Machine Learning Assignment-5 Answers :-

Ans.1 :-

- R-squared is a better measure of goodness of fit model in regression because it representing the proportion of the variance in your data which is explained by your model so the closer to one, the better the fit.

Ans.2 :-

TSS, ESS and RSS:-

TSS:-

- The Total SS (TSS or SST) tells you how much variation there is in the dependent variable.

- The sum of squares total, denoted SST, is the squared differences between the observed *dependent variable* and its mean.

- You can think of this as the dispersion of the observed variables around the mean – much like the variance in descriptive statistics.

- Total SS = $\Sigma(Yi - \text{mean of } Y)^2$.

ESS:-

- The Explained SS tells you how much of the variation in the dependent variable your model explained.

- Explained SS = $\Sigma(Y\text{-Hat} - \text{mean of } Y)^2$.

RSS:-

- The residual sum of squares tells you how much of the dependent variable's variation your model did not explain.

- It is the sum of the squared differences between the actual Y and the predicted Y

- Residual Sum of Squares = Σ e2

Ans.3 :-

Regularization:-

- Regularization is a technique to prevent the model from overfitting by adding extra information to it.

- Sometimes the machine learning model performs well with the training data but does not perform well with the test data.

- In Machine Learning we often divide the dataset into training and test data, the algorithm while training the data can either

  1) learn the data too well, even the noises which is called over fitting
  2) do not learn from the data, cannot find the pattern from the data which is called under fitting.

- Now, both over fitting and underfitting are problems one need to address while building models.

- Regularization in Machine Learning is used to minimize the problem of overfitting, the result is that the model generalizes well on the unseen data once overfitting is minimized.

- To avoid overfitting, regularization discourages learning a more sophisticated or flexible model. Regularization will try to minimize a loss function by inducing penalty.

- For Example,
  The residual sum of squares is our optimization function or loss function in simple linear regression (RSS).

$$\mathrm{RSS} = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 .$$

Here ,

y is the dependent variable,
x_1, x_2, x_3,.....x_n are independent variables.
b_0, b_1, b_2..... b_n, are the coefficients estimates for different variables of x, these can also be called weights or magnitudes

Regularization will shrink these coefficients towards Zero,
Minimizing the loss means less error and model will be a good fit.

The way regularization can be done is by

1) RIDGE also known as L-2 Regularization.
2) LASSO (Least Absolute and Selection Operator) also known as L-1 Regularization.

Ans.4 :-

Gini Impurity:-

- Gini Impurity is a measurement used to build Decision Trees to determine how the features of a dataset should split nodes to form the tree.

- The Gini Impurity of a dataset is a number between 0-0.5

- Which indicates the likelihood of new, random data being misclassified if it were given a random class label according to the class distribution in the dataset.

- For example, say you want to build a classifier that determines if someone will default on their credit card.

- You have some labelled data with features, such as bins for age, income, credit rating, and whether or not each person is a student.

- To find the best feature for the first split of the tree – the root node – you could calculate how poorly each feature divided the data into the correct class, default ("yes") or didn't default ("no").

- This calculation would measure the impurity of the split, and the feature with the lowest impurity would determine the best feature for splitting the current node.

- This process would continue for each subsequent node using the remaining features.

Ans.5 :-

decision-trees prone to overfitting? If yes then why?

- Out of all machine learning techniques, decision trees are amongst the most prone to overfitting.

- No practical implementation is possible without including approaches that mitigate this challenge.

- In this module, through various visualizations and investigations, you will investigate why decision trees suffer from significant overfitting problems.

- Using the principle of Occam's razor, you will mitigate overfitting by learning simpler trees.

- At first, you will design algorithms that stop the learning process before the decision trees become overly complex.

- In an optional segment, you will design a very practical approach that learns an overly-complex tree, and then simplifies it with pruning.

- Your implementation will investigate the effect of these techniques on mitigating overfitting on our real-world loan data set.

Ans.6 :-

Ensemble Technique:-

- Ensemble methods are techniques that create multiple models and then combine them to produce improved results.

