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- R-squared is a better to measure goodness of fit of model in regression as compare to RSS, because R- Squared is a statistical measure of fit that indicates how much variation of a dependent variable is explained by the independent variables in a regression model. In investing, R-squared is generally explained as the percentage of a fund or security's movements that can be explained by movements in a benchmark index. An R-squared of 100% means that all movements of a security or dependent variable are completely explained by movements in the index or independent variable. If R-squared values between 100-70% it means strongly correlated. When below 70% it means correlation going to break.

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 sum of squares total, denoted SST or TSS, 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.

It is a measure of the total variability of the dataset.

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

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, yi = a + b1x1i + b2x2i + ..., then it is the i th predicted value of the response variable.

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

RSS: The residual sum of squares (RSS) is the sum of the squared distances between your actual versus your predicted values and the question is the

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

Where:
yi – the observed value

$\bar{y}$ – the mean value of a sample

$\hat{y}i$ – the value estimated by the regression line

The actual number we get depends largely on the scale of your response variable.
The relationship between the three types of sum of squares can be summarized by the following equation:

TSS = RSS + ESS

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

ANS: Regularization is a technique used to reduce the errors by fitting the function appropriately on the given training set and avoid overfitting. The commonly used regularisation techniques are:

1) L1 regularization
2) L2 regularization
3) Dropout regularization

4. What is Gini–impurity index?

ANS:- Gini index or Gini impurity measures the degree or probability of a particular variable being wrongly classified when it is randomly chosen. But what is actually meant by 'impurity' If all the elements belong to a single class, then it can be called pure. The degree of Gini index varies between 0 and 1, where 0 denotes that all elements belong to a certain class or if there exists only one class, and 1 denotes that the elements are randomly distributed across various classes. A Gini Index of 0.5 denotes equally distributed elements into some classes.

Formula: $Gini=1-(p1,p2,p3,p4,p5.....pn)^2$
where: p=is the probability of an object being classified to a particular class.

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

ANS:-Yes, 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.
This is the disadvantage of the decision tree.

6. What is an ensemble technique in machine learning?

ANS: - Ensemble learning combines the predictions from multiple models to reduce the variance of predictions and reduce generalization error. Techniques for ensemble learning can be grouped by the element that is varied, such as training data, the model, and how predictions are combined.
Type of ensemble methods is,

1) Bagging
2) Boosting

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

ANS:- Bagging and Boosting are similar in that they are both ensemble techniques, where a set of weak learners are combined to create a strong learner that obtains better performance that a single one.

Bagging	Boosting
Simplest way of combining predictions that belongs to the same type.	A way of combining predictions that belong to the different types
Aim to decrease variance, not bias.	Aim to decrease bias not variance.
Each model is built independently.	New models are influenced by performance of previously built models
Different training data subsets are randomly drawn with replacement from the entire training dataset	Every new subset contains the elements that were misclassified by previous model
Bagging tries to solve over-fitting problem	Boosting tries to reduce bias.
If the classifier is unstable, then apply bagging.	If the classifier is stable and simple, then apply boosting

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

ANS:- Out of bag (OOB) score is a way of validating the Random forest model. Below is a simple intuition of how is it calculated followed by a description of how it is different from validation score and where it is advantageous.
In an ideal case, about 36.8 % of the total training data forms the OOB sample. This can be shown as follows.
If there are N rows in the training data set. Then, the probability of not picking a row in a random draw is (N-1)/N
Using sampling-with-replacement the probability of not picking N rows in random draws is $((N-1) / N)^N$
which in the limit of large N becomes equal to $\lim_{N\to\infty} (1 - (1/N))^N = e^{-1} = 0.368$
Therefore, about 36.8 % of total training data are available as OOB sample for each DT and hence it can be used for evaluating or validating the random forest model.

9. What is K-fold cross-validation?

ANS:-Cross-validation is a resampling procedure used to evaluate machine learning models on a limited data sample. The procedure has a single parameter called k that refers to the number of groups that a given data sample is to be split into. As such, the procedure is often called k-fold cross-validation.
When a specific value for k is chosen, it may be used in place of k in the reference to the model, such as k=10 place of k in the reference to the model, such as k=10 becoming 10-fold cross-validation.

