MACHINE LEARNING ASSIGNMENT -5

1. **Q.1) 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?**

   Ans:

   - Both R-squared and Residual Sum of Squares are used as measures of goodness of fit for a model.
   - R-squared, the coefficient of determination, indicates the proportion of variance in the dependent variable that can be explained by the regression equation. It is commonly used because it is relatively easy to interpret; the closer to 1.0, the better the model explains the variance in the data.
   - RSS, on the other hand, is the sum of the squares of the differences between the observed values and the values predicted by the model, minimizing RSS is the objective of the least-squares criteria. However, while RSS provides an absolute measure, it is not comparable across models with different sample sizes or numbers of predictors.
   - Comparing both, R-squared is better and commonly used measure of goodness of fit in regression, indicating the proportion of variance explained by the model, whereas RSS provides an absolute measure of fit but is not as interpretable or comparable across different models.

2. **Q.2) 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?**

   Ans:

   TSS (Total Sum of Squares):

   - TSS tell us how much variation is there in the dependent variable.
   - Total Sum of Squares is the sum of squared differences between the observed dependent variables and the overall mean.
   - TSS measures the total variability of a dataset, commonly used in regression analysis and ANOVA.
   - Formula:

     TSS = $\sum_{i=1}^{n}(y_i - \bar{y})^2$

     Where: y_i– observed dependent variable.

     $\bar{y}$– mean of the dependent variable.

   ESS (Explained Sum of Squares):

   - Explained Sum of Squares is the sum of the differences between the predicted value and the mean of the dependent variable.
   - It describes how well our line fits the data.
   - Formula:

     ESS = $\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2$

     Where: $\hat{y}_i$ – the predicted value of the dependent variable.

     $\hat{y}_i$ – mean of the dependent variable.

RSS (Residual Sum of Squares):

- Residual Sum of Squares is a statistical technique used to measure the amount of variance in a data set that is not explained by a regression model.
- It estimates the variance in the residuals or error term.
- It is the sum of the squared differences between the actual variable and the predicted variable.
- Formula:
  RSS = $\sum_{i=1}^{n}(y_i - f(x_i))^2$
  Where: y_i– ith value of the variable to be predicted.
  $f(x_i)$– predicted value of y_i.

3. **Q.3) What is need of Regularization in machine learning?**
   Ans:
   - Regularization techniques is used in machine learning training model to avoid and reduce the possibility of overfitting or underfitting and help us to obtain an optimal model.
   - Using Regularization, we can fit out machine learning model appropriately on a given test set and hence reduce the errors in it.

4. **Q.4) What is Gini-impurity index?**
   Ans:
   - Gini-impurity index is the measure of the diversity of dataset.
   - Gini-impurity index is an additional approach for dividing a Decision tree.
   - Formula: Gini(P) = 1 - ∑(Px)^2, where P is the proportion of the instances of class x in a set.
   - The range of Gini-impurity index varies from [0,1], where 0 indicates perfect purity and 1 indicates maximum impurity.
   - Gini-impurity index is used in Classification and Regression Trees algorithms.

5. **Q.5) Are unregularized decision-trees prone to overfitting? If yes, why?**
   Ans:
   - Yes unregularized decision-trees is prone to overfitting, because in decision tree's overfitting happens when learning processing in a training set to a point of high granularity that makes them easily overfit. Allowing a decision tree to split to a granular degree is the behaviour of this model that makes it prone to learning every point extremely well, to the point of perfect classification which is overfitting.
6. **Q.6) What is ensemble technique in machine learning?**
   Ans:
   - Ensemble Technique is the technique that enhances accuracy and resilience in forecasting by merging predictions from multiple models.
   - It aims to mitigate errors or biases that may exist in individual models by leveraging the collective intelligence of the ensemble.
   - This technique provides more robust and reliable forecasts.

7. **Q.7) What is the difference between Bagging and Boosting techniques?**
   Ans:
   Bagging Technique:
   - Bagging technique is used to reduce the overfitting or variance of weak learners.
   - This technique is used for combining weak learners of high variance.
   - This technique aims to produce a model with lower variance than the individual weak models. These weak learners are homogeneous, meaning they are of the same type.
   - Bagging technique trains the models in parallel.

   Boosting Technique:
   - Boosting technique is used to reduce the bias or underfitting of weak learners.
   - This technique is used for combining weak learners with high bias.
   - This technique aims to produce a model with a lower bias than that of the individual models. These weak learners are homogeneous.
   - Boosting technique trains the models sequentially.

8. **Q.8) What is out-of-bag error in random forest?**
   Ans:
   - The out-of-bag error is a performance metric that estimates the performance of the Random Forest model using samples not included in the bootstrap sample for training.
   - Out-of-bag error is the number of wrongly classifying the out-of-bag sample.

