Module Code: CT050-3-M-DAP

Module Name: Data Analytical Programming

Individual Assignment

Loan Approval Prediction for LFI

Student Name: Muhammad Yousouf Ali Budullah
TP Number: TP086704
Intake Code: APDMF2501DSBA(BI)(PR)
Programme: MSc Data Science & Business Analytics
Module Lecturer: Mr. Dhason Padmakumar

Date of Submission: September 7, 2025

Contents

Part 1

Foundations & Data Setup

Chapter 1

Introduction

In today's fast-paced financial environment, automation and data-driven decision-making are essential to maintain efficiency, accuracy, and competitiveness. As a data scientist at Lasiandra Finance Inc. (LFI), I have been tasked with developing a predictive model to improve the company's loan approval process. LFI is a leading private lender serving small and medium enterprises across the United States. To continue delivering customer-centric services, LFI must adopt innovative approaches to improve the current manual loan approval processes.

This project aims to build a predictive model using SAS analytics that can determine customer loan applications approvals and rejections with greater accuracy than the current process. By focusing on historical loan data, this project seeks to minimize human error in decision-making and improve the reliability of approvals. The model will serve as the foundation for automating LFI's loan eligibility screening, reduce processing time, support fair decision-making and help the business move toward a more scalable solution.

Chapter 2

Background

Lasiandra Finance Inc. is a private financing company based in New York that provides tailored loan services to SMEs. The business offers tailor-made financial solutions and flexibility to help small businesses access funding for growth and operations. LFI's strength lies in its personalised customer service and flexibility, but its current operations now faces challenges with its manual loan approval system.

Currently the manual process begins when an applicant submits their loan request, including personal and financial details. Loan officers then manually verify each application against internal criteria and past trends. They review each application one by one, checking details like gender, marital status, number of family members, income levels, employment type, and loan history. Each application is reviewed individually, requiring time-consuming cross-validation and decision-making.

This process causes delays and inconsistency. Without a data-driven model it also increases the risk of biases and errors in determining eligibility especially when dealing with a large number of loan applications. Therefore, there is an urgent need for an automated, predictive system that can promptly and accurately identify eligible applicants based on historical trends and attributes.

Chapter 3

Assumptions and Justification

Several assumptions have been made to guide the analysis and development of the predictive model. These assumptions have been made based on the provided dataset and the goal of the project to improve the loan approval process at LFI.

The following are assumed:

- The dataset provided is accurate
- The variables included are relevant to the decision-making process
- The data reflects real historical decisions made by LFI officers
- External factors like economic shifts or policy changes have not significantly affected the loan outcomes in the dataset
- Each record in the dataset represents a single loan application
- There is no major class imbalance that would distort model training
- All numeric and categorical variables are properly formatted and interpretable in SAS

Justification for the Use of SAS.

SAS was chosen because of its reliability, maturity, and strong support for statistical analysis . Its strengths include combining data management and analytics in one platform, producing reproducible reports through macros and ODS, and providing optimized procedures (e.g., `PROC LOGISTIC`, `PROC FREQ`, `PROC MEANS`) that simplify analysis. These features, along with its wide adoption in financial institutions, make SAS a natural choice for LFI's business needs.

At the same time, SAS has limitations. It is less flexible for modern machine learning integration, can be resource-heavy on large datasets, and is more costly than open-source alternatives. Compared with big-data pipelines such as NiFi or Spark, it also offers less scalability.

Despite these drawbacks, SAS is well suited for this project because the dataset is moderate in size, the focus is on interpretable statistical modeling, and the priority is transparent, auditable results. Using SAS ensures outputs that are reliable, consistent and aligned with the operational environment of the LFI.

Chapter 4

Data Dictionary / Metadata

Upload the Datasets to SAS Assignment Folder

Output

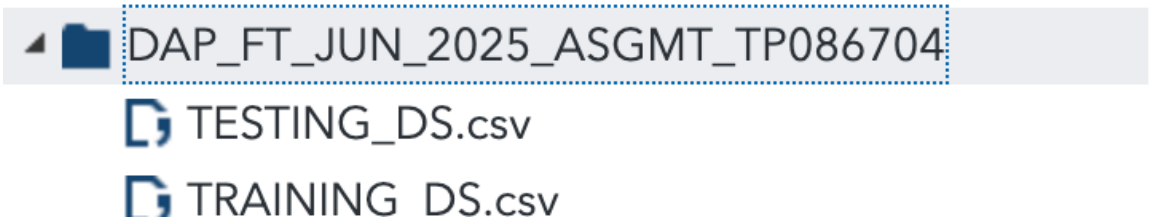

Figure 1: Datasets visible in the SAS Assignment folder

The `TRAINING_DS` and `TESTING_DS` datasets were successfully uploaded to the SAS Assignment folder using the SAS Studio interface. The data is comprised of loan application records and are the basis for all future pre-processing and analysis steps.

Transfer the Datasets to ASGMTLIB Permanent Library

Output

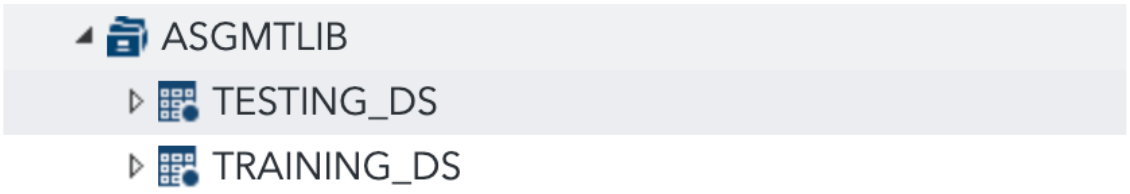

Figure 2: `TRAINING_DS` and `TESTING_DS` saved under `ASGMTLIB`

Using the Task Utilities in the SAS Studio interface, the data scientist selected the `TRAINING_DS` and `TESTING_DS` datasets from the uploaded files in the SAS assignment folder. By ensuring the library location was set to `ASGMTLIB` and executing the import, the datasets were saved permanently. This ensures they remain accessible across SAS sessions, supporting consistency throughout the project.

Data Dictionary / Metadata Structure

SAS Code

```sas
/* Display variable details and
   structure of the dataset */
PROC CONTENTS DATA=ASGMTLIB.
    TRAINING_DS VARNUM;
RUN;
```

Listing 1: SAS code to describe and inspect the permanent table structure

Outputs

Alphabetic List of Variables and Attributes					
#	Variable	Type	Len	Format	Informat
7	CANDIDATE_INCOME	Num	8	BEST12.	BEST32.
6	EMPLOYMENT	Char	3	$3.	$3.
4	FAMILY_MEMBERS	Char	2	$2.	$2.
2	GENDER	Char	6	$6.	$6.
8	GUARANTEE_INCOME	Num	8	BEST12.	BEST32.
9	LOAN_AMOUNT	Num	8	BEST12.	BEST32.
13	LOAN_APPROVAL_STATUS	Char	1	$1.	$1.
10	LOAN_DURATION	Num	8	BEST12.	BEST32.
11	LOAN_HISTORY	Num	8	BEST12.	BEST32.
12	LOAN_LOCATION	Char	7	$7.	$7.
3	MARITAL_STATUS	Char	11	$11.	$11.
5	QUALIFICATION	Char	14	$14.	$14.
1	SME_LOAN_ID_NO	Char	8	$8.	$8.

Table 1: `ASGMTLIB.TRAINING_DS` metadata schema (from `PROC CONTENTS`)

Using `PROC CONTENTS`, the metadata of the `TRAINING_DS` dataset was obtained. The schema includes both categorical and numeric variables and confirms consistent formats/informats, supporting later preprocessing and analysis.

Part 2

Literature Review & Variable Analysis

Chapter 5

Literature Review

Loan Application Process

Traditionally the loan application process has been littered with inefficiencies due to manual workflows, departmental handoffs, and limited compliance validation. Traditionally, even processing a single loan could take 11–13 business days, with approval rates as low as 42% and error rates exceeding 25%. These delays are the result of fragmented data entry, inconsistent application of underwriting criteria, and incomplete documentation checks (Munnangi, 2024). Consequently, many institutions had high abandonment rates, reducing competitiveness and customer satisfaction.

Munnangi (2024) provides a breakdown of these inefficiencies. As illustrated in Figure 3, manual loan processing involved several departmental focal points, with error rates of up to 27%, and processing times took longer than 11 business days. These results illustrate why automation has become essential, as systematic problems, including compliance backlogs and inadequate paperwork checks, were more accountable for inefficiency than applicant quality.

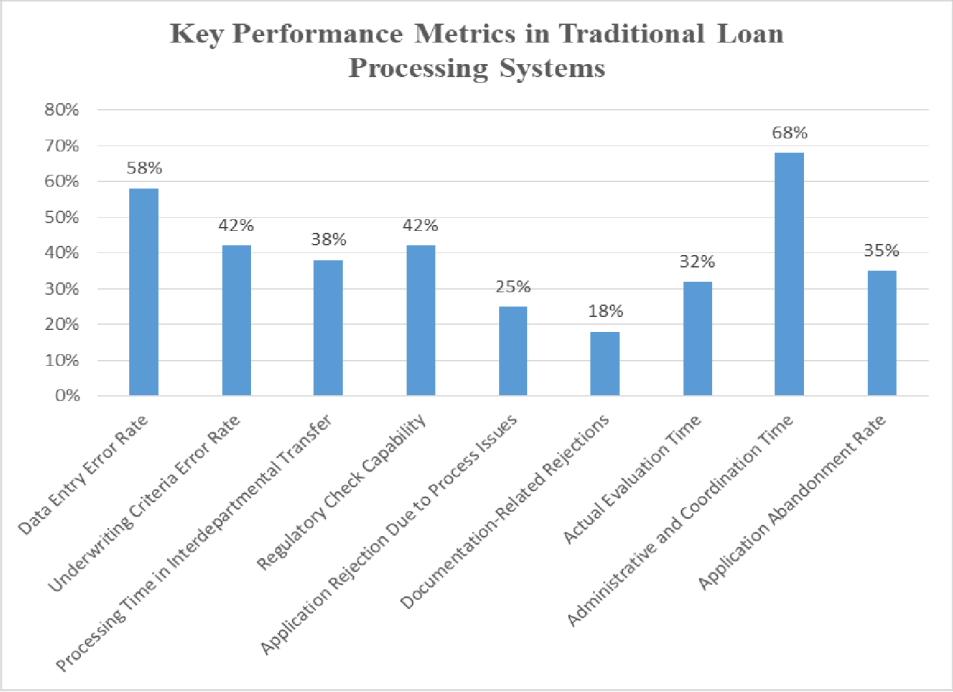

Figure 3: Analysis of manual loan processing inefficiencies in banking operations (adapted from (Munnangi, 2024)).

Recent developments illustrate how AI-driven document processing and intelligent automation are influencing this environment. Loan processing times decreased from almost two weeks to

less than 24 hours due to the introduction of a Loan Origination System (LOS) that combines intelligent routing, real-time compliance validation, and workflow automation (Munnangi, 2024). Customer satisfaction jumped to 89% and approval rates increased to 65%. When compared with manual processing, the LOS greatly reduced errors and delays, as seen in Figure 4, demonstrating visible efficiency gains.

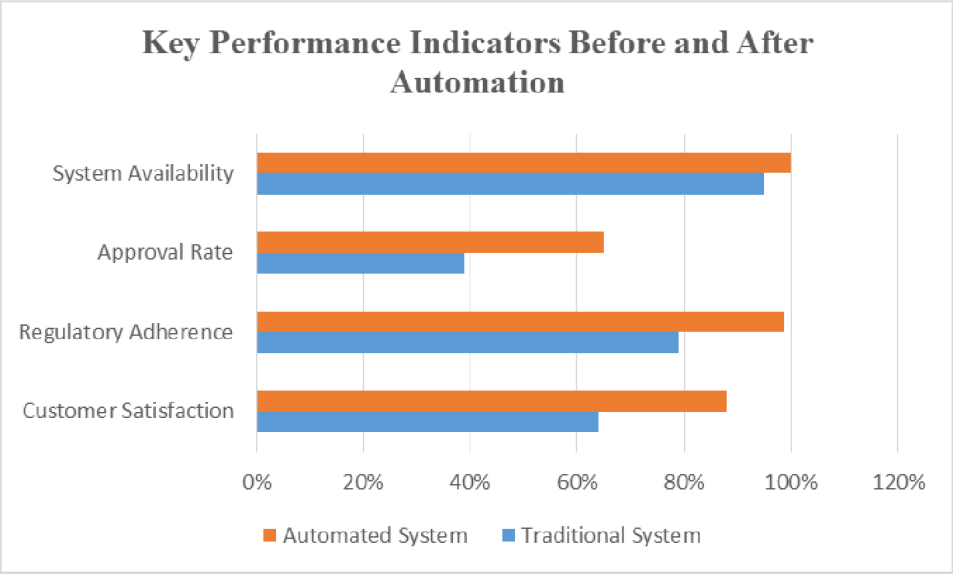

Figure 4: Before and after performance of loan processing systems, showing reduced delays and error rates through automation (adapted from (Munnangi, 2024)).

AI-driven Intelligent Document Processing (IDP) is a complementing invention. IDP uses robotic process automation, machine learning, and natural language processing to extract, validate, and filter applicant data in real time since loan applications involve numerous pieces of papers. According to Ramesh Pingili (2025), banks that adopted IDP have claimed 70% faster approvals, 50% higher fraud detection, and 40% lower compliance costs. Figure 5 reveals these cost reductions, comparing savings across categories including fraud detection and compliance.

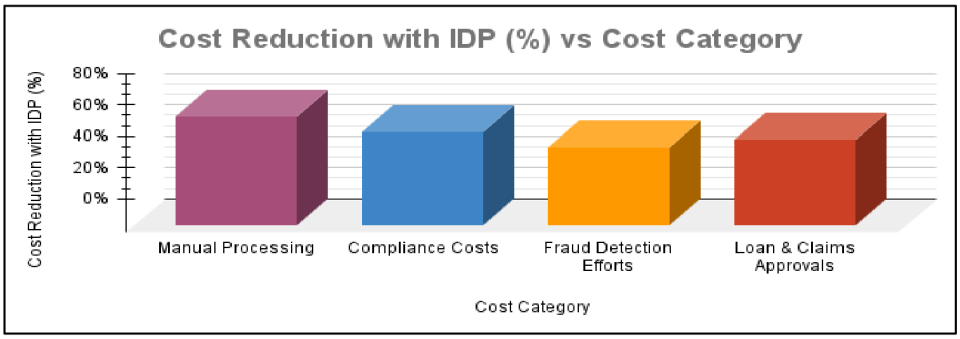

Figure 5: Cost savings achieved with AI-driven intelligent document processing in banking (adapted from (Ramesh Pingili, 2025)).

When used together, LOS and IDP offer an important change in loan applications from inef-

ficient, error prone systems to intelligent, flexible platforms that achieve a balance between accuracy, speed, and compliance. These tools enable financial institutions to deliver efficiency and confidence by reducing processing times.

Financing Companies and Credit Assessment

Predicting loan applications involves understanding how financing companies decide which applications are approved. Different criteria are used by different types of institutions, including banks, equity-based lenders, and non-bank financial institutions. Businesses often mix different forms of funding instead of depending only on one. For predictive modeling, this means that applicant qualities alone are not sufficient to explain loan approvals, institutional procedures and available funding options must also be taken into consideration (Santos et al., 2024).

Credit assessment has gone through a pretty substantial systematic change. Traditional approaches often introduced subjectivity and inconsistency since they rely on human judgment or basic rule-based scorecards. The use of machine learning is being employed by finance companies to produce faster and more accurate decisions. On a dataset of more than 1.3 million loans, Suhadolnik et al. (2023) examined ten methods and discovered that ensemble models in particularly, XGBoost consistently performed better than decision trees and logistic regression. The most important variables were the loan period and interest rate, which together accounted for over half of the predictive power, according to their feature importance analysis (Figure 6).

