Internship Project Report On…

“Loan Approval Prediction Analysis”  
  
 Submitted to…  
  
Prof.Suvarna Ranade  
Program Head  
M.Sc. Statistics

By…  
  
Mr. [Your Name Here]  
(PRN No.- [Your PRN Here])  
  
Date of Submission: [Date]

**Certificate**

This is to certify that the “Internship Report” submitted by [Your Full Name] during the academic year 2024-25 in partial fulfillment of the requirements for the award of the degree of Master Of Science in Statistics, MIT–WPU(Pune).  
  
Place: Pune Dr.Shubhalaxmi Joshi Prof. Suvarna Ranade  
Date: [Submission Date] Associate Dean Program Head  
 M.Sc. (Statistics)

**Acknowledgement**

First of all, I would like to thank Prof. Suvarna Ranade, my supervisor, for her invaluable guidance in preparation of this internship report. I would like to thank the Department of Mathematics and Statistics MIT-WPU and to my batchmates and friends for their constant support and valuable suggestions throughout the project.

**Executive Summary**

This report is based on an internship project that uses machine learning to automate loan approval prediction. Python programming and statistical methods were used to analyze applicant data and build predictive models that assist in identifying eligible loan applicants efficiently. The goal is to reduce processing time and financial risk for banks by leveraging data-driven solutions.

**Contents**

1. Introduction
2. Problem Statement
3. Data Dictionary
4. Data Preprocessing
5. EDA
6. Feature Engineering
7. Data Transformation
8. Logistic Regression
9. Decision Tree
10. Random Forest
11. Hyperparameter Tuning
12. Conclusion

**Introduction**

In today’s data-driven world, automating loan approval processes is crucial for improving efficiency and lowering risk in the banking sector. This project aims to build a machine learning model to predict whether a loan should be approved based on applicant details. We utilize the "Loan Approval Prediction Dataset" from Kaggle to conduct data analysis and model building and data preprocessing tasks. The dataset contains essential parameters which include gender, income, education, credit history and loan amount because these factors determine loan approval outcomes. The system seeks to help banking institutions make quick and objective decisions regarding loan approvals.

**Problem Statement**

Financial institutions currently use manual loan evaluations for their approval process but this approach consumes time and shows human distortions. An automated system needs development to accurately predict loan approvals because it will improve both efficiency and consistency.

The goal is to automate the loan eligibility process in real time using customer details like income, dependents, CIBIL score, and asset values. A binary classification model is built to identify customers eligible for loan approval.

**Data Dictionary**

Variable Description  
loan\_id Unique Loan ID  
no\_of\_dependents Number of dependents  
education Graduate / Not Graduate  
self\_employed Self-employed (Y/N)  
income\_annum Annual income of applicant  
loan\_amount Requested loan amount  
loan\_term Duration of the loan in years  
cibil\_score Credit score from 300–900  
residential\_assets\_value Value of residential assets  
commercial\_assets\_value Value of commercial assets  
luxury\_assets\_value Value of luxury assets  
bank\_asset\_value Amount in bank  
loan\_status Approved / Rejected

**Data Preprocessing**

Data preprocessing is a crucial step to ensure the dataset is suitable for machine learning models. The dataset was processed through three main stages: **Data Cleaning**, **Outlier Detection and Treatment**, and **Encoding**.

**1. Data Cleaning**

The dataset contains 4269 entries and 13 features. During the **data cleaning** process, we checked for missing values across all columns, and no missing values were found means that data is fully cleaned for further analysis.

