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 ofgoodness of fit model in regression and why?**

Ans:

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.

The residual sum of squares (RSS) is a statistical technique used to measure the amount of variance in a data set that is not explained by a regression model itself. Instead, it estimates the variance in the residuals, or error term.

- The residual sum of squares (RSS) measures the level of variance in the error term, or residuals, of a regression model.
- The smaller the residual sum of squares, the better your model fits your data; the greater the residual sum of squares, the poorer your model fits your data.
- A value of zero means your model is a perfect fit.
- Statistical models are used by investors and portfolio managers to track an investment's price and use that data to predict future movements.
- The RSS is used by financial analysts in order to estimate the validity of their econometric models.

R squared, the proportion of variation in the outcome Y, explained by the covariates X, is commonly described as a measure of goodness of fit. This of course seems very reasonable, since R squared measures how close the observed Y values are to the predicted (fitted) values from the model.

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

Ans:

TSS: The total sum of squares (TSS or SST) tells you how far the data points in a dataset are from the center. It's a descriptive statistic called a measure of spread or dispersion. Dividing the TSS by the number of observations in the dataset gives you the average variability within the data, which is called the variance

ESS: The explained sum of squares (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 — for example, $y_i = a + b_1x_{1i} + b_2x_{2i} + ...$

RSS: The residual sum of squares (RSS) is a statistical technique used to measure the amount of variance in a data set that is not explained by a regression model itself. Instead, it estimates the variance in the residuals, or error term.

The value estimated by the regression line .
total sum of squares (TSS) = explained sum of squares (ESS) + residual sum of squares (RSS).

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

Ans: Regularization refers to techniques that are used to calibrate machine learning models in order to minimize the adjusted loss function and prevent overfitting or underfitting. Using Regularization, we can fit our machine learning model appropriately on a given test set and hence reduce the errors in it

4. What is Gini–impurity index?

Ans: 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.

Gini impurity is:

Gini impurity=1-Gini

Here Gini denotes the purity and hence Gini impurity tells us about the impurity of nodes. Lower the Gini impurity we can safely infer the purity will be more and hence a higher chance of the homogeneity of the nodes.

Gini works only in those scenarios where we have **categorical** targets. It does not work with continuous targets.

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

Ans: 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.

Overfitting can be identified by **checking validation metrics such as accuracy and loss**. The validation metrics usually increase until a point where they stagnate or start declining when the model is affected by overfitting

6. What is an ensemble technique in machine learning?

Ans: Ensembling is nothing but the technique to combine several individual predictive models to come up with the final predictive model.

Simple Ensemble Techniques

1. Max Voting
2. Averaging
3. Weighted Averaging

1. Max Voting

The max voting method is generally used for classification problems. In this technique, multiple models are used to make predictions for each data point. The predictions by each model are considered as a 'vote'. The predictions which we get from the majority of the models are used as the final prediction.

2. Averaging

Similar to the max voting technique, multiple predictions are made for each data point in averaging. In this method, we take an average of predictions from all the models and use it to make the final prediction. Averaging can be used for making predictions in regression problems or while calculating probabilities for classification problems.

3. Weighted Average

This is an extension of the averaging method. All models are assigned different weights defining the importance of each model for prediction. For instance, if two of your colleagues are critics, while others have no prior experience in this field, then the answers by these two friends are given more importance as compared to the other people.

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

Ans: Bagging tries to solve the over-fitting problem. Boosting tries to reduce bias.

Bagging	Boosting
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.
Bagging attempts to tackle the over-fitting issue.	Boosting tries to reduce bias.
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.
Every model receives an equal weight.	Models are weighted by their performance.
Objective to decrease variance, not bias.	Objective to decrease bias, not variance.
It is the easiest way of connecting predictions that belong to the same type.	It is a way of connecting predictions that belong to the different types.
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?

Ans: 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

9. What is K-fold cross-validation?

Ans: 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.

In each set (fold) training and the test would be performed precisely once during this entire process. It helps us to avoid overfitting. As we know when a model is trained using all of the data in a single short and give the best performance accuracy. To resist this k-fold cross-validation helps us to build the model is a generalized one.

