

### Example 3:

**Input:** `s = "2[abc]3[cd]ef"`

**Output:** "abcabcccdcdcddef"

### Constraints:

- $1 \leq s.length \leq 30$
- `s` consists of lowercase English letters, digits, and square brackets '[]'.
- `s` is guaranteed to be **a valid** input.
- All the integers in `s` are in the range [1, 300].
- class Solution:
- def decodeString(self, s):
- it, num, stack = 0, 0, [""]
- while it < len(s):

```

•     if s[it].isdigit():
•         num = num * 10 + int(s[it])
•     elif s[it] == "[":
•         stack.append(num)
•         num = 0
•         stack.append("")
•     elif s[it] == "]":
•         str1 = stack.pop()
•         rep = stack.pop()
•         str2 = stack.pop()
•         stack.append(str2 + str1 * rep)
•     else:
•         stack[-1] += s[it]
•     it += 1
• return "".join(stack)

```

(or)

```

class Solution(object):
    def decodeString(self, s):
        """
        :type s: str
        :rtype: str
        """
        stack = []
        for i in range(len(s)):
            if not stack:
                stack.append(s[i])
                continue
            if s[i] == ']':
                currStr = ""
                while stack:
                    currChar = stack.pop()
                    if currChar == '[':
                        count = 0
                        power = 1
                        while stack:
                            n = stack.pop()
                            if not n.isnumeric():
                                stack.append(n)
                                break
                            count += int(n)*power
                            power *= 10
                    else:
                        currStr += currChar
                stack.append(currStr)
            else:
                stack.append(s[i])

```

```

currStr *= count
stack.extend([c for c in currStr])
break
currStr = currChar + currStr
else:
    stack.append(s[i])

return ''.join(stack)

```

### Question

"We can rotate digits by 180 degrees to form new digits. When 0, 1, 6, 8, 9 are rotated 180 degrees, they become 0, 1, 9, 8, 6 respectively. When 2, 3, 4, 5, and 7 are rotated 180 degrees, they become invalid. A *confusing number* is a number that when rotated 180 degrees becomes a different number with each digit valid.(Note that the rotated number can be greater than the original number.) Given a positive integer  $N$ , return the number of confusing numbers between 1 and  $N$  inclusive."?

### Question

Q A **transformation sequence** from word `beginWord` to word `endWord` using a dictionary `wordList` is a sequence of words `beginWord` ->  $s_1$  ->  $s_2$  -> ... ->  $s_k$  such that:

- Every adjacent pair of words differs by a single letter.
- Every  $s_i$  for  $1 \leq i \leq k$  is in `wordList`. Note that `beginWord` does not need to be in `wordList`.
- $s_k == endWord$

"Given two words (`beginWord` and `endWord`), and a dictionary's word list, find the length of the shortest transformation sequence from `beginWord` to `endWord`, such that: 1) Only one letter can be changed at a time, and 2) Each transformed word must exist in the word list."

#### **Example 1:**

**Input:** `beginWord` = "hit", `endWord` = "cog", `wordList` = ["hot", "dot", "dog", "lot", "log", "cog"]

**Output:** 5

**Explanation:** One shortest transformation sequence is "hit" -> "hot" -> "dot" -> "dog" -> "cog", which is 5 words long.

#### **Example 2:**

**Input:** beginWord = "hit", endWord = "cog", wordList = ["hot", "dot", "dog", "lot", "log"]

**Output:** 0

**Explanation:** The endWord "cog" is not in wordList, therefore there is no valid transformation sequence.

### Constraints:

- $1 \leq \text{beginWord.length} \leq 10$
- $\text{endWord.length} == \text{beginWord.length}$
- $1 \leq \text{wordList.length} \leq 5000$
- $\text{wordList[i].length} == \text{beginWord.length}$
- beginWord, endWord, and wordList[i] consist of lowercase English letters.
- $\text{beginWord} \neq \text{endWord}$
- All the words in wordList are **unique**.

### Solution:

```
• def ladderLength(self, beginWord: str, endWord: str, wordList: List[str])  
    -> int:  
•     # set of neighbors and nodes  
•     nodes = defaultdict(set)  
•     neighbors = defaultdict(set)  
•  
•     # for word in the list of beginWord with wordList unpacked so that you  
•     # only have beginWord once  
•     for word in [beginWord, *wordList]:  
•         # for i in range length of the word  
•         for i in range(len(word)):  
•             # nodes -> dictionary with key of word from forward to i with  
•             # word from i+1 forward to end, i --> set words  
•             # nodes as a graph has the key of word[:i]+word[i+:], i for the  
•             # key --> set words  
•             nodes[word[:i] + word[i+:], i].add(word)  
•  
•             # nodes is our graph with keys of specific words and indices -> words  
•             # this allows us to have specific breakdowns that can be done by index  
•             # this makes the union process below possible to facilitate  
•  
•             # loop over list again
```

```

•   for word in [beginWord, *wordList]:
•       # loop over length of the word
•       for i in range(len(word)):
•           # neighbors at word is union of ns[w[:i] + w[i+1:], i]
•           # union the words that match at ns for word up to i and word past
•               i using index i
•               neighbors[word] |= nodes[word[:i] + word[i+1:], i]
•           # neighbors at word has word discarded since it does not need to
•               keep itself
•           neighbors[word].discard(word)
•
•
•       # distance function
•       def dist(word):
•           # return sum of 1 for char_word, char_end in zip of word, endWord
•           if char_word != char_end
•               # this is the overall edit distance between the words
•               # basically, the same as looping over the words and if they are not
•                   equal at the specific char, add 1 to a list and
•               # at the end of it all, sum it up. This might not be very friendly
•                   timewise actually.
•           # This is also our heuristic distance for the a* algorithm
•           return sum(1 for char_word, char_end in zip(word, endWord) if
•               char_word != char_end)
•
•
•       # queue is [(1+dist(beginWord), 1, beginWord)]
•       # this is a list of tuples, which lets us use heaps.
•       queue = [(1 + dist(beginWord), 1, beginWord)]
•       # visited is the set containing the beginWord
•       visited = set([beginWord])
•
•
•       # while you have a queue
•       while queue:
•           # _, distance, word is a heappop of q. Basically, using the forward
•               distance to get us the priority
•           # means the _ can be discarded here
•           _, distance, word = heappop(queue)
•
•
•           # if w is endWord, return distance as the distance between the words
•           if word == endWord:
•               return distance
•
•
•           # for neighbor in neighbors at word

```

```

•   for neighbor in neighbors[word]:
•       # if neighbor not in visited
•       if neighbor not in visited:
•           # add neighbor to the visited set
•           visited.add(neighbor)
•           # heappush the neighbor into the queue using the distance + 1 +
•           # distance of the neighbor, distance+1 and the neighbor
•           heappush(queue, (distance + 1 + dist(neighbor), distance + 1,
•                             neighbor))
•
•       # at the end of 1 loop pass, you'll have added all the neighbors to the
•       # current queue and it will go from there to find the rest
•
•   # if you have not returned, return 0
•   return 0

```

"Given a matrix of N rows and M columns. From  $m[i][j]$ , we can move to  $m[i+1][j]$ , if  $m[i+1][j] > m[i][j]$ , or can move to  $m[i][j+1]$  if  $m[i][j+1] > m[i][j]$ . The task is print longest path length if we start from  $(0, 0)$ ."

### Examples:

Input :  $N = 4, M = 4$

```

m[][] = { { 1, 2, 3, 4 },
          { 2, 2, 3, 4 },
          { 3, 2, 3, 4 },
          { 4, 5, 6, 7 } };

```

Output : 7

Longest path is 1 2 3 4 5 6 7.

Input :  $N = 2, M = 2$

```

m[][] = { { 1, 2 },
          { 3, 4 } };

```

Output : 3

Longest path is either 1 2 4 or

1 3 4.

The idea is to use dynamic programming. Maintain the 2D matrix,  $dp[][]$ , where  $dp[i][j]$  store the value of the length of the longest increasing sequence for submatrix starting from the  $i$ th row and  $j$ th column.

Let the longest increasing sub sequence values for  $m[i+1][j]$  and  $m[i][j+1]$  be known already as  $v_1$  and  $v_2$  respectively. Then the value for  $m[i][j]$  will be  $\max(v_1, v_2) + 1$ .

We can start from  $m[n-1][m-1]$  as the base case with the length of longest increasing subsequence be 1, moving upwards and leftwards updating the value of cells. Then the LIP value for cell  $m[0][0]$  will be the answer.

```
# Python3 program to find longest
# increasing path in a matrix.
MAX = 20

# Return the length of
# LIP in 2D matrix
def LIP(dp, mat, n, m, x, y):

    # If value not calculated yet.
    if (dp[x][y] < 0):
        result = 0

        # // If reach bottom right cell, return 1
        if (x == n - 1 and y == m - 1):
            dp[x][y] = 1
            return dp[x][y]

    # If reach the corner
    # of the matrix.
    if (x == n - 1 or y == m - 1):
        result = 1

    # If value greater than below cell.
    if (x + 1 < n and mat[x][y] < mat[x + 1][y]):
        result = 1 + LIP(dp, mat, n,
                          m, x + 1, y)

    # If value greater than left cell.
    if (y + 1 < m and mat[x][y] < mat[x][y + 1]):
        result = max(result, 1 + LIP(dp, mat, n,
                                      m, x, y + 1))

    dp[x][y] = result
return dp[x][y]
```

```
# Wrapper function
def wrapper(mat, n, m):
    dp = [[-1 for i in range(MAX)]
          for i in range(MAX)]
    return LIP(dp, mat, n, m, 0, 0)
```

```
# Driver Code
mat = [[1, 2, 3, 4 ],
       [2, 2, 3, 4 ],
       [3, 2, 3, 4 ],
       [4, 5, 6, 7 ]]
n = 4
m = 4
print(wrapper(mat, n, m))
```

### Output

7

Implement a SnapshotArray that supports pre-defined interfaces (note: see link for more details).

- `SnapshotArray(int length)` initializes an array-like data structure with the given length. **Initially, each element equals 0.**
- `void set(index, val)` sets the element at the given index to be equal to val.
- `int snap()` takes a snapshot of the array and returns the `snap_id`: the total number of times we called `snap()` minus 1.
- `int get(index, snap_id)` returns the value at the given index, at the time we took the snapshot with the given `snap_id`

### Example 1:

**Input:** ["SnapshotArray","set","snap","set","get"]  
[[3],[0,5],[],[0,6],[0,0]]

**Output:** [null,null,0,null,5]

### Explanation:

```
SnapshotArray snapshotArr = new SnapshotArray(3); // set the length to be 3
snapshotArr.set(0,5); // Set array[0] = 5
snapshotArr.snap(); // Take a snapshot, return snap_id = 0
snapshotArr.set(0,6);
snapshotArr.get(0,0); // Get the value of array[0] with snap_id = 0, return 5
```

## Constraints:

- $1 \leq \text{length} \leq 5 * 10^4$
- $0 \leq \text{index} < \text{length}$
- $0 \leq \text{val} \leq 10^9$
- $0 \leq \text{snap\_id} < (\text{the total number of times we call snap()})$
- At most  $5 * 10^4$  calls will be made to set, snap, and get.
- class SnapshotArray(object):
  - 
  - def \_\_init\_\_(self, n):
    - self.A = [[[ -1, 0 ]]] for \_ in xrange(n)]
    - self.snap\_id = 0
  - 
  - def set(self, index, val):
    - self.A[index].append([self.snap\_id, val])
  - 
  - def snap(self):
    - self.snap\_id += 1
    - return self.snap\_id - 1
  - 
  - def get(self, index, snap\_id):
    - i = bisect.bisect(self.A[index], [snap\_id + 1]) - 1
    - return self.A[index][i][1]

or

```
class SnapshotArray:  
    def __init__(self, length: int):  
        self.map = defaultdict(list)  
        self.snapId = 0  
  
    def set(self, index: int, val: int) -> None:  
        if self.map[index] and self.map[index][-1][0] == self.snapId:  
            self.map[index][-1][1] = val  
        else:  
            self.map[index].append([self.snapId, val])  
  
    def snap(self) -> int:  
        self.snapId += 1  
        return self.snapId - 1  
  
    def get(self, index: int, snap_id: int) -> int:  
        arr = self.map[index]
```

```

left, right, ans = 0, len(arr) - 1, -1
while left <= right:
    mid = (left + right) // 2
    if arr[mid][0] <= snap_id:
        ans = mid
        left = mid + 1
    else:
        right = mid - 1
if ans == -1: return 0
return arr[ans][1]

```

## Approach #1. Brutal Force

- Straight forward solution, actual do snap every time `snap` is called
- This is quick to access, but take lots of extra space and it takes time to take snap shot, as we need to make a copy

```

class SnapshotArray:
    def __init__(self, length: int):
        self.cache = []
        self.d = dict()
        self.i = 0

```

```

    def set(self, index: int, val: int) -> None:
        self.d[index] = val

```

```

    def snap(self) -> int:
        self.cache.append(dict(self.d))
        self.i += 1
        return self.i-1

```

```

    def get(self, index: int, snap_id: int) -> int:
        snap = self.cache[snap_id]
        return snap[index] if index in snap else 0

```

## Approach #2. `defaultdict` + `OrderedDict` + Binary Search

- Take individual snap shot when `set` is called, increment snap id (`self.i`), when `snap` is called
- This is fast to `set` & `snap` but relatively slow when you do an `get`
  - even if it's binary search, make `keys` indexable take time
- It save space too

```

class SnapshotArray:

```

```

def __init__(self, length: int):
    self.cache = collections.defaultdict(lambda : collections.OrderedDict())
    self.i = 0

def set(self, index: int, val: int) -> None:
    self.cache[index][self.i] = val

def snap(self) -> int:
    self.i += 1
    return self.i-1

@lru_cache(maxsize=None)
def get(self, index: int, snap_id: int) -> int:
    if index not in self.cache: return 0
    else:
        idx_cache = self.cache[index]
        if snap_id in idx_cache: return idx_cache[snap_id]
        else:
            keys = list(idx_cache.keys())
            i = bisect.bisect(keys, snap_id)
            if snap_id > keys[-1]: return idx_cache[keys[-1]]
            elif i == 0: return 0
            else: return idx_cache[keys[i-1]]

```

### Approach #3. List of list + Binary Search

- A little combination of two above
- Space used in greater than approach #2 and less than approach #1
- Fast to `set` and `snap`, `get` speed should be faster than approach #2 but slower than approach #1

```

class SnapshotArray(object):
    def __init__(self, n):
        self.cache = [[[ -1, 0]] for _ in range(n)]
        self.i = 0

    def set(self, index, val):
        self.cache[index].append([self.i, val])

    def snap(self):
        self.i += 1
        return self.i - 1

```

```

@lru_cache(maxsize=None)
def get(self, index, snap_id):
    i = bisect.bisect(self.cache[index], [snap_id + 1]) - 1
    return self.cache[index][i][1]

```

"In a row of dominoes, A[i] and B[i] represent the top and bottom halves of the i-th domino. (A domino is a tile with two numbers from 1 to 6 - one on each half of the tile.) We may rotate the i-th domino, so that A[i] and B[i] swap values. Return the minimum number of rotations so that all the values in A are the same, or all the values in B are the same. If it cannot be done, return -1."

### 1007. Minimum Domino Rotations For Equal Row

Medium    12    10    Favorite    Share

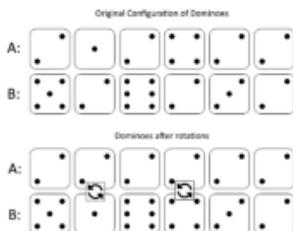
In a row of dominoes, A[i] and B[i] represent the top and bottom halves of the i-th domino. (A domino is a tile with two numbers from 1 to 6 - one on each half of the tile.)

We may rotate the i-th domino, so that A[i] and B[i] swap values.

Return the minimum number of rotations so that all the values in A are the same, or all the values in B are the same.

If it cannot be done, return -1 .

#### Example 1:



**Input:** A = [2,1,2,4,2,2], B = [5,2,6,2,3,2]

**Output:** 2

**Explanation:**

The first figure represents the dominoes as given by A and B: before we do any rotations. If we rotate the second and fourth dominoes, we can make every value in the top row equal to 2, as indicated by the second figure.

#### Example 2:

**Input:** A = [3,5,1,2,3], B = [3,6,3,3,4]

**Output:** -1

**Explanation:**

In this case, it is not possible to rotate the dominoes to make one row of values equal.

"Your friend is typing his *name* into a keyboard. Sometimes, when typing a character *c*, the key might get *long pressed*, and the character will be typed 1 or more times. You examine the *typed* characters of the keyboard. Return *True* if it is possible that it was your friends name, with some characters (possibly none) being long pressed."

### **Example 1:**

**Input:** name = "alex", typed = "aaleex"

**Output:** true

**Explanation:** 'a' and 'e' in 'alex' were long pressed.

### **Example 2:**

**Input:** name = "saeed", typed = "ssaaedd"

**Output:** false

**Explanation:** 'e' must have been pressed twice, but it was not in the typed output.

### **Constraints:**

- $1 \leq \text{name.length}, \text{typed.length} \leq 1000$
- name and typed consist of only lowercase English letters.

```
def isLongPressedName(self, name, typed):  
    i = 0  
    for j in range(len(typed)):  
        if i < len(name) and name[i] == typed[j]:  
            i += 1  
        elif j == 0 or typed[j] != typed[j - 1]:  
            return False  
    return i == len(name)
```

```
from itertools import groupby, zip_longest
```

```
class Solution:
```

```
    """
```

```
    Time: O(max(n, m))
```

```
    Memory: O(1)
```

```
    """
```

```
    def isLongPressedName(self, name: str, typed: str) -> bool:  
        for (a, a_gr), (b, b_gr) in zip_longest(groupby(name),  
groupby(typed), fillvalue=(None, None)):  
            if a != b or sum(1 for _ in a_gr) > sum(1 for _ in b_gr):
```

```

    return False
return True

```

class Solution:

"""

Time: O(max(n, m))

Memory: O(1)

"""

```

def isLongPressedName(self, name: str, typed: str) -> bool:
    return all(
        a == b and sum(1 for _ in a_gr) <= sum(1 for _ in b_gr)
        for (a, a_gr), (b, b_gr) in zip_longest(groupby(name),
groupby(typed), fillvalue=(None, None)))

```

"Given a string S and a string T, find the minimum window in S which will contain all the characters in T in complexity O(n)."

## Approach 1: Sliding Window

### Intuition

The question asks us to return the minimum window from the string SS $\boxed{S}$  which has all the characters of the string TT $\boxed{T}$ . Let us call a window **desirable** if it has all the characters from TT $\boxed{T}$ .

We can use a simple sliding window approach to solve this problem.

In any sliding window based problem we have two pointers.

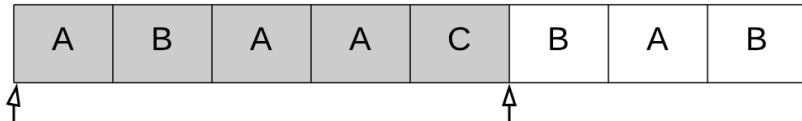
One right $\boxed{right}$  pointer whose job is to expand the current window and then we have the left $\boxed{left}$  pointer whose job is to contract a given window. At any point in time only one of these pointers move and the other one remains fixed.

The solution is pretty intuitive. We keep expanding the window by moving the right pointer. When the window has all the desired characters, we contract (if possible) and save the smallest window till now.

The answer is the smallest desirable window.

For eg.  $S = "ABAACBAB"$   $T = "ABC"$ . Then our answer window is  $"ACB"$  and shown below is one of the possible desirable windows.

String s = "ABAACBABA"

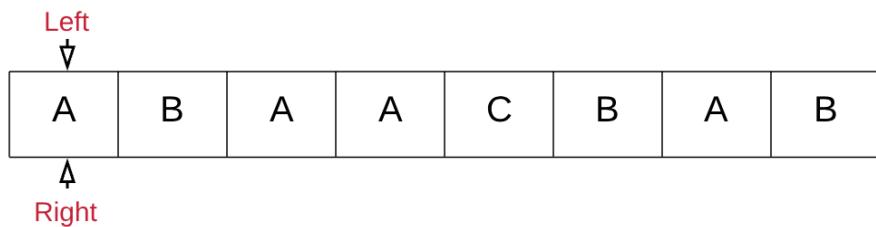


Desirable Window- Has all the characters from t.

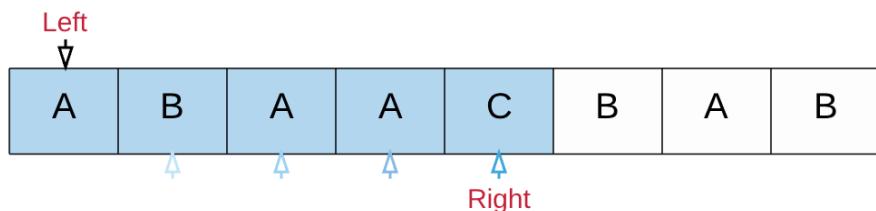
## Algorithm

1. We start with two pointers, `left` initially pointing to the first element of the string `S`.
2. We use the `right` pointer to expand the window until we get a desirable window i.e. a window that contains all of the characters of `T`.
3. Once we have a window with all the characters, we can move the left pointer ahead one by one. If the window is still a desirable one we keep on updating the minimum window size.
4. If the window is not desirable any more, we repeat step 2; `step2` onwards.

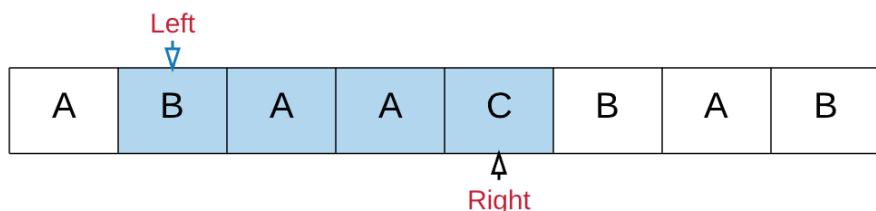
1. Initial State: Left and Right pointers are at index 0.



2. Moving the right pointer until the window has all the elements from string T.  
Record this desirable window.

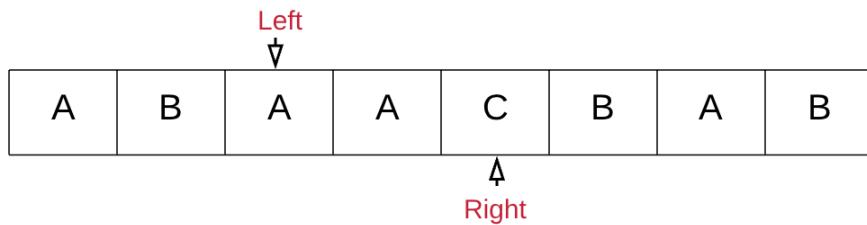


3. Now move the left pointer. Notice the window is still desirable and smaller than the previous window.

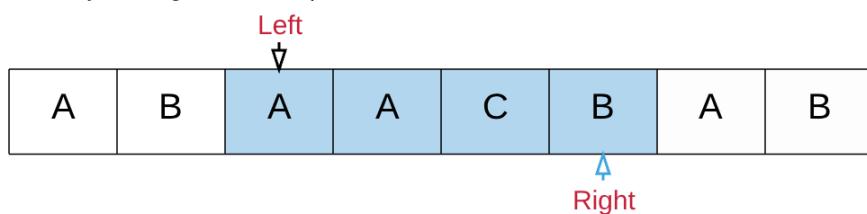


The above steps are repeated until we have looked at all the windows. The smallest window is returned.

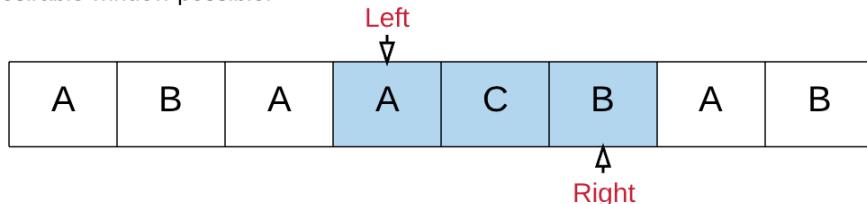
4. After moving left pointer again, the window is no more desirable. Hence we need to increment the right pointer and look for another desirable window.



5. As shown below we again have a desirable window. Now we can contract the window by moving ahead Left pointer and see if the window is still desirable.



6. The window shown below is the smallest desirable window for this example. We continue these steps on the remaining array to see if there is even smaller desirable window possible.



```
def minWindow(self, s, t):
```

```
    """

```

```
    :type s: str
    :type t: str
    :rtype: str
    """

```

```
if not t or not s:
```

```
    return ""
```

```
# Dictionary which keeps a count of all the unique characters in t.
dict_t = Counter(t)
```

```
# Number of unique characters in t, which need to be present in the desired window.
```

```
required = len(dict_t)
```

```
# left and right pointer
```

```
l, r = 0, 0
```

```
# formed is used to keep track of how many unique characters in t are present in the current window in its desired frequency.
```

```
# e.g. if t is "AABC" then the window must have two A's, one B and one C. Thus formed would be = 3 when all these conditions are met.
```

```
formed = 0
```

```
# Dictionary which keeps a count of all the unique characters in the current window.
```

```
window_counts = {}
```

```
# ans tuple of the form (window length, left, right)
```

```
ans = float("inf"), None, None
```

```
while r < len(s):
```

```
# Add one character from the right to the window
```

```
character = s[r]
```

```
window_counts[character] = window_counts.get(character, 0) + 1
```

```
# If the frequency of the current character added equals to the desired count in t then increment the formed count by 1.
```

```
if character in dict_t and window_counts[character] == dict_t[character]:
```

```
    formed += 1
```

```
# Try and contract the window till the point where it ceases to be  
'desirable'.
```

```
while l <= r and formed == required:
```

```
    character = s[l]
```

```
    # Save the smallest window until now.
```

```
    if r - l + 1 < ans[0]:
```

```
        ans = (r - l + 1, l, r)
```

```
    # The character at the position pointed by the `left` pointer is no longer  
    a part of the window.
```

```
    window_counts[character] -= 1
```

```
    if character in dict_t and window_counts[character] < dict_t[character]:
```

```
        formed -= 1
```

```
    # Move the left pointer ahead, this would help to look for a new  
    window.
```

```
    l += 1
```

```
# Keep expanding the window once we are done contracting.
```

```
r += 1
```

```
return "" if ans[0] == float("inf") else s[ans[1] : ans[2] + 1]
```

## Complexity Analysis

- Time Complexity:  $O(|S|+|T|)O(|S| + |T|)O(|S|+|T|)$  where  $|S|$  and  $|T|$  represent the lengths of strings  $S$  and  $T$ . In the worst case we might end up visiting every element of string  $S$  twice, once by left pointer and once by right pointer.  $|T||T||T|$  represents the length of string  $T$ .
- Space Complexity:  $O(|S|+|T|)O(|S| + |T|)O(|S|+|T|)$ .  $|S||S||S|$  when the window size is equal to the entire string  $S$ .  $|T||T||T|$  when  $T$  has all unique characters.

## Approach 2: Optimized Sliding Window

### Intuition

A small improvement to the above approach can reduce the time complexity of the algorithm to  $O(2 * |\text{filtered\_S}| + |S| + |T|)$ , where  $\text{filtered\_S}$  is the string formed from  $S$  by removing all the elements not present in  $T$ .

This complexity reduction is evident when  $|\text{filtered\_S}| \ll |S|$ .

This kind of scenario might happen when length of string  $T$  is way too small than the length of string  $S$  and string  $S$  consists of numerous characters which are not present in  $T$ .

### Algorithm

We create a list called  $\text{filtered\_S}$  which has all the characters from string  $S$  along with their indices in  $S$ , but these characters should be present in  $T$ .

```
S = "ABCDDDDDEEAFFBC" T = "ABC"  
filtered_S = [(0, 'A'), (1, 'B'), (2, 'C'), (11, 'A'), (14, 'B'), (15, 'C')]  
Here (0, 'A') means in string S character A is at index 0.
```

We can now follow our sliding window approach on the smaller string  $\text{filtered\_S}$ .

```
def minWindow(self, s, t):
```

```
    """
```

```
    :type s: str
```

```
    :type t: str
```

```
    :rtype: str
```

```
    """
```

```
    if not t or not s:
```

```
        return ""
```

```
dict_t = Counter(t)
```

```
required = len(dict_t)
```

```
# Filter all the characters from s into a new list along with their index.
```

```
# The filtering criteria is that the character should be present in t.
```

```
filtered_s = []
```

```
for i, char in enumerate(s):
```

```
    if char in dict_t:
```

```
        filtered_s.append((i, char))
```

```
l, r = 0, 0
```

```
formed = 0
```

```
window_counts = {}
```

```
ans = float("inf"), None, None
```

```
# Look for the characters only in the filtered list instead of entire s. This helps to reduce our search.
```

```
# Hence, we follow the sliding window approach on as small list.
```

```
while r < len(filtered_s):
```

```
    character = filtered_s[r][1]
```

```
    window_counts[character] = window_counts.get(character, 0) + 1
```

```
    if window_counts[character] == dict_t[character]:
```

```
        formed += 1
```

```
# If the current window has all the characters in desired frequencies i.e. t is present in the window
```

```
while l <= r and formed == required:
```

```
    character = filtered_s[l][1]
```

```
    # Save the smallest window until now.
```

```
    end = filtered_s[r][0]
```

```
    start = filtered_s[l][0]
```

```
    if end - start + 1 < ans[0]:
```

```
        ans = (end - start + 1, start, end)
```

```
    window_counts[character] -= 1
```

```
    if window_counts[character] < dict_t[character]:
```

```
        formed -= 1
```

```
    l += 1
```

```
r += 1
```

```
return "" if ans[0] == float("inf") else s[ans[1] : ans[2] + 1]
```

"Given a list of query words, return the number of words that are stretchy."

Note: see link for more details.

Sometimes people repeat letters to represent extra feeling. For example:

- "hello" -> "heeelooo"
- "hi" -> "hiiii"

In these strings like "heeelooo", we have groups of adjacent letters that are all the same: "h", "eee", "ll", "ooo".

You are given a string  $s$  and an array of query strings  $\text{words}$ . A query word is **stretchy** if it can be made to be equal to  $s$  by any number of applications of the following extension operation: choose a group consisting of characters  $c$ , and add some number of characters  $c$  to the group so that the size of the group is **three or more**.

- For example, starting with "hello", we could do an extension on the group "o" to get "hellooo", but we cannot get "heloo" since the group "oo" has a size less than three. Also, we could do another extension

like "ll" -> "lllll" to get "helllllooo". If  $s = \text{"helllllooo"}$ , then the query word "hello" would be **stretchy** because of these two extension operations: query = "hello" -> "hellooo" -> "helllllooo" = s.

Return *the number of query strings that are stretchy*.

### Example 1:

**Input:**  $s = \text{"heeeellooo"}$ , words = ["hello", "hi", "helo"]

**Output:** 1

#### Explanation:

We can extend "e" and "o" in the word "hello" to get "heeeellooo".

We can't extend "helo" to get "heeeellooo" because the group "ll" is not size 3 or more.

### Example 2:

**Input:**  $s = \text{"zzzzzzyyyyyy"}$ , words = ["zzyy", "zy", "zyy"]

**Output:** 3

### Constraints:

- $1 \leq s.length, \text{words.length} \leq 100$
- $1 \leq \text{words}[i].length \leq 100$
- $s$  and  $\text{words}[i]$  consist of lowercase letters.
- Here's the plan:
- For the string  $s$  and for each string  $\text{word}$  in  $\text{words}$ , we use `groupby` to construct two tuples per string. The first lists the distinct characters in the string, and second lists the multiplicity of each character in the string.
- I think, knowing that, one can figure out the code below. 'The secret of being a bore is to tell everything.' --Voltaire

#### class Solution:

• def expressiveWords(self, s: str, words: List[str]) -> int:

•

•     f = lambda x: (tuple(zip(\*[(k, len(tuple(g)))  
                      for k,g in groupby(x)])))

•     (sCh, sCt), ans = f(s), 0

•

```

•   for word in words:
•       wCh, wCt = f(word)
•
•       if sCh == wCh:
•           for sc, wc in zip(sCt, wCt):
•
•               if sc < wc or (sc < 3 and sc != wc): break
•
•           else: ans+= 1
•
•       return ans

```

```

• class Solution:
•     def expressiveWords(self, s: str, words: List[str]) -> int:
•         # edge cases
•         if len(s) == 0 and len(words) != 0:
•             return False
•         if len(words) == 0 and len(s) != 0:
•             return False
•         if len(s) == 0 and len(words) == 0:
•             return True
•
•         # helper function, compressing string and extract counts
•         def compressor(s_word):
•             init_string =[s_word[0]]
•             array = []
•             start = 0
•             for i,c in enumerate(s_word):
•                 if c == init_string[-1]:
•                     continue
•                 array.append(i-start)
•                 start = i
•                 init_string += c
•             array.append(i-start+1)
•             return init_string,array
•
•         res = len(words)
•         s_split, s_array = compressor(s)
•         for word in words:
•             word_split = []
•             word_array = []

```

```

•     word_split,word_array = compressor(word)
•     if s_split == word_split:
•         for num_s,num_word in zip(s_array,word_array):
•             if num_s != num_word and num_s < 3 or num_word >
•                 num_s:
•                     res -= 1
•                     break
•                 else:
•                     res -= 1
•             return res

```

- **Python:**

```

•     def expressiveWords(self, S, words):
•         return sum(self.check(S, W) for W in words)
•
•     def check(self, S, W):
•         i, j, n, m = 0, 0, len(S), len(W)
•         for i in range(n):
•             if j < m and S[i] == W[j]: j += 1
•             elif S[i - 1:i + 2] != S[i] * 3 != S[i - 2:i + 1]: return False
•         return j == m

```

- Another approach use 4 pointers, but will be much easier to understand and debug.

- Consider "heeeellooo" and "heloo".

We can create pairs of <letter,repeated\_count> of the two words.

Then we will get their unique characters and the repeated count in-order.

Then we can compare the two list of pairs to check if they are stretchy or not.

"heeeellooo"

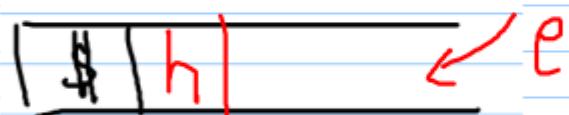
↓      ↓      ↓      ↓  
h(1) e(3) l(2) o(3)

STACK

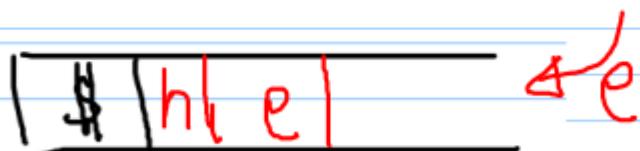


$h \neq \$ \therefore$  insert h

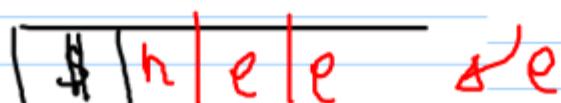
$(\$ , 1) \rightarrow$  Always created



$h \neq e \therefore$  insert  
 $\text{create}(h, 1)$



$e == e \therefore$  insert & inc count



$e == e \therefore$  insert

h|e|l|e|l|e & l  
l != e  
 $\therefore \text{create}(e, 3)$   
reset count = 1  
&& insert l

h|e|e|l|l|e & l  
l == l  $\therefore \text{count}++$   
:  
[('\$', 1), ('h', 1), ('e', 3), ('l', 2), ('o', 3)]

Finally ↗

- We actually don't need the stack but it's easy to see what's happening.
- Next we compare the list of pairs and see if the word is the stretchy version or not.  
First the lengths of the lists to match && the letters must be in the same order.  
If their repeated\_length doesn't match, see if the letter of the word under consideration has smaller length. If so, the letter of the original word should have a length of 3 or more.

Else its not a stretchy word.

[('\$', 1), ('h', 1), ('e', 3), ('l', 2), ('o', 3)]

[('\$', 1), ('h', 1), ('e', 1), ('l', 1), ('o', 1)]

e,3 vs e,1 | l,2 vs l,1  
e == e & 3 > 3 | l == l but 2 < 1  
∴ valid | ∴ invalid

- Putting this to code:

- class Solution:

- def expressiveWords(self, s: str, words: List[str]) -> int:

```
•
•     #generate the pairs <letter,repeated_len> for every string
•     def pairs(s):
•         original = []
•         prev = '$'
•         s+='$' #to make sure the last character of string is not left out.
•
•         count = 1
•         for i in s:
•             if prev == i:
•                 count+=1
•             else:
•                 original.append((prev,count))
•                 prev = i
•                 count=1
•         return original
•
•     original = pairs(s)
•
•     count = 0
•     for w in words:
•         #generate the pair for every word w
•         woriginal = pairs(w)
```

```

•
•     #if the dont have same length, then not stretchy
•     if len(original)!=len(woriginal): continue
•
•         o,wo = original,woriginal
•         match = True
•         for i in range(len(original)):
•
•             #the unique letters present should match in-order
•             if o[i][0]==wo[i][0]:
•                 #if the counts of both the letters are same, no worries
•                 if o[i][1]==wo[i][1]:
•                     continue
•                 else:
•                     #check if the word has smaller length and the original has
•                     3+ len so that we can convert
•                     if wo[i][1]<o[i][1] and o[i][1]>=3:
•                         continue
•                     else:
•                         match = False
•                     break
•                 else:
•                     #the letters itself dont match
•                     match = False
•                     break
•
•         #if everything works
•         if match: count+=1
•
•     return count

```

"Given a *matrix* and a *target*, return the number of non-empty submatrices that sum to target."

A submatrix  $x_1, y_1, x_2, y_2$  is the set of all cells  $\text{matrix}[x][y]$  with  $x_1 \leq x \leq x_2$  and  $y_1 \leq y \leq y_2$ .

Two submatrices  $(x_1, y_1, x_2, y_2)$  and  $(x'_1, y'_1, x'_2, y'_2)$  are different if they have some coordinate that is different: for example, if  $x_1 \neq x'_1$ .

**Example 1:**

0	1	0
1	1	1
0	1	0

**Input:** matrix = [[0,1,0],[1,1,1],[0,1,0]], target = 0

**Output:** 4

**Explanation:** The four 1x1 submatrices that only contain 0.

### Example 2:

**Input:** matrix = [[1,-1],[-1,1]], target = 0

**Output:** 5

**Explanation:** The two 1x2 submatrices, plus the two 2x1 submatrices, plus the 2x2 submatrix.

### Example 3:

**Input:** matrix = [[904]], target = 0

**Output:** 0

### Constraints:

- $1 \leq \text{matrix.length} \leq 100$
- $1 \leq \text{matrix[0].length} \leq 100$
- $-1000 \leq \text{matrix[i]} \leq 1000$
- $-10^8 \leq \text{target} \leq 10^8$

### Intuition

Prequis: [560. Subarray Sum Equals K](#)

Find the Subarray with Target Sum in linear time.

## Explanation

For each row, calculate the prefix sum.

For each pair of columns,

calculate the accumulated sum of rows.

Now this problem is same to, "Find the Subarray with Target Sum".

## Complexity

Time  $O(mnn)$

Space  $O(m)$

```
def numSubmatrixSumTarget(self, A, target):
    m, n = len(A), len(A[0])
    for row in A:
        for i in xrange(n - 1):
            row[i + 1] += row[i]
    res = 0
    for i in xrange(n):
        for j in xrange(i, n):
            c = collections.defaultdict(int)
            cur, c[0] = 0, 1
            for k in xrange(m):
                cur += A[k][j] - (A[k][i - 1] if i > 0 else 0)
                res += c[cur - target]
                c[cur] += 1
    return res
```

### Idea:

This problem is essentially a **2-dimensional** version of **#560. Subarray Sum Equals K (S.S.E.K)**. By using a **prefix sum** on each row or each column, we can compress this problem down to either  $N^2$  iterations of the  $O(M)$  SSEK, or  $M^2$  iterations of the  $O(N)$  SSEK.

In the SSEK solution, we can find the number of subarrays with the target sum by utilizing a **result map** (**res**) to store the different values found as we iterate through the array while keeping a running sum (**csum**). Just as in the case with a prefix sum array, the sum of a subarray between **i** and **j** is equal to the sum of the subarray from **0** to **j** minus the sum of the subarray from **0** to **i-1**.

Rather than iteratively checking if  $\text{sum}[0,j] - \text{sum}[0,i-1] = T$  for every pair of  $i, j$  values, we can flip it around to  $\text{sum}[0,j] - T = \text{sum}[0,i-1]$  and since every earlier sum value has been stored in  $\text{res}$ , we can simply perform a lookup on  $\text{sum}[0,j] - T$  to see if there are any matches.

When extrapolating this solution to our **2-dimensional** matrix (**M**), we will need to first prefix sum the rows or columns, (which we can do **in-place**) to avoid extra space, as we will not need the original values again). Then we should iterate through **M** again in the opposite order of rows/columns where the prefix sums will allow us to treat a group of columns or rows as if it were a **1-dimensional** array and apply the SSEK algorithm.

### **Implementation:**

Python, oddly, has **much** better performance with the use of a simple **dict** instead of a **defaultdict** for **res** (Thanks, [@slcheungcasado!](#)!)

The best result for the code below is **552ms / 15.0MB** (beats 100% / 85%).

#### class Solution:

```
def numSubmatrixSumTarget(self, M: List[List[int]], T: int) -> int:
    xlen, ylen, ans = len(M[0]), len(M), 0
    for r in M:
        for j in range(1, xlen):
            r[j] += r[j-1]
        for j in range(xlen):
            for k in range(j, xlen):
                res, csum = {0: 1}, 0
                for r in M:
                    csum += r[k] - (r[j-1] if j else 0)
                    if csum - T in res: ans += res[csum-T]
                    res[csum] = res[csum] + 1 if csum in res else 1
    return ans
```

Let us define by  $dp[i, j, k]$  sum of numbers in the rectangle  $i \leq x < j$  and  $0 \leq y < m$ . Why it is enough to evaluate only values on these matrices? Because then we can use **2Sum** problem: any sum of elements in submatrix with coordinates  $a \leq x < b$  and  $c \leq y < d$  can be evaluated as difference between sum of  $a \leq x < b, 0 \leq y < d$  and sum of  $a \leq x < b, 0 \leq y < c$ . So, let us fix  $[a, b]$  and  $[c, d]$ , and say we have sums  $S_1, S_2, \dots, S_m$ . Then we want to find how many differences between these values give us our **target**. The idea is to

calculate cumulative sums and keep counter of values, and then check how many we have (we can not use sliding window, because we can have negative values), see problem [560](#). Subarray Sum Equals K for more details.

So, we have in total two stages of our algorithm:

1. Precompute all sums in rectangles of the type  $i \leq x < j$  and  $0 \leq y < m$ .
2. For each  $\frac{n*(n-1)}{2}$  problems with fixed  $i$  and  $j$ , solve sumproblem in  $O(m)$  time.

## Complexity

Time complexity is  $O(n^2m)$ , we need it for both stages. Space complexity is the same.

## Code

class Solution:

```
def numSubmatrixSumTarget(self, matrix, target):
    m, n = len(matrix), len(matrix[0])
    dp, ans = { }, 0
    for k in range(m):
        t = [0] + list(accumulate(matrix[k]))
        for i, j in combinations(range(n+1), 2):
            dp[i, j, k] = dp.get((i,j,k-1), 0) + t[j] - t[i]

    for i, j in combinations(range(n+1), 2):
        T = Counter([0])
        for k in range(m):
            ans += T[dp[i, j, k] - target]
            T[dp[i, j, k]] += 1

    return ans
```

For each row, calculate the prefix sum. For each pair of columns, calculate the sum of rows.

Now this problem is changed to problem 560 Subarray Sum Equals K.

```
def numSubmatrixSumTarget(self, matrix: List[List[int]], target: int) -> int:
    m, n = len(matrix), len(matrix[0])
    for x in range(m):
        for y in range(n - 1):
```

```

matrix[x][y+1] += matrix[x][y]
res = 0
for y1 in range(n):
    for y2 in range(y1, n):
        preSums = {0: 1}
        s = 0
        for x in range(m):
            s += matrix[x][y2] - (matrix[x][y1-1] if y1 > 0 else 0)
            res += preSums.get(s - target, 0)
            preSums[s] = preSums.get(s, 0) + 1
return res

```

"Given a *rows* x *cols* binary *matrix* filled with 0's and 1's, find the largest rectangle containing only 1's and return its area."

### **Example 1:**

1	0	1	0	0
1	0	1	1	1
1	1	1	1	1
1	0	0	1	0

**Input:** matrix =  
`[[1,"0","1","0","0"],["1","0","1","1","1"],["1","1","1","1","1"],["1","0","0","1","0"]]`

**Output:** 6

**Explanation:** The maximal rectangle is shown in the above picture.

### **Example 2:**

**Input:** matrix = [[0]]

**Output:** 0

### **Example 3:**

**Input:** matrix = [["1"]]

**Output:** 1

**Constraints:**

- rows == matrix.length
- cols == matrix[i].length
- 1 <= row, cols <= 200
- matrix[i][j] is '0' or '1'.

The DP solution proceeds row by row, starting from the first row. Let the maximal rectangle area at row i and column j be computed by [right(i,j) - left(i,j)]\*height(i,j).

All the 3 variables left, right, and height can be determined by the information from previous row, and also information from the current row. So it can be regarded as a DP solution. The transition equations are:

left(i,j) = max(left(i-1,j), cur\_left), cur\_left can be determined from the current row

right(i,j) = min(right(i-1,j), cur\_right), cur\_right can be determined from the current row

height(i,j) = height(i-1,j) + 1, if matrix[i][j]=='1';

height(i,j) = 0, if matrix[i][j]=='0'

The code is as below. The loops can be combined for speed but I separate them for more clarity of the algorithm.

```
class Solution {public:  
int maximalRectangle(vector<vector<char> > &matrix) {  
    if(matrix.empty()) return 0;  
    const int m = matrix.size();  
    const int n = matrix[0].size();  
    int left[n], right[n], height[n];  
    fill_n(left,n,0); fill_n(right,n,n); fill_n(height,n,0);  
    int maxA = 0;  
    for(int i=0; i<m; i++) {
```

```

int cur_left=0, cur_right=n;
for(int j=0; j<n; j++) { // compute height (can do this from either side)
    if(matrix[i][j]=='1') height[j]++;
    else height[j]=0;
}
for(int j=0; j<n; j++) { // compute left (from left to right)
    if(matrix[i][j]=='1') left[j]=max(left[j],cur_left);
    else {left[j]=0; cur_left=j+1;}
}
// compute right (from right to left)
for(int j=n-1; j>=0; j--) {
    if(matrix[i][j]=='1') right[j]=min(right[j],cur_right);
    else {right[j]=n; cur_right=j;}
}
// compute the area of rectangle (can do this from either side)
for(int j=0; j<n; j++)
    maxA = max(maxA,(right[j]-left[j])*height[j]);
}
return maxA;
};


```

};

If you think this algorithm is not easy to understand, you can try this example:

0	0	0	1	0	0	0
0	0	1	1	1	0	0
0	1	1	1	1	1	0

The vector "left" and "right" from row 0 to row 2 are as follows

row 0:

l: 0	0	0	3	0	0	0
r: 7	7	7	4	7	7	7

row 1:

l: 0	0	2	3	2	0	0
r: 7	7	5	4	5	7	7

row 2:

l: 0 1 2 3 2 1 0
r: 7 6 5 4 5 6 7

The vector "left" is computing the left boundary. Take  $(i,j)=(1,3)$  for example. On current row 1, the left boundary is at  $j=2$ . However, because matrix[1][3] is 1, you need to consider the left boundary on previous row as well, which is 3. So the real left boundary at  $(1,3)$  is 3.

"Your car starts at position 0 and speed +1 on an infinite number line. (Your car can go into negative positions.) Your car drives automatically according to a sequence of instructions A (accelerate) and R (reverse)...Now for some target position, say the length of the shortest sequence of instructions to get there."

- When you get an instruction 'A', your car does the following:
  - position  $\pm$ = speed
  - speed \*= 2
- When you get an instruction 'R', your car does the following:
  - If your speed is positive then speed = -1
  - otherwise speed = 1

Your position stays the same.

For example, after commands "AAR", your car goes to positions 0 --> 1 --> 3 --> 3, and your speed goes to 1 --> 2 --> 4 --> -1.

Given a target position target, return *the length of the shortest sequence of instructions to get there*.

### Example 1:

**Input:** target = 3

**Output:** 2

**Explanation:**

The shortest instruction sequence is "AA".

Your position goes from 0 --> 1 --> 3.

### Example 2:

**Input:** target = 6

**Output:** 5

**Explanation:**

The shortest instruction sequence is "AAARA".

Your position goes from 0 --> 1 --> 3 --> 7 --> 7 --> 6.

### Constraints:

- $1 \leq \text{target} \leq 10^4$

Step 3 makes the difference between TLE and a 52ms submission time (89%). Maintaining a priority queue decreases the speed from 52ms to 72ms (80%).

class Solution:

```
def racecar(self, target: int) -> int:
```

```
#1. Initialize double ended queue as 0 moves, 0 position, +1 velocity
```

```
queue = collections.deque([(0, 0, 1)])
```

```
while queue:
```

```
# (moves) moves, (pos) position, (vel) velocity)
```

```
moves, pos, vel = queue.popleft()
```

```
if pos == target:
```

```
    return moves
```

```
#2. Always consider moving the car in the direction it is already going
```

```
queue.append((moves + 1, pos + vel, 2 * vel))
```

```
#3. Only consider changing the direction of the car if one of the  
following conditions is true
```

```
# i. The car is driving away from the target.
```

```
# ii. The car will pass the target in the next move.
```

```
if (pos + vel > target and vel > 0) or (pos + vel < target and vel < 0):
```

```
    queue.append((moves + 1, pos, -vel / abs(vel)))
```

## Example

For the input 5, we can reach with only 7 steps: AARARAA. Because we can step back.

## Explanation

Let's say  $n$  is the length of  $\text{target}$  in binary and we have  $2^{n-1} \leq \text{target} < 2^n$

We have 2 strategies here:

### 1. Go pass our target , stop and turn back

We take  $n$  instructions of  $A$ .

$$1 + 2 + 4 + \dots + 2^{n-1} = 2^n - 1$$

Then we turn back by one  $R$  instruction.

In the end, we get closer by  $n + 1$  instructions.

### 2. Go as far as possible before pass target, stop and turn back

We take  $n - 1$  instruction of  $A$  and one  $R$ .

Then we take  $m$  instructions of  $A$ , where  $m < n$

## Complexity

Time  $O(T \log T)$

Space  $O(T)$

```
dp = {0: 0}
def racecar(self, t):
    if t in self.dp:
        return self.dp[t]
    n = t.bit_length()
    if 2**n - 1 == t:
        self.dp[t] = n
    else:
        self.dp[t] = self.racecar(2**n - 1 - t) + n + 1
        for m in range(n - 1):
            self.dp[t] = min(self.dp[t], self.racecar(t - 2**(n - 1) + 2**m) + n + m
+ 1)
    return self.dp[t]
```

```

from collections import deque

class Solution:
    def racecar(self, target: int) -> int:
        queue = deque([])
        visited = set()
        queue.append((0, 1, 0))
        visited.add((0, 1))
        while queue:
            size = len(queue)
            while size > 0:
                current_position, current_speed, moves = queue.popleft()
                if current_position == target:
                    return moves
                new_position = current_position + current_speed
                new_speed = 2 * current_speed
                if (new_position, new_speed) not in visited and abs(new_position - target) < target:
                    visited.add((new_position, new_speed))
                    queue.append((new_position, new_speed, moves + 1))
                # Explore option : "R"
                new_position = current_position
                new_speed = 1 if current_speed < 0 else -1
                if (new_position, new_speed) not in visited and abs(new_position - target) < target:
                    visited.add((new_position, new_speed))
                    queue.append((new_position, new_speed, moves + 1))
        return -1

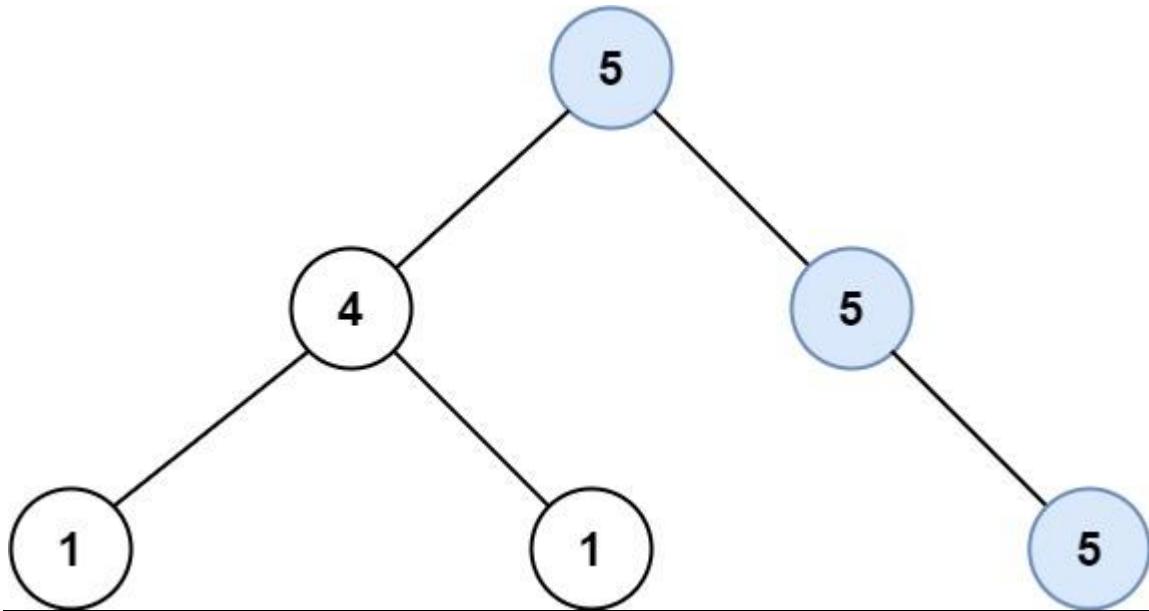
```

"Given a binary tree, find the length of the longest path where each node in the path has the same value. This path may or may not pass through the root. The length of path between two nodes is represented by the number of edges between them."

Given the root of a binary tree, return *the length of the longest path, where each node in the path has the same value*. This path may or may not pass through the root.

**The length of the path** between two nodes is represented by the number of edges between them.

**Example 1:**

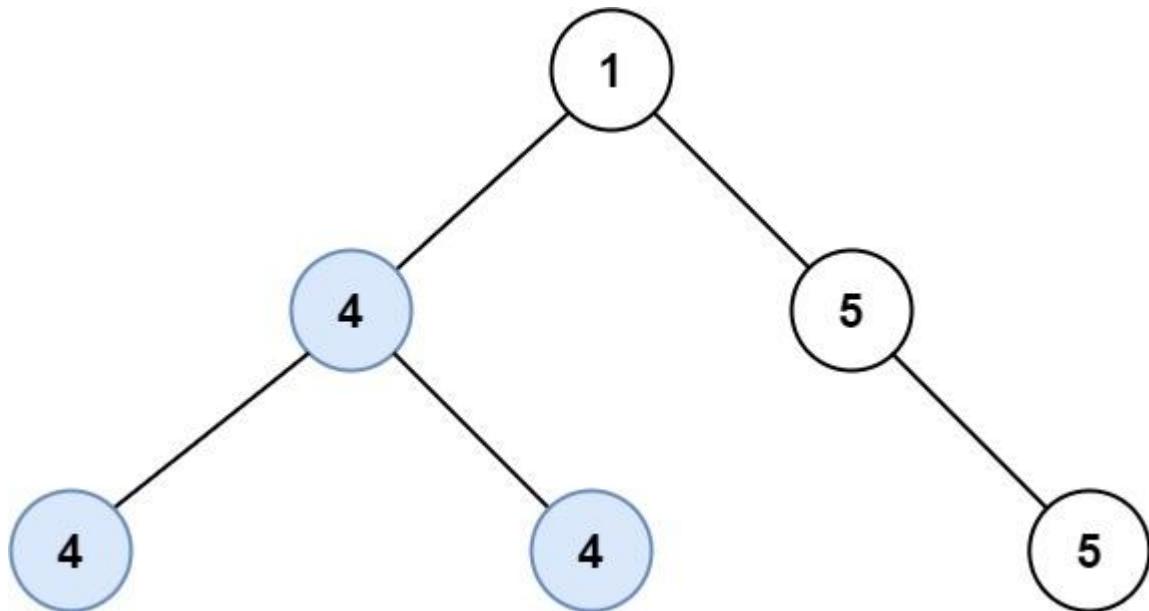


**Input:** root = [5,4,5,1,1,null,5]

**Output:** 2

**Explanation:** The shown image shows that the longest path of the same value (i.e. 5).

**Example 2:**



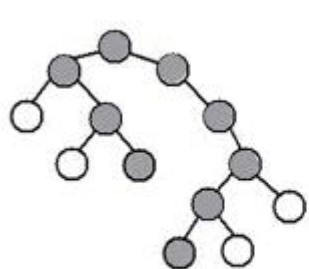
**Input:** root = [1,4,5,4,4,null,5]

**Output:** 2

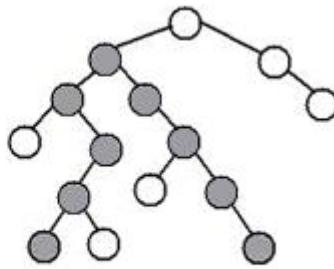
**Explanation:** The shown image shows that the longest path of the same value (i.e. 4).

## Constraints:

- The number of nodes in the tree is in the range  $[0, 10^4]$ .
  - $-1000 \leq \text{Node.val} \leq 1000$
  - The depth of the tree will not exceed 1000.
  - The approach is similar to the [Diameter of Binary Tree](#) question except that we reset the left/right to 0 whenever the current node does not match the children node value.
  - In the Diameter of Binary Tree question, the path can either go through the root or it doesn't.



*diameter, 9 nodes, through root*



*diameter, 9 nodes, NOT through root*

- Hence at the end of each recursive loop, return the longest length using that node as the root so that the node's parent can potentially use it in its longest path computation.
  - We also use an external variable `longest` that keeps track of the longest path seen so far.

```
• class Solution(object):
•     def longestUnivaluePath(self, root):
•         """
•             :type root: TreeNode
•             :rtype: int
•         """
•         # Time: O(n)
•         # Space: O(n)
•         longest = [0]
•
•         def traverse(node):
•             if not node:
•                 return 0
•             left_len, right_len = traverse(node.left), traverse(node.right)
•             left = (left_len + 1) if node.left and node.left.val == node.val else
• 0
•             right = (right_len + 1) if node.right and node.right.val == node.val
•             else 0
•             longest[0] = max(left, right, longest[0])
•             return max(left, right)
```

- `longest[0] = max(longest[0], left + right)`
- `return max(left, right)`
- `traverse(root)`
- `return longest[0]`

Python] Simple DFS | Just take care of Different Conditions for combining results from sub-Tree

```
class Solution:
    def longestUnivalPath(self, root: Optional[TreeNode]) -> int:
        if not root: return 0
        self.ans = 0
        # dfs() returns the longest path starting from the current node
        def dfs(node):
            temp1, temp2 = 0, 0
            if node.left: temp1 = dfs(node.left)
            if node.right: temp2 = dfs(node.right)
            # Taking care of all the conditions if we need to combine results from
            left sub-tree or right sub-tree or from neither
            if node.left and node.right and node.val == node.left.val and node.val ==
node.right.val:
                self.ans = max(self.ans, 2 + temp1 + temp2)
            return 1 + max(temp1, temp2)
            elif node.left and node.left.val == node.val:
                self.ans = max(self.ans, 1 + temp1)
            return 1 + temp1
            elif node.right and node.right.val == node.val:
                self.ans = max(self.ans, 1 + temp2)
            return 1 + temp2
            else:
                return 0

        dfs(root)
        return self.ans
```

This question is definitely not `easy` question. Its `medium` at best.

We will do post-order traversal to compute paths so that at each parent node there will be information about the paths from left and right child.

We will maintain following variables:

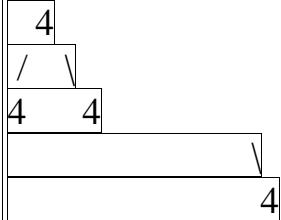
global: `max_val` which will always have the max value.

local: `cur_val` which will compute the value at the node.

local: `ret_val` which will compute the value to be returned to parent.

To compute the solution of this question we have to consider 4 possible conditions at each level.

1. if `left_child.val == right_child.val == root.val`

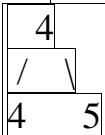


In this case,

`cur_val = lval + rval + 2` (+ 2 because two paths are now added from right and left)

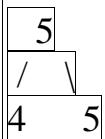
`ret_val = max(lval, rval) + 1` (because we can only send one path to the parent. To get the longest path we will have to send the `max(lval, rval)`)

2. if `only left_child.val == root.val`



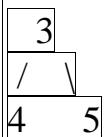
`cur_val = ret_val = lval + 1`

3. If `only right_child.val == root.val`



`cur_val = ret_val = rval + 1`

4. If `both right_child and left_child don't match with root.val`



`cur_val = ret_val = 0` Since there no path between the child and parent node.

`class Solution:`

```

def longestUnivalPath(self, root: TreeNode) -> int:
    if not root: return 0
    self.res = 0
    self.postOrder(root)
    return self.res
  
```

```

def postOrder(self, node):
    if not node: return 0
    l_val = r_val = b_val = 0
    l_val = self.postOrder(node.left)
    r_val = self.postOrder(node.right)

    if node.left and node.right and (node.val == node.left.val ==
node.right.val):
        self.res = max(self.res, l_val + r_val + 2)
        return (max(l_val, r_val) + 1)

    elif node.left and (node.val == node.left.val):
        l_val += 1
        self.res = max(self.res, l_val)
        return l_val

    elif node.right and (node.val == node.right.val):
        r_val += 1
        self.res = max(self.res, r_val)
        return r_val

    else:
        return 0

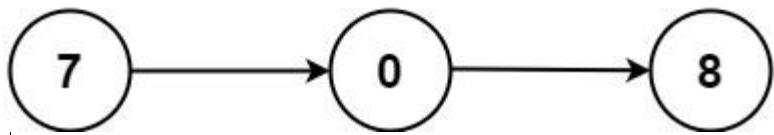
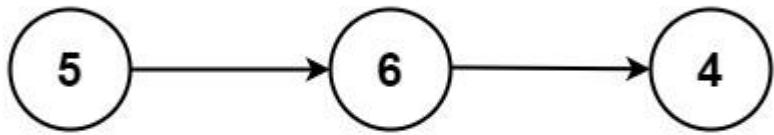
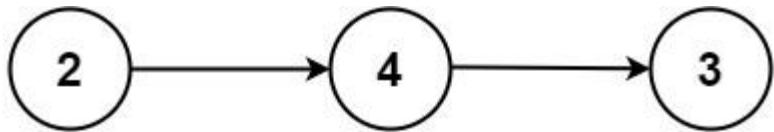
```

"You are given two non-empty linked lists representing two non-negative integers. The digits are stored in reverse order and each of their nodes contain a single digit. Add the two numbers and return it as a linked list."

You are given two **non-empty** linked lists representing two non-negative integers. The digits are stored in **reverse order**, and each of their nodes contains a single digit. Add the two numbers and return the sum as a linked list.

You may assume the two numbers do not contain any leading zero, except the number 0 itself.

**Example 1:**



**Input:** l1 = [2,4,3], l2 = [5,6,4]

**Output:** [7,0,8]

**Explanation:** 342 + 465 = 807.

**Example 2:**

**Input:** l1 = [0], l2 = [0]

**Output:** [0]

**Example 3:**

**Input:** l1 = [9,9,9,9,9,9,9], l2 = [9,9,9,9]

**Output:** [8,9,9,9,0,0,0,1]

**Constraints:**

- The number of nodes in each linked list is in the range [1, 100].
- $0 \leq \text{Node.val} \leq 9$
- It is guaranteed that the list represents a number that does not have leading zeros.
- class ListNode:
- def \_\_init\_\_(self, val=0, next=None):
- self.val = val
- self.next = next
-

```

• class Solution:
•     def addTwoNumbers(self, l1: Optional[ListNode], l2: Optional[ListNode]) -> Optional[ListNode]:
•         num1, num2 = "", ""
•         node1, node2 = l1, l2
•         while node1 is not None:
•             num1 = str(node1.val) + num1
•             node1 = node1.next
•         while node2 is not None:
•             num2 = str(node2.val) + num2
•             node2 = node2.next
•         num1 = int(num1)
•         num2 = int(num2)
•         summ = num1 + num2
•         summ = str(summ)
•
•         digitnodes = list(summ)
•
•         for idx, digit in enumerate(summ):
•             if idx == 0:
•                 digitnodes[idx] = ListNode(val=digit, next=None)
•             else:
•                 digitnodes[idx] = ListNode(val=digit, next=digitnodes[idx-1])
•
•         return(digitnodes[len(digitnodes)-1])
•
• class Solution(object):
•     def addTwoNumbers(self, l1, l2):
•         """
•             :type l1: ListNode
•             :type l2: ListNode
•             :rtype: ListNode
•         """
•         root = ListNode(0)
•         result = root
•         excess = 0
•         while l1 or l2 or excess:
•             if l1:
•                 excess += l1.val
•                 l1 = l1.next
•             if l2:
•

```

- excess += l2.val
- l2 = l2.next
- 
- result.next = ListNode(excess% 10)
- result = result.next
- excess = excess//10
- 
- return root.next

Definition for singly-linked list.

class ListNode:

```
def init(self, val=0, next=None):
```

```
    self.val = val
```

```
    self.next = next
```

class Solution:

```
    def addTwoNumbers(self, l1: Optional[ListNode], l2: Optional[ListNode]) ->
Optional[ListNode]:
```

        head=n=ListNode(0) #initiating our resultant linkedlist (head is taken to call linked list and n is taken to iterate through it to implement operations)

```
        carry=0
```

#continue loop till l1,l2 or carry exists.

```
        while l1 or l2 or carry:
```

            v1=v2=0 # taken for storing values each time in both linkedlists and in case we have extra carry .

```
            if l1: #iterating through first linked list
```

```
                v1=l1.val
```

```
                l1=l1.next
```

```
            if l2: # iterating through second linkedlist
```

```
                v2=l2.val
```

```
                l2=l2.next
```

            carry,value=divmod(v1+v2+carry,10) # it is used to divide and store both quotient and remainder at same time in form of tuple

```
                n.next=ListNode(value) #making nodes of final values
```

```
                n=n.next
```

```
        return head.next
```

""

All edge cases are covered with this approach

```
class Solution:  
    def addTwoNumbers(self, l1: Optional[ListNode], l2: Optional[ListNode]) ->  
Optional[ListNode]:  
        dummy = ListNode()  
        cur = dummy  
  
        carry = 0  
        while l1 or l2 or carry:  
            val1 = l1.val if l1 else 0  
            val2 = l2.val if l2 else 0  
  
            #new digit  
            val = val1 + val2 + carry  
            carry = val // 10  
            val = val % 10  
            cur.next = ListNode(val)  
  
            #update pointers  
            cur = cur.next  
            l1 = l1.next if l1 else None  
            l2 = l2.next if l2 else None  
  
        return dummy.next
```

Probability (19 questions)

**1. Bobo the amoeba has a 25%, 25%, and 50% chance of producing 0, 1, or 2 offspring, respectively. Each of Bobo's descendants also have the same probabilities. What is the probability that Bobo's lineage dies out?**

- $p=1/4+1/4p+1/2p^2 \Rightarrow p=1/2$

**2. In any 15-minute interval, there is a 20% probability that you will see at least one shooting star. What is the probability that you see at least one shooting star in the period of an hour?**

- $1-(0.8)^4$ . Or, we can use Poisson processes

**3. How can you generate a random number between 1 - 7 with only a die?**

- Launch it 3 times: each throw sets the nth bit of the result.

- For each launch, if the value is 1-3, record a 0, else 1. The result is between 0 (000) and 7 (111), evenly spread (3 independent throw). Repeat the throws if 0 was obtained: the process stops on evenly spread values.

**4. How can you get a fair coin toss if someone hands you a coin that is weighted to come up heads more often than tails?**

- Flip twice and if HT then H, TH then T.

**5. You have an 50-50 mixture of two normal distributions with the same standard deviation. How far apart do the means need to be in order for this distribution to be bimodal?**

- more than two standard deviations

**6. Given draws from a normal distribution with known parameters, how can you simulate draws from a uniform distribution?**

- plug in the value to the CDF of the same random variable

**7. A certain couple tells you that they have two children, at least one of which is a girl. What is the probability that they have two girls?**

- 1/3

**8. You have a group of couples that decide to have children until they have their first girl, after which they stop having children. What is the expected gender ratio of the children that are born? What is the expected number of children each couple will have?**

- gender ratio is 1:1. Expected number of children is 2. let X be the number of children until getting a female (happens with prob 1/2). this follows a geometric distribution with probability 1/2

**9. How many ways can you split 12 people into 3 teams of 4?**

- the outcome follows a multinomial distribution with n=12 and k=3. but the classes are indistinguishable

**10. Your hash function assigns each object to a number between 1:10, each with equal probability. With 10 objects, what is the probability of a hash**

**collision? What is the expected number of hash collisions? What is the expected number of hashes that are unused.**

- the probability of a hash collision:  $1-(10!/10^{10})$
- the expected number of hash collisions:  $1-10*(9/10)^{10}$
- the expected number of hashes that are unused:  $10*(9/10)^{10}$

**11. You call 2 UberX's and 3 Lyfts. If the time that each takes to reach you is IID, what is the probability that all the Lyfts arrive first? What is the probability that all the UberX's arrive first?**

- All Lyft's first
  - probability that the first car is Lyft =  $3/5$
  - probability that the second car is Lyft =  $2/4$
  - probability that the third car is Lyft =  $1/3$  Therefore, probability that all the Lyfts arrive first =  $(3/5) * (2/4) * (1/3) = 1/10$
- All Uber's first
  - probability that the first car is Uber =  $2/5$
  - probability that the second car is Uber =  $1/4$  Therefore, probability that all the Ubers arrive first =  $(2/5) * (1/4) = 1/10$

**12. I write a program should print out all the numbers from 1 to 300, but prints out Fizz instead if the number is divisible by 3, Buzz instead if the number is divisible by 5, and FizzBuzz if the number is divisible by 3 and 5. What is the total number of numbers that is either Fizzed, Buzzed, or FizzBuzzed?**

- $100+60-20=140$

**13. On a dating site, users can select 5 out of 24 adjectives to describe themselves. A match is declared between two users if they match on at least 4 adjectives. If Alice and Bob randomly pick adjectives, what is the probability that they form a match?**

- $24C5*(1+5(24-5))/24C5*24C5 = 4/1771$

**14. A lazy high school senior types up application and envelopes to n different colleges, but puts the applications randomly into the envelopes. What is the expected number of applications that went to the right college?**

- 1

**15. Let's say you have a very tall father. On average, what would you expect the height of his son to be? Taller, equal, or shorter? What if you had a very short father?**

- Shorter. Regression to the mean

**16. What's the expected number of coin flips until you get two heads in a row? What's the expected number of coin flips until you get two tails in a row?**

- After the first two flips, you can see this problem as a Markov chain, with states HH, HT, TH, TT.
- HH is the final state. You can then define the expected number of steps  $N$  before reaching HH:  $E(N) = 2 + 0.25n_{HH}, 0.25n_{HT}, 0.25n_{TH}, 0.25n_{TT}$ .  $n_{XX}$  represents the expected number of steps before reaching HH starting from state XX.
- Solve linear equation:
  - $n_{HH} = 0$
  - $n_{HT} = 1 + 0.5n_{TT} + 0.5n_{TH}$
  - $n_{TH} = 1 + 0.5n_{HH} + 0.5n_{HT}$
  - $n_{TT} = 1 + 0.5n_{TH} + 0.5n_{TT}$
- Result gives  $E(N) = 6$ .

**17. Let's say we play a game where I keep flipping a coin until I get heads. If the first time I get heads is on the  $n$ th coin, then I pay you  $2^{n-1}$  dollars. How much would you pay me to play this game?**

- less than \$3

**18. You have two coins, one of which is fair and comes up heads with a probability  $1/2$ , and the other which is biased and comes up heads with probability  $3/4$ . You randomly pick coin and flip it twice, and get heads both times. What is the probability that you picked the fair coin?**

- 4/13

**19. You have a 0.1% chance of picking up a coin with both heads, and a 99.9% chance that you pick up a fair coin. You flip your coin and it comes**

**up heads 10 times. What's the chance that you picked up the fair coin, given the information that you observed?**

- Events:  $F = \text{"picked a fair coin"}$ ,  $T = \text{"10 heads in a row"}$
- (1)  $P(F|T) = P(T|F)P(F)/P(T)$  (Bayes formula)
- (2)  $P(T) = P(T|F)P(F) + P(T|\neg F)P(\neg F)$  (total probabilities formula)
- Injecting (2) in (1):  $P(F|T) = P(T|F)P(F)/(P(T|F)P(F) + P(T|\neg F)P(\neg F)) = 1 / (1 + P(T|\neg F)P(\neg F)/(P(T|F)P(F)))$
- Numerically:  $1/(1 + 0.001 * 2^{10} / 0.999)$ .
- With  $2^{10} \approx 1000$  and  $0.999 \approx 1$  this simplifies to  $1/2$

## 20. What is a P-Value ?

- The probability to obtain a similar or more extreme result than observed when the null hypothesis is assumed.
- ⇒ If the p-value is small, the null hypothesis is unlikely

## 1. What Is the Difference Between Descriptive and Inferential Statistics?

Descriptive and inferential statistics are two different branches of the field. The former summarizes the characteristics and distribution of a dataset, such as mean, median, variance, etc. You can present those using tables and data visualization methods, like box plots and [histograms](#).

In contrast, inferential statistics allows you to formulate and test hypotheses for a sample and generalize the results to a wider population. Using confidence intervals, you can estimate the population parameters.

You must be able to explain the mechanisms behind these concepts, as entry-level statistics questions for a data analyst interview often revolve around sampling, the generalizability of results, etc.

## 2. What Are the Main Measures Used to Describe the Central Tendency of Data?

[Centrality measures](#) are essential for exploratory data analysis. They all indicate the center of the data distribution but yield different results. You must

understand the difference between the main types to interpret and use them in analyses.

During a statistics job interview, you might need to explain the meaning of each measure of centrality – [mean, median, and mode](#):

- **Mean**, also called average, is the sum of all observations divided by the total number of participants or cases (n).
- **Median** is the mid-point in a dataset ordered from the smallest to the largest when n is odd. With an even number of data points, it's the average of the values in position  $n/2$  and  $(n+1)/2$  (i.e., the two values in the middle).
- **Mode** is the most frequently appearing data point. It is a useful measure when working with categorical variables.

### 3. What Are the Main Measures of Variability?

[Variability measures](#) are also crucial in describing data distribution. They show how spread-out data points are and how far away they are from the mean.

Some of the basic questions during a statistics interview might require you to explain the meaning and usage of variability measures. Here's your cheat sheet:

- **Variance** measures the average squared distance of data points from the mean. A small variance corresponds to a narrow spread of the values, while a big variance implies that data points are far from the mean.
- **Standard deviation** is the square root of the variance. It shows the amount of variation of values in a dataset.
- **Range** is the difference between the maximum and minimum data value. It is a good indicator of variability when there are no outliers in a dataset, but when there are, it can be misleading.
- **Interquartile range (IQR)** measures the spread of the middle part of a dataset. It's essentially the difference between the third and the first quartile.

## 4. What Are Skewness and Kurtosis?

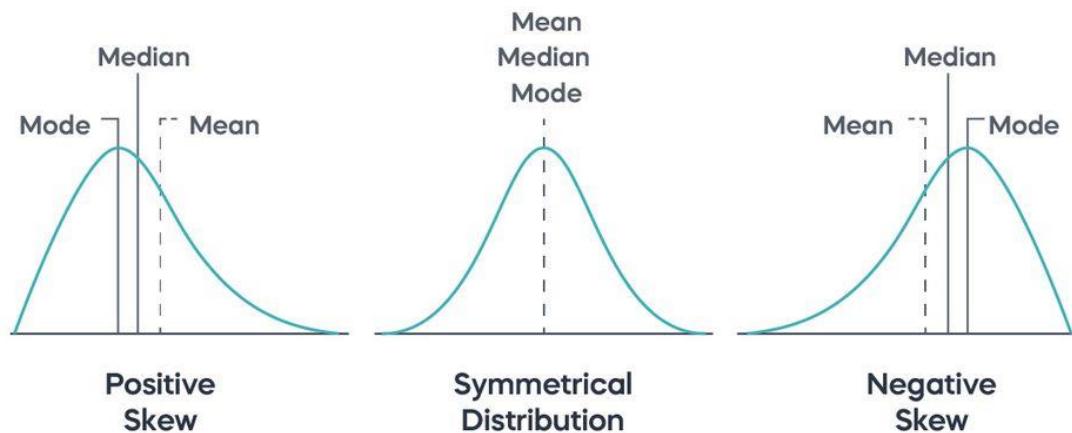
Next on our list of statistics questions for a data science interview are the measures of the shape of data distribution – skewness and kurtosis.

Let's start with the former.

**Skewness** is a great way to measure the symmetry of distribution and the likelihood of a given value falling in the tails. With **symmetrical distribution**, the mean and the median coincide. If the data distribution isn't symmetrical, it is skewed.

There are two types of skewness:

- **Positive** is when the right tail is longer, most values are clustered around the left tail, and the median is smaller than the mean.
- **Negative** is when the left tail is longer, most values are clustered around the right tail, and the median is greater than the mean.



**Kurtosis**, on the other hand, reveals how heavy or light-tailed data is compared to the normal distribution. There are three types of kurtoses:

- **Mesokurtic** distributions approximate a normal distribution.

- **Leptokurtic** distributions have a pointy shape and heavy tails, indicating a high probability of extreme events occurring.
- **Platykurtic** distributions have a flat shape and light tails. They reveal a low probability of the occurrence of extreme events.

For an entry-level job, it may be enough to know the meaning and calculations of these measures. However, statistics interview questions for advanced data science positions may revolve around the usage of these concepts in practice.

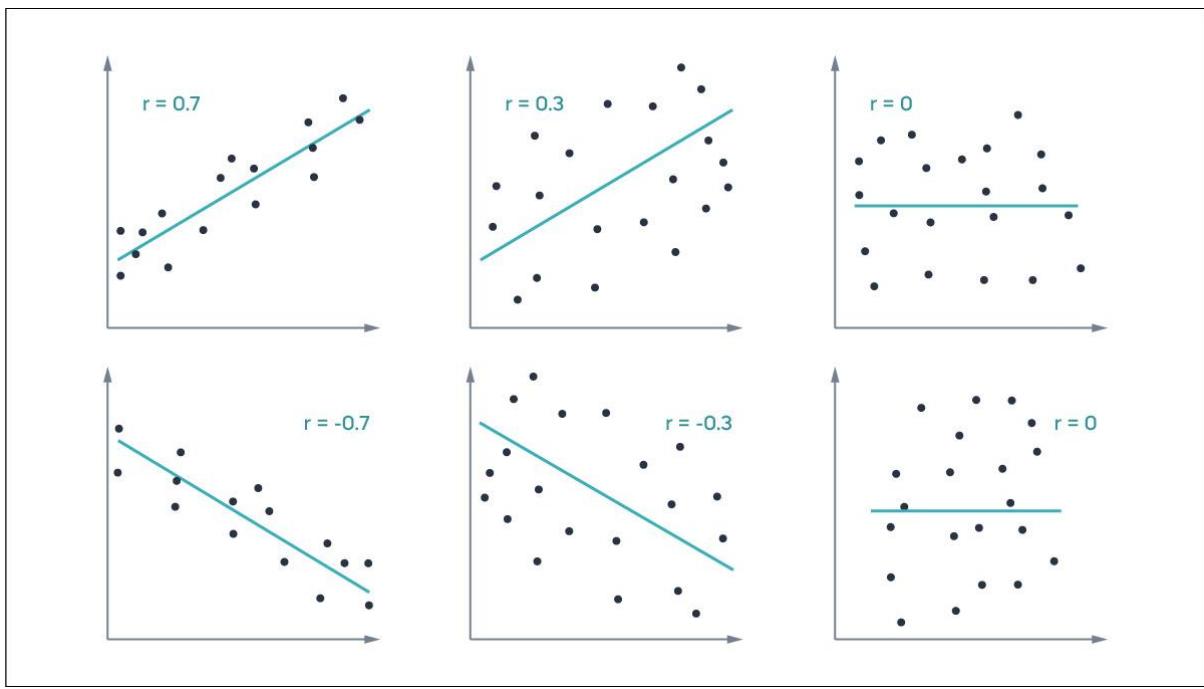
If you want to prepare for more advanced positions, try the [365 Data Scientist Career Track](#). It starts from the basics with Statistics and Probability, builds up your knowledge with programming languages, SQL, Machine Learning, and AI, and ends with portfolio, resume, and interview preparation.

## 5. Describe the Difference Between Correlation and Autocorrelation

These two concepts tend to be confused, which makes it a good trick question for a statistics interview. To avoid surprises, we'll explain the difference.

A **correlation** measures the linear relationship between two or more variables. It ranges between -1 and 1. It's positive if the variables increase or decrease together. If it's negative, one variable decreases while the other increases. When the value is 0, the variables aren't related.

Here's a scatterplot illustrating the different types of correlation:



In contrast, **autocorrelation** measures the linear relationship between two values of the same variable. Typically, we use it when we deal with a time series, i.e., different observations of the same construct. Just like correlation, it can be positive or negative.

## 6. Explain the Difference Between Probability Distribution and Sampling Distribution

As we mentioned, you may be asked various statistics interview questions regarding sampling and the generalizability of results. The difference between probability and sampling distribution is just one example.

A **probability distribution** is a function used to calculate the probability of a random variable  $X$  taking different values. There are two main types depending on the variable – discrete and continuous. Examples of the former are the binomial and Poisson distributions, and of the latter – normal and uniform distributions.

A **sampling distribution** is the probability distribution of a statistic based on a range of random samples from a population. The definition sounds confusing but it's encountered very often in practice.

For example, imagine you're a clinical data analyst working on the development of a new treatment for patients with Alzheimer's. You'll likely be working with samples from the entire population of individuals with the disease. Hence, you'll use the sampling distribution during the data analysis.

## 7. What Is the Normal Distribution?

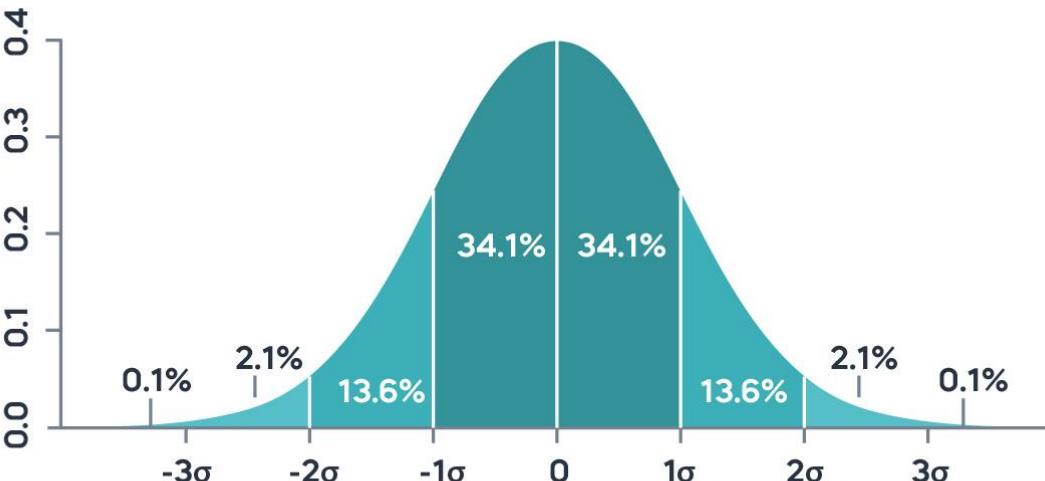
**Normal distribution** is a central concept in mathematics and data analysis. As such, it often appears in statistics interview questions.

The normal, also known as Gaussian, distribution is the most important probability distribution in statistics. It's often called a "bell curve" because of its shape – tall in the middle, flat toward the ends.

A key characteristic of the normal distribution is that the mean and the median coincide. The mean is equal to 0 and the standard deviation is 1. With this information, we can calculate that:

- 27% of the data falls within  $+/-1$  standard deviation of the mean.
- 45% of the data falls within  $+/-2$  standard deviations of the mean.
- 7% of the data falls within  $+/-3$  standard deviations of the mean.

This is known as the empirical rule.



But what is so special about it?

It is considered that naturally occurring phenomena tend to have a normal distribution. As such, we often use it in data analysis to determine the probability of a data point being above or below a given value or for a sample mean being above or below the population mean.

## 8. What Are the Assumptions of Linear Regression?

Next, we move on from basic to intermediate probability and statistics interview questions. To further advance your knowledge on these topics, check out 365's [Statistics course](#) for data scientists.

But for now, let's continue with linear regression, which is at the basis of predictive analysis.

It investigates the relationship between one or more independent variables (predictors) and a dependent variable (outcome). More concretely, it examines whether and to what extent the independent variables are good predictors of the outcome.

The residual (or error term) is equal to the predictor variable minus the actual observed value. Linear regression models aim to find the "line of best fit" where the error is minimal.

The typical statistics interview questions for a data analyst job might involve these definitions or the four main assumptions that must be met to conduct linear regression analysis.

These are the following:

- **Linear relationship:** There is a linear relationship between the predictors and the dependent variable.
- **Normality:** The dependent variable has a normal distribution for any fixed value of the predictor.
- **Homoscedasticity:** The variance of the error term is constant for every value of the independent variable.
- **Independence:** All observations are independent of each other, meaning there is no autocorrelation between the residuals.

## 9. What Is Hypothesis Testing?

We already touched on this topic with some of the previous statistics and probability interview questions. But since it is a fundamental part of data analysis, we cover it in more detail.

Hypothesis testing allows us to make an inference about the population based on data from a sample. Here's how to conduct it:

First, we formulate a **null hypothesis or H<sub>0</sub>**. This is an assumption that there is no difference or no relationship between the variables. For each null hypothesis, there is an alternative one assuming the opposite. If H<sub>0</sub> is rejected, the **alternative hypothesis** is supported.

To determine whether the data supports a particular hypothesis, we need to choose an appropriate statistical test. If the probability of the null hypothesis is below a predetermined significance level, we can reject it.

On that note, statistics questions for a data analyst interview may also be regarding different types of statistical tests. To help you prepare, we cover the basic ones.

## **10. What Are the Most Common Statistical Tests Used?**

There are numerous statistical tests, each one serving a different purpose. Here are some of the most common ones:

- The Shapiro-Wilk test is a statistical tool testing if a data distribution is normal.
- A t-test is used to assess whether the difference between two groups is statistically significant.
- Analysis of Variance (ANOVA) tests the statistical difference between more than two variables.

## **11. What Is the p-Value and How to Interpret It?**

A p-value is the probability of obtaining given results if the null hypothesis is correct. To reject it, the p-value must be lower than a predetermined significance level  $\alpha$ .

The most commonly used significance level is 0.05. This means that if the p-value is below 0.05, we can reject the null hypothesis and accept the alternative one.

In that case, we say that the results are statistically significant.

This is a fundamental part of data analysis, hence a common statistics interview question.

## **12. What Is the Confidence Interval?**

The confidence interval is the range within which we expect the results to lie if we repeat the experiment. It is the mean of the result plus and minus the expected variation.

The latter is determined by the standard error of the estimate, while the center of the interval coincides with the mean of the estimate. The most common confidence interval is 95%.

### 13. What Are the Main Ideas of the Law of Large Numbers?

The Law of Large Numbers is a key theorem in probability and statistics with many practical applications in finance, business, etc. It states that if an experiment is repeated independently multiple times, the mean of all results will approximate the expected value.

A classic example is coin flipping. We know that the probability ( $P$ ) of getting tails is 50%. If the number of tails after 100 trials is  $X$ , then the expected value  $E(X) = n \times P(X) = 100 \times 0.5 = 50$ .

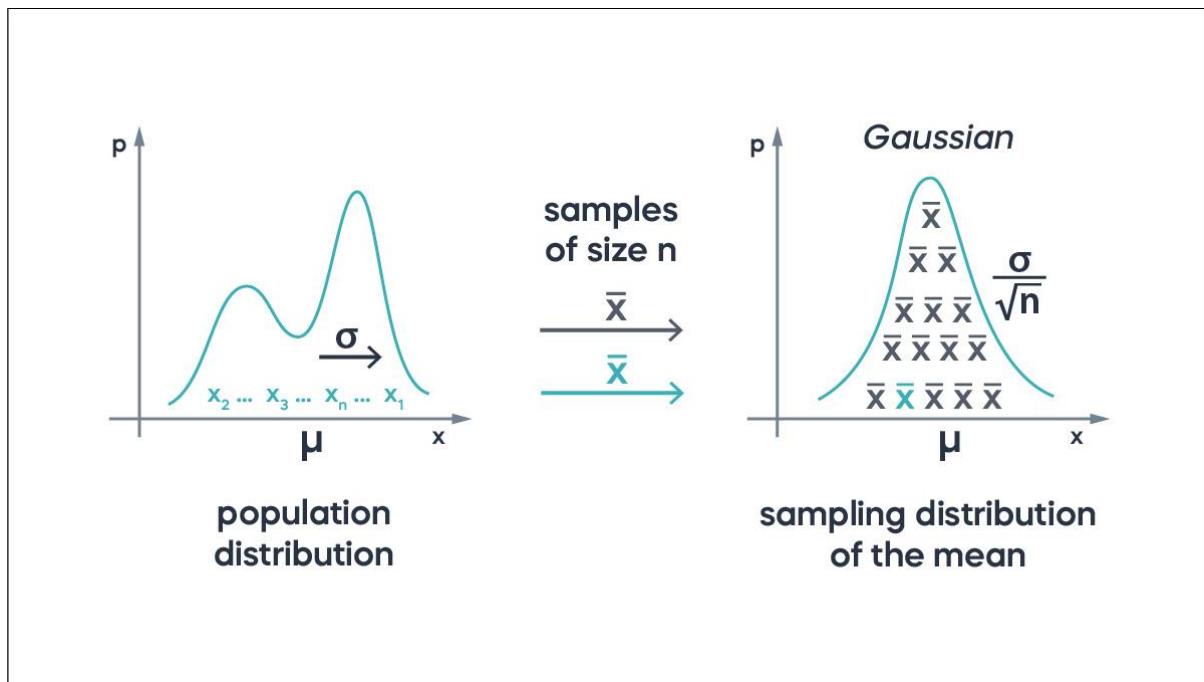
Let's suppose we repeat the experiment multiple times.

The first time, we get  $X_1 = 65$  tails, the second,  $X_2 = 50$  tails, and so on. In the end, we calculate the mean of all trials by adding up the random variables ( $X_1, X_2, \dots, X_n$ ) and dividing the sum by the number of experiments. Following the Law of Large Numbers, the mean of these results will approximate the expected value  $E(X) = 50$ .

This is a basic theorem in Statistics with applications in machine learning, so you can expect questions about it during a job interview.

### 14. What Is the Central Limit Theorem?

The [Central Limit Theorem](#) states that the distribution of sample means starts to resemble a normal distribution as the size of the sample increases. Interestingly, this happens even when the underlying population doesn't have a Gaussian distribution. This is illustrated in the figure below.



On the right, we see that, regardless of the population distribution, the sample means have a symmetrical bell shape distribution as the sample size increases. A sample size equal to or greater than 30 is usually considered large enough for the Central Limit Theorem to apply.

## 15. Explain the Difference Between Probability and Likelihood

Last but not least, we cover one of the fundamental principles of Bayesian statistics, as data science interview questions may include that subject too.

The difference between probability and likelihood is subtle but key. **Probability** is the chance of a particular outcome to occur given the obtained values. When calculating it, we assume the parameters are trustworthy.

