Credit Risk Scorecard Development

Contents

1. Introduction

Credit score models play a crucial role in retail financial sectors to help lenders manage credit risk effectively. Predicting client default can ease the processes of decision for loans.

Objective of this report is to apply a credit risk model analysis to create a credit scorecard. The scorecard will use for decision making and evaluating loan applicants based on the provided loan historical information. The methodology used for this scorecard is binning with WoE transformation and a logistic regression model. This report will guide readers to understand the analytical process and steps.

2. Data Description and Preparation

Data set loan.xlx include 1000rows and 8variables (IDClient, Age, Net_Income, Emp_Years, Home_Ownership, Default_status, Debt_Inc_Ratio, and Loan_Duration). Default_statuse variable is target variable for prediction model and this project (1= default, 0=not default).

Loan.xlx imported to SAS then sorting data set by default_statuse variable. The id variable has been deleted since it does not have any statistical value and was an identifier.

The LOAN_TRAINING dataset was created using a Simple Random Sampling method, stratified by the Default_status variable, also containing 1000 observations. The entire dataset has been used for data in EDA, WoE**.** Therefore, using `N=1000` with stratification does not remove any data. All records are retained, and the class balance is preserved.

Initial Exploratory Data Analysis (EDA):

This part aims to demonstrate initial features of variables.

Numerical

Table1 shows number of each numerical variables which shows Emp_years has missing values.

Table 1: Initial statistics for numerical variables

Variable	Label	N	Mean	Median	Mode	1st Pctl	99th Pctl
Age	Age	1000	36.3330	34.0000	26.0000	18.5000	65.0000
Net_Income	Net_Income	1000	25237.0890	23416.5000	13793.0000	13793.0000	56337.5000
Emp_Years	Emp_Years	805	5.9851	4.0000	2.0000	0.0000	30.0000
Debt_Inc_Ratio	Debt_Inc_Ratio	1000	0.3717	0.3355	0.2870	0.0950	0.8130
Loan_Duration	Loan_Duration	1000	6.6670	7.0000	10.0000	2.0000	10.0000

Net Income: The distribution of this variable is right-skewed, with the mean higher than the median, showing a large number of low-income applicants and a tail stretching towards higher incomes. In addition, there are some very high-income individuals known as outliers.

Table 2: Initial statistics for numerical variable

Moments			
Std Deviation	9980.71333	Variance	99614638.5
Skewness	9.06583052	Kurtosis	159.785214
Uncorrected SS	7.36426E11	Corrected SS	9.9515E10
Coeff Variation	39.5477994	Std Error Mean	315.617868

Basic Statistical Measures			
Location		Variability	
Mean	25237.09	Std Deviation	9981
Median	23416.50	Variance	99614639
Mode	13793.00	Range	212968
		Interquartile Range	7978

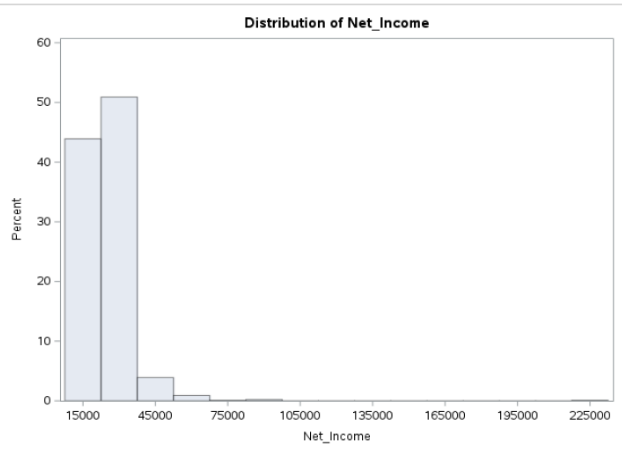


Figure1: Net income

Age: This variable has a relatively symmetric distribution, and the mean and median are close. Also, this shows normality, which is usual for demographic data.

