MACHINE LEARNING

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?

Answer→

R-Squared is a statistical measure in a regression model that determines the proportion of variance in the dependent variable that can be explained by the independent variable where The residual sum of squares (RSS) is a statistical technique used to measure the variance in a data set that is not explained by the regression model. Before assessing a numeric measure of the goodness of fit, like r-squared, you should evaluate the residual plot. The residual plot can expose a biased model for more effectively than the numeric output by displaying problematic patterns in the residuals. If your model is biased, you can not trust the result. If your residual plots look good, go ahead and access your r-squared and other statistics. : Residual Sum of Squares is a better measure of goodness of fit model in regression as it measures the amount of error remaining between the regression function and the data set after the model has been run.

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.

Answer→

The Equation is TSS = ESS + RSS

The TSS tells you how much variation there is in the dependent variable.

The ESS tells you how much of the variation in the dependent variable your model explained.

The RSS tells you how much of the dependent variable's variation your model did not explain.

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

Answer→

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?

Answer→

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 index calculates the amount of probability of a specific feature that is classified incorrectly when selected randomly.

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

Answer→

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. That means decision-trees prone to overfitting when :

1. Overfitting Due to Presence of Noise.
2. Overfitting Due to Lack of Representative Instances.
3. Overfitting and the Multiple Comparison Procedure

6. What is an ensemble technique in machine learning?

Answer→

Ensemble methods is a machine learning technique that combines several base models in order to produce one optimal predictive model. Basically in Ensemble technique multiple weak learners are ensemble into one unit make them boosted and amplify their learning and compute the results. Ensemble technique aims 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.

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

Answer→

Bagging is a method of merging the same type of predictions. Boosting is a method of merging different types of predictions.

Bagging decreases variance, not bias, and solves over-fitting issues in a model. Boosting decreases bias, not variance.

In Bagging, each model receives an equal weight. In Boosting, models are weighed based on their performance.

Models are built independently in Bagging. New models are affected by a previously built model’s performance in Boosting.

In Bagging, training data subsets are drawn randomly with a replacement for the training dataset. In Boosting, every new subset comprises the elements that were misclassified by previous models.

Bagging is usually applied where the classifier is unstable and has a high variance. Boosting is usually applied where the classifier is stable and simple and has high bias.

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

Answer→

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.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 Random Forest Classifier to be fit and validated whilst being trained.

9. What is K-fold cross-validation?

Answer→

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 becoming 10-fold cross-validation

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

Answer→

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.

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

Answer→

For the gradient descent algorithm to reach the local minimum we must set the learning rate to an appropriate value, which is neither too low nor too high . 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. This is important because if the steps it takes are too big, it may not reach the local minimum because it bounces back and forth between the convex function of gradient descent.

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

Answer→

No because Logistic regression is considered a generalized linear model because the outcome always depends on the sum of the inputs and parameters. Or in other words, the output cannot depend on the product (or quotient, etc.) of its parameters.

13. Differentiate between Adaboost and Gradient Boosting.

Answer→

Adaboost is more about ‘voting weights’ and Gradient boosting is more about ‘adding gradient optimization’. Adaboost increases the accuracy by giving more weightage to the target which is misclassified by the model. At each iteration, Adaptive boosting algorithm changes the sample distribution by modifying the weights attached to each of the instances. It increases the weights of the wrongly predicted instances and decreases the ones of the correctly predicted instances. Gradient boosting calculates the gradient (derivative) of the Loss Function with respect to the prediction (instead of the features). Gradient boosting increases the accuracy by minimizing the Loss Function (error which is difference of actual and predicted value) and having this loss as target for the next iteration. Gradient boosting algorithm builds first weak learner and calculates the Loss Function. It then builds a second learner to predict the loss after the first step. The step continues for third learner and then for fourth learner and so on until a certain threshold is reached.

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

Answer→

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. 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 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.

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

Answer→

. Linear kernel: A linear kernel is used as normal dot product of any two given observation. The product between two vectors is the sum of the multiplication of each pair of input values. It is mostly used when there are a Large number of Features in a particular Data Set. Training a SVM with a Linear Kernel is faster than with any other Kernel.

RBF(Radial Basis Function): In machine learning, the radial basis function kernel, or RBF kernel, is a popular kernel function used in various kernelized learning algorithms’ can map an input space in infinite dimensional space. In particular, it is commonly used in support vector machine classification

A polynomial kernel is a more generalized form of the linear kernel. The polynomial kernel can distinguish curved or nonlinear input space. 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.