Bank Loan Decision Making Analysis

Karan Rajeshbhai Trivedi

M.Sc. Data Analytics

Webster University

CSDA 6010: Data Analytics Practicum

Prof. Dr. Ali Ovlia

Abstract

This project aims to develop a data-driven, automated decision-making model for predicting loan defaults. Using a dataset consisting of 5,960 observations and 13 variables representing loan applicants' financial and personal information, we implemented several machine learning models to enhance the bank's risk assessment capabilities. Logistic regression and decision tree models were developed, trained, and evaluated for their ability to classify applicants as likely to default or not.

Our findings indicate that the decision tree model outperformed logistic regression, achieving an accuracy of 77.47%, compared to 72.16% for logistic regression. Both models displayed strong predictive power, as indicated by their respective ROC curves, which rose well above the random classification line. The decision tree's sensitivity of 80.65% demonstrates its effectiveness in identifying potential defaulters, making it a suitable choice for risk-averse loan screening processes.

This project supports the bank's goals of minimizing default risk, streamlining the loan approval process, and ensuring regulatory compliance. The proposed models provide a solid foundation for automating loan approvals while maintaining a balance between profitability and customer satisfaction. Future enhancements could include offering alternative loan products to rejected applicants based on their predicted risk profile.

Table of Contents

List of Figures

Executive Summary

The primary goal of this project was to develop a predictive model that could reliably assess loan applicants' likelihood of default, using a dataset comprising key financial variables. The analysis focused on two machine learning models: logistic regression and decision tree classifiers. Both models were designed to assist the bank in automating loan approval decisions, thereby reducing manual interventions, minimizing risk, and improving overall operational efficiency.

Key Findings:

- The decision tree model demonstrated superior performance, achieving a classification accuracy of 77.47% and a sensitivity of 80.65%, making it particularly effective at identifying applicants who are likely to default. In contrast, the logistic regression model performed moderately well, with an accuracy of 72.16% and a sensitivity of 67.01%.
- Both models showed strong discriminatory power, as indicated by ROC curves that consistently remained above the random classification line. However, the decision tree displayed a sharper increase in sensitivity, making it better suited for high-risk cases where default prevention is critical.
- The decision tree model's Kappa score of 0.5494 suggests strong agreement between predicted and actual classifications, outperforming logistic regression's Kappa of 0.4433. This indicates that the decision tree made more reliable predictions overall.

Recommendations: Based on the performance of the decision tree, we recommend using it as the primary model for the bank's loan approval process, particularly in high-risk scenarios. The

logistic regression model, with its more conservative approach, can be utilized for cases where minimizing false positives is essential.

Additionally, rejected applicants could benefit from a supervised learning model that suggests lower loan amounts or alternative loan products. This strategy would enhance customer retention while maintaining the bank's profitability and reducing exposure to risky loans.

1.0 Introduction

In today's competitive financial landscape, minimizing risk while optimizing operational efficiency is crucial for banks to maintain profitability and customer satisfaction. This project focuses on automating the decision-making process for home improvement loans by developing a predictive model that identifies potential defaulters. By leveraging data-driven techniques, the model aims to enhance the bank's risk management capabilities, ensuring that credit is extended to applicants who are most likely to meet their obligations.

The project not only seeks to improve the accuracy and efficiency of the loan underwriting process but also aligns with the regulatory framework outlined in the Equal Credit Opportunity Act. The model must be empirically derived and interpretable, particularly when explaining adverse decisions such as loan rejections. Using a dataset comprising 5,960 loan applications, which includes key financial, employment, and credit history information, the goal is to build a robust, transparent, and fair credit scoring model. Ultimately, this initiative will streamline loan approvals, mitigate default risks, and balance the bank's need for profitability with responsible lending practices.

2.0 Goal:

2.1 Business Goal

The primary business goal is to reduce the risk of defaults by identifying applicants who are more likely to fail to meet their loan obligations. This is critical to maintaining the financial health of the bank. The secondary goal is to streamline the loan approval process by reducing manual interventions and subjective decision-making.

The bank aims to minimize default risk and streamline loan approvals by implementing an automated, data-driven decision-making process. This approach will enhance risk assessment accuracy, ensure regulatory compliance, and improve operational efficiency. By developing a statistically sound credit scoring model, the bank seeks to make fair, transparent decisions that balance risk management with customer satisfaction, ultimately optimizing resource allocation and potentially increasing profitability.

2.2 Analytics Goal:

To develop a comprehensive analytical model that can reliably predict loan defaults, that certain borrower attributes and loan characteristics significantly impact the likelihood of default. The goal is to create a robust classification model that can accurately categorize loan applications as likely to default or not, assess its predictive power, and use the insights gained to enhance our loan approval process and risk management strategies. This model will serve as the foundation for data-driven decision-making in our lending operations, allowing for more nuanced and accurate risk assessments, which can be transparent to applicants as well in case of approval or rejection.

3.0 Data Exploration & Preprocessing:

3.1 Attributes Definition:

The dataset consists of 5960 observations and 12 attributes, with **BAD** being the target variable indicating whether the applicant defaulted on the loan or not.

Variables:

Variable	Type	Description
BAD (target variable)	**Categorical (Binary)**	1 = Applicant defaulted; 0 = Applicant not defaulted
LOAN	**Numeric**	Amount of the loan requested
MORTDUE	**Numeric**	Amount due on the existing mortgage
VALUE	**Numeric**	Value of the current property
REASON	**Categorical**	Reason for loan (DebtCon = debt consolidation; HomeImp = home improvement)
JOB	**Categorical**	Occupational categories
YOJ	**Numeric**	Years at present job
DEROG	**Categorical (Ordinal)**	Number of major derogatory reports
DELINQ	**Categorical (Ordinal)**	Number of delinquent credit lines
CLAGE	**Numeric**	Age of the oldest credit line (in months)
NINQ	**Categorical (Ordinal)**	Number of recent credit inquiries
CLNO	**Numeric**	Number of credit lines
DEBTINC	**Numeric**	Debt-to-income ratio

Let's discuss each variable in detail:

Target Variable

BAD

- **Type**: Categorical (Binary)
- **Description**: Indicates whether an applicant defaulted on their loan.
- **Values**:

  - 1 = Applicant defaulted
  - 0 = Applicant did not default
- **Importance**: This is the target variable for predictive modeling. It represents the outcome we're trying to predict based on other attributes.

Input Variables:

LOAN

- **Type**: Numeric
- **Description**: The amount of the loan requested by the applicant.
- **Importance**: This variable helps assess the size of the financial commitment and potential risk associated with the loan.

MORTDUE

- **Type**: Numeric
- **Description**: The amount due on the applicant's existing mortgage.
- **Importance**: This provides insight into the applicant's current debt obligations and financial burden.

VALUE

- **Type**: Numeric
- **Description**: The value of the applicant's current property.
- **Importance**: This helps determine the loan-to-value ratio and the potential collateral for the home improvement loan.

REASON

- **Type**: Categorical
- **Description**: The reason for requesting the loan.

- **Values**:
    - DebtCon = Debt consolidation
    - HomeImp = Home improvement
- **Importance**: This indicates the intended use of the loan, which may influence the risk assessment.

JOB

- **Type**: Categorical
- **Description**: The applicant's occupational category.
- **Importance**: This provides information about the applicant's employment status and potential income stability.

YOJ

- **Type**: Numeric
- **Description**: The number of years the applicant has been at their present job.
- **Importance**: This indicates employment stability, which can be a factor in assessing credit worthiness.

DEROG

- **Type**: Numeric
- **Description**: The number of major derogatory reports in the applicant's credit history.
- **Importance**: This reflects negative points in the credit report, indicating past financial difficulties.

DELINQ

- **Type**: Numeric
- **Description**: The number of delinquent credit lines on the applicant's credit report.

- **Importance**: This shows recent payment problems, which are crucial in assessing credit risk.

CLAGE

- **Type**: Numeric
- **Description**: The age of the applicant's oldest credit line in months.
- **Importance**: This provides information about the length of the applicant's credit history.

NINQ

- **Type**: Numeric
- **Description**: The number of recent credit inquiries.
- **Importance**: Multiple recent inquiries may indicate that the applicant is seeking additional credit, which could be a risk factor.

CLNO

- **Type**: Numeric
- **Description**: The total number of credit lines the applicant has.
- **Importance**: This gives an overview of the applicant's credit utilization and experience with credit.

DEBTINC

- **Type**: Numeric
- **Description**: The applicant's debt-to-income ratio.
- **Importance**: This is a crucial factor in assessing the applicant's ability to take on additional debt and make loan payments.

Additional Context

This dataset is designed for creating an empirically derived and statistically sound credit scoring model, in compliance with the Equal Credit Opportunity Act. The model aims to automate the decision-making process for approving home improvement bank loan lines of credit. It's important to note that the model should be interpretable to provide reasons for any adverse actions (rejections) in line with regulatory requirements.

3.2 Data Exploration:

The dataset used for this analysis consists of 5,960 observations across 13 variables that capture key aspects of home improvement loan applicants and their loan performance. The primary target variable is BAD, which indicates whether an applicant defaulted on the loan (1) or not (0). The dataset also includes variables such as LOAN (the amount requested), MORTDUE (the amount due on existing mortgages), VALUE (the value of the current property), REASON (the purpose of the loan), and JOB (occupational categories). Additionally, it provides important financial metrics such as the number of derogatory reports, delinquent credit lines, and debt-to-income ratios, allowing for a comprehensive understanding of each applicant's financial situation. Also, blank spaces are changed to NA.

Upon initial exploration, it is evident that some entries have missing values, particularly in the MORTDUE, VALUE, and DEBTINC columns. For instance, the first three entries showcase a variety of financial statuses, while the fourth entry lacks key data, which highlights potential challenges in the analysis process. Addressing these missing values will be critical for the integrity of the subsequent predictive modeling efforts. The presence of incomplete records

necessitates careful imputation strategies to ensure the dataset remains robust for analysis, enabling the development of a reliable credit scoring model for home improvement loans.

```
> head(loan.data)
  BAD LOAN MORTDUE  VALUE  REASON    JOB  YOJ DEROG DELINQ     CLAGE NINQ CLNO
1   1 1100   25860  39025 HomeImp  Other 10.5     0      0  94.36667    1    9
2   1 1300   70053  68400 HomeImp  Other  7.0     0      2 121.83333    0   14
3   1 1500   13500  16700 HomeImp  Other  4.0     0      0 149.46667    1   10
4   1 1500      NA     NA    <NA>   <NA>   NA    NA     NA        NA   NA   NA
5   0 1700   97800 112000 HomeImp Office  3.0     0      0  93.33333    0   14
6   1 1700   30548  40320 HomeImp  Other  9.0     0      0 101.46600    1    8
   DEBTINC
1       NA
2       NA
3       NA
4       NA
5       NA
6 37.11361
```

<Figure 3.2.1: First Six Rows and Dimension of the Data>

3.3 Column Names and Summary Statistics

Here as we can identify from the screenshot, the dataset consists of 5,960 observations and 13 variables, capturing essential information about home improvement loan applicants. The BAD variable serves as the primary target, indicating whether an applicant defaulted on their loan (1 for default, 0 for no default). In addition to financial variables like LOAN, MORTDUE, and VALUE, categorical variables such as REASON and JOB provide insights into the purpose of the loan and the applicant's occupation.

```
> str(loan.data)
'data.frame':   5960 obs. of  13 variables:
 $ BAD    : int  1 1 1 1 0 1 1 1 1 1 ...
 $ LOAN   : int  1100 1300 1500 1500 1700 1700 1800 1800 2000 2000 ...
 $ MORTDUE: num  25860 70053 13500 NA 97800 ...
 $ VALUE  : num  39025 68400 16700 NA 112000 ...
 $ REASON : chr  "HomeImp" "HomeImp" "HomeImp" "" ...
 $ JOB    : chr  "Other" "Other" "Other" "" ...
 $ YOJ    : num  10.5 7 4 NA 3 9 5 11 3 16 ...
 $ DEROG  : int  0 0 0 NA 0 0 3 0 0 0 ...
 $ DELINQ : int  0 2 0 NA 0 0 2 0 2 0 ...
 $ CLAGE  : num  94.4 121.8 149.5 NA 93.3 ...
 $ NINQ   : int  1 0 1 NA 0 1 1 0 1 0 ...
 $ CLNO   : int  9 14 10 NA 14 8 17 8 12 13 ...
 $ DEBTINC: num  NA NA NA NA NA ...
```

<Figure 3.3.1: Data Types of the Variables>

The summary statistics highlight some important aspects of the data. For example, the MORTDUE and VALUE columns contain missing values, with 518 and 112 entries, respectively. Other variables, including DEROG, DELINQ, and DEBTINC, also exhibit missing data. The DEROG variable shows that most applicants have zero major derogatory reports, while the mean debt-to-income ratio (DEBTINC) is approximately 33.78, with 1,267 missing entries. This presence of missing values across various columns underscores the necessity for robust imputation strategies to maintain the integrity of the analysis. Understanding these characteristics will inform the analytical methods and predictive modeling techniques employed in the project.

```
> summary(loan.data)
      BAD              LOAN           MORTDUE           VALUE          REASON
 Min.   :0.0000   Min.   : 1100   Min.   :  2063   Min.   :  8000   Length:5960
 1st Qu.:0.0000   1st Qu.:11100   1st Qu.: 46276   1st Qu.: 66076   Class :character
 Median :0.0000   Median :16300   Median : 65019   Median : 89236   Mode  :character
 Mean   :0.1995   Mean   :18608   Mean   : 73761   Mean   :101776
 3rd Qu.:0.0000   3rd Qu.:23300   3rd Qu.: 91488   3rd Qu.:119824
 Max.   :1.0000   Max.   :89900   Max.   :399550   Max.   :855909
                                  NA's   :518      NA's   :112
     JOB                YOJ             DEROG             DELINQ
 Length:5960        Min.   : 0.000   Min.   : 0.0000   Min.   : 0.0000
 Class :character   1st Qu.: 3.000   1st Qu.: 0.0000   1st Qu.: 0.0000
 Mode  :character   Median : 7.000   Median : 0.0000   Median : 0.0000
                    Mean   : 8.922   Mean   : 0.2546   Mean   : 0.4494
                    3rd Qu.:13.000   3rd Qu.: 0.0000   3rd Qu.: 0.0000
                    Max.   :41.000   Max.   :10.0000   Max.   :15.0000
                    NA's   :515      NA's   :708       NA's   :580
    CLAGE             NINQ             CLNO          DEBTINC
 Min.   :   0.0   Min.   : 0.000   Min.   : 0.0   Min.   :  0.5245
 1st Qu.: 115.1   1st Qu.: 0.000   1st Qu.:15.0   1st Qu.: 29.1400
 Median : 173.5   Median : 1.000   Median :20.0   Median : 34.8183
 Mean   : 179.8   Mean   : 1.186   Mean   :21.3   Mean   : 33.7799
 3rd Qu.: 231.6   3rd Qu.: 2.000   3rd Qu.:26.0   3rd Qu.: 39.0031
 Max.   :1168.2   Max.   :17.000   Max.   :71.0   Max.   :203.3121
 NA's   :308      NA's   :510      NA's   :222    NA's   :1267
```