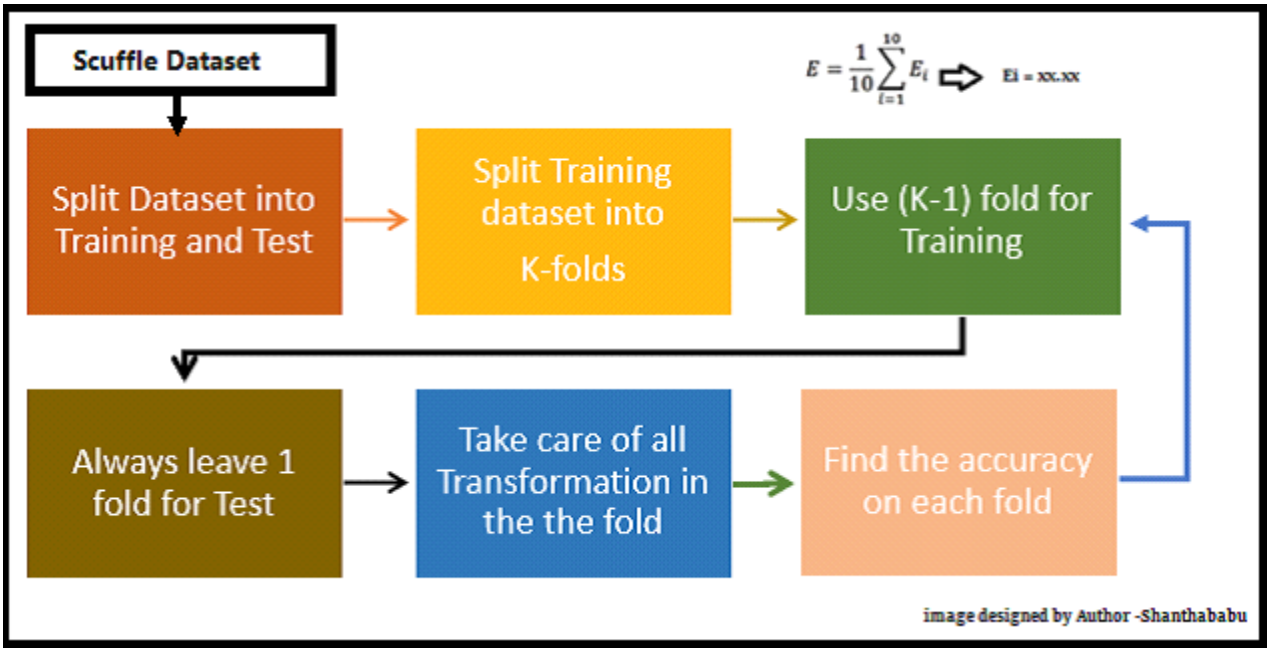
To achieve this K-Fold Cross Validation, we have to split the data set into three sets, Training, Testing, and Validation, with the challenge of the volume of the data.

Here Test and Train data set will support building model and hyperparameter assessments.

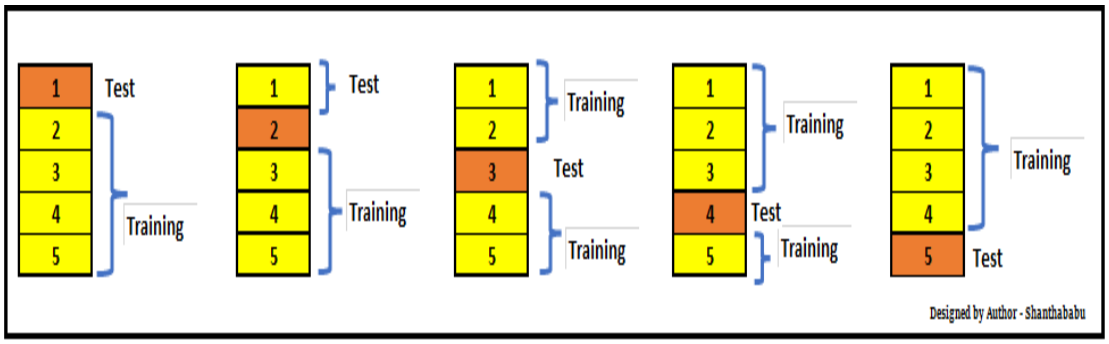
In which the model has been validated multiple times based on the value assigned as a parameter and which is called K and it should be an INTEGER.

