MACHINE LEARNING

Answer.1 A) Least Square Error

Answer.2 A) Linear regression is sensitive to outliers

Answer.3 B) Negative

Answer.4 B) Correlation

Answer.5 C) Low bias and high variance

Answer.6 B) Predictive modal

Answer.7 D) Regularization

Answer.8 D) SMOTE

Answer.9 A) TPR and FPR

Answer.10 B) False

Answer.11 B) Apply PCA to project high dimensional data

Answer.12 D) It does not make use of dependent variable

Question 13. Explain the term regularization?

Answer. Regularization is a technique used in machine learning to prevent overfitting, which occurs when a model learns to perform well on training data but fails to generalize to unseen data. It does this by adding a penalty term to the loss function, discouraging the model from fitting the training data too closely.

Key Concepts of Regularization

1. **Overfitting vs. Underfitting**:
    - **Overfitting**: The model captures noise and fluctuations in the training data, leading to poor performance on new data.
    - **Underfitting**: The model is too simple to capture the underlying patterns in the data.

2. **Loss Function**: In machine learning, we often minimize a loss function (e.g., Mean Squared Error for regression). Regularization modifies this loss function by adding a penalty for large coefficients.

3. **Regularization Techniques**:

- **L1 Regularization (Lasso Regression)**:
  - Adds the absolute value of the coefficients as a penalty term.
  - The loss function becomes: $L = \text{Loss} + \lambda \sum |wi|$
  - Where λ is the regularization parameter, and w_i are the coefficients.
  - L1 regularization can lead to sparse models, meaning some coefficients may be exactly zero, effectively selecting a subset of features.

L2 Regularization (Ridge Regression):

- Adds the squared value of the coefficients as a penalty term.
- The loss function becomes: $L = \text{Loss} + \lambda \sum wi2$
- L2 regularization tends to distribute the weight across all features and is less likely to produce sparse models.

Elastic Net: Combines both L1 and L2 regularization. This approach is useful when there are many correlated features in the dataset.

4. **Regularization Parameter (λ)**: The strength of the regularization is controlled by the parameter λ:

- A higher λ value increases the penalty for larger coefficients, promoting simpler models.
- A lower λ value makes the regularization effect weaker, allowing the model to fit the training data more closely.

Benefits of Regularization

- **Improved Generalization**: By reducing overfitting, regularization helps the model perform better on unseen data.
- **Feature Selection**: In the case of L1 regularization, it can help identify the most important features by shrinking less important feature weights to zero.

Conclusion

Regularization is a crucial concept in machine learning that helps balance the trade-off between fitting the training data well and ensuring that the model generalizes effectively to new data. It’s widely used in various algorithms, including linear regression, logistic regression, and neural networks.

Question 14. Which particular algorithms are used for regularization?

Answer. Regularization is a fundamental technique in machine learning used to prevent overfitting by adding a penalty term to the model's loss function, discouraging excessive complexity. Several algorithms incorporate regularization to enhance their generalization capabilities:

1. Linear Regression

- **Ridge Regression (L2 Regularization):** Introduces a penalty proportional to the square of the coefficients, effectively shrinking them towards zero but not eliminating any (GeeksforGeeks).

- **Lasso Regression (L1 Regularization):** Adds a penalty equal to the absolute value of the coefficients, which can drive some coefficients to zero, performing feature selection (GeeksforGeeks).

- **Elastic Net:** Combines both L1 and L2 regularization, balancing between coefficient shrinkage and feature selection (GeeksforGeeks).

2. Logistic Regression

- **L2 Regularization:** Penalizes the square of the coefficients to prevent large weights, aiding in generalization (Simplilearn).

- **L1 Regularization:** Encourages sparsity by penalizing the absolute value of the coefficients, leading to simpler models (Simplilearn).

3. Neural Networks

- **Weight Decay (L2 Regularization):** Adds a penalty to the weights during training, preventing them from becoming too large and ensuring smoother training dynamics.

- **Dropout:** Randomly sets a fraction of the input units to zero at each update during training, which helps prevent overfitting by ensuring the network doesn't rely too heavily on any single input.

- **Early Stopping:** Monitors the model's performance on a validation set and halts training when performance starts to degrade, preventing overfitting to the training data.

4. Support Vector Machines (SVM)

- **L2 Regularization:** Incorporated in the hinge loss function, it helps balance the trade-off between maximizing the margin and minimizing classification errors (GeeksforGeeks).

5. Decision Trees

- **Pruning:** Involves removing branches that have little significance, thereby reducing the complexity of the model and preventing overfitting (GeeksforGeeks).

These regularization techniques are pivotal in developing models that generalize well to unseen data, ensuring robust and reliable predictions.

Question 15. Explain the term error present in linear regression equation?

Answer. In linear regression, **error** (also known as **residual**) refers to the difference between the actual values of the dependent variable and the values predicted by the linear regression model. The error captures how far off the model's predictions are from the true observed data points.

Mathematical Representation:

The basic equation for linear regression is:$y = \beta 0 + \beta 1x1 + \beta 2x2 + \cdots + \beta nxn + \epsilon$

Where:

- y: Actual observed value (target).
- $y\hat{} = \beta 0 + \beta 1x1 + \cdots + \beta nxn$: Predicted value from the linear regression model.
- $\epsilon = y - y\hat{}$: he error or residual, which represents the difference between the actual value (y) and the predicted value (y^).

Types of Errors:

1. **Residual Error (ϵ\epsilonϵ)**: This is the individual error for each data point, the difference between the actual and predicted values.

$$\epsilon = y - y\hat{}$$

2. **Total Error**: When assessing the overall error in a model, you can aggregate residuals across the dataset using common error metrics, such as:

.Mean Absolute Error (MAE): Average of the absolute differences between actual and predicted values.

$$\mathrm{MAE} = n1i = 1\sum n \mid yi - y\text{^}i \mid$$

.Mean Squared Error (MSE): Average of the squared differences between actual and predicted values.

$$\mathrm{MSE} = n1i = 1\sum n(yi - y\text{^}i)2$$

.Root Mean Squared Error (RMSE): The square root of the MSE, used to keep the error in the same unit as the dependent variable.

$$\mathrm{RMSE} = \mathrm{MSE}$$

Importance of Error in Linear Regression:

- **Model Evaluation**: The size and pattern of errors indicate how well the model is capturing the relationships in the data. Large errors suggest poor model performance.

- **Overfitting and Underfitting**: If errors are too small on training data but large on new data, the model might be overfitting (too complex). Conversely, large errors on both training and testing data indicate underfitting (too simple).

- **Residual Analysis**: Analysing residuals can help detect issues like non-linearity, outliers, or heteroscedasticity (when the variance of residuals varies across the range of predicted values).

Example: If a model predicts a house price $(\mathrm{y}^{=250,000})$ but the actual price is (y=280,000), the error is:
$\epsilon = 280{,}000 - 250{,}000 = 30{,}000$

This residual indicates that the model underpredicted the house price by 30,000 units.

In summary, error in linear regression measures how far the model's predictions are from the actual values and is used to assess the model's performance and fit.