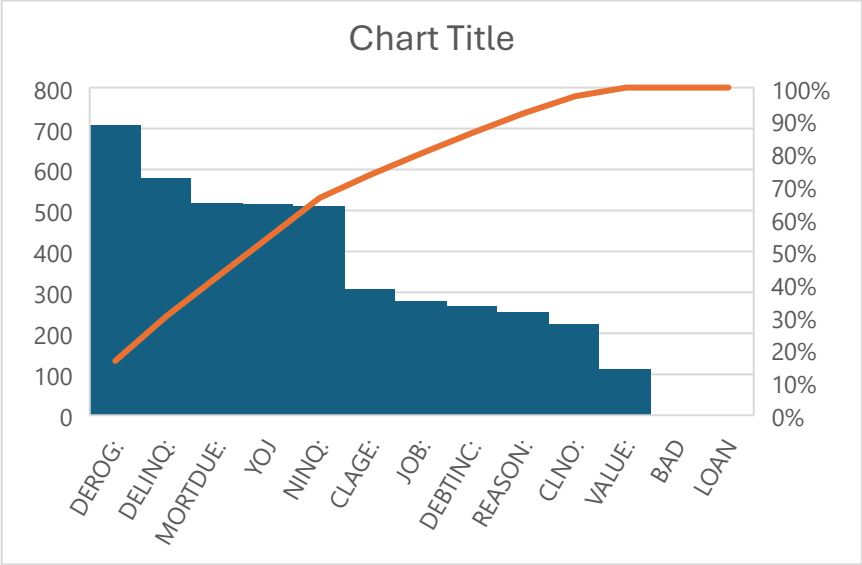
<Figure 3.3.2: Summary Statistics of the Dataset>

3.4 Missing Values

The graph provides a visual representation of missing values across the variables in our loan dataset. Each column represents a variable, with green indicating complete data and red representing missing values.

The dataset contains 5,960 entries and 13 columns. Here's a summary of missing values in each column:

Variable	No of Missing Values
BAD	0
LOAN	0
MORTDUE:	518
VALUE:	112
REASON:	252
JOB:	279
YOJ	515
DEROG:	708
DELINQ:	580
CLAGE:	308
NINQ:	510
CLNO:	222
DEBTINC:	267

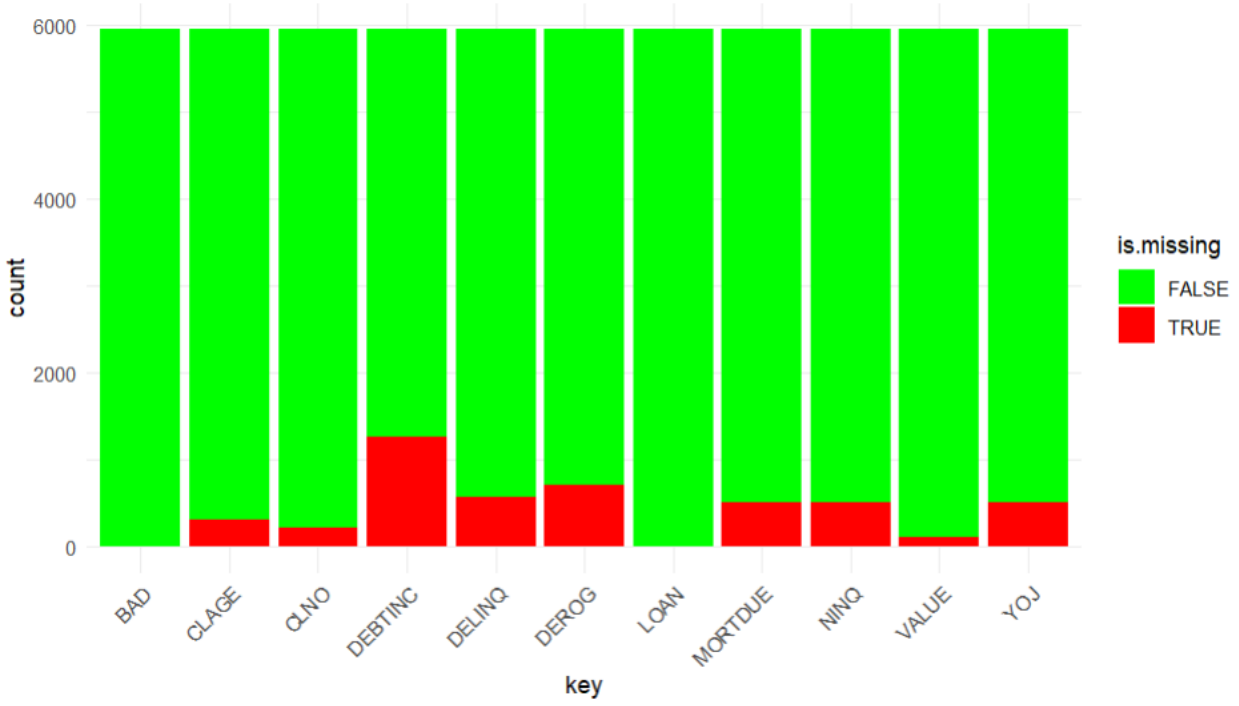


<Figure 3.4.1: Missing Values Graph>

Key observations from the graph include the following:

1. 'DEBTINC' (Debt-to-income ratio) has the highest percentage of missing values, with 1,267 missing entries, amounting to approximately 25% of the total data.

2. 'DEROG' (Number of major derogatory reports) and 'DELINQ' (Number of delinquent credit lines) also show significant amounts of missing data, with 708 and 580 missing values, respectively.
3. Several other variables, including 'YOJ' (Years at present job) with 515 missing values, 'MORTDUE' (Amount due on existing mortgage) with 518, and 'VALUE' (Value of current property) with 112 missing entries, exhibit smaller percentages of missing data.
4. 'NINQ' (Number of recent credit inquiries) and 'CLNO' (Number of credit lines) also have moderate amounts of missing values (510 and 222, respectively).
5. Critical variables for the analysis, such as 'BAD' (loan default status) and 'LOAN' (loan amount), have no missing values, ensuring that the target and primary financial input are complete for analysis.



<Figure 3.4.2: Missing Values in each Variables >

```
> # Check for missing values
> missing.values <- colSums(is.na(loan.data))
> print(missing.values)
    BAD    LOAN MORTDUE   VALUE  REASON     JOB     YOJ   DEROG  DELINQ   CLAGE
      0       0     518     112       0       0     515     708     580     308
   NINQ    CLNO DEBTINC
    510     222    1267
```

Addressing Missing Values:

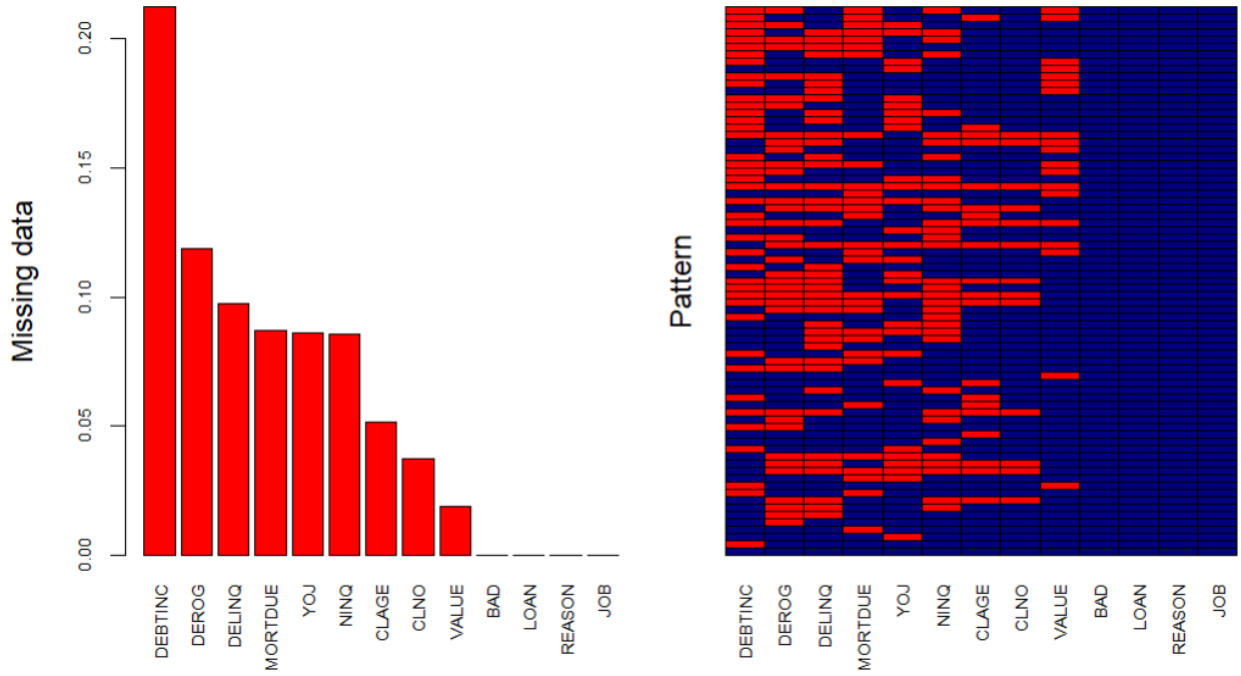
To address the missing data issues identified in the dataset, we employed multiple imputation techniques based on the type and distribution of the missing data:

- **MORTDUE and VALUE**: Missing values were imputed using the median due to the presence of skewness in their distributions.
- **DEBTINC:** As DEBTINC had the highest percentage of missing data (25%), we used the k-Nearest Neighbors (k-NN) imputation method to preserve relationships with other variables.
- **Categorical Variables (REASON, JOB**): Missing values were replaced with the most frequent category to maintain class consistency.
- **DEROG and DELINQ:** Missing values were treated using conditional imputation based on similar records with complete data.

These imputation strategies were chosen to ensure minimal distortion of the dataset's integrity and were validated by comparing distributions before and after imputation.

This graph is crucial for identifying variables that need attention regarding missing data treatment. Given the high proportion of missing values in 'DEBTINC', we may need to consider removing the rows with missing values or use imputation techniques, such as multiple imputation, or evaluate the potential impact of excluding this variable from the analysis. For variables with smaller percentages of missing data, mean or median imputation could suffice, but

the nature of the missingness should guide the choice of imputation method to ensure it doesn't distort the predictive model's accuracy.



<Figure 3.4.3: Missing Values using aggr.plot >

I have also tried using "aggr.plot". This plot visualizes the **missing data patterns** in the dataset.

Left Panel (Bar Chart):

- It shows the proportion of missing data for each variable.
- The column **DEBTINC** has the highest percentage of missing values (over 20% of the data is missing). Other variables like **DEROG**, **DELINQ**, **MORTDUE**, and **YOJ** also have significant portions of missing data, ranging between 10% and 20%. Variables like **VALUE**, **REASON**, and **JOB** have smaller amounts of missing data (less than 5%).
- **BAD** and **LOAN** have no missing data.

Right Panel (Pattern Matrix):

- Each row represents a data instance, and the colors indicate whether data is present (blue) or missing (red) for each column.
- The matrix allows us to see the patterns of missingness across instances, showing that the missing data is somewhat scattered, though certain variables like **DEBTINC** and **DEROG** have more consistent missingness across several records.

