1. Introduction

In this project I build a machine learning model to predict whether a loan application will be approved or rejected based on applicant and asset information.

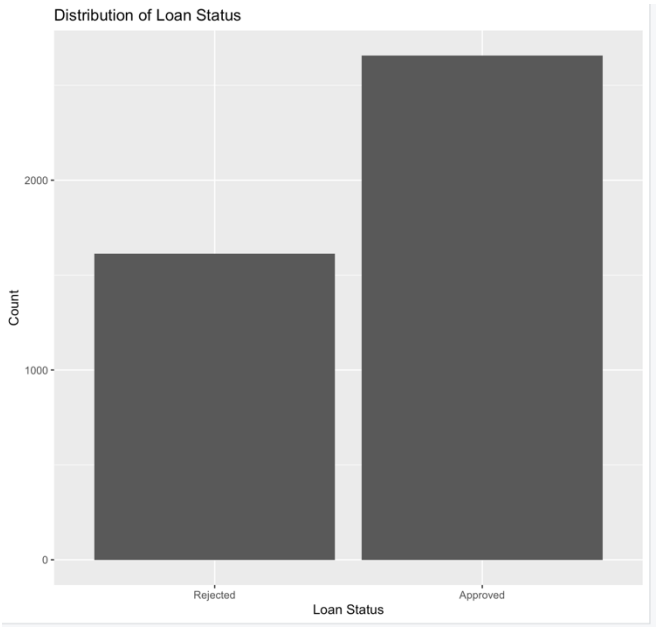
Automating this decision can help financial institutions speed up approvals, reduce manual effort and maintain more consistent decision-making. The goal of the project is to compare a simple, interpretable baseline model (logistic regression) with a more powerful tree-based model (random forest), and evaluate how well they can predict loan approval.

2. Data Description

The dataset contains **4,269 loan applications** with **12 predictor variables** and one target variable:

- **Target:** loan_status – whether the loan was **Approved** or **Rejected**.
- **Applicant features:** number of dependents, education level, employment type, annual income.
- **Loan details:** loan amount, loan term, CIBIL credit score.
- **Assets:** residential, commercial, luxury and bank asset values.

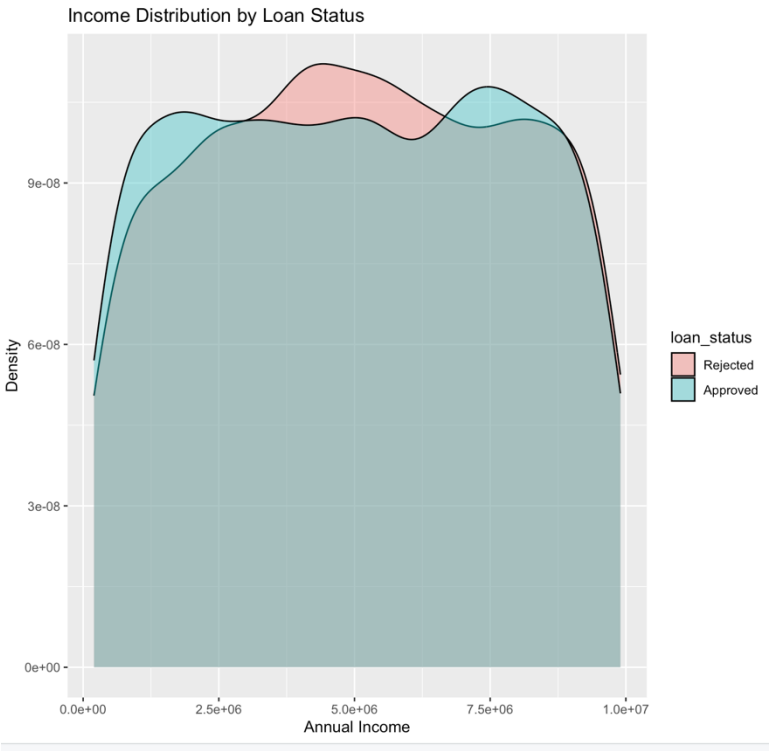
There are slightly more **Approved** loans than Rejected, as shown in the class distribution plot. There are no missing values in this dataset, so no imputation was required. Categorical variables (education, self_employed, loan_status) were converted to factors, and loan_id was removed as an ID column



3. Exploratory Data Analysis (EDA)

I first explored how some predictors relate to loan approval. For example, the density plot of income_annum by loan_status shows that approved applicants tend to have slightly higher annual income, although the distributions overlap.

