❖ Feature

1. **What is Feature in ML?**
   Features are the variables or inputs that are used to make predictions or decisions in a model. For example, in a spam email detection system, the features could include the frequency of certain words, the presence of specific email addresses, or the length of the email. In image recognition, features could be pixel values or higher-level representations extracted by convolutional neural networks.

2. **Why Feature is Important in Machine Learning?**
   Features in machine learning is very important, being building a block of datasets, the quality of the features in your dataset has major impact on the quality of the insights you will get while using the dataset for machine learning.

3. **Different types of Features.**
   Numerical Features. Categorical Features. Ordinal Features. Binary Features. Text Features. Image Features.

4. **What is Variable? Explain types of variables.**
   In machine learning, a variable refers to a characteristic or attribute of the data that is used as input or output in a model.
   Normally variable is two types:
   - Numerical
     Numerical have two different types:
     - Discreate: A variable which values are integers or terms/words is called discrete. Example- Number of legs of a robot, Number of people/cat/cars etc.
     - Continuous: A variable that may contain any real/rational value within some range is called continuous. Example: Temperature, Voltage, Current etc.
   - Categorical
     Categorical also have two types:
     - Nominal: A variable which values can be meaningful ordered. Example- Day of week, Low, medium, high, Age groups: 18-29,30-48,65+ etc.
     - Ordinal: A variable which values cannot be meaningful ordered. Example- Mobile network provider, Error codes etc.

5. **Write down difference between parameters and hyperparameters.**
   - Parameters allow the model to learn the rules from the data. A model parameter is a configuration variable that is internal to the model and whose value can be estimated from data. In other word, something that is learnt during machine learning process.
   - Hyperparameters control how the model is training. In other word, manually specified i.e. how many layers want to use or skip.

6. **What is feature generation?**
   Feature generation, also known as feature engineering, is the process of creating new features or transforming existing features in a dataset to enhance the performance of a machine learning model.

7. **Write some feature generation techniques.**
   - Polynomial Features:
     Introducing polynomial terms to capture nonlinear relationships. For instance, adding squared or cubed versions of a feature to account for quadratic or cubic dependencies.
   - Binning or Discretization:
     Grouping continuous variables into discrete bins or categories. This can help capture nonlinear relationships and patterns in the data.
   - Encoding Categorical Variables:
     Converting categorical variables into a numerical format that can be used by machine learning algorithms. Common encoding methods include one-hot encoding and label encoding.

8. **What is Visual Feature?**
   Visual features refer to specific characteristics or patterns within an image or visual data that are used for analysis, interpretation, or processing. These features can be extracted from images to enable machines, particularly in the field of computer vision, to understand and make sense of visual information.

9. **Write down some methods of Visual features.**
   - Sobel Operator:
     Sobel Operator is like a magic filter for finding edges in pictures. The Sobel Operator is like a special tool to help machines see where the edges are in pictures, which is super useful for understanding and analysing images in computer vision.
   - SIFT (Scale-Invariant Feature Transform):
     SIFT is like a superhero algorithm that makes computers really good at recognizing and matching things in pictures, no matter the size or rotation.
   - SURF (Speeded-Up Robust Features):
     SURF (Speeded-Up Robust Features) is like a faster, efficient superhero for recognizing and matching things in pictures.

10. **What is redundancy in ML?**
    In the context of machine learning (ML), redundancy refers to the presence of unnecessary or duplicate information in the data.The goal in managing redundancy in machine learning is to create a more efficient and effective model by focusing on the most relevant and informative features and data points. Techniques such as feature selection, dimensionality reduction (e.g., Principal Component Analysis), and careful data preprocessing are commonly used to address redundancy issues in ML workflows.

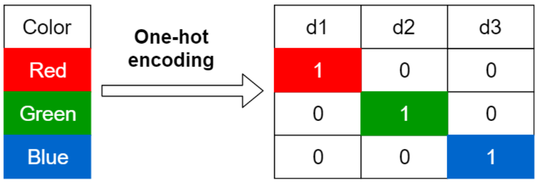
11. **What is skewness?**
    Skewness is a measure that tells us how lopsided or asymmetrical a set of numbers is.In essence, skewness helps us understand the shape of a group of numbers. It's a way to spot if the data is leaning to one side or if it's nicely balanced.

12. **What is kurtosis?**
    Kurtosis is a measure that tells us how much the tails of a distribution differ from a normal distribution. Kurtosis helps us understand how much a distribution deviates from the "normal" bell-shaped curve, especially in terms of extreme values in the tails.

13. **How does Fourier Series work in ML?**
Fourier Series is like a magical tool for understanding and working with repeating patterns or signals. In machine learning, this can be handy for understanding and working with signals or time-series data. It helps find important patterns (features) related to frequencies.

14. **How does Pearson correlation coefficient help?**
Pearson correlation coefficient can help to identify useful and redundant features.

15. **What is one-hot encoding?**
One-hot encoding is a technique used to represent categorical variables with a binary matrix, where each category or label is represented by a unique binary code.



Color
Red
Green
Blue

d1	d2	d3
1	0	0
0	1	0
0	0	1

16. **Why One-Hot Encoding is Useful?**
- Numerical Representation:
  Many machine learning algorithms require numerical input. One-hot encoding allows the representation of categorical variables in a way that can be processed by these algorithms.
- Avoiding Ordinal Interpretation:
  When dealing with categorical variables with no inherent ordinal relationship (no meaningful order), one-hot encoding prevents the algorithm from interpreting the categories as having numerical significance.
- Preventing Bias:
  Avoids introducing unintentional bias by representing categorical variables numerically. For instance, using ordinal encoding might imply an order that doesn't exist in the data.
- Handling Multiple Categories:
  One-hot encoding is flexible and can handle variables with multiple categories, providing a binary representation for each category.

17. **What is dimensionality reduction?**
Dimensionality reduction is a process of reducing the number of input variables, features, or dimensions in a dataset while preserving its important characteristics. The main goal of dimensionality reduction is to simplify the dataset, making it more manageable, computationally efficient, and often improving the performance of machine learning models.

18. **Types of dimensional reduction.**
There are two main types of dimensionality reduction techniques:
- Feature Selection
- Feature Extraction:
  Feature extraction methods transform the original features into a new set of features, typically of lower dimensionality.

19. **Explain some Common Dimensionality Reduction Techniques.**
    - Principal Component Analysis (PCA):
      Principal Component Analysis (PCA) is a dimensionality reduction technique commonly used in data analysis and machine learning. Its primary goal is to transform high-dimensional data into a lower-dimensional representation, capturing the most important information while minimizing the loss of variance.

