Name: Prachi Shedge
Roll No: 281050
Batch: A2
Assignment 7

Statement

In this assignment, we aim to:
a) Apply Data Pre-processing techniques such as Label Encoding and Data Transformation.
b) Perform Data Preparation by splitting the dataset into training and testing sets.
c) Implement a Decision Tree Classifier to predict student admission based on GRE and academic scores.
d) Evaluate the model using accuracy, confusion matrix, and classification report.

Objective

1. Understand how to analyze structured student admission data using Python.
2. Apply classification techniques to predict admission chances.
3. Evaluate model performance using classification metrics.
4. Develop visualization techniques to interpret the decision tree.

Resources Used

- **Software:** Visual Studio Code
- **Libraries:** Pandas, NumPy, Matplotlib, Seaborn, Scikit-learn

Introduction to Classification and Decision Trees

Decision Tree Classification is a supervised machine learning technique used for predicting categorical outcomes. In this assignment, we use a Decision Tree Classifier to predict whether a student will get admitted based on GRE and CGPA scores.

Key Concepts:

- **Decision Tree Algorithm**: A tree-based model that makes decisions by splitting data at different thresholds.
- **Gini Impurity & Entropy**: Measures of purity used to decide the best feature split.
- **Model Evaluation**: Uses accuracy, confusion matrix, and classification report.
- **Data Pre-processing**: Normalizing numerical features for better model performance.

Basic Functions Used

1. pd.read_csv() - Loads data from a CSV file.
2. rename() - Renames dataset columns for consistency.
3. StandardScaler() - Normalizes feature values.
4. train_test_split() - Splits data into training and testing sets.
5. DecisionTreeClassifier() - Implements the classification model.

6. fit() - Trains the model on the dataset.
7. predict() - Predicts admission status for new students.
8. accuracy_score() - Measures the model's performance.
9. confusion_matrix() - Evaluates classification errors.
10. classification_report() - Provides precision, recall, and F1-score.
11. plot_tree() - Visualizes the decision tree structure.

Methodology

1. **Data Collection and Exploration**
    - The dataset contains information on GRE scores, TOEFL scores, CGPA, and research experience.
    - Initial steps include loading and inspecting the dataset.
2. **Feature Selection & Preprocessing**
    - Select relevant features (GRE Score, CGPA) for classification.
    - Normalize features using StandardScaler() for consistency.
    - Create a binary target variable (Admitted = 1 if Chance of Admit >= 0.5, else 0).
3. **Model Training**
    - Split data into training (80%) and testing (20%) sets.
    - Train a Decision Tree Classifier with a maximum depth of 4.
4. **Model Evaluation**
    - Calculate accuracy, confusion matrix, and classification report.
    - Analyze how well the model predicts admission status.
5. **Visualization**
    - Plot the Decision Tree to understand decision splits.

Results & Observations

- The Decision Tree model provided a structured approach to predicting student admissions.
- Evaluation metrics indicated good classification accuracy.
- The visualization helped interpret decision-making in the model.

Advantages of Decision Trees

1. **Interpretability**: Easy to understand and visualize.
2. **No Data Assumptions**: Unlike regression, decision trees do not assume a linear relationship.
3. **Feature Importance**: Helps identify the most influential factors for admission.

Disadvantages

1. **Overfitting**: Trees can become too complex without proper pruning.
2. **Data Sensitivity**: Small changes in data can significantly alter the tree structure.

Conclusion

This assignment demonstrated the use of a **Decision Tree Classifier** for predicting student admission chances based on **GRE Score** and **CGPA**. The analysis covered data preprocessing, model training, evaluation, and visualization. The model provided valuable insights into factors affecting admission chances, showcasing the significance of classification techniques in real-world applications.