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?

R-squared is better measure of goodness of fit model in regression due to below reasons.

R-squared and Residual Sum of Squares (RSS) are both measures of the goodness of fit of a regression model, but they serve slightly different purposes.

R-squared is a statistical measure that represents the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a regression model. In other words, R-squared measures the extent to which changes in the dependent variable can be predicted by changes in the independent variable(s). Higher R-squared values indicate a better fit of the regression model to the data. Therefore, R-squared is often used to compare different models and select the best one.

On the other hand, Residual Sum of Squares (RSS) measures the difference between the observed values of the dependent variable and the predicted values by the model. It represents the sum of the squared differences between the actual and predicted values of the dependent variable. The goal is to minimize the residual sum of squares to obtain a better model fit.

In terms of determining the goodness of fit of a model, R-squared is generally considered a better measure than RSS. This is because R-squared provides an overall measure of the proportion of variance in the dependent variable that is explained by the model, whereas RSS only measures the magnitude of the residuals. Additionally, R-squared is a standardized measure and ranges from 0 to 1, making it easy to compare the fit of different models. In contrast, the magnitude of the RSS value depends on the scale of the dependent variable and can't be easily compared across models.

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.

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

Regularization 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. Coming to linear models like logistic regression, the model might perform very well on your training data and it is trying to predict each data point with so much precision.

4. What is Gini–impurity index?

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. More precisely, the 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?

Yes unregularized decision-trees prone to overfittings, so regularization is required to avoid overfittings.

Regularization in machine learning is the process of regularizing the parameters that constrain, regularizes, or shrinks the coefficient estimates towards zero. In other words, this technique discourages learning a more complex or flexible model, avoiding the risk of Overfitting.

6. What is an ensemble technique in machine learning?

Ensemble methods is a machine learning technique that combines several base models in order to produce one optimal predictive model.

The goal of any machine learning problem is to find a single model that will best predict our wanted outcome. Rather than making one model and hoping this model is the best/most accurate predictor we can make, ensemble methods take a myriad of models into account, and average those models to produce one final model. It is important to note that Decision Trees are not the only form of ensemble methods, just the most popular and relevant in Data Science today.

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

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 than a single one.

- The bagging technique combines multiple models trained on different subsets of data, whereas boosting trains models sequentially, focusing on the error made by the previous model.
- Bagging is best for high variance and low bias models while boosting is effective when the model must be adaptive to errors, suitable for bias and variance errors.
- Generally, boosting techniques are not prone to overfitting. Still, it can be if the number of models or iterations is high, whereas the Bagging technique is less prone to overfitting.
- Bagging improves accuracy by reducing variance, whereas boosting achieves accuracy by reducing bias and variance.
- Boosting is suitable for bias and variance, while bagging is suitable for high-variance and low-bias models.

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

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

9. What is K-fold cross-validation?

The data sample is split into ‘k’ number of smaller samples, hence the name: K-fold Cross Validation. You may also hear terms like four fold cross validation, or ten fold cross validation, which essentially means that the sample data is being split into four or ten smaller samples respectively.

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

Hyperparameters directly control model structure, function, and performance. Hyperparameter tuning allows data scientists to tweak model performance for optimal results. This process is an essential part of machine learning, and choosing appropriate hyperparameter values is crucial for success.

We can choose from three hyperparameter tuning methods — grid search, random search, and Bayesian optimization. If evaluating our model with training data will be quick, we can choose the grid search method. Otherwise, we should select random search or Bayesian optimization to save time and computing resources.

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

It determines how quickly or slowly our model learns, and it plays an important role in controlling both convergence and divergence of the algorithm. When the learning rate is too large, gradient descent can suffer from divergence.

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

The logistic function used to transform a linear combination of the input into a non-linear output, so we can't use logistic regression for classification of non-linear data.

13. Differentiate between Adaboost and Gradient Boosting.

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.

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

Bias-Variance Tradeoff is crucial in machine learning because it directly impacts a model's predictive performance. A model with high bias will consistently produce predictions that are far from the actual values, while a model with high variance will produce widely varying predictions for different training datasets.

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

1. **Linear Kernel**
   - The linear kernel is the simplest and is used when the data is linearly separable.
   - It calculates the dot product between the feature vectors.

2. **Polynomial Kernel**
   - The polynomial kernel is effective for non-linear data.
   - It computes the similarity between two vectors in terms of the polynomial of the original variables.

3. **Radial Basis Function (RBF) Kernel**
   - The RBF kernel is a common type of Kernel in SVM for handling non-linear decision boundaries.
   - It maps the data into an infinite-dimensional space.