Top 50+ Machine Learning Interview Questions and Answers

Last Updated : 27 Dec, 2024

Machine Learning involves the development of algorithms and statistical models that enable computers to improve their performance in tasks through experience. Machine Learning is one of the booming careers in the present-day scenario.

If you are preparing for **machine learning interview**, this interview preparation guide is a one-stop destination for you. We will discuss the **top 50+ most frequently asked machine learning interview questions** in 2025. Our focus will be on real-life situations and questions that are commonly asked by companies like **Google**, **Microsoft,** and **Amazon** during their interviews.

Top Machine Learning Interview Questions

In this **ML interview questions**, we have covered a wide range of machine learning questions for both freshers and experienced individuals, ensuring thorough preparation for your next ML interview.

> ***Note:*** *These **ML Questions** are also beneficial for individuals who are looking for a quick revision of their machine-learning concepts.*

Machine Learning Interview Questions For Freshers

1. What are some real-life applications of clustering algorithms?

- Recommendation systems for personalized suggestions
- Anomaly detection in fraud prevention
- Image compression to reduce storage
- Healthcare for grouping patients with similar conditions
- Document categorization in search engines

2. How to choose an optimal number of clusters?

- **Elbow Method**: Plot the explained variance or within-cluster sum of squares (WCSS) against the number of clusters. The "elbow" point, where the curve starts to flatten, indicates the optimal number of clusters.
- **Silhouette Score**: Measures how similar each point is to its own cluster compared to other clusters. A higher silhouette score indicates better-defined clusters. The optimal number of clusters is the one with the highest average silhouette score.
- **Gap Statistic**: Compares the clustering result with a random clustering of the same data. A larger gap between the real and random clustering suggests a more appropriate number of clusters.

3. What is feature engineering? How does it affect the model's performance?

Feature engineering refers to developing some new features by using existing features. Sometimes there is a very subtle mathematical relation between some features which if explored properly then the new features can be developed using those mathematical operations.

Also, there are times when multiple pieces of information are clubbed and provided as a single data column. At those times developing new features and using them help us to gain deeper insights into the data as well as if

4. What is overfitting in machine learning and how can it be avoided?

Overfitting happens when the model learns patterns as well as the noises present in the data this leads to high performance on the training data but very low performance for data that the model has not seen earlier.

To avoid overfitting there are multiple methods that we can use:

- Early stopping of the model's training in case of validation training stops increasing but the training keeps going on.
- Using regularization methods like L1 or L2 regularization which is used to penalize the model's weights to avoid overfitting.

5. Why we cannot use linear regression for a classification task?

The main reason why we cannot use linear regression for a classification task is that the output of linear regression is continuous and unbounded, while classification requires discrete and bounded output values.

If we use linear regression for the classification task the error function graph will not be convex. A convex graph has only one minimum which is

the local minima we do not use the linear regression algorithm for a classification task.

6. Why do we perform normalization?

To achieve stable and fast training of the model we use normalization techniques to bring all the features to a certain scale or range of values. If we do not perform normalization then there are chances that the gradient will not converge to the global or local minima and end up oscillating back and forth.

7. What is the difference between precision and recall?

Precision is the ratio between the true positives(TP) and all the positive examples (TP+FP) predicted by the model. In other words, precision measures how many of the predicted positive examples are actually true positives. It is a measure of the model's ability to avoid false positives and make accurate positive predictions.

$$\text{Precision} = \frac{TP}{TP + FP}$$

In recall, we calculate the ratio of true positives (TP) and the total number of examples (TP+FN) that actually fall in the positive class. Recall measures how many of the actual positive examples are correctly identified by the model. It is a measure of the model's ability to avoid false negatives and identify all positive examples correctly.

$$\text{Recall} = \frac{TP}{TP + FN}$$

8. What is the difference between upsampling and downsampling?

In upsampling method, we increase the number of samples in the minority class by randomly selecting some points from the minority class and

epoch but the same high accuracy is not observed in the validation accuracy.

