MACHINE LEARNING EXAM REVISION

1.Explain the differences between Traditional Programming and Machine Learning?

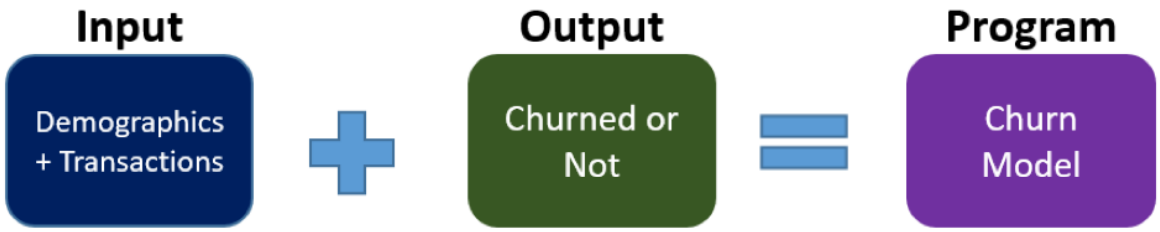
A) Traditional computer programming has been around for more than a century, with the first known computer program dating back to the mid 1800 s. **Traditional Programming** refers to any manually created program that uses input data and runs on a computer to produce the output. In **Machine Learning** programming, also known as augmented analytics, the input data and output are fed to an algorithm to create a program. This yields powerful insights that can be used to predict future outcomes.

Traditional programming is a manual process—meaning a person (programmer) creates the program. But without anyone programming the logic, one has to manually formulate or code rules.

In machine learning, on the other hand, the algorithm automatically formulates the rules from the data.

Unlike traditional programming, machine learning is an automated process. It can increase the value of your embedded analytics in many areas, including data prep, natural language interfaces, automatic outlier detection, recommendations, and causality and significance detection. All of these features help speed user insights and reduce decision bias.

For example, if you feed in customer demographics and transactions as input data and use historical customer churn rates as your output data, the algorithm will formulate a program that can predict if a customer will churn or not. That program is called a **predictive model**.



You can use this model to predict business outcomes in any situation where you have input and historical output data:

1. Identify the business question you would like to ask.
2. Identify the historical input.
3. Identify the historically observed output (i.e., data samples for when the condition is true and for when it's false).

For instance, if you want to predict who will pay the bills late, identify the input (customer demographics, bills) and the output (pay late or not), and let the machine learning use this data to create your model.

2. Describe any one of the applications of Machine Learning in detail

Machine learning is a buzzword for today's technology, and it is growing very rapidly day by day. We are using machine learning in our daily life even without knowing it such as Google Maps, Google assistant, Alexa, etc. Below are some most trending real-world applications of Machine Learning

1. Image Recognition:

Image recognition is one of the most common applications of machine learning. It is used to identify objects, persons, places, digital images, etc. The popular use case of image recognition and face detection is, **Automatic friend tagging suggestion**:

Facebook provides us a feature of auto friend tagging suggestion. Whenever we upload a photo with our Facebook friends, then we automatically get a tagging suggestion with name, and the technology behind this is machine learning's **face detection** and **recognition algorithm**.

It is based on the Facebook project named "**Deep Face**," which is responsible for face recognition and person identification in the picture.

3.Give any one use-case of Facebook using Machine Learning

Let’s take an example. It is mind-boggling how Facebook can guess the people you might be familiar with in real life using **“People You May Know”**. And they are right most of the time!!! Well, this magical effect is achieved by using Machine Learning algorithms that analyze your profile, your interests, your current friends, and also their friends and various other factors to calculate the people you might potentially know. And that’s only one aspect in which Facebook uses Machine Learning! Other aspects are the Facebook News Feed, Facial Recognition system, Targeted Advertising on your page, etc.