      PCA is widely applied in various fields, including image processing, signal processing, finance, and biology, for tasks such as feature extraction, data compression, and visualization.
    - Linear Discriminant Analysis (LDA):
      LDA is a supervised dimensionality reduction technique that aims to maximize the separation between classes in a classification task.
    - t-Distributed Stochastic Neighbor Embedding (t-SNE):
      t-SNE is a non-linear technique commonly used for visualization. It emphasizes preserving the pairwise similarities between data points, making it suitable for visualizing high-dimensional data in two or three dimensions.

20. **What is feature selection?**
    Feature selection is the process of choosing a subset of relevant features from the original set of features in a dataset to build a machine learning model. The goal of feature selection is to improve the model's performance, reduce computational complexity, and enhance interpretability by focusing on the most informative features while discarding irrelevant or redundant ones.

21. **Why are we interested in reducing the number of features?**
    Reducing the number of features, also known as feature reduction or feature selection, is often motivated by several practical considerations in machine learning and data analysis. In summary, reducing the number of features is crucial for building more efficient, interpretable, and generalizable machine learning models.

22. **Why feature selection is important?**
    - Curse of Dimensionality:
      As the number of features increases, the complexity of the model also increases. This can lead to overfitting and reduced model generalization. Feature selection helps mitigate the curse of dimensionality by selecting only the most relevant features.
    - Improved Model Performance:
      Irrelevant or redundant features can introduce noise into the model, making it harder for the algorithm to identify meaningful patterns. Feature selection can lead to more accurate and efficient models.
    - Reduced Training Time:
      Removing irrelevant features can significantly reduce the computational resources required to train a model. This is especially important for large datasets with a high number of features.

23. **What is the objective of Correlation-based Feature Selection (CFS)?**
    The primary objective of Correlation-based Feature Selection is to identify a subset of features that are highly correlated with the target variable while minimizing redundancy, ultimately leading to improved model performance and interpretability.

24. **Feature selection categories.**
    - Filter Methods:
      These methods evaluate the relevance of features based on statistical measures or information-theoretic criteria before the model training process.
    - Wrapper Methods:
      Wrapper methods evaluate the performance of different feature subsets by using a specific machine learning algorithm. Recursive Feature Elimination (RFE) is an example of a wrapper method.
    - Embedded Methods:
      These methods incorporate feature selection within the model training process itself. For example, regularization techniques like LASSO (L1 regularization) in linear regression or tree-based algorithms with built-in feature importance scores.

25. **Explain Greedy Hill-Climbing Heuristic in the terms of feature selection in ML.**
    The Greedy Hill-Climbing Heuristic is like climbing a hill, step by step, always choosing the path that takes you higher. In the context of feature selection, the Greedy Hill-Climbing Heuristic helps identify a subset of features that contributes the most to the predictive power of the model. However, it may not guarantee finding the globally optimal subset, and the result may depend on the starting point and the specific criterion used for evaluation.

26. **What is the main objective of Principal Component Analysis (PCA)?**
    Detect correlations in data and reveal the internal structure of data.

❖ Preprocessing

27. **What is Preprocessing? Explain data preprocessing in the context of Machine Learning.**
    Data preprocessing is a crucial step in the machine learning pipeline that involves cleaning and transforming raw data into a format suitable for training a machine learning model. The goal of data preprocessing is to enhance the quality of the data, address any issues or inconsistencies, and prepare it for effective model training and evaluation. This process significantly influences the performance and reliability of machine learning models.

28. **What’s types of problems we can face with raw data?**
    - Missing values for certain observations.
    - Nominal variables must be transformed.
    - Infrequent categories.
    - Outliers
    - Noise

29. **Tools and methods used for preprocessing.**
    There are several different tools and methods used for preprocessing data, including the following:
    - **sampling**, which selects a representative subset from a large population of data.
    - **transformation**, which manipulates raw data to produce a single input.
    - **denoising**, which removes noise from data.
    - **imputation**, which synthesizes statistically relevant data for missing values.
    - **normalization**, which organizes data for more efficient access. And

- **feature extraction**, which pulls out a relevant feature subset that is significant in a particular context.

30. Why we need to do normalization?

The goal of normalization is to transform features to be on a similar scale. This improves the performance and training stability of the model.

31. What are the key steps in data preprocessing?

- Data profiling.
- Data cleansing
- Data reduction
- Data transformation
- Data enrichment
- Data validation

32. How is data preprocessing used?

Data preprocessing is used to ensure that raw data is in a suitable form for machine learning model training. It involves a combination of cleaning, transformation, and enhancement techniques to address various challenges associated with real-world data.

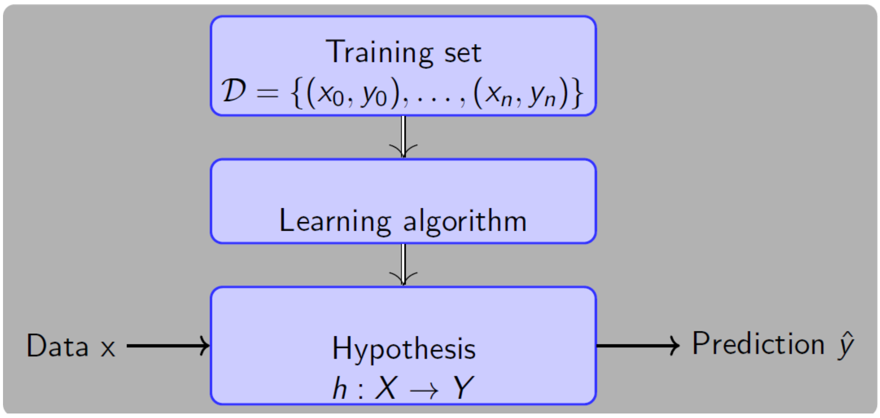
33. Applications in robotics:

- Determine relevant sensors (reduce sensors and save money)
- Reduce dimensionality of data
- Improve data processing.

❖ Supervised Learning

- Supervised machine learning requires labelled input and output data during the training phase of the machine learning model lifecycle by a data scientist in the preparation phase, before being used to train and test the model. Once the model has learned the relationship between the input and output data, it can be used to classify new and unseen datasets and predict outcomes.

  The reason it is called supervised machine learning is because at least part of this approach requires human oversight. Most available data is unlabelled, raw data. Human interaction is generally required to accurately label ready for supervised learning.



Here, for example x part is a image and y part is labelling. Suppose x is the

Image (x) of the dog (y). We have data set/training set. Learn it and compare with given data/input (Data x) and provide result (prediction y) which is the basic idea of supervised learning.

If Y is discrete domain (like label for image) is called classification.

If Y is continuous (like real number) is called regression.

If we don't have the y in training is called unsupervised learning.

34. Supervised machine learning is often used for-

- Classifying different file types such as images, documents, or written words.
- Forecasting future trends and outcomes through learning patterns in training data.