In downsampling, we decrease the number of samples in the majority class by selecting some random number of points that are equal to the number of data points in the minority class so that the distribution becomes balanced. In this case, we have to suffer from data loss which may lead to the loss of some critical information as well.

9. What is data leakage and how can we identify it?

If there is a high correlation between the target variable and the input features then this situation is referred to as data leakage. This is because when we train our model with that highly correlated feature then the model gets most of the target variable's information in the training process only and it has to do very little to achieve high accuracy. In this situation, the model gives pretty decent performance both on the training as well as the validation data but as we use that model to make actual predictions then the model's performance is not up to the mark. This is how we can identify data leakage.

10. Explain the classification report and the metrics it includes.

The **classification report** provides key metrics to evaluate a model's performance, including:

- **Precision**: The proportion of true positives to all predicted positives, measuring accuracy of positive predictions.
- **Recall**: The proportion of true positives to all actual positives, indicating how well the model finds positive instances.
- **F1-Score**: The harmonic mean of precision and recall, balancing the two metrics.

- **Macro Average**: The average of precision, recall, and F1-score across all classes, treating them equally.
- **Weighted Average**: The average of metrics, weighted by class support, giving more importance to frequent classes.

11. What are some of the hyperparameters of the random forest regressor which help to avoid overfitting?

The most important hyperparameters of a Random Forest are:

- **max_depth:** Sometimes the larger depth of the tree can create overfitting. To overcome it, the depth should be limited.
- **n-estimator:** It is the number of decision trees we want in our forest.
- **min_sample_split:** It is the minimum number of samples an internal node must hold in order to split into further nodes.
- **max_leaf_nodes:** It helps the model to control the splitting of the nodes and in turn, the depth of the model is also restricted.

12. What is the bias-variance tradeoff?

First, let's understand what is bias and variance:

- **Bias** refers to the difference between the actual values and the predicted values by the model. Low bias means the model has learned the pattern in the data and high bias means the model is unable to learn the patterns present in the data i.e the underfitting.
- **Variance** refers to the change in accuracy of the model's prediction on which the model has not been trained. Low variance is a good case but high variance means that the performance of the training data and the validation data vary a lot.

If the bias is too low but the variance is too high then that case is known as overfitting. So, finding a balance between these two situations is known as

13. Is it always necessary to use an 80:20 ratio for the train test split?

No, there is no such necessary condition that the data must be split into 80:20 ratio. The main purpose of the splitting is to have some data which the model has not seen previously so, that we can evaluate the performance of the model.

If the dataset contains let's say 50,000 rows of data then only 1000 or maybe 2000 rows of data is enough to evaluate the model's performance.

14. What is Principal Component Analysis?

PCA(Principal Component Analysis) is an unsupervised machine learning dimensionality reduction technique in which we trade off some information or patterns of the data at the cost of reducing its size significantly. In this algorithm, we try to preserve the variance of the original dataset up to a great extent let's say 95%. For very high dimensional data sometimes even at the loss of 1% of the variance, we can reduce the data size significantly.

By using this algorithm we can perform image compression, visualize high-dimensional data as well as make data visualization easy.

15. What is one-shot learning?

One-shot learning is a concept in machine learning where the model is trained to recognize the patterns in datasets from a single example instead of training on large datasets. This is useful when we haven't large datasets. It is applied to find the similarity and dissimilarities between the two images.

16. What is the difference between Manhattan Distance and

Both Manhattan Distance and Euclidean distance are two distance measurement techniques.

- Manhattan Distance (MD) is calculated as the sum of absolute differences between the coordinates of two points along each dimension.

$$MD = |x_1 - x_2| + \ |y_1 - y_2|$$

- Euclidean Distance (ED) is calculated as the square root of the sum of squared differences between the coordinates of two points along each dimension.

$$ED = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Generally, these two metrics are used to evaluate the effectiveness of the clusters formed by a clustering algorithm.

