Predictive Analytics for Minimizing Default Risks in Loan Portfolios

Olapeju Esuola

2024-04-9

INTRODUCTION

The business problem at hand is to develop a strategy to accurately predict and identify customers who are at risk of defaulting on their loans. This is crucial as the majority of a bank's earnings still stem from net interest earnings, making it imperative to minimize default rates while maximizing loan distribution.

In this report, I conducted an exploratory data analysis of the loan_default data-set. Subsequently, I applied Logistic Regression and decision trees to analyze the data and predict which customers are likely to default on their loans. This approach aims to provide actionable insights for mitigating default risks effectively.

Description of Variables for loan_default data-set:

1. Checking_Amount (Numeric): The amount of money in the borrower's checking account.
2. Term (Numeric): The duration of the loan in months.
3. Credit_score (Numeric): The credit score of the borrower.
4. Marital_status (Categorical): The marital status of the borrower.
5. Car_loan (Numeric): Indicates whether the borrower owns a car loan (1 for owning a car loan, 0 otherwise).
6. Personal_loan (Numeric): Indicates whether the borrower owns a personal loan (1 for owning a personal loan, 0 otherwise).
7. Home_loan (Numeric): Indicates whether the borrower owns a home loan (1 for owning a home loan, 0 otherwise).
8. Education_loan (Numeric): Indicates whether the borrower owns an education loan (1 for owning an education loan, 0 otherwise).
9. Emp_status (Categorical): The employment status of the borrower.
10. Amount (Numeric): The amount of the loan.
11. Saving_amount (Numeric): The amount of savings the borrower has.
12. Emp_duration (Numeric): The duration of employment in months.
13. Age (Numeric): The age of the borrower in years.
14. No_of_credit_account (Numeric): The number of credit accounts the borrower has.
15. Default (Response Variable): Indicates whether the borrower defaulted on the bank loan (1 for default, 0 for non-default).

EXPLORATORY DATA ANALYSIS

Data Summary

```
#import libraries
library(readr)
library(tidyverse)
library(skimr)
library(ggplot2)
library(knitr)

#Load the data-set
loan_default<- read.csv("C:/Users/olape/Downloads/loan_default.csv")

#Use the skim function to check the data summary
skim(loan_default)
```

Data summary

Name	loan_default
Number of rows	1000
Number of columns	16
________________________	
Column type frequency:	
character	3
numeric	13
________________________	
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace

Sex	0	1	4	6	0	2	0
Marital_status	0	1	6	7	0	2	0
Emp_status	0	1	8	10	0	2	0

Variable type: numeric

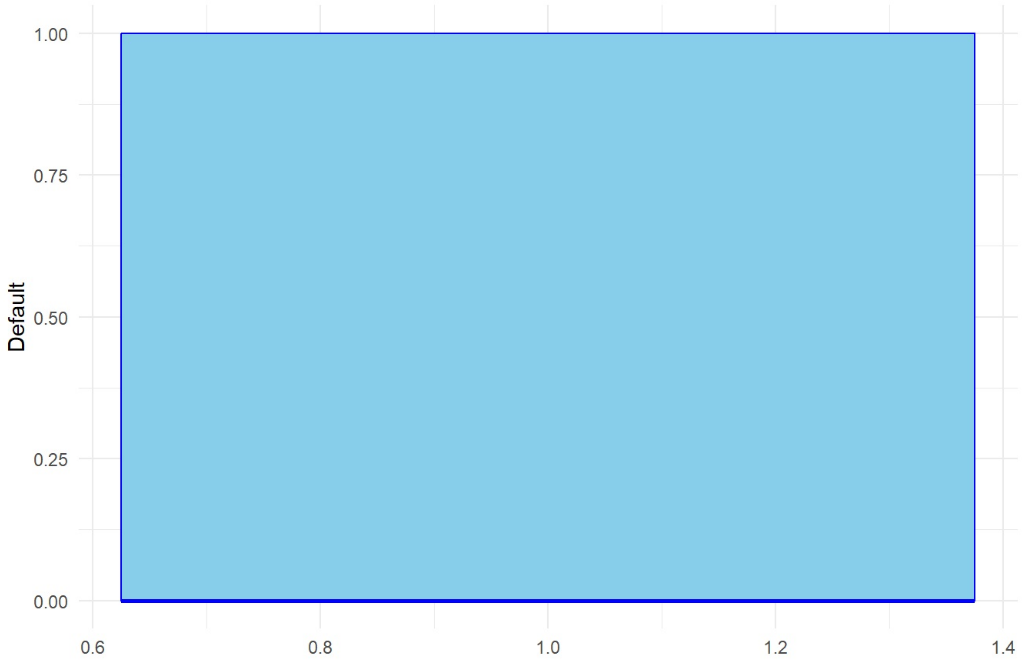
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Default	0	1	0.30	0.46	0	0.00	0.0	1.00	1	▇▁▁▁▃
Checking_amount	0	1	362.41	300.90	-665	164.75	351.5	553.50	1319	▁▂▇▂▁
Term	0	1	17.82	3.24	9	16.00	18.0	20.00	27	▁▆▇▆▁
Credit_score	0	1	760.48	77.56	376	725.75	770.5	812.00	1029	▁▁▅▇▁
Car_loan	0	1	0.35	0.48	0	0.00	0.0	1.00	1	▇▁▁▁▅
Personal_loan	0	1	0.47	0.50	0	0.00	0.0	1.00	1	▇▁▁▁▇
Home_loan	0	1	0.06	0.23	0	0.00	0.0	0.00	1	▇▁▁▁▁
Education_loan	0	1	0.11	0.32	0	0.00	0.0	0.00	1	▇▁▁▁▁
Amount	0	1	1218.68	305.75	244	1016.00	1225.5	1419.75	2362	▁▅▇▂▁
Saving_amount	0	1	3179.27	339.55	2082	2951.00	3203.0	3402.25	4108	▁▂▇▆▁
Emp_duration	0	1	49.39	37.76	0	15.00	41.0	85.00	120	▇▅▂▃▃
Age	0	1	31.21	4.09	18	29.00	32.0	34.00	42	▁▂▇▆▁
No_of_credit_acc	0	1	2.55	1.65	1	1.00	2.0	3.00	9	▇▂▂▁▁

Upon utilizing a skim function to review the data summary, it was observed that the data-set comprises 1000 rows and 16 columns. Notably, no missing values were identified within the data-set. Furthermore, the summary revealed the presence of 3 character variables and 13 numeric variables.

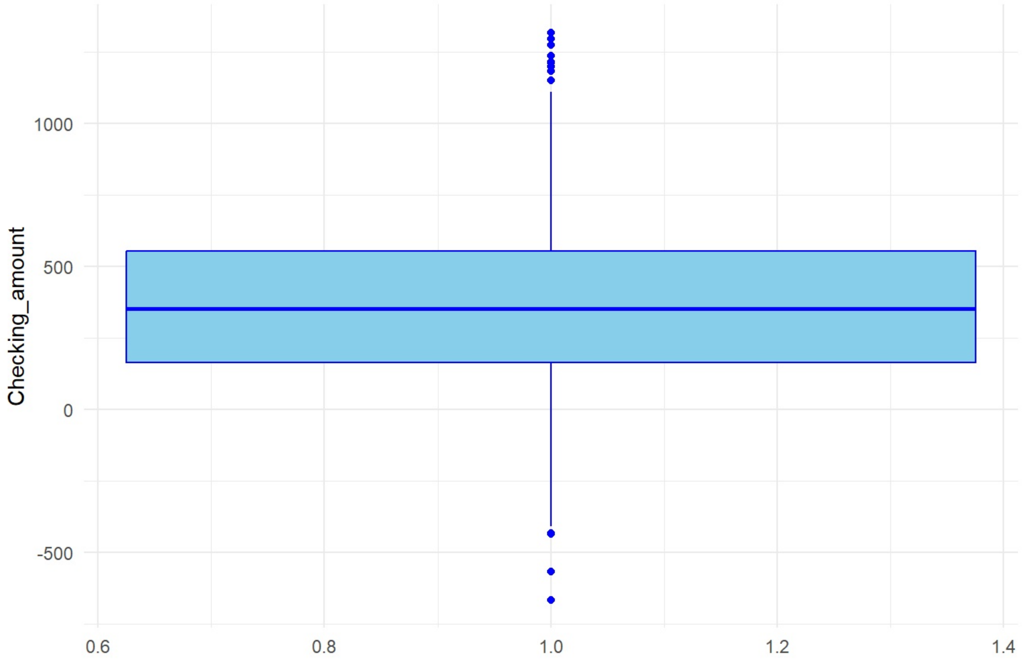
Exploring Outliers in the loan_default Dataset: An Analysis of Potential Anomalies

```
# Create boxplots for each numeric variable
for (col in names(loan_default)) {
  if (is.numeric(loan_default[[col]])) {
    plot_output <- ggplot(loan_default, aes(x = 1, y = loan_default[[col]])) +
      geom_boxplot(fill = "skyblue", color = "blue") +
      ggtitle(paste("Boxplot for", col)) +
      xlab("") +
      ylab(col) +
      theme_minimal()  # Optional: Use a minimal theme

    print(plot_output)
  }
}
```