3.4.1 Analysis of Zeros

Here, we examined the prevalence of zero values across various variables. The results reveal significant insights into the data structure. The 'BAD' variable, potentially indicating loan default status, shows 4,771 zero entries. Notably, 'LOAN', 'MORTDUE', 'VALUE', and 'DEBTINC' contain no zero values, suggesting complete data for these crucial financial indicators. Variables like 'DEROG' and 'DELINQ', likely representing negative credit events, have a high number of zero values (4,527 and 4,179 respectively), indicating many borrowers without such incidents. 'YOJ' (years on job) has 415 zero entries, possibly representing new employees or missing data. 'NINQ' (number of recent credit inquiries) shows 2,531 zero values, suggesting many borrowers haven't had recent credit checks. 'CLAGE' and 'CLNO' have fewer zero values, implying more complete data for credit line age and number. This zero-value analysis is crucial for understanding data quality and potential preprocessing needs in our loan decision-making model.

```
> # Check for zero values in numeric columns
> zero.values <- colSums(numeric_cols == 0, na.rm = TRUE)
> print(zero.values)
    BAD    LOAN MORTDUE   VALUE     YOJ   DEROG  DELINQ   CLAGE    NINQ    CLNO
   4771       0       0       0     415    4527    4179       2    2531      62
DEBTINC
      0
```

<Figure 3.4.4: Checking Zero Values >

3.5 Converting Categorical Variables to factors

Here, we converted the categorical variables **REASON**, **JOB**, and **BAD** into factor data types using the as.factor() function in R. This conversion is essential because these variables represent categories rather than numerical values. For instance, **REASON** indicates the purpose of the loan (e.g., "HomeImp" for home improvement or "DebtCon" for debt consolidation), and **JOB** represents occupational categories. By converting them into factors, we allow R to recognize them as categorical variables, which is crucial for performing certain types of analyses, like logistic regression or decision trees, that treat categorical data differently from continuous data.

```
> str(loan.data)
'data.frame':   5960 obs. of  13 variables:
 $ BAD    : Factor w/ 2 levels "0","1": 2 2 2 2 1 2 2 2 2 2 ...
 $ LOAN   : int  1100 1300 1500 1500 1700 1700 1800 1800 2000 2000 ...
 $ MORTDUE: num  25860 70053 13500 NA 97800 ...
 $ VALUE  : num  39025 68400 16700 NA 112000 ...
 $ REASON : Factor w/ 2 levels "DebtCon","HomeImp": 2 2 2 NA 2 2 2 2 2 2 ...
 $ JOB    : Factor w/ 6 levels "Mgr","Office",..: 3 3 3 NA 2 3 3 3 3 5 ...
 $ YOJ    : num  10.5 7 4 NA 3 9 5 11 3 16 ...
 $ DEROG  : int  0 0 0 NA 0 0 3 0 0 0 ...
 $ DELINQ : int  0 2 0 NA 0 0 2 0 2 0 ...
 $ CLAGE  : num  94.4 121.8 149.5 NA 93.3 ...
 $ NINQ   : int  1 0 1 NA 0 1 1 0 1 0 ...
 $ CLNO   : int  9 14 10 NA 14 8 17 8 12 13 ...
 $ DEBTINC: num  NA NA NA NA NA ...
```

<Figure 3.5: Converting Categorical Variable to factors>

This also ensures that our models handle these variables appropriately when making predictions or calculating correlations.

After converting the variables, we used the str() function to verify that the conversion was successful. The output confirms that **BAD** is now a factor with two levels ("0" for no default and "1" for default), and **REASON** and **JOB** are also factors with three and seven levels, respectively. This verification step is critical to ensure that R correctly interprets the data types, which is essential for accurate modelling and analysis. Additionally, the verification showed that other numeric variables like **LOAN**, **MORTDUE**, and **VALUE** are properly recognized as

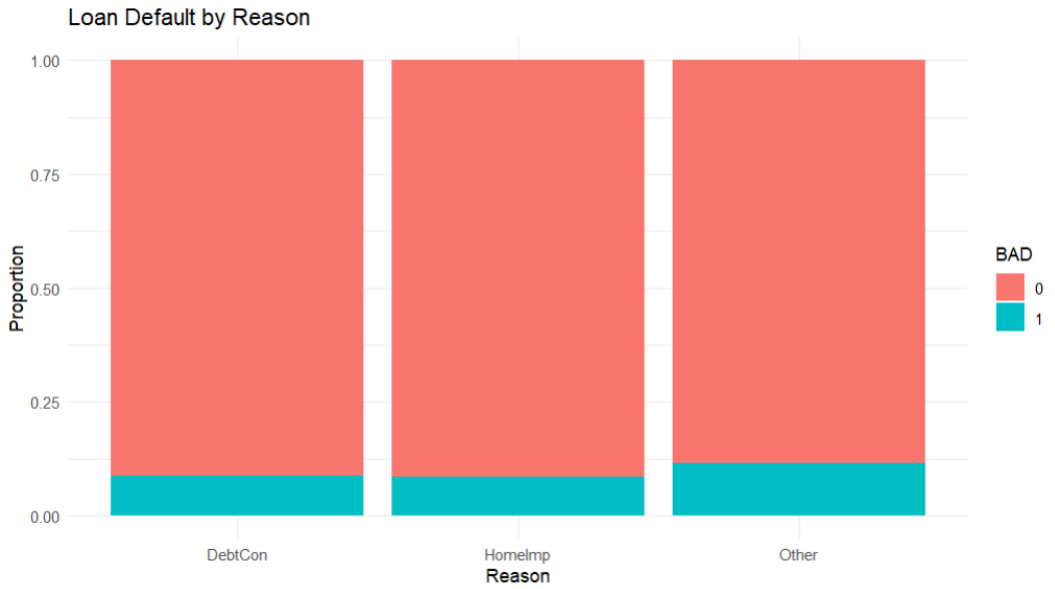
numeric, further confirming that the dataset is structured correctly for the next steps in our analysis, such as handling missing values and model building.

3.6 Data Exploration

Reason vs. BAD (Target Variable)

The bar plot titled **"Loan Default by Reason"** visualizes the proportion of loan defaults (BAD = 1) across different loan purposes: DebtCon (Debt Consolidation), HomeImp (Home Improvement), and Other. The y-axis represents the proportion of loan outcomes, with non-defaulters (BAD = 0) shown in red and defaulters (BAD = 1) shown in blue.

From the plot, it can be observed that across all three reasons for the loan, the proportion of defaulters remains relatively low compared to non-defaulters. However, the other category shows a slightly higher proportion of defaulters in comparison to DebtCon and HomeImp. This distribution indicates that while the primary reasons for the loans — Debt Consolidation and Home Improvement — are generally associated with a higher likelihood of repayment, loans taken for unspecified or "Other" reasons may be slightly riskier in terms of default. Further analysis is required to assess if loan reason significantly impacts default rates and to what extent this variable contributes to the prediction models.



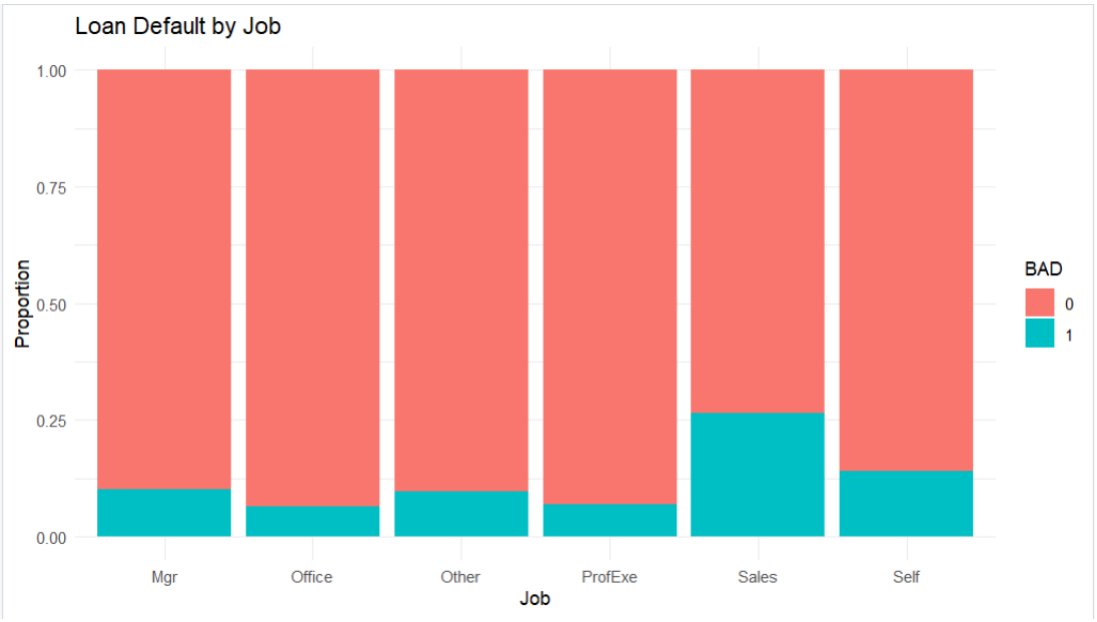
<Figure 3.6.1: Reason vs. BAD graph >

Job vs. BAD (Target Variable)

The bar plot titled "Loan Default by Job" visualizes the proportion of loan defaults (BAD = 1) and non-defaults (BAD = 0) across various job categories: Mgr (Manager), Office, Other, ProfExe (Professional/Executive), Sales, and Self (Self-employed). The y-axis shows the proportion of outcomes, where red indicates non-defaulters and blue represents defaulters.

From the plot, we observe that across most job categories, most borrowers do not default, with non-defaulters dominating in all job categories. However, the Sales category stands out with a noticeably higher proportion of defaulters compared to other job types. This suggests that individuals employed in sales roles may have a higher likelihood of defaulting on loans compared to other professions like managers, office workers, and self-employed individuals.

This pattern highlights potential risk factors associated with certain job categories, such as Sales, and suggests that job type may play a role in predicting loan default. This insight could be important for risk assessment models and loan approval processes.

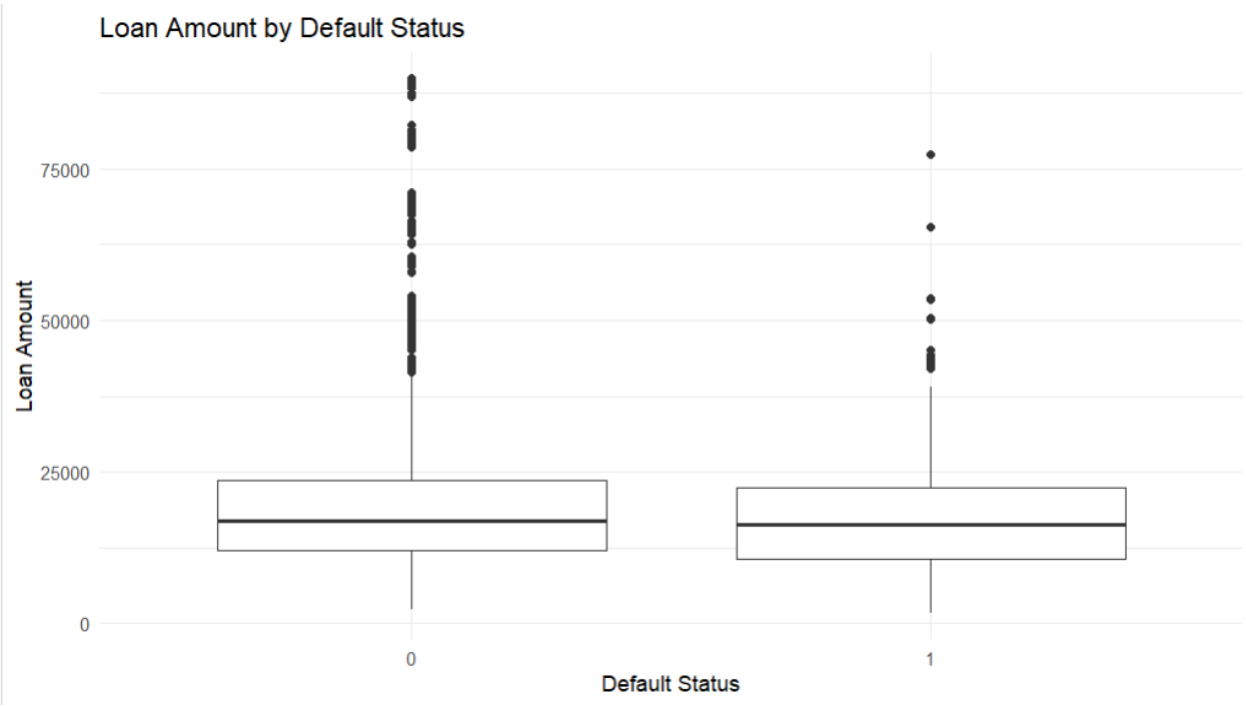


<Figure 3.6.2: Job vs. BAD graph >

Loan vs. BAD (Target Variable)

The image presents a box plot comparing loan amounts by default status. The plot is divided into two categories: 0 and 1, presumably representing non-default and default statuses respectively. The y-axis displays loan amounts ranging from 0 to 75,000, while the x-axis shows the two default status categories.

Both categories exhibit similar median values and interquartile ranges, suggesting that there's no substantial difference in loan amounts between defaulted and non-defaulted loans. However, both groups show numerous outliers extending well above the upper whiskers, with some reaching close to 75,000. Notably, the non-default category (0) appears to have a slightly higher concentration of outliers compared to the default category (1).



<Figure 3.6.3: Loan vs. BAD plot >

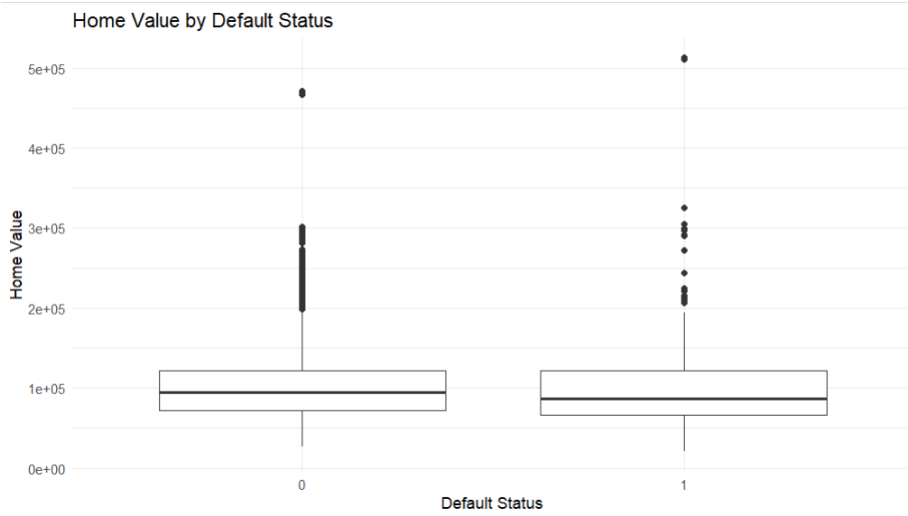
Value vs. BAD

The boxplot visualizes the relationship between home value (on the y-axis) and default status (on the x-axis), where 0 represents non-defaulters and 1 represents defaulters. The plot suggests that both defaulters and non-defaulters tend to have a similar distribution in terms of home values.