2. Textual Analysis

While you may believe photos are the most important on Facebook (especially your photos!), the text is equally as important. And there is a lot of text on Facebook!!! To understand and manage this text in the correct manner, Facebook uses **Deep Text** which is a text engine based on deep learning that can understand thousands of posts in a second in more than 20 languages with as much accuracy as you can!
But understanding a language-based text is not that easy as you think! To truly understand the text, Deep Text has to understand many things like **grammar, idioms, slang words, context**, etc. For example: If there is a sentence “I love Apple” in a post, then does the writer mean the fruit or the company? Most probably it is the company (Except for Android users!) but it depends on the context and DeepText has to learn this. Because of these complexities, and that too in multiple languages, DeepText uses Deep Learning, and therefore it handles labelled data much more efficiently than traditional Natural Language Processing models.

4. What are the various Learning Methods in Machine Learning? Explain any one method in detail and examples

the ten methods described offer an overview — and a foundation you can build on as you hone your machine learning knowledge and skill:

1. Regression
2. Classification
3. Clustering
4. Dimensionality Reduction

5. Ensemble Methods
6. Neural Nets and Deep Learning
7. Transfer Learning
8. Reinforcement Learning
9. Natural Language Processing
10. Word Embeddings

Another class of supervised ML, classification methods predict or explain a class value. For example, they can help predict whether or not an online customer will buy a product. The output can be yes or no: buyer or not buyer. But classification methods aren't limited to two classes. For example, a classification method could help to assess whether a given image contains a car or a truck. In this case, the output will be 3 different values: 1) the image contains a car, 2) the image contains a truck, or 3) the image contains neither a car nor a truck. The simplest classification algorithm is logistic regression — which makes it sounds like a regression method, but it's not. Logistic regression estimates the probability of an occurrence of an event based on one or more inputs. For instance, a logistic regression can take as inputs two exam scores for a student in order to estimate the probability that the student will get admitted to a particular college. Because the estimate is a probability, the output is a number between 0 and 1, where 1 represents complete certainty. For the student, if the estimated probability is greater than 0.5, then we predict that he or she will be admitted. If the estimated probability is less than 0.5, we predict the he or she will be refused. The chart below plots the scores of previous students along with whether they were admitted. Logistic regression allows us to draw a line that represents the decision boundary. Because logistic regression is the simplest classification model, it's a good place to start for classification. As you progress, you can dive into non-linear classifiers such as decision trees, random forests, support vector machines, and neural nets, among others.

5.What type of learning method is best suitable for understanding hidden patterns

Pattern recognition is the use of computer algorithms to recognize data regularities and patterns. This type of recognition can be done on various input types, such as biometric recognition, colors, image recognition, and facial recognition. It has been applied in various fields such as image analysis, computer vision, healthcare, and seismic analysis.

Pattern recognition is the use of machine learning algorithms to identify patterns. It classifies data based on statistical information or knowledge gained from patterns and their representation.In this technique, labeled training data is used to train pattern recognition systems. A label is attached to a specific input value that is used to produce a pattern-based output. In the absence of labeled data, other computer algorithms may be employed to find unknown patterns.

Features of pattern recognition

Pattern recognition has the following features:

It has great precision in recognizing patterns

It can recognize unfamiliar objects.

It can recognize objects accurately from various angles.

It can recover patterns in instances of missing data.

A pattern recognition system can discover patterns that are partly hidden.

How pattern recognition works

Pattern recognition is achieved by utilizing the concept of learning. Learning enables the pattern recognition system to be trained and to become adaptable to provide more accurate results. A section of the dataset is used for training the system while the rest is used for testing it. The training set contains images or data used for training or building the model. Training rules are used to provide the criteria for output decisions.

Training algorithms are used to match a given input data with a corresponding output decision. The algorithms and rules are then applied to facilitate training. The system uses the information collected from the data to generate results.

The testing set is used to validate the accuracy of the system. The testing data is used to check whether the accurate output is attained after the system has been trained. This data represents approximately 20% of the entire data in the pattern recognition system.

6.Why is Python widely used in Basic Data Exploration and Machine Learning?