In contrast, **likelihood** aims to verify if the parameters in a model are trustworthy given the obtained results. In other words, we calculate the likelihood of a model being correct with the observed measurements.

*Q1:*

**What is a *Probability Distribution*?**

Answer

**A Probability Distribution** is a statistical function that describes all the possible values and likelihood that a random variable can take within a given range.

There are two main types of probability distribution:

- **Discrete probability distributions:** used for random variables with discrete outcomes, for example, the number of heads in five consecutive coin tosses, the number of rainy days in a given week, the number of goals scored by a player, and so on.
- **Continuous probability distributions:** used for random variables with continuous outcomes, for example, the height of male students, median house prices in San Francisco, claim amounts experienced by an insurance company, and so on.

*Q2:*

**Presidential Election Challenge: What is the probability that A or B wins the election?**

Problem

In a presidential election, there are four candidates. Call them A, B, C, and D. Based on our polling analysis, we estimate that A has a **20%** chance of winning the election, while B has a **40%** chance of winning. What is the probability that A or B wins the election?

Answer

We first notice the events that A wins, B wins, C wins, and D wins are mutually exclusive events since more than one of them cannot occur at the same time. For example, if A wins, then B cannot win. We know that the probability of the union of two disjoint events is the summation of individual probabilities.

Therefore,

$$\begin{aligned}(A \text{ wins or } B \text{ wins}) &= (\{A \text{ wins}\} \cup \{B \text{ wins}\}) P(A \text{ wins or } B \text{ wins}) = P(\{A \text{ wins}\} \cup \{B \text{ wins}\}) \\ &= (\{A \text{ wins}\}) + (\{B \text{ wins}\}) P(A \text{ wins or } B \text{ wins}) = P(\{A \text{ wins}\}) + P(\{B \text{ wins}\}) \\ &= 0.2 + 0.4 = 0.6 P(A \text{ wins or } B \text{ wins}) = 0.2 + 0.4 = 0.6\end{aligned}$$

*Q3:*

**Rolling a fair die event challenge**

Problem

Let's suppose I roll a fair die. Let **A** be the event that an outcome is an odd number and let **B** be the event that the outcome is less than or equal to 3. What is the probability  $P(A|B)$ ?

Answer

Given  $A = \{1,3,5\}$  and  $B = \{1,2,3\}$ , if we know **B** has occurred, the outcome must be among  $\{1,2,3\}$ . For **A** to also happen the outcome must be in  $A \cap B = \{1,3\}$ . Since all die rolls are equally likely, then we calculate the conditional probability as:

$$P(A|B) = |B|/|A \cap B| = 2/3$$

**Q4:**

**What is the difference between a *Combination* and a *Permutation*?**

Answer

- A **Combination** is the choice of  $r$  elements from a set of  $n$  elements *without replacement* and where *order does not matter*. Is most used to group data. For example, picking three team members from a group, picking two colors from a color brochure, etc. It is mathematically defined as:

$$Crn = \frac{n!}{r!(n-r)!}$$

- A **Permutation** is the choice of  $r$  elements from a set of  $n$  elements *without replacement* and where the *order matters*. Is used to list data, for example picking first, second and third place winners, picking two favorite colors -in order- from a color brochure, etc. It is mathematically defined as:

$$Pn,r = r!(n-r)!$$

**Q5:**

**What is the difference between the *Bernoulli* and *Binomial* distribution?**

Answer

- The **Bernoulli distribution** is the *discrete probability distribution* of a random variable which takes a **binary** output: 1 with probability  $p$ , and 0 with probability  $(1-p)$ . The idea is that, whenever we are running an experiment that might lead either to *success* or to a *failure*, we can

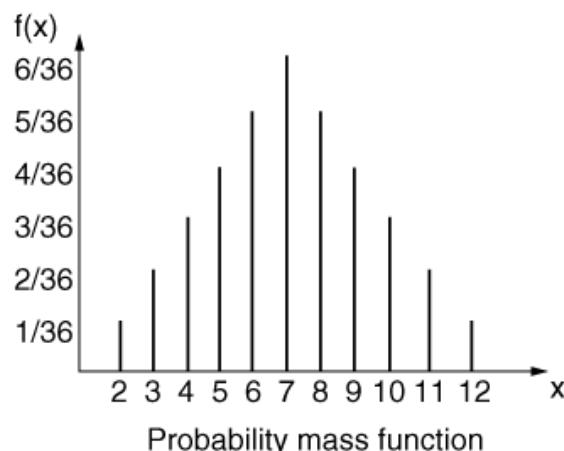
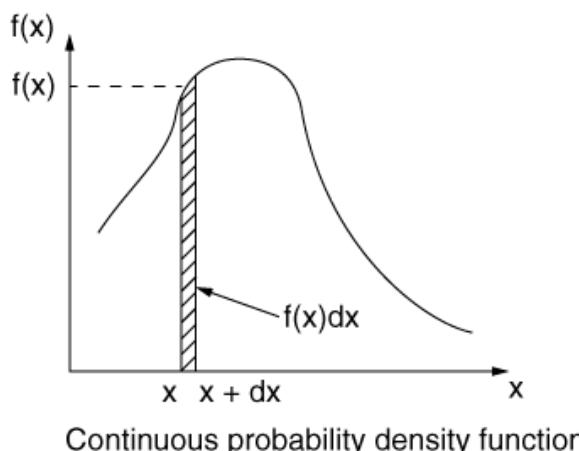
associate with our *success* (labeled with 1) a probability  $p$ , while our *failure* (labeled with 0) will have probability  $(1-p)$ .

- In the **Binomial distribution** we keep the same idea as before: we count *probability* of a *success* or *failure* outcome in an experiment, but this time it is *repeated multiple times*.

**Q6:**

**What's the difference between *Probability Mass Functions* and *Density Probability Functions*?**

- **Probability mass functions** are used to describe *discrete probability* distributions and allow us to determine the probability of an observation being *exactly equal* to a target value.
- **Density functions** are used to describe *continuous probability* distributions and allows us to determine the probability of an observation being within a *range around our target value* by computing the *area under the curve* for our interval.



**Q7:**

A coin was flipped 1000 times, and 550 times it showed up heads. Do you think the coin is biased?

To answer this question let's say  $\diamond X$  is the number of heads and let's assume that the coin is not biased. Since each individual flip is a Bernoulli random variable, we can assume it has a probability of showing up heads as  $p = 0.5$ , so this will lead to the following expected number of heads:

$$1000 \times 0.5 = 500 \mu = np = 1000 \times 0.5 = 500$$

And the following standard deviation:

$$1000 \times 0.5 \times 0.5 = 250 \approx 16 \sigma = np(1-p) = 1000 \times 0.5 \times 0.5 = 250 \approx 16$$

Given that we got a **1000** sample size, we can apply the Central Limit Theorem to approximate the total number of heads as normal distribution and calculate the corresponding z-score to test the hypothesis that the coin is fair.

$$550 - 500 / 16 = 3.125 > 3$$

This means that, if the coin were fair, the event of seeing 550 heads should occur with a **< 1%** chance under normality assumptions. Therefore, the coin is likely biased.

**Q8:**

**Find a probability of dangerous Fire when there is Smoke**

Let's say:

- the probability of dangerous fires are rare (1%)
- but smoke is fairly common (10%) due to barbecues,
- and 90% of dangerous fires make smoke

Can you find the probability of dangerous Fire when there is Smoke?

Answer

We can then discover the **probability of dangerous Fire when there is Smoke** using the Bayes' Theorem:

$$\begin{aligned} P(\text{Fire} | \text{Smoke}) &= P(\text{Fire}) * P(\text{Smoke} | \text{Fire}) / P(\text{Smoke}) \\ &= 0.01 * 0.9 / 0.1 \\ &= 0.09 (9\%) \end{aligned}$$

So probability of dangerous fire when there is a smoke is 9%.

**1. Which of the following relation is correct for a negative skewed distribution?**

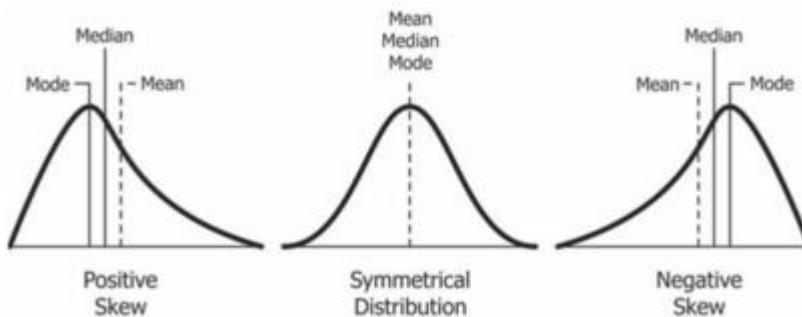
(a) Mean=Mode=Median

(b) Mean>Median>Mode

(c) Mode>Median>Mean

(d) Mean>Mode=Median

**Solution:(c)**



**Explanation:**

## 2. In the symmetric covariance matrix:

- (a) Diagonal elements must be positive and other elements are always zero.
- (b) Diagonal elements can never be negative and other elements are always positive.
- (c) Diagonal elements can never be negative and other elements can be negative or positive.
- (d) Diagonal elements can be negative and positive and other elements are always negative.

**Solution: (c)**

**Explanation:** In a covariance matrix, the diagonal entries represent covariance of the variable with itself which is equal to the variance of that variable and is calculated as the square of standard deviation. Since variance is always positive, therefore diagonal entries are always positive.

**3. Presence of Outliers in a dataset not affects:**

- (a) Standard deviation
- (b) Range
- (c) Mean
- (d) Inter-quartile Range(IQR)

**Solution: (d)**

**Explanation: The IQR is essentially the range of the middle 50% of the data. Since it uses the middle 50%, therefore it is not affected by the outliers.**

**4. If X and Y are independent random variables, then which of the following is TRUE?**

- (a)  $E(XY)=E(X)E(Y)$  [ E represents Expectation value ]
- (b)  $\text{Cov}(X,Y)=0$  [ Cov represents covariance between variables ]
- (c)  $\text{Var}(X+Y)=\text{Var}(X)+\text{Var}(Y)$  [ Var represents variance ]
- (d) All of the above

**Solution: (d)**

**Explanation:** If X and Y are independent then  $\text{Cov}(X,Y)=0$  and  $\text{Var}(X+Y) = \text{Var}(X)+\text{Var}(Y)$  ( $\because 2\text{Cov}(X, Y) = 0$ )

**5. For a normal distribution Z, which option is TRUE?**

- (a) Coefficient of skewness ( $E(Z^3))=0$
- (b)  $E(Z)=0 ; E(Z^2)=\text{Var}(Z)=1$
- (c) Kurtosis ( $E(Z^4))=3$
- (d) Its density is symmetric about the mean.

**Solution:** (d)

**Explanation:**

**6. Let X and Y be normal random variables with their respective means 3 and 4 and variances 9 and 16, then  $2X-Y$  will have normal distribution with parameters:**

- (a) Mean=2 and Variance=52
- (b) Mean=0 and Variance=1
- (c) Mean=2 and Variance=1
- (d) None of the above

**Solution:** (d)

**Hint:**  $\text{Var}(aX + bY) = a^2 \text{Var}(X) + b^2 \text{Var}(Y) + 2ab\text{Cov}(X,Y)$

**7. Suppose X and Y take values {0,1} and are independent with  $P(X=1)=1/2$  and  $P(Y=1)=1/3$ . What is the probability that  $P(X+Y=1)$ ?**

(a) 5/18

(b) 1/2

(c) 5/6

(d) 1/6

**Solution:(b)**

**Explanation:**  $P(X+Y=1) = P(X=0).P(Y=1) + P(X=1).P(Y=0) = (1/2)(1/3) + (1/2)(2/3) = 1/2.$

**8. Let X and Y are random variables with  $E(X)=\mu/2$  and  $E(Y)=\mu$ , then which one is TRUE?**

(a)  $g=X+Y$  is an unbiased estimator of  $\mu$

(b)  $g = X+Y$  is a biased estimator of  $\mu$  with bias equals to  $\mu$

(c)  $h=X+(Y/2)$  is an unbiased estimator of  $\mu$

(d)  $h= X+(Y/2)$  is a biased estimator of  $\mu$  with bias equals to  $\mu/2$

**Solution: (c)**

**Explanation:**  $E(g)= E(X+Y)= E(X) + E(Y)=3\mu/2 ; \text{Bias}(g)= E(g)-\mu = \mu/2$

$E(h)= E(X+(Y/2))= E(X) + 1/2E(Y) = \mu, \text{Bias}(h)= E(h)-\mu = 0$

**9. Suppose that X takes values between 0 and 1 and has probability density function(PDF)  $2x$ , then the value of Variance of  $X^2$  is :**

(a) 1/12

(b) 1/18

(c) 1/6

(d) 5/18

**Solution:(a)**

**Hint:** Use  $\text{Var}(X^2) = E(X^4) - (E(X^2))^2$

**10.** For random variables X and Y, we have  $\text{Var}(X)=1$ ,  $\text{Var}(Y)=4$ , and  $\text{Var}(2X-3Y)=34$ , then the correlation between X and Y is:

- (a) 1/2
- (b) 1/4
- (c) 1/3
- (d) None of the above

**Solution:(b)**

**Explanation:**  $\text{Var}(2X-3Y) = 34$

$$= 4\text{Var}(X) + 9\text{Var}(Y) - 12\text{Cov}(X, Y)$$

$$= 4(1) + 9(4) - 12\text{Cov}(X, Y) = 34$$

$$\therefore \text{Cov}(X, Y) = 1/2$$

**11.** A fair die is rolled repeatedly until a number larger than 4 is observed. If K is the total number of times that the die is rolled, then  $P(K=4)$  is equal to:

(a) 16/81

(b) 8/81

(c) 8/27

(d) 16/27

**Solution:** (b)

**Explanation:**  $P(K=4) = (P(\text{#less than 4 or equal}))^3 \cdot P(\{4\}) = (2/3)^3 \cdot (1/3) = 8/81.$

**12. Let X and Y be independent uniform (0, 1) random variables. Define A=X+Y and B=X-Y. Then,**

(a) A and B are independent random variables

(b) A and B are uncorrelated random variables

(c) A and B are both uniforms (0,1) random variables.

(d) None of these

**Solution:** (b)

**Explanation:**  $\text{Cov}(X+Y, X-Y) = \text{Cov}(X, X) - \text{Cov}(X, Y) + \text{Cov}(Y, X) - \text{Cov}(Y, Y) \Rightarrow \text{Var}(X) - \text{Var}(Y) = 0$

**13. If g is a point estimator of X, then Mean Square error(MSE) for g is:**

(a)  $\text{Variance}(g) + \text{Bias}(g)$

(b)  $\text{Variance}(g) + \text{Bias}(g^2)$

(c)  $\text{Variance}(g) + (\text{Bias}(g))^2$

(d)  $\text{Variance}(g^2) + \text{Bias}(g)$

**Solution:** (c)

**Explanation:**  $\text{MSE}(g) = E[(g-X)^2] = \text{Var}(g-X) + (E[g-X])^2 = \text{Var}(g) + (\text{Bias}(g))^2$

**14.** Let  $X$  and  $Y$  be two random variables and let  $a, b, c, d$  be real numbers, then which one of the following is FALSE?

(a)  $\text{Cov}(X+b, Y+d) = \text{Cov}(X, Y)$

(b)  $\text{Cov}(aX, cY) = ac * \text{Cov}(X, Y)$

(c)  $\text{Cov}(aX+b, cY+d) = ac * \text{Cov}(X, Y)$

(d)  $\text{Corr}(aX+b, cY+d) = ac * \text{Corr}(X, Y)$  for  $a,c > 0$

**Solution:** (d)

**Explanation:**  $\text{Corr}(aX+b, cY+d) = \text{Corr}(X, Y)$

**15.** Let  $X$  and  $Y$  be jointly(bivariate) normal with  $\text{Var}(X) = \text{Var}(Y)$ , then:

(a)  $X+Y$  and  $X-Y$  are jointly normal

(b)  $X+Y$  and  $X-Y$  are uncorrelated

(c)  $X+Y$  and  $X-Y$  are independent

(d) All of the above

**Solution:** (d)

**Explanation:** If  $X$  and  $Y$  be the bivariate normal distribution, then any linear combination of  $X$  and  $Y$  is also normally distributed.

**16.** Let  $X_1, X_2, X_3, \dots, X_n$  be a random sample from a distribution with  $E(X_i) = \mu$  and  $\text{Var}(X_i) = \sigma^2$ . Now, consider two estimators:

$$g_1 = X_1 \quad g_2 = \bar{X} = (X_1 + X_2 + X_3 + \dots + X_n)/n$$

Which of these estimator has high mean squared error(MSE)?

- (a)  $g_1$
- (b)  $g_2$
- (c) Same for both  $g_1$  and  $g_2$
- (d) None of the above

**Solution:** (a)

**Explanation:**  $\text{MSE}(g_1) = E[(g_1 - \mu)^2] = E[(X_1 - E(X_1))^2] = \text{Var}(X_1) = \sigma^2$

$$\text{MSE}(g_2) = E[(g_2 - \mu)^2] = E[(\bar{X} - \mu)^2] = \text{Var}(\bar{X}) + (E[\bar{X}] - \mu)^2 = \text{Var}(\bar{X}) = \sigma^2/n$$

**17.** A random sample of  $n=6$  taken from the population has the elements 6, 10, 13, 14, 18, 20. Then, which option is False?

- (a) Point estimate for population mean is 13.5
- (b) Point estimate for population standard deviation is 4.68
- (c) Point estimate for population standard deviation is 3.5

(d) Point estimate for standard error of mean is 1.91

**Solution:** (c)

**Explanation:** Population mean( $X'$ ) = ( $\Sigma X_i/n$ ) = 13.5

$$\begin{aligned} \text{Population standard deviation}(S) &= \sqrt{(\Sigma X_i^2/n) - (\Sigma X_i/n)^2} \\ &= 4.68 \end{aligned}$$

$$\text{Standard error of mean} = S/\sqrt{n} = 4.68/\sqrt{6} = 1.91$$

## Section-2

**18. True or False:** If the Pearson's correlation between 2 variables is zero, then they are necessarily independent.

**Solution:** False

**Explanation:** Correlation is a measure of linear dependence between the variables.

**19. True or False:** Let  $g$  be an unbiased estimator of  $X$  and  $U$  be a random variable with zero means, then  $h=g+U$  is also unbiased for  $X$ .

**Solution:** True.

**Explanation:**  $E(h) = E(g) + E(U) = 0+0=0$  ( $\because E(g)=0$  due to unbiased estimator)

**20. True or False:** Let  $X$  and  $Y$  be two independent standard normal random variables and  $T=XY^2+X+1$  and  $P=X-3$ , then  $Cov(T, P)=1$

**Solution:** False.

**Hint:** Use properties mentioned in Question-14.

**21. True or False:** Let  $X$  has a normal distribution with parameters  $\mu$  and  $\sigma^2$ , then  $X^2$  follows a chi-square distribution with parameter 1.

**Solution:** False.

**Explanation:** For the given statement to be True,  $X$  should be Standard normal distribution( $\mu=0, \sigma^2=1$ )

**22. True or False:** If the characteristic function of a random variable exists, then its expectation and variance will also exist.

**Solution:** False.

**Hint:** Moment Generating Function(MGF)

**23. True or False:** Let  $X$  has uniform distribution  $U(a, b)$  such that  $E(X)=2$  and  $\text{Var}(X)=3/4$ , then  $P(X<1)=1/6$ .

**Solution:** True.

**Explanation:**  $E(X) = (a+b)/2 = a+b=4$  ;  $\text{Var}(X) = (b-a)^2/12 = (b-a)=3 \Rightarrow X \sim U(0.5, 3.5)$

**24. True or False:** The correlation coefficient between  $X+Y$  and  $X-Y$ , where  $X$  and  $Y$  are independent random variables with variances 36 and 16 respectively is 6/13.

**Solution:** False.

**Explanation:**  $\text{Corr}(X+Y, X-Y) = \text{Cov}(X+Y, X-Y) / \text{Std}(X+Y).\text{Std}(X-Y)$  [Std= Standard Deviation]

**25. True or False:** In interval estimation, As the confidence level increases the margin of error decreases.

**Solution:** False.

**Explanation:** The Confidence Interval is defined as  $X \pm Z(s/\sqrt{n})$

**1. What is the difference between quantitative and qualitative data?**

Quantitative data is data defined by a numeric value such as a count or range—for example, a person's height in cm. Qualitative data is described as a quality or characteristic and is usually presented in words. For example, using words like 'tall' or 'short' to describe a person's height.

**2. Name three different types of encoding techniques when dealing with qualitative data.**

Label Encoding, One-Hot Encoding, Binary Encoding

### **3. Explain the bias-variance trade-off.**

The bias-variance trade-off is the trade-off between the error introduced by the bias and the error introduced by a model's variance. A highly biased model is too simple and doesn't fit well enough to the training data.

A model with high variance fits exceptionally well to the training data and cannot generalize outside the data it was trained on. The bias-variance trade-off involves finding a sweet spot to build a machine learning model that fits well enough onto the training data and can generalize and perform well on test data.

### **1. How is the statistical significance of an insight assessed?**

Hypothesis testing is used to find out the statistical significance of the insight. To elaborate, the null hypothesis and the alternate hypothesis are stated, and the p-value is calculated.

After calculating the p-value, the null hypothesis is assumed true, and the values are determined. To fine-tune the result, the alpha value, which denotes the significance, is tweaked. If the p-value turns out to be less than the alpha, then the null hypothesis is rejected. This ensures that the result obtained is statistically significant.

### **2. Where are long-tailed distributions used?**

A long-tailed distribution is a type of distribution where the tail drops off gradually toward the end of the curve.

The Pareto principle and the product sales distribution are good examples to denote the use of long-tailed distributions. Also, it is widely used in classification and regression problems.

### **3. What is the central limit theorem?**

The central limit theorem states that the normal distribution is arrived at when the sample size varies without having an effect on the shape of the population distribution.

This central limit theorem is the key because it is widely used in performing hypothesis testing and also to calculate the confidence intervals accurately.

#### **4. What is observational and experimental data in Statistics?**

Observational data correlates to the data that is obtained from observational studies, where variables are observed to see if there is any correlation between them.

Experimental data is derived from experimental studies, where certain variables are held constant to see if any discrepancy is raised in the working.

#### **5. What is meant by mean imputation for missing data? Why is it bad?**

Mean imputation is a rarely used practice where null values in a dataset are replaced directly with the corresponding mean of the data.

It is considered a bad practice as it completely removes the accountability for feature correlation. This also means that the data will have low variance and increased bias, adding to the dip in the accuracy of the model, alongside narrower confidence intervals.

#### **6. What is an outlier? How can outliers be determined in a dataset?**

Outliers are data points that vary in a large way when compared to other observations in the dataset. Depending on the learning process, an outlier can worsen the accuracy of a model and decrease its efficiency sharply.

Outliers are determined by using two methods:

- Standard deviation/z-score

- Interquartile range (IQR)

## 7. How is missing data handled in statistics?

There are many ways to handle missing data in Statistics:

- Prediction of the missing values
- Assignment of individual (unique) values
- Deletion of rows, which have the missing data
- Mean imputation or median imputation
- Using random forests, which support the missing values

## 8. What is exploratory data analysis?

Exploratory data analysis is the process of performing investigations on data to understand the data better.

In this, initial investigations are done to determine patterns, spot abnormalities, test hypotheses, and also check if the assumptions are right.

## 9. What is the meaning of selection bias?

Selection bias is a phenomenon that involves the selection of individual or grouped data in a way that is not considered to be random. Randomization plays a key role in performing analysis and understanding model functionality better.

If correct randomization is not achieved, then the resulting sample will not accurately represent the population.

## 10. What are the types of selection bias in statistics?

There are many types of selection bias as shown below:

- Observer selection
- Attrition

- Protopathic bias
- Time intervals
- Sampling bias

## **11. What is the meaning of an inlier?**

An inlier is a data point that lies at the same level as the rest of the dataset. Finding an inlier in the dataset is difficult when compared to an outlier as it requires external data to do so. Inliers, similar to outliers reduce model accuracy. Hence, even they are removed when they're found in the data. This is done mainly to maintain model accuracy at all times.

## **12. What is the probability of throwing two fair dice when the sum is 5 and 8?**

There are 4 ways of rolling a 5 (1+4, 4+1, 2+3, 3+2):

$$P(\text{Getting a 5}) = 4/36 = 1/9$$

Now, there are 7 ways of rolling an 8 (1+7, 7+1, 2+6, 6+2, 3+5, 5+3, 4+4)

$$P(\text{Getting an 8}) = 7/36 = 0.194$$

## **13. State the case where the median is a better measure when compared to the mean.**

In the case where there are a lot of outliers that can positively or negatively skew data, the median is preferred as it provides an accurate measure in this case of determination.

## **14. Can you give an example of root cause analysis?**

Root cause analysis, as the name suggests, is a method used to solve problems by first identifying the root cause of the problem.

Example: If the higher crime rate in a city is directly associated with the higher sales in a red-colored shirt, it means that they are having a positive correlation. However, this does not mean that one causes the other.

Causation can always be tested using A/B testing or hypothesis testing.

## **15. What is the meaning of six sigma in statistics?**

Six sigma is a quality assurance methodology used widely in statistics to provide ways to improve processes and functionality when working with data.

A process is considered as six sigma when 99.99966% of the outcomes of the model are considered to be defect-free.

## **16. What is DOE?**

DOE is an acronym for the Design of Experiments in statistics. It is considered as the design of a task that describes the information and the change of the same based on the changes to the independent input variables.

## **17. What is the meaning of KPI in statistics?**

KPI stands for Key Performance Analysis in statistics. It is used as a reliable metric to measure the success of a company with respect to its achieving the required business objectives.

There are many good examples of KPIs:

- Profit margin percentage
- Operating profit margin
- Expense ratio

## **18. What type of data does not have a log-normal distribution or a Gaussian distribution?**

Exponential distributions do not have a log-normal distribution or a Gaussian distribution. In fact, any type of data that is categorical will not have these distributions as well.

Example: Duration of a phone call, time until the next earthquake, etc.

## **19. What is the Pareto principle?**

The Pareto principle is also called the 80/20 rule, which means that 80 percent of the results are obtained from 20 percent of the causes in an experiment.

A simple example of the Pareto principle is the observation that 80 percent of peas come from 20 percent of pea plants on a farm.

## **20. What is the meaning of the five-number summary in Statistics?**

The five-number summary is a measure of five entities that cover the entire range of data as shown below:

- Low extreme (Min)
- First quartile (Q1)
- Median
- Upper quartile (Q3)
- High extreme (Max)

## **21. What are population and sample in Inferential Statistics, and how are they different?**

A population is a large volume of observations (data). The sample is a small portion of that population. Because of the large volume of data in the population, it raises the computational cost. The availability of all data points in the population is also an issue.

In short:

- We calculate the statistics using the sample.
- Using these sample statistics, we make conclusions about the population.

## **22. What are quantitative data and qualitative data?**

- Quantitative data is also known as numeric data.
- Qualitative data is also known as categorical data.

## **23. What is Mean?**

Mean is the average of a collection of values. We can calculate the mean by dividing the sum of all observations by the number of observations.

## **24. What is the meaning of standard deviation?**

Standard deviation represents the magnitude of how far the data points are from the mean. A low value of standard deviation is an indication of the data being close to the mean, and a high value indicates that the data is spread to extreme ends, far away from the mean.

## **25. What is a bell-curve distribution?**

A normal distribution can be called a bell-curve distribution. It gets its name from the bell curve shape that we get when we visualize the distribution.

## **26. What is skewness?**

Skewness measures the lack of symmetry in a data distribution. It indicates that there are significant differences between the mean, the mode, and the median of data. Skewed data cannot be used to create a normal distribution.

## **27. What is kurtosis?**

Kurtosis is used to describe the extreme values present in one tail of distribution versus the other. It is actually the measure of outliers present in the distribution. A high value of kurtosis represents large amounts of outliers being present in data. To overcome this, we have to either add more data into the dataset or remove the outliers.

## **28. What is correlation?**

Correlation is used to test relationships between quantitative variables and categorical variables. Unlike covariance, correlation tells us how strong the relationship is between two variables. The value of correlation between two variables ranges from -1 to +1.

The -1 value represents a high negative correlation, i.e., if the value in one variable increases, then the value in the other variable will drastically decrease. Similarly, +1 means a positive correlation, and here, an increase in one variable will lead to an increase in the other. Whereas, 0 means there is no correlation.

If two variables are strongly correlated, then they may have a negative impact on the statistical model, and one of them must be dropped.

Next up on this top Statistics Interview Questions and Answers blog, let us take a look at the intermediate set of questions.

## **29. What are left-skewed and right-skewed distributions?**

A left-skewed distribution is one where the left tail is longer than that of the right tail. Here, it is important to note that the mean < median < mode.

Similarly, a right-skewed distribution is one where the right tail is longer than the left one. But, here mean > median > mode.

## **30. What is the difference between Descriptive and Inferential Statistics?**

**Descriptive Statistics:** Descriptive statistics is used to summarize a sample set of data like the standard deviation or the mean.

**Inferential statistics:** Inferential statistics is used to draw conclusions from the test data that are subjected to random variations.

### **31. What are the types of sampling in Statistics?**

There are four main types of data sampling as shown below:

- **Simple random:** Pure random division
- **Cluster:** Population divided into clusters
- **Stratified:** Data divided into unique groups
- **Systematical:** Picks up every ‘n’ member in the data

### **32. What is the meaning of covariance?**

Covariance is the measure of indication when two items vary together in a cycle. The systematic relation is determined between a pair of random variables to see if the change in one will affect the other variable in the pair or not.

### **33. Imagine that Jeremy took part in an examination. The test is having a mean score of 160, and it has a standard deviation of 15. If Jeremy's z-score is 1.20, what would be his score on the test?**

To determine the solution to the problem, the following formula is used:

$$X = \mu + Z\sigma$$

Here:

$\mu$ : Mean

$\sigma$ : Standard deviation

X: Value to be calculated

Therefore,  $X = 160 + (15*1.2) = 173.8$  (Approximated to 174)

**34. If a distribution is skewed to the right and has a median of 20, will the mean be greater than or less than 20?**

If the given distribution is a right-skewed distribution, then the mean should be greater than 20, while the mode remains to be less than 20.

**35. What is Bessel's correction?**

Bessel's correction is a factor that is used to estimate a populations' standard deviation from its sample. It causes the standard deviation to be less biased, thereby, providing more accurate results.

**36. The standard normal curve has a total area to be under one, and it is symmetric around zero. True or False?**

True, a normal curve will have the area under unity and the symmetry around zero in any distribution. Here, all of the measures of central tendencies are equal to zero due to the symmetric nature of the standard normal curve.

**37. In an observation, there is a high correlation between the time a person sleeps and the amount of productive work he does. What can be inferred from this?**

First, correlation does not imply causation here. Correlation is only used to measure the relationship, which is linear between rest and productive work. If both vary rapidly, then it means that there is a high amount of correlation between them.

**38. What is the relationship between the confidence level and the significance level in statistics?**

The significance level is the probability of obtaining a result that is extremely different from the condition where the null hypothesis is true. While the confidence level is used as a range of similar values in a population.

Both significance and confidence level are related by the following formula:

$$\text{Significance level} = 1 - \text{Confidence level}$$

**39. A regression analysis between apples (y) and oranges (x) resulted in the following least-squares line:  $y = 100 + 2x$ . What is the implication if oranges are increased by 1?**

If the oranges are increased by one, there will be an increase of 2 apples since the equation is:

$$y = 100 + 2x.$$

**40. What types of variables are used for Pearson's correlation coefficient?**

Variables to be used for the Pearson's correlation coefficient must be either in a ratio or in an interval.

Note that there can exist a condition when one variable is a ratio, while the other is an interval score.

**41. In a scatter diagram, what is the line that is drawn above or below the regression line called?**

The line that is drawn above or below the regression line in a scatter diagram is called the residual or also the prediction error.

**42. What are the examples of symmetric distribution?**

Symmetric distribution means that the data on the left side of the median is the same as the one present on the right side of the median.

There are many examples of symmetric distribution, but the following three are the most widely used ones:

- Uniform distribution
- Binomial distribution
- Normal distribution

#### **43. Where is inferential statistics used?**

Inferential statistics is used for several purposes, such as research, in which we wish to draw conclusions about a population using some sample data. This is performed in a variety of fields, ranging from government operations to quality control and quality assurance teams in multinational corporations.

#### **44. What is the relationship between mean and median in a normal distribution?**

In a normal distribution, the mean is equal to the median. To know if the distribution of a dataset is normal, we can just check the dataset's mean and median.

#### **45. What is the difference between the I<sup>st</sup> quartile, the II<sup>nd</sup> quartile, and the III<sup>rd</sup> quartile?**

Quartiles are used to describe the distribution of data by splitting data into three equal portions, and the boundary or edge of these portions are called quartiles.

That is,

- **The lower quartile (Q1)** is the 25th percentile.
- **The middle quartile (Q2)**, also called the median, is the 50th percentile.
- **The upper quartile (Q3)** is the 75th percentile.

## **46. How do the standard error and the margin of error relate?**

The standard error and the margin of error are quite closely related to each other. In fact, the margin of error is calculated using the standard error. As the standard error increases, the margin of error also increases.

## **47. What is one sample t-test?**

This T-test is a statistical hypothesis test in which we check if the mean of the sample data is statistically or significantly different from the population's mean.

## **48. What is an alternative hypothesis?**

The alternative hypothesis (denoted by  $H_1$ ) is the statement that must be true if the null hypothesis is false. That is, it is a statement used to contradict the null hypothesis. It is the opposing point of view that gets proven right when the null hypothesis is proven wrong.

## **49. Given a left-skewed distribution that has a median of 60, what conclusions can we draw about the mean and the mode of the data?**

Given that it is a left-skewed distribution, the mean will be less than the median, i.e., less than 60, and the mode will be greater than 60.

## **50. What are the types of biases that we encounter while sampling?**

Sampling biases are errors that occur when taking a small sample of data from a large population as the representation in statistical analysis. There are three types of biases:

- The selection bias
- The survivorship bias
- The undercoverage bias

Next up on this top Statistics Interview Questions and answers blog, let us take a look at the advanced set of questions.

### **51. What are the scenarios where outliers are kept in the data?**

There are not many scenarios where outliers are kept in the data, but there are some important situations when they are kept. They are kept in the data for analysis if:

- Results are critical
- Outliers add meaning to the data
- The data is highly skewed

### **52. Briefly explain the procedure to measure the length of all sharks in the world.**

Following steps can be used to determine the length of sharks:

- Define the confidence level (usually around 95%)
- Use sample sharks to measure
- Calculate the mean and standard deviation of the lengths
- Determine t-statistics values
- Determine the confidence interval in which the mean length lies

### **53. How does the width of the confidence interval change with length?**

The width of the confidence interval is used to determine the decision-making steps. As the confidence level increases, the width also increases.

The following also apply:

- Wide confidence interval: Useless information
- Narrow confidence interval: High-risk factor

#### **54. What is the meaning of degrees of freedom (DF) in statistics?**

Degrees of freedom or DF is used to define the number of options at hand when performing an analysis. It is mostly used with t-distribution and not with the z-distribution.

If there is an increase in DF, the t-distribution will reach closer to the normal distribution. If  $DF > 30$ , this means that the t-distribution at hand is having all of the characteristics of a normal distribution.

#### **55. How can you calculate the p-value using MS Excel?**

Following steps are performed to calculate the p-value easily:

- Find the Data tab above
- Click on Data Analysis
- Select Descriptive Statistics
- Select the corresponding column
- Input the confidence level

#### **56. What is the law of large numbers in statistics?**

The law of large numbers in statistics is a theory that states that the increase in the number of trials performed will cause a positive proportional increase in the average of the results becoming the expected value.

Example: The probability of flipping a fair coin and landing heads is closer to 0.5 when it is flipped 100,000 times when compared to 100 flips.

#### **57. What are some of the properties of a normal distribution?**

A normal distribution, regardless of its size, will have a bell-shaped curve that is symmetric along the axes.

Following are some of the important properties:

- Unimodal: It has only one mode.
- Symmetrical: Left and right halves of the curve are mirrored.
- Central tendency: The mean, median, and mode are at the midpoint.

**58. If there is a 30 percent probability that you will see a supercar in any 20-minute time interval, what is the probability that you see at least one supercar in the period of an hour (60 minutes)?**

The probability of not seeing a supercar in 20 minutes is:

$$\begin{aligned} &= 1 - P(\text{Seeing one supercar}) \\ &= 1 - 0.3 \\ &= 0.7 \end{aligned}$$

Probability of not seeing any supercar in the period of 60 minutes is:

$$= (0.7)^3 = 0.343$$

Hence, the probability of seeing at least one supercar in 60 minutes is:

$$\begin{aligned} &= 1 - P(\text{Not seeing any supercar}) \\ &= 1 - 0.343 = 0.657 \end{aligned}$$

**59. What is the meaning of sensitivity in statistics?**

Sensitivity, as the name suggests, is used to determine the accuracy of a classifier (logistic, random forest, etc.):

The simple formula to calculate sensitivity is:

$$\text{Sensitivity} = \text{Predicted True Events} / \text{Total number of Events}$$

**60. What are the types of biases that you can encounter while sampling?**

There are three types of biases:

- Selection bias
- Survivorship bias
- Under coverage bias

## 61. What is the meaning of TF/IDF vectorization?

TF-IDF is an acronym for Term Frequency – Inverse Document Frequency. It is used as a numerical measure to denote the importance of a word in a document. This document is usually called the collection or the corpus.

The TF-IDF value is directly proportional to the number of times a word is repeated in a document. TF-IDF is vital in the field of Natural Language Processing (NLP) as it is mostly used in the domain of text mining and information retrieval.

## 62. What are some of the low and high-bias Machine Learning algorithms?

There are many low and high-bias Machine Learning algorithms, and the following are some of the widely used ones:

- **Low bias:** **SVM**, decision trees, KNN algorithm, etc.
- **High bias:** Linear and logistic regression

## 63. What is the use of Hash tables in statistics?

Hash tables are the data structures that are used to denote the representation of key-value pairs in a structured way. The hashing function is used by a hash table to compute an index that contains all of the details regarding the keys that are mapped to their associated values.

## 64. What are some of the techniques to reduce underfitting and overfitting during model training?

Underfitting refers to a situation where data has high bias and low variance, while overfitting is the situation where there are high variance and low bias.

Following are some of the techniques to reduce underfitting and overfitting:

#### **For reducing underfitting:**

- Increase model complexity
- Increase the number of features
- Remove noise from the data
- Increase the number of training epochs

#### **For reducing overfitting:**

- Increase training data
- Stop early while training
- Lasso regularization
- Use random dropouts

#### **65. Can you give an example to denote the working of the central limit theorem?**

Let's consider the population of men who have normally distributed weights, with a mean of 60 kg and a standard deviation of 10 kg, and the probability needs to be found out.

If one single man is selected, the weight is greater than 65 kg, but if 40 men are selected, then the mean weight is far more than 65 kg.

The solution to this can be as shown below:

$$Z = (x - \mu) / \sigma = (65 - 60) / 10 = 0.5$$

For a normal distribution  $P(Z > 0.5) = 0.409$

$$Z = (65 - 60) / 5 = 1$$

$$P(Z > 1) = 0.090$$

## **66. How do you stay up-to-date with the new and upcoming concepts in statistics?**

This is a commonly asked question in a statistics interview. Here, the interviewer is trying to assess your interest and ability to find out and learn new things efficiently. Do talk about how you plan to learn new concepts and make sure to elaborate on how you practically implemented them while learning.

If you are looking forward to learning and mastering all of the Data Analytics and Data Science concepts and earn a certification in the same, do take a look at Intellipaat's latest **Data Science with R Certification** offerings.

## **67. What is the benefit of using box plots?**

Box plots allow us to provide a graphical representation of the 5-number summary and can also be used to compare groups of histograms.

## **68. Does a symmetric distribution need to be unimodal?**

A symmetric distribution does not need to be unimodal (having only one mode or one value that occurs most frequently). It can be bi-modal (having two values that have the highest frequencies) or multi-modal (having multiple or more than two values that have the highest frequencies).

## **69. What is the impact of outliers in statistics?**

Outliers in statistics have a very negative impact as they skew the result of any statistical query. For example, if we want to calculate the mean of a dataset that contains outliers, then the mean calculated will be different from the actual mean (i.e., the mean we will get once we remove the outliers).

## **70. When creating a statistical model, how do we detect overfitting?**

Overfitting can be detected by cross-validation. In cross-validation, we divide the available data into multiple parts and iterate on the entire dataset. In each iteration, one part is used for testing, and others are used for training. This way, the entire dataset will be used for training and testing purposes, and we can detect if the data is being overfitted.

## 71. What is a survivorship bias?

The survivorship bias is the flaw of the sample selection that occurs when a dataset only considers the ‘surviving’ or existing observations and fails to consider those observations that have already ceased to exist.

## 72. What is an undercoverage bias?

The undercoverage bias is a bias that occurs when some members of the population are inadequately represented in the sample.

## 74. What is the relationship between standard deviation and standard variance?

Standard deviation is the square root of standard variance. Basically, standard deviation takes a look at how the data is spread out from the mean. On the other hand, standard variance is used to describe how much the data varies from the mean of the entire dataset.

1. **[Facebook - Easy]** [\[Coin Fairness Test on DataLemur\]](#) There is a fair coin (one side heads, one side tails) and an unfair coin (both sides tails). You pick one at random, flip it 5 times, and observe that it comes up as tails all five times. What is the chance that you are flipping the unfair coin?
2. **[Lyft - Easy]** You and your friend are playing a game. The two of you will continue to toss a coin until the sequence HH or TH shows up. If HH shows up first, you win. If TH shows up first, your friend wins. What is the probability of you winning?
3. **[Google - Easy]** What is the probability that a seven-game series goes to 7 games?

4. **[Facebook - Easy]** Facebook has a content team that labels pieces of content on the platform as spam or not spam. 90% of them are diligent raters and will label 20% of the content as spam and 80% as non-spam. The remaining 10% are non-diligent raters and will label 0% of the content as spam and 100% as non-spam. Assume the pieces of content are labeled independently from one another, for every rater. Given that a rater has labeled 4 pieces of content as good, what is the probability that they are a diligent rater?
5. **[Bloomberg - Easy]** Say you draw a circle and choose two chords at random. What is the probability that those chords will intersect?
6. **[Amazon - Easy]** 1/1000 people have a particular disease, and there is a test that is 98% correct if you have the disease. If you don't have the disease, there is a 1% error rate. If someone tests positive, what are the odds they have the disease?
7. **[Facebook - Easy]** There are 50 cards of 5 different colors. Each color has cards numbered between 1 to 10. You pick 2 cards at random. What is the probability that they are not of same color and also not of same number?
8. **[Tesla - Easy]** A fair six-sided die is rolled twice. What is the probability of getting 1 on the first roll and not getting 6 on the second roll?
9. **[Facebook - Easy]** What is the expected number of rolls needed to see all 6 sides of a fair die?
10. **[Microsoft - Easy]** Three friends in Seattle each told you it's rainy, and each person has a 1/3 probability of lying. What is the probability that Seattle is rainy? Assume the probability of rain on any given day in Seattle is 0.25.
11. **[Uber - Easy]** Say you roll three dice, one by one. What is the probability that you obtain 3 numbers in a strictly increasing order?
12. **[Bloomberg - Medium]** Three ants are sitting at the corners of an equilateral triangle. Each ant randomly picks a direction and starts moving along the edge of the triangle. What is the probability that none of the ants collide? Now, what if it is  $k$  ants on all  $k$  corners of an equilateral polygon?
13. **[Two Sigma - Medium]** What is the expected number of coin flips needed to get two consecutive heads?
14. **[Amazon - Medium]** How many cards would you expect to draw from a standard deck before seeing the first ace?
15. **[Robinhood - Medium]** A and B are playing a game where A has  $n+1$  coins, B has  $n$  coins, and they each flip all of their coins. What is the probability that A will have more heads than B?

16. **[Airbnb - Medium]** Say you are given an unfair coin, with an unknown bias towards heads or tails. How can you generate fair odds using this coin?
17. **[Quora - Medium]** Say you have  $N$  i.i.d. draws of a normal distribution with parameters  $\mu$  and  $\sigma$ . What is the probability that  $k$  of those draws are larger than some value  $Y$ ?
18. **[Spotify - Hard]** A fair die is rolled  $n$  times. What is the probability that the largest number rolled is  $r$ , for each  $r$  in  $1..6$ ?
19. **[Snapchat - Hard]** There are two groups of  $n$  users, A and B, and each user in A is friends with those in B and vice versa. Each user in A will randomly choose a user in B as their best friend and each user in B will randomly choose a user in A as their best friend. If two people have chosen each other, they are mutual best friends. What is the probability that there will be no mutual best friendships?
20. **[Tesla - Hard]** Suppose there is a new vehicle launch upcoming. Initial data suggests that any given day there is either a malfunction with some part of the vehicle or possibility of a crash, with probability  $p$  which then requires a replacement. Additionally, each vehicle that has been around for  $n$  days must be replaced. What is the long-term frequency of vehicle replacements?

## 20 Statistics Problems Asked By FAANG & Hedge Funds

1. **[Facebook - Easy]** How would you explain a confidence interval to a non-technical audience?
2. **[Two Sigma - Easy]** Say you are running a multiple linear regression and believe there are several predictors that are correlated. How will the results of the regression be affected if they are indeed correlated? How would you deal with this problem?
3. **[Uber - Easy]** Describe p-values in layman's terms.
4. **[Facebook - Easy]** How would you build and test a metric to compare two user's ranked lists of movie/tv show preferences?
5. **[Microsoft - Easy]** Explain the statistical background behind power.
6. **[Twitter - Easy]** Describe A/B testing. What are some common pitfalls?
7. **[Google - Medium]** How would you derive a confidence interval from a series of coin tosses?
8. **[Stripe - Medium]** Say you model the lifetime for a set of customers using an exponential distribution with parameter  $\lambda$ , and you have the lifetime history (in months) of  $n$  customers. What is your best guess for  $\lambda$ ?

9. **[Lyft - Medium]** Derive the mean and variance of the uniform distribution  $U(a, b)$ .
10. **[Google - Medium]** Say we have  $X \sim \text{Uniform}(0, 1)$  and  $Y \sim \text{Uniform}(0, 1)$ . What is the expected value of the minimum of  $X$  and  $Y$ ?
11. **[Spotify - Medium]** You sample from a uniform distribution  $[0, d]$   $n$  times. What is your best estimate of  $d$ ?
12. **[Quora - Medium]** You are drawing from a normally distributed random variable  $X \sim N(0, 1)$  once a day. What is the approximate expected number of days until you get a value of more than 2?
13. **[Facebook - Medium]** Derive the expectation for a geometric distributed random variable.
14. **[Google - Medium]** A coin was flipped 1000 times, and 550 times it showed up heads. Do you think the coin is biased? Why or why not?
15. **[Robinhood - Medium]** Say you have  $n$  integers  $1 \dots n$  and take a random permutation. For any integers  $i, j$  let a swap be defined as when the integer  $i$  is in the  $j$ th position, and vice versa. What is the expected value of the total number of swaps?
16. **[Uber - Hard]** What is the difference between MLE and MAP? Describe it mathematically.
17. **[Google - Hard]** Say you have two subsets of a dataset for which you know their means and standard deviations. How do you calculate the blended mean and standard deviation of the total dataset? Can you extend it to  $K$  subsets?
18. **[Lyft - Hard]** How do you randomly sample a point uniformly from a circle with radius 1?
19. **[Two Sigma - Hard]** Say you continually sample from some i.i.d. uniformly distributed  $(0, 1)$  random variables until the sum of the variables exceeds 1. How many times do you expect to sample?
20. **[Uber - Hard]** Given a random Bernoulli trial generator, how do you return a value sampled from a normal distribution

## Solutions To Probability Interview Questions

### Problem #1 Solution:

We can use Bayes Theorem here. Let  $U$  denote the case where we are flipping the unfair coin and  $F$  denote the case where we are flipping a fair coin. Since the coin is chosen randomly, we know that  $P(U) = P(F) = 0.5$ . Let  $5T$  denote the event where we flip 5 heads in a row. Then we are interested in solving for

$P(U|5T)$ , i.e., the probability that we are flipping the unfair coin, given that we saw 5 tails in a row.

We know  $P(5T|U) = 1$  since by definition the unfair coin will always result in tails. Additionally, we know that  $P(5T|F) = 1/2^5 = 1/32$  by definition of a fair coin. By Bayes Theorem we have:

$$P(U|5T) = \frac{P(5T|U) * P(U)}{P(5T|U) * P(U) + P(5T|F) * P(F)} = \frac{0.5}{0.5 + 0.5 * 1/32} = 0.97$$

Therefore the probability we picked the unfair coin is about 97%.

### Problem #5 Solution:

By definition, a chord is a line segment whereby the two endpoints lie on the circle. Therefore, two arbitrary chords can always be represented by any four points chosen on the circle. If you choose to represent the first chord by two of the four points then you have:

$$\binom{4}{2} = 6$$

choices of choosing the two points to represent chord 1 (and hence the other two will represent chord 2). However, note that in this counting, we are duplicating the count of each chord twice since a chord with endpoints p1 and p2 is the same as a chord with endpoints p2 and p1. Therefore the proper number of valid chords is:

$$\frac{1}{2} \binom{4}{2} = 3$$

Among these three configurations, only exactly one of the chords will intersect, hence the desired probability is:

$$p = \frac{1}{3}$$

### Problem #13 Solution:

Let  $X$  be the number of coin flips needed until two heads. Then we want to solve for  $E[X]$ . Let  $H$  denote a flip that resulted in heads, and  $T$  denote a flip that resulted in tails. Note that  $E[X]$  can be written in terms of  $E[X|H]$  and  $E[X|T]$ , i.e. the expected number of flips needed, conditioned on a flip being either heads or tails respectively.

Conditioning on the first flip, we have:

$$E[X] = \frac{1}{2}(1 + E[X|H]) + \frac{1}{2}(1 + E[X|T])$$

Note that  $E[X|T] = E[X]$  since if a tail is flipped, we need to start over in getting two heads in a row.

To solve for  $E[X|H]$ , we can condition it further on the next outcome: either heads (HH) or tails (HT).

Therefore, we have:

$$E[X|H] = \frac{1}{2}(1 + E[X|HH]) + \frac{1}{2}(1 + E[X|HT])$$

Note that if the result is HH, then  $E[X|HH] = 0$  since the outcome was achieved, and that  $E[X|HT] = E[X]$  since a tail was flipped, we need to start over again, so:

$$E[X|H] = \frac{1}{2}(1 + 0) + \frac{1}{2}(1 + E[X]) = 1 + \frac{1}{2}E[X]$$

Plugging this into the original equation yields  $E[X] = 6$  coin flips

### Problem #15 Solution:

Consider the first n coins that A flips, versus the n coins that B flips.

There are three possible scenarios:

1. A has more heads than B
2. A and B have an equal amount of heads
3. A has less heads than B

Notice that in scenario 1, A will always win (irrespective of coin n+1), and in scenario 3, A will always lose (irrespective of coin n+1). By symmetry, these two scenarios have an equal probability of occurring.

Denote the probability of either scenario as x, and the probability of scenario 2 as y.

We know that  $2x + y = 1$  since these 3 scenarios are the only possible outcomes. Now let's consider coin n+1. If the flip results in heads, with probability 0.5, then A will have won after scenario 2 (which happens with probability y). Therefore, A's total chances of winning the game are increased by 0.5y.

Thus, the probability that A will win the game is:

$$x + \frac{1}{2}y = x + \frac{1}{2}(1 - 2x) = \frac{1}{2}$$

**Problem #18 Solution:**

Let B be the event that all n rolls have a value less than or equal to r. Then we have:

$$P(B_r) = \frac{r^n}{6^n}$$

since all n rolls must have a value less than or equal to r. Let A be the event that the largest number is r. We have:

$$B_r = B_{r-1} \cup A_r$$

and since the two events on the right hand side are disjoint, we have:

$$P(B_r) = P(B_{r-1}) + P(A_r)$$

Therefore, the probability of A is given by:

$$P(A_r) = P(B_r) - P(B_{r-1}) = \frac{r^n}{6^n} - \frac{(r-1)^n}{6^n}$$

## Solutions To Statistics Interview Questions

**Problem #2 Solution:**

There will be two main problems. The first is that the coefficient estimates and signs will vary dramatically, depending on what particular variables you include in the model. In particular, certain coefficients may even have confidence intervals that include 0 (meaning it is difficult to tell whether an increase in that X value is associated with an increase or decrease in Y). The second is that the resulting p-values will be misleading - an important variable might have a high p-value and deemed insignificant even though it is actually important.

You can deal with this problem by either removing or combining the correlated predictors. In removing the predictors, it is best to understand the causes of the correlation (i.e. did you include extraneous predictors or such as both X and 2X). For combining predictors, it is possible to include interaction terms (the

product of the two). Lastly, you should also 1) center data, and 2) try to obtain a larger sample size (which will lead to narrower confidence intervals).

### **Problem #9 Solution:**

For  $X \sim U(a, b)$  we have the following:

$$f_X(x) = \frac{1}{b - a}$$

Therefore we can calculate the mean as:

$$E[X] = \int_a^b x f_X(x) dx = \int_a^b \frac{x}{b - a} dx = \frac{x^2}{2(b - a)} \Big|_a^b = \frac{a + b}{2}$$

Similarly for variance we want:

$$Var(X) = E[X^2] - E[X]^2$$

And we have:

$$E[X^2] = \int_a^b x^2 f_X(x) dx = \int_a^b \frac{x^2}{b - a} dx = \frac{x^3}{3(b - a)} \Big|_a^b = \frac{a^2 + ab + b^2}{3}$$

Therefore:

$$Var(X) = \frac{a^2 + ab + b^2}{3} - \left(\frac{a + b}{2}\right)^2 = \frac{(b - a)^2}{12}$$

### **Problem #12 Solution:**

Since  $X$  is normally distributed, we can look at the cumulative distribution function (CDF) of the normal distribution:

$$\Phi(x) = P(X \leq x)$$

To check the probability  $X$  is at least 2, we can check (knowing that  $X$  is distributed as standard normal):

$$\Phi(2) = P(X \leq 2) = P(X \leq \mu + 2\sigma) = 0.977$$

Therefore  $P(X > 2) = 1 - 0.977 = 0.023$  for any given day. Since the draws are independent each day, then the expected time until drawing an  $X > 2$  follows a geometric distribution, with  $p = 0.023$ . Let  $T$  be a random variable denoting the number of days, then we have:

$$E[T] = \frac{1}{p} = \frac{1}{.023} \approx 43 \text{ days}$$

### **Problem #14 Solution:**

Because the sample size of flips is large (1000), we can apply the Central Limit Theorem. Since each individual flip is a Bernoulli random variable, we can assume it has a probability of showing up heads as  $p$ . Then we want to test whether  $p$  is 0.5 (i.e. whether it is fair). The Central Limit Theorem allows us to approximate the total number of heads seen as being normally distributed.

More specifically, the number of heads seen should follow a Binomial distribution since it is a sum of Bernoulli random variables. If the coin is not biased ( $p = 0.5$ ), then we have the following on the expected number of heads:

$$\mu = np = 1000 * 0.5 = 500$$

and the variance is given by:

$$\sigma^2 = np(1 - p) = 1000 * 0.5 * 0.5 = 250, \sigma = \sqrt{250} \approx 16$$

Since this mean and standard deviation specify the normal distribution, we can calculate the corresponding z-score for 550 heads:

$$z = \frac{550 - 500}{16} > 3$$

This means that, if the coin were fair, the event of seeing 550 heads should occur with a  $< 1\%$  chance under normality assumptions. Therefore, the coin is likely biased.

### **Problem #20 Solution:**

Assume we have  $n$  Bernoulli trials each with a success probability of  $p$ :

$$x_1, x_2, \dots, x_n,$$

$$x_i \sim Ber(p)$$

Assuming iid trials, we can compute the sample mean for p from a large number of trials:

$$\mu = \frac{1}{n} \sum_{i=1}^n x_i$$

Assuming iid trials, we can compute the sample mean for p from a large number of trials:

$$\mu = \frac{1}{n} \sum_{i=1}^n x_i$$

We know the expectation of this sample mean is:

$$E[\mu] = \frac{np}{n} = p$$

Additionally, we can compute the variance of this sample mean:

$$Var(\mu) = \frac{np(1-p)}{n^2} = \frac{p(1-p)}{n}$$

Assume we sample a large n. Due to the Central Limit Theorem, our sample mean will be normally distributed:

$$\mu \sim N\left(p, \frac{p(1-p)}{n}\right)$$

Therefore we can take a z-score of our sampled mean as:

$$z(\mu) = \frac{\mu - p}{\sqrt{\frac{p(1-p)}{n}}}$$

This z-score will then be a simulated value from a standard normal distribution.

*“What is the probability of choosing 2 queens out of a deck of cards?”*

This is also an example of a dependent event. In the first draw, our probability of getting a queen is 4/52. If we do get a queen in the first draw, then our probability to get another queen in the second draw would be 3/51. Hence:

$$P(2\text{queens})=P(X1=\text{queen}) \cap P(X2=\text{queen})$$

$$P(2\text{queens})=131*171$$

$$P(2\text{queens})=2211$$

Question from [Facebook](#):

*“Let’s say you have 2 dice. What is the probability of getting at least one 4?”*

Different from previous questions, this question is one of the examples of independent events since the outcome from throwing a die wouldn’t have any effect on the outcome from throwing the second die.

Let’s say that:

**A** = getting a 4 in the first die

**B** = getting a 4 in the second die

The probability of independent events **A** and **B** both to occur can be defined as:

$$P(A \cap B)=P(A)*P(B)$$

And the probability of getting at least one 4 can be computed with the probability of union of two events:

$$P(A \cup B)=P(A)+P(B)-P(A \cap B)$$

We know that the probability of us getting any specific outcome from throwing a die is  $\frac{1}{6}$ . Thus,

$$P(A \cup B)=61+61-(61*61)=3611$$

Question from [Facebook](#):

*“Three ants are sitting at the three corners of an equilateral triangle. Each ant randomly picks a direction and starts to move along the edge of the triangle. What is the probability that none of the ants collide?”*

Although it’s implicit, this is the case of an independent event. Each ant can randomly pick the direction, either to the left or to the right. The decision of one

ant to go to the left wouldn't affect the decision of the other two ants whether they want to go to the left or right.

Since the decision is random, then the probability of an ant to pick a certain direction is 0.5. The three ants wouldn't collide if all of them go to the left or all of them go to the right.

Hence:

$$P(\text{none of the ants collide}) = (2)(3) + (2)(3) = 41$$

## Permutations and Combinations

Permutations and combinations probably sound similar and we have probably used the two words interchangeably in real life. However, they have a distinct difference in terms of their concept and it is important for us to know how to differentiate between combination and permutation because they have different formulas.

One big difference between permutation and combination is the importance of order. The order is very important in permutation but not in combination. This concept of order will be explained more deeply in the examples of data science interview questions below.

Question from [Kabbage](#):

*“How to find who cheated on essay writing in a group of 200 students?”*

There are different ways on how we can find who's cheating in an exam. One way to do this is by comparing a pair of student exams one-by-one.

If we think about it, comparing the exam of student **A** with student **B** is the same as comparing the exam of student **B** with student **A**. In other words, **A, B = B, A**. The order doesn't matter.

Since the order doesn't matter, then we can use the concept of combination. The general equation of combination is:

$$C(n,k) = P(n,k) / k! = n! / (n-k)!k!$$

where **n** is the total number of items and **k** is the total number of items to be ordered.

Since there are 200 students and there 2 exams that will be compared, then we have:

$$C(200,2) = 200! / (200-2)!2! = 200! / 198!*2! //$$

Question from [IBM](#):

*"From a deck of cards numbered from 1 to 100, we draw two cards at random. What is the probability that a number on one of the cards is exactly double the number on the second card?"*

This question can also be answered with the concept of combination. This is because when we draw two cards from the same deck of cards, the order is not important. This means that if we get a card number 10 in the first draw and number 40 in the second draw, this is the same as getting card number 40 in the first draw and number 10 in the second draw.

Thus, by plugging values that we know from the question into the combination equation we will get:

$$C(100,2) = 100! / (100-2)!2! = 4950 \text{ combinations}$$

which means that we have 4950 combination pairs.

Now out of those 4950 combinations, the number of possibilities that one card is the double of the other card is 50, since we have 100 cards in total. Thus, we can compute the probability as:

$$50/4950 = 0.01 = 1\%$$

Question from [Peak6](#):

*"Three people, and 1st, 2nd and 3rd place at a competition, how many different combinations are there?"*

In this question, the order actually matters because being in the 1st position is not the same as being in the 2nd or 3rd position.

This means that if we have athletes A, B, C and position 1, 2, 3, then the composition of **A, B, C** is not the same as **C, B, A** nor **B, A, C**. Thus, we're dealing with the concept of permutation in this question.

The general equation for permutation problem is:

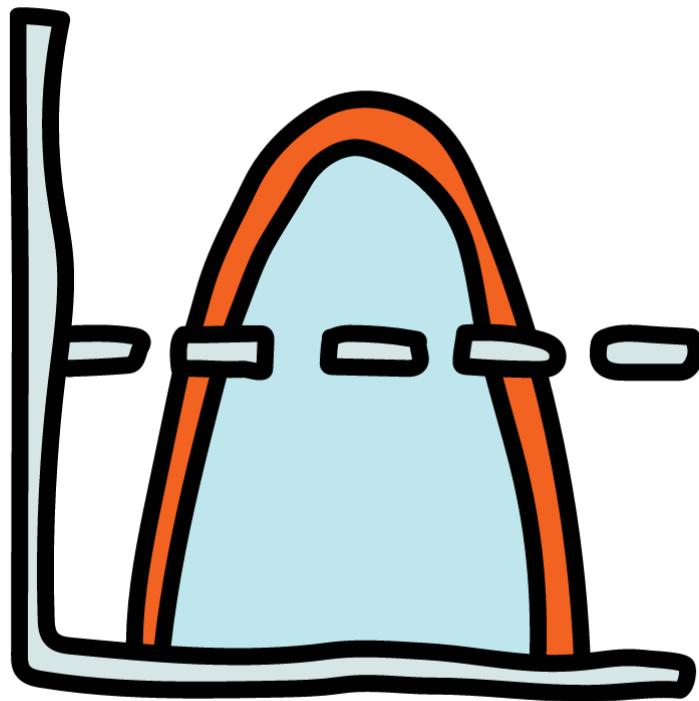
$$P(n,k) = n! / (n! - k!)$$

where  $n$  is the total number of items and  $k$  is the total number of items to be ordered.

In the questions, we have three athletes and three places to be ordered, hence:

$$P(3,3) = 3! / (3! - 3!) = 3! = 3 * 2 * 1 = 6$$

## Probability Distributions



A knowledge about probability distribution is a must before you're going to a data science interview. Question about probability distributions is one, if not, the most popular data science interview question out there.

Below is one interview question that test your general knowledge about probability distributions:

Question from [IBM](#):

*“What is an example of a dataset with a non-Gaussian distribution?”*

We can answer this question by providing an example of data with binomial distribution, such as the frequencies you'll get 500 tails from tossing a coin

1000 times, the frequencies of us getting two 5 from throwing a die 10 times, etc.

The thing is, you can't answer this question if you don't know about probability distributions in the first place. To make things worse, there are a lot of different probability distributions out there. So do we need to know all of the probability distributions?

Of course not.

Binomial, uniform, and Gaussian distributions are the most popular ones in a data science interview among all probability distributions. And if you're really new to probability distribution, you can start with these three before branching out to the other probability distributions.

There are two types of questions related to probability distributions that are commonly asked in a data science interview: either you're asked to compute the probability mass function (PMF) / probability density function (PDF) of a distribution or to compute the expected value of a distribution.

Let's start with binomial distribution.

## Binomial Distribution

Binomial distribution is one of discrete probability distributions and it measures the probability of success of an event in a certain number of trials.

The probability mass function (PMF) of binomial distribution is as follows:

$$P(X=k) = \binom{n}{k} p^k (1-p)^{n-k}$$

where  $n$  is the number of trials and  $k$  is the number of successes. Meanwhile, the expected value of binomial distribution can be computed as follows:

$$E(x) = n * p$$

Below are the examples of data science interview questions from various companies that cover the concept of binomial distribution.

Question from [Verizon Wireless](#):

*“What is the probability of getting one 5 on throwing dice 7 times?”*

This question can be answered by simply plugging in values into the equation of binomial distribution. We can consider that the number of successes is 1 (because we’re looking at one 5) and the number of trials is 7. Meanwhile, the probability of getting a 5 in a single throw is, as we all know,  $\frac{1}{6}$ . Hence:

$$P(X=1) = \binom{7}{1} \left(\frac{1}{6}\right)^1 \left(1 - \frac{1}{6}\right)^{6} = 0.397$$

Question from [Jane Street](#):

*“What's the probability of obtaining 2 tails in 5 coin flips?”*

Same as the previous question, this can be answered by simply plugging in values into the PMF equation of binomial distribution. In this scenario, the number of successes is 2 because we’re looking at obtaining 2 tails and the total number of trials is 5. The probability of getting a tail in each fair coin toss is 0.5. Hence:

$$P(X=2) = \binom{5}{2} \left(\frac{1}{2}\right)^2 \left(1 - \frac{1}{2}\right)^3 = \frac{10}{32}$$

Question from [Lyft](#):

*“A discount coupon is given to  $N$  riders. The probability of using a coupon is  $P$ . What is the probability that one of the coupons will be used?”*

Again, this question can also be answered by plugging in the values into the PMF equation of binomial distribution.

From the question, we can say that the number of successes is 1 (because only one coupon will be used) and the number of items is  $N$ , while the probability of success of a single trial is  $P$ .

Thus,

$$P(X=1) = \binom{N}{1} P * (1-P)^{N-1} = N * P * (1-P)^{N-1}$$



Question from [Lyft](#):

*“A \$5 discount coupon is given to  $N$  riders. The probability of using a coupon is  $P$ . What is the expected cost for the company?”*

Different from the previous question, now we need to compute the expected value of a variable with binomial distribution instead of computing the PMF. We can answer this question by plugging in the values into the equation of expected value of binomial distribution.

From the equation above, we have  $N$  coupons and the probability of using a coupon is  $P$ .

Thus, the expected value would be:

$$E(x)=N*P$$

And the expected cost would be:

$$E(X)*5\$=N*P*5$$

Question from [Facebook](#):

*“We have two options for serving ads within Newsfeed:*

- 1. Out of every 25 stories, one will be an ad*
- 2. Every story has a 4% chance of being an ad.*

*For each option, what is the expected number of ads shown in 100 news stories? If we go with option 2, what is the chance a user will be shown only a single ad in 100 stories?”*

This question tests your knowledge on both expected value and the PMF of binomial distribution.

The first question, which is the expected number of ads shown in 100 news stories would be:

$$E(adsshown)=100*4/100=4$$

Meanwhile, the second question can be answered with the PMF of binomial distribution, where the total number of trials is 100, the total number of

successes is 1 (only a single ad), and every story has 0.04 probability of being an ad.

$$P(X=1) = (1100) * 0.04 * (1 - 0.04)^{99} = 0.07$$



## Uniform Distribution

Uniform distribution can be classified as both discrete and continuous probability distribution, depending on the use case. It measures the probability of an event with  $n$  possible outcomes, where each  $n$  is equally likely to happen. Because of this, it has a flat PMF/PDF.

The common example of a uniform distribution is throwing a die. Our probability of getting any of the sides from a 6-sided die would always be  $\frac{1}{6}$ .

The expected value of a discrete uniform distribution is:

$$E(x) = \frac{1}{2}(a+b)$$

where  $a$  is the minimum possible outcome and  $b$  is the maximum outcome. As an example, when we roll a 6-sided die, the minimum possible outcome would be 1 and the maximum possible outcome is 6.

Below are the examples of data science interview questions that test your knowledge about uniform distribution.

Question from [Jane Street](#):

*“What is the expectation of a roll of a die?”*

We can solve this question easily by plugging in the values into the formula of expected value of a uniform distribution as follows:

$$E(x) = \frac{1}{2}(1+6) = 3.5$$

Question from [Walmart](#):

*“Suppose you roll a die and earn whatever face you get. Now suppose you have a chance to roll a second die. If you roll, you earn whatever face you get but*

*you forfeit earnings from the first round. When should you roll the second time?"*

This question is somewhat an extension from the previous question. As you already know from the previous question, the expected value of a roll of a 6-sided die is:

$$E(x)=1/2(1+6)=3.5$$

To answer this question, we need to think it like this:

If we get more than 3.5 (the expected value of a single roll) in the first roll, then we shouldn't roll the second die and keep the earnings. Meanwhile, if we get less than 3.5, then we should roll the second die.

Question from [PayPal](#):

*"What has the larger expected value: sampling a number between 1 and N from a uniform distribution and multiplying it by itself, or sampling two numbers between 1 and N from a uniform distribution and multiplying them?"*

This question can be interpreted to either one of these two:

**First interpretation: Take one sample, multiply the sample by itself, then compute the expected value after we multiply the sample.**

To answer this question, we need to know the general equation of variance for a variable with uniform distribution:

$$Var(X)=E(X^2)-E(X)^2$$

- For the first case, we sample a number between 1 to  $N$ , let's call this  $X$ . If we multiply this  $X$  by itself, then we have  $X^2$  and its expected value would be  $E(X^2)$ .
- For the second case, we sample two numbers independently between 1 to  $N$ , hence the expected value for both numbers after we multiply them would be  $E(X)E(X) = E(X)^2$ .

If we look at the variance equation above, we know that the value of variance should always be positive. To fulfill this condition, then  $E(X^2)$  has to be larger than  $E(X)^2$ . Hence, the expected value of sampling a number between 1 and  $N$  and multiplying it by itself will always be larger.

**Second interpretation: Take one sample, compute the expected value of that one sample, then multiply that expected value by itself.**

- For the first case, we first sample a number between 1 and  $N$  from a uniform distribution, then we multiply the expected value of that number by itself, hence we have  $E(X)^2$ .
- For the second case, we sample two independent numbers, and multiply their expected value, hence we have  $E(X)E(X) = E(X)^2$ .

Thus, we can conclude that both methods result in similar expected values.

## Gaussian Distribution

Gaussian distribution or normal distribution is a bell-shaped curve that is characterized with two parameters: the mean and the standard deviation.

Interview questions about normal distribution are normally coupled with other themes in the scope of inferential statistics, such as how to infer p-Value, sample size, margin of error, confidence interval, and hypothesis testing.

You can see the example interview questions of any of these in the following section.

There are also at least three big topics in statistics that are commonly asked in a data science interviews, which are:

1. Measure of center and spreads (mean, variance, standard deviation)
2. Inferential statistics
3. Bayes' theorem

Let's discuss the measure of center and spread first.

## Mean, Variance, Standard Deviation

The concept of measure of center (mean, median, mode) and measure of spread (variance, standard deviation) are the very first concepts that you should master before delving deep into statistics.

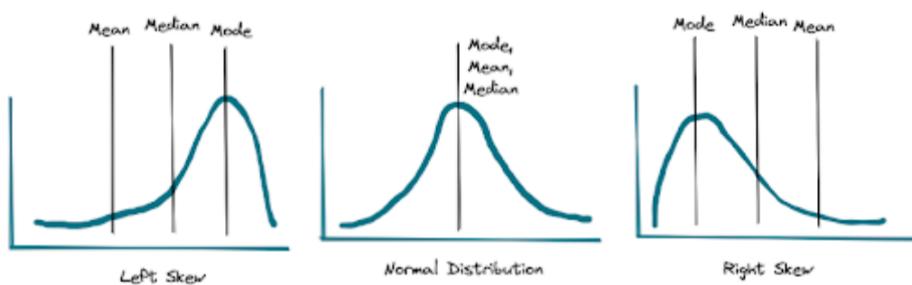
This is why questions about these concepts are very popular in a data science interview. Companies want to know whether you have a basic knowledge of statistics or not. Below is an example that asks about these concepts.

Question from [Facebook](#):

*"In Mexico, if you take the mean and the median age, which one will be higher and why?"*

This question tests your knowledge about the concept of measure of center. To find out which one between mean and median that will be higher, we need to find out the age distribution in Mexico.

According to [Statista](#), Mexico has a right-skewed distribution in terms of its age distribution. If you take a look at the below image, a right-skewed distribution has a higher mean compared to the median.



Thus, the mean age is higher than the median age in Mexico.

Question from [Microsoft](#):

*"What is the definition of the variance?"*

As the concept says, the variance measures the spread of data points of a dataset with respect to its mean value. Below is the general equation of a variance:

$$S^2 = \frac{1}{n-1} \sum (x_i - \bar{x})^2$$

where  $S$  is the variance,  $x$  is the sample,  $\bar{x}$  is the sample mean, and  $n$  is the total number of samples.

## Inferential Statistics

Inferential statistics is a very popular topic in data science interviews. This is because by asking a question within this topic, companies can gauge your thought process when deriving some insights that come from the data.

Inferential statistics come with different steps and terms, such as hypothesis testing, confidence interval, p-Value, margin of error, and sample size.

Let's start with several example questions about p-Value.

### p-Value

Question from [Amazon](#):

*"How would you explain to an engineer how to interpret a p-value?"*

Question from [State Farm](#):

*"What is a p-value? Would your interpretation of p-value change if you had a different (much bigger, 3 mil records for ex.) data set?"*

p-Value stands for probability value in the field of statistics and is normally used during hypothesis testing. p-Value describes how unlikely the data that you just observed given the fact that your null hypothesis is true.

Normally, we set a significance level before hypothesis testing. If the p-Value is below our significance level, then we reject the null hypothesis. Meanwhile if the p-Value is above significance level, then we go with our null hypothesis.

The interpretation of p-Value wouldn't change if your dataset is getting bigger, but bigger dataset means a more robust and more reliable result from our p-Value.

## Confidence Interval, Sample Size, and Margin of Error

The concepts of confidence interval, sample size, and the margin of error normally come along together due to their relationships, as you will see in the following confidence interval equation:

$$CI = \bar{X} \mp Z * \sigma$$

In the equation above,  $X_{\bar{}}$  is the sample mean,  $Z$  is the confidence value,  $\sigma$  is sample standard deviation, and  $n$  is the sample size.

Meanwhile, margin of error can be defined as:

$$\text{Margin of error} = Z \cdot \sigma / \sqrt{n}$$

As you can see above, there is a relationship between confidence interval, margin of error, and sample size value. And this is why the questions between these terms are closely tied together.

Now let's take a look at interview questions that talk about these concepts.

Question from [Google](#):

*"For sample size n, the margin of error is 3. How many more samples do we need to bring the margin of error down to 0.3?"*

This question can be answered by simply plugging in values into the equation of margin of error above.

$$3/0.3 = \sqrt{n}/\sqrt{3}$$

$$100 \cdot n/3 = n/0.3$$

which means that we need 100 times more samples than our initial sample size to bring down the margin of error to 0.3.

Question from [Facebook](#):

*"Let's say the population on Facebook clicks ads with a click-through-rate of P. We select a sample of size N and examine the sample's conversion rate, denoted by  $\hat{P}$ , what is the minimum sample size N such that Probability(ABS( $\hat{P}$  - P) < DELTA ) = 95%? In other words, find the minimum sample size N such that our sample estimate  $\hat{P}$  is within DELTA of the true click-through rate P, with 95% confidence."*

This question tests our knowledge about confidence intervals, margin of error, sample size, and binomial distribution. The conversion rate in the question follows a binomial distribution, which means we need to estimate standard deviation  $\sigma$  with the square-root of variance of binomial distribution.

The general equation for the variance of binomial distribution is:

$$\text{Var}(X) = p(1-p)$$

From the question, we know that we have a 95% confidence interval, which translates to a Z-score equals to 1.96 (see the Z-table to obtain this value). Plugging in this equation into the equation margin of error, we get:

$$\delta=1.96*NP(1-P)$$

$$N>=1.96^2*(\delta)^2P(1-P)$$

Question from [Tesla](#):

*"There are 100 products and 25 of them are bad. What is the confidence interval?"*

Same as the previous question, this question also tests our knowledge about confidence interval, margin of error, sample size, and binomial distribution.

The problem stated in the question follows a binomial distribution, so we need to compute the sample mean from the expected value of binomial distribution and the standard deviation from the variance of binomial distribution.

$$E(X)=100*0.25$$

$$Var(X)=100*0.25*(1-0.25)=18.75$$

Once we compute the mean and the standard deviation from the binomial distribution formula, then we can just plug those values into the equation of confidence intervals to find out the answer.

$$CI=25\pm 1.96*18.75$$

## Hypothesis Testing

Interview questions about hypothesis testing are normally presented as an example use case. Companies will give you a specific use case about their products and they will ask you about how you would know if one product performs better in the market compared to the other product. Below is the example of that:

Question from [Facebook](#):

*"We have a product that is getting used differently by two different groups.*

*1. What will be your hypothesis?*

*2. How would you go about testing your hypothesis?"*

This question tests our knowledge about different steps that we should take to conduct a hypothesis testing.

Below is the step-by-step example on how we should conduct a hypothesis testing:

- Formulate our null hypothesis and the alternative hypothesis
- Choose the significance level. The significance level can vary depending on our use case. However, we can pick the default value, which is 0.05.
- Compute the sample mean and sample standard error from our data.
- Compute the t-statistics which correspond to your use case, whether it is paired t-test, one sample t-test for population mean, two sample t-test, ANOVA, or Chi-Squared.
- If the p-Value from the resulting test is below our significance value, then we reject our null hypothesis in favour of the alternative hypothesis. Meanwhile, if the resulting p-Value is above the significance value, then we take the null hypothesis.

Question from [Amazon](#):

*“In an A/B test, how can you check if assignment to the various buckets was truly random?”*

If the buckets were truly assigned at random, then in terms of statistics we wouldn't notice any significant differences between variable samples in each bucket. But how do we know whether the sample differences between buckets is significant or not?

We can use a statistical test to measure this.

If the variables that we observe are continuous variables and there is only one treatment, we can use the two-sample t-test. Meanwhile, if there are multiple treatments, then we can use ANOVA.

After conducting a statistical test, we will get a p-Value and we can use this p-Value to conclude whether there is a significant difference between buckets.

## Bayes' Theorem

Bayes' theorem is also commonly asked in data science interview questions. This statistical approach has a different approach compared to frequentist statistics that we saw in the Inferential Statistics section above. In fact, some companies might ask you what is the difference between Bayesian and frequentist statistics as below:

Question from [Yelp](#):

*“What is the difference between Bayesian vs frequentist statistics?”*

The main difference between these two is:

- In frequentist statistics, the inference is interpreted as long run frequencies. This means that if we repeat the trial for an infinite amount of times, we want to measure how many times that the mean of each trial is within 95% of the population's confidence interval.
- In Bayesian statistics, the procedure is interpreted as a subjective belief. In the end, the goal is to update your belief based on the evidence of the data.

Bayesian inference has the same analogy as how we as a human make an inference. In the beginning, we always have a certain degree of belief in how likely something is going to happen. Then, as we see more and more evidence, our belief will be updated.

There are four fundamental terms in Bayes' rule: prior, posterior, likelihood, and marginal as you can see from the equation below.

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

*Likelihood*      *Prior*  
↑                    ↑  
*Posterior*          *Marginal*  
↓

Below are some examples of data science interview questions that test our knowledge about Bayes' theorem.

Question from [Facebook](#):

*"You're about to get on a plane to Seattle. You want to know if you should bring an umbrella. You call 3 random friends of yours who live there and ask each independently if it's raining. Each of your friends has a 2/3 chance of telling you the truth and a 1/3 chance of messing with you by lying. All 3 friends tell you that "Yes" it is raining. What is the probability that it's actually raining in Seattle?"*

To answer this question, you need to make an assumption about the probability of rain in Seattle. Let's say it's 0.5.

Each of our friends has a  $\frac{2}{3}$  chance of telling the truth, so the probability of rain in Seattle given that our friends say that it will be raining in Seattle would be  $\frac{2}{3}$ . Likewise, the probability of not raining given that our friends say that it won't be raining in Seattle is also  $\frac{2}{3}$ .

Based on this, let's define an event as below:

- $A$  = raining in Seattle
- $A'$  = not raining in Seattle
- $X_i$  = random variable with Bernoulli distribution, where the value of this variable represents the answer of our friends: raining (1) or not (0)

Thus, we can approximate the probability that it will rain in Seattle given that our friends say that it will rain with Bayes' theorem.

$$P(A|X_1=1 \cap X_2=1 \cap X_3=1)$$

$$(1/2 * 2/3)^3 / (2/3 * 1/2)^3 + (1 - 1/2 * 2/3)^3 = 0.888$$

Question from [Facebook](#):

*"You randomly draw a coin from 100 coins - 1 unfair coin (head-head), 99 fair coins (head-tail) and roll it 10 times. If the result is 10 heads, what's the probability that the coin is unfair?"*

To answer this question, we need to define an event:

$A$  = The coin is unfair

$A'$  = The coin is fair

$B$  = The result of rolling random coins 10 times is 10 heads.

After defining the above events, we can plug the values into the Bayes' equation as follows:

$$P(B|A)*P(A)/P(B|A')*P(A')P(B|A)*P(A) \\ =1*0.011*0.01+(12)10*0.99=0.9118=1*0.01+(21)10*0.991*0.01=0.9118$$

Question from [Zenefits](#):

*"There are 30 red marbles and 10 black marbles in Urn #1. You have 20 red and 20 Black marbles in Urn #2. Randomly you pull a marble from the random urn and find that it is red. What is the probability that it was pulled from Urn #1?"*

We can answer this question with the same approach as the previous question by defining an event first:

- A** = The marble was pulled from Urn #1
- A'** = The marble was pulled from Urn #2
- B** = The marble is red

Now after we define the above event, we can plug the values into Bayes' theorem equation as follows:

$$P(B|A)*P(A)/P(B|A')*P(A')P(B|A)*P(A) \\ =3040*123040*12+2040*12=35=4030*21+4020*214030*21=53$$

Question from [Lyft](#):

*"A discount coupon is given to 2 riders. The probability of using a coupon is P. Given that at least one of them uses a coupon, what is the probability that both riders use the coupons?"*

This question tests your knowledge about two concepts: Bayes' theorem and binomial distribution.

With the PMF of binomial distribution, the probability of exactly one rider uses the coupon can be computed as follows.

$$P(X=1)=(12)*P1*(1-P)=2*P*(1-P)$$



.

The probability that both of them use the coupon can be computed as well with the PMF of binomial distribution.

$$P(X=2) = \binom{2}{2} * P^2 * (1-P)^{2-2} = P^2$$

$$\binom{2}{2}$$

The probability that at least one coupon being used is the example of a mutual exclusive event, which means:

$$P(X=1 \cup X=2) = P(X=1) + P(X=2)$$

$$P(X=1 \cup X=2) = 2 * P * (1-P) + P^2$$

Next, as usual, we need to define an event to make it easier for us to understand what each term in Bayes' theorem equation represents.

**A** = at least one of the riders uses the coupon

**B** = both riders use the coupon

Now we can plug the values into Bayes' theorem equation as follows:

$$P(B|A) = P(A)P(A|B)*P(B)$$

$$= 1 * P^2 / 2 * P * (1-P) + P^2 = P/2 - P$$

1. *What's your background in data science?*
2. *Would you explain your educational history?*
3. *What's a p-value?*
4. *Can you define the term random variables?*
5. *What's the difference between discrete and continuous variables?*
6. *What are permutations?*
7. *Can you define expected value?*
8. *Define variance and explain its importance.*
9. *Can you explain combinations as they're related to probability?*
10. *Explain when an event A can be independent of itself?*
11. *How might you explain a confidence interval to an audience with no technical or data science background?*
12. *How many rolls would it take to see all six sides of a fair die?*
13. *A co-worker tells you they have two children, and one is a boy. What's the probability that the other is also a boy?*

14. If an insect has a lifespan of 13 to 15 days, what's the probability it might die in exactly 14 days?
15. How many coin flips would you expect to perform to get heads twice in a row?
- 1.
- How can you generate fair odds using a coin with an unknown bias toward heads or tails?
2. How many cards would you expect to draw from a standard deck of cards before seeing your first queen?
3. If there are 25 cards with five different colors and each card has a number from one to five, what's the probability they aren't the same color if you pick two cards at random?
4. An online dating service allows users to choose six out of 30 words to describe their personality. The dating service creates a match based on five of the same words. If client A and client B both choose random personality words, what's the probability they can find a match?
5. Given a mother who's very tall, on average, what could you expect the height of her daughter to be? Shorter, equal or taller?
6. How many possible ways can you split 12 people into three teams of four?
7. If three friends in London told you it's raining, and there is a  $1/3$  probability that each person is lying, what's the probability that it's raining in London? The probability of rain on any day in London is  $0.25$ .
8. If you picked randomly from a fair coin (one side heads, one side tails) or an unfair coin (both sides tails), flipped it five times and got tails five times, what's the chance you picked the unfair coin?
9. A jar holds five lollipops: three red and two yellow. If you remove and replace three lollipops after every draw, find the probability of drawing the same color lollipop twice.
10. If there's a  $15\%$  probability that you might see at least one airplane in a five-minute interval, what is the probability that you might see at least one airplane in a period of half an hour?

## 1. Explain probability distribution

Understanding probability distributions is key to understanding predictive and inferential statistics. Interviewers might ask this question to see how well you understand the basics of higher-level statistical analysis. Answer with a clear definition that's simple enough for your interviewer to understand yet that contains the necessary language to demonstrate your statistical knowledge.

**Example:** "Probability distribution describes the likelihood that a random variable is equal to a certain value or set of values during a single observation. Property distribution explains how every method has a range of possible values from a single draw. They form the basis for all potential values in any given process."

## 2. How would you generate a random number between one and seven with only a single die?

Your interviewer may ask you questions like this to test your practical probability knowledge. In answering this question, it might be beneficial to write notes on a notepad or whiteboard to help explain your answer. In your answer, explain the steps you would take to generate the random number concisely.

**Example:** "First, I would roll the die three times, with each throw setting the nth bit of the result. For each roll, I would notice if the value is one to three and record a zero or else a one. My result would be between zero (000) and seven (111) evenly spread across the three independent throws. If I repeat the throws when the result is a zero, then the process stops on evenly spread values."

## 3. If you draw from a normal distribution with known values of parameters, how do you generate draws in a uniform distribution?

This is a common question that assesses how well you know the concepts of uniform distribution and normal distribution. Answer it by explaining your process for generating draws clearly. If there are multiple ways to do this, you can also state an alternative to the first method.

**Example:** "I'd generate draws in a uniform distribution by entering the value from the normal distribution cumulative distribution function for the same random variable. Another way to explain this would be to plug in the value of the parameters to the cumulative distribution function of the same random variable. This function is the probability of a random variable taking a value that's less than or equal to X."

## 4. If 75 customers fall randomly into three equal-sized databases, all partitions are equally likely. Bob and Ben are two randomly selected customers. What is the probability that Bob and Ben are in the same customer database?

This question helps an interviewer determine whether you can apply your statistical knowledge to find an answer to a realistic business scenario. For

questions like this, it may be necessary for you to show calculations on a whiteboard or notepad, or even use the note function on your phone. This can help your interviewer better understand your thought process.

**Example:** *"I'd assign a different number from one to 75 for each student with numbers one through 25 in group one, numbers 26 to 50 in group two and numbers 51 to 75 in group three. Next, I would assign a random number to Bob and Ben. Once Bob has a number, there are 74 random numbers remaining with 24 that will result in him being part of the same group as Ben. The probability is 24/79."*

## **5. Can you explain the Bayesian approach to probability?**

Bayesian statistics is a method for applying probability to a statistical problem. Employers may ask this question in an interview because the Bayesian approach has a number of practical applications in quantitative finance and data science. In your answer, convey your understanding of this approach and make it clear that you also understand the frequentist approach and how it differs from the Bayesian method.

**Example:** *"The Bayesian approach defines probability as the measure of believability one has about how a particular event occurs. It uses mathematical tools to help you update beliefs about random events once you've seen new evidence about the event. You can use Bayesian statistics to create different beliefs after discovering new evidence. It differs from frequentist statistics that rely only on data from repeated trials."*

## **Q1. What is Statistics?**

**Ans.** Statistics is the science concerned with developing and studying methods for collecting and analyzing, interpreting, and presenting empirical data (information that comes from research).

## **Q2. What are the types of data?**

**Ans. Categorical** – Describe category or groups

Example – Car Brands( Audi, BMW, TATA)

**Numerical** – Represent numbers

These are of two types:

- Discrete

Example – Grade, Number of Objects

- Continuous

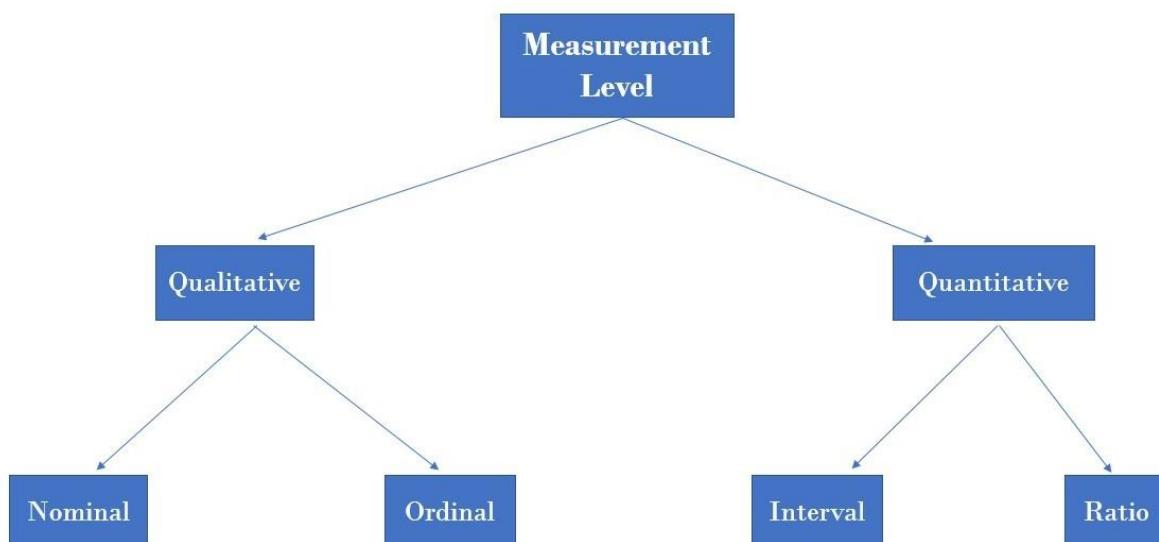
Example – Weight, Height, Area

**Q3.** Difference between Population and Sample?

**Ans.** The Population is a collection of all items of interest while the Sample is the subset of the population. The numbers obtained from the population are called Parameters while the numbers obtained from the sample are called Statistics. Sample data are used to make conclusions on Population data.

**Q4.** What are the different types of variables or measurement levels?

**Ans.**



**Q5.** Difference between Descriptive and Inferential Statistics?

**Descriptive**

**Inferential**

Summarize the characteristics(properties) of the data.	Used to conclude the population.
It helps to organize, analyze, and present data in a meaningful way.	It allows comparing data and making predictions through hypotheses.
Done using charts, tables, and graphs.	Achieved through probability.

## Q6. What are the Measures of Central Tendency?

The measure of central tendency is a single value that describes(represents) the central position within the dataset. Three most common measures of central tendency are Mean, Median, and Mode.

### Mean:

Mean(**Arithmetic Mean**) is defined as the sum of all values divided by the number of values. If there are  $n$  values given ( $x_1, x_2, x_3, \dots, x_n$ ) then,

$$\text{Mean} (\bar{x}) = \frac{x_1 + x_2 + x_3 + \dots + x_n}{n} = \frac{\sum_{i=1}^n x_i}{n}$$

### Median:

Median is the exact middle value when the data is ordered(i.e. arranged either in ascending or descending order ). If there are  $n$  values given ( $x_1, x_2, x_3, \dots, x_n$ ) then,

### Case – I: if $n$ is odd:

$$\text{Median} = \left(\frac{n+1}{2}\right)^{\text{th}} \text{term}$$

**Case – II:** if  $n$  is even:

$$\text{Median} = \frac{\left(\frac{n}{2}\right)^{\text{th}} + \left(\frac{n}{2} + 1\right)^{\text{th}}}{2}$$

i.e. mean of two middle values

### Mode:

Mode is the most frequent value in the dataset. It may or may not be unique. i.e. in the dataset, more than one value can be the mode.