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

ANS:- 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 parameters, 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.
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.

5. The aim of this article is to explore various strategies to tune hyperparameter for Machine learning model.

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

1) Grid Search CV
2) Randomized Search CV

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

ANS:- Gradient Descent is an optimization algorithm used for minimizing the cost function in various machine learning algorithms. It is basically used for updating the parameters of the learning model. When the learning rate is too large, gradient descent can inadvertently increase rather than decrease the training error. […] When the learning rate is too small, training is not only slower, but may become permanently stuck with a high training error.

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

ANS:- Logistic Regression has traditionally been used as a linear classifier, i.e. when the classes can be separated in the feature space by linear boundaries. That can be remedied however if we happen to have a better idea as to the shape of the decision boundary. Logistic regression has traditionally been used to come up with a hyperplane that separates the feature space into classes. But if we suspect that the decision boundary is nonlinear we may get better results by attempting some nonlinear functional forms for the logit function.

13. Differentiate between Ad boost and Gradient Boosting.

Gradient boosting	AdaBoost
This approach trains learners based upon minimizing 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, minimize 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.
In Gradient boost "shortcomings" are identified by gradients	In AdaBoost "shortcomings "are identified by high-weight data points
Gradient boost further dissects error components to bring in more explanation	Exponential loss of AdaBoost gives more weights for those samples fitted worse

Concepts of Gradients are more general in nature	AdaBoost is considered as a special case of Gradient boost in terms of loss function, in which exponential losses.

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

ANS:- Bias:- 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:- Variance is the variability of model prediction for a given data point or a value which tells us spread of our data. Model with high variance pays a lot of attention to training data and does not generalize 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.

Bias-Variance trade off :- If our model is too simple and has very few parameters then it may have high bias and low variance. On the other hand if our model has large number of parameters then it's going to have high variance and low bias. So we need to find the right/good balance without overfitting and underfitting the data. This tradeoff in complexity is why there is a tradeoff between bias and variance. An algorithm can't be more complex and less complex at the same time

$$\text{Total error} = \text{Bias}^2 + \text{Variance} + \text{Irreducible error}$$

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

Ans:- SVM algorithm use asset 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 kernel functions. Some of these functions can be different types are as follows

1. Linear
2. Polynomial
3. Radial basis function (RBF)

Kernel Rules:- K(Xn, Xi), which transforms the original data space into a new space with a higher dimension.

$K(Xn, Xi) = \varphi(Xn)\varphi(Xi)$.....(1)

$$f(Xi) = \sum_{n=1}^{N} \alpha n\, yn\, K(Xn, Xi) + b \quad \text{.....(2)}$$

The aim is the data, which already transformed into a higher dimension, can be separated easily. Thus the hyperplane function is written in Equation (2)

C = cost, γ = gamma, r= coefficient, d= degree

Linear	RBF	Polynomial
It is useful when data is Linearly separable, that is, it can be separated using a single line. It is one of the most common ernes to be used	It is a general purpose kernel, used when there is no prior knowledge about the data. It is also called Gaussian function.	It is commonly used with SVM and other kernelized models, that represents the similarity of vectors (training samples) in a features space over polynomial of the original variables allowing learning of nonlinear models.
Formula:- $K(Xn, Xi) = (Xn, Xi)$	Formula:- $K(Xn, Xi) = \exp(-\gamma \lVert Xn - Xi \rVert^2 + C)$	Formula:- $K(Xn, Xi) = (\gamma (Xn, Xi) + r)^d$
Optimization Parameter:- C & γ Optimal pair value :- $C = 2^{-5}$ $\gamma = 2^{-10}$ r = N/A d = N/A Classification error : - 0.17	Optimization Parameter:- C & γ Optimal pair value:- $C = 2^{-1}$ $\gamma = 2^{-3}$ r = N/A d = N/A Classification error :- 0.15	Optimization Parameter:- C, γ, r & d Optimal pair value :- $C = 2^{-8}$ $\gamma = 2^{-1}$ $r = 2^2$ d = 3 Classification error :- 0.12