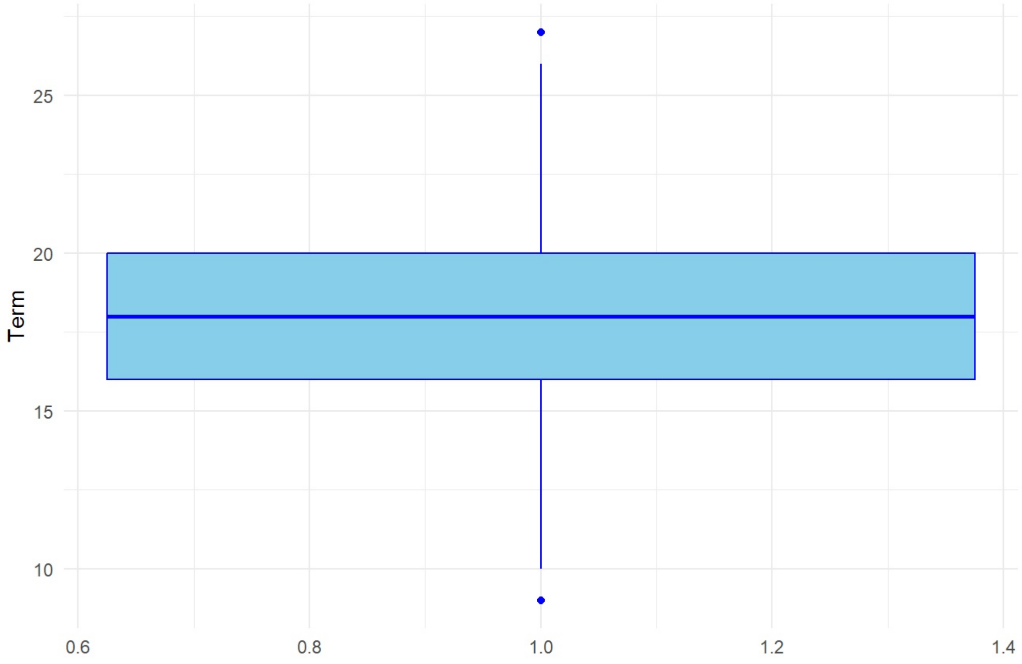
Boxplot for Default



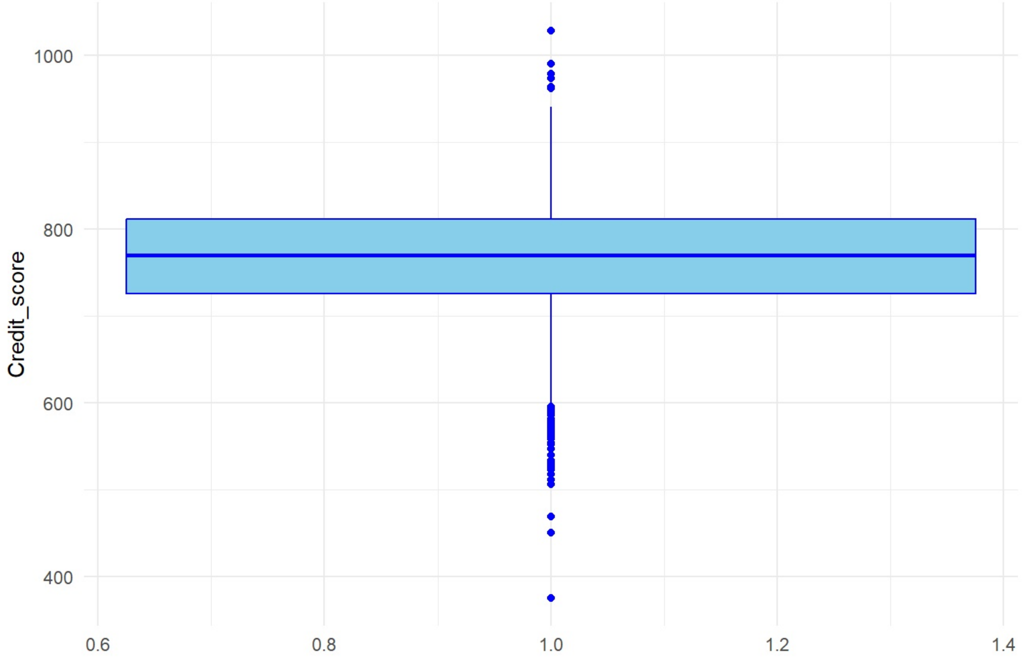
Boxplot for Checking_amount



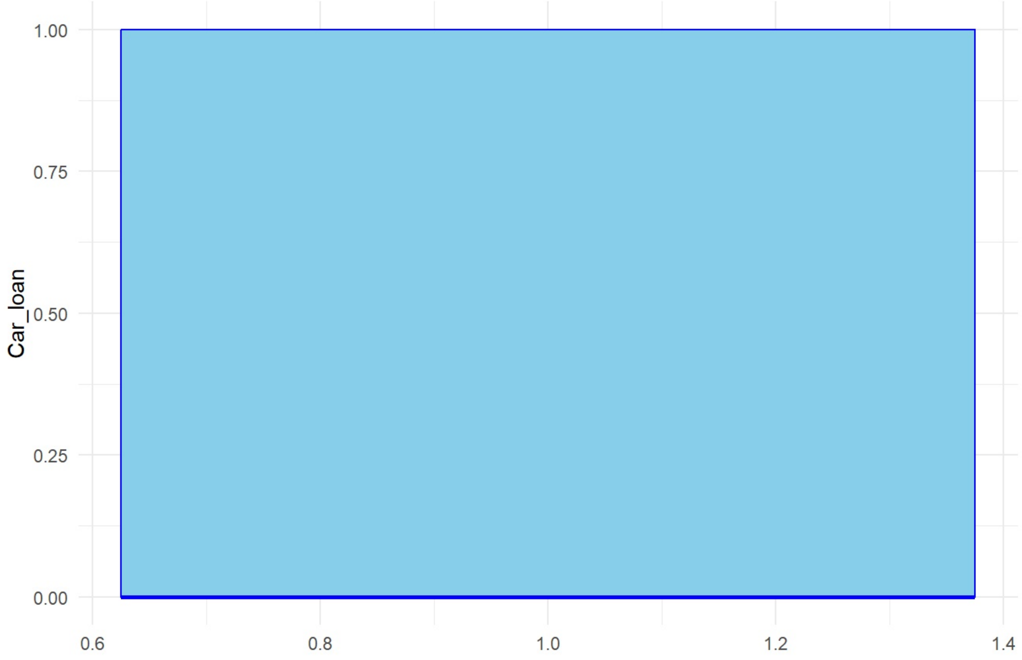
Boxplot for Term

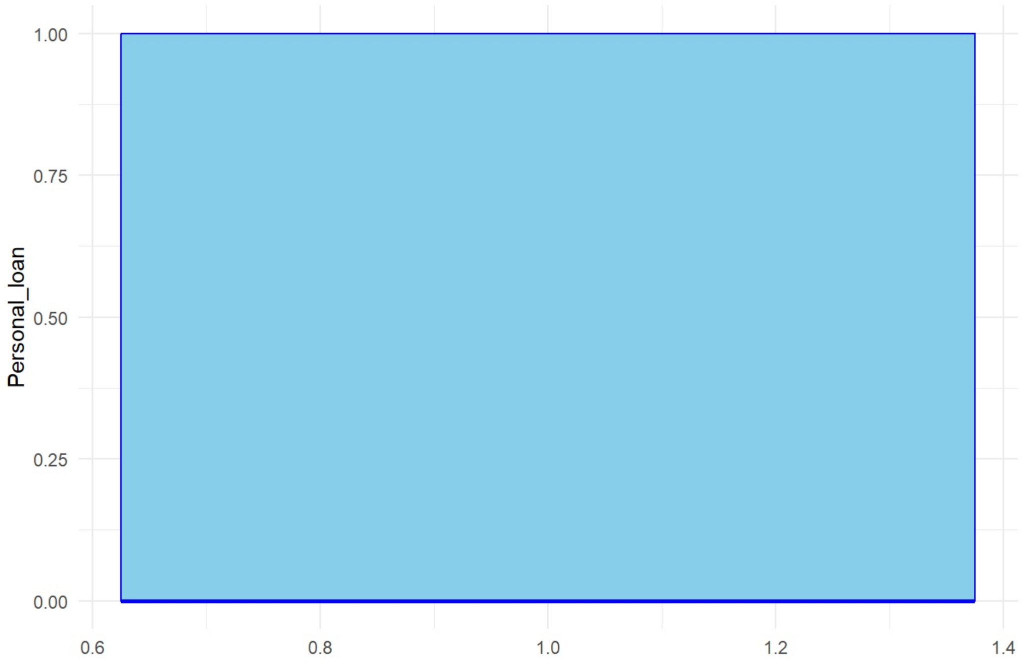


Boxplot for Credit_score

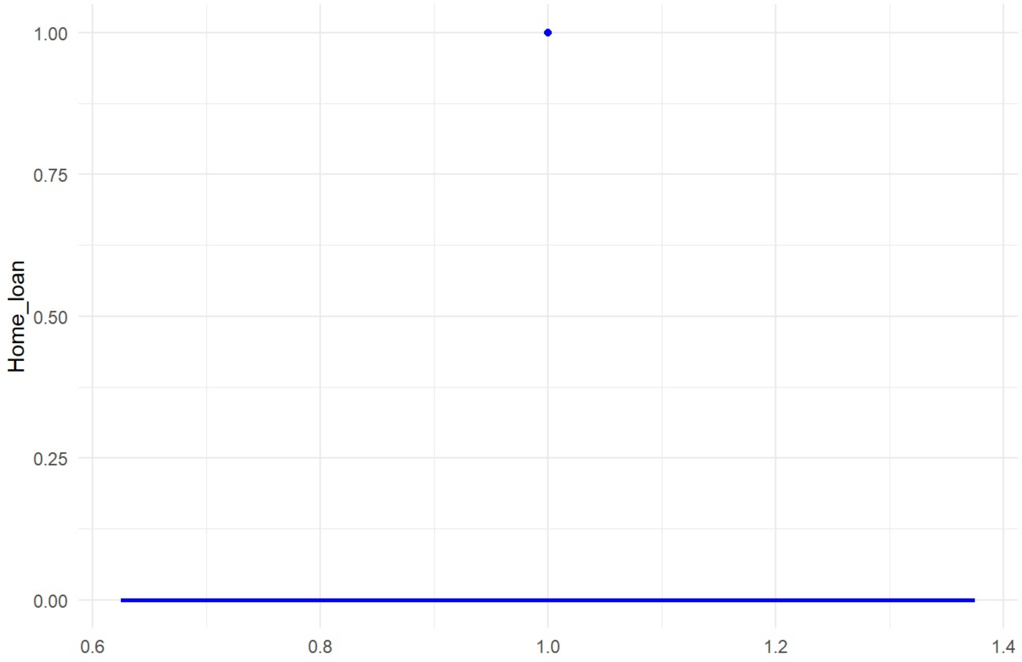


Boxplot for Car_loan

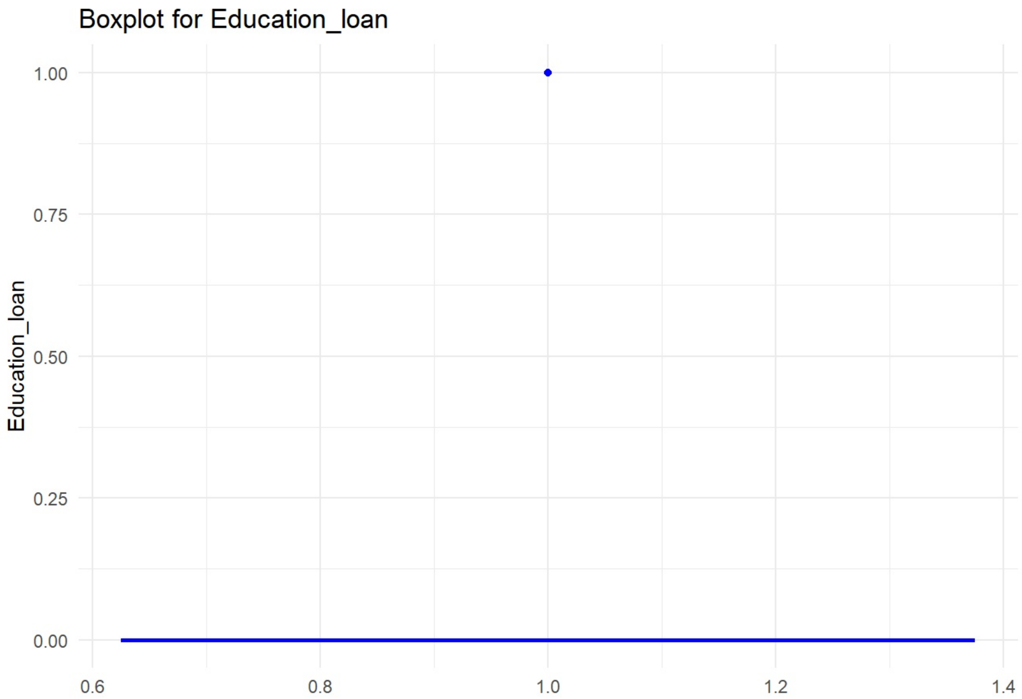




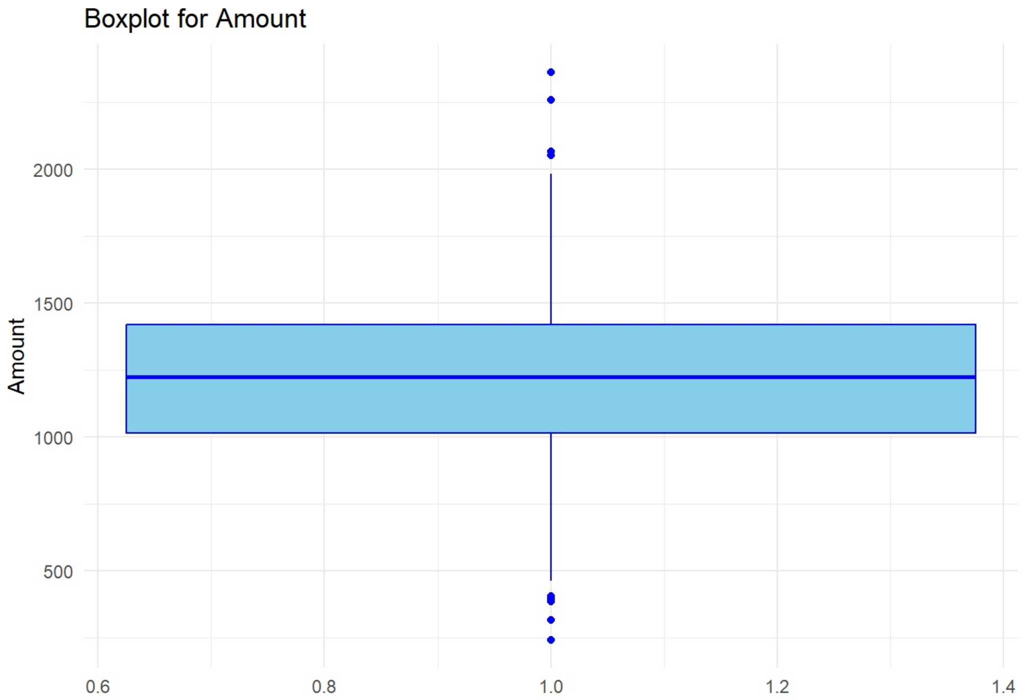
Boxplot for Personal_loan



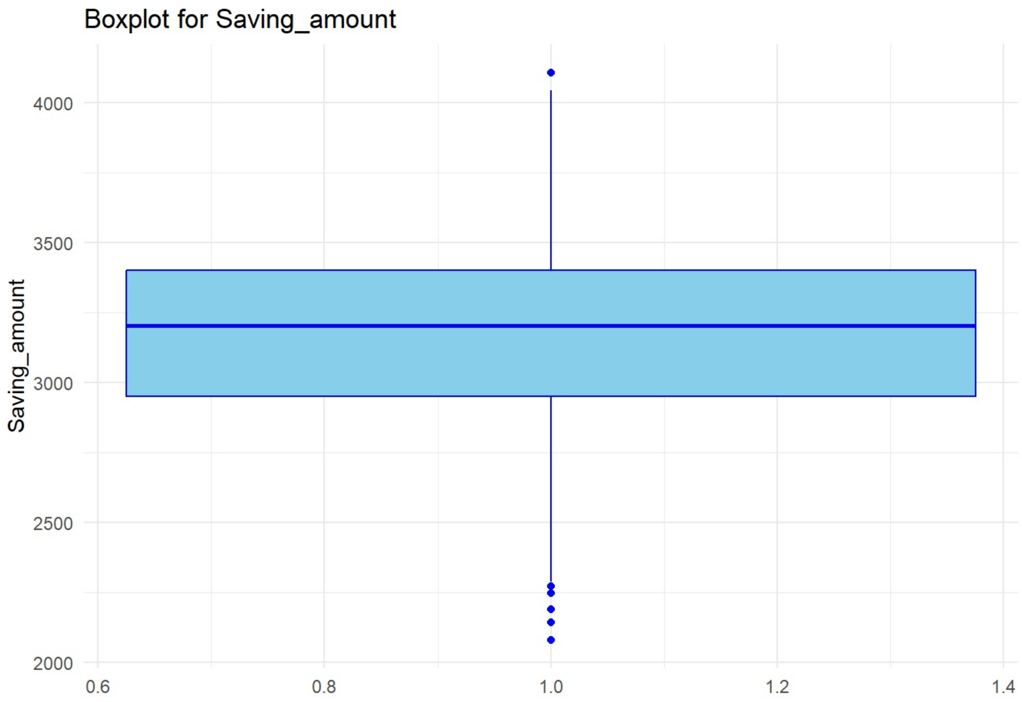
Boxplot for Home_loan

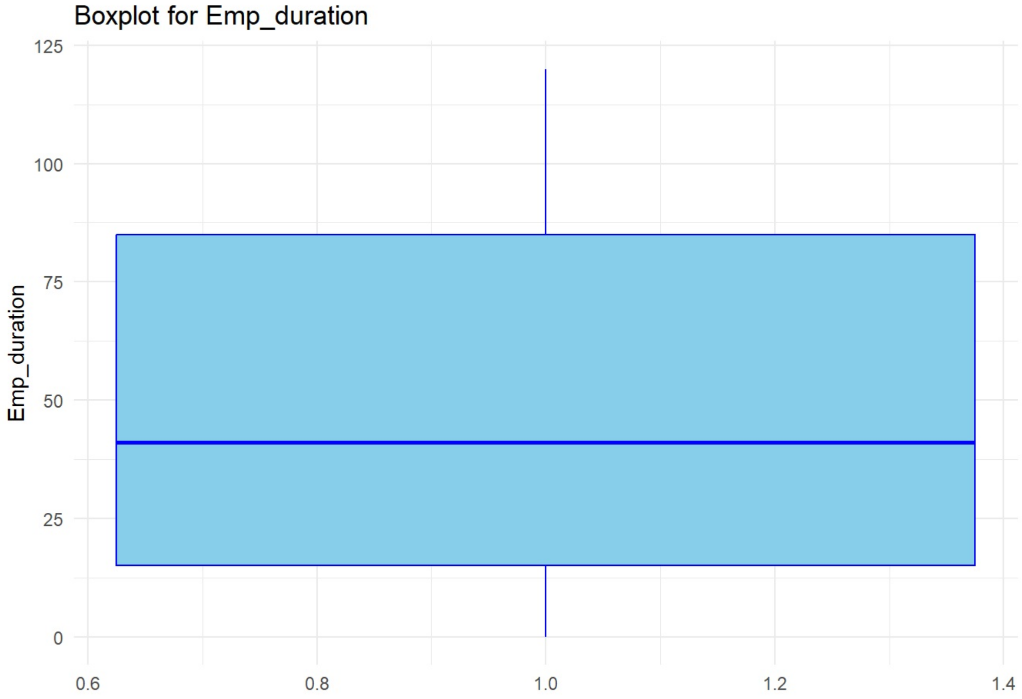


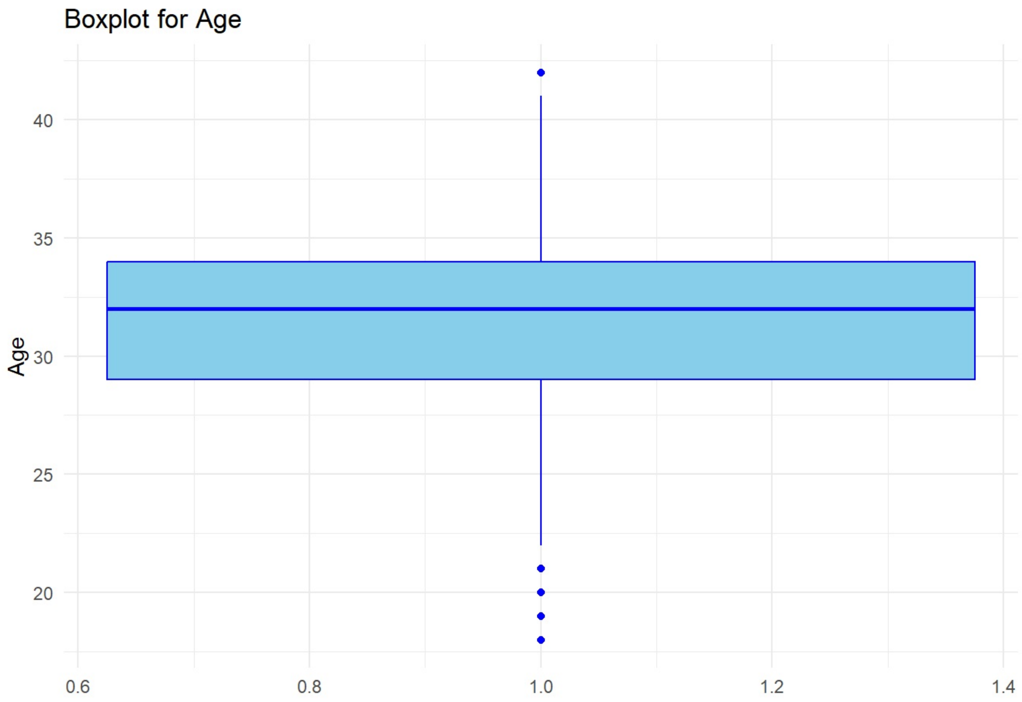
Boxplot for Education_loan

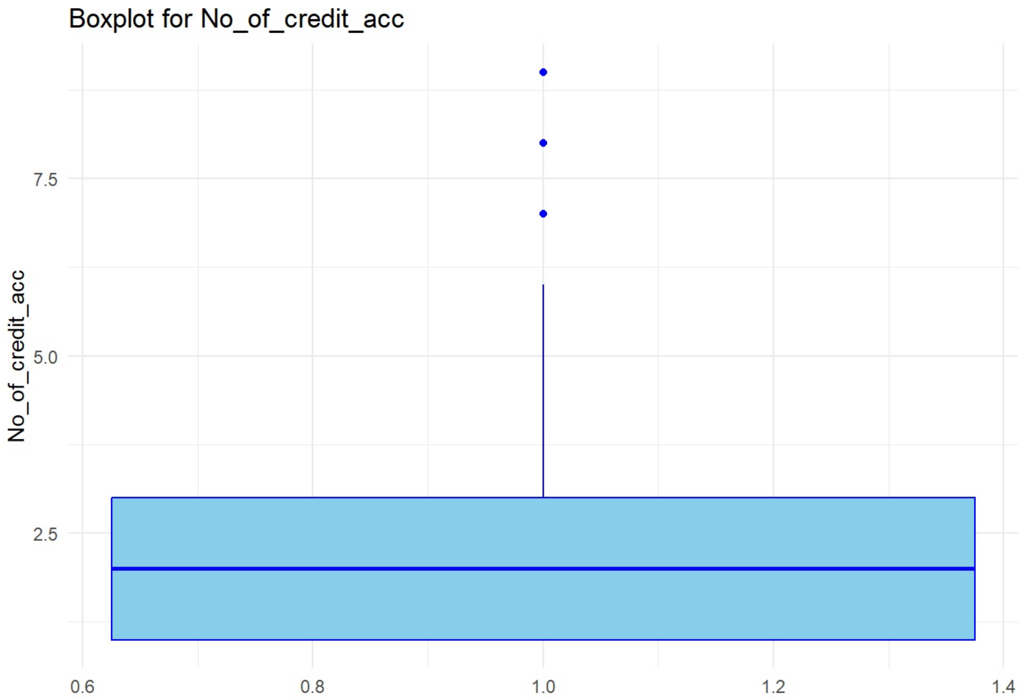


Boxplot for Amount


Boxplot for Saving_amount


Boxplot for Emp_duration





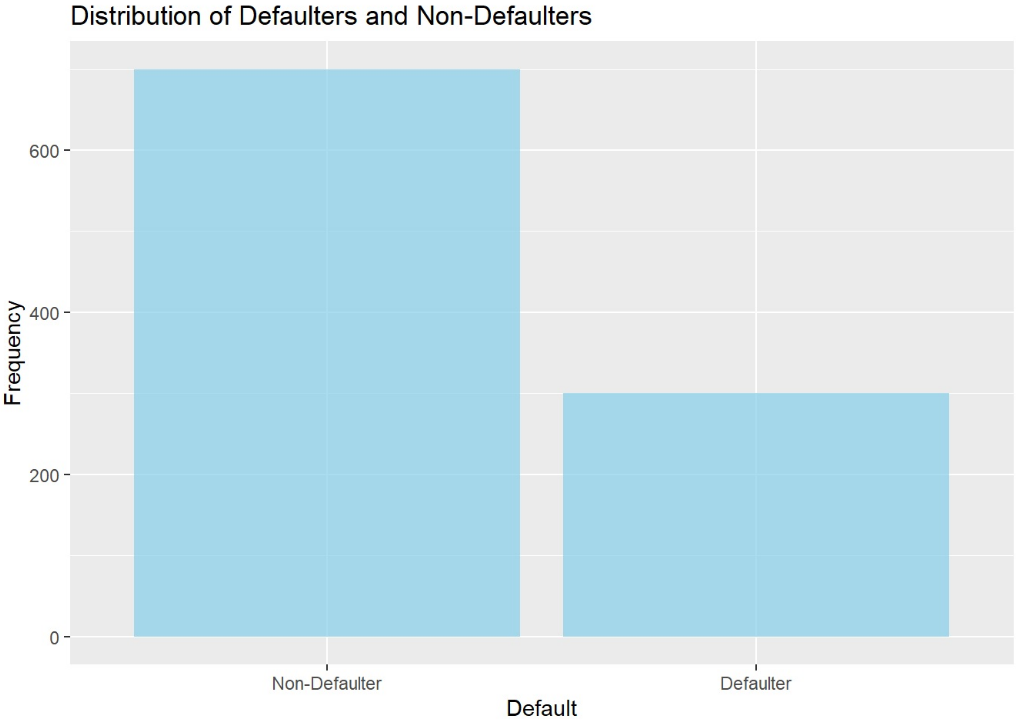
To identify outliers, I utilized box plots and analyzed each plot meticulously to detect potential anomalies in the numeric variables. Subsequently, it was observed that the following variables exhibited outliers: Checking Amount, Term, Credit Score, Home Loan, Education Loan, Amount, Saving Amount, Age, and Number of Credit Accounts. The boxplot for Default, Car Loan, Personal Loan, Education Loan and Personal Loan cannot be interpreted because they are binary variables.