Q7. Which is the best measure of central tendency – Mean, Median, Mode?

- If the data is symmetrically distributed then

### Mean = Median = Mode

- If the distribution is Skewed, then Median is the best measure of central tendency
- Mean is most sensitive for skewed data.
- Mode is the best measure for all levels of measurement, but more meaningful for Qualitative Data
- Variables and the corresponding best measures:

<b>Types of Variable</b>	<b>Best Measurement</b>
<b>Nominal</b>	<ul style="list-style-type: none"> <li>• Mode</li> </ul>
<b>Ordinal</b>	<ul style="list-style-type: none"> <li>• Mode</li> <li>• Median</li> </ul>
<b>Interval/Ratio(Skew)</b>	<ul style="list-style-type: none"> <li>• Median</li> </ul>
<b>Interval/Ration(Not Skew)</b>	<ul style="list-style-type: none"> <li>• Mean</li> </ul>

#### Q8. What are the Measures of Dispersion?

Dispersion or variability describes how items are distributed from each other and the centre of a distribution.

The measure of dispersion is a statistical method that helps to know how the data points are spread in the dataset.

There are 4 methods to measure the dispersion of the data:

- Range
- Interquartile Range
- Variance
- Standard Deviation

#### Q9. Which measure of dispersion is best?

Standard Deviation is considered the best measure of dispersion as

- Help to make a comparison between the distribution of two or more different datasets
- Based on all values
- Capable of further algebraic treatment

## Q10. What is the Central Limit Theorem?

Central limit theorem states that, if you have a population mean ( $\mu$ ) and standard deviation ( $\sigma$ ) and take large random samples from the population with replacement.

Then the distribution of the sample means will be approximately normally distributed regardless of whether the population is normal or skewed.

Provided that the sample size is sufficiently large ( $n > 30$ ).

## Q11. What is the difference between Covariance and Correlation?

### Covariance

- Signifies the direction of the linear relationship between two variables
- In simple terms, It is a measure of variance between two variables
- It can take any value from positive infinity to negative infinity

### Correlation

- It measures the relationship between two variables, as well as the strength between these two variables.
- It can take any value from -1 to 1

## Q12. What are the different types of Correlation?

There are mainly three types of correlation:

### • Pearson

- Normalized measurement of covariance
- Assumes both the variables are normally distributed
- Measure linear relationship but fail to measure the non-linear relationship between variables

### • Spearman Rank

- It is a non-parametric measure
- Measures both linear and non-linear relationship between two variables

### •Kendall Rank

- Non-parametric measure for calculating the rank of the correlation coefficient
- Measures both linear and non-linear relationship between two variables

Q13. What is the difference between Probability and Likelihood?

Probability attaches to possible results (chances) while Likelihood attaches to the hypothesis.

Let's understand the difference by an example of cricket,

Problem: Captain have to decide to bat first

**Probability:** Only two possibilities

- Choose to Bat
- Doesn't choose to Bat
- $P(\text{choose to bat}) = P(\text{doesn't choose to bat}) = \frac{1}{2} = 0.5$

**Likelihood:** Choosing to bat first will depend on

- Weather Conditions ( Rainfall, wind speed)
- Due on Pitch
- Humidity

Q14. What are the different types of Probability Distribution used in Data Science?

A probability distribution is a statistical function that describes all the possible values and likelihoods that a random variable can take within a given range. Probability distribution depends on various factors like maximum, minimum, mean, standard deviation, skewness, and kurtosis.

The six, most common probability distributions are:

- Normal Distribution
- Poisson Distribution

- [Binomial Distribution](#)
- Uniform Distribution
- Exponential Distribution
- Bernoulli Distribution

## Q15. What is Normal Distribution?

Normal Distribution is a probability distribution that is symmetric about the mean. It is also known as Gaussian Distribution. The distribution appears as a Bell-shaped curve which means the mean is the most frequent data in the given data set.

In Normal Distribution:

- Mean = Median = Mode
- Total area under the curve is 1.
- The probability distribution function(PDF) of a random variable  $x$  of a Normal Distribution is given by:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2},$$

where

$\sigma$ : Standard Deviation

$\mu$ : Mean

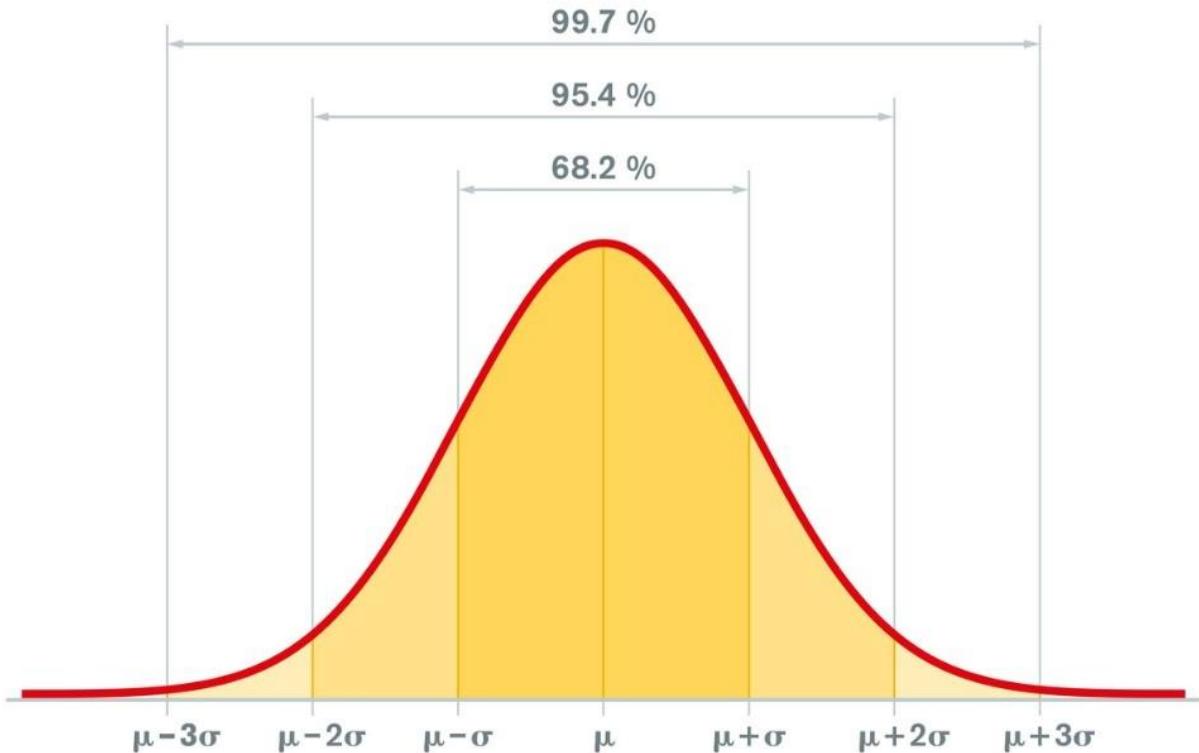
$x$ : random variable

## Q16. What is the empirical rule?

Empirical Rule is often called the **68 – 95 – 99.7** rule or **Three Sigma Rule**. It states that on a Normal Distribution:

- 68% of the data will be within one Standard Deviation of the Mean

- 95% of the data will be within two Standard Deviations of the Mean
- 99.7 of the data will be within three Standard Deviations of the Mean



### Q17. What is Skewness?

It is a measure of lack of symmetry i.e. it measures the deviation of the given distribution of a random variable from a symmetric distribution (like normal Distribution).

There are two types of skewness:

- Positive Skewness
- Negative Skewness

### Q18. What are the different measures of Skewness?

There are different ways to measure the skewness like

- Pearson Mode
- Pearson Median
- Momental

- Kelly's Measure
- Bowley

But we mainly use the first two, Pearson mode and Pearson median skewness.

### Q19. What is Conditional Probability?

Let there be two events A and B of any random experiment,

then the probability of occurrence of event A, such that event B has already occurred is known as Conditional Probability.

$$P(A|B) = \frac{P(A \cap B)}{P(B)},$$

*where,*

$P(A \cap B)$ : Probability that both the events A and B occurs

$P(B)$ : Probability of event B

### Q20. What is Bayes' Theorem?

Bayes' theorem is an extension of Conditional Probability.

It includes two conditional probabilities.

It gives the relation between conditional probability and its reverse form.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

*Where,*

$P(A)$ : Marginal Probability of A

$P(B)$ : Marginal Probability of B

$P(A|B)$ : Conditional Probability of A given B

$P(B|A)$ : Conditional Probability of B given A

## Q21. What is Regression Analysis?

It is a statistical method to model the relationship between a dependent (target) variable and independent (one or more) variables.

It gives a clear understanding of factors that are affecting the target variable in building machine learning models.

These models are used to predict the continuous data.

Example: Predicting Rainfall depends on humidity, temperature, direction and speed of the wind.

## Q22. What are the different types of Regression?

There are mainly 5 types of regression:

### • Linear

- Gives the linear relationship between dependent and independent variables
- Example: House Price Prediction
  - Price of the house increases as the size of the house increases

### • Polynomial

- As linear regression works only with the linear data but for non-linear data (data points are in the form of a curve) we will use polynomial regression.

### • Logistic

- Used when we have to compute the probability of mutually exclusive occurrence such as True/False, Yes/No etc
- Use when the dependent variable is discrete

### • Ridge

- It is an extension of Linear regression that is used to minimize the loss.
- Ridge solves the problem of multicollinearity through shrinking parameter
- Which shrinks the value of coefficient but not to zero.
- It is also known as L2 regression

### • Lasso

- Lasso stands for Least Absolute Shrinkage and Selection Operator.
- It also penalizes the absolute size of the regression coefficients, similar to Ridge
- Reduces the variability and improves the accuracy of the linear regression model.
- It is also known as L1 regression

### Q23. What is Sampling?

It is a process of selecting a group of observations from the population, to study the characteristics of the data to make conclusions about the population.

Example: Covaxin (a covid-19 vaccine) is tested over thousand of males and females before giving it to all the people of the country.

### Q24. What is a Sampling Error and how it can be reduced?

Errors which occur during the sampling process are known as Sampling Errors

$$\text{Sampling Error} = z \times \frac{\sigma}{\sqrt{n}}$$

Where,

*z: z – score value based on confidence interval (approx  $\approx 1.96$ )*

*$\sigma$ : population standard deviation*

*n: sample size*

They can be reduced by:

- Increasing the sample size
- Classifying the population into different groups

### Q25. What is Resampling and what are the common methods of resampling?

Resampling is the method that consists of drawing repeatedly drawing samples from the population.

It involves the selection of randomized cases with replacements from samples.

There are two types of resampling methods:

- K-fold cross-validation
- Bootstrapping

Q26. What is an outlier in any dataset?

An outlier is a value in the data set that is extremely distinct from most of the other values.

Example:

Let there are 5 children having weights of 30 kg, 35 kg, 40kg, 50 kg and 300 kg.

Then the student's weight having 300 kg is an outlier.

An outlier in the data is due to

- Variability in the data
- Experimental Error
- Heavy skewness in data
- Missing values

Q27. What are the different methods to detect outliers in a dataset?

There are mainly 3 ways to detect outliers in a dataset:

• **Box-Plot**

- Data points are divided into 4 different quartiles.
- Box-plot marks Maximum, Minimum, lower quartile (Q1), median (Q2) and upper quartile (Q3).
- Points outside the whisker are Outliers.

• **Inter Quartile Range**

- Arrange the data orderly (ascending)
- Compute  $IQR = Q3 - Q1$
- Calculate bound (upper and lower)  $1.5 IQR$
- Any point outside the upper and lower bound are the outlier.

• **Z-score**

In a normal distribution, any data point whose z-score is outside the 3rd standard deviation is an outlier.

## Q28. What is Cost Function?

The cost function measures the performance of machine learning models.

It quantifies the error between the actual and predicted value of the observation data.

In linear regression, there are many evaluation metrics (mean absolute error, mean squared error, R squared, RMSLE, RMSE etc) to quantify the error, but we generally use Mean Squared Error:

$$\text{Mean Square Error} = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}$$

where,

$y_i$ :  $i^{th}$  actual value

$\hat{y}_i$ : corresponding predicted value

$n$ : sample size

This Mean squared function is also referred to as Cost Function.

Note: Depending upon the evaluation metrics, cost functions are different.

## Q29. What is Gradient Descent?

Gradient Descent is an optimisation algorithm used to find the value of the parameters of a function that minimizes the cost function.

In the gradient descent, we calculate the next point using the gradient of the cost function at the current position.

The process is given by:

$$a_{n+1} = a_n - \alpha \nabla f(a_n),$$

where

$a_n$ : current position

$a_{n+1}$ : next iterative position

$\alpha$ : learning rate

$\nabla f(a_n)$ : gradient at the current position

### Q30. What is Hypothesis Testing?

Hypothesis testing is a form of statistical inference that uses data from a sample to draw conclusions about a population parameter or a population probability distribution.

There are 3 steps in Hypothesis Testing:

- State Null and Alternate Hypothesis
- Perform Statistical Test
- Accept or reject the Null Hypothesis

### Q31. What is the Null and Alternate Hypothesis?

A null and alternate hypothesis is used in statistical hypothesis testing.

#### Null Hypothesis

- It states that the population parameter is equal to the assumed value
- It is an initial claim based on previous analysis or experience

#### Alternate Hypothesis

- It states that population parameters are equal or different to the assumed value
- It is what you might believe to be true or want to prove true

### Q32. What are a p-value and its role in Hypothesis Testing?

P-value is the probability that a random chance generated the data or something else that is equal or rare.

P-values are used in hypothesis testing to decide whether to reject the null hypothesis or not.

- $p\text{-value} < \alpha$  – value

Means results are not in favor of the null hypothesis, reject the null hypothesis

- $p\text{-value} > \alpha$  – value

Means results are in favor of the null hypothesis, accept the null hypothesis.

### Q33. What Chi-square test?

A statistical method is used to find the difference or correlation between the observed and expected categorical variables in the dataset.

Example: A food delivery company wants to find the relationship between gender, location and food choices of people in India.

It is used to determine whether the difference between 2 categorical variables is:

- Due to chance or
- Due to relationship

### Q34. What is a t-test?

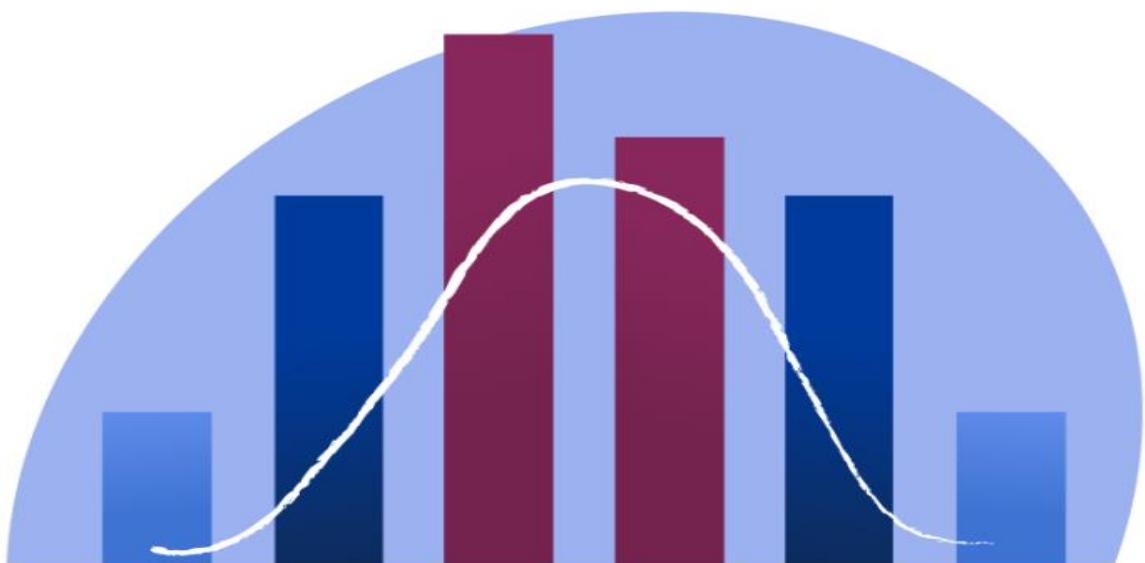
Statistical method for the comparison of the mean of the two groups of the normally distributed sample(s).

It is used when:

- Population parameter (mean and standard deviation) is not known
- Sample size (number of observations)  $< 30$

### Q35. What is the ANOVA test?

Analysis of Variance (ANOVA) is a statistical formula used to compare variances across the means (or average) of different groups. A range of scenarios uses it to determine if there is any difference between the means of different groups.



A bar graph representation of data with normal distribution

### 1. What is the Central Limit Theorem?

[Central Limit Theorem](#) is the cornerstone of statistics. It states that the distribution of a sample from a population comprising a large sample size will have its mean normally distributed. In other words, it will not have any effect on the original population distribution.

Central Limit Theorem is widely used in the calculation of confidence intervals and hypothesis testing. Here is an example – We want to calculate the average height of people in the world, and we take some samples from the general population, which serves as the data set. Since it is hard or impossible to obtain data regarding the height of every person in the world, we will simply calculate the mean of our sample.

By multiplying it several times, we will obtain the mean and their frequencies which we can plot on the graph and create a normal distribution. It will form a bell-shaped curve that will closely resemble the original data set.

## **2. What is the assumption of normality?**

The assumption of normality dictates that the mean distribution across samples is normal. This is true across independent samples as well.

## **3. Describe Hypothesis Testing. How is the statistical significance of an insight assessed?**

[Hypothesis Testing](#) in statistics is used to see if a certain experiment yields meaningful results. It essentially helps to assess the statistical significance of insight by determining the odds of the results occurring by chance. The first thing is to know the null hypothesis and then state it. Then the p-value is calculated, and if the null hypothesis is true, other values are also determined. The alpha value denotes the significance and is adjusted accordingly.

If the p-value is less than alpha, the null hypothesis is rejected, but if it is greater than alpha, the null hypothesis is accepted. The rejection of the null hypothesis indicates that the results obtained are statistically significant.

## **4. What are observational and experimental data in statistics?**

Observational data is derived from the observation of certain variables from observational studies. The variables are observed to determine any correlation between them.

Experimental data is derived from those experimental studies where certain variables are kept constant to determine any discrepancy or causality.

## **5. What is an outlier?**

Outliers can be defined as the data points within a data set that varies largely in comparison to other observations. Depending on its cause, an outlier can decrease the accuracy as well as the efficiency of a model. Therefore, it is crucial to remove them from the data set.

## **6. How to screen for outliers in a data set?**

There are many ways to screen and identify potential outliers in a data set. Two key methods are described below –

- Standard deviation/z-score – Z-score or standard score can be obtained in a normal distribution by calculating the size of one standard deviation and multiplying it by 3. The data points outside the range are then identified. The Z-score is measured from the mean. If the z-score is positive, it means the data point is above average.

If the z-score is negative, the data point is below average.

If the z-score is close to zero, the data point is close to average.

If the z-score is above or below 3, it is an outlier and the data point is considered unusual.

The formula for calculating a z-score is –

$$z = \text{data point} - \text{mean} / \text{standard deviation} \text{ OR } z = x - \mu / \sigma$$

- Interquartile range (IQR) – IQR, also called midspread, is a method to identify outliers and can be described as the range of values that occur throughout the length of the middle of 50% of a data set. It is simply the difference between two extreme data points within the observation.

$$\text{IQR} = Q_3 - Q_1$$

Other methods to screen outliers include Isolation Forests, Robust Random Cut Forests, and DBScan clustering.

## 7. What is the meaning of an inlier?

An Inliner is a data point within a data set that lies at the same level as the others. It is usually an error and is removed to improve the model accuracy. Unlike outliers, inlier is hard to find and often requires external data for accurate identification.

## 8. What is the meaning of six sigma in statistics?

Six sigma in statistics is a quality control method to produce an error or defect-free data set. Standard deviation is known as Sigma or  $\sigma$ . The more the standard

deviation, the less likely that process performs with accuracy and causes a defect. If a process outcome is 99.99966% error-free, it is considered six sigma. A six sigma model works better than  $1\sigma$ ,  $2\sigma$ ,  $3\sigma$ ,  $4\sigma$ ,  $5\sigma$  processes and is reliable enough to produce defect-free work.

## **9. What is the meaning of KPI in statistics?**

KPI is an acronym for a key performance indicator. It can be defined as a quantifiable measure to understand whether the goal is being achieved or not. KPI is a reliable metric to measure the performance level of an organization or individual with respect to the objectives. An example of KPI in an organization is the expense ratio.

## **10. What is the Pareto principle?**

Also known as the 80/20 rule, the Pareto principle states that 80% of the effects or results in an experiment are obtained from 20% of the causes. A simple example is – 20% of sales come from 80% of customers.

## **11. What is the Law of Large Numbers in statistics?**

According to the law of large numbers, an increase in the number of trials in an experiment will result in a positive and proportional increase in the results coming closer to the expected value. As an example, let us check the probability of rolling a six-sided dice three times. The expected value obtained is far from the average value. And if we roll a dice a large number of times, we will obtain the average result closer to the expected value (which is 3.5 in this case).

## **12. What are some of the properties of a normal distribution?**

Also known as Gaussian distribution, Normal distribution refers to the data which is symmetric to the mean, and data far from the mean is less frequent in occurrence. It appears as a bell-shaped curve in graphical form, which is symmetrical along the axes.

The properties of a normal distribution are –

- Symmetrical – The shape changes with that of parameter values
- Unimodal – Has only one mode.
- Mean – the measure of central tendency

- Central tendency – the mean, median, and mode lie at the centre, which means that they are all equal, and the curve is perfectly symmetrical at the midpoint.

### **13. How would you describe a ‘p-value’?**

P-value in statistics is calculated during hypothesis testing, and it is a number that indicates the likelihood of data occurring by a random chance. If a p-value is 0.5 and is less than alpha, we can conclude that there is a probability of 5% that the experiment results occurred by chance, or you can say, 5% of the time, we can observe these results by chance.

### **14. How can you calculate the p-value using MS Excel?**

The formula used in MS Excel to calculate p-value is –

`=tdist(x,deg_freedom,tails)`

The p-value is expressed in decimals in Excel. Here are the steps to calculate it –

- Find the Data tab
- On the Analysis tab, click on the data analysis icon
- Select Descriptive Statistics and then click OK
- Select the relevant column
- Input the confidence level and other variables

### **15. What are the types of biases that you can encounter while sampling?**

Sampling bias occurs when you lack the fair representation of data samples during an investigation or a survey. The six main types of biases that one can encounter while sampling are –

- Undercoverage bias
- Observer Bias
- Survivorship bias
- Self-Selection/Voluntary Response Bias
- Recall Bias
- Exclusion Bias

### **16. What is cherry-picking, P-hacking, and significance chasing?**

Cherry-picking can be defined as the practice in statistics where only that information is selected which supports a certain claim and ignores any other claim that refutes the desired conclusion.

P-hacking refers to a technique in which data collection or analysis is manipulated until significant patterns can be found who have no underlying effect whatsoever.

Significance chasing is also known by the names of Data Dredging, Data Fishing, or Data Snooping. It refers to the reporting of insignificant results as if they are almost significant.

## **17. What is the difference between type I vs type II errors?**

A type 1 error occurs when the null hypothesis is rejected even if it is true. It is also known as false positive.

A type 2 error occurs when the null hypothesis fails to get rejected, even if it is false. It is also known as a false negative.

## **18. What is a statistical interaction?**

A statistical interaction refers to the phenomenon which occurs when the influence of an input variable impacts the output variable. A real-life example includes the interaction of adding sugar to the stirring of tea. Neither of the two variables has an impact on sweetness, but it is the combination of these two variables that do.

## **19. Give an example of a data set with a non-Gaussian distribution?**

A non-Gaussian distribution is a common occurrence in many processes in statistics. This happens when the data naturally follows a non-normal distribution with data clumped on one side or the other on a graph. For example, the growth of bacteria follows a non-Gaussian or exponential distribution naturally and Weibull distribution.

## **20. What is the Binomial Distribution Formula?**

The binomial distribution formula is:

$$b(x; n, P) = nCx * Px * (1 - P)^{n-x}$$

Where:

b = binomial probability

x = total number of “successes” (pass or fail, heads or tails, etc.)

P = probability of success on an individual trial

n = number of trials

## **21. What are the criteria that Binomial distributions must meet?**

Here are the three main criteria that Binomial distributions must meet –

- The number of observation trials must be fixed. It means that one can only find the probability of something when done only a certain number of times.
- Each trial needs to be independent. It means that none of the trials should impact the probability of other trials.
- The probability of success remains the same across all trials.

## **22. What is linear regression?**

In statistics, linear regression is an approach that models the relationship between one or more explanatory variables and one outcome variable. For example, linear regression can be used to quantify or model the relationship between various predictor variables such as age, gender, genetics, and diet on height, outcome variables.

## **23. What are the assumptions required for linear regression?**

Four major assumptions for linear regression are as under –

- There's a linear relationship between the predictor (independent) variables and the outcome (dependent) variable. It means that the relationship between X and the mean of Y is linear.
- The errors are normally distributed with no correlation between them. This process is known as Autocorrelation.
- There is an absence of correlation between predictor variables. This phenomenon is called multicollinearity.
- The variation in the outcome or response variable is the same for all values of independent or predictor variables. This phenomenon of assumption of equal variance is known as homoscedasticity.

## **24. What are some of the low and high-bias Machine Learning algorithms?**

Some of the widely used low and high-bias Machine Learning algorithms are –

Low bias -Decision trees, Support Vector Machines, k-Nearest Neighbors, etc.

High bias -Linear Regression, Logistic Regression, Linear Discriminant Analysis, etc.

## **25. When should you use a t-test vs a z-test?**

The z-test is used for hypothesis testing in statistics with a normal distribution. It is used to determine population variance in the case where a sample is large.

The t-test is used with a t-distribution and used to determine population variance when you have a small sample size.

In case the sample size is large or  $n > 30$ , a z-test is used. T-tests are helpful when the sample size is small or  $n < 30$ .

## **26. What is the equation for confidence intervals for means vs for proportions?**

To calculate the confidence intervals for mean, we use the following equation –

### **For $n > 30$**

Use the Z table for the standard normal distribution.

### **For $n < 30$**

Use the t table with  $df = n - 1$

### **Confidence Interval for the Population Proportion –**

## **27. What is the empirical rule?**

In statistics, the empirical rule states that every piece of data in a normal distribution lies within three standard deviations of the mean. It is also known as the 68–95–99.7 rule. According to the empirical rule, the percentage of values that lie in a normal distribution follow the 68%, 95%, and 99.7% rule. In other words, 68% of values will fall within one standard deviation of the mean, 95% will fall within two standard deviations, and 99.75 will fall within three standard deviations of the mean.

## **28. How are confidence tests and hypothesis tests similar? How are they different?**

Confidence tests and hypothesis tests both form the foundation of statistics.

The confidence interval holds importance in research to offer a strong base for research estimations, especially in medical research. The confidence interval provides a range of values that helps in capturing the unknown parameter.

Hypothesis testing is used to test an experiment or observation and determine if the results did not occur purely by chance or luck using the below formula where ‘p’ is some parameter.

Confidence and hypothesis testing are inferential techniques used to either estimate a parameter or test the validity of a hypothesis using a sample of data from that data set. While confidence interval provides a range of values for an accurate estimation of the precision of that parameter, hypothesis testing tells us how confident we are inaccurately drawing conclusions about a parameter from a sample. Both can be used to infer population parameters in tandem.

In case we include 0 in the confidence interval, it indicates that the sample and population have no difference. If we get a p-value that is higher than alpha from hypothesis testing, it means that we will fail to reject the null hypothesis.

## **29. What general conditions must be satisfied for the central limit theorem to hold?**

Here are the conditions that must be satisfied for the central limit theorem to hold –

- The data must follow the randomization condition which means that it must be sampled randomly.
- The Independence Assumptions dictate that the sample values must be independent of each other.
- Sample sizes must be large. They must be equal to or greater than 30 to be able to hold CLT. Large sample size is required to hold the accuracy of CLT to be true.

## **30. What is Random Sampling? Give some examples of some random sampling techniques.**

Random sampling is a sampling method in which each sample has an equal probability of being chosen as a sample. It is also known as probability sampling.

Let us check four main types of random sampling techniques –

- Simple Random Sampling technique – In this technique, a sample is chosen randomly using randomly generated numbers. A sampling frame with the list of members of a population is required, which is denoted by ‘n’. Using Excel, one can randomly generate a number for each element that is required.
- Systematic Random Sampling technique -This technique is very common and easy to use in statistics. In this technique, every k'th element is sampled. For instance, one element is taken from the sample and then the next while skipping the pre-defined amount or ‘n’.

In a sampling frame, divide the size of the frame N by the sample size (n) to get ‘k’, the index number. Then pick every k'th element to create your sample.

- Cluster Random Sampling technique -In this technique, the population is divided into clusters or groups in such a way that each cluster represents the population. After that, you can randomly select clusters to sample.
- Stratified Random Sampling technique – In this technique, the population is divided into groups that have similar characteristics. Then a random sample can be taken from each group to ensure that different segments are represented equally within a population.

### **31. What is the difference between population and sample in inferential statistics?**

A population in inferential statistics refers to the entire group we take samples from and are used to draw conclusions. A sample, on the other hand, is a specific group we take data from and this data is used to calculate the statistics. Sample size is always less than that of the population.

### **32. What are descriptive statistics?**

Descriptive statistics are used to summarize the basic characteristics of a data set in a study or experiment. It has three main types –

- Distribution – refers to the frequencies of responses.

- Central Tendency – gives a measure or the average of each response.
- Variability – shows the dispersion of a data set.

### **33. What are quantitative data and qualitative data?**

Qualitative data is used to describe the characteristics of data and is also known as Categorical data. For example, how many types. Quantitative data is a measure of numerical values or counts. For example, how much or how often. It is also known as Numeric data.

### **34. How to calculate range and interquartile range?**

The range is the difference between the highest and the lowest values whereas the Interquartile range is the difference between upper and lower medians.

$$\text{Range (X)} = \text{Max(X)} - \text{Min(X)}$$

$$\text{IQR} = \text{Q3} - \text{Q1}$$

Here, Q3 is the third quartile (75 percentile)

Here, Q1 is the first quartile (25 percentile)

### **35. What is the meaning of standard deviation?**

Standard deviation gives the measure of the variation of dispersion of values in a data set. It represents the differences of each observation or data point from the mean.

$$(\sigma) = \sqrt{(\sum(x-\mu)^2 / n)}$$

Where the variance is the square of standard deviation.

### **36. What is the relationship between mean and median in normal distribution?**

In a normal distribution, the mean and the median are equal.

### **37. What is the left-skewed distribution and the right-skewed distribution?**

In the left-skewed distribution, the left tail is longer than the right side.

Mean < median < mode

In the right-skewed distribution, the right tail is longer. It is also known as positive-skew distribution.

Mode < median < mean

### **38. How to convert normal distribution to standard normal distribution?**

Any point ( $x$ ) from the normal distribution can be converted into standard normal distribution ( $Z$ ) using this formula –

$$Z(\text{standardized}) = (x - \mu) / \sigma$$

Here,  $Z$  for any particular  $x$  value indicates how many standard deviations  $x$  is away from the mean of all values of  $x$ .

### **39. What can you do with an outlier?**

Outliers affect A/B testing and they can be either removed or kept according to what situation demands or the data set requirements.

Here are some ways to deal with outliers in data –

- Filter out outliers especially when we have loads of data.
- If a data point is wrong, it is best to remove the outliers.
- Alternatively, two options can be provided – one with outliers and one without.
- During post-test analysis, outliers can be removed or modified. The best way to modify them is to trim the data set.
- If there are a lot of outliers and results are critical, then it is best to change the value of the outliers to other variables. They can be changed to a value that is representative of the data set.
- When outliers have meaning, they can be considered, especially in the case of mild outliers.

### **40. How to detect outliers?**

The best way to detect outliers is through graphical means. Apart from that, outliers can also be detected through the use of statistical methods using tools

such as Excel, Python, SAS, among others. The most popular graphical ways to detect outliers include box plot and scatter plot.

#### **41. Why do we need sample statistics?**

Sampling in statistics is done when population parameters are not known, especially when the population size is too large.

#### **42. What is the relationship between standard error and margin of error?**

Margin of error = Critical value X Standard deviation for the population

and

Margin of error = Critical value X Standard error of the sample.

The margin of error will increase with the standard error.

#### **43. What is the proportion of confidence intervals that will not contain the population parameter?**

Alpha is the probability in a confidence interval that will not contain the population parameter.

$$\alpha = 1 - CL$$

Alpha is usually expressed as a proportion. For instance, if the confidence level is 95%, then alpha would be equal to 1-0.95 or 0.05.

#### **44. What is skewness?**

Skewness provides the measure of the symmetry of a distribution. If a distribution is not normal or asymmetrical, it is skewed. A distribution can exhibit positive skewness or negative skewness if the tail on the right is longer and the tail on the left side is longer, respectively.

#### **45. What is the meaning of covariance?**

In statistics, covariance is a measure of association between two random variables from their respective means in a cycle.

## **46. What is a confounding variable?**

A confounding variable in statistics is an ‘extra’ or ‘third’ variable that is associated with both the dependent variable and the independent variable, and it can give a wrong estimate that provides useless results.

For example, if we are studying the effect of weight gain, then lack of workout will be the independent variable, and weight gain will be the dependent variable. In this case, the amount of food consumption can be the confounding variable as it will mask or distort the effect of other variables in the study. The effect of weather can be another confounding variable that may later affect the experiment design.

## **47. What does it mean if a model is heteroscedastic?**

A model is said to be heteroscedastic when the variation in errors comes out to be inconsistent. It often occurs in two forms – conditional and unconditional.

## **48. What is selection bias and why is it important?**

Selection bias is a term in statistics used to denote the situation when selected individuals or a group within a study differ in a manner from the population of interest that they give systematic error in the outcome.

Typically selection bias can be identified using bivariate tests apart from using other methods of multiple regression such as logistic regression.

It is crucial to understand and identify selection bias to avoid skewing results in a study. Selection bias can lead to false insights about a particular population group in a study.

Different types of selection bias include –

- Sampling bias – It is often caused by non-random sampling. The best way to overcome this is by drawing from a sample that is not self-selecting.
- Participant attrition – The dropout rate of participants from a study constitutes participant attrition. It can be avoided by following up with the participants who dropped off to determine if the attrition is due to the presence of a common factor between participants or something else.

- Exposure – It occurs due to the incorrect assessment or the lack of internal validity between exposure and effect in a population.
- Data – It includes dredging of data and cherry-picking and occurs when a large number of variables are present in the data causing even bogus results to appear significant.
- Time-interval – It is a sampling error that occurs when observations are selected from a certain time period only. For example, analyzing sales during the Christmas season.
- Observer selection- It is a kind of discrepancy or detection bias that occurs during the observation of a process and dictates that for the data to be observable, it must be compatible with the life that observes it.

## **49. What does autocorrelation mean?**

Autocorrelation is a representation of the degree of correlation between the two variables in a given time series. It means that the data is correlated in a way that future outcomes are linked to past outcomes. Autocorrelation makes a model less accurate because even errors follow a sequential pattern.

## **50. What does Design of Experiments mean?**

The Design of Experiments or DOE is a systematic method that explains the relationship between the factors affecting a process and its output. It is used to infer and predict an outcome by changing the input variables.

## **51. What is Bessel's correction?**

Bessel's correction advocates the use of  $n-1$  instead of  $n$  in the formula of standard deviation. It helps to increase the accuracy of results while analyzing a sample of data to derive more general conclusions.

## **52. What types of variables are used for Pearson's correlation coefficient?**

Variables (both the dependent and independent variables) used for Pearson's correlation coefficient must be quantitative. It will only test for the linear relationship between two variables.

## **53. What is the use of Hash tables in statistics?**

In statistics, hash tables are used to store key values or pairs in a structured way. It uses a hash function to compute an index into an array of slots in which the desired elements can be searched.

#### **54. Does symmetric distribution need to be unimodal?**

Symmetrical distribution does not necessarily need to be unimodal, they can be skewed or asymmetric. They can be bimodal with two peaks or multimodal with multiple peaks.

#### **55. What is the benefit of using box plots?**

Boxplot is a visually effective representation of two or more data sets and facilitates quick comparison between a group of histograms.

#### **56. What is the meaning of TF/IDF vectorization?**

TF/IDF is an acronym for Term Frequency – Inverse Document Frequency and is a numerical measure widely used in statistics in summarization. It reflects the importance of a word or term in a document. The document is called a collection or corpus.

#### **57. What is the meaning of sensitivity in statistics?**

Sensitivity refers to the accuracy of a classifier in a test. It can be calculated using the formula –

Sensitivity = Predicted True Events/Total number of Events

#### **58. What is the difference between the first quartile, the second quartile, and the third quartile?**

The first quartile is denoted by Q1 and it is the median of the lower half of the data set.

The second quartile is denoted by Q2 and is the median of the data set.

The third quartile is denoted by Q3 and is the median of the upper half of the data set.

About 25% of the data set lies above Q3, 75% lies below Q3 and 50% lies below Q2. The Q1, Q2, and Q3 are the 25<sup>th</sup>, 50<sup>th</sup>, and 75<sup>th</sup> percentile respectively.

#### **59. What is kurtosis?**

Kurtosis is a measure of the degree of the extreme values present in one tail of distribution or the peaks of frequency distribution as compared to the others. The standard normal distribution has a kurtosis of 3 whereas the values of symmetry and kurtosis between -2 and +2 are considered normal and acceptable. The data sets with a high level of kurtosis imply that there is a presence of outliers. One needs to add data or remove outliers to overcome this problem. Data sets with low kurtosis levels have light tails and lack outliers.

## **60. What is a bell-curve distribution?**

A bell-curve distribution is represented by the shape of a bell and indicates normal distribution. It occurs naturally in many situations especially while analyzing financial data. The top of the curve shows the mode, mean and median of the data and is perfectly symmetrical. The key characteristics of a bell-shaped curve are –

- The empirical rule says that approximately 68% of data lies within one standard deviation of the mean in either of the directions.
- Around 95% of data falls within two standard deviations and
- Around 99.7% of data fall within three standard deviations in either direction.

## **How do I prepare for a statistics interview?**

To prepare for a statistics interview, you can read this blog on the top commonly asked interview questions. These questions will help you brush up your skills and ace your upcoming interview.

## **What are the most important topics in statistics?**

Estimation: bias, maximum likelihood, method of moments, Rao-Blackwell theorem, fisher information. Central limit theorem, hypothesis testing, likelihood ratio tests, law of large numbers – These are some of the most important topics in statistics.

## **What are basics of statistics?**

A collection of methods to display, analyze, and draw conclusions from data. Statistics can be of two types, descriptive statistics and inferential statistics.

## **What are the 7 steps in hypothesis testing?**

1. State the null hypothesis
2. State the alternate hypothesis
3. Which test and test statistic to be performed

4. Collect Data
5. Calculate the test statistic
6. Construct Acceptance / Rejection regions
7. Based on steps 5 and 6, draw a conclusion about  $H_0$

## **1. What is the chance of rolling at least one five with two dice?**

Let's assume that event A is that we get 5 on 1st dice and B is that we get 5 on 2nd dice

- Since the outcome of throwing the second die wouldn't be affected by the outcome of throwing the first dice, we can calculate the probability of independent events A and B both occurring as:  $P(A \cap B) = P(A) * P(B)$
- The probability of getting at least one 5 can be computed using the probability of the union of two events:
  - $P(A \cup B) = P(A) + P(B) - P(A \cap B)$  The probability of getting any specific outcome from a die is  $\frac{1}{6}$ .
  - Thus,  $P(A \cup B) = \frac{1}{6} + \frac{1}{6} - \frac{1}{6 \cdot 6} = \frac{1}{3} - \frac{1}{36} = \frac{11}{36}$

Thus the probability of rolling at least one five with two dice is **11/36**.

## **2. Given that a die is rolled twice and the sum of the numbers is noted to be 6, what is the conditional probability that the number 4 has occurred at least once?**

If you roll the dice twice, you'll get the following sample space:

$$S = \{(1,1)(1,2)(1,3)(1,4)(1,5)(1,6)$$

$$(2,1)(2,2)(2,3)(2,4)(2,5)(2,6)$$

$$(3,1)(3,2)(3,3)(3,4)(3,5)(3,6)$$

$$(4,1)(4,2)(4,3)(4,4)(4,5)(4,6)$$

$$(5,1)(5,2)(5,3)(5,4)(5,5)(5,6)$$

$$(6,1)(6,2)(6,3)(6,4)(6,5)(6,6)\}$$

Provided the given data, calculate the probability that 4 has appeared at least once, given that the sum of the numbers is 6.

Assume that F: The total of two numbers is six.

Take E, for example, 4 has appeared at least once.

As a result, we must locate  $P(E|F)$ .

Obtaining  $P(E)$ :

The chances of collecting four at least once are:

$$E = \{(1, 4), (2, 4), (3, 4), (4, 4), (5, 4), (6, 4), (4, 1), (4, 2), (4, 3), (4, 5), (4, 6), (4, 1), (4, 2), (4, 3), (4, 5), (4, 6)\}$$

As a result,  $P(E) = 11/36$ .

Identifying  $P(F)$ :

The chance of getting the sum of two numbers is 6:

$$F = \{(1, 5), (5, 1), (2, 4), (4, 2), (3, 3)\}$$

As a result,  $P(F) = 5/36$

$$\text{In addition, } E \cap F = \{(2, 4), (4, 2)\}$$

$$P(E \cap F) = 2/36$$

$$\text{As a result, } P(E|F) = (P(E \cap F)) / (P(F))$$

Now, Substitute the computed probability values=  $(2/36) / (5/36)$

Hence,  $2/5$  is the required probability.

### **3. Team A and B are competing in a game in which they must win four of the seven rounds to win.**

**What is the likelihood that they will play all seven rounds if A's chance of winning is  $p$  and B's chance of winning is  $1-p$  (no chance of a tie)? What if the chances of A winning differ on the home field ( $p$ ) and the away field ( $q$ )?**

If two teams compete in all seven rounds, both A and B must win three times in the first six rounds, regardless of who wins the final round. Each round can be thought of as a Bernoulli trial, with the number of times A wins in the first six games following a binomial distribution. The probability of A winning is  $B(n, k, p)$ , with  $n=6$ ,  $k=3$ , and  $p=p$ . The probability of A winning three times out of six games, according to the Binomial distribution, is:

$$Bi(6,3,p) = \binom{6}{3} * p^3 * (1-p)^3 = \frac{6!}{3! * (6-3)!} * p^3 * (1-p)^3 = 20p^3(1-p)^3$$



Because team A has won three times, team B has won three times as well. We can suppose that Team A's chance of winning at home is p, away is q, and Team A has won x games at home if the two teams have different winning percentages at home and away. x, p, and q will determine the likelihood of both teams playing all seven rounds. We know that both teams A and B must win three rounds and that Team A must win x games at home and 3-x games away, while Team B must win 3-x games away (Team A's home being Team B's visit site, and Team B wins aways when A loses at home) and win x rounds at home. Playing seven rounds has a probability of:

$$\binom{3}{x} * p^x * (1-p)^{3-x} * \binom{3}{x} * q^{3-x} * (1-q)^x$$



#### **4. What is the likelihood of drawing two cards with the same suite (from the same deck)?**

This is an illustration of a dependent event. According to this definition, the likelihood that two events will occur in the case of a dependent event is:

The probability of events A and B occurring simultaneously is equal to the chance of events A occurring multiplied by the likelihood of events B occurring given the outcome of events A.

This is expressed as  $P(A \cap B) = P(A) * P(B|A)$

In this scenario, a deck of cards comprises four suites, each of which contains 13 cards.

Our chance of getting a card from a certain suite in the initial draw would be 13/52. Our chances of drawing a card from the same suite as the previous one in the subsequent draw would drop from 13/52 to 12/51.

Hence  $P(\text{two cards same suite})$

$$= 4 * 13/52 * 12/51$$

$$= 4/17$$

**5. We choose two cards at random from a deck of cards with numbers ranging from 1 to 100. What is the likelihood that a number on one card is precisely twice the number on the other?**

With the idea of a combination, this query can also be resolved. This is due to the fact that the order is unimportant when we draw two cards from the same deck of cards. As a result, receiving a card with the number 10 in the first draw and the number 40 in the second draw is equivalent to receiving a card with the number 40 in the first draw and the number 10 in the second.

Input values from the question into the combination equation will provide the following results:

$$C(100,2) = 100!/(100-2)!2! = 4950 \text{ combinations}$$

Hence there are 4950 combination pairings available.

Given that we have 100 cards altogether, there are 50 ways out of those 4950 combinations where one card is double another.

As a result, we may determine the probability as:

$$50/4950 = 0.01 = 1\%$$

**6. What is the probability of rolling the dice seven times and receiving a 5?**

Simply entering values into the binomial distribution equation will provide the solution to this query. Assuming that there have been 1 success and 7 trials, we can say that there have been 1 success and 7 trials. As we all know, there is a 1 in 6 chance of getting a 5 on a single throw.

The probability mass function (PMF) of the binomial distribution is as follows:

$$P(X=k) = \binom{n}{k} p^k (1-p)^{n-k}$$



Hence by putting values we get,

$$P(X=k) = \binom{7}{1} \frac{1}{6} \left(1 - \frac{1}{6}\right)^6 = 0.397$$



## **7. N riders receive a \$5 off voucher. P is the likelihood that a coupon will be used. What is the company's anticipated cost?**

Different from the previous question, now we need to compute the expected value of a variable with binomial distribution instead of computing the PMF. We can answer this question by plugging the values into the equation of the expected value of the binomial distribution.

From the equation above, we have  $N$  coupons and the probability of using a coupon is  $P$ .

Thus, the expected value would be:  $E(X)=N*P$

And the anticipated expense would be:  $E(X)*5\$=N*P*5$

## **8. What is the variance's definition?**

The variance measures the range of data points in a dataset in relation to its mean value, as the notion suggests. The general formula for a variance is given below:

$$S^2 = \frac{\sum (x_i - \bar{x})^2}{n-1}$$



**InterviewBit**

Where n is the total number of samples, x is the sample, x bar is the sample mean, and S is the variance.

**9. What is a p-value? If you had a different (far larger, 3 mil records, for example) data set, would it affect how you interpret the p-value?**

In the world of statistics, the term "p-Value" refers to the probability value, which is typically applied during hypothesis testing. When your null hypothesis is true, the facts you just observed are considered to be very unlikely, according to the p-value.

Typically, a significance level is established before hypothesis testing. We reject the null hypothesis if the p-value is less than the threshold we consider significant. In the meantime, if the p-Value exceeds the level of significance, we proceed with our null hypothesis.

The meaning of the p-Value is unaffected by the size of your dataset, although a larger dataset yields a more solid and trustworthy conclusion from our p-Value.

**10. Assume Facebook users click on ads P times out of every 100. What is the smallest sample size N such that Probability(ABS(hat{P}-P) <= DELTA) = 95 percent when we choose a sample of size N and analyze the sample's conversion rate, denoted by hat{P}?**

**Find the smallest sample size N that will allow our sample estimate of P to be, with a 95% confidence level, within a DELTA of the actual click-through rate P.**

The understanding of confidence intervals, the margin of error, sample size, and binomial distribution is tested by this question. The conversion rate in the hypothetical situation has a binomial distribution, so we must calculate the standard deviation using the square root of the variance of the binomial distribution.

The following is the general equation for the binomial distribution's variance:  $\text{Var}(X) = p(1-p)$

Our understanding of the question's 95 per cent confidence interval results in a Z-score of 1.96. (see the Z-table to obtain this value). When we enter the following equation into the equation margin of error, we obtain:

$$\delta = 1.96 * \sqrt{\frac{P(1-P)}{N}}$$

$$N \geq 1.96^2 * \frac{P(1-P)}{(\delta)^2}$$



**11. Out of 100 goods, 25 are of poor quality. What is the range of confidence?**

We must calculate the sample mean from the expected value of the binomial distribution and the standard deviation from the variance of the binomial distribution because the problem in the question has a binomial distribution:

$$E(X) = 100 * 0.25 \quad E(X) = 100 * 0.25$$

$$\text{Var}(X) = 100 * 0.25 * (1 - 0.25) = 18.75$$

We can simply enter the computed mean and standard deviation into the equation for confidence intervals to determine the result after computing them using the binomial distribution method.

$$CI = \bar{X} \pm \frac{Z^* \sigma}{\sqrt{n}}$$



**InterviewBit**

$\bar{X}$  is the sample mean,  $Z$  is the confidence value,  $\sigma$  is the sample standard deviation, and  $n$  is the sample size in the equation above.

So after putting values we get

$$CI = 25 \pm \frac{1.96 * 18.75}{\sqrt{100}}$$

## 12. How do you determine whether assignment to the different buckets in an A/B test was indeed random?

In terms of statistics, there wouldn't be any discernible disparities between the variable samples in each bucket if the buckets were actually chosen at random. However, how can we tell if the sample differences between the buckets are meaningful or not?

To assess this, we can perform a statistical test.

The two-sample t-test can be used if the variables we observe are continuous variables and there is only one treatment. In the meanwhile, we can utilize ANOVA if there are numerous treatments.

We can utilize the p-Value obtained after running a statistical test to determine whether there is a significant difference between buckets.

### **13. What distinguishes probability density functions from probability mass functions?**

- **Probability mass function:-** The probability distribution for discrete variables is provided by the Probability Mass Function (PMF). For instance, throwing dice. The complete sample space is defined by 6 unique outcomes: 1, 2, 3, 4, 5, and 6. We only have whole numbers; there are no fractions like 1.2 or 3.75. Each discrete variable's probability is mapped to the PMF. Each of the six factors has a 1/6 chance of being rolled in an ideal environment when rolling a die.
- **Probability density function:-** The probability distribution of the continuous random variables is provided by the Probability Density Function. we use a probability distribution function where outcome values are not fixed like we have a range of values as outcomes for example if we try to calculate the height of students in a class we get the value in range. The probability density function is calculated by the area under the curve of the interval in which or outcome values lie.

### **14. What is a Probability Distribution?**

A probability distribution is a statistical function that outlines every conceivable value and likelihood that a random variable could have within a specified range.

Two main categories of probability distribution exist:

- **Discrete probability distributions:** used random variables having discrete outcomes, such as the frequency of heads in five consecutive coin tosses, the number of rainy days in a particular week, the number of goals a player scores, etc..
- **Continuous probability distributions:** used for random variables with continuous outcomes, such as the average height of male students, the median price of a home in San Francisco, the number of claims a given insurance provider receives, and so forth.

### **15. How would you use Bayes' Rule to test hypotheses?**

We can calculate a conditional probability using new information we already have using the Bayes' Rule equation. In the case of two distinct events A and B, we can consider the following two events in terms of Baye's theorem:

- Hypothesis A, which may be true or untrue
- Evidence A supporting B, which may be present or not.

As a result, A is the occurrence whose likelihood we are interested in. The issue can be stated as follows using Bayes' theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Where:

- The probability that event A will occur if event B is true is expressed as  $P(A|B)$ . We are attempting to estimate what is also known as the posterior.
- $P(B|A)$  is the likelihood that event B will occur assuming that event A is true. It can alternatively be understood as the probability of witnessing the new data given our prior hypothesis, and it can also be understood as the likelihood.
- The probabilities of seeing A and B,  $P(A)$ , and  $P(B)$ , respectively, in the absence of any specific conditions, are also known as the prior and marginal probabilities.

Now, in order to compare the two hypotheses, we compute their respective probabilities using the previous technique. Acceptance is given to the hypothesis with the highest posterior probability.

## **16. How many advertisements should be expected to appear in 100 news stories for each option?**

**There are two ways we can serve advertising inside Newsfeed:**

1. Every 25 stories, there will be one advertisement.
2. There is a 4% possibility that every story contains an advertisement.

**How many advertisements should be expected to appear in 100 news stories for each option? What is the likelihood that a user will only see one ad out of every 100 stories if we choose option 2?**

The expected value and binomial distribution PMF are both tested in this question.

The first query, which concerns the anticipated number of advertisements displayed in 100 news pieces, is:  $E(\text{ads shown}) = 100 * 4 / 100 = 4$

The PMF of a binomial distribution can be used to provide an answer to the second query, where there are 100 total trials, one success (a single ad), and a 0.04 probability that each story would contain an advertisement.

$$P(X=k) = \binom{100}{1} * 0.04 * (1-0.04)^{99} = 0.07$$



**17. Let's say you roll a dice and get the face you roll. Let's say you get another chance to roll the die. If you roll, you receive the face you obtain and keep your winnings from the previous round. When should the second roll be made?**

The outcome of a roll of a 6-sided die is predicted to be:  $E(x) = 1/2 * (1+6) = 3.5$

To respond to this query, we must consider it in the following manner:

We shouldn't roll the second die and instead keep the winnings if the first roll yields more than 3.5 (the expected value of one roll). While waiting, we should roll the second die if our result is less than 3.5.

**18. A discount coupon is given to 2 riders. The probability of using a coupon is P. Given that at least one of them uses a coupon, what is the probability that both riders use the coupons?**

You will be tested on your understanding of the Bayes theorem and the binomial distribution in this question.

The probability that exactly one cyclist will use the coupon can be calculated using the binomial PMF:

$$P(X=1) = \binom{2}{1} * P^1 * (1-P) = 2 * P * (1-P)$$



The PMF of a binomial distribution can also be used to calculate the likelihood that both of them will use the coupon.

$$P(X=2) = \binom{2}{2} * P^2 * (1-P)^{2-2} = P^2$$



As per normal, the next step is to describe an event so that we may better grasp what each item in the Bayes' theorem equation means,

- A = At least one rider has used the coupon.
- B = The voucher is used by both riders

The values can now be entered as follows into the Bayes' theorem equation:

$$\begin{aligned} P(B|A) &= [P(A|B)*P(B)] / P(A) \\ &= (1* P^2) / 2*P * (1-P) + P^2 \\ &= P / (2-P) \end{aligned}$$

The 30 probability and statistics interview questions from various companies are concluded at this point. We believe these inquiries will help you hone your abilities so you can ace your data science interview. Remember that you won't be able to answer interview questions about statistics and probability in one sitting; rather, you'll develop the capacity over time by learning consistently.

**19. There are two types of coins: one fair (one side heads, one side tails) and one unfair(both sides tails). You choose one at random, flip it five times, and note that it lands on tails each time. What is the probability of you tossing an unfair coin?**

Here, the Bayes Theorem can be used. Let U stand for the scenario in which we flip an unfair coin and F for the scenario in which we flip a fair coin. We are aware that  $P(U) = P(F) = 0.5$  since the coin is picked at random. Let 5T stand for the scenario in which we consistently flip 5 heads. After that, assuming that we saw 5 tails in a row, we are interested in finding a solution for  $P(U|5T)$ , or the likelihood that we are tossing an unfair coin.

Since the unjust coin will always land on heads, we know  $P(5T|U) = 1$ . Furthermore, we are aware that  $P(5T|F) = 1/25 = 1/32$  according to the concept of a fair coin. Using the Bayes Theorem, we can:

$$P(U|5T) = \frac{P(5T|U) * P(U)}{P(5T|U) * P(U) + P(5T|F) * P(F)} = \frac{0.5}{0.5 + 0.5 * 1/32} = 0.97$$



Therefore, there is a 97 per cent chance that we chose the unjust coin.

**20. A and B are playing a game where each player flips all of their coins. A has n+1 coins, and B has n coins. What is the probability that A will have more heads than B?**

Compare the first n coins flipped by A to the n coins flipped by B.

There are three potential outcomes:

1. More heads are on A than B.
2. Equal numbers of heads are present in A and B.
3. A is headless compared to B.

Keep in mind that A will always win in scenario 1 (regardless of coin n+1), while A will always lose in scenario 3 (regardless of coin n+1). These two scenarios have an equal chance of happening because of symmetry.

Put x for any scenario's probability and y for scenario 2's probability.

Since there are only 3 events that can occur, we know that  $2x + y = 1$ . Let's now think about coin n+1. A will have prevailed if the coin lands face up with a probability of 0.5 under scenario 2. (which happens with probability y). As a result, A's overall odds of winning the game increase by 0.5y.

So, the likelihood that player A will prevail in the game is:  $x+1/2y = x+!/2(1-2x) = 1/2$

**21. A and B are participating in a game of archery together. Assume that both of their arrow-firing skills are identical and that both have a 0.5 chance of hitting the target.**

**What is the probability that A will hit more targets than B given that A has fired 201 arrows and B has fired 200?**

Since 201 is not an even number, let's start with 200 games. Assume that in 200 games, event A is A shooting more arrows on target than event B, event B is B shooting more arrows on target, and event C is they both shoot the same number of arrows on targets. We possess

Given that A and B compete evenly in 200 games of archery, we obtain  $P(A) = P(B)$ . Thus:

Now switch to the additional game that player A plays. if over the previous 200 games:

- If A is higher than B, then A remains higher than B whether A hits the target in this additional game or not.
- If A is less than B, then A will still not be more than B even if A fires on target for the additional game.
- If  $A=B$  and A hits the target in the additional game, then A will be higher than B and there is a 0.5 chance that A will hit the target in any game.

As a result, the overall likelihood that A surpasses B is:

Since  $2P(A) + P(C) = 1$ , we can divide 2 into both sides to get the following result:  $\textcolor{red}{P(A)+P(C)=0.5}$

When A plays 201 games and B plays 200 games, there is a 0.5 per cent chance that A will score more targets than B.

**22. You have 40 cards total, with 10 each of red, green, blue, and yellow. There is a number from 1 to 10 for each color. What is the likelihood that two cards you draw without a replacement will not be the same color or have the same number?**

The odds of receiving two cards with the same number and two cards of the same color can be calculated first, and the result is one less than the total of the two probabilities.

The likelihood of drawing two identical cards is as follows:  $P(\text{Same Number}) = 40/40 * (9/39) = 9/39$

Any number can be drawn in the first draw, regardless of significance. As a result, the likelihood is unaffected by the first draw; but, because there are only 39 cards available, you must choose the same number for the second draw. There are four cards with the same number on them, each in a different color. You can only choose three cards from a total of 39 for the second draw.

The same reasoning applies if you get two cards of the same color:  $P(\text{Same colour}) = 40/40 * (9/39) = 9/39$

In the initial draw, we can choose any color, but we can only select nine cards from the remaining 39 of the same hue. The likelihood of not receiving the same card AND the same number is:

$$P = 1 - P(\text{Same Number}) - P(\text{Same colour}) = 27/39$$

**23. Eight people enter an elevator in a building with ten floors. What is the expected number of stopping? What assumptions do you need to calculate this expectation?**

We can use the binomial distribution to answer this question if we model each passenger's decision to halt on a particular floor as a Bernoulli trial. The presumptions comprise:

- 8 passengers make autonomous choices;
- assume that all occupants enter on the first floor and that there are ten options ranging from one to ten floors. (There are only 9 options if you assume no one stops at the first floor.)

There are eight passengers in all, and the elevator will stop if anyone wishes to exit for any reason. Instead of figuring out how likely it is that the elevator will stop at a specific floor, we may figure out how likely it is that it won't stop. The likelihood that the elevator won't stop at any floor, for any floor, is:  $(9/10)^8$

The likelihood that the elevator will stop at any floor is  $1 - (9/10)^8$

Assume that X is a random variable with the elevator's stopping frequency and that X has a binomial distribution to get the predicted number of stops in this situation

$$X \sim Bi(10, 1 - (9/10)^8)$$

If  $n=10$ , then  $p=1 - (9/10)^8$  The binomial distributed random variable's expected value is  $np$ :  $E(x)=10*1-(9/10)^8$

**24. Assume there is a highly uncommon disease in the world. There is a 0.1 per cent chance that anyone will contract this illness. You decide to take a test to find out if you are infected, and the results are affirmative.**

**99% of those who have the condition will test positive, and 99 percent of those who do not will test positive, according to the test's accuracy (many thanks to Xavier Lavenir for clarifying the assumptions in the question). What are the odds that you are really ill? (We are grateful to Dennis Meisner for spotting the misinterpretation here.)**

Assume that event A has the illness and that event B tested positive. According to the details in the query:

$P(B|A) = 99.9$  percent, and 1 percent of those who tested positive don't have the condition, therefore  $P(B|\text{not } A) = 1$  percent; if  $P(A) = 0.1$  percent, then  $P(\text{not } A)$  equals 99.9 per cent

$P(A|B)$  is what?

Bayes' Theorem:  $P(A|B) = [P(B|A) P(A)] / P(B)$

And,

$$P(B) = P(B|A) * P(A) + P(B | \text{not } A) * P(\text{not } A)$$

Plug in every number:

$$P(A|B) = (0.99 * 0.001) / (0.99 * 0.001 + 0.01 * 0.999) = 9\%$$

All of the questions have answers in the list below. I hope that reading this essay will help you hone your probability theory skills.

**25. In order to win the game, Team A and Team B must win 4 of the game's 7 rounds.**

**What is the likelihood that they will play all seven rounds if the probability of A winning is p, the probability of B winning is 1-p, and there is no chance of a tie? What if the odds that team A wins differ on the home field (p) and the visiting field (q)?**

If two teams compete in all 7 rounds, both A and B must win exactly 3 times in the first 6 rounds; the last round's winner is irrelevant. If we think of each round as a Bernoulli trial, then the distribution of how many times A wins in the first 6 games is binomial.  $B(n,k,p)$  with  $n=6$ ,  $k=3$ , and  $p=p$  gives the likelihood that A

will prevail. The likelihood that player A will win 3 out of every 6 games is, according to the binomial distribution:

$$Bi(6, 3, p) = \binom{2}{2} * p^3 * (1-p)^3 = \frac{6!}{3! * (6-3)!} * p^3 * (1-p)^3 = 20p^3(1-p)^3$$



Keep in mind that when team A wins three times, team B automatically wins three times

We can assume Team A's likelihood of winning at home is  $p$ , Team A's probability of winning away is  $q$ , and Team A has won  $x$  games at home if the two teams have different winning percentages playing at home and playing away. The likelihood that both teams will participate in all 7 rounds will depend on the variables  $x$ ,  $p$ , and  $q$ . We know that team A and team B must each win 3 rounds in order to advance, and that team A must win  $x$  games at home and  $3-x$  games away, while team B must win  $3-x$  games away (team B visits team A's home, therefore team B wins away games when A loses at home) and win  $x$  rounds at home. The likelihood of seven rounds being played is

$$\binom{3}{x} * p^x * (1-p)^{3-x} * \binom{3}{x} * q^{3-x} * (1-q)^x$$



We can learn more about the likelihood if we have additional information about the distribution of  $x$ .

**26. On each corner of an equilateral triangle, three zebras are seated. Each zebra chooses a direction at random and proceeds solely around the triangle's perimeter to either of its opposite edges. What is the likelihood that no zebras will collide?**

Consider the zebras as being arranged in an equilateral triangle. If they are going down the outline to each edge, they each have two possible ways to travel in. Let's calculate the likelihood that they won't collide given that the scenario is random.

Actually, there are just two options. Either all of the zebras will opt to move in a clockwise or counterclockwise direction as they run.

Let's determine the likelihood of each. The likelihood that each zebra will choose to move in a clockwise direction will be determined by the sum of their individual decisions. Given that there are two options (clockwise or counterclockwise), the answer is  $1/2 * 1/2 * 1/2$ , which equals  $1/8$ .

The likelihood of each zebra turning counterclockwise is  $1/8$ . As a result, when the probabilities are added together, we obtain the proper probability of  $1/4$ , or 25%.

**27. Your flight to Seattle is ready to take off. You dial the numbers of three unrelated random friends who reside there to inquire about the weather.**

**Each of your buddies has a  $2/3$  chance of being honest with you and a  $1/3$  chance of playing a practical joke on you by lying. It is raining, as all three of your buddies confirm. What is the probability that Seattle is experiencing rain right now?**

You must assume something about the likelihood of rain in Seattle in order to respond to this question. Say the value is 0.5.

Since each of our pals has a  $2/3$  chance of being honest, there is a  $2/3$  chance that Seattle will experience rain if our friends are correct. Given that our friends predict that it won't rain in Seattle, the likelihood of it not raining is also  $2/3$ .

Let's define an event as follows in light of this:

- $A =$  a rainy day in Seattle.
- $A' =$  not raining in Seattle
- $X_i =$  random variable with a Bernoulli distribution, and its value corresponds to the response provided by our friends: raining (1) or not (0)

By using Bayes' theorem, we can therefore approximatively determine the likelihood that it will rain in Seattle provided that our friends predict that it will.

- **When is an event A independent of itself?**

Solution – An event may just be separate from itself when there's zero possibility of it happening or perhaps when it's certainly going to take place. Events A and B are actually independent when  $P(A \cap B) = P(A) * P(B)$

Given  $B=A$ ,  $P(A \cap A) = P(A)$  when  $P(A) = \text{zero or perhaps one}$ .

- **Does the frequentist approach always give the same result as the Bayesian approach?**  
Solution – No. The frequentist strategy is determined by the way the hypothesis is actually identified while our prior faiths are actually updated by the Bayesian strategy. Therefore the frequentist approach may well lead to an exactly opposite conclusion in case the hypotheses are reported in a unique fashion. Hence the 2 procedures might not deliver the exact same results.
- **If you should generate a random number between 1 – 7 with only one die, how will you do it?**  
Solution – We have to release the die three times: each and every throw sets the nth component of the outcome. For every launch, if the great is actually between one and three, it is going to be zero, else one. The end result is actually between zero (zero) and seven (111), spread quite uniformly because there are actually three independent throws. If we repeat the throws when zero was obtained: the procedure prevents uniformly sent out values.
- **If you are given draws from a normal distribution with known values of parameters, how can you generate draws from a uniform distribution?**  
Solution – We have to type in the value from the regular distribution collective distribution function of the very same random variable.
- **A jar contains 4 marbles. 3 Red & 1 white. Two marbles are removed with replacement after each draw. Find the probability that the same color marble is drawn twice?**  
Solution – Suppose that the marbles are of the same color. Then the calculation will be  $3/4 * 3/4 + 1/4 * 1/4 = 5/8$
- **In a website offering dating services, users can select 5 out of 24 adjectives to describe themselves.**  
Solution – A match is said to be between two users if they match on at least 4 adjectives.
- **If Alice and Bob randomly pick adjectives, what is the probability that they are able to find a match?**  
Solution – The probability here is calculated as  $24C5 * (1+5(24-5))/24C5 * 24C5 = 4/1771$

- There's a 0.1 % possibility of getting a coin with the two heads, along with a 99.9 % chance you buy a reasonable coin. A coin is actually flipped plus it comes up heads ten times. What is the possibility that the reasonable coin was picked, because of the info that you observed?

Solution – The possible incidents are actually  $F = \text{"picked a good coin"}$ ,  $T = \text{"10 heads inside a row"}$

$$P(F|T) = P(T|F)P(F)/P(T) \text{ (Bayes theorem) } -(1)$$

$$P(T) = P(T|F)P(F) + P(T|\neg F)P(\neg F) \text{ (total probabilities) } -(2)$$

Injecting (2) in (1):

$$\begin{aligned} P(F|T) &= P(T|F)P(F)/(P(T|F)P(F) + P(T|\neg F)P(\neg F)) = 1 / (1 + \\ &P(T|\neg F)P(\neg F)/(P(T|F)P(F))) \\ &= 1/(1 + 0.001 * 2^{10} / 0.999) \end{aligned}$$

With  $2^{10} \approx 1000$  and  $0.999 \approx \text{one}$ , this is equal to 0.5

- If a life insurance business sells a 1dollar1 240,000 living insurance policy with a one-year phrase to a 25 year older woman for 1dollar1 210, the likelihood that she survives the season is actually,999592. Find the anticipated value of this particular policy for the insurance business?

Solution – The probability that the company loses the money,  $P(\text{company loses the money}) = 0.99592$

The probability that the company doesn't lose the money  $P(\text{the company does not lose the money}) = 0.000408$

The amount of money the company loses in case of loss =  $240,000 - 210 = 239790$

The money gained is \$210

Expected money the company should give =  $239790 * 0.000408 = 97.8$

Expect money the company receives = 210.

Therefore the required value =  $210 - 98 = \$112$

- Alice has 2 children, one of which is a girl. In what probability will the child be also a girl?
- Solution – Assuming there are an equal number of males and females in the world, the outcomes for two kids can be {BB, BG, GB, GG}
- Since it is given that one of them is a girl, the BB option can be removed. Therefore the sample space has 3 options. In those, only one fits the second condition. Therefore the probability that the second child will be a girl too is 1/3
- In a category of thirty pupils, what's the probability 2 of the pupils have the birthday of theirs on the exact same (assuming it's not really

**a leap year)?**

Solution – An example of a favorable event would-be students with birthdays 3rd Jan 1998 and 3rd Jan 1997. The total number of possible combinations for no two persons to have the same birthday in a class of 30 is  $30 * (30-1)/2 = 435$ .

Now, a year has 365 days (if not a leap year). Thus, the probability of two persons having a different birthday would be  $364/365$ . Out of 870 possible combinations, no two people having the same birthday is  $(364/365)435 = 0.303$ . Thus, the probability of two people having their birthdays on the same date would be  $1 - 0.303 = 0.696$  or 69.6%

- **According to hospital records, 75% of patients suffering from a disease die from that disease. Find out the probability that 4 out of the 6 randomly selected patients survive.**

Solution – This has to be a binomial there are only 2 outcomes – death or life. Here  $n = 6$ , and  $x=4$ .  $p=0.25$ (probability if life)  $q = 0.75$ (probability of death)

$$P(X) =$$

$$nCx * p^x * q^{(n-x)} = 6C4 * (0.25)^4 * (0.75)^2 = 0.03295$$

- **A fly carries a lifetime of between 4 6 days. What's the likelihood that the fly is going to die in precisely five days?**

Solution – The continuous probabilities below create a mass function. The likelihood of the event is actually estimated by discovering the part under the curve. Here since we need to estimate the likelihood of the fly expiring for precisely five days – the part under the curve can be zero.

- **A roulette wheel has thirty-eight slots – eighteen are red, eighteen are black, and two are going green. You're playing 5 games and usually bet on red. What's the probability you go on to win five games?**

Solution – The probability that this color is packaged as white in any spin is actually  $18/38$ . The game is actually being played five times and most of the games are actually independent of one another. Therefore, the likelihood that all the video games are actually won is actually  $(18/38)^5 = 0.0238$

- **Say you have a sandwich shop. Out of the readily available choices, 70 % of individuals pick an egg, as well as the rest select chicken. What's the probability that you sell two egg sandwiches to the following three customers?**

Solution – The likelihood of promoting an egg sandwich is actually 0.7 and selling a chicken sandwich is actually 0.3. The probability that

the next three customers will purchase two egg sandwiches is actually 0.7  
 $* 0.7 * 0.3 = 0.147$

$$P(A | X_1=1 \cap X_2=1 \cap X_3=1) \\ \frac{\frac{1}{2} * \frac{1}{2}^3}{\left(\frac{2}{3} * \frac{1}{2}\right) + \left(\frac{2}{3} * \frac{1}{2}\right)} = 0.888$$



**InterviewBit**

## **28. Imagine you attended a technical job interview.**

**50% of those who participated in the first interview received a call for the second interview. Those who received a call for a second interview felt positive about it in 95% of cases. Seventy-five per cent of those who did not get a follow-up contact were satisfied with their initial interview. What is the probability that you will be called back for a second interview if your first interview went well?**

Assume that 100 participants participated in the initial interview process. For the second round of interviews, 50 candidates received calls. Out of this, 47.5 per cent, or 95 per cent, agreed that their interview went well. 50 persons were not called for the interview; of those, 37.5 (or 75% of them) felt happy about it.

Thus, a total of  $(37.5 + 47.5)$  85 participants reported feeling positive after conducting their interview.

As a result, only 47.5 of the 85 candidates who felt excellent received the call for the following stage. The likelihood of success is, therefore  $(47.5/85) = 0.558$ .

Using the Bayes theorem, one can elegantly answer the following question:

- A: Feeling good after your initial interview.
- B: A call inviting you to a second interview.

Now,

$$P(A) = 0.5 * 0.95 + 0.5 * 0.75 = 0.85$$

$$P(B) = 0.$$

$$P(A|B) = 0.95$$

Hence,

$$P(B|A) \text{ equals } [P(A|B)*P(B)]. / P(A) = (0.95 * 0.5)/0.85 = \mathbf{0.558}$$

## **29. The likelihood that people A and B could each independently solve the given problem is 1/2 and 1/3, respectively.**

Given that, the two events say A and B are independent if  $P(A \cap B) = P(A) \cdot P(B)$

We can see from the data that  $P(A) = 1/2$  and  $P(B) = 1/3$ .

The likelihood that a problem will be solved is equal to the likelihood that either person A or person B will do so.

The following can be written:

$$= P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

$P(A \cup B)$  Equals  $P$  if A and B are independent ( $A$ ).  $P(B)$

Change the values now,

$$= (1/2) \times (1/3)$$

$$P(A \cap B) = 1/6$$

The likelihood of a problem being solved is now expressed as

$P(\text{Problem solved})$  equals  $P(A)$  plus  $P(B)$  minus  $P(A \cap B)$ .

$$= (1/2) + (1/3) - (1/6)$$

$$= (3/6) + (2/6) - (1/6)$$

$$= 4/6$$

$$= 2/3$$

The likelihood that the problem will be solved is therefore  $2/3$ .

**30. Three groups A, B & C are competing for positions on the Board of directors of a company.**

**The probabilities of their winning are 0.5, 0.3, and 0.2 respectively. If group A wins, the probability of introducing a new product is 0.7 and the corresponding probabilities for groups B & C are 0.6 & 0.5 respectively. Find the probability that the new product will be introduced.**

Given  $P(A) = 0.5$ ,  $P(B) = 0.3$  and  $P(C) = 0.2$

therefore  $P(A) + P(B) + P(C) = 1$

then events A, B, and C are exhaustive.

If  $P(E) =$  Probability of introducing a new product, then as given

$P(EIA) = 0.7$ ,  $P(EIB) = 0.6$  and  $P(EIC) = 0.5$

$P(E) = P(A) \cdot P(EIA) + P(B) \cdot P(EIB) + P(C) \cdot P(EIC)$

$$= 0.5 \times 0.7 + 0.3 \times 0.6 + 0.2 \times 0.5 = 0.35 + 0.18 + 0.10 = 0.63$$

**31. Given three identical boxes I, II, and III, each containing two coins.**

**In box I, both coins are gold coins, in box II, both are silver coins and in box III, there is one gold and one silver coin. A person chooses a box at random and takes out a coin. If the coin is of gold, what is the probability that the other coin in the box is also of gold?**

Let  $E_1$ ,  $E_2$ , and  $E_3$  be the events that boxes I, II and III are chosen, respectively. Then  $P(E_1) = P(E_2) = P(E_3) = 1/3$

Also, let  $A$  be the event that 'the coin drawn is of gold'

Then  $P(A|E_1) = P(\text{a gold coin from box I}) = 2/2=1$

$P(A|E_2) = P(\text{a gold coin from box II}) = 0$

$P(A|E_3) = P(\text{a gold coin from box III}) = 1/2$

Now, the probability that the other coin in the box is gold

= the probability that a gold coin is drawn from box I.

=  $P(E_1|A)$

By Baye's theorem, we know that

$$P(E_1|A) = \frac{P(E_1)P(A|E_1)}{P(E_1)P(A|E_1) + P(E_2)P(A|E_2) + P(E_3)P(A|E_3)} = \frac{\frac{2}{3} * 1}{\frac{2}{3} * 1 + \frac{1}{3} * 0 + \frac{1}{3} * \frac{1}{2}} = \frac{2}{3}$$

1.

One of Sam's two children is a female. How likely is it that the other child is a girl as well?

Males and females in the world are roughly equal in number, you can presume.

- 0.75
- 0.333
- 0.25
- 0.50

2.

Two fair six-sided dice are rolled. What is the likelihood that the first roll results in a 3 and the second roll does not result in a 6?

- $3/17$
- $5/36$
- $1/18$

2/9

1/3

3.

Roll a tetrahedral die twice and think about it. What is the likelihood that the first roll's number will be clearly higher than the second roll's number?

Note: A tetrahedral die only has four sides (1, 2, 3, and 4).

1/2

7/16

3/8

3/16

4.

Three rolls are made using an impartial cubic die marked with 1, 2, 2, 3, 3, 3. Having a total score of 4 or 6 is likely to occur.

40/216

25/108

30/108

None

5.

Three classes of equal size are created by randomly dividing a group of 60 pupils. Each division has an equal likelihood. There are two students in that group named Jack and Jill. How likely is it that Jack and Jill will be placed in the same class?

16/58

3/17

1/7

19/59

6.

Red flowers are created when a red and a white flower cross-fertilize 25% of the time. Five pairs of red and white flowers are now cross-fertilized, resulting in five offspring. What is the likelihood that none of the five kids will have red flowering plants?

26.9%

33.8%

23.7%

26.5%

7.

There are 38 slots on a roulette wheel, including 2 green, 18 red, and 18 black. You bet on red in each of your five games. What is your chance of winning all five games?

0.0378

0.0373

0.0525

0.0238

8.

There are 4 purple marbles, 8 red marbles, and 5 white marbles in a bag. What is the chance of not obtaining a purple marble if we pick a marble at random?

0.23

0.77

0.44

0.39

9.

Consider that you sell sandwiches. 70% of individuals pick eggs, while the remaining 30% chose chicken. What is the likelihood that the following three customers will purchase 2 egg sandwiches?

- 0.033
- 0.44
- 0.027
- 0.153

10.

What is the likelihood that two dice will be thrown together and one will have an even number and the other an odd number?

- $1/4$
- $1/2$
- $2/5$
- $3/4$

#### QUES 1- WHAT DOES MEAN AND STANDARD DEVIATION TELL YOU ABOUT ANY DISTRIBUTION?

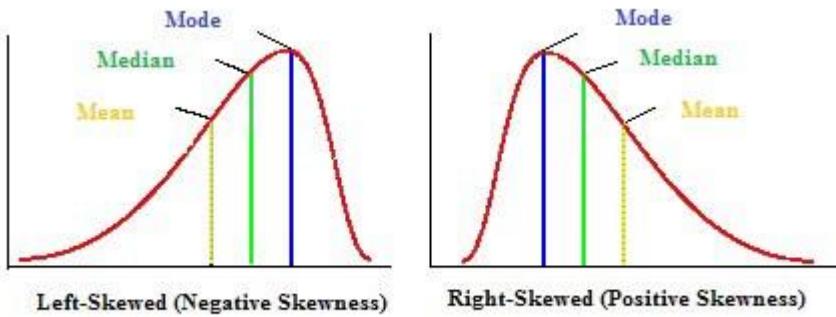
Ans- Mean tell us about the central value of distribution, or where the central value of the distribution lies, Standard deviation tell you about the spread of the distribution.

#### QUES 2- WHAT IS KURTOSIS AND SKEWNESS?

Ans- Skewness is the measure of assymetry or it can be define as How dissimilar is the distribution from the Normal Distribution.

**RIGHT SKEW-** Its simply means outliers are to the right. Here  
**mean>median>mode**

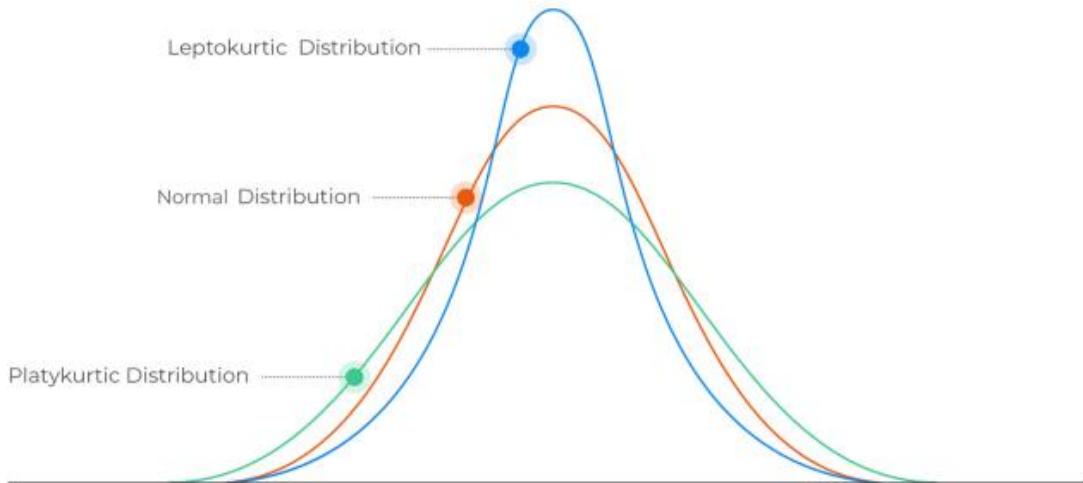
**LEFT SKEW-** It simply means outliers are to the left. Here  
**mode>median>mean**



**Kurtosis-** It is defined as how heavy the tail of distribution differs from the tail of normal distribution.



### Kurtosis



**QUES 3- HOW TO DO STANDARD NORMAL VARIATE(z) AND STANDARIZATION?**

**Ans-** Lets see what is z, lets say z is a gaussian distributed random variable with mean 0 and standard deviation 1.

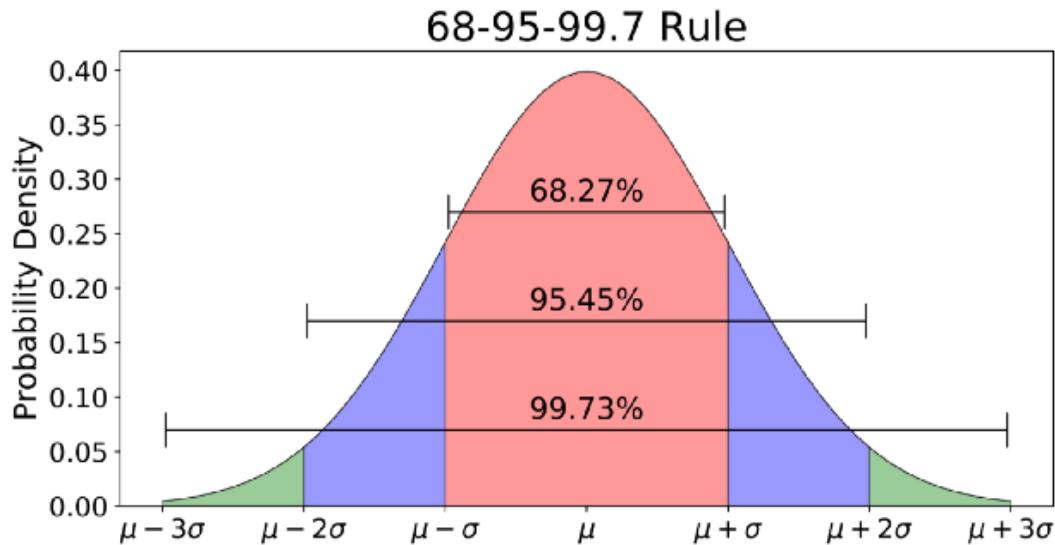
Now lets say we have a gaussian distributed random variable  $X$  with mean ( $\mu$ ) and standard deviation( $\sigma$ ) , now we know  $X$  can take various values so lets say  $x$  can take  $(x_1,x_2,x_3,x_4,x_5)$

Now if we subtract the mean( $\mu$ ) from every observation of  $X$  and divide it with  $\sigma$ , we will get  $z$ .

### BUT WHY DO WE DO STANDARDIZATION?

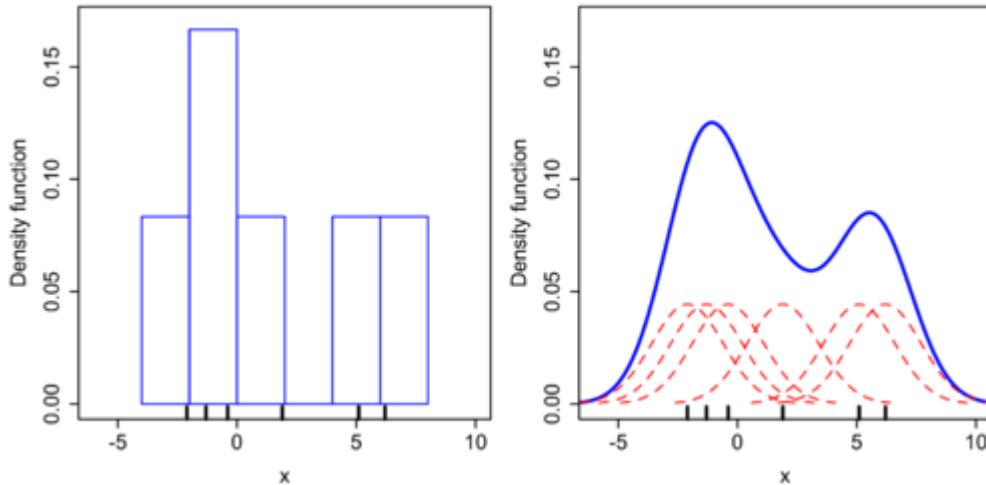
The main reason we do this standardization is the moment we do standarization we know between -1 to 1, 68% of the data lies. -2 to 2 ,85% of the data lies.

So we know some properties of  $z$  so its always good to convert into a standard normal distribution.



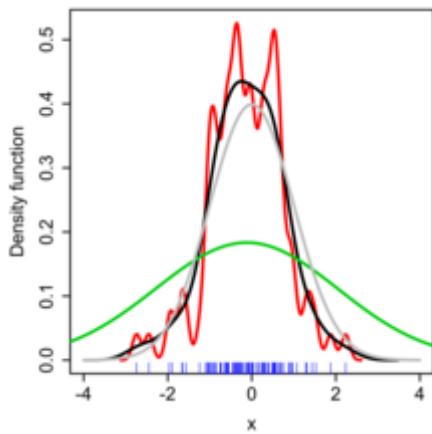
### QUES 4- WHAT IS KERNEL DENSITY ESTIMATION?

Ans- So it is the process of smoothening the histograms to probablity density function.



From the image above we can see that a histogram is converted to a density function so how do we do it, at every point lets say “5” in histogram we build a gaussian kernel (see red lines in density function) by making 5 as a mean , we will repeat it with every point in histogram, in the end we just add up all the values occurring at a single point, for example, at “5” ther are 3 gaussian kernels, we will add up the values of these kernel to get the pdf.

Now what about the standard deviation of every gaussian kernel(as we fixed mean already) so here standard deviation is also called bandwidth.



So if we make bandwidth too small the it would appear as Red line above, if we make bandwidth make too big, it would be very flat liek green lien above, if we make it normal, it will look like Black.

## QUES 5- IMPORTANCE OF SAMPLING THEOREM AND CENTRAL LIMIT THEOREM?

Ans-

### Sampling Theorem

It simply says lets say  $X$  is any random distribution not necessarily gaussian, lets say we take random sample of size  $n$  lets say 30 ,we call it  $(s_1)$ , again we will take a random sample of size  $n$ , we call it  $(s_2)$ , lets say we take  $m$  samples like this so last sample would be  $(s_m)$

if we will take mean of all the samples , lets say for  $s_1$  sample,  $x_1'$  is the mean, for  $s_2$ ,  $x_2'$  is the mean and so on.

$x_1', x_2', x_3', \dots, x_m'$  (mean values of all the samples)

Distribution of  $x_i$ 's is called the sampling distribution of sampling means.

Suppose you have a random variable that has a population mean,  $\mu$ , and a population standard deviation,  $\sigma$ . If a sample of size  $n$  is taken, then the sample mean,  $\bar{x}$  has a mean  $\mu_{\bar{x}} = \mu$  and standard deviation of  $\sigma_{\bar{x}} = \sigma/\sqrt{n}$ . The standard deviation of  $\bar{x}$ 's lower because by taking the mean you are averaging out the extreme values, which makes the distribution of the original random variable spread out.

This we will use in Central Limit Theorem.

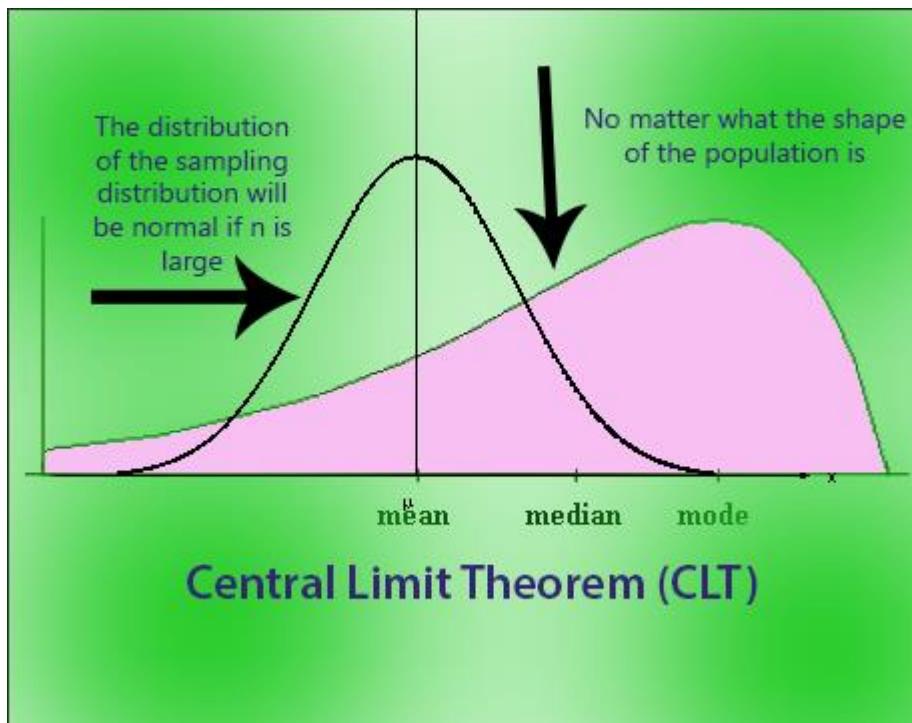
## CENTRAL LIMIT THEOREM

**Theorem states — Suppose a random variable X (population distribution) with a finite mean and standard deviation forms any distribution. If a sample of size  $n$  is taken, then the sample mean,  $\bar{x}$ , becomes normally distributed as  $n$  increases.**

Lets take a random variable X(population distribution)with any distribution, with a finite mean and standard deviation, (pareto distribution had infinite mean and standard deviation) , and we take m sample of size n, lets say

$s_1, s_2, s_3 \dots s_m$  and then we took the mean of all the sample  $x'_1, x'_2, x'_3, \dots, x'_m$ , central limit theorem says if we plot the distribution of these means it will tends to form NORMAL DISTRIBUTION with mean equal to population distribution and variance will be (variance of population/ $n^2$ ).

**Lets say  $m=1000$  and  $n=30$  so by just looking into 30k data points, we are able to estimate the whole population mean and population variance thats make it the most fundamental Theorem.**



## QUES 6- IMPORTANCE OF Q-Q PLOT?

Ans 6- There are various things to check if your distribution is gaussian or not,  
TWO MOST USED TECHNIQUES ARE

1. Q-Q PLOT(QUANTILE QUANTILE PLOT)
2. KS TEST( WE WILL SEE LATER)

So how do we plot Q-Q plot?

So lets assume we have a random variable X and we take 500 observations out of them, lets say  $x_1, x_2, \dots, x_{500}$ .

HERE WE DO NOT KNOW THE DISTRIBUTION OF X, AT THE END OF QQ PLOT WE SHOULD KNOW IS IT NORMAL DISTRIBUTED OR NOT.

STEPS TO FOLLOW

1. Sort  $x_i$ 's in ascending order and find percentile

(if you dont know how to find percentile or what is exactly percentile, lets assume i have 100 values and i sort them into ascending order.