- Ensemble methods usually produces more accurate solutions than a single model would.

- This has been the case in a number of machine learning competitions, where the winning solutions used ensemble methods.

- Ensemble techniques are classified into three types:
    1) Bagging
    2) Boosting
    3) Stacking

Ans.7 :-

Difference between Bagging and Boosting:-

Bagging:-

- The simplest way of combining predictions that belong to the same type.
- Aim to decrease variance, not bias.
- Each model receives equal weight.
- Each model is built independently.
- Different training data subsets are selected using row sampling with replacement and random sampling methods from the entire training dataset.
- Bagging tries to solve the over-fitting problem.
- If the classifier is unstable (high variance), then apply bagging.
- In this base classifiers are trained parallelly.
- Example: The Random forest model uses Bagging.

Boosting:-

- A way of combining predictions that belong to the different types.
- Aim to decrease bias, not variance.
- Models are weighted according to their performance.
- New models are influenced by the performance of previously built models.
- Every new subset contains the elements that were misclassified by previous models.
- Boosting tries to reduce bias.
- If the classifier is stable and simple (high bias) the apply boosting.
- In this base classifiers are trained sequentially.
- Example: The AdaBoost uses Boosting techniques.

Ans.8 :-

Out-Of-Bag error:-

- 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.
- OOB error is the mean prediction error on each training sample x_i, using only the trees that did not have x_i in their bootstrap sample.

- The RandomForestClassifier is trained using *bootstrap aggregation*, where each new tree is fit from a bootstrap sample of the training observations zi=(xi,yi).

- The *out-of-bag* (OOB) error is the average error for each zi calculated using predictions from the trees that do not contain zi in their respective bootstrap sample.

- This allows the RandomForestClassifier to be fit and validated whilst being trained.

Ans.9 :-

K-fold Cross-Validation:-

- K-fold Cross-Validation is when the dataset is split into a K number of folds and is used to evaluate the model's ability when given new data.

- K refers to the number of groups the data sample is split into.
- For example, if you see that the k-value is 5, we can call this a 5-fold cross-validation.

- Each fold is used as a testing set at one point in the process.

- K-fold Cross-Validation Process:-

  1. Choose your k-value
  2. Split the dataset into the number of k folds.
  3. Start off with using your k-1 fold as the test dataset and the remaining folds as the training dataset
  4. Train the model on the training dataset and validate it on the test dataset
  5. Save the validation score

6. Repeat steps 3 – 5, but changing the value of your k test dataset. So we chose k-1 as our test dataset for the first round, we then move onto k-2 as the test dataset for the next round.
7. By the end of it you would have validated the model on every fold that you have.
8. Average the results that were produced in step 5 to summarize the skill of the model.

Ans.10 :-

Hyper parameter tuning and why it is does?:-

- A Machine Learning model is defined as a mathematical model with a number of parameters that need to be learned from the data.
- By training a model with existing data, we are able to fit the model parameters.
- However, there is another kind of parameter, known as Hyperparameters, that cannot be directly learned from the regular training process.
- They are usually fixed before the actual training process begins.
- These parameters express important properties of the model such as its complexity or how fast it should learn.
- Hyperparameter tuning consists of finding a set of optimal hyperparameter values for a learning algorithm while applying this optimized algorithm to any data set.
- That combination of hyperparameters maximizes the model's performance, minimizing a predefined loss function to produce better results with fewer errors.

- Some examples of model hyperparameters include:
    1. The penalty in Logistic Regression Classifier i.e. L1 or L2 regularization
    2. The learning rate for training a neural network.
    3. The C and sigma hyperparameters for support vector machines.
    4. The k in k-nearest neighbors.

- Models can have many hyperparameters and finding the best combination of parameters can be treated as a search problem. The two best strategies for Hyperparameter tuning are:

    1. GridSearchCV
    2. RandomizedSearchCV

Ans.11 :-

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

- If a learning rate that is too large can cause the model to converge too quickly to a suboptimal solution.

- we may never converge to a local minimum because we overshoot it every time.

- If we are lucky and the algorithm converges anyway, it still might take more steps than it needed.

