PPML MINIPROJECT REPORT

LOAN APPROVAL PREDICTION USING MACHINE LEARNING

-AMIRTA VARSHINI R

220901005

III-EEE-A

**Aim**

To analyze a dataset on loan approval prediction, preprocess the data, and evaluate the performance of various machine learning classifiers (Decision Tree) for predicting loan approval status.

**Abstract**

Loan approval prediction is crucial for financial institutions to minimize risks and streamline decision-making processes. This project involves the exploration and preprocessing of a dataset containing loan application details. The data is analyzed to understand the distribution of categorical variables and correlations between features. Missing values are handled using imputation, and categorical variables are encoded. The dataset is then split into training and testing sets to evaluate the performance of four machine learning models. Performance metrics, including accuracy scores, are used to compare the classifiers.

**Program Code:**

**import pandas as pd**

**import numpy as np**

**import matplotlib.pyplot as plt**

**import seaborn as sns**

**data = pd.read\_csv("LoanApprovalPrediction.csv")**

**print(data.head())**

**categorical\_cols = data.select\_dtypes(include='object').columns**

**print("Categorical variables:", len(categorical\_cols))**

**plt.figure(figsize=(18, 36))**

**for index, col in enumerate(categorical\_cols, 1):**

**y = data[col].value\_counts()**

**plt.subplot(11, 4, index)**

**plt.xticks(rotation=90)**

**sns.barplot(x=y.index, y=y.values)**

**plt.title(f"Value counts for {col}")**

**plt.tight\_layout()**

**plt.show()**

**from sklearn.preprocessing