1. Rich library ecosystem

A programming language library refers to a module that comes with a pre-written code that helps the user to use the same functionality to perform different actions. Python contains libraries that help in saving developer's time as they do not have to start from scratch.

List of some common libraries used for AI and machine learning:

- Pandas
- Scikit-learn
- Keras
- TensorFlow
- Caffe
- PyBrain

With the help of these libraries, AI and ML algorithms can be implemented more easily. These libraries are useful for data analysis, deep learning, machine learning, computer vision, and advanced computing. This helps in the faster development of the product as the developers can now resolve complex problems without rewriting code lines.

2. Flexibility

Python is a flexible language, which means that it can be used along with other programming languages to achieve the desired result. It offers an option to the developer to choose between OOPs or scripting. Also, it does not require recompilation of the

source code, making it easier to view the results. Due to its flexibility, it gives the developer a safe environment and reduces the chances of mistakes.

3. Simple and Consistent

This programming language offers concise, readable codes. As complex algorithms stand behind AI and ML, the simplicity of the language helps in developing reliable systems. Now the entire focus is on solving an ML problem instead of worrying about the technical details of the language.

Another reason which makes Python so popular is that it is an easy-to-learn programming language. Due to its easier understandability by humans, it is easier to make models for machine learning. Furthermore, many coders say that Python is more intuitive than other programming languages. It is suitable for a collaborative implementation as and when multiple developers are involved. Being a general-purpose language, it allows you to build prototypes faster so that you can test your product for machine learning.

4. Platform Independent

Platform independence of a programming language means that it can run on a variety of platforms and software architectures. The code has to be written once and it can be compiled and run on multiple platforms.

Python is easy to learn and use and scores high on versatility. It can run on any platform, be it Windows, MacOS, Linux, Unix, and more. If one wants to run the code of different platforms, packages like PyInstaller come in handy. Let’s say a coder wants to shift from one platform to another, it is far easier with Python. This saves time and money for tests on multiple platforms. As a result, the overall process becomes more convenient.

5. Huge Community and Popularity

Having strong community support helps a programming language in multiple ways, especially when it is an open-source language. Python is free, comes with useful libraries and tools, and its documentation can be accessed online. Programmers can discuss their problem statement on forums and have conversations with others to find solutions. The increasing popularity of Python was reinstated in the *Developer Survey 2020 by Stack Overflow*, which named Python as the top 5 most popular programming languages.

7. Enumerate any 6 Python Libraries that are used for Machine Learning

1. TensorFlow

TensorFlow is a free and open-source library that is used for numerical computations. The Google Brain research team developed it in 2015. It offers an exhaustive math library suitable for neural network applications and large-scale systems. The library supports probabilistic methods such as Bayesian models by providing access to several distribution functions like Bernoulli, Chi2, Gamma, and others.

2. PyTorch

PyTorch is a free and open-source library typically used for computer vision and natural language processing applications. The library was developed by Facebook's AI research group and adopted by companies such as **Microsoft, Walmart, Uber,** and **Facebook**. Moreover, PyTorch is used to build several deep learning software, such as Uber's Pyro, which is used for deep probabilistic modeling.

3. Keras

Keras is an open-source and standalone Python ML library suitable for neural network computations. Keras extends support to convolutional and recurrent neural networks, apart from standard neural nets. The library can operate over known frameworks of TensorFlow and Theano. It enables faster experimentation as the library is easy to interpret, modular, and even extensible.

4. Pandas

Pandas is primarily designed to perform data manipulation and analysis. It is known that dataset preparation is essential before the training phase. The Pandas library comes in handy in such a scenario as it provides a variety of data structures, functions, and components that help in data extraction and preparation tasks. Data preparation refers to data organization, wherein various methods are employed to a group, combine, reshape, and filter out different datasets.

5. Matplotlib

Similar to Pandas library, Matplolib is not a machine learning heavy library. It is typically used for data visualization where developers can derive insights from the visualized data patterns. Some of its modules, such as **Pyplot,** provide functionalities to control line styles, manage fonts, and others while plotting 2D graphs and plots.The features offered by Matplotlib are in line with those of **MATLAB,** and all the Python packages are freely available in this library.