$X = \{x_1, x_2, x_3, \dots, x_{100}\}$ , here  $x_1 < x_2 < \dots < x_{100}$

In this set, lets say i am ranking each value from 1 to 100 so first value will get rank 1 and the last value will get rank 100.

I can say that below the value of  $x_{10}$  or below the value of 10th rank or 10th percentile, ***10% of the values lies*** and above  $x_{10}$  or above 10th percentile, ***90% of the value lies***.

That is the meaning of percentile.

so we will get 100 percentile values for the orginal 500 samples

$x_5, x_{10}, x_{15}, \dots, x_{500}$  {these are the percentile values}

***x5 is the value below which only 1% of the values lies(because here the sample size is 500 not 100)***

***x10 is the value below which only 2% of the value lies***

***x25 is the value below which only 5% of the value lies***

2. Second step is to create a Random Variable Y which has a Normal Distribution and has a mean=0 and standard deviation =1.

Again we will take 500 observation, sort them and find their percentile

so lets say we have  $y_1, y_2, y_3 \dots y_{100}$ (same as we did with our original distribution X)

LET ME REMIND YOU WE DON'T KNOW WHAT IS THE DISTRIBUTION OF X, THAT'S WHY WE ARE USING Q-Q TEST TO DETERMINE WHETHER THE DISTRIBUTION OF X IS GAUSSIAN/NORMAL OR NOT.

3. Third step is to plot QQ plot between X and Y

so we have  $\{x_1, y_1\}, \{x_2, y_2\}, \{x_3, y_3\} \dots \{x_{100}, y_{100}\}$

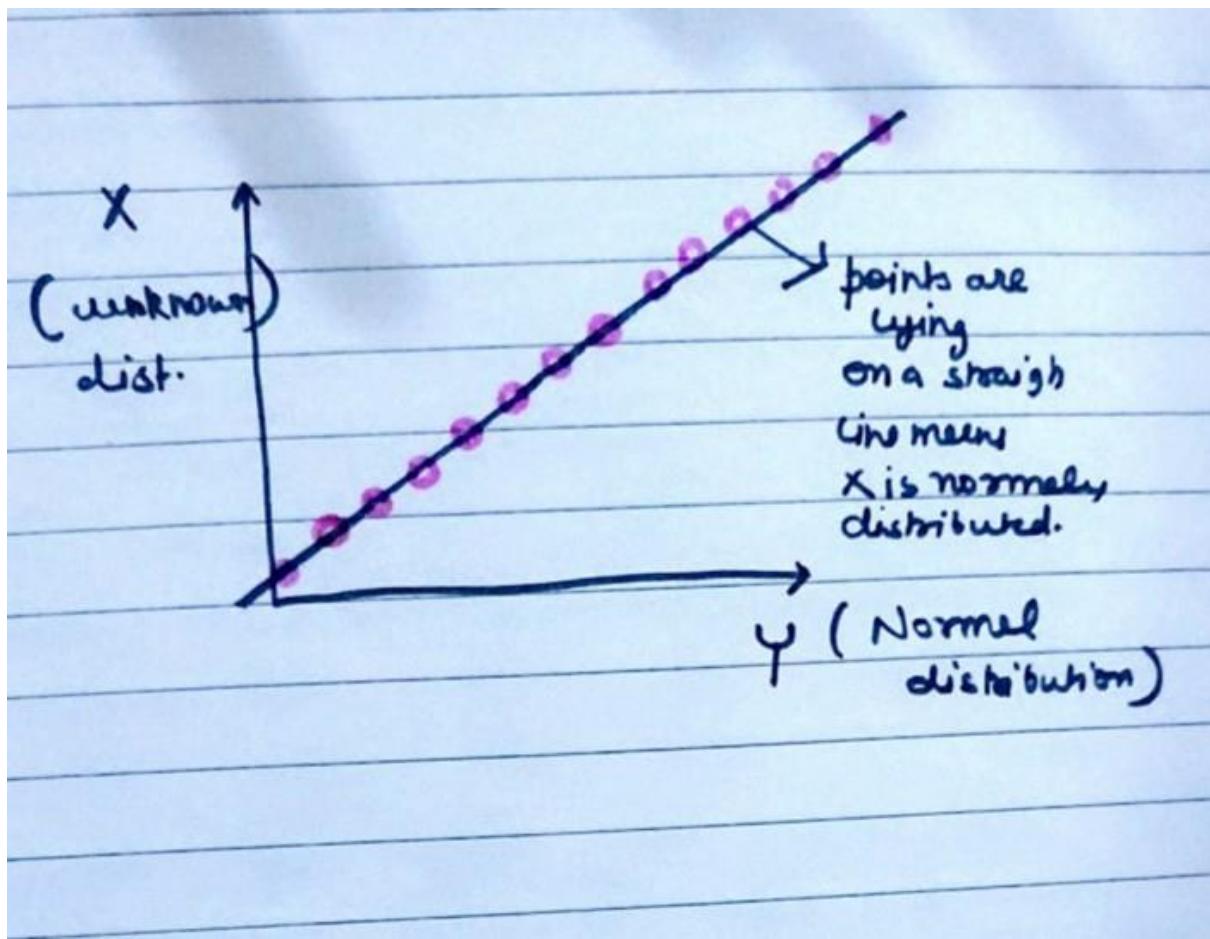
we will plot and if all the points lie in the same line, it means X is NORMALLY DISTRIBUTED but need not have mean= 0 and standard deviation =1.

if all points does not lie in the same line, it means X is not NORMALLY DISTRIBUTED.

In the picture below points are deviating in the end, it means sample quantiles is not normally distributed.

*If number of observations are small , it is hard to interpret QQ plot.*

*Q-Q plots are also used to check if two random variable X and Y have same distribution or not by the same method.*



## CODE

```
import scipy.stats as stats
```

```
import numpy as np
```

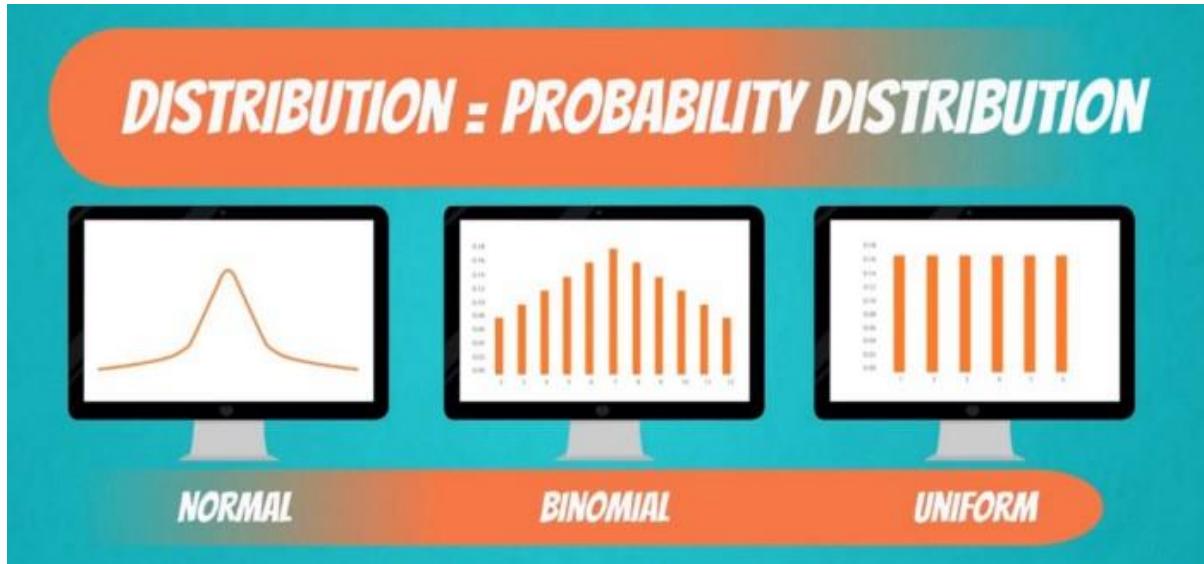
```
import pylab
```

```
stats.probplot(Y, dist= 'norm', plot=pylab)
```

```
pylab.show
```

## QUES 7- WHAT IS UNIFORM DISTRIBUTION?

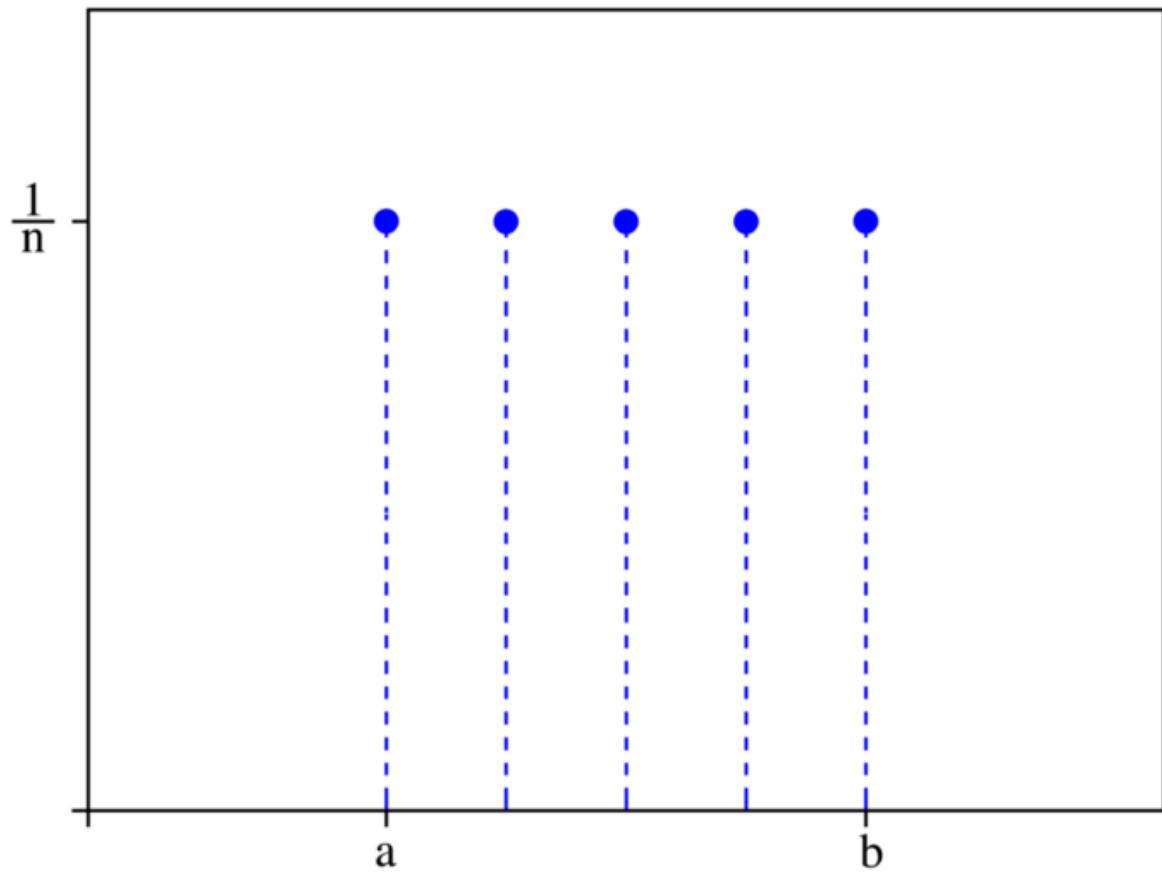
Ans-7 In statistics, a type of probability distribution in which all outcomes are equally likely. A [deck](#) of cards has within it uniform distributions because the likelihood of drawing a heart, a club, a diamond or a spade is equally likely. A coin also has a uniform distribution because the probability of getting either heads or tails in a coin toss is the same.



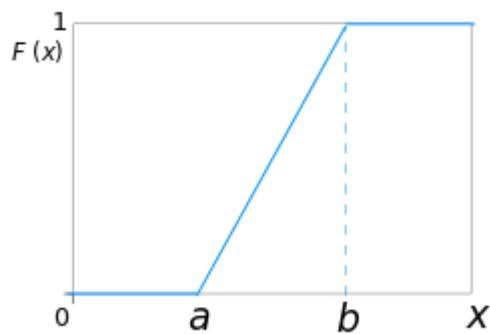
## QUES 8- WHAT IS DISCRETE AND CONTINUOUS UNIFORM DISTRIBUTION?

Ans 8-

### 1. DISCRETE



## 2. CONTINUOUS



## QUES 9-HOW TO RANDOMLY SAMPLE DATA POINTS?

Ans 9-Simple random sampling is the most basic and common type of [sampling method](#) used in quantitative social science research and in scientific research generally. The main benefit of the simple random sample is that each member of the population has an equal chance of being chosen for the study. This means that it guarantees that the sample chosen is representative of the population and

that the sample is selected in an unbiased way. In turn, the statistical conclusions drawn from the analysis of the sample will be [valid](#).

There are multiple ways of creating a simple random sample. These include the lottery method, using a random number table, using a computer, and sampling with or without replacement.

### Lottery Method of Sampling

The lottery method of creating a simple random sample is exactly what it sounds like. A researcher randomly picks numbers, with each number corresponding to a subject or item, in order to create the sample. To create a sample this way, the researcher must ensure that the numbers are well mixed before selecting the sample population.

### Sampling With Replacement

[Sampling with replacement](#) is a method of random sampling in which members or items of the population can be chosen more than once for inclusion in the sample. Let's say we have 100 names each written on a piece of paper. All of those pieces of paper are put into a bowl and mixed up. The researcher picks a name from the bowl, records the information to include that person in the sample, then puts the name back in the bowl, mixes up the names, and selects another piece of paper. The person that was just sampled has the same chance of being selected again. This is known as sampling with replacement.

## Sampling Without Replacement

Sampling without replacement is a method of random sampling in which members or items of the population can only be selected one time for inclusion in the sample. Using the same example above, let's say we put the 100 pieces of paper in a bowl, mix them up, and randomly select one name to include in the sample. This time, however, we record the information to include that person in the sample and then set that piece of paper aside rather than putting it back into the bowl. Here, each element of the population can only be selected one time.

## QUES 10- EXPLAIN BERNOULLI AND BINOMIAL DISTRIBUTION?

Ans 10-

**BERNOULLI DISTRIBUTION-** This distribution is used when you have two outcomes, probability of getting one outcome is  $p$  and probability of getting another is  $1-p$ . This distribution is a discrete distribution.

## Bernoulli distribution

The Bernoulli distribution is the “coin flip” distribution.

X is Bernoulli if its probability function is:

$$X = \begin{cases} 1 & w.p. \quad p \\ 0 & w.p. \quad 1-p \end{cases}$$

X=1 is usually interpreted as a “success.” E.g.:

X=1 for heads in coin toss

X=1 for male in survey

X=1 for defective in a test of product

X=1 for “made the sale” tracking performance

21

**BINOMIAL DISTRIBUTION**-A binomial distribution can be thought of as simply the probability of a SUCCESS or FAILURE outcome in an experiment or survey that is repeated multiple times. The binomial is a type of distribution that has **two possible outcomes** (the prefix “bi” means two, or twice). For example, a coin toss has only two possible outcomes: heads or tails and taking a test could have two possible outcomes: pass or fail.

- The first variable in the binomial formula, n, stands for the number of times the experiment runs.
- The second variable, p, represents the probability of one specific outcome.

## Binomial Distribution Formula

$$P(x) = \binom{n}{x} p^x q^{n-x} = \frac{n!}{(n-x)! x!} p^x q^{n-x}$$

where

$n$  = the number of trials (or the number being sampled)

$x$  = the number of successes desired

$p$  = probability of getting a success in one trial

$q = 1 - p$  = the probability of getting a failure in one trial

## QUES 11- WHAT IS CHEBYSHEV'S INEQUALITY?

Ans- So this is a very interesting topic , Now lets say i have random variable X which says about the height of all the students in the school or office or anywhere.

**CASE1-** Lets say we know that the distribution of X and it is gaussian distributed.

Now when we know it is gaussian distributed we know gaussian distribution follows 68%, 95% and 99.7% rule, means 68% of the total data lies between first standard deviation, 95% of the total data lies between second standard deviation and 99.7% of the total data lies between third standard deviation.

We can easily plot a cdf of the data and can answer any question, for example lets say we know the mean=150cm and standard deviation=10cm, so by this rule 95% of the total data lies between second standard deviation.

$p(u-2\sigma < X < u+2\sigma) = 95\%$ , means 95% of the total heights of the people lies between  $[130 < X < 170]$ .

**CASE2-** What if we do not know the distribution, lets take an example, lets say we have a random variable  $X$  which tells us about the salaries of all the people in the country but we do not know the distribution but with central limit theorem we got mean and standard deviation, we have to make sure the mean should be finite and standard deviation must be non zero and finite.

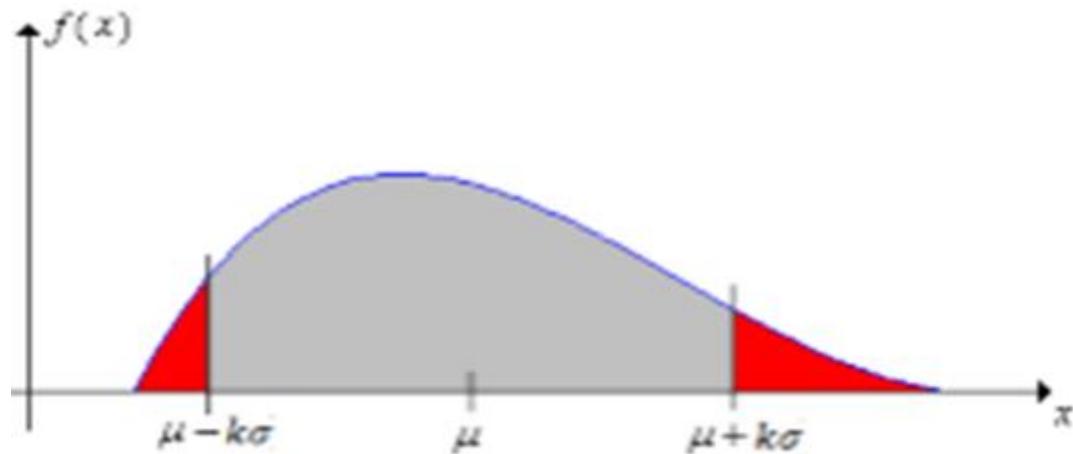
Now the question is can we know what % of salaries lies with second standard deviation which will be  $p(u-2\sigma < X < u+2\sigma)$ .

Lets say  $u=40k$  and  $\sigma=10k$ , now can we know what % of individuals have a salary in range of  $[20k, 60k]$  which is just under second standard deviation.

Here comes CHEBSEV'S INEQUALITY, IT SAYS,

$$P(|X-u| \geq k\sigma) \leq 1/k^2$$

where  $k$  is a constant value



$$P(X \geq u+k\sigma \text{ AND } X \leq u-2\sigma) \leq 1/k^2$$

It can be written as

$$P(u-2\sigma \leq X \leq u+2\sigma) > 1-(1/k^2)$$

Now we can easily answer that salary question because,  $u-2\sigma=20$  and  $u+2\sigma=60$ , from this we got  $k=2$ , so from this

$$P(20 < X < 60) > 1-(1/4)$$

$P(20 < X < 60) > 0.75$ , means atleast 75% of the people salary lies between this region.

## QUES 12- EXPLAIN BOX COX TRANSFORMATION?

**Ans 12-**

**So Box Cox transformation is that mathematical trick which converts Pareto distribution to Gaussian Distribution.**

**Lets understand how Box Cox works.**

Lets say we have a Pareto distribution( $X$ ) and its data points are  $x_1, x_2, x_3, \dots, x_n$

**Step1**

$$\text{boxcox}(X) = \lambda$$

So basically you will be giving “ $n$ ” observations of  $x$  to box-cox and it will give you  $\lambda$ .

Now how box cox will give you lamda is involves a lot of mathematics and it is not necessary to get into that maths.

so lets assume Box Cox is a box in which n observations goes inside and lamda comes outside.

## STEP 2

$$\textcircled{2} \quad y_i = \begin{cases} \frac{x_i^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0 \\ \ln(x_i) & \text{if } \lambda = 0 \end{cases}$$

$\forall i \in 1:n$

(gaussian distribution)      so if  $\lambda = 0$        $y_i \sim \text{log normal}$

Now from the above picture, we can clearly understand, if lamda we got in step 1 is "0" then just taking log of ( $x_i$ ) will give us Gaussian Distribution otherwise we have to use the formula in the picture above.

## CODE FOR BOX COX

```
scipy.stats.boxcox(X, lambda=“ ”)
```

- 1) What is Statistics?

Statistics is a discipline that concerns the study of collection, organization, analysis, interpretation, and presentation of data. Statistics study is generally used in scientific, industrial, and social problems to understand the statistical population or a statistical model of the related data. For example, to get the population statistics, we can use diverse it into the groups of people or objects such as "all people living in a country".

Statistics is the study of every aspect of data, including the planning of data collection in terms of the design of surveys and experiments.

---

## 2) What are the different types of Statistics?

There are mainly two types of Statistics:

- Descriptive statistics
- Inferential statistics

### **Descriptive Statistics**

Descriptive statistics is a type of statistics where data is summarized through the given observations. The summarization is done from a population sample using parameters such as the mean or standard deviation. Descriptive statistics provides a way to organize, represent and describe a collection of data using tables, graphs, and summary measures. For example, a collection of people in a city using specific services such as the internet or television channels.

**The descriptive statistics can be categorized into the following four different categories:**

- Measure of frequency
- Measure of position
- Measure of dispersion
- A measure of central tendency

### **Inferential Statistics**

Inferential statistics is a type of statistics used to interpret the meaning of descriptive statistics. These statistics are used to conclude the data that depends on random variations such as observational errors, sampling variation, etc. Once we have collected, analyzed, and summarized the data, we use these statistics to describe the meaning of the collected data.

In this method, we use the information collected from a sample to make decisions, predictions, or inferences from a population. It also facilitates us to give statements that go beyond the available data or information.

---

### 3) What is the key difference between data and statistics?

In general, people often use the terms "data" and "statistics" interchangeably, but there is a key difference between them. Data can be specified as the individual pieces of factual information recorded and used for analysis. In other terms, data is raw information from which statistics are created. On the other hand, statistics are the results of data analysis, its interpretation, and presentation.

In other words, we can say that statistics is a process of some computation to provide some understanding of what the data means. Statistics are generally presented in the form of a table, chart, or graph. For research purposes, we require both statistics and data frequently. Statistics are often reported and used by government agencies. For example, unemployment statistics, educational literacy statistics, etc. These types of statistics are called "statistical data".

---

### 4) What are the main things you should know before studying data analysis?

Following are the four main things you should know before studying data analysis. These things are:

- Descriptive statistics
  - Inferential statistics
  - Distributions (normal distribution / sampling distribution)
  - Hypothesis testing
- 

### 5) What are the four different types of data statistics?

Data statistics can be divided into mainly two categories:

- Qualitative data
- Quantitative data

Later, these can be subdivided into 4 types of data where nominal data and ordinal data come under qualitative data, and interval and ratio data come under quantitative data.

**Qualitative data:** Qualitative data is a set of information that cannot be measured in the form of numbers. It is also called categorical data. It normally contains words, narratives, etc., that we label with names. It mainly focuses on the qualities of things in data, and after the qualitative data analysis, the outcome comes in featuring keywords, extracting data, and ideas elaboration.

For example, a person's hair color such as black, brown, red, blonde, etc. The qualitative data can be divided into two subcategories: nominal and ordinal.

- **Nominal Data:** The nominal data are used to label variables with no quantitative value and no order. It doesn't change the meaning if you change the order of the value, and after that meaning will remain the same. So, you can only observe the nominal data and can't measure.
  - **Ordinal Data:** The ordinal data is very much similar to the nominal data but not in the case of an order. The ordinal data is ordered, and their categories can be ordered like 1st, 2nd, etc.
- 

## 6) What is the Central Limit Theorem? Why is it used?

Central Limit Theorem is the most important part of statistics. It specifies that the distribution of a sample from a population that consists of large sample size will have its mean normally distributed. In other words, we can say that it will not affect the original population distribution even if the sample size gets larger, regardless of the population's distribution. Generally, it is considered sufficient for the CLT to hold if the sample sizes are equal to or more than 30.

Central Limit Theorem or CTL is mainly used to calculate confidence intervals and hypothesis testing. It also facilitates us to calculate the confidence intervals accurately. For example, if you want to calculate the average height of the people in the world, you have to take some samples from the general population, which serves as the data set. Here, it is very difficult or nearly impossible to get data regarding the height of every person in the world, so you have to calculate the mean of your sample data.

By multiplying the get data set several times, you will get the mean and their frequencies which you can plot on the graph and create a normal distribution

curve. Here, you will get a bell-shaped curve that closely resembles the original data set.

---

## 7) What do you understand by observational and experimental data in Statistics?

Observational data is a type of data obtained from observational studies. In observational data, we observe the variables to see if there is any correlation between them. On the other hand, experimental data is a type of data that is collected from experimental studies. Here, we hold certain variables as constant to see if there is any discrepancy raised in the working.

---

## 8) How can you assess the statistical significance of an insight?

We can use hypothesis testing to determine the statistical significance of an insight. Here, we state the null and alternate hypotheses and then calculate the p-value. Once the p-value is calculated, the null hypothesis is assumed true, and the values are determined. To ensure the value's correctness, we compare it with the alpha value, which denotes the significance, which is tweaked. If the p-value is less than the alpha value, the null hypothesis is rejected, otherwise considered. This is used to ensure that the result obtained is statistically significant.

---

## 9) What is the difference between data analysis and machine learning?

Following is a list of key differences between data analysis and machine learning:

Data Analysis	Machine Learning
Data analysis is a process where we inspect, clean, transform, and model data to find useful information, informing conclusions, and support decision-making, which can enhance the decision-making process.	Machine learning is mainly used to automate entire data analysis workflow to provide deeper, faster, and more comprehensive insights.
Data analysis requires a deep knowledge of coding and basic knowledge of statistics.	On the other hand, machine learning requires basic knowledge of coding and deep knowledge of statistics and business.

<p>We mainly focus on generating valuable insights from the available data in data analysis. Companies use the data analysis process to make better decisions regarding several matters such as marketing, production, etc.</p>	<p>We mainly focus on studying algorithms to improve the overall user experience in machine learning. It is a subset of artificial intelligence that leverages algorithms to analyze huge amounts of data.</p>
<p>Data analysis may require human intervention to inspect, clean, transform, and model data to find useful and trustworthy information.</p>	<p>In machine learning, we use algorithms that learn from data automatically and apply the learned knowledge without human intervention.</p>
<p>The average salary of a data analysis professional in India is less than the salary of a machine learning professional.</p>	<p>The average salary of a machine learning professional in India is more than the salary of a data analysis professional.</p>
<p>A data analysis professional has to deal with data, so they should have deep knowledge of coding and basic knowledge of statistics.</p>	<p>A machine learning professional must know about Deep Learning, Natural Language Processing (NLP), Computer Vision, Analytics Skills, Statistical Analysis, SQL, knowledge of R and Python programming language.</p>

## 10) What is the difference between inferential statistics and descriptive statistics?

Inferential statistics provide information about a sample. It is required to conclude the population. On the other hand, descriptive statistics provide exact and accurate information.

## 11) What is Normality in Statistics?

In Statistics, Normality is behaviour consistent with the usual way of behaving of a person. It is an accepted way of social standards and thinking and behaving similarly to the majority, and generally seen as a good way in this context. According to the situation, it can also be specified as expected and appropriate behaviour.

In the case of psychological statistics, it can also be just being average. It specifies how you adjust to the surroundings, manage or control emotions, work satisfactorily, and build satisfactory, fulfilling, or at least acceptable relationships.

## 12) What are the criteria for Normality?

For any specified behaviour or trait, the criteria for Normality are being average or close to the average. It means the scores falling within one standard deviation above or below the mean is normal. The most average 68.3% of the population is considered normal.

---

## 13) What is the assumption of Normality?

In technical terms, the assumption of Normality states that the sampling distribution of the mean is normal or that the distribution of means across samples is normal. In other words, the assumption of Normality specifies that the mean distribution across samples is normal. This is true across independent samples as well.

---

## 14) What is the main usage of long-tailed distributions? Where are they mainly used?

The long-tailed distributions are the type of distribution where the tail gradually drops off toward the curve's end. They are most widely used in classification and regression problems. The Pareto principle and the product sales distribution are good examples of using long-tailed distributions.

---

## 15) What do you understand by Hypothesis Testing?

In Statistics, Hypothesis Testing is mainly used to see if a certain experiment generates meaningful results. It helps assess the statistical significance of insight by finding the odds of the results occurring by chance. In Hypothesis Testing, the first thing is to know the null hypothesis and then specify it. After that, the p-value is calculated, and if the null hypothesis is true, the other values are also determined. The alpha value specifies the significance, and you can adjust it accordingly.

If the p-value is less than the alpha value, the null hypothesis is rejected, but the null hypothesis is accepted if the p-value is greater than the alpha value. If the null hypothesis is rejected, it indicates that the results obtained are statistically significant.

---

## 16) How can you handle the missing data in Statistics?

There are several ways to handle the missing data in Statistics:

- By predicting the missing values.
  - By assigning the individual or unique values.
  - By deleting the rows which have the missing data.
  - By mean imputation or median imputation.
  - By using the random forests, which support the missing values.
- 

## 17) What do you understand by mean imputation for missing data? Why is it considered bad?

Mean imputation is a way where null values in a dataset are replaced directly with the corresponding mean of the data. It is a rarely used practice nowadays. Mean imputation is considered bad practice because it completely removes the accountability for feature correlation. It also means that the data will have low variance and increased bias that may cause a dip in the model's accuracy, along with the narrower confidence intervals.

---

## 18) What do you understand by six Sigma in Statistics?

In Statistics, six Sigma is a quality control method used to produce an error or defect-free data set. In this method, the standard deviation is known as Sigma or  $\sigma$ . The more the standard deviation is, the less likely that process would perform with accuracy and causes a defect. A six sigma model works better than  $1\sigma$ ,  $2\sigma$ ,  $3\sigma$ ,  $4\sigma$ ,  $5\sigma$  processes and is reliable enough to provide a defect-free work. If you get the outcome of the process 99.99966% error-free, it is considered six Sigma.

---

## 19) What is an exploratory data analysis in Statistics?

In Statistics, an exploratory data analysis is the process of performing investigations on data to understand the data better. In this process, the initial investigations are done to determine patterns, spot abnormalities, test hypotheses, and check if the assumptions are correct.

---

## 20) What do you understand by selection bias?

In Statistics, the selection bias is a phenomenon that involves the selection of individual or grouped data in a way that is not considered to be random. Randomization plays a vital role in performing analysis and understanding the model functionality better. If we don't achieve the correct randomization, the resulting sample will not accurately represent the population.

---

## 21) What is an outlier in Statistics? How can you determine an outlier in a data set?

In Statistics, outliers are data points that usually vary largely as compared to other observations in the dataset. Based on the learning process, an outlier can decrease a model's accuracy and decrease its efficiency sharply.

**We can determine an outlier by using two methods:**

- Standard deviation/z-score
  - Interquartile range (IQR)
- 

## 22) What do you understand by an inlier in Statistics?

An inlier is a data point within a data set that lies at the same level as the rest of the data set. It isn't easy to find an inlier in the dataset compared to an outlier as it requires external data.

Similar to outliers, inliers also reduce the model accuracy. Unlike outliers, inlier is hard to find and often requires external data for accurate identification. So, it is usually an error, and we have to remove it to improve the model accuracy. This is mainly done to maintain the model accuracy at all times.

---

## 23) What do you understand by KPI in Statistics?

KPI is an acronym that stands for Key Performance Indicator. A KPI is a quantifiable measure to understand if we can achieve the goal or not. KPI is a reliable metric that is generally used to measure the performance level of an

organization or individual for the objectives. An example of KPI in an organization is the expense ratio.

---

#### 24) What are the different types of selection bias in Statistics?

There are several types of selection bias in Statistics:

- Attrition selection bias
  - Observer selection bias
  - Protopathic selection bias
  - Time intervals selection bias
  - Sampling selection bias
- 

#### 25) What is the law of large numbers in Statistics?

In Statistics, the law of large numbers is used to specify that if we increase the number of trials in an experiment, we will get a positive and proportional increase in the results coming closer to the expected value. For example, if you roll a six-sided dice three times and check the probability, you will see that the expected value obtained is far from the average value. On the other hand, if you roll a dice a large number of times, you will obtain the average result closer to the expected value, which is 3.5 in this case. This is a good example of the law of large numbers in Statistics.

---

#### 26) What is root cause analysis in Statistics? Can you give an example to explain it?

As the name suggests, root cause analysis is a method used in Statistics to solve problems by first identifying the root cause of the problem.

For example, If you see that the higher crime rate in a city is directly associated with the higher sales in a black-coloured shirt, it means that they have a positive correlation. However, it does not mean that one causes the other. Correlation is always tested using A/B testing or hypothesis testing.

---

## 27) What are some important properties of a normal distribution in Statistics?

Normal distribution is used to specify the data, which is symmetric to the mean, and data far from the mean occurred less frequently. It appears as a bell-shaped curve in graphical form, which is symmetrical along the axes. In Statistics, a normal distribution is also known as Gaussian distribution. It appears as a bell-shaped curve in graphical form, which is symmetrical along the axes. In Statistics, a normal distribution is also known as Gaussian distribution.

### A normal distribution consists of the following properties:

- **Symmetrical:** The symmetrical property specifies the shape changes with that of parameter values.
  - **Unimodal:** As the name specifies, this property has only one mode.
  - **Mean:** This property is used to measure the central tendency.
  - **Central tendency:** It specifies that the mean, median, and mode lie at the centre, which means they are all equal, and the curve is perfectly symmetrical at the midpoint.
- 

## 28) In which cases median is a better measure than the mean?

In the cases where there are a lot of outliers that can positively or negatively skew data, we prefer the median as it provides an accurate measure in this case of determination.

---

## 29) What is the 'p-value' in Statistics? How would you describe it?

In Statistics, a p-value is a number that indicates the likelihood of data occurring by a random chance. It is calculated during hypothesis testing. If the p-value is 0.5 and is less than alpha, we can conclude that there is a probability of 5% that the experiment results occurred by chance. In other words, we can say that 5% of the time, we can observe these results by chance.

---

## 30) How can you calculate the p-value using MS Excel in Statistics?

In Excel, the p-value is called probability value. It is used to understand the statistical significance of a finding. The main use of the p-value is to test the

validity of the Null Hypothesis. If the Null Hypothesis is not seemed according to the p-value, we have to believe that the alternative hypothesis might be true. P-value allows us to determine whether the provided results are caused by chance or whether we are testing two unrelated things. So, the p-value is considered an investigator and not a judge.

It is a number between 0 and 1, but it is generally denoted in percentages. If the p-value is 0.05, it will be denoted as 5%. A smaller p-value leads to the rejection of the Null Hypothesis.

**Following is the formula to calculate the p-value using MS Excel in Statistics:**

$$\text{p-value} = \text{tdist}(x, \text{deg\_freedom}, \text{tails})$$

The p-value is expressed in decimals in Excel. Follow the steps given below to calculate the p-value in Excel:

- First, find the Data tab.
  - After that, click on the data analysis icon on the Analysis tab.
  - Select Descriptive Statistics and then click OK.
  - Select the relevant column.
  - Input the confidence level and other variables.
- 

### 31) What do you understand by DOE in Statistics?

DOE is an acronym that stands for the Design of Experiments in Statistics. In this process, we design a task that describes the information and the change of the same based on the changes to the independent input variables.

---

### 32) What do you understand by Covariance?

Covariance is a measure that specifies how much two random variables vary together. It indicates how two variables move in sync with each other. It also specifies the direction of the relationship between two variables. There are two types of Covariance: positive and negative Covariance. The positive Covariance specifies that both variables tend to be high or low simultaneously. On the other hand, the negative Covariance specifies that one tends to be below when the other is high.

---

### 33) What is the Pareto principle used in Statistics?

The Pareto principle used in Statistics is also called the 80/20 principle or 80/20 rule. This principle specifies that 80 per cent of the results are obtained from 20 per cent of the causes in an experiment.

For example, you will have observed in your real life that 80 per cent of the wheat comes from the 20 per cent of the wheat plants on a farm.

---

### 34) What type of data does not have a log-normal or Gaussian distribution?

The exponential distributions types of data do not have a log-normal distribution or a Gaussian distribution. Any type of categorized data will not have these distributions as well.

For example, duration of a phone call, time until the next earthquake, etc.

---

### 35) What is IQR in Statistics? How can you calculate the IQR?

IQR is an acronym that stands for interquartile range. It is a measurement of the "**middle fifty**" in a data set. The IQR describes the middle 50% of values when ordered from lowest to highest.

**Follow the steps given below to find the interquartile range (IQR) in Statistics:**

- First, find the median (middle value) of the lower and upper half of the data.
- These values are quartile 1 (Q1) and quartile 3 (Q3).
- The IQR is the difference between Q3 and Q1.

$$\text{IQR} = Q3 - Q1$$

Q3 is the third quartile (75 percentile), and Q1 is the first quartile (25 percentile).

---

### 36) What do you understand by the five-number summary in Statistics?

In Statistics, the five-number summary is used to measure five entities covering the entire data range. It is mainly used in descriptive analysis or during the preliminary investigation of a large data set.

**The five-number summary contains the following five values:**

- Low extreme (Min)
  - The first quartile (Q1)
  - Median
  - Upper quartile (Q3)
  - High extreme (Max)
- 

**37) What is the advantage of using the box plot?**

The box plot shows the 5-number summary pictorially. It is mainly used to compare a group of histograms.

---

**38) What is the difference between the 1st quartile, the 2nd quartile, and the 3rd quartile?**

In Statistics, quartiles are used to describe data distribution by dividing the data into three equal portions. In this partition of the data, the boundary or edge of these portions is called quartiles.

**There are three types of quartile:**

- **The lower quartile (Q1)** specifies the 25th percentile of the data.
  - **The middle quartile (Q2):** It is also called the median and specifies the 50th percentile of the data.
  - **The upper quartile (Q3)** specifies the 75th percentile of the data.
- 

**39) What do you understand by skewness?**

Skewness can be described as a distortion or asymmetry that deviates from a data set's symmetrical bell curve or normal distribution. You can assume it as a degree of asymmetry observed in a probability distribution.

Depending on the varying degrees, skewness can be of two types, i.e. the right (positive) skewness and the left (negative) skewness. Skewness is centred on the mean. If skewness is negative, the data is spread more on the left of the mean than the right. If skewness is positive, the data moves more to the right. A normal distribution (bell curve) shows zero skewness.

---

#### 40) What is the difference between a left-skewed distribution and right-skewed distribution?

The key difference between the left-skewed distribution and the right-skewed distribution is that the left tail is longer than the right side in the left-skewed distribution. Here,  $\text{mean} < \text{median} < \text{mode}$ . On the other hand, the right tail is longer than the right side in the right-skewed distribution. Here,  $\text{mode} < \text{median} < \text{mean}$ .

---

#### 41) What are the different types of data sampling in Statistics?

There are mainly four types of data sampling in Statistics:

- **Simple random:** This data sampling type specifies the pure random division.
  - **Cluster:** The population is divided into clusters
  - **in this data sampling type.**
  - **Stratified:** Data is divided into unique groups in this data sampling type.
  - **Systematical:** This data sampling type picks up every 'n' member in the data.
- 

#### 42) What is Bessel's correction? Why is it used in Statistics?

In Statistics, Bessel's correction is a factor used to estimate the standard deviation of populations from its sample. It causes a less biased standard deviation and is mainly used to provide more accurate results.

---

#### 43) What is the difference between type I vs type II errors?

Type I errors occur when the null hypothesis is rejected, even if true. It is also known as false positive. On the other hand, type II errors occur when the null hypothesis fails to get rejected, even if false. It is also known as a false negative.

---

#### 44) What is the relationship between the significance level and the confidence level in Statistics?

In Statistics, the significance level is the probability of getting a completely different result from the condition where the null hypothesis is true. On the other hand, the confidence level is used as a range of similar values in a population.

We can specify the similarity between the significance level and the confidence level by the following formula:

$$\text{Significance level} = 1 - \text{Confidence level}$$

---

#### 45) What do you understand by the Binomial Distribution formula?

Following is the formula for the Binomial Distribution:

$$b(x; n, P) = nCx * Px * (1 - P)^{n-x}$$

#### Parameter explanation:

- $b$  = It specifies the binomial probability.
  - $x$  = It specifies the total number of "successes" (pass or fail, heads or tails, etc.)
  - $P$  = It specifies the probability of success on an individual trial.
  - $n$  = It specifies the number of trials.
- 

#### 46) What are the examples of symmetric distribution in Statistics?

Symmetric distribution specifies that the data on the left side of the median is the same as the data on the right side of the median.

**Following are the three most widely used examples of symmetric distribution:**

- Normal distribution
  - Uniform distribution
  - Binomial distribution
- 

#### 47) What is the empirical rule in Statistics?

In Statistics, the empirical rule is also known as the 68-95-99.7 rule. It specifies that every piece of data in a normal distribution lies within three standard deviations of the mean.

According to the empirical rule,

- 68% of values fall within one standard deviation of the mean.
  - 95% of values fall within two standard deviations of the mean.
  - 75% of values fall within three standard deviations of the mean.
- 

#### 48) What is the relationship between mean and median in a normal distribution?

Mean and median are equal in a normal distribution. So, if the distribution of a dataset is normal, the mean and median would be the same.

### **What Are Some Key Concepts in Statistics?**

Statistics is the study of data. It's an important field that helps scientists, analysts, and researchers make sense of large amounts of information. There are several key concepts in statistics that can help you better understand what it is and how it works.

We'll discuss four of them here: population and sample, standard deviation, covariance and correlation, and probability.

	Variance	Standard Deviation
Population	$\sigma^2 = \frac{\sum(x_i - \mu)^2}{N}$	$\sigma = \sqrt{\frac{\sum(x_i - \mu)^2}{N}}$
Sample	$s^2 = \frac{\sum(x_i - \bar{x})^2}{N - 1}$	$s = \sqrt{\frac{\sum(x_i - \bar{x})^2}{N - 1}}$

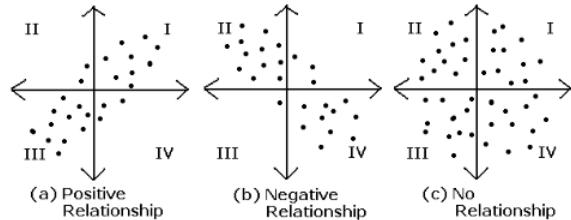
Population and Sample Variance and Standard Deviation

A **population** is a collection of elements that have specific characteristics in common. For example, all the people who live in a particular city are part of that city's population. A **sample** is a subset of the population being studied. It has been selected to reflect the characteristics of the population as a whole but is not necessarily representative of the entire group.

**Standard deviation** measures how far away from the average value for a set of values something is likely to be. It's calculated by taking all those values, finding their mean (average) value, and then calculating their percentage from that mean value.

$$cov(x, y) = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})$$

$$cor(x, y) = \frac{cov(x, y)}{\sqrt{var(x) var(y)}}$$



Covariance and Correlation

**Covariance and correlation** measure how two sets of data relate to each other; they represent whether there's any kind of pattern between them or whether one set causes changes in another set over time by using random variables.

Finally, **probability** represents how likely something might happen given certain conditions or how unlikely it would be given those same conditions (for example: "The probability that it will rain tomorrow is 30%").

## What Are Descriptive Statistics?

## Descriptive Statistics

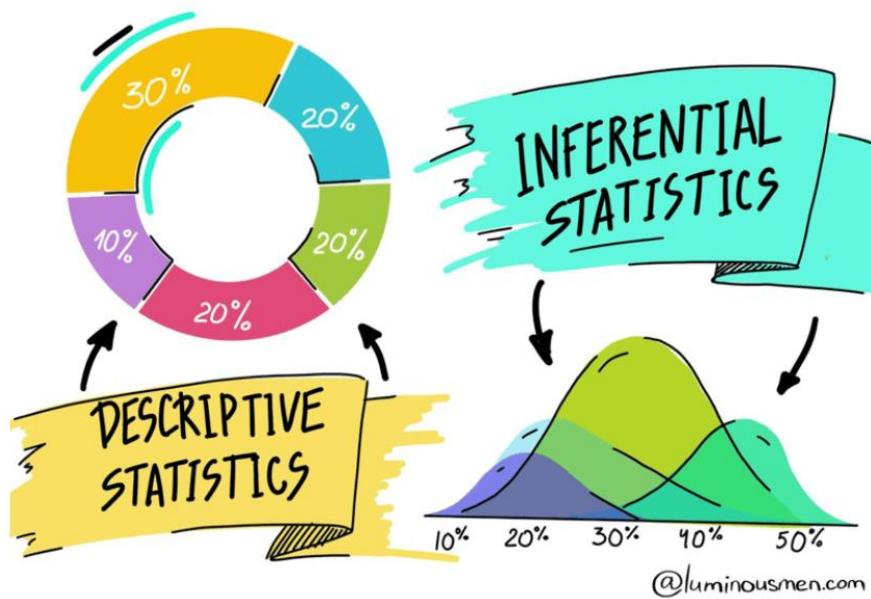


WallStreetMojo

Descriptive statistics are a set of numbers that describe a group or population, and they're usually used to summarize information about a set of data.

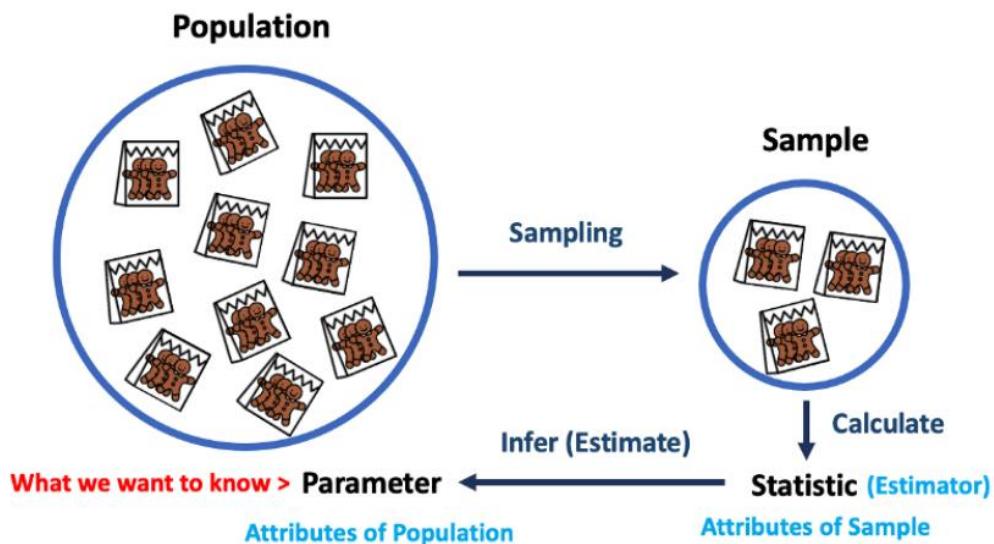
For example, if you're looking at the average height of the male population in your country, you might use descriptive statistics to find out that the mean height is 5'11". Descriptive statistics can be used to describe any kind of data—from test scores to how much money a company makes during a year.

### What Is the Difference Between Inferential Statistics and Descriptive Statistics?



The difference between **inferential statistics** and **descriptive statistics** is that inferential statistics are used to draw conclusions about a population based on the data you've collected. In contrast, descriptive statistics are used to summarize your data.

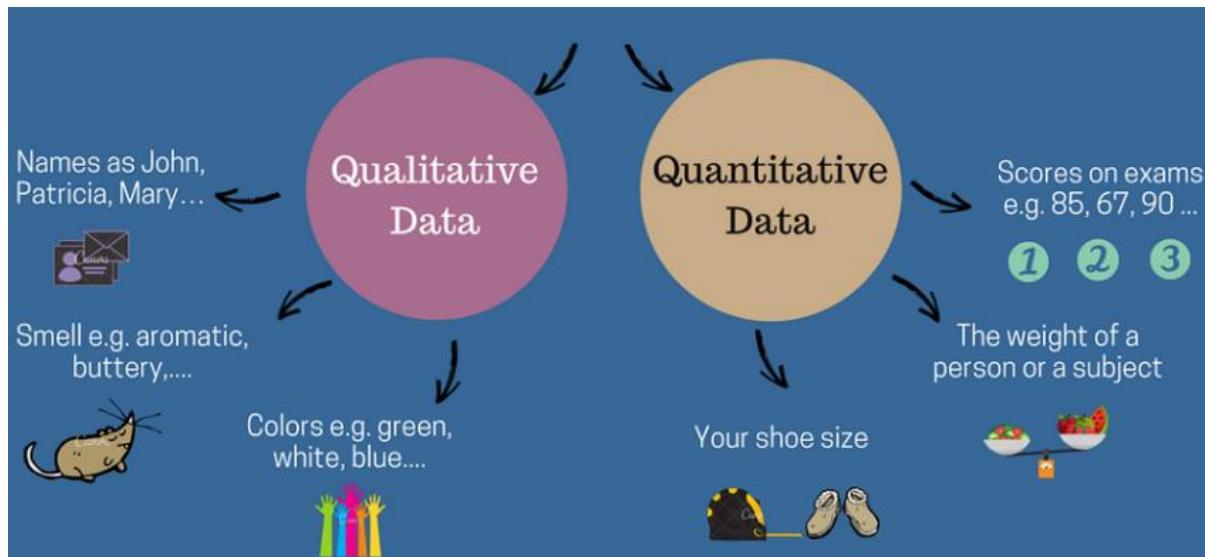
## What Is the Difference Between Population and Sample in Inferential Statistics?



In inferential statistics, the difference between population and sample is that a population is the complete set of objects in a specific category. In contrast, a sample is a subset of that category.

You can think about it like this: if you have a jar full of marbles, the entire contents of the jar are the “population”—that’s all the marbles. But if you randomly remove ten marbles from the jar, those ten marbles are your “sample.”

## What Is the Difference Between Quantitative Data and Qualitative Data?

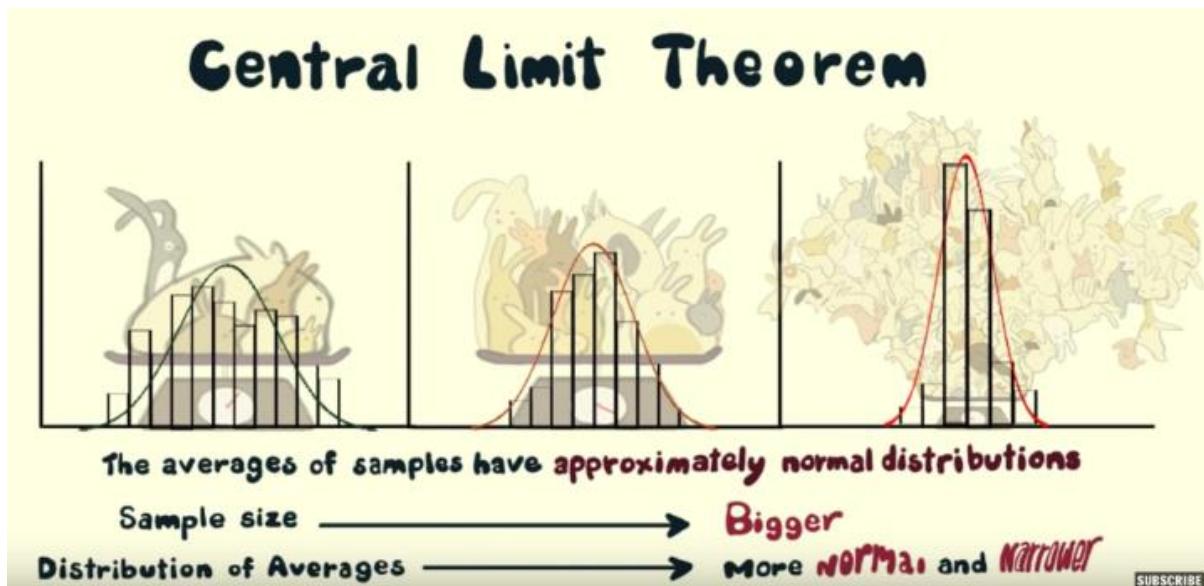


Quantitative data is numerical data that can be measured, counted, and expressed as a percentage. For example, if you have 100 people in a room, how

many of them are women? If you ask them to fill out a survey and write down their age and gender, that would be quantitative data.

Qualitative data is non-numerical information that describes subjective experiences or opinions about an event or topic. Qualitative data can be examined using methods like surveys and interviews. For example, if you wanted to understand how people feel about sports, you might ask them questions like: “What kind of sports do you like?” or “How much time do you spend watching sports?”

### Explain the Central Limit Theorem.



The Central Limit Theorem is a mathematical principle describing how the mean of a large number of samples approaches their population mean as the sample size increases.

It's important because it can be used to test an alternative hypothesis about populations by looking at the means of random samples from those populations.

For example, if you gather 100 samples from a population and find that they all have a mean equal to some number, you can conclude that the population's mean is also equal to that

### What Is Sampling? What Are the Different Sampling Methods? List Some Examples of Sampling Biases.

Sampling is collecting information from a population to make inferences about the whole. It's used in statistical analysis, scientific research, and other fields.

There are many different sampling methods: simple random sampling, stratified random sampling, systematic sampling, cluster sampling, convenience sampling (also known as judgmental or non-probability sampling), and quota sampling. These methods have their strengths and weaknesses that can result in some form of selection bias.

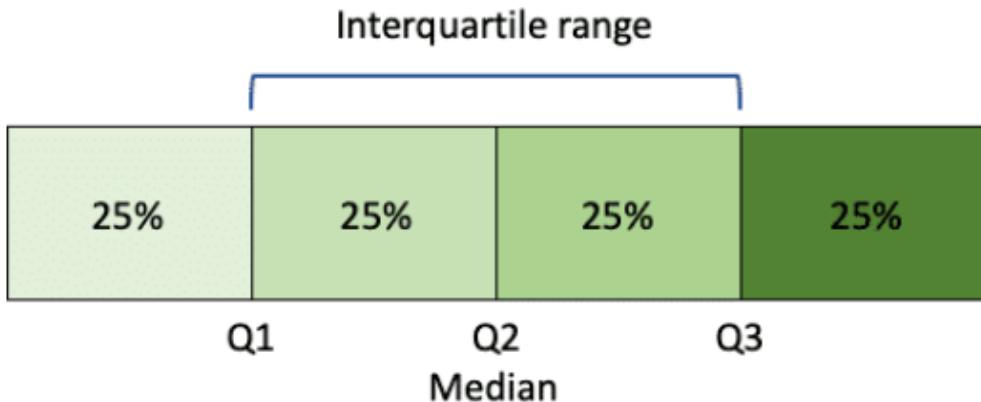
Sampling bias occurs when the sample does not represent the population it is supposed to represent. This can happen if the sample is too small or biased towards certain groups of people who are more likely to answer questions or participate in surveys.



Some examples of biases that can occur in a sample include:

- Self-selection: where people volunteer for a study because they believe they will benefit from it (e.g., they want to win a prize);
- Recruiting at places where people tend to congregate (e.g., bars).

## How Do You Calculate Range and Interquartile Range?

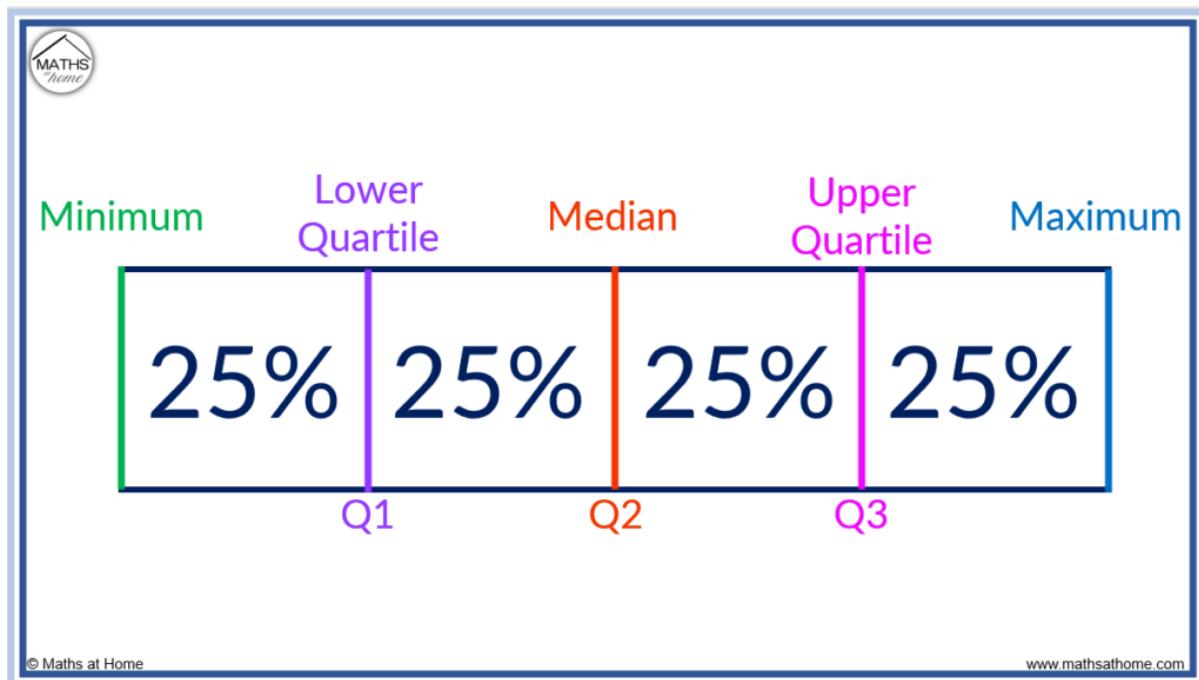


Range and interquartile range are two ways to calculate the spread of data. The range is the difference between the highest and lowest value in a set of data. The interquartile range is the difference between the 75th percentile and 25th percentile of a set of data.

To calculate the interquartile range, first, you need to sort your data from smallest to largest. Then find the 75th percentile by calculating three-quarters of the way across your sorted list (i.e.,  $3/4 = .75$ ). Next, find 25% of your sorted list by calculating one-fourth of the way across your sorted list (i.e.,  $1/4 = .25$ ). Finally, subtract these numbers; this is your interquartile range.

To calculate the range: find the absolute value difference between each number in your list and add all those differences.

### **What's the 5-number Summary, and How Do We Visualize It?**



The five-number summary is a statistical description of a data set. It consists of the smallest value, the largest value, the median, the first quartile, and the third quartile.

The five-number summary can be visualized with box plots or histograms. A box plot shows the normal distribution of data using boxes extending from one quantile (the 25th percentile) to another (the 75th percentile). A histogram shows the distribution by giving equal area to bars representing each value in a data set.

## What Is the Relationship Between Standard Deviation and Variance?

Variance	$\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2$
Standard deviation	$\sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2}$

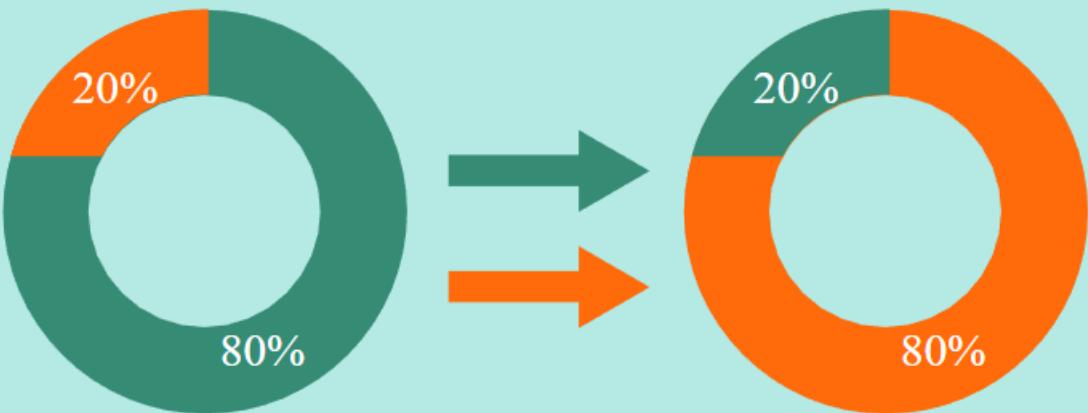
Standard deviation and variance are statistical measures of how values within a [data set](#) are distributed. The standard deviation measures the average distance between each value in the data set and the mean, while variance measures how much each value in the data set varies around its mean.

The standard deviation is always greater than or equal to the variance, regardless of which method is used to calculate either. This is because standard deviation accounts for only one measure of dispersion (distance from the mean), whereas variance accounts for two: distance from the mean and each other.

### Explain the Following:

Pareto Principle

## THE PARETO PRINCIPLE

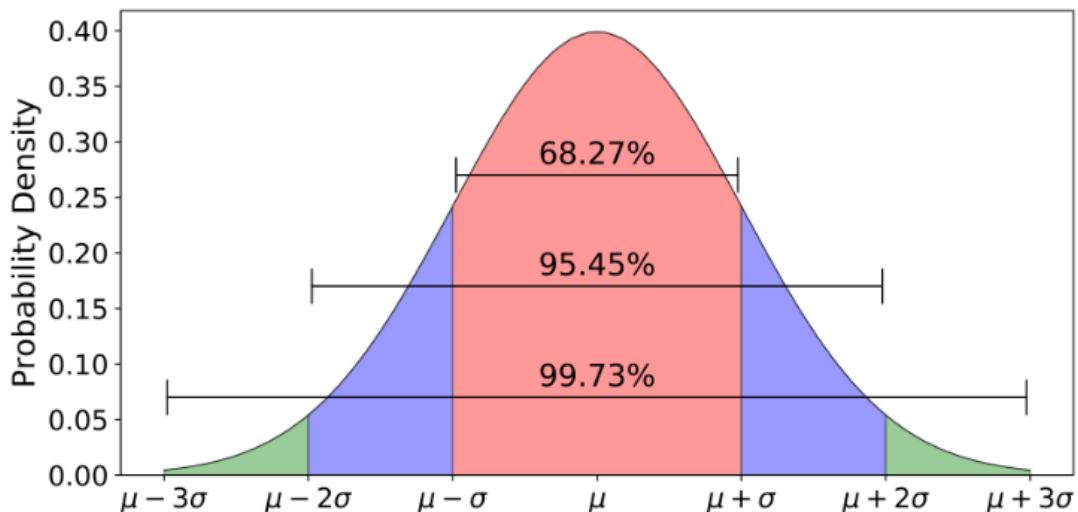


The Pareto Principle, also known as the 80-20 rule, is a principle that states that 20% of causes are responsible for 80% of effects.

The principle was named after Italian economist Vilfredo Pareto, who noticed that 80% of his country's land was owned by 20% of the population. He found this to be true in other places and industries as well—that a small portion of causes (and effects) were responsible for a large portion of what happens.

In statistics, this means that while many variables are at play in any situation, only a few will account for most of the results you see. For example, suppose you wanted to predict how many people will come to an event based on how much money you spend on advertising alone. In that case, you'd need to know which variables accounted for most of the total amount spent on advertising (e.g. which variables had the most significant impact).

Three-Sigma Rule



The Three-Sigma Rule is a statistical concept that states that if you have a sample of data and want to determine the probability that the average of your sample will fall within three standard deviations of the actual value, you must calculate the appropriate z-score using a normal distribution.

For example, we have a sample of five values, and they are 2, 5, 7, 10, and 11. The average of those values is 6.2. Let's say we want to determine our chances that this average will fall within three standard deviations of the true value, which would be between 6.5 and 7.5. We can do this by calculating  $Z = (6 - 6.5) / 0.5 = -0.1$ , then plugging this into our calculator to find that our chances are about 70%.

### Law of Large Numbers

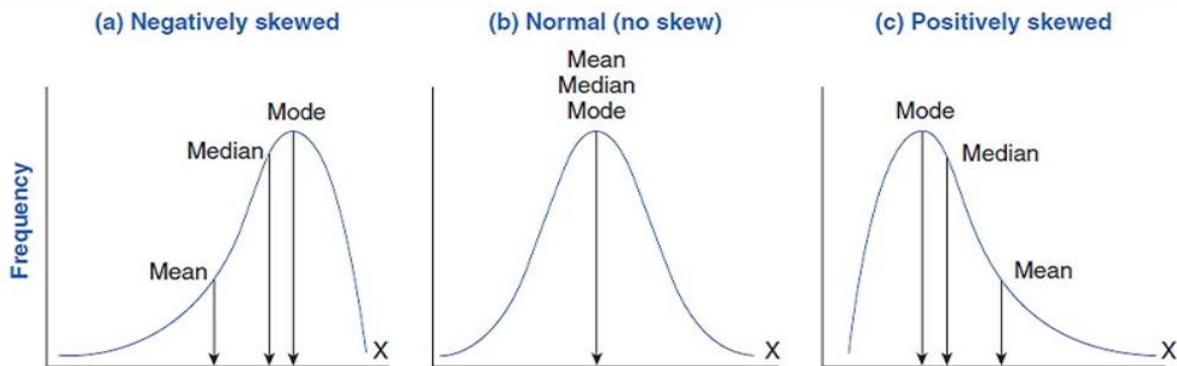
**(THE LAW OF LARGE NUMBERS)**

$$\frac{N_n(\text{outcome})}{n} \rightarrow P(\text{outcome}) \quad \text{as} \quad n \rightarrow \infty$$

Source: [Probabilistic World](#)

The Law of Large Numbers states that the average of many trials is close to the expected value. It is a fundamental principle of probability theory used to describe how an experiment's results converge on the true value as more and more trials are conducted.

## What Are Left-Skewed Distribution and Right-Skewed Distribution?



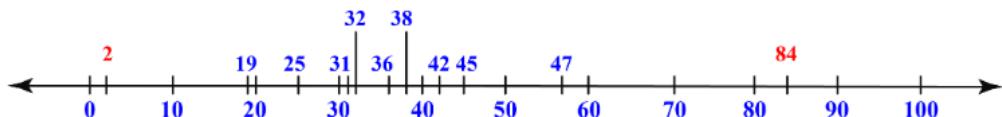
Source: [StudiousGuy](#)

Left-skewed distributions have a longer tail to the left (lower values), while right-skewed distributions have a longer tail to the right (higher values).

For example, if you were looking at the distribution of test scores on an exam, a left-skewed distribution would mean that more students scored lower than average than higher than average. A right-skewed distribution would mean that more students scored higher than average than lower than average.

## What Is an Outlier, and How Can You Find One?

**Example:** For a data set containing 2, 19, 25, 32, 36, 38, 31, 42, 57, 45, and 84

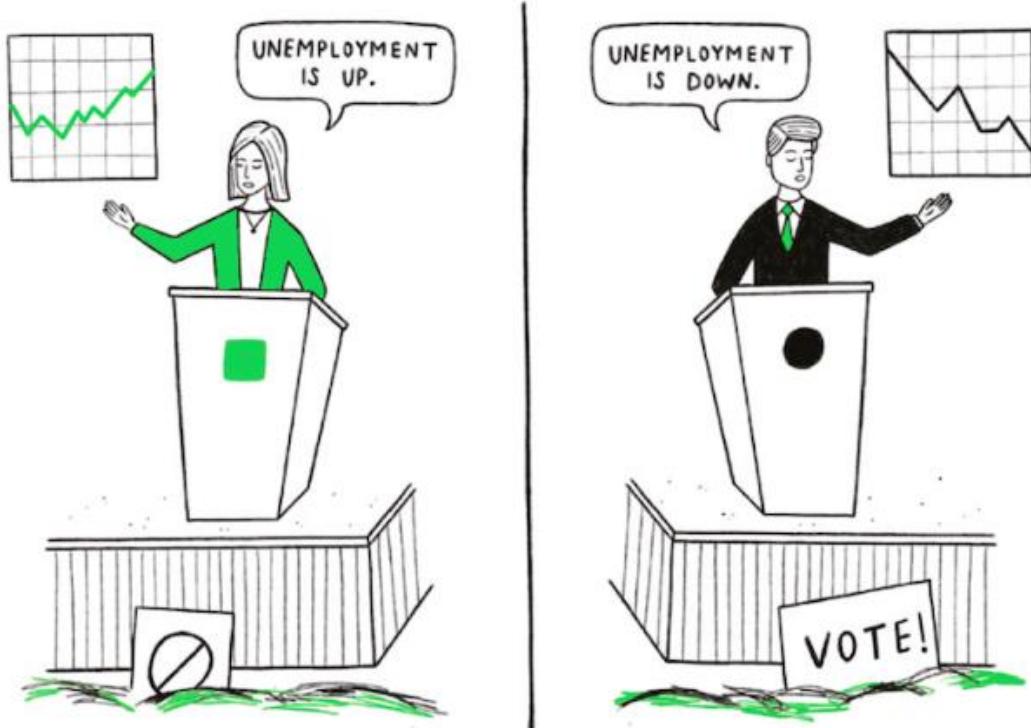


An outlier is an observation point that is distant from other data points. It's important to note that the term "outlier" doesn't refer to the numerical value of a data point but rather the distance between it and all other values.

You can use statistical tools like box plots or stem-and-leaf plots to find outliers in your dataset.

## Describe:

Cherry-Picking



Cherry-picking is a term used in statistics to describe the practice of selecting data points that support a conclusion. It's also called data mining or using an inappropriate statistical test.

In statistics, cherry-picking can be done intentionally or unintentionally. If a researcher wants to prove that one treatment is better, they might use only the data points that support their claim and discard any that don't fit their narrative. On the other hand, if a researcher doesn't know what they're doing or doesn't realize that they're making decisions about what data points to include and exclude, they could accidentally be cherry-picking their data set.

### P-Hacking or Data Dredging



P-hacking or Data Dredging is manipulating data to get the desired result. You can do this by changing the way you analyze your data until you get the desired outcome. This is not a good way to conduct research because it means that you're not actually looking at your data objectively, and you're more likely to find false positives (results that seem significant but are not).

### Significance Chasing

Significance chasing is the practice of using statistics to confirm a hypothesis rather than using it to explore unproven ideas. This is done by setting a very low threshold for significance and then either finding barely statistically significant results or manipulating data to achieve statistical significance.

Significance chasing is unethical and ineffective because it leads researchers to draw conclusions that are not well-supported by the data.

### If Four Coins Are Tossed Simultaneously, What Is the Possibility of Getting Three Heads and One Tail?

The probability of getting three heads and one tail is  $\frac{1}{4}$  or 25%.

## **How Many Possible Permutations Does a License Plate With 5 Digits Have?**

There are 100,000 permutations of a license plate with five digits. This can be found by multiplying the number of possible digits (ten) five times.

## **Take a Fair Dice. On Average, How Many Times Must You Roll the Dice Before Rolling a Six?**

On average, you'll need to roll the dice about four times before you land on six.

## **How Would You Go About Finding the Mean Height of Women in the World?**

To find the mean height of women worldwide, you should gather data from surveys of women across all countries. Then, use appropriate statistical methods to calculate an average for each country. Finally, create a mean for all countries combined using these numbers and mathematical formulas.

## **Two Fair Dice Are Rolled Together. What Is the Probability of Getting a Total Of:**

**3**

The probability of getting a total of 3 when two fair dice are rolled together is 1/18. This is because there are 36 possible outcomes when two dice are rolled, and only two of them result in a three.

**10**

The probability of getting a total of 10 when two fair dice are rolled together is 1/9. This is because there are 36 possible outcomes for the roll, and four of them result in a 10.

## **You Have To Draw Three Cards Successively From a Full Deck of Cards. What Is the Probability That You Draw a Face Card, a Seven, and a Two in That Order?**

The probability that you draw a face card, a seven, and a two in that order is 1/676. You can find this by multiplying the chances of pulling each card respectively.

There are twelve face cards in a deck, so the chance of pulling a face card is 1/4.

There are only four seven cards in a deck, so the chance of pulling a seven is 1/13.

There are only four two cards in a deck, so the chance of pulling a two is also 1/13.

### How Would You Go About Choosing a Sample Size?

#### Sample Size Formula



$$n = N \times \frac{\frac{Z^2 \times p \times (1 - p)}{e^2}}{N - 1 + \frac{Z^2 \times p \times (1 - p)}{e^2}}$$



There are a few ways to choose the sample size, but the most common method is to use the margin of error (ME) formula. The margin of error is the amount of error expected in your result. You can use this formula to determine the desired sample size.

Choose a sample size by first identifying the population of interest. Then, decide on a sample size that will allow you to represent that population accurately.

A good rule of thumb is to have a maximum sample of around 10% of the population.

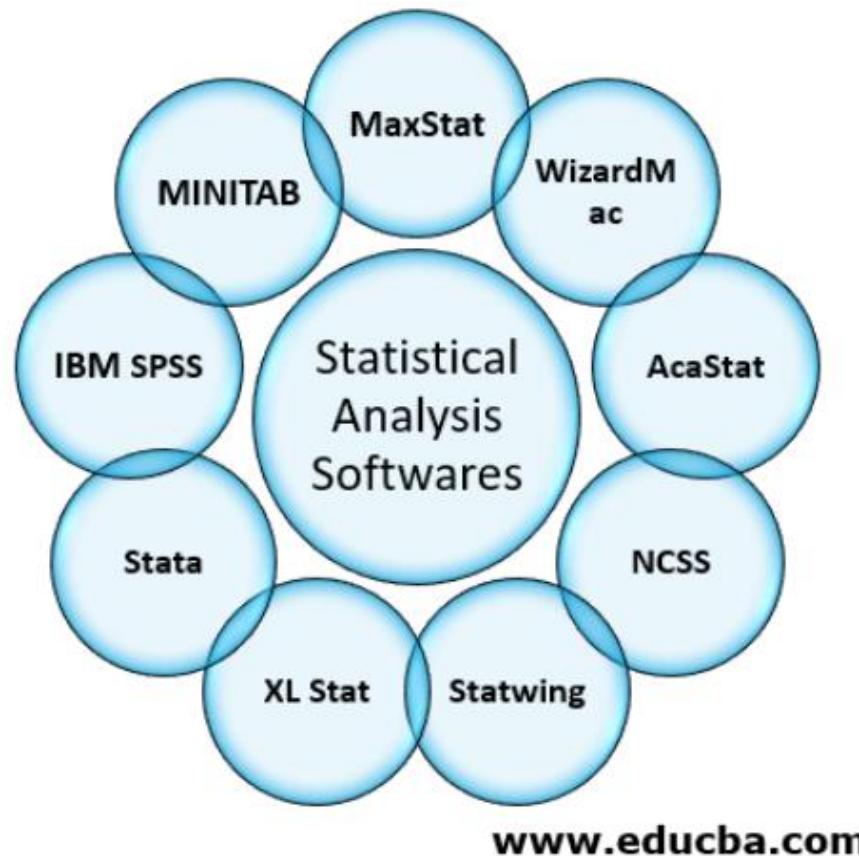
### Why Is Bessel's Correction Important?

$$s = \sqrt{\frac{\sum (x - \bar{x})^2}{n - 1}}$$

Source: [Statistics How To](#)

Bessel's correction is important because it allows us to solve differential equations, which tell you how an unknown function (the output) changes in time or space depending on the values of the dependent variables (the inputs).

## What Statistical Analysis Software Are You Familiar With?



When asked about software, you want to highlight the fact that you're familiar with a variety of software. You should also highlight your analytical thinking and problem-solving skills, as these are two key traits employers look for when hiring statisticians.

You might say, "I'm familiar with both Excel and Stata. I've used them for statistical analysis in different fields, including economics, psychology, and ecology."

You could also mention that you have experience using other software packages such as SPSS or R (or both).

## 1. What is the difference between Machine Learning and Deep Learning?

**Machine Learning** forms a subset of Artificial Intelligence, where we use statistics and algorithms to train machines with data, thereby, helping them improve with experience.

**Deep Learning** is a part of Machine Learning, which involves mimicking the human brain in terms of structures called neurons, thereby, forming **neural networks**.

## 2. What is a perceptron?

A perceptron is similar to the actual neuron in the human brain. It receives inputs from various entities and applies functions to these inputs, which transform them to be the output.

A perceptron is mainly used to perform binary classification where it sees an input, computes functions based on the weights of the input, and outputs the required transformation.

## 3. How is Deep Learning better than Machine Learning?

Machine Learning is powerful in a way that it is sufficient to solve most of the problems. However, **Deep Learning** gets an upper hand when it comes to working with data that has a large number of dimensions. With data that is large in size, a Deep Learning model can easily work with it as it is built to handle this.

.

## 4. What are some of the most used applications of Deep Learning?

Deep Learning is used in a variety of fields today. The most used ones are as follows:

- Sentiment Analysis
- Computer Vision
- Automatic Text Generation
- Object Detection

- [Natural Language Processing](#)
- Image Recognition

## **5. What is the meaning of overfitting?**

Overfitting is a very common issue when working with Deep Learning. It is a scenario where the Deep Learning algorithm vigorously hunts through the data to obtain some valid information.

This makes the Deep Learning model pick up noise rather than useful data, causing very high variance and low bias. This makes the model less accurate, and this is an undesirable effect that can be prevented.

## **6. What are activation functions?**

Activation functions are entities in Deep Learning that are used to translate inputs into a usable output parameter. It is a function that decides if a neuron needs activation or not by calculating the weighted sum on it with the bias.

Using an activation function makes the model output to be non-linear. There are many types of activation functions:

- ReLU
- Softmax
- Sigmoid
- Linear
- Tanh

## **7. Why is Fourier transform used in Deep Learning?**

Fourier transform is an effective package used for analyzing and managing large amounts of data present in a database. It can take in real-time array data and process it quickly. This ensures that high efficiency is maintained and also makes the model more open to processing a variety of signals.

## **8. What are the steps involved in training a perception in Deep Learning?**

There are five main steps that determine the learning of a perceptron:

1. Initialize thresholds and weights
2. Provide inputs
3. Calculate outputs
4. Update weights in each step
5. Repeat steps 2 to 4

## **9. What is the use of the loss function?**

The loss function is used as a measure of accuracy to see if a neural network has learned accurately from the training data or not. This is done by comparing the training dataset to the testing dataset.

The loss function is a primary measure of the performance of the neural network. In Deep Learning, a good performing network will have a low loss function at all times when training.

## **10. What are some of the Deep Learning frameworks or tools that you have used?**

This question is quite common in a Deep Learning interview. Make sure to answer based on the experience you have with the tools.

However, some of the top Deep Learning frameworks out there today are:

- TensorFlow
- Keras
- PyTorch
- Caffe2
- CNTK
- MXNet
- Theano

## **11. What is the use of the swish function?**

The swish function is a self-gated activation function developed by Google. It is now a popular activation function used by many as Google claims that it outperforms all of the other activation functions in terms of computational efficiency.

## **12. What are autoencoders?**

Autoencoders are artificial neural networks that learn without any supervision. Here, these networks have the ability to automatically learn by mapping the inputs to the corresponding outputs.

Autoencoders, as the name suggests, consist of two entities:

- Encoder: Used to fit the input into an internal computation state
- Decoder: Used to convert the computational state back into the output

## **13. What are the steps to be followed to use the gradient descent algorithm?**

There are five main steps that are used to initialize and use the gradient descent algorithm:

- Initialize biases and weights for the network
- Send input data through the network (the input layer)
- Calculate the difference (the error) between expected and predicted values
- Change values in neurons to minimize the loss function
- Multiple iterations to determine the best weights for efficient working

## **14. Differentiate between a single-layer perceptron and a multi-layer perceptron.**

Single-layer Perceptron	Multi-layer Perceptron
Cannot classify non-linear data points	Can classify non-linear data

Takes in a limited amount of parameters	Withstands a lot of parameters
Less efficient with large data	Highly efficient with large datasets

•

## 15. What is data normalization in Deep Learning?

Data normalization is a preprocessing step that is used to refit the data into a specific range. This ensures that the network can learn effectively as it has better convergence when performing backpropagation.

## 16. What is forward propagation?

Forward propagation is the scenario where inputs are passed to the hidden layer with weights. In every single hidden layer, the output of the activation function is calculated until the next layer can be processed. It is called forward propagation as the process begins from the input layer and moves toward the final output layer.

## 17. What is backpropagation?

**Backpropagation** is used to minimize the cost function by first seeing how the value changes when weights and biases are tweaked in the neural network. This change is easily calculated by understanding the gradient at every hidden layer. It is called backpropagation as the process begins from the output layer, moving backward to the input layers.

## 18. What are hyperparameters in Deep Learning?

Hyperparameters are variables used to determine the structure of a neural network. They are also used to understand parameters, such as the learning rate and the number of hidden layers, and more, present in the neural network.

## 19. How can hyperparameters be trained in neural networks?

Hyperparameters can be trained using four components as shown below:

- Batch size: This is used to denote the size of the input chunk. Batch sizes can be varied and cut into sub-batches based on the requirement.
- Epochs: An epoch denotes the number of times the training data is visible to the neural network so that it can train. Since the process is iterative, the number of epochs will vary based on the data.
- Momentum: Momentum is used to understand the next consecutive steps that occur with the current data being executed at hand. It is used to avoid oscillations when training.
- Learning rate: Learning rate is used as a parameter to denote the time required for the network to update the parameters and learn.

Next up on this top Deep Learning interview questions and answers blog, let us take a look at the intermediate questions.

## **20. What is the meaning of dropout in Deep Learning?**

Dropout is a technique that is used to avoid overfitting a model in Deep Learning. If the dropout value is too low, then it will have minimal effect on learning. If it is too high, then the model can under-learn, thereby, causing lower efficiency.

## **21. What are tensors?**

Tensors are multidimensional arrays in Deep Learning that are used to represent data. They represent the data with higher dimensions. Due to the high-level nature of the programming languages, the syntax of tensors is easily understood and broadly used.

## **22. What is the meaning of model capacity in Deep Learning?**

In Deep Learning, model capacity refers to the capacity of the model to take in a variety of mapping functions. Higher model capacity means a large amount of information can be stored in the network.

We will check out neural network interview questions alongside as it is also a vital part of Deep Learning.

### **23. What is a Boltzmann machine?**

A Boltzmann machine is a type of recurrent neural network that uses binary decisions, alongside biases, to function. These neural networks can be hooked up together to create deep belief networks, which are very sophisticated and used to solve the most complex problems out there.

### **24. What are some of the advantages of using TensorFlow?**

TensorFlow has numerous advantages, and some of them are as follows:

- High amount of flexibility and platform independence
- Trains using CPU and GPU
- Supports auto differentiation and its features
- Handles threads and asynchronous computation easily
- Open-source
- Has a large community

### **25. What is a computational graph in Deep Learning?**

A computation graph is a series of operations that are performed to take inputs and arrange them as nodes in a graph structure. It can be considered as a way of implementing mathematical calculations into a graph. This helps in parallel processing and provides high performance in terms of computational capability.

### **26. What is a CNN?**

CNNs are [convolutional neural networks](#) that are used to perform analysis on images and visuals. These classes of neural networks can input a multi-channel image and work on it easily.

These Deep Learning questions must be answered in a concise way. So make sure to understand them and revisit them if necessary.

## **27. What are the various layers present in a CNN?**

There are four main layers that form a convolutional neural network:

- Convolution: These are layers consisting of entities called filters that are used as parameters to train the network.
- ReLu: It is used as the activation function and is always used with the convolution layer.
- Pooling: Pooling is the concept of shrinking the complex data entities that form after convolution and is primarily used to maintain the size of an image after shrinkage.
- Connectedness: This is used to ensure that all of the layers in the neural network are fully connected and activation can be computed using the bias easily.

## **28. What is an RNN in Deep Learning?**

RNNs stand for [recurrent neural networks](#), which form to be a popular type of artificial neural network. They are used to process sequences of data, text, genomes, handwriting, and more. RNNs make use of backpropagation for the training requirements.

## **29. What is a vanishing gradient when using RNNs?**

Vanishing gradient is a scenario that occurs when we use RNNs. Since RNNs make use of backpropagation, gradients at every step of the way will tend to get smaller as the network traverses through backward iterations. This equates to the model learning very slowly, thereby, causing efficiency problems in the network.

## **30. What is exploding gradient descent in Deep Learning?**

Exploding gradients are an issue causing a scenario that clumps up the gradients. This creates a large number of updates of the weights in the model when training.

The working of gradient descent is based on the condition that the updates are small and controlled. Controlling the updates will directly affect the efficiency of the model.

### **31. What is the use of LSTM?**

LSTM stands for long short-term memory. It is a type of RNN that is used to sequence a string of data. It consists of feedback chains that give it the ability to perform like a general-purpose computational entity.

### **32. Where are autoencoders used?**

Autoencoders have a wide variety of usage in the real world. The following are some of the popular ones:

- Adding color to black–white images
- Removing noise from images
- Dimensionality reduction
- Feature removal and variation

### **33. What are the types of autoencoders?**

There are four main types of autoencoders:

- Deep autoencoders
- Convolutional autoencoders
- Sparse autoencoders
- Contractive autoencoders

### **34. What is a Restricted Boltzmann Machine?**

A Restricted Boltzmann Machine, or RBM for short, is an undirected graphical model that is popularly used in Deep Learning today. It is an algorithm that is used to perform:

- Dimensionality reduction
- Regression
- Classification
- Collaborative filtering
- Topic modeling

Next up on this top Deep Learning interview questions and answers blog, let us take a look at the advanced questions.

### **35. What are some of the limitations of Deep Learning?**

There are a few disadvantages of Deep Learning as mentioned below:

- Networks in Deep Learning require a huge amount of data to train well.
- Deep Learning concepts can be complex to implement sometimes.
- Achieving a high amount of model efficiency is difficult in many cases.

These are some of the vital advanced deep learning interview questions that you have to know about!

### **36. What are the variants of gradient descent?**

There are three variants of gradient descent as shown below:

- Stochastic gradient descent: A single training example is used for the calculation of gradient and for updating parameters.
- Batch gradient descent: Gradient is calculated for the entire dataset, and parameters are updated at every iteration.

- Mini-batch gradient descent: Samples are broken down into smaller-sized batches and then worked on as in the case of stochastic gradient descent.

### **37. Why is mini-batch gradient descent so popular?**

Mini-batch gradient descent is popular as:

- It is more efficient when compared to stochastic gradient descent.
- Generalization is done by finding the flat minima.
- It helps avoid the local minima by allowing the approximation of the gradient for the entire dataset.

### **38. What are deep autoencoders?**

Deep autoencoders are an extension of the regular autoencoders. Here, the first layer is responsible for the first-order function execution of the input. The second layer will take care of the second-order functions, and it goes on.

Usually, a deep autoencoder is a combination of two or more symmetrical deep-belief networks where:

- The first five shallow layers consist of the encoding part
- The other layers take care of the decoding part

On the next set of Deep Learning questions, let us look further into the topic.

### **39. Why is the Leaky ReLU function used in Deep Learning?**

Leaky ReLU, also called LReLU, is used to manage a function to allow the passing of small-sized negative values if the input value to the network is less than zero.

### **40. What are some of the examples of supervised learning algorithms in Deep Learning?**

There are three main supervised learning algorithms in Deep Learning:

- Artificial neural networks
- Convolutional neural networks
- Recurrent neural networks

#### **41. What are some of the examples of unsupervised learning algorithms in Deep Learning?**

There are three main unsupervised learning algorithms in Deep Learning:

- Autoencoders
- Boltzmann machines
- Self-organizing maps

Next up, let us look at more neural network interview questions that will help you ace the interviews.

#### **42. Can we initialize the weights of a network to start from zero?**

Yes, it is possible to begin with zero initialization. However, it is not recommended to use because setting up the weights to zero initially will cause all of the neurons to produce the same output and the same gradients when performing backpropagation. This means that the network will not have the ability to learn at all due to the absence of asymmetry between each of the neurons.

#### **43. What is the meaning of valid padding and same padding in CNN?**

- Valid padding: It is used when there is no requirement for padding. The output matrix will have the dimensions  $(n - f + 1) \times (n - f + 1)$  after convolution.
- Same padding: Here, padding elements are added all around the output matrix. It will have the same dimensions as the input matrix.

#### **44. What are some of the applications of transfer learning in Deep Learning?**

Transfer learning is a scenario where a large model is trained on a dataset with a large amount of data and this model is used on simpler datasets, thereby resulting in extremely efficient and accurate neural networks.

The popular examples of transfer learning are in the case of:

- BERT
- ResNet
- GPT-2
- VGG-16

#### **45. How is the transformer architecture better than RNNs in Deep Learning?**

With the use of sequential processing, programmers were up against:

- The usage of high processing power
- The difficulty of parallel execution

This caused the rise of the transformer architecture. Here, there is a mechanism called attention mechanism, which is used to map all of the dependencies between sentences, thereby making huge progress in the case of NLP models.

#### **46.What are the steps involved in the working of an LSTM network?**

There are three main steps involved in the working of an LSTM network:

- The network picks up the information that it has to remember and identifies what to forget.
- Cell state values are updated based on Step 1.
- The network calculates and analyzes which part of the current state should make it to the output.

## **47. What are the elements in TensorFlow that are programmable?**

In TensorFlow, users can program three elements:

- Constants
- Variables
- Placeholders

## **48. What is the meaning of bagging and boosting in Deep Learning?**

Bagging is the concept of splitting a dataset and randomly placing it into bags for training the model.

Boosting is the scenario where incorrect data points are used to force the model to produce the wrong output. This is used to retrain the model and increase accuracy.

## **49. What are generative adversarial networks (GANs)?**

Generative adversarial networks are used to achieve generative modeling in Deep Learning. It is an unsupervised task that involves the discovery of patterns in the input data to generate the output.

The generator is used to generate new examples, while the discriminator is used to classify the examples generated by the generator.

## **50. Why are generative adversarial networks (GANs) so popular?**

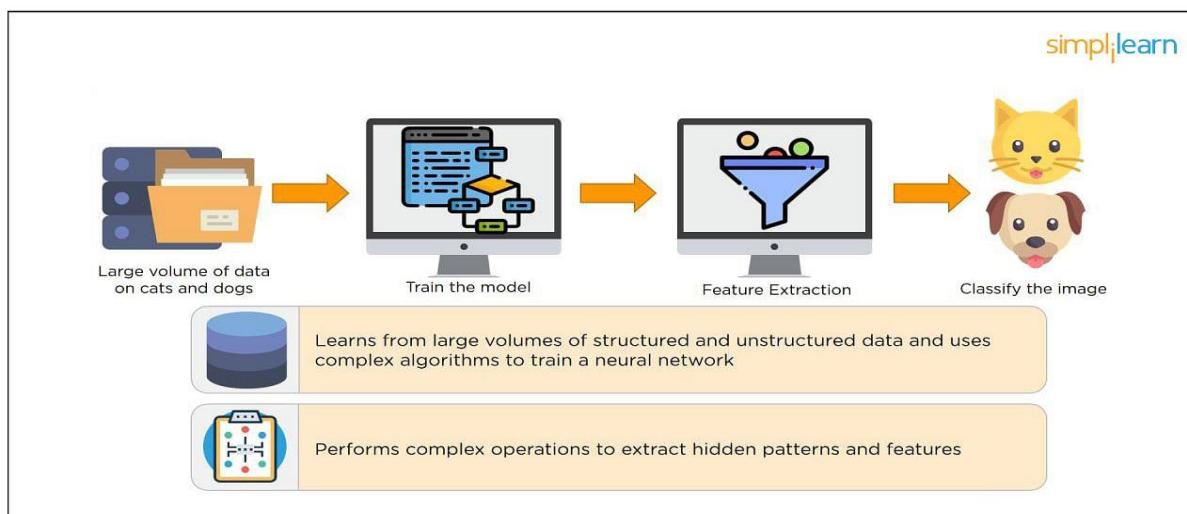
Generative adversarial networks are used for a variety of purposes. In the case of working with images, they have a high amount of traction and efficient working.

- Creation of art: GANs are used to create artistic images, sketches, and paintings.
- Image enhancement: They are used to greatly enhance the resolution of the input images.

- Image translation: They are also used to change certain aspects, such as day to night and summer to winter, in images easily.