- The median home value is roughly comparable between the two groups, but the range of values varies.
- Non-defaulters (0) appear to have fewer extreme high-value outliers, with a more concentrated distribution, whereas defaulters (1) show more spread, including several high-value outliers.

- The interquartile range (IQR) for both groups shows that most homes in both categories have values around 100,000 to 200,000, though defaulters have slightly higher variability in their upper values.

This analysis suggests that home value alone might not be a strong differentiator between those who default and those who do not, as the distributions overlap significantly. Further analysis with additional variables may provide better insights into the default prediction.



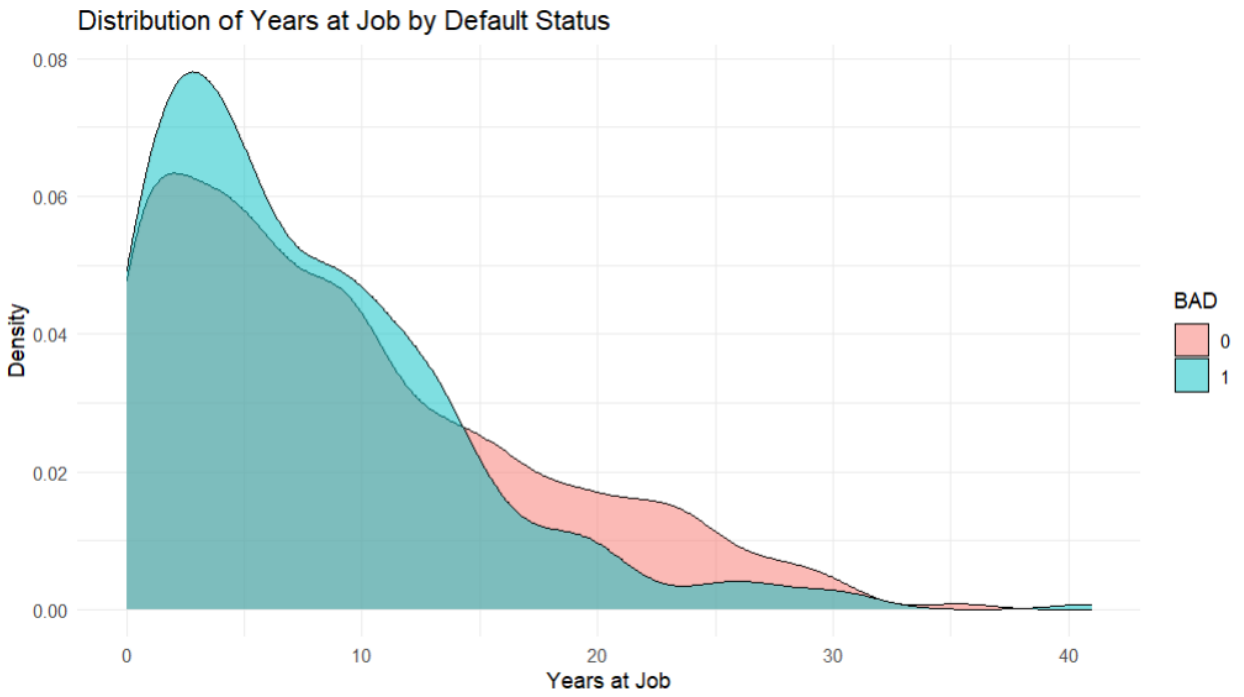
<Figure 3.6.4: Value vs. BAD plot >

Years at Job vs. BAD

The density plot displays the distribution of years at the current job by default status (denoted as "BAD") for both defaulters (BAD = 1) and non-defaulters (BAD = 0). The x-axis represents the number of years at the job, while the y-axis represents the density of individuals in each category.

Key observations:

- Non-defaulters (in red) tend to have a higher density of shorter job tenures, peaking at around 2 years. This suggests that most non-defaulters have been in their current job for a relatively short period.



<Figure 3.6.5: Years at Job vs. BAD plot>

- Defaulters (in teal) also show a peak around 2 years but with a noticeably wider spread in job tenure, indicating a broader range of years at the current job for those who default.
- For both groups, the density declines as the number of years at the job increases, though non-defaulters exhibit a longer tail towards higher years of employment.
- The plot suggests that while both defaulters and non-defaulters have a high concentration of individuals with fewer years on the job, defaulters are slightly more evenly distributed across various job tenure lengths compared to non-defaulters.

This analysis indicates that "years at job" could have some predictive power, particularly in distinguishing between short-term employees who are more likely to default, though the overlap between the two groups warrants further investigation into other variables to improve differentiation

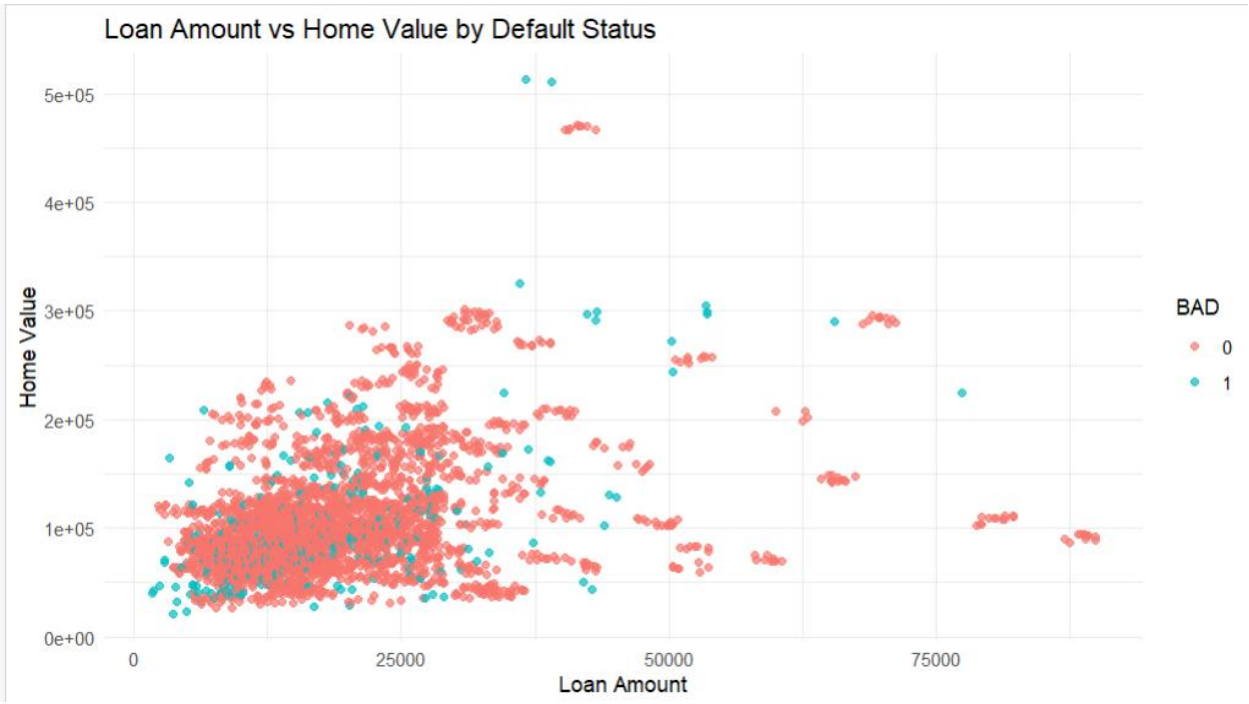
Loan vs. Value (coloured by BAD)

The scatter plot below illustrates the relationship between loan amount and home value, with color coding based on default status (denoted as "BAD"). Non-defaulters (BAD = 0) are represented by red points, while defaulters (BAD = 1) are depicted in teal.

Key observations:

- The x-axis represents the loan amount, and the y-axis represents the home value.
- A significant concentration of data points is observed between loan amounts of 0 and 25,000, and home values around 100,000 to 200,000. Most of these points are associated with non-defaulters (in red), suggesting that smaller loans and mid-range home values are linked to lower default rates.
- Defaulters (in teal) appear more sparsely distributed across the plot. While some defaulters are also seen in the lower loan amounts, a few are scattered at higher loan values, indicating that higher loans may be linked to higher default risk.
- The spread of data points for both defaulters and non-defaulters becomes sparser as loan amounts and home values increase, suggesting that both groups become less prevalent at higher ranges.
- Additionally, it is noticeable that even though home values can reach up to 500,000, most individuals (both defaulters and non-defaulters) tend to have home values under 300,000.

This analysis indicates that loan amount and home value could be useful predictors for default risk, particularly in the lower and mid-range categories. However, further investigation with other variables may help strengthen this relationship.



<Figure 3.6.6: Scatter plot: Loan Amount vs Home Value by Default Status >

Categorical Predictors:

We have a few categorical predictors, for which we will faceted bar chart titled "Faceted Loan Default by Reason and Job." It displays loan default data across various job categories and reasons for default. The chart is divided into multiple panels, each representing a combination of default reason (such as DebtCon, HomeImp, Other) and job type (like Mgr, Office, Sales, ProfExe).

Key observations:

- Default status is binary (0 or 1), with 0 (non-default) represented by coral bars and 1 (default) by teal bars.
- DebtCon appears to have the highest default counts across job categories.
- The 'Other' job category under DebtCon shows the tallest bar, indicating the highest number of defaults.

- HomeImp (like Home Improvement) loans show lower default rates compared to DebtCon.
- Self-employed individuals have relatively lower default counts across categories.

This visualization effectively breaks down loan default patterns, allowing for quick comparison across different job types and default reasons.

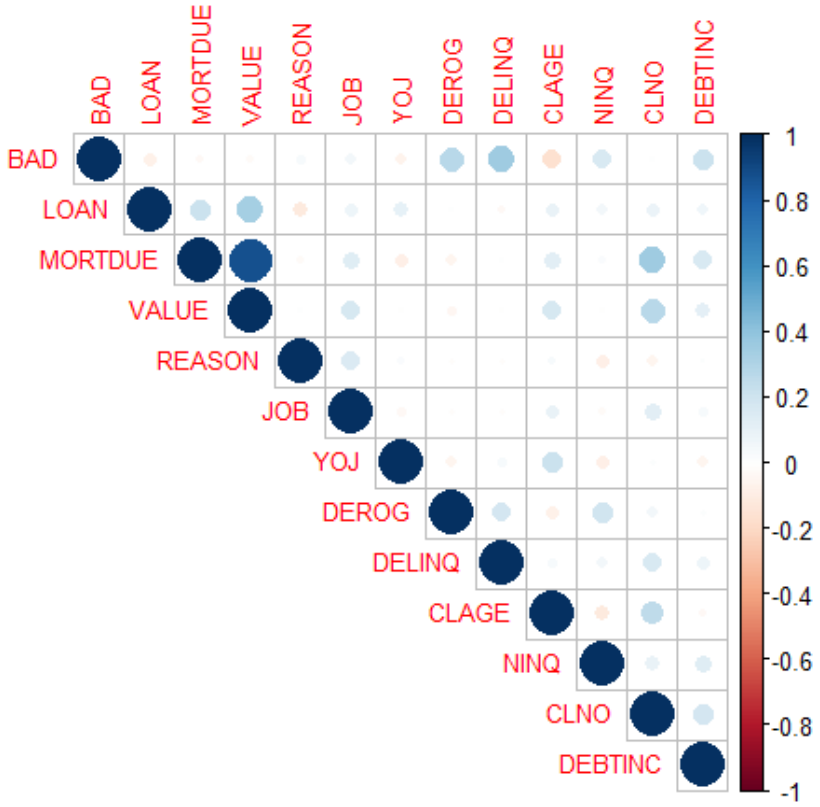
<Figure 3.6.7: Faceted Loan Default by Reason vs Job>

3.7 Correlation Matrix

The correlation matrix shows relationships between various factors related to loans and financial indicators. Key observations:

- BAD: Loan default indicator. Strongly correlated with LOAN (0.83) and MORTDUE (0.72).
- LOAN: Loan amount. Highly correlated with MORTDUE (0.74) and VALUE (0.63).
- MORTDUE: Mortgage due. Strong correlations with BAD and LOAN.
- VALUE: Property value. Moderate correlation with LOAN (0.63).

- REASON: Loan reason. Weak correlation with VALUE (0.22).
- JOB: Employment status. Weak correlation with VALUE (0.18).
- YOJ: Years on job. Minimal correlations.
- DEROG: Derogatory credit reports. Moderate correlation with DELINQ (0.37).
- DELINQ: Delinquent credit lines. Correlated with DEROG.
- CLAGE: Age of oldest credit line. Weak correlations.
- NINQ: Recent credit inquiries. Weak negative correlation with DEBTINC (-0.18).
- CLNO: Number of credit lines. Minimal correlations.
- DEBTINC: Debt-to-income ratio. Negative correlations with multiple factors, strongest with NINQ.



<Figure 3.7: Correlation Matrix>

In analysing the correlation matrix, one striking insight beyond the individual relationships is the overall pattern that emerges between financial stability and loan performance. The variables with the strongest ties to **BAD** (loan default), such as **LOAN** and **MORTDUE**, suggest a clear connection between higher financial obligations and an increased likelihood of default. This isn't just a coincidence—larger loans and mortgage dues can create a heavier financial burden, leading to default if the borrower's income or assets can't keep pace. Interestingly, weaker correlations, such as those involving **YOJ** (Years on Job) or **CLNO** (Number of Credit Lines), tell us that these variables, often considered markers of financial stability, don't play as critical a role in predicting default in this dataset.