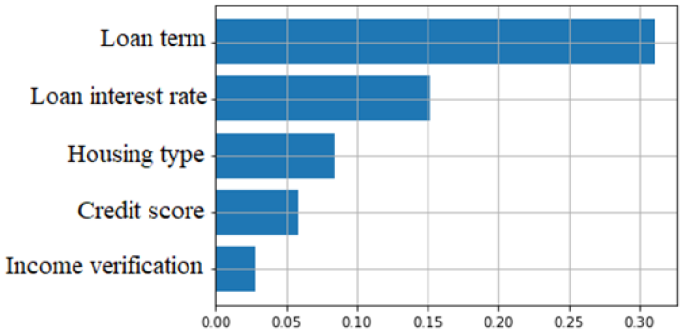

Figure 12. Relative importance of features.

Figure 6: Relative importance of features in loan application classification (adapted from (Suhadolnik et al., 2023)).

Model performance was also assessed using accuracy, precision, recall, F1, and AUC. As shown in Figure 7, ensemble models achieved higher predictive accuracy and better separation compared to traditional methods.

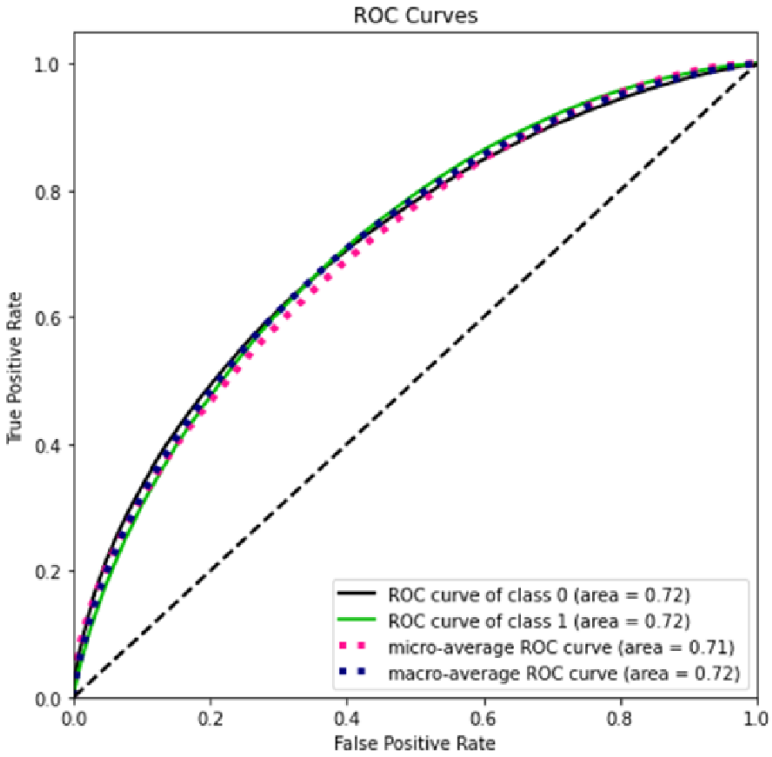

Figure 7: ROC curve comparison of classifiers in loan classification (adapted from (Suhadolnik et al., 2023)).

While financing firms offer a variety of tools, the factor that significantly impacts the prediction of loan applications is their increasing usage of machine learning. These methodological developments allow prediction models to more closely resemble the approval decisions taken by actual human lenders.

Bad Debts and Risk Management

Although loan approvals are a necessary step, financing companies also have to manage the default risk in order to ensure long-term sustainability of their decisions. Because they reduce profitability and threaten financial stability, bad debts, which are often referred to as non-performing loans (NPLs) are a significant problem to lenders. Consequently, it is vital to properly understand the factors that impact default risk to link loan application choices with future credit results.

Recent data from the European peer-to-peer (P2P) lending market, a sector of the financial system that has expanded substantially as part of FinTech lending, are presented by Nigmonov et al. (2024). Their study uses complex econometric techniques, such as endogeneity correction and the least absolute shrinkage and selection operator (LASSO), to analyse default risk using a loan-book dataset dating from 2014 to 2020. The results revealed that the chances of loan defaults are increased by both higher interest rates and larger stock market returns.

This is particularly relevant since it illustrates that credit risk is influenced by both borrowers' characteristics and the state of the financial market.

The study also reveals that the degree of FinTech adoption and the difficulties faced in the banking system have an impact on how severe these consequences are. Essentially, this means that default risk on alternative lending platforms can be increased by systemic stress and market volatility. Nigmonov et al. (2024) highlight the importance of incorporating macro-financial data into risk management frameworks by directly linking interest rates, stock market movements, and delinquency rates.

These findings illustrate how important the role of thorough screening during the application process is for predictive models of loan approvals. Although approval databases might not monitor final defaults, understanding the factors that contribute to bad debts adds substantial context. Effective approval models could help lower the long-term probability of non-performing loans in addition to predicting immediate results.

Chapter 6

Analysis of the Variables

Variable Overview in (ASGMTLIB.TRAINING_DS)

This section provides an overview of the categorical, numerical, and bivariate variables in the `ASGMTLIB.TRAINING_DS` dataset. Variables marked with a checkmark (✓) were selected for analysis, while those with a cross (✗) were excluded.

Categorical Variable	**Analysed**
EMPLOYMENT	✗
FAMILY_MEMBERS	✓
GENDER	✗
LOAN_APPROVAL_STATUS	✗
LOAN_LOCATION	✗
MARITAL_STATUS	✓
QUALIFICATION	✓
SME_LOAN_ID_NO	✗
LOAN_HISTORY	✗

Table 2: Univariate categorical variables in `ASGMTLIB.TRAINING_DS`

Numerical Variable	**Analysed**
CANDIDATE_INCOME	✓
GUARANTEE_INCOME	✓
LOAN_AMOUNT	✓
LOAN_DURATION	✓
LOAN_HISTORY	✗

Table 3: Univariate numerical variables in `ASGMTLIB.TRAINING_DS`

Variable 1	Variable 2	Analysed
MARITAL_STATUS	FAMILY_MEMBERS	✓
MARITAL_STATUS	QUALIFICATION	✓
GENDER	EMPLOYMENT	✓

Table 4: Bivariate categorical variable pairs in `ASGMTLIB.TRAINING_DS`

Categorical	Numerical	Analysed
EMPLOYMENT	LOAN_AMOUNT	✓
QUALIFICATION	LOAN_DURATION	✓
GENDER	GUARANTEE_INCOME	✓

Table 5: Bivariate categorical vs numerical variable pairs in `ASGMTLIB.TRAINING_DS`

Univariate Analysis of Categorical Variables in ASGMTLIB.TRAINING_DS

Univariate Analysis of Categorical Variable: Family Members

SAS Code

```
/* Family Members: Frequency
    distribution and bar chart */
TITLE "Univariate Analysis -
    Frequency of Family Members";
PROC FREQ DATA = ASGMTLIB.
    TRAINING_DS;
    TABLE FAMILY_MEMBERS;
RUN;

TITLE "Univariate Analysis - Bar
    Chart of Family Members";
ODS GRAPHICS / RESET WIDTH=3.0in
    HEIGHT=4.0in IMAGEMAP;
PROC SGPLOT DATA = ASGMTLIB.
    TRAINING_DS;
    VBAR FAMILY_MEMBERS;
RUN;
```

Listing 2: SAS code for frequency table and bar chart of FAMILY_MEMBERS

Outputs

FAMILY_MEMBERS	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	345	57.60	345	57.60
1	102	17.03	447	74.62
2	101	16.86	548	91.49
3+	51	8.51	599	100.00
Frequency Missing = 15				

Table 6: Frequency distribution of FAMILY_MEMBERS

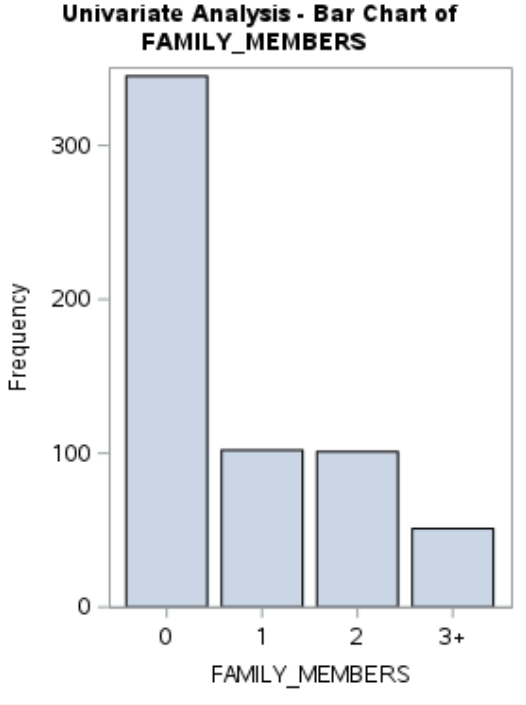

Figure 8: Bar chart of FAMILY_MEMBERS

Most applicants have no family members (57.6%), followed by one (17.0%) and two (16.9%). Only 8.5% lies in the category three-or-more, indicating that the distribution is skewed toward smaller households. This makes sense since many SME loan applicants apply on their own rather than as part of a larger household. There are also 15 missing records which should be flagged in the table so they don't get overlooked later.

Univariate Analysis of Categorical Variable: Marital Status

SAS Code

```
/* Marital Status: Frequency
    distribution and bar chart */
TITLE "Univariate Analysis -
    Frequency of Marital Status";
PROC FREQ DATA = ASGMTLIB.
    TRAINING_DS;
    TABLE MARITAL_STATUS;
RUN;

TITLE "Univariate Analysis - Bar
    Chart of Marital Status";
ODS GRAPHICS / RESET WIDTH=3.0in
    HEIGHT=4.0in IMAGEMAP;
PROC SGPLOT DATA = ASGMTLIB.
    TRAINING_DS;
    VBAR MARITAL_STATUS;
RUN;
```

Listing 3: SAS code for frequency table and bar chart of MARITAL_STATUS

Outputs

MARITAL_STATUS	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Married	398	65.14	398	65.14
Not Married	213	34.86	611	100.00
Frequency Missing = 3				

Table 7: Frequency distribution of MARITAL_STATUS

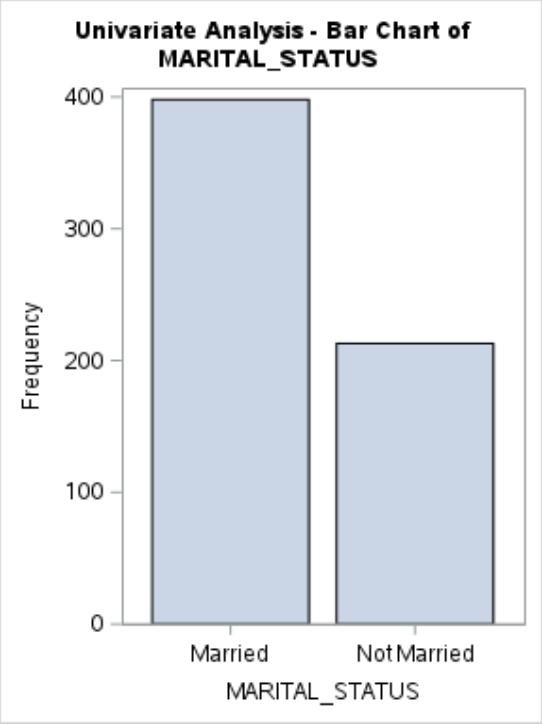

Figure 9: Bar chart of MARITAL_STATUS

The majority of applicants are married (65.1%) while 34.9% are not married. The distribution is skewed toward married applicants, as they make up almost two-thirds of the dataset. A possible reason is that married individuals may be seen as more financially stable or more likely to apply for loans with family responsibilities in mind. There are also 3 missing cases that need to be highlighted for future modeling.

Univariate Analysis of Categorical Variable: Qualification

SAS Code

```
/* Qualification: Frequency
    distribution and bar chart */
TITLE "Univariate Analysis -
    Frequency of Qualification";
PROC FREQ DATA = ASGMTLIB.
    TRAINING_DS;
    TABLE QUALIFICATION;
RUN;

TITLE "Univariate Analysis - Bar
    Chart of Qualification";
ODS GRAPHICS / RESET WIDTH=3.0in
    HEIGHT=4.0in IMAGEMAP;
PROC SGPLOT DATA = ASGMTLIB.
    TRAINING_DS;
    VBAR QUALIFICATION;
RUN;
```

Listing 4: SAS code for frequency table and bar chart of `QUALIFICATION`

Outputs

QUALIFICATION	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Graduate	480	78.18	480	78.18
Under Graduate	134	21.82	614	100.00

Table 8: Frequency distribution of `QUALIFICATION`

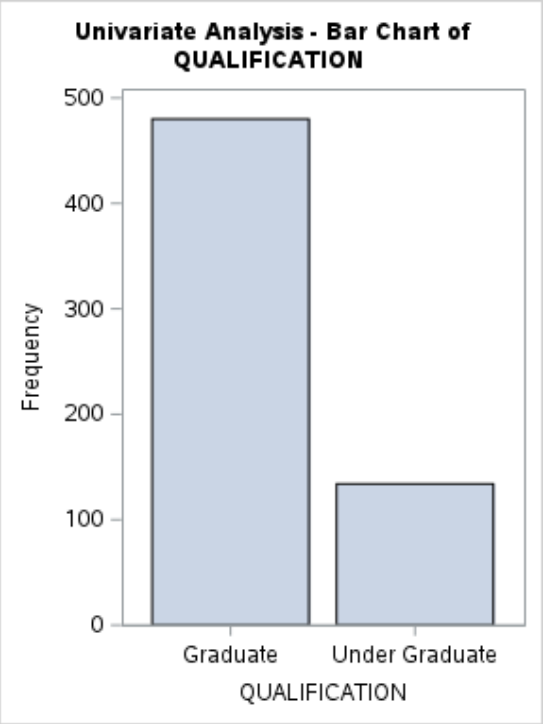

Figure 10: Bar chart of `QUALIFICATION`

Most of the applicants are graduates (78.2%) while undergraduates make up only 21.8%. The distribution a strong bias toward higher education. It's likely because graduates have more access to opportunities, resources, and financial literacy which makes them more inclined to apply for SME loans. No missing values were recorded here, so the variable is complete and easy to work with.

Univariate Analysis of Numerical Variables in ASGMTLIB.TRAINING_DS

Univariate Analysis of Numerical Variable: Candidate Income

SAS Code

```
/* Candidate Income: Summary
    statistics and histogram */
TITLE "Univariate Analysis - Summary
     of Candidate Income";
PROC MEANS DATA = ASGMTLIB.
    TRAINING_DS N NMISS MIN MAX MEAN
     MEDIAN STD;
    VAR CANDIDATE_INCOME;
RUN;

TITLE "Univariate Analysis -
    Histogram of Candidate Income";
PROC SGPLOT DATA = ASGMTLIB.
    TRAINING_DS;
    HISTOGRAM CANDIDATE_INCOME;
RUN;
```

Listing 5: SAS code for descriptive statistics and histogram of CANDIDATE_INCOME

Outputs

Analysis Variable : CANDIDATE_INCOME						
N	N Miss	Minimum	Maximum	Mean	Median	Std Dev
614	0	150.0000000	81000.00	5403.46	3812.50	6109.04

Table 9: Descriptive statistics of CANDIDATE_INCOME

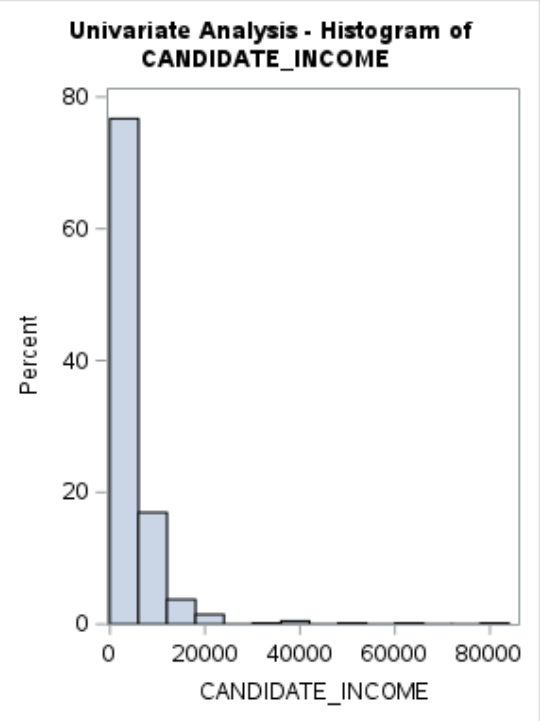

Figure 11: Histogram of CANDIDATE_INCOME

The histogram (Figure 11) shows that the majority of the applicants have a low income, with a very small number of them having a significantly higher income. This trend is also supported by the summary statistics table (Table 9), which displays a mean income of 5,403.46, which is higher than the median income of 3,812.50. This difference indicates that the distribution is skewed to the right, with the few high-income applicants acting as outliers.

