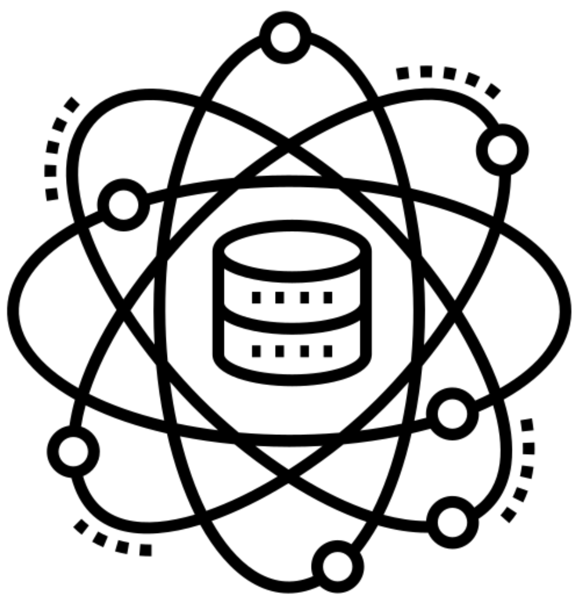
College of Applied Computer Sciences

King Saud University

Introduction to data science

Project ISY351

Using Data Science to Classify Loan-Eligible Customers



Group 7

Student:

Ibrahim alatyan 442102290

Osama alamri 442170142

Table of Contents:

Introduction:

About the Dataset

The **"Loan Prediction"** dataset is from Kaggle. It's a **classification problem**, where we try to predict if a loan will be approved (1) or not (0). The dataset includes information like the applicant's Gender, Marital Status, Income, Loan Amount, Credit History, etc.

Why This Dataset?

This dataset is good for learning because it's based on the **insurance industry**, which uses a lot of data science. It helps us understand:

- How to handle missing or incorrect data.
- Which details are important for making loan decisions.
- How to build models to predict loan eligibility.

The Problem

The goal is to **automate the loan approval process**. By looking at the applicant's details, we predict if they'll be approved for a loan. This can help companies save time and make quicker, more accurate decisions. This dataset is a small example of what real businesses deal with, making it a great way to practice.

Data Exploration:

Descriptive Statistics

To better understand the dataset, we begin by examining the summary statistics of the numerical columns. The descriptive statistics provide important insights, such as the mean, median, standard deviation, and range for each variable.

summary(org_data)

for example:

the mean for **<u>Loan Amount</u>** is 146.4 K

<u>Loan Amount Term</u> min is 12 month and max is 480 months

The median for **<u>Applicant Income</u>** is 3812

Various visualizations were created to explore the data distribution and relationships between variables.

Histograms were plotted for continuous variables, such as LoanAmount, to visualize the distribution.

Bar plots were used to visualize categorical variables like Married, Gender, and Loan_Status.

