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:

- R-squared quantifies the proportion of the total variation in the dependent variable (response) that is explained by the independent variables (predictors) in the model.

Residual sum of errors:

- The RSS quantifies the difference between your observed data and the predictions made by your regression model.
- Use RSS to evaluate how well your model fits the data.
- In least squares regression, minimizing the RSS results in the best possible fit for a given model.

Residual sum of squares is useful when we want to evaluate the goodness of fit focusing on the magnitude of the prediction errors is a primary concern. RSS is better measure.

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.

Total Sum of Squares (TSS):

- TSS is also known as the sum of squares total, quantifies the overall variability in the dependent variable (response variable). It represents the sum of squared differences between the observed dependent variable values and the overall mean.

Explained Sum of Squares (ESS):

- ESS is also called the sum of squares due to regression, measures the variation explained by the regression model. It quantifies how well the regression line fits the data.

- ESS is computed as the sum of squared differences between the predicted values and the mean of the dependent variable
- ESS = Σ(Y-Hat – mean of Y)^2

Residual sum of Squares(RSS):

- RSS, also known as the sum of squares error, captures the unexplained variation in the data.
- It represents the sum of squared differences between the observed values and the predicted values (residuals).
- Equation : TSS(Total Sum of Squares) = ESS(Explained Sum of Squares) + (RSS)Residual Sum of Squares

$$TSS = \Sigma(Yi - \text{mean of } Y)^2.$$

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

- Regularization is a technique prevent the model from overfitting. It discourages the model from fitting the training data too closely and promotes simpler models that generalize better to unseen data.

Overfitting and Underfitting:

- Overfitting: Occurs when a model performs well on the training data but poorly on unseen data. It memorizes noise in the training set instead of learning meaningful patterns.

Underfitting: Happens when a model is too simplistic and fails to learn even basic patterns in the data.

- Bias-Variance Tradeoff:
- Striking a balance between bias and variance is crucial. Regularization helps achieve this balance.

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.

4]What is Gini–impurity index?

- The Gini Impurity, also known as the Gini Index, is a measurement used in building Decision Trees. It helps determine how the features of a dataset should split nodes to form the tree.
- It is the probability of misclassifying a randomly chosen element in a set.
- The range of the Gini index is [0, 1], where 0 indicates perfect purity and 1 indicates maximum impurity.
- Gini index is a linear measure. It can be interpreted as the expected error rate in a classifier.
- It has a bias toward selecting splits that result in a more balanced distribution of classes
- Gini index is typically used in CART (Classification and Regression Trees) algorithms

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

- Yes, Unregularized decision tress prone to overfitting. Decision trees are powerful models that can learn intricate patterns from training data. However, without proper constraints, they tend to overfit
- When a decision tree is unregularized, it can grow without restrictions. It may:
- Memorize noise: By adding nodes for features that don't genuinely contribute to the model's generalization ability.
- Overfit to training data: The tree might learn specific rules for individual training examples rather than capturing broader patterns.

Essentially, an unpruned decision tree can become too deep and complex, leading to overfitting.

6. What is an ensemble technique in machine learning?

Ensemble Technique:

- Ensemble methods is a machine learning technique that combines several base models in order to produce one optimal predictive model.
- Instead of relying solely on a single model, ensemble techniques merge predictions from diverse models to create a more precise prediction.

- By considering multiple perspectives and utilizing the strengths of different models, ensemble learning improves the overall performance of the learning system. [1]
- Increased Accuracy: By combining multiple models, ensemble methods often yield more accurate predictions than individual models.
- Robustness: They reduce the risk of overfitting by averaging out biases and errors from different models.
- Versatility: Ensemble methods can be applied to various base learners beyond just decision trees.

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

Bagging Technique:

- The simplest way of combining predictions that belong to the same type.
- Aim to decrease variance, not bias.
- Different training data subsets are selected using row sampling with replacement and random sampling methods from the entire training dataset.
- Each model receives equal weight.
- Bagging tries to solve the over-fitting problem.

Boosting Technique:

- A way of combining predictions that belong to the different types.
- Aim to decrease bias, not variance.
- Every new subset contains the elements that were misclassified by previous models.
- Models are weighted according to their performance.
- Boosting tries to reduce bias.

