**Name**: Shubham Dahane  
**Class**: CSE (AI) B Division  
**Roll No.**: 282015  
**PRN No.**: 22310312  
**Subject**: Machine Learning

**Assignment 7 – Admission Prediction using Decision Tree Classifier**

**Objective**

To build a classification model using **Decision Tree Classifier** to assist counselors in predicting whether a student will be admitted (1) or not admitted (0) to a foreign university based on GRE, TOEFL, CGPA, and other academic indicators.

**Dataset Description**

The dataset contains the following features:

* **GRE Score** (out of 340)
* **TOEFL Score** (out of 120)
* **University Rating** (1 to 5)
* **SOP** – Statement of Purpose strength (1 to 5)
* **LOR** – Letter of Recommendation strength (1 to 5)
* **CGPA** – Undergraduate GPA (out of 10)
* **Research** – Research Experience (0 = No, 1 = Yes)
* **Chance of Admit** – Continuous values between 0 and 1, converted to binary:
  + < 0.75 → **0** (Not Admitted)
  + ≥ 0.75 → **1** (Admitted)

**1. Data Preprocessing**

* **Dataset Loading**: Imported using **pandas**.
* **Missing Values**: Checked using isnull().sum(), and none were found.
* **Column Renaming**: Cleaned up for easier processing (if needed).
* **Target Variable Transformation**:
  + The Chance of Admit column was **binarized** to create a new target column called **Admitted**.

df['Admitted'] = (df['Chance of Admit'] >= 0.75).astype(int)

* **Exploratory Visualization**:
  + TOEFL score histogram and GRE vs TOEFL scatter plot were plotted to understand score distribution and correlation.

**2. Data Preparation**

* **Irrelevant Columns Removed**:
  + Serial No. and Chance of Admit were dropped.
* **Feature Matrix (X)**: All independent variables.
* **Target Variable (y)**: Admitted (binary).
* **Train-Test Split**:
  + 80% for training and 20% for testing using train\_test\_split.

**3. Model Building**

* **Model Used**: DecisionTreeClassifier from sklearn.tree.
* **Training**: Applied .fit() method on training data.
* **Prediction**: Used .predict() on test data.

**4. Model Evaluation**

The following metrics were calculated using sklearn.metrics:

| **Metric** | **Value** |
| --- | --- |
| **Accuracy Score** | 0.85 |
| **Precision Score** | 0.744 |
| **Recall Score** | 0.97 |
| **F1 Score** | 0.842 |

* **Confusion Matrix**: Generated to visualize true positives, true negatives, false positives, and false negatives.
* **Classification Report**: Detailed class-wise metrics printed using classification\_report.

A blue squares with white text

AI-generated content may be incorrect.

**5. Decision Tree Visualization**

* Used plot\_tree() from sklearn.tree to visualize the trained decision tree with all feature splits and paths.

plt.figure(figsize=(40, 40))

tree.plot\_tree(clf, filled=True, fontsize=16)

plt.show()

A graph with a bar and a number of text

AI-generated content may be incorrect.

A diagram of a structure

AI-generated content may be incorrect.

**6. Additional Observations & Enhancements**

* **Class Imbalance Check**: Verified class distribution in the target variable to ensure balanced training.
* **Hyperparameter Tuning**: Could be applied in future (e.g., max\_depth, min\_samples\_split) to prevent overfitting.
* **Feature Importance**: The tree inherently provides insights into which features most influence predictions.
* **Model Alternatives**: To further improve accuracy and generalization:
  + Try **Random Forest**, **Gradient Boosting**, or **XGBoost** classifiers.
* **Cross-Validation**: Implementing cross\_val\_score can give a better estimate of model performance.

**Conclusion**

The Decision Tree Classifier achieved **85% accuracy** in predicting admission chances. With high **recall**, the model is particularly good at identifying students who are likely to be admitted. This tool can assist educational counselors in quickly assessing student profiles. Further improvements using ensemble techniques can boost predictive performance and robustness.