# 

Submitted To: Amanpal Singh Rayal Submitted By: Chiranjeev Sunamudi

Registration no: 12211481

Roll no.: 08

Section: EM301

Course code: INT234

# INDEX

1. INTRODUCTION…………………………………………………………………………………………………1
2. OBJECTIVE…………………………………………………………………………………………………………1
3. FULL CODE PREVIEW……………………………………………………………………………………….2-6
4. FULL CODE OUTPUT………………………………………………………………………………………..7-16
5. LIBRARY EXPALINATION…………………………………………………………………………………17-21
6. CONCLUSION……………………………………………………………………………………………………21

# Loan Prediction Project - Detailed Explanation & Code

# Introduction

Predictive Analytics leverages historical data to make informed predictions about future outcomes. In the context of financial institutions, predicting whether a loan application will be approved is a critical task. Machine learning enables this by learning patterns from past data and using them to make predictions for new cases.

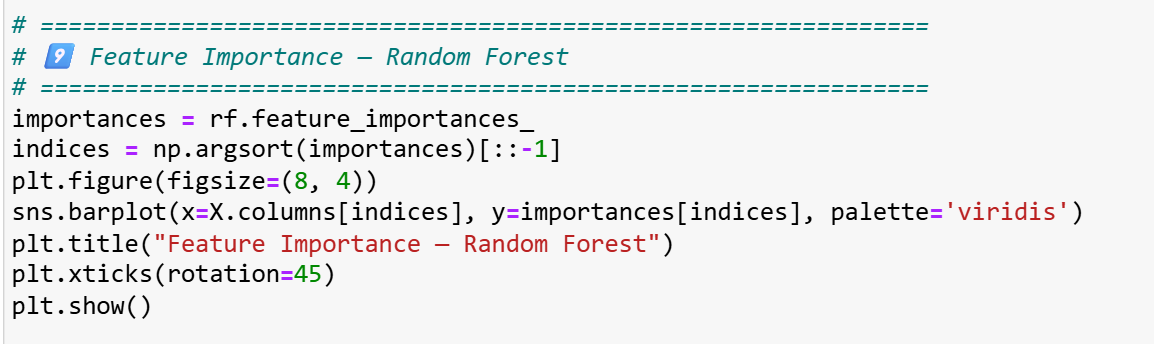
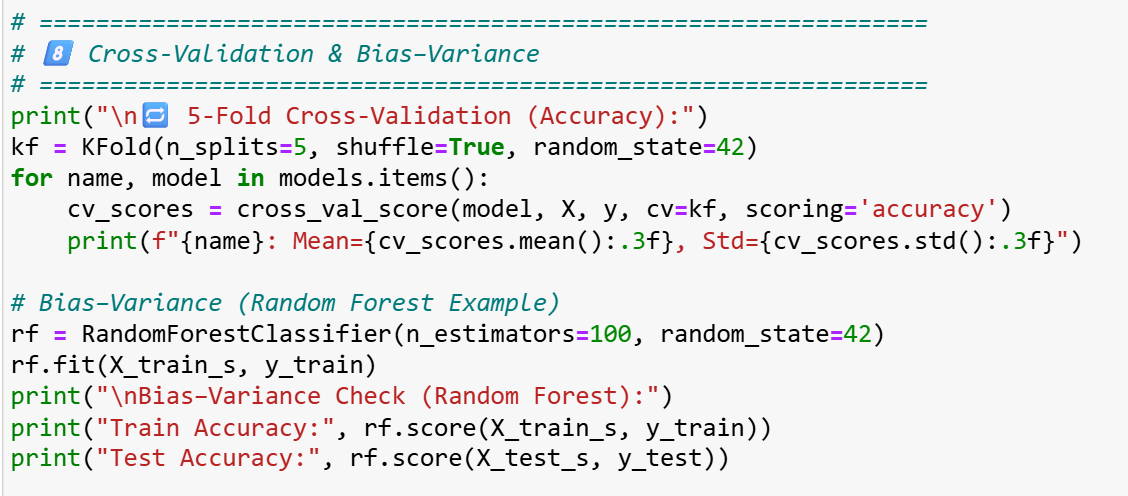
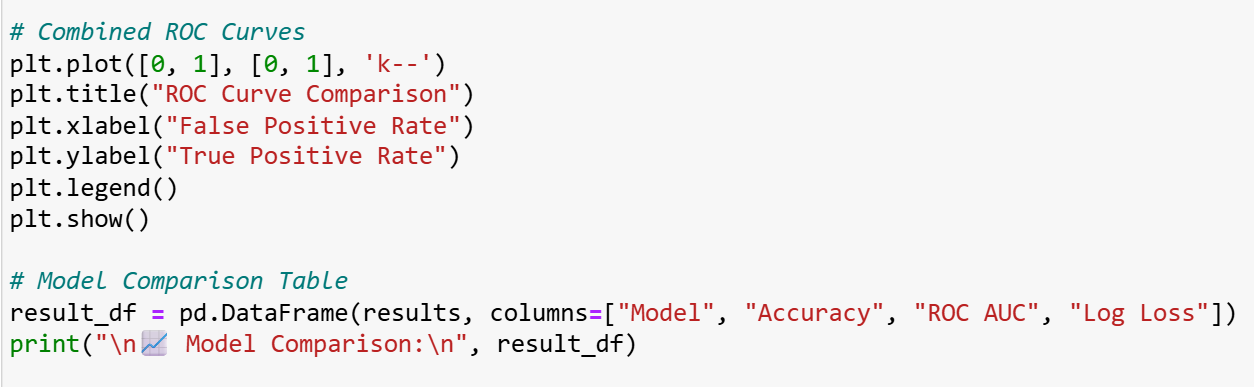
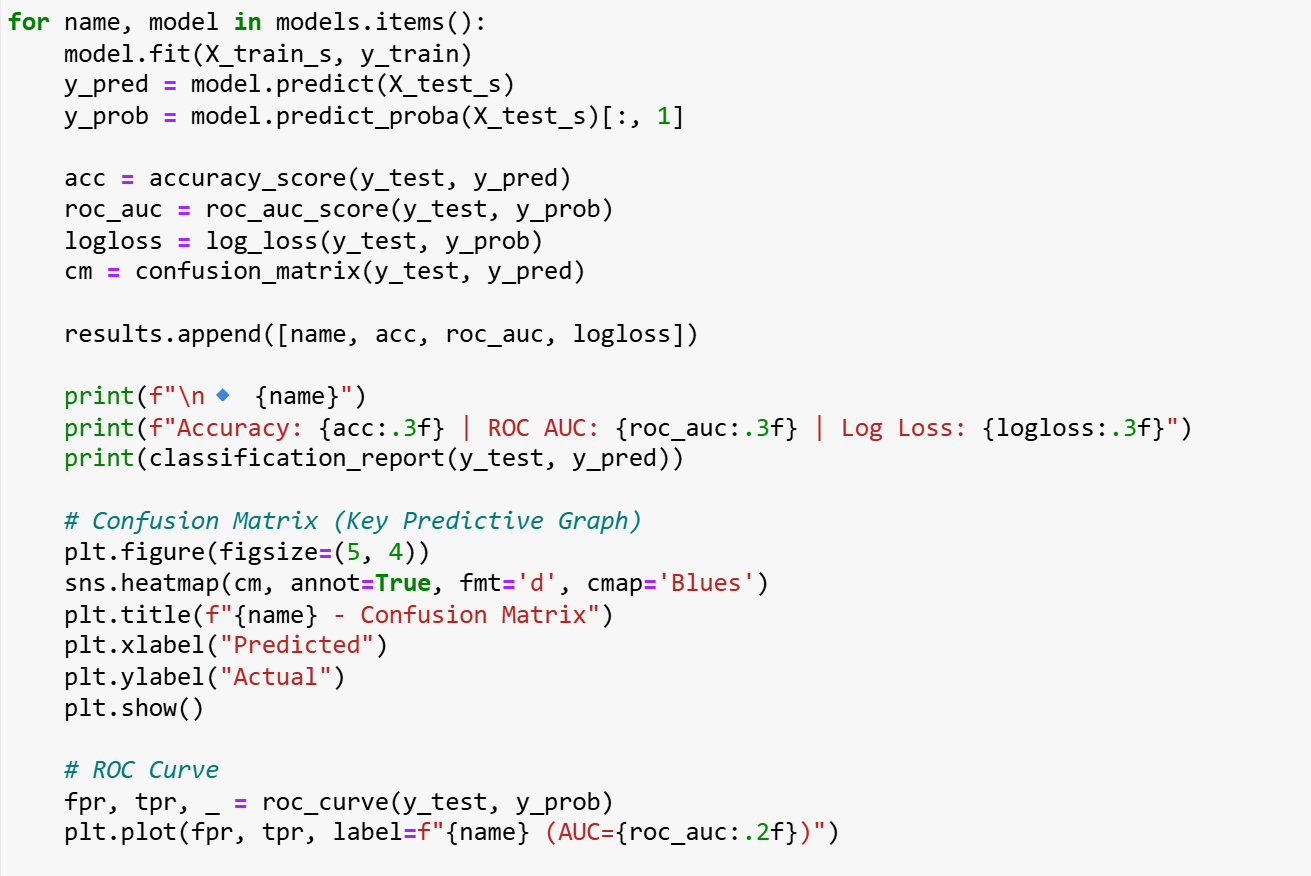
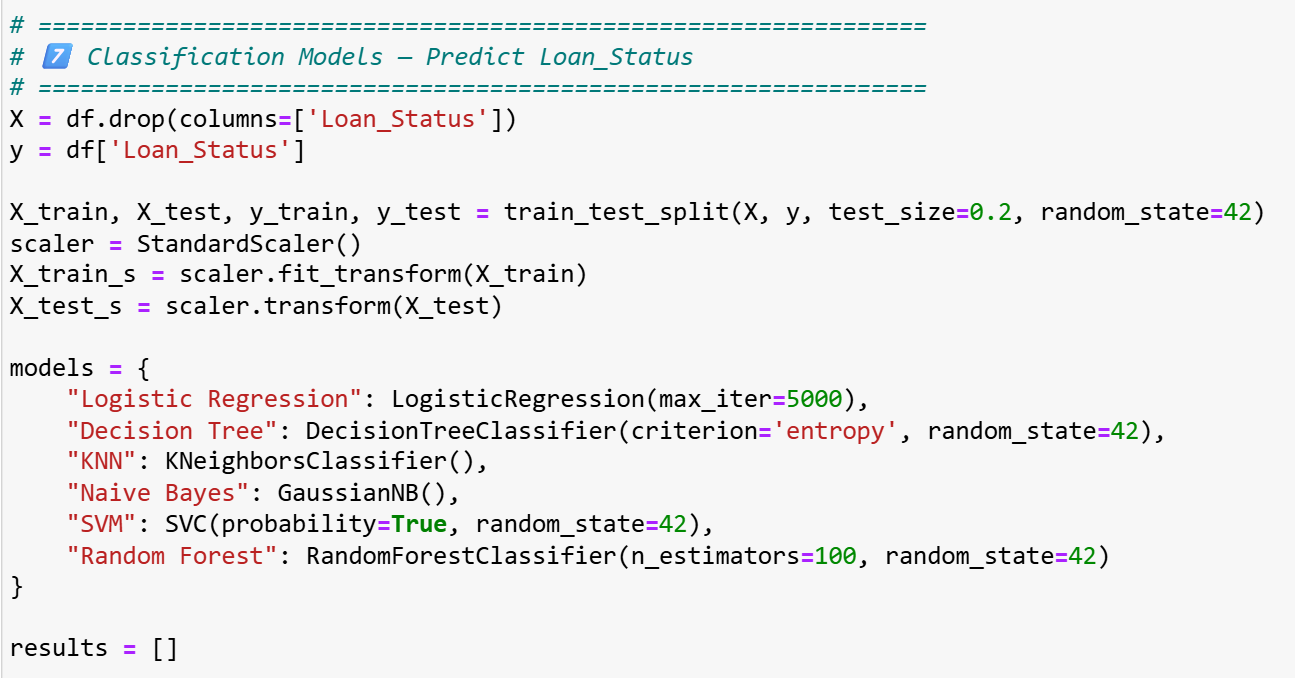
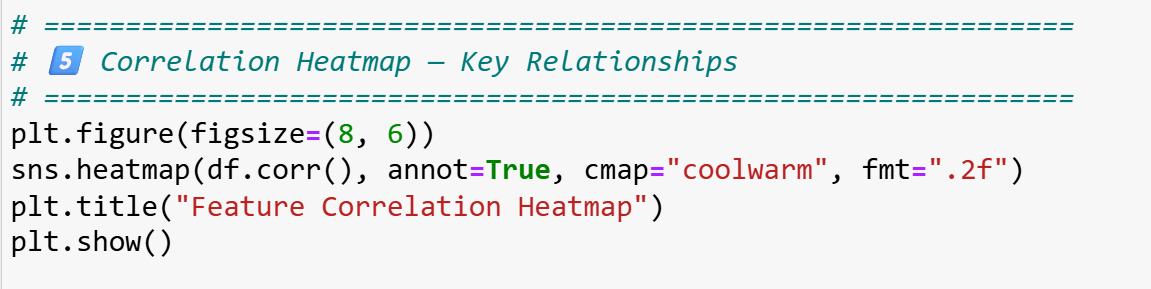