❖ **Unsupervised Learning**

35. What is unsupervised learning?

Unsupervised machine learning is the training of models on row and unlabelled training data. It is often used to identify patterns and trends in raw datasets, or to cluster similar data into a specific number of groups. It's also often an approach used in the early exploratory phase to better understand the datasets.

As the name suggests, unsupervised machine learning is more of a hands-off approach compared to supervised machine learning. A human will set model hyperparameters such as the number of cluster points, but the model will process huge arrays of data effectively and without human oversight.

36. Unsupervised learning is often used for-

- Cluster datasets in similarities between features or segment data.
- Understand relationship between different data points such as automated music recommendation.
- Perform initial data analysis.

37. Difference between supervised and unsupervised learning.

The main difference between supervised and unsupervised learning is the problem the model is deployed to solve. Supervised machine learning is generally used to classify data or make predictions, whereas unsupervised learning is generally used to understand relationships within datasets.

❖ Clustering

38. What is clustering? Why do we need it?

Normally cluster is an unsupervised data. The idea of cluster is i.e. I have data/group of data, or I want to make a group of data and I want to learn from it by applying algorithms.

Clustering is a technique in machine learning and data analysis that involves grouping a set of objects or data points into clusters, where items within the same cluster are more like each other than to those in other clusters. The primary goal of clustering is to discover inherent structures or patterns within the data based on similarities, without requiring pre-defined labels or categories.

For robotics, first must make a cluster then analysis. In robotics, there is no leveling, so robot must create level depends on cluster. For computer vision, colouring of cluster is important.

Clustering is used to identify groups of similar objects in datasets with two or more variable quantities.

39. Types of clusters.

- Centroid-based Clustering

  Centroid-based clustering is a machine learning technique that partitions a dataset into groups of similar data points, known as clusters. This technique uses centroids, the center of each cluster, to minimize the sum of the distances between the data points and their corresponding cluster centroids. As a result, the data points are as close as possible to the center of the cluster and the inter-cluster distance is maximized. k-means is the most widely used centroid-based clustering algorithm.

- Density-based Clustering

  Density-based clustering is an unsupervised machine learning algorithm that groups similar data points in a dataset based on their density. The algorithm identifies core points with a minimum number of neighbouring points within a specified distance (known as the epsilon radius). It expands clusters by connecting these core points to their neighbouring points until the density falls below a certain threshold. Points that do not include any cluster are considered outliers or noise.

- Distribution-based Clustering

  Distribution-Based Clustering is a clustering model in which we will fit the data on the probability of it belonging to the same distribution. This clustering approach assumes data is composed of distributions, such as Gaussian, binomial, etc.

- Hierarchical Clustering

  Hierarchical clustering is a method of cluster analysis in data mining that creates a hierarchical representation of the clusters in a dataset.

40. Write down clustering methods.

- Gaussian Mixture Models (GMM) is a probabilistic model that assumes that the data is generated by a mixture of several Gaussian distributions. In other words, it assumes that the dataset is a combination of multiple underlying Gaussian distributions, each characterized by its mean and covariance.

  GMM is like trying to figure out the different patterns in your data by modelling it as a mix of bells. It iteratively refines its guesses until it finds the best combination of bells that explains the data. This probabilistic approach allows for more flexibility in capturing complex cluster shapes and sizes.

- DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a clustering algorithm that groups together data points that are close to each other based on a density criterion. Unlike K-means, DBSCAN doesn't require the number of clusters to be specified in advance, and it can identify clusters of arbitrary shapes.

  DBSCAN finds groups of individuals who are densely packed together, considering each person's personal space. It doesn't need to know in advance how many groups there are, and it's good at dealing with outliers or loners.

  - Pros:
    - Number of clusters not defined beforehand as in k-means or GMM.
    - Arbitrary shapes can be modelled.
    - Noise is modelled.
  - Cons:

- Dependence on the distance measure.
- Large differences in densities cannot be modelled.

41. **What is Expectation-Maximisation and how it works?**

Expectation-Maximization (EM) is a general framework for finding maximum likelihood estimates of parameters in models with latent (present and capable of emerging or developing but not now visible) variables. It is often used for statistical models where some of the variables cannot be observed directly.

EM is like solving a puzzle where some pieces are missing. You make a guess about the missing pieces, use that guess to figure out more about the puzzle, and then update your guess based on what you've learned. Keep doing this until you have a good overall picture of the puzzle.

One common application of EM is in the training of Gaussian Mixture Models (GMMs), where each cluster is modelled as a Gaussian distribution. The E-step involves calculating the probability that each data point belongs to each cluster, while the M-step updates the parameters of these Gaussian distributions based on these probabilities.

42. **What is k-means clustering? Explain k-means clustering.**

K-means is a clustering algorithm in machine learning that aims to partition a dataset into k distinct (presenting a clear unmistakable impression) and non-overlapping clusters. The algorithm operates iteratively, assigning data points to clusters and updating cluster centroids in a way that minimizes the sum of squared distances within each cluster.

Key features of K-means include its unsupervised nature, as it does not rely on labelled data, and its application in tasks like customer segmentation, image compression, and data grouping.

43. **How k-means clustering work?**

Start by Picking a Number: Decide how many groups you want to find in your data (let's call this number k).

Randomly Guess Centers: Imagine k points in your data as the initial centers of your groups. These are like the first guesses.

Assign Points: For each data point, figure out which one of these k guessed centers is closest to it. Assign the point to that group.

Find New Centers: Recalculate the center of each group based on the points assigned to it. Now, these centers are better guesses.

Repeat: Keep repeating the assignment and recalculation steps until the groups don't change much. This is when you've found stable centers.

The final centers represent the centers of your groups, and each data point belongs to the group whose center is closest.

44. **Write down pros and cons of k-means.**

Pros:

- Simple:easy to understand and to implement
- Efficient: O(n) time complexity for n data-points.
- Guaranteed convergence because it is a coordinate descent algorithm.

- Cons:
  - The user needs to specify k.
  - Sensitivity to outliers.

- Final clustering is only locally optimal.

45. **What is Density Estimation?**
    Density estimation is the construction of an estimate of an unknown probability density function (pdf). This estimate is constructed based on a set of observed data points that are thought of as a random sample drawn from this pdf.

46. **What is Kernel Density Estimation (KDE) and how it works?**
    Kernel Density Estimation (KDE) is a non-parametric way to estimate the probability density function of a random variable. It is often used for visualizing the distribution of data points or generating smooth representations of probability distributions.
    KDE is like placing smooth bells on each data point and adding them up to create a smooth curve that gives you an idea of how your data is distributed. It's a way to make sense of your data's overall shape.

47. **Clustering applications in robotics.**
    - Cluster different substrates a robot has walked over.
    - Cluster different objects a robot can interact with into different categories.

