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: - I think R-squared is better measure of goodness of fit in a regression than Residual sum of squares.

Reason: - R-squared is easy to interpret the ratio of the variance in the dependent variable which is explained by independent variables in the model. If range is 0 to 1, in very rare case it can slightly negative. If R- squared of 0.75 it means that 75% of the variance in the target variable is explained by the model.

R-squared is Good to understand how better model is performing. It is easy to predict the mean of the data and easy to understand the model performance.

RSS is depending on the scale and units of the data, it is difficult to compare models with different target variables or scales. But RSS is useful in Calculating absolute error.

Absolute error = |Actual value – predicted value.

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: - These metrics together help to evaluate the goodness of fit of a regression model.
TSS: Total Variability in the data.
ESS: Variability explained by the model.
RSS: Variability not explained by the model.

TSS. Represent the total variability in the observed data. It measures how much the actual data points deviate from their mean, reflecting the overall spread or variation in the data.

These three metrics are related by the following equation

TSS = ESS + RSS

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

Answer: - Regularization is a process of introducing additional information to prevent overfitting. Regularization helps the model to generalize better to new data set by simplifying the learned pattern.

In Regularization there are 2 types

1) L1 Regularization Lasso regression
2) L2 Regularization Ridge regression

Ridge regression is more important to handle overfitting situations.

In ridge regression have to use scikit learn library to implement these regularization techniques.

4. What is Gini–impurity index?

Answer: - Gini-impurity is a metric used to measure the degree or probability of particular variable wrongly classified when selected randomly. If the elements are randomly distributed across different classes, The Gini impurity is high up to a maximum of 0.5 for a binary classification problem 'perfectly impure'

we have a dataset with two classes, A and B. If 80% of the data points belong to class A and 20% belong to class B, the Gini impurity is calculated as

Gini=1−(0.82+0.22) =1−(0.64+0.04) =1−0.68=0.32

Entropy and Gini-impurity both doing the same task, that is to calculate the purity of split in a decision tree. Most of the time Gini-impurity is better as compared to Entropy.

(Personal comment mentioned below)

(I was not clear of this index however I learned from you tub and understand the concept also tried on Jupiter notebook to understand on this more)

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

Answer: - Yes unregularized decision trees are prone to overfitting.

Unregularized tree, the algorithm will continue to split the data until all leaf nodes are pure. In doing so, the tree may capture noise and anomalies in the dataset, which does not contribute to the true underlying patterns. This causes poor performance on test data because the model is too tailored to the training set.

By applying regularization the tree avoids becoming too complex and can generalize better to unseen data, reducing overfitting. Regularization helps to control the complexity of the tree, making it better at generalizing to unseen data reducing overfitting.

Unregularized decision trees tend to overfit because they grow too complex and capture noise in the data.

6. What is an ensemble technique in machine learning?

Answer: - Ensemble techniques in machine learning are powerful tools for improving model accuracy, reducing overfitting, and enhancing robustness. By combining predictions from multiple models, ensemble methods like **bagging**, **boosting**, and **stacking** can outperform individual models and offer more reliable predictions in many machine learning tasks.

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

Answer: - Boosting is more powerful to improve the accuracy of data. But there is high risk of overfitting. Boosting: Models are weighted based on their performance. Models that perform better have higher weights, while models that perform worse are given less importance in the final prediction.

Bagging is useful to reduce the overfitting. Generally, reduces overfitting by averaging predictions across multiple models, making it more robust to noisy data.

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

Answer: - Out-of-Bag error is a built in cross-validation method for Random Forests that provides an unbiased estimate of the model's performance using only the training data. It's efficient, avoids the need for a separate validation set, and is a useful tool for detecting overfitting.

9. What is K-fold cross-validation?

Answer: - K-fold cross-validation is used to assess the generalization performance in ML model. Also, it also ensures model is not overfitting on unseen data. K-fold cross-validation is powerful technique.

Reduced overfitting, more reliable performance Estimates.

There are few advantages of K-fold cross-validation, better generalization. Reduced Bias and More data efficient.

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

Answer: - Hyperparameter tuning in machine learning refers to the process of finding the optimal set of hyperparameters for a machine learning algorithm to improve its performance on a specific task.

Hyperparameters are parameters that are not learned by the model itself but are set before the learning process begins. These parameters can significantly impact the model's accuracy, performance, and generalization ability.

Properly tuned hyperparameters lead to better model performance in terms of accuracy.

A well-tuned model is more likely to generalize to unseen data, improving its predictive power.

Hyperparameters control how a model learns from data.

Tuning these parameters is critical to getting the best performance out of the model.

Techniques like grid search, random search, and Bayesian optimization help in finding the optimal hyperparameters.

Properly tuned hyperparameters can significantly improve a model's performance and its ability to generalize to new data.

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

Answer: - If we have a large learning rate, Low learning rate 0.1. Moderate learning rate 0.5. High learning rate 1.2.

A large learning rate in gradient descent can cause.

Overshooting the minimum, missing the optimal point.

Divergence where the cost function increases rather than decreases.

Failure to converge because of erratic updates.

High variance in the learning process, leading to unstable training.

Settling for a suboptimal solution, as the model cannot fine-tune toward the true minimum.

Finding the right balance in the learning rate is essential to ensure stable and efficient convergence. This is why hyperparameter tuning is important in machine learning.

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

Answer: - No, logistic regression is not suitable for the classification of non-linear data without modifications. This is because logistic regression is a linear classifier, meaning that it tries to separate classes using a linear decision boundary.

Logistic regression is not well-suited for non-linear data unless modifications are made, such as feature transformation (e.g., polynomial features) or using kernel methods.

For non-linear data, more sophisticated techniques like decision trees, SVMs with non-linear kernels, or neural networks are typically better choices.

13. Differentiate between Adaboost and Gradient Boosting.

Answer: - **AdaBoost** and **Gradient Boosting** are both popular ensemble techniques in machine learning that aim to combine weak learners (typically decision trees) to create a strong predictive model.

AdaBoost (Adaptive Boosting)

Focuses on reweighting the training data points based on the errors of the previous weak learners.

Emphasizes misclassified data points by assigning them higher weights so that the next learner can focus on getting these points right.

It combines the weak learners in a sequential manner by adjusting their weights based on performance.

Gradient Boosting

Focuses on reducing the residual errors made by previous models.

Instead of reweighting the data points, Gradient Boosting builds the next model to minimize the **loss function** (e.g., mean squared error) of the residuals from the previous model.

Models are combined by learning the gradient of the loss function, adjusting iteratively.

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

Answer: - The **bias-variance tradeoff** is a fundamental concept in machine learning that describes the relationship between the error introduced by the bias of a model and the variance of the model. It helps us understand how different model complexities affect the model's performance and its ability to generalize to new, unseen data.

Bias is reduced by increasing model complexity, but it comes with the risk of increasing **variance**.

Variance is reduced by simplifying the model, but it comes with the risk of increasing **bias**.

The **bias-variance tradeoff** is about finding the right balance to minimize the total error and improve the model's generalization to new data.

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

Answer: - This is the simplest kernel used when the data is linearly separable. It maps the input features directly without any transformation.

Best suited for linearly separable datasets where a straight line or hyperplane can distinguish between the classes.