Make it simple, based on the K value, the data set would be divided, and train/testing will be conducted in a sequence way equal to K time.

Life Cycle of K-Fold Cross-Validation



Let's have a generalised K value. If K=5, it means, in the given dataset and we are splitting into 5 folds and running the Train and Test. During each run, one fold is considered for testing and the rest will be for training and moving on with iterations, the below pictorial representation would give you an idea of the flow of the fold-defined size.



In which each data point is used, once in the hold-out set and K-1 in Training. So, during the full iteration at least once, one fold will be used for testing and the rest for training.

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

Ans: ***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.
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:

GridSearchCV
RandomizedSearchCV

Why is hyper parameter tuning done?

Hyperparameters are parameters whose values control the learning process and determine the values of model parameters that a learning algorithm ends up learning.
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?

Ans: A learning rate that is too large can cause the model to converge too quickly to a suboptimal solution, whereas a learning rate that is too small can cause the process to get stuck.

Learning rate is used to scale the magnitude of parameter updates during gradient descent. The choice of the value for learning rate can impact two things: 1) how fast the algorithm learns and 2) whether the cost function is minimized or not.

Gradient Descent is too sensitive to the learning rate. If it is too big, the algorithm may bypass the local minimum and overshoot. If it too small, it might increase the total computation time to a very large extent.

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

Ans: .
Logistic Regression should not be used if the number of observations is lesser than the number of features, otherwise, it may lead to overfitting.

By using Logistic Regression, non-linear problems can't be solved because it has a linear decision surface.

13. Differentiate between Adaboost and Gradient Boosting.

Ans:

AdaBoost is the first designed boosting algorithm with a particular loss function. On the other hand, 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

Gradient boosting	Adaptive Boosting
This approach trains learners based upon minimising the loss function of a learner (i.e., training on the residuals of the model)	This method focuses on training upon misclassified observations. Alters the distribution of the training dataset to increase weights on sample observations that are difficult to classify.
Weak learners are decision trees constructed in a greedy manner with split points based on purity scores (i.e., Gini, minimise loss). Thus, larger trees can be used with around 4 to 8 levels. Learners should still remain weak and so they should be constrained (i.e., the maximum number of layers, nodes, splits, leaf nodes)	The weak learners incase of adaptive boosting are a very basic form of decision tree known as stumps.
All the learners have equal weights in the case of gradient boosting. The weight is usually set as the learning rate which is small in magnitude.	The final prediction is based on a majority vote of the weak learners' predictions weighted by their individual accuracy.

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

Ans: **Bias** and **Variance** are the **core parameters** to tune while training a Machine Learning model.

prediction errors can be decomposed into two main subcomponents: error due to **bias,** and error due to **variance.**

Errors due to Bias

An error due to ***Bias*** is the ***distance between the predictions*** of a model and the ***true values***. In this type of error, the model pays little attention to training data and ***oversimplifies*** the model and doesn't learn the patterns. The model **learns the wrong relations** by **not taking in account all the features**

Errors due to Variance

Variability of model prediction for a given data point or a value that ***tells us the spread of our data***. In this type of error, the model pays a l***ot of attention in training data***, to the point to memorize it instead of learning from it. A model with a high error of variance is not flexible to generalize on the data which it hasn't seen before.

Bias-variance trade-off is **tension between the error introduced by the bias and the error produced by the variance**. To understand how to make the most of this trade-off and avoid underfit or overfit our model,

Bias — Variance Trade-Off

Bias- Variance trade-off is about balancing and ***about finding a sweet spot*** between error due to bias and errors due to *variance*.

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

Ans: The linear, polynomial and RBF or Gaussian kernel are simply different in case of making the hyperplane decision boundary between the classes. The kernel functions are used to map the original dataset (linear/nonlinear) into a higher dimensional space with view to making it linear dataset..

Linear SVM: Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.

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.

Polynomial kernel: The polynomial kernel is a general representation of kernels with a degree of more than one. It's useful for image processing.