9. **Q.9) What is K-fold cross-validation?**
   Ans:
   - K-fold cross-validation a technique which is used for evaluating predictive models, where dataset is divided into k folds and then the model is trained and evaluated k times, using a different fold as the validation set each time.
   - The model's generalization performance is estimated after averaging the performance metrics from each fold.

10. **Q.10) What is hyper parameter tuning in machine learning and why it is done?**
    Ans:
    - Hyperparameter tuning is the process of selecting the optimal values fro a machine learning model's hyperparameters.
    - Here, Hyperparameters are configuration variables that are set before the training process of a model begins. These are settings that control the learning process of the model.
    - The goal of hyperparameter tuning is to find the values that lead to the best performance on a given task.

      Hyperparameter Tuning in machine learning is done for:
      - Model selection, choosing the right model architecture for the task.
      - Regularization parameter tuning, controlling model complexity for optimal performance.

- Feature preprocessing optimization, enhancing data quality and model performance.
- Algorithmic parameter tuning, adjusting algorithm-specific parameters for optimal results.

11. **Q.11) What issues can occur if we have a large learning rate in Gradient Descent?**
    Ans:
    The learning rate can significantly impact the performance of gradient descent. If we have a large learning rate, the algorithm may overshoot the minimum, and if it is too low, the algorithm may take too long to converge, i.e. Gradient descent can overfit the training data if the model is too complex or the learning rate is too high. This can lead to poor generalization performance on new data.

12. **Q.12) Can we use Logistic Regression for classification of Non-Linear Data? If not, why?**
    Ans: No, we cannot use Logistic Regression for classification of Non-Linear Data, because 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. And also it is sensitive to outliers and noise, which can affect the accuracy and stability of the model. Logistic regression also has a limited capacity to learn from multiple features, as it can only combine them linearly.

13. **Q.13) Differentiate between Adaboost and Gradient Boosting?**
    Ans:

    Adaboost
    - An additive model where shortcomings of previous models are identified by high-weight data points.
    - The trees are usually grown as decision stumps.
    - Each classifier has different weights assigned to the final prediction based on its performance.
    - It gives weights to both classifiers and observations thus capturing maximum variance within data.

    Gradient Boosting
    - An additive model where shortcomings of previous models are identified by the gradient.
    - The trees are grown to a greater depth usually ranging from 8 to 32 terminal nodes.
    - All classifiers are weighed equally and their predictive capacity is restricted with learning rate to increase accuracy.
    - It builds trees on previous classifier's residuals thus capturing variance in data.

14. **Q.14) What is bias-variance trade off in machine learning?**

Ans:

- The bias-variance trade off concept in machine learning refers to the delicate balance between two sources of error in a predictive model, bias and variance.
- Where bias represents the error due to overly simplistic assumptions in the learning algorithm and variance reflects the model's sensitivity to small fluctuations in the training data.
- Thus bias-variance trade off is used to find the right level of complexity in a model to minimize both bias and variance, achieving good generalization to new data, which helps in building models that perform well on a variety of datasets.

15. **Q.15) Give short description each of Linear, RBF, Polynomial kernels used in SVM?**

Ans:

- Support Vector Machine (SVM) is a powerful supervised algorithm that works best on smaller and complex datasets. SVM can be used for both regression and classification tasks, but generally, they work best in classification problems.
- SVM algorithms use a set of mathematical functions that are defined as the kernel. The function of kernel is to take data as input and transform it into the required form. Different SVM algorithms use different types of kernel functions. Like, linear, nonlinear, polynomial, radial basis function (RBF), and sigmoid.
- **Linear SVM:** When the data is perfectly linearly separable i.e. the data points can be classified into two classes by using a single straight line, only then we can use Linear SVM.
  Formula:
  K(x,y) = x.y
  Where, x and y are the input feature vectors. The dot product of the input vectors is a measure of their similarity or distance in the original feature space.
- **Radial Basis Kernel (RBF):** in SVM is a kernel function that is used in machine learning to find a non-linear classifier or regression line. RBF is a function whose value depends on the Euclidean distance to a centre in the input space.
  Formula: K(x,y) = exp(-gamma * ||x-y||^2)
  Where x and y are the input feature vectors, gamma is a parameter that controls the width of RBF function, and ||x-y||^2 is the squared Euclidean distance between the input vectors.
- **Polynomial Kernel** in SVM is a nonlinear kernel function that employs polynomial functions to transfer the input data into a higher- dimensional feature space.
  Formula:
  K (x, y) = (x.y + c)d
  Where, x and y are the input feature vectors, c is a constant term and d is the degree of polynomial.