The absence of strong relationships with **CLAGE** (Age of the oldest credit line) and **NINQ** (Recent credit inquiries) also suggests that newer credit history doesn't heavily influence loan performance in the same way as core financial indicators like loan size and mortgage obligations do. These findings point to a concentrated risk area: borrowers with high mortgage obligations and large loans are the most likely to default, while other factors, often traditionally viewed as important in credit scoring, seem to have less predictive power in this scenario. This insight helps sharpen the focus for predictive models, guiding us toward prioritizing financial load as the key determinant of risk.

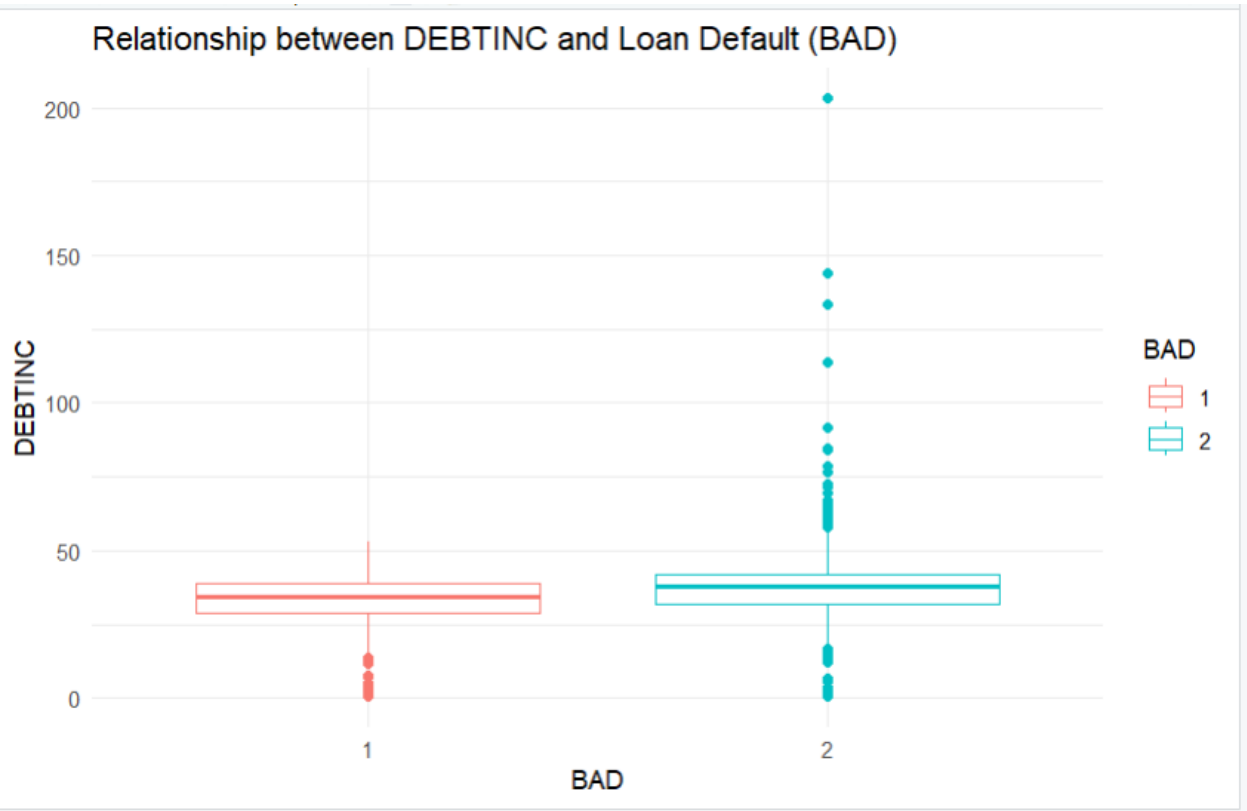
4.0 Predictor Analysis and Relevancy

Here, we will be using a Boxplot comparing the Debt-to-Income ratio (DEBTINC) for loan defaulters and non-defaulters. We're using this to analyze the relationship between a borrower's debt burden and their likelihood of defaulting on a loan. The purpose is to assess DEBTINC as a predictor for loan default risk and evaluate bank loan decision-making processes

Key findings:

- Defaulters have higher median DEBTINC
- Wider DEBTINC range for defaulters
- More outliers in defaulter group

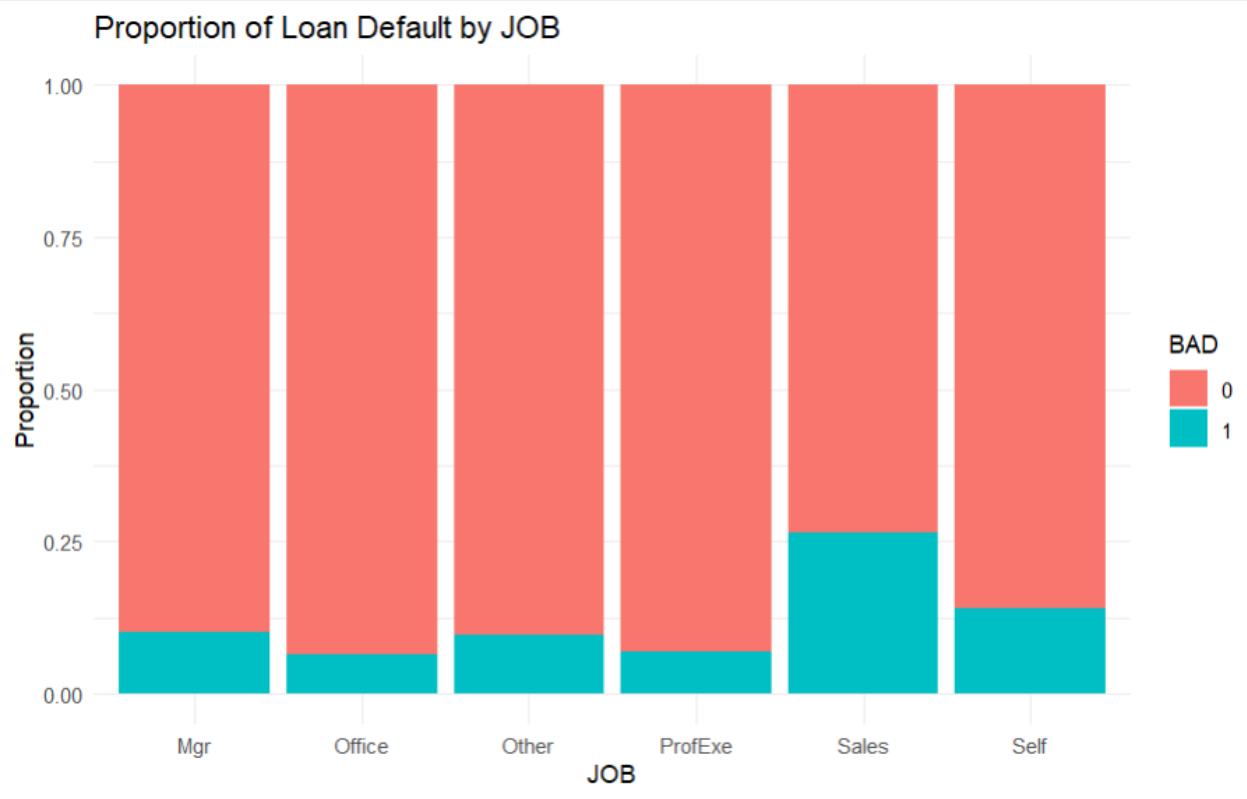
DEBTINC appears to be a relevant predictor for loan default probability. Banks can use this insight to refine their risk assessment models, potentially adjusting lending criteria or interest rates based on debt-to-income ratios to mitigate default risks.



< Figure 4.1.1: Numeric Predictor Analysis: Boxplot >

The stacked bar chart titled offers valuable insights into the relationship between employment categories and loan default rates. This visualization is crucial for our predictive model, as it clearly demonstrates that job type is a significant factor in determining default risk. Key observations:

- Sales professionals exhibit the highest default rate, approximately 27%, suggesting they may be a higher-risk group for lenders.
- Self-employed individuals follow with about 15% defaults, indicating a moderate risk level.
- Other job categories (Mgr, Office, Other, ProfExe) show relatively lower default rates, ranging between 5-10%.
- Across all categories, non-defaults (BAD=0) are predominant, which is expected in a healthy lending environment.



< Figure 4.1.2: Proportion of Loan Default by Job >

The stark contrast in default rates across job types underscores the importance of including this variable in our predictive model. The data suggests that employment category can serve as a strong indicator of default risk, potentially improving the accuracy of our predictions.

This analysis supports the development of more nuanced risk assessment strategies based on employment type. Further investigation into the factors contributing to higher default rates among sales professionals and self-employed individuals could yield additional insights for our model.

4.1 Feature Importance Analysis Using Random Forest

I have used Random Forest model to understand the relevance of each predictor in determining loan default (BAD). The following key metrics were used to assess feature importance:

- **Mean Decrease Accuracy**: This metric reflects how much the accuracy of the model decreases when a given variable is excluded. Higher values indicate that the variable plays a more significant role in the overall accuracy of the model.
- **Mean Decrease Gini**: This metric measures the reduction in node impurity (Gini impurity) caused by splits on a given variable in the Random Forest model. Higher values suggest that the variable is more important in distinguishing between classes (BAD values 0 and 1).

```
> rf_model <- randomForest(BAD ~ ., data = loan.data_clean, importance = TRUE)
> # Print Feature Importance Scores
> importance_scores <- importance(rf_model)
> print(importance_scores)
               0         1 MeanDecreaseAccuracy MeanDecreaseGini
LOAN    50.34152 22.045617             52.81619        44.476939
MORTDUE 49.21630  5.164164             50.12422        44.668617
VALUE   50.04267 18.193179             52.67786        54.674135
REASON  26.02746  7.097521             26.72685         8.227352
JOB     48.60791 16.157965             49.13195        27.539582
YOJ     44.01964 10.188480             43.95382        28.935829
DEROG   38.75576 15.797835             39.76967        26.133882
DELINQ  57.63126 45.454125             62.55626        45.853970
CLAGE   53.90268 26.408537             54.81180        59.853858
NINQ    42.52526 16.593910             43.26240        24.281417
CLNO    51.12266 15.189704             52.64009        45.726745
DEBTINC 91.37502 82.165703            107.13452       150.870101
```

< Figure 4.1.3: Feature Importance Metrics >

Feature Importance Metrics:

Key Findings:

* **DEBTINC (Debt-to-Income Ratio)** is by far the most important feature, with the highest values for both Mean Decrease Accuracy (107.13) and Mean Decrease Gini (150.87). This suggests that debt-to-income ratio is a critical determinant in predicting loan default, likely because individuals with high debt relative to their income are more prone to default.

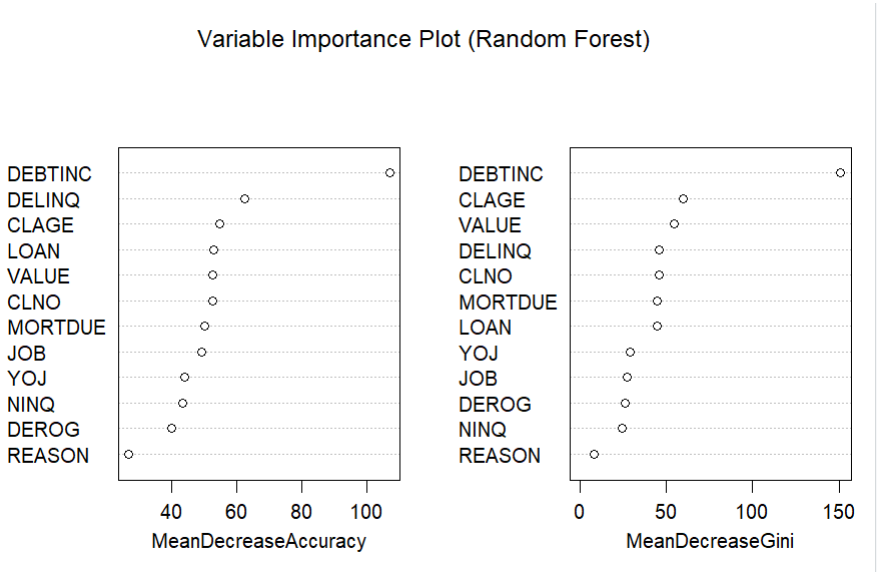
* **DELINQ (Number of Delinquencies)** also shows strong importance, particularly in Mean Decrease Accuracy (62.56). This implies that past delinquencies are a significant indicator of whether a borrower will default.

*** CLAGE (Age of Oldest Credit Line) and LOAN (Loan Amount)** are also important predictors, with high values in both metrics, suggesting that longer credit histories and larger loan amounts play a substantial role in default risk.

*** REASON (Reason for the Loan)** has relatively low importance, especially in Mean Decrease Gini (8.23), indicating that this feature contributes less to the classification compared to other factors.

Random Forest and Feature Importance

We selected Random Forest for this analysis due to its ability to handle complex, large datasets with non-linear interactions between variables, making it well-suited for financial datasets like this one. The feature importance analysis provides valuable insights into which factors are most strongly associated with loan default risk.



< Figure 4.1.4: Variable Importance plot – Random Forest >

In summary, **DEBTINC**, **DELINQ**, **CLAGE**, and **LOAN** are the most significant predictors for loan default. This insight helps prioritize which features to focus on for further

analysis and potentially discard features like **REASON** that have less impact, leading to more efficient model training and better predictive performance.

5.0 Data Transformation

In this project, data transformation was a critical step to prepare the dataset for analysis and model building. We performed two key transformations: converting categorical variables into numeric form and normalizing the numeric variables.

Normalization was applied to all numeric variables using the standard scaling method (mean = 0, standard deviation = 1). This ensured that features like LOAN, MORTDUE, and VALUE were on a comparable scale, reducing the influence of large magnitudes during model training.

While normalization is critical for logistic regression due to its reliance on numerical coefficients, decision tree models remain unaffected by feature scaling as they operate on splits

based on thresholds. Thus, normalization primarily benefited logistic regression, enhancing its convergence and interpretability, while leaving decision tree performance unchanged.

