MACHINE LEARNING

Machine Learning Assignment

1. Which of the following in sk-learn library is used for hyper parameter tuning? (D)
   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? (D)
   A) Random forest B) Adaboost
   C) Gradient Boosting D) All of the above

3. In machine learning, if in the below line of code: (A)
   *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: (A)
   *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? (C)
   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? (C)
   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? (B)
   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 (B)
   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

MACHINE LEARNING

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)

Gini Index: The Gini Index is computed by the following formula: Gini = 1 - (p1^2 + p2^2), where and p2 represent the proportion of each class. In this case, p1 = 0.4 and p2 = 0.6, so Gini = 1 - (0.4^2 + 0.6^2) = 0.48.

Entropy: The Entropy is calculated using the following formula: Entropy = -p1log2p1 - p2log2p2. In this case, Entropy = -0.4log2(0.4) - 0.6log2(0.6) = 0.97.

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

Ans) There are several advantages of using a random forest over a decision tree, including:

-> Improved accuracy: Random forests typically produce more accurate predictions than individual decision trees because they combine the predictions of multiple trees.

-> Reduced overfitting: Because a random forest is composed of multiple decision trees, each tree is less likely to overfit the training data, resulting in improved generalization to new data.

-> Handling missing values: Random forests can handle missing values in the input data without the need for imputation.

-> Handling outliers: Random forests are less sensitive to outliers in the input data than decision trees.

-> Feature importance: Random forests can provide feature importance measures, which can be used to identify the most important features in the input data.

-> Handle high dimensional data: Random forests are less prone to the curse of dimensionality, they handle high dimensional data well.

-> Robustness: Random forests are more robust to noise in the data and can generalize well from noisy data.

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

Ans) In Data Processing, we try to change the data in such a way that the model can process it without any problems. And Feature Scaling is one such process in which we transform the data into a better version. Feature Scaling is done to normalize the features in the dataset into a finite range.

MACHINE LEARNING

The two widely used scaling techniques are:

MinMax Scaler:

The MinMax scaler is one of the simplest scalers to understand. It just scales all the data between 0 and 1. The formula for calculating the scaled value is-

x_scaled = (x – x_min)/(x_max – x_min)

Thus, a point to note is that it does so for every feature separately. Though (0, 1) is the default range, we can define our range of max and min values as well.

Standard Scaler:

Just like the MinMax Scaler, the Standard Scaler is another popular scaler that is very easy to understand and implement.

For each feature, the Standard Scaler scales the values such that the mean is 0 and the standard deviation is 1(or the variance).

x_scaled = x – mean/std_dev

However, Standard Scaler assumes that the distribution of the variable is normal. Thus, in case, the variables are not normally distributed, we

-> either choose a different scaler
-> or first, convert the variables to a normal distribution and then apply this scaler

Implementing the standard scaler is much similar to implementing a min-max scaler. Just like before, we will first import StandardScaler and then use it to transform our variable.

MACHINE LEARNING

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

Ans) Scaling provides a number of advantages when using gradient descent for optimization. Firstly, it helps to ensure that the cost function is well conditioned, meaning that the gradients are of similar magnitude in different parts of the parameter space. This helps to reduce the chances of the optimization process getting stuck in local minima. Secondly, it helps to speed up the convergence of the optimization process, allowing for larger learning rates and faster convergence times. Lastly, it helps to reduce the computational complexity of the optimization process, as the gradients can be computed more efficiently.

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?

Ans) No, Accuracy is not a good metric to measure the performance of a model in case of a highly imbalanced dataset. This is because the majority class will always be predicted correctly and the minority class will be ignored or misclassified. This will result in an overall high accuracy value, but the model may not be performing as expected. Better metrics to use in this case include Average Precision, F1 Score, Balanced Accuracy, and Quality Measures.

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

Ans) F-Measure provides a way to combine both precision and recall into a single measure that captures both properties.

Alone, neither precision or recall tells the whole story. We can have excellent precision with terrible recall, or alternately, terrible precision with excellent recall. F-measure provides a way to express both concerns with a single score.

Once precision and recall have been calculated for a binary or multiclass classification problem, the two scores can be combined into the calculation of the F-Measure.

The traditional F measure is calculated as follows:

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

This is the harmonic mean of the two fractions. This is sometimes called the F-Score or the F1-Score and might be the most common metric used on imbalanced classification problems.

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

Ans)
-> The fit(data) method is used to compute the mean and std dev for a given feature to be used further for scaling. The fit function computes the formulation to transform the column based on Standard scaling but doesn’t apply the actual transformation. The computation is stored as a fit object. The fit method doesn’t return anything.

-> The transform(data) method is used to perform scaling using mean and std dev calculated using the .fit() method. The transform method takes advantage of the fit object in the fit() method and applies the actual transformation onto the column. So, fit() and transform() is a two-step process that completes the transformation in the second step. Here, Unlike the fit() method the transform method returns the actually transformed array.

MACHINE LEARNING

-> The fit_transform() method does both fits and transform. fit() and transform() is a two-step process, which can be brought down to a one-shot process using the fit_transform method. When the fit_transform method is used, we can compute and apply the transformation in a single step.