Table 3: Initial statistics for age

Basic Statistical Measures			
Location		Variability	
Mean	36.33300	Std Deviation	11.70184
Median	34.00000	Variance	136.93304
Mode	26.00000	Range	57.00000
		Interquartile Range	17.00000

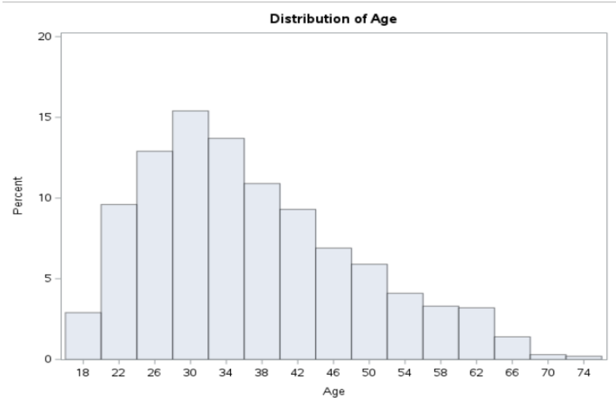


Figure2: age

Loan Duration: loan duration is a numerical variable, but frequency is more valuable here. The highest number is in 9.9, also, the mean and median are close.

Table 4: Initial statistics for loan duration

Basic Statistical Measures			
Location		Variability	
Mean	6.66700	Std Deviation	2.23587
Median	7.00000	Variance	4.99911
Mode	10.00000	Range	9.00000
		Interquartile Range	3.00000

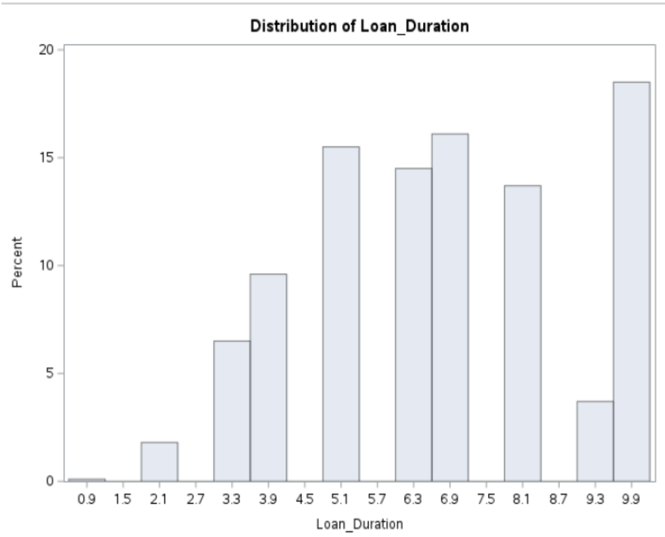


Figure 3: loan duration

Debt-to-Income Ratio (Debt_Inc_Ratio): This variable is crucial for credit risk, which is right-skewed; a minority have exceptionally high numbers. Also, most applicants fall within a reasonable range.

Table 5: Initial statistics for Debt-to-Income Ratio

Basic Statistical Measures			
Location		Variability	
Mean	0.371660	Std Deviation	0.16765
Median	0.335500	Variance	0.02811
Mode	0.287000	Range	0.80800
		Interquartile Range	0.22800

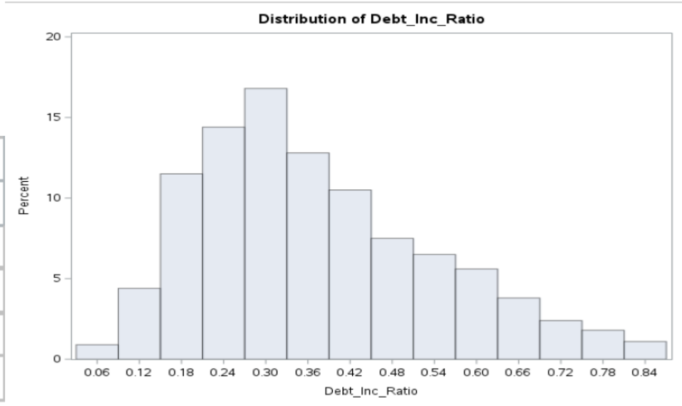


Figure 4: debt_inc_ratio

Employment Years (Emp_Years): Same as net income this shows right-skewed showing concentration of fewer years of employment, and fewer clients have high employment years.