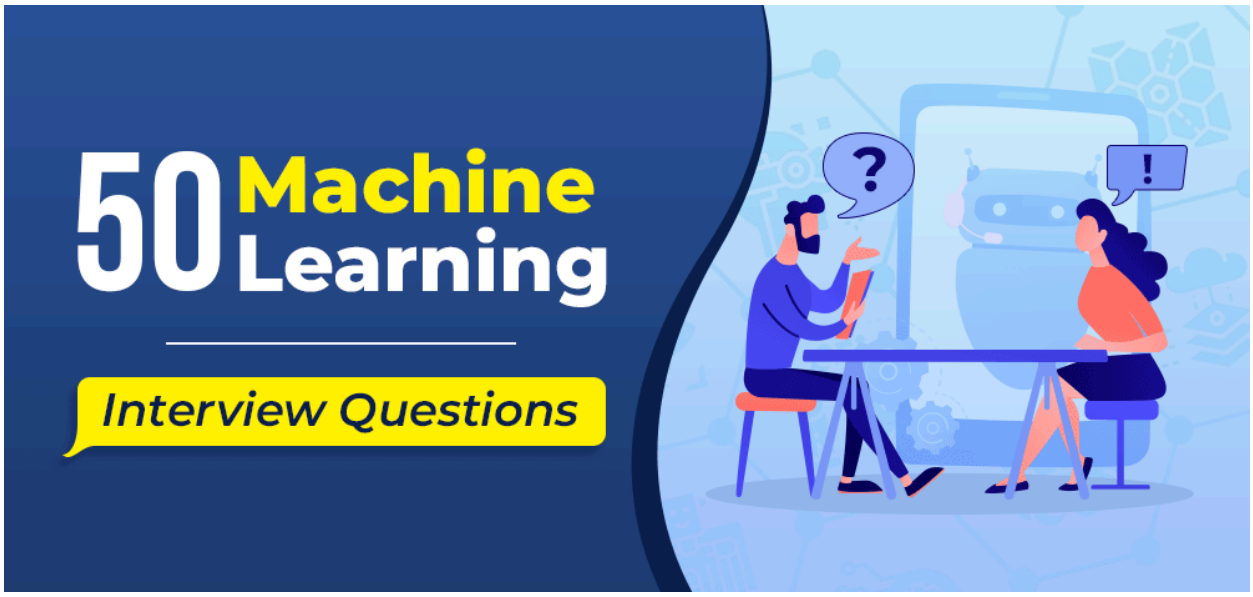
17. What is the difference between one hot encoding and ordinal encoding?

One Hot encoding and ordinal encoding both are different methods to convert categorical features to numeric ones the difference is in the way they are implemented. In one hot encoding, we create a separate column for each category and add 0 or 1 as per the value corresponding to that row.

In ordinal encoding, we replace the categories with numbers from 0 to n-1 based on the order or rank where n is the number of unique categories present in the dataset. The main difference between one-hot encoding and ordinal encoding is that one-hot encoding results in a binary matrix representation of the data in the form of 0 and 1, it is used when there is no order or ranking between the dataset whereas ordinal encoding

18. How can you conclude about the model's performance using the confusion matrix?

Confusion matrix summarizes the performance of a classification model. In a confusion matrix, we get four types of output (in case of a binary classification problem) which are TP, TN, FP, and FN. As we know that there are two diagonals possible in a square, and one of these two diagonals represents the numbers for which our model's prediction and the true labels are the same. Our target is also to maximize the values along these diagonals. From the confusion matrix, we can calculate various evaluation metrics like accuracy, precision, recall, F1 score, etc.



Machine Learning Interview Questions and Answers

19. Explain the working principle of SVM.

A data set that is not separable in different classes in one plane may be separable in another plane. This is exactly the idea behind the SVMin this a low dimensional data is mapped to high dimensional data so, that it becomes separable in the different classes. A hyperplane is determined after mapping the data into a higher dimension which can separate the data into categories

which the data has been categorized. To perform this mapping different types of kernels are used like radial basis kernel, gaussian kernel, polynomial kernel, and many others.