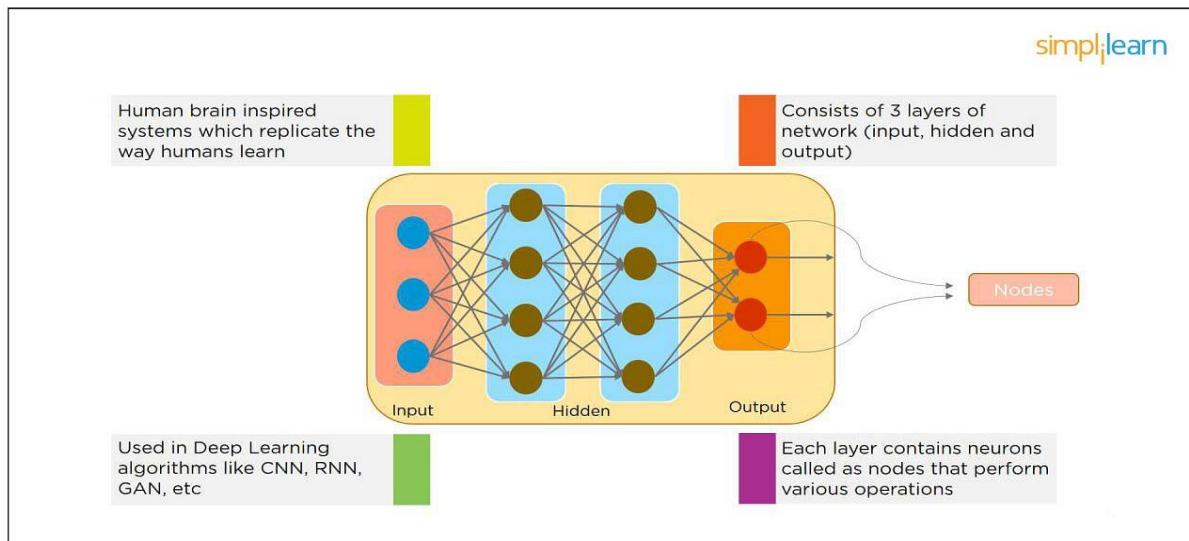
## 1. What is Deep Learning?

If you are going for a deep learning interview, you definitely know what exactly deep learning is. However, with this question the interviewee expects you to give an in-detail answer, with an example. **Deep Learning** involves taking large volumes of structured or unstructured data and using complex algorithms to train neural networks. It performs complex operations to extract hidden patterns and features (for instance, distinguishing the image of a cat from that of a dog).



## 2. What is a Neural Network?

**Neural Networks** replicate the way humans learn, inspired by how the neurons in our brains fire, only much simpler.



The most common Neural Networks consist of three network layers:

1. An input layer
2. A hidden layer (this is the most important layer where feature extraction takes place, and adjustments are made to train faster and function better)
3. An output layer

Each sheet contains neurons called “nodes,” performing various operations. Neural Networks are used in **deep learning algorithms** like CNN, RNN, GAN, etc.

### 3. What Is a Multi-layer Perceptron(MLP)?

As in Neural Networks, **MLPs** have an input layer, a hidden layer, and an output layer. It has the same structure as a single layer **perceptron** with one or more hidden layers. A single layer perceptron can classify only linear separable classes with binary output (0,1), but MLP can classify nonlinear classes.

Except for the input layer, each node in the other layers uses a nonlinear activation function. This means the input layers, the data coming in, and the activation function is based upon all nodes and weights being added together, producing the output. MLP uses a **supervised learning** method called “backpropagation.” In backpropagation, the neural network calculates the error with the help of cost function. It propagates this error backward from where it came (adjusts the weights to train the model more accurately).

## 4. What Is Data Normalization, and Why Do We Need It?

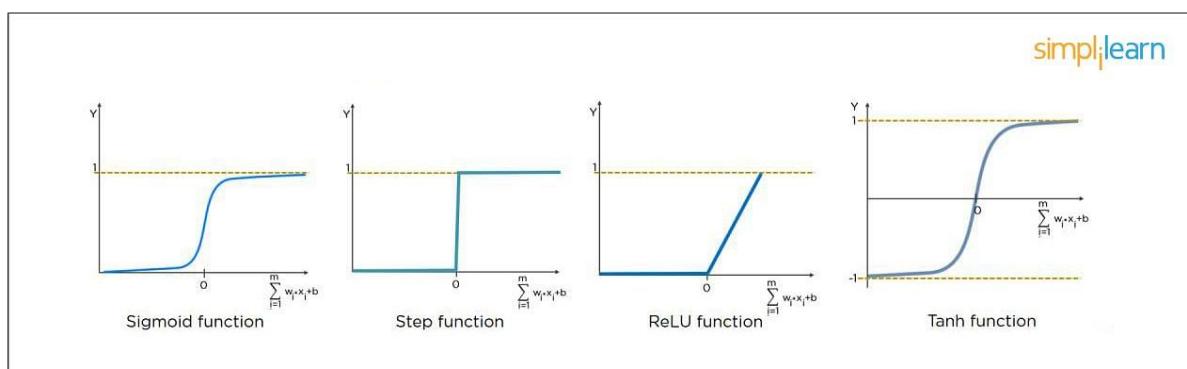
The process of standardizing and reforming data is called “Data Normalization.” It’s a pre-processing step to eliminate data redundancy. Often, data comes in, and you get the same information in different formats. In these cases, you should rescale values to fit into a particular range, achieving better convergence.

## 5. What is the Boltzmann Machine?

One of the most basic Deep Learning models is a Boltzmann Machine, resembling a simplified version of the Multi-Layer Perceptron. This model features a visible input layer and a hidden layer -- just a two-layer neural net that makes stochastic decisions as to whether a neuron should be on or off. Nodes are connected across layers, but no two nodes of the same layer are connected.

## 6. What Is the Role of Activation Functions in a Neural Network?

At the most basic level, an activation function decides whether a neuron should be fired or not. It accepts the weighted sum of the inputs and bias as input to any activation function. Step function, Sigmoid, ReLU, Tanh, and Softmax are examples of activation functions.



## 7. What Is the Cost Function?

Also referred to as “loss” or “error,” cost function is a measure to evaluate how good your model’s performance is. It’s used to compute the error of the output layer during backpropagation. We push that error backward through the neural network and use that during the different training functions.

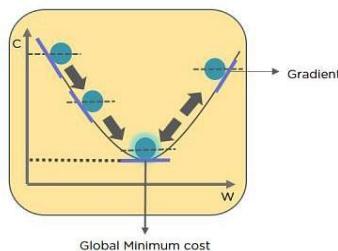
Cost Function:

$$C = \frac{1}{2} (Y - \hat{Y})^2$$

$Y$  -----> Original Output  
 $\hat{Y}$  -----> Predicted Output

## 8. What Is Gradient Descent?

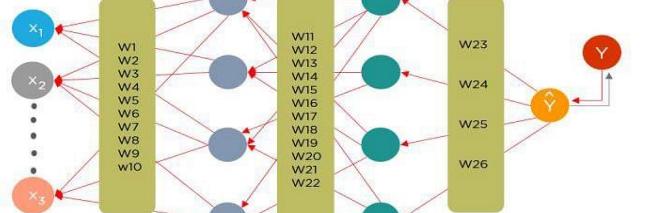
Gradient Descent is an optimal algorithm to minimize the cost function or to minimize an error. The aim is to find the local-global minima of a function. This determines the direction the model should take to reduce the error.



## 9. What Do You Understand by Backpropagation?

This is one of the most frequently asked deep learning interview questions. Backpropagation is a technique to improve the performance of the network. It backpropagates the error and updates the weights to reduce the error.

- Neural Network technique to minimize the cost function
- Helps to improve the performance of the network
- Backpropagates the error and updates the weights to reduce the error

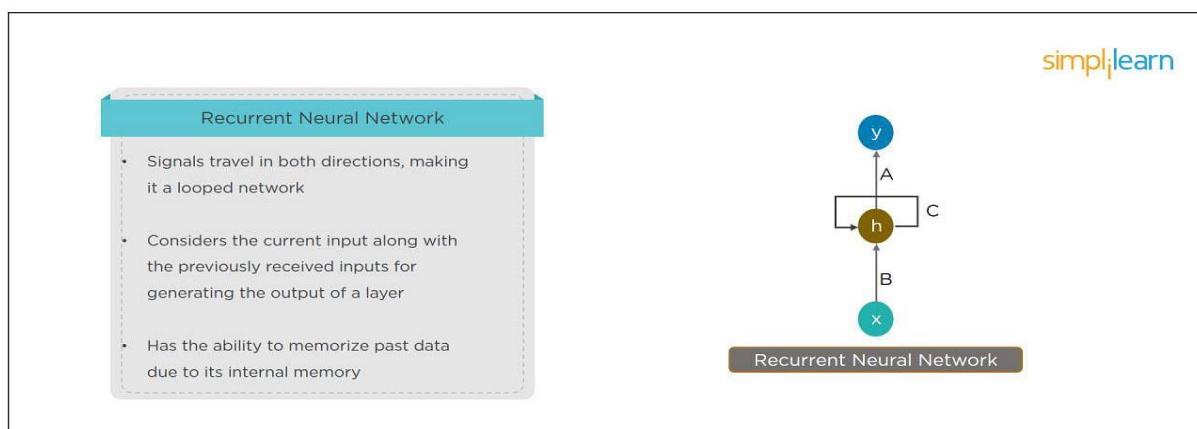


## 10. What Is the Difference Between a Feedforward Neural Network and Recurrent Neural Network?

In this deep learning interview question, the interviewee expects you to give a detailed answer.

A Feedforward Neural Network signals travel in one direction from input to output. There are no feedback loops; the network considers only the current input. It cannot memorize previous inputs (e.g., [CNN](#)).

A Recurrent Neural Network's signals travel in both directions, creating a looped network. It considers the current input along with the previously received inputs for generating the output of a layer and can memorize past data due to its internal memory.



## 11. What Are the Applications of a Recurrent Neural Network (RNN)?

The [RNN](#) can be used for sentiment analysis, text mining, and image captioning. Recurrent Neural Networks can also address time series problems such as predicting the prices of stocks in a month or quarter.

## 12. What Are the Softmax and ReLU Functions?

Softmax is an activation function that generates the output between zero and one. It divides each output, such that the total sum of the outputs is equal to one. Softmax is often used for output layers.

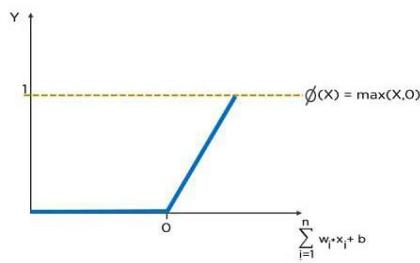
- Softmax is an activation function that generates the output between 0 and 1
- It divides each output such that the total sum of the outputs is equal to 1
- It is often used in the output layers

$$\text{Softmax}(L_n) = \frac{e^{L_n}}{\| e^{L_n} \|}$$



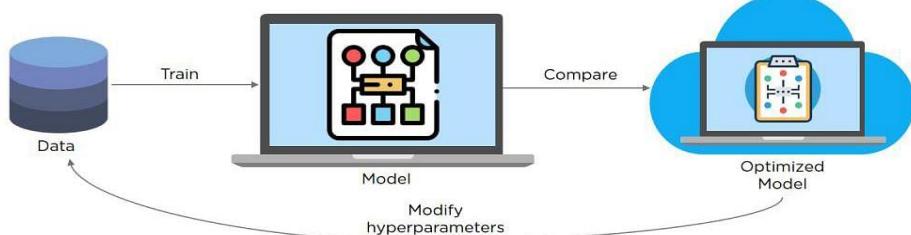
ReLU (or Rectified Linear Unit) is the most widely used activation function. It gives an output of X if X is positive and zeros otherwise. ReLU is often used for hidden layers.

- ReLU stands for Rectified Linear Unit and is the most widely used activation function
- It gives an output of X if X is positive and 0 otherwise
- It is often used in the hidden layers



### 13. What Are Hyperparameters?

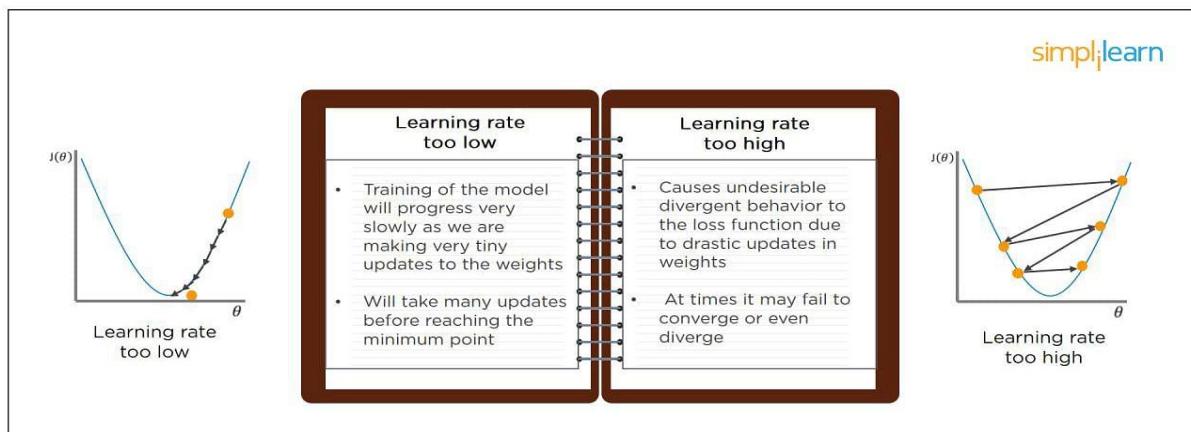
This is another frequently asked deep learning interview question. With neural networks, you're usually working with hyperparameters once the data is formatted correctly. A hyperparameter is a parameter whose value is set before the learning process begins. It determines how a network is trained and the structure of the network (such as the number of hidden units, the learning rate, epochs, etc.).



## 14. What Will Happen If the Learning Rate Is Set Too Low or Too High?

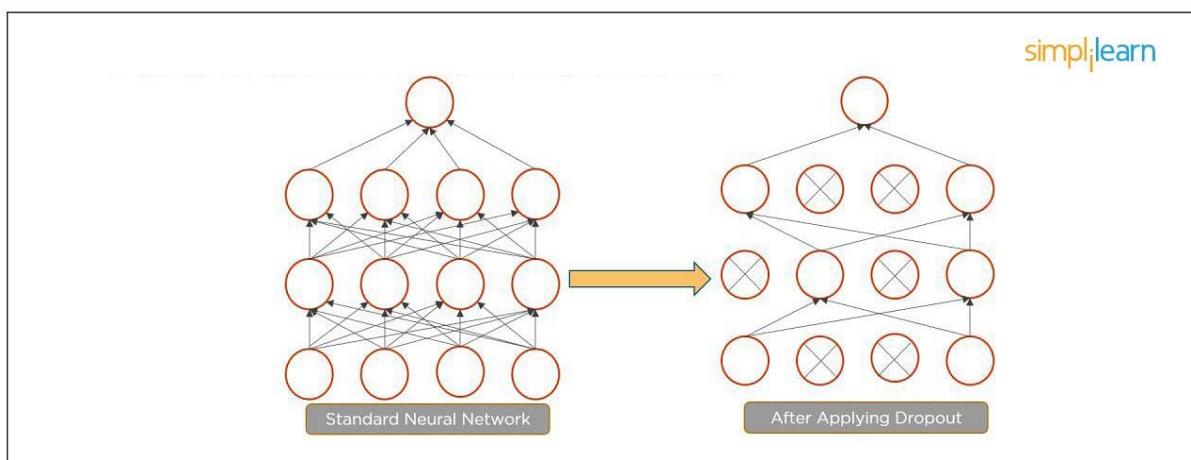
When your learning rate is too low, training of the model will progress very slowly as we are making minimal updates to the weights. It will take many updates before reaching the minimum point.

If the learning rate is set too high, this causes undesirable divergent behavior to the loss function due to drastic updates in weights. It may fail to converge (model can give a good output) or even diverge (data is too chaotic for the network to train).



## 15. What Is Dropout and Batch Normalization?

Dropout is a technique of dropping out hidden and visible units of a network randomly to prevent overfitting of data (typically dropping 20 percent of the nodes). It doubles the number of iterations needed to converge the network.



Batch normalization is the technique to improve the performance and stability of neural networks by normalizing the inputs in every layer so that they have mean output activation of zero and standard deviation of one.

The next step on this top Deep Learning interview questions and answers blog will be to discuss intermediate questions.

## 16. What Is the Difference Between Batch Gradient Descent and Stochastic Gradient Descent?

Batch Gradient Descent	Stochastic Gradient Descent
<p>The batch gradient computes the gradient using the entire dataset.</p> <p>It takes time to converge because the volume of data is huge, and weights update slowly.</p>	<p>The stochastic gradient computes the gradient using a single sample.</p> <p>It converges much faster than the batch gradient because it updates weight more frequently.</p>

## 17. What is Overfitting and Underfitting, and How to Combat Them?

Overfitting occurs when the model learns the details and noise in the training data to the degree that it adversely impacts the execution of the model on new information. It is more likely to occur with nonlinear models that have more flexibility when learning a target function. An example would be if a model is looking at cars and trucks, but only recognizes trucks that have a specific box shape. It might not be able to notice a flatbed truck because there's only a particular kind of truck it saw in training. The model performs well on training data, but not in the real world.

Underfitting alludes to a model that is neither well-trained on data nor can generalize to new information. This usually happens when there is less and incorrect data to train a model. Underfitting has both poor performance and accuracy.

To combat overfitting and underfitting, you can resample the data to estimate the model accuracy (k-fold cross-validation) and by having a validation dataset to evaluate the model.

## 18. How Are Weights Initialized in a Network?

There are two methods here: we can either initialize the weights to zero or assign them randomly.

Initializing all weights to 0: This makes your model similar to a linear model. All the neurons and every layer perform the same operation, giving the same output and making the deep net useless.

Initializing all weights randomly: Here, the weights are assigned randomly by initializing them very close to 0. It gives better accuracy to the model since every neuron performs different computations. This is the most commonly used method.

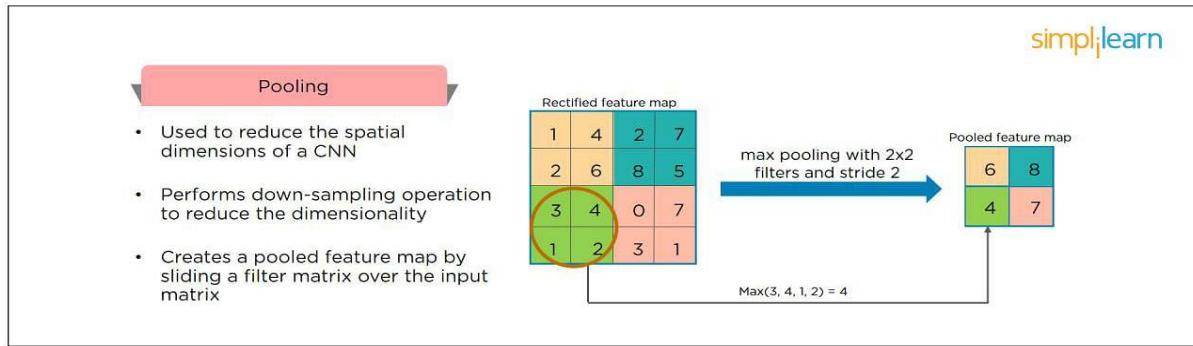
## 19. What Are the Different Layers on CNN?

There are four layers in CNN:

1. Convolutional Layer - the layer that performs a convolutional operation, creating several smaller picture windows to go over the data.
2. ReLU Layer - it brings non-linearity to the network and converts all the negative pixels to zero. The output is a rectified feature map.
3. Pooling Layer - pooling is a down-sampling operation that reduces the dimensionality of the feature map.
4. Fully Connected Layer - this layer recognizes and classifies the objects in the image.

## 20. What is Pooling on CNN, and How Does It Work?

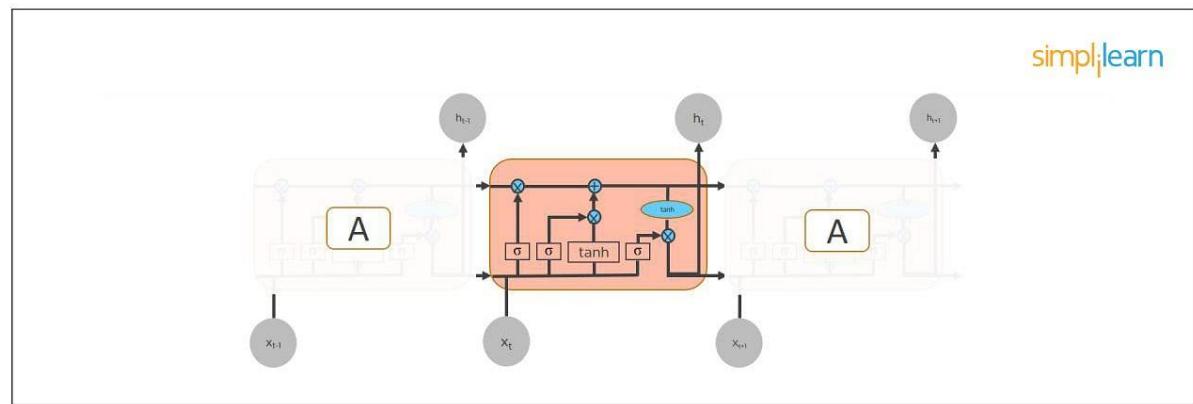
Pooling is used to reduce the spatial dimensions of a CNN. It performs down-sampling operations to reduce the dimensionality and creates a pooled feature map by sliding a filter matrix over the input matrix.



## 21. How Does an LSTM Network Work?

Long-Short-Term Memory (LSTM) is a special kind of recurrent neural network capable of learning long-term dependencies, remembering information for long periods as its default behavior. There are three steps in an LSTM network:

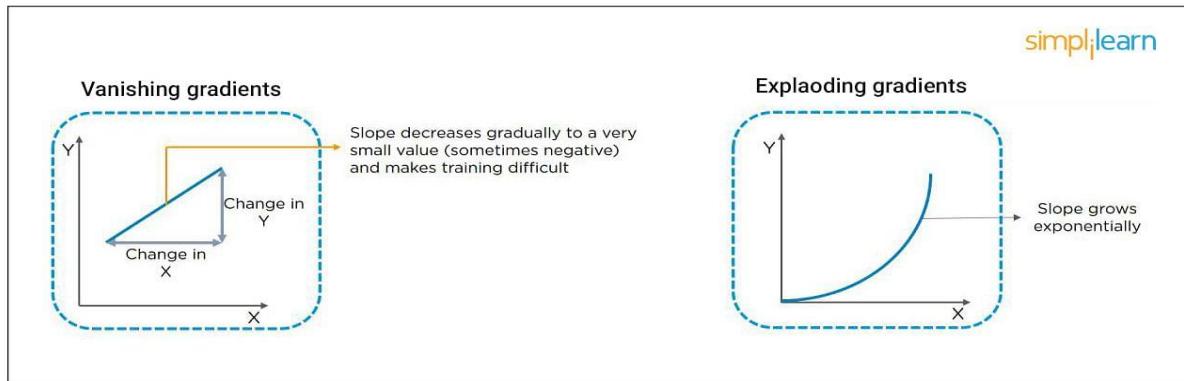
- Step 1: The network decides what to forget and what to remember.
- Step 2: It selectively updates cell state values.
- Step 3: The network decides what part of the current state makes it to the output.



## 22. What Are Vanishing and Exploding Gradients?

While training an RNN, your slope can become either too small or too large; this makes the training difficult. When the slope is too small, the problem is known as a “Vanishing Gradient.” When the slope tends to grow exponentially

instead of decaying, it's referred to as an "Exploding Gradient." Gradient problems lead to long training times, poor performance, and low accuracy.



### 23. What Is the Difference Between Epoch, Batch, and Iteration in Deep Learning?

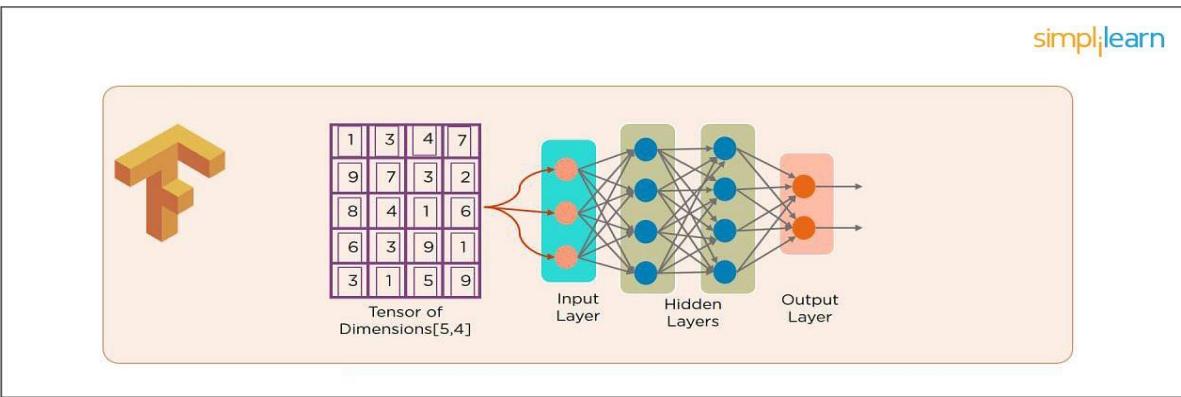
- Epoch - Represents one iteration over the entire dataset (everything put into the training model).
- Batch - Refers to when we cannot pass the entire dataset into the neural network at once, so we divide the dataset into several batches.
- Iteration - if we have 10,000 images as data and a batch size of 200. then an epoch should run 50 iterations (10,000 divided by 50).

### 24. Why is Tensorflow the Most Preferred Library in Deep Learning?

**Tensorflow** provides both **C++** and Python APIs, making it easier to work on and has a faster compilation time compared to other Deep Learning libraries like **Keras** and **Torch**. Tensorflow supports both CPU and GPU computing devices.

### 25. What Do You Mean by Tensor in Tensorflow?

This is another most frequently asked deep learning interview question. A tensor is a mathematical object represented as arrays of higher dimensions. These arrays of data with different dimensions and ranks fed as input to the neural network are called "Tensors."



## 26. What Are the Programming Elements in Tensorflow?

**Constants** - Constants are parameters whose value does not change. To define a constant we use `tf.constant()` command. For example:

```
a = tf.constant(2.0, tf.float32)
```

```
b = tf.constant(3.0)
```

```
Print(a, b)
```

**Variables** - Variables allow us to add new trainable parameters to graph. To define a variable, we use the `tf.Variable()` command and initialize them before running the graph in a session. An example:

```
W = tf.Variable([.3], dtype=tf.float32)
```

```
b = tf.Variable([-3], dtype=tf.float32)
```

**Placeholders** - these allow us to feed data to a tensorflow model from outside a model. It permits a value to be assigned later. To define a placeholder, we use the `tf.placeholder()` command. An example:

```
a = tf.placeholder(tf.float32)
```

```
b = a*2
```

```
with tf.Session() as sess:
```

```
result = sess.run(b,feed_dict={ a:3.0 })
```

```
print result
```

Sessions - a session is run to evaluate the nodes. This is called the “Tensorflow runtime.” For example:

```
a = tf.constant(2.0)
```

```
b = tf.constant(4.0)
```

```
c = a+b
```

```
# Launch Session
```

```
Sess = tf.Session()
```

```
# Evaluate the tensor c
```

```
print(sess.run(c))
```

## 27. Explain a Computational Graph.

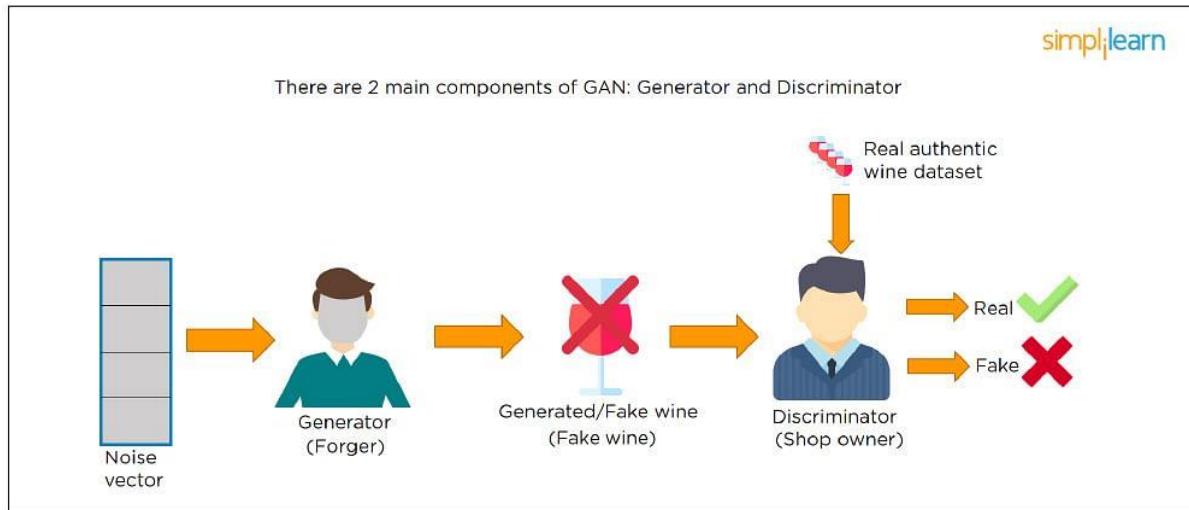
Everything in a **tensorflow** is based on creating a computational graph. It has a network of nodes where each node operates, Nodes represent mathematical operations, and edges represent tensors. Since data flows in the form of a graph, it is also called a “DataFlow Graph.”

## 28. Explain **Generative Adversarial Network**.

Suppose there is a wine shop purchasing wine from dealers, which they resell later. But some dealers sell fake wine. In this case, the shop owner should be able to distinguish between fake and authentic wine.

The forger will try different techniques to sell fake wine and make sure specific techniques go past the shop owner’s check. The shop owner would probably get some feedback from wine experts that some of the wine is not original. The owner would have to improve how he determines whether a wine is fake or authentic.

The forger's goal is to create wines that are indistinguishable from the authentic ones while the shop owner intends to tell if the wine is real or not accurately.



Let us understand this example with the help of an image shown above.

There is a noise vector coming into the forger who is generating fake wine.

Here the forger acts as a Generator.

The shop owner acts as a Discriminator.

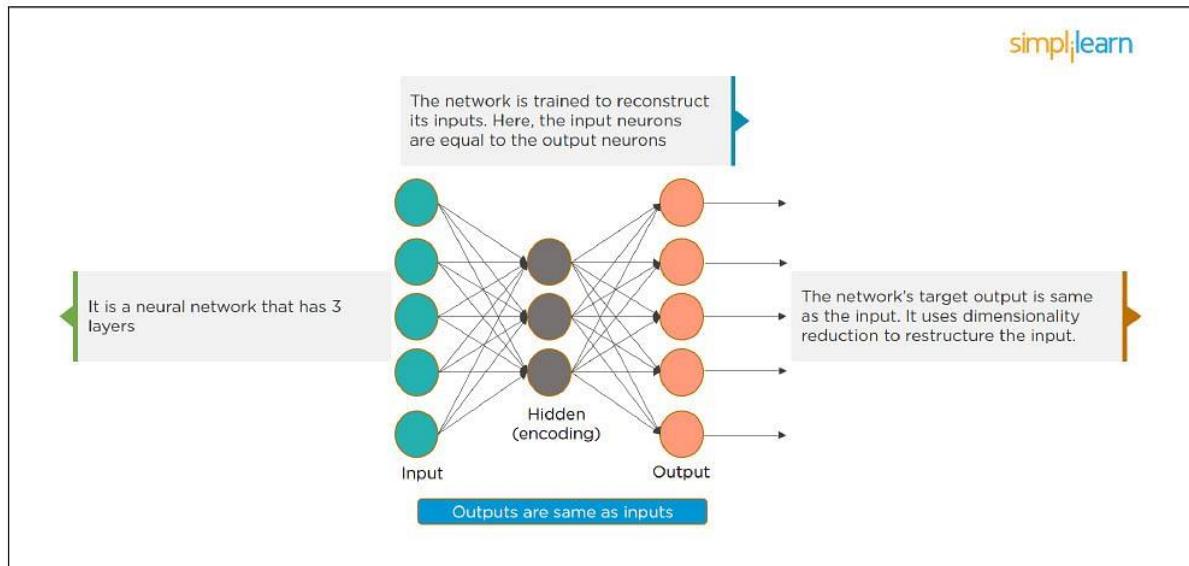
The Discriminator gets two inputs; one is the fake wine, while the other is the real authentic wine. The shop owner has to figure out whether it is real or fake.

So, there are two primary components of Generative Adversarial Network (GAN) named:

1. Generator
2. Discriminator

The generator is a CNN that keeps keys producing images and is closer in appearance to the real images while the discriminator tries to determine the difference between real and fake images. The ultimate aim is to make the discriminator learn to identify real and fake images.

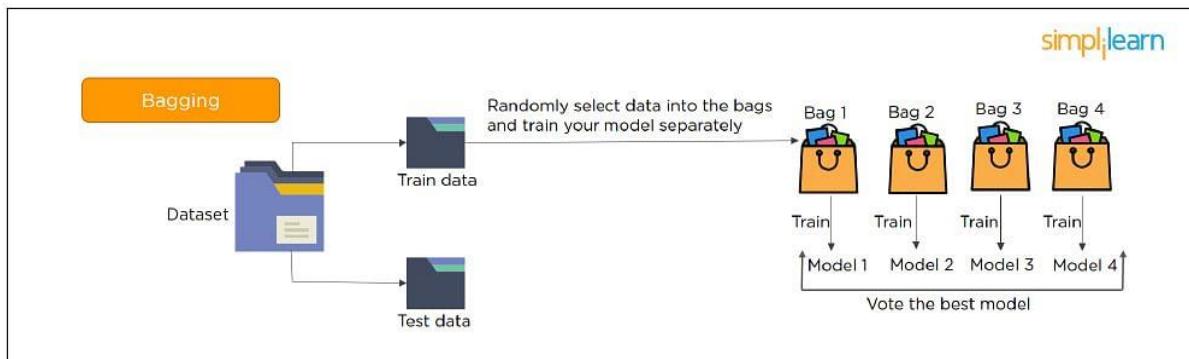
## 29. What Is an Auto-encoder?



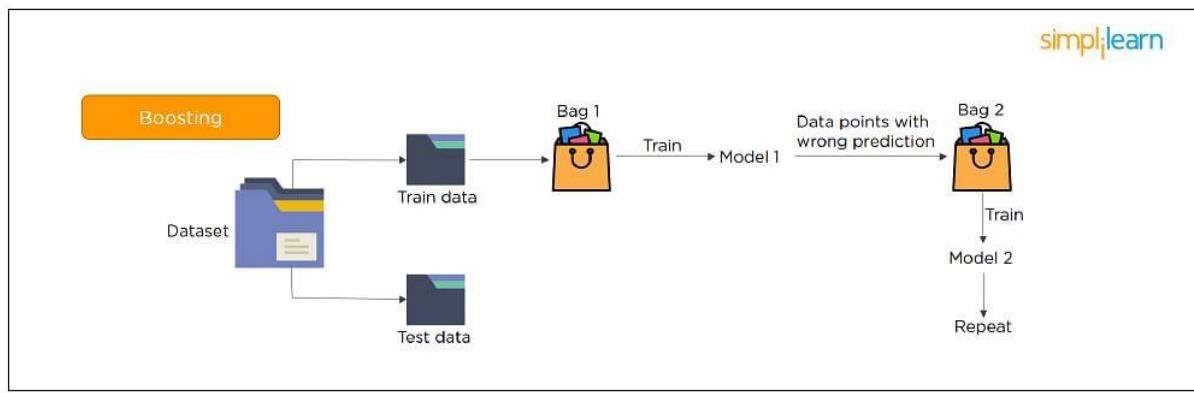
This Neural Network has three layers in which the input neurons are equal to the output neurons. The network's target outside is the same as the input. It uses dimensionality reduction to restructure the input. It works by compressing the image input to a latent space representation then reconstructing the output from this representation.

### 30. What Is Bagging and Boosting?

Bagging and Boosting are ensemble techniques to train multiple models using the same learning algorithm and then taking a call.



With Bagging, we take a dataset and split it into training data and test data. Then we randomly select data to place into the bags and train the model separately.



With Boosting, the emphasis is on selecting data points which give wrong output to improve the accuracy.

The following are some of the most important advanced deep learning interview questions that you should know!

**31. What is the significance of using the Fourier transform in Deep Learning tasks?**

The Fourier transform function efficiently analyzes, maintains, and manages large datasets. You can use it to generate real-time array data that is helpful for processing multiple signals.

**32. What do you understand by transfer learning? Name a few commonly used transfer learning models.**

Transfer learning is the process of transferring the learning from a model to another model without having to train it from scratch. It takes critical parts of a pre-trained model and applies them to solve new but similar machine learning problems.

Some of the popular transfer learning models are:

- VGG-16
- BERT
- GTP-3
- Inception V3
- XCEPTION

### 33. What is the difference between SAME and VALID padding in Tensorflow?

Using the Tensorflow library, `tf.nn.max_pool` performs the max-pooling operation. `Tf.nn.max_pool` has a padding argument that takes 2 values - SAME or VALID.

With padding == “SAME” ensures that the filter is applied to all the elements of the input.

The input image gets fully covered by the filter and specified stride. The padding type is named SAME as the output size is the same as the input size (when stride=1).

With padding == “VALID” implies there is no padding in the input image. The filter window always stays inside the input image. It assumes that all the dimensions are valid so that the input image gets fully covered by a filter and the stride defined by you.

### 34. What are some of the uses of Autoencoders in Deep Learning?

- Autoencoders are used to convert black and white images into colored images.
- Autoencoder helps to extract features and hidden patterns in the data.
- It is also used to reduce the dimensionality of data.
- It can also be used to remove noises from images.

### 35. What is the Swish Function?

Swish is an activation function proposed by Google which is an alternative to the ReLU activation function.

It is represented as:  $f(x) = x * \text{sigmoid}(x)$ .

The Swish function works better than ReLU for a variety of deeper models.

The derivative of Swist can be written as:  $y' = y + \text{sigmoid}(x) * (1 - y)$

### 36. What are the reasons for mini-batch gradient being so useful?

- Mini-batch gradient is highly efficient compared to stochastic gradient descent.
- It lets you attain generalization by finding the flat minima.
- Mini-batch gradient helps avoid local minima to allow gradient approximation for the whole dataset.

### 37. What do you understand by Leaky ReLU activation function?

Leaky ReLU is an advanced version of the ReLU activation function. In general, the ReLU function defines the gradient to be 0 when all the values of inputs are less than zero. This deactivates the neurons. To overcome this problem, Leaky ReLU activation functions are used. It has a very small slope for negative values instead of a flat slope.

### 38. What is Data Augmentation in Deep Learning?

Data Augmentation is the process of creating new data by enhancing the size and quality of training datasets to ensure better models can be built using them. There are different techniques to augment data such as numerical data augmentation, image augmentation, GAN-based augmentation, and text augmentation.

### 39. Explain the Adam optimization algorithm.

Adaptive Moment Estimation or Adam optimization is an extension to the stochastic gradient descent. This algorithm is useful when working with complex problems involving vast amounts of data or parameters. It needs less memory and is efficient.

Adam optimization algorithm is a combination of two gradient descent methodologies -

Momentum and Root Mean Square Propagation.

### 40. Why is a convolutional neural network preferred over a dense neural network for an image classification task?

- The number of parameters in a convolutional neural network is much more diminutive than that of a Dense Neural Network. Hence, a CNN is less likely to overfit.
- CNN allows you to look at the weights of a filter and visualize what the network learned. So, this gives a better understanding of the model.
- CNN trains models in a hierarchical way, i.e., it learns the patterns by explaining complex patterns using simpler ones.

41. Which strategy does not prevent a model from over-fitting to the training data?

1. Dropout
2. Pooling
3. Data augmentation
4. Early stopping

Answer: b) Pooling - It's a layer in CNN that performs a downsampling operation.

42. Explain two ways to deal with the vanishing gradient problem in a deep neural network.

- Use the ReLU activation function instead of the sigmoid function
- Initialize neural networks using Xavier initialization that works with tanh activation.

43. Why is a deep neural network better than a shallow neural network?

Both deep and shallow neural networks can approximate the values of a function. But the deep neural network is more efficient as it learns something new in every layer. A shallow neural network has only one hidden layer. But a deep neural network has several hidden layers that create a deeper representation and computation capability.

44. What is the need to add randomness in the weight initialization process?

If you set the weights to zero, then every neuron at each layer will produce the same result and the same gradient value during backpropagation. So, the neural network won't be able to learn the function as there is no asymmetry between the neurons. Hence, randomness to the weight initialization process is crucial.

#### 45. How can you train hyperparameters in a neural network?

Hyperparameters in a neural network can be trained using four components:

Batch size: Indicates the size of the input data.

Epochs: Denotes the number of times the training data is visible to the neural network to train.

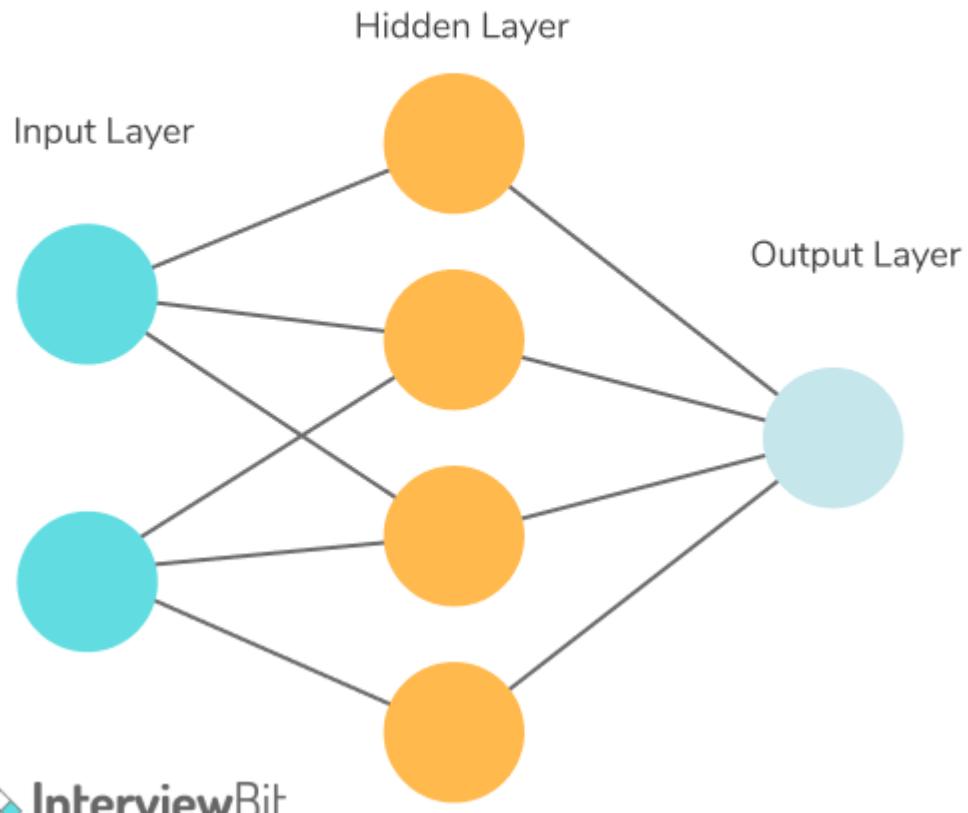
Momentum: Used to get an idea of the next steps that occur with the data being executed.

Learning rate: Represents the time required for the network to update the parameters and learn

### **1. What do you understand about Neural Networks in the context of Deep Learning?**

Neural Networks are artificial systems that have a lot of resemblance to the biological neural networks in the human body. A neural network is a set of algorithms that attempts to recognize underlying relationships in a batch of data using a method that mimics how the human brain works. Without any task-specific rules, these systems learn to do tasks by being exposed to a variety of datasets and examples. The notion is that instead of being programmed with a pre-coded understanding of these datasets, the system derives identifying traits from the data it is fed to. Neural networks are built on threshold logic computational models. Because neural networks can adapt to changing input, they can produce the best possible outcome without requiring the output criteria to be redesigned.

# A Simple Neural Network



## 2. What are the applications of deep learning?

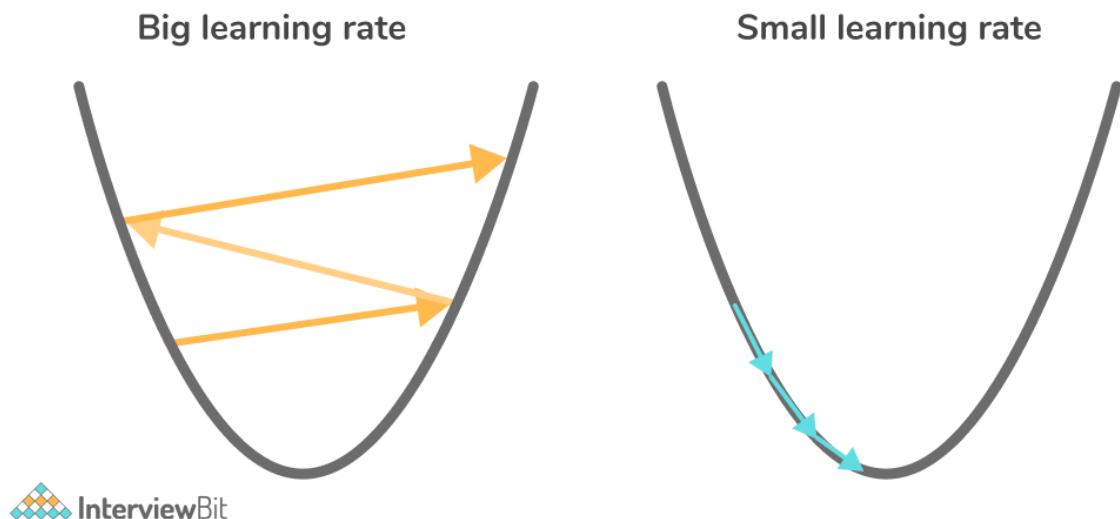
Following are some of the applications of deep learning:-

- Pattern recognition and natural language processing.
- Recognition and processing of images.
- Automated translation.
- Analysis of sentiment.
- System for answering questions.
- Classification and Detection of Objects.
- Handwriting Generation by Machine.
- Automated text generation.
- Colorization of Black and White images.

## 3. Explain learning rate in the context of neural network models. What happens if the learning rate is too high or too low?

Learning rate is a number that ranges from 0 to 1. It is one of the most important tunable hyperparameters in neural network training models. The learning rate

determines how quickly or slowly a neural network model adapts to a given situation and learns. A higher learning rate value indicates that the model only needs a few training epochs and produces rapid changes, whereas a lower learning rate indicates that the model may take a long time to converge or may never converge and become stuck on a poor solution. As a result, it is recommended that a good learning rate value be established by trial and error rather than using a learning rate that is too low or too high.



In the above image, we can clearly see that a big learning rate leads us to move away from the desired output. However, having a small learning rate leads us to the desired output eventually.

**You can download a PDF version of Deep Learning Interview Questions.**

[Download PDF](#)

#### 4. What are the advantages of neural networks?

Following are the advantages of neural networks:

- Neural networks are extremely adaptable, and they may be used for both classification and regression problems, as well as much more complex problems. Neural networks are also quite scalable. We can create as many layers as we wish, each with its own set of neurons. When there are a lot of data points, neural networks have been shown to generate the best

outcomes. They are best used with non-linear data such as images, text, and so on. They can be applied to any data that can be transformed into a numerical value.

- Once the neural network mode has been trained, they deliver output very fast. Thus, they are time-effective.

## **5. What are the disadvantages of neural networks?**

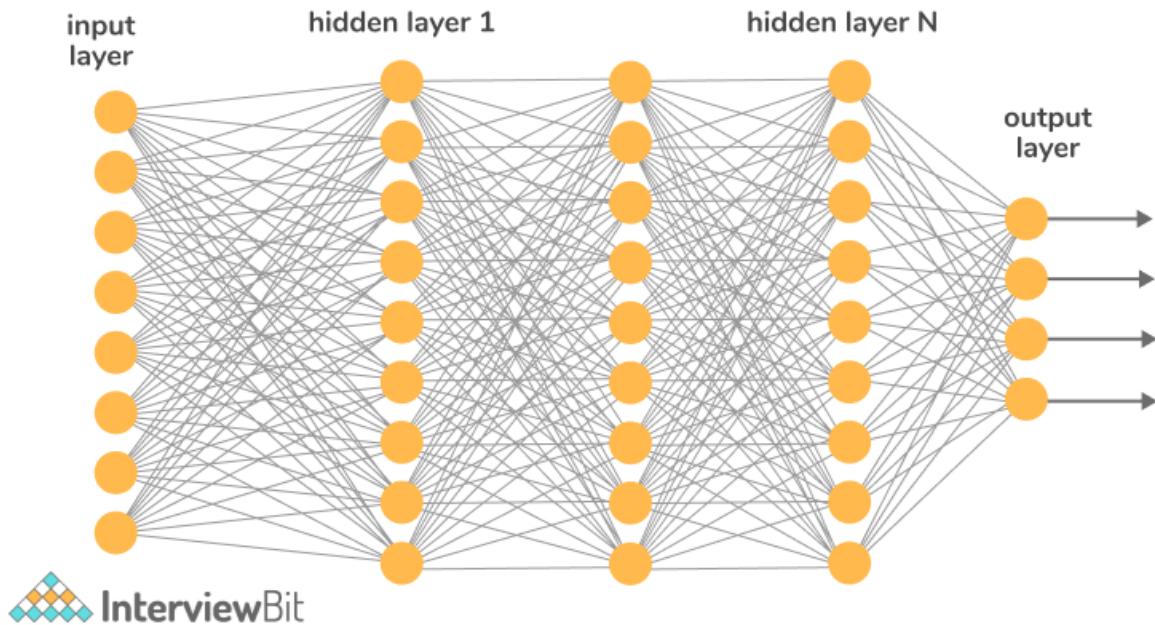
Following are the disadvantages of neural networks:-

- The "black box" aspect of neural networks is a well-known disadvantage. That is, we have no idea how or why our neural network produced a certain result. When we enter a dog image into a neural network and it predicts that it is a duck, we may find it challenging to understand what prompted it to make this prediction.
- It takes a long time to create a neural network model.
- Neural networks models are computationally expensive to build because a lot of computations need to be done at each layer.
- A neural network model requires significantly more data than a traditional machine learning model to train.

## **6. Explain what a deep neural network is.**

An artificial neural network (ANN) having numerous layers between the input and output layers is known as a deep neural network (DNN). Deep neural networks are neural networks that use deep architectures. The term "deep" refers to functions that have a higher number of layers and units in a single layer. It is possible to create more accurate models by adding more and larger layers to capture higher levels of patterns. The below image depicts a deep neural network.

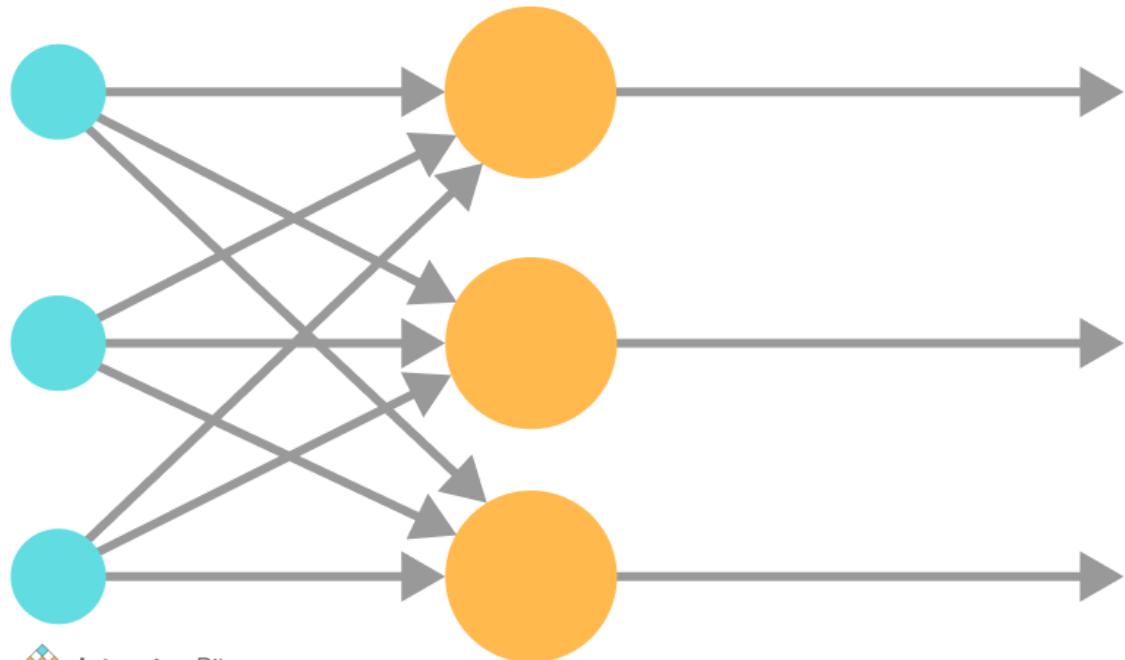
## Deep neural network



### 7. What are the different types of deep neural networks?

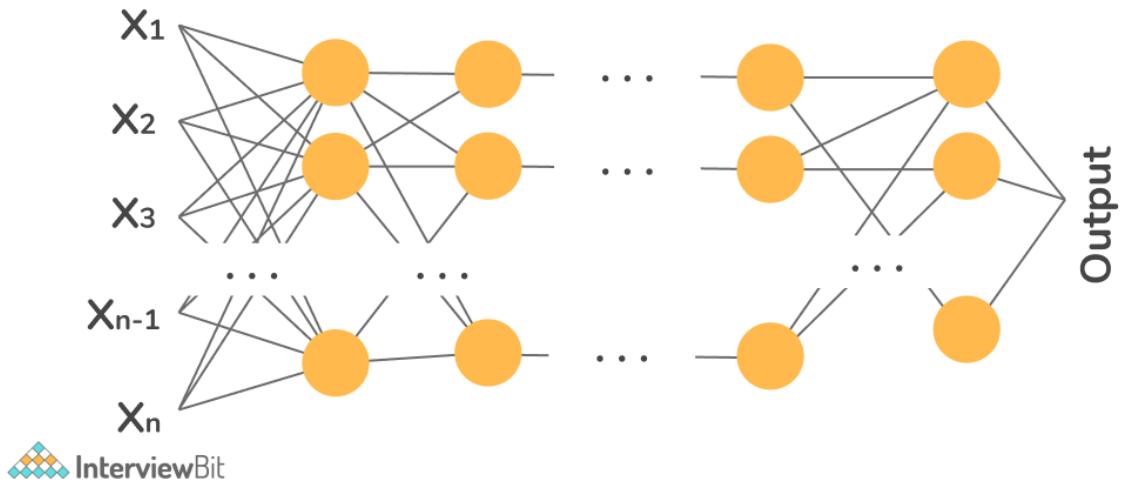
Following are the different types of deep neural networks:-

- **FeedForward Neural Network:-** This is the most basic type of neural network, in which flow control starts at the input layer and moves to the output layer. These networks only have a single layer or a single hidden layer. There is no backpropagation mechanism in this network because data only flows in one way. The input layer of this network receives the sum of the weights present in the input. These networks are utilised in the computer vision-based facial recognition method.

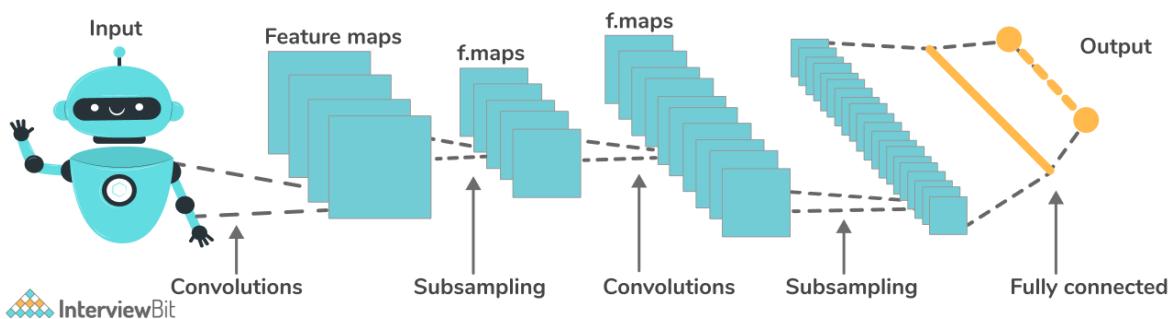


- **Radial Basis Function Neural Network:-** This type of neural network usually has more than one layer, preferably two. The relative distance from any location to the center is determined in this type of network and passed on to the next layer. In order to avoid blackouts, radial basis networks are commonly employed in power restoration systems to restore power in the shortest period possible.
- **Multi-Layer Perceptrons (MLP):-** A multilayer perceptron (MLP) is a type of feedforward artificial neural network (ANN). MLPs are the simplest deep neural networks, consisting of a succession of completely linked layers. Each successive layer is made up of a collection of nonlinear functions that are the weighted sum of all the previous layer's outputs (completely linked). Speech recognition and other machine learning systems rely heavily on these networks.

## Multi Layer Perceptrons (MLP):-

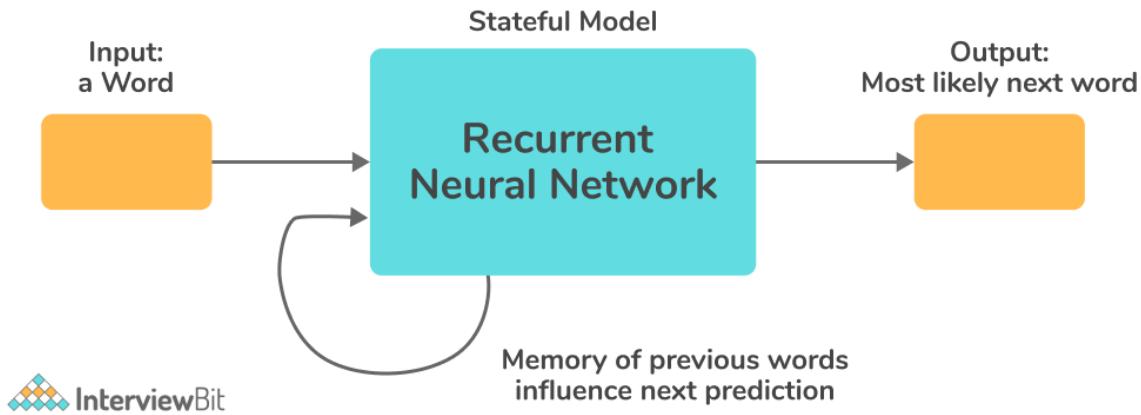


- **Convolutional Neural Network (CNN):-** Convolutional Neural Networks are mostly used in computer vision. In contrast to fully linked layers in MLPs, one or more convolution layers extract simple characteristics from input by performing convolution operations in CNN models. Each layer is made up of nonlinear functions of weighted sums at various coordinates of spatially close subsets of the previous layer's outputs, allowing the weights to be reused. The AI system learns to automatically extract the properties of these inputs to fulfill a specific task, such as picture classification, face identification, and image semantic segmentation, given a sequence of images or videos from the actual world.

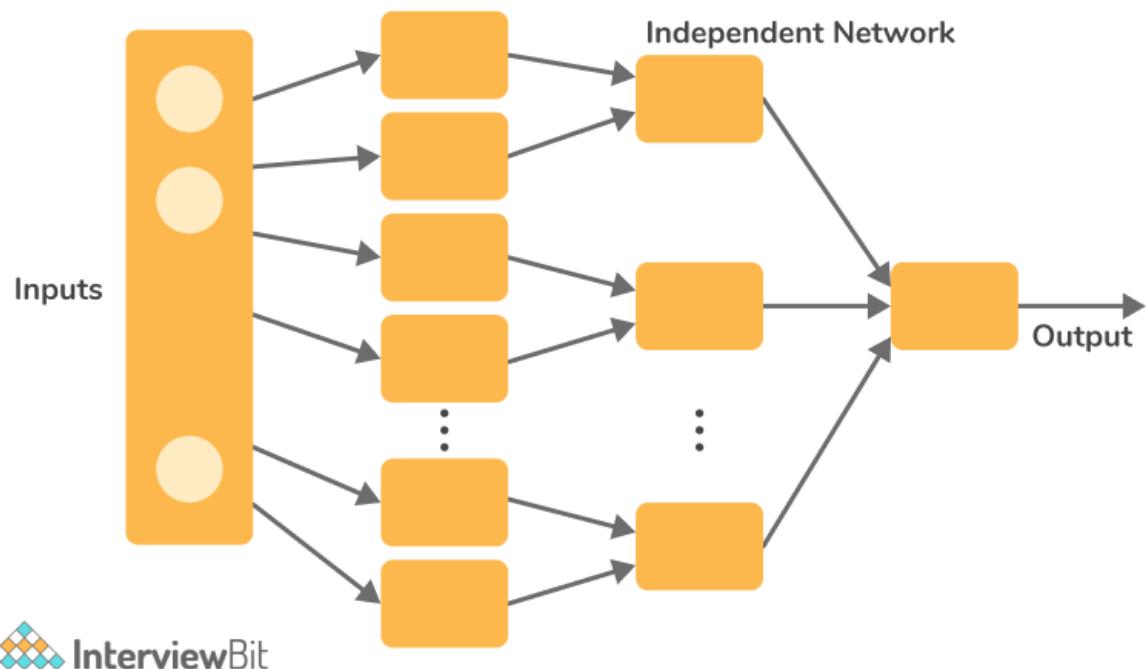


- **Recurrent Neural Network (RNN):-** Recurrent Neural Networks were created to solve the sequential input data time-series problem. RNN's input is made up of the current input and prior samples. As a result, the node connections create a directed graph. Furthermore, each neuron in an RNN has an internal memory that stores the information from previous samples' computations. Because of their superiority in processing data with a variable input length, RNN models are commonly employed in natural language processing (NLP). The goal of AI in this case is to create

a system that can understand human-spoken natural languages, such as natural language modeling, word embedding, and machine translation. Each successive layer in an RNN is made up of nonlinear functions of weighted sums of outputs and the preceding state. As a result, the basic unit of RNN is termed "cell," and each cell is made up of layers and a succession of cells that allow recurrent neural network models to be processed sequentially.



- **Modular Neural Network:-** This network is made up of numerous tiny neural networks, rather than being a single network. The sub-networks combine to form a larger neural network, which operates independently to achieve a common goal. These networks are extremely useful for breaking down a large-small problem into smaller chunks and then solving it.



- **Sequence to Sequence Model:-** In most cases, this network is made up of two RNN networks. The network is based on encoding and decoding, which means it has an encoder that processes the input and a decoder that processes the output. This type of network is commonly employed for text processing when the length of the inputting text differs from the length of the outputted text.

## 8. What do you mean by end-to-end learning?

It's a deep learning procedure in which a model is fed raw data and the entire data is trained at the same time to create the desired result with no intermediate steps. It is a deep learning method in which all of the different steps are trained simultaneously rather than sequentially. End-to-end learning has the advantage of eliminating the requirement for implicit feature engineering, which usually results in lower bias. Driverless automobiles are an excellent example that you may use in your end-to-end learning content. They are guided by human input and are programmed to learn and interpret information automatically using a CNN to fulfill tasks. Another good example is the generation of a written transcript (output) from a recorded audio clip (input). The model here skips all of the steps in the middle, focusing instead on the fact that it can manage the entire sequence of steps and tasks.

## 9. What do you understand about gradient clipping in the context of deep learning?

Gradient Clipping is a technique for dealing with the problem of exploding gradients (a situation in which huge error gradients build up over time, resulting in massive modifications to neural network model weights during training) that happens during backpropagation. The problem of exploding gradients occurs when the gradients get excessively big during training, causing the model to become unstable. If the gradient has crossed the anticipated range, the gradient values are driven element-by-element to a specific minimum or maximum value. Gradient clipping improves numerical stability while training a neural network, but it has little effect on the performance of the model.

## 10. Explain Forward and Back Propagation in the context of deep learning.

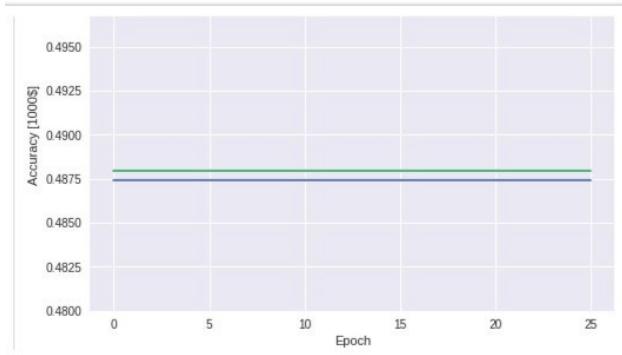
- **Forward Propagation:** The hidden layer, between the input layer and the output layer of the network, receives inputs with weights. We calculate the output of the activation at each node at each hidden layer, and this propagates to the next layer until we reach the final output layer. We go forward from the inputs to the final output layer, which is known as the forward propagation.
- **Back Propagation:** It sends error information from the network's last layer to all of the weights within the network. It's a technique for fine-tuning the weights of a neural network based on the previous epoch's (i.e., iteration) error rate. By fine-tuning the weights, you may lower error rates and improve the model's generalization, making it more dependable. The process of backpropagation can be broken down into the following steps: It can generate output by propagating training data through the network. It, then, computes the error derivative for output activations using the target and output values. It can backpropagate to compute the derivative of the error in the previous layer's output activation, and so on for all hidden layers. It calculates the error derivative for weights using the previously obtained derivatives and all hidden layers. The weights are updated based on the error derivatives obtained from the next layer.

## 11. Explain Data Normalisation. What is the need for it?

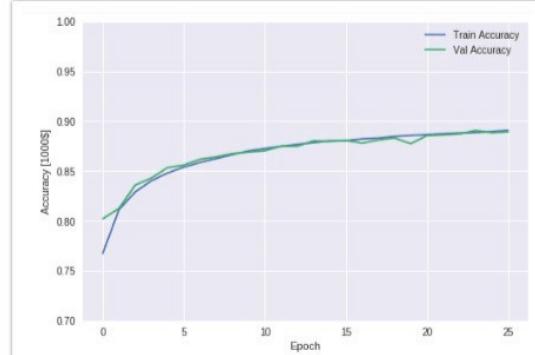
Data Normalisation is a technique in which data is transformed in such a way that they are either dimensionless or have a similar distribution. It is also known as standardization and feature scaling. It's a pre-processing procedure for the input data that removes redundant data from the dataset.

Normalization provides each variable equal weights/importance, ensuring that no single variable biases model performance in its favour simply because it is larger. It vastly improves model precision by converting the values of numeric columns in a dataset to a similar scale without distorting the range of values.

Model Accuracy, without normalized data



Model Accuracy, with normalized data



## 12. What are the different techniques to achieve data normalization?

Following are the different techniques employed to achieve data normalization:-

- **Rescaling:** Rescaling data is the process of multiplying each member of a data set by a constant term k, or changing each integer x to f(X), where  $f(x) = kx$  and k and x are both real values. The simplest of all approaches, rescaling (also known as "min-max normalization"), is calculated as:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

This represents the rescaling factor for every data point x.

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$



- **Mean Normalisation:** In the transformation process, this approach employs the mean of the observations:

$$x' = \frac{x - \bar{x}}{s}$$

This represents the mean normalizing factor for every data point x.

$$x' = \left( \frac{x - \text{average}(x)}{\max(x) - \min(x)} \right)$$



- **Z-score Normalisation:** This technique, also known as standardization, employs the Z-score or "standard score." SVM and logistic regression are two examples of machine learning algorithms that utilise it:

$$z = \frac{x - \mu}{\sigma}$$

This represents the Z-score.

$$z = \left( \frac{x - \mu}{\sigma} \right)$$

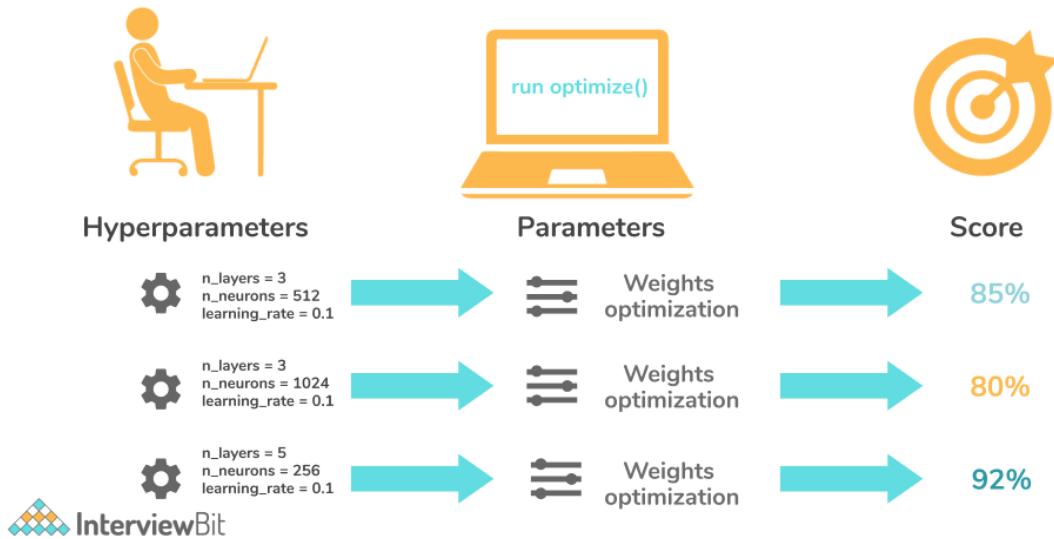


### 13. What do you mean by hyperparameters in the context of deep learning?

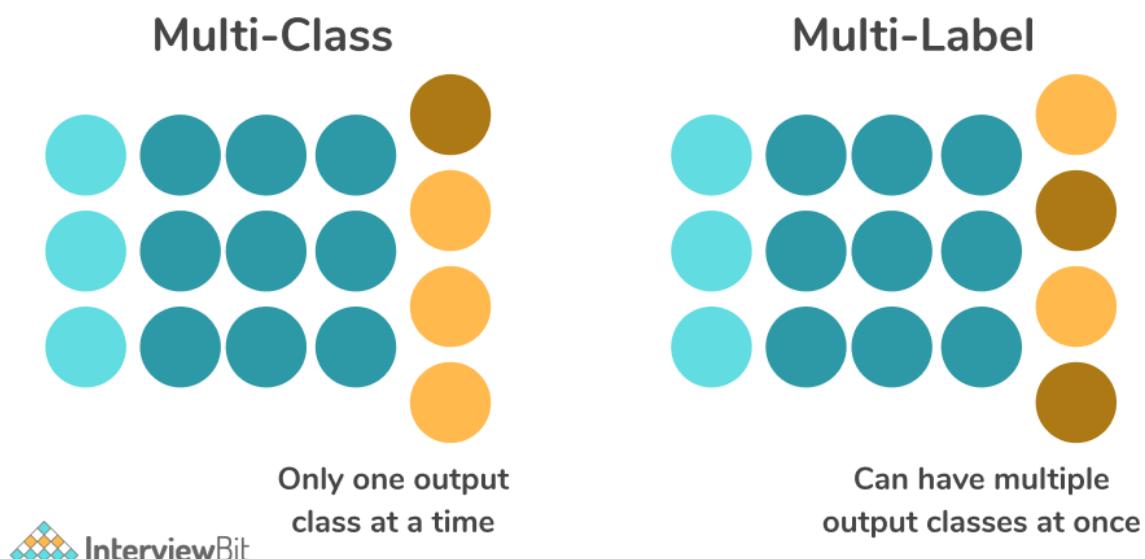
Hyperparameters are variables that determine the network topology (for example, the number of hidden units) and how the network is trained (Eg: Learning Rate). They are set before training the model, that is, before optimizing the weights and the bias.

Following are some of the examples of hyperparameters:-

- **Number of hidden layers:** With regularisation techniques, many hidden units inside a layer can boost accuracy. Underfitting may occur if the number of units is reduced.
- **Learning Rate:** The learning rate is the rate at which a network's parameters are updated. The learning process is slowed by a low learning rate, but it eventually converges. A faster learning rate accelerates the learning process, but it may not converge. A declining Learning rate is usually desired.



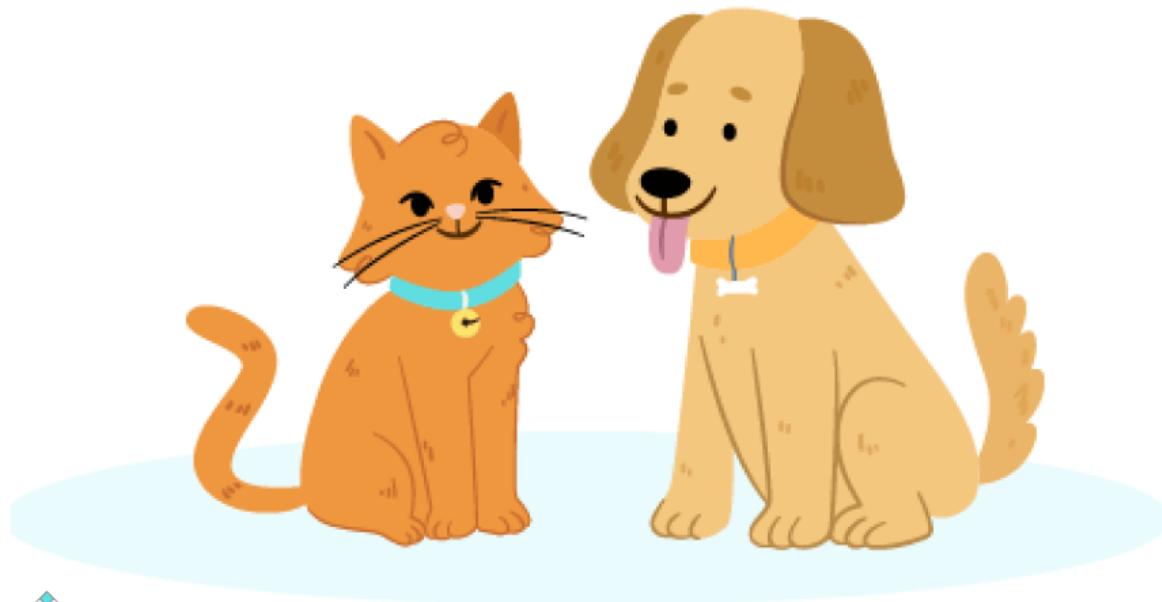
#### 14. Difference between multi-class and multi-label classification problems.



The classification task in a multi-class classification problem has more than two mutually exclusive classes (classes that have no intersection or no attributes in common), whereas in a multi-label classification problem, each label has a

different classification task, although the tasks are related in some way. For example, classifying a group of photographs of animals that could be cats, dogs, or bears is a multi-class classification problem that assumes each sample can be of only one type, implying that an image can be categorized as either a cat or a dog, but not both at the same time.

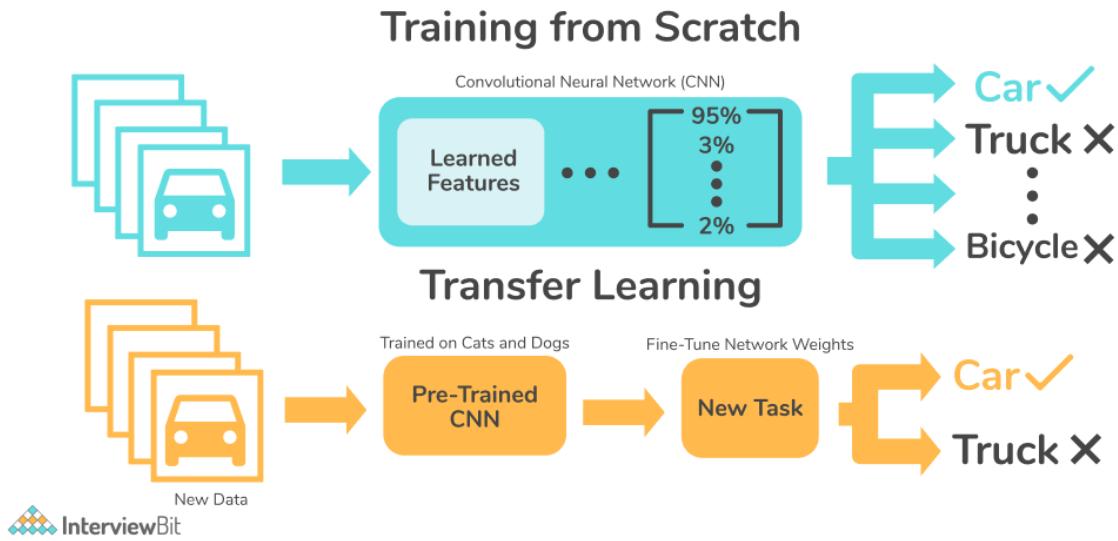
Now let us assume you wish to manipulate the image below.



The image above must be categorized as both a cat and a dog because it depicts both creatures. A set of labels is allocated to each sample in a multi-label classification issue, and the classes are not mutually exclusive. In a multi-label classification problem, a pattern can belong to one or more classes.

## **15. Explain transfer learning in the context of deep learning.**

Transfer learning is a learning technique that allows [data scientists](#) to use what they've learned from a previous machine learning model that was used for a similar task. The ability of humans to transfer their knowledge is used as an example in this learning. You can learn to operate other two-wheeled vehicles more simply if you learn to ride a bicycle. A model trained for autonomous automobile driving can also be used for autonomous truck driving. The features and weights can be used to train the new model, allowing it to be reused. When there is limited data, transfer learning works effectively for quickly training a model.



In the above image, the first diagram represents training a model from scratch while the second diagram represents using a model already trained on cats and dogs to classify the different class of vehicles, thereby representing transfer learning.

## 16. What are the advantages of transfer learning?

Following are the advantages of transfer learning :

- **Better initial model:** In other methods of learning, you must create a model from scratch. Transfer learning is a better starting point because it allows us to perform tasks at a higher level without having to know the details of the starting model.
- **Higher learning rate:** Because the problem has already been taught for a similar task, transfer learning allows for a faster learning rate during training.
- **Higher accuracy after training:** Transfer learning allows a deep learning model to converge at a higher performance level, resulting in more accurate output, thanks to a better starting point and higher learning rate.

## 17. Is it possible to train a neural network model by setting all biases to 0? Also, is it possible to train a neural network model by setting all of the weights to 0?

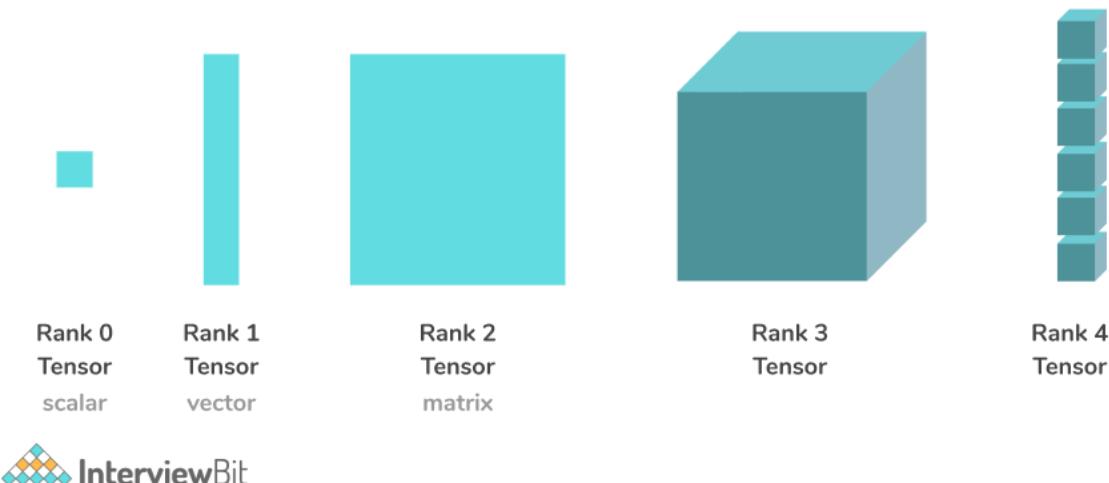
Yes, even if all of the biases are set to zero, the neural network model has a chance of learning.

No, training a model by setting all of the weights to 0 is impossible since the neural network will never learn to complete a task. When all weights are set to zero, the derivatives for each  $w$  remain constant, resulting in neurons learning the same features in each iteration. Any constant initialization of weights, not simply zero, is likely to generate a poor result.

## 18. What is a tensor in deep learning?

A tensor is a multidimensional array that represents a generalization of vectors and matrices. It is one of the key data structures used in deep learning. Tensors are represented as n-dimensional arrays of base data types. The data type of each element in the Tensor is the same, and the data type is always known. It's possible that only a portion of the shape (that is, the number of dimensions and the size of each dimension) is known. Most operations yield fully-known tensors if their inputs are likewise fully known, however, in other circumstances, the shape of a tensor can only be determined at graph execution time.

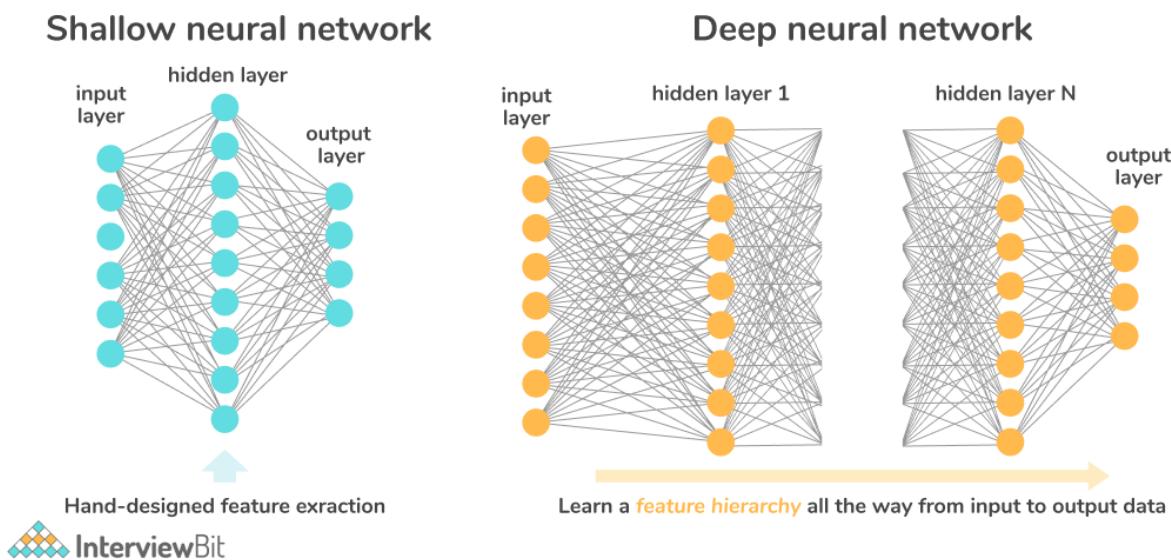
**A tensor is an N-dimensional array of data**



## 19. Explain the difference between a shallow network and a deep network.

A hidden layer, as well as input and output layers, are present in every neural network. Shallow neural networks are those that have only one hidden layer, whereas deep neural networks include numerous hidden layers. Both shallow and deep networks can fit into any function, however, shallow networks require a large number of input parameters, whereas deep networks, because of their several layers, can fit functions with a small number of input parameters. Deep networks are currently favored over shallow networks because the model learns a new and abstract representation of the input at each layer. In comparison to

shallow networks, they are also far more efficient in terms of the number of parameters and computations.



## 20. In a Convolutional Neural Network (CNN), how can you fix the constant validation accuracy?

When training any neural network, constant validation accuracy is a common issue because the network just remembers the sample, resulting in an overfitting problem. Over-fitting a model indicates that the neural network model performs admirably on the training sample, but the model's performance deteriorates on the validation set. Following are some ways for improving CNN's constant validation accuracy:

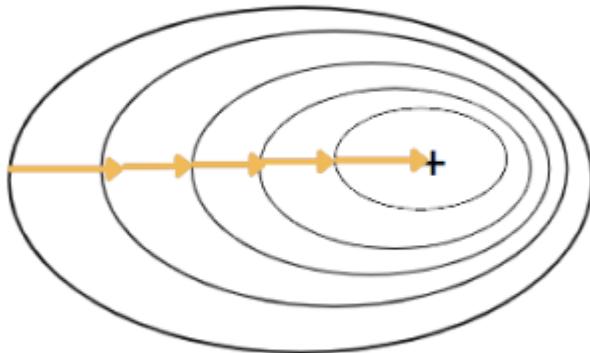
- It is always a good idea to split the dataset into three sections: training, validation, and testing.
- When working with limited data, this difficulty can be handled by experimenting with the neural network's parameters.
- By increasing the training dataset's size.
- By using batch normalization.
- By implementing regularization
- By reducing the complexity of the network

## 21. Explain Batch Gradient Descent.

**Batch Gradient Descent:** Batch Gradient Descent entails computation (involved in each step of gradient descent) over the entire training set at each step and hence it is highly slow on very big training sets. As a result, Batch Gradient Descent becomes extremely computationally expensive. This is ideal

for error manifolds that are convex or somewhat smooth. Batch Gradient Descent also scales nicely as the number of features grows.

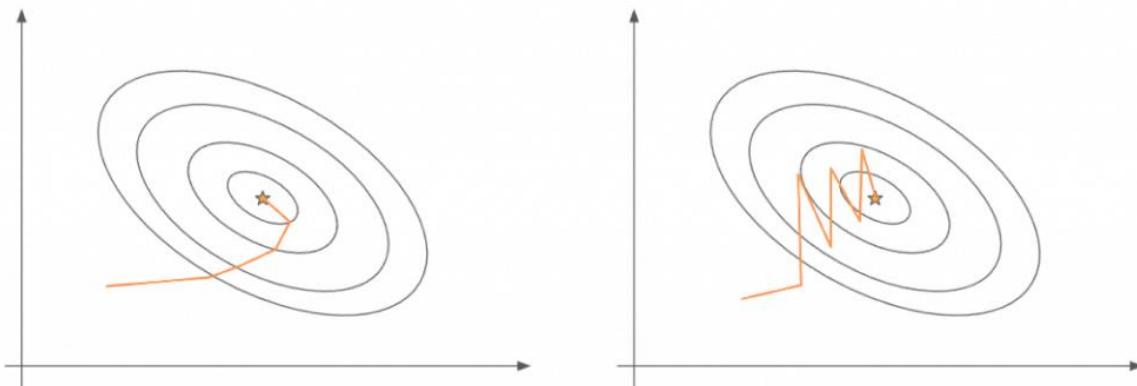
## Batch Gradient Descent



### 22. Explain Stochastic Gradient Descent. How is it different from Batch Gradient Descent?

**Stochastic Gradient Descent:** Stochastic Gradient Descent seeks to tackle the major difficulty with Batch Gradient Descent, which is the use of the entire training set to calculate gradients at each step. It is stochastic in nature, which means it chooses up a "random" instance of training data at each step and then computes the gradient, which is significantly faster than Batch Gradient Descent because there are much fewer data to modify at once. Stochastic Gradient Descent is best suited for unconstrained optimization problems. The stochastic nature of SGD has a drawback in that once it gets close to the minimum value, it doesn't settle down and instead bounces around, giving us a good but not optimal value for model parameters. This can be solved by lowering the learning rate at each step, which will reduce the bouncing and allow SGD to settle down at the global minimum after some time.

Following are the differences between the two:-



**Gradient Descent**

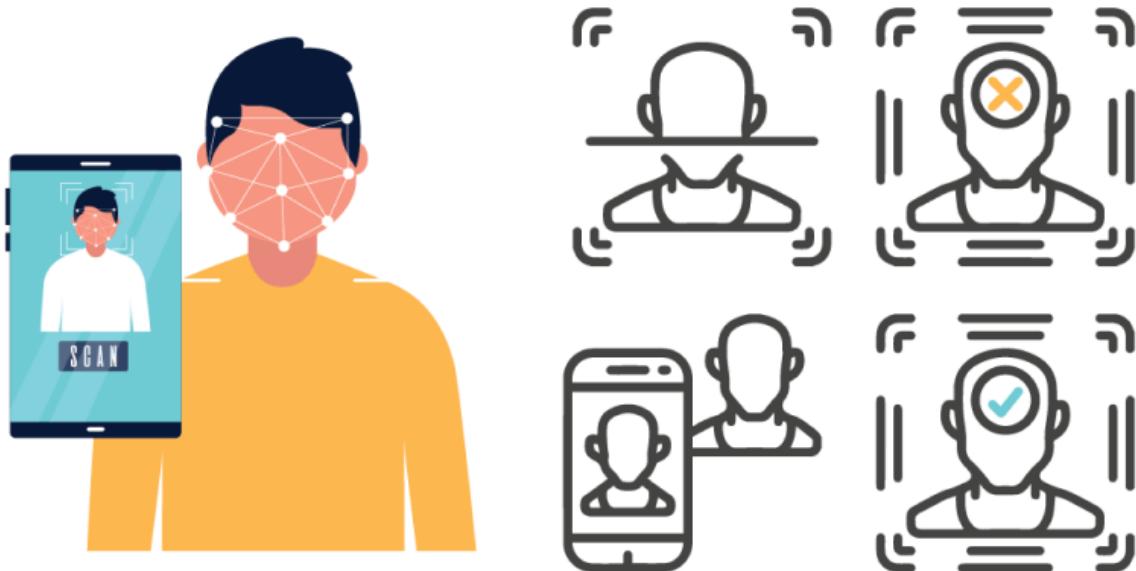


**Stochastic Gradient Descent**

<b>Batch Gradient Descent</b>	<b>Stochastic Gradient Descent</b>
The gradient is calculated using the entire training dataset.	A single training sample is used to compute the gradient.
It is slow and computationally more expensive than Stochastic Gradient Descent.	It is faster and less computationally expensive than Batch Gradient Descent.
It is not recommended for large training samples.	It is recommended for large training samples.
It is deterministic (not random) in nature.	It is stochastic (random) in nature.
Given enough time to converge, it returns the best answer.	It provides a good solution, but not the best.
There is no need to shuffle the data points at random.	Because we want the data sample to be in a random order, we'll shuffle the training set for each epoch.
In this, it is difficult to get out of shallow local minimas.	It has a better chance of escaping shallow local minimas.
In this, the convergence is slow.	It arrives at the convergence point substantially faster.

**23. Which deep learning algorithm is the best for face detection?**

## Face Recognition: FACE ID

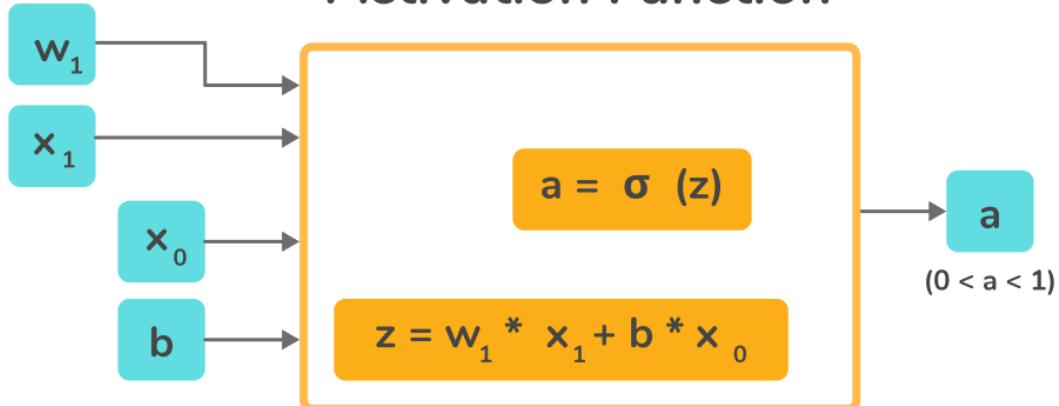


Face identification may be accomplished using a variety of machine learning methods, but the best ones use Convolutional Neural Networks and deep learning. The following are some notable face detection algorithms: FaceNet, Probabilistic, Face Embedding, ArcFace, Cosface, and Spherface.

