Machine Learning-2

Q1.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?

RSS is the sum of the squares of the errors made by the model on each data point. So, RSS depend upon the number of data-points in the data. If dataset is large, then naturally it will have large RSS because RSS is the sum of squares of the errors made by all the data points. While on the other hand R-squared is given as follows:

$$RSS = \sum^{n}_{i=1} (y^{i} - f(x_{i}))^{2}$$

Where:

y_i = the ith value of the variable to be predicted

$f(x_i)$ = predicted value of y_i

n = upper limit of summation

So, R-squared does not depend upon the number of data-points in the data, rather it depends only on the quality of the fit of the curve on the data, while RSS depends on both the quality of fit and the number of data points in the data.

Q2. What is the need of regularization in machine learning?

Regularization is one of the most important concepts of machine learning. It is a technique to prevent the model from overfitting by adding extra information to it.

Sometimes the machine learning model performs well with the training data but does not perform well with the test data. It means the model is not able to predict the output when deals with unseen data by introducing noise in the output, and hence the model is called overfitted. This problem can be deal with the help of a regularization technique.

This technique can be used in such a way that it will allow to maintain all variables or features in the model by reducing the magnitude of the variables. Hence, it maintains accuracy as well as a generalization of the model.

It mainly regularizes or reduces the coefficient of features toward zero. In simple words, "In regularization technique, we reduce the magnitude of the features by keeping the same number of features."

Q3. 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.

Q4. What is an ensemble technique in machine learning?

Ensemble methods are 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. Voting and averaging are two of the easiest ensemble methods. They are both easy to understand and implement. Voting is used for classification and averaging is used for regression.

Q5. What is the difference between Bagging and Boosting technique?

Bagging:
Bagging is an acronym for 'Bootstrap Aggregation' and is used to decrease the variance in the prediction model. Bagging is a parallel method that fits different, considered learners independently from each other, making it possible to train them simultaneously.
Bagging generates additional data for training from the dataset. This is achieved by random sampling with replacement from the original dataset. Sampling with replacement may repeat some observations in each new training data set. Every element in Bagging is equally probable for appearing in a new dataset.
These multi datasets are used to train multiple models in parallel. The average of all the predictions from different ensemble models is calculated. The majority vote gained from the voting mechanism is considered when classification is made. Bagging decreases the variance and tunes the prediction to an expected outcome.

Boosting:
Boosting is a sequential ensemble method that iteratively adjusts the weight of observation as per the last classification. If an observation is incorrectly classified, it increases the weight of that observation. The term 'Boosting' in a layman language, refers to algorithms that convert a weak learner to a stronger one. It decreases the bias error and builds strong predictive models.
Data points mis predicted in each iteration are spotted, and their weights are increased. The Boosting algorithm allocates weights to each resulting model during training. A learner with good training data prediction results will be assigned a higher weight. When evaluating a new learner, Boosting keeps track of learner's errors.
If a provided input is inappropriate, its weight is increased. The purpose behind this is that the forthcoming hypothesis is more likely to properly categorize it by combining the entire set, at last, to transform weak learners into superior performing models.

Q6. What is out-of-bag error in random forests?

The out-of-bag error is **an error estimation technique** often used to evaluate the accuracy of a random forest and to select appropriate values for tuning parameters, such as the number of candidate predictors that are randomly drawn for a split, referred to as try. However, for binary classification problems with metric predictors it has been shown that the out-of-bag error can overestimate the true prediction error depending on the choices of random forests parameters.

Q7. What is K-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 5, we can call this a 5-fold cross-validation. Each fold is used as a testing set at one point in the process.

Q8. What is hyper parameter tuning in machine learning and why it is done?

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.

Q9. What issues can occur if we have a large learning rate in Gradient Descent?

Learning Rate is the hyperparameter that determines the steps the gradient descent algorithm takes. Gradient Descent is too sensitive to the learning rate. If it is too big, the algorithm may bypass the local minimum and overshoot.

Q10. Can we use Logistic Regression for classification of Non-Linear Data? If not, why?

Non-linear problems cannot be solved with logistic regression because it has a linear decision surface. The decision boundary is a line or a plane that separates the target variables into different classes that can be either linear or nonlinear. In the case of a Logistic Regression model, the decision boundary is a straight line.

Q11. Differentiate between Adaboost and Gradient Boosting.

AdaBoost or Adaptive Boosting is the first Boosting ensemble model. The method automatically adjusts its parameters to the data based on the actual performance in the current iteration. Meaning, both the weights for re-weighting the data and the weights for the final aggregation are re-computed iteratively.

In practice, this boosting technique is used with simple classification trees or stumps as base-learners, which resulted in improved performance compared to the classification by one tree or other single base-learner.

Gradient Boosting

Gradient Boost is a robust machine learning algorithm made up of Gradient descent and Boosting. The word ‘gradient’ implies that you can have two or more derivatives of the same function. Gradient Boosting has three main components: additive model, loss function and a weak learner.

The technique yields a direct interpretation of boosting methods from the perspective of numerical optimisation in a function space and generalises them by allowing optimisation of an arbitrary loss function.

Q12. What does bias variance trade off in machine learning?

While building the machine learning model, it is important to take care of bias and variance in order to avoid overfitting and underfitting in the model. If the model is very simple with fewer parameters, it may have low variance and high bias. Whereas, if the model has many parameters, it will have high variance and low bias. So, it is required to make a balance between bias and variance errors, and this balance between the bias error and variance error is known as the Bias-Variance trade-off.

Q13. Give short description each of Linear, RBF, Polynomial kernels used in SVM.

Linear kernels - Linear Kernel is used when the data is Linearly separable, that is, it can be separated using a single Line. It is one of the most common kernels to be used. It is mostly used when there are many Features in a particular Data Set. Polynomial kernels- It represents the similarity of vectors in the training set of data in a feature space over polynomials of the original variables used in the kernel. RBF kernel - When the data set is linearly inseparable or in other words, the data set is non-linear, it is recommended to use kernel functions such as RBF.

Q14. 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.

TSS is total sum of squares. It is equal to the variance of the data.

• ESS is called the explained sum of squares. It is the variance of the data which has been explained by the model. The model can explain this much variance of the data. To refer the formula please look above in the image.

- RSS is called the residual sum of squares. It is equal to the sum of squares of all the errors or residuals made by the model on the data. The relationship between tss, rss and ess is given as: TSS = ESS + RSS

Q15. Are unregularized decision-trees prone to overfitting? If yes, why?

Unregularized decision-trees are highly prone to overfitting. If we do not restrict the depth up to which a tree can be grown or control it in any other way, the decision tree will most likely learn each data point in the training dataset. So, it will learn the training data patterns too closely and when it will be tested on unseen data, it will most likely perform poorly. So, to solve this problem, we regularize decision trees by several ways, either by controlling the depth of the tree, or controlling the maximum number of leaves tree can have etc.