Exploring the response variable- Default

```
library(dplyr)
# Calculate the frequency of defaulters (Default = 1) and non-defaulters (Default = 0)
default_freq <- loan_default %>%
  group_by(Default) %>%
  summarize(Frequency = n())

# Print the frequency table
print(default_freq)
```

```
## # A tibble: 2 × 2
##   Default Frequency
##     <int>     <int>
## 1       0       700
## 2       1       300
```

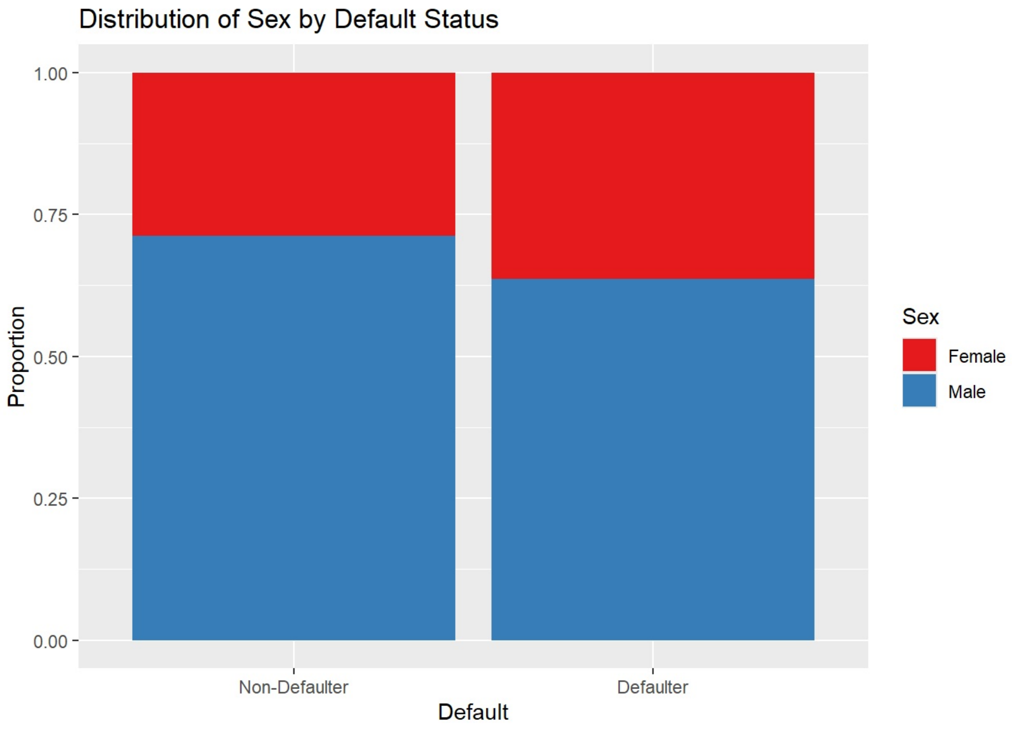
```
# Create a bar chart to visualize the distribution of defaulters and non-defaulters
ggplot(loan_default, aes(x = factor(Default))) +
  geom_bar(fill = "skyblue", alpha = 0.7) +
  labs(title = "Distribution of Defaulters and Non-Defaulters",
       x = "Default",
       y = "Frequency") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter"))
```



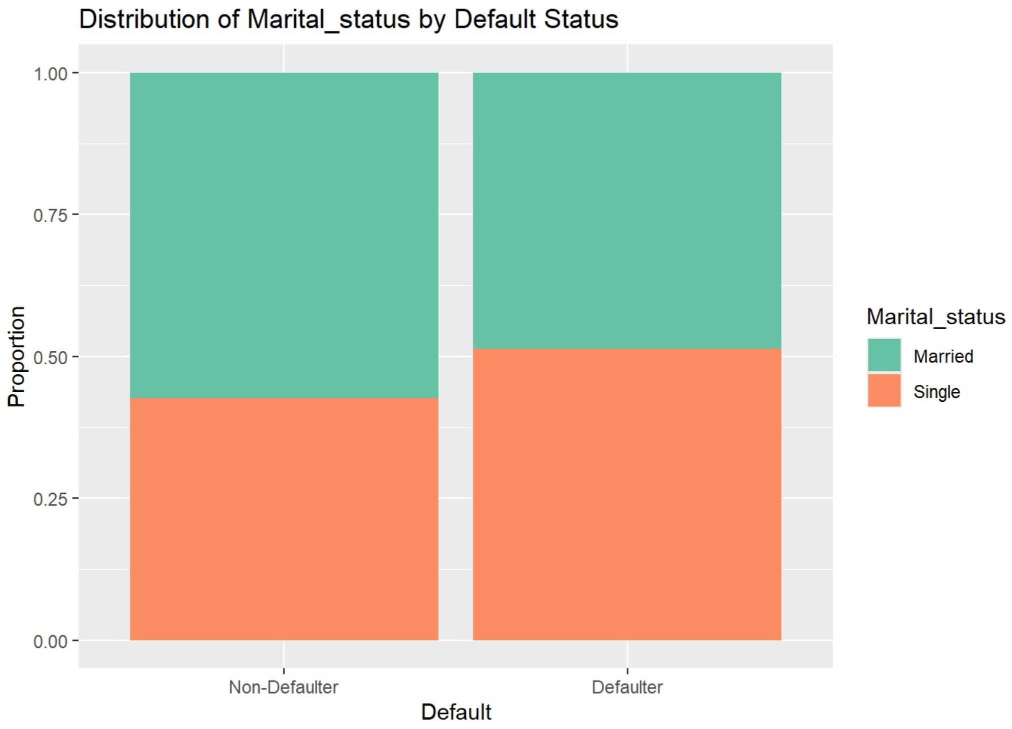
After comparing defaulters and non-defaulters, it’s evident that the number of non-defaulters is higher. This indicates a lower number of individuals who default on bank loans.

Exploring the relationship between the response variable (Default) and categorical variables.