Univariate Analysis of Numerical Variable: Guarantee Income

SAS Code

```
/* Guarantee Income: Summary
    statistics and histogram */
TITLE "Univariate Analysis - Summary
     of Guarantee Income";
PROC MEANS DATA = ASGMTLIB.
    TRAINING_DS N NMISS MIN MAX MEAN
     MEDIAN STD;
    VAR GUARANTEE_INCOME;
RUN;

TITLE "Univariate Analysis -
    Histogram of Guarantee Income";
PROC SGPLOT DATA = ASGMTLIB.
    TRAINING_DS;
    HISTOGRAM GUARANTEE_INCOME;
RUN;
```

Listing 6: SAS code for descriptive statistics and histogram of GUARANTEE_INCOME

Outputs

Analysis Variable : GUARANTEE_INCOME						
N	N Miss	Minimum	Maximum	Mean	Median	Std Dev
614	0	0	41667.00	1621.25	1188.50	2926.25

Table 10: Descriptive statistics of GUARANTEE_INCOME

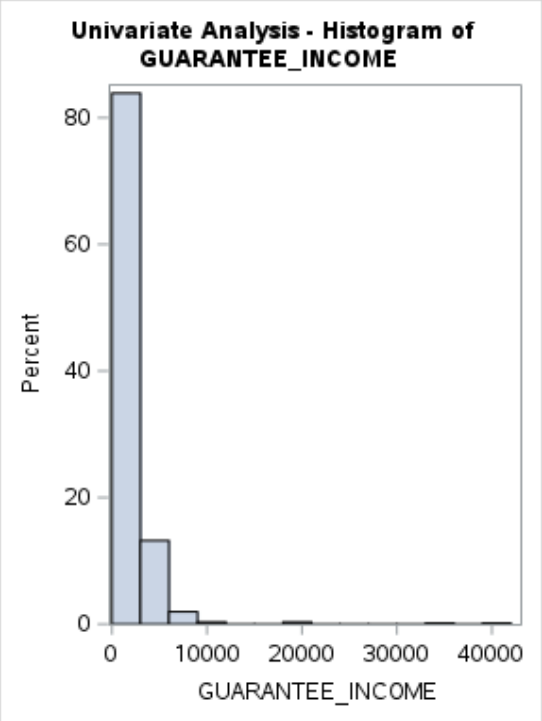

Figure 12: Histogram of GUARANTEE_INCOME

GUARANTEE_INCOME indicates a highly right-skewed distribution, meaning that most applicants have a low guarantee income, with a few having very high incomes. The histogram (Figure 12) shows a large number of values at the lowest income levels, with a long tail extending to the right. The summary table (Table 10) confirms this skewness, as the mean ($1,621.25) is higher than the median ($1,188.50), pulled upward by high-income outliers.

Univariate Analysis of Numerical Variable: Loan Amount

SAS Code

```
/* Loan Amount: Summary statistics
    and histogram */
TITLE "Univariate Analysis - Summary
     of Loan Amount";
PROC MEANS DATA = ASGMTLIB.
    TRAINING_DS N NMISS MIN MAX MEAN
     MEDIAN STD;
    VAR LOAN_AMOUNT;
RUN;

TITLE "Univariate Analysis -
    Histogram of Loan Amount";
PROC SGPLOT DATA = ASGMTLIB.
    TRAINING_DS;
    HISTOGRAM LOAN_AMOUNT;
RUN;
```

Listing 7: SAS code for descriptive statistics and histogram of LOAN_AMOUNT

Outputs

Analysis Variable : LOAN_AMOUNT						
N	N Miss	Minimum	Maximum	Mean	Median	Std Dev
592	22	9.0000000	700.0000000	146.4121622	128.0000000	85.5873252

Table 11: Descriptive statistics of LOAN_AMOUNT

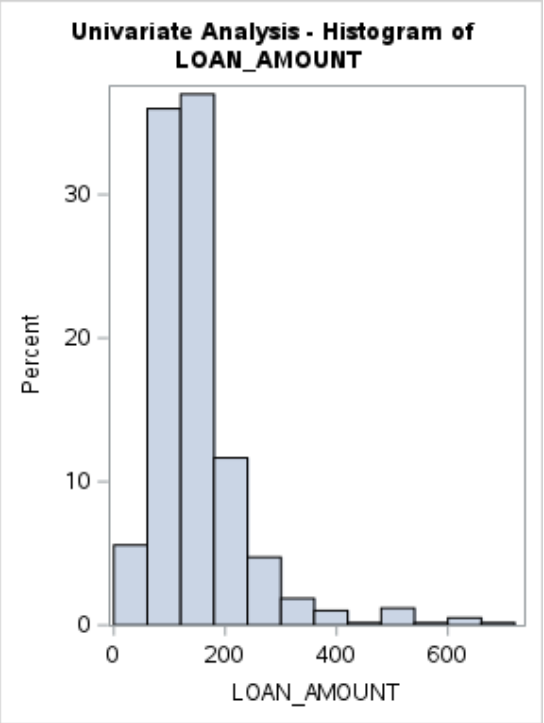

Figure 13: Histogram of LOAN_AMOUNT

LOAN_AMOUNT shows that the distribution is skewed to the right, with the majority of loan amounts falling between 100 and 200. This is clearly seen in the histogram (Figure 13), which shows a clear peak in this range. The summary table (Table 11) further supports this, as the mean ($146.41) is slightly higher than the median ($128.00), a sign of a right-skewed distribution caused by a few larger loan amounts. It also worth noting that there are 22 missing values LOAN_AMOUNT.

Univariate Analysis of Numerical Variable: Loan Duration

SAS Code

```
/* Loan Duration: Summary statistics
    and histogram */
TITLE "Univariate Analysis - Summary
    of Loan Duration";
PROC MEANS DATA = ASGMTLIB.
    TRAINING_DS N NMISS MIN MAX MEAN
    MEDIAN STD;
    VAR LOAN_DURATION;
RUN;

TITLE "Univariate Analysis -
    Histogram of Loan Duration";
PROC SGPLOT DATA = ASGMTLIB.
    TRAINING_DS;
    HISTOGRAM LOAN_DURATION;
RUN;
```

Listing 8: SAS code for descriptive statistics and histogram of LOAN_DURATION

Outputs

Analysis Variable : LOAN_DURATION						
N	N Miss	Minimum	Maximum	Mean	Median	Std Dev
600	14	12.0000000	480.0000000	342.0000000	360.0000000	65.1204099

Table 12: Descriptive statistics of LOAN_DURATION

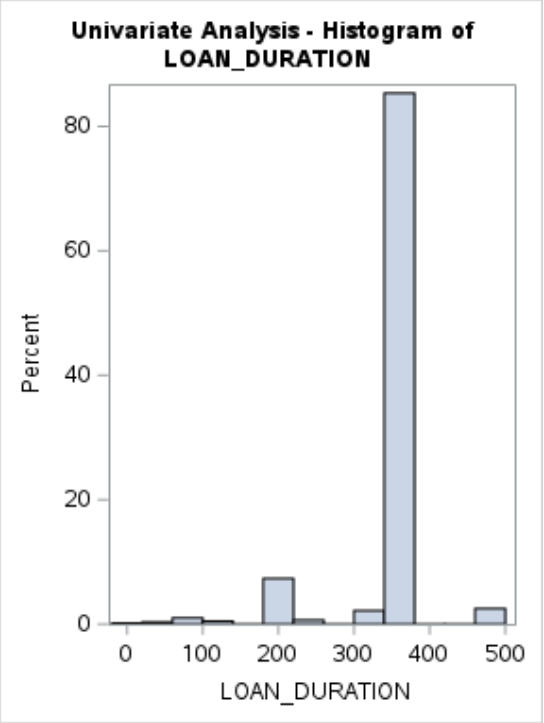

Figure 14: Histogram of LOAN_DURATION

LOAN_DURATION shows a strong focus of loans at around 360 days , with over 80% of the loans serving this duration. This is clearly seen by the tall bar in the histogram (Figure 14). The summary table (Table 12) shows that the median is 360 days, reflecting this concentration. The mean is 342 days, which is slightly lower than the median, suggesting a small number of short duration loans are pulling the mean down. It also worth noting that there are 14 missing values in LOAN_DURATION.

Bivariate Analysis of Categorical Variables in ASGMTLIB.TRAINING_DS

Bivariate Analysis of Categorical Variable: Gender vs Employment

SAS Code

```
/* Relationship between Gender and
    Employment */
TITLE1 "Bivariate Analysis of
    Categorical Variables";
TITLE2 "Gender vs Employment";
PROC FREQ DATA = ASGMTLIB.
    TRAINING_DS;
    TABLE GENDER * EMPLOYMENT /
        PLOTS = FREQPLOT(TWOWAY=
    STACKED SCALE=GROUPPCT);
RUN;
```

Listing 9: SAS code for cross-tabulation and stacked bar chart of GENDER vs EMPLOYMENT

Outputs

Table of GENDER by EMPLOYMENT

GENDER	EMPLOYMENT		
	No	Yes	Total
Female	89 15.64 85.58 18.13	15 2.64 14.42 19.23	104 18.28
Male	402 70.65 86.45 81.87	63 11.07 13.55 80.77	465 81.72
Total	491 86.29	78 13.71	569 100.00

Frequency Missing = 45

Table 13: Crosstab of GENDER by EMPLOYMENT

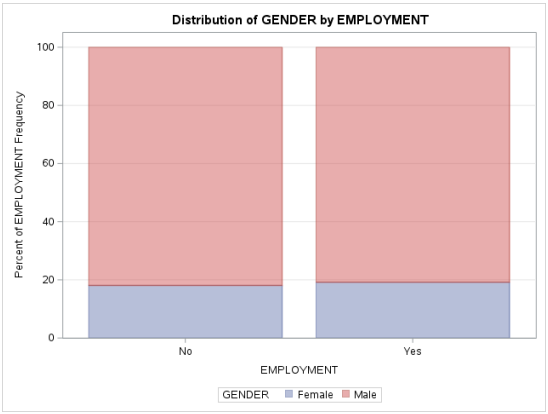

Figure 15: Stacked bar chart of GENDER by EMPLOYMENT

The bivariate analysis of GENDER and EMPLOYMENT show that employment rates are consistent between genders. Given on the table's row percentages (Table 13), 14.42% of females are employed, which is only marginally higher than 13.55% for males. The stacked bar chart (Figure 15) validates this, as the proportion of employed applicants (the "Yes" bar) has a equal gender composition to the non-employed applicants (the "No" bar). However, due to the much larger number of males present in ASGMTLIB.TRAINING_DS, they represent a larger percentage of the total employed population at 80.77% compared to females at 19.23%. It is important to note that the analysis was conducted on 569 applicants due to 45 missing values, a significant amount.

Bivariate Analysis of Categorical Variable: Marital Status vs Family Members

SAS Code

```
/* Relationship between Marital
    Status and Family Members */
TITLE1 "Bivariate Analysis of
    Categorical Variables";
TITLE2 "Marital Status vs Family
    Members";
PROC FREQ DATA = ASGMTLIB.
    TRAINING_DS;
    TABLE MARITAL_STATUS *
    FAMILY_MEMBERS /
        PLOTS = FREQPLOT(TWOWAY=
    STACKED SCALE=GROUPPCT);
RUN;
```

Listing 10: SAS code for cross-tabulation and stacked bar chart of MARITAL_STATUS vs FAMILY_MEMBERS

Outputs

Table of MARITAL_STATUS by FAMILY_MEMBERS

MARITAL_STATUS	FAMILY_MEMBERS 0	1	2	3+	Total
Married	174 29.05 44.62 50.43	79 13.19 20.26 77.45	93 15.53 23.85 92.08	44 7.35 11.28 86.27	390 65.11
Not Married	171 28.55 81.82 49.57	23 3.84 11.00 22.55	8 1.34 3.83 7.92	7 1.17 3.35 13.73	209 34.89
Total	345 57.60	102 17.03	101 16.86	51 8.51	599 100.00

Frequency Missing = 15

Table 14: Crosstab of MARITAL_STATUS by FAMILY_MEMBERS

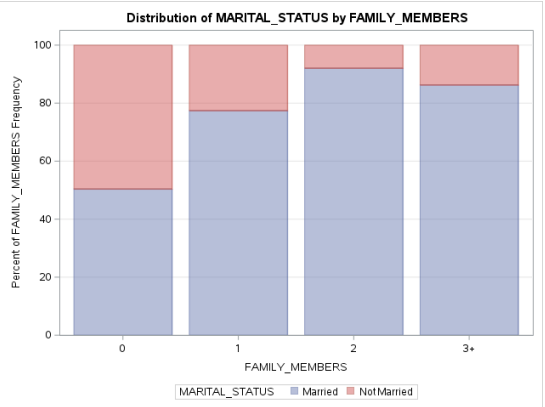

Figure 16: Stacked bar chart of MARITAL_STATUS by FAMILY_MEMBERS

Through the bivariate analysis we can see MARITAL_STATUS and FAMILY_MEMBERS has a clear relationship. Where, as the number of family members increases, so does the chance of a person being married. The stacked bar chart (Figure 16) demonstrates this trend, with the "Married" (blue) section growing as the FAMILY_MEMBERS category increases from 0 to 3+. The table's row percentages (Table 14) validates this, as applicants who are married rises from 50.43% for those with 0 family members to 86.27% for those with 3 or more family members. It is important to note that the analysis was conducted on 599 individuals due to 15 missing values, a moderate amount.

Bivariate Analysis of Categorical Variable: Marital Status vs Qualification

SAS Code

```
/* Relationship between Marital
    Status and Qualification */
TITLE1 "Bivariate Analysis of
    Categorical Variables";
TITLE2 "Marital Status vs
    Qualification";
PROC FREQ DATA = ASGMTLIB.
    TRAINING_DS;
    TABLE MARITAL_STATUS *
    QUALIFICATION /
        PLOTS = FREQPLOT(TWOWAY=
    STACKED SCALE=GROUPPCT);
RUN;
```

Listing 11: SAS code for cross-tabulation and stacked bar chart of MARITAL_STATUS vs QUALIFICATION

Outputs

Table of MARITAL_STATUS by QUALIFICATION

MARITAL_STATUS	QUALIFICATION		
	Graduate	Under Graduate	Total
Married	309 50.57 77.64 64.78	89 14.57 22.36 66.42	398 65.14
Not Married	168 27.50 78.87 35.22	45 7.36 21.13 33.58	213 34.86
Total	477 78.07	134 21.93	611 100.00

Frequency Missing = 3

Table 15: Crosstab of MARITAL_STATUS by QUALIFICATION

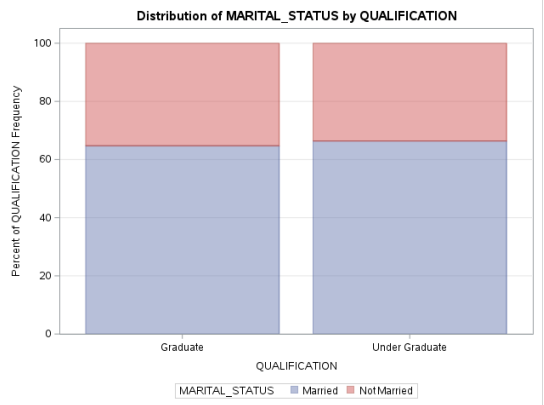

Figure 17: Stacked bar chart of MARITAL_STATUS by QUALIFICATION

The bivariate analysis of MARITAL_STATUS and QUALIFICATION shows a weak relationship. The stacked bar chart (Figure 17) illustrates that the proportion of married and not married individuals is nearly the same for both "Graduate" and "Under Graduate" qualifications. The table's row percentages (Table 15) validates this, as 77.64% of married individuals are graduates and 78.87% of not married individuals are graduates. This could imply that a person's qualification is not a determining factor of their marital status. It is important to note that the analysis was conducted on 611 individuals due to 3 missing values, a minimal amount.