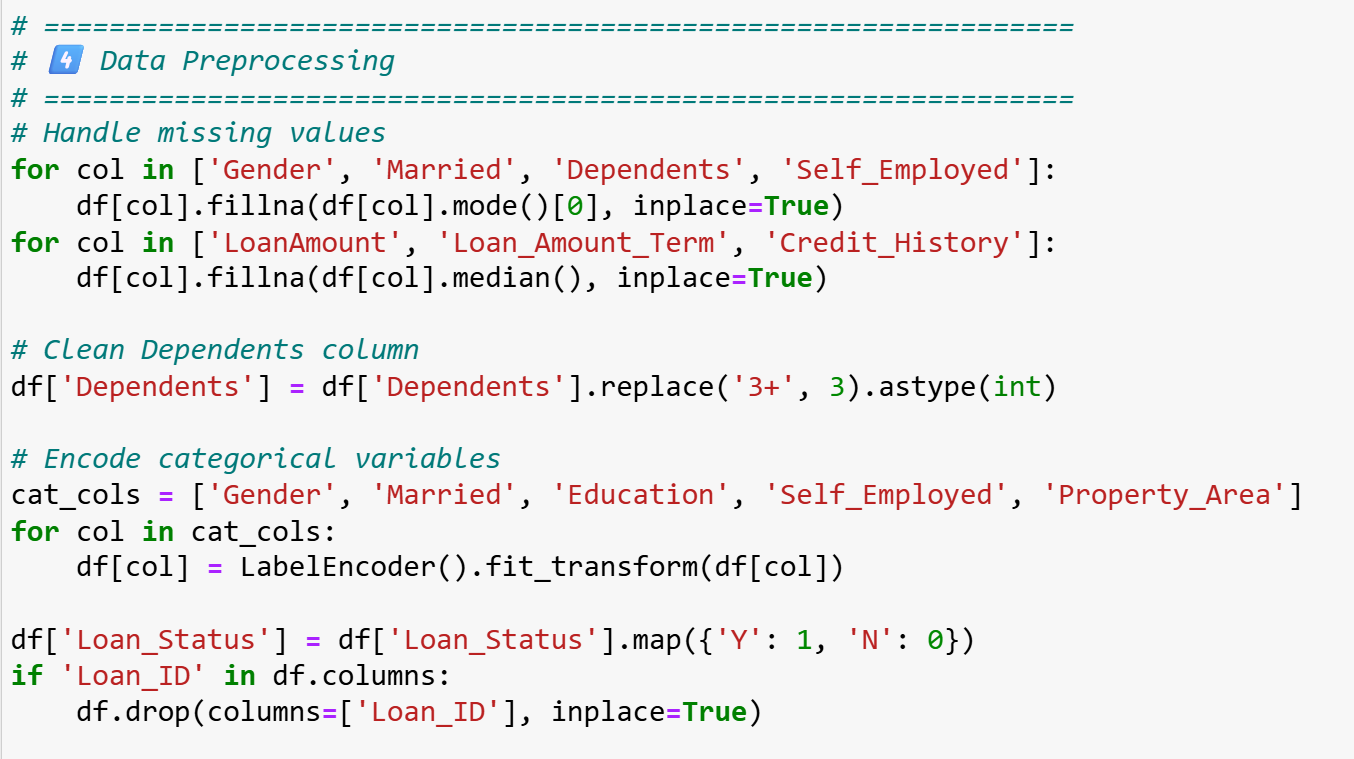
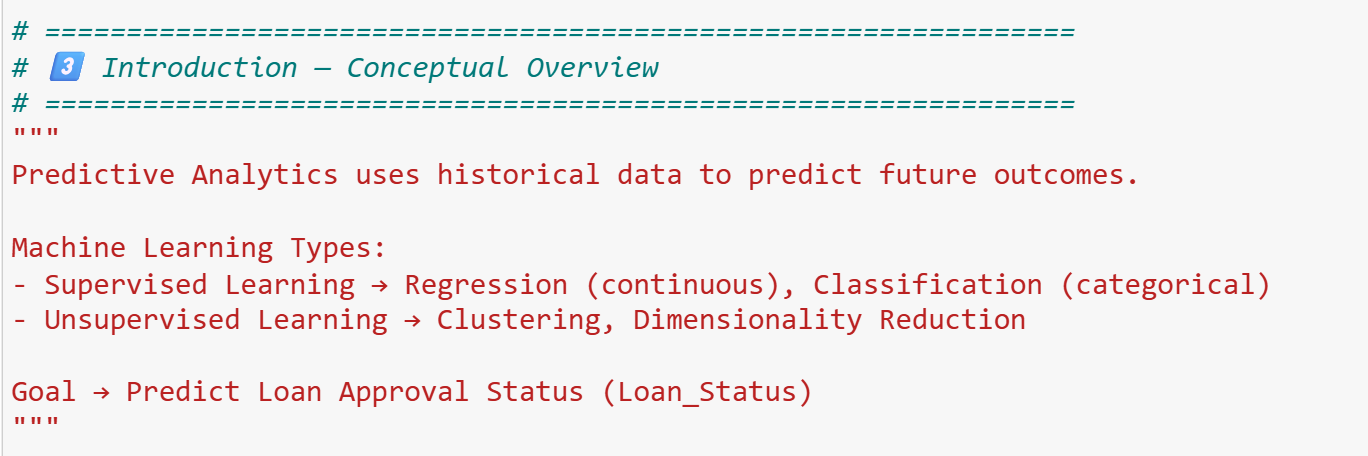
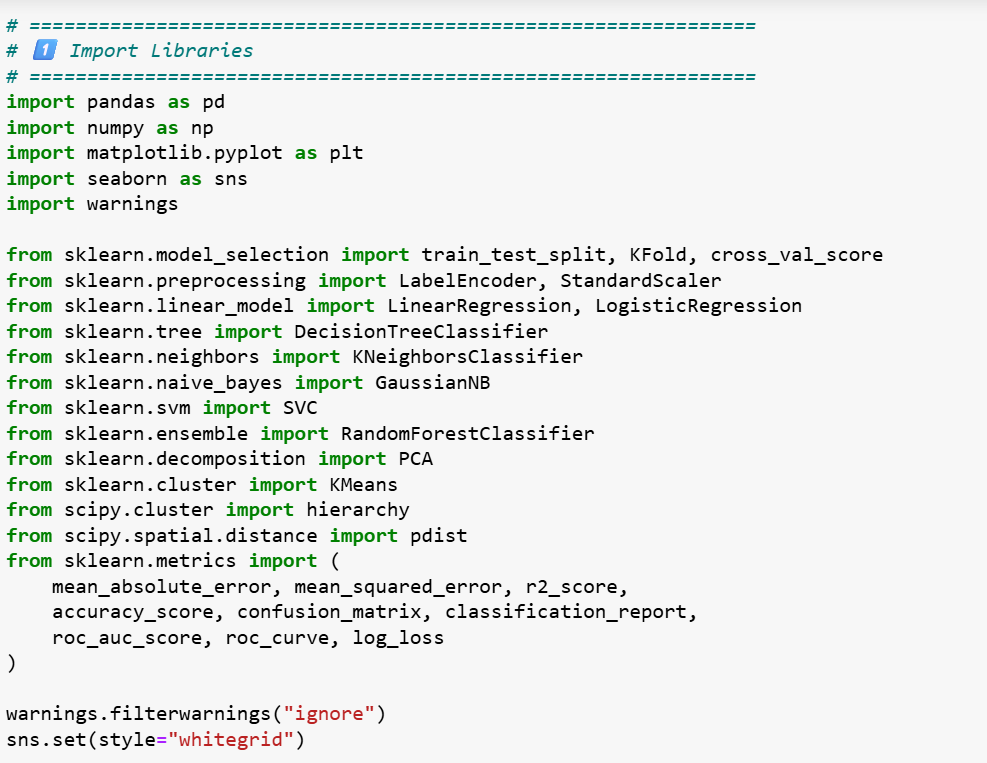
This project demonstrates the application of **supervised learning** techniques (regression and classification) to predict loan approval, along with **unsupervised learning** (clustering and dimensionality reduction) for deeper data insights. Key machine learning concepts such as data preprocessing, model evaluation, cross-validation, feature importance, and handling convergence/iteration limitations are also covered.