Summary statistics also show that applicants with higher **CIBIL scores** and higher asset values are more likely to be approved, which matches the intuition of how credit scoring works.



4. Modelling Approach

I split the data into **70% training** and **30% testing** using stratified sampling on loan_status to preserve the class balance in both sets.

I trained and compared two models:

1. **Logistic Regression**
    - A linear, interpretable classification model.
    - Predicts the log-odds of approval as a function of all predictors.

- Useful as a baseline to understand which features are associated with approval.

2. **Random Forest**
    - An ensemble of decision trees trained on bootstrapped samples.
    - Handles non-linear relationships and interactions automatically.
    - Tends to give higher accuracy in exchange for less interpretability.

All models were implemented in **R** using tidyverse, caret, randomForest and pROC.

5. Evaluation Metrics

Model performance was evaluated on the **test set** using:

- **Confusion matrix**
- **Accuracy**
- **F1-score**
- **ROC curve and AUC (Area Under the Curve)**

I treated **"Approved"** as the positive class because in this context we are most interested in correctly identifying approved applications.

6. Results

You can summarise the numbers you got:

The table below compares the two models on the test data:

Model	**Accuracy**	**F1-score**
Logistic Regression	0.91	0.93
Random Forest	0.98	0.98

Logistic Regression

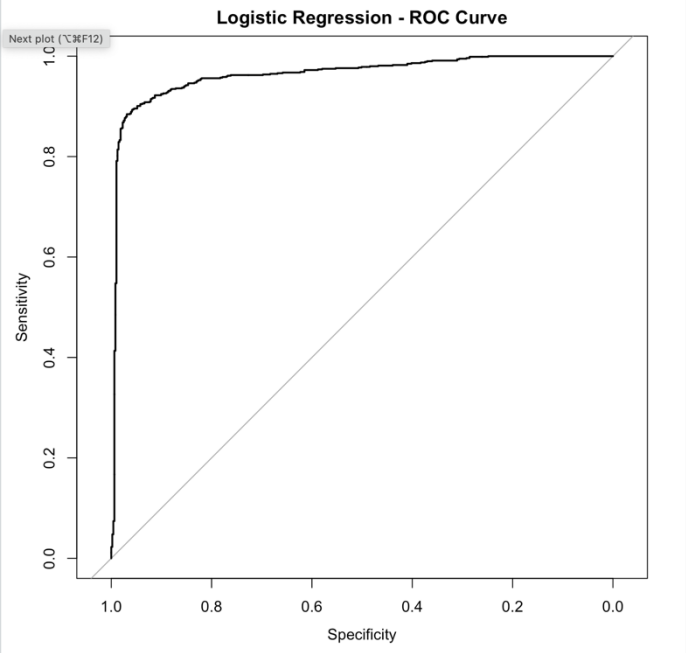
- Achieved an accuracy of around **91%** with an AUC of **0.96**, indicating very good discrimination between approved and rejected loans.

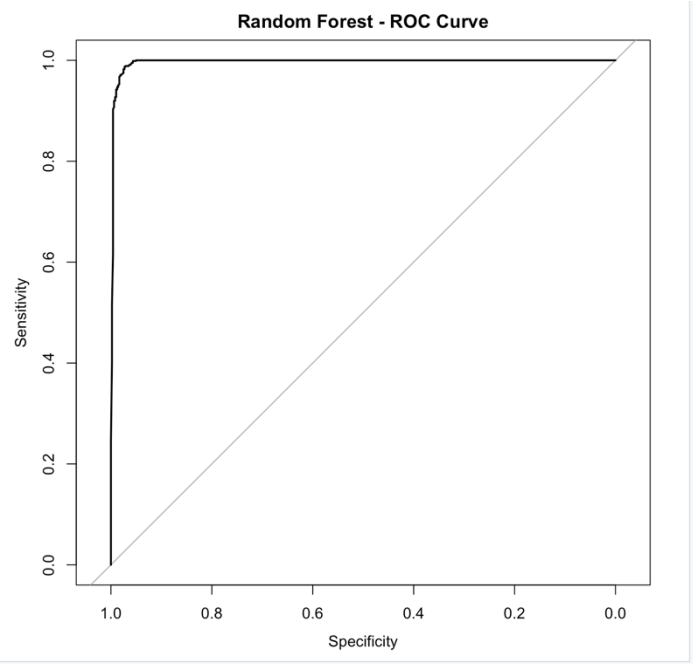
- The coefficients suggest that higher **CIBIL scores** increase the probability of approval, while longer **loan terms** and higher income have different effects depending on the combination of variables.

