MACHINE LEARNING ASSIGNMENT 5

NAME -SUBHAM

BATCH – DS2401

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?**

   Ans) R-squared (coefficient of determination) and Residual Sum of Squares (RSS) are both measures used to assess the goodness of fit in regression models, but they capture different aspects of model performance.

   R Squared(R^2):

   R-squared represents the proportion of the variance in the dependent variable explained by the independent variables in the model. It ranges from 0 to 1, where 1 indicates a perfect fit. It provides an easily interpretable measure of the overall goodness of fit, with higher values indicating a better-fitting model.

   However, R-squared can be sensitive to the number of predictors and may not reveal problems with individual predictors.

   Residual sum of squares(RSS):

   RSS measures the total squared difference between the predicted and observed values of the dependent variable.

   It directly quantifies the errors of the model, providing insight into how well the model predicts individual data points.

   While RSS is sensitive to model fit, it doesn't give a clear indication of the proportion of variance explained.

   Which is better:

   R squared: It is often preferred when you want a single, comprehensive measure of overall model fit. However, high R-squared doesn't guarantee that the model's predictions are accurate for every data point.

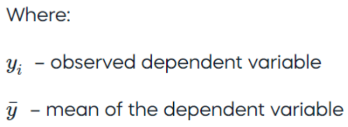
   RSS: It is valuable for understanding the magnitude of prediction errors. It helps identify if the model is systematically underestimating or overestimating the target variable.

   R-squared is often considered to be a better measure of goodness of fit than RSS because it provides a single number that summarizes the proportion of variance in the dependent variable that is explained by the model, which is more interpretable and easier to compare 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.**

   Ans) TSS (Total sum of squares) or SST(Sum of squares total) is the sum of squared differences between the observed *dependent variables* and the overall **mean**. It tells you how much variation there is in the independent variable

$$SST = \sum_{i=1}^{n} (y_i - \bar{y})^2$$



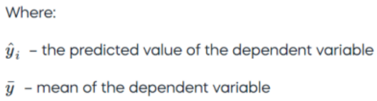
Where:

y_i – observed dependent variable

$\bar{y}$ – mean of the dependent variable

explained sum of squares (ESS) or sum of squares due to regression (SSR) is the sum of the differences between the *predicted value* and the **mean** of the *dependent variable*. In other words, it describes how well our line fits the data. It tells you how much the variation in the dependent variable your model explained

$$SSR = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2$$



Where:

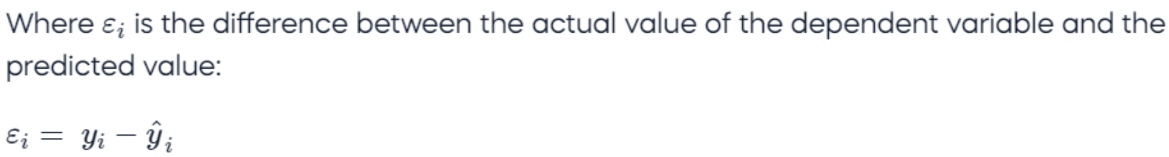
$\hat{y}_i$ – the predicted value of the dependent variable

$\bar{y}$ – mean of the dependent variable

If **SSR** equals **SST**, our **regression model** perfectly captures all the observed variability, but that's rarely the case**.**

The **sum of squares error (SSE) or residual sum of squares (RSS, where** residual means remaining or unexplained**)** is the difference between the *observed* and *predicted* values. It tells you how much of the dependent variable's variation your model does not explain.