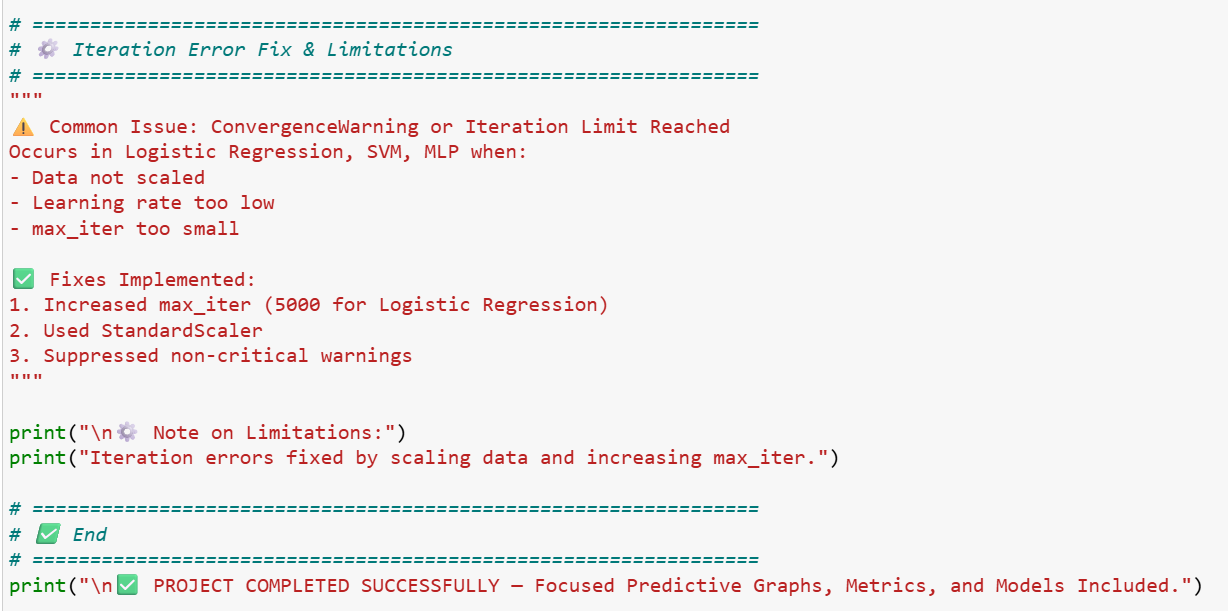
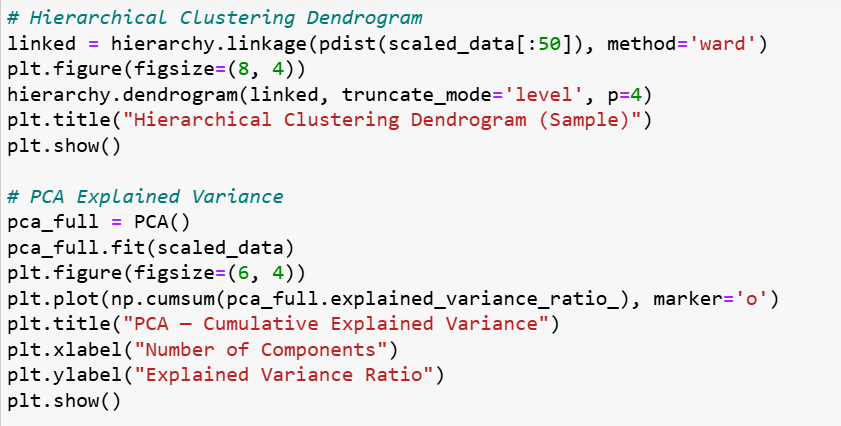
# Objectives