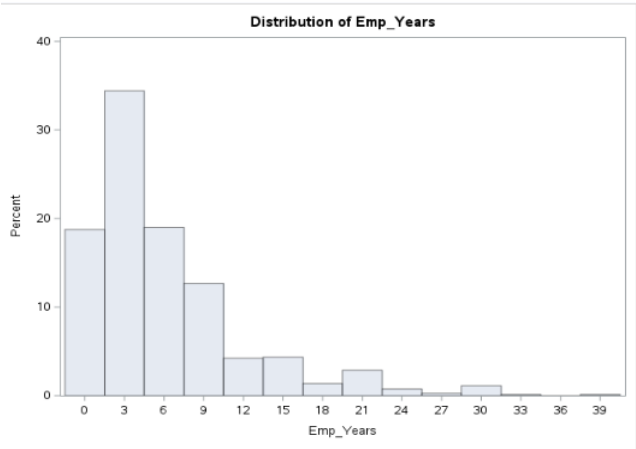


Figure5: Emp_years

Categorical

Home Ownership: The proportion of applicants in different categories, such as owner, renter, and others, is important to understand demographic characteristics.

Table 6: frequency for home ownership

Home_Ownership				
Home_Ownership	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Other	269	26.90	269	26.90
Owner	309	30.90	578	57.80
Renter	422	42.20	1000	100.00

Default_status: The frequency shows the class balance, which is crucial for a robust predictive model in credit risk.

Table 7: Frequency for default status

Default_status				
Default_status	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	750	75.00	750	75.00
1	250	25.00	1000	100.00

Missing values:

195 missing values in the variable emp-years will be treated as a separate group in the WoE transformation section.

Table 8: missing value

The MEANS Procedure

Variable	Label	N	N Miss
Default_status	Default_status	1000	0
Age	Age	1000	0
Net_Income	Net_Income	1000	0
Emp_Years	Emp_Years	805	195
Debt_Inc_Ratio	Debt_Inc_Ratio	1000	0
Loan_Duration	Loan_Duration	1000	0
Total	Total Number of Sampling Units	1000	0
AllocProportion	Allocation Proportion	1000	0
SampleSize	Sample Size	1000	0
ActualProportion	Actual Proportion of Total Sample Size	1000	0
SelectionProb	Probability of Selection	1000	0
SamplingWeight	Sampling Weight	1000	0

4. Feature Engineering: Weight of Evidence (WOE) and Information Value (IV)

This part transforms raw data into proper features for a credit scoring model based on the Weight of Evidence (WOE) and Information Value (IV) techniques. These are crucial steps to build a predictive model and a scorecard.

Concept of WOE and IV

Weight of Evidence (WOE) shows the predictive power of a variable's bin relative to the target variable, the natural logarithm of the ratio of the proportion of default and non-default. This transformation makes a characteristic proper for linear models and handles nonlinearity.

Information Value (IV) quantifies the overall predictive value of a variable, the power it can distinguish between default and non-default. Strong predictive power in variables is shown with high IV. It is calculated using this formula (WOE for each bin * (Proportion of Goods - Proportion of Bads)) for each bin.

WOE and IV are crucial since they are linearizing relationships, handle missing values, and help feature selection combine categories.

Methodology for Continuous Variables: For 195 missing values, inputted with 999 to avoid elimination, they were used as a separate bin for the clustering section in the SAS. Also, IV and WoE are calculated in this section and will be used for further binning in the Excel file.