20. What is the difference between the k-means and k-means++ algorithms?

The only difference between the two is in the way centroids are initialized. In the k-means algorithm, the centroids are initialized randomly from the given points. There is a drawback in this method that sometimes this random initialization leads to non-optimized clusters due to maybe initialization of two clusters close to each other.

To overcome this problem k-means++ algorithm was formed. In k-means++, the first centroid is selected randomly from the data points. The selection of subsequent centroids is based on their separation from the initial centroids. The probability of a point being selected as the next centroid is proportional to the squared distance between the point and the closest centroid that has already been selected. This guarantees that the centroids are evenly spread apart and lowers the possibility of convergence to less-than-ideal clusters. This helps the algorithm reach the global minima instead of getting stuck at some local minima.

Read more about it here.

21. Explain some measures of similarity which are generally used in Machine learning.

Some of the most commonly used similarity measures are as follows:

- **Cosine Similarity:** By considering the two vectors in n - dimension we evaluate the cosine of the angle between the two. The range of this similarity measure varies from [-1, 1] where the value 1 represents that

- **Euclidean or Manhattan Distance:** These two values represent the distances between the two points in an n-dimensional plane. The only difference between the two is in the way the two are calculated.
- **Jaccard Similarity:** It is also known as IoU or Intersection over union it is widely used in the field of object detection to evaluate the overlap between the predicted bounding box and the ground truth bounding box.

22. Whether decision tree or random forest is more robust to the outliers.

Decision trees and random forests are both relatively robust to outliers. A random forest model is an ensemble of multiple decision trees so, the output of a random forest model is an aggregate of multiple decision trees.

So, when we average the results the chances of overfitting get reduced. Hence we can say that the random forest models are more robust to outliers.

23. What is the difference between L1 and L2 regularization? What is their significance?

L1 regularization (Lasso regularization)adds the sum of the absolute values of the model's weights to the loss function. This penalty encourages sparsity in the model by pushing the weights of less important features to exactly zero. As a result, L1 regularization automatically performs **feature selection**, removing irrelevant or redundant features from the model, which can improve interpretability and reduce overfitting.

L2 regularization (Ridge regularization) in which we add the square of the weights to the loss function. In both of these regularization methods, weights are penalized but there is a subtle difference between the

In L2 regularization the weights are not penalized to 0 but they are near zero for irrelevant features. It is often used to prevent overfitting by shrinking the weights towards zero, especially when there are many features and the data is noisy.

24. What is a radial basis function?

RBF (radial basis function) is a real-valued function used in machine learning whose value only depends upon the input and fixed point called the center.

The formula for the radial basis function is as follows:

$$K\left(x,\ x^{'}\right) = exp\left(-\frac{\left|\left|x-x^{'}\right|\right|^2}{2\sigma^2}\right)$$

Machine learning systems frequently use the RBF function for a variety of functions, including:

- RBF networks can be used to approximate complex functions. By training the network's weights to suit a set of input-output pairs,
- RBF networks can be used for unsupervised learning to locate data groups. By treating the RBF centers as cluster centers,
- RBF networks can be used for classification tasks by training the network's weights to divide inputs into groups based on how far from the RBF nodes they are.

It is one of the very famous kernels which is generally used in the SVM algorithm to map low dimensional data to a higher dimensional plane so, we can determine a boundary that can separate the classes in different regions of those planes with as much margin as possible.

25. Explain SMOTE method used to handle data imbalance.

method is that the model does not get trained on the same data. But the disadvantage of using this method is that it adds undesired noise to the dataset and can lead to a negative effect on the model's performance.

26. Does the accuracy score always a good metric to measure the performance of a classification model?

No, there are times when we train our model on an imbalanced dataset the accuracy score is not a good metric to measure the performance of the model. In such cases, we use precision and recall to measure the performance of a classification model.

