Q1. Ans:- R-squared is a better way measure how well a model fits the data because it shows what percentage of the data the model explains.

Residual Sum of Squares shows how far off the model's predictions are from the actual data, but it's harder to understand and depends on the size of the data.

We can think of R-square like a report card grade for our model while RSS is like a list of mistakes the model made. R-square gives us a clearer picture of how well the model is doing.

Q2. Ans:- Total sum of squares tells us how much the values in our data differ from the average value. It's like measuring how spread out our data is.

Explained sum of squares shows how much of that spread (TSS) is captured by our model. It's how well our model explains the differences in the data.

Residual sum of squares measures what's left after our model does its job. It's the part of the spread that our model couldn't explain.

The equation relating these three metrics with each other-

TSS=ESS+RSS\text{TSS} = \text{ESS} + \text{RSS}TSS=ESS+RSS

Q3. Ans:- Regularization in machine learning is like adding a bit of control to keep our model from getting too complicated or too "fancy." Here's why it's needed:

1. Prevent Overfitting happens when our model is too closely tied to the training data. It learns the details and noise in the training data too well, which means it might not perform well on new, unseen data.
2. Simplifying the model- A model with too many features or too much flexibility can become overly complex. This makes it harder to understand and can lead to errors.
3. Improve Generalization- Generalization is the model's ability to perform well on new, unseen data.

In summary, regularization is like putting limits on our model to keep it from becoming too complex and overfitting the training data. This helps make sure it performs better when faced with new data.

Q4. Ans:- The Gini impurity index is a way to measure how mixed up the different classes are in a set of data. It helps us decide how good a split in a decision tree is. It's purpose is to tells us how "impure" or mixed the classes are in a data group. The lower the Gini impurity, the better the split, because it means the group is more homogeneous.

And when building a decision tree, we use the Gini impurity to choose the best split. We pick the split that makes the resulting groups as pure as possible, meaning each group has items mostly from one class.

Q5. Ans:- Yes, unregularized decision trees are prone to overfitting.
Overfitting happens when a model learns the details and noise of the training data too well. As a result, it performs great on the training data but poorly on new, unseen data. Unregularized decision trees tend to overfit because they can become overly complex and tailored to the training data. They might capture too many details and noise, which makes them less effective at handling new, unseen data. Regularization techniques, like pruning, help simplify the tree and reduce overfitting.

Q6. Ans:- An ensemble technique involves using a group of models together to improve prediction accuracy and robustness. Instead of relying on one model, we use several models and combine their predictions.
An ensemble technique in machine learning combines multiple models to make better predictions than any single model on its own. Think of it like getting a second opinion from multiple experts instead of just one.

Q7. Ans:- Bagging and boosting are techniques used to make machine learning models better, but they do it in different ways. **Bagging** involves training several models separately on different random parts of the data and then combining their results, which helps make the predictions more stable and less sensitive to the specific data used. **Boosting**, on the other hand, trains models one after another, with each new model trying to fix the mistakes of the previous ones, which helps improve overall accuracy by focusing on errors. So, bagging reduces variation by averaging out model results, while boosting reduces errors by building on mistakes.

Q9. Ans:- K-fold cross-validation is a technique used to assess how well a machine learning model will perform on unseen data.

Q10. Ans:- Hyperparameter tuning in machine learning is the process of finding the best settings for a model to improve its performance. Unlike regular model parameters that the model learns during training, hyperparameters are set before training begins (like the number of trees in a random forest or the learning rate in gradient boosting). Tuning involves trying different values for these settings and evaluating how well the model performs with each set of values. This is done to ensure the model performs as accurately as possible, as the right combination of hyperparameters can significantly boost the model’s ability to make correct predictions.

Q12. Ans:- Logistic Regression is typically not the best choice for classifying non-linear data because it assumes a linear relationship between the input features and the target variable. This means it works well when the data can be separated by a straight line. For non-linear data, where the classes are not linearly separable, Logistic Regression may struggle to find the correct boundaries and make accurate predictions. However, we can improve its performance on non-linear data by transforming the features or using techniques like polynomial features or kernel methods to help capture non-linear patterns.

Q13. Ans:- AdaBoost and Gradient Boosting are both boosting techniques used to improve the performance of machine learning models, but they differ in how they build and combine their models AdaBoost adjusts weights on data points to correct errors from previous models, while Gradient Boosting builds models to directly address the residual errors, making it more flexible and powerful for complex data.

Q14. Ans:- The bias-variance trade-off in machine learning is about finding the right balance between two types of errors that affect model performance. Bias refers to errors from overly simplistic models that can't capture the complexity of the data, leading to underfitting. Variance refers to errors from overly complex models that fit the training data too closely and don't generalize well to new data, leading to overfitting. The trade-off is finding a model that is complex enough to capture the data patterns (low bias) but not so complex that it gets confused by noise in the training data (low variance). The goal is to minimize both bias and variance to achieve the best performance on unseen data.