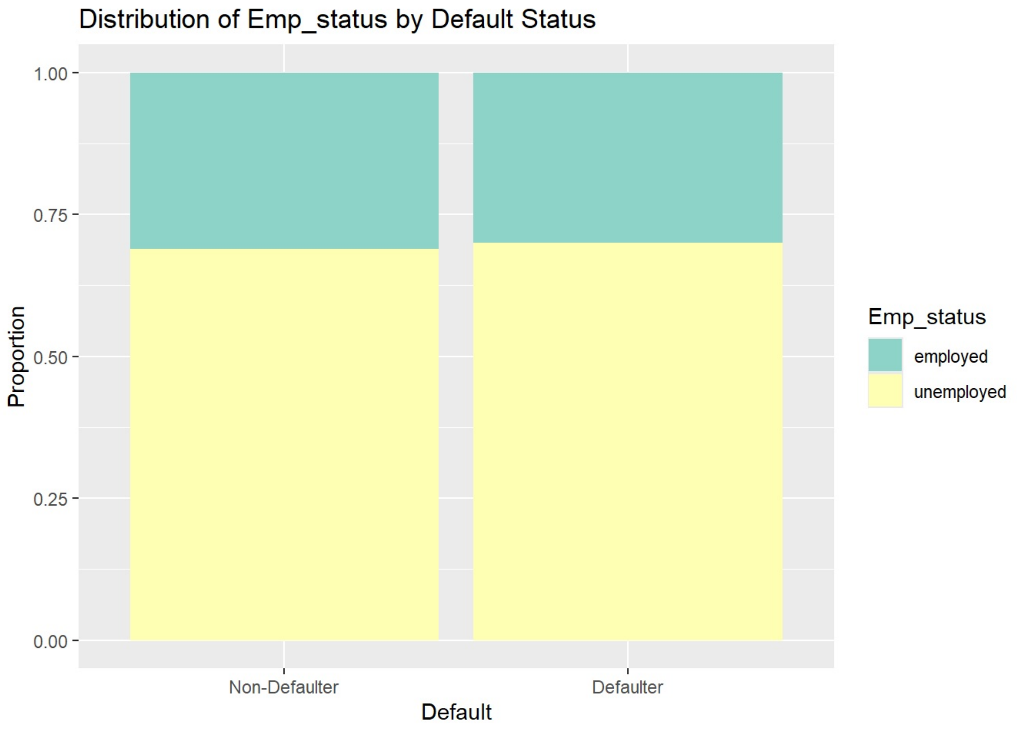
```
#Create a stacked bar chart to compare the distribution of Sex between defaulters and non-defaulters
ggplot(loan_default, aes(x = factor(Default), fill = Sex)) +
  geom_bar(position = "fill") +
  labs(title = "Distribution of Sex by Default Status",
       x = "Default",
       y = "Proportion") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_brewer(palette = "Set1")
```



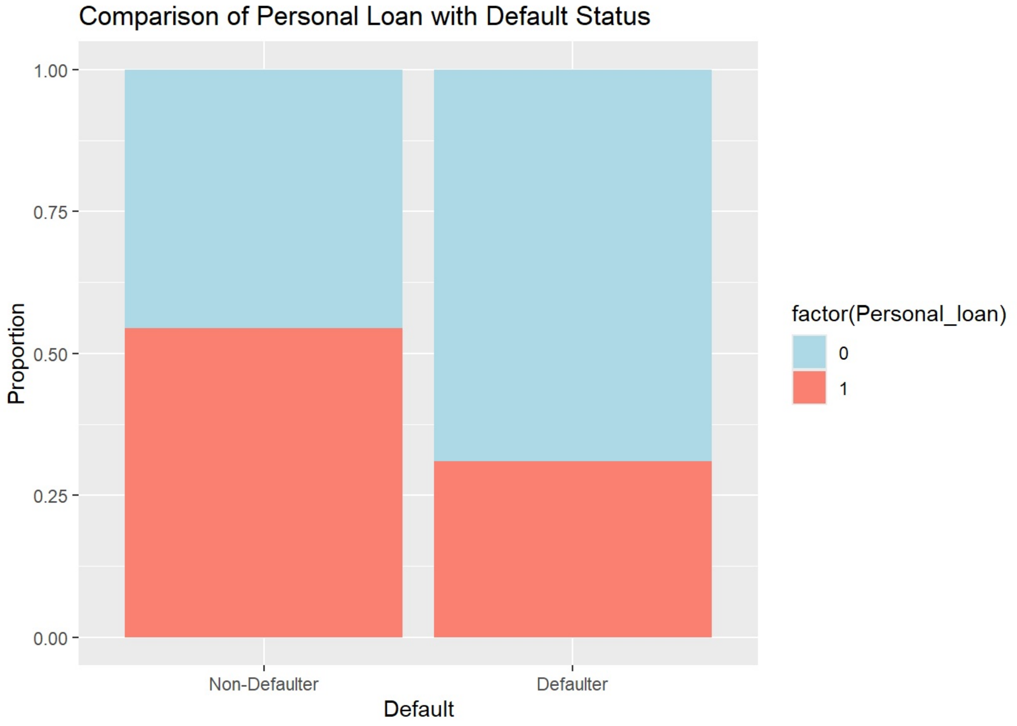
```
#Create a stacked bar chart to compare the distribution of Marital_status between defaulters and non-defaulters
ggplot(loan_default, aes(x = factor(Default), fill = Marital_status)) +
  geom_bar(position = "fill") +
  labs(title = "Distribution of Marital_status by Default Status",
       x = "Default",
       y = "Proportion") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_brewer(palette = "Set2")
```



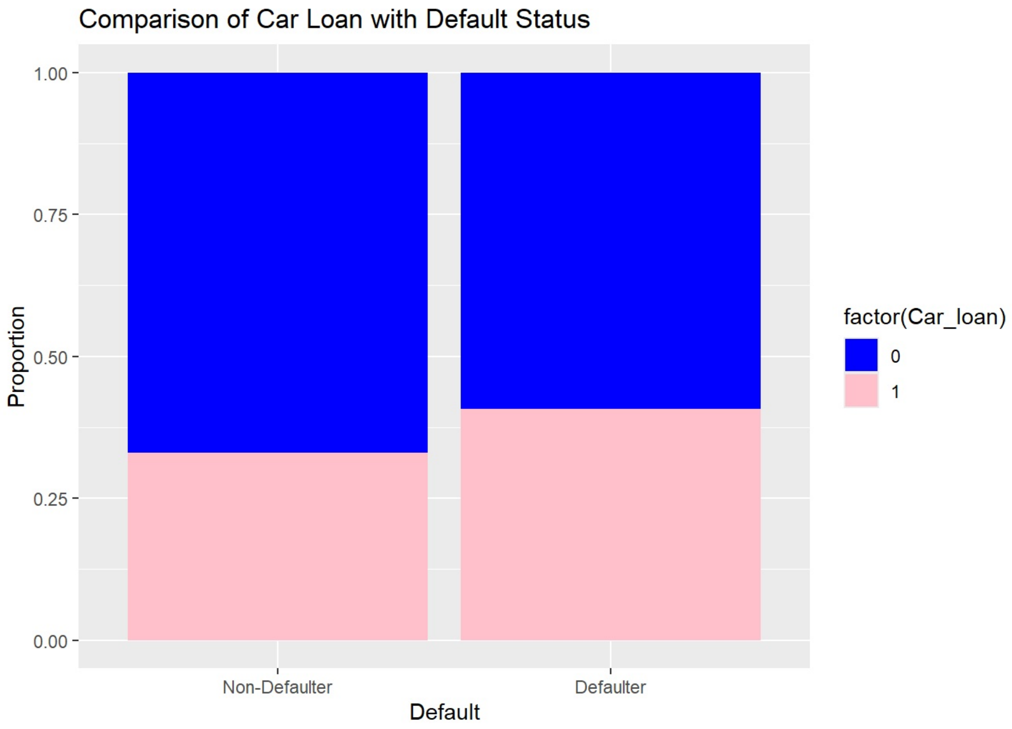
```
#Create a stacked bar chart to compare the distribution of Emp_status between defaulters and non-defaulters
ggplot(loan_default, aes(x = factor(Default), fill = Emp_status)) +
  geom_bar(position = "fill") +
  labs(title = "Distribution of Emp_status by Default Status",
       x = "Default",
       y = "Proportion") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_brewer(palette = "Set3")
```



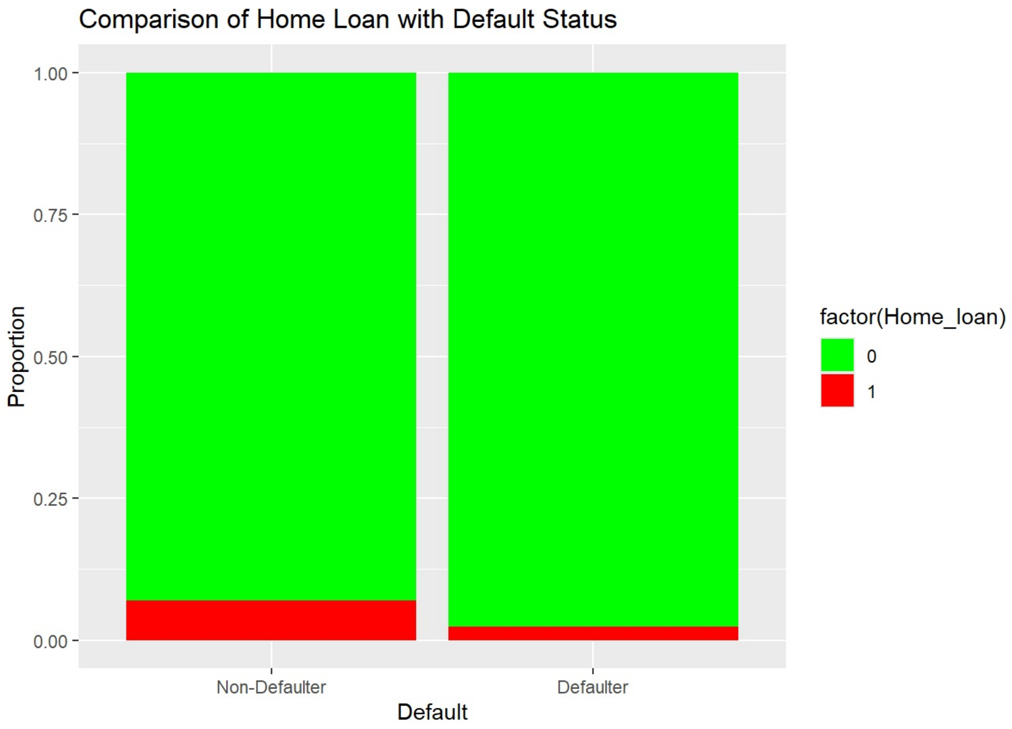
```
# Create a stacked bar chart to compare binary categorical variables with the 'Default' response variable
ggplot(loan_default, aes(x = factor(Default), fill = factor(Personal_loan))) +
  geom_bar(position = "fill") +
  labs(title = "Comparison of Personal Loan with Default Status",
       x = "Default",
       y = "Proportion") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_manual(values = c("lightblue", "salmon"))
```



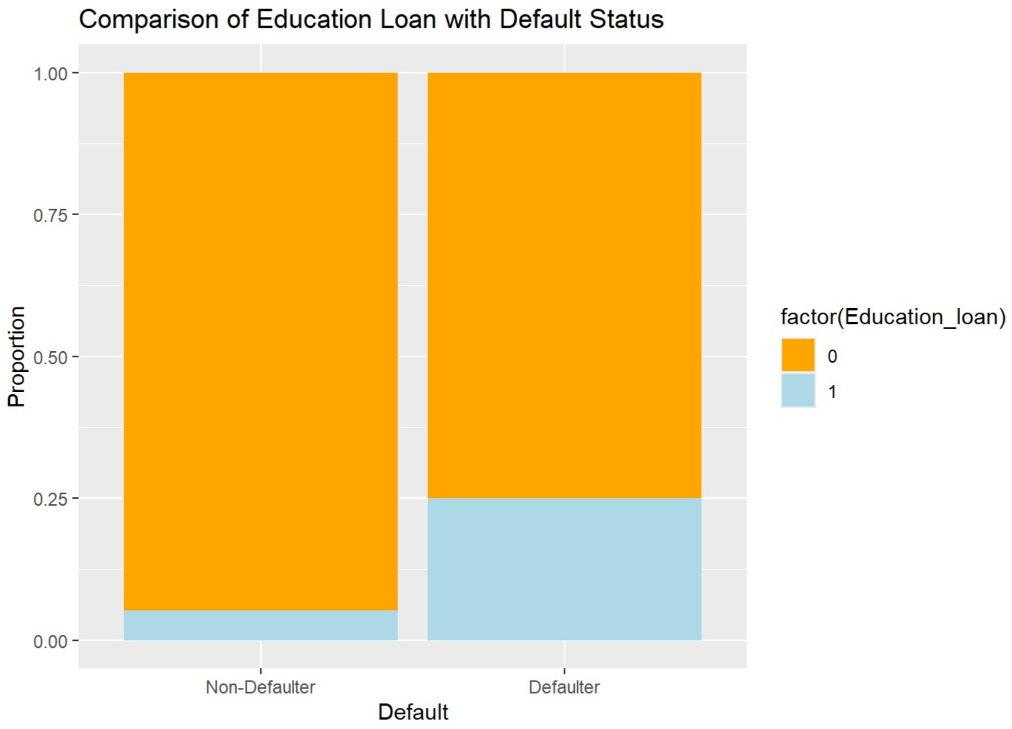
```
# Create a stacked bar chart to compare binary categorical variables with the 'Default' response variable
ggplot(loan_default, aes(x = factor(Default), fill = factor(Car_loan))) +
  geom_bar(position = "fill") +
  labs(title = "Comparison of Car Loan with Default Status",
       x = "Default",
       y = "Proportion") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_manual(values = c("blue", "pink"))
```



```
# Create a stacked bar chart to compare binary categorical variables with the 'Default' response variable
ggplot(loan_default, aes(x = factor(Default), fill = factor(Home_loan))) +
  geom_bar(position = "fill") +
  labs(title = "Comparison of Home Loan with Default Status",
       x = "Default",
       y = "Proportion") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_manual(values = c("green", "red"))
```



```
# Create a stacked bar chart to compare binary categorical variables with the 'Default' response variable
ggplot(loan_default, aes(x = factor(Default), fill = factor(Education_loan))) +
  geom_bar(position = "fill") +
  labs(title = "Comparison of Education Loan with Default Status",
       x = "Default",
       y = "Proportion") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_manual(values = c("orange", "lightblue"))
```

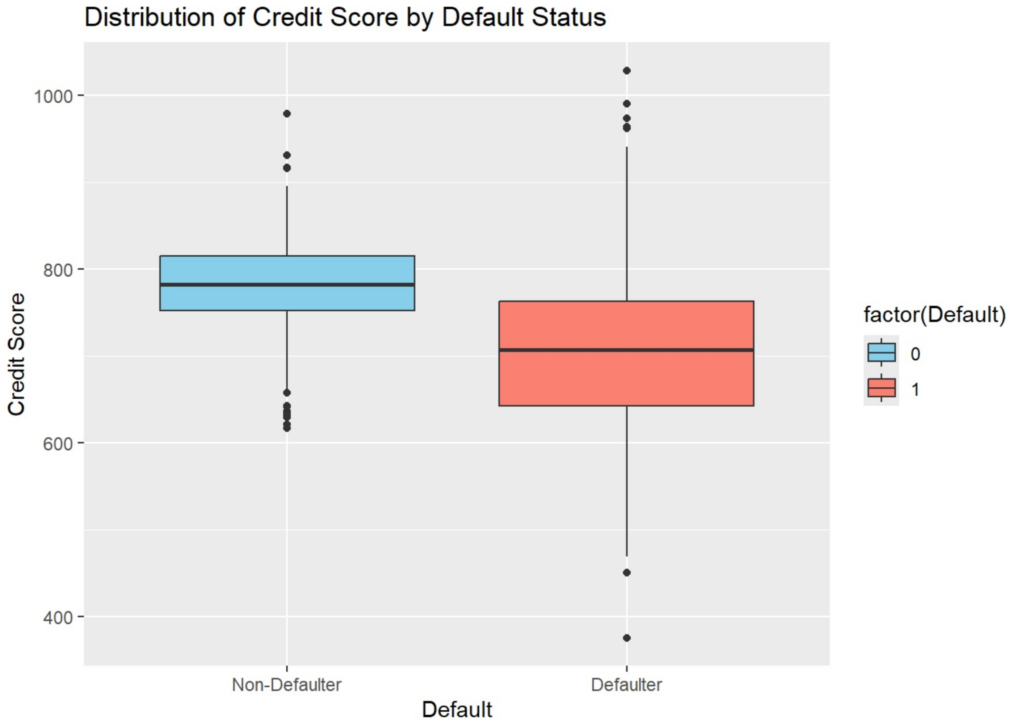


Based on the stacked bar chart analysis, it appears that males are more likely to default on bank loans compared to females. Additionally, both single and married individuals show similar likelihoods of defaulting, but there are more married non-defaulters. Moreover, the analysis suggests that unemployed individuals are more likely to default on bank loans compared to those who are employed.