❖ Regression

48. **What is regression? What is the goal of regression?**
    Regression is a statistical technique and a type of supervised learning algorithm in machine learning that analyses the relationship between a dependent variable (target) and one or more independent variables (features). The goal of regression is to model and quantify the relationship between variables, allowing for the prediction of the dependent variable based on new input data.

49. **Types of regression model.**
    - Linear Regression
      Linear regression is a statistical method used to model the relationship between a dependent variable and one or more independent variables by fitting a linear equation to observed data. The goal of linear regression is to find the best-fitting straight line (or hyperplane, in the case of multiple independent variables) that minimizes the sum of the squared differences between the observed and predicted values of the dependent variable.

    - Logistic Regression
      Logistic regression is a statistical method used for binary classification problems, where the dependent variable is dichotomous (having only two possible outcomes). Despite its name, logistic regression is a classification algorithm, not a regression algorithm, as it is used for predicting the probability of an observation belonging to one of the two classes.
      In logistic regression, the logistic function (also called the sigmoid function) is used to model the relationship between the independent variables and the probability of the event occurring. The logistic function is an S-shaped curve that maps any real-valued number to a value between 0 and 1.
        - Pro:
            - Can be seen as single-layer neural network used with sigmoidal.
            - Output can be interpreted as probability.
            - Weights indicate which features are important.

- Cons:
    - For an n-dimensional feature space, this model requires adaption of n parameters.
    - Requires large training sets the more explanatory variables to not overfit.
    - Not online feasible.

- Ridge Regression
  Ridge regression, also known as **Tikhonov regularization** or **L2 regularization**, is a linear regression technique that adds a regularization term to the ordinary least squares (OLS) method. The primary purpose of ridge regression is to prevent overfitting and address multicollinearity in multiple linear regression models.

50. What is sum of squared errors (SSE)?
The Sum of Squared Errors (SSE) is a metric used to evaluate the performance of a regression model. SSE is calculated by taking the sum of the squared differences between each predicted value and its corresponding actual value.

51. What is Gaussian Process Regression?
Gaussian process regression (GPR) models are nonparametric kernel-based probabilistic models.

52. Explain Polynomial model.
A polynomial model is a type of regression model where the relationship between the independent variable (input) and the dependent variable (output) is modelled as an nth-degree polynomial. In simpler terms, instead of fitting a straight line (linear model), a polynomial model uses a polynomial function to capture more complex patterns in the data.

53. Explain the difference between hard and soft margins?
- The hard margin SVM aims to find a decision boundary (hyperplane) that perfectly separates the classes without allowing any misclassified points. It assumes that the data is linearly separable, meaning a clear gap exists between the two classes.
- The soft margin SVM allows for some misclassification. It seeks a balance between maximizing the margin and tolerating a certain degree of classification errors. It is more flexible and can handle datasets where the classes are not perfectly separable.

54. Applications in Robotics.
- Learning a model of the dynamics of a robot
- Modelling of sensors and actuators: Detect failures and wear off.

❖ Classification

55. What is classification? What is the basic algorithm?
Classification is a supervised machine learning method where the model tries to predict the correct label of a given input data.
Algorithm: First compare each unlabelled instance with all instances in the training set. Second find the instance from the training set that matches the unlabelled one best, i.e. find the nearest neighbour. Third, lathe the unlabelled instance with the class of the nearest neighbour.

56. Types of learners in machine learning classification.

There are two types of learners in machine learning classification: lazy and eager learners.

- Eager learners are machine learning algorithms that first build a model from the training dataset before making any prediction on future datasets. They spend more time during the training process because of their eagerness to have a better generalization during the training from learning the weights, but they require less time to make predictions.

  Most machine learning algorithms are eager learners, and below are some examples:
  - Logistic Regression.
  - Support Vector Machine
  - Decision Trees
  - Artificial Neural Networks
- Lazy learners or instance-based learners, on the other hand, do not create any model immediately from the training data, and this is where the lazy aspect comes from. They just memorize the training data, and each time there is a need to make a prediction, they search for the nearest neighbour from the whole training data, which makes them very slow during prediction. Some examples of this kind are:
  - K-Nearest Neighbour.
  - Case-based reasoning.

57. What is k-nearest neighbour? How it works?

K-Nearest Neighbors (KNN) is a simple and widely used supervised machine learning algorithm for classification and regression tasks. It's a type of instance-based learning, where the algorithm makes predictions based on the majority class (for classification) or average value (for regression) of the K-nearest data points in the feature space.

- For classification:
  - Imagine a Map: Think of your dataset as points on a map, where each point has coordinates representing its features.
  - Pick a Point: When a new point (data to be classified) comes in, identify its location on the map based on its features.
  - Look at Neighbors: Find the K-nearest points (neighbors) to the new point. "K" is a predefined number.
  - Majority Vote: For classification, let the neighbors vote on the category of the new point. The category with the most votes becomes the predicted category for the new point.

- For Regression:
  - Imagine a Number Line: Picture your dataset as points on a number line, where each point has a numerical value (target).
  - Pick a Point: When a new point (data for regression) comes in, identify its location on the number line based on its features.
  - Look at Neighbours: Find the K-nearest points (neighbours) to the new point.
  - Average Value: For regression, take the average value of the numerical targets of the K-nearest points. This average becomes the predicted value for the new point.

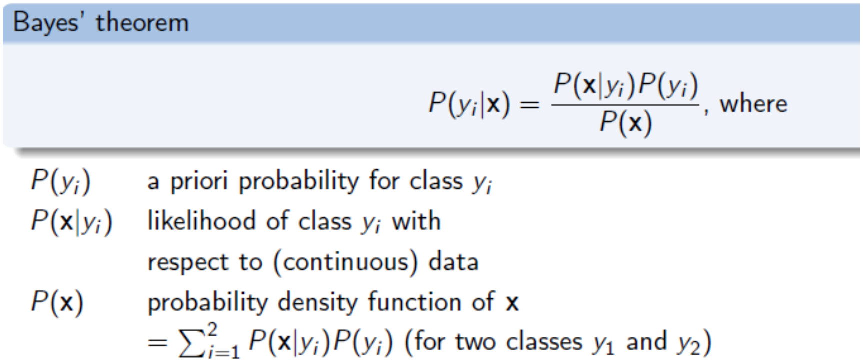
58. What are Distance Measures?

Distance measure/matrix are used in supervised and unsupervised learning to calculate similarity in data points. They improve the performance, heather that's for classification tasks or clustering. The four types to distance matrices are:

i) Euclidean Distance ii) Manhattan Distance iii) Minkowski Distance iv) Hamming Distance

59. Explain Bayes' theorem. How can we use it for classification?

Bayes' Theorem is a fundamental concept in probability theory that describes the probability of an event based on prior knowledge of conditions that might be related to the event.