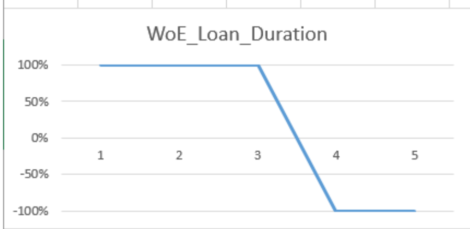
Table 9: Binning in excel for age

CLUSTER	Total	Total_Bac	Total_Goo	N_Class	Min	Max	PCT_B	PCT_G	Bads	Goods	WOE	IVi	perct_obs	class
1	1000	250	750	29	18	19	2.8	2.93	7	22	0.04652	0.0062	2.9	1
3	1000	250	750	47	20	21	2.8	5.33	7	40	0.64436	1.63237	4.7	1
5	1000	250	750	23	22	22	2	2.4	5	18	0.18232	0.07293	2.3	1
6	1000	250	750	26	23	23	1.6	2.93	4	22	0.60614	0.80818	2.6	1
7	1000	250	750	16	24	24	1.2	1.73	3	13	0.36772	0.19612	1.6	1
8	1000	250	750	34	25	25	3.2	3.47	8	26	0.08004	0.02134	3.4	1
9	1000	250	750	44	26	26	6	3.87	15	29	-0.4394	0.93732	4.4	1
10	1000	250	750	189	27	31	27.6	16	69	120	-0.5452	6.32463	18.9	1
11	1000	250	750	379	32	45	34.4	39.07	86	293	0.12721	0.59366	37.9	2
12	1000	250	750	186	46	62	14.8	19.87	37	149	0.29442	1.49171	18.6	3
2	1000	250	750	26	63	72	3.6	2.27	9	17	-0.4626	0.61683	2.6	3
4	1000	250	750	1	75	75	0	0.13	0	1			0.1	3

Row La	Sum of Goods	Sum of Bads	%Good	%Bad	WoE	Ivi
1	290	118	0.386667	0.472	-0.1994	0.01702
2	293	86	0.390667	0.344	0.12721	0.00594
3	167	46	0.222667	0.184	0.19074	0.00738
Grand Tot	**750**	**250**	1	1	0	0
						0.03033

Table 10: Binning in excel for loan duration

CLUSTER	Total	Total_Bad	Total_Goc	N_Class	Min	Max	PCT_B	PCT_G	Bads	Goods	WOE	IVi	perct_obs	class
1	1000	250	750	1	1	1	0.4	0	1	0			0.1	1
2	1000	250	750	18	2	2	1.6	1.87	4	14	0.15415	0.04111	1.8	1
3	1000	250	750	65	3	3	4.8	7.07	12	53	0.38677	0.87669	6.5	1
4	1000	250	750	96	4	4	7.2	10.4	18	78	0.36772	1.17672	9.6	2
5	1000	250	750	155	5	5	10.8	17.07	27	128	0.45758	2.86751	15.5	2
6	1000	250	750	145	6	6	14.4	14.53	36	109	0.00922	0.00123	14.5	3
7	1000	250	750	161	7	7	21.2	14.4	53	108	-0.38677	2.63006	16.1	4
8	1000	250	750	137	8	8	16.8	12.67	42	95	-0.28241	1.16727	13.7	4
9	1000	250	750	37	9	9	4.8	3.33	12	25	-0.36464	0.53481	3.7	5
10	1000	250	750	185	10	10	18	18.67	45	140	0.03637	0.02425	18.5	5



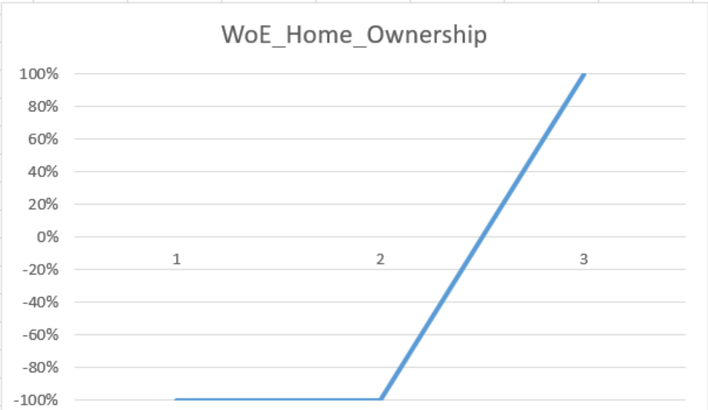
Row Labels	Sum of Goods	Sum of Bads	%Good	%Bad	WoE	Ivi
1	67	17	0.08933	0.068	0.27287	0.00582
2	206	45	0.27467	0.18	0.4226	0.04001
3	109	36	0.14533	0.144	0.00922	1.2E-05
4	203	95	0.27067	0.38	-0.33928	0.03709
5	165	57	0.22	0.228	-0.03572	0.00029
Grand Total	**750**	**250**	1	1	0	0
			0	0		
			0	0		
						0.08322

