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

Ans: 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

Ans: 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

Ans: B) 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

Ans: A)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

Ans: C) 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

Ans: C) Both 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.

Ans: Ans-B

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

Ans: C) model is performing good

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.
   Ans: The Gini index of a dataset is calculated as 1 - (p1^2 + p2^2) where p1 and p2 are the proportions of the two classes in the dataset. In this case, the Gini index would be:
   1 - (0.4^2 + 0.6^2) = 1 - (0.16 + 0.36) = 1 - 0.52 = 0.48

   The entropy of a dataset is calculated as -p1 * log2(p1) - p2 * log2(p2) where p1 and p2 are the proportions of the two classes in the dataset. In this case, the entropy would be:
   -0.4 * log2(0.4) - 0.6 * log2(0.6) = -0.6309
10. What are the advantages of Random Forests over Decision Tree?
    - **Random Forests have several advantages over decision trees. Some of the main advantages include:**
    - **Improved accuracy: Random Forests tend to be more accurate than individual decision trees because they are able to reduce overfitting by averaging the predictions of multiple trees.**
    - **Reduced variance: Random Forests also tend to have lower variance than individual decision trees, which means they are less likely to be affected by outliers or noise in the data.**
    - **Better Handling of missing values: Random Forests are able to handle missing values in the data better than decision trees because they can use the values from other trees in the ensemble.**
    - **Better Handling of categorical variable: Random Forests can handle categorical variables without the need for one-hot encoding.**
    - **Feature importance: Random Forests allow you to measure the feature importance.**
    - **Handling non-linearity: Random Forests can handle non-linearity and interactions between features without the need for feature selection.**
11. What is the need of scaling all numerical features in a dataset? Name any two techniques used for scaling.
    Scaling numerical features in a dataset is important for several reasons:
    Algorithms that use distance measures, such as K-nearest neighbors or K-means, are sensitive to the scale of the features, and may produce different results depending on whether or not the features are scaled.
    Some machine learning algorithms, such as linear regression, gradient boosting, and support vector machines, are sensitive to the scale of the features, and may perform better or worse depending on whether or not the features are scaled.
    Two common techniques used for scaling numerical features in a dataset are:
    Min-Max Scaling: also known as normalization, it scales the feature values to a given range, typically [0, 1].
    Standardization: it scales the feature values so that they have zero mean and unit variance.
    Note that some of the algorithms don't require scaling or it depends on the data and the problem, so it's important to check if scaling is really needed or not.

MACHINE LEARNING

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

Scaling numerical features in a dataset can provide several advantages when optimization using gradient descent algorithm:

Faster convergence: Scaling the features to have similar ranges allows the optimization algorithm to converge faster, as the gradients calculated for each feature will be of a similar scale.

Avoid oscillations: With unscaled features, optimization algorithm can oscillate around the optimal solution, requiring more iterations to converge. Scaling helps to avoid these oscillations and makes the optimization more stable.

Better solution: Scaling can lead to a better solution, as the optimization algorithm can find the global minimum more easily.

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?

Accuracy is not always a good metric to measure the performance of a model in case of a highly imbalanced dataset for a classification problem. The reason is that accuracy only considers the number of correct predictions, but not the distribution of the classes. In a highly imbalanced dataset, even a model that always predicts the majority class will have a high accuracy.

For example, if a dataset has 95% of observations from class A and only 5% from class B, even a model that always predicts class A will have an accuracy of 95%, which may give a false impression of a good model performance.

In such cases, metrics such as precision, recall, F1-score, area under the ROC curve (AUC-ROC) are more suitable to evaluate the model performance as they take into account the distribution of the classes.

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

The F-score, also known as F1-score, is a measure of a model's accuracy that combines precision and recall. It is defined as the harmonic mean of precision and recall, where precision is the number of true positive predictions divided by the number of true positive and false positive predictions, and recall is the number of true positive predictions divided by the number of true positive and false negative predictions.

The mathematical formula for F-score is:

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

Where,

True Positives (TP) are the number of observations correctly classified as the positive class,
False Positives (FP) are the number of observations incorrectly classified as the positive class,
True negatives (TN) are the number of observations correctly classified as the negative class,
False negatives (FN) are the number of observations incorrectly classified as the negative class.

Precision = TP / (TP + FP)

Recall = TP / (TP + FN)

The F1-score is often used as a single-number summary of classifier performance. It is the harmonic mean of precision and recall. It gives more weight to low values, therefore it gives more emphasis on the classifier

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

- **fit() method is used to calculate the internal parameters of the estimator based on the training data. For example, for a preprocessing method, the fit() method will calculate the mean and standard deviation of the data.**
- **transform() method is used to apply the transformation from the fit() method to the data. For example, for a preprocessing method, the transform() method will scale the data using the mean and standard deviation calculated from the fit() method.**
- **fit_transform() method is a combination of fit() and transform() methods. It first fits the estimator to the data and then applies the transformation to the data. This method is useful when the transformation is learned from the data and needs to be applied to the same data.**