Random Forest

- Achieved a much higher accuracy of around **98%** and an AUC of **0.996**, indicating near-perfect separation on this dataset.
- The confusion matrix shows both sensitivity and specificity above **96%**, meaning the model rarely misclassifies either approved or rejected applications.

The ROC curves for both models lie well above the diagonal line, with the random forest curve almost touching the top-left corner (perfect classifier).





7. Feature Importance

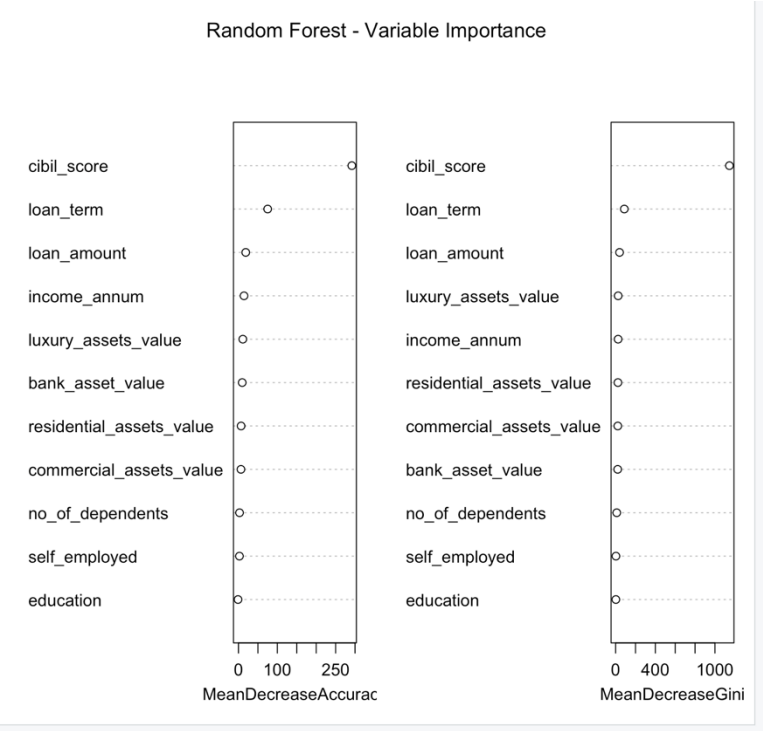
The variable importance plot from the random forest highlights which features contribute most to the predictions.

The most important predictors are:

- **CIBIL score**
- **Loan term**

- **Loan amount**
- **Annual income and asset values**

This makes sense in a credit-risk context: applicants with higher credit scores, reasonable loan amounts, and stronger asset positions are more likely to be approved.



8. Discussion

Both models perform extremely well on this dataset, with random forest clearly outperforming logistic regression on accuracy, F1 and AUC.

Logistic regression remains valuable because it is more interpretable – we can examine the coefficients to understand how individual variables affect the odds of approval. Random forest performs better but behaves more like a black box, which may be an issue in highly regulated financial environments.

Another limitation is that the dataset appears synthetic and may not reflect real-world biases, noise or changing economic conditions. In practice we would also need to check for overfitting, class imbalance over time, and fairness across different groups.

9. Conclusion and Future Work

In summary, I built an end-to-end loan approval prediction pipeline in R, from data loading and cleaning through to modelling, evaluation and visualisation. The best model (random forest) achieved around **98% accuracy** and an AUC of **0.996** on unseen test data.

In future work I would:

- Perform hyperparameter tuning for the random forest (e.g. using caret::train).
- Try additional models such as gradient boosting (XGBoost, LightGBM).
- Investigate calibration of predicted probabilities and cost-sensitive learning, since approving a bad loan is more expensive than rejecting a good one.
- Deploy the model as a simple API or Shiny app to demonstrate how it could be used in a real system.