**2. Outlier Detection and Treatment**

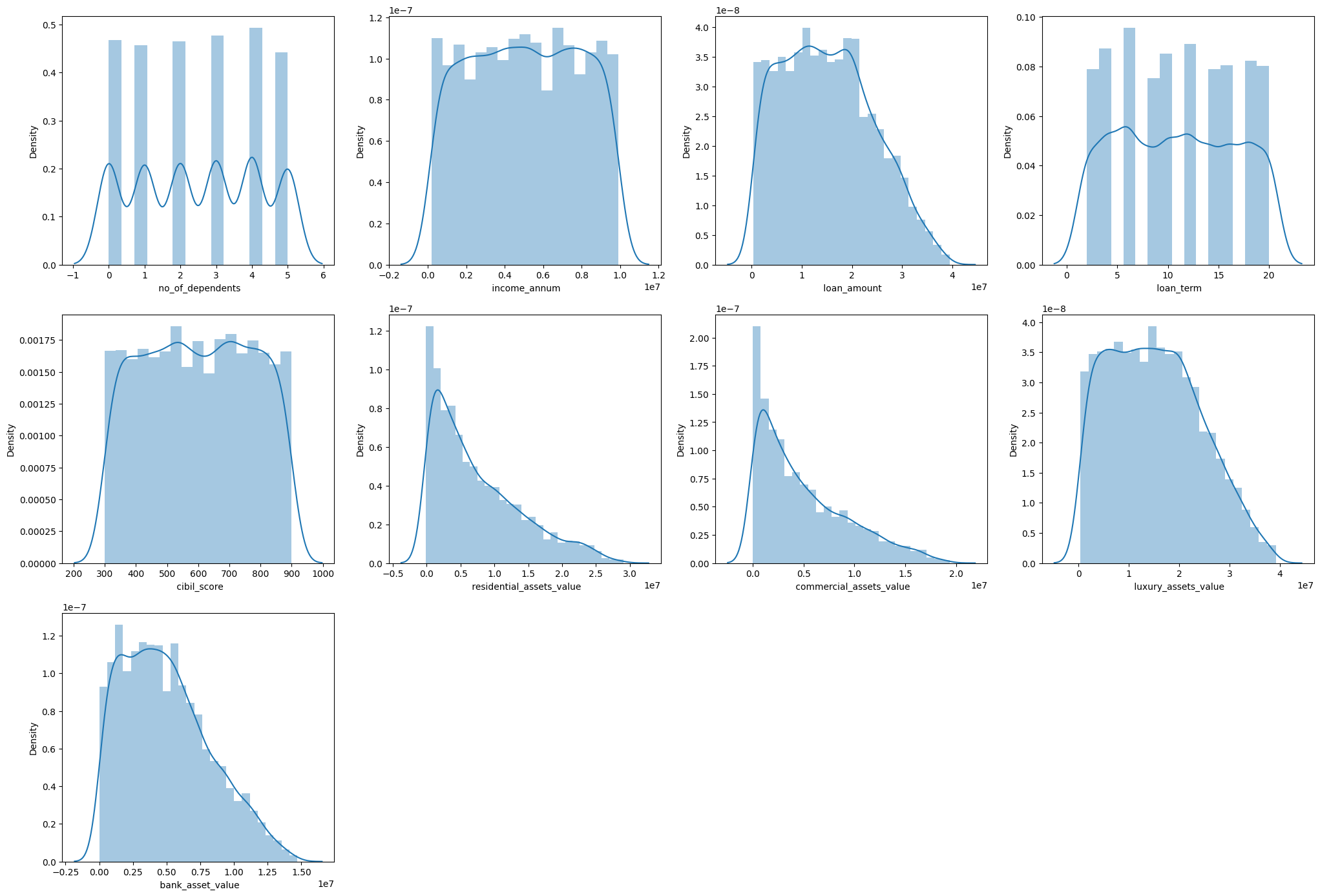
Outliers were detected using the Interquartile Range (IQR) method. Features such as **residential\_assets\_value**, **commercial\_assets\_value**, and **bank\_asset\_value** contained outliers. These outliers were capped using the 1st and 99th percentiles to reduce their impact on the model's performance, ensuring that extreme values did not unduly influence the predictions.

**EDA**

Exploratory Data Analysis (EDA) is an approach that is used to analyze the data and discover trends, patterns, or check assumptions in data with the help of statistical summaries and graphical representations.