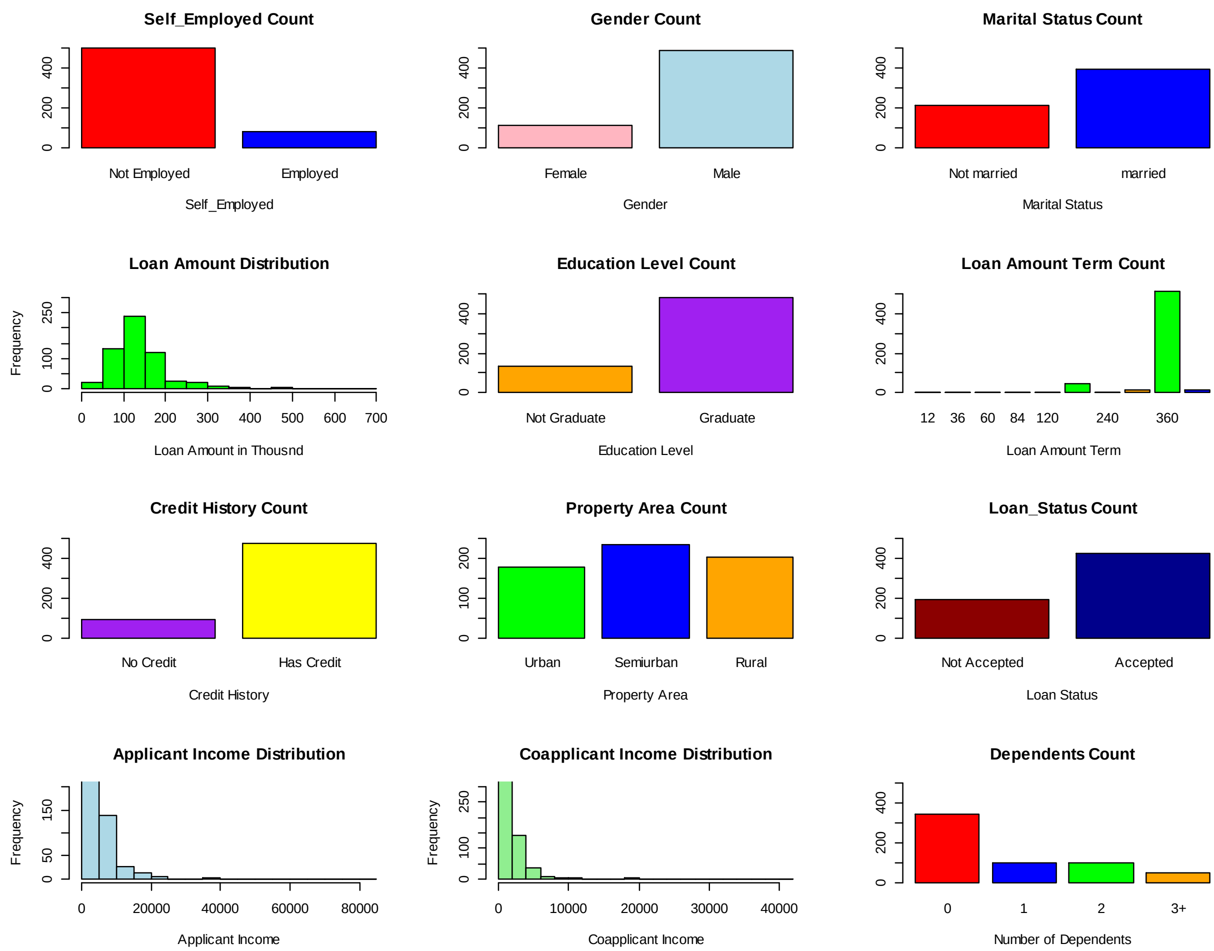
```
> summary(org_data) #summary of each column (mean, min ,max , etc..)
   Loan_ID             Gender            Married            Dependents          Education         Self_Employed      ApplicantIncome CoapplicantIncome   LoanAmount    Loan_Amount_Term
 Length:614         Length:614         Length:614         Length:614         Length:614         Length:614         Min.   :  150   Min.   :    0     Min.   :  9.0   Min.   : 12
 Class :character   Class :character   Class :character   Class :character   Class :character   Class :character   1st Qu.: 2878   1st Qu.:    0     1st Qu.:100.0   1st Qu.:360
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :character   Median : 3812   Median : 1188     Median :128.0   Median :360
                                                                                                                 Mean   : 5403   Mean   : 1621     Mean   :146.4   Mean   :342
                                                                                                                 3rd Qu.: 5795   3rd Qu.: 2297     3rd Qu.:168.0   3rd Qu.:360
                                                                                                                 Max.   :81000   Max.   :41667     Max.   :700.0   Max.   :480
                                                                                                                                                   NA's   :22      NA's   :14
 Credit_History   Property_Area        Loan_Status
 Min.   :0.0000   Length:614         Length:614
 1st Qu.:1.0000   Class :character   Class :character
 Median :1.0000   Mode  :character   Mode  :character
 Mean   :0.8422
 3rd Qu.:1.0000
 Max.   :1.0000
 NA's   :50
```

head(org_data)

```
> head(org_data) #first 6
   Loan_ID Gender Married Dependents     Education Self_Employed ApplicantIncome CoapplicantIncome LoanAmount Loan_Amount_Term Credit_History Property_Area Loan_Status
1 LP001002   Male      No          0      Graduate            No            5849                 0         NA              360              1         Urban           Y
2 LP001003   Male     Yes          1      Graduate            No            4583              1508        128              360              1         Rural           N
3 LP001005   Male     Yes          0      Graduate           Yes            3000                 0         66              360              1         Urban           Y
4 LP001006   Male     Yes          0  Not Graduate            No            2583              2358        120              360              1         Urban           Y
5 LP001008   Male      No          0      Graduate            No            6000                 0        141              360              1         Urban           Y
6 LP001011   Male     Yes          2      Graduate           Yes            5417              4196        267              360              1         Urban           Y
```

Visualization:

Before data cleaning:



Data Preprocessing:

Handling Missing Values

Missing values were identified in the dataset, particularly in numerical variables like ApplicantIncome, CoapplicantIncome, Loan_Amount_Term and LoanAmount. These missing values were imputed with the mean of the respective columns.