6. Theano:

The Theano Python library manipulates, evaluates, and optimizes mathematical models. It was developed by the Montreal Institute for Learning Algorithms (MILA), University of Montreal, in 2007, to define and execute mathematical expressions. The library uses **multi-dimensional arrays** to process these expressions

Q8) Explain the "Exploratory Data Analysis" with respect to requirements in Machine Learning

Exploratory data analysis (EDA) is used by data scientists to analyze and investigate data sets and summarize their main characteristics, often employing data visualization methods. It helps determine how best to manipulate data sources to get the answers you need, making it easier for data scientists to discover patterns, spot anomalies, test a hypothesis, or check assumptions. EDA is primarily used to see what data can reveal beyond the formal modeling or hypothesis testing task and provides a provides a better understanding of data set variables and the relationships between them. It can also help determine if the statistical techniques you are considering for data analysis are appropriate. Originally developed by American mathematician John Tukey in the 1970s, EDA techniques continue to be a widely used method in the data discovery process today. The main purpose of EDA is to help look at data before making any assumptions. It can help identify obvious errors, as well as better understand patterns within the data, detect outliers or anomalous events, find interesting relations among the variables.Data scientists can use exploratory analysis to ensure the results they produce are valid and applicable to any desired business outcomes and goals. EDA also helps stakeholders by confirming they are asking the right questions. EDA can help answer questions about standard deviations, categorical variables, and confidence intervals. Once EDA is complete and insights are drawn, its features can then be used for more sophisticated data analysis or modeling, including machine learning.

Q9) Would data cleaning and pre-processing improve performance of a system? Justify.

Q10)How does the reward and punishment type of learning work?

Q11) What are the different steps involved in Machine Learning process?

1 - Data Collection

- The quantity & quality of your data dictate how accurate our model is
- The outcome of this step is generally a representation of data (Guo simplifies to specifying a table) which we will use for training
- Using pre-collected data, by way of datasets from Kaggle, UCI, etc., still fits into this step

2 - Data Preparation

- Wrangle data and prepare it for training
- Clean that which may require it (remove duplicates, correct errors, deal with missing values, normalization, data type conversions, etc.)
- Randomize data, which erases the effects of the particular order in which we collected and/or otherwise prepared our data
- Visualize data to help detect relevant relationships between variables or class imbalances (bias alert!), or perform other exploratory analysis
- Split into training and evaluation sets

3 - Choose a Model

- Different algorithms are for different tasks; choose the right one

4 - Train the Model

- The goal of training is to answer a question or make a prediction correctly as often as possible

- Linear regression example: algorithm would need to learn values for *m* (or *W*) and *b* (*x* is input, *y* is output)
- Each iteration of process is a training step

5 - Evaluate the Model

- Uses some metric or combination of metrics to "measure" objective performance of model
- Test the model against previously unseen data
- This unseen data is meant to be somewhat representative of model performance in the real world, but still helps tune the model (as opposed to test data, which does not)
- Good train/eval split? 80/20, 70/30, or similar, depending on domain, data availability, dataset particulars, etc.

6 - Parameter Tuning

- This step refers to *hyperparameter* tuning, which is an "artform" as opposed to a science
- Tune model parameters for improved performance
- Simple model hyperparameters may include: number of training steps, learning rate, initialization values and distribution, etc.

7 - Make Predictions

- Using further (test set) data which have, until this point, been withheld from the model (and for which class labels are known), are used to test the model; a better approximation of how the model will perform in the real world

Q12) How are Datasets and Databases different?

A dataset is a structured collection of data generally associated with a unique body of work. A database is an organized collection of data stored as multiple datasets.

Q13) Differentiate between Nominal, Ordinal, Discrete and Continuous types of data?