Bivariate Analysis of Categorical vs. Numerical Variables in ASGMTLIB.TRAINING_DS

Bivariate Analysis of Categorical Variable & Continuous: Guarantee Income vs Gender

SAS Code

```
1  /* Compare Guarantee Income across
       Gender categories */
2  TITLE1 "Bivariate Analysis of
       Variables";
3  TITLE2 "Gender vs Guarantee Income";
4  PROC MEANS DATA = ASGMTLIB.
       TRAINING_DS;
5       CLASS GENDER;              /*
       Grouping by Gender (categorical)
        */
6       VAR GUARANTEE_INCOME;      /*
       Summarizing Guarantee Income (
       numeric) */
7  RUN;
```

Listing 12: SAS code for summarizing GUARANTEE_INCOME by GENDER

Outputs

Analysis Variable : GUARANTEE_INCOME						
GENDER	N Obs	N	Mean	Std Dev	Minimum	Maximum
Female	112	112	1108.01	4094.60	0	41667.00
Male	489	489	1742.93	2606.51	0	33837.00

Table 16: Descriptive statistics of GUARANTEE_INCOME by GENDER

The bivariate analysis of GUARANTEE_INCOME by GENDER shows a significant difference in income between the two genders. The mean guarantee income for males is 1,742.93, which is greater than the mean of 1,108.01 for females. The standard deviation for females (4,094.60) is also significantly larger than for males (2,606.51), which possibly means a greater spread and more extreme outliers in female guarantee incomes (Table 16).

Bivariate Analysis of Categorical Variable & Continuous: Loan Amount vs Employment

SAS Code

```
/* Compare Loan Amount across
    Employment categories */
TITLE1 "Bivariate Analysis of
    Variables";
TITLE2 "Employment vs Loan Amount";
PROC MEANS DATA = ASGMTLIB.
    TRAINING_DS;
    CLASS EMPLOYMENT;        /*
    Grouping by Employment (
    categorical) */
    VAR LOAN_AMOUNT;         /*
    Summarizing Loan Amount (numeric
    ) */
RUN;
```

Listing 13: SAS code for summarizing LOAN_AMOUNT by EMPLOYMENT

Outputs

Analysis Variable : LOAN_AMOUNT						
EMPLOYMENT	N Obs	N	Mean	Std Dev	Minimum	Maximum
No	500	482	141.7489627	79.7809192	9.0000000	700.0000000
Yes	82	79	172.0000000	108.6346500	25.0000000	650.0000000

Table 17: Descriptive statistics of LOAN_AMOUNT by EMPLOYMENT

The bivariate analysis of LOAN_AMOUNT by EMPLOYMENT illustrates that employed applicants are more likely to take out larger loan amounts. The mean loan amount for employed applicants is 172, which is greater than the mean loan amount of 141.75 for unemployed applicants. Furthermore, the standard deviation is higher for employed individuals, at 108.63, compared to 79.78 for unemployed applicants. This indicates a wider range of loan amounts within the employed group compared to the unemployed group,(Table 17).

Bivariate Analysis of Categorical Variable & Continuous: Loan Duration vs Qualification

SAS Code

```
/* Compare Loan Duration across Qualification categories */
TITLE1 "Bivariate Analysis of Variables";
TITLE2 "Qualification vs Loan Duration";
PROC MEANS DATA = ASGMTLIB.TRAINING_DS;
    CLASS QUALIFICATION;      /* Grouping by Qualification (categorical) */
    VAR LOAN_DURATION;        /* Summarizing Loan Duration (numeric) */
RUN;
```

Listing 14: SAS code for summarizing LOAN_DURATION by QUALIFICATION

Outputs

Analysis Variable : LOAN_DURATION						
QUALIFICATION	N Obs	N	Mean	Std Dev	Minimum	Maximum
Graduate	480	472	344.6694915	61.2996615	12.0000000	480.0000000
Under Graduate	134	128	332.1562500	77.0796129	36.0000000	480.0000000

Table 18: Descriptive statistics of LOAN_DURATION by QUALIFICATION

The bivariate analysis of LOAN_DURATION by QUALIFICATION reveals that there is a minor difference in the loan duration between graduates and under graduates. The mean loan duration for graduate candidates is 344.67 days, which is only marginally longer than the mean of 332.16 days for under graduate candidates. The standard deviations are also nearly identical, indicating that both groups have a similar spread in their loan durations (Table 18).

Variable Overview in (ASGMTLIB.TESTING_DS)

This section provides an overview of the categorical, numerical, and bivariate variables in the ASGMTLIB.TESTING_DS dataset. Variables marked with a checkmark (✓) were analysed, while those with a cross (✗) were excluded due to time constraints.

Categorical Variable	Analysed
EMPLOYMENT	✗
FAMILY_MEMBERS	✓
GENDER	✓
LOAN_APPROVAL_STATUS	✗
LOAN_LOCATION	✗
MARITAL_STATUS	✓
QUALIFICATION	✓
SME_LOAN_ID_NO	✗
LOAN_HISTORY	✗

Table 19: Univariate categorical variables in ASGMTLIB.TESTING_DS

Numerical Variable	Analysed
CANDIDATE_INCOME	✓
GUARANTEE_INCOME	✓
LOAN_AMOUNT	✓
LOAN_DURATION	✓
LOAN_HISTORY	✗

Table 20: Univariate numerical variables in ASGMTLIB.TESTING_DS

Variable 1	Variable 2	Analysed
MARITAL_STATUS	FAMILY_MEMBERS	✓
MARITAL_STATUS	QUALIFICATION	✓
GENDER	EMPLOYMENT	✓

Table 21: Bivariate categorical variable pairs in ASGMTLIB.TESTING_DS

Categorical	Numerical	Analysed
EMPLOYMENT	LOAN_AMOUNT	✓
QUALIFICATION	LOAN_DURATION	✓
GENDER	GUARANTEE_INCOME	✓

Table 22: Bivariate categorical vs numerical variable pairs in `ASGMTLIB.TESTING_DS`

Univariate Analysis of Categorical Variables in ASGMTLIB.TESTING_DS

Univariate Analysis of Categorical Variables: Family Members, Marital Status, Qualification & Gender

SAS Code

```
/* Enable compile-time notes for macros */
OPTIONS MCOMPILENOTE=ALL;

/* Macro to generate frequency table for a categorical variable */
%MACRO UVACATE_VAR(ptitle, pds, pvar);
    TITLE "&PTITLE";
    PROC FREQ DATA=&PDS;
        TABLE &PVAR;
    RUN;
%MEND UVACATE_VAR;

/* Macro calls: categorical variables */
%UVACATE_VAR(UNIVARIATE ANALYSIS OF THE
    CATEGORICAL VARIABLE: Family_Members,
     ASGMTLIB.TESTING_DS, FAMILY_MEMBERS)
    ;
%UVACATE_VAR(UNIVARIATE ANALYSIS OF THE
    CATEGORICAL VARIABLE: Marital_Status,
     ASGMTLIB.TESTING_DS, MARITAL_STATUS)
    ;
%UVACATE_VAR(UNIVARIATE ANALYSIS OF THE
    CATEGORICAL VARIABLE: Qualification,
     ASGMTLIB.TESTING_DS, QUALIFICATION);
%UVACATE_VAR(UNIVARIATE ANALYSIS OF THE
    CATEGORICAL VARIABLE: Gender,
    ASGMTLIB.TESTING_DS, GENDER);
```

Listing 15: SAS macro and calls for categorical frequencies in ASGMTLIB.TESTING_DS

Outputs

FAMILY_MEMBERS	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	200	56.02	200	56.02
1	58	16.25	258	72.27
2	59	16.53	317	88.80
3+	40	11.20	357	100.00
Frequency Missing = 10				

Table 23: Frequency distribution of FAMILY_MEMBERS

GENDER	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Female	70	19.66	70	19.66
Male	286	80.34	356	100.00
Frequency Missing = 11				

Table 24: Frequency distribution of GENDER

MARITAL_STATUS	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Married	233	63.49	233	63.49
Not Married	134	36.51	367	100.00

Table 25: Frequency distribution of MARITAL_STATUS

QUALIFICATION	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Graduate	283	77.11	283	77.11
Under Graduate	84	22.89	367	100.00

Table 26: Frequency distribution of QUALIFICATION

The analysis of the applicants shows a clear trend of concentration within a few categories. A significant amount of the applicants are male, making up over 80% of the testing dataset (Table 24). Applicants are predominantly married, accounting for nearly two-thirds of the candidates (Table 25), and a large majority hold a Graduate qualification, representing 77.11% (Table). The frequency table for family members also shows a concentration, with more than half the candidates (56.02%) having no family members (Table 23). It is important to note that a small number of missing values exist for both the Gender and Family Members variables.

Univariate Analysis of Numerical Variables in ASGMTLIB.TESTING_DS

Univariate Analysis of Numerical Variables: Candidate Income, Guarantee Income, Loan Amount & Loan Duration

SAS Code

```
/* Enable compile-time notes for macros */
OPTIONS MCOMPILENOTE=ALL;
/* Macro to generate summary statistic for a continuous variable */
%MACRO UVA_CONTI_VAR(ptitle, pds, pvar);
    TITLE "&PTITLE";
    PROC MEANS DATA=&pds N NMISS MIN MAX MEAN MEDIAN STD;
        VAR &pvar;
    RUN;
%MEND UVA_CONTI_VAR;
/* Macro calls: continuous variables */
%UVA_CONTI_VAR(UNIVARIATE ANALYSIS OF THE CONTINUOUS VARIABLE: Candidate_Income, ASGMTLIB.TESTING_DS, CANDIDATE_INCOME);
%UVA_CONTI_VAR(UNIVARIATE ANALYSIS OF THE CONTINUOUS VARIABLE: Guarantee_Income, ASGMTLIB.TESTING_DS, GUARANTEE_INCOME);
%UVA_CONTI_VAR(UNIVARIATE ANALYSIS OF THE CONTINUOUS VARIABLE: Loan_Amount, ASGMTLIB.TESTING_DS, LOAN_AMOUNT);
%UVA_CONTI_VAR(UNIVARIATE ANALYSIS OF THE CONTINUOUS VARIABLE: Loan_Duration, ASGMTLIB.TESTING_DS, LOAN_DURATION);
```

Listing 16: SAS macro and calls for continuous variables in ASGMTLIB.TESTING_DS

Outputs

Analysis Variable : CANDIDATE_INCOME						
N	N Miss	Minimum	Maximum	Mean	Median	Std Dev
367	0	0	72529.00	4805.60	3786.00	4910.69

Table 27: Descriptive statistics of CANDIDATE_INCOME

Analysis Variable : GUARANTEE_INCOME						
N	N Miss	Minimum	Maximum	Mean	Median	Std Dev
367	0	0	24000.00	1569.58	1025.00	2334.23

Table 28: Descriptive statistics of GUARANTEE_INCOME

Analysis Variable : LOAN_AMOUNT						
N	N Miss	Minimum	Maximum	Mean	Median	Std Dev
362	5	28.0000000	550.0000000	136.1325967	125.0000000	61.3666524

Table 29: Descriptive statistics of LOAN_AMOUNT

Analysis Variable : LOAN_DURATION						
N	N Miss	Minimum	Maximum	Mean	Median	Std Dev
361	6	6.0000000	480.0000000	342.5373961	360.0000000	65.1566434

Table 30: Descriptive statistics of LOAN_DURATION

Based on the descriptive statistics, the mean candidate income is 4,805.60, which is significantly higher than the average guarantor income of 1,569.58, which implies that applicants tend to have a greater financial capacity than their guarantors (Table 27, Table 28). The average loan amount is 136.13, with a typical loan duration of about 342.54 days (Table 29, Table 30). It is also important to note that data for LOAN_AMOUNT and LOAN_DURATION is not complete, with 5 and 6 missing values respectively, while the income data for both candidates and guarantors are complete with 367 observations each.

Bivariate Analysis of Categorical Variables in ASGMTLIB.TESTING_DS

SAS Code

```
/* Enable compile-time notes for macros */
OPTIONS MCOMPILENOTE=ALL;

/* Macro: BVA_CATE_CATE
   Purpose : Cross-tabulation and stacked frequency plot for two categorical
    variables
   Parameters:
       PTITLE1 - First line of title
       PTITLE2 - Second line of title (variables being compared)
       PDS     - Dataset
       PVAR1   - First categorical variable
       PVAR2   - Second categorical variable */
%MACRO BVA_CATE_CATE(ptitle1, ptitle2, pds, pvar1, pvar2);
    TITLE1 "&ptitle1";
    TITLE2 "&ptitle2";
    PROC FREQ DATA=&pds;
        TABLE &pvar1 * &pvar2 /
            PLOTS=FREQPLOT(TWOWAY=STACKED SCALE=GROUPPCT);
    RUN;
%MEND BVA_CATE_CATE;

/* Macro calls */
%BVA_CATE_CATE(BIVARIATE ANALYSIS OF THE VARIABLES:, Categorical Variable -
    Marital_Status vs Categorical Variable - Family_Members, ASGMTLIB.TESTING_DS
    , MARITAL_STATUS, FAMILY_MEMBERS);
%BVA_CATE_CATE(BIVARIATE ANALYSIS OF THE VARIABLES:, Categorical Variable -
    Marital_Status vs Categorical Variable - Qualification, ASGMTLIB.TESTING_DS,
     MARITAL_STATUS, QUALIFICATION);
%BVA_CATE_CATE(BIVARIATE ANALYSIS OF THE VARIABLES:, Categorical Variable -
    Gender vs Categorical Variable - Employment, ASGMTLIB.TESTING_DS, GENDER,
    EMPLOYMENT);
```

Listing 17: SAS macro for bivariate analysis of categorical variables in ASGMTLIB.TESTING_DS

Bivariate Analysis of Categorical Variable: Marital Status vs Family Members

Output

Table of MARITAL_STATUS by FAMILY_MEMBERS					
	FAMILY_MEMBERS				
MARITAL_STATUS	0	1	2	3+	Total
Married	95 26.61 41.67 47.50	45 12.61 19.74 77.59	53 14.85 23.25 89.83	35 9.80 15.35 87.50	228 63.87
Not Married	105 29.41 81.40 52.50	13 3.64 10.08 22.41	6 1.68 4.65 10.17	5 1.40 3.88 12.50	129 36.13
Total	200 56.02	58 16.25	59 16.53	40 11.20	357 100.00
Frequency Missing = 10					

Table 31: Cross-tabulation of MARITAL_STATUS by FAMILY_MEMBERS (TESTING_DS)

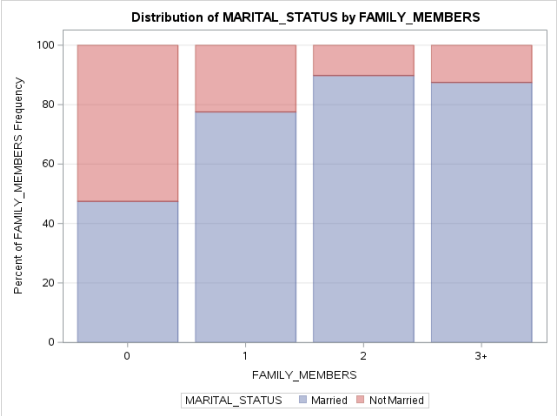

Figure 18: Stacked bar chart of MARITAL_STATUS vs FAMILY_MEMBERS (TESTING_DS)

Through the bivariate analysis we can see MARITAL_STATUS and FAMILY_MEMBERS has a clear relationship similiar to the training dataset. The stacked bar chart (Figure 18) validates this, demonstrating that as the number of family members increases, the numbers of applicants who are married also increases . The table (Table 31) validates this as well, showing that the percentage of married individuals rises from 47.50% for those with 0 family members to 87.50% for those with 3 or more. It is important to note that the analysis was conducted on 357 applicants due to 10 missing values, a minimal amount.