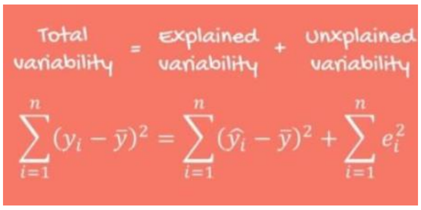
$$SSE = \sum_{i=1}^{n} \varepsilon_i^2$$



Where ε_i is the difference between the actual value of the dependent variable and the predicted value:

$$\varepsilon_i = y_i - \hat{y}_i$$

Relation between them mathematically it is, TSS = ESS + RSS



Total variability = Explained variability + Unexplained variability

$$\sum_{i=1}^{n}(y_i - \bar{y})^2 = \sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2 + \sum_{i=1}^{n} e_i^2$$

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

Ans) regularization is used in machine learning to reduce errors by fitting the function appropriately on the given training set and avoiding overfitting . while training a machine learning model , the model can be overfitted or underfitted. To avoid this , we use regularization to properly fit a model onto our test set.

4. What is Gini–impurity index?

Ans) The Gini impurity index is a measure used in decision tree algorithms, such as CART (Classification and Regression Trees), to evaluate the impurity or disorder of a set of data points. In the context of decision trees, the Gini impurity is employed to determine the optimal splits at each node of the tree. in simple terms, Gini impurity quantifies the likelihood of incorrectly classifying a randomly chosen element in the dataset. A Gini impurity score of 0 indicates perfect purity (all elements belong to the same class), while a score of 0.5 implies maximum impurity (elements are evenly distributed across classes). Decision tree algorithms aim to minimize the Gini impurity when selecting feature splits, thereby creating nodes that result in more homogenous subsets and, consequently, a more accurate and predictive tree.

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

Ans) Yes, unregularized decision trees are prone to overfitting because they can become excessively complex, capturing noise and specific details in the training data. Without constraints, they may fit the training data too closely, leading to poor generalization on new data. Regularization techniques, such as limiting tree depth or pruning, are essential to strike a balance and prevent overfitting by promoting a more generalized tree structure.

6. What is an ensemble technique in machine learning?

Ans) An ensemble technique in machine learning involves combining the predictions of multiple individual models to improve overall performance and predictive accuracy. The idea is to leverage the strengths of diverse models and mitigate their weaknesses, leading to a more robust and accurate prediction. The two main types of ensemble techniques are: bagging and boosting

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

Ans)

BAGGING (BOOTSTRAP AGGREGATING)	BOOSTING
The simplest way of combining predictions that belong to the same type	A way of combining predictions that belong to the different types
Aim to decrease variance, not bias.	Aim to decrease bias, not variance.
Each model receives equal weight	Models are weighted according to their performance
Bagging tries to solve the over-fitting problem.	Boosting tries to reduce bias.
In this base classifiers are trained parallelly	In this base classifiers are trained sequentially
Example: The Random forest model	Example: The AdaBoost
Each model is built independently	New model are influenced by the performance of previously built models.

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

Ans) out of bag error are an estimate of the performance of a random forest classifier or regressor on unseen data. OOB error is calculated using out of bag samples , which are samples that are not used in the training of the model , it provides an unbiased estimate of the model performance. It is also used to tune the hyperparameters of a model. By using the OOB error as a performance metric , the hyperparameters of the model can be adjusted to improve its performance on unseen data. It can also be used to diagnose whether the model is overfitting or underfitting. In scikit-learn , the OOB error can be obtained by using OOB_SCORE_ attributeof the random forest classifier.

9. What is K-fold cross-validation?

Ans) machine learning‘s prime aim is to train the machine to be able to predict the outcome of the unseen data point . cross validation is a technique used in machine learning to evaluate the performance of a model on unseen data. It involves dividing the dataset into multiple fold or subsets , using one of these sets as a validation set and training the model on the remaining folds. This process is repeated many times. And results are averaged to produce a robust estimate. K fold cross validation is a type of cross validation where we split the data into k number of subsets and perform training on all the subsets and leave one k-1 subset for the evaluation of the trained set(validation set). We iterate it k times with different subset reserved for testing purpose each time.

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

Ans) Hyperparameter tuning is the process of selecting the optimal values for a machine learning 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 purpose of hyperparameter tuning is to find the best set of hyperparameters for a given machine learning model. This can improve the model’s performance on unseen data , prevent overfitting and reduce training time.

