Machine Learning Assignment-5

Q1. R-squared or Residual Sum of Squares (RSS) which one of these two is a better measure of goodness of fit model in regression and why?

Ans1.

The residual sum of squares (RSS) is the absolute amount of explained variation, whereas R-squared is the absolute amount of variation as a proportion of total variation.

R-squared is generally a better measure of the goodness of fit for a regression model than the residual sum of squares (RSS).

Q2. What are TSS (Total Sum of Squares), ESS (Explained Sum of Squares) and RSS (Residual Sum of Squares) in regression? Also mention the equation relating these three metrics with each other.

Ans2.

TSS (Total Sum of Squares):

The sum of squares total (SST) or the total sum of squares (TSS) is the sum of squared differences between the observed dependent variables and the overall mean. SST measures the total variability of a dataset, commonly used in regression analysis and ANOVA.

Mathematically, the difference between variance and SST is that we adjust for the degree of freedom by dividing by n–1 in the variance formula.

ESS (Explained Sum of Squares):

The sum of squares due to regression (SSR) or explained sum of squares (ESS) is the sum of the differences between the predicted value and the mean of the dependent variable. In other words, it describes how well our line fits the data.

If SSR equals SST, our regression model perfectly captures all the observed variability, but that’s rarely the case.

RSS (Residual Sum of Squares):

The sum of squares error (SSE) or residual sum of squares (RSS, where residual means remaining or unexplained) is the difference between the observed and predicted values.

Relationship between SSR, SSE, and SST is--

Mathematically, SST = SSR + SSE.

Q3. What is the need of regularization in machine learning?

Ans3.

The primary goal of regularization is to reduce the model's complexity to make it more generalizable to new data, thus improving its performance on unseen datasets.

Q4. What is Gini–impurity index?

Ans4. Gini-impurity index is a measure of how mixed or impure a dataset is. The Gini impurity ranges between 0 and 1, where 0 represents a pure dataset and 1 represents a completely impure dataset. In a pure dataset, all the samples belong to the same class or category.

Q5. Are unregularized decision-trees prone to overfitting? If yes, why?

Ans5. Yes, unregularized decision-trees are prone to overfitting. Overfitting in decision tree models occurs when the tree becomes too complex and captures noise or random fluctuations in the training data, rather than learning the underlying patterns that generalize well to unseen data. Other reasons for overfitting include:

Complexity: Decision trees become overly complex, fitting training data perfectly but struggling to generalize to new data.

Memorizing Noise: It can focus too much on specific data points or noise in the training data, hindering generalization.

Overly Specific Rules: Might create rules that are too specific to the training data, leading to poor performance on new data.

Feature Importance Bias: Certain features may be given too much importance by decision trees, even if they are irrelevant, contributing to overfitting.

Sample Bias: If the training dataset is not representative, decision trees may overfit to the training data's idiosyncrasies, resulting in poor generalization.

Lack of Early Stopping: Without proper stopping rules, decision trees may grow excessively, perfectly fitting the training data but failing to generalize well.

Q6. What is an ensemble technique in machine learning?

Ans6. Ensemble technique refers to a machine learning approach where several models are trained to address a common problem, and their predictions are combined to enhance the overall performance.

Stacking, bagging, and boosting are the three most popular ensemble learning techniques. Each of these techniques offers a unique approach to improving predictive accuracy.

Q7. What is the difference between Bagging and Boosting techniques?

Ans7. As we know, Ensemble learning helps improve machine learning results by combining several models. This approach allows the production of better predictive performance compared to a single model. Basic idea is to learn a set of classifiers (experts) and to allow them to vote. Bagging and Boosting are two types of Ensemble Learning. These two decrease the variance of a single estimate as they combine several estimates from different models. So the result may be a model with higher stability. Let's understand these two terms in a glimpse.

Bagging: It is a homogeneous weak learners' model that learns from each other independently in parallel and combines them for determining the model average.

Boosting: It is also a homogeneous weak learners' model but works differently from Bagging. In this model, learners learn sequentially and adaptively to improve model predictions of a learning algorithm.

