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

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

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

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

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

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

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.

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

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% andpercentage of class B is 60%. Calculate the Gini index and entropy of the dataset.**

   Answer –

   Gini index is a measure of the impurity of a set of examples. It is defined as:

   Gini = 1 - (p(A)^2 + p(B)^2)

MACHINE LEARNING

Entropy is a measure of impurity in a set of examples. It is defined as:

Entropy = -p(A) * log2(p(A)) - p(B) * log2(p(B))

For the given dataset:

p(A) = 0.4, p(B) = 0.6

Gini = 1 - (0.4^2 + 0.6^2) = 0.48

Entropy = -0.4 * log2(0.4) - 0.6 * log2(0.6) = 0.97

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

Answer – There are several advantages of Random Forests over Decision Trees:

- Reduced Overfitting: Random Forests is an ensemble method, which means it combines the predictions of multiple decision trees. This averaging process reduces the overfitting problem, which is a common issue in decision trees.
- Improved Accuracy: Random Forests often have higher accuracy than a single decision tree, because it reduces the variance by averaging the results of multiple decision trees.
- Handling Missing Values: Random Forests can handle missing values and maintain accuracy, whereas decision trees can't handle missing values and require imputation.
- Handling Categorical Variables: Random Forests can handle categorical variables, whereas decision trees can't.
- Feature Importance: Random Forests provide feature importance, which can be used to select the most important features for the model.
- Handling Non-Linearity: Random Forest is a non-parametric model, which means it can handle non-linearity and provide accurate results for non-linear data.

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

Answer - Scaling is important when working with numerical features, because it ensures that all features are on the same scale and have similar magnitude. This is important because many machine learning algorithms use distance-based metrics to calculate similarity between examples, and if the features are not on the same scale, some features may dominate the distance calculation.

Two common techniques used for scaling numerical features are:

- Min-Max Scaling: Also known as Min-Max Normalization, it scales the values of a feature to a given range (usually 0 to 1) by subtracting the minimum value and dividing by the range.
- Standardization: It scales the values of a feature by subtracting the mean and dividing by the standard deviation. This is useful when the data has a Gaussian (normal) distribution.

MACHINE LEARNING

It is important to note that some models like tree based models are not sensitive to the scale of the feature.

12. **Write down some advantages which scaling provides in optimization using gradient descentalgorithm.**

Answer –

- Scaling provides several advantages when using gradient descent optimization algorithm:
- Faster Convergence: Scaling the features to have similar magnitudes helps the optimization algorithm converge faster, because the optimizer is able to make larger steps in the direction of the optimal solution.
- Stable Convergence: Scaling also helps the optimization algorithm converge more stably, because it ensures that all features contribute equally to the optimization process. Without scaling, features with larger magnitudes may dominate the optimization process.
- Handling Non-Linearity: Scaling helps the optimization algorithm handle non-linearity by making the optimization process more robust to variations in the feature magnitudes.
- Avoiding Local Minima: Scaling helps the optimization algorithm avoid getting stuck in local minima, because it helps the optimizer make larger steps towards the global minimum.
- Handling Large Data: Scaling can help the optimization algorithm handle large data set by reducing the computation time and memory usage, as scaling the data makes it more manageable for the algorithm to optimize.

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

Answer – Accuracy is not a good metric to measure the performance of the model in case of a highly imbalanced dataset for a classification problem because accuracy only takes into account the number of correct predictions and does not take into account the balance of the classes.

In a highly imbalanced dataset, even a model that always predicts the majority class will have a high accuracy. However, such a model would be of little value in practice, because it would not be able to correctly identify the minority class.

Instead of accuracy, other metrics such as precision, recall, F1-score, AUC-ROC and Matthew's Correlation Coefficient (MCC) are more suitable for imbalanced datasets. These metrics take into account the balance of the classes and provide a more complete picture of the model's performance.

Precision is a metric that calculates the proportion of correctly predicted positive observations out of total predicted positives. Recall is a metric that calculates the proportion of correctly predicted positive observations out of all actual positives. The F1-score is the harmonic mean of precision and recall, and can be seen as a balance between precision and recall. AUC-ROC is a metric that measures the ability of the classifier to distinguish between classes, and MCC is a metric that considers all possible combinations of predicted and actual class, hence providing a robust metric for imbalanced datasets.

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

Answer – F-score, also known as F1-score or F-measure, is a measure of a test's accuracy that balances precision and recall. It is commonly used in information retrieval, natural language processing, and other fields where the goal is to balance the precision and recall of a model.

MACHINE LEARNING

The F-score is the harmonic mean of precision and recall, where precision is the proportion of true positive predictions among all positive predictions, and recall is the proportion of true positive predictions among all actual positive observations.

The mathematical formula for F-score is:

F-score = 2 * (Precision * Recall) / (Precision + Recall)

Precision = True Positive / (True Positive + False Positive)

Recall = True Positive / (True Positive + False Negative)

It is a trade-off between precision and recall, the F1 Score might be more useful than accuracy, especially if you need to seek a balance between precision and recall. A harmonic mean is used rather than a simple average because it punishes extreme values, such as a low precision and high recall or vice versa.

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

Answer- In machine learning and data preprocessing, the terms fit(), transform(), and fit_transform() refer to methods used to prepare data for modeling.

- fit() is used to calculate the parameters of the transformation method. For example, the mean and standard deviation of a dataset for standardization. This method is used to learn the parameters from the training data.

- transform() is used to apply the transformation method to the dataset. It uses the parameters learned from the fit() method to transform the data.

- fit_transform() is a combination of the fit() and transform() methods. It first learns the parameters from the training data using the fit() method and then applies the transformation using the transform() method.

For example, in the case of standardizing the data, we use fit_transform() method to calculate the mean and standard deviation of the data and then apply this standardization on the data.

In summary, fit_transform() is used to learn the parameters of the transformation method and apply the transformation in one step, while fit() is used to learn the parameters and transform() is used to apply the transformation using the learned parameters.