Nominal Data

Nominal Data is used to label variables without any order or quantitative value. The colour of hair can be considered nominal data, as one colour can't be compared with another colour. The name "nominal" comes from the Latin name "nomen," which means "name." With the help of nominal data, we can't do any numerical tasks or can't give any order to sort the data. These data don't have any meaningful order; their values are distributed to distinct categories.

Ordinal Data

Ordinal data have natural ordering where a number is present in some kind of order by their position on the scale. These data are used for observation like customer satisfaction, happiness, etc., but we can't do any arithmetical tasks on them. The ordinal data is qualitative data for which their values have some kind of relative position. These kinds of data can be considered as "in-between" the qualitative data and quantitative data. The ordinal data only shows the sequences and cannot use for statistical analysis. Compared to the nominal data, ordinal data have some kind of order that is not present in nominal data.

Discrete Data

The term discrete means distinct or separate. The discrete data contain the values that fall under integers or whole numbers. The total number of students in a class is an example of discrete data. These data can't be broken into decimal or fraction values.The discrete data are countable and have finite values; their subdivision is not possible. These data are represented mainly by a bar graph, number line, or frequency table.

Continuous Data

Continuous data are in the form of fractional numbers. It can be the version of an android phone, the height of a person, the length of an object, etc. Continuous data represents information that can be divided into smaller levels. The continuous variable can take any value within a range. The key difference between discrete and continuous data is that

discrete data contains the integer or whole number. Still, continuous data stores the fractional numbers to record different types of data such as temperature, height, width, time, speed, etc.

14. What is the significance of “Random States” in train_test_split?

The random state hyperparameter in the train_test_split() function controls the shuffling process. With random_state=None , we get different train and test sets across different executions and the shuffling process is out of control. With random_state=0 , we get the same train and test sets across different executions.

15. Is there any relation between the final accuracies and the train-test-split ratio?

The train-test split procedure is used to estimate the performance of machine learning algorithms when they are used to make predictions on data not used to train the model. It is a fast and easy procedure to perform, the results of which allow you to compare the performance of machine learning algorithms for your predictive modelling problem. Although simple to use and interpret, there are times when the procedure should not be used, such as when you have a small dataset and situations where additional configuration is required, such as when it is used for classification and the dataset is not balanced. The idea of “sufficiently large” is specific to each predictive modeling problem. It means that there is enough data to split the dataset into train and test datasets and each of the train and test datasets are suitable representations of the problem domain. This requires that the original dataset is also a suitable representation of the problem domain.A suitable representation of the problem domain means that there are enough records to cover all common cases and most uncommon cases in the domain. This might mean combinations of input variables observed in practice. It might require thousands, hundreds of thousands, or millions of examples.Conversely, the train-test procedure is not appropriate when the dataset available is small. The reason is that when the dataset is split into train and test sets, there will not be enough data in the training dataset for the model to learn an effective mapping of inputs to outputs. There will also not be enough data in the test set to effectively evaluate the model performance. The estimated performance could be overly optimistic (good) or overly pessimistic (bad).If you have insufficient data, then a suitable alternate model evaluation procedure would be the k-fold cross-validation procedure. In addition to dataset size, another reason to use the train-test split evaluation procedure is computational efficiency.

Some models are very costly to train, and in that case, repeated evaluation used in other procedures is intractable. An example might be deep neural network models. In this case, the train-test procedure is commonly used. Alternately, a project may have an efficient model and a vast dataset, although may require an estimate of model performance quickly. Again, the train-test split procedure is approached in this situation. Samples from the original training dataset are split into the two subsets using random selection. This is to ensure that the train and test datasets are representative of the original dataset.

Q16) Differentiate between Feature Extraction, Feature Transformation and Feature Engineering

Feature engineering refers to a process of selecting
and **transforming** variables/features in your dataset when creating a **predictive model** using machine learning.
Therefore you have to extract the features from the **raw dataset** you have collected before training your data in machine learning algorithms.
Otherwise, it will be hard to gain good insights in your data.
Feature engineering has two goals:

Preparing the proper input dataset, compatible with the machine learning algorithm requirements.