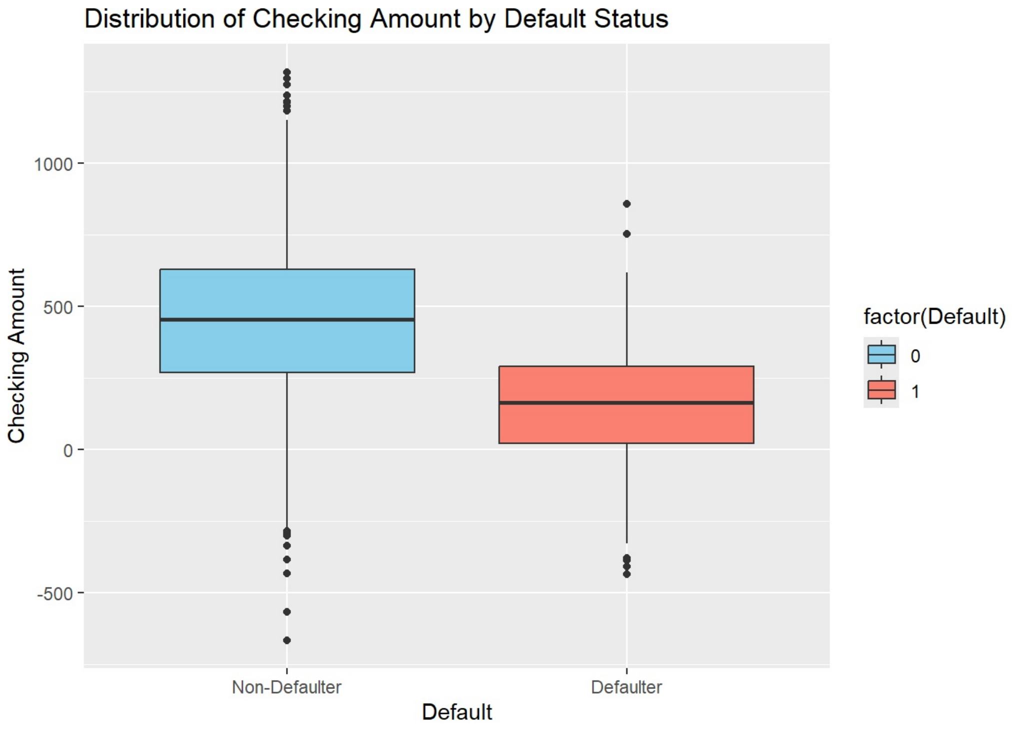
Individuals with a value of 0 (indicating the absence of a personal loan, car loan, education loan, or home loan) are more likely to default on their bank loans. This insight suggests that the absence of these types of loans may be associated with a higher risk of default. It's essential for financial institutions to consider this finding when assessing the creditworthiness of borrowers and managing loan portfolios.

Exploring the relationship between the response variable (Default) and categorical variables.

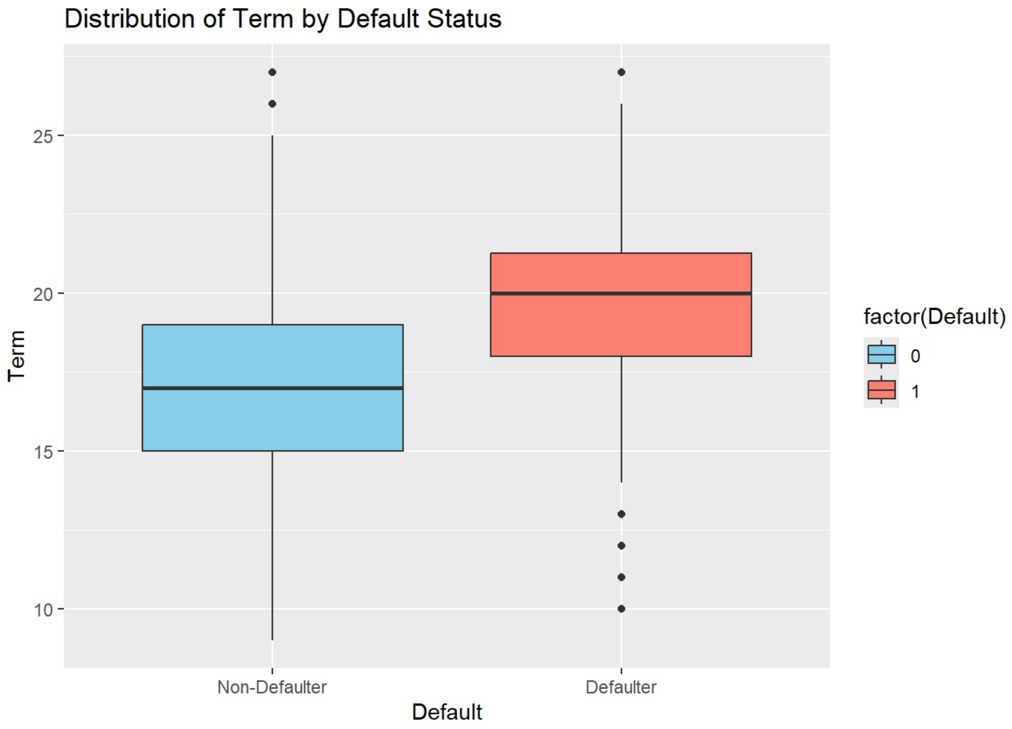
```
# Create a box plot to compare the distribution of 'Credit_score' between defaulters and non-defaulters
ggplot(loan_default, aes(x = factor(Default), y = Credit_score, fill = factor(Default))) +
  geom_boxplot() +
  labs(title = "Distribution of Credit Score by Default Status",
       x = "Default",
       y = "Credit Score") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_manual(values = c("skyblue", "salmon"))
```



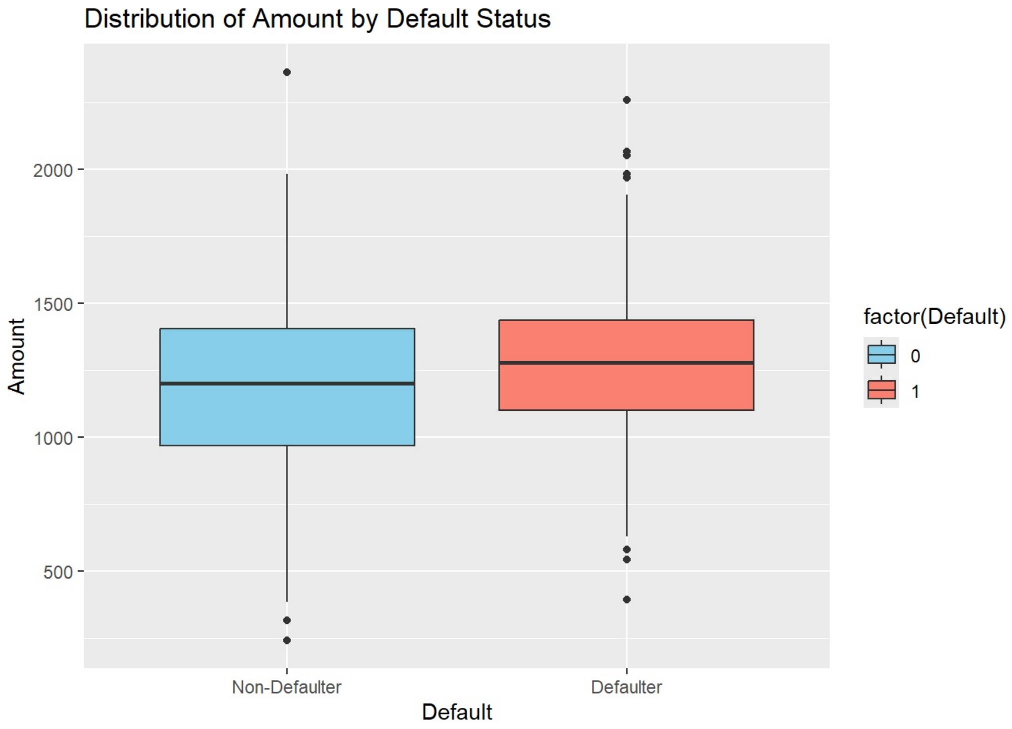
```
# Create a box plot to compare the distribution of 'Checking_amount' between defaulters and non-defaulters
ggplot(loan_default, aes(x = factor(Default), y = Checking_amount, fill = factor(Default))) +
  geom_boxplot() +
  labs(title = "Distribution of Checking Amount by Default Status",
       x = "Default",
       y = "Checking Amount") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_manual(values = c("skyblue", "salmon"))
```



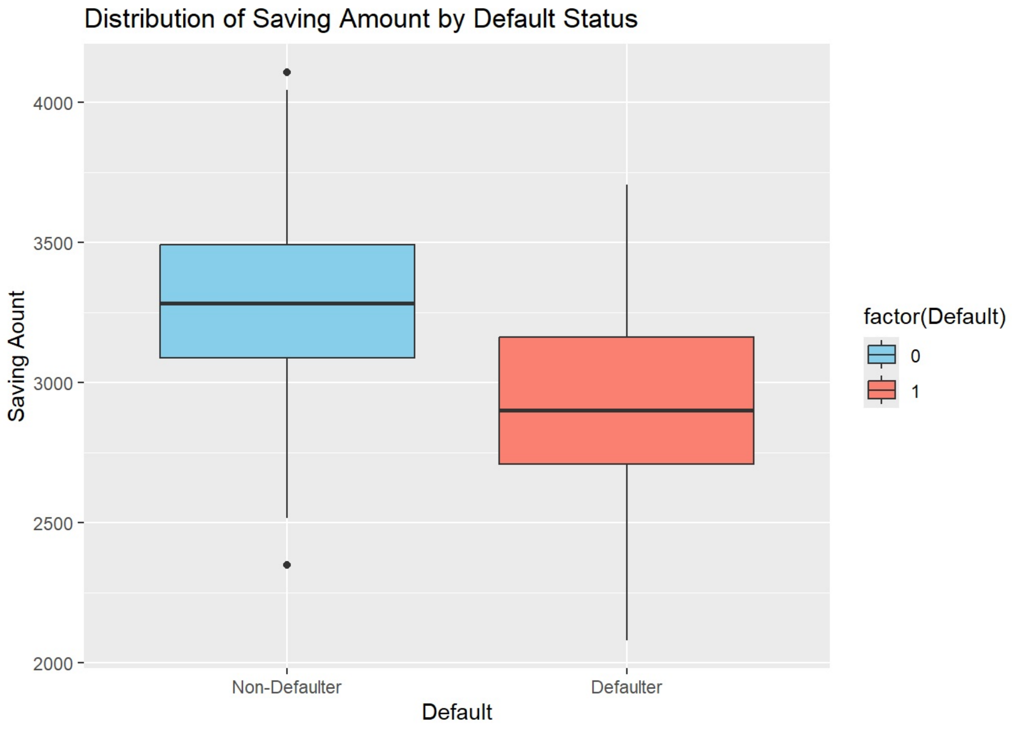
```
# Create a box plot to compare the distribution of 'Term' between defaulters and non-defaulters
ggplot(loan_default, aes(x = factor(Default), y = Term, fill = factor(Default))) +
  geom_boxplot() +
  labs(title = "Distribution of Term by Default Status",
       x = "Default",
       y = "Term") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_manual(values = c("skyblue", "salmon"))
```



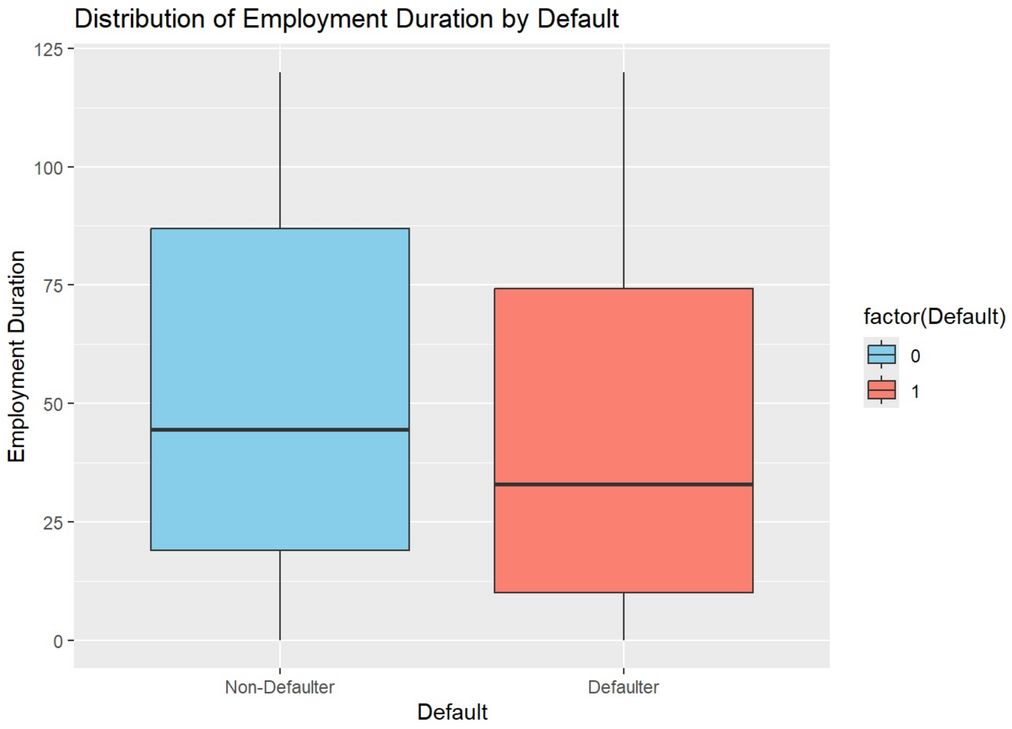
```
# Create a box plot to compare the distribution of 'Amount' between defaulters and non-defaulters
ggplot(loan_default, aes(x = factor(Default), y = Amount, fill = factor(Default))) +
  geom_boxplot() +
  labs(title = "Distribution of Amount by Default Status",
       x = "Default",
       y = "Amount") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_manual(values = c("skyblue", "salmon"))
```