Bivariate Analysis of Categorical Variable: Marital Status vs Qualifaction

Output

Table of MARITAL_STATUS by QUALIFICATION			
	QUALIFICATION		
MARITAL_STATUS	Graduate	Under Graduate	Total
Married	176 47.96 75.54 62.19	57 15.53 24.46 67.86	233 63.49
Not Married	107 29.16 79.85 37.81	27 7.36 20.15 32.14	134 36.51
Total	283 77.11	84 22.89	367 100.00

Table 32: Cross-tabulation of MARITAL_STATUS by QUALIFICATION (TESTING_DS)

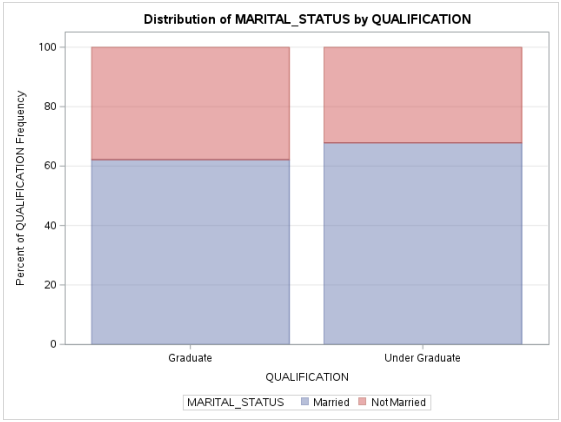

Figure 19: Stacked bar chart of MARITAL_STATUS vs QUALIFICATION (TESTING_DS)

The bivariate analysis of MARITAL_STATUS and QUALIFICATION displays a weak relationship. The stacked bar chart (Figure 19) shows that the percentage of married and not married individuals is nearly the same for both ”Graduate” and ”Under Graduate” qualifications. The table (Table 32) also reveals the same, with 75.54% of married applicants being graduates and 79.85% of not married applicants also being graduates. This implies that a person’s qualification does not have a strong influence on their marital status. This analysis was conducted on the all 367 individuals, with no missing values.

Bivariate Analysis of Categorical Variable: Gender vs Employment

Output

Table of GENDER by EMPLOYMENT			
	EMPLOYMENT		
GENDER	No	Yes	Total
Female	63 18.92 94.03 21.14	4 1.20 5.97 11.43	67 20.12
Male	235 70.57 88.35 78.86	31 9.31 11.65 88.57	266 79.88
Total	298 89.49	35 10.51	333 100.00
Frequency Missing = 34			

Table 33: Cross-tabulation of GENDER by EMPLOYMENT (TESTING_DS)

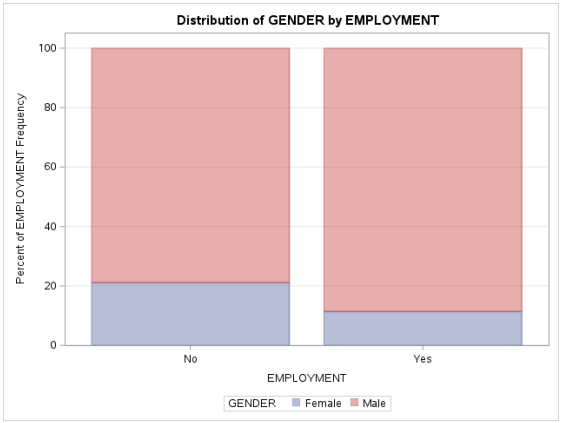

Figure 20: Stacked bar chart of GENDER vs EMPLOYMENT (TESTING_DS)

The bivariate analysis of GENDER and EMPLOYMENT displays a difference in employment rates between male and female applicants. The stacked bar chart (Figure 20) reveals that employed applicants have a much higher share of males than females compared to the unemployed applicants. Based on the cross-tabulation (Table 33), the employment rate for males (11.65%) is significantly higher than for females (5.97%). It is also important to note that males represent the majority of the total employed population with 88.57%, compared to 11.43% for females. It is important to note that the analysis was conducted on 333 applicants due to 34 missing values, a substantial amount.

Bivariate Analysis of Categorical vs. Numerical Variables in ASGMTLIB.TESTING_DS

SAS Code

```
/* Enable compile-time notes for macros */
OPTIONS MCOMPILENOTE=ALL;

/* Macro: BVA_CATE_CONTI
   Purpose : Summary statistics of a continuous variable grouped by a
    categorical variable
   Parameters:
       PTITLE1 - First line of title
       PTITLE2 - Second line of title (variables being compared)
       PDS     - Dataset
       PCATE   - Categorical variable
       PCONTI  - Continuous variable */
%MACRO BVA_CATE_CONTI(ptitle1, ptitle2, pds, pcate, pconti);
    TITLE1 "&ptitle1";
    TITLE2 "&ptitle2";
    PROC MEANS DATA=&pds;
        CLASS &pcate;
        VAR &pconti;
    RUN;
%MEND BVA_CATE_CONTI;

/* Macro calls */
%BVA_CATE_CONTI(BIVARIATE ANALYSIS OF THE VARIABLES:, Categorical Variable -
    Employment vs Continuous Variable - Loan_Amount, ASGMTLIB.TESTING_DS,
    EMPLOYMENT, LOAN_AMOUNT);
%BVA_CATE_CONTI(BIVARIATE ANALYSIS OF THE VARIABLES:, Categorical Variable -
    Qualification vs Continuous Variable - Loan_Duration, ASGMTLIB.TESTING_DS,
    QUALIFICATION, LOAN_DURATION);
%BVA_CATE_CONTI(BIVARIATE ANALYSIS OF THE VARIABLES:, Categorical Variable -
    Gender vs Continuous Variable - Guarantee_Income, ASGMTLIB.TESTING_DS,
    GENDER, GUARANTEE_INCOME);
```

Listing 18: SAS macro and calls for categorical vs. numerical analysis in ASGMTLIB.TESTING_DS

Bivariate Analysis of Categorical Variable & Continuous: Guarantee Income vs Gender

Output

Analysis Variable : GUARANTEE_INCOME						
GENDER	N Obs	N	Mean	Std Dev	Minimum	Maximum
Female	70	70	1171.96	1979.82	0	11666.00
Male	286	286	1670.87	2433.94	0	24000.00

Table 34: Descriptive statistics of `GUARANTEE_INCOME` by `GENDER` in `ASGMTLIB.TESTING_DS`

The bivariate analysis of `GUARANTEE_INCOME` by `GENDER` (Table 34) displays a difference in income between the male and female. Males have a larger mean guarantee income of 1,670.87 compared to 1,171.96 for females. However, the standard deviation for males is also larger, at 2,433.94, implying a wider range of income among males compared to females, who only have a standard deviation of 979.82.

Bivariate Analysis of Categorical Variable & Continuous: Loan Amount vs Employment

Output

Analysis Variable : LOAN_AMOUNT						
EMPLOYMENT	N Obs	N	Mean	Std Dev	Minimum	Maximum
No	307	302	133.7218543	57.9927432	28.0000000	460.0000000
Yes	37	37	150.1891892	84.6797358	50.0000000	550.0000000

Table 35: Descriptive statistics of `LOAN_AMOUNT` by `EMPLOYMENT` in `ASGMTLIB.TESTING_DS`

The bivariate analysis of `LOAN_AMOUNT` by `EMPLOYMENT` (Table 35) reveals that employed applicants tend to take out much larger loans. The mean loan amount for employed applicants is 150.19, compared to a mean loan amount of 133.72 for unemployed applicants. In addition, the standard deviation is higher for employed applicants (84.68) compared to the unemployed applicants (57.99), which could imply a greater variation in loan amounts for employed applicants.

Bivariate Analysis of Categorical Variable & Continuous: Loan Duration vs Qualification

Output

Analysis Variable : LOAN_DURATION						
QUALIFICATION	N Obs	N	Mean	Std Dev	Minimum	Maximum
Graduate	283	279	340.2867384	66.9216513	6.0000000	480.0000000
Under Graduate	84	82	350.1951220	58.4884581	36.0000000	480.0000000

Table 36: Descriptive statistics of `LOAN_DURATION` by `QUALIFICATION` in `ASGMTLIB.TESTING_DS`

The bivariate analysis of `LOAN_DURATION` by `QUALIFICATION` (Table 36) demonstrate a marginal difference in the loan duration between graduates and under graduates. The mean loan duration for graduate applicants is 340.29 days, which is only a little bite smaller than the mean of 350.20 days for under graduate applicant. The standard deviations are also nearly identical, implying that both have a similar spread in their loan durations.

Chapter 7

Data Cleaning and Imputation

Backup & Restore Utilities for ASGMTLIB.TRAINING_DS and ASGMTLIB.TESTING_DS

SAS Code

```
/* Enable compile-time notes for macros */
OPTIONS MCOMPILENOTE=ALL;
/* Macro: BKP_MAKE
   Purpose : Create a backup copy of a dataset ( _BK)
   Param   : PDS - Dataset */
%MACRO BKP_MAKE(PDS);
PROC SQL;
  CREATE TABLE &PDS._BK AS
  SELECT * FROM &PDS;
QUIT;
%MEND BKP_MAKE;
/* Macro: BKP_RESTORE
   Purpose : Restore the dataset from its _BK copy
   Param   : PDS - Dataset */
%MACRO BKP_RESTORE(PDS);
PROC SQL;
  CREATE TABLE &PDS AS
  SELECT * FROM &PDS._BK;
QUIT;
%MEND BKP_RESTORE;
/* Make backups before any imputation runs --- */
%BKP_MAKE(ASGMTLIB.TRAINING_DS);
%BKP_MAKE(ASGMTLIB.TESTING_DS);
```

Listing 19: SAS macros for creating and restoring dataset backups

Output

Figure 21: Created backup table ASGMTLIB.TRAINING_DS_BK

Figure 22: Created backup table ASGMTLIB.TESTING_DS_BK

Before any imputation, both the training and testing datasets were backed up into separate copies (_BK). This ensures that the original datasets remain intact and can be restored if needed. Figures 21 and 22 confirm the successful creation of backup tables.

Categorical Imputation of Variables in ASGMTLIB.TRAINING_DS and ASGMTLIB.TESTING_DS

SAS Code

```
/* Enable compile-time notes for macros */
OPTIONS MCOMPILENOTE=ALL;
/* Macro: IMPUTE_CATE_MODE
   Purpose : Impute a categorical variable's missing/blank values with its MODE.
   Params  : PTITLEVAR - Pretty variable name for TITLES
             PDS       - Dataset
             PVAR      - Categorical variable name
             PSTATS    - Output dataset to store counts/mode */
%MACRO IMPUTE_CATE_MODE(PTITLEVAR, PDS, PVAR, PSTATS);
  /*Imputing values - STEP 1*/
  TITLE1 "STEP 1: LIST THE DETAILS OF THE LOAN APPLICANTS";
  TITLE2 "WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT &PTITLEVAR DATA";
  FOOTNOTE "----End----";
  PROC SQL;
    SELECT *
    FROM &PDS t
    WHERE (t.&PVAR IS MISSING OR t.&PVAR = '');
  QUIT;
  /*Imputing values - STEP 2*/
  TITLE1 "STEP 2: COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS";
  TITLE2 "WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT &PTITLEVAR DATA";
  FOOTNOTE "----End----";
  PROC SQL;
    SELECT COUNT(*) LABEL = "Total number of loan applicants"
    FROM &PDS t
    WHERE (t.&PVAR IS MISSING OR t.&PVAR = '');
  QUIT;
  /*Imputing values - STEP 3 */
  TITLE1 "STEP 3: FIND THE MOD VALUE AND SAVE THE STATISTICS IN A TEMPORARY DATASET";
  FOOTNOTE "----End----";
  PROC SQL;
    CREATE TABLE &PSTATS AS
    SELECT t.&PVAR AS &PVAR, COUNT(*) AS counts
    FROM &PDS t
    WHERE ( (t.&PVAR IS NOT MISSING) OR (t.&PVAR NE '') )
    GROUP BY t.&PVAR;
  QUIT;
  /*Imputing values - STEP 4: impute the missing values found...*/
  TITLE1 "STEP 4: IMPUTE THE MISSING VALUES FOUND...";
  FOOTNOTE "----End----";
  PROC SQL;
    UPDATE &PDS
      SET &PVAR = (
        SELECT to.&PVAR
        FROM &PSTATS to
        WHERE to.counts = (SELECT MAX(ti.counts) FROM &PSTATS ti)
      )
    WHERE (&PVAR IS MISSING OR &PVAR = '');
  QUIT;
  /*Imputing values - STEP 5(AI)*/
  TITLE1 "STEP 5(AI): LIST THE DETAILS OF THE LOAN APPLICANTS";
  TITLE2 "WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT &PTITLEVAR DATA";
  FOOTNOTE "----End----";
  PROC SQL;
    SELECT *
    FROM &PDS t
    WHERE (t.&PVAR IS MISSING OR t.&PVAR = '');
  QUIT;
/*Imputing values - STEP 6(AI)*/
  TITLE1 "STEP 6(AI): COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS";
  TITLE2 "WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT &PTITLEVAR DATA";
  FOOTNOTE "----End----";
  PROC SQL;
    SELECT COUNT(*) LABEL = "Total number of loan applicants"
    FROM &PDS t
    WHERE (t.&PVAR IS MISSING OR t.&PVAR = '');
  QUIT;
%MEND IMPUTE_CATE_MODE;
```

Listing 20: Mode-based imputation macro for (ASGMTLIB.TRAINING_DS and ASGMTLIB.TESTING_DS)

SAS Code

```
/* TRAINING: MARITAL_STATUS */
%IMPUTE_CATE_MODE(Marital_Status, ASGMTLIB.TRAINING_DS, MARITAL_STATUS,
    ASGMTLIB.TRAINING_STATS_DS);

/* TRAINING: FAMILY_MEMBERS */
%IMPUTE_CATE_MODE(Family_Members,  ASGMTLIB.TRAINING_DS, FAMILY_MEMBERS,
     ASGMTLIB.TRAINING_STATS_DS);

/* TESTING: MARITAL_STATUS */
%IMPUTE_CATE_MODE(Family_Members, ASGMTLIB.TESTING_DS,  FAMILY_MEMBERS,
    ASGMTLIB.TESTING_STATS_DS);
```

Listing 21: Macro calls for imputing categorical variables for (ASGMTLIB.TRAINING_DS and ASGMTLIB.TESTING_DS

Categorical Imputation

Categorical variables were imputed using the mode strategy. The SAS macro followed these steps: (i) list applicants with missing or blank values, (ii) count total missing, (iii) build a temporary statistics table to identify the mode, (iv) replace missing values with the mode using an UPDATE query, and (v) re-list and re-count post-imputation to confirm that no missing values remained.

Imputation of `MARITAL_STATUS` in `ASGMTLIB.TRAINING_DS`

Output

STEP 1: LIST THE DETAILS OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Marital_Status DATA

SME_LOAN_ID_NO	GENDER	MARITAL_STATUS	FAMILY_MEMBERS	QUALIFICATION	EMPLOYMENT	CANDIDATE_INCOME	GUARANTEE_INCOME	LOAN_AMOUNT	LOAN_DURATION	LOAN_HISTORY	LOAN_LOCATION	LOAN_APPROVAL_STATUS
LP001357	Male			Graduate	No	3816	754	160	360	1	City	Y
LP001760	Male			Graduate	No	4758	0	158	480	1	Town	Y
LP002393	Female			Graduate	No	10047	0	.	240	1	Town	Y

----End----

STEP 2: COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Marital_Status DATA

Total number of loan applicants
3

----End----

STEP 5(AI): LIST THE DETAILS OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Marital_Status DATA

----End----

STEP 6(AI): COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Marital_Status DATA

Total number of loan applicants
0

----End----

Figure 23: Training set: Pre- and post-imputation checks for `MARITAL_STATUS` (list+count before; list+count after)

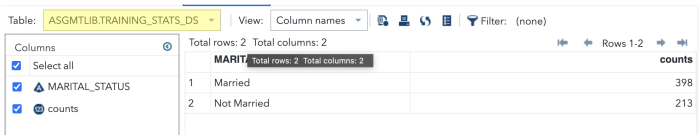

Figure 24: Training set: Temporary stats table for `MARITAL_STATUS` (mode counts)

Figure 25: Training set: SAS log confirming successful imputation ("3 rows were updated").