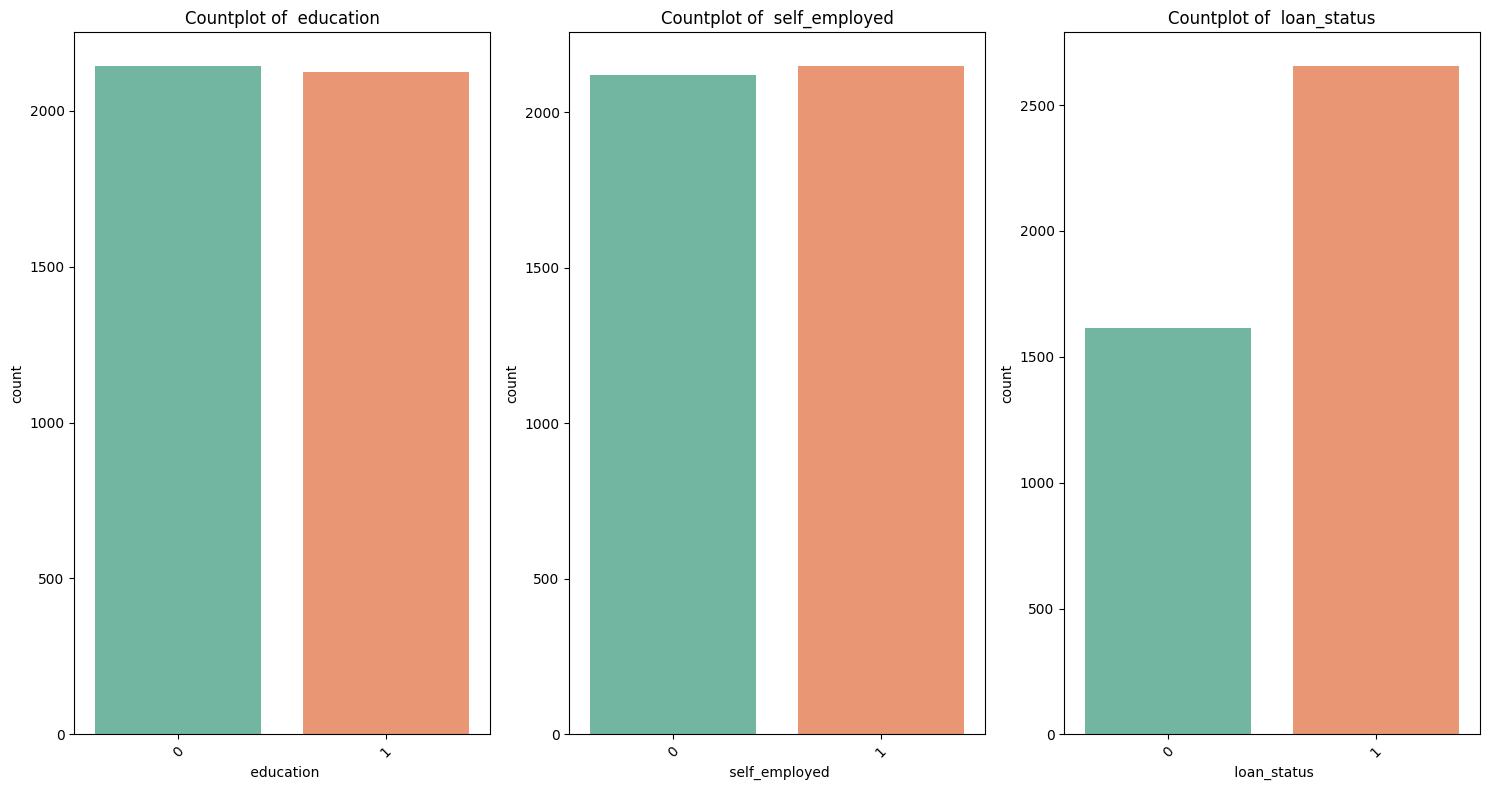
Univariate Analysis

1. **Distribution Plots (Distplots for Numerical Features)**

****

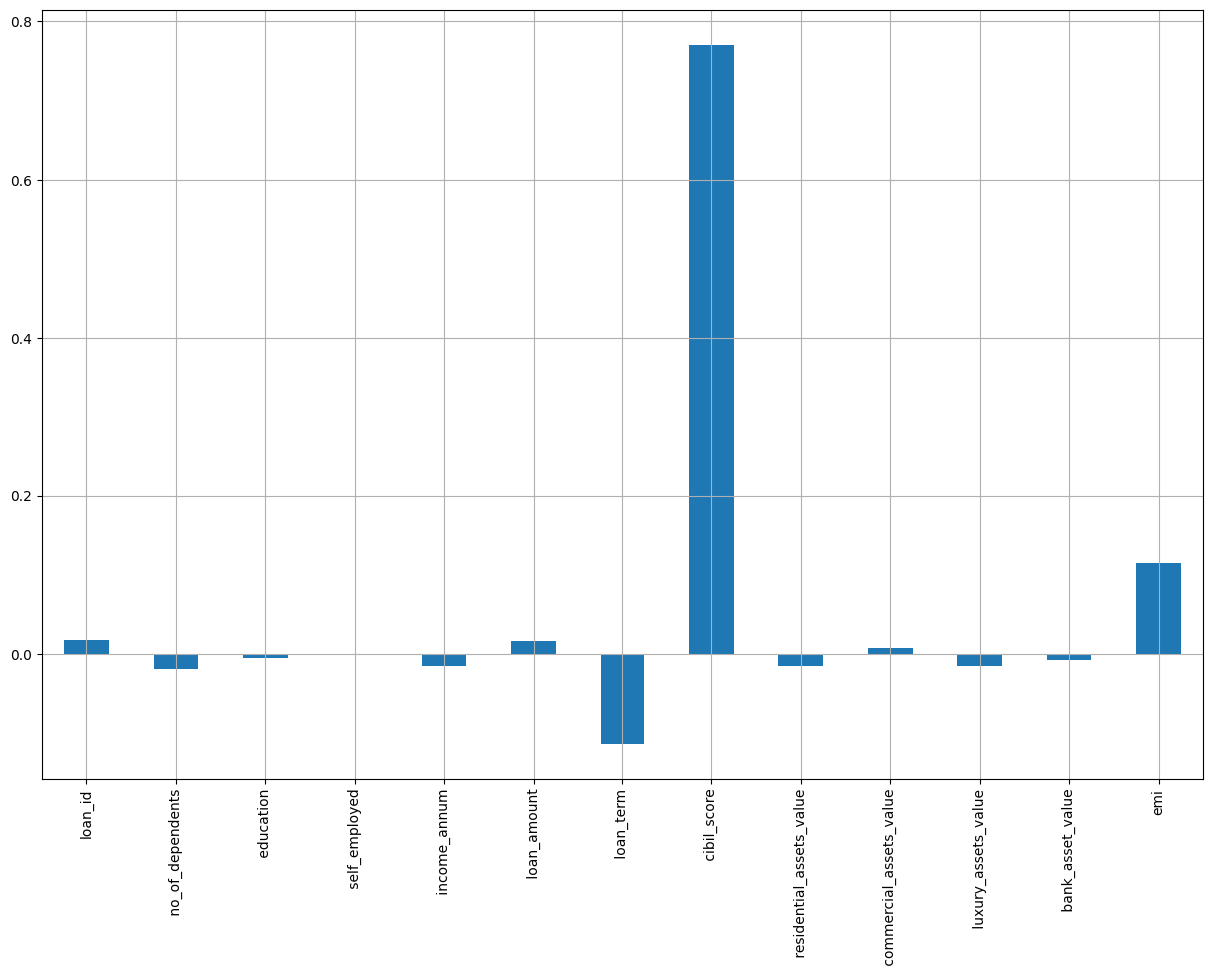
* These show how data like income, loan amount, and CIBIL score are spread.
* Most features are **right-skewed**, especially loan amount and income, indicating a majority of people fall into lower ranges with a few extreme high values.

1. **Count Plots (Categorical Features)**

****

* Breakdown of applicants by **education**, **employment type**, and **loan status**.
* Balanced distribution in education and employment.
* More loans are **approved** than rejected.

