MACHINE LEARNING

Ans1: The Residual Sum of Squares (RSS) is a statistical technique used to measure the amount of Variance in a data set. 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. Where as R-Squared is statistical measure in a regression model that determines the proportion of variance in the dependent variable that can be explained by the independent variable. In other word, R-squared shows how well the data fit the regression model the goodness of fit. A higher R-squared value indicates a higher amount of variability being explained by our model and vice-versa. If we had a really low RSS value, it would mean that the regression line was very close to the actual points. This means the independent variables explain the majority of variation in the target variable. The residual sum of squares is the absolute amount of explained variation, whereas R-squared is the absolute amount of variation as a proportion of total variation.

Ans2: The Total sum of squares (TSS) measures how much variation there is in the observed data. While the residual sum of squares measures the variation in the error between the observed data and modeled values. Explained sum of squares (ESS) gives an estimate of how well a model explains the observed data for the process. It tells how much of the variation between observed data and predicted data is being explained by the model proposed. Residual sum of squares (RSS) is a statistical technique used to measure the amount of variance in a dataset. RSS measures the level of variance in the error term or residuals, of a regression model. In statistics, the ESS alternatively, known as the model sum of squares or sum of squares due to regression. RSS or sum of squares of errors, is a quantity used in describing how well a model, often a regression model, represents the data being modelled. In particular, the ESS measures how much variation there is in the modelled values and this is compared to the TSS, which measures how much variation there is in the observed data and to the RSS, which measures the

variation in the error between the observed data and modelled value. the equation relating these three metrics with each other is.

TSS = ESS + RSS
Where
TSS = Total sum of squares
ESS = Estimated sum of squares
RSS = Residual sum of squares
The aim of regression analysis is explained the variation of dependent variable "Y".

Ans3: Regularization refers to a technique that are used to calibrate machine learning models in order to minimize the adjusted loss function and prevent overfitting and underfitting. It is used in machine learning models to cope with the problem of overfitting i.e, when the difference between training error and the test error is too high. Using regularization, we can fit our machine learning model appropriately on a given test set and hence reduce the errors in it. It is a technique that penalize the coefficient. In an overfit model, the coefficients are generally inflated. Thus, regularization adds penalties to the parameters and avoids them weigh heavily. The coefficients are added to the cost function of the linear equation. Regularization is important in machine learning because it can help to improve the performance of a learning algorithm. In, particular, it can help to avoid overfitting and therefore improve the generalizability of the model.

Ans4: Gini index also known as Gini impurity, calculates the amount of probability of a specific feature that is classified incorrectly when selected randomly. If all the elements are linked with a single class then it can be called pure.
Gini impurity is a measurement used to build Decision Tree to determine how the features of a dataset should split nodes to form the tree. It is calculated by subtracting the sum of squared probabilities of each class from one. It favors larger partitions and is easy to implement. The lower the impurity measure the better the split the lower impurity measure is better. Gini index varies between 0 and 1, where 0 expresses the purity of classification i.e, all the elements belong to a specified class or only one class exists there, and 1 indicates the random distribution of elements across various classes. The value of 0.5 of the Gini index

shows an equal distribution of elements over some classes. The Gini index is determined by deducting the sum of squared of probabilities of each class.

Ans5: One of the limitations of decision tree is that they are largely unstable compared to other decision predictors. A small change in the data can result in a major change in the structure of the decision tree. 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 events that meet the previous assumptions. Overfitting affects the accuracy of predictions made from the samples which are not part of the training set. One can build a perfect decision tree model on the training data with 100% accuracy, but with significantly low accuracy on test data.

Ans6: Ensemble 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. The ensemble techniques such as Boosting and Bagging. Noise , Variance and Bias are the major source of error. The Ensemble methods in machine learning help minimize theses error causing factors, there by ensuring the accuracy and stability of machine learning algorithms. The primary goal of 'Bagging' ensemble method is to minimize variance errors. Bagging often considers homogenous weak learners, learns them independently from each other in parallel and combines them following some kind of deterministic averaging process. An alternative ensemble technique "Boosting", adjusts an observation's weight based on its last classification. In case observation is incorrectly classified, boosting increases the observation's weight and reduce bias errors and produce super predictive models. Where as Random Forest Models can be thought of as Bagging with a slight tweak when deciding where to split and how to make decisions, Bagged Decision Tree have the full disposal of features to choose from. Random forest models decide where to split based on a random selection of features. Rather than splitting at similar features at each node throughout, random forest models implement a level of differentiation because each tree will split based on different features. This level of differentiation provides a greater ensemble to aggregate over, ergo producing a more accurate predictor.