import LabelEncoder**

**label\_encoder = LabelEncoder()**

**for col in categorical\_cols:**

**data[col] = label\_encoder.fit\_transform(data[col].astype(str))**

**remaining\_categorical\_cols = data.select\_dtypes(include='object').columns**

**print("Remaining categorical variables:", len(remaining\_categorical\_cols))**

**plt.figure(figsize=(12, 6))**

**sns.heatmap(data.corr(), cmap='BrBG', annot=True, fmt='.2f', linewidths=2)**

**plt.title("Correlation Heatmap")**

**plt.show()**

**data.fillna(data.mean(), inplace=True)**

**print("Remaining missing values:\n", data.isna().sum())**

**from sklearn.model\_selection import train\_test\_split**

**X = data.drop(['Loan\_Status'], axis=1)**

**Y = data['Loan\_Status']**

**X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, Y, test\_size=0.4, random\_state=1)**

**print("Training set shape:", X\_train.shape, Y\_train.shape)**

**print("Testing set shape:", X\_test.shape, Y\_test.shape)**

**from sklearn.tree import DecisionTreeClassifier**

**from sklearn import metrics**

**dtc = DecisionTreeClassifier(criterion='entropy', random\_state=7)**

**dtc.fit(X\_train, Y\_train)**

**Y\_train\_pred = dtc.predict(X\_train)**

**train\_accuracy = metrics.accuracy\_score(Y\_train, Y\_train\_pred) \* 100**

**print(f"Decision Tree Training Accuracy: {train\_accuracy:.2f}%")**

**Y\_test\_pred = dtc.predict(X\_test)**

**test\_accuracy = metrics.accuracy\_score(Y\_test, Y\_test\_pred) \* 100**

**print(f"Decision Tree Testing Accuracy: {test\_accuracy:.2f}%")**

**from sklearn.tree import plot\_tree**

**plt.figure(figsize=(20, 10))**