The primary objectives of this project are:

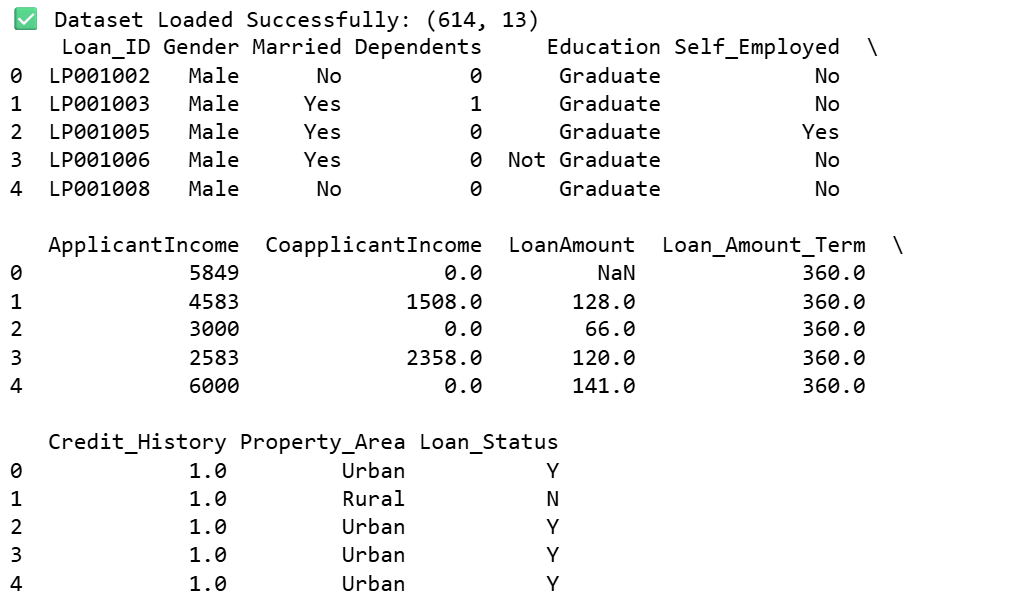
1. **Data Understanding & Preprocessing**
   * Clean the dataset, handle missing values, encode categorical variables, and scale features for better model performance.
2. **Regression Analysis**
   * Predict numerical outcomes (Loan Amount) using linear regression and evaluate model performance using MAE, MSE, RMSE, and R².
3. **Classification Modelling**
   * Predict whether a loan will be approved using various classification algorithms: Logistic Regression, Decision Tree, KNN, Naive Bayes, SVM, and Random Forest.
   * Evaluate models using Accuracy, ROC AUC, Log Loss, Confusion Matrix, and ROC Curves.
4. **Model Validation & Performance Analysis**
   * Apply K-Fold cross-validation to assess model stability.
   * Analyze bias–variance trade-off to understand underfitting and overfitting.
5. **Unsupervised Learning & Visualization**
   * Perform K-Means clustering, hierarchical clustering, and PCA for pattern detection, dimensionality reduction, and visualization of class separability.
6. **Handling Model Limitations**
   * Address common issues such as convergence warnings and iteration limits in models like Logistic Regression and SVM.