Handling Duplicates

Duplicates were checked and removed from the dataset to ensure data integrity

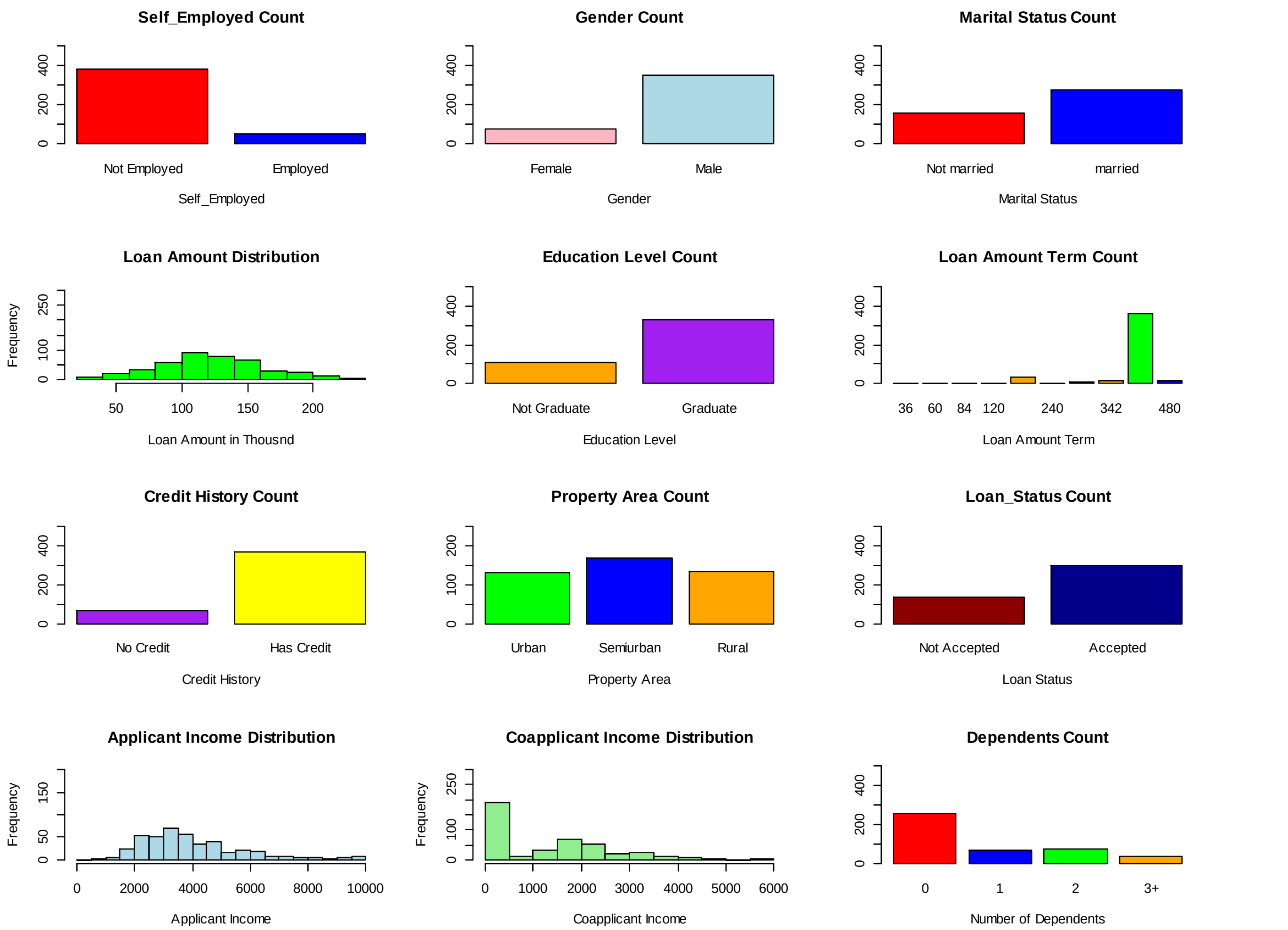
Encoding Categorical Variables

Categorical variables such as Gender, Married, Education, Self_Employed, Dependents and Loan_Status were encoded as numeric variables to make them suitable for machine learning models.