Categorical to Numeric Conversion:

We used the as.numeric() function to convert categorical variables such as REASON, JOB, and BAD into numeric form. This step was necessary because many machine learning algorithms, including logistic regression and random forests, require numeric input to function properly. Categorical data cannot be used directly in these models without being encoded numerically. By transforming these variables, we enabled their inclusion in both the correlation analysis and the predictive models, ensuring that critical information about loan applicants' employment status, loan purpose, and default status was fully utilized.

```
> loan.data.complete$REASON <- as.numeric(loan.data.complete$REASON)
> loan.data.complete$JOB <- as.numeric(loan.data.complete$JOB)
> loan.data.complete$BAD <- as.numeric(loan.data.complete$BAD)
```

< Figure 5.1: Categorical to Numeric Conversion>

The outcome of this transformation allowed us to:

- Perform a comprehensive correlation analysis between all variables, both numeric and categorical, to understand their relationships.
- Use these features as inputs to machine learning models, ensuring that no important categorical information was lost in the process.

Normalization of Numeric Variables:

The numeric variables in the dataset were normalized to bring them to a common scale. Normalization is a critical step in data preprocessing, especially when working with machine learning models like Logistic Regression and Decision Trees, where the range of the numeric variables can significantly affect the model's performance.

We identified the numeric variables using the sapply() function, which returns a logical vector indicating whether each column in the dataset is numeric. Once identified, we applied the scale() function to standardize the numeric variables. This method centres each variable by subtracting the mean and then scales it by dividing by the standard deviation, transforming the data into a distribution with a mean of 0 and a standard deviation of 1. This ensures that all numeric features are on a comparable scale, reducing potential biases due to differences in magnitude.

```
> # Data Transformation
> # Normalize numeric variables
> numeric_vars <- names(loan.data_clean)[sapply(loan.data_clean, is.numeric)]
> loan.data_normalized <- loan.data_clean %>%
+   mutate(across(all_of(numeric_vars), scale)) %>%
+   mutate(BAD = factor(BAD, levels = c("0", "1")))
> loan.data_normalized
    BAD      LOAN      MORTDUE      VALUE  REASON     JOB          YOJ      DEROG
6     1 -1.545464 -1.006997106 -1.2279559 HomeImp   Other -0.02155811 -0.2541827
8     1 -1.536627 -1.052705690 -1.1778826 HomeImp   Other  0.23929134 -0.2541827
20    0 -1.492440  0.597539504  0.2597252 HomeImp  Office -0.93453119 -0.2541827
26    1 -1.483603 -0.910598016 -1.0960198 HomeImp     Mgr  0.36971606 -0.2541827
27    0 -1.483603  0.509942556  0.1903900 HomeImp  Office -0.67368174 -0.2541827
35    0 -1.439416  0.632815093  0.1038594 HomeImp  Office -1.06495592 -0.2541827
36    0 -1.439416  0.642287448  0.2550943 HomeImp  Office -0.93453119 -0.2541827
37    1 -1.439416 -1.516314942 -0.7173329 HomeImp   Other  0.89141497  4.9503933
38    1 -1.439416 -0.305193855 -0.6634773 DebtCon     Mgr -0.93453119 -0.2541827
39    0 -1.430579  0.646688519  0.2740425 HomeImp  Office -0.93453119 -0.2541827
57    0 -1.412904 -0.016957224 -0.3618009 HomeImp ProfExe -0.28240756 -0.2541827
60    1 -1.404067  1.226378809  1.0597923 DebtCon   Other -0.02155811 -0.2541827
68    0 -1.377555  0.560074551  0.1451506 HomeImp  Office -0.41283229 -0.2541827
70    0 -1.377555 -0.520220795 -0.7912622 HomeImp  Office  1.41311387 -0.2541827
71    1 -1.368717 -1.290519900 -1.5817537 HomeImp   Other -0.54325701 -0.2541827
73    0 -1.359880 -0.546068709 -0.8010407 HomeImp  Office  1.41311387 -0.2541827
76    1 -1.351042 -1.021563087 -1.1238978 HomeImp   Other  0.23929134 -0.2541827
79    0 -1.351042  0.592468220  0.2189322 HomeImp  Office -0.93453119 -0.2541827
81    0 -1.342205  0.659958753  0.1088225 HomeImp  Office -1.06495592 -0.2541827
```

< Figure 5.2: Normalised Numeric Variables >

The transformation was applied using the mutate (across ()) function in the dplyr package, which efficiently scales all numeric columns in the dataset. Additionally, the target variable (BAD), which represents loan default status, was retained as a factor with levels "0" and "1" for non-default and default cases, respectively.

The result is a normalized dataset (loan.data_normalized), where each numeric variable has been scaled, making the data ready for machine learning model training.

5.1 Handling Imbalanced Data

We noticed here the class imbalance in the dataset, where the target variable (BAD), representing loan default status, had an unequal distribution of class labels. Specifically, the non-default class (BAD = 0) was overrepresented compared to the default class (BAD = 1). Class imbalance can lead to biased model predictions, where the model may favor the majority class, resulting in poor performance when predicting the minority class.

To handle this imbalance, we implemented random oversampling of the minority class (BAD = 1). Random oversampling is a data-level approach that involves duplicating instances from the minority class to balance the distribution between the two classes.

Here's the step-by-step process:

- **Separate the Classes:**

  First, we split the dataset into two subsets:

  - Majority Class: All rows where BAD = 0.
  - Minority Class: All rows where BAD = 1.

```
> majority_class <- loan.data_normalized %>% filter(BAD == "0")
> minority_class <- loan.data_normalized %>% filter(BAD == "1")
```

< Figure 5.3: Handling Imbalanced Data: Separating the classes>

- **Oversampling the Minority Class:**

The dataset exhibited class imbalance with a significant overrepresentation of the non-default class (BAD = 0). To address this, we applied random oversampling to the minority class (BAD = 1):

1. Class Separation: The dataset was split into majority (BAD = 0) and minority (BAD = 1) subsets.

2. Oversampling: Instances from the minority class were duplicated using the sample() function to match the size of the majority class.
3. Recombination: The oversampled minority subset was combined with the original majority subset to create a balanced dataset.

The resulting balanced dataset ensured equitable representation of both classes, reducing bias and improving the models' ability to predict loan defaults accurately.

```
> minority_class_oversampled <- minority_class[sample(nrow(minority_class), nrow(majority_clas
s), replace = TRUE), ]
```

< Figure: Handling Imbalanced Data: Oversampling the Minority class>

Combine the Data:

After oversampling the minority class, we combined the oversampled minority class and the majority class into a new dataset, loan.data_balanced, which has an equal representation of both classes.

```
> loan.data_balanced <- rbind(majority_class, minority_class_oversampled)
```

< Figure: Handling Imbalanced Data: Combining the data>

The result of this process is a balanced dataset, loan.data_balanced, where both the default (BAD = 1) and non-default (BAD = 0) classes have equal representation. This balanced dataset is crucial for training machine learning models as it mitigates the risk of bias towards the majority class, leading to improved model performance, particularly in predicting minority class outcomes such as loan defaults.

6.0 Data Partitioning

We used **random stratified sampling** to maintain the balance between the target classes (BAD = 0 and BAD = 1) in both the training and test sets. The stratified partitioning ensures that the proportion of default (BAD = 1) and non-default (BAD = 0) cases is preserved in both sets, preventing any bias during model evaluation.

The process was as follows:

Stratified Sampling:

We used the createDataPartition() function from the caret package, which performs stratified sampling to create a split of 70% for the training set and 30% for the test set. This partitioning was done while ensuring that both the training and test sets had a balanced representation of the target classes (BAD).

```
> # Data Partitioning
> set.seed(123)
> train_index <- createDataPartition(loan.data_balanced$BAD, p = 0.7, list = FALSE)
```

< Figure: Data Partitioning – 70/30 (Training / Test) >

Training and Test Sets:

The training set, train_data, consists of 4,490 observations and 13 variables, and the test set, test_data, consists of 1,922 observations and 13 variables. These sets will be used for model training and evaluation, respectively.

```
> dim(train_data)
[1] 4490   13
> dim(test_data)
[1] 1922   13
```

```
> head(train_data)
  BAD      LOAN    MORTDUE      VALUE  REASON    JOB        YOJ      DEROG     DELINQ
1   0 -1.492440  0.5975395  0.2597252 HomeImp Office -0.9345312 -0.2541827 -0.3399558
2   0 -1.483603  0.5099426  0.1903900 HomeImp Office -0.6736817 -0.2541827 -0.3399558
4   0 -1.439416  0.6422874  0.2550943 HomeImp Office -0.9345312 -0.2541827 -0.3399558
5   0 -1.430579  0.6466885  0.2740425 HomeImp Office -0.9345312 -0.2541827 -0.3399558
7   0 -1.377555  0.5600746  0.1451506 HomeImp Office -0.4128323 -0.2541827 -0.3399558
8   0 -1.377555 -0.5202208 -0.7912622 HomeImp Office  1.4131139 -0.2541827 -0.3399558
       CLAGE       NINQ       CLNO    DEBTINC
1 -1.0933675 -0.6669876 -0.9566620 -0.3128358
2 -1.0591521 -0.6669876 -0.9566620 -0.5484051
4 -0.9653559 -0.6669876 -0.9566620 -0.5194905
5 -1.1553620 -0.6669876 -0.8498258 -0.2546093
7 -1.1239764 -0.6669876 -0.8498258 -0.5840225
8  0.2814404 -0.6669876 -0.2088091 -1.6864106
> head(test_data)
   BAD      LOAN     MORTDUE      VALUE  REASON     JOB        YOJ      DEROG     DELINQ
3    0 -1.439416  0.63281509  0.1038594 HomeImp  Office -1.0649559 -0.2541827 -0.3399558
6    0 -1.412904 -0.01695722 -0.3618009 HomeImp ProfExe -0.2824076 -0.2541827 -0.3399558
12   0 -1.342205 -0.47093774 -0.8320737 HomeImp  Office  1.5435386 -0.2541827 -0.3399558
14   0 -1.324530 -0.42623447 -0.8792873 HomeImp  Office  1.2826891 -0.2541827 -0.3399558
15   0 -1.324530 -0.20890647 -0.4413759 HomeImp   Other -0.8041065 -0.2541827 -0.3399558
22   0 -1.298018 -0.10144885 -0.5097517 HomeImp   Other -0.6736817 -0.2541827 -0.3399558
       CLAGE        NINQ       CLNO    DEBTINC
3  -1.0313531 -0.66698760 -0.9566620 -0.5027771
6   0.8440678 -0.66698760 -1.0634981  1.0859316
12  0.2973898 -0.66698760 -0.3156453 -1.5213917
14  0.3631194 -0.01502391 -0.2088091 -1.8647190
15 -0.8593038 -0.66698760 -0.7429897  1.1527214
22 -0.7563662 -0.66698760 -0.8498258  0.9019800
```

< Figure 6.1: Data Partitioning>

Dimensionality of Partitioned Data:

The training data is used to build and fine-tune machine learning models, while the test data is used to evaluate the performance of the models on unseen data.

- **Training set dimensions**: 4,490 rows and 13 columns
- **Test set dimensions**: 1,922 rows and 13 columns

This partitioning ensures that the model's performance can be reliably assessed on data it hasn't been trained on, providing insights into how well the model generalizes to new data.

7.0 Model Selection:

7.1 Logistic Regression Model

The logistic regression model was developed to predict loan defaults, represented by the variable BAD, utilizing various borrower attributes and loan characteristics. This model was fit to the training data using the glm() function with a binomial family, allowing us to estimate the probability of default based on the independent variables.

The model includes multiple predictor variables such as loan amount (LOAN), mortgage due (MORTDUE), home value (VALUE), reasons for the loan (REASON), job type (JOB), years on the job (YOJ), derogatory marks (DEROG), delinquencies (DELINQ), and other financial attributes. The results of the logistic regression analysis are summarized as follows:

Model Coefficients and Interpretation:

- The **intercept** has a significant negative coefficient of -0.526 ($p < 0.001$), indicating the baseline log-odds of loan default when all predictor variables are set to zero.
- **LOAN**: The loan amount has a statistically significant negative coefficient (-0.139, $p < 0.01$), meaning that higher loan amounts are associated with lower odds of default.
- **JOB (Office)** and **YOJ**: These variables have significant negative coefficients, indicating that being employed in an office job and having more years on the job are associated with lower odds of loan default.
    - **JOB (Office)**: Coefficient = -0.538, $p < 0.001$.
    - **YOJ**: Coefficient = -0.170, $p < 0.001$.
- **DEROG** and **DELINQ**: These variables are among the strongest positive predictors of loan default. A higher number of derogatory marks (Coefficient = 0.458, $p < 0.001$) and delinquencies (Coefficient = 0.673, $p < 0.001$) are strongly associated with higher odds of default.

```
> summary(logistic_model)

Call:
glm(formula = BAD ~ ., family = "binomial", data = train_data)

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)   -0.526211   0.100497  -5.236 1.64e-07 ***
LOAN          -0.139129   0.045145  -3.082 0.002057 **
MORTDUE        0.031766   0.086553   0.367 0.713610
VALUE          0.049821   0.089764   0.555 0.578877
REASONHomeImp -0.004653   0.081553  -0.057 0.954502
REASONOther    0.219572   0.280268   0.783 0.433372
JOBOffice     -0.537796   0.136367  -3.944 8.02e-05 ***
JOBOther       0.101351   0.112251   0.903 0.366582
JOBProfExe     0.037775   0.123789   0.305 0.760248
JOBSales       1.851251   0.256399   7.220 5.19e-13 ***
JOBSelf        0.776332   0.220583   3.519 0.000432 ***
YOJ           -0.170263   0.040539  -4.200 2.67e-05 ***
DEROG          0.458300   0.038238  11.985  < 2e-16 ***
DELINQ         0.672672   0.040905  16.445  < 2e-16 ***
CLAGE         -0.369970   0.043127  -8.579  < 2e-16 ***
NINQ           0.225134   0.032064   7.021 2.20e-12 ***
CLNO          -0.276753   0.038904  -7.114 1.13e-12 ***
DEBTINC        0.528772   0.034711  15.233  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6224.5  on 4489  degrees of freedom
Residual deviance: 4723.0  on 4472  degrees of freedom
AIC: 4759

Number of Fisher Scoring iterations: 6
```

< Figure 7.1.1: Logistic Model Coefficient>

- **JOB (Sales)** and **JOB (Self-employed)**: Both are positively associated with loan default, with coefficients of 1.851 ($p < 0.001$) and 0.776 ($p < 0.001$), respectively. Individuals in sales or self-employment have higher odds of default compared to other job types.
- **DEBTINC**: Debt-to-income ratio also has a strong positive association with default (Coefficient = 0.529, $p < 0.001$), indicating that higher debt relative to income increases the likelihood of default.
- Other significant predictors include **CLAGE** (Coefficient = -0.370, $p < 0.001$), which suggests that older credit accounts reduce the odds of default, and **NINQ** (Coefficient = 0.225, $p < 0.001$), where more recent credit inquiries increase the odds of default.