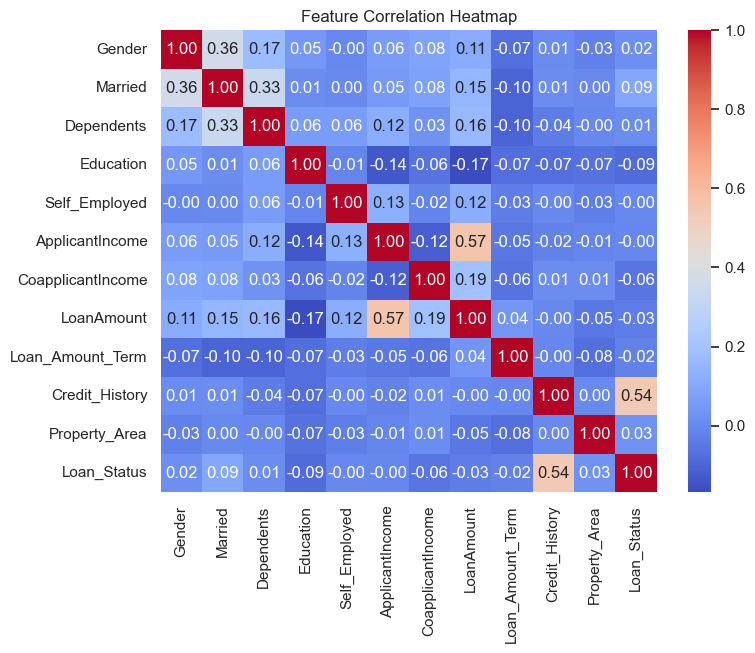
## Full Code

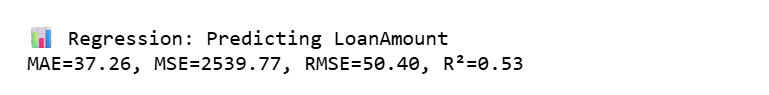


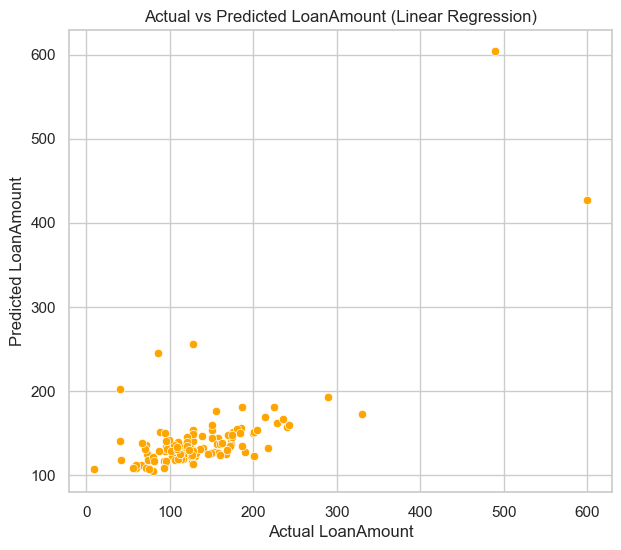


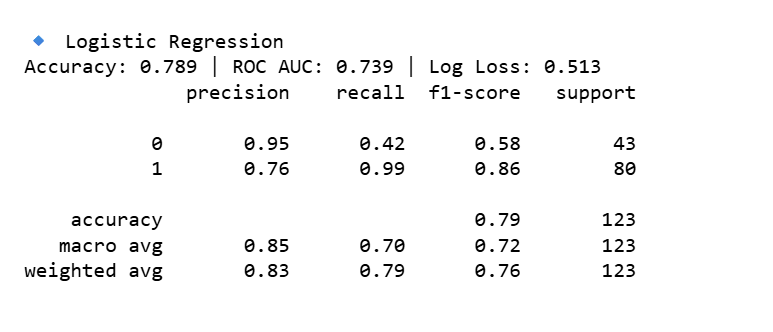
# Full Code Output

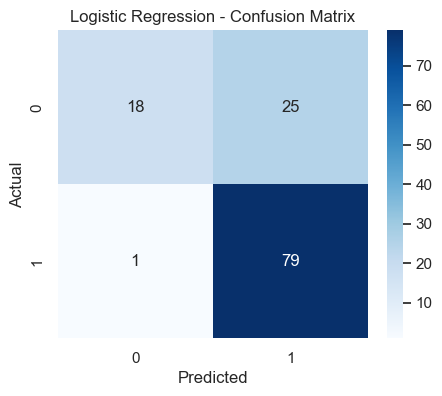


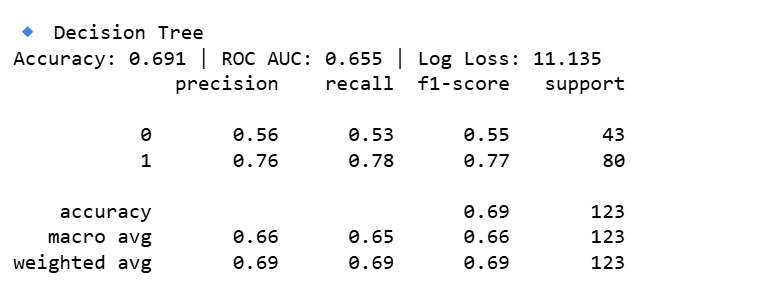


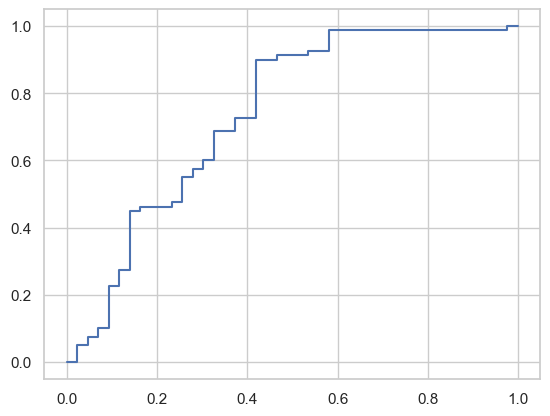


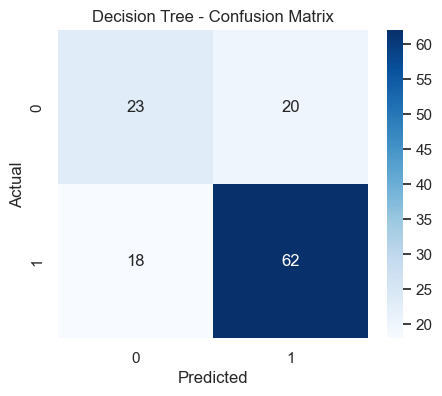


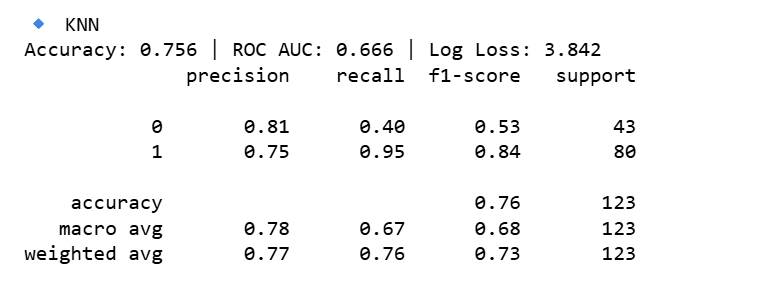


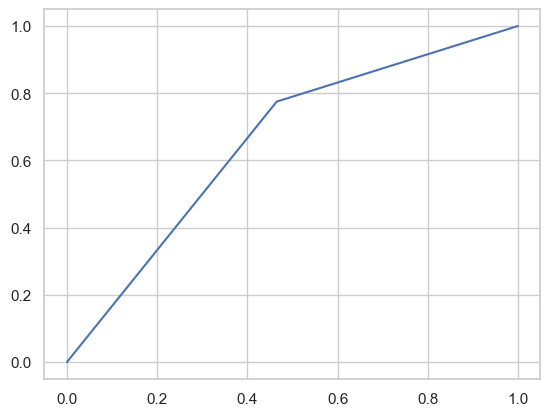


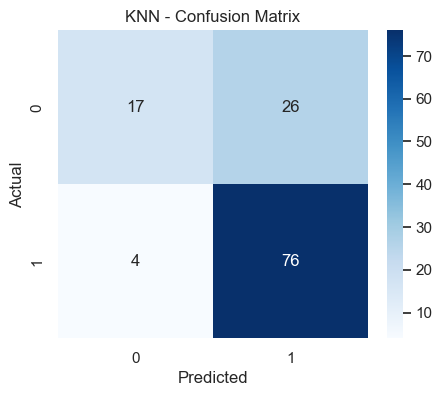


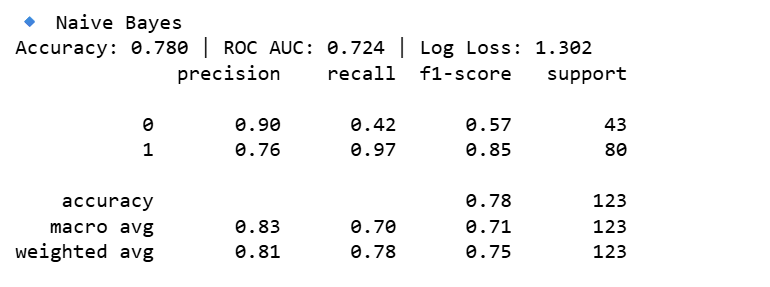


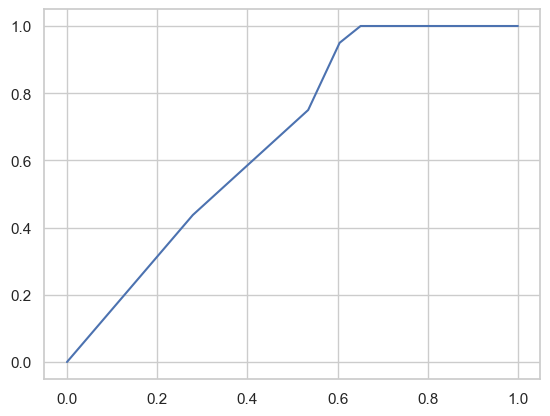


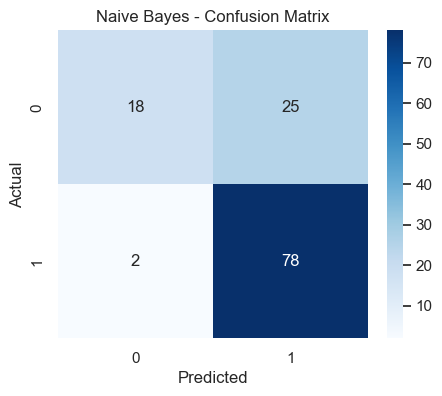


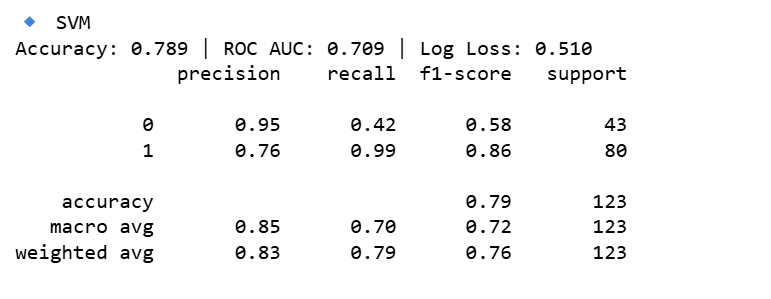


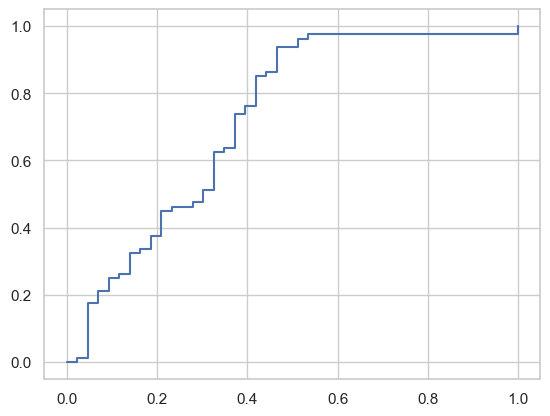


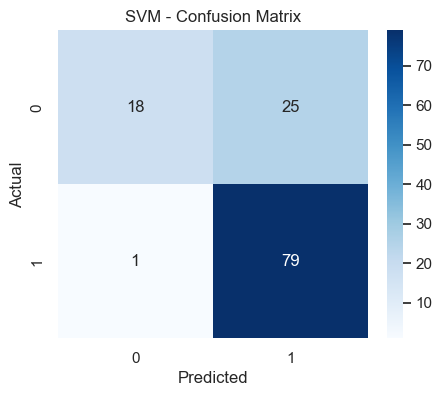


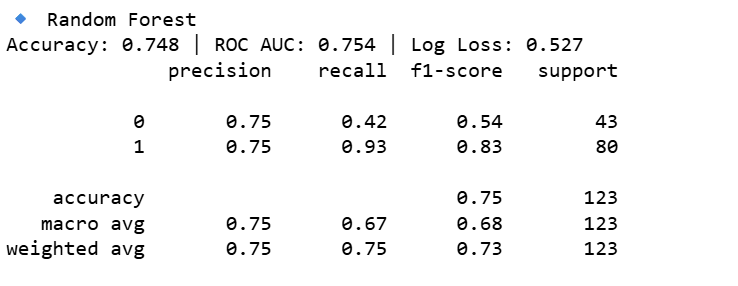


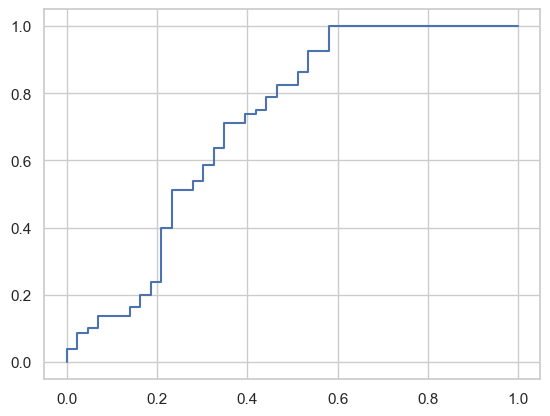


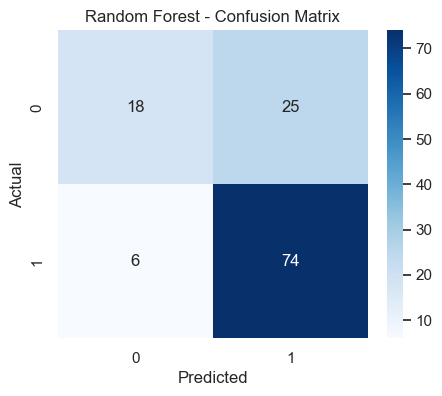


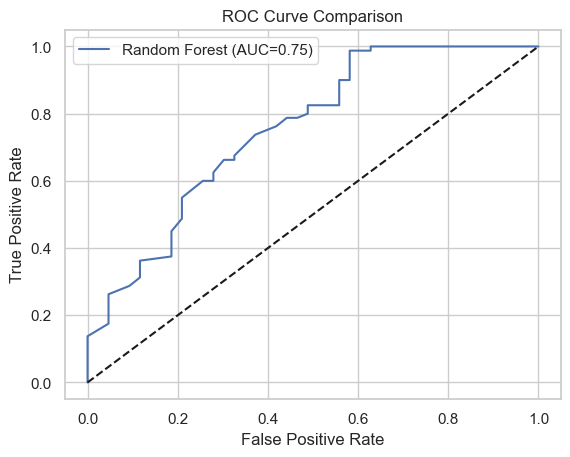


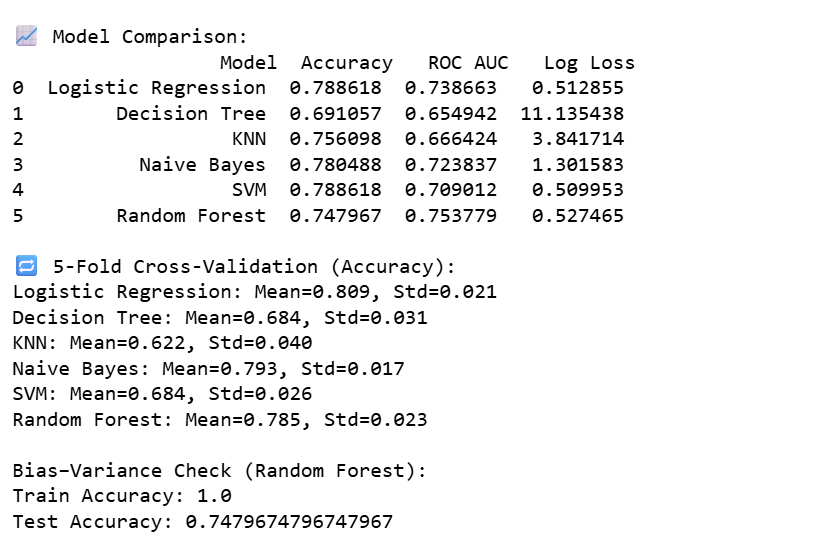


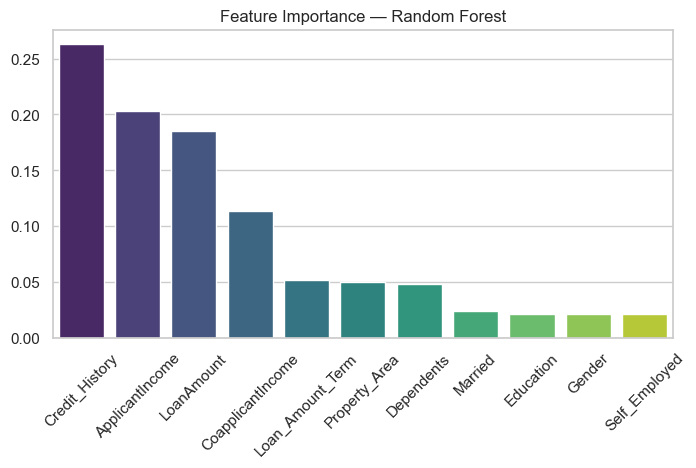


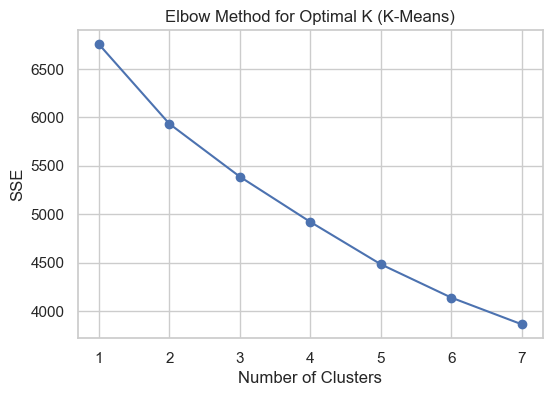


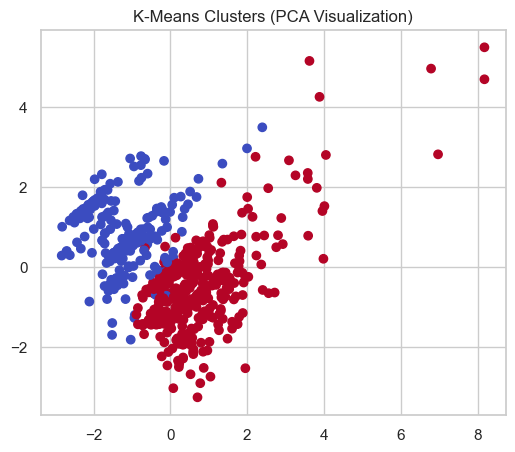


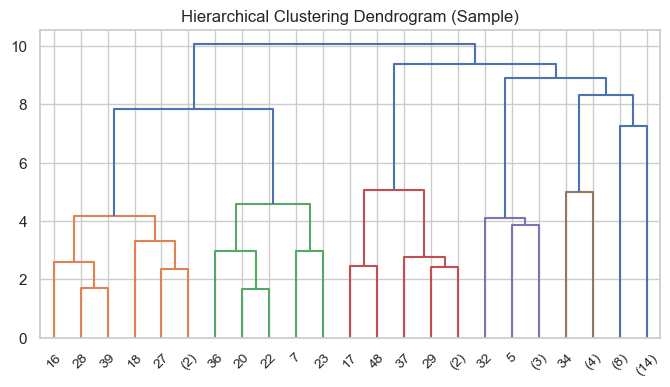


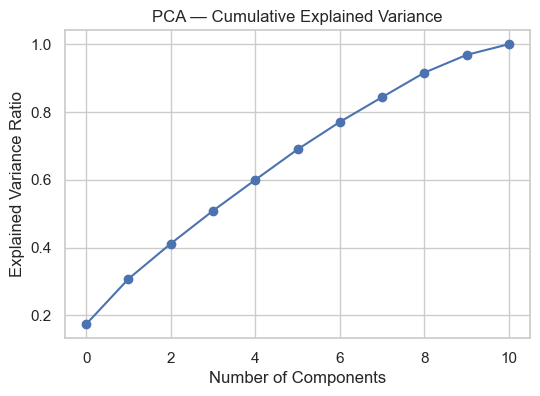


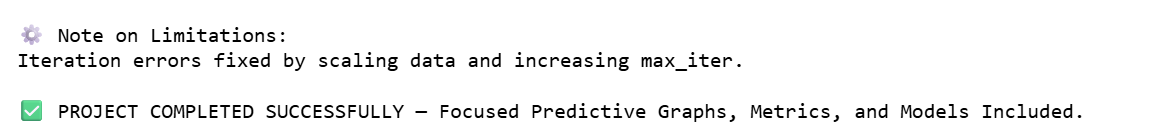












# LIBRARY AND TERMS EXPLAINATION

| Library / Module | Purpose | Explanation & Used in This Project |
| --- | --- | --- |
| pandas | Data handling and manipulation | Pandas is used to read, clean, and manipulate data in tabular form (DataFrames). In this project, it was used to load the CSV dataset, handle missing values, and preprocess data before feeding it into machine learning models. |
| numpy | Numerical operations | NumPy provides support for large, multidimensional arrays and mathematical operations. It was used for calculations like RMSE (np.sqrt) and array transformations during scaling and clustering. |
| matplotlib.pyplot | Visualization | Matplotlib is a plotting library used for basic graphs like scatter plots, line plots, and bar charts. In this project, it visualized Actual vs Predicted Loan Amounts, ROC curves, and PCA plots. |
| seaborn | Statistical visualization | Seaborn provides advanced visualization based on Matplotlib. It was used for creating correlation heatmaps, confusion matrices, and feature importance plots with better aesthetics. |
| warnings | Suppressing warnings | The warnings library is used to ignore unnecessary warning messages (like convergence warnings) for cleaner output in notebooks. |
| sklearn.model\_selection | Model training and validation tools | Contains functions like train\_test\_split, KFold, and cross\_val\_score. These were used to split the dataset into training and test sets, perform cross-validation, and evaluate model stability. |
| train\_test\_split | Splitting data | Used to split data into training and testing subsets (e.g., 80% train, 20% test) to check model generalization. |