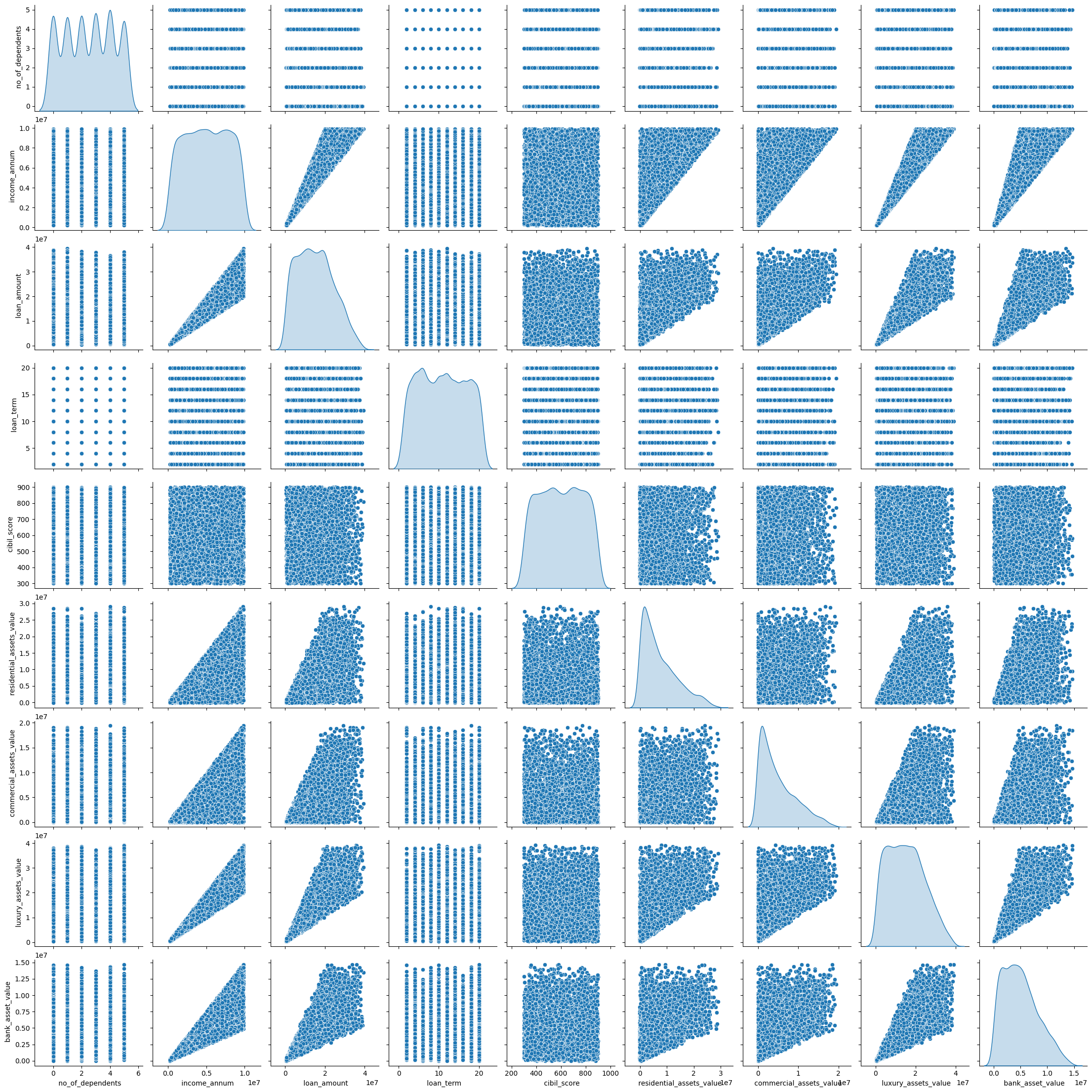
1. Correlation with Loan Status (Bar Plot)



* Quantifies how each feature correlates with loan approval.
* CIBIL score is the most impactful feature for loan approval.
* Other features have minimal influence individually.

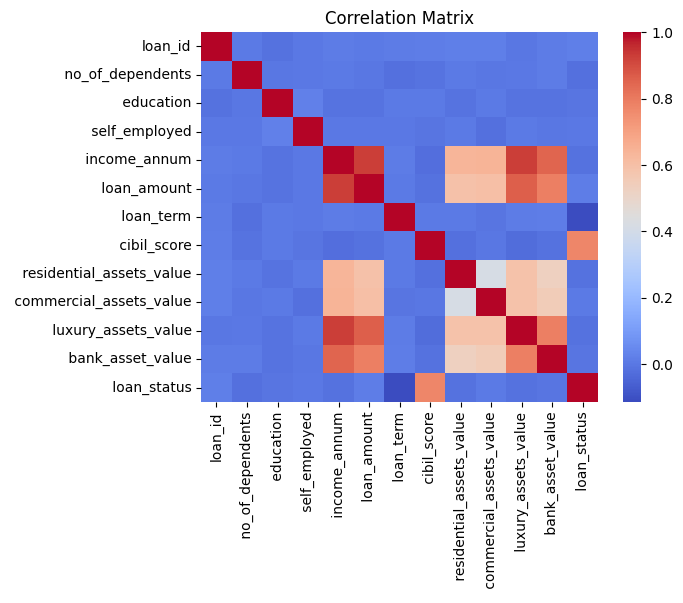
Bivariate Analysis

1. **Pairplot**

****

* Visualizes pairwise relationships between numeric features.
* Helps detect **correlation and clustering**. For instance, CIBIL score and loan amount may show distinct grouping between approved and rejected loans.