Also, f1-score is another metric that can be used to measure performance but in the end, f1-score is also calculated using precision and recall as the f1-score is nothing but the harmonic mean of the precision and recall.

27. What is KNN Imputer and how does it work?

KNN Imputer imputes missing values in a dataset compared to traditional methods like using mean, median, or mode. It is based on the **K-Nearest Neighbors (KNN)** algorithm, which fills missing values by referencing the values of the nearest neighbors.

Here's how it works:

- **Neighborhood-based Imputation:** The KNN Imputer identifies the **k nearest neighbors** to the data point with the missing value, based on a distance metric (e.g., Euclidean distance).
- **Imputation Process:** Once the nearest neighbors are found, the missing value is imputed (filled) using a statistical measure, such as the mean or median, of the values from these neighbors.
- **Distance Parameter:** The **k parameter** is used to define how many neighbors to consider when imputing a missing value, and the distance

XGBoost model is an ensemble technique of machine learning in this method weights are optimized in a sequential manner by passing them to the decision trees. After each pass, the weights become better and better as each tree tries to optimize the weights, and finally, we obtain the best weights for the problem at hand. Techniques like regularized gradient and mini-batch gradient descent have been used to implement this algorithm so, that it works in a very fast and optimized manner.

29. What is the purpose of splitting a given dataset into training and validation data?

The main purpose is to keep some data left over on which the model has not been trained so, that we can evaluate the performance of our machine learning model after training. Also, sometimes we use the validation dataset to choose among the multiple state-of-the-art machine learning models. Like we first train some models let's say LogisticRegression, XGBoost, or any other than test their performance using validation data and choose the model which has less difference between the validation and the training accuracy.

30. Explain some methods to handle missing values in that data.

Some of the methods to handle missing values are as follows:

- Removing the rows with null values may lead to the loss of some important information.
- Removing the column having null values if it has very less valuable information. it may lead to the loss of some important information.
- Imputing null values with descriptive statistical measures like mean, mode, and median.
- Using methods like KNN Imputer to impute the null values in a more sophisticated way

31. What is the difference between k-means and the KNN algorithm?

K-means algorithm is one of the popular unsupervised machine learning algorithms which is used for clustering purposes. But, KNN is a model which is generally used for the classification task and is a supervised machine learning algorithm. The k-means algorithm helps us to label the data by forming clusters within the dataset.

32. What is Linear Discriminant Analysis?

Linear Discriminant Analysis (LDA) is a supervised machine learning dimensionality reduction technique because it uses target variables also for dimensionality reduction. It is commonly used for classification problems. The LDA mainly works on two objectives:

- Maximize the distance between the means of the two classes.
- Minimize the variation within each class.

33. How can we visualize high-dimensional data in 2-d?

One of the most common and effective methods is by using the t-SNE algorithm which is a short form for t-Distributed Stochastic Neighbor Embedding. This algorithm uses some non-linear complex methods to reduce the dimensionality of the given data. We can also use PCA or LDA to convert n-dimensional data to 2 - dimensional so, that we can plot it to get visuals for better analysis. But the difference between the PCA and t-SNE is that the former tries to preserve the variance of the dataset but the t-SNE tries to preserve the local similarities in the dataset.

34. What is the reason behind the curse of dimensionality?

from the limited number of datasets or we can say that the weights are not optimized properly due to the high dimensionality of the data and the limited number of examples used to train the model. Due to this after a certain threshold for the dimensionality of the input data, we have to face the curse of dimensionality.

35. Which metric is more robust to outliers: MAE, MSE, or RMSE?

Out of the three metrics—Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE)—**MAE** is more robust to outliers.

The reason behind this is the way each metric handles error values:

- **MSE and RMSE** both square the error values. When there are outliers, the error is typically large, and squaring it results in even larger error values. This causes outliers to disproportionately affect the overall error, leading to misleading results and potentially distorting the model's performance.
- **MAE**, on the other hand, takes the absolute value of the errors. Since it does not square the error terms, the influence of large errors (outliers) is linear rather than exponential, making MAE less sensitive to outliers.