| KFold, cross\_val\_score | Cross-validation | These functions divide data into multiple folds to measure model performance on different subsets — reducing bias in results. |
| sklearn.preprocessing | Data encoding and scaling | This module includes tools like LabelEncoder and StandardScaler, which are used to convert categorical variables into numbers and scale numeric features to a standard range. |
| LabelEncoder | Categorical encoding | Converts text categories (e.g., “Male”, “Female”) into numerical codes (0, 1). |
| StandardScaler | Feature scaling | Standardizes features by removing the mean and scaling to unit variance. It ensures all features contribute equally to model training, especially for distance-based algorithms like SVM and KNN. |
| sklearn.linear\_model | Linear and Logistic Regression models | Contains models for regression and classification. Used here for Linear Regression (predicting LoanAmount) and Logistic Regression (predicting Loan\_Status). |
| LinearRegression | Regression model | Predicts continuous numeric values — here, used to predict LoanAmount and measure error metrics. |
| LogisticRegression | Classification model | Predicts binary outcomes (Approved/Not Approved). Used with max\_iter=5000 to avoid convergence issues. |
| sklearn.tree | Decision Tree algorithms | Contains Decision Tree classifiers and regressors. Used to build tree-based classification models for loan approval prediction. |
| DecisionTreeClassifier | Classification model | Predicts class labels using a tree structure — splits data based on feature values for better interpretability. |
| sklearn.neighbors | K-Nearest Neighbors algorithm | Provides algorithms for KNN, a model that predicts outcomes based on similarity to nearest data points. |
| KNeighborsClassifier | Classification model | Used to predict loan approval by comparing each applicant with its closest data neighbors in the feature space. |
| sklearn.naive\_bayes | Probabilistic models | Implements Naive Bayes algorithms. Used GaussianNB to classify loan approvals based on conditional probability. |
| GaussianNB | Classification model | Based on Bayes’ theorem, assumes feature independence; efficient for small datasets and fast prediction. |
| sklearn.svm | Support Vector Machine models | Implements algorithms for SVM classification and regression. |
| SVC | Support Vector Classifier | Used for binary classification (loan approval). Works well for high-dimensional data when features are scaled. |
| sklearn.ensemble | Ensemble learning models | Contains ensemble algorithms like Random Forests, which combine multiple trees for higher accuracy. |
| RandomForestClassifier | Classification model | Builds multiple decision trees and averages their predictions. Used here to identify key features and improve accuracy. |
| sklearn.decomposition | Dimensionality reduction | Contains the PCA algorithm used for visualization and noise reduction. |