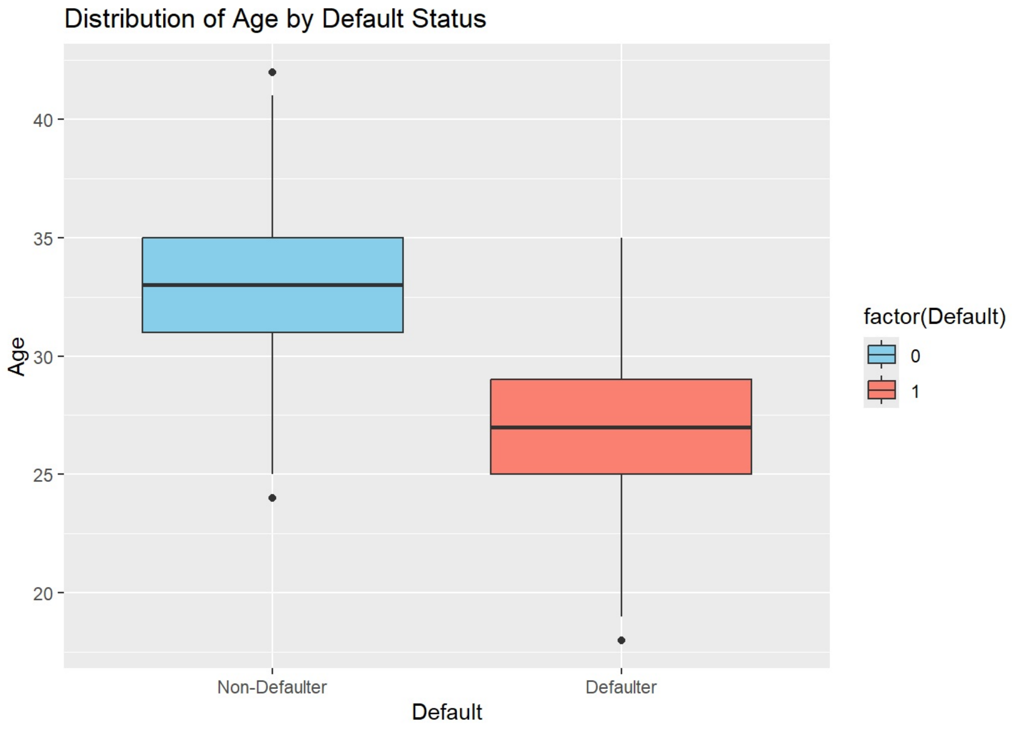
```
# Create a box plot to compare the distribution of 'Saving_amount' between defaulters and non-defaulters
ggplot(loan_default, aes(x = factor(Default), y = Saving_amount, fill = factor(Default))) +
  geom_boxplot() +
  labs(title = "Distribution of Saving Amount by Default Status",
       x = "Default",
       y = "Saving Aount") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_manual(values = c("skyblue", "salmon"))
```



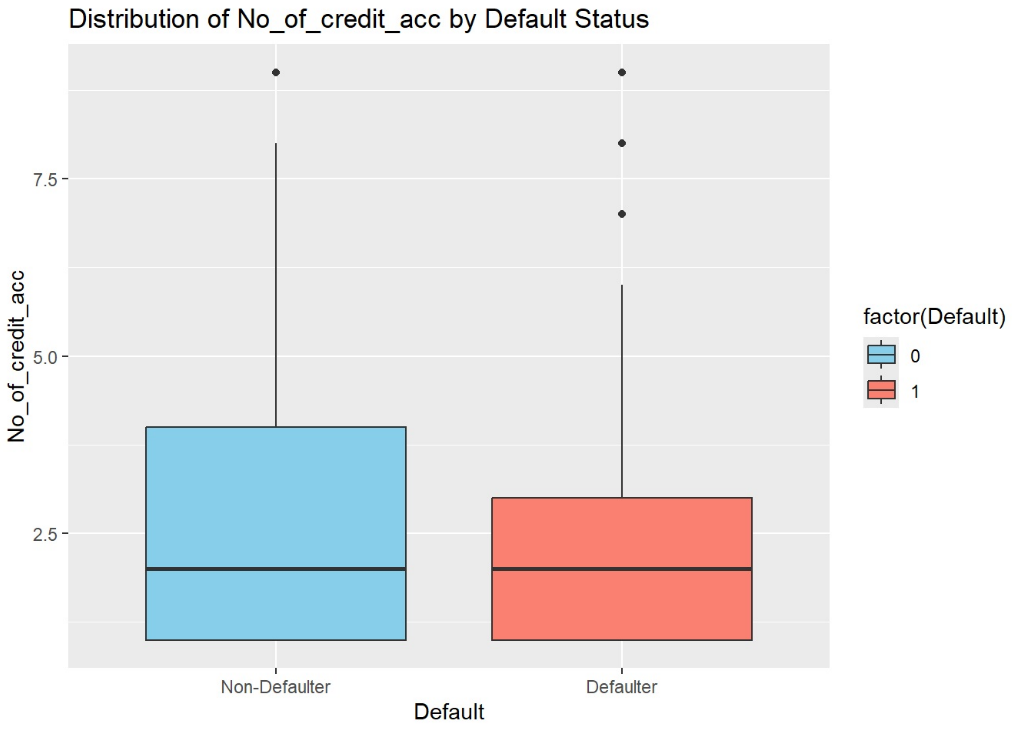
```r
# Create a box plot to compare the distribution of 'Emp_duration' between defaulters and non-defaulters
ggplot(loan_default, aes(x = factor(Default), y = Emp_duration, fill = factor(Default))) +
  geom_boxplot() +
  labs(title = "Distribution of Employment Duration by Default",
       x = "Default",
       y = "Employment Duration") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_manual(values = c("skyblue", "salmon"))
```



```r
# Create a box plot to compare the distribution of 'Age' between defaulters and non-defaulters
ggplot(loan_default, aes(x = factor(Default), y = Age, fill = factor(Default))) +
  geom_boxplot() +
  labs(title = "Distribution of Age by Default Status",
       x = "Default",
       y = "Age") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_manual(values = c("skyblue", "salmon"))
```



```
# Create a box plot to compare the distribution of 'No_of_credit_acc' between defaulters and non-defaulters
ggplot(loan_default, aes(x = factor(Default), y = No_of_credit_acc, fill = factor(Default))) +
  geom_boxplot() +
  labs(title = "Distribution of No_of_credit_acc by Default Status",
       x = "Default",
       y = "No_of_credit_acc") +
  scale_x_discrete(labels = c("Non-Defaulter", "Defaulter")) +
  scale_fill_manual(values = c("skyblue", "salmon"))
```



After plotting a boxplot to compare the response variable (default status) with various features such as credit score, checking amount, loan term, loan amount, saving amount, employment duration, and age, several patterns emerge:

Individuals with lower credit scores tend to default on their payments more frequently, indicating a correlation between creditworthiness and default risk.

Those with lower checking amounts are more likely to default on their loans, suggesting a potential financial strain or instability among this group.

Higher loan terms are associated with a higher likelihood of default, implying that longer loan durations may pose greater repayment challenges for borrowers.

Borrowers with higher loan amounts are more prone to default, possibly due to the increased financial burden associated with larger loans.

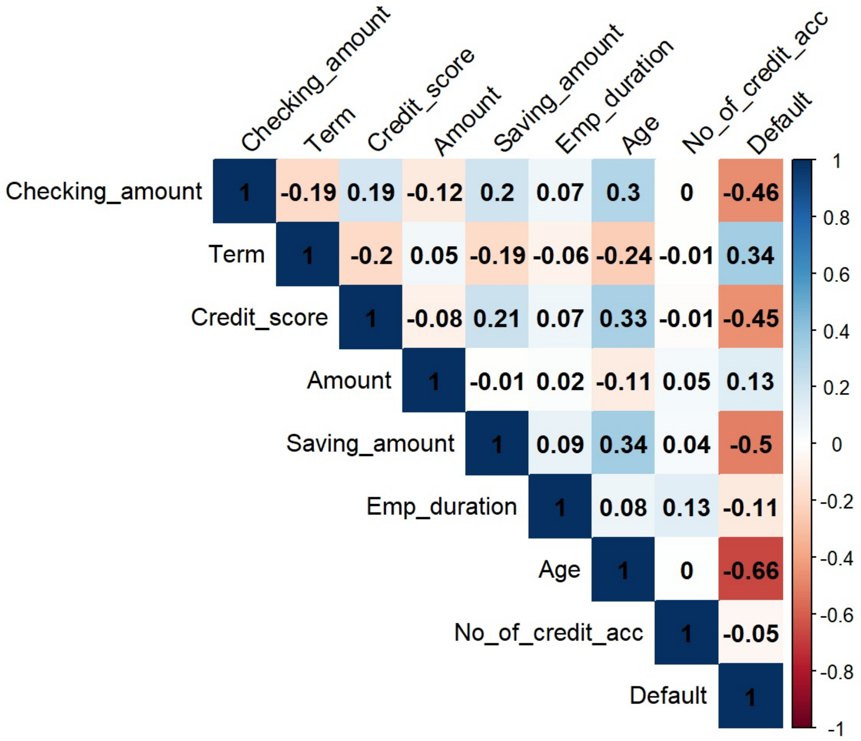
Individuals with lower saving amounts are more likely to default, indicating a lack of financial resilience or emergency funds to cover loan payments.

Shorter employment durations are linked to a higher likelihood of default, suggesting that stability and longevity in employment may contribute to better loan repayment capability.

Younger individuals, or those with lower ages, are more likely to default on their loans, possibly due to limited financial experience or resources.

Correlation Analysis

```
library(corrplot)
# Calculate correlation matrix with numerical variables
correlation_matrix <- cor(loan_default[, c("Checking_amount", "Term", "Credit_score", "Amount", "Saving_amount",
"Emp_duration", "Age", "No_of_credit_acc", "Default")])

# Visualize correlation matrix using a heatmap
corrplot(correlation_matrix, method = "color", type = "upper", addCoef.col = "black", tl.col = "black", tl.srt =
45)
```



	Checking_amount	Term	Credit_score	Amount	Saving_amount	Emp_duration	Age	No_of_credit_acc	Default
Checking_amount	1	-0.19	0.19	-0.12	0.2	0.07	0.3	0	-0.46
Term		1	-0.2	0.05	-0.19	-0.06	-0.24	-0.01	0.34
Credit_score			1	-0.08	0.21	0.07	0.33	-0.01	-0.45
Amount				1	-0.01	0.02	-0.11	0.05	0.13
Saving_amount					1	0.09	0.34	0.04	-0.5
Emp_duration						1	0.08	0.13	-0.11
Age							1	0	-0.66
No_of_credit_acc								1	-0.05
Default									1

Checking Amount to Default: There is a moderate negative correlation of -0.46 between the checking amount and the likelihood of default. This suggests that individuals with lower checking amounts are more likely to default on their loans.

Term to Default: There is a moderate positive correlation of 0.34 between the loan term and the likelihood of default. This indicates that longer loan terms are associated with a higher likelihood of default.

Credit Score to Default: There is a moderate negative correlation of -0.45 between the credit score and the likelihood of default. This suggests that individuals with lower credit scores are more likely to default on their loans.

Amount to Default: There is a weak positive correlation of 0.13 between the loan amount and the likelihood of default. This indicates that higher loan amounts may be slightly associated with a higher likelihood of default.

Saving Amount to Default: There is a moderate negative correlation of -0.50 between the saving amount and the likelihood of default. This suggests that individuals with lower saving amounts are more likely to default on their loans.

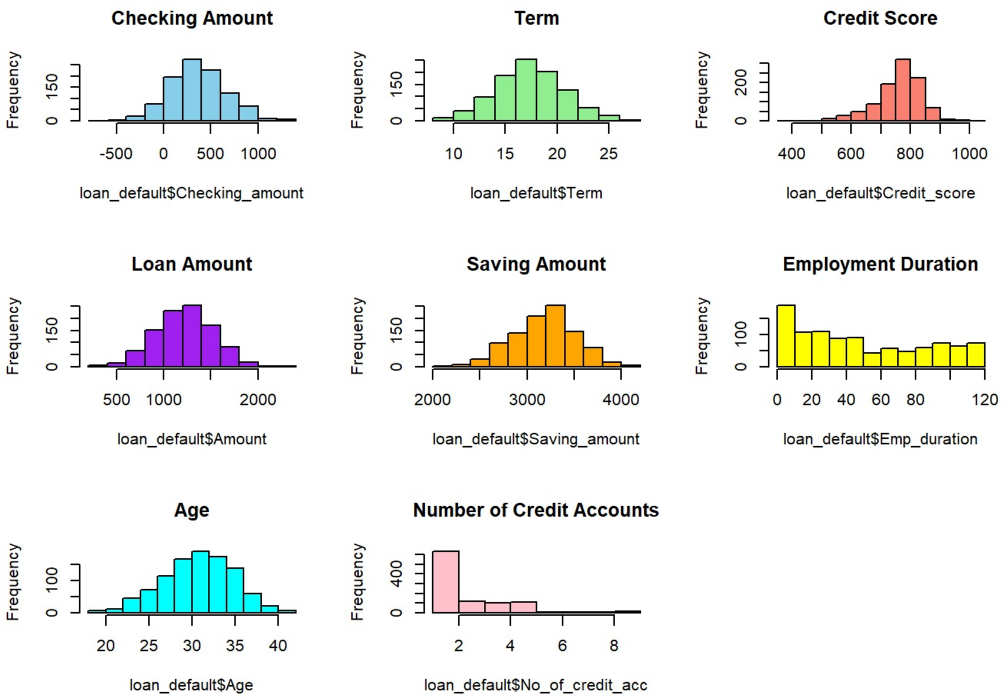
Employment Duration to Default: There is a weak negative correlation of -0.11 between the employment duration and the likelihood of default. This indicates a slight tendency for individuals with shorter employment durations to default more frequently.

Age to Default: There is a strong negative correlation of -0.66 between the age and the likelihood of default. This suggests that younger individuals are more likely to default on their loans compared to older individuals.