Bayes' theorem

$$P(y_i|\mathbf{x}) = \frac{P(\mathbf{x}|y_i)P(y_i)}{P(\mathbf{x})}, \text{ where}$$

$P(y_i)$ a priori probability for class y_i

$P(\mathbf{x}|y_i)$ likelihood of class y_i with respect to (continuous) data

$P(\mathbf{x})$ probability density function of $\mathbf{x}$ $= \sum_{i=1}^{2} P(\mathbf{x}|y_i)P(y_i)$ (for two classes y_1 and y_2)

60. Explain decision trees and working principle.

A decision tree is a non-parametric supervised learning algorithm, which is utilized for both classification and regression tasks.

Working principle:

- Choosing Features:
  The algorithm selects the best feature to split the data based on certain criteria (commonly information gain or Gini impurity for classification and mean squared error for regression).
- Splitting Data:
  The data is split into subsets based on the chosen feature. Each subset goes to a different branch of the tree.
- Repeating the Process:
  The process is repeated for each subset at the internal nodes, selecting the best feature to split the data until a stopping criterion is met (e.g., a maximum depth is reached or a minimum number of samples in a node).
- Leaf Nodes and Predictions:
  The final leaf nodes contain the predictions or classifications based on the majority class in the case of classification or the mean value in the case of regression.

Example:

- Classification:
  - Imagine classifying fruits as "Apple" or "Orange" based on features like colour and shape. The decision tree might start by asking whether the colour is red. If yes, it may further ask about the shape to distinguish between apples and other red fruits.
- Regression:
  - Suppose predicting house prices based on features like size and number of bedrooms. The decision tree may start by asking whether the size is above a certain threshold, leading to further questions about bedrooms, ultimately reaching a leaf node with the predicted house price.

61. **How can we build decision trees efficiently?**

Building decision trees efficiently involves optimizing the construction process to create accurate and effective models without unnecessary complexity. Here are some key strategies for building decision trees efficiently:

- Use Feature Importance:
  Start by identifying the most informative features. Features that better discriminate between classes or have higher predictive power should be considered first during tree construction.
- Parallelization:
  Decision tree algorithms can be parallelized, especially when working with large datasets. Take advantage of parallel processing capabilities to speed up tree construction.
- Cross-Validation:
  Use cross-validation to assess the performance of the tree and fine-tune hyperparameters. This helps in selecting the optimal configuration for building an efficient and accurate decision tree.
- Feature Selection:
  Consider feature selection techniques to identify and use only the most relevant features. This reduces the dimensionality of the problem and can improve efficiency.

  By combining these strategies, you can build decision trees efficiently while ensuring they generalize well to new, unseen data. The goal is to strike a balance between model complexity and predictive accuracy.

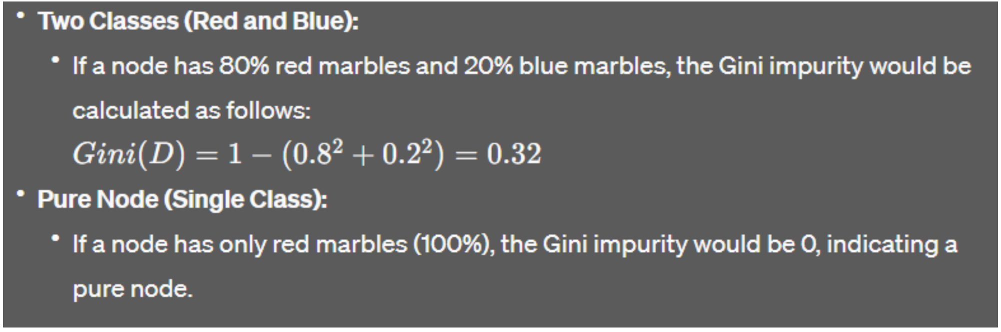
62. **Write down a key step to make a good Decision tree.**

- Select root node.
- If node is pure or no more attributes to test.
- Then stop.
- Else construct subset of nodes using the rest of attributes. Choose the attribute which is associated with the purest node.
- Recurse on each subnode.

63. **Explain Gini impurity for classification.**

Gini Impurity is a measure used in decision tree algorithms for classification tasks. It quantifies the likelihood of an incorrect classification of a randomly chosen element in the dataset if it were randomly labeled according to the distribution of labels in the subset. In simpler terms, Gini impurity measures the degree of disorder or impurity in a set of labels.

For example:



- **Two Classes (Red and Blue):**
  - If a node has 80% red marbles and 20% blue marbles, the Gini impurity would be calculated as follows:
    $Gini(D) = 1 - (0.8^2 + 0.2^2) = 0.32$
- **Pure Node (Single Class):**
  - If a node has only red marbles (100%), the Gini impurity would be 0, indicating a pure node.

64. **What is NP hard problem?**

An NP-hard problem is a problem for which we cannot prove that a polynomial time solution exists.

65. What is Decision tree overfitting/pruning and how to solve the problem?

Applying greedy heuristics such as entropy has a drawback. The resulting decision trees overfit to training data and might not generalize well.

Solution:

Stop growing a branch during building the decision tree (pre-pruning) or cut branches off (post-pruning) according to some heuristics:

- Identify irrelevant attributes:
    - The resulting subsets have same proportions as the original set.
    - Information gain is close to zero, i.e., below a threshold.
    - Apply a statistical significance test.
- Reduced Error pruning:

  Each decision node is a candidate. Use a test set and compute the expected misclassification from bottom up and compare I to a leaf that sets the class label to the most common class label. Prune where the error is highest and repeat the process until there is no error reduction possible.

66. Write Pros and Cons of Decision tree.

- Pro:
    - Classification without much computation, though training can be expensive.
    - Can handle continuous and categorical features and variables, though less appropriate to estimate a continuous variable.
    - Performs well in many situations.
- Con:
    - Tend to overfit on training data.
    - Not online feasible, new data might change splitting attribute.
    - Large trees with large data sets.

67. What is Random Forest and importance? Explain it.

The main idea of random forest is to create a large set of decision trees in a randomized fashion to obtain more robust decision.

Steps:

- First step to choose random set of features during each attribute splitting in a tree (This can give more importance to correlated features).
- Second step, create random training sets by selecting m samples with replacement- also called bagging.
- Third step, repeat both repetitions and combine trees via majority vote.

The combination of predictive accuracy, reduced overfitting, feature importance analysis, and ease of use makes Random Forests a valuable tool in various machine learning applications. They are particularly well-suited for complex and diverse datasets.

68. What are support vector machines?

Support Vector Machines (SVM) are a class of supervised machine learning algorithms used for classification and regression tasks. SVMs are particularly effective in high-dimensional spaces and are well-suited for problems where the data is not linearly separable.

