MACHINE LEARNING

1. Which of the following in sk-learn library is used for hyper parameter tuning?
   A) GridSearchCV() B) RandomizedCV()
   C) K-fold Cross Validation D) All of the above
   Answer – GridSearchCV()

2. In which of the below ensemble techniques trees are trained in parallel?
   A) Random forest B) Adaboost
   C) Gradient Boosting D) All of the above
   Answer – Random forest

3. In machine learning, if in the below line of code:
   *sklearn.svm.**SVC** (C=1.0, kernel='rbf', degree=3)*
   we increasing the C hyper parameter, what will happen?
   A) The regularization will increase B) The regularization will decrease
   C) No effect on regularization D) kernel will be changed to linear
   Answer – The regularization will decrease

4. Check the below line of code and answer the following questions:
   *sklearn.tree.**DecisionTreeClassifier**(*criterion='gini',splitter='best',max_depth=None, min_samples_split=2)*
   Which of the following is true regarding max_depth hyper parameter?
   A) It regularizes the decision tree by limiting the maximum depth up to which a tree can be grown.
   B) It denotes the number of children a node can have.
   C) both A & B
   D) None of the above
   Answer – It regularizes the decision tree by limiting the maximum depth up to which a tree can be grown.

5. Which of the following is true regarding Random Forests?
   A) It's an ensemble of weak learners.
   B) The component trees are trained in series
   C) In case of classification problem, the prediction is made by taking mode of the class labels predicted by the component trees.
   D) None of the above
   Answer – In case of classification problem, the prediction is made by taking mode of the class labels predicted by the component trees.

6. What can be the disadvantage if the learning rate is very high in gradient descent?
   A) Gradient Descent algorithm can diverge from the optimal solution.
   B) Gradient Descent algorithm can keep oscillating around the optimal solution and may not settle.
   C) Both of them
   D) None of them
   Answer – Gradient Descent algorithm can diverge from the optimal solution.

7. As the model complexity increases, what will happen?
   A) Bias will increase, Variance decrease B) Bias will decrease, Variance increase
   C)both bias and variance increase D) Both bias and variance decrease.
   Answer – Bias will decrease, variance increase

MACHINE LEARNING

8. Suppose I have a linear regression model which is performing as follows:
   Train accuracy=0.95 and Test accuracy=0.75
   Which of the following is true regarding the model?
   A) model is underfitting B) model is overfitting
   C) model is performing good D) None of the above
   Answer – model is overfitting

Q9 to Q15 are subjective answer type questions, Answer them briefly.

9. Suppose we have a dataset which have two classes A and B. The percentage of class A is 40% and percentage of class B is 60%. Calculate the Gini index and entropy of the dataset.

Answer - The Gini index and entropy are two commonly used impurity measures in decision tree algorithms. They measure the degree of impurity or randomness in a dataset.

To calculate the Gini index and entropy of the given dataset with two classes A and B, where class A has a percentage of 40% and class B has a percentage of 60%, we can use the following formulas:

Gini index = 2 * (P_A * P_B)

Entropy = - (P_A * log2(P_A)) - (P_B * log2(P_B))

where P_A and P_B are the probabilities of each class.

Substituting the values in the formulas, we get:

Gini index = 2 * (0.4 * 0.6) = 0.48

Entropy = - (0.4 * log2(0.4)) - (0.6 * log2(0.6)) = 0.971

Therefore, the Gini index of the dataset is 0.48, and the entropy of the dataset is 0.971.

10. What are the advantages of Random Forests over Decision Tree?

Answer - Random Forests have several advantages over Decision Trees, including:

1. Reduced overfitting: Random Forests reduce the overfitting problem by creating multiple decision trees using bootstrap samples of the data and random subsets of the features. This helps to reduce the variance and improve the generalization performance of the model.
2. Better accuracy: Random Forests have been shown to provide better accuracy than Decision Trees in many cases, especially when dealing with complex datasets.
3. Robustness to noise and outliers: Random Forests are less sensitive to noise and outliers in the data compared to Decision Trees, as the randomness introduced in the model-building process helps to reduce their impact.
4. Feature importance: Random Forests provide a measure of feature importance, which can help in feature selection and understanding the underlying patterns in the data.
5. Parallel processing: Random Forests can be easily parallelized, as the individual decision trees in the forest can be trained independently of each other.

   Overall, Random Forests are a more robust and accurate model compared to Decision Trees and are well-suited for a wide range of machine-learning problems.

MACHINE LEARNING

11. What is the need of scaling all numerical features in a dataset? Name any two techniques used for scaling.

Answer - Scaling is the process of transforming the numerical features of a dataset to a common scale. The need for scaling arises because many machine learning algorithms are sensitive to the scale of the features. If the features are on different scales, some features may dominate others in the algorithm's objective function, leading to biased results.