Q8. What is out-of-bag error in random forests?

Ans8. Out-of-bag (OOB) error, also called out-of-bag estimate, is a method of measuring the prediction error of random forests, boosted decision trees, and other machine learning models utilizing bootstrap aggregating (bagging). Bagging uses subsampling with replacement to create training samples for the model to learn from.

The out-of-bag (OOB) error is the average error for each calculated using predictions from the trees that do not contain in their respective bootstrap sample. This allows the RandomForestClassifier to be fit and validated whilst being trained

Q9. What is K-fold cross-validation?

Ans9. In K-Fold cross-validation, the input data is divided into 'K' number of folds, hence the name K Fold. The model undergoes training with K-1 folds and is evaluated on the remaining fold. This procedure is performed K times, where each fold is utilized as the testing set one time. The performance metrics are averaged across K iterations to offer a more reliable evaluation of the model's performance.

Example: Suppose we specified the fold as 10 (k = 10), then the K-Fold cross-validation splits the input data into 10 folds, which means we have 10 sets of data to train and test our model. So for every iteration, the model uses one

fold as test data and the remaining as training data (9 folds). Every time, it picks a different fold for evaluation, and the result is an array of evaluation scores for each fold.

Q10. What is hyper parameter tuning in machine learning and why it is done?

Ans10. Hyperparameters are parameters that control the behaviour of a machine-learning model but are not learned during training. Some common examples of hyperparameters include:

Regularization strength: This parameter controls how much the model is penalized for overfitting.

Number of trees: This parameter controls the number of trees in a random forest model.

Learning rate: This parameter controls how quickly the model learns during training.

Tuning hyperparameters is done because it can improve the performance of a training model on new data. For example, a poorly calibrated model will have high bias, meaning it is unsuitable for new data. On the other hand, a well-calibrated model will have bias and high variance, meaning it will extend well to new data and be accurate.

Q11. What issues can occur if we have a large learning rate in Gradient Descent?

Ans11. If the learning rate is too high, the algorithm may overshoot the minimum, and if it is too low, the algorithm may take too long to converge. Overfitting: Gradient descent can overfit the training data if the model is too complex or the learning rate is too high.

Q12. Can we use Logistic Regression for classification of Non-Linear Data? If not, why?

Ans12. We cannot use Logistic Regression for Classification of Non-Linear Data as it assumes a linear relationship between the input features and the output. This means that it cannot capture the complexity and non-linearity of the data.

Q13. Differentiate between Adaboost and Gradient Boosting.

Ans13. The most significant difference is that gradient boosting minimizes a loss function like MSE or log loss while AdaBoost focuses on instances with high error by adjusting their sample weights adaptively.

Gradient boosting models apply shrinkage to avoid overfitting which AdaBoost does not do. Gradient boosting also performs subsampling of the training instances while AdaBoost uses all instances to train every weak learner.

Overall gradient boosting is more robust to outliers and noise since it equally considers all training instances when optimizing the loss function. AdaBoost is faster but more impacted by dirty data since it fixates on hard examples.

Q14. What is bias-variance trade off in machine learning?

Ans14. The bias-variance trade-off is about finding the right balance between simplicity and complexity in a machine learning model. High bias means the model is too simple and consistently misses

the target, while high variance means the model is too complex and shoots all over the place.

Q15. Give short description each of Linear, RBF, Polynomial kernels used in SVM.

Ans15.

Linear kernel:The linear kernel can be defined as:

The dot product of the input vectors is a measure of their similarity or distance in the original feature space. When using a linear kernel in an SVM, the decision boundary is a linear hyperplane that separates the different classes in the feature space.

RBF (Radial Basis Function): The RBF (Radial Basis Function) kernel function is a popular kernel function used in SVM (Support Vector Machine) classification algorithms. It is widely used for its ability to handle non-linearly separable data by mapping the data to higher dimensions.

Polynomial kernels: In machine learning, the polynomial kernel is a kernel function commonly used with support vector machines (SVMs) and other kernelized models, that represents the similarity of vectors (training samples) in a feature space over polynomials of the original variables, allowing learning of non-linear models.