Ans7:

Bagging	Boosting
The simplest way of combining predictions that belong to the same type. Each model receives equal weight.	A way of combining predictions that belong to the different types. Models are weighted according to their performance
Each model is built independently	New models are influenced by the performance of previously built models.
Bagging tries to solve the over-fitting problem.	Boosting tries to reduce bias.
If the classifier is instable (high variance) then apply bagging	If the classifier is stable and simple (high bias) then apply boosting.
Aim to decrease variance, not Bias.	Aim to decrease bias, not variance.
In Bagging base classifiers are trained parallelly.	In Boosting base classifier are trained sequentially.
For example: the random forest model uses Bagging	For example: the AdaBoost uses Boosting

Ans8: This ensemble method, known as random forest, often outperforms using a single tree. During Bootstrap process, random resamples of variables and records are often taken. Random forest is trained using Bootstrap aggregation, where each new tree is fit from a bootstrap sample of the training observations. The out of the 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 random forest to be fit and validates whilst being trained. The out of bag error is the average error for each predicted outcome calculated using predictions from the trees that do not contain that data point in their respective bootstrap sample. This way, random forest model is constantly being validated while being trained.

Ans9: Cross- validation is a statistical method used to estimate the skill of machine learning models. It is commonly used in applied machine learning to compare and select a model for a given predictive modeling problem because it is

easy to understand, implement and results in skill estimates that generally have a lower bias than other methods. 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 refers to the number of groups that a given data sample is to be split into. As such, procedure is often called as 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 becoming 10-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 10, we can call a 10-fold-cross-validation.

Ans10: In Machine learning, we need to differentiate between parameters and hyperparameters. A learning algorithm learns or estimates model parameters for the given data set, then continues updating these values as it continues to learn. After learning is complete, these parameters become part of the model. Hyperparameters are specific to the algorithms itself, so we can't calculate their values from the data. We use hyperparameters to calculate the model parameter. Different hyperparameter values produce different model parameter values for a given dataset. Hyperparameter tuning consists of finding a set of optimal hyperparameter values for a learning algorithm while applying this optimized algorithms to any data set. That combination of hyperparameter maximizes the model's performance, minimizing a predefined loss function to produce better results with fewer errors. Note that the learning algorithm optimizes the loss based on the input data and tries to find an optimal solution within the given setting. However, hyperparameters describe this setting exactly. Hyperparameter tuning is an essential part of controlling the behavior of a machine learning model. If we don't correctly tune our hyperparameters, our estimated model parameters produce suboptimal results, as they don't minimize the loss function. This means our model makes more errors. In practice, key indicators like the accuracy or the confusion matrix will be worse.

Ans11: Learning rate is one such hyper-parameter that defines the adjustment in the weights of our network with respect to the loss gradient descent. It

determines how fast or slow we will move towards the optimal weights. The Gradient Descent Algorithms estimates the weights of the model in many iterations by minimizing a cost function at every step. 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. If it is too many iterations to converge to the best values. So using a good learning rate is crucial. When the learning rate is to large 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. However, if the learning rate is set too large it can cause undesirable divergent behavior in your loss function.

Ans12: 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. Logistic regression has traditionally been used as a linear classifier. When the classes can be separated in the feature space by linear boundaries may be non-linear. No, we cannot use Logistic Regression for classification of Non-Linear Data as logistic regression only forms linear decision surface. And logistic regression is indeed non -linear in terms of odds and probability, however it is linear in terms of Log Odds

Ans13: Both AdaBoost 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 or weighted addition of the weak learners.

AdaBoost	Gradient Boost
In AdaBoost, shift is done by up-weighting observations that were misclassified before.	In Gradient Boost identities difficult observations by large residuals computed in the previous iterations
AdaBoost is considered as a special case of Gradient boost in terms of loss function, in which exponential losses.	Concepts of gradients are more general in nature.
In AdaBoost, shortcomings are identified by high-weight data points.	In Gradient Boost shortcomings are identified by gradients
Exponential loss of AdaBoost gives more weights for those samples fitted worse.	Gradient Boost further dissect error components to bring in more explanation.

Ans14: In statistics and machine learning, the bias-variance tradeoff is the property of a model that the variance of the parameter estimated across samples can be reduced by increasing the bias in the estimated parameters. This is the conflict in trying to simultaneously minimize these two sources of error that prevent supervised machine learning algorithms from generalizing beyond the training set. If our model is too simple and has very few parameters then it may have high bias and low variance. But if our model has large number of parameters then it's going to have high variance and low bias. When we modify the machine learning algorithm to better fit a given dataset, it will in turn lead to low bias and but will increase the variance. This way, the model will fit with the dataset while increasing the chances of inaccurate predictions.

Ans15: short description:

Linear: The linear model is one of the simplest models in machine learning, but linear models are the building blocks for deep neural networks. There are two main classes in supervised machine learning problems, regression and classification. In reversal, the target value is the actual value.

RBF: Radial Basis Functions (RBF) are real-valued functions that use supervised machine learning to perform as a non-linear classifier. Its value depends on the distance between the input and a certain fixed point.

Polynomial kernels used in SVM: In machine learning polynomial kernel is a kernel function commonly used with Support Vector Machines (SVM) 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.