1. **Correlation Heatmap**

****

* Displays relationships between numerical features.
* Strong positive correlation between **CIBIL score** and **loan approval**; weak or no correlation among many asset-related features.

**Feature Engineering**

In feature engineering, EMI feature was created to show the monthly repayment amounts for every loan. The standard amortization formula calculated the EMI feature by combining loan amount and interest rate with the loan term. The new feature enables better prediction of loan approval by assessing how much applicants need to pay each month.

**Data Transformation**

1. **Yeo–Johnson Transformation:** This transformation was used to handle skewed data by stabilizing the variance and making the distribution more normal (bell-shaped).
2. **Standard Scaling:** After applying the Yeo–Johnson transformation, the features were standardized to have a mean of 0 and a standard deviation of 1. This ensures that all features are on the same scale, preventing any one feature from dominating due to its larger numerical values and improving the performance of machine learning algorithms.

**Encoding**

For categorical features such as **education**, **self\_employed**, and **loan\_status**, encoding was applied. **Label Encoding** was used to transform these categorical variables into numerical values. Specifically, the **education** and **self\_employed** features were encoded as binary variables, and the **loan\_status** variable was converted into binary values (1 for 'Approved' and 0 for 'Rejected') to make them compatible with machine learning algorithms.

Model Building

LOGISTIC REGRESSION.

This type of statistical model (also known as logit model) is often used for classification and predictive analytics. Logistic regression estimates the probability of an event occurring, such as voted or didn't vote, based on a given dataset of independent variables.

Training Data Assumptions for Logistic Regression.

1. The predicted outcome is strictly binary or dichotomous.

2. The factors, or the independent variables, that influence the outcome are independent of each other. In other words, there is little or no multicollinearity among the independent variables

3. The independent variables can be linearly related to log odds.

4. Fairly large sample sizes. The Accuracy of Logistic Regression Model: 91.33

**K-Fold Cross Validation**

In k-fold cross-validation, you split the input data into k subsets of data (also known as folds). You train an ML model on all but one (k-1) of the subsets, and then evaluate the model on the subset that was not used for training. This process is repeated k times, with a different subset reserved for evaluation (and excluded from training) each time.

Cross-validation is a technique for evaluating ML models by training several ML models on subsets of the available input data and evaluating them on the complementary subset of the data. Use cross-validation to detect overfitting, i.e., failing to generalize a pattern.

1 of K Fold 5 accuracy\_score 0.931

2 of K Fold 5 accuracy\_score 0.935

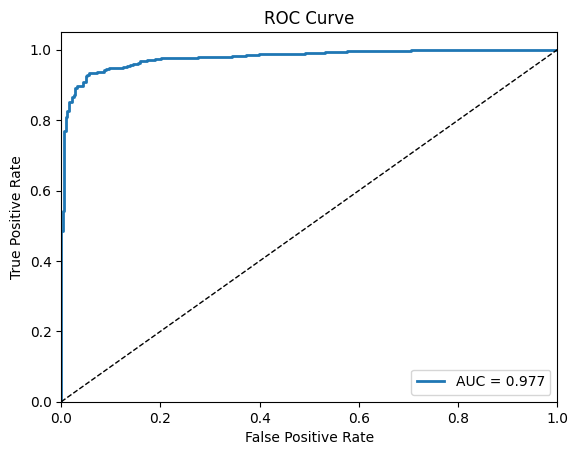
3 of Kfold 5 accuracy\_score 0.920

4 of Kfold 5 accuracy\_score 0.937

5 of Kfold 5 accuracy\_score 0.915

**Area under curve (AUC)**

**AUC provides an aggregate measure of performance across all possible classification threshold**

****

**Decision Tree**

The Decision Tree classifier achieved 98% accuracy but showed signs of overfitting. It’s an interpretable model but can be less stable with minor data changes.

