**Assignment 4**

**Aim:** Implementation of Decision Tree Classifier on Loan Dataset

**Objective:** To implement and evaluate a Decision Tree Classifier using Python to classify loan approval status based on various features

**Importance of Decision Tree**: Decision Trees are a strong supervised learning algorithm for classification and regression problems. A Decision Tree divides the data into branches on the basis of feature values in order to predict. The purpose of this assignment is to pre-process data, train a Decision Tree Classifier, evaluate its performance, and display the results on the loan dataset.

Decision Trees are popular in machine learning because of their intuitive nature and interpretability. The key benefits are:

● **Easy to Interpret and Understand:** The tree structure is intuitive.

● **Handles Both Categorical and Numerical Data:** Unlike many other algorithms, Decision Trees can process mixed data types.

● **Minimal Data Preprocessing Required:** No need for extensive feature scaling or transformation.

● **Captures Non-Linear Relationships:** Can efficiently model complex decision boundaries.

● **Useful for Feature Selection:** Identifies important variables based on how frequently they are used to split nodes.

**Dataset:** The data used in this task is loan.csv, which has numerical and categorical features influencing the approval of the loan. The most significant features are:

● ApplicantIncome: Salary of the applicant.

● CoapplicantIncome: Salary of co-applicant.

● LoanAmount: Loan amount applied for.

● Loan\_Amount\_Term: Length of the loan in months.

● Credit\_History: Applicant's credit history (1 = good, 0 = bad/missing). ● Property\_Area: Location of the property (Urban, Semiurban, Rural). ● Loan\_Status: Target variable of loan approval (Y/N).

**Steps of Implementation:**

1. **Importing Libraries:**

o Python libraries such as **Pandas, NumPy, Matplotlib, Seaborn, and Scikit-Learn** are used for data handling, visualization, and model training.

2. **Loading the Dataset:**

o The dataset is imported using Pandas, and an initial exploration is performed using .shape(), .head(), and .info().

3. **Data Preprocessing:**

o Encoding categorical variables using one-hot encoding.

o Handling missing values by filling categorical columns with mode and numerical columns with median.

o Splitting the dataset into 67% training and 33% testing.

4. **Training the Decision Tree Model:**

o A Decision Tree Classifier with the **Gini Index** as the splitting criterion and a maximum depth of 3 is used.

5. **Making Predictions:**

o The trained model predicts loan approval on the test dataset.

6. **Model Evaluation:**

o Performance metrics such as **Accuracy Score, Confusion Matrix, and Classification Report** are computed.

7. **Visualization of Results:**

o The **Decision Tree structure** is plotted to visualize decision-making. **Conclusion:**

● The Decision Tree Classifier was successfully trained for loan approval prediction. ● Accuracy Score: Reports the overall performance of the model.

● Confusion Matrix: Gives insight into the mistakes in classification. ● Classification Report: Shows Precision, Recall, and F1-score for validating performance.

● Decision Tree Visualization: Aids in the interpretation of decision-making using input features.

**References:**

● https://www.geeksforgeeks.org/decision-tree/

**Github Repository:**

https://github.com/AdhikSarak/MachineLearningLab/blob/main/Assignment4\_ML.ipynb