- It can cause undesirable divergent behaviour in your loss function.

Ans.12 :-

- We cannot use Logistic Regression for classification of Non-Linear Data.

Ans.13 :-

Differentiate between Adaboost and Gradient Boosting:-

Loss Function:-

- In case of Adaptive Boosting or AdaBoost, it minimises the exponential loss function that can make the algorithm sensitive to the outliers.

- With Gradient Boosting, any differentiable loss function can be utilised. Gradient Boosting algorithm is more robust to outliers than AdaBoost.

Flexibility:-

- AdaBoost is the first designed boosting algorithm with a particular loss function.

- Gradient Boosting is a generic algorithm that assists in searching the approximate solutions to the additive modelling problem. This makes Gradient Boosting more flexible than AdaBoost.

Benefits:-

- AdaBoost minimises loss function related to any classification error and is best used with weak learners. The method was mainly designed for binary classification problems and can be utilised to boost the performance of decision trees.

- Gradient Boosting is used to solve the differentiable loss function problem. The technique can be used for both classification and regression problems.

Shortcomings:-

- In the case of Gradient Boosting, the shortcomings of the existing weak learners can be identified by gradients.

- With AdaBoost, it can be identified by high-weight data points.

Ans.14 :-

Bias:-

- The bias is known as the difference between the prediction of the values by the ML model and the correct value.

- Being high in biasing gives a large error in training as well as testing data.

- It's recommended that an algorithm should always be low biased to avoid the problem of underfitting.

- By high bias, the data predicted is in a straight line format, thus not fitting accurately in the data in the data set.

- Such fitting is known as Underfitting of Data. This happens when the hypothesis is too simple or linear in nature.

Variance:-

- The variability of model prediction for a given data point which tells us spread of our data is called the variance of the model.

- The model with high variance has a very complex fit to the training data and thus is not able to fit accurately on the data which it hasn't seen before.

- As a result, such models perform very well on training data but has high error rates on test data.
- When a model is high on variance, it is then said to as Overfitting of Data.

- Overfitting is fitting the training set accurately via complex curve and high order hypothesis but is not the solution as the error with unseen data is high.

Bias-Variance trade off:-

- It is important to understand prediction errors (bias and variance) when it comes to accuracy in any machine learning algorithm.

- There is a trade off between a model's ability to minimize bias and variance which is referred to as the best solution for selecting a value of Regularization constant.

- Proper understanding of these errors would help to avoid the overfitting and underfitting of a data set while training the algorithm.

- If the algorithm is too simple (hypothesis with linear eq.) then it may be on high bias and low variance condition and thus is error-prone.

- If algorithms fit too complex (hypothesis with high degree eq.) then it may be on high variance and low bias

- In the latter condition, the new entries will not perform well. Well, there is something between both of these conditions, known as Trade-off or Bias Variance Trade-off.

- This trade off in complexity is why there is a trade off between bias and variance.

- An algorithm can't be more complex and less complex at the same time.

Ans.15 :-

Linear, RBF, Polynomial kernels used in SVM:-

Linear kernel:-

- Linear Kernel is used when the data is Linearly separable, that is, it can be separated using a single Line.

- It is one of the most common kernels to be used.

- It is mostly used when there are a Large number of Features in a particular Data Set.
- One of the examples where there are a lot of features, is Text Classification, as each alphabet is a new feature.

- So, we mostly use Linear Kernel in Text Classification.

Polynomial kernel:-

- 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.

- The polynomial kernel looks not only at the given features of input samples to determine their similarity, but also combinations of these.

- In the context of regression analysis, such combinations are known as interaction features.

- The (implicit) feature space of a polynomial kernel is equivalent to that of polynomial regression, but without the combinatorial blow up in the number of parameters to be learned.

- When the input features are binary-valued (Booleans), then the features correspond to logical conjunctions of input features.

RBF kernel:-

- In machine learning, the radial basis function kernel, or RBF kernel, is a popular kernel function used in various kernelized learning algorithms.

- In particular, it is commonly used in support vector machine classification.