### 24. What is an activation function? What is the use of an activation function?

An artificial neural network's activation function is a function that is introduced to help the network learn complex patterns in the data. When compared to a neuron-based model seen in our brains, the activation function is responsible for determining what is to be fired to the next neuron at the end of the process. In an ANN, an activation function performs the same job. It takes the preceding cell's output signal and turns it into a format that may be used as input to the next cell.

## Activation Function



Here,  $x_0$  and  $x_1$  are the inputs.  $w_1$  is the weight and  $a$  is the activation function.

The activation function introduces non-linearity into the neural network, allowing it to learn more complex functions. The neural network would only be able to learn a function that is a linear combination of its input data if it didn't have the Activation function.

The activation function converts inputs to outputs. The activation function is in charge of determining whether or not a neuron should be stimulated. It arrives at a decision by calculating the weighted total and then adds bias. The activation function's main goal is to introduce non-linearity into a neuron's output.

### 25. What do you mean by an epochs in the context of deep learning?

An epoch is a terminology used in deep learning that refers to the number of passes the deep learning algorithm has made across the full training dataset. Batches are commonly used to group data sets (especially when the amount of data is very large). The term "iteration" refers to the process of running one batch through the model.

The number of epochs equals the number of iterations if the batch size is the entire training dataset. This is frequently not the case for practical reasons. Several epochs are used in the creation of many models.

There is a general relation which is given by:-

$$d * e = i * b$$

where,

d is the dataset size

e is the number of epochs

i is the number of iterations

b is the batch size

## Deep Learning Interview Questions for Experienced

### 26. While building a neural network architecture, how will you decide how many neurons and the hidden layers should the neural network have?

There is no clear and fast rule for determining the exact number of neurons and hidden layers required to design a neural network architecture given a business problem. The size of the hidden layer in a neural network should be somewhere between the size of the output layers and that of the input layers. However, there are a few basic ways that might help you get a head start on constructing a neural network architecture:

- The best method to approach any unique real-world predictive modelling problem is to start with some basic systematic experimentation to see what would perform best for any given dataset based on previous experience working with neural networks in similar real-world situations. The network configuration can be chosen based on one's understanding of the problem domain and previous expertise with neural networks. The number of layers and neurons employed on similar issues is always a good place to start when evaluating a neural network's configuration.
- It is best to start with simple neural network architecture and gradually increase the complexity of the neural network based on predicted output and accuracy.

### 27. Can a deep learning model be solely built on linear regression?

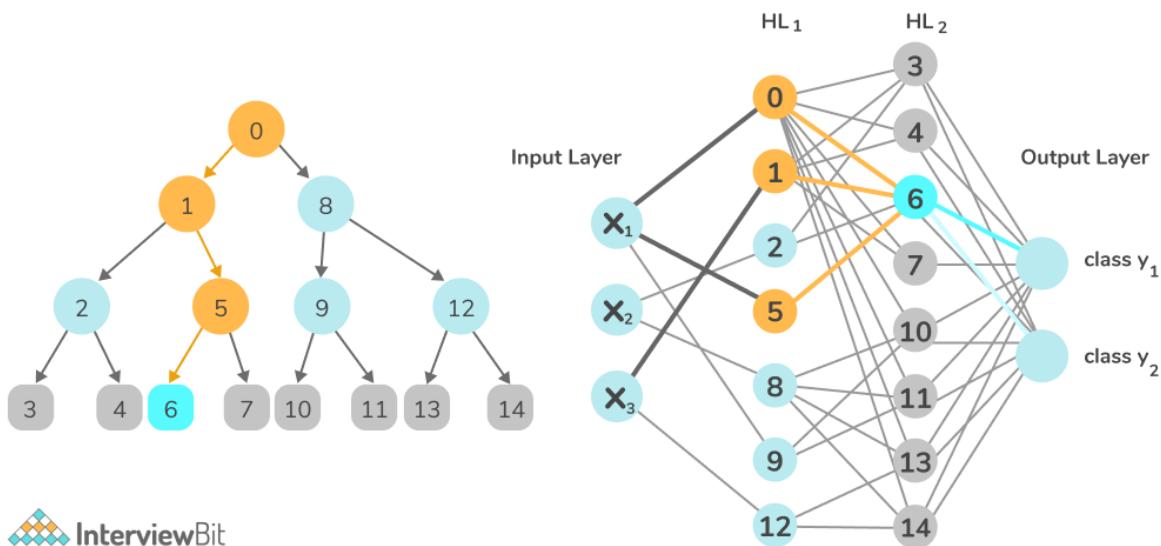
Yes, if the problem is represented by a linear equation, deep networks can be built using a linear function as the activation function for each layer. A problem that is a composition of linear functions, on the other hand, is a linear function, and there is nothing spectacular that can be accomplished by implementing a deep network because adding more nodes to the network will not boost the machine learning model's predictive capacity.

### 28. According to you, which one is more powerful - a two layer neural network without any activation function or a two layer decision tree?

A two-layer neural network is made up of three layers: one input layer, one hidden layer, and one output layer. When dealing with neural networks, an activation function is essential since it is required when dealing with complex and nonlinear functional mappings between inputs and response variables. When there is no activation function in a two-layer neural network, it is simply a linear network. A Neural Network without an Activation function is just a Linear Regression Model, which has limited capability and frequently fails to perform well.

A decision tree with a depth of two layers is known as a two-layer decision tree. Decision Trees are a type of supervised machine learning (that is, the machine is fed with what the input is and what the related output is in the training data) in which the data is continually split according to a parameter. Two entities, decision nodes, and leaves can be used to explain the tree. The decisions or final outcomes are represented by the leaves. And the data is separated at the decision nodes.

When comparing these two models, the two-layer neural network (without activation function) is more powerful than the two-layer decision tree, because the two-layer neural network will consider more attributes while building a model, whereas the two-layer decision tree will only consider 2 or 3 attributes.



The figure on the left depicts a 2 layer decision tree and the figure on the right depicts a 2 layer neural network.

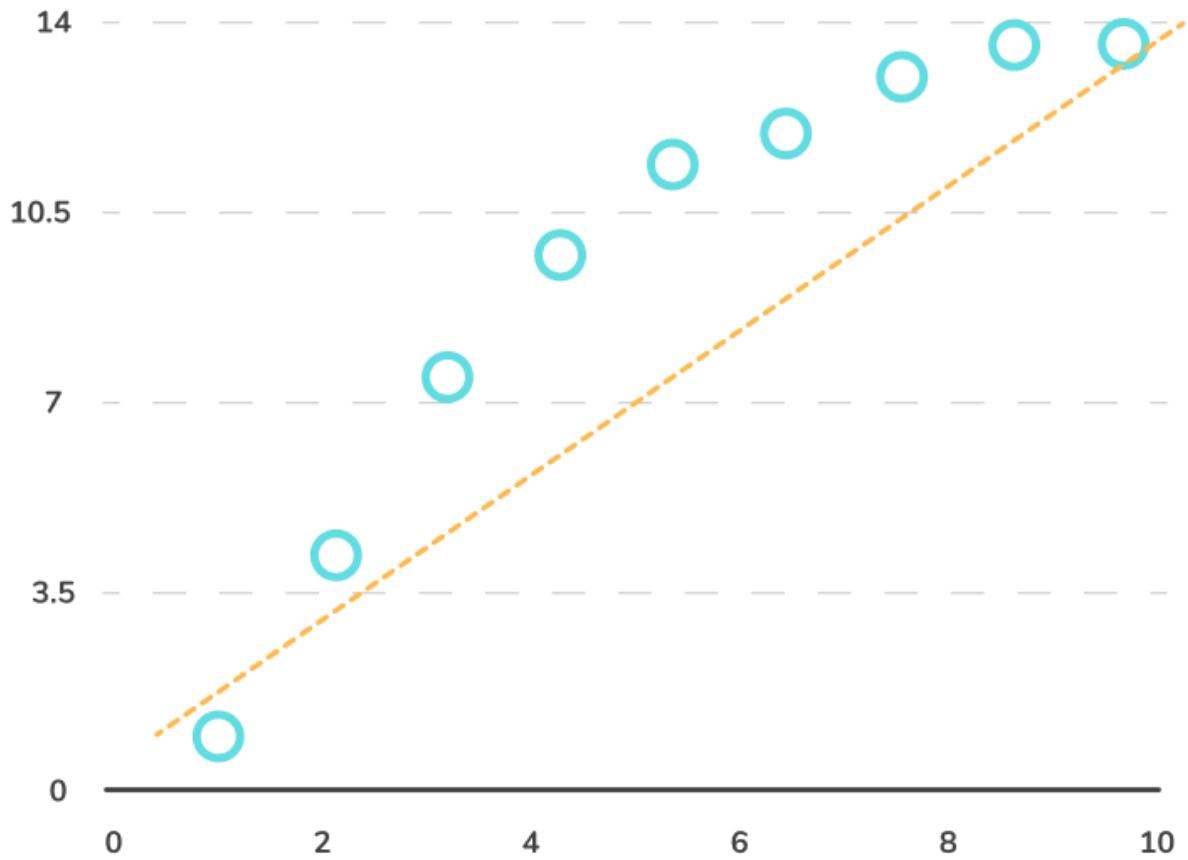
**29. Differentiate between bias and variance in the context of deep learning models. How can you achieve balance between the two?**

Comprehending prediction errors is crucial when it comes to understanding predictions. Reducible (errors that arise due to squared bias or squared variance) and irreducible (errors that arise due to the randomness or natural variability in a system and cannot be reduced by varying the model) mistakes are the two primary types of errors. There are two types of reducible errors: bias and variance. Gaining a thorough grasp of these flaws aids in the construction of an accurate model by preventing overfitting and underfitting.

### **Bias:**

The bias is defined as the difference between the ML model's predicted values and the actual value. Biassing results in a substantial inaccuracy in both training and testing data. To avoid the problem of underfitting, it is advised that an algorithm be low biassed at all times.

The data predicted is in a straight line format due to significant bias, and hence does not fit accurately in the data set. Underfitting of data is the term for this type of fitting. This occurs when the theory is too straightforward or linear. Consider the graph below as an illustration of a situation like this.

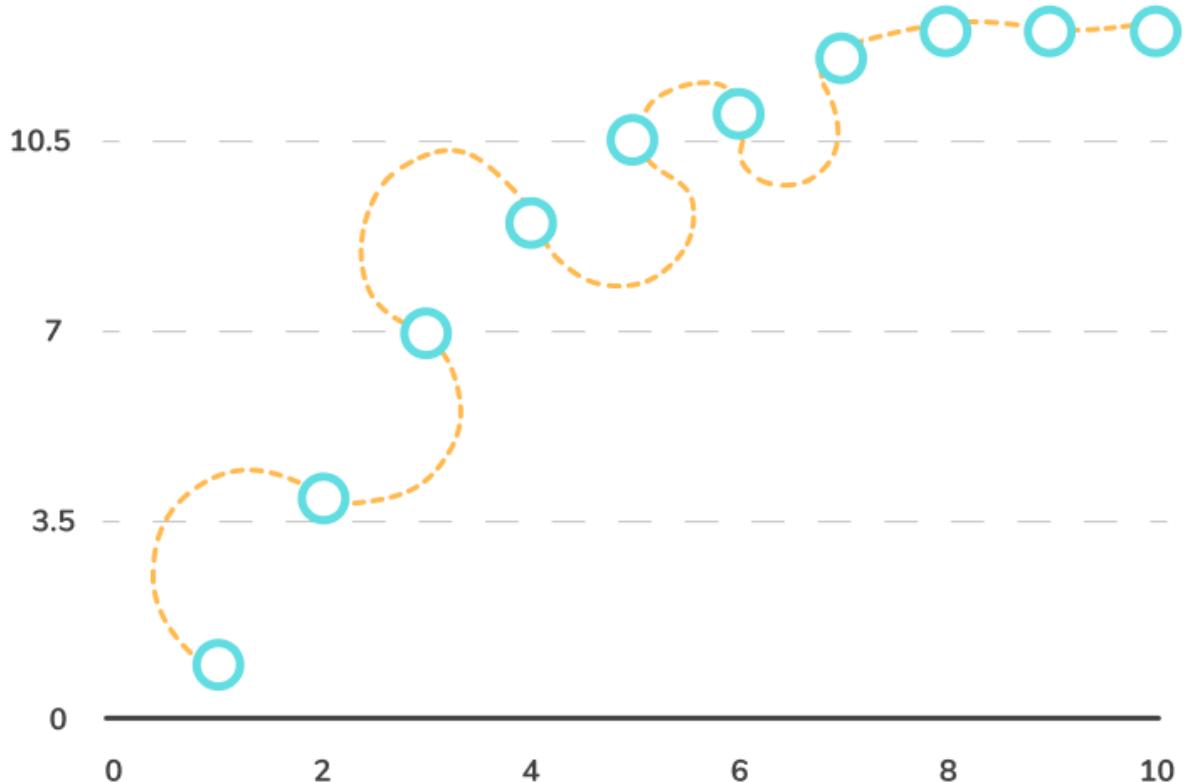


### Variance:

The variance of the model is the variability of model prediction for a given data point, which tells us about the dispersion of our data. It is the difference between the validation error and the training error. The model with high variance has a very complex fit to the training data and so is unable to fit accurately on new data. As a result, while such models perform well on training data, they have high error rates when testing data.

When a model's variance is excessive, it's referred to as Overfitting of Data. Overfitting, which involves accurately fitting the training set using a complicated curve and a high order hypothesis, is not a viable option because the error with unknown data is considerable.

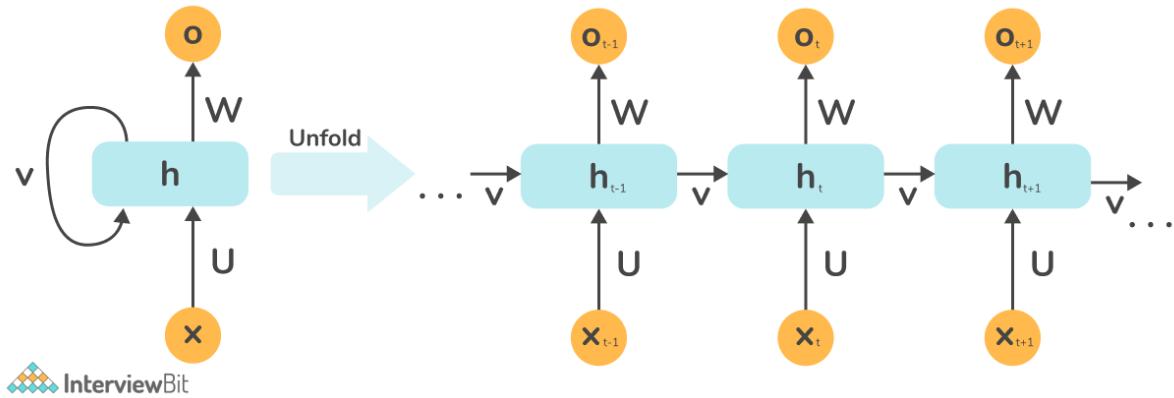
Variance should be kept to a minimum when training a data model.



The model must always aim for a low bias and a low variance in order to achieve the best balance between the two mistakes.

### 30. How does Recurrent Neural Network backpropagation vary from Artificial Neural Network backpropagation?

Backpropagation in Recurrent Neural Networks differ from that of Artificial Neural Networks in the sense that each node in Recurrent Neural Networks has an additional loop as shown in the following image:



This loop, in essence, incorporates a temporal component into the network. This allows for the capture of sequential information from data, which is impossible with a generic artificial neural network.

### **31. What exactly do you mean by exploding and vanishing gradients?**

By taking incremental steps towards the minimal value, the gradient descent algorithm aims to minimize the error. The weights and biases in a neural network are updated using these processes.

However, at times, the steps grow excessively large, resulting in increased updates to weights and bias terms — to the point where the weights overflow (or become NaN, that is, Not a Number). An exploding gradient is the result of this, and it is an unstable method.

On the other hand, if the steps are excessively small, it results in minor – even negligible – changes in the weights and bias terms. As a result, we may end up training a deep learning model with nearly identical weights and biases every time, never reaching the least error function. The vanishing gradient is what it's called.

### **32. What are autoencoders? Explain the different layers of autoencoders.**

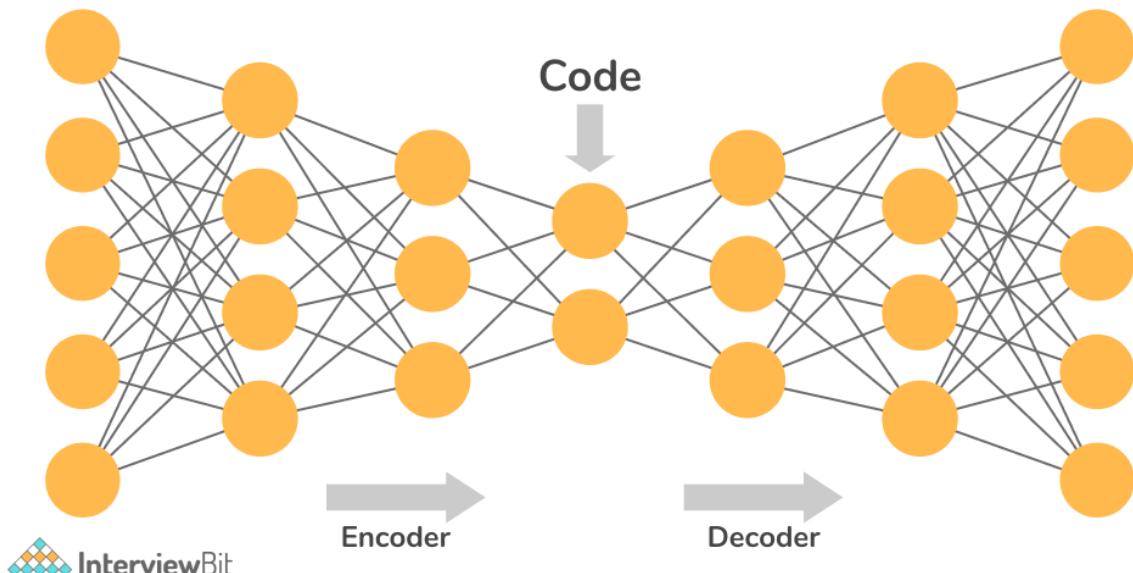
An autoencoder is a type of neural network with the condition that the output layer has the same dimension as that of the input layer. In other words, the number of output units in the output layer is equal to the number of input units in the input layer. An autoencoder is also known as a replicator neural network since it duplicates data from the input to the output in an unsupervised way.

By sending the input through the network, the autoencoders rebuild each dimension of the input. It may appear simple to use a neural network to replicate an input, however, the size of the input is reduced during the replication process, resulting in a smaller representation. In comparison to the

input and output layers, the middle layers of the neural network have fewer units. As a result, the reduced representation of the input is stored in the middle layers. This reduced representation of the input is used to recreate the output.

Following are the different layers in the architecture of autoencoders :

- **Encoder:** An encoder is a fully connected, feedforward neural network that compresses the input image into a latent space representation and encodes it as a compressed representation in a lower dimension. The deformed representation of the original image is the compressed image.
- **Code:** The reduced representation of the input that is supplied into the decoder is stored in this section of the network.
- **Decoder:** Like the encoder, the decoder is a feedforward network with a structure identical to the encoder. This network is in charge of reassembling the input from the code to its original dimensions.



As we can see in the above image, the input is compressed in the encoder, then stored in the Code, and then the original input is decompressed from the code by the decoder. The autoencoder's principal goal is to provide an output that is identical to the input.

### 33. Mention the applications of autoencoders.

Following are the applications of autoencoders:-

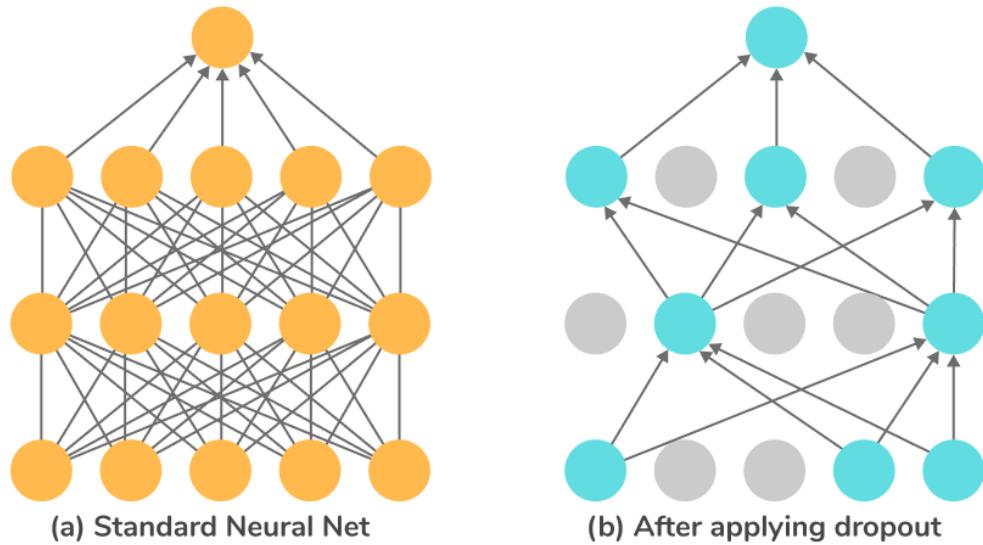
- **Image Denoising:** Denoising images is a skill that autoencoders excel at. A noisy image is one that has been corrupted or has a little amount of noise (that is, random variation of brightness or color information in

images) in it. Image denoising is used to gain accurate information about the image's content.

- **Dimensionality Reduction:** The input is converted into a reduced representation by the autoencoders, which is stored in the middle layer called code. This is where the information from the input has been compressed, and each node may now be treated as a variable by extracting this layer from the model. As a result, we can deduce that by removing the decoder, an autoencoder can be utilised for dimensionality reduction, with the coding layer as the output.
- **Feature Extraction:** The encoding section of Autoencoders aids in the learning of crucial hidden features present in the input data, lowering the reconstruction error. During encoding, a new collection of original feature combinations is created.
- **Image Colorization:** Converting a black-and-white image to a coloured one is one of the applications of autoencoders. We can also convert a colourful image to grayscale.
- **Data Compression:** Autoencoders can be used for data compression. Yet they are rarely used for data compression because of the following reasons:
  - **Lossy compression:** The autoencoder's output is not identical to the input, but it is a near but degraded representation. They are not the best option for lossless compression.
  - **Data-specific:** Autoencoders can only compress data that is identical to the data on which they were trained. They differ from traditional data compression algorithms like jpeg or gzip in that they learn features relevant to the provided training data. As a result, we can't anticipate a landscape photo to be compressed by an autoencoder trained on handwritten digits.

#### 34. What do you know about Dropout?

Dropout is a regularization approach that helps to avoid overfitting and hence improves generalizability (that is, the model predicts correct output for most of the inputs in general, rather than only being limited to the training data set). In general, we should utilize a low dropout value of 20 percent to 50 percent of neurons, with 20% being a decent starting point. A probability that is too low has no effect, whereas a number that is too high causes the network to under-learn.

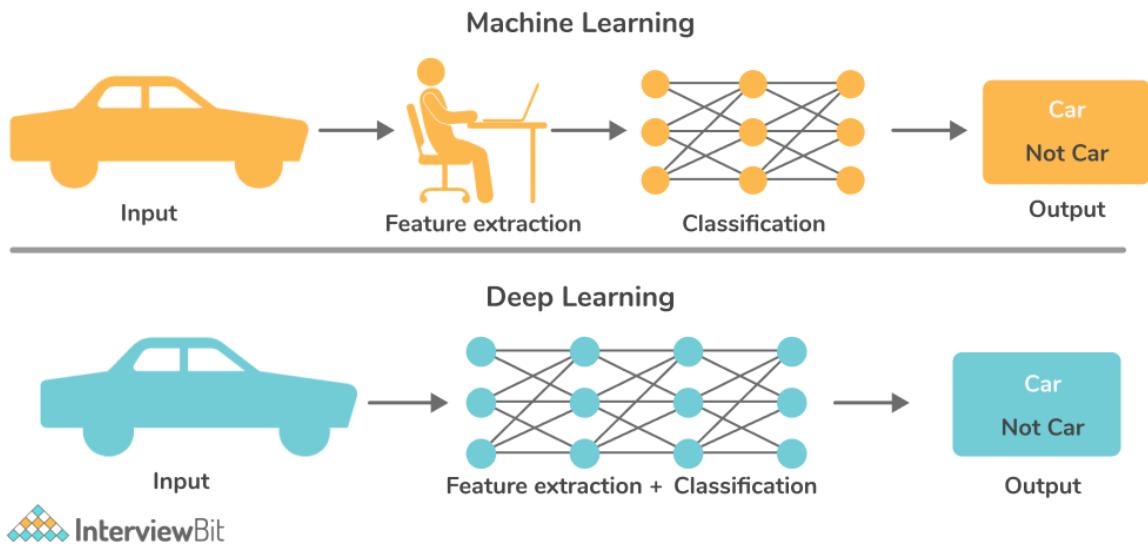


When you employ dropout on a larger network, you're more likely to achieve better results because the model has more opportunities to learn independent representations.

### 35. Differentiate between Deep Learning and Machine Learning.

**Deep Learning:** Deep Learning is a subclass of Machine Learning in which a recurrent neural network and an artificial neural network are linked. The algorithms are constructed in the same way as machine learning algorithms are, however, there are many more levels of algorithms. The artificial neural network refers to all of the algorithm's networks put together. In much simpler terms, it mimics the human brain by connecting all of the neural networks in the brain, which is the concept of deep learning. It uses algorithms and a technique to tackle all types of complex problems.

**Machine Learning:** Machine learning is a subset of Artificial Intelligence (AI) that allows a system to learn and grow from its experiences without having to be programmed to that level. Data is used by Machine Learning to learn and get accurate outcomes. Machine learning algorithms have the ability to learn and improve their performance by gaining more data. Machine learning is currently employed in self-driving cars, cyber fraud detection, face recognition, and Facebook friend suggestion, among other applications. [Learn More](#).



The following table illustrates the difference between them:

Deep Learning	Machine Learning
Deep Learning is a subclass of Machine Learning.	Machine Learning is a super-class of Deep Learning.
Deep Learning employs neural networks to represent data, which is a very distinct data representation (ANN).	Machine Learning represents data in a different way than Deep Learning since it uses structured data.
In this, the output ranges from numerical values to free-form elements such as text or sound.	In this, the output consists of numerical values
It uses a neural network to evaluate data features and relationships by passing data through processing layers.	It uses a variety of automated techniques to convert input into model functions and forecast future actions.
It usually deals with millions of data points.	It usually deals with thousands of data points.
Machine Learning has evolved into Deep Learning. Essentially, it refers to the depth of machine learning.	Artificial Intelligence has evolved into Machine Learning.

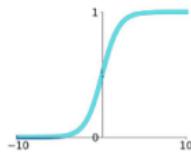
### 36. Explain the different types of activation functions.

Following are the different types of activation functions:

## Activation Functions

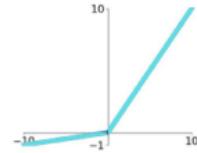
### Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



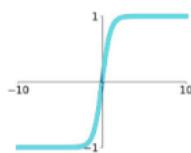
### Leaky ReLU

$$\max(0.1x, x)$$



### tanh

$$\tanh(x)$$

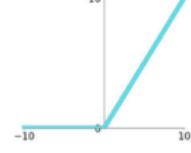


### Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

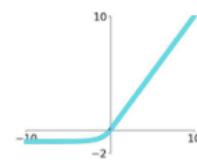
### ReLU

$$\max(0, x)$$



### ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



**Sigmoid function:** The sigmoid function is a non-linear activation function in an ANN that is mostly utilised in feedforward neural networks. It's a differentiable real function with positive derivatives everywhere and a certain degree of smoothness, defined for real input values. The sigmoid function is found in the deep learning models' output layer and is used to anticipate probability-based outputs. The sigmoid function is written as follows:

$$f(x) = \frac{1}{1+e^{-x}} \quad (1.11)$$

**Hyperbolic Tangent Function (Tanh):** The Tanh function is a smoother and zero-centered function having a range of -1 to 1. The output of the tanh function is represented by:

$$f(x) = \frac{\exp(x)}{\sum_j \exp(x_j)} \quad (1.12)$$

$$f(x) = \frac{\exp(x)}{\sum_j \exp(x_j)} \quad (1.12)$$

Because it provides higher training performance for multilayer neural networks, the tanh function is considerably more widely utilised than the sigmoid function. The tanh function's primary advantage is that it gives a zero-centered output, which helps with backpropagation.

**Softmax function:** The softmax function is another type of activation function used in neural networks to generate probability distribution from a vector of real numbers. This function returns a number between 0 and 1, with the sum of the probabilities equal to 1. The softmax function is written like this:

$$f(x_i) = \frac{\exp(x_i)}{\sum_j \exp(x_j)} \quad (1.12)$$

$$f(x_i) = \frac{\exp(x_i)}{\sum_j \exp(x_j)} \quad (1.12)$$

This function is most commonly used in multi-class models, returning probabilities for each class, with the target class having the highest probability. It can be found in practically all of the output layers of the DL architecture.

**Softsign function:** This is most commonly used in regression computation issues and text-to-speech applications based on deep learning. It's a quadratic polynomial with the following representation:

$$f(x) = \frac{x}{x+1} \quad (1.13)$$

**Rectified Linear Unit Function:** The rectified linear unit (ReLU) function is a fast-learning artificial intelligence (AI) that promises to give cutting-edge performance and outstanding results. In deep learning, the ReLU function outperforms other AFs like the sigmoid and tanh functions in terms of performance and generalisation. The function is a roughly linear function that preserves the features of linear models, making gradient-descent approaches easier to optimise.

On each input element, the ReLU function performs a threshold operation, setting all values less than zero to zero. As a result, the ReLU is written as:

$$f(x) = \max(0, x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \quad (1.14)$$

**Exponential Linear Unit Function:** The exponential linear units (ELUs) function is a type of AF that can be used to speed up neural network training (just like ReLU function). The ELU function's major advantage is that it can solve the vanishing gradient problem by employing identity for positive values and boosting the model's learning properties. The exponential linear unit function has the following representation:

$$f(x) = \begin{cases} x, & \text{if } x > 0 \\ e^{x-1} - 1, & \text{if } x \leq 0 \end{cases} \quad (1.22)$$

Which of the following gives non linearity to a neural network?

- Rectified Linear Unit
- Stochastic Gradient Descent
- Convolution Function
- None of the above

2.

The input image has been transformed into a  $28 \times 28$  matrix and a  $7 \times 7$  kernel/filter with a stride of 1. What will the convoluted matrix's size be?

- $20 \times 20$
- $21 \times 21$
- $22 \times 22$
- $25 \times 25$

3.

The input layer has ten nodes, whereas the hidden layer has five. From the input layer to the hidden layer, the maximum number of connections is

- 50
- less than 50
- more than 50
- It is an arbitrary value

4.

If we want to forecast the probabilities of n classes ( $p_1, p_2..p_k$ ), which of the following functions can be utilised as an activation function in the output layer so that the sum of p over all n equals 1?

- Softmax
- ReLu
- Sigmoid
- Tanh

5.

Assume a three-neuron MLP model with inputs 1, 2, and 3. The input neurons' weights are 4,5 and 6, respectively. Assume the activation function is a linear constant value of 3 for the activation function. What will the result be?

- 32
- 64
- 96
- 128

6.

When using Convolutional Neural Network, does max pooling always result in a decrease in parameters?

- True
- False
- Can be true or false
- Cannot say

7.

Which of the following options represents the correct sequence of steps involved in employing a gradient descent algorithm?

1. Calculate the difference between the actual and predicted values.
2. Repeat until you've found the best network weights.
3. Get values from the output layer by passing an input across the network.
4. Create a random weight and bias.
5. To lessen the error, go to each neuron that contributes to the error and modify its value.

- 4, 3, 1, 5, 2
- 1, 2, 3, 4, 5
- 3, 2, 1, 5, 4
- 5, 4, 3, 2, 1

8.

Which strategy does not prevent a model from over-fitting to the training data?

- Early stopping
- Dropout
- Data augmentation
- Pooling

9.

Weight sharing occurs in which neural network architecture?

- Convolutional neural Network
- Recurrent Neural Network
- Fully Connected Neural Network
- Both A and B

10.

What does a Boltzmann machine encompass?

- fully connected network with both hidden and visible units
- asynchronous operation
- stochastic update
- All of the mentioned

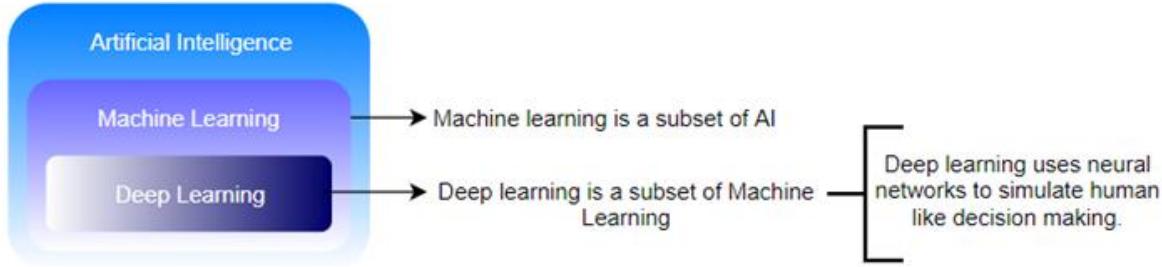
### 1) What is deep learning?

Deep learning is a part of machine learning with an algorithm inspired by the structure and function of the brain, which is called an **artificial neural network**. In the mid-1960s, **Alexey Grigorevich Ivakhnenko** published the first general, while working on deep learning network. Deep learning is suited over a range of fields such as computer vision, speech recognition, natural language processing, etc.

---

### 2) What are the main differences between AI, Machine Learning, and Deep Learning?

- AI stands for Artificial Intelligence. It is a technique which enables machines to mimic human behavior.
- Machine Learning is a subset of AI which uses statistical methods to enable machines to improve with experiences.



- Deep learning is a part of Machine learning, which makes the computation of multi-layer neural networks feasible. It takes advantage of neural networks to simulate human-like decision making.
- 

### 3) Differentiate supervised and unsupervised deep learning procedures.

- Supervised learning is a system in which both input and desired output data are provided. Input and output data are labeled to provide a learning basis for future data processing.
  - Unsupervised procedure does not need labeling information explicitly, and the operations can be carried out without the same. The common unsupervised learning method is **cluster analysis**. It is used for exploratory data analysis to find hidden patterns or grouping in data.
- 

### 4) What are the applications of deep learning?

There are various applications of deep learning:

- Computer vision
- Natural language processing and pattern recognition
- Image recognition and processing
- Machine translation
- Sentiment analysis
- Question Answering system
- Object Classification and Detection
- Automatic Handwriting Generation
- Automatic Text Generation.

---

## 5) Do you think that deep network is better than a shallow one?

Both shallow and deep networks are good enough and capable of approximating any function. But for the same level of accuracy, deeper networks can be much more efficient in terms of computation and number of parameters. Deeper networks can create deep representations. At every layer, the network learns a new, more abstract representation of the input.

---

## 6) What do you mean by "overfitting"?

Overfitting is the most common issue which occurs in deep learning. It usually occurs when a deep learning algorithm apprehends the sound of specific data. It also appears when the particular algorithm is well suitable for the data and shows up when the algorithm or model represents high variance and low bias.

---

## 7) What is Backpropagation?

Backpropagation is a training algorithm which is used for multilayer neural networks. It transfers the error information from the end of the network to all the weights inside the network. It allows the efficient computation of the gradient.

Backpropagation can be divided into the following steps:

- It can forward propagation of training data through the network to generate output.
  - It uses target value and output value to compute error derivative concerning output activations.
  - It can backpropagate to compute the derivative of the error concerning output activations in the previous layer and continue for all hidden layers.
  - It uses the previously calculated derivatives for output and all hidden layers to calculate the error derivative concerning weights.
  - It updates the weights.
- 

## 8) What is the function of the Fourier Transform in Deep Learning?

Fourier transform package is highly efficient for analyzing, maintaining, and managing a large databases. The software is created with a high-quality feature known as the **special portrayal**. One can effectively utilize it to generate real-time array data, which is extremely helpful for processing all categories of signals.

---

9) Describe the theory of autonomous form of deep learning in a few words.

There are several forms and categories available for the particular subject, but the autonomous pattern represents independent or unspecified mathematical bases which are free from any specific categorizer or formula.

---

10) What is the use of Deep learning in today's age, and how is it adding data scientists?

Deep learning has brought significant changes or revolution in the field of machine learning and data science. The concept of a **complex neural network** (CNN) is the main center of attention for data scientists. It is widely taken because of its advantages in performing next-level machine learning operations. The advantages of deep learning also include the process of clarifying and simplifying issues based on an algorithm due to its utmost flexible and adaptable nature. It is one of the rare procedures which allow the movement of data in independent pathways. Most of the data scientists are viewing this particular medium as an advanced additive and extended way to the existing process of machine learning and utilizing the same for solving complex day to day issues.

---

11) What are the deep learning frameworks or tools?

Deep learning frameworks or tools are:

Tensorflow, Keras, Chainer, Pytorch, Theano & Ecosystem, Caffe2, CNTK, DyNetGensim, DSSTNE, Gluon, Paddle, Mxnet, BigDL

---

12) What are the disadvantages of deep learning?

There are some disadvantages of deep learning, which are:

- Deep learning model takes longer time to execute the model. In some cases, it even takes several days to execute a single model depends on complexity.
  - The deep learning model is not good for small data sets, and it fails here.
- 

### 13) What is the meaning of term weight initialization in neural networks?

In neural networking, weight initialization is one of the essential factors. A bad weight initialization prevents a network from learning. On the other side, a good weight initialization helps in giving a quicker convergence and a better overall error. Biases can be initialized to zero. The standard rule for setting the weights is to be close to zero without being too small.

---

### 14) Explain Data Normalization.

Data normalization is an essential preprocessing step, which is used to rescale values to fit in a specific range. It assures better convergence during backpropagation. In general, data normalization boils down to subtracting the mean of each data point and dividing by its standard deviation.

---

### 15) Why is zero initialization not a good weight initialization process?

If the set of weights in the network is put to a zero, then all the neurons at each layer will start producing the same output and the same gradients during backpropagation.

As a result, the network cannot learn at all because there is no source of asymmetry between neurons. That is the reason why we need to add randomness to the weight initialization process.

---

### 16) What are the prerequisites for starting in Deep Learning?

There are some basic requirements for starting in Deep Learning, which are:

- Machine Learning

- Mathematics
  - Python Programming
- 

17) What are the supervised learning algorithms in Deep learning?

- Artificial neural network
  - Convolution neural network
  - Recurrent neural network
- 

18) What are the unsupervised learning algorithms in Deep learning?

- Self Organizing Maps
  - Deep belief networks (Boltzmann Machine)
  - Auto Encoders
- 

19) How many layers in the neural network?

- **Input** **Layer**  
The input layer contains input neurons which send information to the hidden layer.
  - **Hidden** **Layer**  
The hidden layer is used to send data to the output layer.
  - **Output** **Layer**  
The data is made available at the output layer.
- 

20) What is the use of the Activation function?

The activation function is used to introduce nonlinearity into the neural network so that it can learn more complex function. Without the Activation function, the neural network would be only able to learn function, which is a linear combination of its input data.

Activation function translates the inputs into outputs. The activation function is responsible for deciding whether a neuron should be activated or not. It makes the decision by calculating the weighted sum and further adding bias with it. The basic purpose of the activation function is to introduce non-linearity into the output of a neuron.

---

## 21) How many types of activation function are available?

- Binary Step
  - Sigmoid
  - Tanh
  - ReLU
  - Leaky ReLU
  - Softmax
  - Swish
- 

## 22) What is a binary step function?

The binary step function is an activation function, which is usually based on a threshold. If the input value is above or below a particular threshold limit, the neuron is activated, then it sends the same signal to the next layer. This function does not allow multi-value outputs.

---

## 23) What is the sigmoid function?

The sigmoid activation function is also called the logistic function. It is traditionally a trendy activation function for neural networks. The input data to the function is transformed into a value between **0.0** and **1.0**. Input values that are much larger than 1.0 are transformed to the value 1.0. Similarly, values that are much smaller than 0.0 are transformed into 0.0. The shape of the function for all possible inputs is an S-shape from zero up through 0.5 to 1.0. It was the default activation used on neural networks, in the early 1990s.

---

## 24) What is Tanh function?

The hyperbolic tangent function, also known as tanh for short, is a similar shaped nonlinear activation function. It provides output values between **-1.0** and **1.0**. Later in the 1990s and through the 2000s, this function was preferred over the sigmoid activation function as models. It was easier to train and often had better predictive performance.

---

## 25) What is ReLU function?

A node or unit which implements the activation function is referred to as a **rectified linear activation unit** or ReLU for short. Generally, networks that use the rectifier function for the hidden layers are referred to as **rectified networks**.

Adoption of ReLU may easily be considered one of the few milestones in the deep learning revolution.

---

## 26) What is the use of leaky ReLU function?

The Leaky ReLU (LReLU or LReL) manages the function to allow small negative values when the input is less than zero.

---

## 27) What is the softmax function?

The softmax function is used to calculate the probability distribution of the event over 'n' different events. One of the main advantages of using softmax is the output probabilities range. The range will be between 0 to 1, and the sum of all the probabilities will be equal to one. When the softmax function is used for multi-classification model, it returns the probabilities of each class, and the target class will have a high probability.

---

## 28) What is a Swish function?

Swish is a new, self-gated activation function. Researchers at Google discovered the Swish function. According to their paper, it performs better than ReLU with a similar level of computational efficiency.

---

### 29) What is the most used activation function?

**Relu function** is the most used activation function. It helps us to solve vanishing gradient problems.

---

### 30) Can Relu function be used in output layer?

No, Relu function has to be used in hidden layers.

---

### 31) In which layer softmax activation function used?

Softmax activation function has to be used in the output layer.

---

### 32) What do you understand by Autoencoder?

Autoencoder is an artificial neural network. It can learn representation for a set of data without any supervision. The network automatically learns by copying its input to the output; typically, internet representation consists of smaller dimensions than the input vector. As a result, they can learn efficient ways of representing the data. Autoencoder consists of two parts; an encoder tries to fit the inputs to the internal representation, and a decoder converts the internal state to the outputs.

---

### 33) What do you mean by Dropout?

Dropout is a cheap regulation technique used for reducing overfitting in neural networks. We randomly drop out a set of nodes at each training step. As a result, we create a different model for each training case, and all of these models share weights. It's a form of model averaging.

---

### 34) What do you understand by Tensors?

Tensors are nothing but a de facto for representing the data in deep learning. They are just multidimensional arrays, which allows us to represent the data having higher dimensions. In general, we deal with high dimensional data sets where dimensions refer to different features present in the data set.

---

### 35) What do you understand by Boltzmann Machine?

A Boltzmann machine (also known as stochastic Hopfield network with hidden units) is a type of recurrent neural network. In a Boltzmann machine, nodes make binary decisions with some bias. Boltzmann machines can be strung together to create more sophisticated systems such as deep belief networks. Boltzmann Machines can be used to optimize the solution to a problem.

Some important points about Boltzmann Machine-

- It uses a recurrent structure.
  - It consists of stochastic neurons, which include one of the two possible states, either 1 or 0.
  - The neurons present in this are either in an adaptive state (free state) or clamped state (frozen state).
  - If we apply simulated annealing or discrete Hopfield network, then it would become a Boltzmann Machine.
- 

### 36) What is Model Capacity?

The capacity of a deep learning neural network controls the scope of the types of mapping functions that it can learn. Model capacity can approximate any given function. When there is a higher model capacity, it means that the larger amount of information can be stored in the network.

---

### 37) What is the cost function?

A cost function describes us how well the neural network is performing with respect to its given training sample and the expected output. It may depend on variables such as weights and biases. It provides the performance of a neural

network as a whole. In deep learning, our priority is to minimize the cost function. That's why we prefer to use the concept of gradient descent.

---

### 38) Explain gradient descent?

An optimization algorithm that is used to minimize some function by repeatedly moving in the direction of steepest descent as specified by the negative of the gradient is known as gradient descent. It's an iteration algorithm, in every iteration algorithm, we compute the gradient of a cost function, concerning each parameter and update the parameter of the function via the following formula:

$$\Theta := \Theta - \alpha \frac{d}{d\Theta} J(\Theta)$$

Where,

**$\Theta$  - is the parameter vector,**

**$\alpha$  - learning rate,**

**$J(\Theta)$  - is a cost function**

In machine learning, it is used to update the parameters of our model. Parameters represent the coefficients in linear regression and weights in neural networks.

---

### 39) Explain the following variant of Gradient Descent: Stochastic, Batch, and Mini-batch?

- |  |                 |                |
|--|-----------------|----------------|
| ○ <b>Stochastic</b>  | <b>Gradient</b> | <b>Descent</b> |
| Stochastic gradient descent is used to calculate the gradient and update the parameters by using only a single training example. |                 |                |
| ○ <b>Batch</b>   | <b>Gradient</b> | <b>Descent</b> |
| Batch gradient descent is used to calculate the gradients for the whole dataset and perform just one update at each iteration.   |                 |                |
| ○ <b>Mini-batch</b>  | <b>Gradient</b> | <b>Descent</b> |
| Mini-batch gradient descent is a variation of stochastic gradient descent.   |                 |                |

Instead of a single training example, mini-batch of samples is used. Mini-batch gradient descent is one of the most popular optimization algorithms.

---

#### 40) What are the main benefits of Mini-batch Gradient Descent?

- It is computationally efficient compared to stochastic gradient descent.
  - It improves generalization by finding flat minima.
  - It improves convergence by using mini-batches. We can approximate the gradient of the entire training set, which might help to avoid local minima.
- 

#### 41) What is matrix element-wise multiplication? Explain with an example.

Element-wise matrix multiplication is used to take two matrices of the same dimensions. It further produces another combined matrix with the elements that are a product of corresponding elements of matrix a and b.

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \circ \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix} =$$

$$\begin{pmatrix} a_{11}b_{11} & a_{12}b_{12} & a_{13}b_{13} \\ a_{21}b_{21} & a_{22}b_{22} & a_{23}b_{23} \\ a_{31}b_{31} & a_{32}b_{32} & a_{33}b_{33} \end{pmatrix}$$

---

#### 42) What do you understand by a convolutional neural network?

A convolutional neural network, often called CNN, is a feedforward neural network. It uses convolution in at least one of its layers. The convolutional layer contains a set of filter (kernels). This filter is sliding across the entire input image, computing the dot product between the weights of the filter and the input image. As a result of training, the network automatically learns filters that can detect specific features.

---

#### 43) Explain the different layers of CNN.

There are four layered concepts that we should understand in CNN (Convolutional Neural Network):

- **Convolution**

This layer comprises of a set of independent filters. All these filters are initialized randomly. These filters then become our parameters which will be learned by the network subsequently.

- **ReLU**

The ReLu layer is used with the convolutional layer.

- **Pooling**

It reduces the spatial size of the representation to lower the number of parameters and computation in the network. This layer operates on each feature map independently.

- **Full**

**Collectedness**

Neurons in a completely connected layer have complete connections to all activations in the previous layer, as seen in regular Neural Networks. Their activations can be easily computed with a matrix multiplication followed by a bias offset.

---

#### 44) What is an RNN?

RNN stands for Recurrent Neural Networks. These are the artificial neural networks which are designed to recognize patterns in sequences of data such as handwriting, text, the spoken word, genomes, and numerical time series data. RNN use backpropagation algorithm for training because of their internal memory. RNN can remember important things about the input they received, which enables them to be very precise in predicting what's coming next.

---

#### 45) What are the issues faced while training in Recurrent Networks?

Recurrent Neural Network uses backpropagation algorithm for training, but it is applied on every timestamp. It is usually known as **Back-propagation Through Time** (BTT).

There are two significant issues with Back-propagation, such as:

- **Vanishing Gradient**  
When we perform Back-propagation, the gradients tend to get smaller and smaller because we keep on moving backward in the Network. As a result, the neurons in the earlier layer learn very slowly if we compare it with the neurons in the later layers. Earlier layers are more valuable because they are responsible for learning and detecting simple patterns. They are the building blocks of the network. If they provide improper or inaccurate results, then how can we expect the next layers and complete network to perform nicely and provide accurate results. The training procedure takes long, and the prediction accuracy of the model decreases.
  - **Exploding Gradient**  
Exploding gradients are the main problem when large error gradients accumulate. They provide result in very large updates to neural network model weights during training. Gradient Descent process works best when updates are small and controlled. When the magnitudes of the gradient accumulate, an unstable network is likely to occur. It can cause poor prediction of results or even a model that reports nothing useful.
- 

#### 46) Explain the importance of LSTM.

LSTM stands for **Long short-term memory**. It is an artificial RNN (Recurrent Neural Network) architecture, which is used in the field of deep learning. LSTM has feedback connections which makes it a "general purpose computer." It can process not only a single data point but also entire sequences of data.

They are a special kind of RNN which are capable of learning long-term dependencies.

---

#### 47) What are the different layers of Autoencoders? Explain briefly.

An autoencoder contains three layers:

- **Encoder**  
The encoder is used to compress the input into a latent space representation.

It encodes the input images as a compressed representation in a reduced dimension. The compressed images are the distorted version of the original image.

- **Code**

The code layer is used to represent the compressed input which is fed to the decoder.

- **Decoder**

The decoder layer decodes the encoded image back to its original dimension. The decoded image is a reduced reconstruction of the original image. It is automatically reconstructed from the latent space representation.

---

#### 48) What do you understand by Deep Autoencoders?

Deep Autoencoder is the extension of the simple Autoencoder. The first layer present in DeepAutoencoder is responsible for first-order functions in the raw input. The second layer is responsible for second-order functions corresponding to patterns in the appearance of first-order functions. Deeper layers which are available in the Deep Autoencoder tend to learn even high-order features.

A deep autoencoder is the combination of two, symmetrical deep-belief networks:

- First four or five shallow layers represent the encoding half.
  - The other combination of four or five layers makes up the decoding half.
- 

#### 49) What are the three steps to developing the necessary assumption structure in Deep learning?

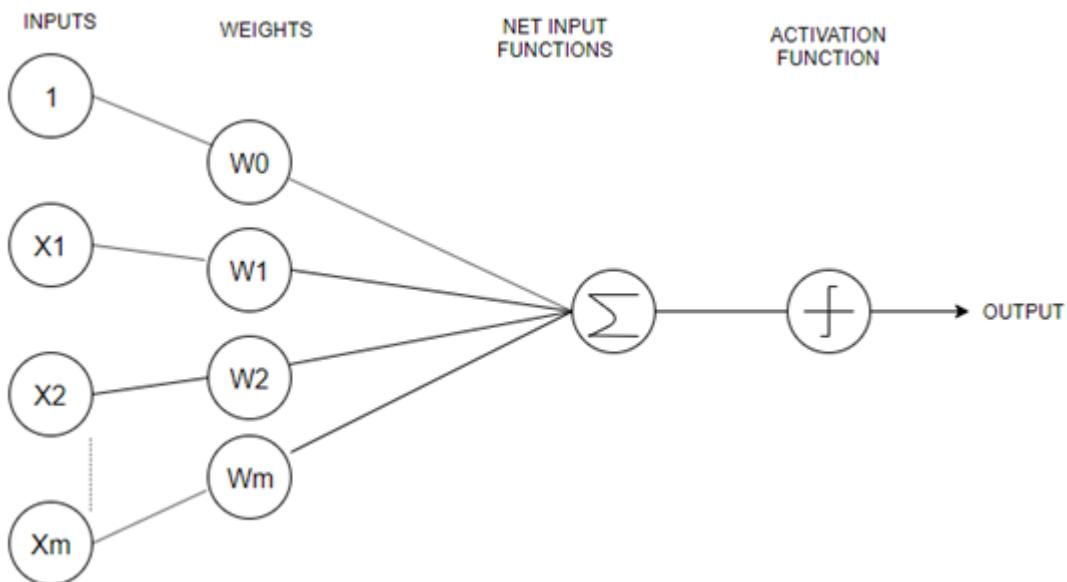
The procedure of developing an assumption structure involves three specific actions.

- The first step contains algorithm development. This particular process is lengthy.
- The second step contains algorithm analyzing, which represents the in-process methodology.

- The third step is about implementing the general algorithm in the final procedure. The entire framework is interlinked and required for throughout the process.
- 

50) What do you understand by Perceptron? Also, explain its type.

A perceptron is a neural network unit (an artificial neuron) that does certain computations to detect features. It is an algorithm for supervised learning of binary classifiers. This algorithm is used to enable neurons to learn and processes elements in the training set one at a time.



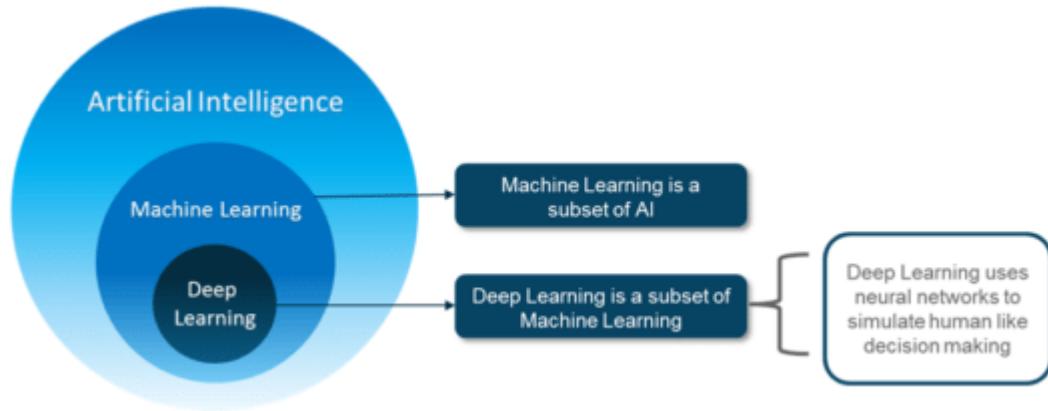
There are two types of perceptrons:

- Single-Layer Perceptron**  
Single layer perceptrons can learn only linearly separable patterns.
- Multilayer Perceptrons**  
Multilayer perceptrons or feedforward neural networks with two or more layers have the higher processing power.

## Q1. Differentiate between AI, Machine Learning and Deep Learning.

Artificial Intelligence is a technique that enables machines to mimic human behavior.

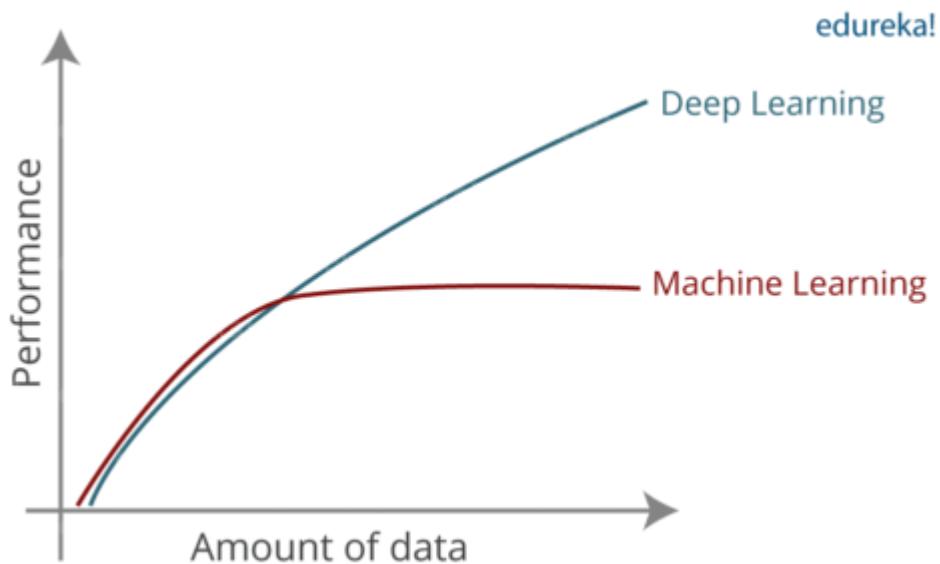
Machine Learning is a subset of AI technique which uses statistical methods to enable machines to improve with experience.



Deep learning is a subset of ML which make the computation of multi-layer neural network feasible. It uses Neural networks to simulate human-like decision making.

**Q2. Do you think Deep Learning is Better than Machine Learning? If so, why?**

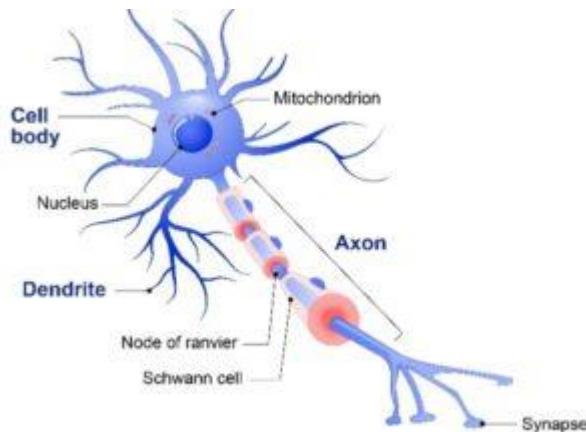
Though traditional ML algorithms solve a lot of our cases, they are not useful while working with high dimensional data, that is where we have a large number of inputs and outputs. For example, in the case of handwriting recognition, we have a large amount of input where we will have a different type of inputs associated with different type of handwriting.



The second major challenge is to tell the computer what are the features it should look for that will play an important role in predicting the outcome as well as to achieve better accuracy while doing so.

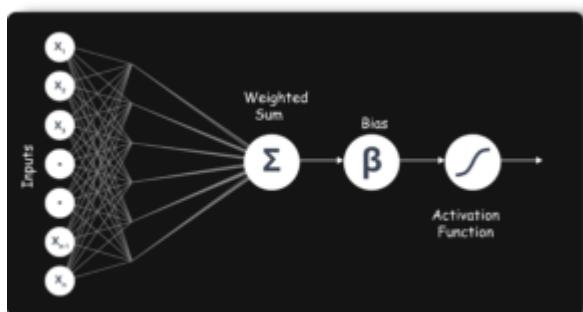
### Q3. What is Perceptron? And How does it Work?

If we focus on the structure of a biological neuron, it has dendrites which are used to receive inputs. These inputs are summed in the cell body and using the Axon it is passed on to the next biological neuron as shown below.



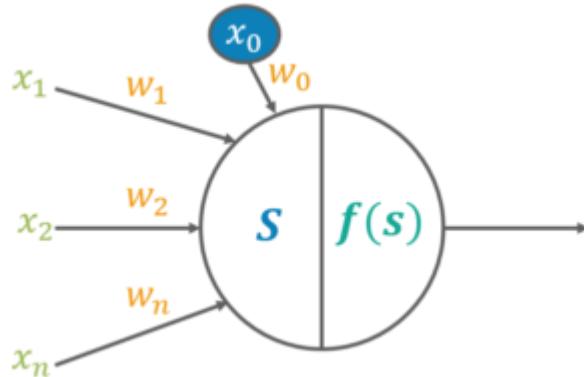
- **Dendrite:** Receives signals from other neurons
- **Cell Body:** Sums all the inputs
- **Axon:** It is used to transmit signals to the other cells

Similarly, a perceptron receives multiple inputs, applies various transformations and functions and provides an output. A Perceptron is a linear model used for binary classification. It models a neuron which has a set of inputs, each of which is given a specific weight. The neuron computes some function on these weighted inputs and gives the output.



#### **Q4. What is the role of weights and bias?**

For a perceptron, there can be one more input called **bias**. While the weights determine the **slope** of the classifier line, bias allows us to shift the line towards left or right. Normally bias is treated as another weighted input with the input value  $x_0$ .



#### **Q5. What are the activation functions?**

Activation function translates the inputs into outputs. Activation function decides whether a neuron should be activated or not by calculating the weighted sum and further adding bias with it. The purpose of the activation function is to introduce non-linearity into the output of a neuron.

There can be many Activation functions like:

- Linear or Identity
- Unit or Binary Step
- Sigmoid or Logistic
- Tanh
- ReLU
- Softmax

#### **Q6. Explain Learning of a Perceptron.**

1. Initializing the weights and threshold.
2. Provide the input and calculate the output.
3. Update the weights.

4. Repeat Steps 2 and 3

$$W_j(t+1) = W_j(t) + n(d-y)x$$

**W<sub>j</sub> (t+1)** – Updated Weight

**W<sub>j</sub> (t)** – Old Weight

**d** – Desired Output

**y** – Actual Output

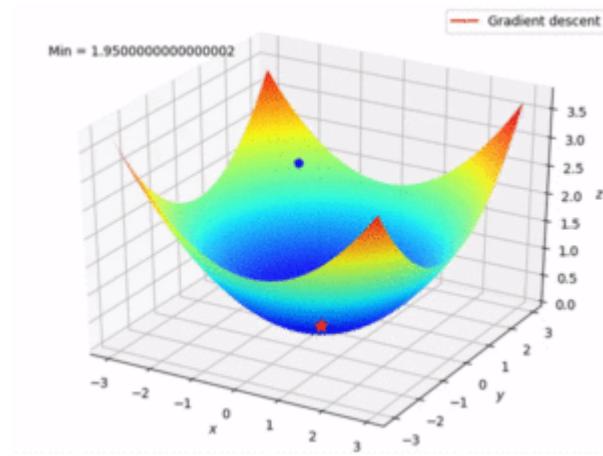
**x** – Input

## Q7. What is the significance of a Cost/Loss function?

A cost function is **a measure of the accuracy** of the neural network with respect to a given training sample and expected output. It provides the performance of a neural network as a whole. In deep learning, the goal is to minimize the cost function. For that, we use the concept of gradient descent.

## Q8. What is gradient descent?

**Gradient descent** is an optimization algorithm used to minimize some function by iteratively moving in the direction of steepest descent as defined by the negative of the gradient.



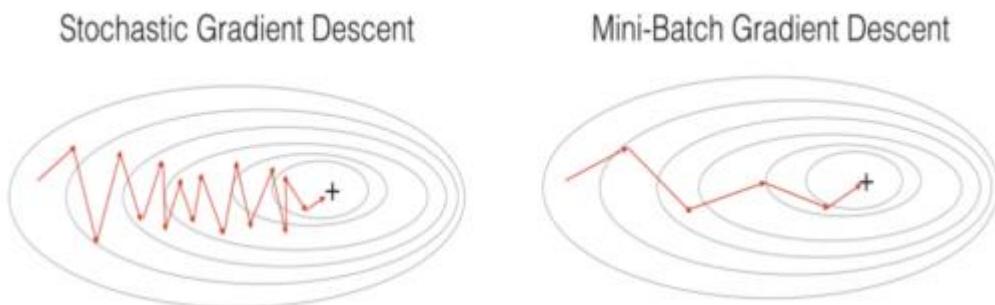
**Stochastic Gradient Descent:** Uses only a single training example to calculate the gradient and update parameters.

**Batch Gradient Descent:** Calculate the gradients for the whole dataset and perform just one update at each iteration.

**Mini-batch Gradient Descent:** Mini-batch gradient is a variation of stochastic gradient descent where instead of single training example, mini-batch of samples is used. It's one of the most popular optimization algorithms.

### Q9. What are the benefits of mini-batch gradient descent?

- This is more efficient compared to stochastic gradient descent.
- The generalization by finding the flat minima.
- Mini-batches allows help to approximate the gradient of the entire training set which helps us to avoid local minima.



### Q10. What are the steps for using a gradient descent algorithm?

- Initialize random weight and bias.
- Pass an input through the network and get values from the output layer.
- Calculate the error between the actual value and the predicted value.
- Go to each neuron which contributes to the error and then change its respective values to reduce the error.
- Reiterate until you find the best weights of the network.

### Q11. Create a Gradient Descent in python.

```
1           params = [weights_hidden, weights_output, bias_hidden, bias_output]
2
3           def sgd(cost, params, lr=0.05):
```

```

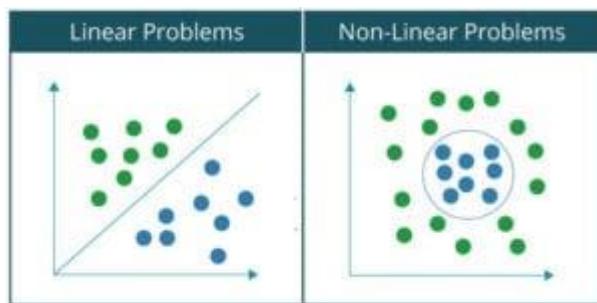
4
5             grads = T.grad(cost=cost, wrt=params)
6             updates = []
7
8             for p, g in zip(params, grads):
9                 updates.append([p, p - g * lr])
10
11            return updates
12
13        updates = sgd(cost, params)

```

## Q12. What are the shortcomings of a single layer perceptron?

Well, there are two major problems:

- Single-Layer Perceptrons cannot classify non-linearly separable data points.
- Complex problems, that involve a lot of parameters cannot be solved by Single-Layer Perceptrons



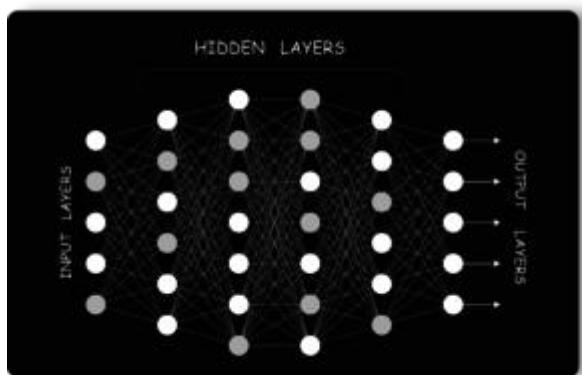
## Q13. What is a Multi-Layer-Perceptron

A multilayer perceptron (MLP) is a deep, artificial neural network. It is composed of more than one perceptron. They are composed of an input layer to receive the signal, an output layer that makes a decision or prediction about the input, and in between those two, an arbitrary number of hidden layers that are the true computational engine of the MLP.

## **Q14. What are the different parts of a multi-layer perceptron?**

**Input Nodes:** The Input nodes provide information from the outside world to the network and are together referred to as the “Input Layer”. No computation is performed in any of the Input nodes – they just pass on the information to the hidden nodes.

**Hidden Nodes:** The Hidden nodes perform computations and transfer information from the input nodes to the output nodes. A collection of hidden nodes forms a “Hidden Layer”. While a network will only have a single input layer and a single output layer, it can have zero or multiple Hidden Layers.



**Output Nodes:** The Output nodes are collectively referred to as the “Output Layer” and are responsible for computations and transferring information from the network to the outside world.

## **Q15. What Is Data Normalization And Why Do We Need It?**

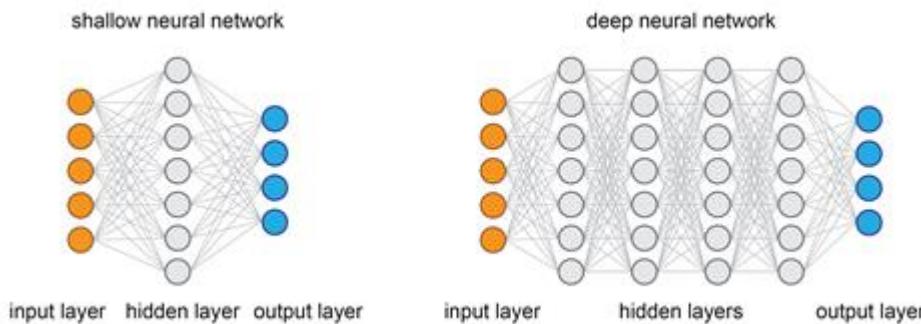
Data normalization is very important preprocessing step, used to rescale values to fit in a specific range to assure better convergence during backpropagation. In general, it boils down to subtracting the mean of each data point and dividing by its standard deviation.

These were some basic Deep Learning Interview Questions. Now, let's move on to some advanced ones.

## **Q16. Which is Better Deep Networks or Shallow ones? and Why?**

Both the Networks, be it shallow or Deep are capable of approximating any function. But what matters is how precise that network is in terms of getting the

results. A shallow network works with only a few features, as it can't extract more. But a deep network goes deep by computing efficiently and working on more features/parameters.



## Q17. Why is Weight Initialization important in Neural Networks?

Weight initialization is one of the very important steps. A bad weight initialization can prevent a network from learning but good weight initialization helps in giving a quicker convergence and a better overall error.

Biases can be generally initialized to zero. The rule for setting the weights is to be close to zero without being too small.

## Q18. What's the difference between a feed-forward and a backpropagation neural network?

A Feed-Forward Neural Network is a type of Neural Network architecture where the connections are “fed forward”, i.e. do not form cycles. The term “Feed-Forward” is also used when you input something at the input layer and it travels from input to hidden and from hidden to the output layer.

Backpropagation is a training algorithm consisting of 2 steps:

- Feed-Forward the values.
- Calculate the error and propagate it back to the earlier layers.

So to be precise, forward-propagation is part of the backpropagation algorithm but comes before back-propagating.

## **Q19. What are the Hyperparameters? Name a few used in any Neural Network.**

Hyperparameters are the variables which determine the network structure(Eg: Number of Hidden Units) and the variables which determine how the network is trained(Eg: Learning Rate). Hyperparameters are set before training.

- Number of Hidden Layers
- Network Weight Initialization
- Activation Function
- Learning Rate
- Momentum
- Number of Epochs
- Batch Size

## **Q20. Explain the different Hyperparameters related to Network and Training.**

### **Network Hyperparameters**



**The number of Hidden Layers:** Many hidden units within a layer with regularization techniques can increase accuracy. Smaller number of units may cause underfitting.

**Network Weight Initialization:** Ideally, it may be better to use different weight initialization schemes according to the activation function used on each layer. Mostly uniform distribution is used.

**Activation function:** Activation functions are used to introduce nonlinearity to models, which allows deep learning models to learn nonlinear prediction boundaries.

### **Training Hyperparameters**



**Learning Rate:** The learning rate defines how quickly a network updates its parameters. Low learning rate slows down the learning process but converges smoothly. Larger learning rate speeds up the learning but may not converge.

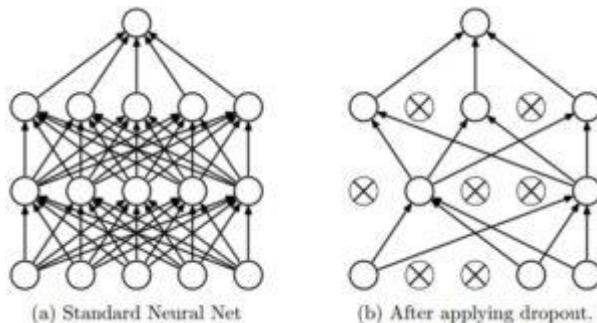
**Momentum:** Momentum helps to know the direction of the next step with the knowledge of the previous steps. It helps to prevent oscillations. A typical choice of momentum is between 0.5 to 0.9.

**The number of epochs:** Number of epochs is the number of times the whole training data is shown to the network while training. Increase the number of epochs until the validation accuracy starts decreasing even when training accuracy is increasing(overfitting).

**Batch size:** Mini batch size is the number of sub-samples given to the network after which parameter update happens. A good default for batch size might be 32. Also try 32, 64, 128, 256, and so on.

## Q21. What is Dropout?

Dropout is a regularization technique to avoid overfitting thus increasing the generalizing power. Generally, we should use a small dropout value of 20%-50% of neurons with 20% providing a good starting point. A probability too low has minimal effect and a value too high results in under-learning by the network.



Use a larger network. You are likely to get better performance when dropout is used on a larger network, giving the model more of an opportunity to learn independent representations.

**Q22. In training a neural network, you notice that the loss does not decrease in the few starting epochs. What could be the reason?**

The reasons for this could be:

- The learning rate is low
- Regularization parameter is high
- Stuck at local minima

**Q23. Name a few deep learning frameworks**

- TensorFlow
- Caffe
- The Microsoft Cognitive Toolkit/CNTK
- Torch/PyTorch
- MXNet
- Chainer
- Keras

**Q24. What are Tensors?**

Tensors are nothing but a de facto for representing the data in deep learning. They are just multidimensional arrays, that allows you to represent data having higher dimensions. In general, Deep Learning you deal with high dimensional data sets where dimensions refer to different features present in the data set.

't'
'e'
'n'
's'
'o'
'r'

*Tensor of dimension[1]*

3	1	4	1
5	9	2	6
5	3	5	8
9	7	9	3
2	3	8	4
6	2	6	4

*Tensor of dimensions[2]*

2	7	8	8	8
2	8	5	0	5
2	4	9	4	4
2	3	5	0	8
7	4	1	5	6
7	7	3	2	6

*Tensor of dimensions[3]*

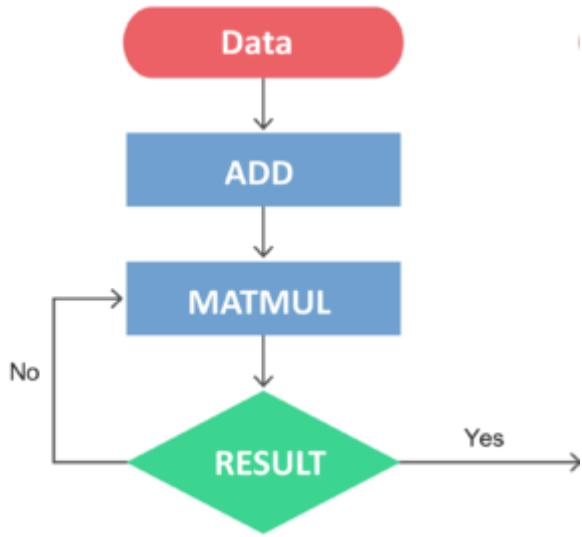
**Q25.** List a few advantages of TensorFlow?



- It has platform flexibility
- It is easily trainable on CPU as well as GPU for distributed computing.
- TensorFlow has auto differentiation capabilities
- It has advanced support for threads, asynchronous computation, and queues.
- It is a customizable and open source.

**Q26.** What is Computational Graph?

A computational graph is a series of TensorFlow operations arranged as nodes in the graph. Each node takes zero or more tensors as input and produces a tensor as output.



Basically, one can think of a Computational Graph as an alternative way of conceptualizing mathematical calculations that takes place in a TensorFlow program. The operations assigned to different nodes of a Computational Graph can be performed in parallel, thus, providing better performance in terms of computations.

## Q27. What is a CNN?

Convolutional neural network (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery. Unlike neural networks, where the input is a vector, here the input is a multi-channeled image. CNNs use a variation of multilayer perceptrons designed to require minimal preprocessing.

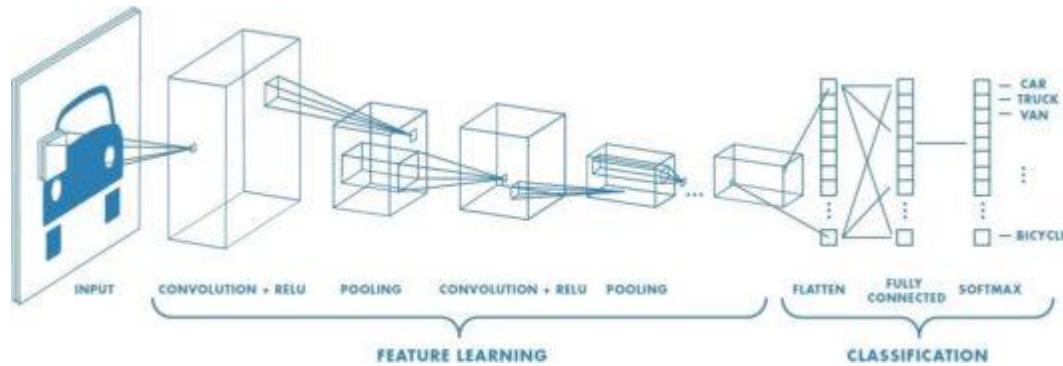
## Q28. Explain the different Layers of CNN.

There are four layered concepts we should understand in Convolutional Neural Networks:

**Convolution:** The convolution layer comprises of a set of independent filters. All these filters are initialized randomly and become our parameters which will be learned by the network subsequently.

**ReLU:** This layer is used with the convolutional layer.

Next



**Pooling:** Its function is to progressively reduce the spatial size of the representation to reduce the number of parameters and computation in the network. Pooling layer operates on each feature map independently.

**Full Connectedness:** Neurons in a fully connected layer have full connections to all activations in the previous layer, as seen in regular Neural Networks. Their activations can hence be computed with a matrix multiplication followed by a bias offset.

## Q29. What is an RNN?

Recurrent Networks are a type of artificial neural network designed to recognize patterns in sequences of data, such as text, genomes, handwriting, the spoken word, numerical times series data. Recurrent Neural Networks use **backpropagation** algorithm for training. Because of their **internal memory**, RNN's are able to remember important things about the input they received, which enables them to be very precise in predicting what's coming next.

## Q30. What are some issues faced while training an RNN?

Recurrent Neural Networks use backpropagation algorithm for training, but it is applied for every timestamp. It is commonly known as **Back-propagation Through Time** (BTT).

There are some issues with Back-propagation such as:

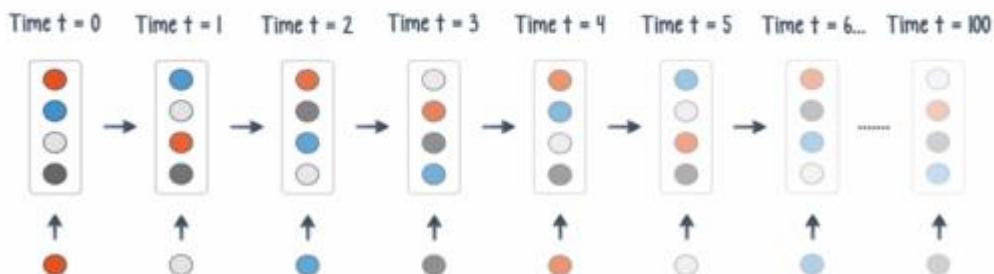
- Vanishing Gradient
- Exploding Gradient

### **Q31. What is Vanishing Gradient? And how is this harmful?**

When we do Back-propagation, the gradients tend to get smaller and smaller as we keep on moving backward in the Network. This means that the neurons in the Earlier layers learn very slowly as compared to the neurons in the later layers in the Hierarchy.

Earlier layers in the Network are important because they are responsible to learn and detecting the simple patterns and are actually the building blocks of our Network.

## **Decay of information through time**



Obviously, if they give improper and inaccurate results, then how can we expect the next layers and the complete Network to perform nicely and produce accurate results. The Training process takes too long and the Prediction Accuracy of the Model will decrease.

### **Q32. What is Exploding Gradient Descent?**

Exploding gradients are a problem when large error gradients accumulate and result in very large updates to neural network model weights during training.

Gradient Descent process works best when these updates are small and controlled. When the magnitudes of the gradients accumulate, an unstable network is likely to occur, which can cause poor prediction of results or even a model that reports nothing useful what so ever.

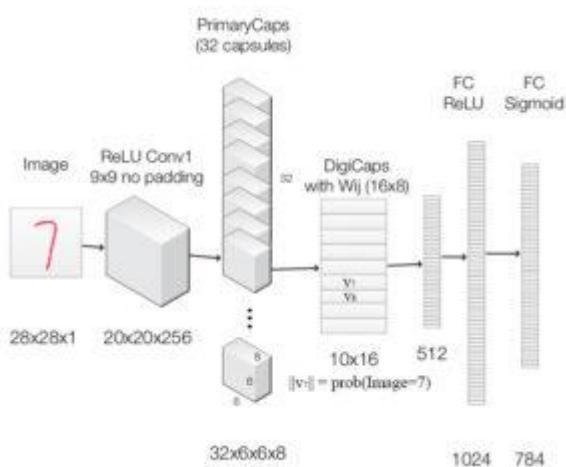
### Q33. Explain the importance of LSTM.

Long short-term memory(LSTM) is an artificial recurrent neural network architecture used in the field of deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections that make it a “general purpose computer”. It can not only process single data points, but also entire sequences of data.

They are a special kind of Recurrent Neural Networks which are capable of learning long-term dependencies.

### Q34. What are capsules in Capsule Neural Network?

**Capsules** are a vector specifying the features of the object and its likelihood. These features can be any of the instantiation parameters like position, size, orientation, deformation, velocity, hue, texture and much more.

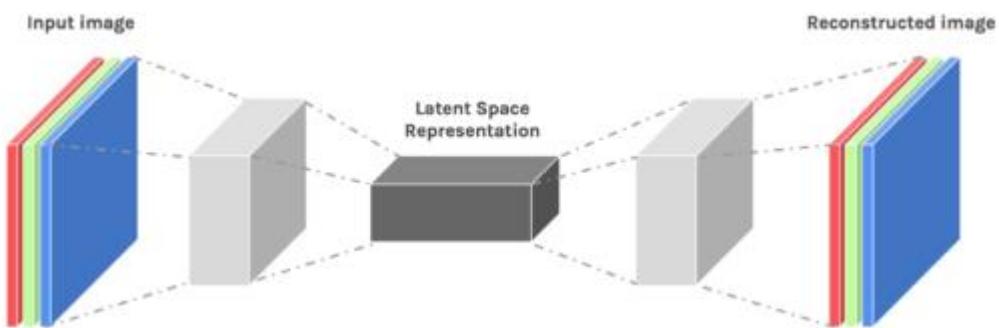


A capsule can also specify its attributes like angle and size so that it can represent the same generic information. Now, just like a neural network has layers of neurons, a capsule network can have layers of capsules.

Now, let's continue this Deep Learning Interview Questions and move to the section of autoencoders and RBMs.

### **Q35. Explain Autoencoders and it's uses.**

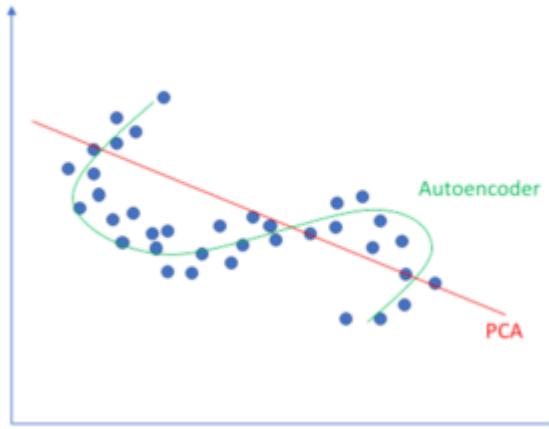
An **autoencoder** neural network is an Unsupervised Machine learning algorithm that applies backpropagation, setting the target values to be equal to the inputs. Autoencoders are used to reduce the size of our inputs into a smaller representation. If anyone needs the original data, they can reconstruct it from the compressed data.



### **Q36. In terms of Dimensionality Reduction, How does Autoencoder differ from PCAs?**

- An autoencoder can learn non-linear transformations with a non-linear activation function and multiple layers.
- It doesn't have to learn dense layers. It can use convolutional layers to learn which is better for video, image and series data.
- It is more efficient to learn several layers with an autoencoder rather than learn one huge transformation with PCA.
- An autoencoder provides a representation of each layer as the output.
- It can make use of pre-trained layers from another model to apply transfer learning to enhance the encoder/decoder.

### Linear vs nonlinear dimensionality reduction



### Q37. Give some real-life examples where autoencoders can be applied.

**Image Coloring:** Autoencoders are used for converting any black and white picture into a colored image. Depending on what is in the picture, it is possible to tell what the color should be.

**Feature variation:** It extracts only the required features of an image and generates the output by removing any noise or unnecessary interruption.

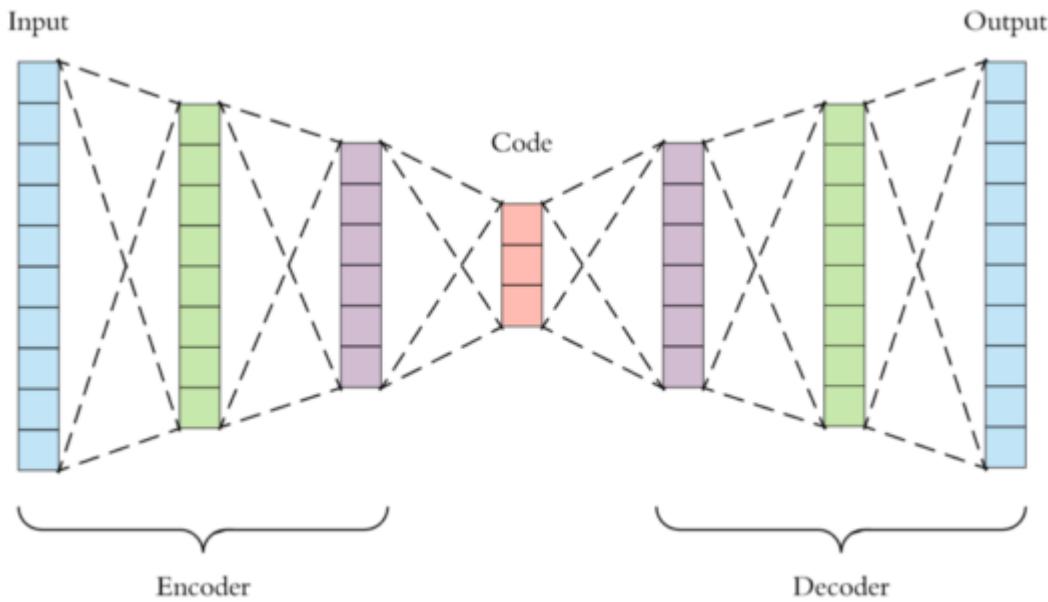
**Dimensionality Reduction:** The reconstructed image is the same as our input but with reduced dimensions. It helps in providing a similar image with a reduced pixel value.

**Denoising Image:** The input seen by the autoencoder is not the raw input but a stochastically corrupted version. A denoising autoencoder is thus trained to reconstruct the original input from the noisy version.

### Q38. what are the different layers of Autoencoders?

An Autoencoder consists of three layers:

- Encoder
- Code
- Decoder



### Q39. Explain the architecture of an Autoencoder.

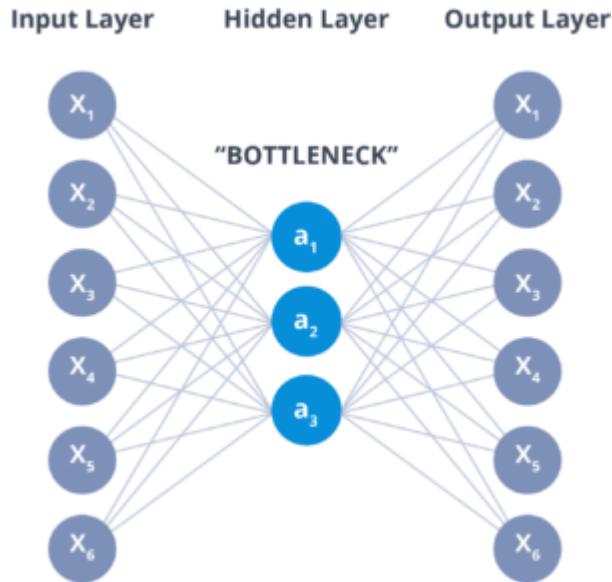
**Encoder:** This part of the network compresses the input into a latent space representation. The encoder layer encodes the input image as a compressed representation in a reduced dimension. The compressed image is the distorted version of the original image.

**Code:** This part of the network represents the compressed input which is fed to the decoder.

**Decoder:** This layer decodes the encoded image back to the original dimension. The decoded image is a lossy reconstruction of the original image and it is reconstructed from the latent space representation.

### Q40. What is a Bottleneck in autoencoder and why is it used?

The layer between the encoder and decoder, ie. the code is also known as Bottleneck. This is a well-designed approach to decide which aspects of observed data are relevant information and what aspects can be discarded.



It does this by balancing two criteria:

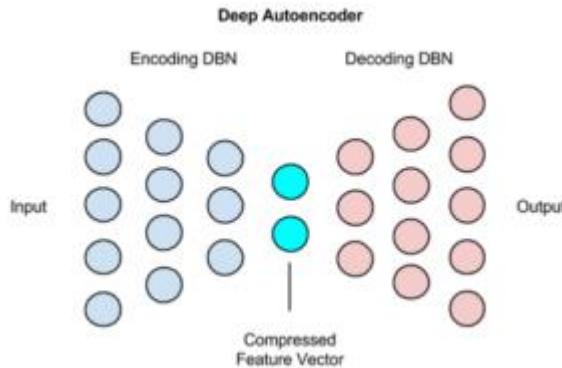
- Compactness of representation, measured as the compressibility.
- It retains some behaviourally relevant variables from the input.

#### **Q41. Is there any variation of Autoencoders?**

- Convolution Autoencoders
- Sparse Autoencoders
- Deep Autoencoders
- Contractive Autoencoders

#### **Q42. What are Deep Autoencoders?**

The extension of the simple Autoencoder is the Deep Autoencoder. The first layer of the Deep Autoencoder is used for first-order features in the raw input. The second layer is used for second-order features corresponding to patterns in the appearance of first-order features. Deeper layers of the Deep Autoencoder tend to learn even higher-order features.

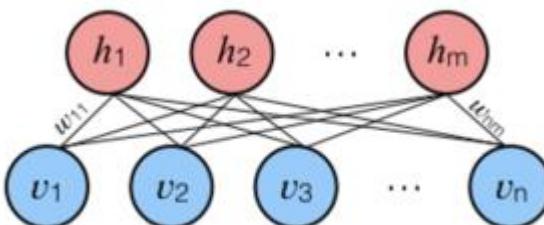


A deep autoencoder is composed of two, symmetrical deep-belief networks:

- First four or five shallow layers representing the encoding half of the net.
- The second set of four or five layers that make up the decoding half.

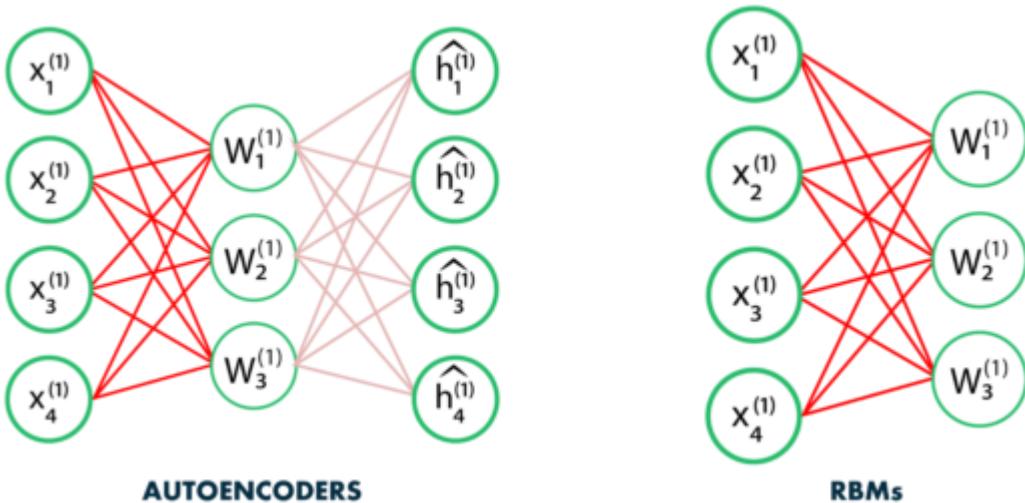
#### **Q43. What is a Restricted Boltzmann Machine?**

**Restricted Boltzmann Machine** is an undirected graphical model that plays a major role in Deep Learning Framework in recent times. It is an algorithm which is useful for dimensionality reduction, classification, regression, collaborative filtering, feature learning, and topic modeling.



#### **Q44. How Does RBM differ from Autoencoders?**

An Autoencoder is a simple 3-layer neural network where output units are directly connected back to input units. Typically, the number of hidden units is much less than the number of visible ones. The task of training is to minimize an error or reconstruction, i.e. find the most efficient compact representation for input data.