Improving the **performance** of machine learning models.

Feature selection is the process where you automatically or manually select the features that contribute the most to your prediction variable or output.

Having irrelevant features in your data can *decrease* the accuracy of the machine learning models.
The top reasons to use feature selection are:

- It enables the machine learning algorithm to train faster.
- It reduces the complexity of a model and makes it easier to interpret.
- It improves the accuracy of a model if the right subset is chosen.
- It reduces overfitting.

Feature extraction is a part of the dimensionality reduction process, in which, an initial set of the raw data is divided and reduced to more manageable groups. So when you want to process it will be easier. The most important characteristic of these large data sets is that they have a large number of variables. These variables require a lot of computing resources to process. So Feature extraction helps to get the best feature from those big data sets by selecting and combining variables into features, thus, effectively reducing the amount of data. These features are easy to process, but still able to describe the actual data set with accuracy and originality.

Q17) How do we identify whether a given dataset could be used for Regression or Classification?

Regression and Classification algorithms are Supervised Learning algorithms. Both the algorithms are used for prediction in Machine learning and work with the labeled datasets. But the difference between both is how they are used for different machine learning problems.The main difference between Regression and Classification algorithms that Regression algorithms are used to **predict the continuous** values such as price, salary, age, etc. and Classification algorithms are used to **predict/Classify the discrete values** such as Male or Female, True or False, Spam or Not Spam, etc.

Classification: Classification is a process of finding a function which helps in dividing the dataset into classes based on different parameters. In Classification, a computer program is trained on the training dataset and based on that training, it categorizes the data into different classes.

ypes of ML Classification Algorithms:

Classification Algorithms can be further divided into the following types:

- Logistic Regression
- K-Nearest Neighbours
- Support Vector Machines
- Kernel SVM
- Naïve Bayes
- Decision Tree Classification
- Random Forest Classification

Regression:

Regression is a process of finding the correlations between dependent and independent variables. It helps in predicting the continuous variables such as prediction of **Market Trends**, prediction of House prices, etc.

The task of the Regression algorithm is to find the mapping function to map the input variable(x) to the continuous output variable(y).

Example: Suppose we want to do weather forecasting, so for this, we will use the Regression algorithm. In weather prediction, the model is trained on the past data, and once the training is completed, it can easily predict the weather for future days.

- Simple Linear Regression
- Multiple Linear Regression
- Polynomial Regression
- Support Vector Regression
- Decision Tree Regression
- Random Forest Regression

Q18) What is an "algorithm/model parameter" and a "hyperparameter"?

A model parameter is a configuration variable that is internal to the model and whose value can be estimated from data.

- They are required by the model when making predictions.
- They 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.

Parameters are key to machine learning algorithms. They are the part of the model that is learned from historical training data.

In classical machine learning literature, we may think of the model as the hypothesis and the parameters as the tailoring of the hypothesis to a specific set of data.

Often model parameters are estimated using an optimization algorithm, which is a type of efficient search through possible parameter values.

- **Statistics**: In statistics, you may assume a distribution for a variable, such as a Gaussian distribution. Two parameters of the Gaussian distribution are the mean (*mu*)

and the standard deviation (*sigma*). This holds in machine learning, where these parameters may be estimated from data and used as part of a predictive model.

- **Programming**: In programming, you may pass a parameter to a function. In this case, a parameter is a function argument that could have one of a range of values. In machine learning, the specific model you are using is the function and requires parameters in order to make a prediction on new data.

Whether a model has a fixed or variable number of parameters determines whether it may be referred to as "*parametric*" or "*nonparametric*".

A hyperparameter is a configuration variable that is external to the model. It is defined manually before the training of the model with the historical dataset. Its value cannot be evaluated from the datasets.

It is not possible to know the best value of the hyperparameter. But we can use rules of thumb or select a value with trial and error for our system.