Outliers

To improve the accuracy of our modeling process, we identified and removed outliers from the ApplicantIncome, CoapplicantIncome, and LoanAmount variables using the Interquartile Range (IQR) method.

Data visualization after cleaning:



Modeling:

Machine Learning Algorithms

We applied two machine learning algorithms to predict the loan approval status:

1. Logistic Regression
2. Decision Tree

We chose **Logistic Regression** and **Decision Tree** for this project due to their suitability for binary classification tasks and their interpretability.

- **Logistic Regression** was selected for its simplicity, efficiency, and ability to predict binary outcomes, providing clear insights into the relationship between predictor variables and the target.
- **Decision Tree** was chosen for its ability to handle non-linear relationships and its interpretability, offering a visual representation of how decisions are made based on feature values.

Both models were ideal for analyzing the loan approval dataset, balancing effectiveness with ease of understanding.

Code:

Here are some line of code we use to create a train, test set and model:

```
library(caret)

set.seed(100)  # For reproducibility

# Split data into training and testing (80% train, 20% test)
trainIndex <- createDataPartition(data$Loan_Status, p = 0.8,
list = FALSE)
trainData <- data[trainIndex, ] #train 80%
testData <- data[-trainIndex, ] #test 20%

# Remove the Loan_ID column from both training and test data
trainData <- trainData[, -which(names(trainData) == "Loan_ID")]
testData <- testData[, -which(names(testData) == "Loan_ID")]
```

```r
#model 1 - Logistic Regression
# training model
model_logistic <- glm(Loan_Status ~ ., data = trainData, family
= "binomial")

# Summary of the model
summary(model_logistic)

# Make predictions on test set
pred_logistic <- predict(model_logistic, newdata = testData,
type ="response")

# Convert probabilities to binary predictions
pred_logistic_class <- ifelse(pred_logistic > 0.5, 1, 0)


#model 2 - desicion tree
# Load library for decision trees
library(rpart)
library(rpart.plot) #Visualize decision tree library

# Train decision tree
model_tree <- rpart(Loan_Status ~ ., data = trainData, method =
"class")

# Visualize decision tree
rpart.plot(model_tree)

# Make predictions on test set
pred_tree <- predict(model_tree, newdata = testData, type =
"class")
```

Results and Evaluation:

The performance of both models was evaluated using confusion matrices and accuracy scores.

Confusion Matrix

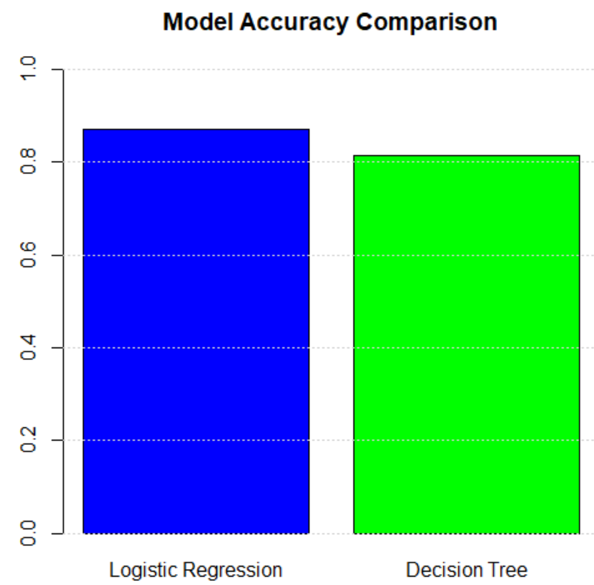
The confusion matrix for both models was generated, and accuracy was calculated for:

Logistic Regression

```
               Accuracy : 0.8721
                 95% CI : (0.7827, 0.9344)
    No Information Rate : 0.686
    P-Value [Acc > NIR] : 5.462e-05

                  Kappa : 0.6736

 Mcnemar's Test P-Value : 0.01586

            Sensitivity : 0.6296
            Specificity : 0.9831
         Pos Pred Value : 0.9444
         Neg Pred Value : 0.8529
             Prevalence : 0.3140
         Detection Rate : 0.1977
   Detection Prevalence : 0.2093
      Balanced Accuracy : 0.8063

       'Positive' Class : 0
```