RBM shares a similar idea, but it uses stochastic units with particular distribution instead of deterministic distribution. The task of training is to find out how these two sets of variables are actually connected to each other.

One aspect that distinguishes RBM from other autoencoders is that **it has two biases**. The **hidden bias** helps the RBM produce the activations on the forward pass, while **The visible layer's biases** help the RBM learn the reconstructions on the backward pass.

Now, Coming to the final questions of this “Deep Learning Interview Questions” series.

#### **Q45. What are some limitations of deep learning?**

- Deep learning usually requires large amounts of training data.
- Deep neural networks are easily fooled.
- Successes of deep learning are purely empirical, deep learning algorithms have been criticized as uninterpretable “black-boxes”.
- Deep learning thus far has not been well integrated with prior knowledge.

#### **1. What is Deep Learning?**

Deep learning is a machine learning technology that involves neural networks. The term ‘deep’ in deep learning refers to the hierarchical structure of the networks used to teach computers natural human actions.

It is commonly used in medical research, driverless cars, and other cases where precision and accuracy are important.

#### **2. What is the difference between deep learning, machine learning and AI?**

Both **Deep Learning** and **Machine Learning** are part of **Artificial intelligence** and the difference between all the three domains is really just about the specificities. While deep learning deals with neural networks attempting to train machines through several layers of logic, machine learning is all about algorithms which uses historical data to teach machines. Artificial intelligence, of course, is the broader term which refers to any method which helps machines to mimic basic human actions.

### **3. What is the difference between supervised and unsupervised deep learning?**

Supervised learning refers to the learning method which trains machines through labelled data. This data is already categorised and tagged to the correct set of answers. When a machine is fed this data, it analyses the training set and produces the correct result.

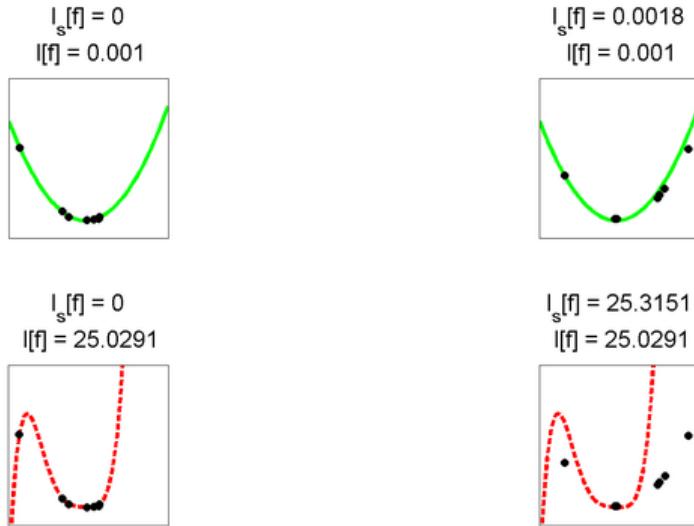
Unsupervised learning, on the other hand, does not require the data to be labelled. Machines self-learn from identifying patterns and model data according to probability densities.

### **4. What are data visualisation libraries?**

Data visualisation libraries help in understanding complex ideas by using visual elements such as graphs, charts, maps and more. The visualisation tools help you to recognise patterns, trends, outliers and more, making it possible to design your data according to the requirement. Popular data visualisation libraries include D3, React-Vis, Chart.js, vx, and more.

### **5. What is overfitting?**

Overfitting is a type of modelling error which results in the failure to predict future observations effectively or fit additional data in the existing model. It occurs when a function is too closely fit to a limited set of data points and usually ends with more parameters than the data can accommodate. It is common for huge data sets to have some anomalies, so when this data is used for any kind of modelling, it can result in inaccuracies in the analysis.



## 6. How to prevent overfitting?

Overfitting can be prevented by following a few methods namely-

- **Cross-validation:** Where the initial training data is split into several mini-test sets and each mini data set is used to tune the model.
- **Remove features:** Remove irrelevant features manually from the algorithms and use feature selection heuristics to identify the important features
- **Regularisation:** This involves various ways of making your model simpler so that there's little room for error due to obscurity. Adding penalty parameters and pruning your decision tree are ways of doing that.
- **Ensembling:** These are machine learning techniques for combining multiple separate predictions. The most popular methods of ensembling are bagging and boosting.

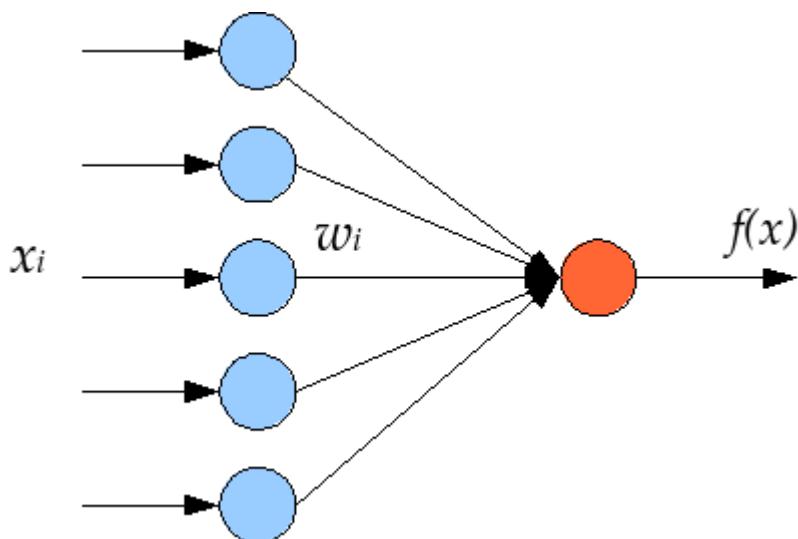
## 7. How are Deep networks better than shallow networks?

Neural networks include hidden layers apart from input and output layers. Shallow neural networks use a single hidden layer between the input and output layers whereas Deep neural networks, use multiple layers. For a shallow network to fit into any function, it needs to have a lot of parameters. However, since deep networks have several layers, it can fit functions better even with a limited number of parameters. Today deep networks have become preferable owing to its ability to work on any kind of data modelling, whether it is for voice or image recognition.

## 8. What is Perceptron? And how does it work?

Perceptron is a machine learning algorithm which came to exist from the 1950s. It is a single layer neural network with a linear classifier to work on a set of input data. Since perceptron uses classified data points which are already labelled, it is a supervised learning process.

Perceptron algorithms often present visual charts for users where output datasets are processed to provide the required output. The input data goes through an iterative loop to teach machines. This loop not only iterates but also evolves every time a **dataset is fed to the machine**. The algorithm improvises its output based on its findings each time so that after a period of time, the output data is more sophisticated and accurate.



## 9. What is Multilayer Perceptron and Boltzmann Machine?

Similar to single layer perceptron, multilayer perceptrons have input, output and hidden layers. However, since MLPs have more than one hidden layer, they are capable of classifying non-linear classes. Each node in the hidden layers uses a nonlinear activation function along with the input layers to produce the output through ‘backpropagation’. In this method, the neural networks calculate the errors using cost function and pushes the error backwards to the source to train the model more accurately.

The Boltzmann machine is a simplified version of the multilayer perceptron. This is a two layer model with a visible input layer and a hidden layer which makes stochastic decisions for the neurons. The nodes of this model are connected across layers without being connected to each other.

## 10. What are activation functions and its types?

Activation functions introduce non-linear properties to our network, allowing it to learn more complex functions. The main purpose of an activation function is to convert an input signal of a node in an A-NN to an output signal. This output signal is then used as an input in the next layer in the stack. To get the output of that layer and feed it as an input to the next layer, we must take the sum of the products of the inputs ( $x$ ) and their corresponding weights ( $w$ ) and apply the activation function  $f(x)$  to it, in an A-NN.

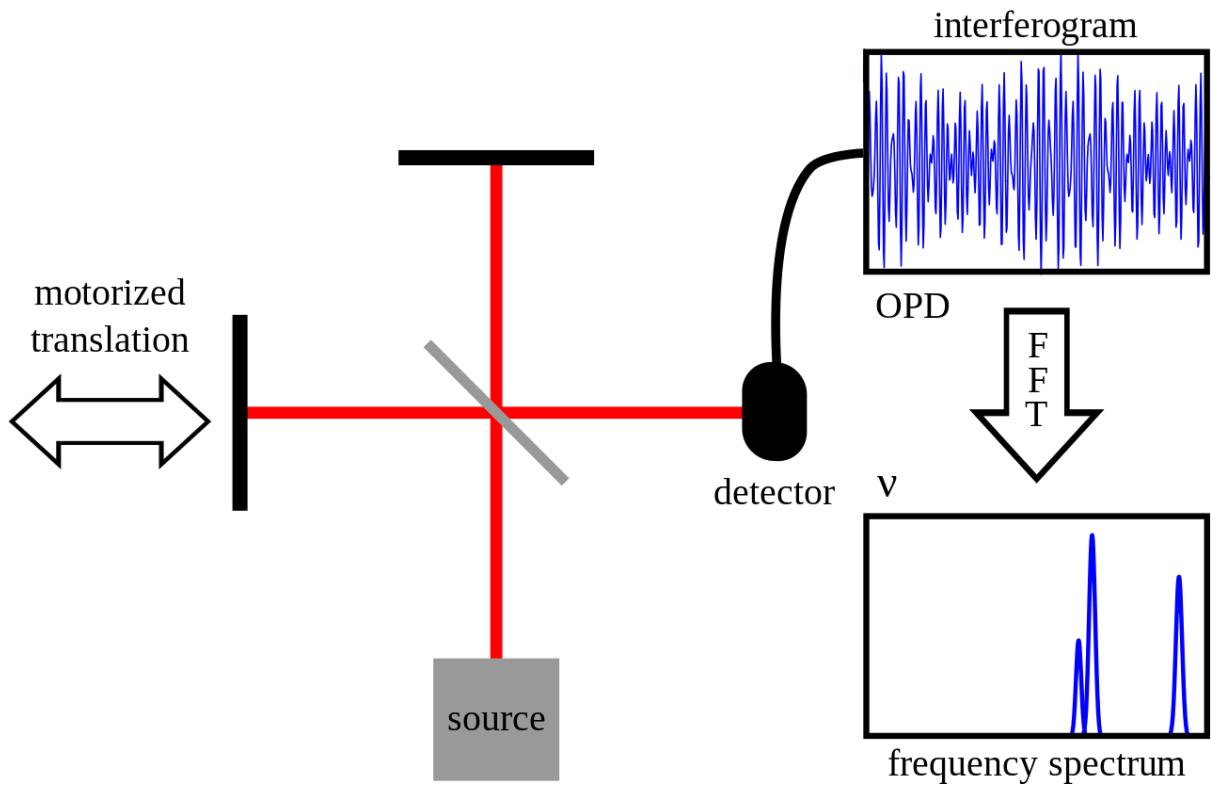
There are various types of activation functions, such as-  
Linear or Identity,  
Unit or Binary Step,  
Sigmoid or Logistic,  
Tanh,  
ReLU, and  
Softmax.

## **11. What is inductive reasoning machine learning?**

Inductive reasoning focuses as much on the conclusion as the premises and treats the conclusion as part of the reasoning to justify any behaviour. Also known as the ‘cause and effect reasoning’, inductive reasoning tries to prove a conclusion by backtracing to the inputs and picks up that logic as part of its learning.

## **12. What is the use of Fourier Transform in Deep Learning?**

Fourier transform is used in machine learning to process signals. The fourier series is a method of breaking down signals into frequency components. It is applicable to non-periodic signals such as a delta function and enables such signals to be measured in terms of frequencies instead of time. Fourier transform is useful when you are working on a system where the transfer function is known.



### 13. What is gradient descent? What are the steps for using a gradient descent algorithm?

An optimization algorithm which is used to learn the value of parameters that minimize cost function is known as a gradient descent. It is an iterative algorithm and moves in the direction of steepest descent. It is defined by the negative of the gradient. It was first proposed in 1847 by Augustin-Louis Cauchy. The steps involved in using a gradient descent algorithm are as follows-

- Initialize a random weight and bias
- Pass an input through the network and get the value from the output layer
- Calculate if there is an error between the actual value and the predicted value
- Go to each neuron which is contributing to the error and change its respective value so that the error is reduced
- Reiterate the steps until the best weights of the network are found

### 14. What are the benefits of mini-batch gradient descent?

Mini-batch gradient descent, a variation of the gradient descent algorithm, splits any training dataset into small batches to study data model errors and update accordingly. It is the most commonly used gradient descent used for deep learning.

The benefits of using a mini-batch gradient descent is that it allows a more

robust convergence without involving the local minima. Even computationally, it is more efficient than other gradient descents (stochastic and batch gradient descents). Mini-batching can work even with zero training data in memory and algorithm implementation.

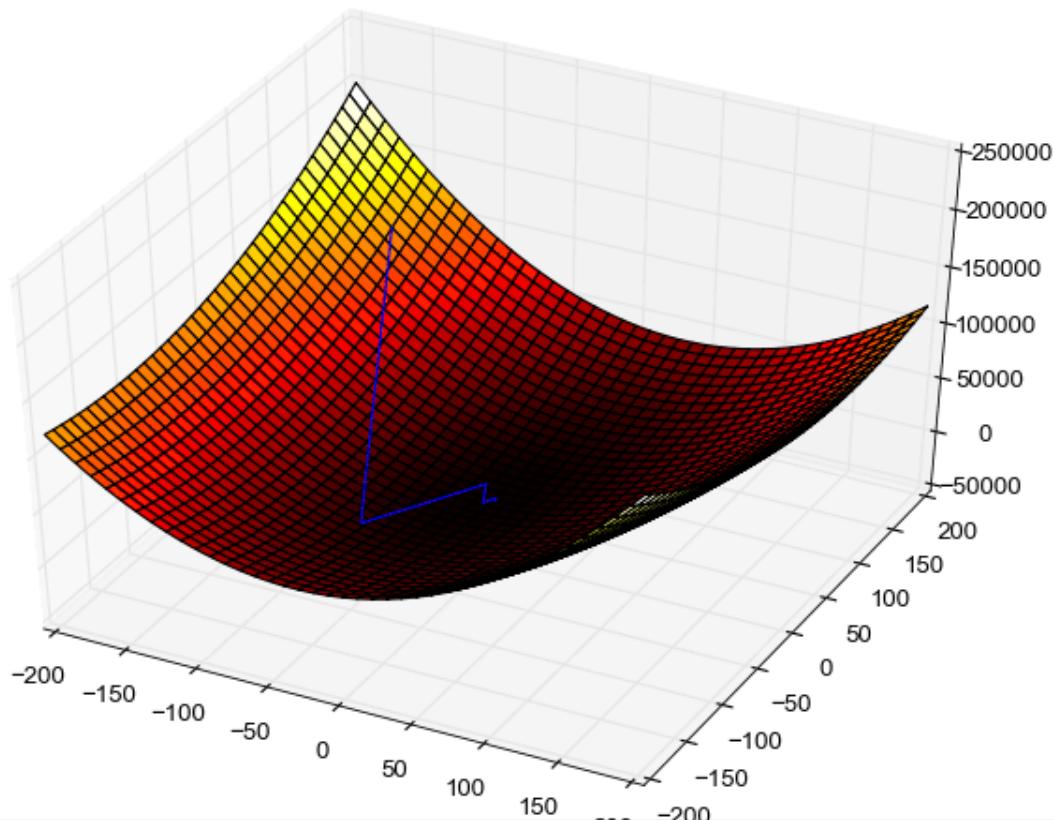
## **15. What are Vanishing and Exploding gradients?**

Vanishing gradient occurs when backpropagation does not function properly while training the neural networks. In such cases, the network parameters and hyperparameters do not match resulting in the slope becoming too small and decreasing.

In the case of an exploding gradient, there is a significant increase in the norm of the gradient during training. This results in unstable networks that are unable to learn from the training data, especially when the input has a long series of data.

## **16. What is the difference between Epoch, Batch, and Iteration?**

Epoch, iteration and batch are different types of processing datasets and algorithms for gradient descent. Epoch is the process of passing an entire dataset forward and backward through a neural network just once. Often the dataset is too big to be passed in a single attempt so it is passed several times to generate accurate results. When this happens, i.e., a limited set of data is passed through a neural network several times, it is known as an iterative process. However, if the data set is broken down into a number of batches or subsets to ensure it passes through the neural networks successfully, it is known as batch. All the three methods, i.e., epoch, iteration and batch size are basically ways of working on the gradient descent depending on the size of the data set.



## **17. How to choose the appropriate formula to solve issues on classification?**

Choosing the right metrics and formula to classify your data is extremely important for understanding and optimising the model. Use a model evaluation procedure to choose between different model types, features and tuning parameters. Train and test these models on the same set of data, split-test the models or cross-validate the models by comparing average results with test split results.

## **18. What is backpropagation?**

A training algorithm which is used for a multilayer neural network is known as backpropagation. In backpropagation, the error is moved from the end of the network to all weights, thus allowing efficient computing of the gradient. It is typically divided into several steps, such as-

- Forward propagation of the training data so that the output is generated.
- By using the target value and the output value, error derivative can be computed. (with respect to output activation)
- We then propagate for computing the derivative of the error (with respect to output activation) and continue to do so for all of the hidden layers.
- By using the previously calculated derivatives, we can calculate error

derivatives with respect to the weights.

-Update the weights.

## 19. What are hyperparameters?

Hyperparameters are created from prior observation before a dataset is captured and used in deep learning algorithms to train a model. Hyperparameters need to be initialised before training a model. The benefits of using hyperparameters are that they can control the behaviour of a training model and impact its performance significantly.

Hyperparameters can be optimised by Grid search, Random search, and Bayesian optimisation. Choosing good hyperparameters ensure easy management of large datasets.

## 20. What is underfitting and how can it be prevented?

When a model cannot train data or generalise new data, it is referred to as underfitting.

If a model has a good performance metric, it is easy to detect the error. In the case of underfitting, the model does not learn enough and is incapable of predicting correct results.

Underfitting can be prevented by using more training data, adding dropouts, reducing the capacity of networks and regularising weight.

## 21. What are Recurrent Neural Networks?

Recurrent Neural Networks are neural networks which uses the output from the previous step as inputs for the current step. Unlike a traditional neural network, where the inputs and outputs are independent of each other, in a recurrent neural network, the preceding outputs are crucial to decide the next. It features a hidden layer which carries data regarding a sequence.

## 22. What are the different layers of a convoluted neural network?

The different types of layers of a CNN include:

- **Convolutional Layer:** This is the core layer which has sets of learnable filters with receptive field. This is the first layer which extracts features from the input data.
- **ReLU Layer:** This layer converts negative pixels to zero by making the networks non-linear.

- **Pooling Layer:** This layer progressively reduces the spatial size of the representation by reducing computation and parameters in the network. The most common approach of pooling is max pooling.

## **23. What is the most preferred library in Deep Learning and why?**

Tensorflow is the most preferred library in deep learning.

Tensorflow provides high flexibility owing to its low-level structure. It can fit into any kind of functionality for any model. Tensorflow is popular among researchers as it can be changed according to the requirement and control networks better.

## **24. What do you understand by Tensors?**

Tensors are multidimensional arrays which allow us to represent data that have a higher dimension. Deep learning deals with high dimensional datasets. Here, dimensions refer to the various features that are present in the dataset.

## **25. What are deep autoencoders?**

Two symmetrical deep-belief networks which typically have four or five shallow layers that represent the encoding half of the net and a second set of four or five layers that represent the decoding half are together known as deep autoencoder. These layers are restricted Boltzmann machines and the building blocks of deep belief networks.

To process the dataset MNIST, a deep autoencoder uses binary transformations after each RBM. They can also be used for other datasets on which you would use Gaussian rectified transformations instead of the RBM.

## **26. What is the ultimate use of Deep learning in today's age and how is it aiding data scientists?**

Deep learning is used for a number of cases including language recognition, self-driving cars, text generation, video and image editing and more. However, the most important use of deep learning is perhaps in the **field of computer vision** where computers are fed relevant data to learn object detection, image restoration and segmentation, medical diagnostics, monitoring crops and livestock, and more. Scientists are using deep learning across industries to automate surveillance-based and repetitive tasks to improve productivity and accuracy.

## **27. What are the benefits of supervised learning procedure?**

With supervised learning, you can train the classifier perfectly so that it has a perfect decision boundary. Specific definitions of the classes helps machines distinguish between various classes accurately. Once the training is completed, the decision boundary can be reused for the mathematical formula, instead of the training data. Supervised learning is especially helpful for predictions on data with numerical values.

## **28. How does unsupervised learning aid in deep learning?**

Unsupervised learning has been heralded as the future of deep learning. It actually mimics how humans learn. The biggest advantage of using this method is that unlike supervised learning, unsupervised can scaled up. A strong unsupervised algorithm will be capable of learning by distinguishing without many examples.

- 1. What kind of a neural network will you use in deep learning regression via Keras-TensorFlow? Or How will you decide the best neural network model for a given problem?**

The foremost step when deciding on choosing a [neural network](#) model is to have a good know-how of the data and then decide the best model for it. Also, factoring in whether it is a linearly separable problem or not is important when deciding on a neural network model. So, the task at hand and the data play a vital role in choosing the best neural network model for a given problem. However, it is always better to start with a simple model like multi-layer perceptron (MLP) that has just one hidden layer unlike CNN, LSTM, or RNN that require configuring the nodes and layers. MLP is considered the simplest neural network because the weight initialization is not sensitive and also there is no need to define a structure for the network beforehand.

- 2. Why do we need autoencoders when there are already powerful dimensionality reduction techniques like Principal Component Analysis?**

The curse of dimensionality (the problems that arise when working with high-dimensional data) is a common problem when [working on machine learning](#) or deep learning projects. Curse of Dimensionality causes lots of difficulties while training a model because it requires training a lot of parameters on a scarce dataset leading to issues like overfitting, large training times, and poor generalization. PCA and autoencoders are used to tackle these issues. PCA is an unsupervised technique wherein the actual data is projected to the direction of high variance while autoencoders are [neural networks](#) used for compressing the

data into a low dimensional latent space and then try to reconstruct the actual high dimensional data.

PCA or autoencoders are effective only when the features have some relationship with each other. A general thumb rule between choosing PCA and Autoencoders is the size of data. Autoencoders work great for larger datasets and PCA works well for smaller datasets. Autoencoders are usually preferred when there is a need for modeling non-linearities and relatively complex relationships. Autoencoders can encode a lot of information with fewer dimensions when there is a curvature in low dim structure or non-linearity, making them a better choice over PCA in such scenarios.

Autoencoders are usually preferred for identifying data anomalies rather than for reducing data. Anomalous data points can be identified using the reconstruction error, PCA is not good for reconstructing data particularly when there are non-linear relationships.

**3. Say you have to build a neural network architecture; how will you decide how many neurons and hidden layers are needed for the network?**

Given a business problem, there is no hard and fast rule to determine the exact number of neurons and hidden layers required to build a neural network architecture. The optimal size of the hidden layer in a neural network lies between the size of the output layers and the size of the input. However, here are some common approaches that have the advantage of making a great start to building a neural network architecture –

- To address any specific real-world predictive modeling problem, the best way is to start with rough systematic experimentation and find out what would work best for any given dataset based on prior experience [working with neural networks](#) on similar real-world problems. Based on the understanding of any given problem domain and one's experience working with neural networks, one can choose the network configuration. The number of layers and neurons used on similar problems is always a great way to start testing the configuration of a neural network.
- It is always advisable, to begin with, simple neural network architecture and then go on to enhance the complexity of the neural network.
- Try working with varying depths of networks and configure deep neural networks only for challenging predictive modeling problems where depth can be beneficial.

**4. Why CNN is preferred over ANN for Image Classification tasks even though it is possible to solve image classification using ANN?**

One common problem with using ANN's for image classification is that ANN's react differently to input images and their shifted versions. Let's consider a simple example where you have the picture of a dog in the top left of an image and in another image, there is a picture of a dog at the bottom right. ANN will assume that a dog will always appear in this section of any image, however, that's not the case. ANN's require concrete data points meaning if you are building a deep learning model to distinguish between cats and dogs, the length of the ears, the width of the nose, and other features should be provided as data points while if using CNN for image classification spatial features are extracted from the input images. When there are thousands of features to be extracted, CNN is a better choice because it gathers features on its own, unlike ANN where each individual feature needs to be measured.

Training a neural network model becomes computationally heavy (requiring additional storage and processing capability) as the number of layers and parameters increases. Tuning the increased number of parameters can be a tedious task with ANN, unlike CNN where the time for tuning parameters is reduced making it an ideal choice for image classification problems.

## **5. Why Sigmoid or Tanh is not preferred to be used as the activation function in the hidden layer of the neural network?**

A common problem with Tanh or Sigmoid functions is that they saturate. Once saturated, the learning algorithms cannot adapt to the weights and enhance the performance of the model. Thus, Sigmoid or Tanh activation functions prevent the neural network from learning effectively leading to a vanishing gradient problem. The vanishing gradient problem can be addressed with the use of Rectified Linear Activation Function (ReLU) instead of sigmoid and using a Xavier initialization.

## **6. Why does the exploding gradient problem happen?**

When the model weights grow exponentially and become unexpectedly large in the end when training the model, exploding gradient problem happens. In a neural network with n hidden layers, n derivatives are multiplied together. If the weights that are multiplied are greater than 1 then the gradient increases exponentially greater than the usual one and eventually explodes as you propagate through the model. The situation wherein the value of weights is more than 1 makes the output exponentially larger hindering the model training and impacting the overall accuracy of the model is referred to as the exploding gradients problem. Exploding gradients is a serious problem because the model cannot learn from its training data resulting in a poor loss. One can deal with the exploding gradient problem either by gradient clipping, weight regularization, or with the use of LSTM's.

## **7. How to fix the constant validation accuracy in CNN model training?**

Constant validation accuracy is a common problem when training any neural network because the network just remembers the sample and results in an overfitting problem. Overfitting of a model means that the neural network model works fantastic on the training sample but the performance of the model sinks in on the validation set. Here are some tips to try to fix the constant validation accuracy in CNN –

- It is always advisable to divide the dataset into training, validation, and test set.
- When working with little data, this problem can be solved by changing the parameters of the neural network by trial and error.
- Increasing the size of the training dataset.
- Use batch normalization.
- Regularization
- Reduce the network complexity

## **8. What do you understand by learning rate in a neural network model? What happens if the learning rate is too high or too low?**

Learning rate is one of the most important configurable hyperparameters used in the training of a neural network. The value of the learning rate lies between 0 and 1. Choosing the learning rate is one of the most challenging aspects of training a neural network because it is the parameter that controls how quickly or slowly a neural network model adapts to a given problem and learns. A higher learning rate value means that the model requires few training epochs and results in rapid changes while a smaller learning rate implies that the model will take a long time to converge or might never converge and get stuck on a suboptimal solution. Thus, it is advisable not to use a learning rate that is too low or too high but instead a good learning rate value should be discovered through trial and error.

## **9. What kind of a network would you prefer – a shallow network or a deep network for voice recognition?**

Every neural network has a hidden layer along with input and output layers. Neural networks that use a single hidden layer are known as shallow neural networks while those that use multiple hidden layers are referred to as deep neural networks. Both shallow and deep networks are capable of fitting into any function but shallow networks require a lot of parameters, unlike deep networks that can fit functions even with a limited number of parameters because of several layers. Deep networks are preferred today over shallow networks because at every layer the model learns a novel and abstract representation of

the input. Also, they are much more efficient in terms of the number of parameters and computations compared to shallow networks.

#### **10. Can you train a neural network model by initializing all biases as 0?**

Yes, there is a possibility that the neural network model will learn even if all the biases are initialized to 0.

#### **11. Can you train a neural network model by initializing all the weights to 0?**

No, it is not possible to train a model by initializing all the weights to 0 because the neural network will never learn to perform a given task. Initializing all weights to zeros will cause the derivatives to remain the same for every  $w$  in  $W$  [1] because of which neurons will learn the same features in every iteration. Not just 0, but any kind of constant initialization of weights is likely to produce a poor result.

#### **12. Why is it important to introduce non-linearities in a neural network?**

Without non-linearities, a neural network will act like a perceptron regardless of how many layers are there making the output linearly dependent on the input. In other words, having a neural network with  $n$  layers and  $m$  hidden units with linear activation functions is just like having a linear neural network without hidden layers that can only find linear separation boundaries. A neural network without non-linearities cannot find appropriate solutions and classify the data correctly for complex problems.

#### **13. Why dropout is effective in deep networks?**

The problem with deep neural networks is that they are most likely to overfit training data with few examples. Overfitting can be reduced by ensembles of networks with different model configurations but this requires the additional effort of maintaining multiple models and is also computationally expensive. Dropout is one of the easiest and exceptionally successful methods to reduce dependencies in deep neural networks and overcome overfitting problems. When using the dropout regularization method, a single neural network model is used to similar different network architecture by dropping out nodes while training. It is considered an effective method of regularization as it improves generalization errors and is also computationally cheap.

#### **14. A deep learning model finds close to 12 million face vectors. How will you find a new face quickly?**

You will need to know about One-Shot Learning for Face Recognition which is a classification task where one or more examples(faces in this case) are used for classifying new examples(faces) in the future. One needs to know about the method of indexing data to retrieve a new face faster. A new face can be recognized by finding the vectors that are close (most similar) to the input face but in this case, the system would have become super slow if we were to calculate the distance to 12 million vectors. A convenient way would be to index data on real vector space by dividing the data into easy structures for querying (almost like a tree data structure). It is easier to find the vector that is in close proximity with time very quickly whenever new data is available. Techniques like Annoy Indexing, Locality Sensitive Hashing, and Approximate Nearest Neighbours can be used for this purpose.

#### **15. What has fostered the implementation and experimentation of powerful neural network architectures in the industry?**

Flexibility makes deep learning powerful. Neural networks are universal function approximators so even if it is a complex enough problem at hand(where the formula between input and output is not known), a neural network can be approximated. Also, transfer learning (where the trained weights of an existing neural network can be used to initialize the weights of another network that performs similar tasks) makes the application of deep learning much easier under situations when training a neural network from scratch is costly or almost impossible when there is data scarcity.

Faster and powerful computational resources are also a prime reason for the adoption of neural network architectures. One cannot deny the fact that it is faster to train a neural network in just minutes with GPU acceleration which would otherwise take days for the network to learn.

#### **16. Can you build deep learning models based solely on linear regression?**

Yes, it is definitely possible to build deep networks using a linear function as the activation function for each layer if the problem is represented by a linear equation. However, a problem that is a composition of linear functions is a linear function and there is nothing extraordinary that can be achieved with the implementation of a deep network because adding more nodes to the network will not increase the predictive power of the [machine learning model](#).

#### **17. When training a deep learning model you observe that after a few epochs the accuracy of the model decreases. How will you address this problem?**

The decrease in the accuracy of a deep learning model after a few epochs implies that the model is learning from the characteristics of the dataset and not considering the features. This is referred to as the overfitting of the deep learning model. You can either use dropout regularization or early stopping to fix this issue. Early stopping as the phrase implies stops training the deep learning model any further the moment you notice a drop inaccuracy of the model. Dropout regularization is a technique wherein a few nodes or output layers are dropped so that the remaining nodes have varying weights.

**18.What is the impact on a model with an improperly set learning rate on weights?**

With images as inputs, an improperly set learning rate can cause noisy features. Having an ill-chosen learning rate determines the prediction quality of a model and can result in an un converged neural network.

**19)What do you understand by the terms Batch, Iterations, and Epoch in training a neural network model?**

- Epoch refers to the iteration where the complete dataset is passed forward and backward through the neural network only once.
- It is not possible to pass the complete dataset to the network in one go so the dataset is divided into parts. This is referred to as the Batch.
- The total number of batches needed to complete one epoch is referred to as iteration. For example, if you have 60,000 data rows and the batch size is 1000 then each epoch will run 60 iterations.

**20) Is it possible to calculate the learning rate for a model a priori?**

For simple models, it could be possible to set the best learning rate value a priori. However, for complex models, it is not possible to calculate the best learning rate through theoretical deductions that can actually make accurate predictions. Observations and experiences do play a vital role in defining the optimal learning rate.

**21) What is the theoretical foundation of neural networks?**

To answer this question one needs to explain the universal approximation theorem that forms the base on why neural networks work.

*Introducing non-linearity via an activation function allows us to approximate any function. It's quite simple, really*

According to the Universal Approximation Theorem, a neural network having a single hidden layer containing a finite number of neurons can approximate any continuous function to a reasonable accuracy for inputs in a specific range. However, if the function has large gaps it is not possible to approximate it. Meaning, if a neural network is trained with inputs between 20 and 30, we cannot be assured that it will work well for inputs between 60 and 70.

## **22) What are the commonly used approaches to set the learning rate?**

- Using a fixed learning rate value for the complete learning process.
- Using a learning rate schedule
- Making use of adaptive learning rates
- Adding momentum to the classical SGD equation.

## **23) Is there any difference between neural networks and deep learning?**

Ideally, there is no significant difference between deep learning networks and neural networks. Deep learning networks are neural networks but with a slightly complex architecture than they were in 1990s. It is the availability of hardware and computational resources that has made it feasible to implement them now.

## **24) You want to train a deep learning model on a 10GB dataset but your machine has 4GB RAM. How will you go about implementing a solution to this deep learning problem?**

One of the possible ways to answer this question would be to say that a neural network can be trained by loading the data into the NumPy array and defining a small batch size. NumPy doesn't load the complete dataset into the memory but creates a complete mapping of the dataset. NumPy offers several tools for compressing large datasets that can be integrated with other NN packages like PyTorch, [TensorFlow](#), or Keras.

## **25) How will the predictability of a neural network impact if you use a ReLu activation function and then use the Sigmoid function in the final layer of the network?**

The neural network will predict only one class for all types of inputs because the output of a ReLu activation function is always a non-negative result.

## **26) What are the limitations of using a perceptron?**

A major drawback to using a perceptron is that they can only linearly separable functions and cannot handle non-linear inputs.

## **27) How will you differentiate between a multi-class and multi-label classification problem?**

In a multi-class classification problem, the classification task has more than two mutually exclusive classes whereas in a multi-label problem each label has a different classification task, however, the tasks are related somehow. For example, classifying a set of images of animals which may be cats, dogs, or bears is a multi-class classification problem that assumes that each sample has only one label meaning an image can be classified as either a cat or a dog but not both at the same time. Now imagine that you want to process the below image. The image shown below needs to be classified as both cat and dog because the image shows both the animals. In a multi-label classification problem, a set of labels are assigned to each sample and the classes are not mutually exclusive. So, a pattern can belong to one or more classes in a multi-label classification problem.

## **28) What do you understand by transfer learning?**

You know how to ride a bicycle, so it will be easy for you to learn to drive a bike. This is transfer learning. You have some skill and you can learn a new skill that relates to it without having to learn it from scratch. Transfer learning is a process in which the learning can be transferred from one model to another without having to make the model learn everything from scratch. The features and weights can be used for training the new model providing reusability. Transfer learning works well in training a model easily when there is limited data.

## **29) What is fine-tuning and how is it different from transfer learning?**

In transfer learning, the feature extraction part remains untouched and only the prediction layer is retrained by changing the weights based on the application. On the contrary in fine-tuning, the prediction layer along with the feature extraction stage can be retrained making the process flexible.

## **30) Why do we use convolutions for images instead of using fully connected layers?**

Each convolution kernel in a CNN acts like its own feature detector and has a partially in-built translation in-variance. Using convolutions lets one preserve, encode and make use of the spatial information from the image, unlike fully connected layers that do not have any relative spatial information.

### **31) What do you understand by Gradient Clipping?**

Gradient Clipping is used to deal with the exploding gradient problem that occurs during the backpropagation. The gradient values are forced element-wise to a particular minimum or maximum value if the gradient has crossed the expected range. Gradient clipping provides numerical stability while training a neural network but does not provide any performance improvements.

### **32) What do you understand by end-to-end learning?**

It is a deep learning process where a model gets raw data as the input and all the various parts are trained simultaneously to produce the desired outcome with no intermediate tasks. The advantage of end-to-end learning is that there is no need for implicitly doing [feature engineering](#) which usually leads to a lower bias. A good example that you can quote in the context of end-to-end learning is driverless cars. They use human-provided input as guidance and are trained to automatically learn and process the information using a CNN to complete tasks.

### **33) Are convolutional neural networks translation-invariant?**

Convolutional neural networks are translation invariant only to a certain extent but pooling can make them translation invariant. Making a CNN completely translation-invariant might not be possible. However, by feeding the right kind of data this can be achieved although this might not be a feasible solution.

### **34) What is the advantage of using small kernels like 3x3 than using a few large ones.**

Smaller kernels let you use more filters so you can use a greater number of activations functions and let the CNN learn a more discriminative mapping function. Also, smaller kernels capture more spatial context and use fewer computations and parameters making them a better choice over large ones.

### **35) How can you generate a dataset on multiple cores in real-time that can be fed to the deep learning model?**

One of the major challenges today in CV is the need to load large datasets of videos and images but there is not enough memory on the machine. In such situations, data generators act as a magic wand when it comes to loading a dataset that is memory-consuming. You can talk about the various data generators Keras model class provides. When working with big data, in most of the cases it might not be required to load all the data into RAM as it would be memory wastage, could lead to memory overflow, and also take a longer time to

process. Making use of generative functions is highly beneficial then as they generate the data to be directly fed into the model in each batch for training.

### **36) How do you bring balance to the force when handling imbalanced datasets in deep learning?**

It is next to impossible to have a perfectly balanced real-world dataset when working on deep learning problems so there will be some level of class imbalance within the data that can be tackled either by –

- Weight Balancing -
- Over and Under Sampling

### **37) What are the benefits of using batch normalization when training a neural network?**

- Batch normalization optimizes the network training process making it easier to build and faster to train a deep neural network.
- Batch normalization regulates the values going into each activation function making activation functions more viable because non-linearities that don't seem to work well become viable with the use of batch normalization.
- Batch normalization makes it easier to initialize weights and also allows the use of higher learning rates ultimately increasing the speed at which the network trains.

### **38) Which is better LSTM or GRU?**

LSTM works well for problems where accuracy is critical and sequence is large whereas if you want less memory consumption and faster operations, opt for GRU.

### **39) RMSProp and Adam optimizer adjust gradients? Does this mean that they perform gradient clipping?**

This does not inherently mean that they perform gradient clipping because gradient clipping involves setting up predetermined values beyond which the gradients cannot go, unlike Adam and RMSProp that make multiplicative adjustments to gradients.

### **40) Can you name a few hyperparameters used for training a neural network.**

When training any neural networks there are two types of hyperparameters-one that define the structure of the neural network and the other determining how a neural network is trained. Listed are a few hyperparameters that are set before training any neural network –

- Initialization of weights
- Setting the number of hidden layers
- Learning Rate
- Number of epochs
- Activation Functions
- Batch Size
- Momentum

#### **41) When is multi-task learning usually preferred?**

Multi-task learning with deep neural networks is a subfield wherein several tasks are learned by a shared model. This reduces overfitting, enhances data efficiency, and speeds up the learning process with the use of auxiliary information. Multi-task learning is useful when there is a small amount of data for any given task and we can benefit from training a deep learning model on a large dataset.

#### **42) Explain the Adam Optimizer in one minute.**

Adaptive momentum or Adam optimizer is an optimization algorithm designed to deal with sparse gradients on noisy problems. Adam optimizer improves convergence through momentum that ensures that a model does not get stuck in saddle point and also provides per-parameter updates for faster convergence.

#### **43) Which loss function is preferred for multi-category classification?**

Cross-Entropy loss function

#### **44) To what kind of problems can the cross-entropy loss function be applied?**

- Binary Classification Problems
- Multi-Label Classification Problems
- Multi-Category Classification Problems

#### **45) List the steps to implement a gradient descent algorithm.**

- The first step is to initialize random weight and bias.

- Get values from the output layer by passing the input through the neural network.
- Determine the error between the actual and predicted value.
- Based on the neurons that contribute to the error, modify the values to minimize the error.
- Repeat the process until the optimal weights are found for the neural network.

**46) How important is it to shuffle the training data when using batch gradient descent?**

Shuffling the training dataset will not make much of a difference because the gradient is calculated at every epoch using the complete training dataset.

**47) What is the benefit of using max-pooling in classification convolutional neural networks?**

The feature maps become smaller after max-pooling in CNN and hence help reduce the computation and also give more translation in-variance. Also, we don't lose much semantic information because we're taking the maximum activation.

**48) Can you name a few data structures that are commonly used in deep learning?**

You can talk about computational graphs, tensors, matrices, data frames, and lists.

**49) Can you add an L2 regularization to a recurrent neural network to overcome the vanishing gradient problem?**

This can actually worsen the vanishing gradient problem because the L2 regularization will shrink weights towards zero.

**50) How will you implement Batch Normalization in RNN?**

It is not possible to use batch normalization in RNN because statistics are computed per batch and thus batch normalization will not consider the recurrent part of the neural network. An alternative to this could be layer normalization in RNN or reparameterizing the LSTM layer that allows the use of batch normalization.

**1. Given that there are so many deep learning algorithms, how will you determine which deep learning algorithm has to be used for a dataset.**

Artificial Neural Network Artificial [Neural Network](#) or sometimes called Classic Neural Network is a connection of multilayered perceptrons. This algorithm can be used when the data is properly structured in a tabular form. Both Classification and regression problems can be solved using ANNs Convolutional Neural Networks These networks are the best proven ones to build any prediction model involving image data as input. To put it in general terms, CNN works best on data with spatial relationships and hence these can also produce state-of-the-art results for NLP problems such as topic modelling, document classification and so on. Recurrent Neural Networks RNNs come into picture when we have sequential data where the order of the data entered is also important. RNNs can provide solutions for problems involving [Time Series](#) data. More often, rather than vanilla RNNs, gated networks like LSTMs (Long short term memory) and GRUs(Gated Recurrent units) are proven to give much better results. Autoencoders Autoencoders are widely used in the deep learning community these days because of its ability to operate automatically based on its inputs even before taking an activation function and final output decoding. These can be used when we have problems such as feature detection, recommendation systems and other compelling problems.

## 2. How do one-hot encoding and label encoding affect the dimensionality of a dataset?

Label encoding does not really affect the dataset in any way because in label encoding, we only provide labels to each category in the column.

For example,

Place of birth (before label encoding)	Place of birth (after label encoding)
Delhi	0
Hyderabad	1
Chennai	2
Delhi	0

In the above example, we are mapping Delhi -> 0, Hyderabad -> 1, and Chennai -> 2.

In one hot encoding, we create columns to each of the category in the dataset. Thus, the more the number of categories in the column, the more are the columns generated after one hot encoding. Let us consider the very same dataset that we saw above. After one hot encoding it will look like the table shown below

Place of birth (Delhi)	Place of birth (Hyderabad)	Place of birth (Chennai)
1	0	0
0	1	0
0	0	1
1	0	0

If the value is ‘Delhi’, then only the column meant for ‘Delhi’ takes the value 1 and the other columns takes the value 0.

Often, we don't consider the last/first category after one hot encoding the variable because it can be clearly understood that if all the existing entries for the category are 0, then it belongs to the category that we dropped. This is much clearly explained with the example below

Place of birth (Delhi)	Place of birth (Hyderabad)
1	0
0	1
0	0
1	0

Here , we already know that there are 3 unique categories in the variable (Delhi, Hyderabad, and Chennai). There are two zeros in the 3rd row which clearly implies that it does not belong to both the categories and the one which remains in Chennai. Therefore, the decoded value for that row is Chennai.

### 3. Why are GPUs important for implementing deep learning models?

Whenever we are trying to build any neural network model, the model training phase is the most resource-consuming job. Each iteration of model training comprises thousands (or even more) of matrix multiplication operations taking place. If there are less than around 1 lakh parameters in a neural network model, then it would not take more than a few minutes (or few hours at most) to train. But when we have millions of parameters, that is when our sizable computers would probably give up. This is where GPUs come into the picture. GPUs (Graphics Processing Units) are nothing but CPUs but with more ALUs (Arithmetic logic units) than our normal CPUs which are specifically meant for this kind of heavy mathematical computation.

#### **4. Which is the best algorithm for face detection ?**

There are several machine learning algorithms available for face detection but the best ones are the ones which involve CNNs and deep learning. Some notable algorithms for face detection are listed below FaceNet Probabilistic Face Embedding ArcFace Cosface Spherface

#### **5. What evaluation approaches do you use to gauge the effectiveness of deep learning models?**

#### **6. When training a neural network, you observe that the loss does not decrease in the first few epochs. What are the possible reasons for this?**

7. What are the commonly used techniques to deal with the overfitting of a deep learning model?

8. What kind of gradient descent variant is the best for handling data that is too big to handle in RAM simultaneously?

9. How will you explain the success and recent rise in demand for deep learning in the industry?

10. How do you select the depth of a neural network?

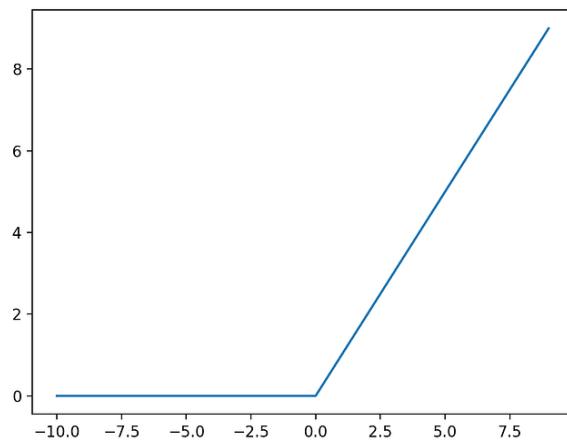
**1)What is Deep Learning?**

**2)Which deep learning framework do you prefer to work with – PyTorch or TensorFlow and why? Refer [PyTorch vs Tensorflow](#) for answer**

**3) Talk about a deep learning project you've worked on and the tools you used?**

**4) Have you used the ReLu activation function in your neural network? Can you explain how does the ReLu activation function works?**

Yes, I have used ReLu in my neural networks. ReLu stands for Rectified Linear Unit. Basically, the function returns the input value as it is if it is positive or returns zero if it is negative. If the function is plotted in a line graph, it would look like the graph shown below



The main purpose of formulating this function was to overcome the Vanishing gradient problem caused by preliminary activation functions like Sigmoid and TanH which prevented us from building deeper neural network models. Now a days, this function has become a default activation function for many types of neural network models because models that use this function are easily trainable and don't suffer from the vanishing gradient problem.

**5) How often do you use pre-trained models for your neural network?**

**6) What does the future of video analysis look like with the use of deep learning solutions? How effective/good is video analysis currently?**

**7) Tell us about your passion for deep learning. Do you like to participate in deep learning/machine learning hackathons, write blogs around novel deep learning tools, or attend local meetups, etc ?**

**8) Describe the last time you felt frustrated solving a deep learning challenge, and how did you overcome it?**

9. What is more important to you the performance of your deep learning model or its accuracy?

10. Given the dataset, how will you decide which deep learning model to use and how to implement it?

11. What is the last deep learning research paper you've read?

12. What are the most commonly used neural network paradigms ? (Hint: Talk about Encoder-Decoder Structures, LSTM, GAN, and CNN)

13. Is it possible to use a neural network as a tool of dimensionality reduction?

14. How deep learning models tackle the curse of dimensionality?

## **15) What are the pros and cons of using neural networks?**

### **Pros :**

Neural networks are highly flexible and can be used for both classification and regression problems and sometimes for problems much more complex than that. Neural networks are highly scalable. We can add as many layers with as many neurons as we want. Neural networks are proven to produce best results when we have a lot of data points. They work best for non linear data such as image data, text data and so on. They can be used on any data that can be converted to numbers.

### **Cons :**

1. The well known disadvantage of neural networks is their "black box" nature. That is, we don't know how or why our neural network came up with a certain output. For example, when we feed an image of a dog into a neural network and it predicts it to be a duck, we may find it difficult to understand what caused it to arrive at this prediction.

2. Developing a neural network model takes much time.

3. Neural networks are more computationally expensive than traditional algorithms.

4. The amount of computational power needed for a neural network depends mostly on the size of data, depth and complexity of the network.

5. To train a neural network model, it requires much more data than training a traditional machine learning model.

**16) How is a Capsule Neural Network different from a Convolutional Neural Network?**

**17) What is a GAN and what are the different types of GAN you've worked with?**

**18) For any given problem, how do you decide if you have to use transfer learning or fine-tuning?**

Transfer learning is a method used when a model is developed for one task is reused to work on a second task. Fine tuning is one approach to achieve transfer learning. In Transfer Learning we train the model with a dataset and after we train the same model with another dataset that has a different distribution of classes. In Fine-tuning, an approach of Transfer Learning, we have a dataset, and we make an 80-20 split and use 80% of it in training. Then we train the same model with the remaining 20%. Usually, we change the learning rate to a smaller one, so it does not have a significant impact on the already adjusted weights. To decide which method to choose, one should experiment first by using transfer learning as it is easy and fast, and if it does not suffice the purpose, then use fine tuning.

**19) Can you share some tricks or techniques that you use to fight to overfit a deep learning model and get better generalization?**

Overfitting of a model is defined when, a model performs well on the training data (low bias) and performs badly / poorly on the test data (high variance). In short, the model has learned over a certain pattern of data and is not useful for any other data. Overfitting can be detected by checking the performance metrics like loss and accuracy of a given model. There are several tips and techniques one can use in order to reduce the over fitting of a deep learning model. •

Increase the size of training data. • Reduce number of layers in the hidden layer, this will reduce the networks capacity.

- Apply regularization •
- Add dropout layers.
- Early stopping – try to stop the training before the validation loss increases.
- Make use of data augmentation.

**20) Explain the difference between Gradient Descent and Stochastic Gradient Descent.**

To begin with, Gradient descent and stochastic gradient descent both are popular machine learning and deep learning optimization algorithms which are used for updating a set of parameters in an iterative way in order to minimize an error function. In gradient descent in order to update parameters, the entire dataset set is to be considered for a particular iteration while in stochastic gradient descent, computation is carried over only one single training sample. For example, if a dataset has 10000 datapoints, then GD, will train on all the 10000 datapoints and this will take a longer time, while on the other hand, Stochastic GD, will be much faster as we will train on only a single sample and update the parameters. This is because Stochastic gradient descent usually converges faster than gradient descent on large datasets, because updates are more frequent.

**21) Which one do you think is more powerful – a two-layer NN without any activation function or a two-layer decision tree?**

- When you say a two-layer neural network, it basically contains, one input layer, one hidden layer and one output layer. An activation function is important while dealing with neural networks as they are needed while dealing with complex and nonlinear complex functional mappings between inputs and response variable.
- When a two-layer neural network has no activation function, it is just a linear network. A Neural Network without Activation function would simply be a Linear regression Model, which has limited power and does not perform good most of the times.
- Two-layer decision tree is just a decision tree with depth of 2.
- So, while comparing between these two models, two-layer neural network (without activation function) is more powerful than the two-layer decision tree, since two-layer neural network will take more attributes into consideration while building a model and in case of 2-layer decision tree, only 2 or 3 attributes will be considered.

**22) Can you name the breakthrough project that garnered the popularity and adoption of deep learning?**

- The last decade has seen remarkable improvements in the ability of computers to understand the world around them. One of these breakthroughs is, an artificial intelligence technique called deep learning.
- Deep learning, unlike machine learning is based on neural networks, a type of data structure loosely inspired by networks of biological neurons. Neural networks are organized into layers, with inputs from one layer connected to outputs from the next layer.

- Computer scientists have been experimenting with neural networks since the 1950s. But two significant breakthroughs—one in 1986, the other in 2012—laid the foundation for today's vast deep learning industry.
- The fortunes of neural networks were revived by a famous 1986 paper that introduced the concept of backpropagation, a practical method to train deep neural networks.
- Backpropagation made deeper networks more computationally tractable, but those deeper networks still required more computing resources than shallower networks.
- Research results in the 1990s and 2000s often suggested diminishing returns to making neural networks more complex. Then a famous 2012 paper which described a neural network dubbed AlexNet after lead researcher Alex Krizhevsky—transformed people's thinking.
- Dramatically deeper networks could deliver breakthrough performance, but only if they were combined with ample computing power and lots and lots of data.

### **23) Differentiate between bias and variance with respect to deep learning models and how can you achieve a balance between the two?**

While understanding predictions, understanding the prediction errors is most important. There are mainly two broad types of errors, reducible and irreducible. In reducible errors we have two kinds, bias and variance. Gaining a proper understanding of these errors helps one built an accurate model by avoiding overfitting and underfitting of the model.

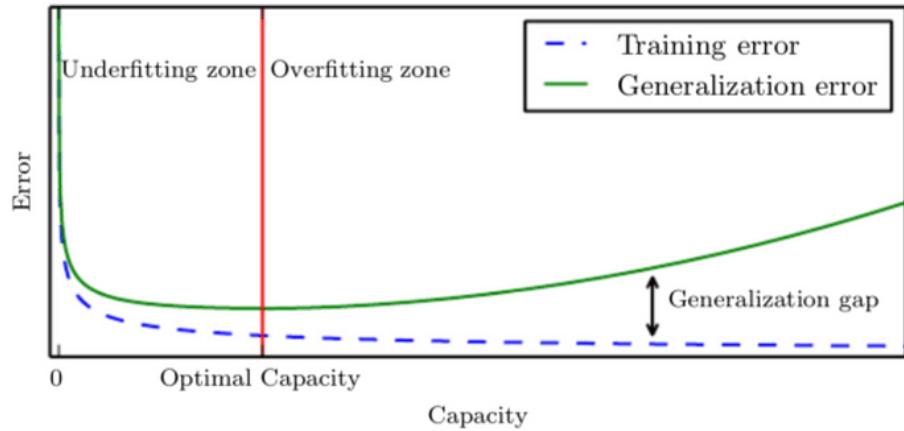
In order to obtain the optimal balance between the two errors, the model must always aim at maintain a low bias and a low variance. An optimal balance of bias and variance would never overfit or underfit the model.

Bias – In the above diagram, the training error (blue dotted line) is high in the initial stage (high bias) and then decreases sustainably (low bias). High bias means, the data is under fitting, and hence the data must have a low bias to achieve good results. In order to achieve low bias:

- I. Try increasing the number of iterations / epochs
- II. Try a bigger network

Variance – the variance in deep learning is nothing but the difference between the validation error and the training error. In the above figure, we can see that the gap between the training error and validation error is high, i.e., the variance is high. This is the case of overfitting. The model should have low variance and can be achieved by: i. Increasing the training data ii. Using regularization iii.

Using different neural network architectures.



#### 24) What are your thoughts about using GPT3 for our business?

GPT-3, or the third generation Generative Pre-trained Transformer, is a neural network machine. GPT-3 is a text predictor. Given a text or phrase, GPT-3 returns a human-like response to text completion in natural language. GPT-3 has a wide range of applications serving the industry today. It is a powerful tool that can create applications for responding to customer queries, language translator (say, asking a question in English and expecting an answer in Spanish) etc.

GPT3 can also do everything from creating spreadsheets to building complex CSS or even deploying Amazon Web Services (AWS) instances. So, can using GPT-3 help your business? Well, it can help in many ways. It all depends on what you need it to do, but it is a super versatile deep learning model applied to many applications.

Some more applications of GPT-3 that you can probably use in your business are:

- Generate emails from short descriptions. An application that can expand the given brief description into a formatted and grammatically correct professional email.
- Generate python codes from a description. Generate Flask (Python) API code just by describing the functions in English using GPT-3.
- Generate a deep learning model based on a description. For more details related to GPT-3 applications.

**25) Can you train a neural network without using back-propagation? If yes, what technique will you use to accomplish this?**

- In a neural network, back propagation is the process of repeatedly adjusting the weights of the layers in the network in order to minimise the difference between the actual output and the desired output, i.e., the loss.
- These adjusted weights result in making the hidden units of neural network to represent key features of the data. Are there any other ways to carry on the process rather than back propagation?
- Indeed, there are various optimization algorithms that does not require back-propagation to train the neural network.
- Among them are evolutionary optimization and Jeff Hinton's capsule routing. However, none of these methods exhibit a competitive performance against back-propagation based algorithms.

**26) Describe your research experience in the field of deep learning?**

**27) Explain the working of a perceptron.**

- Perceptron's were developed in the 1950s and 1960s by the scientist Frank Rosenblatt, inspired by earlier work by Warren McCulloch and Walter Pitts.
- A perceptron is one of the simplest ANN (artificial neural network) unit that does certain computations in order to detect features or business intelligence in the input data.
- Perceptron is based on an artificial neuron called a threshold logic unit (TLU)
- The inputs and output are numbers rather than binary values and each input connection is associated with a weight.
- The TLU computes a weighted sum of its inputs:  $(z = w_1 x_1 + w_2 x_2 + \dots + w_n x_n = w^T x)$ , then applies a step function to that sum and outputs the result:  $h_w(x) = \text{step}(z)$ , where  $z = w^T x$ .
- A single TLU can be used for simple linear binary classification.

**28) Differentiate between a feed-forward neural network and a recurrent neural network.**

**29) Why don't we see the exploding or vanishing gradient problem in feed-forward neural networks?**

**30) How do you decide the size of the filter when performing a convolution operation in a CNN?**

- While performing a convolution operation in CNN, filters detect spatial patterns such as edges in images by detecting the changes in the intensity of values of the images.
- There is no particular answer to how many filters or the best number of filters one can use.
- To decide the filter size, I would say it strongly depends on the type and complexity of the image data.
- A fair number of features is learned from experience after repeatedly working with similar types of datasets.
- In general, the more features you want to capture in an image, the higher the number of filters required in a CNN. The number of filters is a hyper-parameter that can be later tuned.

**31) When designing a CNN, can we find out how many convolutional layers should we use?**

- While designing a CNN, Convolutional layers are the layers where filters are applied to the original image, or to other feature maps in a deep CNN.
- The more convolutional layers the better as each convolutional layer reduces the number of input features to the fully connected layers, although after about two or three layers the accuracy gain becomes rather small so you need to decide whether your main focus is generalisation accuracy or training time.
- All image recognition tasks are different so the best method is to simply try incrementing the number of convolutional layers one at a time until you are satisfied by the result.

**32) What do you understand by a computational graph?**

**33) Differentiate between PCA and Autoencoders.**

34.Which one is better for reconstruction linear autoencoder or PCA?

35.How is deep learning related to representation learning?

36.Explain the Borel Measurable function.

37.How are Gradient Boosting and Gradient Descent different from each other?

- 38.In a logistic regression model, will all the gradient descent algorithms lead to the same model if run for a long time?
- 39.What is the benefit of shuffling a training dataset when using batch gradient descent?
40. Explain the cross-entropy loss function.
41. Why is cross-entropy preferred as the cost function for multi-class classification problems?
- 42.What happens if you do not use any activation functions in a neural network?
- 43.What is the importance of having residual neural networks?
- 44.There is a neuron in the hidden layer that always results in a large error in backpropagation. What could be the reason for this?
- 45.Explain the working of forwarding propagation and backpropagation in deep learning.
- 46.Is there any difference between feature learning and feature extraction?
- 47.Do you know the difference between the padding parameters valid and the same padding in a CNN?
- 48.How does deep learning outperform traditional machine learning models in time series analysis?
- 49.Can you explain the parameter sharing concept in deep learning?
- 50.How many trainable parameters are there in a Gated Recurrent Unit cell and in a Long Short Term Memory cell
51. What are the key components of LSTM ?
- 51.What are the components of a General Adversarial Network?

**1) State the main differences between supervised and unsupervised Deep learning procedures?**

Supervised learning is information examining function that theorizes an activity from the labeled training data. The set of training data is composed of training samples which are arranged in combinations fused with input objects. Unlike the

supervised process, the unsupervised procedure does not need labeling information explicitly, and the operations can be carried out without the same.

## **2) Explain the concept of 'overfitting' in the specific field.**

Overfitting is one of the most common issues that take place in deep learning. It generally appears when the sound of specific data is apprehended by a deep learning algorithm. It also occurs when the particular algorithm is well suitable for the data and shows up when the algorithm or model indicates high variance and low bias.

## **Machine Learning Interview Questions & Answers for 2021**

### **3) What is inductive reasoning machine learning?**

The idea of inductive justification mainly aids in making the right judgments based on the previously assembled pieces of evidence and data. Inductive reasoning operates mostly the entire function of analytical learning and is highly beneficial for making accurate decisions and theoretical assumptions in complicated project works.

### **4) State few methods in which you will demonstrate the core concept of machine learning**

The idea of deep learning is similar to that of machine learning. The technical ideology can often sound complicated to a general mind. Thus it is best to pick examples from universal laws of decision making. The [deep learning](#) interface includes making sound decisions based on the gathered data from the past. For instance, if a kid gets hurt by a particular object while playing, he is likely to reconsider the occurred event before touching it again. The concept of deep learning functions in a comparably similar manner.

### **5) Name the categories of issues that are solved by regularization**

The process of regularization is mainly used to determine issues related to overfitting. It is primarily due to the castigation of the loss function and is managed by enumerating a multiplex of L2 (Ridge) ORL1 (LASSO).

## **6) How to predict and choose the appropriate formula to solve issues on classification?**

Choosing a suitable algorithm can often be critical and using the correct strategy is very important. The process of cross-confirmation is highly advantageous in this scenario which involves examining a bulk of formulas together. Analyzing a stack of systems together will break down the core hindrances and provide the right method for issues of categorization or classification.

## **7) What is the use of Fourier Transform in Deep Learning?**

The particular package is highly efficient for analyzing and managing and maintaining large databases. The software is infused with a high-quality feature called spectral portrayal, and you can effectively utilize it to generate real-time array data. This is extremely helpful for processing all categories of signals.

## **8) What can be some of the most effective schemes to lower dimensionality issues?**

This particular issue mainly occurs while evaluating and interpreting massive organizational databases. The foremost approach to trim down this problem is to use system dimensionality contraction anatomies like the PCA or ICA. This will be helpful for getting first-hand preparation for diminishing the capacity issue. Other than that, attributes with multiple nodes and points present in the system can cause similar errors time and again and this is dismissing the complex features.

## **9) Provide an overview of PCA and mention the numerical steps of the same.**

The package as mentioned earlier is one of the most popular software in today's industry. It is used to detect the data specifications that are often not identified with a generic approach. It makes it easier for researchers and evaluators to understand the fundamental briefing and lowdown of complex information. The most significant advantage of the Principal component analysis is that it allows simplified presentation of the collected outcomes with crisp and simple explanatory that are easy to understand.

- Assimilate
- Evaluate covariance
- Consider Eigenvalues
- Realign information
- Contemplate the gathered data
- Bi-conspire the collected data

## **10) How shall you know that it is the right time to utilize classification other than reversion?**

As the former terminology suggests, classification involves the technique of recognition. The purpose of regression is to use intuitive methodologies to predict specific stimulation, whereas categorization is used to interpret the affinity of the data to a particular group. Therefore, the method of categorization is mainly second-handed when the outcomes of the algorithm are to be sent back to definite sections of data sets. It is not a straight-cut way of detecting a particular data but can always be utilized while searching for similar categories of information. This is highly effective for system learning via provided input and eventually using it for accurate data detection in project work.

## **11) Describe the concept of Machine learning in your own words**

Deep learning is often termed hierarchical learning due to its hyper-rich design that utilizes the neural net to run the operation of machine learning, and the inputs are fused in a specific order. It is also known as hierarchical learning is an extension of the clan of machine learning. The field of Machine learning is vast and holds the most peak complexities of the data science and is mainly used for fostering web applications, detecting patterns in data sets, labeling out key features, and recognizing imageries.

## **12) State some of the simplest ways to dodge overfitting**

The issue generally occurs when a limited stack of information is used. To obtain a smooth functional flow, the system demands a widened data set. The problem can be prevented from recurrence by merely utilizing the maximum information stack or utilizing the process of cross-affirmation. You will be able to overcome the issue quite easily as during this particular process; the information multiplies into several units while validating the information and shall finally conclude with the algorithm.

## **13) Name the several initiatives used in the particular field**

There are ample access ways to machine learning, but there are a certain amount of recorded skills that are mostly used in today's industry.

1. Cognitive approach
2. Analyzing approach
3. Problem-solving

4. Allegorical approach
5. Approach to classification
6. Elementary approach

**14) Explain the theory of autonomous form of deep learning in few words**

There are multiple forms and categories of the particular subject, but the autonomous pattern indicates independent or unspecified mathematical bases that are free from any specific categorizer or formula.

**15) What is referred to as ‘genetic computerizing’ in the field of data science?**

As the name of the method already suggests, the notion of genetic computerizing aids is one of the critical procedures used in deep learning. This exemplary involves analyzing and picking out the appropriate out of the stack of outcomes.

**16) State one of the finest procedures often utilized to overcome the issue of overfitting**

Usually, the problem of overfitting can be interrupted with the help of increased data usage, but if the problem is still appearing, one can apply the method of ‘Isotonic regression.’

**17) What do you know about the PAC learning procedure?**

Among the various evaluating techniques, the PAC is another form of learning scheme that is widely utilized to understand the learning set of rules and figure out their respective adeptness in an analytical method. The particular technique was first introduced to the industry in the year 1984 and has undergone several advancements since then.

**18) What is the ultimate use of Deep learning in today’s age and how is it aiding data scientists?**

The particular subject area has brought about a significant change or revolution in the domain of machine learning and data science. The concept of a complex neural network (DNN) is the main center of attention for data scientists and is widely taken advantage of to proceed with the next level of machine learning operations. The emergence of deep learning has also aided in clarifying and simplifying issues based on algorithms due to its utmost flexibility and adaptable nature. It is one of the rare procedures that allow the movement of data in independent pathways. Data scientists are viewing this particular medium as an

extended and advanced additive to the existing process of machine learning and utilizing for the same for solving complex day-to-day issues.

### **19) State the critical segments of affiliated analyzing strategies**

The essential components of the above mention techniques include the following,

- Information recovery
- Ground Truth recovery
- Cross-confirmation strategy
- Query category
- Accounting metric
- Connotation test

### **20) Differentiate between deep learning and fictitious or artificial learning**

The concept of factitious learning or artificial learning has taken over the new-age business spectrum. It is used in various fields to break down or simplify complex and hyper-rich databases and improve business strategies. The method of artificial learning is a supplementary character to the process of deep learning and involves artificial intelligence, automatic language convention, loop filling, and other automated mechanisms along with the core methodology. On the other hand, deep learning includes introducing formulas and a set of rules concerning assembled records and data from the past.

### **Q20) Explain the role of supervised learning procedure in the particular field**

Supervised learning is a mere combination of an expected output and input element. This kind of model helps in evaluating the training information and finally generates a fundamental objective that is often utilized for calibrating upcoming samples. To break it down in a more simplified manner, the particular model is used for intact categorization, dialect recognition, backsliding, commentate strings, and also forecast time arrays.

### **21) How does the method of unsupervised learning aid in deep learning?**

Unlike supervised learning, this is a type of process where the involvement of categorization is nil. It is solely used to detect the unrevealed or uncovered attributes and formation in an unidentified set of information. Other than the mentioned function, the specific method is also utilized to perform the following tasks.

- Detect data jamming or data entanglement

- Detect low spatial data depiction
- Point out the appropriate data alignment
- Locate alluring data intersection and links
- Data clarification

**22) Mention the three steps to build the necessary assumption structure in deep learning**

The process of developing an assumption structure involves three specific actions. The foremost step includes algorithm development. This particular process is lengthy as the out has to undergo several processes prior to the outcome generation. The second step involves algorithm analyzing which indicates the in-process methodology. The third step is all about implementing the generated algorithm in the final procedure. The entire framework is interlinked and requires utmost continuity throughout the process.

**23) Define the concept of the perceptron**

The above-titled terminology fundamentally refers to the model used for supervised categorization that indicates a single input among the various existing non-binary outcomes.

**24) Demonstrate the significant elements suffused in the Bayesian logic system**

There are mainly two elements involved in the particular system, and the former one includes rational explanatory infused with an array of Bayesian specifications that grasps the approximate framework of the specific field. The other element holds a quantitative approach towards the same and is mainly used to record or capture the calculable data in the specific domain.

**25) Define the concept of an additive learning algorithm**

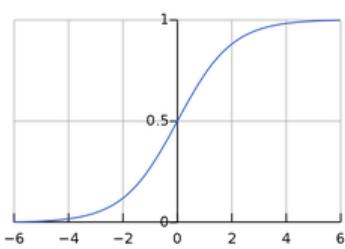
The above-mentioned technique is referred to as the method of algorithms capturing learning elements from a given set of information which is an accessible post to the generation of a classifier that has been produced from the existing set of data.

1. What is the difference between a Perceptron and Logistic Regression?

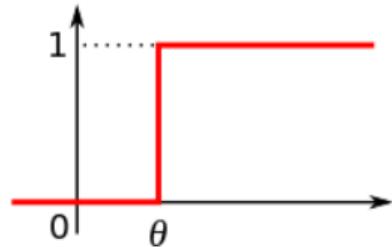
A Multi-Layer Perceptron (MLP) is one of the most basic [neural networks](#) that we use for classification. For a binary classification problem, we know that the output can be either 0 or 1. This is just like our simple logistic regression, where

we use a logit function to generate a probability between 0 and 1. So, what's the difference between the two?

Simply put, it is just the difference in the threshold function! When we restrict the logistic regression model to give us either exactly 1 or exactly 0, we get a Perceptron model:



Logistic regression - Logit function



Perceptron step function

## 2. Can we have the same bias for all neurons of a hidden layer?

Essentially, you can have a different bias value at each layer or at each neuron as well. However, it is best if we have a bias matrix for all the neurons in the hidden layers as well. A point to note is that both these strategies would give you very different results.

## 3. What if we do not use any activation function(s) in a neural network?

The main aim of this question is to understand why we need **activation functions** in a neural network. You can start off by giving a simple explanation of how neural networks are built:**Step 1:** Calculate the sum of all the inputs ( $X$ ) according to their weights and include the bias term:

$$Z = (\text{weights} * X) + \text{bias}$$

**Step 2:** Apply an activation function to calculate the expected output:

$$Y = \text{Activation}(Z)$$

Steps 1 and 2 are performed at each layer. If you recollect, this is nothing but forward propagation! Now, what if there is no activation function?

Our equation for Y essentially becomes:

$$Y = Z = (\text{weights} * X) + \text{bias}$$

Wait – isn't this just a simple linear equation? Yes – and that is why we need activation functions. A linear equation will not be able to capture the complex patterns in the data – this is even more evident in the case of deep learning problems.

In order to capture non-linear relationships, we use activation functions, and that is why a neural network without an activation function is just a linear regression model.

4. In a neural network, what if all the weights are initialized with the same value?

In simplest terms, if all the neurons have the same value of weights, each hidden unit will get exactly the same signal. While this might work during forward propagation, the derivative of the cost function during backward propagation would be the same every time. In short, there is no learning happening by the network! What do you call the phenomenon of the model being unable to learn any patterns from the data? Yes, [underfitting](#).

Therefore, if all weights have the same initial value, this would lead to underfitting.

*Note: This question might further lead to questions on exploding and vanishing gradients, which I have covered below.*

## 5. List the supervised and unsupervised tasks in Deep Learning.

Now, this can be one tricky question. There might be a misconception that deep learning can only solve unsupervised learning problems. This is not the case.

Some example of Supervised Learning and Deep learning include:

- Image classification
- Text classification
- Sequence tagging

On the other hand, there are some unsupervised deep learning techniques as well:

- Word embeddings (like Skip-gram and Continuous Bag of Words):
- Autoencoders

## 6. What is the role of weights and bias in a neural network?

This is a question best explained with a real-life example. Consider that you want to go out today to play a cricket match with your friends. Now, a number of factors can affect your decision-making, like:

- How many of your friends can make it to the game?
- How much equipment can all of you bring?
- What is the temperature outside?

And so on. These factors can change your decision greatly or not too much. For example, if it is raining outside, then you cannot go out to play at all. Or if you

have only one bat, you can share it while playing as well. The magnitude by which these factors can affect the game is called the weight of that factor.

Factors like the weather or temperature might have a higher weight, and other factors like equipment would have a lower weight.

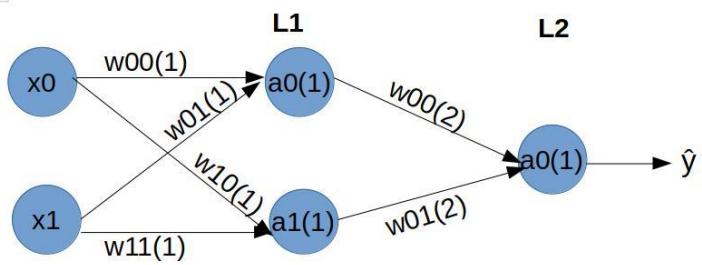
However, does this mean that we can play a cricket match with only one bat? No – we would need 1 ball and 6 wickets as well. This is where bias comes into the picture. Bias lets you assign some threshold which helps you activate a decision-point (or a neuron) only when that threshold is crossed.

## 7. How does forward propagation and backpropagation work in deep learning?

Now, this can be answered in two ways. If you are on a phone interview, you cannot perform all the calculus in writing and show the interviewer. In such cases, it best to explain it as such:

- **Forward propagation:** The inputs are provided with weights to the hidden layer. At each hidden layer, we calculate the output of the activation at each node and this further propagates to the next layer till the final output layer is reached. Since we start from the inputs to the final output layer, we move forward and it is called forward propagation
- **Backpropagation:** We minimize the cost function by its understanding of how it changes with changing the weights and biases in a neural network. This change is obtained by calculating the gradient at each hidden layer (and using the chain rule). Since we start from the final cost function and go back each hidden layer, we move backward and thus it is called backward propagation

For an in-person interview, it is best to take up the marker, create a simple neural network with 2 inputs, a hidden layer, and an output layer, and explain it.



## Forward propagation:

### Forward Propagation

Assuming Activation function = Sigmoid( $\sigma$ )

At Layer L1,

$$z_0^{(1)} = [w_{00}^{(1)} \cdot x_0 + b_{00}^{(1)}] + [w_{01}^{(1)} \cdot x_1 + b_{01}^{(1)}] \quad \text{and}$$

$$z_1^{(1)} = [w_{10}^{(1)} \cdot x_0 + b_{10}^{(1)}] + [w_{11}^{(1)} \cdot x_1 + b_{11}^{(1)}]$$

After applying Activation function at L1,

$$a_0^{(1)} = \sigma(z_0^{(1)}) \quad | \quad a_1^{(1)} = \sigma(z_1^{(1)})$$

At Layer L2,

$$z_0^{(2)} = [w_{00}^{(2)} \cdot a_0^{(1)} + b_{00}^{(2)}] + [w_{01}^{(2)} \cdot a_1^{(1)} + b_{01}^{(2)}]$$

Final Output Layer,

$$\hat{y} = \sigma(z_0^{(2)}) = a_0^{(2)}$$

## Backpropagation:

At layer L2, for all weights:

At Layer L2,

$$\frac{\delta C}{\delta w_{00}^{(2)}} = \frac{\delta C}{\delta a_0^{(2)}} \cdot \frac{\delta a_0^{(2)}}{\delta z_0^{(2)}} \cdot \frac{\delta z_0^{(2)}}{\delta w_{00}^{(2)}}$$

$$\frac{\delta C}{\delta w_{01}^{(2)}} = \frac{\delta C}{\delta a_0^{(2)}} \cdot \frac{\delta a_0^{(2)}}{\delta z_0^{(2)}} \cdot \frac{\delta z_0^{(2)}}{\delta w_{01}^{(2)}}$$

At layer L1, for all weights:

At Layer L1,

1.

$$\frac{\delta C}{\delta w_{00}^{(1)}} = \frac{\delta C}{\delta a_0^{(1)}} \cdot \frac{\delta a_0^{(1)}}{\delta z_0^{(1)}} \cdot \frac{\delta z_0^{(1)}}{\delta w_{00}^{(1)}} = \left[ \frac{\delta C}{\delta a_0^{(2)}} \cdot \frac{\delta a_0^{(2)}}{\delta z_0^{(2)}} \cdot \frac{\delta z_0^{(2)}}{\delta a_0^{(1)}} \right] \cdot \frac{\delta a_0^{(1)}}{\delta z_0^{(1)}} \cdot \frac{\delta z_0^{(1)}}{\delta w_{00}^{(1)}}$$

2.

$$\frac{\delta C}{\delta w_{01}^{(1)}} = \frac{\delta C}{\delta a_0^{(1)}} \cdot \frac{\delta a_0^{(1)}}{\delta z_0^{(1)}} \cdot \frac{\delta z_0^{(1)}}{\delta w_{01}^{(1)}} = \left[ \frac{\delta C}{\delta a_0^{(2)}} \cdot \frac{\delta a_0^{(2)}}{\delta z_0^{(2)}} \cdot \frac{\delta z_0^{(2)}}{\delta a_0^{(1)}} \right] \cdot \frac{\delta a_0^{(1)}}{\delta z_0^{(1)}} \cdot \frac{\delta z_0^{(1)}}{\delta w_{01}^{(1)}}$$

3.

$$\frac{\delta C}{\delta w_{10}^{(1)}} = \frac{\delta C}{\delta a_1^{(1)}} \cdot \frac{\delta a_1^{(1)}}{\delta z_1^{(1)}} \cdot \frac{\delta z_1^{(1)}}{\delta w_{10}^{(1)}} = \left[ \frac{\delta C}{\delta a_0^{(2)}} \cdot \frac{\delta a_0^{(2)}}{\delta z_0^{(2)}} \cdot \frac{\delta z_0^{(2)}}{\delta a_1^{(1)}} \right] \cdot \frac{\delta a_0^{(1)}}{\delta z_0^{(1)}} \cdot \frac{\delta z_0^{(1)}}{\delta w_{00}^{(1)}}$$

4.

$$\frac{\delta C}{\delta w_{11}^{(1)}} = \frac{\delta C}{\delta a_1^{(1)}} \cdot \frac{\delta a_1^{(1)}}{\delta z_1^{(1)}} \cdot \frac{\delta z_1^{(1)}}{\delta w_{11}^{(1)}} = \left[ \frac{\delta C}{\delta a_0^{(2)}} \cdot \frac{\delta a_0^{(2)}}{\delta z_0^{(2)}} \cdot \frac{\delta z_0^{(2)}}{\delta a_1^{(1)}} \right] \cdot \frac{\delta a_0^{(1)}}{\delta z_0^{(1)}} \cdot \frac{\delta z_0^{(1)}}{\delta w_{11}^{(1)}}$$

You need not explain with respect to the bias term as well, though you might need to expand the above equations substituting the actual derivatives.

## 8. What are the common data structures used in Deep Learning?

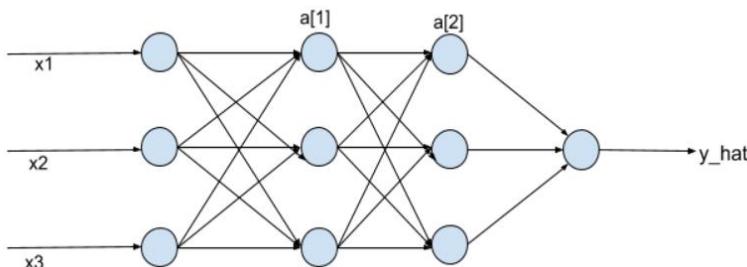
Deep Learning goes right from the simplest data structures like lists to complicated ones like computation graphs. Here are the most common ones:

- **List:** An ordered sequence of elements (You can also mention NumPy ndarrays here)
- **Matrix:** An ordered sequence of elements with rows and columns
- **Dataframe:** A dataframe is just like a matrix, but it holds actual data with the column names and rows denoting each datapoint in your dataset. If marks of 100 students, their grades, and their details are stored in a dataframe, their details are stored as columns. Each row will represent the data of each of the 100 students
- **Tensors:** You will work with them on a daily basis if you have ventured into deep learning. Used both in PyTorch and TensorFlow, tensors are like the basic programming unit of deep learning. Just like multidimensional arrays, we can perform numerous mathematical operations on them. Read more about tensors [here](#)

- **Computation Graphs:** Since deep learning involves multiple layers and often hundreds, if not thousands of parameters, it is important to understand the flow of computation. A computation graph is just that. A computation graph gives us the sequence of operations performed with each node denoting an operation or a component in the neural network

## 9. Why should we use Batch Normalization?

Once the interviewer has asked you about the fundamentals of deep learning architectures, they would move on to the key topic of improving your deep learning model's performance. Batch Normalization is one of the techniques used for reducing the training time of our deep learning algorithm. Just like normalizing our input helps improve our logistic regression model, we can normalize the activations of the hidden layers in our deep learning model as well:



We basically normalize  $a[1]$  and  $a[2]$  here. This means we normalize the inputs to the layer, and then apply the activation functions to the normalized inputs.

## 10. List the activation functions you have used so far in your projects and how you would choose one.

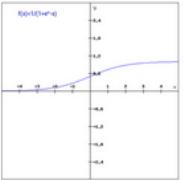
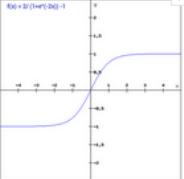
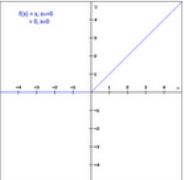
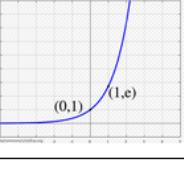
The most common activation functions are:

- Sigmoid
- Tanh
- ReLU

- Softmax

While it is not important to know all the activation functions, you can always score points by knowing the range of these functions and how they are used.

Here is a handy table for you to follow:

Function	Mathematical Expression	Range	Plot
Sigmoid	$\frac{1}{1 + e^{-x}}$	(0, 1)	
tanh	$2 * \text{sigmoid}(2x) - 1$	(-1, 1)	
ReLU	$\max(0, x)$	[0, inf)	
Softmax	$s(x_i) = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}}$	[0, 1]	

## 11. Why does a Convolutional Neural Network (CNN) work better with image data?

The key to this question lies in the Convolution operation. Unlike humans, the machine sees the image as a matrix of pixel values. Instead of interpreting a shape like a petal or an ear, it just identifies curves and edges. Thus, instead of looking at the entire image, it helps to just read the image in parts. Doing this for a 300 x 300 pixel image would mean dividing the matrix into smaller 3 x 3 matrices and dealing with them one by one. This is convolution.

Mathematically, we just perform a small operation on the matrix to help us detect features in the image – like boundaries, colors, etc.

$$Z = X * f$$

Here, we are convolving (\* operation – not multiplication) the input matrix X with another small matrix f, called the kernel/filter to create a new matrix Z. This matrix is then passed on to the other layers.

If you have a board/screen in front of you, you can always illustrate this with a simple example:

3	9	4
11	1	8
2	13	7

0	0
1	1

Thus, the filter 'f' considers  $2 \times 2$  subparts of the X matrix at the time and performs the convolution operation

- $(3 \times 0) + (9 \times 0) + (11 \times 1) + (1 \times 1) = 12$
- $(9 \times 0) + (4 \times 0) + (1 \times 1) + (8 \times 1) = 9$
- $(11 \times 0) + (1 \times 0) + (2 \times 1) + (13 \times 1) = 15$
- $(1 \times 0) + (8 \times 0) + (13 \times 1) + (7 \times 1) = 20$

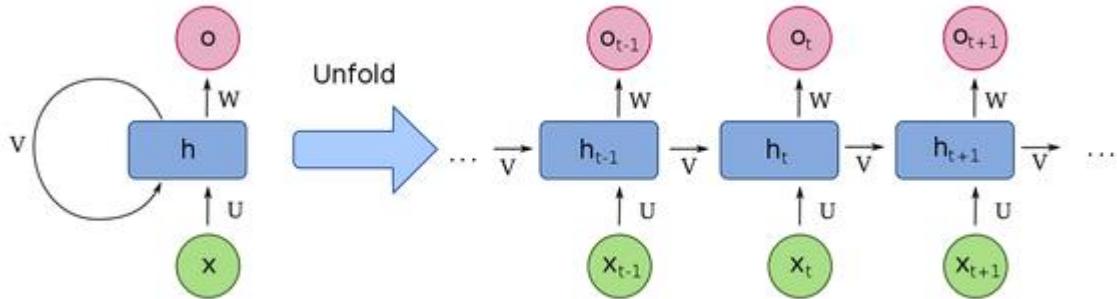
Thus,  $Z =$

12	9
15	20

## 12. Why do RNNs work better with text data?

The main component that differentiates Recurrent Neural Networks (RNN) from the other models is the addition of a loop at each node. This loop brings the **recurrence** mechanism in RNNs. In a basic Artificial Neural Network (ANN), each input is given the same weight and fed to the network at the same time. So, for a sentence like “I saw the movie and hated it”, it would be difficult

to capture the information which associates “it” with the “movie”.



The addition of a loop is to denote preserving the previous node's information for the next node, and so on. This is why RNNs are much better for sequential data, and since text data also is sequential in nature, they are an improvement over ANNs.

**13. In a CNN, if the input size 5 X 5 and the filter size is 7 X 7, then what would be the size of the output?**

This is a pretty intuitive answer. As we saw above, we perform the convolution on ‘x’ one step at a time, to the right, and in the end, we got Z with dimensions 2 X 2, for X with dimensions 3 X 3. Thus, to make the input size similar to the filter size, we make use of padding – adding 0s to the input matrix such that its new size becomes at least 7 X 7. Thus, the output size would be using the formula:

$$\text{Dimension of image} = (n, n) = 5 \times 5$$

$$\text{Dimension of filter} = (f, f) = 7 \times 7$$

Padding = 1 (adding 1 pixel with value 0 all around the edges)

Dimension of output will be  $(n+2p-f+1) \times (n+2p-f+1) = 1 \times 1$

**14. What's the difference between valid and same padding in a CNN?**

This question has more chances of being a follow-up question to the previous one. Or if you have explained how you used CNNs in a computer vision task, the interviewer might ask this question along with the details of the padding parameters.

- Valid Padding: When we do not use any padding. The resultant matrix after convolution will have dimensions  $(n - f + 1) \times (n - f + 1)$
- Same padding: Adding padded elements all around the edges such that the output matrix will have the same dimensions as that of the input matrix

## 15. What do you mean by exploding and vanishing gradients?

The key here is to make the explanation as simple as possible. As we know, the [gradient descent algorithm](#) tries to minimize the error by taking small steps towards the minimum value. These steps are used to update the weights and biases in a neural network. However, at times, the steps become too large and this results in larger updates to weights and bias terms – so much so as to cause an overflow (or a NaN) value in the weights. This leads to an unstable algorithm and is called an exploding gradient.

On the other hand, the steps are too small and this leads to minimal changes in the weights and bias terms – even negligible changes at times. We thus might end up training a deep learning model with almost the same weights and biases each time and never reach the minimum error function. This is called the vanishing gradient.

A point to note is that both these issues are specifically evident in Recurrent Neural Networks – so be prepared for follow-up questions on RNN!

## 16. What are the applications of transfer learning in Deep Learning?

I am sure you would have a doubt as to why a relatively simple question was included in the Intermediate Level. The reason is the sheer volume of subsequent questions it can generate! The use of [transfer learning](#) has been one of the key milestones in deep learning. Training a large model on a huge dataset, and then using the final parameters on smaller simpler datasets has led to defining breakthroughs in the form of Pretrained Models. Be it Computer Vision or NLP, pretrained models have become the norm in research and in the industry.

Some popular examples include BERT, ResNet, GPT-2, VGG-16, etc and many more.

It is here that you can earn brownie points by pointing out specific examples/projects where you used these models and how you used them as well.

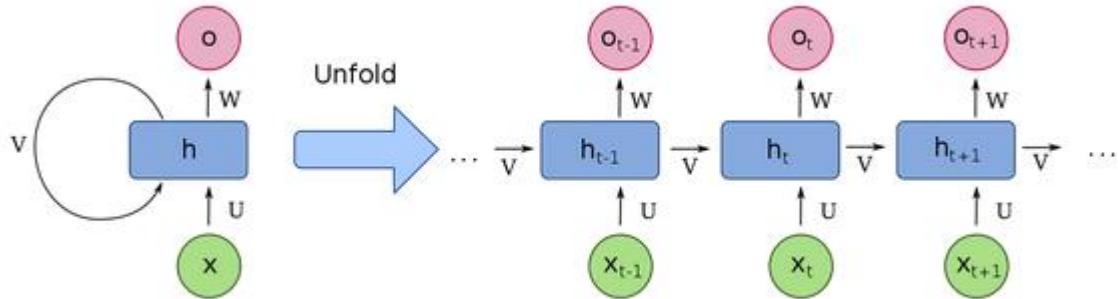
It is not possible to discuss all of them, so here are a few resources to get started:

Also, depending on the domain – with Computer Vision or Natural Language Processing, these questions can change. While it is not important to know the architecture of each model in detail, you would need to know the intuition behind them and why these models were needed in the first place.

Again, just like the intermediate level, it is important to always bring in examples that you have studied or implemented yourself into the discussion.

## 17. How backpropagation is different in RNN compared to ANN?

In Recurrent Neural Networks, we have an additional loop at each node:

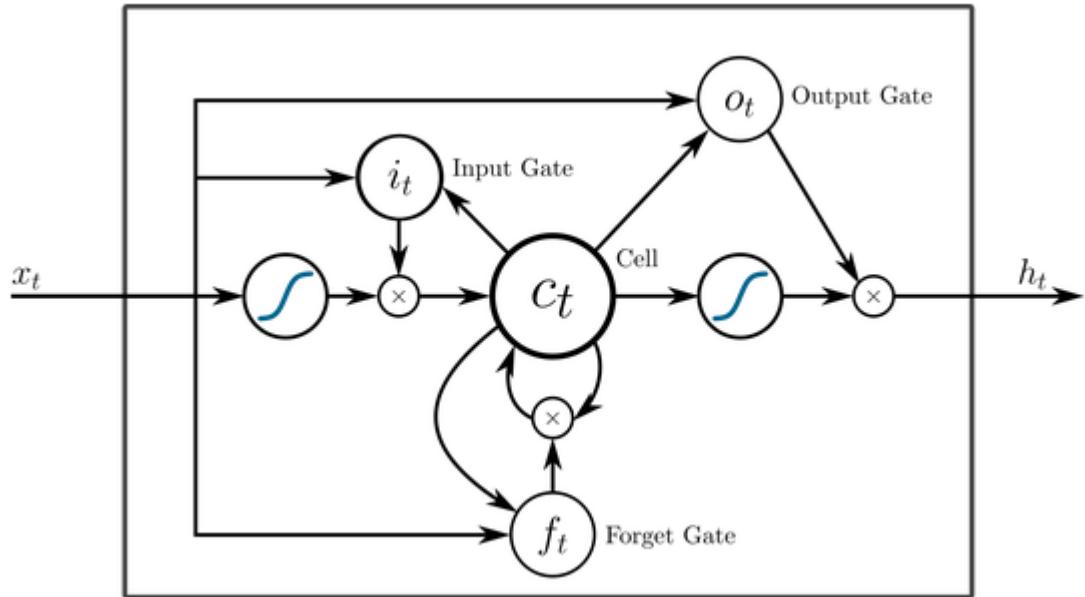


This loop essentially includes a time component into the network as well. This helps in capturing sequential information from the data, which could not be possible in a generic artificial neural network.

This is why the backpropagation in RNN is called Backpropagation through Time, as in backpropagation at each time step.

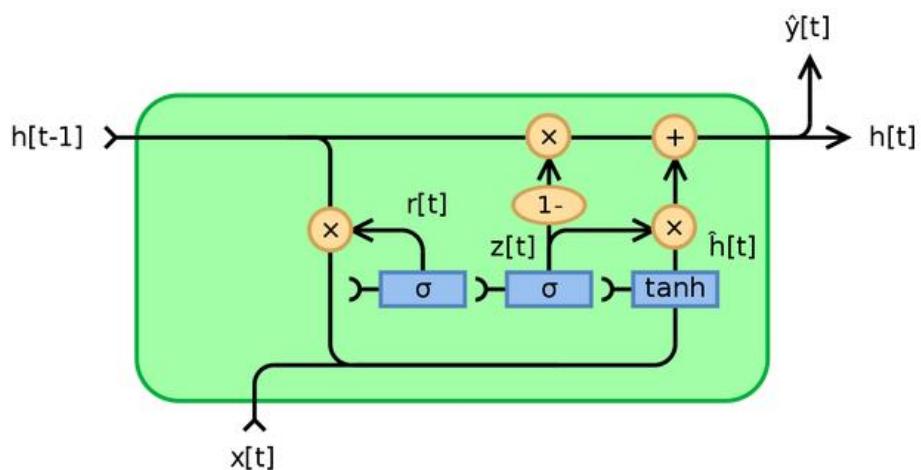
## 18. How does LSTM solve the vanishing gradient challenge?