36. Why removing highly correlated features are considered a good practice?

When two features are highly correlated, they may provide similar information to the model, which may cause overfitting. If there are highly correlated features in the dataset then they unnecessarily increase the dimensionality of the feature space and sometimes create the problem of the curse of dimensionality. If the dimensionality of the feature space is high then the model training may take more time than expected, it will

37. What is the difference between the content-based and collaborative filtering algorithms of recommendation systems?

In a content-based recommendation system, similarities in the content and services are evaluated, and then by using these similarity measures from past data we recommend products to the user. But on the other hand in collaborative filtering, we recommend content and services based on the preferences of similar users.

For example, if one user has taken A and B services in past and a new user has taken service A then service A will be recommended to him based on the other user's preferences.

38. How you would assess the goodness-of-fit for a linear regression model? Which metrics would you consider most important and why?

To evaluate the performance of a linear regression model, important key metrics are: R-squared, Adjusted R-squared, RMSE, and F-Statistics. R-squared is particularly important as it reflects the proportion of variance in the dependent variable that can be explained by the independent variables, providing a measure of how well our model fits the data. However, Adjusted R-squared also plays a crucial role, especially when comparing models with different numbers of predictors. It adjusts for the complexity of the model, helping to prevent overfitting and ensuring the robustness of our findings.

To learn more about regression metrics, check out: **Regression Metrics**

39. What is the null hypothesis in linear regression problem?

In linear regression, the null hypothesis id that there is no relationship

Essentially, the null hypothesis suggests that the predictor variable does not contribute to predicting the outcome. For instance, if the null hypothesis states that the slope of the regression line is zero, then a student's score in an English class would not be a useful predictor of their overall grade-point average.

The alternative hypothesis, denoted as $H_1 : \beta_1 \neq 0$, proposes that changes in the independent variable are indeed associated with changes in the dependent variable, indicating a meaningful relationship.

40. Can SVMs be used for both classification and regression tasks?

Yes, Support Vector Machines (SVMs) can be used for both classification and regression. For classification, SVMs work by finding a hyperplane that separates different classes in the data with the largest gap possible.

For regression, which involves predicting a continuous number, SVMs are adapted into a version called Support Vector Regression (SVR). SVR tries to fit as many data points as possible within a certain range of the predicted line, allowing some errors but penalizing those that are too large. This makes it useful for predicting values in situations where the data shows complex patterns.

To learn how to implement Support Vector Regression, you can refer to: **Support Vector Regression (SVR) using Linear and Non-Linear Kernels in Scikit Learn**

41. Explain the concept of weighting in KNN? What are the different ways to assign weights, and how do they affect the model's predictions?

Weighting in KNN assigns different levels of importance to the neighbors based on their distance from the query point, influencing how each

- **Uniform Weighting:** All neighbors have equal weight regardless of their distance.
- **Distance Weighting:** Weights are inversely proportional to the distance, giving closer neighbors more influence.
- **User-defined Weights:** Weights are assigned based on domain knowledge or specific data characteristics.

Effect on Model's Prediction:

- **Uniform Weighting:** Simple but may not perform well with noisy data or varied distances.
- **Distance Weighting:** Improves accuracy by emphasizing closer neighbors, useful for irregular class boundaries but sensitive to anomalies.
- **User-defined Weights:** Optimizes performance when specific insights about the dataset are applied, though less generalizable.

42. What are the assumptions behind the K-means algorithm? How do these assumptions affect the results?

The assumptions of K-Means algorithm include:

1. **Cluster Shape:** Assumes clusters are spherical and of similar size, affecting how well it handles non-spherical groups.
2. **Scale of Features:** Assumes features are on similar scales; different ranges can distort the distance calculation.
3. **Clusters are Balanced:** Assumes clusters have a roughly equal number of observations, which can bias results against smaller clusters.
4. **Similar Density:** Assumes all clusters have similar density, impacting the algorithm's effectiveness with clusters of varying densities.

