**LINEAR REGRESSSION**

1. **What is the difference between simple linear regression and multiple linear regression?**

* Simple linear regression involves one independent variable predicting a dependent variable whereas multiple linear regression contains two or more independent variables in the prediction model.
* Simple Linear Regression follows the equation - Y=a+bX+ε whereas multiple linear regression follows the equation - Y=a+b1​X1​+b2​X2​+…+bk​Xk​+ε.
* Simple linear regression has a straightforward equation with one independent variable whereas Multiple linear regression involves a more elaborate equation, accommodating multiple independent variables.

1. **Explain the concept of the cost function in linear regression.**

* The cost function in linear regression quantifies the difference between predicted and actual values.
* The Mean Squared Error (MSE) or Mean Absolute Error (MAE) is commonly used.
* It calculates the average of squared or absolute differences, respectively.
* It adjusts model parameters through optimization methods like gradient descent to achieve the best fit for the data.

MSE = (1/n) \* sum((y\_pred - y\_true)^2)

 MAE = (1/n) Σ(i=1 to n) |y\_i – ŷ\_i|

1. **How do you interpret the coefficients in a linear regression model?**

Thet are interepted by Intercept and slope.

Intercept(alpha) - Represents the estimated value of the dependent variable when the independent variable is zero.

Slope(beta) - Indicates the change in the dependent variable for a one-unit change in the independent variable.

1. **What are the assumptions of linear regression?**

The key assumptions of linear regression are:

* Linearity: The relationship between variables is linear.
* Independence of Residuals: Residuals are independent of each other.
* Normality of Residuals: Residuals are normally distributed.

**LOGISTIC REGRESSION**

1. **How does logistic regression differ from linear regression?**

* Linear regression predicts continuous outcomes with a linear relationship, whereas logistic regression models probabilities for binary or ordinal classification tasks using the logistic function.
* Linear regression outputs on a continuous scale whereas logistic regression outputs probabilities constrained between 0 and 1.

1. **Explain the sigmoid function and its role in logistic regression.**

* The sigmoid function is a mathematical function that transforms any real-valued number into a range between 0 and 1.
* It is a crucial element in logistic regression and plays a key role in converting linear predictions into probabilities.

The sigmoid function is defined as:

sigma(z) = [1/(1 + e^(-z)) ]

where:

- sigma (z) is the sigmoid function of z,

- e is the base of the natural logarithm,

- z is the input to the function.

The sigmoid function has an S-shaped curve

The logistic regression equation is as follows:

P(Y=1|X) = sigma(beta\_0 + beta\_1X\_1 + beta\_2X\_2 + … + beta\_kX\_k)

where:

- P(Y=1|X) is the probability of the event (e.g., class 1),

- sigma() is the sigmoid function,

- beta\_0, beta\_1, …, beta\_k are the coefficients of the model,

- X\_1, X\_2, …, X\_k are the input features.

1. **What are the key performance metrics used to evaluate a logistic regression model?**

* Accuracy: The ratio of correctly predicted instances to the total number of instances.

Accuracy=True Positives+True Negatives/Total Instances

* Precision: The ratio of correctly predicted positive observations to the total predicted positives.
* Precision=True Positives/False Positives + True Positive  
  Recall (Sensitivity or True Positive Rate):

The ratio of correctly predicted positive observations to the total actual negatives

Recall=True Positives/False Negatives + True Positives​

* F1 Score:

The harmonic mean of precision and recall.

F1 Score=2×(Precision×Recall/Precision+Recall)​

1. **How do you handle multicollinearity in logistic regression?**

To handle multicollinearity in logistic regression:

1.Correlation Analysis:

- Identify highly correlated variables.

2.Variable Selection:

- Remove one of the correlated variables.

3. Combine Variables:

- Create new variables (interaction terms or PCA).

4. Regularization Techniques:

- Apply L1 or L2 regularization.

5. Feature Engineering:

- Create new independent features.

**NAÏVE BAYES**

1. **What is the Naive Bayes algorithm based on?**

* Naïve Bayes algorithm is based on conditional probability.
* It calculates the probability of an event given prior knowledge of related conditions.
* In classification, Naive Bayes assumes conditional independence among features given the class label, simplifying probability calculations.
* It's widely used for its efficiency and effectiveness in various applications, especially in scenarios with limited data.