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

- The out-of-bag (OOB) error is a performance metric specifically used for ensemble models like random forests.

- It is Calculated using samples not used during training (out-of-bag samples).Provides an unbiased estimate of the model's performance.
- These errors estimate how well a random forest classifier or regressor performs on unseen data.
- Computed using samples not included in training individual trees.
- Different from errors computed using usual training and validation sets.

9. What is K-fold cross-validation?

- K-fold cross-validation is a statistical technique used to estimate the performance of machine learning models.
- The dataset is divided into k subsets (or folds).
- The model is trained and evaluated k times, each time using a different fold as the validation set.
- The remaining k-1 folds are used for training.
- This process helps you gain more confidence in your model's performance by assessing it across different subsets of data.

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

- Hyperparameter tuning is the process of selecting the optimal values for model's hyperparameters. Hyperparameters are settings that control the learning process of the model, such as the learning rate, the number of neurons in a neural network, or the kernel size in a support vector machine.
- The performance of a machine learning model depends on finding the right hyperparameters.
- Optimal hyperparameters lead to better model performance, improved accuracy, and generalization to unseen data.
- Poorly chosen hyperparameters can result in overfitting, slow convergence, or instability.

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

Divergence:

- A very high learning rate can lead to divergence, where the optimization process fails to converge to any solution.

Trajectory Behavior:

- Large learning rates can significantly alter the trajectory of optimization.

Overshooting the Minimum:

- If the learning rate (step size) is too large, the algorithm can overshoot the minimum point in the loss function.

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

- Logistic regression is not inherently designed to handle non-linear relationships between features and the target variable.
- When the decision boundary is curved or complex, logistic regression may struggle to capture these patterns effectively.Logistic regression is not inherently designed to handle non-linear relationships between features and the target variable.
- Logistic regression assumes a linear decision boundary in the feature space. This means that it tries to find a hyperplane that separates observations belonging to a class from those that do not. If the true decision boundary is non-linear, logistic regression may not perform well.

13. Differentiate between Adaboost and Gradient Boosting.

Adaboost :

- AdaBoost aims to minimize the exponential loss function.
- It assigns weights to instances based on their performance in previous iterations. Misclassified instances receive higher weights.
- AdaBoost trains weak learners sequentially, adjusting weights at each step.
- AdaBoost is less flexible than Gradient Boosting.

Gradient Boosting:

- Gradient Boosting minimizes the gradient of the loss function.
- It focuses on residuals (differences between actual and predicted values) from the previous model.
- Weak learners are added stagewise, with each one correcting the errors of the previous model.
- Gradient Boosting includes regularization (e.g., shrinkage) to prevent overfitting.
- It's more flexible and can handle various loss functions.

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

Bias-variance trade off:

- Bias measures the average difference between predicted values and true values. As bias increases, a model predicts less accurately on a training dataset. High bias refers to high error in training.
- Variance measures the difference between predictions across various realizations of a given model. As variance increases, a model predicts less accurately on unseen data. High variance refers to high error during testing and validation.
- Simultaneous reduction in both is not always possible, resulting in the need for regularization. Regularization decreases model variance at the cost of increased bias.

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

Linear kernel:

- The linear kernel is the simplest and most straightforward kernel. It works well when the data is linearly separable.
- It computes the dot product between the input features and is suitable for problems where the decision boundary is a straight line or hyperplane.
- The linear kernel is efficient and easy to interpret.However, it may not perform well when dealing with complex, nonlinear data.

Polynomial Kernel:

- The polynomial kernel introduces nonlinearity by mapping the data into a higher-dimensional space using polynomial functions.
- It allows SVM to capture more complex decision boundaries. The degree of the polynomial determines the complexity of the mapping.
- Polynomial kernels are useful when the data is not linearly separable but exhibits some degree of curvature.

RBF or Radius Basis Function Kernel:

- The RBF kernel, also known as the Gaussian kernel, is one of the most powerful and widely used kernels. It maps the data into an infinite-dimensional space using a radial basis function.
- The RBF kernel is effective for capturing complex, nonlinear relationships. It works well even when the decision boundary is irregular or has intricate shapes.
- The RBF kernel has a free parameter, σ (sigma), which controls the width of the Gaussian function. Tuning σ affects the smoothness of the decision boundary.