**plot\_tree(dtc, feature\_names=X.columns, class\_names=["Not Approved", "Approved"], filled=True)**

**plt.title("Decision Tree Visualization")**

**plt.show()**

**OUTPUT:**

**Education Self\_Employed \**

**0 LP001002 Male No 0.0 Graduate No**

**1 LP001003 Male Yes 1.0 Graduate No**

**2 LP001005 Male Yes 0.0 Graduate Yes**

**3 LP001006 Male Yes 0.0 Not Graduate No**

**4 LP001008 Male No 0.0 Graduate No**

**ApplicantIncome CoapplicantIncome LoanAmount Loan\_Amount\_Term \**

**0 5849 0.0 NaN 360.0**

**1 4583 1508.0 128.0 360.0**

**2 3000 0.0 66.0 360.0**

**3 2583 2358.0 120.0 360.0**

**4 6000 0.0 141.0 360.0**

**Credit\_History Property\_Area Loan\_Status**

**0 1.0 Urban Y**

**1 1.0 Rural N**

**2 1.0 Urban Y**

**3 1.0 Urban Y**

**4 1.0 Urban Y**

**Categorical variables: 7**

**Remaining missing values:**

**Loan\_ID 0**

**Gender 0**

**Married 0**

**Dependents 0**

**Education 0**

**Self\_Employed 0**

**ApplicantIncome 0**

**CoapplicantIncome 0**

**LoanAmount 0**

**Loan\_Amount\_Term 0**

**Credit\_History 0**

**Property\_Area 0**

**Loan\_Status 0**

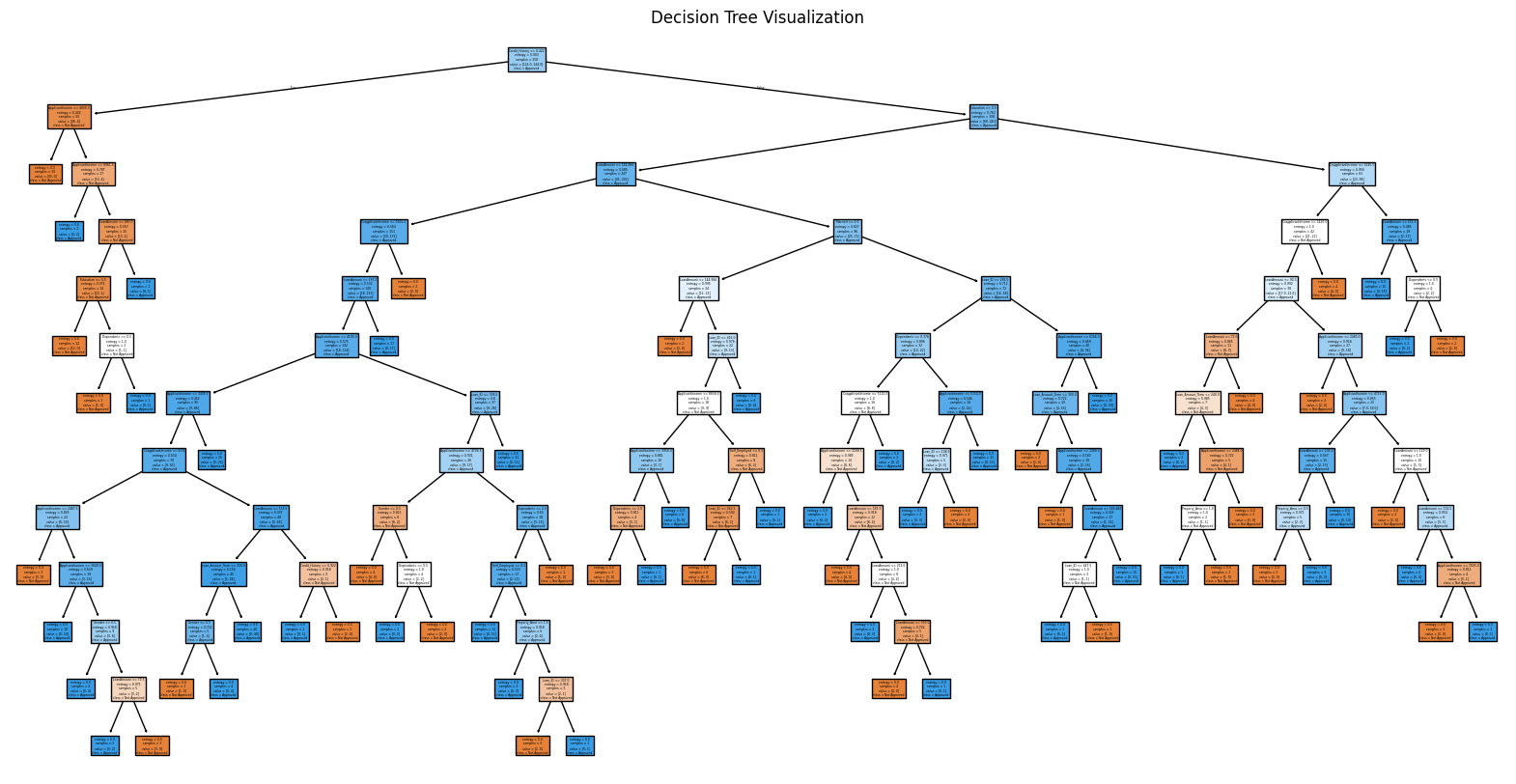
**dtype: int64**

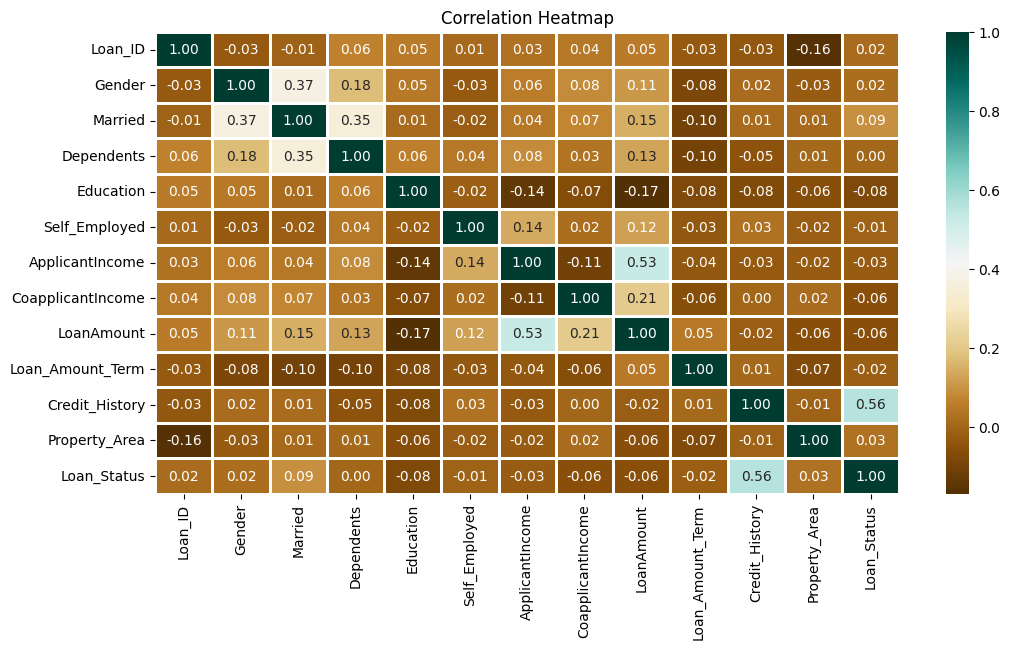
**Training set shape: (358, 12) (358,)**

**Testing set shape: (240, 12) (240,)**

**Decision Tree Training Accuracy: 100.00%**

**Decision Tree Testing Accuracy: 66.67%**





**Colab link:**

[**https://colab.research.google.com/drive/1hDS1NTeVqw1RjLw10zAWVkbQT8cTenQL?usp=sharing**](https://colab.research.google.com/drive/1hDS1NTeVqw1RjLw10zAWVkbQT8cTenQL?usp=sharing)

**Result:**

Thus the loan approval prediction is done using machine learning algorithm.