The key idea behind SVMs is to find the optimal hyperplane that best separates different classes in the feature space. The term "support vector" refers to the data points that lie closest to the decision boundary (hyperplane) and have the most influence on determining the optimal separation. SVMs aim to maximize the margin between classes, which is the distance between the decision boundary and the nearest data points from each class.

Here are the key concepts associated with Support Vector Machines:

- Hyperplane:
  In a two-dimensional space, a hyperplane is a line. In higher dimensions, it becomes a plane, and so on. SVMs find the hyperplane that best separates the classes.
- Margin:
  The margin is the distance between the hyperplane and the nearest data point from each class. SVMs aim to maximize this margin.
- Support Vectors:
  Support vectors are the data points that lie closest to the decision boundary. They are crucial in determining the position and orientation of the hyperplane.
- Linear and Non-linear SVMs:
  SVMs can be applied to linearly separable as well as non-linearly separable data. The choice of the kernel determines the decision boundary shape.
- Kernel Trick:
  SVMs can efficiently handle non-linear decision boundaries by using the kernel trick. Kernels transform the input features into higher-dimensional spaces, making it easier to find linear separation in those spaces.
  In other word, Replace inner product with a symmetric positive (semi-) definite function.
  Advantages:
    - Handling Non-linearity: The Kernel Trick allows linear models to handle non-linear relationships effectively.
    - Computationally Efficient: It avoids the computational cost of explicitly transforming data into higher-dimensional spaces.

69. What are drawbacks of Kernel-trick and how to solve it?

- Drawbacks:
    - Choice of $\emptyset$ limits the function class used for classification to the chosen function.
    - Finding a suitable hyper-parameter we had to use CV (costly).
    - Transformation of data costs calculation power.
- Solution:
  If we somehow could calculate the inner product of our transformed data and get the solution without transforming the data to that higher dimension, we could benefit of the higher dimension while not paying for the costly transformation.

70. What is the difference between Probability and likelihood?

Probability and likelihood are related concepts in statistics, but they have distinct meanings and are used in different contexts. Probability is about predicting future events, while likelihood is about assessing how well a model explains observed data given certain parameters.

71. What is online learning in ML?

Online learning in machine learning refers to a paradigm where a model is trained and updated continuously as new data becomes available. In contrast to traditional batch learning, where the model is trained on a fixed dataset, online learning allows the model to adapt and evolve over time with each new data point or batch of data.

Challenges:

Handling concept drift (changes in the underlying patterns), deciding when to update the model, and dealing with streaming data efficiently are some challenges in online learning.

72. **What is Outlier Detection?**

Outlier detection, also known as anomaly detection, is the process of identifying data points that deviate significantly from the normal behaviour of most of the dataset. The primary goal of outlier detection is to identify observations that are different from most of the data. Outliers can provide valuable insights into potential issues, novel patterns, or anomalies in a dataset.

Challenges in outlier detection include determining an appropriate definition of what constitutes an outlier, handling different types of data (numeric, categorical, time series), and addressing the imbalanced nature of outlier detection problems.

73. **What is misclassification and how to minimize it?**

Misclassification refers to the situation in which a predictive model makes an incorrect prediction or classification for a data point. In other words, it occurs when the predicted label or class assigned by the model does not match the true label or class of the actual data point.

Minimizing misclassification involves improving the performance of a predictive model to reduce the number of incorrect predictions. The specific strategies to achieve this depend on the type of model and the nature of the data. Here are some general approaches:

Use a Better Model, Ensemble Methods, Handle Imbalanced Data, Cross-Validation, Threshold Adjustment etc.

74. **What is Online Passive-Aggressive (PA) algorithms?**

Online Passive-Aggressive (PA) algorithms are a family of machine learning algorithms designed for online learning scenarios. Online learning refers to the continuous update of a model as new data points arrive, and Passive-Aggressive algorithms are specifically designed for online and incremental learning tasks. These algorithms are particularly useful when dealing with large streams of data and situations where the underlying patterns may change over time.

75. **What is One-Class Support Vector Machine (One-Class SVM)?**

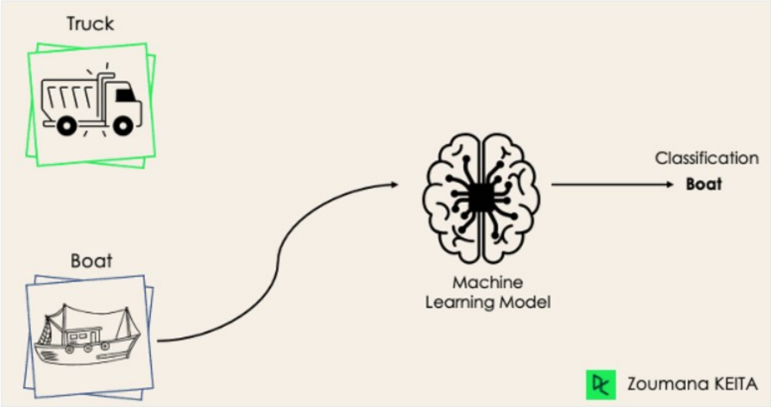
One-Class Support Vector Machine (One-Class SVM) is a machine learning algorithm that falls under the category of unsupervised learning. It is designed for the task of outlier detection or novelty detection, where the objective is to identify observations that are significantly different from most of the data.

One limitation of One-Class SVM is its sensitivity to the choice of parameters, especially the ν parameter. Proper tuning is crucial for effective outlier detection.

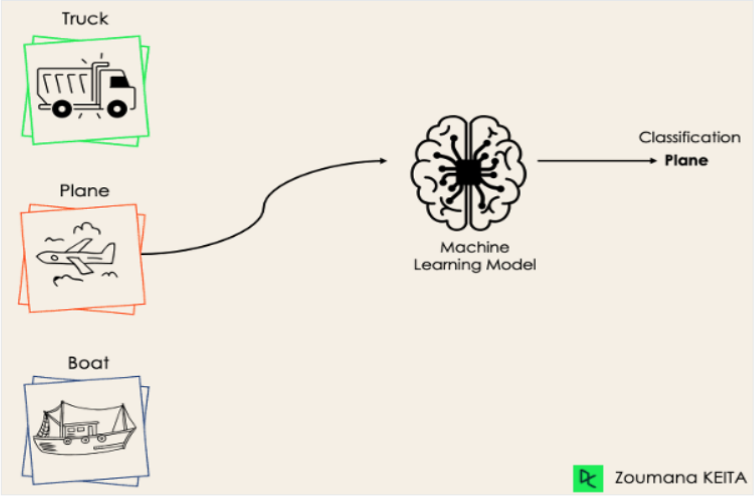
76. **Explain Different Types of Classification Tasks in Machine Learning.**

