**ASSIGNMENT – 5**

# MACHINE LEARNING

**Q1 to Q15 are subjective answer type questions, Answer them briefly.**

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?**

In a regression, r squared is a measure of goodness of fit. It tells us how much of the variance in the dependent variable is explained by the independent variables. So, higher the r-squared value, better is the model fit. However, if you perform a regression, and go on adding independent variables, it will observe that the r-squared value increases, which is intuitive. This happens due to multicollinearity. This increase in r-squared value is a taken care of by adjusted r-squared, thereby making it a better metric to assess model performance.

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.**

**TSS:** This gives you the distance from the linear line drawn to each particular variable. You could also describe TSS as the dispersion of observed variables around the mean, or the variance. So, the goal of TSS is to measure the total variability of the dataset.

**ESS:** ESS is the sum of the squares of the deviations of the predicted values from the mean value of a response variable, in a standard regression model

**RSS**: Residual sum of squares (RSS), also known as the sum of squared residuals (SSR) or the sum of squared errors of prediction (SSE), is the sum of the squares of residuals (deviations of predicted from actual empirical values of data)

**3. What is the need of regularization in machine learning?**

One of the major aspects of training your machine learning model is avoiding overfitting. *The model will have a low accuracy if it is overfitting.* This happens because your model is trying too hard to capture the noise in your training dataset. ***By noise we mean the data points that don't really represent the true properties of your data, but random chance***. Learning such data points, makes your model more flexible, at the risk of overfitting.***The concept of balancing bias and variance, is helpful in understanding the phenomenon of overfitting.***

**4. What is Gini–impurity index?**

Gini Index, also known as Gini impurity, calculates the amount of probability of a specific feature that is classified incorrectly when selected randomly.

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

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

Decision trees are prone to overfitting, especially when a tree is particularly deep. This is due to the amount of specificity we look at leading to smaller sample of events that meet the previous assumptions. This small sample could lead to unsound conclusions

**6. What is an ensemble technique in machine learning?**

Ensemble methods are **techniques that aim at improving the accuracy of results in models by combining multiple models instead of using a single model**. The combined models increase the accuracy of the results significantly. This has boosted the popularity of ensemble methods in machine learning

**7. What is the difference between Bagging and Boosting techniques?**

| Sl No | Bagging | Boosting |
|---|---|---|
| **1** | Various training data subsets are randomly drawn with replacement from the whole training dataset. | Each new subset contains the components that were misclassified by previous models |
| **2** | Bagging attemps to tackle the over –fitting issue. | Boosting tries to reduce bias. |
| **3** | If the classifier is unstable (high variance), then we need to apply bagging. | If the classifier is steady and straightforward (high bias), then we need to apply boosting. |
| **4.** | Every model received an equal weight | Models are weighted by their performance |
| **5.** | Objective to decrease variance, not bias. | Objective to decrease bias, not variance. |
| **6.** | It is the easiest way of connecting predictions that belong to the same type. | It is the way of connecting predictions that belong to the different types. |
| **7.** | Every model is constructed independently. | New models are affected by the performance of the previously developed model. |

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

**An error estimate is made for cases that were not used when constructing the tree**. This is called an out-of-bag(OOB) error estimate mentioned as a percentage. The decision trees are prone to overfitting, and this is the main drawback of it.

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 xi, using only the trees that did not have xi in their bootstrap sample

**9. What is K-fold cross-validation?**

Cross validation is an evaluation method used in machine learning to find out how well your machine learning model can predict the outcome of unseen data. It is a method that is easy to comprehend, works well for a limited data sample and also offers an evaluation that is less biased, making it a popular choice

1. First, shuffle the dataset and split into k number of subsamples. (It is important to try to make the subsamples equal in size and ensure k is less than or equal to the number of elements in the dataset).

2. In the first iteration, the first subset is used as the test data while all the other subsets are considered as the training data.

3. Train the model with the training data and evaluate it using the test subset. Keep the evaluation score or error rate, and get rid of the model.

4. Now, in the next iteration, select a different subset as the test data set, and make everything else (including the test set we used in the previous iteration) part of the training data.

5. Re-train the model with the training data and test it using the new test data set, keep the evaluation score and discard the model.

6. Continue iterating the above k times. Each data subsamples will be used in each iteration until all data is considered. You will end up with a k number of evaluation scores.

7. The total error rate is the average of all these individual evaluation scores.

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

Hyperparameter tuning consists of **finding a set of optimal hyperparameter values for a learning algorithm while applying this optimized algorithm to any data set**.

Hyperparameter tuning takes advantage of the processing infrastructure of Google Cloud **to test different hyperparameter configurations when training your model**. It can give you optimized values for hyperparameters, which maximizes your model's predictive accuracy.

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

Gradient Descent is too sensitive to the learning rate. If it is too big, **the algorithm may bypass the local minimum and overshoot**.

A learning rate that is too small may never converge or may get stuck on a suboptimal solution. When the learning rate is too large, gradient descent can inadvertently increase rather than decrease the training error

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

**Logistic regression is neither linear nor is it a classifier**. The idea of a "decision boundary" has little to do with logistic regression, which is instead a direct probability estimation method that separates predictions from decision

In order for Gradient Descent to work, we must set the learning rate to an appropriate value. This parameter determines how fast or slow we will move towards the optimal weights. If the learning rate is very large we will **skip the optimal solution**

**13. Differentiate between Adaboost and Gradient Boosting.**

| **Ada Boost** | **Gradient Boost** |
|---|---|
| Ada Boos and Gradient Boost use a base weak learner and they try to boost the performance of a weak learner by iteratively shifting the focus towards problematic observations that were difficult to predict .At the end , a strong learner is formed by addition of the weak learner. | |

| | |
|---|---|
| Shift is done by up weighting observations that were misclassified before. | Gradient boost identifies difficult observation by large residuals computed in the previous iterations. |
| In Ada Boost "shortcoming " are identified by high weight data point | In Gradient boost "short coming are identified by gradients. |
| Exponential loss of Ada Boost gives more weight for those sample fitled worse. | Gradient boost further dissect error components to bring in more explanation |

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

Whenever we discuss model prediction, it's important to understand prediction errors (bias and variance). There is a tradeoff between a model's ability to minimize bias and variance. Gaining a proper understanding of these errors So let's start with the basics and see how they make difference to our machine learning Models would help us not only to build accurate models but also to avoid the mistake of overfitting and underfitting.

So let's start with the basics and see how they make difference to our machine learning Models.

Bias is the difference between the average prediction of our model and the correct value which we are trying to predict. Model with high bias pays very little attention to the training data and oversimplifies the model. It always leads to high error on training and test data

Variance is the variability of model prediction for a given data point or a value which tells us spread of our data

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

**Unlike linear or polynomial kernels**, RBF is more complex and efficient at the same time that it can combine multiple polynomial kernels multiple times of different degrees to project the non-linearly separable data into higher dimensional space so that it can be separable using a hyperplane.

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

SVM or Support Vector Machine is a linear model for classification and regression problems. It can solve linear and non-linear problems and work well for many practical problems. The idea of SVM is simple: **The algorithm creates a line or a hyperplane which separates the data into classes**