If these assumptions are not met, the model will perform poorly making difficult to process and select clustering techniques that align with the

43. Can you explain the concept of convergence in K-means? What conditions must be met for K-means to converge?

Convergence in K-means occurs when the cluster centroids stabilize, and the assignment of data points to clusters no longer changes. This happens when the algorithm has minimized the sum of squared distances between points and their corresponding centroids.

Conditions for K-means to Converge:

1. **Proper Initialization:** The initial placement of centroids significantly impacts convergence. Techniques like k-means++ help ensure a better start.
2. **Data Characteristics:** The algorithm converges more effectively if the data naturally clusters into well-separated groups. Overlapping or complex cluster shapes can hinder convergence.
3. **Correct Number of Clusters (k):** Choosing the right number of clusters is critical; too many or too few can lead to slow convergence or convergence to suboptimal solutions.
4. **Algorithm Parameters:** Setting a maximum number of iterations and a small tolerance for centroid change helps prevent infinite loops and determines when the algorithm should stop if centroids move minimally between iterations.

44. What is the significance of tree pruning in XGBoost? How does it affect the model?

Tree pruning in XGBoost is used to reduce model complexity and prevent overfitting. XGBoost implements a "pruning-as-you-grow" strategy where it starts by growing a full tree up to a maximum depth, then prunes back the branches that contribute minimal gains in terms of loss reduction. This is guided by the gamma parameter, which sets a minimum loss reduction

1. **Reduces Overfitting:** By trimming unnecessary branches, pruning helps in creating simpler models that generalize better to unseen data, reducing the likelihood of overfitting.
2. **Improves Performance:** Pruning helps in removing splits that have little impact, which can enhance the model's performance by focusing on more significant attributes.
3. **Optimizes Computational Efficiency:** It decreases the complexity of the final model, which can lead to faster training and prediction times as there are fewer nodes to traverse during decision making.

45. How does Random Forest ensure diversity among the trees in the model?

Random Forest ensures diversity among the trees in its ensemble through two main mechanisms:

1. **Bootstrap Aggregating (Bagging):** Each tree in a Random Forest is trained on a different bootstrap sample, a random subset of the data. This sampling with replacement means that each tree sees different portions of the data, leading to variations in their learning and decision-making processes.
2. **Feature Randomness:** When splitting a node during the construction of the tree, Random Forest randomly selects a subset of features instead of using all available features. This variation in the feature set ensures that trees do not follow the same paths or use the same splits, thereby increasing the diversity among the trees.

The diversity among trees reduces the variance of the model without significantly increasing the bias.

46. What is the concept of information gain in decision trees? How does it guide the creation of the tree structure?

Information gain is a measure used in decision trees to select the best feature that splits the data into the most informative subsets. It is calculated based on the reduction in entropy or impurity after a dataset is split on an attribute. Entropy is a measure of the randomness or uncertainty in the data set, and information gain quantifies how much splitting on a particular attribute reduces that randomness.

47. How does the independence assumption affect the accuracy of a Naive Bayes classifier?

Naive Bayes classifier operates under the assumption that all features in the dataset are independent of each other given the class label. This assumption simplifies the computation of the classifier's probability model, as it allows the conditional probability of the class given multiple features to be calculated as the product of the individual probabilities for each feature.