Based on the outputs, `MARITAL_STATUS` was missing for 3 applicants (as shown in the pre-imputation checks, Fig. 23). The mode of this variable was identified as `Married` in the temporary statistics table (Table 24). A total of 3 rows were updated, confirmed in the SAS log (Fig. 25), and no applicants remained with missing values after imputation (Fig. 23).

Imputation of FAMILY_MEMBERS in ASGMTLIB.TRAINING_DS

Output

STEP 1: LIST THE DETAILS OF THE LOAN APPLICANTS WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Family_Members DATA

SME_LOAN_ID_NO	GENDER	MARITAL_STATUS	FAMILY_MEMBERS	QUALIFICATION	EMPLOYMENT	CANDIDATE_INCOME	GUARANTEE_INCOME	LOAN_AMOUNT	LOAN_DURATION	LOAN_HISTORY	LOAN_LOCATION	LOAN_APPROVAL_STATUS
LP001350	Male	Married		Graduate	No	13650	0	.	360	1	City	Y
LP001357	Male	Married		Graduate	No	3816	754	160	360	1	City	Y
LP001426	Male	Married		Graduate	No	5667	2667	180	360	1	Village	Y
LP001754	Male	Married		Under Graduate	Yes	4735	0	138	360	1	City	N
LP001760	Male	Married		Graduate	No	4758	0	158	480	1	Town	Y
LP001945	Female	Not Married		Graduate	No	5417	0	143	480	0	City	N
LP001972	Male	Married		Under Graduate	No	2875	1750	105	360	1	Town	Y
LP002100	Male	Not Married		Graduate	No	2833	0	71	360	1	City	Y
LP002106	Male	Married		Graduate	Yes	5503	4490	70	.	1	Town	Y
LP002130	Male	Married		Under Graduate	No	3523	3230	152	360	0	Village	N
LP002144	Female	Not Married		Graduate	No	3813	0	116	180	1	City	Y
LP002393	Female	Married		Graduate	No	10047	0	.	240	1	Town	Y
LP002682	Male	Married		Under Graduate	No	3074	1800	123	360	0	Town	N
LP002847	Male	Married		Graduate	No	5116	1451	165	360	0	City	N
LP002943	Male	Not Married		Graduate	No	2987	0	88	360	0	Town	N

----End----

STEP 2: COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Family_Members DATA

Total number of loan applicants
15

----End----

STEP 5(AI): LIST THE DETAILS OF THE LOAN APPLICANTS WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Family_Members DATA

----End----

STEP 6(AI): COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Family_Members DATA

Total number of loan applicants
0

----End----

Figure 26: Training set: Pre- and post-imputation checks for FAMILY_MEMBERS (list+count before; list+count after)

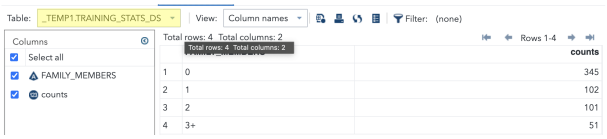

Figure 27: Training set: Temporary stats table for MARITAL_STATUS (mode counts)

Figure 28: Training set: SAS log confirming successful imputation ("rows were updated").

Based on the outputs, FAMILY_MEMBERS was missing for 15 applicants (as indicated in the pre-imputation checks, Fig. 26). The mode was found to be 0 family members in the statistics table (Table 27). The SAS log confirmed that 15 rows were updated (Fig. 28), and the after imputation (See Fig. 26) verified that no records remained with missing values.

Imputation of FAMILY_MEMBERS in ASGMT.TESTING_DS

Output

STEP 1: LIST THE DETAILS OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Family_Members DATA

SME_LOAN_ID_NO	GENDER	MARITAL_STATUS	FAMILY_MEMBERS	QUALIFICATION	EMPLOYMENT	CANDIDATE_INCOME	GUARANTEE_INCOME	LOAN_AMOUNT	LOAN_DURATION	LOAN_HISTORY	LOAN_LOCATION	LOAN_APPROVAL_STATUS
LP001237	Male	Married		Under Graduate	No	4163	1475	162	360	1	City	
LP001366	Female	Not Married		Graduate	No	3250	0	95	360	1	Town	
LP001587	Male	Married		Graduate	No	4082	0	93	360	1	Town	
LP001769		Not Married		Graduate	No	3333	1250	110	360	1	Town	
LP002111	Male	Married		Graduate	No	3016	1300	100	360	.	City	
LP002360	Male	Married		Graduate	No	10000	0	.	360	1	City	
LP002385	Male	Married		Graduate	No	3863	0	70	300	1	Town	
LP002441	Male	Not Married		Graduate	No	3579	3308	138	360	.	Town	
LP002654	Female	Not Married		Graduate	Yes	14987	0	177	360	1	Village	
LP002754	Male	Not Married		Graduate	No	2066	2108	104	84	1	City	

----End----

STEP 2: COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Family_Members DATA

Total number of loan applicants
10

----End----

STEP 5(AI): LIST THE DETAILS OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Family_Members DATA

----End----

STEP 6(AI): COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Family_Members DATA

Total number of loan applicants
0

----End----

Figure 29: Testing set: Pre- and post-imputation checks for FAMILY_MEMBERS (list+count before; list+count after)

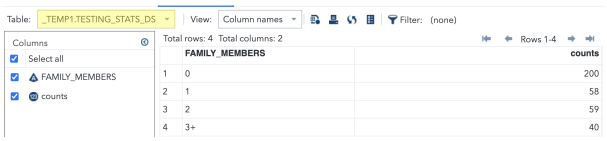

Figure 30: Testing set: Temporary stats table for MARITAL_STATUS (mode counts)

Figure 31: Testing set: SAS log confirming successful imputation ("rows were updated").

Based on the outputs, FAMILY_MEMBERS was missing for 5 applicants (as shown in the pre-imputation steps, Fig. 29). The mode of this variable was again 0, identified in the statistics table (Table 30). The SAS log reported that 5 rows were successfully updated (Fig. 31), and this was further confirmed in the after imputation check (Fig. 29).

Continuous Imputation of Variables in ASGMTLIB.TRAINING_DS and ASGMTLIB.TESTING_DS

SAS Code

```
/* Enable compile-time notes for macros */
OPTIONS MCOMPILENOTE=ALL;

/* Macro: IMPUTE_CONT_MEAN
   Purpose : Impute missing values of a continuous variable using MEAN
   Params  : PTITLEVAR - Pretty variable name for titles (e.g., Loan_Amount)
             PDS       - Dataset (libref.member)
             PVAR      - Continuous variable name
*/
%MACRO IMPUTE_CONT_MEAN(PTITLEVAR, PDS, PVAR);

  /*************************************************************
   STEP 1: List applicants with missing values (continuous)
  *************************************************************/
  TITLE1 "STEP 1: LIST THE DETAILS OF THE LOAN APPLICANTS";
  TITLE2 "WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT &PTITLEVAR DATA";
  FOOTNOTE "----End----";
  PROC SQL;
    SELECT *
    FROM &PDS t
    WHERE (t.&PVAR IS MISSING OR t.&PVAR = .);
  QUIT;

  /*************************************************************
   STEP 2: Count applicants with missing values (continuous)
  *************************************************************/
  TITLE1 "STEP 2: COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS";
  TITLE2 "WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT &PTITLEVAR DATA";
  FOOTNOTE "----End----";
  PROC SQL;
    SELECT COUNT(*) LABEL='Number of Loan Applicants'
    FROM &PDS t
    WHERE (t.&PVAR IS MISSING OR t.&PVAR = .);
  QUIT;

  /*************************************************************
   STEP 3: Impute missing values with MEAN
  *************************************************************/
  PROC STDIZE DATA=&PDS REPONLY METHOD=MEAN OUT=&PDS;
    VAR &PVAR;
  QUIT;

  /*************************************************************
   STEP 4(AI): Post-imputation listing (should be empty)
  *************************************************************/
  TITLE1 "STEP 4(AI): LIST THE DETAILS OF THE LOAN APPLICANTS";
  TITLE2 "WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT &PTITLEVAR DATA";
  FOOTNOTE "----End----";
  PROC SQL;
    SELECT *
    FROM &PDS t
    WHERE (t.&PVAR IS MISSING OR t.&PVAR = .);
  QUIT;

  /*************************************************************
   STEP 5(AI): Post-imputation count (should be zero)
  *************************************************************/
  TITLE1 "STEP 5(AI): COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS";
  TITLE2 "WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT &PTITLEVAR DATA";
  FOOTNOTE "----End----";
  PROC SQL;
    SELECT COUNT(*) LABEL='Number of Loan Applicants'
    FROM &PDS t
    WHERE (t.&PVAR IS MISSING OR t.&PVAR = .);
  QUIT;

%MEND IMPUTE_CONT_MEAN;
```

Listing 22: Mean-based imputation macro for (ASGMTLIB.TRAINING_DS and ASGMTLIB.TESTING_DS)

SAS Code

```
/* TRAINING: LOAN_AMOUNT */
%IMPUTE_CONT_MEAN(Loan_Amount, ASGMTLIB.TRAINING_DS, LOAN_AMOUNT);

/* TRAINING: LOAN_DURATION */
%IMPUTE_CONT_MEAN(Loan_Duration, ASGMTLIB.TRAINING_DS, LOAN_DURATION);

/* TESTING: LOAN_AMOUNT */
%IMPUTE_CONT_MEAN(Loan_Amount, ASGMTLIB.TESTING_DS, LOAN_AMOUNT);
```

Listing 23: Macro calls for imputing continuos variables for (`ASGMTLIB.TRAINING_DS` and `ASGMTLIB.TESTING_DS`

Continuous Imputation

Continuous variables were imputed using the mean strategy. The SAS macro followed these steps: (i) list applicants with missing values, (ii) count total missing, (iii) replace missing values with the mean using `PROC STDIZE`, and (iv) re-list and re-count post-imputation to confirm that all missing values had been handled.

Imputation of LOAN_AMOUNT in ASGMTLIB.TRAINING_DS

Output

STEP 1: LIST THE DETAILS OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Loan_Amount DATA

SME_LOAN_ID_NO	GENDER	MARITAL_STATUS	FAMILY_MEMBERS	QUALIFICATION	EMPLOYMENT	CANDIDATE_INCOME	GUARANTEE_INCOME	LOAN_AMOUNT	LOAN_DURATION	LOAN_HISTORY	LOAN_LOCATION	LOAN_APPROVAL_STATUS
LP001002	Male	Not Married	0	Graduate	No	5849	0	.	360	1	City	Y
LP001106	Male	Married	0	Graduate	No	2275	2067	.	360	1	City	Y
LP001213	Male	Married	1	Graduate	No	4945	0	.	360	0	Village	N
LP001266	Male	Married	1	Graduate	Yes	2395	0	.	360	1	Town	Y
LP001326	Male	Not Married	0	Graduate		6782	0	.	360	.	City	N
LP001350	Male	Married	0	Graduate	No	13650	0	.	360	1	City	Y
LP001356	Male	Married	0	Graduate	No	4652	3583	.	360	1	Town	Y
LP001392	Female	Not Married	1	Graduate	Yes	7451	0	.	360	1	Town	Y
LP001449	Male	Not Married	0	Graduate	No	3865	1640	.	360	1	Village	Y
LP001682	Male	Married	3+	Under Graduate	No	3992	0	.	180	1	City	N
LP001922	Male	Married	0	Graduate	No	20667	0	.	360	1	Village	N
LP001990	Male	Not Married	0	Under Graduate	No	2000	0	.	360	1	City	N
LP002054	Male	Married	2	Under Graduate	No	3601	1590	.	360	1	Village	Y
LP002113	Female	Not Married	3+	Under Graduate	No	1830	0	.	360	0	City	N
LP002243	Male	Married	0	Under Graduate	No	3010	3136	.	360	0	City	N
LP002393	Female	Married	0	Graduate	No	10047	0	.	240	1	Town	Y
LP002401	Male	Married	0	Graduate	No	2213	1125	.	360	1	City	Y
LP002533	Male	Married	2	Graduate	No	2947	1603	.	360	1	City	N
LP002697	Male	Not Married	0	Graduate	No	4680	2087	.	360	1	Town	N
LP002778	Male	Married	2	Graduate	Yes	6633	0	.	360	0	Village	N
LP002784	Male	Married	1	Under Graduate	No	2492	2375	.	360	1	Village	Y
LP002960	Male	Married	0	Under Graduate	No	2400	3800	.	180	1	City	N

----End----

STEP 2: COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Loan_Amount DATA

Number of Loan Applicants
22

----End----

STEP 4(AI): LIST THE DETAILS OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Loan_Amount DATA

----End----

STEP 5(AI): COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Loan_Amount DATA

Number of Loan Applicants
0

----End----

Figure 32: Training: pre-/post-imputation checks for LOAN_AMOUNT (Steps 1–5(AI))

Based on the outputs, LOAN_AMOUNT was missing for 22 applicants (as shown in the pre-imputation checks, Fig. 32). The variable was imputed using the mean, replacing all missing values. After imputation, no applicants remained with missing loan amount values (Fig. 32).

Imputation of LOAN_DURATION in ASGMTLIB.TRAINING_DS

Output

STEP 1: LIST THE DETAILS OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Loan_Duration DATA

SME_LOAN_ID_NO	GENDER	MARITAL_STATUS	FAMILY_MEMBERS	QUALIFICATION	EMPLOYMENT	CANDIDATE_INCOME	GUARANTEE_INCOME	LOAN_AMOUNT	LOAN_DURATION	LOAN_HISTORY	LOAN_LOCATION	LOAN_APPROVAL_STATUS
LP001041	Male	Married	0	Graduate		2600	3500	115	.	1	City	Y
LP001109	Male	Married	0	Graduate	No	1828	1330	100	.	0	City	N
LP001136	Male	Married	0	Under Graduate	Yes	4695	0	96	.	1	City	Y
LP001137	Female	Not Married	0	Graduate	No	3410	0	88	.	1	City	Y
LP001250	Male	Married	3+	Under Graduate	No	4755	0	95	.	0	Town	N
LP001391	Male	Married	0	Under Graduate	No	3572	4114	152	.	0	Village	N
LP001574	Male	Married	0	Graduate	No	3707	3166	182	.	1	Village	Y
LP001669	Female	Not Married	0	Under Graduate	No	1907	2365	120	.	1	City	Y
LP001749	Male	Married	0	Graduate	No	7578	1010	175	.	1	Town	Y
LP001770	Male	Not Married	0	Under Graduate	No	3189	2598	120	.	1	Village	Y
LP002106	Male	Married	0	Graduate	Yes	5503	4490	70	.	1	Town	Y
LP002188	Male	Not Married	0	Graduate	No	5124	0	124	.	0	Village	N
LP002357	Female	Not Married	0	Under Graduate	No	2720	0	80	.	0	City	N
LP002362	Male	Married	1	Graduate	No	7250	1667	110	.	0	City	N

----End----

STEP 2: COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Loan_Duration DATA

Number of Loan Applicants
14

----End----

STEP 4(AI): LIST THE DETAILS OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Loan_Duration DATA

----End----

STEP 5(AI): COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Loan_Duration DATA

Number of Loan Applicants
0

----End----

Figure 33: Training: pre-/post-imputation checks for LOAN_DURATION (Steps 1–5(AI))

Based on the outputs, LOAN_DURATION was missing for 14 applicants (as shown in the pre-imputation checks, Fig. 33). The variable was imputed using the mean, ensuring all missing values were replaced. After imputation, no applicants remained with missing loan duration values (Fig. 33).