Decision Tree

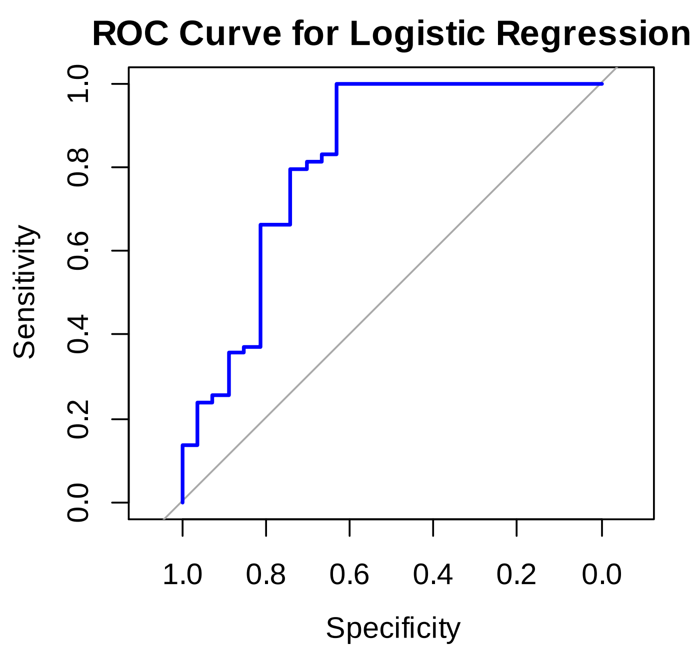
```
               Accuracy : 0.814
                 95% CI : (0.7155, 0.8898)
    No Information Rate : 0.686
    P-Value [Acc > NIR] : 0.00563

                  Kappa : 0.55

 Mcnemar's Test P-Value : 0.45325

            Sensitivity : 0.6296
            Specificity : 0.8983
         Pos Pred Value : 0.7391
         Neg Pred Value : 0.8413
             Prevalence : 0.3140
         Detection Rate : 0.1977
   Detection Prevalence : 0.2674
      Balanced Accuracy : 0.7640

       'Positive' Class : 0
```



Logistic Regression Accuracy: **0.872093** **Decision Tree Accuracy:** **0.8139535**

