Machine Learning

Q1)a. Explain Lasso Regression. Explain how Lasso Regression is used for

feature selection.

Ans: **Lasso Regression** (Least Absolute Shrinkage and Selection Operator) is a type of linear regression that uses **L1 regularization** to penalize the absolute size of the coefficients. The regularization term is added to the cost function to reduce the model's complexity and prevent overfitting.

Where:

1. **RSS** is the residual sum of squares.
2. **λ** is the regularization parameter (controls the strength of the penalty).
3. **βi** are the model coefficients.

Feature Selection in Lasso:

Lasso performs automatic feature selection by forcing the coefficients of some features to be exactly zero. The process works as follows:

* The L1 penalty added to the cost function (the sum of absolute values of coefficients) encourages sparsity.

* Features with coefficients shrinking to zero are effectively removed from the model.

* This allows Lasso to select only the most important feature & naking the model more efficient and reducing multicollinearity.

In short, Lasso regression not only fits the data but also helps identify and retain only the most relevant features for the prediction

task.

Q1)b. Define different regression models.

Ans: Here's a concise overview of different regression models:

1. Linear Regression: Models the relationship between a dependent variable and independent variables using a straight line. Best for linear relationships. 2. Ridge Regression: A variant of linear regression with L2 regularization to prevent overfitting by shrinking coefficients.

3. Lasso Regression: Similar to Ridge but uses L1 regularization, which can shrink some coefficients zero, thus performing feature selection.

4. Elastic Net Regression: Combines L1 and L2 regularization, balancing feature selection and coefficient shrinkage.

5. Polynomial Regression: Extends linear regression by modeling nonlinear relationships using polynomial terms.

6. Logistic Regression: Used for binary classification, mo-'-Is the probability of a categorical outcome (0 or 1)

7. K-Nearest Neighbors Regression

(KNN): Predicts the target value based on the average of the k nearest neighbors, useful for non-linear relationships.

Each model is suited for different types of data and tasks.

Q1)c. Describe the bias-variance trade-off and its relationship to under fitting and overfitting.

Ans: The bias-variance trade-off is a fundamental concept in machine learning that describes the balance between two sources of error in a

model:

1. Bias: The error introduced by assuming a simplistic model that doesn't capture the underlying patterns of the data. High bias leads to underfitting, where the model is too simple and performs poorly on both training and test data.

2. Variance: The error due to the model's sensitivity to fluctuations in the training data. High variance leads to overfitting, where the model comes too complex and captures noise or irrelevant patterns

Relationship to Underfitting and Overfitting:

* Underfitting occurs when the model has high bias (too simple), resulting in poor performance on both training and test

data.

* Overfitting occurs when the model has high variance (too complex), resulting in excellent performance on training data but poor generalization to new, unseen data.

The trade-off is about finding the right balance: minimizing both bias and variance.

Q3)a. Explain kernel methods which are suitabie for SVM.

Ans: Kernel methods in Support Vector Machines

(SVM) are used to transform non-linearly separable data into a higher-dimensional space where it becomes linearly separable.

This is achieved without explicitly computing the coordinates of the data in this higher-dimensional space, using a kernel function to compute the inner product in the transformed space directly.

Common Kernel Functions:

1. Linear Kernel: No transformation is applied, and the data is assumed to be already linearly separable.

* K(x, y) = xy

2. Polynomial Kernel: Maps the data into a higher-dimensional polynomial feature space.

* K(x, y) = (xy + c) d

where C is a constant and d is the degree of the polynomial.

3. Radial Basis Function (RBF) Kernel:

Maps the data into an infinite dimensional space and is effective

Role in SVM:

* Kernel methods allow SVM to learn non-

linear decision boundaries by implicitly working in a higher-dimensional space, without the need to compute the

transformed data explicitly.

* This helps SVM classify complex data that is not linearly separable in the original feature space.

Q3)b. What are advantages and disadvantages of K-NN?

Ans: Advantages of K-Nearest Neighbors (K-NN):

1. Simple and Intuitive: Easy to understand and implement.

2. No Training Phase: K-NN is a lazy learner, meaning it doesn't require a training phase, which can be beneficial for large datasets.

3. Non-Parametric: It makes no assumptions about the data distribution, useful for complex and non-linear data.

4. Flexible: Can be used for both classification and Regression tasks.

Disadvantages of K-NN:

1. Computationally Expensive: The prediction phase is slow, as it requires calculating distances to all training samples.

2. Sensitive to Irrelevant Features:

Performance can degrade with high-dimensional or noisy data.

3. Memory Intensive Requires storing the entire training dataset for prediction which can be inefficient for large

datasets.

4.Choosing K: The choice of the right value for K can significantly affect performance, and it might require cross-validation to tune.

Q3)c. What are different distance metrics are used in K-NN?

Ans: In K-Nearest Neighbors (K-NN), different distance metrics are used to calculate the

similarity between data points. Common ones include:

1. Euclidean Distance:

* The most widely used metric, measures the straight-line distance

between two points.

$$\textbf{Euclidean Distance} = |X - Y| = \sqrt{\sum_{i=1}^{i=n} (x_i - y_i)^2}$$

X: Array or vector X

Y: Array or vector Y

x_i: Values of horizontal axis in the coordinate plane

y_i: Values of vertical axis in the coordinate plane

n: Number of observations

2. Manhattan Distance:

Measures the sum of the absolute differences of the coordinates, useful in grid-like data.