Imputation of LOAN_AMOUNT in ASGMTLIB.TESTING_DS

Output

STEP 1: LIST THE DETAILS OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Loan_Amount DATA

SME_LOAN_ID_NO	GENDER	MARITAL_STATUS	FAMILY_MEMBERS	QUALIFICATION	EMPLOYMENT	CANDIDATE_INCOME	GUARANTEE_INCOME	LOAN_AMOUNT	LOAN_DURATION	LOAN_HISTORY	LOAN_LOCATION	LOAN_APPROVAL_STATUS
LP001415	Male	Married	1	Graduate	No	3413	4053	.	360	1	Town	
LP001542	Female	Married	0	Graduate	No	2262	0	.	480	0	Town	
LP002057	Male	Married	0	Under Graduate	No	13083	0	.	360	1	Village	
LP002360	Male	Married	0	Graduate	No	10000	0	.	360	1	City	
LP002593	Male	Married	1	Graduate	No	8333	4000	.	360	1	City	

----End----

STEP 2: COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Loan_Amount DATA

Number of Loan Applicants
5

----End----

STEP 4(AI): LIST THE DETAILS OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Loan_Amount DATA

----End----

STEP 5(AI): COUNT THE TOTAL NUMBER OF THE LOAN APPLICANTS
WHO SUBMITTED THEIR LOAN APPLICATIONS WITHOUT Loan_Amount DATA

Number of Loan Applicants
0

----End----

Figure 34: Testing: pre-/post-imputation checks for LOAN_AMOUNT (Steps 1–5(AI))

Based on the outputs, LOAN_AMOUNT was missing for 5 applicants (as shown in the pre-imputation checks, Fig. 34). The mean was used to impute these values, and after imputation, no applicants remained with missing values in loan amount (Fig. 34).

Chapter 8

Model Creation

Logistic Regression Model Creation in ASGMTLIB.TRAINING_DS

SAS Code

```
/* Train logistic regression model */
TITLE "Logistic Regression - Model Creation (Training)";
PROC LOGISTIC DATA=ASGMTLIB.TRAINING_DS
              OUTMODEL=ASGMTLIB.TRAINING_DS_LR_MODEL;   /* save model */
    CLASS
        GENDER
        MARITAL_STATUS
        FAMILY_MEMBERS
        QUALIFICATION
        EMPLOYMENT
        LOAN_HISTORY
        LOAN_LOCATION
    ;
    MODEL LOAN_APPROVAL_STATUS =
        GENDER
        MARITAL_STATUS
        FAMILY_MEMBERS
        QUALIFICATION
        EMPLOYMENT
        LOAN_HISTORY
        LOAN_LOCATION
        CANDIDATE_INCOME
        GUARANTEE_INCOME
        LOAN_AMOUNT
        LOAN_DURATION
    ;
    /* Save predicted probabilities to an output dataset */
    OUTPUT OUT=ASGMTLIB.TRAINING_OUT_DS
           P=PRED_PROB;   /* predicted probability of modeled event */
RUN;
```

Listing 24: SAS code for training the logistic regression model on ASGMT.TRAINING_DS

Output

Model Information	
Data Set	ASGMTLIB.TRAINING_DS
Response Variable	LOAN_APPROVAL_STATUS
Number of Response Levels	2
Model	binary logit
Optimization Technique	Fisher's scoring

Figure 35: Model Information for logistic regression in `ASGMT.TRAINING_DS`

Figure 35 summarizes the modeling setup: binary logistic regression on `LOAN_APPROVAL_STATUS` using Fisher's scoring. Categorical predictors were specified in the CLASS statement, ensuring correct handling of nominal variables.

Output

Model Convergence Status
Convergence criterion (GCONV=1E-8) satisfied.

Figure 36: Convergence status of logistic regression (criterion satisfied)

The optimization converged successfully as shown in Figure 36, where the convergence criterion (GCONV=1E-8) is satisfied. This ensures stable coefficient estimates and valid inference.

Output

Number of Observations Read	614
Number of Observations Used	526

Figure 37: Number of observations read and used in model estimation

Figure 37 shows that 614 observations were read, of which 526 were used. This implies that 88 records were excluded, most likely due to missing values. Minimizing exclusions allows us to preserve statistical power.

Output

Model Fit Statistics		
Criterion	Intercept Only	Intercept and Covariates
AIC	663.975	↓ 501.945
SC	668.240	↓ 565.925
-2 Log L	661.975	471.945

Figure 38: Model fit statistics (AIC, SC, and $-2\log L$)

Model fit improved markedly as shown in Figure 38. AIC decreased from 663.98 to 501.95, SC from 668.24 to 565.93, and $-2\log L$ from 661.98 to 471.95. These reductions indicate that the full model fits substantially better than the intercept-only model.

Output

Type 3 Analysis of Effects			
Effect	DF	Wald Chi-Square	Pr > ChiSq
GENDER	1	0.4878	0.4849
MARITAL_STATUS	1	3.7881	0.0516
FAMILY_MEMBERS	3	1.7620	0.6232
QUALIFICATION	1	3.3770	0.0661
EMPLOYMENT	1	0.1640	0.6855
LOAN_HISTORY	1	81.0825	<.0001
LOAN_LOCATION	2	12.5217	0.0019
CANDIDATE_INCOME	1	0.0010	0.9743
GUARANTEE_INCOME	1	1.2406	0.2654
LOAN_AMOUNT	1	2.3906	0.1221
LOAN_DURATION	1	0.0705	0.7906

Figure 39: Type 3 analysis of effects (Wald chi-square tests)

The Wald chi-square tests in Figure 39 show that `LOAN_HISTORY` ($p < 0.0001$) and `LOAN_LOCATION` ($p = 0.0019$) are significant predictors. `MARITAL_STATUS` ($p = 0.0516$) and `QUALIFICATION` ($p = 0.0661$) are borderline significant, while all other predictors were not statistically significant at $\alpha = 0.05$.

Results Summary

The logistic regression converged and showed improved fit compared to the intercept-only baseline (AIC 663.98 $\rightarrow$ 501.95; SC 668.24 $\rightarrow$ 565.93; $-2\log L$ 661.98 $\rightarrow$ 471.95). `LOAN_HISTORY` and `LOAN_LOCATION` emerged as strong predictors of approval, while marital status and qualification were borderline. Other demographic or financial variables did not reach significance, indicating potential improvement from further cleaning.

Logistic Regression Scoring on ASGMTLIB.TESTING_DS

SAS Code — Scoring the Testing Set

```
/*Predict the Loan Approval Status using the model created*/
TITLE "Logistic Regression - Scoring (Testing)";
PROC LOGISTIC INMODEL=ASGMTLIB.TRAINING_DS_LR_MODEL;   /* load saved model */
    SCORE DATA=ASGMTLIB.TESTING_DS
          OUT=ASGMTLIB.TESTING_LAS_PRED_TP086704_DS;
QUIT;

TITLE "Report - Loan Approval Status Predicted";
FOOTNOTE "----End----";
/* Display the details of the loan approval status predicted */
PROC SQL;
    SELECT *
    FROM ASGMTLIB.TESTING_LAS_PRED_TP086704_DS;
QUIT;
```

Listing 25: SAS code for scoring and predicting loan approval status on ASGMTLIB.TESTING_DS

Outputs

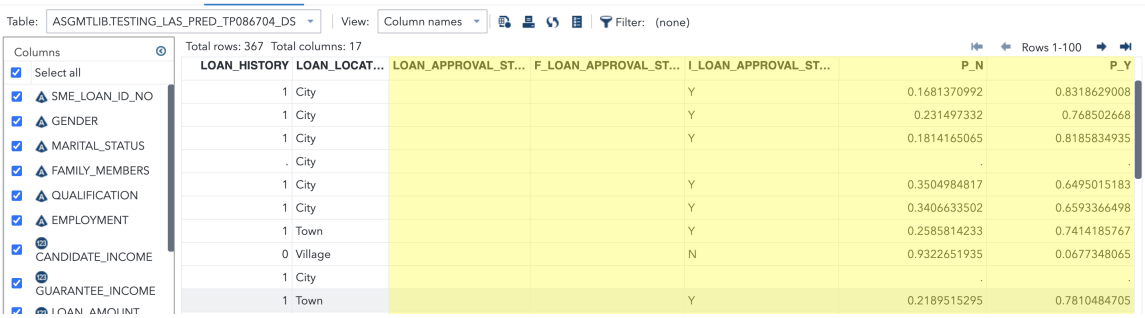

Figure 40: Predicted loan approval status results for ASGMTLIB.TESTING_DS

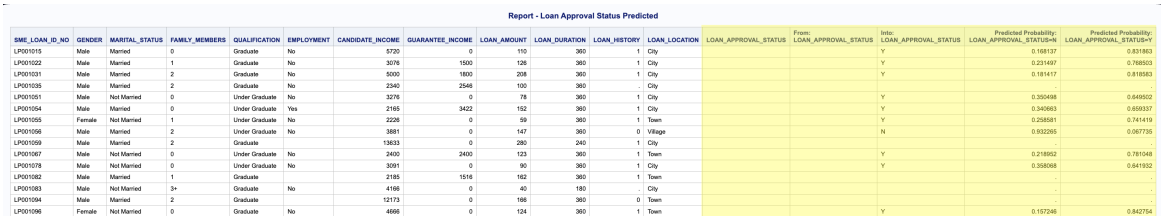

Table 37: Detailed prediction output for ASGMTLIB.TESTING_DS

The logistic regression model was applied to the testing dataset (ASGMT.TESTING_DS), and predicted loan approval statuses were generated. Figure 40 summarizes the predicted outcomes for each applicant, while Table 37 provides a more detailed view of the predictions. These results confirm that the trained logistic regression model can be successfully deployed to score new applications and support decision making for loan applications.

Part 3

Report Generation, Data Visualization & Conclusion

Chapter 9

Report Generation

Report Generation of Predicted Loan Approval Status in ASGMTLIB.TESTING_DS

SAS Code

```
/* Close any open ODS destinations (clean start) */
ods html close;
ods pdf  close;
ods listing close;

/* PDF output location & basic options */
ods pdf file="/home/u64179868/sasuser.v94/DAP_FT_JUN_2025_ASGMT_TP086704/LAS_REPORT_TP086704.pdf";
options nodate;

/* Report titles/footers */
title1 "Predicted Bank Loan Approval Status Report";
title2 "APU, TPM";
footnote " End of Report ";

/* Main report */
ods proclabel "Loan Approval Predictions";
proc report data= ASGMTLIB.TESTING_LAS_PRED_TP086704_DS nowindows;
    by SME_LOAN_ID_NO;

    /* Column definitions (labels chosen for clean PDF headings) */
    define SME_LOAN_ID_NO   / group  "Loan ID";
    define GENDER           / group  "Gender";
    define MARITAL_STATUS   / group  "Marital Status";
    define FAMILY_MEMBERS   / group  "Family Members";
    define CANDIDATE_INCOME / group  "Monthly Income";
    define GUARANTEE_INCOME / group  "Co-Applicant's Income";
    define LOAN_AMOUNT      / group  "Loan Amount";
    define LOAN_DURATION    / group  "Loan Duration";
    define LOAN_HISTORY     / group  "Loan History";
    define LOAN_LOCATION    / group  "Loan Location";
run;
```

Listing 26: SAS code used to generate the PDF report with PROC REPORT

Output

Figure 41: Folder view showing the generated PDF `LAS_REPORT_TP086704.pdf`

Figure 42: ODS log confirming PDF was written to disk (highlighted path)

Output

Figure 43: Results table (sample row) with predicted probabilities

Figure 44: Results table (second sample row) with predicted probabilities

Output

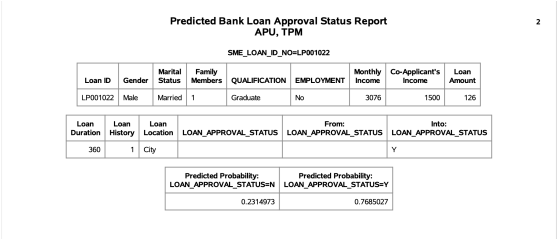

Figure 45: PDF report page showing `SME_LOAN_ID_NO = LP001022`

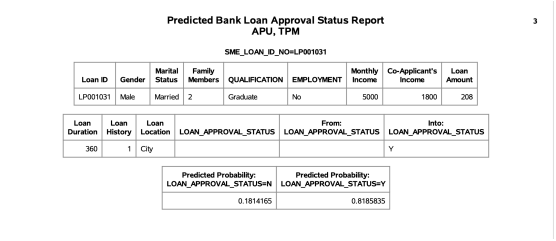

Figure 46: PDF report page showing `SME_LOAN_ID_NO = LP001031`

The code in Listing 26 closes open ODS destinations, opens an ODS PDF target, and renders a grouped `PROC REPORT` with clear column labels. The folder view and ODS note (Figures 41–42) verify that `LAS_REPORT_TP086704.pdf` was created at the specified path. Sample rows from the Results window (Figures 43–44) illustrate the predicted probabilities stored in the scored dataset. Figures 45–46 show the final PDF layout per loan ID, including both probabilities P_N and P_Y.

Complex Report Generation of Predicted Loan Approval Status in ASGMTLIB.TESTING_DS

SAS Code

```
/* Enable compile-time notes for macros */
OPTIONS MCOMPILENOTE=ALL;

/* Macro: MACRO_RPT_3
   Purpose : Generate a complex report filtered by loan location (CITY/VILLAGE/TOWN)
             and create a location-specific output table with derived fields.
   Params  : PLOAN_LOCATION - Location filter (CITY / VILLAGE / TOWN)
*/
%MACRO MACRO_RPT_3(PLOAN_LOCATION);

    /* Map location parameter to dataset suffix  */
    %LOCAL SFX OUTDS;
    %IF %UPCASE(&PLOAN_LOCATION) = CITY    %THEN %LET SFX = CTY;
    %ELSE %IF %UPCASE(&PLOAN_LOCATION) = VILLAGE %THEN %LET SFX = VLG;
    %ELSE %IF %UPCASE(&PLOAN_LOCATION) = TOWN    %THEN %LET SFX = TWN;
    %ELSE %DO;
        %PUT ERROR: (MACRO_RPT_3) INVALID LOAN_LOCATION=&PLOAN_LOCATION. USE CITY, VILLAGE, OR TOWN.;
        %RETURN;
    %END;

    %LET OUTDS = ASGMTLIB.TESTING_LAS_PRED_TP086704_&SFX;

    /* Drop output table if it already exists */
    %IF %SYSFUNC(EXIST(&OUTDS)) %THEN %DO;
        PROC SQL;
            DROP TABLE &OUTDS;
        QUIT;
    %END;

    /* Create specific location output table */
    PROC SQL;
        CREATE TABLE &OUTDS AS
        SELECT
            L.SME_LOAN_ID_NO AS Loan_ID,

            /* Gender short name */
            CASE
                WHEN UPCASE(L.GENDER) = 'MALE'   THEN 'M'
                WHEN UPCASE(L.GENDER) = 'FEMALE' THEN 'F'
                ELSE 'N/A'
            END AS Gender_short_name,

            /* Remarks based on gender + loan history */
            CASE
                WHEN UPCASE(L.GENDER) = 'MALE'   AND L.LOAN_HISTORY = 1 THEN
                     'He is a good applicant. For he has settled his past loan(s) on time.'
                WHEN UPCASE(L.GENDER) = 'MALE'   AND L.LOAN_HISTORY = 0 THEN
                     'He is not a good applicant. For he has not settled his past loan(s) on time.'
                WHEN UPCASE(L.GENDER) = 'FEMALE' AND L.LOAN_HISTORY = 1 THEN
                     'She is a good applicant. For she has settled her past loan(s) on time.'
                WHEN UPCASE(L.GENDER) = 'FEMALE' AND L.LOAN_HISTORY = 0 THEN
                     'She is not a good applicant. For she has not settled her past loan(s) on time.'
                ELSE 'N/A'
            END AS Remarks
        FROM ASGMTLIB.TESTING_LAS_PRED_TP086704_DS L
        WHERE UPCASE(L.LOAN_LOCATION) = "%UPCASE(&PLOAN_LOCATION)";
    QUIT;

    /* Print Final Report */
    TITLE1 "COMPLEX REPORT: %SYSFUNC(DATE(),WORDDATE.)";
    TITLE2 "DETAILS OF THE LOAN APPLICANTS  &PLOAN_LOCATION";
    PROC PRINT DATA=&OUTDS;
        VAR Loan_ID Gender_short_name Remarks;
    RUN;

%MEND MACRO_RPT_3;

/*Location: City*/
%MACRO_RPT_3(CITY);
/*Location: Village*/
%MACRO_RPT_3(VILLAGE);
/*Location: Town*/
%MACRO_RPT_3(TOWN);
```

Output

COMPLEX REPORT: September 3, 2025
DETAILS OF THE LOAN APPLICANTS — CITY

Obs	Loan_ID	Gender_short_name	Remarks
1	LP001015	M	He is a good applicant. For he has settled his past loan(s) on time.
2	LP001022	M	He is a good applicant. For he has settled his past loan(s) on time.
3	LP001031	M	He is a good applicant. For he has settled his past loan(s) on time.
4	LP001035	M	N/A
5	LP001051	M	He is a good applicant. For he has settled his past loan(s) on time.