Hyperparameters affect the speed and accuracy of the learning process of the model. Different systems need different numbers of hyper-parameters. Simple systems might not need any hyperparameters at all.

Learning rate, the value of K in k-nearest neighbors, and batch-size are examples of hyper-parameters.

Q19) What is the need of Hyperparameter tuning?

Hyperparameter tuning (or hyperparameter optimization) is the process of determining the right combination of hyperparameters that maximizes the model performance. It works by running multiple trials in a single training process. Each trial is a complete execution of your training application with values for your chosen hyperparameters, set within the limits you specify. This process once finished will give you the set of hyperparameter values that are best suited for the model to give optimal results.

Needless to say, It is an important step in any Machine Learning project since it leads to optimal results for a model. If you wish to see it in action, here's a research paper that talks about the importance of hyperparameter optimization by experimenting on datasets.

Q20) Explain bias and variance

Bias is a phenomenon that skews the result of an algorithm in favor or against an idea. Bias is considered a systematic error that occurs in the machine learning model itself due to incorrect assumptions in the ML process. Technically, we can define bias as the error between average model prediction and the ground truth. Moreover, it describes how well the model matches the training data set:

A model with a higher bias would not match the data set closely. A low bias model will closely match the training data set.

Variance refers to the changes in the model when using different portions of the training data set. Simply stated, variance is the variability in the model prediction—how much the ML function can adjust depending on the given data set. Variance comes from highly complex models with a large number of features.

- Models with high bias will have low variance.
- Models with high variance will have a low bias.

All these contribute to the flexibility of the model. For instance, a model that does not match a data set with a high bias will create an inflexible model with a low variance that results in a suboptimal machine learning model.

Q21) Contrast between Overfitting, Underfitting and an Ideal Situation

Overfitting and Underfitting are the two main problems that occur in machine learning and degrade the performance of the machine learning models. The main goal of each machine learning model is **to generalize well**. Here **generalization** defines the ability of an ML model to provide a suitable output by adapting the given set of unknown input. It means after providing training on the dataset, it can produce reliable and accurate output. Hence, the underfitting and overfitting are the two terms that need to be checked for the performance of the model and whether the model is generalizing well or not.

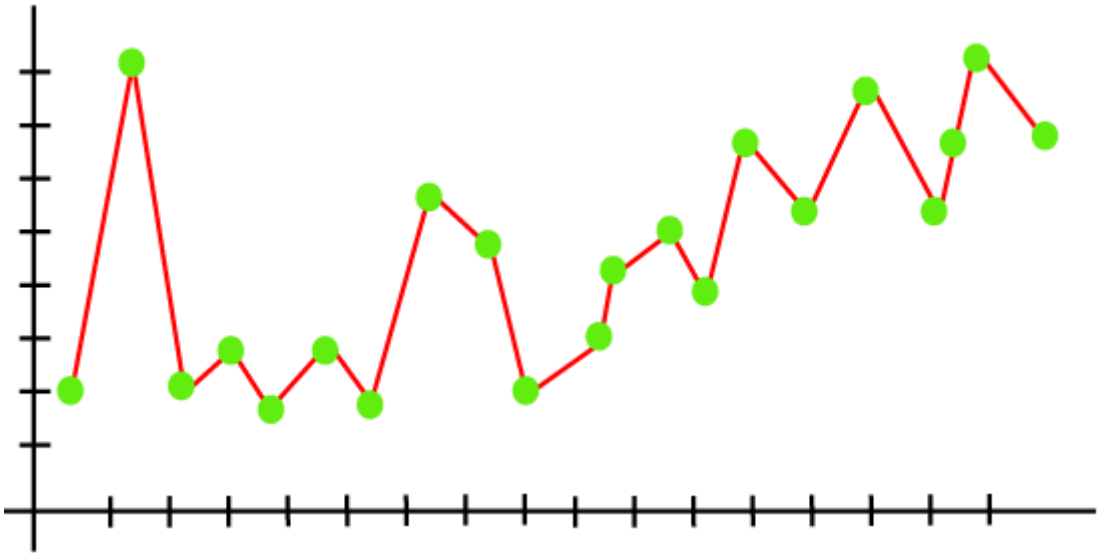
Overfitting occurs when our ML model tries to cover all the data points or more than the required data points present in the given dataset. Because of this, the model starts caching noise and inaccurate values present in the dataset, and all these factors reduce the efficiency and accuracy of the model. The overfitted model has **low bias** and **high variance.**

The chances of occurrence of overfitting increase as much we provide training to our model. It means the more we train our model, the more chances of occurring the overfitted model.