1. **Explain the concept of conditional probability in the context of Naive Bayes.**

Conditional probability in Naive Bayes is the likelihood of observing a specific feature given the knowledge that the instance belongs to a particular class.

The algorithm assumes that features are conditionally independent given the class label, simplifying probability calculations for classification.

The probability of an instance belonging to a class is computed based on the product of these conditional probabilities, making predictions efficiently.

P(A∣B)=P(B∣A)⋅P(A)​/P(B)

P(A∣B) is the probability of event A given that event B has occurred.

P(B∣A) is the probability of event B given that event A has occurred.P(A) and P(B) are the probabilities of events A and B, respectively.

1. **What are the advantages and disadvantages of Naive Bayes?**

**Advantages of Naive Bayes:**

1. Simplicity and efficiency.

2. Fast training and prediction.

3. Handles high-dimensional data well.

4. Handles missing data.

5. Effective for text classification.

6. Works well with small datasets.

**Disadvantages of Naive Bayes:**

1. Assumption of feature independence.

2. Sensitive to irrelevant features.

3. Limited expressiveness.

4. Biased probability estimates in certain cases.

5. Assumption of normality for Gaussian Naive Bayes.

6. Difficulty with continuous features.

1. **How does Naive Bayes handle missing values and categorical features?**

Handling Missing Values:

- Naive Bayes can handle missing values by excluding the corresponding feature from probability calculations during prediction, making it robust to missing data.

Handling Categorical Features:

- For categorical features, Naive Bayes calculates the probability of each category within the class. It assumes feature independence given the class label, making it effective for categorical data, especially in text classification scenarios.

**DCISION TREES**

1. **How does a decision tree make decisions?**

A decision tree makes decisions by recursively splitting the dataset based on features. It selects the best feature at each node to maximize information gain or Gini impurity reduction. This process continues until a stopping criterion is met, forming a tree structure where each leaf node represents a decision or classification outcome. During prediction, a new instance traverses the tree, following the path of feature-based decisions until it reaches a leaf node, determining the final decision or classification.

1. **What are the main criteria for splitting nodes in a decision tree?**

The main criteria for splitting nodes in a decision tree are:

1. Information Gain (Entropy):

- Measures the reduction in uncertainty or disorder in the dataset after a split. The goal is to maximize information gain.

2. Gini Impurity:

- Measures the probability of incorrectly classifying a randomly chosen element in the dataset. The objective is to minimize Gini impurity.

These criteria guide the decision tree algorithm in selecting the best feature and value to split the data at each node, creating a tree structure that optimally separates classes or minimizes variance in the target variable.

1. **How do decision trees handle categorical variables?**

1. Label Encoding:

- Assigns a unique numerical label to each category, enabling the algorithm to process categorical variables as if they were numerical.

2. One-Hot Encoding:

- Represents each category as a binary vector, with a separate binary feature for each category. This approach avoids imposing an ordinal relationship among categories.

1. **What are some common techniques to prevent overfitting in decision trees?**

1. Pruning: Remove unnecessary branches.

2. Minimum Split Size: Set a minimum samples requirement for splitting.

3. Maximum Depth: Limit the maximum tree depth.

4. Minimum Samples per Leaf: Specify a minimum leaf size.

5. Ensemble Methods: Use Random Forests for a combination of trees.

6. Feature Engineering: Carefully select and engineer features.

**SUPPORT VECTOR MACHINE(SVM)**

1. **What is the basic idea behind SVM?**

- SVM aims to find a hyperplane that best separates data into different classes in a way that maximizes the margin between the classes.

1. **Explain the concepts of margin and support vectors in SVM.**

- Margin: The margin is the distance between the hyperplane and the nearest data points of the classes. SVM seeks to maximize this margin.

- Support Vectors: Support vectors are the data points that lie on the edges of the margin. They play a crucial role in determining the optimal hyperplane.

1. **What are the different kernel functions used in SVM, and when would you use each?**

- Linear Kernel: Suitable for linearly separable data.

- Polynomial Kernel: Useful for non-linear data, introducing polynomial features.

- Sigmoid Kernel: Applicable for neural network-like problems.

1. **How does SVM handle outliers?**

- SVM is less sensitive to outliers due to its focus on maximizing the margin. Outliers may have minimal impact on the placement of the hyperplane, making SVM robust in handling them.

Applying SVM involves finding a hyperplane that maximizes the margin between classes, utilizing support vectors. Different kernels address linear and non-linear data, and SVM's design makes it robust against outliers.