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

Ans) gradient descent is a widely used optimization algorithm in machine learning and deep learning that minimizes the cost function of a neural network model during training.

a large learning rate in gradient descent can cause the optimization algorithm to overshoot the optimal parameter values, leading to divergence or oscillations that hinder convergence. It may lead to overshooting and instability.

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

Ans) No, Non linear problems can’t be solved with logistic regression because it has linear decision surface. Logistic regression is inherently designed for linear decision boundaries, making it less suitable for capturing complex non linear relationship in data

13. Differentiate between Adaboost and Gradient Boosting.

Ans)

	Adaboost (adaptive boosting)	Gradient boosting
Objective	Focuses on adjusting the weights of incorrectly classified instances to improve model performance iteratively	Works by fitting a series of weak learners (typically decision trees) sequentially, where each subsequent model corrects the errors of the previous one
Weighing of data	Adjusts the weights of misclassified instances, assigning higher weights to them to make the model pay more attention to those cases	Adjusts the model by fitting subsequent weak learners to the residuals (the difference between predicted and actual values) of the previous models
Learning rate	Employs a learning rate to control the contribution of weak learner, reducing the impact of each model to prevent overfitting	Also uses a learning rate, controlling the step size in the direction of minimizing the loss function.
Model complexity	Typically uses simple models as weak learners, often decision stumps (shallow tress)	Can use more complex weak learners, usually decision trees, and allows for tuning parameters like depth
Parallelization	Can be parallelized as the weak learners are trained independently	Sequentially builds weak learners, limiting parallelization opportunities

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

Ans) The bias-variance trade-off is a fundamental concept in machine learning that refers to the balance between bias and variance in the predictive performance of a model.

1.Bias :

definition : Bias measures the error introduced by approximating a real-world problem with a simplified model.

characteristics : High bias often results in underfitting, where the model is too simple and unable to capture the underlying patterns in the data.

impact on predictions : Models with high bias tend to make systematic errors on both training and unseen data.

2.Variance :

Definition : Variance measures the model's sensitivity to changes in the training data.

Characteristics : High variance can lead to overfitting, where the model is too complex and captures noise in the training data, making it less generalizable to new, unseen data.

Impact on predictions : Models with high variance perform well on training data but may struggle with new, diverse datasets.

3.Trade off :

Objective : The bias-variance trade-off aims to find the right level of model complexity that minimizes both bias and variance, achieving good predictive performance on both training and unseen data.

Challenge : Decreasing bias often increases variance, and vice versa. Striking the right balance is crucial to building a model that generalizes well.

4.Model selection :

Simple models : Models with high bias and low variance (e.g., linear regression) may be suitable when the underlying relationship is relatively simple.

Complex models: Models with low bias and high variance (e.g., deep neural networks) may be necessary for complex relationships but require careful regularization to avoid overfitting.

5.Regularization :

Role : Regularization techniques, such as adding penalties to the model complexity, help control overfitting and strike a balance between bias and variance.

Example : L1 or L2 regularization in linear models, dropout in neural networks.

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

Ans) **SVM –** support vector machine

1.Linear kernel:

Description: The linear kernel is the simplest kernel and represents a linear decision boundary.

Function: It computes the dot product of feature vectors in the input space.

Use cases: Suitable for linearly separable data or when a linear decision boundary is appropriate.

2.Radial basis function kernel:

Description: The RBF kernel, also known as the Gaussian kernel, introduces non-linearity and is capable of capturing complex relationships.

Function: It measures the similarity between data points in the transformed space using a Gaussian distribution.

Use cases: Effective for non-linear, complex patterns in the data. It is a versatile choice, but can be sensitive to hyperparameter tuning.

3.Polynomial kernel:

Description: The polynomial kernel introduces non-linearity by computing the dot product raised to a specified power.

Function: It raises the dot product of feature vectors to a polynomial power, allowing the SVM to capture non-linear relationships.

Use cases: Suitable for datasets with polynomial decision boundaries. The degree of the polynomial is a crucial hyperparameter that influences the flexibility of the model.