Affect of accuracy on a Naive Bayes classifier:

1. **Strengths in High-Dimensional Data:** In practice, the independence assumption can sometimes lead to good performance, especially in high-dimensional settings like text classification, despite the interdependencies among features. This is because the errors in probability estimates may cancel out across the large number of features.
2. **Limitations Due to Feature Dependency:** The accuracy of Naive Bayes can be adversely affected when features are not independent, particularly if the dependencies between features are strong and critical to predicting the class. The model may underperform in such scenarios because it fails to capture the interactions between features.
3. **Generalization Capability:** The simplistic nature of the independence assumption often allows Naive Bayes to perform well on smaller

48. Why does PCA maximize the variance in the data?

PCA aims to **maximize the variance** because variance represents how much information is spread out in a given direction. The higher the variance along a direction, the more information that direction holds about the data. By focusing on the directions of highest variance, PCA helps us:

- **Preserve information** while reducing the dimensionality.
- **Simplify the data** by eliminating less important features (those with low variance)

49. How do you evaluate the effectiveness of a machine learning model in an imbalanced dataset scenario? What metrics would you use instead of accuracy?

We can use Precision, Recall, F1 score and ROC-AUC to evaluate the effectiveness of machine learning model in imbalanced dataset scenario. The best metric is F1 score as it combines both precision and recall into single metric that is important in imbalanced datasets where a high number of true negatives can skew accuracy. By focusing on both false positives and false negatives, the F1-score ensures that both the positive class detection and false positives are accounted for.

- If the cost of false positives (Type I errors) and false negatives (Type II errors) is similar, F1-Score strikes a good balance.
- It is especially useful when you need to prioritize performance in detecting the minority class (positive class).

However, if you are more concerned about false positives or false negatives specifically, you may opt for:

- Precision (if false positives are more costly) or
- Recall (if false negatives are more costly).

One-Class SVM is an unsupervised anomaly detection algorithm. It is often used when only normal data is available. The model learns a decision boundary around normal data points using a kernel, typically an RBF, to map the data into a higher-dimensional space. The algorithm identifies support vectors—data points closest to the boundary—and any new data point outside this boundary is flagged as an anomaly. Key parameters like 'nu' control the fraction of outliers allowed, while the kernel defines the boundary shape.

51. Explain the concept of "concept drift" in anomaly detection.

Concept driftrefers to the change in the underlying distribution or patterns in the data over time, which can make previously normal data points appear as anomalies. In anomaly detection, this is particularly challenging because a model trained on old data may not recognize new, evolving patterns as part of the normal data distribution. Concept drift can occur suddenly or gradually and needs to be monitored closely. To address this, models can be adapted through periodic retraining with new data or by using adaptive anomaly detection algorithms.

Next Article

100+ Machine Learning Projects with Source Code [2025]

Similar Reads

1. Top 25 Machine Learning System Design Interview Questions
2. 10 Basic Machine Learning Interview Questions
3. Top 50 Data Mining Interview Questions & Answers
4. Deep Learning Interview Questions

7. Top Artificial Intelligence(AI) Interview Questions and Answers
8. Top 10 Benefits of Machine Learning in Businesses
9. Quantiphi Interview Experience for Machine learning engineer 2024 (On-Campus)
10. Tharsio Limited Interview Experience | Machine Learning Internship

GoLang
SQL
R Language
Android

DSA Roadmap
DSA Interview Questions
Competitive Programming

Data Science & ML

Data Science With Python
Machine Learning
ML Maths
Data Visualisation
Pandas
NumPy
NLP
Deep Learning

Web Technologies

HTML
CSS
JavaScript
TypeScript
ReactJS
NextJS
NodeJs
Bootstrap
Tailwind CSS

Python Tutorial

Python Examples
Django Tutorial
Python Projects
Python Tkinter
Web Scraping
OpenCV Tutorial
Python Interview Question

Computer Science

GATE CS Notes
Operating Systems
Computer Network
Database Management System
Software Engineering
Digital Logic Design
Engineering Maths

DevOps

Git
AWS
Docker
Kubernetes
Azure
GCP
DevOps Roadmap

System Design

High Level Design
Low Level Design
UML Diagrams
Interview Guide
Design Patterns
OOAD
System Design Bootcamp
Interview Questions

School Subjects

Mathematics
Physics
Chemistry
Biology
Social Science
English Grammar

Databases

SQL
MYSQL
PostgreSQL
PL/SQL
MongoDB