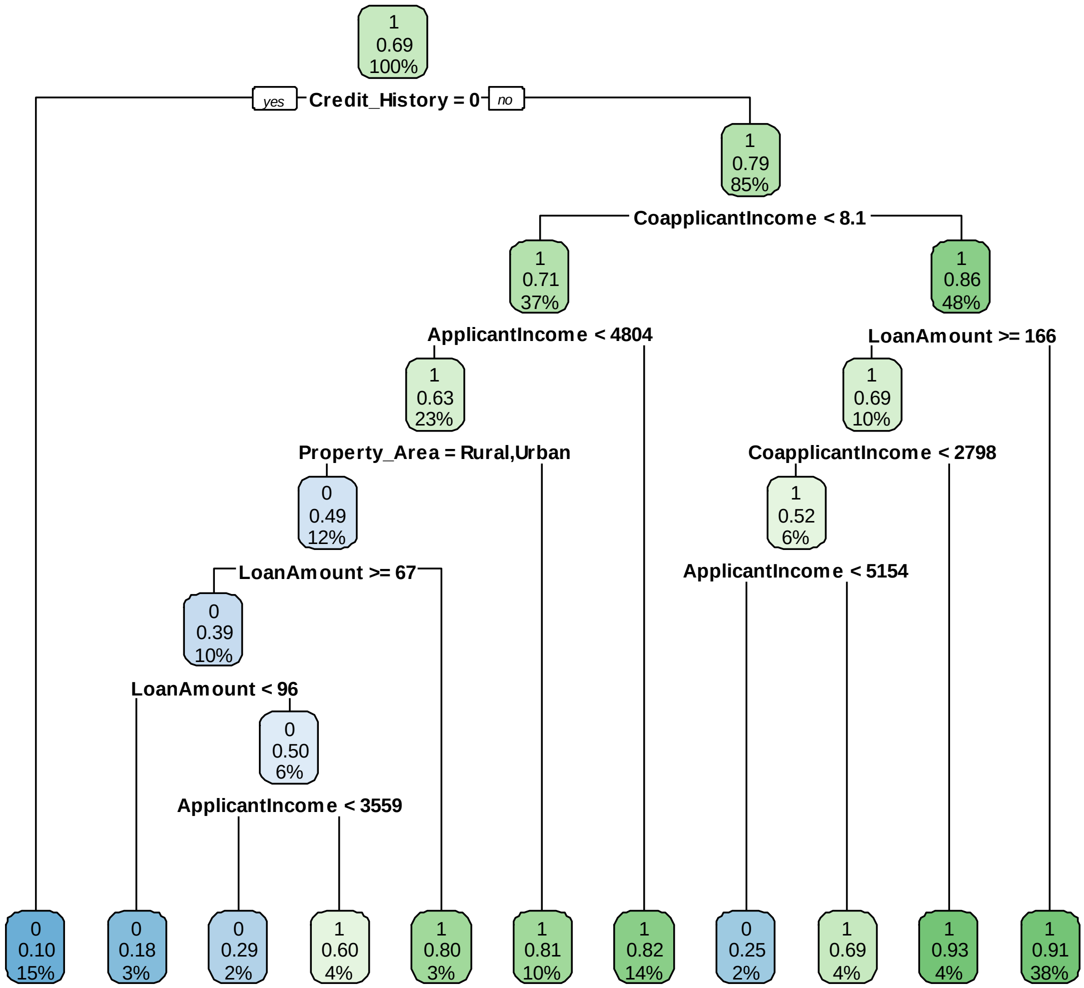
ROC

ROC curve was plotted for the logistic regression model to evaluate its classification performance. The AUC value was calculated to quantify the model's performance.



AUC for Logistic Regression: **0.819209**

Decision Tree



1 -> accepted **0-> not accepted**

Interpretation of Results

1. **Logistic Regression:**
   - Performed well with high accuracy and a good AUC-ROC score, showing it effectively distinguishes between approved and rejected loans.
   - It's simple and robust but limited in capturing complex patterns.
2. **Decision Tree:**
   - Slightly lower accuracy but highly interpretable, highlighting key factors like Credit History and Applicant Income.
   - Prone to overfitting, reducing generalization on unseen data.
3. **Comparison:**
   - Logistic Regression is better for accuracy and reliability, while the Decision Tree provides deeper insights into feature importance.
   - Both models agree that Credit History is the most critical factor for loan approval.

Additional Efforts:

Try model decision with new customer data

try model decision by give him new customer data to get loan accepted or not accepted:

New customer data:

```
custom_input <- data.frame(
  Gender = 1,
  Married = 1,
  Dependents = 2,
  Education = 1,
  Self_Employed = 0,
  ApplicantIncome = 3558,
  CoapplicantIncome = 0,
  LoanAmount = 95,
  Loan_Amount_Term = 360,
  Credit_History = 0,
  Property_Area = "Urban"
)


# Logistic Regression Prediction
custom_pred_logistic <- predict(model_logistic, newdata =
custom_input, type = "response")
custom_pred_logistic_class <- ifelse(custom_pred_logistic > 0.5,
"YES", "NO")
cat("Logistic Regression Prediction (Probability):",
custom_pred_logistic, "\n")
cat("Logistic Regression Prediction (Decision):",
custom_pred_logistic_class, "\n")

# Decision Tree Prediction
custom_pred_tree <- predict(model_tree, newdata = custom_input,
type = "class")
custom_pred_tree_decision <- ifelse(custom_pred_tree == 1,
"YES", "NO")
cat("Decision Tree Prediction (Decision):",
custom_pred_tree_decision, "\n")
```

output(Decision for model):

Logistic Regression Prediction (Probability): 0.06128625

Logistic Regression Prediction (Decision): NOT Accepted

Decision Tree Prediction (Decision): NOT Accepted

Conclusion:

Summary of Findings

We trained two models, logistic regression and a decision tree, to predict loan approval using various features. The logistic regression model was slightly more accurate than the decision tree model. However, the decision tree was easier to understand, as it made the decision-making process more transparent.

Recommendations and Next Steps

- **Model Tuning**: Both models can be improved by tuning hyperparameters, especially for the decision tree.
- **Cross-Validation**: Implementing cross-validation could improve the robustness of the models.
- **Feature Selection**: Further analysis of feature importance could help improve the models by selecting the most relevant features.

References:

Dataset from:

https://www.kaggle.com/datasets/ninzaami/loan-predication