**K-Fold Cross Validation**

1 of K folds 5 Accuracy Score of Random Forest is 0.986

2 of K folds 5 Accuracy Score of Random Forest is 0.974

3 of K folds 5 Accuracy Score of Random Forest is 0.972

4 of K folds 5 Accuracy Score of Random Forest is 0.980

5 of K folds 5 Accuracy Score of Random Forest is 0.977

**Random Forest**

Random forest is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression

**K-Fold Cross Validation**

1 of K folds 5 Accuracy Score of Random Forest is 0.988

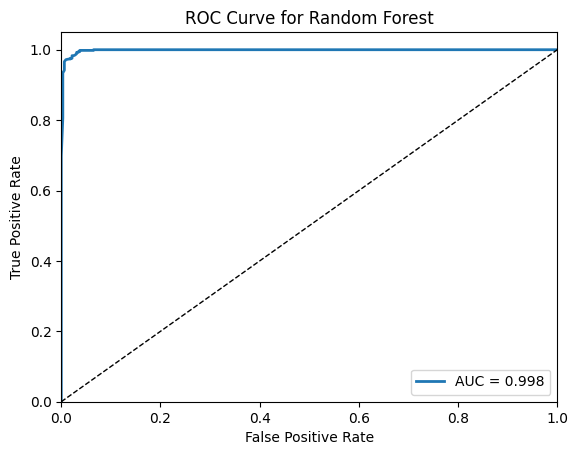
2 of K folds 5 Accuracy Score of Random Forest is 0.980

3 of K folds 5 Accuracy Score of Random Forest is 0.984

4 of K folds 5 Accuracy Score of Random Forest is 0.986

5 of K folds 5 Accuracy Score of Random Forest is 0.984

AUC



Classification Report

precision recall f1-score support

0 0.99 0.97 0.98 323

1 0.98 0.99 0.99 531

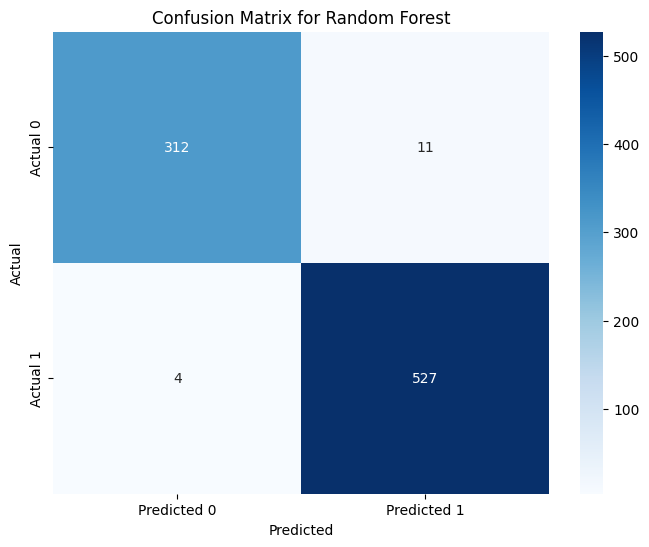
accuracy 0.98 854

macro avg 0.98 0.98 0.98 854

weighted avg 0.98 0.98 0.98 854

ROC-AUC score: 0.9983820468419303

Confusion Matrix



Hyperparameter Tuning

**Random Forest** proved to be the best, achieving an accuracy of **99.5%**. Hyperparameter tuning using **RandomizedSearchCV** optimized parameters, enhancing the model’s performance, especially in handling class imbalance and improving ROC-AUC.

**Conclusion**

* **CIBIL score** emerged as the most influential factor in determining loan approval.
* Correlation analysis showed that most variables had **low correlation** with loan approval, except for the **CIBIL score**.
* **SMOTE** was applied to handle class imbalance, improving the model’s ability to correctly predict rejected loans.
* Among all models tested, the **Random Forest classifier** gave the best performance with a **ROC-AUC of ~0.996**.
* **Logistic Regression** and **Decision Tree** also performed well, with ROC-AUC values above 0.97.
* The project successfully demonstrates how **machine learning and EDA** can be combined to build reliable loan approval systems.