Figure 47: Complex report filtered to `CITY`: derived fields *Gender_short_name* and *Remarks* populated from `GENDER` and `LOAN_HISTORY`.

COMPLEX REPORT: September 3, 2025
DETAILS OF THE LOAN APPLICANTS — TOWN

Obs	Loan_ID	Gender_short_name	Remarks
1	LP001055	F	She is a good applicant. For she has settled her past loan(s) on time.
2	LP001067	M	He is a good applicant. For he has settled his past loan(s) on time.
3	LP001082	M	He is a good applicant. For he has settled his past loan(s) on time.
4	LP001094	M	He is not a good applicant. For he has not settled his past loan(s) on time.
5	LP001096	F	She is a good applicant. For she has settled her past loan(s) on time.

Figure 48: Complex report filtered to `TOWN`: shows both positive and negative *Remarks* narratives reflecting `LOAN_HISTORY` = 1 vs. 0.

COMPLEX REPORT: September 3, 2025
DETAILS OF THE LOAN APPLICANTS — VILLAGE

Obs	Loan_ID	Gender_short_name	Remarks
1	LP001056	M	He is not a good applicant. For he has not settled his past loan(s) on time.
2	LP001153	M	He is not a good applicant. For he has not settled his past loan(s) on time.
3	LP001317	F	She is a good applicant. For she has settled her past loan(s) on time.
4	LP001347	F	She is not a good applicant. For she has not settled her past loan(s) on time.
5	LP001361	M	He is not a good applicant. For he has not settled his past loan(s) on time.

Figure 49: Complex report filtered to `VILLAGE`: concise listing with derived fields consistent with `CITY`/`TOWN` logic.

The complex reports for `CITY`, `TOWN`, and `VILLAGE` (Figs. 47–49) share the same structure, listing loan applicants with derived fields `Gender_short_name` (M/F) and narrative-style `Remarks` based on loan history. Applicants with `LOAN_HISTORY=1` are described as "good applicants" who repaid on time, while those with `LOAN_HISTORY=0` are flagged as "not good applicants." The `TOWN` report (Fig. 48) clearly illustrates both positive and negative cases across genders, demonstrating how the reporting logic captures different applicant profiles.

Chapter 10

Data Visualisation

Stacked Bar Chart: Family Members by Loan Location

SAS Code

```
/************************************************
 STACKED BAR CHART
 - Shows number of FAMILY_MEMBERS by LOAN_LOCATION
 - Groups are stacked to compare location composition
************************************************/
TITLE "Number of Family Members by Loan Location";
PROC SGPLOT DATA=ASGMTLIB.TESTING_LAS_PRED_TP086704_DS;
    VBAR FAMILY_MEMBERS /
         GROUP=LOAN_LOCATION
         GROUPDISPLAY=STACK   /* stack groups above each other */;
    LABEL FAMILY_MEMBERS = "Number of Family Members"
          LOAN_LOCATION  = "Loan Location";
RUN;
```

Output

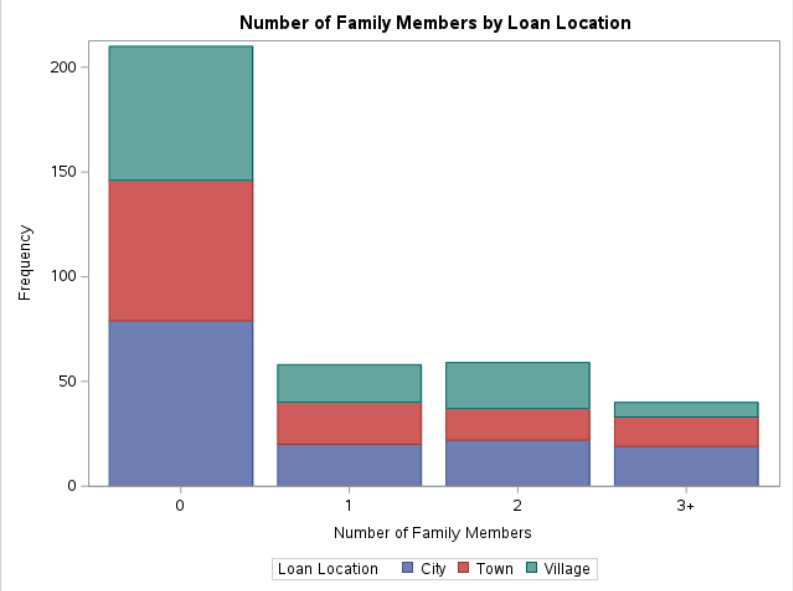

Figure 50: Stacked counts of `FAMILY_MEMBERS` by `LOAN_LOCATION`.

The stacked bar chart displays the number of family members across loan applicants by location (City, Town, Village). Applicants with no additional family members make up the largest group, with smaller proportions having one, two, or three-plus members. The distribution pattern is consistent across all three locations. This reveals that single applicants dominate the dataset regardless of their location.

Pie Chart: Loan Approval Status Predicted

SAS Code

```
/************************************************
 PIE CHART
  - Overall distribution of predicted LOAN APPROVAL STATUS
  - Quick view of class proportions
************************************************/
TITLE "Loan Approval Status Predicted";
PROC GCHART DATA=ASGMTLIB.TESTING_LAS_PRED_TP086704_DS;
    PIE3D I_LOAN_APPROVAL_STATUS;
    LABEL I_LOAN_APPROVAL_STATUS = "Loan Approval Status";
RUN;
QUIT;
```

Output

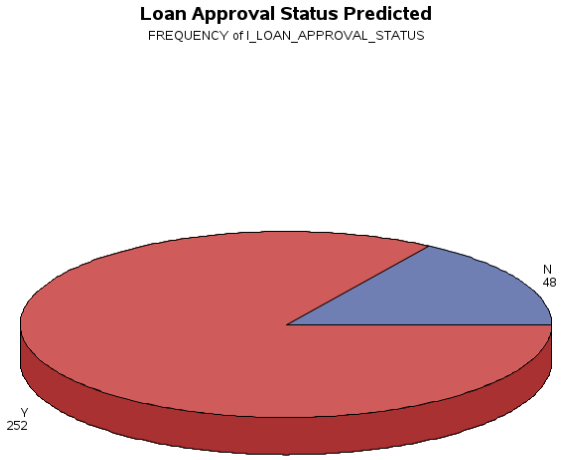

Figure 51: Overall split of predicted `I_LOAN_APPROVAL_STATUS`.

Based on the results, the pie chart shows the distribution of loan approval status. Out of 300 applications, 252 were approved (Y) while 48 were rejected (N). The large approval slice implies that the large majority of applicants satisfied the loan requirements. This shows that most of the applicants in the dataset are eligible to secure loan approval.

Box Plot: Loan Amount by Approval Status

SAS Code

```
/************************************************
 BOX PLOT
 - Compares LOAN_AMOUNT distributions across
   LOAN APPROVAL STATUS categories
 - Highlights median, spread, and outliers
************************************************/
TITLE "Loan Amount by Loan Approval Status";
PROC SGPLOT DATA=ASGMTLIB.TESTING_LAS_PRED_TP086704_DS;
    VBOX LOAN_AMOUNT / CATEGORY=I_LOAN_APPROVAL_STATUS;
    LABEL LOAN_AMOUNT           = "Loan Amount"
          I_LOAN_APPROVAL_STATUS = "Loan Approval Status";
RUN;
```

Output

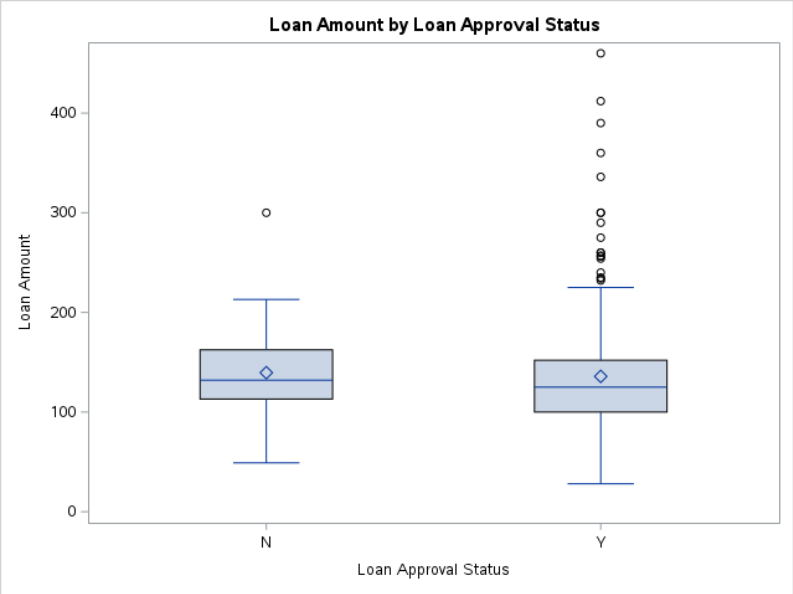

Figure 52: Distribution of LOAN_AMOUNT across predicted approval classes, showing median, spread, and outliers.

The box plot displays the distribution of loan amounts by approval status. Approved applications (Y) show a wider spread, with several outliers exceeding 400, while rejected ones (N) remain mostly below 200. Both groups have similar medians, but the the variability is greater for approved loans. This indicates that larger loan amounts are more likely to be approved.

Clustered Bar Chart: Loan Approval Status by Qualification

SAS Code

```
/************************************************
 CLUSTERED BAR CHART
 - Shows LOAN APPROVAL STATUS across QUALIFICATION
 - Side-by-side bars for clean comparison
************************************************/
TITLE "Loan Approval Status by Qualification";
PROC SGPLOT DATA=ASGMTLIB.TESTING_LAS_PRED_TP086704_DS;
    VBAR QUALIFICATION /
         GROUP=I_LOAN_APPROVAL_STATUS
         GROUPDISPLAY=CLUSTER;   /* side-by-side grouping */
    LABEL QUALIFICATION          = "Qualification"
          I_LOAN_APPROVAL_STATUS = "Loan Approval Status";
RUN;
```

Output

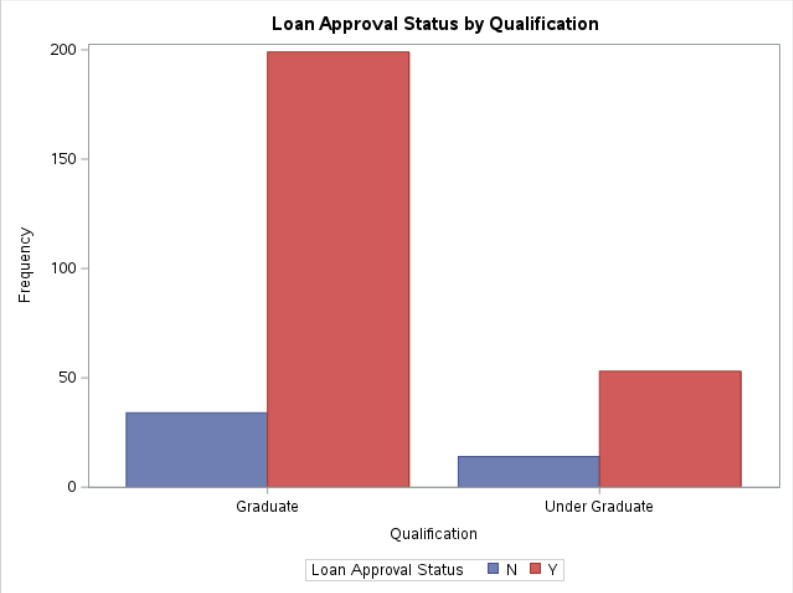

Figure 53: Side-by-side counts of predicted approval status by `QUALIFICATION`.

The clustered bar chart compares loan approval status by qualification (Graduate vs. Under Graduate). Graduates had nearly 200 approvals with relatively few rejections, while undergraduates showed about 50 approvals and 15 rejections. The difference suggests that higher education is associated with greater loan approval success. This highlights an applicant qualification as a key factor in loan decision.

Chapter 11

Conclusion

Loan Application Model Results.

The logistic regression model achieved stable convergence and demonstrated a significant improvement in fit compared to the intercept-only baseline, with AIC decreasing from 663.98 to 501.95 and SC from 668.24 to 565.93. Among the predictors, `LOAN_HISTORY` and `LOAN_LOCATION` emerged as the strongest and most statistically significant factors influencing loan approval. `MARITAL_STATUS` and `QUALIFICATION` were borderline significant, while other demographic and financial variables showed little impact at the 5% level. When applied to the testing dataset, the model generated consistent predictions of approval status, confirming its usefulness in supporting loan decision-making for new applications.

Reflections as a Data Scientist

Working on this project provided significant opportunities for technical and professional growth. I gained hands-on experience with SAS macro functions, logistic regression modeling, and ODS-based reporting, which were new to me. At the same time, I was able to deepen my knowledge of univariate and bivariate analysis, building on prior experience. My background in other programming languages made it easier to adapt to SAS, and I now see it as an additional pathway to perform statistical modeling and reporting efficiently.

The project reinforced the central role of data quality in predictive modeling. Although I already understood the importance of handling missing values, I learned new imputation techniques, such as applying the mode for categorical variables and the mean for continuous variables. This broadened my understanding of how imputation strategies are tailored to different data types.

Beyond technical skills, I became more familiar with the SAS environment and its ecosystem learning to navigate the interface, interpret logs, debug issues, and verify outputs. This process also indirectly improved my proficiency with LaTeX, as I had to design a structured, professional report that balanced code, outputs, and narrative in a clear format. Communicating findings in a way that is accessible to both technical and non-technical audiences was a key part of this learning process.

There were also challenges present. Repetition in code and formatting was a persistent issue, especially when presenting SAS code, outputs, and descriptions in a neat, non-overbearing way. I discovered how macros could reduce redundancy and simplify workflows, and I would use this strategy more extensively in future projects. Similarly, I would prepare more effectively by templating the document, standardising outputs, and minimizing unnecessary code.

Overall, this project strengthened my ability to apply data science tools critically, document findings effectively, and continuously refine workflows for greater clarity and efficiency.

References

Munnangi, S. (2024). Revolutionizing Loan Systems Through Intelligent Automation. *International Journal of Research In Computer Applications and Information Technology (IJRCAIT)*, *7*(2), 1508. https://doi.org/10.5281/zenodo.14220828

Nigmonov, A., Shams, S., & Urbonas, P. (2024). Estimating probability of default via delinquencies? Evidence from European P2P lending market. *Global Finance Journal*, *63*. https://doi.org/10.1016/j.gfj.2024.101050

Ramesh Pingili. (2025). AI-driven intelligent document processing for banking and finance. *International Journal of Management & Entrepreneurship Research*, *7*(2), 98–109. https://doi.org/10.51594/ijmer.v7i2.1802

Santos, A. M., Cincera, M., & Cerulli, G. (2024). Sources of financing: Which ones are more effective in innovation–growth linkage? *Economic Systems*, *48*(2). https://doi.org/10.1016/j.ecosys.2023.101177

Suhadolnik, N., Ueyama, J., & Da Silva, S. (2023). Machine Learning for Enhanced Credit Risk Assessment: An Empirical Approach. *Journal of Risk and Financial Management*, *16*(12). https://doi.org/10.3390/jrfm16120496