| PCA (Principal Component Analysis) | Dimensionality reduction | Reduces feature dimensions while retaining most of the variance — used to visualize clusters in 2D. |
| sklearn.cluster | Clustering algorithms | Contains clustering methods like K-Means for unsupervised learning. |
| KMeans | Clustering algorithm | Groups similar data points together. Used to find hidden patterns and visualize loan applicant clusters. |
| scipy.cluster | Hierarchical clustering | Provides tools for hierarchical clustering and dendrograms. |
| hierarchy, pdist | Clustering utilities | hierarchy helps draw dendrograms (tree-like visual of cluster merging); pdist computes pairwise distances between data points. |
| sklearn.metrics | Model evaluation metrics | Contains functions to measure model performance. Used extensively to evaluate regression and classification models. |
| mean\_absolute\_error (MAE) | Regression metric | Average of absolute differences between predicted and actual values. |
| mean\_squared\_error (MSE) | Regression metric | Measures average of squared errors — penalizes larger errors more. |
| r2\_score (R²) | Regression metric | Indicates how much variance in the target variable is explained by the model. |
| accuracy\_score | Classification metric | Measures overall correctness of classification predictions. |
| confusion\_matrix | Classification metric | Compares actual vs predicted outcomes to understand model performance. |
| classification\_report | Classification metric | Summarizes precision, recall, f1-score, and accuracy. |
| roc\_auc\_score | Classification metric | Measures how well a model distinguishes between classes. Higher AUC = better performance. |
| roc\_curve | Visualization metric | Plots True Positive Rate vs False Positive Rate for evaluating classification thresholds. |
| log\_loss | Classification metric | Measures model performance based on predicted probabilities rather than just labels. |
| sns.set() | Visualization style setting | Used to give graphs a clean “whitegrid” background for better readability. |
| warnings.filterwarnings("ignore") | Runtime control | Suppresses non-critical warnings during model training (like convergence or iteration warnings). |

# Conclusion

The Loan Prediction Project successfully demonstrates the application of machine learning techniques for predictive analytics in finance. By combining **data preprocessing, regression, classification, and unsupervised learning**, the project:

* Accurately predicts loan approval status using multiple classification models.
* Provides insights into the importance of features affecting loan decisions.
* Offers clear visualizations of model performance (confusion matrices, ROC curves, PCA scatter plots, feature importance).
* Highlights model limitations and solutions for convergence or iteration errors.

Overall, this project serves as a practical example of how predictive modelling can guide decision-making in financial institutions, improving both efficiency and reliability in loan approval processes.