- Binary Classification:

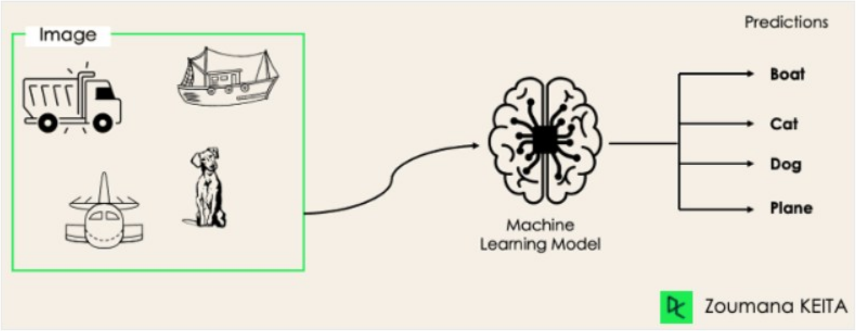
  In a binary classification task, the goal is to classify the input data into two mutually exclusive categories. The training data in such a situation is labelled in a binary format: true and false; positive and negative; O and 1; spam and not spam, etc. depending on the problem being tackled. For instance, we might want to detect whether a given image is a truck or a boat.



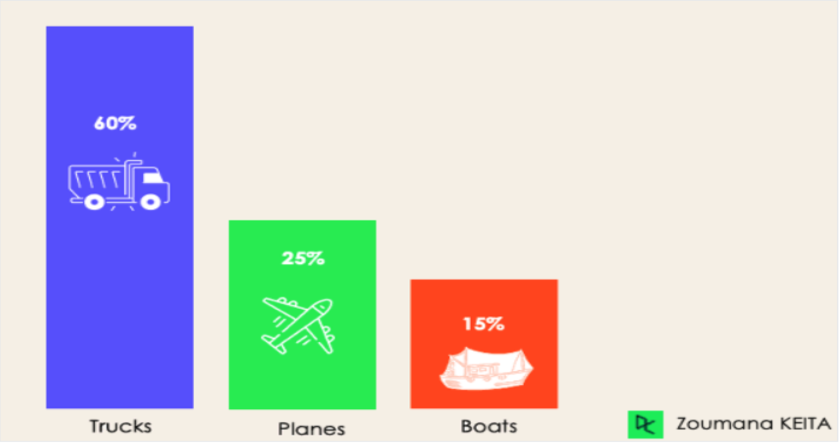
- Multi-Class Classification:
  The multi-class classification, on the other hand, has at least two mutually exclusive class labels, where the goal is to predict to which class a given input example belongs to. In the following case, the model correctly classified the image to be a plane.



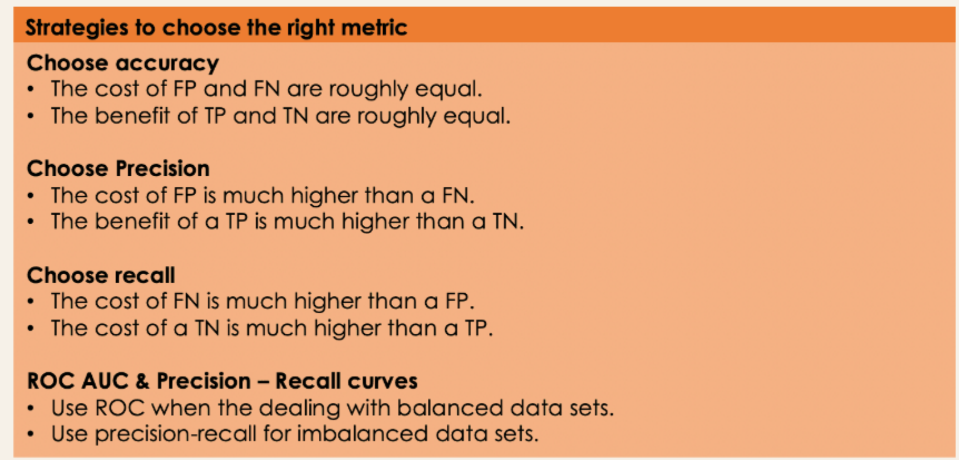
- Multi-Label Classification:
  In multi-label classification tasks, we try to predict 0 or more classes for each input example. In this case, there is no mutual exclusion because the input example can have more than one label.



- Imbalanced Classification:
  For the imbalanced classification, the number of examples is unevenly distributed in each class, meaning that we can have more of one class than the others in the training data. Let’s consider the following 3-class classification scenario where the training data contains: 60% of trucks, 25% of planes, and 15% of boats.



77. **Metrics to Evaluate Machine Learning Classification Algorithms.**
    Confusion matrix, accuracy, precision, recall, F1 score, and area under the ROC (Receiver Operating Characteristic) curve and AUC (Area Under the Curve).



Strategies to choose the right metric

Choose accuracy
- The cost of FP and FN are roughly equal.
- The benefit of TP and TN are roughly equal.

Choose Precision
- The cost of FP is much higher than a FN.
- The benefit of a TP is much higher than a TN.

Choose recall
- The cost of FN is much higher than a FP.
- The cost of a TN is much higher than a TP.

ROC AUC & Precision – Recall curves
- Use ROC when the dealing with balanced data sets.
- Use precision-recall for imbalanced data sets.

78. **What is difference between AUC and ROC?**
    The ROC (Receiver Operating Characteristic) curve is a visual representation of a model's performance at different classification thresholds, the AUC (Area Under the Curve) is a single numerical metric that summarizes the overall performance of the model across all possible thresholds. AUC is a commonly used metric for evaluating the discriminatory power of binary classification models, and it provides a convenient way to compare different models.
    In other words, ROC is a probability curve (CDF of TPR and FPR). AUC represents a degree of distinction between two classes and measures performances of classification models.

79. **Applications of classification.**
    - Application in robotics:
        - Recognize objects that need to be manipulated correctly.
        - Know the type of terrain from sensor data.

  - Allow a robot to understand speech.
  - Recognize the owner/user and his intentions.
  - Anomaly and novelty detection.
- Further applications:
  - Spam detection.
  - Handwriting recognition.
  - Speech recognition.
  - Fraud detection etc.

80. What is overfitting?

Overfitting is a common issue in machine learning where a model learns the training data too well (large values of k), capturing noise or random fluctuations in the data rather than the underlying patterns.

81. What is underfitting?

Underfitting occurs when a machine learning model is too simple (k too small) to capture the underlying patterns in the training data. In simpler terms, the model is not complex enough to represent the relationships between the input features and the output accurately. As a result, it performs poorly on both the training data and new, unseen data.

82. How can k be chosen when true function in unknown?

Test learned hypothesis on hold-out validation data (-Cross validation).