Number of Credit Accounts to Default: There is a weak negative correlation of -0.05 between the number of credit accounts and the likelihood of default. This suggests a slight tendency for individuals with fewer credit accounts to default more frequently.

Distribution Analysis

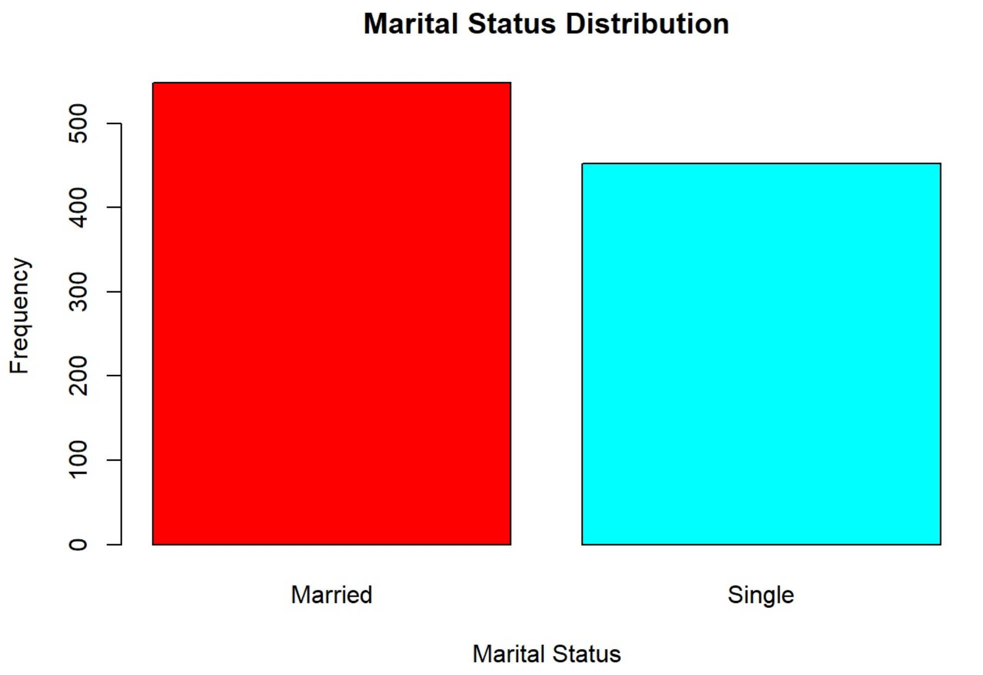
```
# Plot histograms for each numeric variable
par(mfrow = c(3,3))  # Adjust layout for 3 rows and 3 columns of plots
hist(loan_default$Checking_amount, main = "Checking Amount", col = "skyblue")
hist(loan_default$Term, main = "Term", col = "lightgreen")
hist(loan_default$Credit_score, main = "Credit Score", col = "salmon")
hist(loan_default$Amount, main = "Loan Amount", col = "purple")
hist(loan_default$Saving_amount, main = "Saving Amount", col = "orange")
hist(loan_default$Emp_duration, main = "Employment Duration", col = "yellow")
hist(loan_default$Age, main = "Age", col = "cyan")
hist(loan_default$No_of_credit_acc, main = "Number of Credit Accounts", col = "pink")
```



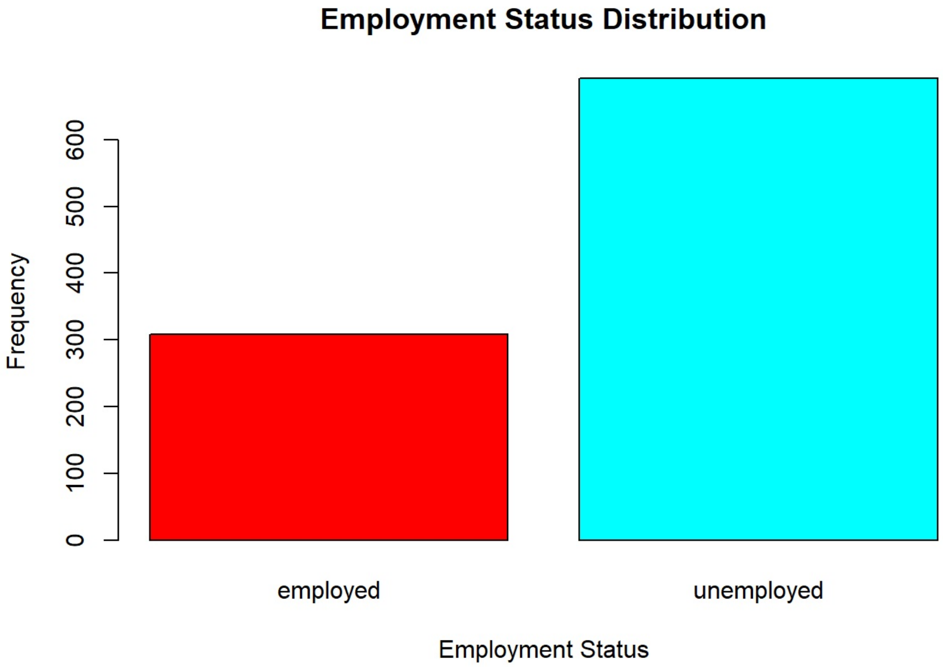
Checking Amount, Term, and Loan Amount: These variables exhibit a normal distribution, indicating that the data is evenly distributed around the mean, forming a bell-shaped curve. Credit Score, Saving Amount, and Age: These variables are slightly skewed to the right. Number of Credit Accounts and Employment Duration: These variables are skewed to the left.

Categorical Variable analysis

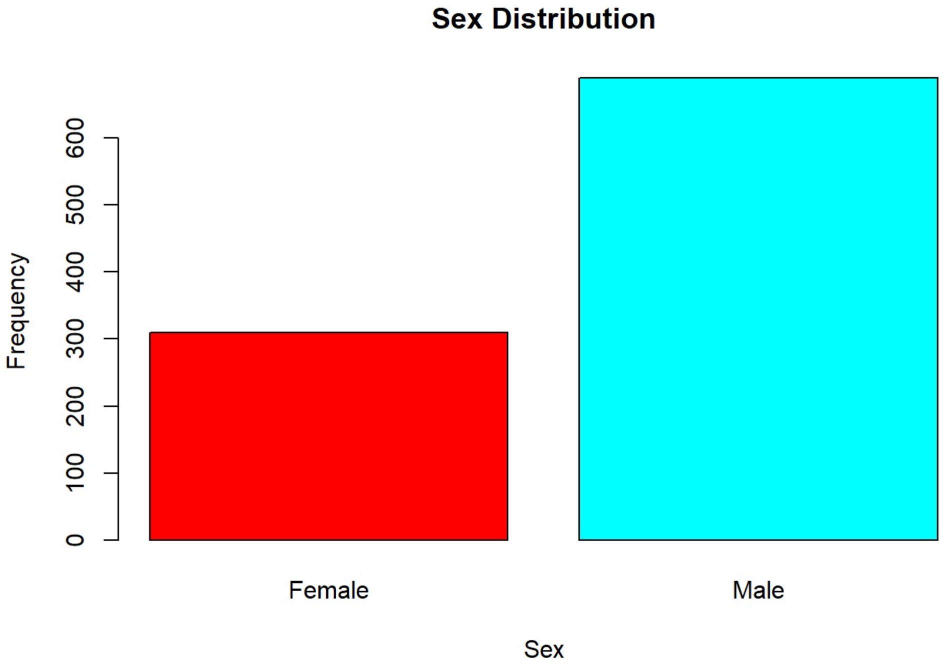
```
# Frequency table for Marital_status
marital_freq <- table(loan_default$Marital_status)
# Bar chart for Marital_status
barplot(marital_freq, main = "Marital Status Distribution", xlab = "Marital Status", ylab = "Frequency", col = ra
inbow(length(marital_freq)))
```



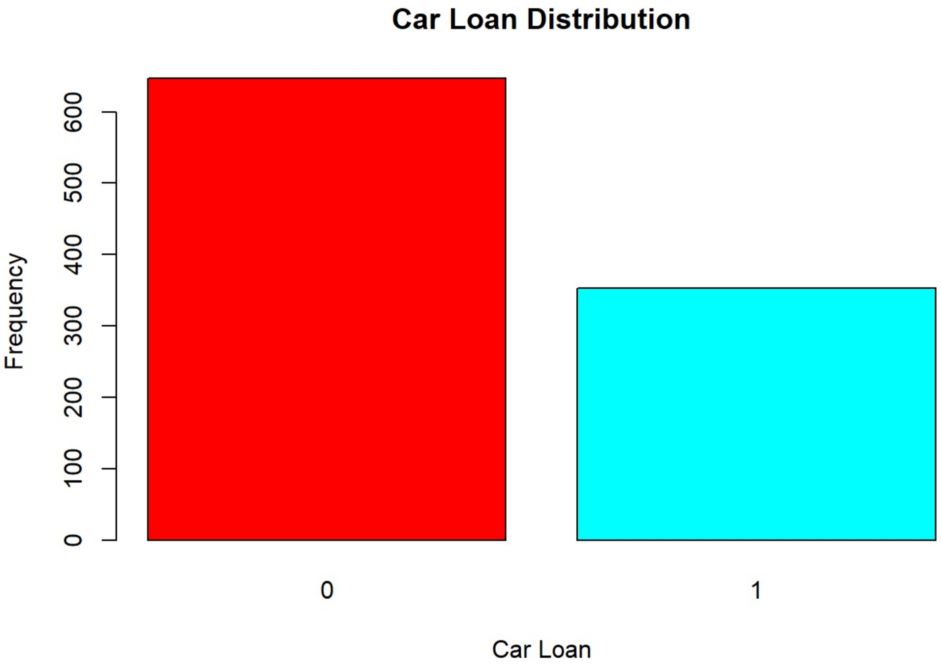
```
# Frequency table for Emp_status
emp_freq <- table(loan_default$Emp_status)
# Bar chart for Emp_status
barplot(emp_freq, main = "Employment Status Distribution", xlab = "Employment Status", ylab = "Frequency", col =
rainbow(length(emp_freq)))
```



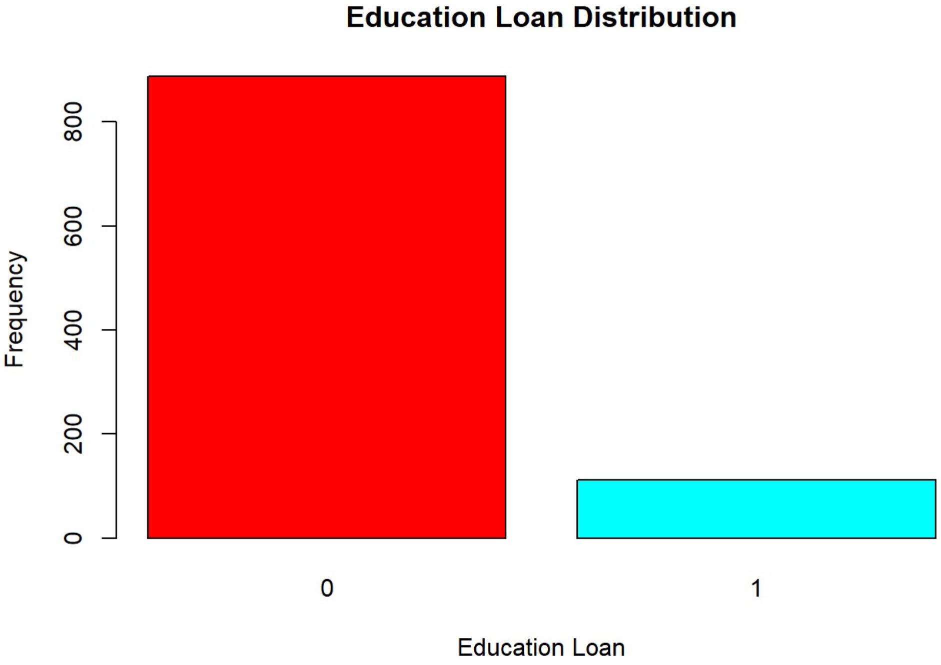
```
# Frequency table for Sex
sex_freq <- table(loan_default$Sex)
# Bar chart for Sex
barplot(sex_freq, main = "Sex Distribution", xlab = "Sex", ylab = "Frequency", col = rainbow(length(sex_freq)))
```



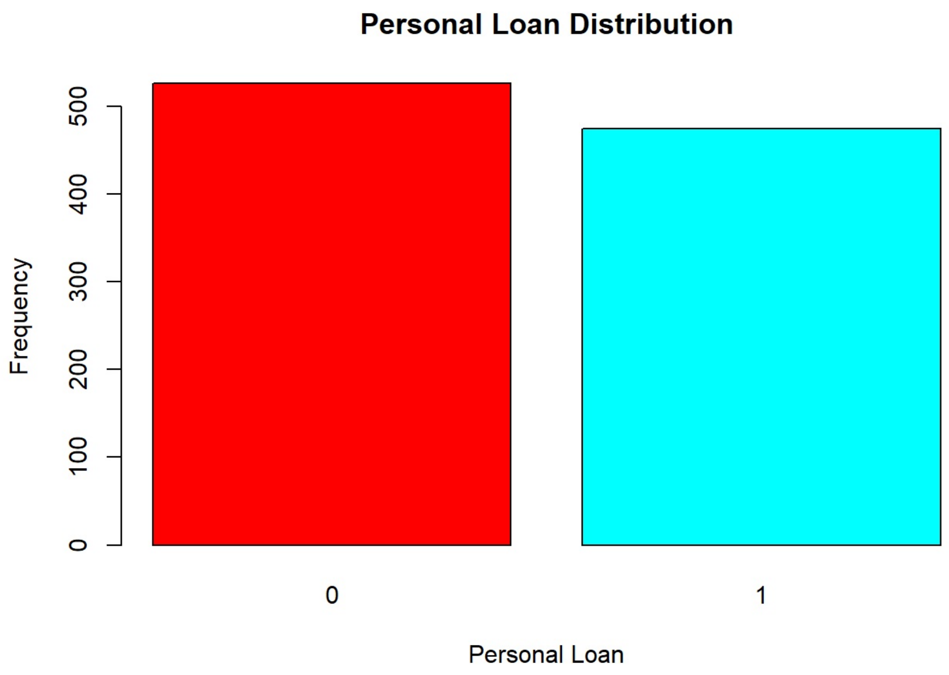
```
# Frequency table for Car_loan
car_freq <- table(loan_default$Car_loan)
# Bar chart for Car_loan
barplot(car_freq, main = "Car Loan Distribution", xlab = "Car Loan", ylab = "Frequency", col = rainbow(length(car
_freq)))
```