Scaling is important to ensure that all features contribute equally to the model and prevent the model from being influenced by the scale of the features. This is particularly important for algorithms that use distance-based calculations, such as K-Nearest Neighbors (KNN) and Support Vector Machines (SVM).

Two commonly used techniques for scaling are:

Min-Max scaling: In this technique, the values of the features are scaled to lie between 0 and 1. It can be done using the formula:

x_scaled = (x - min(x)) / (max(x) - min(x))

Standardization: In this technique, the values of the features are transformed to have zero mean and unit variance. It can be done using the formula:

x_scaled = (x - mean(x)) / std(x)

Where x is the original feature, x_scaled is the scaled feature, min(x) and max(x) are the minimum and maximum values of x, respectively, and mean(x) and std(x) are the mean and standard deviation of x, respectively.
By using these scaling techniques, we can ensure that all numerical features are on the same scale and contribute equally to the model, leading to better performance and more accurate results.

12. Write down some advantages which scaling provides in optimization using gradient descent algorithm.

Answer - Scaling provides several advantages in optimization using gradient descent algorithm, including:

1. Faster convergence: Scaling can help the gradient descent algorithm converge faster by ensuring that the steps taken by the algorithm are more consistent and in the appropriate direction.
2. Avoiding numerical overflow/underflow: In some cases, the range of the input features may be too large, leading to numerical overflow or underflow in the optimization process. Scaling can help prevent this problem by bringing the input features to a more manageable range.
3. Better conditioned problem: Scaling can help ensure that the objective function is well-conditioned, which means that its Hessian matrix has well-behaved eigenvalues. This makes the optimization process more stable and less prone to numerical errors.
4. Preventing the algorithm from getting stuck in local optima: Scaling can help the gradient descent algorithm avoid getting stuck in local optima by ensuring that the objective function has a more consistent shape across all dimensions.

Overall, scaling can make the optimization process more efficient and effective, leading to better performance and more accurate results.

MACHINE LEARNING

13. In case of a highly imbalanced dataset for a classification problem, is accuracy a good metric to measure the performance of the model. If not, why?

Answer - In the case of a highly imbalanced dataset for a classification problem, accuracy may not be a good metric to measure the performance of the model. This is because accuracy is biased towards the majority class and may not reflect the performance of the model on the minority class.

For example, if we have a dataset with 95% of observations in one class and only 5% in the other class, a model that always predicts the majority class will have an accuracy of 95%, even though it is not performing well on the minority class.

In such cases, alternative metrics such as precision, recall, F1-score, and Area Under the Receiver Operating Characteristic Curve (AUROC) are better indicators of model performance. These metrics provide a more balanced view of the model's ability to predict both classes and are less sensitive to class imbalance.

Precision is the proportion of true positive predictions out of all positive predictions, while recall is the proportion of true positive predictions out of all actual positive observations. F1-score is the harmonic mean of precision and recall. AUROC measures the ability of the model to distinguish between positive and negative classes across different probability thresholds.

Therefore, when dealing with a highly imbalanced dataset, it is important to choose appropriate evaluation metrics that consider the performance of the model on both classes.

14. What is “f-score" metric? Write its mathematical formula.

Answer - The F-score is a metric used to evaluate the performance of a binary classification model, which takes into account both precision and recall.

The mathematical formula for the F-score is:

F-score = 2 * (precision * recall) / (precision + recall)

where precision is the proportion of true positive predictions out of all positive predictions, and recall is the proportion of true positive predictions out of all actual positive observations.

The F-score ranges from 0 to 1, with a higher score indicating better model performance. A perfect F-score of 1 is achieved when both precision and recall are 1, indicating that the model has made no false positive or false negative predictions.

The F-score is a popular metric in machine learning and is often used in situations where the data is imbalanced and the classes have different prior probabilities. It provides a more balanced view of the model's performance than accuracy, which can be biased towards the majority class in such situations.

15. What is the difference between fit(), transform() and fit_transform()?

Answer - In machine learning, fit(), transform(), and fit_transform() are commonly used methods for preprocessing data.

fit(): The fit() method is used to calculate the parameters required for preprocessing the data. It calculates the mean, standard deviation, or other statistical measures of the data that will be used to transform it. The fit() method is called on the training data set and then the calculated parameters are used to transform the test data set.

transform(): The transform() method is used to apply the calculated parameters to the data set. It takes

MACHINE LEARNING

the input data and applies the calculated transformation to it. The transform() method is called on the test data set after the fit() method has been called on the training data set.

- fit_transform(): The fit_transform() method is a combination of the fit() and transform() methods. It first calculates the parameters required for preprocessing the data using the fit() method and then applies the transformation to the data using the transform() method. It is called on the training data set and returns the transformed training data set.

  In summary, the fit() method is used to calculate the parameters required for preprocessing the data, the transform() method is used to apply the transformation to the data, and the fit_transform() method is used to perform both the fitting and transformation steps in one step.

MACHINE LEARNING