**Loan Prediction System Report**

**1. Introduction**

**Problem Statement**

In this project, the goal is to predict whether a loan application will be approved or rejected. Financial institutions need an automated and reliable system to assess loan eligibility based on customer information.

**Objective**

Build and evaluate machine learning models (Linear Regression and Random Forest) to predict loan status accurately.

**2. Dataset Overview**

The dataset contains multiple features related to loan applicants and their financial profiles.

Key variables include:

|  |  |
| --- | --- |
| **Feature** | **Description** |
| **no\_of\_dependents** | Number of dependents on the applicant |
| **Education** | Applicant's education level (Graduate or Not Graduate) |
| **self\_employed** | Whether the applicant is self-employed (Yes/No) |
| **income\_annum** | Annual income of the applicant |
| **loan\_amount** | Loan amount requested |
| **loan\_term** | Loan term in months |
| **cibil\_score** | Applicant's CIBIL credit score |
| **residential\_assets\_value** | Value of residential assets |
| **commercial\_assets\_value** | Value of commercial assets |
| **luxury\_assets\_value** | Value of luxury assets |
| **bank\_asset\_value** | Bank asset value |
| **loan\_status** | Target variable: Approved (1) or Rejected (0) |

**Data Preprocessing**

* Combined all asset values into a single feature: Assets.
* Removed unnecessary columns such as loan\_id.
* Encoded categorical variables (education, self\_employed, and loan\_status).
* Handled missing values and removed outliers using the **IQR method**.

**3. Exploratory Data Analysis (EDA)**

**Loan Status Distribution**

The loan status is visualized to understand class distribution.

**Outlier Analysis**

Boxplots were used to identify outliers in numerical features. Outliers were removed to improve model performance.

**Correlation Heatmap**

The heatmap shows correlations between features:

* income\_annum, loan\_amount, and cibil\_score have notable correlations with loan\_status.

**4. Methodology**

**4.1 Feature Engineering**

* Created a new feature Assets by summing asset values.
* Scaled numerical features using **StandardScaler** for consistency.

**4.2 Train-Test Split**

The dataset was split into **80% training data** and **20% testing data**.

**4.3 Machine Learning Models**

Two machine learning models were used:

1. **Linear Regression**: A simple model to predict loan approval probability.
2. **Random Forest**: An ensemble model that handles non-linear relationships effectively.

**4.4 Hyperparameter Tuning**

GridSearchCV was used to optimize the Random Forest model parameters:

* n\_estimators: Number of trees in the forest.
* max\_depth: Depth of the trees.

**5. Results and Model Evaluation**

**Evaluation Metrics**

The models were evaluated using the following metrics:

* **Accuracy**
* **Precision**
* **Recall**
* **F1-Score**

**Model Comparison**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-Score** |
| Linear Regression | 89.3% | 87.1% | 90.0% | 88.5% |
| Random Forest | **98.0%** | **97.5%** | **98.8%** | **98.1%** |

**Key Insights**

* The **Random Forest model** significantly outperformed Linear Regression in all metrics.
* Random Forest's ability to handle complex data and outliers contributed to its superior performance.

**6. Deployment**

The best-performing Random Forest model was deployed using **Streamlit**, creating an interactive web application.

**App Features**

1. Users can input loan applicant details such as income, loan amount, and CIBIL score.
2. The app predicts loan approval status as "Approved" or "Rejected."
3. Displays the probability of loan approval for user confidence.

**7. Conclusion**

**Summary of Findings**

* Outlier removal and feature scaling improved model performance.
* **Random Forest** achieved the highest accuracy of **98%**.
* The loan prediction system is robust, user-friendly, and deployable.

**Future Enhancements**

* Add more features like applicant's debt-to-income ratio.
* Incorporate other ensemble models like Gradient Boosting for comparison.

**8. References**

* Scikit-Learn Documentation
* Streamlit Documentation

**Attachments**

* Streamlit Web Application : https://rashidametascifortechnology-r2avhtghwas9swfwgbj8nr.streamlit.app/
* Code Files : [Rashida\_Meta\_Scifor\_Technology/Major\_Project/loan\_approval\_prediction.py at main · Rashida-kk/Rashida\_Meta\_Scifor\_Technology](https://github.com/Rashida-kk/Rashida_Meta_Scifor_Technology/blob/main/Major_Project/loan_approval_prediction.py)