83. Explain evaluation techniques in ML.

In machine learning, evaluation techniques are methods used to assess the performance of a model. These techniques help determine how well a model is likely to generalize to new, unseen data. Here are some common evaluation techniques:

- Holdout methods:

  Process:

  Like train-test split, but the test set is kept aside and not used for model tuning. A separate validation set is often used for tuning hyperparameters. 2/3 is randomly chosen for training and 1/3 for testing.

  Purpose:

  Helps prevent information leakage from the test set during hyperparameter tuning.

  Stratification:

  Each class should be represented in the right proportion in the training and testing set.

  - Stratified holdout method:

    Training and test set are randomly chosen, but the class ratio should be considered (ideally the equal ratio of both classes in the training and test data).
  - Repeated holdout method:

    Repetition of training and testing with different random samples (possible with stratification) to handle sampling bias.
  - A specific form of repeated holdout method is a cross validation.
    - Cross validation:

      Process: The dataset is divided into multiple subsets (folds). The model is trained and evaluated multiple times, each time using a different fold as the test set and the remaining folds as the training set.

Purpose: Provides a more reliable estimate of a model's performance by averaging performance across multiple train-test splits.

- K-Fold Cross-Validation:
  Process: The dataset is divided into K folds, and the model is trained and evaluated K times. Each time, a different fold is used as the test set, and the remaining folds are used for training.
  Purpose: Balances the trade-off between LOOCV and efficiency, providing a good compromise for most cases.
- Leave-One-Out Cross-Validation (LOOCV):
  Process: Each data point is used as a test set exactly once, while the rest of the data is used for training.
  Purpose: Helpful when the dataset is small, but it can be computationally expensive for larger datasets.
- Bootstrapping:
  Process: Randomly sample the dataset with replacement to create multiple bootstrap samples. Train and evaluate the model on each sample.
  Purpose: Provides insights into the stability of model performance by assessing its variability across different subsets of the data.

84. What is Degree of freedom(df)?

The number of values that can varied for computation.

85. Why does variance have n-1 degree of freedom (not n)?

Only n-1 deviations can freely varied.

86. What is Mutual information (based on confusion matrix)?

Difference between a-priory entropy H(A) which is computed based on class ratio and entropy of classification results H(A|P) which is computed based on confusion matrix.

87. What is Gradient Descent? How does gradient descent work?

Gradient descent is an iterative method that is based on the intuition that one finally reaches the bottom of a valley when always going downwards on a hill.

Mathematical Background: Negative gradient is direction of largest decrease of function value!

Gradient Descent is an optimization algorithm commonly used to minimize the cost or loss function in the context of machine learning models. The primary goal is to iteratively adjust the model parameters to find the minimum of the cost function.

Working principle:

- Imagine a blindfolded hiker on a mountain trying to find the lowest point.
- The hiker feels the ground to figure out which way is downhill (gradient).
- Steps are taken in the opposite direction of the slope to go downhill (parameter update).
- The size of each step is carefully chosen (learning rate).
- This process is repeated until the hiker reaches a point where the slope is almost flat (convergence).

Challenges:

- Learning Rate Tuning:
  Selecting an appropriate learning rate is crucial. A too-large learning rate may cause the algorithm to oscillate or diverge, while a too-small learning rate may result in slow convergence.
- Local Minima:
  Gradient Descent may get stuck in local minima, especially in complex cost functions. Various modifications, such as momentum or adaptive learning rates, can help address this issue.

88. What is regularisation and its importance?

Regularization is a technique used in machine learning to prevent overfitting and improve the generalization of a model. Techniques like cross-validation allow to choose appropriate model complexity. But CV is computationally costly, so alternative is Regularization.

Why regularization:

- Preventing Overfitting:
  Regularization helps prevent overfitting by discouraging the model from fitting noise in the training data.
- Improving Generalization:
  By favouring simpler models, regularization improves a model's ability to generalize to new, unseen data.
- Handling Multicollinearity:
  In linear regression, regularization can handle multicollinearity, where predictor variables are highly correlated.

❖ Ensemble learning

89. What are ensemble methods?

Ensemble methods are machine learning techniques that combine the predictions of multiple base models to create a more robust and accurate predictive model. The idea behind ensemble methods is to leverage the diversity of different models to improve overall performance and reduce overfitting.

Here are two commonly used ensemble methods:

- Bagging (Bootstrap Aggregating):
  Process:
  - Train multiple instances of the same learning algorithm on different random subsets (with replacement) of the training data.
  - Combine their predictions, often by averaging (for regression) or voting (for classification).

  Example: Random Forest is a popular ensemble model that uses bagging with decision trees as base models.
- Boosting:
  Process:
  - Train multiple weak learners sequentially, where each learner corrects the errors of the previous one.
  - Combine their predictions with weighted voting, giving more weight to the more accurate models.

  Example: AdaBoost (Adaptive Boosting) and Gradient Boosting Machines (GBM) are popular boosting algorithms.

90. What is inductive Bias?

Inductive bias is the algorithm's way of bringing prior knowledge and assumptions to the table, guiding it to make smart predictions based on what it has seen and learned before.

91. Explain the bias/variance trade-off.

The bias-variance trade-off is a fundamental concept in machine learning that involves finding the right balance between model simplicity and complexity. It addresses the challenge of building a model that accurately captures the underlying patterns in the data without overfitting or underfitting.

- Bias (Simplicity): Bias is related to how much the model assumptions simplify the underlying patterns. High bias means the model is too simplistic and may miss important details.
- Variance (Complexity): Variance is related to how much the model's predictions vary with different training datasets. High variance means the model is too sensitive to the training data and captures noise.
- Trade-off: There's a trade-off between bias and variance. Increasing model complexity typically reduces bias but increases variance, and vice versa.

92. What is a weak learner?

A weak learner is a model or algorithm that performs slightly better than random chance but is not highly accurate on its own. Weak learners are often used in ensemble learning methods, such as boosting, where the goal is to combine the predictions of multiple weak learners to create a strong and accurate model.

93. Explain the difference between sequential and parallel ensembles.

Sequential ensembles often excel in situations where each model corrects the weaknesses of the previous ones, while parallel ensembles are powerful in creating diverse models that collectively lead to more robust predictions.

94. What is Gradient boosting?

Gradient boosting is a machine learning ensemble technique that combines the predictions of multiple weak learners, typically decision trees, sequentially. It aims to improve overall predictive performance by optimizing the model’s weights based on the errors of previous iterations, gradually reducing prediction errors, and enhancing the model’s accuracy.

95. Why encoder and decoder needed in RNN?

In RNN we can give variable as input and get variable in output. For example, we are translating English to German where we are not transferring every word, we transfer it after modifying. For recognize this modification we need encoder or decoder.