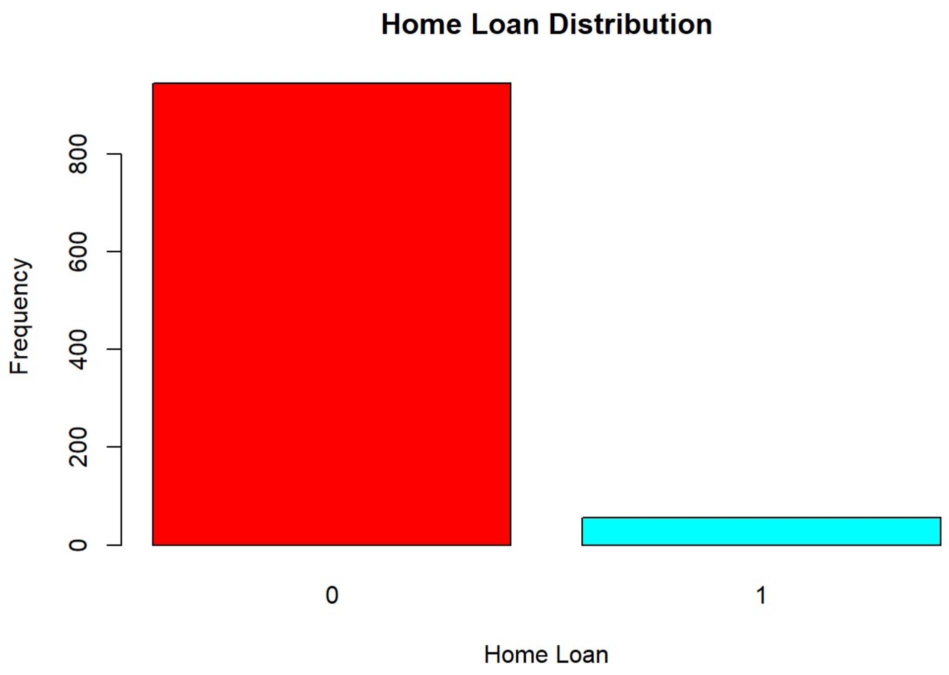
```
# Frequency table for Education Loan
education_freq <- table(loan_default$Education_loan)
# Bar chart for Education_loan
barplot(education_freq, main = "Education Loan Distribution", xlab = "Education Loan", ylab = "Frequency", col =
rainbow(length(education_freq)))
```



```
# Frequency table for Personal Loan
personal_freq <- table(loan_default$Personal_loan)
# Bar chart for Personal Loan
barplot(personal_freq, main = "Personal Loan Distribution", xlab = "Personal Loan", ylab = "Frequency", col = rai
nbow(length(personal_freq)))
```



```
# Frequency table for Home_loan
Home_freq <- table(loan_default$Home_loan)
# Bar chart for Home_loan
barplot(Home_freq, main = "Home Loan Distribution", xlab = "Home Loan", ylab = "Frequency", col = rainbow(length(
Home_freq)))
```



Marital Status: The frequency of individuals who are married is higher than those who are single. This indicates that a larger proportion of individuals in the dataset are married compared to being single.

Employment Status: The frequency of individuals who are unemployed is higher than those who are employed. This suggests that there are more unemployed individuals in the dataset compared to employed individuals.

Gender: The frequency of males is higher than females. This indicates that there are more males in the dataset compared to females.

Binary Variables (Car, Education, Home, Personal Loan): The frequency of individuals without car, education loan, home loan, and personal loan is higher than those with these loans. This suggests that a larger proportion of individuals in the dataset do not have these loans compared to those who do.

MODELLING

Data Processing

```
# Encode categorical variables
# Convert Binary Variable to factor
loan_default <- loan_default %>% mutate_at(7:10, as.factor)

# Convert categorical variable to factor
loan_default <- loan_default %>% mutate_at(c("Sex","Marital_status","Emp_status"), as.factor)

# Convert response variable to factor
loan_default$Default <- as.factor(loan_default$Default)

# Re-level the response variable
loan_default$Default<- relevel(loan_default$Default, ref = "1")

# Convert predictor variables to dummy variables
loan_predictors_dummy <- model.matrix(Default~ ., data = loan_default)
loan_predictors_dummy<- data.frame(loan_predictors_dummy[,-1]) #get rid of intercept
loan_data <- cbind(Default=loan_default$Default, loan_predictors_dummy)

# Split data into testing and training set
library(caret)
set.seed(99) #set random seed
index <- createDataPartition(loan_default$Default, p = 0.8, list = FALSE)
loan_train <- loan_default[index, ]
loan_test <- loan_default[-index, ]


cv_control <- trainControl(method = "cv", number = 5)
```

Summary of Data Processing Steps:

Converting Categorical Variables to Factors:

The categorical variables in the dataset were converted to factors using the as.factor() function in R. This ensures that R treats these variables as categorical rather than continuous. Releveling the Response Variable:

The response variable (usually the outcome variable) was re-leveled to set a specific category as the reference level. This helps in interpreting the model coefficients more intuitively and can aid in comparing different levels of the response variable. Converting Predictor Variables to Dummy Variables:

Categorical predictor variables were converted into dummy variables using one-hot encoding. Each category of a categorical variable was represented by a binary (0 or 1) dummy variable, indicating the presence or absence of that category. Removing the Intercept:

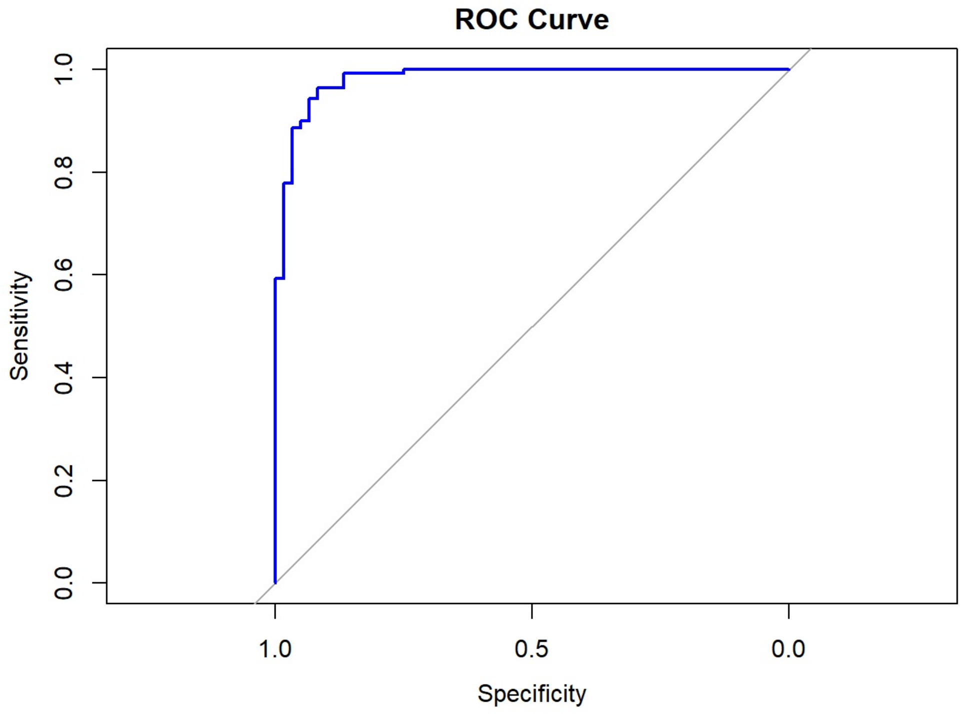
In regression models, the intercept represents the predicted value when all predictor variables are zero. Since we typically don’t want this intercept in the context of dummy variables (to avoid multicollinearity), it was removed before fitting the model. Splitting the Data:

The dataset was split into training and testing sets. This ensures that the model is trained on one subset of the data and evaluated on an independent subset, providing an unbiased estimate of the model’s performance. K-Fold Cross-Validation:

K-fold cross-validation was used to assess the performance of the models. It involves dividing the dataset into k subsets, or “folds”, and training the model k times, each time using a different fold as the validation set and the remaining folds as the training set.

Logistic Regression Modelling

```
library(e1071)
library(glmnet)
library(Matrix)
library(ROCR)
library(pROC)
# Set random seed for reproducibility
set.seed(25)

# Fit LASSO logistic regression model using glmnet
lasso_model <- train(Default ~ .,
                     data = loan_train,
                     method = "glmnet",
                     standardize = TRUE,
                     tuneGrid = expand.grid(alpha = 1, lambda = seq(0.0001, 1, length = 20)),
                     trControl = cv_control)

# Predict probabilities on test data
predictions <- predict(lasso_model, newdata = loan_test, type = "prob")

# Calculate ROC curve using pROC package
roc_curve <- roc(loan_test$Default, predictions[, 2])

# Plot ROC curve
plot(roc_curve, col = "blue", main = "ROC Curve", lwd = 2)
```



```
# Calculate AUC
auc_lasso <- auc(roc_curve)

# Print AUC
auc_lasso
```

```
## Area under the curve: 0.9824
```

I embarked on a journey to predict loan defaults using advanced statistical modeling techniques. My journey began by preparing the data, where I meticulously encoded categorical variables, ensuring they were ready for analysis. I then trained a sophisticated LASSO logistic regression model, renowned for its ability to handle high-dimensional data and select important features.

To ensure my model was finely tuned, I employed cross-validation, a powerful technique that rigorously tests the model’s performance across various parameter combinations. Once trained, I unleashed my model onto the test data, predicting the likelihood of loan defaults with precision.

The moment of truth arrived when I evaluated my model’s performance using the ROC curve, a visual representation of its ability to discriminate between loan defaulters and non-defaulters. The curve revealed an AUC value of 0.9824, signifying exceptional discriminatory power. This means my model excels at distinguishing between those who will default on loans and those who won’t, outperforming random chance by a significant margin.

Decision Tree Modelling

```
# Load required libraries
library(rpart)
library(rpart.plot)

# Set seed for reproducibility
set.seed(99)

# Train decision tree model
decision_tree_model <- train(Default ~ .,
                             data = loan_train,
                             method = "rpart",
                             trControl = cv_control)

# Print the trained decision tree model
print(decision_tree_model)
```

```
## CART
##
## 800 samples
##  15 predictor
##   2 classes: '1', '0'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 640, 640, 640, 640, 640
## Resampling results across tuning parameters:
##
##   cp         Accuracy  Kappa
##   0.0375000  0.88250   0.7062681
##   0.1041667  0.86125   0.6615454
##   0.5208333  0.75625   0.2521879
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was cp = 0.0375.
```

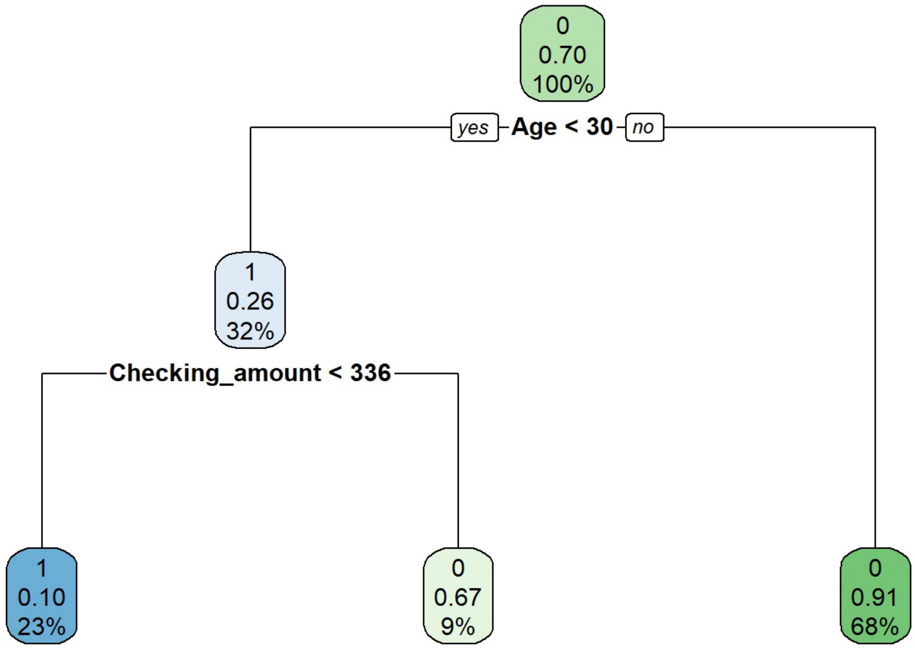
```
# Predict on test data
predictions <- predict(decision_tree_model, newdata = loan_test)

# Confusion matrix
conf_matrix <- confusionMatrix(predictions, loan_test$Default)
print(conf_matrix)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   1   0
##          1  41   4
##          0  19 136
##
##                Accuracy : 0.885
##                  95% CI : (0.8325, 0.9257)
##     No Information Rate : 0.7
##     P-Value [Acc > NIR] : 4.449e-10
##
##                   Kappa : 0.7051
##
##  Mcnemar's Test P-Value : 0.003509
##
##             Sensitivity : 0.6833
##             Specificity : 0.9714
##          Pos Pred Value : 0.9111
##          Neg Pred Value : 0.8774
##              Prevalence : 0.3000
##          Detection Rate : 0.2050
##    Detection Prevalence : 0.2250
##       Balanced Accuracy : 0.8274
##
##        'Positive' Class : 1
##
```

```
# Accuracy
accuracy <- conf_matrix$overall["Accuracy"]
print(paste("Accuracy:", accuracy))
```

```
## [1] "Accuracy: 0.885"
```

```
# Plot decision tree
rpart.plot(decision_tree_model$finalModel)
```



My analysis aimed to develop a decision tree model to predict loan defaults based on demographic and financial factors. Here's a breakdown of our process and the key findings:

Data Preparation and Model Training:

I pre-processed the dataset, encoding categorical variables and splitting it into training and testing sets. Trained a decision tree model using the training data.

Model Evaluation:

The decision tree achieved an accuracy score of 88.5% on the testing data, indicating its effectiveness in classifying loan defaults. The confusion matrix provides further insights into the model's performance: True Positives (Defaulters Correctly Identified): 41 instances True Negatives (Non-Defaulters Correctly Identified): 136 instances Other metrics such as sensitivity, specificity, positive predictive value, and negative predictive value were also computed, demonstrating the model's ability to accurately classify instances from both classes.

Decision Tree Interpretation:

The decision tree offers intuitive decision rules for predicting loan defaults: Individuals aged less than 30 are identified as a higher-risk group for defaults. Among this age group, those with a checking account balance less than 336 are predicted to default, while those with a higher balance are predicted not to default. Individuals aged 30 or older are predicted not to default, regardless of their checking account balance.