The [LSTM model](#) is considered a special case of RNNs. The problems of vanishing gradients and exploding gradients we saw earlier are a disadvantage while using the plain RNN model. In LSTMs, we add a forget gate, which is basically a memory unit that retains information that is retained across timesteps and discards the other information that is not needed. This also necessitates the need for input and output gates to include the results of the forget gate as well.



## 19. Why is GRU faster as compared to LSTM?

As you can see, the LSTM model can become quite complex. In order to still retain the functionality of retaining information across time and yet not make a too complex model, we need GRUs. Basically, in GRUs, instead of having an additional Forget gate, we combine the input and Forget gates into a single Update Gate:



It is this reduction in the number of gates that makes GRU less complex and faster than LSTM.

## 20. How is the transformer architecture better than RNN?

Advancements in deep learning have made it possible to solve many tasks in Natural Language Processing. Networks/Sequence models like RNNs, LSTMs, etc. are specifically used for this purpose – so as to capture all possible information from a given sentence, or a paragraph. However, sequential processing comes with its caveats:

- It requires high processing power
- It is difficult to execute in parallel because of its sequential nature

This gave rise to the Transformer architecture. Transformers use what is called the attention mechanism. This basically means mapping dependencies between all the parts of a sentence.

## Q1. What are the different types of activation functions?

This is an important Deep Learning coding interview question. You must know the following types of activation functions:

1. **Sigmoid function:** It is a nonlinear function in an ANN that is mostly used in feedforward neural networks. It's a differentiable real function with positive derivatives and a certain degree of smoothness. It is written as: {"detectHand":false}.
2. **Hyperbolic tangent function (Tanh):** It is a smoother, zero-centered function (range of -1 to +1). The output is represented by: {"detectHand":false}. The primary advantage of this function is that it gives a zero-centered output that helps in backpropagation.
3. **Softmax function:** It is used to generate probability distribution from a vector of real numbers. This function returns the output between 0 and 1, with the sum of probabilities equals to 1. This is written as:

- { "detectHand":false }. It is used in multi-class models, returning probabilities of each class, with the target having the highest probability.
4. **Softsign function:** It is commonly used in regression computation issues and text-to-speech applications. It's a quadratic polynomial, written as: { "detectHand":false }.
  5. **Rectified linear unit of function:** It outperforms other AFs in generalization and performance. The function is roughly linear and preserves the features of linear models, making gradient-descent approaches easier to optimize. It is written as: { "detectHand":false }.
  6. **Exponential linear unit of function:** The major advantage of this function is that it can solve the vanishing gradient problem by employing identity for positive values and boosting the model's learning properties. It is represented by: { "detectHand":false }.

## **Q2. How does recurrent neural network backpropagation vary from artificial neural network backpropagation?**

Each node in a recurrent neural network has an additional loop. This makes it different from artificial neural network propagation. This loop incorporates a temporal component into the network. The main advantage of recurrent neural networks is that they allow for sequential data information. This is usually impossible with a generic artificial neural network.

## **Q3. Can a deep learning model be solely built on linear regression?**

If you are well-versed in Deep Learning, you can answer these types of Deep Learning interview questions with ease.

A deep learning model may be solely built on linear regression. However, the problem should be represented by a linear equation, which does not boost the machine learning model's predictive capacity due to the addition of nodes. Hence, building a deep learning model solely on linear regression creates no spectacular results.

## **Q4. What is a computational graph in Deep Learning?**

This is one of the important topics asked in Deep Learning interview questions.

A computational graph is a series of operations performed to take inputs and arrange them as nodes in a graph. It is a way of implementing mathematical calculations into a graph. This way, it will help in parallel processing and provide high performance in terms of computational capability.

## **Q5. What are the types of autoencoders, and where are they used?**

This is a commonly asked in Deep Learning interview question. You must have a sound understanding of what autoencoders are to answer this.

Autoencoders are used worldwide. Some of the popular usages of autoencoders are:

1. Adding color to black-white images
2. Removing noise from images
3. Dimensionality reduction
4. Feature removal and variation

You must know there are four types of autoencoders. They are:

1. Deep autoencoders
2. Convolutional autoencoders
3. Sparse autoencoders
4. Contractive autoencoders

## **Q1. What do you understand about text normalization in NLP?**

These types of Deep Learning interview questions test your fundamental knowledge of the subject.

When developing NLP tools to work with exceptional data, it's beneficial to attain a canonical representation of textual content. This is known as textual normalization. Textual normalization captures different kinds of variations into one representation.

## **Q2. Do you know what feature engineering is?**

When you employ machine learning methods to complete your modeling, you need to input pre-processed text into an NLP algorithm. This set of strategies used for this process is known as feature engineering or feature extraction. The main purpose of feature extraction is to convert the text's qualities into a numeric vector that NLP algorithms can understand. This stage is known as text representation.

## **Q3. Explain TF-IDF in NLP.**

TF-IDF is known as Term-Frequency-Inverse Document Frequency. It helps you get the importance of a particular word relative to other words in the corpus. It converts words into vectors and adds semantic information, resulting in weighted unusual words. These words can be utilized in various NLP applications. Moreover, it's a common scoring metric in information retrieval and summarization.

#### **Q4. What do you understand about POS tagging?**

A part-of-speed (POS) tagger reads the text in a language and assigns speed parts to each word, such as noun, verb, adverb, and others. POS taggers employ an algorithm to label terms in text bodies. These labels create various complex categories with tags like "noun plural" or other complicated labels.

#### **Q5. What is the difference between NLP and NLU?**

This is one of the most asked Deep Learning interview questions. The differences between NLP and NLU are:

Natural Language Understanding (NLU)	Natural Language Programming (NLP)
Aids in solving AI's complex problems.	A system that manages end-to-end conversations between computers and people simultaneously.
Allows machines to interpret unstructured input by transforming them into structured text.	Humans and machines are involved in NLP.
Concentrates on extracting meaning and context.	Focuses on interpreting language in its most literal sense.
Helps machines deduce the meaning behind the language content.	Can parse text-based on grammar, typography, structure, and point of view.

Recommended Reading: [Amazon Machine Learning Engineer Interview Prep](#)

#### Deep Learning Computer Vision Interview Questions

If you are applying for a role of a Computer Vision Engineer in any top company, you must practice the following Deep Learning computer vision interview questions to uplevel your preparation:

**Q1. What are the features detected by the initial layers of a neural network used for computer vision? How is it different from what is detected by the later neural network layers?**

Neural network's earlier layers detect simple features of an image (for example, edges or corners). As you go deeper, the features become increasingly complex, detecting patterns and shapes in the neural network. The later layers can detect intricate patterns, such as complete objects.

**Q2. How will you address the edge pixels issue during convolutional operation?**

You can use padding to address the issue of filter or kernel extracting information from the edge pixels less compared to the central pixel. Padding is the addition of one or more rows or columns of pixels along the boundary of the image.

It forms the new pixels of the picture. Therefore, it results in insufficient extraction of information from the original edge pixels. It also prevents the shrinking of an image due to the convolution operations.

**Q3. You are given a 5x5 image with a 3x3 filter and a padding p = 1. What will be the resultant image's size if a convolutional stride of s = 2 is used?**

You should know that for an  $n \times n$  image with an  $f \times f$  filter, padding  $p$ , and stride length  $s$ , resultant image's size after convolution has the shape  $n + 2p - fs + 1 \times n + 2p - fs + 1$ . Therefore, per the data provided, the resulting size of the image will be  $((5 + 2 * 1 - 3) / 2) + 1 \times ((5 + 2 * 1 - 3) / 2) + 1 = 3 \times 3$ .

**Q4. What will be the resultant image size for an RGB image of 10x10x3 convolved with a 3x3 filter?**

The convolution operation is not possible for such dimensions of an RGB image. The third dimension (number of channels) should be the same to achieve convolution. However, if the 10x10x3 image is convolved in a 3x3x3 filter, the dimensions of the resultant image will be 4x4.

**Q5. How many parameters need to be learned in pooling layers?**

The pooling layer contains hyperparameters describing the filter size and the stride length. These parameters are set and work as a fixed computation. Hence, no parameters are to be learned in the pooling layers.

1. What is an ensemble method in NLP?
2. State the steps to build a text classification system.
3. How is parsing done in NLP?
4. Differentiate between deep learning and machine learning.
5. What is a bag of words (BOW)?
6. What is Latent Semantic Indexing (LSI) in NLP?
7. What are some metrics on which NLP models are evaluated?
8. Explain the pipeline for information extraction.
9. What do you understand about autoencoders?
10. Explain the meaning of masked language modeling.
11. Explain pragmatic analysis in NLP.
12. What is the meaning of N-gram in NLP?
13. What do you mean by perplexity in NLP?
14. Explain why the inputs in computer vision problems can get huge.  
Provide a solution to overcome this challenge.
15. What should the padding be for a 10x10 image used with a 5x5 filter to get an image of the same size as the original image?
16. What method can be used to evaluate an object localization model? How does it work?
17. How will you use IoU for resolving the issue of multiple detections of the same object?
18. Give us an example of a scenario that would require the use of anchor boxes.
19. How is the Siamese Network beneficial in addressing the one-shot learning problem?
20. What purpose does grayscaling serve?
21. Explain translational equivariance.
22. Explain the object detection algorithm YOLO.
23. What do you know about dropouts?
24. Explain exploding and vanishing gradients.
25. Differentiate between bias and variance in the context of deep learning models. How can you achieve a balance between the two?

26. According to you, which one is more powerful — a two-layer neural network without any activation function or a two-layer decision tree?
27. While building a neural network architecture, how will you decide how many neurons and hidden layers should the neural network have?
28. What is an activation function? What is the use of an activation function?
29. What deep learning algorithm works best for face detection?
30. What is Stochastic Gradient Descent and how is it different from Batch Gradient Descent?
31. Explain how you would fix the constant validation accuracy in a Convolutional Neural Network (CNN)?
32. What are the differences between a shallow network and a deep network.
33. What is a tensor in deep learning?
34. What are the advantages of transfer learning?
35. Difference between multi-class and multi-label classification problems.
36. What are the different techniques to achieve data normalization?
37. What are Forward and Back Propagation in the context of deep learning?
38. List the different types of deep neural networks.
39. Define deep learning and neural networks.

40. Explain perception with an example.

41. What is the importance of data normalisation in deep learning?

42. What is a multi-layer perceptron (MLP)?

43. Define hyperparameters and discuss some common ones.

44. Explain cost function and gradient descent.

45. Define a feedforward neural network and a recurrent neural network with examples.

46. Explain the importance of activation functions in neural networks.

47. What is the Boltzmann machine?

48. Discuss backpropagation and its benefits in deep learning.

49. Explain the importance of weight initialisation in a neural network.

50. Between shallow and deep networks, which one do you think are better?

51. Discuss the difference between supervised and unsupervised algorithms.

52. What is feature extraction, and why is it required?

53. What is a deep learning model, and how do you deploy one?

54. Explain the utility of Softmax and ReLU functions.

55. Discuss how batch gradient descent and stochastic gradient descent are different.

56. Define the different layers of a convolutional neural network.

- 57.What is a long-short-term memory (LSTM), and how does it function?
- 58.Explain how epoch, batch and iteration are different.
- 59.What is Tensorflow, and why is it preferred?

## **Why do segmentation CNNs typically have an encoder-decoder style / structure?**

The encoder CNN can basically be thought of as a feature extraction network, while the decoder uses that information to predict the image segments by "decoding" the features and upscaling to the original image size.

## **What is the significance of Residual Networks?**

The main thing that residual connections did was allow for direct feature access from previous layers. This makes information propagation throughout the network much easier. One very interesting paper about this shows how using local skip connections gives the network a type of ensemble multi-path structure, giving features multiple paths to propagate throughout the network.

## **What is batch normalization and why does it work?**

Training Deep Neural Networks is complicated by the fact that the distribution of each layer's inputs changes during training, as the parameters of the previous layers change. The idea is then to normalize the inputs of each layer in such a way that they have a mean output activation of zero and standard deviation of one. This is done for each individual mini-batch at each layer i.e compute the mean and variance of that mini-batch alone, then normalize. This is analogous to how the inputs to networks are standardized. How does this help? We know that normalizing the inputs to a network helps it learn. But a network is just a series of layers, where the output of one layer becomes the input to the next. That means we can think of any layer in a neural network as the first layer of a smaller subsequent network. Thought of as a series of neural networks feeding into each other, we normalize the output of one layer before applying the activation function, and then feed it into the following layer (sub-network).

Why would you use many small convolutional kernels such as 3x3 rather than a few large ones?

This is very well explained in the VGGNet paper. There are 2 reasons: First, you can use several smaller kernels rather than few large ones to get the same receptive field and capture more spatial context, but with the smaller kernels you are using less parameters and computations. Secondly, because with smaller kernels you will be using more filters, you'll be able to use more activation functions and thus have a more discriminative mapping function being learned by your CNN.

## **Why do we need a validation set and test set? What is the difference between them?**

When training a model, we divide the available data into three separate sets:

- The training dataset is used for fitting the model's parameters. However, the accuracy that we achieve on the training set is not reliable for predicting if the model will be accurate on new samples.
- The validation dataset is used to measure how well the model does on examples that weren't part of the training dataset. The metrics computed on the validation data can be used to tune the hyperparameters of the model. However, every time we evaluate the validation data and we make decisions based on those scores, we are leaking information from the validation data into our model. The more evaluations, the more information is leaked. So we can end up overfitting to the validation data, and once again the validation score won't be reliable for predicting the behavior of the model in the real world.
- The test dataset is used to measure how well the model does on previously unseen examples. It should only be used once we have tuned the parameters using the validation set.

## **What is vanishing gradient?**

As we add more and more hidden layers, back propagation becomes less and less useful in passing information to the lower layers. In effect, as information is passed back, the gradients begin to vanish and become small relative to the weights of the networks.

1. **What is Deep Learning?**
2. **What is a Neural Network?**
3. **What are the principal differences between AI, Machine Learning, and Deep Learning?**
4. **Distinguish between supervised and unsupervised Deep Learning procedures**
5. **Do you think that a deep network is better than a shallow one?**
6. **What do you understand by 'overfitting'?**
7. **What is Backpropagation?**
8. **What is the function of the Fourier Transform in Deep Learning?**
9. **Describe the theory of the autonomous form of Deep Learning**
10. **What is the application of Deep Learning in today's age, and how does it help Data Scientists?**
11. **What are the uses of a Recurrent Neural Network (RNN)?**
12. **What makes Tensorflow the most favoured library in Deep Learning?**
13. **Explain the meaning of term weight initialization in Neural Networks**

**14. What makes zero initialization not a good weight initialization process?**

**15. Explain the significance of LSTM**

### **1. What is Deep Learning?**

Deep Learning is an advanced form of Machine Learning with an algorithm inspired by the brain's structure and function, called an Artificial Neural Network. Alexey Grigorevich Ivakhnenko published the first general in the mid-1960s while working on a Deep Learning network. Deep Learning includes the acquisition of large volumes of structured or unstructured data and complex algorithms to train Neural Networks.

### **2. What is a Neural Network?**

Neural Networks imitate the way humans learn, inspired by how neurons in our brains work, but much more straightforward. Each sheet comprises neurons called 'nodes,' conducting a variety of operations. Neural Networks get used for Deep Learning algorithms such as CNN, RNN, GAN, etc.

The most typical Neural Networks consist of three layers of the network –

- a) An input layer
- b) A hidden layer (this is the most vital layer where feature extraction takes place)
- c) An output layer

### **3. What are the principal differences between AI, Machine Learning, and Deep Learning?**

AI refers to 'Artificial Intelligence.' It's a technique that allows computers to imitate human interactions and intelligence.

Machine Learning is a subset of AI that uses statistical techniques to allow machines to enhance their performance.

Deep Learning is part of Machine Learning, which uses Neural Networks to replicate human-like decision-making.

#### **4. Distinguish between supervised and unsupervised Deep Learning procedures.**

Supervised Learning is a method in which both input and output data are given. The input and output data are named as a learning basis for future data processing.

The unsupervised procedure does not require specific labeling details, and operations can perform without the same. Cluster Analysis is a traditional Unsupervised Learning process. It gets used for exploratory Data Analysis to identify hidden patterns or Data Clustering.

#### **5. Do you think that a deep network is better than a shallow one?**

Both shallow and deep networks are good enough to approximate any feature. But deeper networks can be much more effective in computing a number of parameters at the same degree of accuracy. Deeper networks can develop deep representations. The network learns a new, more abstract representation of the input at each layer.

#### **6. What do you understand by ‘overfitting’?**

Overfitting is the most prevalent problem in Deep Learning. It typically happens when a Deep Learning algorithm perceives the noise of any specific data set.

#### **7. What is Backpropagation?**

Backpropagation is a training algorithm that gets used for multi-layered Neural Networks. Backpropagation can be described as the following:

- It can forward the distribution of training data over the network to generate output.
- It uses the target value and output value to calculate the derivative error in the output activations.
- It can be repropagated to measure the error derivative for output activations in the previous layer and to proceed for all hidden layers.

- It uses the previously measured output derivatives and all hidden layers to calculate the weight-related error derivative.

## **8. What is the function of the Fourier Transform in Deep Learning?**

Fourier transform package is highly effective for the analysis, maintenance, and management of broad databases. The program gets developed with a high-quality feature known as a special portrayal. It can be used effectively to produce real-time array data, which is extremely useful for processing all categories of signals.

## **9. Describe the theory of the autonomous form of Deep Learning.**

An autonomous pattern represents an individual or non-specific mathematical foundation exempt from any specific categorizer or formula.

## **10. What is the application of Deep Learning in today's age, and how does it help Data Scientists?**

Deep Learning has brought significant improvements and transformations to the world of Machine Learning and Data Science. The definition of a Complex Neural Network (CNN) is the main subject of concern for data scientists. It is commonly used because of its advantages in conducting next-level Machine Learning operations. The benefits of Deep Learning also include the process of clarifying and simplifying algorithm-based issues due to its extraordinarily scalable and adaptable nature. It is one of the rare techniques that allow the movement of data in separate pathways.

## **11. What are the uses of a Recurrent Neural Network (RNN)?**

The RNN gets used for sentiment analysis, text mining, and imaging. Recurrent Neural Networks may also fix time-series issues such as forecasting stock prices in a month or a quarter.

## **12. What makes Tensorflow the most favored library in Deep Learning?**

Tensorflow offers both C++ and Python APIs, making it easier to operate faster than other Deep Learning libraries, like Keras and Torch. Tensorflow supports CPU and GPU computing devices.

## **13. Explain the meaning of term weight initialization in Neural Networks.**

In Neural Networking, the initialization of weight is one of the main factors. Poor initialization of weight restricts a network from learning. On the other hand, a successful initialization of weight helps achieve faster convergence. Biases can all be initialized to zero. The basic rule for setting weights is that it must be close to zero without being too low.

#### **14. What makes zero initialization not a good weight initialization process?**

If the set of weights in the network is set to zero, all neurons on each layer will generate the same output and gradients during backpropagation. As a result, the network cannot learn because there is no source of asymmetry between neurons. That's why we need to add randomness to the method of weight initialization.

#### **15. Explain the significance of LSTM.**

LSTM refers to Long Short-Term Memory. This Artificial RNN (Recurrent Neural Network) architecture gets used in the area of Deep Learning. LSTM has input connections that make it a ‘general-purpose computer.’ It can handle not only a single data point but also entire data sequences. They are a particular form of RNN, capable of learning long-term dependencies. So, we’ve covered 15 of the most frequently asked Deep Learning interview questions that will help you get the dream job.

#### ***Q1. What are different types of Machine Learning and briefly explain them?***

The expected answer should mention supervised, unsupervised, and reinforcement learning.

**Supervised Learning** You give the algorithm labeled data and the algorithm has to learn from it and figure out how to solve future similar problems. Think of it as if you’re giving the algorithm problems and answers, the algorithm has to learn how these problems were solved in order to solve future problems in a similar manner. This is like the example where the bank learns from your habits which credit card transactions are legit and which are fraudulent.

**Unsupervised Learning** You give the algorithm a problem without any labeled data or any prior knowledge of what the answer could be. Think of it as if you're giving the algorithm problems without any answers, the algorithm has to find the best answer by driving insights from the data. This is similar to a bank clustering its customers according to various parameters and deciding who's eligible for a credit card offer, line of credit offer, and who isn't eligible for any offers. This is usually done using a Machine Learning method called **K-Means**.

**Reinforcement Learning** This is when the algorithm learns from its own experience using reward and punishment. The easiest example is self-driving cars where there is an agent that learns from each move it makes. A positive move toward the target earns the agent a reward while a negative move away from the target earns the agent a punishment.

*Q2. Give me an example of supervised learning and another for unsupervised learning?*

Here I usually expect to hear the 3 words: **Classification**, **Regression**, and **clustering**. These are some of the most popular and basic uses for Machine Learning.

Classification and Regression mainly use supervised learning and the candidate can give an example showing how historical data is used to train the model.

For example, if someone steals your credit card and makes an online transaction. You will probably get an email or text from your bank asking to verify this transaction otherwise the bank will consider it fraud. Your bank's algorithm learned your credit card purchasing habits through your purchase history and

when an abnormal transaction was detected the bank suspected it's a fraud. This is a form of Machine Learning and probably it's decision tree **Classification**.

Another example is a car company trying to predict sales for next year based on this year's numbers and historical data, that's a form of Machine Learning and could be linear **Regression**.

**Clustering** mainly uses unsupervised learning where there is no historical data. A simple example is the spam email filter where the algorithm examines different parts of all incoming emails, group them together, then cluster the emails into spam and ham.

*Q3. You built a DL model and while training it you noticed that after a certain number of epochs the accuracy is decreasing. What's the problem and how to fix it?*

The answer should be around **overfitting**.

It seems the model is learning the exact dataset characteristics rather than capturing its features this is called overfitting the model. Probably the model is very complex in comparison to the dataset, the model is complex in terms of having many layers and neurons than needed.

Depending on the situation there are several ways to fix this overfitting model the most common are **early stopping** and **dropout regularization**.

Early stopping is what it sounds like, stop the training early once you start seeing the drop in the accuracy. Dropout regularization is dropping some outputs layers

or nodes thus the remaining nodes have different weights and have to do extra work to capture the characteristics.

***Q4. What's the difference between Bias and Variance in DL models? How to achieve a balance between them?***

This is kinda related to the previous question. The answer should include simple models that underfit, complex models that overfit, and the fact that both Bias and Variance can't be minimized at the same time.

**High Bias** means the model is simple and can't capture many features during the training phase aka underfitting model. **High Variance** means the model is complex and is not only capturing features but also learning anything but those specific training set features, this is also referred to as overfitting.

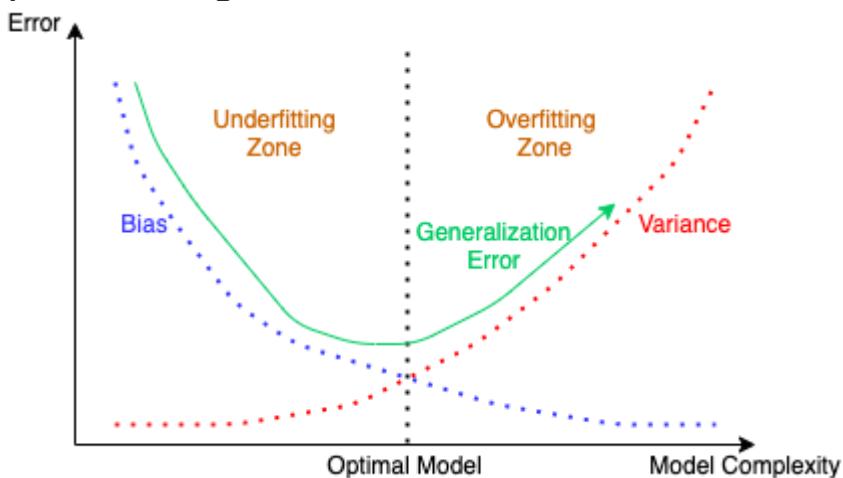


Image by Author

As you can see there is a sweet spot in the middle to balance both Bias and Variance. If your model shift to the right side then it's getting more complicated thus increasing variance and resulting in overfitting. If your model shifts to the left then it's getting too simple thus increasing bias and results in underfitting.

A good data scientist knows how to tradeoff bias and variance by tuning the model's hyperparameters thus achieving optimum model complexity.

A simple model means a small number of neurons and fewer layers while a complex model means a big number of neurons and several layers.

***Q5. What's the confusion matrix? Is it used for both supervised and unsupervised learning? What are Type 1 and Type 2 errors?***

Confusion Matrix is used to assess the performance of supervised learning models only and can't be used with unsupervised models.

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	True Positive (TP)	False Positive (FP)
	Negative (0)	False Negative (FN)	True Negative (TN)

**Confusion Matrix**

**Type 1 Error** is highlighted in yellow in the cell for False Positive (FP).

**Type 2 Error** is highlighted in yellow in the cell for False Negative (FN).

Confusion Matrix is a way to present the 4 outcomes of the model: True Positive, False Positive, False Negative, and True Negative. Recall, Precision, Accuracy, and F1 can all be calculated from the Confusion Matrix.

**Type 1 error** is when your algorithm makes a positive prediction but in fact, it's negative. For example, your algorithm predicted a patient has cancer but in fact, he doesn't.

**Type 2 error** is when your algorithm makes a negative prediction but in fact, it's positive. For example, your algorithm predicted a patient doesn't have cancer but in fact, he does.

#### *Q6. What is a model learning rate? Is a high learning rate always good?*

The learning rate is a tuning parameter that determines the step size of each iteration (epoch) during model training. The step size is how fast (or slow) you update your neurons' weights in response to an estimated error. Model weights are updated using the backpropagation error method. So, the input will flow from the input nodes of your model through the neurons to the output nodes then the error is determined and backpropagated to update the neuron's (model) weights. How fast to update those neurons' weights is the learning rate.

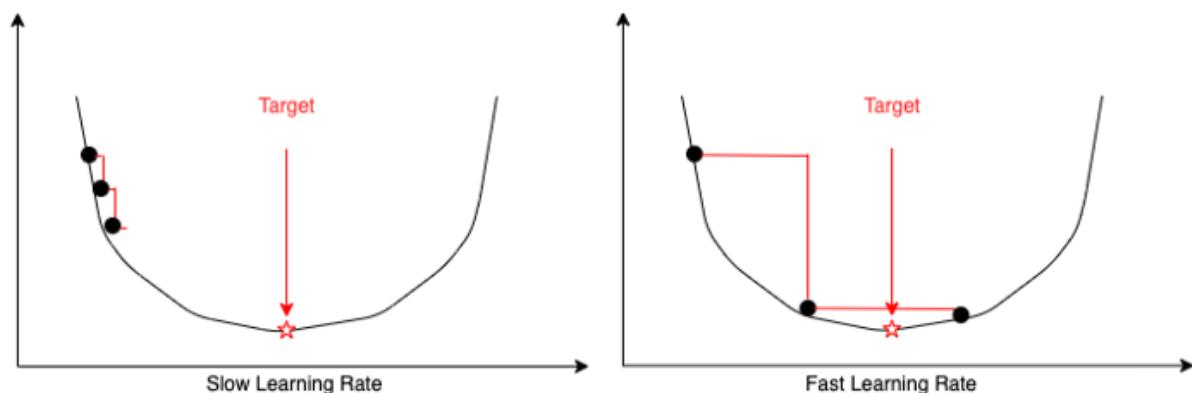


Image by Author

If the learning rate is high thus the model weights are updated fast and frequently your model will converge fast but it may overshoot the true error minima. This means a faster but erroneous model.

If the learning rate is low thus the model weights are updated slowly your model will take a long time to converge but will not overshoot the true error minima. This means a slower but more accurate model.

### ***Q7. What vanishing gradient descent?***

This question is related to the previous one. Here I expect a quick explanation of the gradient descent and how backpropagation affects it.

Think of gradient descent as the weights used to update your neural network during the backpropagation from output to input nodes. Think of Activation as the equation tied to each neuron in your model, this equation decides if this neuron should be activated or not depending on the neuron's input relevancy to the model prediction.

In some cases when you have a deep neural network with several layers and based on your choice of the activation function (along with other hyperparameters), the gradients will become very small and may vanish while backpropagating from the output to input nodes through the layers of the network. The problem here is the weights of the neurons in your model won't get updated (or get updated with very small values) thus your model won't learn (or will get minimal learning). This is a clear case of a vanishing gradient descent problem.

### ***Q8. What's the difference between KNN and K-means?***

I'm personally surprised by how many candidates confuse these two. The answer should state the fact that **KNN is a supervised** model used for classification

and **K-means** is an unsupervised model used for clustering. Then the candidate should give an example of classification and another of clustering.

***Q9. What does it mean to cross-validate a machine learning model?***

This is another easy one where the answer should include testing the model on new data that the model never seen before. The best example is when you use Scikit Learn (or any other library) to split your data into training and test set. The test set data is used to cross-validate your model after it is trained so you can assess how well your model is performing.

***Q10. How to assess your supervised machine learning model? What's Recall and Precision?***

**Precision:** This is the answer for: out of all the times the model said positive, how many were really positive. You care about precision when False Positive is important to your output.

$$Precision = \frac{TP}{TP + FP}$$

**Precision**

Let's say you're a small company and you send samples to potential customers who might buy your product. You don't want to send samples to customers that will never buy your product no matter what. The customer who gets a sample but doesn't buy your product is false positive because you predicted they will buy your product (Predicted = 1) but actually, they never will (Actual = 0). In

such cases, you want to decrease the FP as much as you can in order to have high precision.

$$Recall = \frac{TP}{TP + FN}$$

## Recall

**Recall:** This is the answer for: out of the actual positives, how many were classified correctly. You care about the recall when False Negative is important to your output. Let's take an example of your credit card, someone stole your credit card number and used it to purchase stuff online from a sketchy website that you never visit. That's clearly a fraudulent transaction but unfortunately, your banks' algorithm didn't catch it. What happened here is that your bank predicted it's not a fraud (predicted = 0) but it was actually a fraud (actual =1). In such a case, your bank should develop a fraud detection algorithm that decreases the FN thus increases the recall.

## *Q11. What's the Curse of Dimensionality and how to solve it?*

This is when your dataset has too many features thus it's hard for your model to learn and extract those features.

Two main things could happen

- More features than observations thus the risk of overfitting the model

- Too many features, observations become harder to cluster. Too many dimensions cause every observation in the dataset to appear equidistant from all others and no meaningful clusters can be formed

The main technique to solve this problem is **Principal Component Analysis (PCA)**.

PCA is an unsupervised machine learning algorithm that attempts to reduce the dimensionality (number of features) within a dataset while still retaining as much information as possible. This is done by finding a new set of features called components, which are composites of the original features that are uncorrelated with one another. They are also constrained so that the first component accounts for the largest possible variability in the data, the second component the second most variability, and so on.

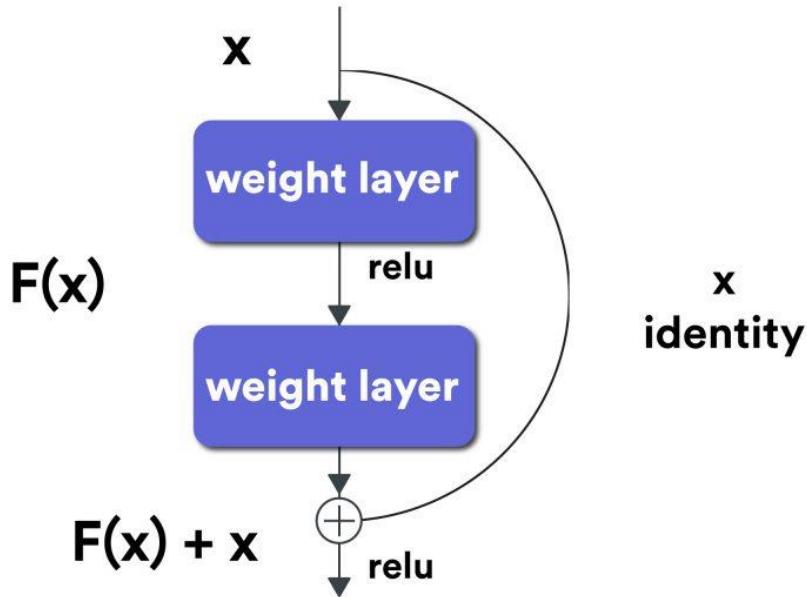
What is a ResNet and where would you use it? Is it efficient?

Among the various neural networks that are used for computer vision, ResNet (Residual Neural Networks), is one of the most popular ones. It allows us to train extremely deep neural networks which is the prime reason for its huge usage and popularity. Before the invention of this network, training extremely deep neural networks was almost impossible.

To understand why, we must look at the vanishing gradient problem which is basically an issue that arises when the gradient is back propagated to all the layers. As a large number of multiplications are performed, the size of the network keeps decreasing till it becomes extremely small and thus, the network starts performing badly. ResNet helps to counter the vanishing gradient problem.

The efficiency of this network is highly dependent on the concept of skip connections. Skip connections is a method of allowing a shortcut path through which the gradient can flow, which in effect helps counter the vanishing gradient problem.

An example of skip connection is shown below:



In general, a skip connection allows us to skip the training of a few layers. Skip connections are also called identity shortcut connections as they allow us to directly compute an identity function by just relying on these connections and not having to look at the whole network.

The skipping of these layers makes ResNet an extremely efficient network.

Dropout is an essential requirement in some neural networks. Why is it necessary?

Overfitting is probably one of the biggest problems when it comes to neural networks. This occurs when a complicated model is used for a very small dataset. It quite obviously results in very poor performance.

To counter overfitting, one of the most useful methods is dropout. Dropout uses different architectures in parallel to train neural networks. Some layers are randomly removed during training which, in effect, is called a dropout.

When a dropout takes place, some of the units are forced to fix errors that were already caused by other units. In general dropout is done on any of the layers apart from the output layer. The use case for dropout is probably all types of networks including convolutional neural networks, Long Short-Term Memory (LSTM) networks etc.

Note that both hidden as well as visible layers can be dropped. At the end of a dropout, a reduced network, with both incoming and outgoing edges removed for every dropped out node, is produced.

The probability in general of a node being dropped is 0.5. In effect, as training is not performed on all nodes, overfitting is reduced. This also leads to the model learning more generic features which can then be used to learn new data quicker and better.

Dropout generally gives better performance on large networks. Dropout generally performs better with a large learning rate but with a decay factor.

What is a sobel filter? How would you implement it in Python?

The sobel filter performs a two-dimensional spatial gradient measurement on a given image which then emphasizes regions which have high spatial frequency. In effect, this means finding edges.

In most cases, sobel filters are used to find the approximate absolute gradient magnitude for every point in a grayscale image. The operator consists of a pair of  $3 \times 3$  convolution kernels. One of these kernels is rotated by 90 degrees.

These kernels respond to edges that run horizontal or vertical with respect to the pixel grid, one kernel for each orientation. A point to note is that these kernels can be applied either separately or can be combined together to find the absolute magnitude of the gradient at every point.

The sobel operator has a large convolution kernel which ends up smoothing the image to a greater extent and thus the operator becomes less sensitive to noise. It also produces higher output values for similar edges compared to other methods.

To overcome the problem of output values from the operator overflowing the maximum allowed pixel value per image type, avoid using image types that support pixel values.

Implementation in Python

To implement it in Python, we can use the OpenCV module (can be installed from pip):

```
import cv2
```

```
import numpy as np
```

```
img = cv2.imread('your image.jpg',0)
```

```
laplacian = cv2.Laplacian(img,cv2.CV_64F)
```

```
sobelx = cv2.Sobel(img,cv2.CV_64F,1,0,ksize=5)
```

```
sobely = cv2.Sobel(img,cv2.CV_64F,0,1,ksize=5)
```

How do you add dropout to a Neural Network?

Dropout can be added very easily to a neural network. The code is as follows

```
if (dropout_flag==True):  
    first_layer *= np.random.binomial([np.ones((len(X), hidden_dim))], 1 -  
    dropout_percentage)[0] * (1.0 / (1 - dropout_percentage))
```

What is the purpose of a Boltzmann Machine?

Boltzmann machines are algorithms which are based on physics, specifically thermal equilibrium. A special and more well known case of Boltzmann machines is the Restricted boltzmann machine which is a type of boltzmann machine where there are no connections between hidden layers of the network.

The concept was coined by Geoff Hinton who most recently won the Turing award. In general, the algorithm uses the laws of thermodynamics and tries to optimise a global distribution of energy in the system.

In discrete mathematical terms, a restricted boltzmann machine can be called a symmetric bipartite graph i.e. two symmetric layers. These machines are a form of unsupervised learning which means that there are no labels provided with data. It uses stochastic binary units to reach this state.

Boltzmann machines are derived from markov state machines. A Markov State Machine is a model that can be used to represent almost any computable function. The restricted boltzmann machine can be regarded as an undirected graphical model. It is used in dimensionality reduction, collaborative filtering, learning features as well as modelling. It can also be used for classification and regression. In general, restricted boltzmann machines are composed of a two layer network which can then be extended further.

Note that these models are probabilistic in nature since each of the nodes present in the system learns low-level features from items in dataset. For example, if we take a grayscale image, each node that is responsible for the visible layer will take just one pixel value from the image.

A part of the process of creating such a machine is feature hierarchy where sequences of activations are grouped in terms of features. In thermodynamics principles, simulated annealing is a process that the machine follows to separate signal and noise.

What is the advantage of Boltzmann Machines?

The advantage of Boltzmann machines is that many of these machines can be piped together to make a system which is generally called a deep belief network.

Deep belief networks are interesting as they can be used to discover many complex features and patterns in the training data. The only disadvantage of these networks is that they are relatively slower than other models. The nodes which are present across layers are connected to each other but none of the nodes in the same layer are connected. Each of these layers compute their respective inputs.

Why do we have gates in neural networks?

To understand gates we must first understand recurrent neural networks.

Recurrent neural networks allow information to be stored as memory by means of loops. Thus, the output of a recurrent neural network is not only based on the current input but also the past inputs which are stored in memory of the network. Back propagation is done through time but in general, the truncated version of this is used for longer sequences.

Gates are generally used in networks that are dependent on time. In effect, any network which would require memory, so to speak, would benefit from the use of gates. These gates are generally used to keep track of any information that is required by the network without leading to a state of either vanishing or exploding gradients. Such a network can also preserve the error through time. Since a sense of constant error is maintained, the network can learn better.

These gated units can be considered as units with a recurrent connections. They also contain additional neurons which are gates. If you relate this process to a signal processing system, the gate is used to regulate which part of the signal passes through. A sigmoid activation function is used which means that the values taken are from 0 to 1.

An advantage of using gates is that it enables the network to either forget information that it has already learnt or to selectively ignore information either based on the state of the network or the input the gate receives.

Gates are extensively used in recurrent neural networks especially in Long Short-Term Memory (LSTM) networks. A general LSTM network will have 3 to 5 gates typically an input gate, output gate, hidden gate and activation gate.

Transfer learning is one of the most useful concepts today. Where can it be used?

Pre-trained models is probably one of the most common use cases for transfer learning.

For anyone who does not have access to huge computational power, training complex models is always a challenge. Transfer learning aims to help by both improving the performance and speeding up your network.

In layman terms, transfer learning is a technique in which a model that has already been trained to do one task is used for another without much change. This type of learning is also called multi-task learning.

Many models that are pre-trained are available online. Any of these models can be used as a starting point in the creation of the new model required. After just using the weights, the model must be refined and adapted on the required data by tuning the parameters of the model.

The general idea behind transfer learning is to transfer knowledge not data. For humans, this task is easy – we can generalise models which we have mentally created a long time ago for a different purpose. One or two samples is almost always enough. However, in the case of neural networks, huge amount of data and computational power are required.

Transfer learning should generally be used when we don't have a lot of labelled training data or if there already exists a network for the task you are trying to achieve, probably trained on a much more massive dataset. Note, however, that the input of the model must have the same size during training. Also, this works only if the tasks are fairly similar to each other and the features learned can be generalised. For example, something like learning how to recognise vehicles can probably be extended to learn how to recognise aeroplanes and helicopters.

What are some real-life examples where Transfer Learning can be used?

An example where transfer learning can be used, is photograph classification. Since it is not possible to train such huge categories of photographs on a normal machine, pre-trained weights can be used directly. If you are using your own dataset, you might need to tune the parameters before the network works accurately.

Transfer learning is very widely used with image data and language data. Since words are mapped to very high dimensional vector spaces, it becomes easy to find words with similar meaning in different languages or even in the same language.

Why are deep learning models referred to as black boxes?

Lately, the concept of deep learning being a black box has been floating around. A black box is a system whose functioning cannot be properly grasped but the output produced can be understood and utilised.

Now, since most models are mathematically sound and are created based on legit equations, how is it possible that we do not know how the system works?

First, it is almost impossible to visualize the functions that are generated by a system. Most machine learning models end up with such complex output that it is not possible for a human to make sense of it.

Second, there are networks with millions of hyperparameters. As a human, we can grasp around 10 to 15 parameters. But analysing a million of them seems out of the question.

Third and most important, it becomes very hard, if not impossible, to trace back why the system made the decisions it did. This may not sound like a huge problem to worry about but consider the case of a self driving car. If the car hits someone on the road, we need to understand why that happened and prevent it. But this isn't possible if we do not understand how the system works.

To make a deep learning model not be a black box, a new field called Explainable Artificial Intelligence or simply, Explainable AI is emerging. This field aims to be able to create intermediate results and trace back the decision making process of a system.

Why is the process of weight initialization an important step in deep learning?

Building even a small neural network is an extremely challenging task and we quite obviously do not want to get results that are less than satisfactory. The first step to making an efficient neural network is weight initialization. A negative effect of improper initialisation is that the neural network might be prohibited from learning at all.

The core objective is to prevent the explosion or vanishing of activation outputs of the layers over the course of iterations. This occurs due to multiplication of large matrices, which is one of the core mathematical operations behind neural networks. In effect, it leads to generation of matrix products which are quite large for the system to handle.

With weight initialization, a network comes to a quick convergence and also has less error. Optimisation is thus achieved in the least time possible.

What are the types of weight initialization?

There are two major types of weight initialisation:- zero initialisation and random initialisation.

**Zero initialisation:** In this process, biases and weights are initialised to 0. If the weights are set to 0, all derivatives with respect to the loss functions in the weight matrix become equal. Hence, none of the weights change during subsequent iterations. Setting the bias to 0 cancels out any effect it may have.

All hidden units become symmetric due to zero initialisation. In general, zero initialisation is not very useful or accurate for classification and thus must be avoided when any classification task is required.

**Random initialisation:** As compared to 0 initialisation, this involves setting random values for the weights. The only disadvantage is that setting very high values will increase the learning time as the sigmoid activation function maps close to 1. Likewise, if low values are set, the learning time increases as the activation function is mapped close to 0.

Setting too high or too low values thus, generally leads to the exploding or vanishing gradient problem.

New types of weight initialisation like “He initialisation” and “Xavier initialisation” have also emerged. These are based on specific equations and are not mentioned here due to their sheer complexity.

What does tuning of hyperparameters signify? Explain with examples.

A hyperparameter is just a variable which defines the structure of the network. Let's go through some hyperparameters and see the effect of tuning them.

1. Number of hidden layers – Most times, the presence or absence of a large number of hidden layers may determine the output, accuracy and training time of the neural network. Having a large number of these layers may sometimes cause an increase in accuracy.
2. Learning rate – This is simply a measure of how fast the neural network will change its parameters. A large learning rate may lead to the network not being able to converge, but might also speed up learning. On the other hand, a smaller value for learning rate will probably slow down the network but might lead to the network being able to converge.
3. Number of epochs – This is the number of times the entire training data is run through the network. Increasing the number of epochs leads to better accuracy.

4. Momentum – Momentum is a measure of how and where the network will go while taking into account all of its past actions. A proper measure of momentum can lead to a better network.
5. Batch Size – Batch size determines the number of subsamples that are inputs to the network before every parameter update.

An example of hyperparameters for a SVM model in tensorflow is shown below:

```
__init__(  
    example_id_column,  
    feature_columns,  
    weight_column_name=None,  
    model_dir=None,  
    l1_regularization=0.0,  
    l2_regularization=0.0,  
    num_loss_partitions=1,  
    kernels=None,  
    config=None,  
    feature_engineering_fn=None  
)
```

What is a Tensor in deep learning?

To represent data before being processed by neural networks, the data should have a regular structure. This data structure is called a tensor.

In general tensors are just multidimensional arrays. They are useful because they allow us to present data in an extremely higher dimensions, in an easy way. This is an important aspect since we cannot visualise data beyond a specific number of dimensions.

In deep learning, most of the data can be represented in the form of n-dimensional vectors and hence, we use tensors. There also exists a type of processing unit called the tpu or tensor processing unit.

An Example of a Tensor using Tensorflow is as follows:

```

import tensorflow as tf
strings = tf.Variable(["Hello"], tf.string)
decimals = tf.Variable([3.14159, 2.71828], tf.float32)
integers = tf.Variable([2, 3, 5, 7, 11], tf.int32)
complexNumbers = tf.Variable([12.3 - 4.85j, 7.5 - 6.23j], tf.complex64)

```

What are capsules? How are they useful in deep learning?

In deep learning, and especially while working with image data, it is essential to preserve the location of specific structures in the image. Most neutral networks fail to do so, especially those in the category of generative networks.

For example, take the case of generating faces using a generator adversarial neural network. At the end of the generation we have a face but with all features jumbled. The nose is above the eye, the mouth below the chin and so on. Quite obviously, this is not a face. But to a machine, since the concept of facial structure does not exist, thus for the system, the output is correct. To prevent such a mistake, we use capsule networks.

Capsule networks help to retain structure and location of features especially while working with images for any task. Capsule networks consist of a vector specifying the features present in the object.

A capsule might also specify many more attributes and parameters. It is obvious how useful these networks can be in preserving the structural integrity of the generated object.

- I have designed a 2 layered deep neural network for a classifier with 2 units in the hidden layer. I use linear activation functions with a sigmoid at the final layer. I use a data visualization tool and see that the decision boundary is in the shape of a sine curve. I have tried to train with 200 data points with known class labels and see that the training error is too high. What do I do ?

Increase number of units in the hidden layer  
 Increase number of hidden layers  
 Increase data set size  
 Change activation function to tanh  
 Try all of the above  
 The answer is d. When I use a ...

- What are the commonly used activation functions ? When are they used.

Ans. The commonly used loss functions are Linear :  $g(x) = x$ . This is the simplest activation function. However it cannot model complex decision boundaries. A deep network with linear ...

- I have used a 4 layered fully connected network to learn a complex classifier boundary. I have used tanh activations throughout except the last layer where I used sigmoid activation for binary classification. I train for 10K iterations with 100K examples (my data points are 3 dimensional and I initialized my weights to 0 to begin with). I see that my network is unable to fit the training data and is leading to a high training error. What is the first thing I try ?

Increase the number of training iterations Make a more complex network – increase hidden layer size Initialize weights to a random small value instead of zeros Change tanh activations to relu Ans : (3) ...

- What are the different ways of preventing over-fitting in a deep neural network ? Explain the intuition behind each

L2 norm regularization : Make the weights closer to zero prevent overfitting. L1 Norm regularization : Make the weights closer to zero and also induce sparsity in weights. Less common ...

- How is long term dependency maintained while building a language model?

Language models can be built using the following popular methods – Using n-gram language model n-gram language models make assumption for the value of n. Larger the value of n, longer the ...

- Suppose you build word vectors (embeddings) with each word vector having dimensions as the vocabulary size(V) and feature values as pPMI between corresponding words: What are the problems with this approach and how can you resolve them ?

Problems As the vocabulary size (V) is large, these vectors will be large in size. They will be sparse as a word may not have co-occurred with all possible words. Resolution Dimensionality Reduction using ...

Given the following two sentences, how do you determine if Teddy is a person or not? “Teddy bears are on sale!” and “Teddy Roosevelt was a great President!”

This is an example of Named Entity Recognition(NER) problem. One can build a sequence model such as an LSTM to perform this task. However, as shown in both the sentences above, ...

- What are the optimization algorithms typically used in a neural network ?

Gradient descent is the most commonly used training algorithm. Momentum is a common way to augment gradient descent such that gradient in each step is accumulated over past steps ...

- When are deep learning algorithms more appropriate compared to traditional machine learning algorithms?

Deep learning algorithms are capable of learning arbitrarily complex non-linear functions by using a deep enough and a wide enough network with the appropriate non-linear activation function. Traditional ML algorithms ...

- What is negative sampling when training the skip-gram model ?

Recap: Skip-Gram model is a popular algorithm to train word embeddings such as word2vec. It tries to represent each word in a large text as a lower dimensional vector in ...

- Why do you typically see overflow and underflow when implementing an ML algorithms ?

A common pre-processing step is to normalize/rescale inputs so that they are not too high or low. However, even on normalized inputs, overflows and underflows can occur: Underflow: Joint probability distribution often ...

- Can you give an example of a classifier with high bias and high variance?

High bias means the data is being underfit. The decision boundary is not usually complex enough. High variance happens due to over fitting, the decision boundary is more complex than ...

- Given a deep learning model, what are the considerations to set mini-batch size ?

The batch size is a hyper parameter. Usually people try various values to see what works best in terms of speed and accuracy. Suppose you have M training instances and ...

## Q #1) What is machine learning?

**Answer:** **Machine Learning** is a study in computer science which deals with making machines intelligent. A machine is called intelligent if it can make its own decisions.

The process of making machines learn is by providing a machine learning algorithm with training data. The output of this learning process is a trained ML model. This model artifact makes predictions on new data for which output is not known.

### Let us see a real-life example of ML: Self Driving Cars



A real-life example of machine learning is self-driving cars. With machine learning, self-driving cars exist. How does ML help Self Driven Cars?

So, the data of all the self-driving cars on the road is collected from the sensors and cameras attached to the cars which are been driven. Now, with machine learning algorithms and the collected data, the cars can learn themselves. Thus, by such training, they can perform tasks like humans.

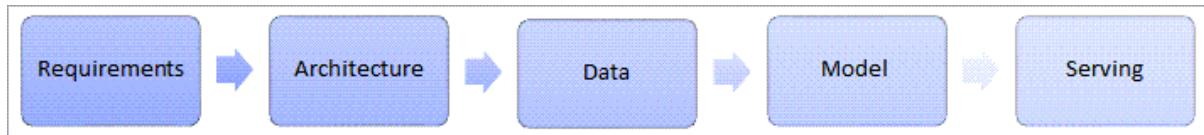
### Q #2) What is machine learning system design?

**Answer:** It is a step-by-step process to define hardware and software requirements for machine learning model design. **The aim of machine learning design is:**

- **Adaptability:** The system should be flexible enough to adapt to new changes, such as new data or changes in business features.
- **Maintainability:** The performance of the system should not degrade with time. The system should have optimal performance with any data distribution changes that occur with time.
- **Scalability:** As the system grows, it should be able to accommodate the growth. Such changes are increases in complexity, data, or traffic.
- **Reliability:** The system should provide correct results or show errors (not show garbage output) for uncertain input data and environments.

### **Q #3) What are the steps involved in Machine Learning system design?**

**Answer:**



**A) Gather Requirements:** The system designer gathers the knowledge about designing the system, such as what size of datasets will be used? Does the system need to be more accurate or faster? What is the type of hardware requirements for the model? Would there be any need to retrain the model?

**B) Identify the Metrics:** Metrics are used to measure the outcome of the model. Functional metrics measure how beneficial the model will be like click-through rate, time spent watching the video, etc.

Some non-functional metrics could be scalability, flexibility, ease to train, etc. While the model is being developed, the dataset is broken into 3 sets- training, evaluation, and test. Some offline methods, such as Mean Squared Error, F1 score, Area under ROC Curve, are also employed to measure the outcome of the model.

### **C) Architecture:**

**When planning architecture:**

- Identify the target variable. **For example**, to design a system that recommends products to users, the target variable would be product.
- Finalize a few features of the variable. In our example, some features may be user age, user hobbies.
- Machine Learning operations such as storing data, data transformation, to be performed.
- Choose a baseline model. A model which does not need to be trained and acts as a baseline for other models.
- Start working on the model. This step deals with activities such as storing logs, using analytics tools that are performed in the production.

**D) Serve the model to the users.**

**Q #4) What are the different types of Machine Learning Algorithms?**

**Answer:** These are classified as below:

- **Supervised Learning Algorithm:** Supervised learning uses labeled data to predict outcomes. The learning happens in the presence of a supervisor, just like learning performed by a small child with the help of his teacher. By using labeled data, the machines can find out their accuracy and learn by themselves.
- **Unsupervised Learning Algorithms:** Unsupervised learning happens without the help of a supervisor. The machine learning algorithms were used to cluster the unlabelled data. These algorithms find out the hidden patterns in the data without any human help.
- **Reinforcement Learning:** The algorithm learns by the feedback mechanism and past experiences. This type of learning takes the feedback from the previous step and learns from experience to decide what the best next step would be. It is an iterative process, also called Markov Decision Process. In Reinforcement Learning, the more the number of feedbacks the more accurate the system would be.

**Q #5) What are the applications of Machine Learning?**

**Answer:** Some of the most seen applications are listed as below:

**Chatbots:**



## Ecommerce:



1. **Chatbot:** These days majority of the websites have a **virtual customer service** assistant which provides automated answers to your queries based on the information present on the website. With the help of machine learning algorithms, chatbots can train themselves with the inputs and provide better answers with time.
2. **Search Engine Results:** In any Web Search Engines, say Google, as we query, it provides some results. As we click on any of the results displayed and spend some time visiting the webpage, Google can find out whether or not the query results are appropriate? With the machine learning algorithms at the backend, the search engines can refine their results.
3. **Ecommerce Shopping:** Whenever user shops online, he/she is presented with product recommendations, some options such as “Customers also bought”, “Products Bought Together”, “Other similar Products” etc. These are nothing but the recommendations provided by machine learning algorithms running behind the website, which try to make the customer experience easy and friendly.
4. **Facial Recognition:** Nowadays the mobile phones, social media platforms such as Instagram, Facebook can automatically identify and suggest tagging the person in the uploaded pic. In such cases, these platforms have ML algorithms that extract the features of the picture and match them with the profile picture of people in your friend list.
5. **Personalized Virtual Assistants such as Siri, Alexa, Bixby:** These assistants run over voice and provide appropriate information. With such assistants, we can also create personalized tasks such as “Creating To-Do List”, “Listing Grocery Items”, “Setting Up Alarm”, “Play Music or Videos”. The machine learning algorithms here capture our previous inputs and refine

their output. Each time, the machine learns by itself to provide a personalized experience.

There are numerous other applications where machine learning is used, like Email Filtering, Security Systems, Fraud Detection, etc. From the above applications, we can see how it plays a vital role in our day-to-day lives.

## **Q #6) Is there a difference between Artificial Intelligence and Machine Learning?**

**Answer:** Artificial Intelligence and Machine Learning terms are used interchangeably always, but it is not so. There is a difference between both. Before going to the difference, let us understand what Artificial Intelligence is. Artificial Intelligence is the ability of a computer machine to show human-like intelligence and perform tasks like humans. A machine competent to think, learn on its own, and make its own decisions is nothing but an artificially intelligent machine.

**Let us compare and differentiate them along with some real-life examples:**  
**Artificial Intelligence**

The art of making machines intelligent is AI

AI robots perform tasks to make the system successful rather than training and retraining

AI computers are programmed extensively

**Machine Learning**

ML is a part of AI. It is a process of learning input data without any help of programming

The machines retrain themselves for accurate reduction of error.

ML mechanism does not involve programming; it learns from data

## **Q #7) Give an example to compare Artificial Intelligence and Machine Learning?**

**Answer:**

### **Example of Artificial Intelligence:**

The most seen example of AI is Tesla Car. All the cars are connected, so if one car learns about an unnoticed sharp turn, it is updated for all cars.

Another example is Drones, nowadays used by Tech Giant, Amazon for Logistics and Transportation. The drones use programming and technology, such as navigation systems, for automated flying. Sensors and cameras are attached to drones to capture data which is used by Machine Learning algorithms.

Some uses of **AI-enabled drones** are agriculture, smart cities, etc.

### **Example of Machine Learning: Drone**

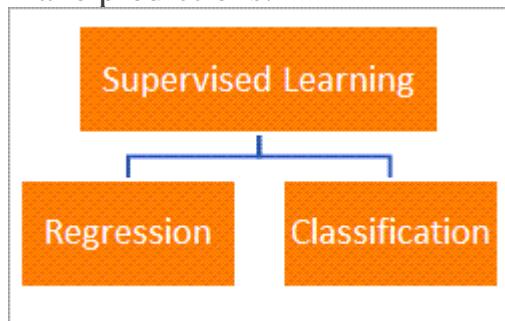
As we read above about Self-Flying Drones, the cameras and sensors attached to the drones capture images that are processed using Computer Vision. The computer vision marks objects for drones to recognize, which helps the drones to go in the right direction without colliding with obstacles.

The machine learning algorithms also learn from the captured images of the objects. The self-flying drones are also enabled by GPS navigation, due to which the destination coordinates are already fed in them. But the GPS system is not enough to avoid a collision, leading to droes crashing with the mountains or walls or trees.

Thus, there is a need to train drones. With the machine learning algorithms, the drones are fed with a large amount of data. The datasets train the drones to detect the objects and avoid such objects which may lead to a collision.

### **Q #8) What is the Classification and Regression in Machine Learning?**

**Answer:** Supervised Learning Methods are classified into Classification and Regression. Both these methods work with labeled data set and are used to make predictions.



**Classification Methods:** These methods categorize the input data into different output classes. In the Classification algorithm, the machine learns and gives the output in form of classes.

In other words, these classification methods provide an output function that maps the input data to an output class. The learned machine will categorize input data into generated output classes. As new data is fed to the machine, it will move to one of the output classes. The output classes are discrete, such as Yes/No, Long/Short.

As we know, the training sets (input data) for classification machine learning algorithms are labeled. By labeled data, we mean the input data is pre-categorized. Such as an image of fruit is labeled with fruit name or fruit description.

**Classification Methods are divided into binary classifiers and multi-class classifiers. Let us see each of them:**

- **Binary Classifier:** This type of classification has the outcome as only 2 classes.
- **Multi-Class Classifier:** In this type of classification, the outcome is more than 2 classes.

**Regression Methods:** Regression methods give the predicted output as a continuous variable like Cost, Price, Age, Salary, etc. In Regression, the machine learning algorithms predict output as continuous variables. The regression problems predict a mapping function based on the input and output variables.

### **Q #9) What are the classification and regression methods?**

**Answer:**

**Classification algorithms are as below:**

1. Decision Tree Classification
2. K Nearest Neighbours
3. Naïve Bayes
4. Support Vector Machine
5. Random Forest
6. Stochastic Gradient Descent

**Some of the Regression Methods are:**

1. Linear Regression
2. Support Vector Regression
3. Regression Tree

### **Q #10) Give an example of Classification and Regression in machine learning?**

**Answer:** Let us see a simple example to understand classification and regression.

Speed is a continuous variable, so if we must determine what is the speed of the car? – It is a Regression Problem

If the speed of the car is given, we can predict if the speed of the car moving at high speed or low speed? – It is a classification problem

### **Q #11) How to build a Machine Learning Model?**

**Answer:** The ML model is built primarily using 3 steps:

1. Choose an algorithm for the model and train it.
2. Test the model by using test data.
3. Retrain the model if there are any changes and use the model for real-time projects

### **Q #12) How to choose an appropriate algorithm to create a Machine Learning Model?**

**Answer:** To choose the most appropriate algorithm to train your machine, some steps to be followed are:

**a) Categorise the problem based on input and output:**

- **Based on Input:** If the data is labeled, we use supervised learning methods while for data that is not labeled unsupervised learning techniques are used. Reinforcement learning is used where feedback from the previous step determines the next best step to follow. Each step takes the model to reach its goal.
- **Based on output:** If the output of the problem is continuous, such as a number, regression methods are used, while if the output is a class, classification techniques are applied.

**b) Prepare the data**

The data play an important role in determining the type of algorithm to be used. Some algorithms use small sets of data while other algorithms may need tons of data. The next step would be to analyze, process, and transform the data to use for modeling.

**c) Check out the available algorithms:**

To choose an appropriate algorithm based on the availability, focus on:

- Time is taken to build the model.
- The complexity of the algorithm.
- Accuracy of the model.
- Scalability of the model.
- How much time does it take to Predict the output?
- Is the model fulfilling the business requirements?

**d) Implement the ML algorithms:**

To choose the appropriate algorithm, run the available ML algorithms on different sets of data and evaluate their performance based on set criteria. Also, we can run a single algorithm on different datasets and find out the best algorithm.

**Q #13) What are test data and training data?**

**Answer:** Training Data in Machine Learning is as important as a Machine Algorithm itself. As the name says, a training dataset is data to train the machine. The machine learns from the training data. The training data is labeled dataset. It means the output variable is mapped to one or more input variables. Test data is data used to check the accuracy of the machine. The machine output should have minimal error.

Now, how do we find out the training data and test data?

The training data and test data may be taken out from the same dataset. While training the machine, we may take out a portion of the data (training data) and pass through the model multiple times to reduce the error. After successful training, we feed the model with the remaining data (test data) to get the output.

If the predicted output variable is equal to the actual labeled output value, the model passes otherwise, we may need to retrain the machine or change the model.

#### **Q #14) What is deep learning? How is it different from Machine Learning?**

**Answer:** Deep learning is a part of the Machine learning process which uses **Artificial Neural Networks** (ANN) for making machines learn and have decision-making capabilities. The ANN corresponds to the neural system of the human brain, where all nerves are interconnected.

The neurons in the human brain correspond to the nodes in ANN. The Artificial Neural Network consists of many layers and intermediate layers between the input and output layers are called hidden layers. The Deep Learning Algorithms are like Machine Learning Algorithms except that the former contains many more layers (hidden layers) than the latter.

**Some differences between deep learning and machine learning are:**

**Deep Learning**

**Machine Learning**

Deep Learning pass the data through multiple processing layers to predict the relation between input and output variables

Machine Learning works with predefined algorithms

The output data could be of any form such as shape, sound, or image

ML algorithms output data in Numbers

Deep Learning uses far more data than ML

The data used in ML is less than Learning.

The Deep Learning Algorithms does not need human intervention

ML algorithms require the attention of data analysts to explore the data

#### **Q #15) What are the most popular algorithms used in machine learning?**

**Answer: The most common algorithms are:**

- 1. K-Nearest Neighbour:** It is a supervised algorithm used for classification and regression problems. This algorithm assumes similar points are near to each other. It works by choosing an appropriate number of examples (k) as the query. By query, we mean the item in question. **For example**, songs recommended of 5 similar songs by the system. So, k here is 5.

2. **Decision Tree:** It is a supervised learning technique mostly used for classification problems. The decision tree is structured like a tree where the nodes represent the dataset, branches show rules on data and the leaf denotes the outcome.
3. **Neural Network Algorithms:** The artificial neural network learns by both supervised, unsupervised learning. An artificial neural network consists of multiple layers, namely input, output, and hidden layers. Two of the neural network training algorithms are Gradient Descent and Back-Propagation Algorithm.
4. **Support Vector Machine:** It is a supervised learning algorithm used for classification and regression problems. In this algorithm, we divide the data points with a hyperplane. The n-dimensional data points are divided into classes where new data points can be classified. Some applications of SVM are image categorization, facial recognition.

#### **Q #16) What do you mean by Genetic Programming?**

**Answer:** Genetic Programming is a form of artificial intelligence. It copies the process of natural selection to find out the optimal result.

This process is iterative in nature where at each step of the algorithm there might be randomly mutating offspring. Only the fittest offspring are chosen to cross and reproduce in the next generation. Thus, the fitness of the algorithm improves with generations. This algorithm terminates once it reaches a pre-defined fitness value.

#### **Q #17) What is Logistic Regression?**

**Answer:** Logistic Regression is an algorithm that comes under classification type. It predicts a binary outcome that is either 0 or 1 for given input variables. The output of Logistic Regression is 0/1. The threshold value is generally taken as 0.5. By threshold value, we mean any input below 0.5 has output 0, and any value more than threshold has output 1.

#### **Q #18) What is Lazy Learning?**

**Answer:** Lazy Learning is a machine learning method where the data is not generalized until the query is made to it. In other words, such learning defers the processing until the request for information is received. An example of a Lazy learning technique is KNN, where the data is just stored. It is processed only when the query is made to it.

#### **Q #19) What is a Perceptron? How does it work?**

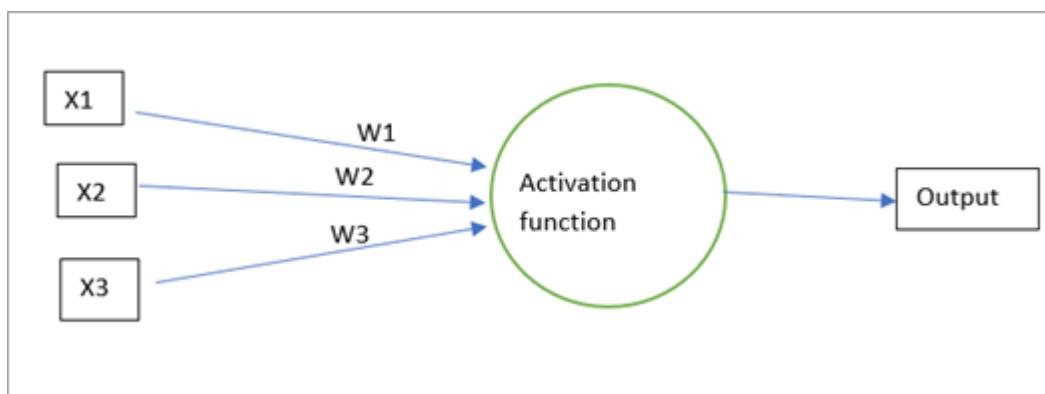
**Answer:** A **Perceptron** is the simplest ML algorithm for linear classification. A single-layer neural network is called a Perceptron. A perceptron model consists of the input layer, hidden layer, and output layer.

The input layer is connected to the hidden layer through weights and the weights are +1,0 or -1. The activation function for a single layer model is a binary step function.

The perceptron learning model is a binary classifier that classifies the inputs to output classes. The net input is fed to the activation function. If the output of the activation function is greater than the threshold value, it will return 1 otherwise, if the output is less than the threshold value, it will return 0.

**The output for the below model will be**

$$O = w_1 * x_1 + w_2 * x_2 + w_3 * x_3$$

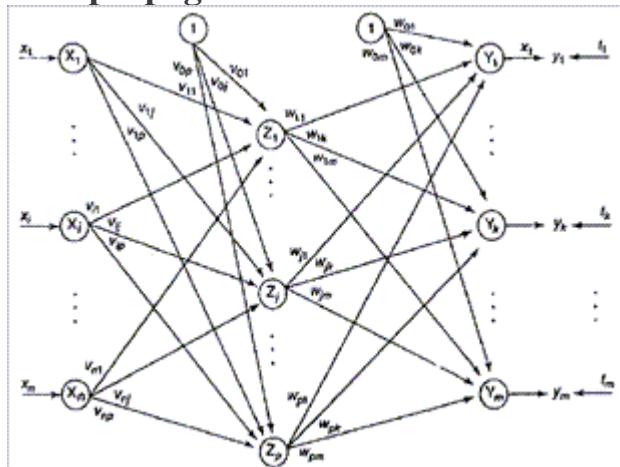


#### **Q #20) What is Backpropagation Technique?**

**Answer:** The backpropagation Method is an artificial neural network training method for machine learning. It is an iterative process for the reduction of error and makes the artificial neural network model more reliable and accurate.

The error is calculated from the previous epoch output and input. The weights of the hidden and input layers are updated. Since the error travels back towards the hidden layer, that is why it is called backpropagation of error.

#### **Backpropagation Network:**



Backpropagation Network is a multilayer perceptron network. It works in 2 phases: Feed Forward and Reverse Phase.

In the first phase, the network is fed with an input set of neurons, and the output is calculated. It is a supervised learning algorithm, therefore the target value is known. The output of the training model is compared with the target. The error is calculated and sent back for updating weight at the input and hidden layers.

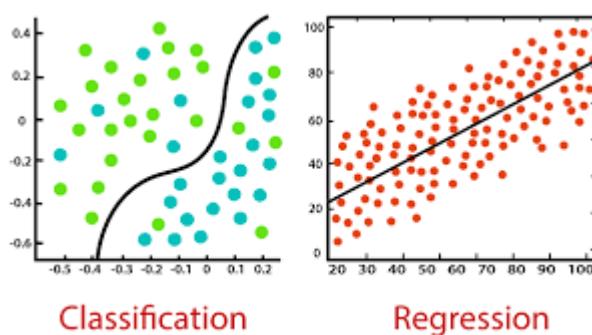
### 1. What is the difference between Supervised and Unsupervised machine learning?

Supervised learning requires training labeled data. For example, in order to do classification (a supervised learning task), you'll need to first label the data you'll use to train the model to classify data into your labeled groups. Unsupervised learning, in contrast, does not require labeling data explicitly.

### 2. What is the difference between classification and regression?

Classification is used to produce discrete results, classification is used to classify data into some specific categories .for example classifying e-mails into spam and non-spam categories.

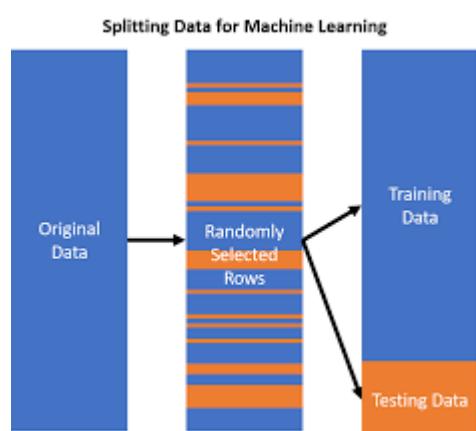
Whereas, We use regression analysis when we are dealing with continuous data, for example predicting stock prices at a certain point of time.



### 3. What is meant by ‘Training set’ and ‘Test Set’?

‘**Training set**’ is the portion of the dataset used to train the model.

‘**Testing set**’ is the portion of the dataset used to test the trained model.

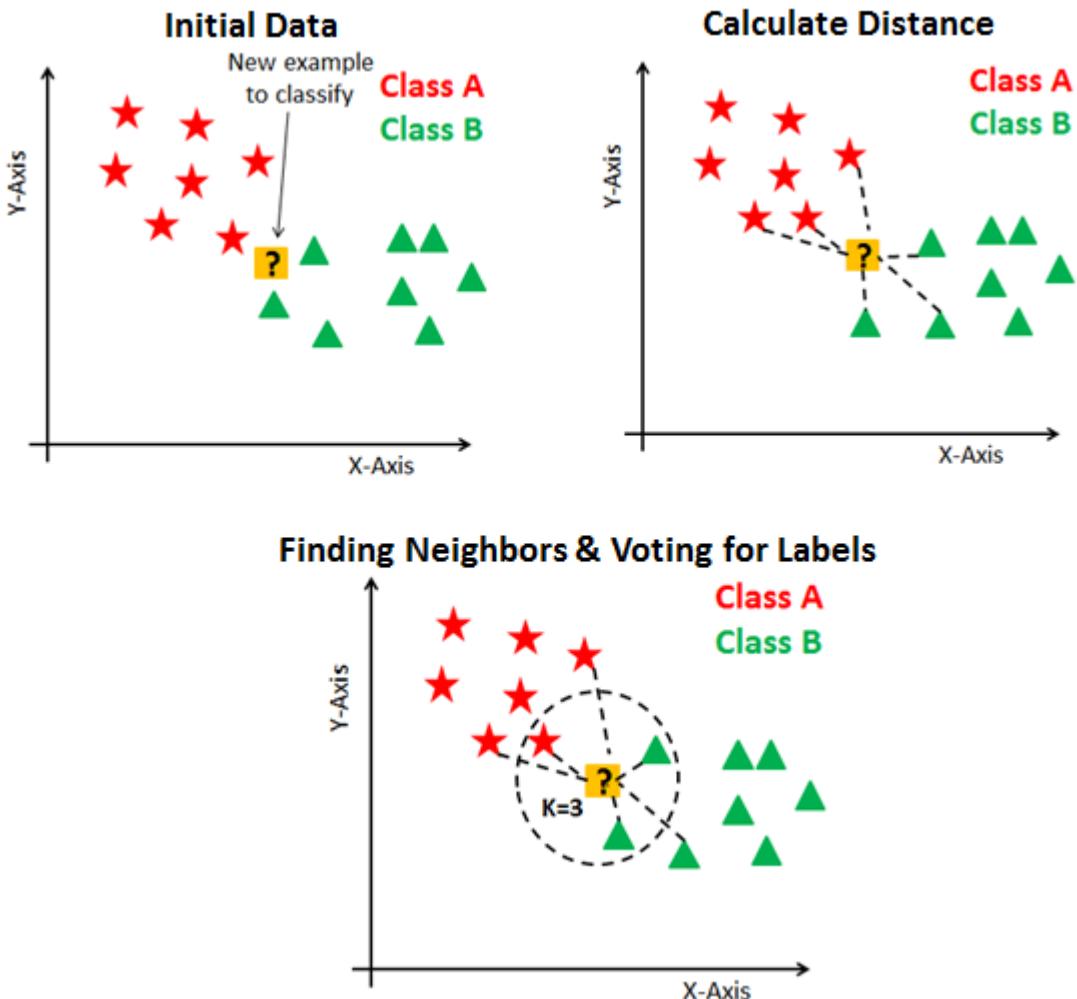


#### 4. How do you handle missing or corrupted data in a dataset?

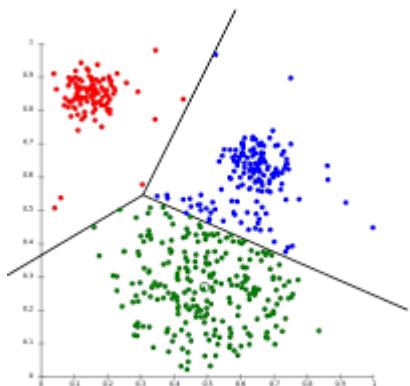
You could find missing/corrupted data in a dataset and either drop those rows or columns, or decide to replace them with another value.

In Pandas, there are two very useful methods: `isnull()` and `dropna()` that will help you find columns of data with missing or corrupted data and drop those values. If you want to fill the invalid values with a placeholder value (for example, 0), you could use the `fillna()` method.

#### 5. How is KNN different from k-means clustering



K-Nearest Neighbors is a supervised classification algorithm, while k-means clustering is an unsupervised clustering algorithm. While the mechanisms may seem similar at first, what this really means is that in order for K-Nearest Neighbors to work, you need labeled data you want to classify an unlabeled point into (thus the nearest neighbor part). K-means clustering requires only a set of unlabeled points and a threshold: the algorithm will take unlabeled points and gradually learn how to cluster them into groups by computing the mean of the distance between different points.



The critical difference here is that KNN needs labeled points and is thus supervised learning, while k-means doesn't — and is thus unsupervised learning. KNN algorithm tries to classify an unlabeled observation based on its  $k$  (can be any number) surrounding neighbors. It is also known as a lazy learner because it involves minimal training of the model. Hence, it doesn't use training data to make generalizations on the unseen data set.

#### 6. What is the main advantage of Naive Bayes?

A Naive Bayes classifier converges very quickly as compared to other models like logistic regression. As a result, we need less training data in the case of naive Bayes classifier.

#### 7. What's the difference between Type I and Type II error?

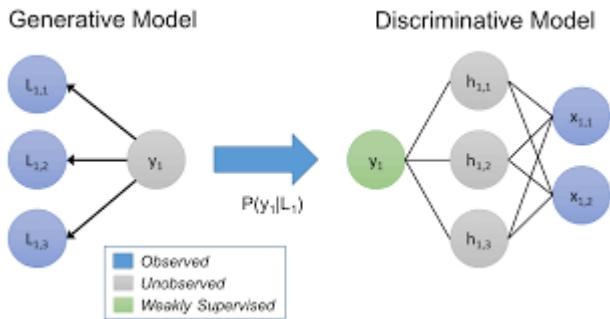
Don't think that this is a trick question! Many machine learning interview questions will be an attempt to lob basic questions at you just to make sure you're on top of your game and you've prepared all of your bases.

Type I error is a false positive, while Type II error is a false negative. Briefly stated, Type I error means claiming something has happened when it hasn't, while Type II error means that you claim nothing is happening when in fact something is.

A clever way to think about this is to think of Type I error as telling a man he is pregnant, while Type II error means you tell a pregnant woman she isn't carrying a baby.

#### 8. What's the difference between a generative and discriminative model?

A generative model will learn categories of data while a discriminative model will simply learn the distinction between different categories of data.



Discriminative models will generally outperform generative models on classification tasks.

## 9. What are Parametric models?

Parametric models are those with a finite number of parameters. To predict new data, you only need to know the parameters of the model. Examples include linear regression, logistic regression, and linear SVMs.

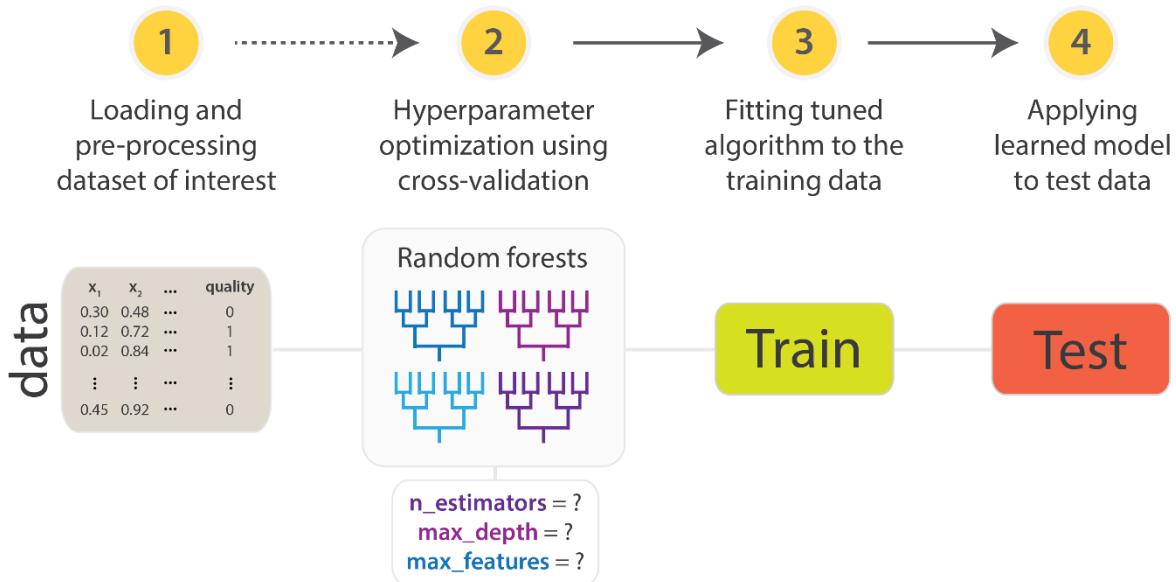
Non-parametric models are those with an unbounded number of parameters, allowing for more flexibility. To predict new data, you need to know the parameters of the model and the state of the data that has been observed. Examples include decision trees, k-nearest neighbors, and topic models using latent Dirichlet analysis.

## 10. How to ensure that your model is not overfitting?

Keep the design of the model simple. Try to reduce the noise in the model by considering fewer variables and parameters. Cross-validation techniques such as K-folds cross-validation help us keep overfitting under control. Regularization techniques such as LASSO help in avoiding overfitting by penalizing certain parameters if they are likely to cause overfitting.

## 11. How Much Data You should have to use For Training and Testing your Model?

You have to find a balance, and there's no right answer for every problem.



If your test set is too small, you'll have an unreliable estimation of model performance (performance statistic will have high variance). If your training set is too small, your actual model parameters will have a high variance.

A good rule of thumb is to use an 80/20 train/test split. Then, your train set can be further split into train/validation or into partitions for cross-validation.

12. What should you do when your model is suffering from low bias and high variance?

When the model's predicted value is very close to the actual value the condition is known as low bias. In this condition, we can use bagging algorithms like random forest regressor

13. What Is Bagging Algorithm?

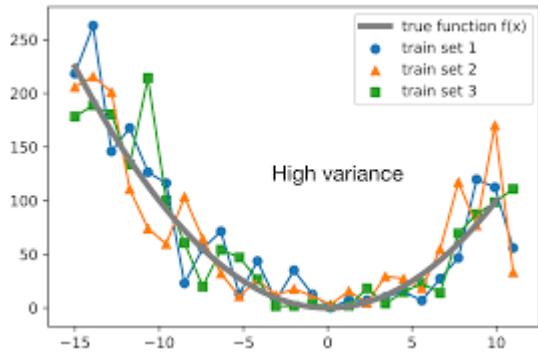
Bagging, or Bootstrap Aggregating, is an ensemble method in which the dataset is first divided into multiple subsets through resampling.

Then, each subset is used to train a model, and the final predictions are made through voting or averaging the component models.

Bagging is performed in parallel.

14. You came to know that your model is suffering from low bias and high variance. Which algorithm should you use to tackle it? Why?

Low bias occurs when the model's predicted values are near to actual values. In other words, the model becomes flexible enough to mimic the training data distribution. While it sounds like a great achievement, but not to forget, a flexible model has no generalization capabilities. It means, when this model is tested on unseen data, it gives disappointing results.



In such situations, we can use a bagging algorithm (like random forest) to tackle high variance problems. Bagging algorithms divide a data set into subsets made with repeated randomized sampling. Then, these samples are used to generate a set of models using a single learning algorithm. Later, the model predictions are combined using voting (classification) or averaging (regression).

Also, to combat high variance, we can:

1. Use the regularization techniques, where higher model coefficients get penalized, hence lowering model complexity.
2. Use top n features from the variable importance chart. Maybe, with all the variables in the data set, the algorithm is having difficulty in finding a meaningful signal.

## 15. List Down Advantages and Disadvantages of Neural Network

**Advantages:** Neural networks (specifically deep NNs) have led to performance breakthroughs for unstructured datasets such as images, audio, and video. Their incredible flexibility allows them to learn patterns that no other ML algorithm can learn.

**Disadvantages:** However, they require a large amount of training data to converge. It's also difficult to pick the right architecture, and the internal "hidden" layers are incomprehensible.

## 16. How do you think Google is training data for self-driving cars?



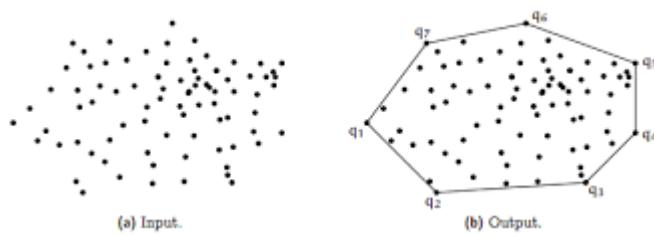
Google is currently using Recaptcha to source labeled data on storefronts and traffic signs. They are also building on training data collected by Sebastian Thrun at GoogleX — some of which was obtained by his grad students driving buggies on desert dunes!

## 17. How would you evaluate a logistic regression model?

A subsection of the question above. You have to demonstrate an understanding of what the typical goals of a logistic regression are (classification, prediction, etc.) and bring up a few examples and use cases.

## 18. What is Convex Hull?

In the case of linearly separable data, the convex hull represents the outer boundaries of the two groups of data points. Once the convex hull is created, we get maximum margin hyperplane (MMH) as a perpendicular bisector between two convex hulls.



MMH is the line which attempts to create the greatest separation between two groups.

1. How will you implement dropout during forward and backward pass?
2. What do you do if Neural network training loss/testing loss stays constant?
3. Why do RNNs have a tendency to suffer from exploding/vanishing gradient? How to prevent this? (Talk about LSTM cell which helps the

gradient from vanishing, but make sure you know why it does so. Talk about gradient clipping, and discuss whether to clip the gradient element wise, or clip the norm of the gradient.)

4. Do you know GAN, VAE, and memory augmented neural network? Can you talk about it?
5. Does using full batch means that the convergence is always better given unlimited power?
6. What is the problem with sigmoid during backpropagation? (Very small, between 0.25 and zero.)
7. Given a black box machine learning algorithm that you can't modify, how could you improve its error? (you can transform the input for example.)
8. How to find the best hyper parameters? (Random search, grid search, Bayesian search (and what it is?))
9. What is transfer learning?
10. Compare and contrast L1-loss vs. L2-loss and L1-regularization vs. L2-regularization.
11. Can you state Tom Mitchell's definition of learning and discuss T, P and E?
12. What can be different types of tasks encountered in Machine Learning?
13. What are supervised, unsupervised, semi-supervised, self-supervised, multi-instance learning, and reinforcement learning?
14. Loosely how can supervised learning be converted into unsupervised learning and vice-versa?
15. Consider linear regression. What are T, P and E?
16. Derive the normal equation for linear regression.
17. What do you mean by affine transformation? Discuss affine vs. linear transformation.
18. Discuss training error, test error, generalization error, overfitting, and underfitting.
19. Compare representational capacity vs. effective capacity of a model.
20. Discuss VC dimension.
21. What are nonparametric models? What is nonparametric learning?
22. What is an ideal model? What is Bayes error? What is/are the source(s) of Bayes error occur?
23. What is the no free lunch theorem in connection to Machine Learning?
24. What is regularization? Intuitively, what does regularization do during the optimization procedure? (expresses preferences to certain solutions, implicitly and explicitly)

- 25.What is weight decay? What is it added?
- 26.What is a hyperparameter? How do you choose which settings are going to be hyperparameters and which are going to be learnt? (either difficult to optimize or not appropriate to learn - learning model capacity by learning the degree of a polynomial or coefficient of the weight decay term always results in choosing the largest capacity until it overfits on the training set)
- 27.Why is a validation set necessary?
- 28.What are the different types of cross-validation? When do you use which one?
- 29.What are point estimation and function estimation in the context of Machine Learning? What is the relation between them?
- 30.What is the maximal likelihood of a parameter vector  $\hat{h}$ ? Where does the log come from?
- 31.Prove that for linear regression MSE can be derived from maximal likelihood by proper assumptions.
- 32.Why is maximal likelihood the preferred estimator in ML? (consistency and efficiency)
- 33.Under what conditions do the maximal likelihood estimator guarantee consistency?
- 34.What is cross-entropy of loss? (trick question)
- 35.What is the difference between an optimization problem and a Machine Learning problem?
- 36.How can a learning problem be converted into an optimization problem?
- 37.What is empirical risk minimization? Why the term empirical? Why do we rarely use it in the context of deep learning?
- 38.Name some typical loss functions used for regression. Compare and contrast. (L2-loss, L1-loss, and Huber loss)
- 39.What is the 0-1 loss function? Why can't the 0-1 loss function or classification error be used as a loss function for optimizing a deep neural network? (Non-convex, gradient is either 0 or undefined.)
- 40.Write the equation describing a dynamical system. Can you unfold it? Now, can you use this to describe a RNN? (include hidden, input, output, etc.)
- 41.What determines the size of an unfolded graph?
- 42.What are the advantages of an unfolded graph? (arbitrary sequence length, parameter sharing, and illustrate information flow during forward and backward pass)
- 43.What does the output of the hidden layer of a RNN at any arbitrary time  $t$  represent?

44. Are the output of hidden layers of RNNs lossless? If not, why?
45. RNNs are used for various tasks. From a RNNs point of view, what tasks are more demanding than others?
46. Discuss some examples of important design patterns of classical RNNs.
47. Write the equations for a classical RNN where hidden layer has recurrence. How would you define the loss in this case? What problems you might face while training it? (Discuss runtime)
48. What is backpropagation through time? (BPTT)
49. Consider a RNN that has only output to hidden layer recurrence. What are its advantages or disadvantages compared to a RNN having only hidden to hidden recurrence?
50. What is Teacher forcing? Compare and contrast with BPTT.
51. What is the disadvantage of using a strict teacher forcing technique? How to solve this?
- 52.
53. Explain the vanishing/exploding gradient phenomenon for recurrent neural networks. (use scalar and vector input scenarios)
54. Why don't we see the vanishing/exploding gradient phenomenon in feedforward networks? (weights are different in different layers - Random block initialization paper)
55. What is the key difference in architecture of LSTMs/GRUs compared to traditional RNNs? (Additive update instead of multiplicative)
56. What is the difference between LSTM and GRU?
57. Explain Gradient Clipping.
58. Adam and RMSProp adjust the size of gradients based on previously seen gradients. Do they inherently perform gradient clipping? If no, why?
59. Discuss RNNs in the context of Bayesian Machine Learning.
60. Can we do Batch Normalization in RNNs? If not, what is the alternative? (BNorm would need future data; Layer Norm)
61. What is an Autoencoder? What does it "auto-encode"?
62. What were Autoencoders traditionally used for? Why there has been a resurgence of Autoencoders for generative modeling?
63. What is recirculation?
64. What loss functions are used for Autoencoders?
65. What is a linear autoencoder? Can it be optimal (lowest training reconstruction error)? If yes, under what conditions?
66. What is the difference between Autoencoders and PCA (can also be used for reconstruction -
67. What is the impact of the size of the hidden layer in Autoencoders?

- 68.What is an undercomplete Autoencoder? Why is it typically used for?
- 69.What is a linear Autoencoder? Discuss it's equivalence with PCA. (only valid for undercomplete) Which one is better in reconstruction?
- 70.What problems might a nonlinear undercomplete Autoencoder face?
- 71.What are overcomplete Autoencoders? What problems might they face? Does the scenario change for linear overcomplete autoencoders? (identity function)
- 72.Discuss the importance of regularization in the context of Autoencoders.
- 73.Why does generative autoencoders not require regularization?
- 74.What are sparse autoencoders?
- 75.What is a denoising autoencoder? What are its advantages? How does it solve the overcomplete problem?
- 76.What is score matching? Discuss it's connections to DAEs.
- 77.Are there any connections between Autoencoders and RBMs?
- 78.What is manifold learning? How are denoising and contractive autoencoders equipped to do manifold learning?
- 79.What is a contractive autoencoder? Discuss its advantages. How does it solve the overcomplete problem?
- 80.Why is a contractive autoencoder named so? (intuitive and mathematical)
- 81.What are the practical issues with CAEs? How to tackle them?
- 82.What is a stacked autoencoder? What is a deep autoencoder? Compare and contrast.
- 83.Compare the reconstruction quality of a deep autoencoder vs. PCA.
- 84.What is predictive sparse decomposition?
- 85.Discuss some applications of Autoencoders.
- 86.What is representation learning? Why is it useful? (for a particular architecture, for other tasks, etc.)
- 87.What is the relation between Representation Learning and Deep Learning?
- 88.What is one-shot and zero-shot learning (Google's NMT)? Give examples.
- 89.What trade offs does representation learning have to consider?
- 90.What is greedy layer-wise unsupervised pretraining (GLUP)? Why greedy? Why layer-wise? Why unsupervised? Why pretraining?
- 91.What were/are the purposes of the above technique? (deep learning problem and initialization)
- 92.Why does unsupervised pretraining work?
- 93.When does unsupervised training work? Under which circumstances?

94. Why might unsupervised pretraining act as a regularizer?
95. What is the disadvantage of unsupervised pretraining compared to other forms of unsupervised learning?
96. How do you control the regularizing effect of unsupervised pretraining?
97. How to select the hyperparameters of each stage of GLUP?
98. What are deterministic algorithms? (nothing random)
99. What are Las vegas algorithms? (exact or no solution, random resources)
100.     What are deterministic approximate algorithms? (solution is not exact but the error is known)
101.    What are Monte Carlo algorithms? (approximate solution with random error)
102.    Discuss state-of-the-art attack and defense techniques for adversarial models.



















**What is Machine**