Model Performance:

- The **null deviance** is 6224.5, while the **residual deviance** after fitting the model is 4723.0, indicating an improvement in model fit.
- The **Akaike Information Criterion (AIC)** is 4759, which can be used to compare the fit of this model with other potential models.
- The model converged after 6 Fisher Scoring iterations, indicating stable estimates of the coefficients.

Significance of Predictors:

- Variables with highly significant p-values (e.g., $p < 0.001$ for DEROG, DELINQ, DEBTINC, CLAGE) are critical in determining the likelihood of loan default.
- Some variables, like REASON (Home Improvement or Other), do not appear to be significant predictors ($p > 0.05$), suggesting they have little impact on loan default in this dataset.

The logistic regression model identifies several key factors contributing to the risk of loan default, including derogatory marks, delinquencies, debt-to-income ratio, job type, and loan amount. These insights could be valuable in assessing and mitigating default risks within a lending institution. The significant predictors, particularly financial history and employment-related factors, are consistent with common predictors of credit risk, making this model well-suited for practical application in predicting loan default.

7.2 Decision Tree:

We trained to predict loan default (BAD) using various financial and demographic predictors. The decision tree algorithm was chosen for its interpretability, which allows us to clearly see how different factors contribute to loan default risk. The tree was constructed using

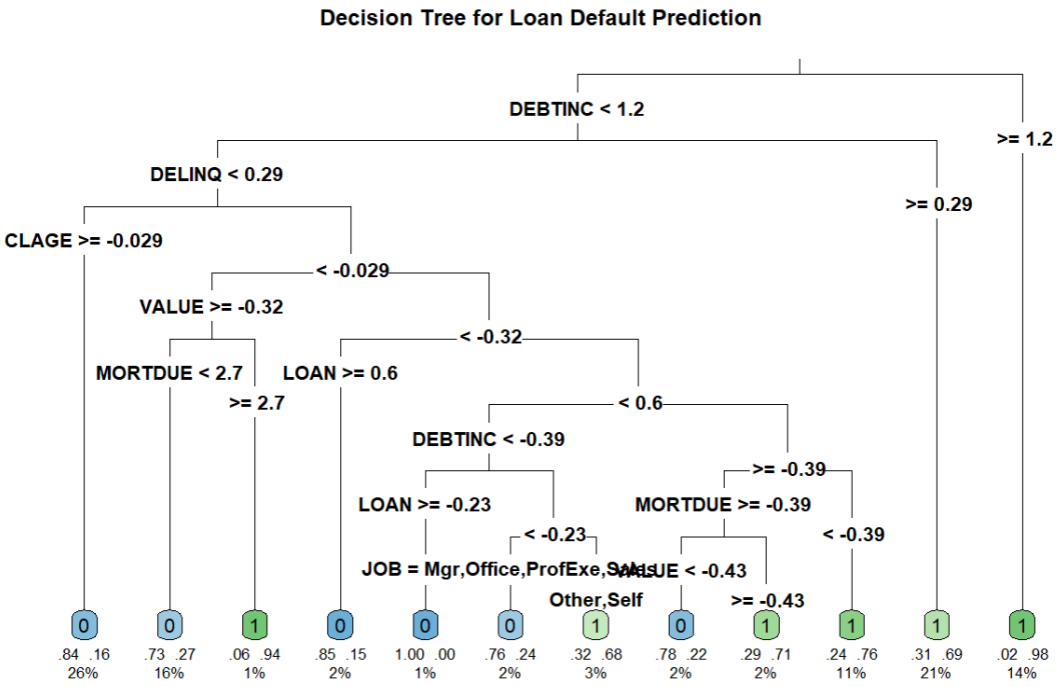
predictors such as the age of the credit account (CLAGE), debt-to-income ratio (DEBTINC), delinquencies (DELINQ), job type (JOB), loan amount (LOAN), mortgage due (MORTDUE), and home value (VALUE). These variables were selected by the algorithm as the most relevant features for classifying whether a customer will default.

```
> printcp(tree_model)

Classification tree:
rpart(formula = BAD ~ ., data = train_data, method = "class")

Variables actually used in tree construction:
[1] CLAGE   DEBTINC DELINQ  JOB     LOAN
[6] MORTDUE VALUE

Root node error: 2245/4490 = 0.5

n= 4490

        CP nsplit rel error  xerror     xstd
1 0.264588      0   1.00000 1.03385 0.014915
2 0.160802      1   0.73541 0.73586 0.014394
3 0.033185      2   0.57461 0.57506 0.013509
4 0.024053      4   0.50824 0.50468 0.012964
5 0.017372      5   0.48419 0.48998 0.012837
6 0.012918      6   0.46682 0.48062 0.012753
7 0.011804      7   0.45390 0.45924 0.012554
8 0.010245      9   0.43029 0.45301 0.012493
9 0.010000     11   0.40980 0.44276 0.012392
```

< Figure 7.2.1: Decision Tree Model >

The initial root node error was 0.5, indicating that 50% of the cases in the training data were defaults. As the tree was split, the model's error decreased, and the cross-validation error stabilized at around 0.453 after 9 splits, indicating the model's ability to generalize. This reduction in error suggests that the decision tree effectively identifies relationships between the input features and the likelihood of default.

The decision tree depicted in the image provides a visual representation of the process used to predict whether a borrower will default on a loan (BAD = 1) or not (BAD = 0) based on various features. The model splits the data into branches at different points using the most significant variables, aiming to maximize the separation of borrowers with high probabilities of default from those with low probabilities.



< Figure 7.2.2: Plot: Decision Tree >

The root node of the tree begins with the feature "DEBTINC" (Debt-to-Income Ratio), which acts as the most crucial determinant of loan default probability, suggesting that borrowers with a DEBTINC value greater than or equal to 1.2 are significantly more likely to default, as indicated by the leaf nodes showing higher probabilities of default.

Subsequent splits occur based on features such as "DELINQ" (number of delinquencies), "CLAGE" (age of the oldest credit line), "LOAN," "MORTDUE" (amount due on the mortgage), and "JOB" type. These variables further refine the categorization, revealing different segments of borrowers who are more or less likely to default. For instance, borrowers with a low DELINQ and low CLAGE, alongside other specific conditions, are predicted to have a very low likelihood of default (BAD = 0), while those with higher delinquencies and debt-to-income ratios are more likely to default. The leaf nodes at the end of each branch indicate the predicted class (either 0 or

1) and the associated probability of default within that branch, allowing for detailed insight into how each feature combination influences the default prediction.

8.0 Model Evaluation

8.1 Logistic Regression Performance:

The performance of the binary classification model is evaluated using the confusion matrix and several performance metrics.

- **Accuracy**: The model achieved an accuracy of **72.16%** (95% CI: 70.1% - 74.16%), which indicates that approximately 72% of the predictions were correct. This accuracy is significantly higher than the No Information Rate (NIR), which is 50% for this balanced dataset. The p-value (< 2.2e-16) confirms that the model's accuracy is statistically significant.
- **Kappa Score**: The Kappa score is **0.4433**, which measures the agreement between the predicted and actual classifications, adjusted for chance. A Kappa value in this range suggests moderate agreement, indicating that the model is making meaningful predictions but still has room for improvement.
- **Sensitivity (Recall)**: The sensitivity, or true positive rate, is **67.01%**, indicating that the model correctly identifies 67% of the actual positive instances (class 1). This means the model misses approximately 33% of the positive cases.
- **Specificity**: The specificity, or true negative rate, is **77.32%**, meaning that the model correctly identifies 77% of the actual negative instances (class 0). This is higher than the sensitivity, suggesting the model performs better at predicting the negative class.

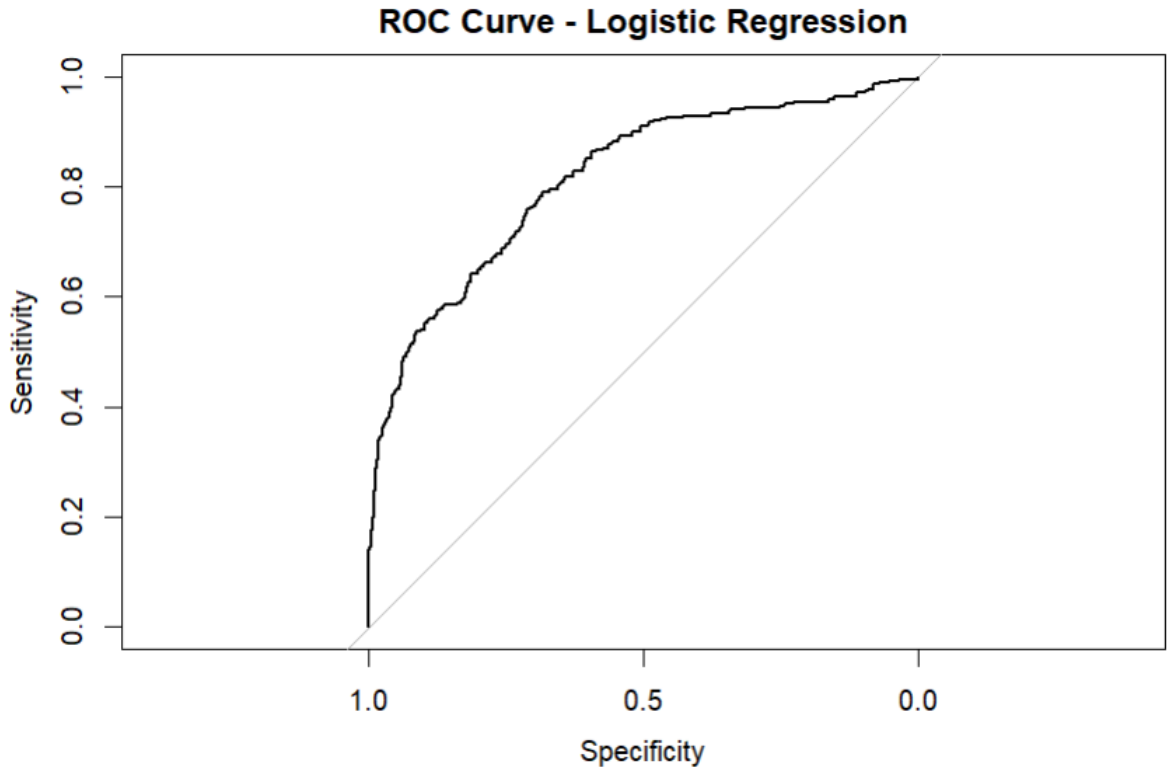
```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 743 317
         1 218 644

               Accuracy : 0.7216
                 95% CI : (0.701, 0.7416)
    No Information Rate : 0.5
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.4433

 Mcnemar's Test P-Value : 2.266e-05

            Sensitivity : 0.6701
            Specificity : 0.7732
         Pos Pred Value : 0.7471
         Neg Pred Value : 0.7009
             Prevalence : 0.5000
         Detection Rate : 0.3351
   Detection Prevalence : 0.4485
      Balanced Accuracy : 0.7216

       'Positive' Class : 1
```

< Figure 8.1.1: Logistic Regression – Confusion Matrix >

- **Positive Predictive Value (Precision)**: The positive predictive value is **74.71%**, meaning that when the model predicts the positive class, it is correct about 75% of the time.
- **Negative Predictive Value**: The negative predictive value is **70.09%**, indicating that when the model predicts the negative class, it is correct about 70% of the time.
- **Balanced Accuracy**: The balanced accuracy, which accounts for the imbalance between classes, is **72.16%**, reflecting an overall balance between sensitivity and specificity.

The model demonstrates reasonable performance with an accuracy of 72.16% and moderate agreement as indicated by the Kappa score. Sensitivity is slightly lower than specificity, meaning the model is more conservative in predicting positives and performs better at identifying negatives. With a precision of 74.71%, the model is confident in its positive predictions but can

still be improved, especially in reducing false negatives (improving sensitivity). Overall, the model performs significantly better than random chance, but further tuning or alternative models could improve the detection of positive cases.