Overfitting is the main problem that occurs in supervised learning

.

Example: The concept of the overfitting can be understood by the below graph of the linear regression output:



As we can see from the above graph, the model tries to cover all the data points present in the scatter plot. It may look efficient, but in reality, it is not so. Because the goal of the regression model to find the best fit line, but here we have not got any best fit, so, it will generate the prediction errors.

Underfitting:

Underfitting occurs when our machine learning model is not able to capture the underlying trend of the data. To avoid the overfitting in the model, the fed of training data can be stopped at an early stage, due to which the model may not learn enough from the training data. As a result, it may fail to find the best fit of the dominant trend in the data .In the case of underfitting, the model is not able to learn enough from the training data, and hence it reduces the accuracy and produces unreliable predictions. An underfitted model has high bias and low variance.

Q22) What is the role of ML engineers to control Bias-Variance trade-off? One of the practices to reduce Bias can be to change the methodologies being used to create models. So for Models having High bias, the correct method will be not to use a Linear model if features and target variables of data do not in fact have a Linear Relationship. The correct way to tackle high variance will be to train the data using multiple models. Ensemble learning methods also help to leverage both weak and strong learners in the model to improve the model prediction. Most best-suited solutions in Machine Learning make use of Ensemble Learning. Another way can be to ensure that the training data is diverse and represents all possible groups or outcomes. In the presence of an imbalanced dataset, using weighted or penalized models can be considered as an alternative. The most common source of error is generally due to the training dataset not being diverse in nature and hence the model being created not having enough training data to clearly identify or differentiate between a problem. Introducing more data increases the data to noise ratio which may help reduce the variance of the present model. When the model is fed with more data, it has shown to be able to come up with a better understanding of the data which is then also applied to newly introduced data points.

Q23) What are the various evaluation metrics that you are familiar with?

1. Confusion Matrix
2. F1 Score
3. Gain and Lift Charts
4. Kolmogorov Smirnov Chart
5. AUC – ROC
6. Log Loss
7. Gini Coefficient
8. Concordant – Discordant Ratio
9. Root Mean Squared Error
10. Cross Validation

Q24) Explain the acronyms - TP, FP, FN, and TN

FN	False negative – number of persons *with* disease who have a negative test result with the assay in question.
FP	False positive – number of persons *without* disease who have a positive test result with the assay in question.
TN	True negative – number of persons without disease who have a negative test result with the assay in question.
TP	True positive – number of persons with disease who have a positive test result with the assay in question.

Q25) Enumerate the needs of different types of performance measures over Accuracy Score

Q26) What is F1 Score? How do we use it?

The **F1-score** combines the precision and recall of a classifier into a single metric by taking their harmonic mean. It is primarily used to compare the performance of two classifiers. Suppose that classifier A has a higher recall, and classifier B has higher precision. In this case, the F1-scores for both the classifiers can be used to determine which one produces better results.

The **F1-score** of a classification model is calculated as follows:

\frac{2(P * R)}{P+R}*P*+*R*2(*P*∗*R*)

How could we use lines (linear separators) for classifying multiple classes?R*R* = the recall of the classification model

It is **primarily used to compare the performance of two classifiers**. Suppose that classifier A has a higher recall, and classifier B has higher precision.

Q27) How could we use lines (linear separators) for classifying multiple classes?

Q28) What does “fitting the curve” means? How is it different in Logistic Regression and in Linear regression?