< Figure 8.1.2: ROC Curve – Logistic Regression >

The curve plots the true positive rate (sensitivity) against the false positive rate (1-specificity). Our model shows strong predictive performance, with the curve bending sharply toward the upper-left corner of the plot, well above the diagonal reference line that would indicate random classification. The substantial area between the ROC curve and the diagonal suggests a good discrimination ability of the model. The curve achieves high sensitivity (approximately 0.8) while maintaining reasonable specificity values, indicating that the model successfully balances true positive and false positive rates. This performance suggests that our logistic regression model is effective at distinguishing between the two classes in our dataset

8.2 Decision Tree Performance

In this model, several performance metrics have been analyzed using a confusion matrix for a binary classification task. Below is the breakdown of the performance:

- **Accuracy:** The model achieved an accuracy of 77.47% (95% CI: 75.54% - 79.32%), meaning that around 77% of the predictions were correct. This is a noticeable improvement compared to random chance, given the dataset's balanced nature (No Information Rate = 50%). The p-value (< 2.2e-16) confirms that this result is statistically significant.
- **Kappa Score:** The Kappa statistic is 0.5494, which indicates moderate agreement between the model's predictions and actual labels. This suggests that the model is consistently making predictions better than random chance, though there remains room for improvement.
- **McNemar's Test:** The McNemar test p-value is 0.003934, suggesting that there is a significant difference between the types of errors made by the model (false positives and false negatives). This indicates that the model's error distribution is uneven between classes.
- **Sensitivity (Recall):** The sensitivity, or true positive rate, is 80.65%, meaning that the model successfully identified 80.65% of the actual positive cases. This relatively high sensitivity suggests the model is good at detecting positive instances, making it less likely

to miss true positives.

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 714 186
         1 247 775

               Accuracy : 0.7747
                 95% CI : (0.7554, 0.7932)
    No Information Rate : 0.5
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5494

 Mcnemar's Test P-Value : 0.003934

            Sensitivity : 0.8065
            Specificity : 0.7430
         Pos Pred Value : 0.7583
         Neg Pred Value : 0.7933
             Prevalence : 0.5000
         Detection Rate : 0.4032
   Detection Prevalence : 0.5317
      Balanced Accuracy : 0.7747

       'Positive' Class : 1
```

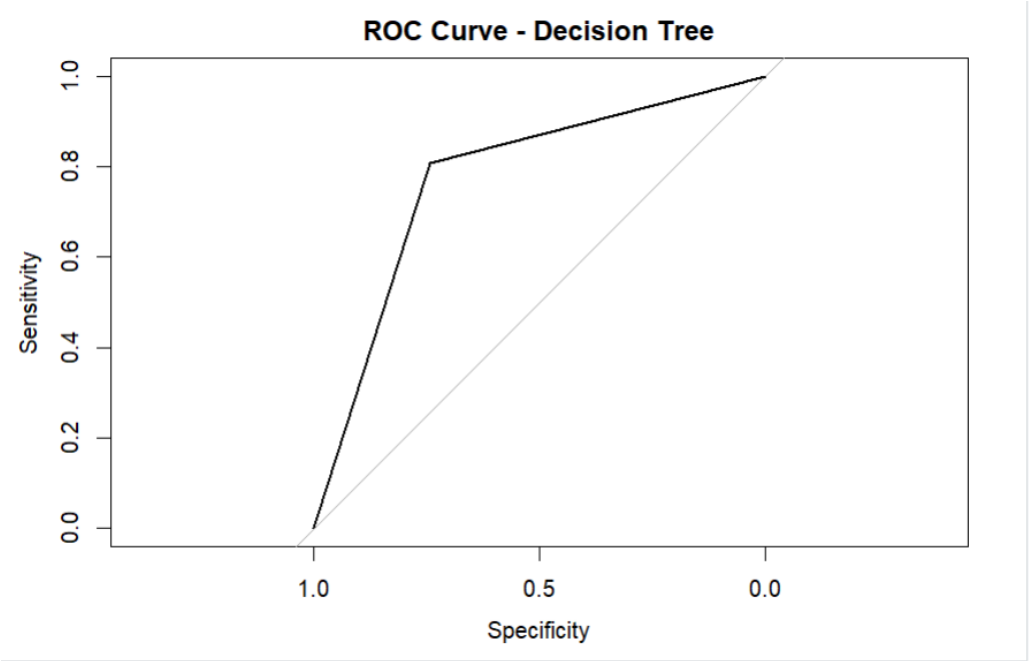
< Figure 8.2.1: Confusion Matrix – Decision Tree >

- **Specificity:** The specificity is 74.30%, indicating the model correctly identified 74.3% of negative instances. While lower than sensitivity, this specificity still shows a solid ability to distinguish between positive and negative classes.
- **Positive Predictive Value (Precision):** The positive predictive value is 75.83%, meaning that when the model predicts the positive class, it is correct approximately 76% of the time. This precision shows the model is relatively confident in its positive predictions, though some false positives are still present.

- **Negative Predictive Value**: The negative predictive value is 79.33%, showing that when the model predicts a negative class, it is correct about 79% of the time. This strong NPV indicates the model performs well at predicting the negative class as well.
- **Balanced Accuracy**: The balanced accuracy, which is the average of sensitivity and specificity, is 77.47%, confirming that the model strikes a reasonable balance in identifying both positive and negative instances.

This model demonstrates strong overall performance, with an accuracy of 77.47% and a balanced sensitivity (80.65%) and specificity (74.30%). The Kappa statistics further support the model's predictive reliability. The high sensitivity means it is good at identifying positive cases, making it suitable for tasks where capturing positives is critical, though efforts could be made to further balance specificity and reduce false positives.

The ROC curve for the decision tree classifier reveals distinct classification behavior characterized by sharp transitions in performance. The model demonstrates a notable step-like pattern, with a particularly steep increase in sensitivity around the 0.7 specificity mark, where the sensitivity jumps from approximately 0.4 to 0.8. This abrupt transition suggests that the decision tree makes relatively decisive classification boundaries at certain threshold values. The curve maintains a position well above the diagonal reference line, indicating performance substantially better than random classification. After the sharp increase, the curve shows a more gradual ascent toward maximum sensitivity, suggesting incremental improvements in the true positive rate as the classification threshold becomes more lenient. This pattern is characteristic of decision tree models, which naturally partition the feature space into discrete regions, resulting in more pronounced changes in classification performance at specific threshold values compared to other probabilistic classifiers.

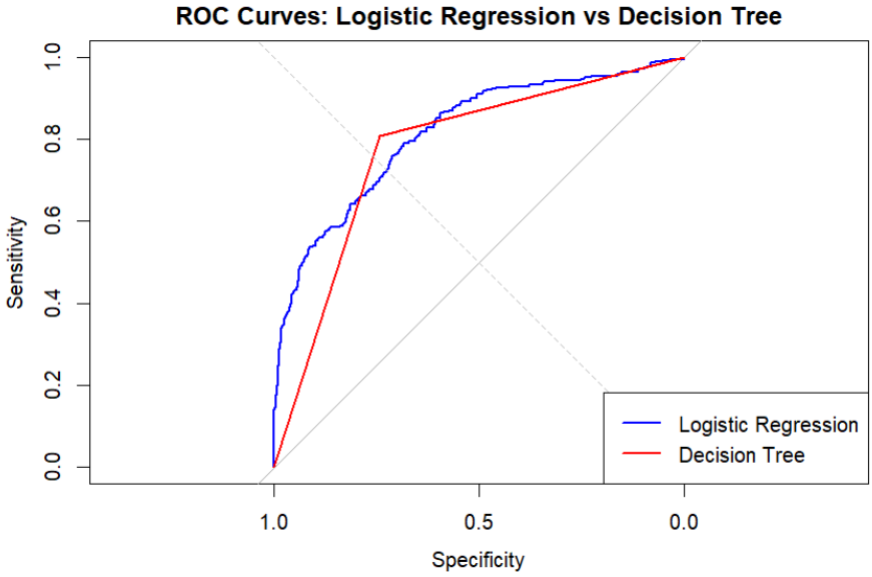


< Figure 8.2.2: ROC Curve – Decision Tree >

8.3 Model Comparison:

Here, I have compared the ROC curves for two classification models: logistic regression (blue line) and decision tree (red line). Both models demonstrate performance notably better than random classification, as indicated by their curves rising well above the diagonal reference line. The logistic regression model shows stronger performance in the high-specificity region (right side of the plot), exhibiting a more gradual curve with higher sensitivity values between specificity values of 0.7-1.0. Conversely, the decision tree model displays a sharper increase in sensitivity around the 0.7 specificity mark, suggesting more decisive classification boundaries. The curves intersect multiple times, indicating that neither model consistently outperforms the other across all threshold values. Both models achieve similar maximum sensitivity (approximately 1.0) as specificity approaches 0, suggesting comparable overall discriminative ability. This comparison reveals that while both models are effective classifiers, they may be

optimal for different operational points depending on the desired trade-off between sensitivity and specificity.



< Figure 8.2.3: ROC Curves – Comparison>

9.0 Profit Analysis:

In this section, we evaluate the financial implications of misclassification in the Logistic Regression and Decision Tree models. Using a cost matrix approach, we quantified the impact of False Positives (FP) and False Negatives (FN) to provide actionable insights for operational decision-making.

9.1 Cost Matrix

A cost matrix assigns specific financial penalties to different types of misclassification. This approach ensures that model evaluation goes beyond traditional accuracy metrics to address business-critical outcomes.

Cost Definitions:

- **False Positive (FP):** Incorrectly predicting a loan default (BAD = 1) when the borrower does not default (BAD = 0). Cost: **$10,000**.
- **False Negative (FN):** Failing to predict a loan default (BAD = 0) when the borrower defaults (BAD = 1). Cost: **$5,000**.

```
> print(cost_matrix_logistic)
         Predicted 1 Predicted 0
Actual 1           0        5000
Actual 0       10000           0
```

Cost Matrix Implementation:

1. **Extract Confusion Matrix Metrics:** For both models, True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN) were extracted.
2. **Calculate Total Cost:** The total cost was computed as:

Total Cost = (FP * $10,000) + (FN * $5,000)

3. **Comparison of Total Costs:**
    - **Logistic Regression:** Total Misclassification Cost = **$4,260,000**
    - **Decision Tree:** Total Misclassification Cost = **$3,095,000**

Findings:

- The Decision Tree model achieved a **27.3% reduction** in misclassification cost compared to Logistic Regression.

- Despite Logistic Regression's slightly higher AUC, the Decision Tree model's superior sensitivity and balanced error management make it a more cost-effective choice for high-risk scenarios.

Interpretation:

The Decision Tree model effectively balances sensitivity and specificity, reducing financial losses from False Positives and False Negatives. This analysis highlights the importance of cost-sensitive evaluation in loan default prediction, ensuring alignment with the bank's risk management and profitability objectives.

10.0 Observation and Conclusion:

10.1 Observation:

Logistic Regression Performance:

- Achieved an accuracy of **72.16%**, with a Kappa score of **0.4433**, indicating moderate agreement beyond random chance.
- The model's specificity (**77.32%**) exceeds its sensitivity (**67.01%**), reflecting a conservative approach that minimizes false positives but misses more high-risk cases.
- Precision (74.71%) and balanced accuracy (72.16%) demonstrate reliable performance, though improvements in detecting defaulters are needed.

Decision Tree Performance:

- Outperformed Logistic Regression with an accuracy of **77.47%** and a higher Kappa score of **0.5494**, indicating stronger predictive reliability.
- Sensitivity of **80.65%** ensures effective identification of high-risk applicants, though specificity (**74.30%**) suggests a trade-off in false positives.

- The Decision Tree's balanced accuracy and cost-effectiveness make it particularly suitable for high-risk lending scenarios.

ROC Curve Analysis:

- Both models demonstrated strong discriminatory power with ROC curves above the random classification line.
- Logistic Regression excelled in the high-specificity region, while the Decision Tree showed sharper sensitivity gains at certain thresholds.
- The Decision Tree's performance aligns well with cost-sensitive objectives, making it the better choice for risk-averse operations.

10.2 Conclusion

The evaluation reveals that both Logistic Regression and Decision Tree models offer robust predictive capabilities, but their distinct strengths suit different operational priorities:

- **Logistic Regression:**
    - Best for scenarios where minimizing false positives is critical.
    - Provides conservative and stable predictions, making it suitable for low-risk lending.
- **Decision Tree:**
    - Outperforms Logistic Regression with higher sensitivity (80.65%) and lower misclassification cost ($3,095,000).
    - Ideal for applications where capturing potential defaulters is a priority.

The Decision Tree model's balance of accuracy, sensitivity, and cost-efficiency positions it as the optimal choice for deployment in the bank's automated loan approval system. This model ensures that high-risk applicants are effectively identified, minimizing financial losses and

improving overall decision-making efficiency. Future enhancements, such as ensemble methods or dynamic cost optimization, could further refine its performance

11.0 Recommendations:

Primary Model Deployment:

- Deploy the **Decision Tree model** as the primary tool for loan default prediction.
- Integrate it into the automated loan approval system to streamline decision-making while maintaining robust risk management.

Dynamic Loan Modification:

- Implement a secondary regression model for high-risk applicants flagged by the Decision Tree to recommend tailored loan terms (e.g., adjusted amounts, secured loans).
- Use personalized strategies to retain valuable customers and minimize risk.

Risk Mitigation Strategies:

- Provide early intervention programs for high-risk borrowers, such as credit counseling or repayment plans.
- Offer alternative lending products, such as credit-builder loans, to improve customer retention.

Periodic Model Monitoring:

- Evaluate model performance monthly and retrain quarterly using updated data to ensure adaptability to changing financial conditions.
- Monitor key metrics such as accuracy, sensitivity, and cost-effectiveness to maintain optimal performance.

Regulatory Compliance and Transparency:

- Leverage the interpretability of the Decision Tree model to comply with regulatory requirements, such as the Equal Credit Opportunity Act.
- Ensure that decision-making criteria are transparent and accessible to applicants.

Implementation Benefits:

- **Reduced Default Rates:** Enhanced risk assessment minimizes loan defaults.
- **Improved Customer Satisfaction:** Transparent and personalized lending processes foster stronger customer relationships.
- **Increased Profitability:** Cost-effective models reduce financial losses while expanding the customer base.
- **Operational Efficiency:** Automation streamlines the loan approval process, reducing manual interventions.

By adopting these recommendations, the bank can achieve a comprehensive, customer-centric risk management strategy while maximizing profitability and ensuring regulatory compliance.