These are the steps applied for final binding in Excel for each variable. All variables are binned to other tables for this part and are attached in the appendix.

Methodology for Categorical Variables: For Home_Ownership, WOE and IV were calculated directly for each existing category in SAS. This variable has a suitable proportion of data in each category, monotonic WoE and moderate IV.

Table 11: Binning in excel for home ownership

Var	Total	Total_Bad	Total_Goc	N_Class	PCT_B	PCT_G	Bads	Goods	WOE	IVi
Renter	1000	250	750	422	57.2	37.2	143	279	-0.43025	0.086049
Other	1000	250	750	269	31.6	25.33	79	190	-0.22104	0.013852
Owner	1000	250	750	309	11.2	37.47	28	281	1.207538	0.31718



0.417081

Characteristic Selection: Variable selection was based on the information value. After extracting the SAS file, merged or grouped the clusters that SAS produced. Merging was used to group the similar clusters and reduce the number of clusters for simplicity. In this step, we have considered the WoE to be monotonic and generated an overall linear trend.

Variables such as age show a W pattern younger and older clients show more risk than other middle-aged ages. While here is important to prioritize keeping patterns, it has been decided to make 3 bins with overall linearity pattern for simplicity and good performance in the model. Also it shows IV 0.03, which is acceptable for using in the model.

Eventually, all variables are binned with monotonic WoEs and acceptable IV for use in the model.

Table 12: IV for all variables

Variable name	Loan duration	age	Home ownership	Emp_years	Net_incom	Debt_inc_Ratio
IV	0.08322	0.03033	0.417081	0.14094	0.083678	0.08865

The most challenging variable in this step is age, since it had not monotonic pattern. In addition, WoE analysis is crucial since a consistent trend across ordered bins is desirable (demonstrates changes).

Correlation: For Dataset mysas.loan_training_WOE and WoE-transformed predictors, is checked multicollinearity and insight of feature relationships for logistic regression modelling.

Table 13: Correlation analysis

Correlation Analysis

The CORR Procedure

6 Variables:	woe_home_ownership woe_age woe_emp_years woe_net_income woe_loan_duration woe_debt_inc_ratio

Simple Statistics						
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum
woe_home_ownership	1000	0.13211	0.72450	132.10705	-0.43025	1.20754
woe_age	970	0.01388	0.17218	13.46214	-0.19942	0.19074
woe_emp_years	1000	0.03579	0.37919	35.78765	-0.41848	0.48456
woe_net_income	1000	0.02205	0.29608	22.04776	-0.23817	0.54362
woe_loan_duration	1000	0.02129	0.29208	21.29438	-0.33928	0.42260
woe_debt_inc_ratio	1000	0.02389	0.30836	23.88653	-0.25934	0.73397

All six variables show non-zero standard deviation (valid variability). There are no extreme outliers based on the min and max values shown. In addition, there are no missing values since they have been treated as separate bins in characters.

Table 14: Pearson Correlation

Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations						
	woe_home_ownership	woe_age	woe_emp_years	woe_net_income	woe_loan_duration	woe_debt_inc_ratio
woe_home_ownership	1.00000 1000	0.30227 <.0001 970	0.18434 <.0001 1000	0.42057 <.0001 1000	0.01505 0.6346 1000	0.29439 <.0001 1000
woe_age	0.30227 <.0001 970	1.00000 970	0.34257 <.0001 970	0.30725 <.0001 970	0.01841 0.5669 970	0.00154 0.9619 970
woe_emp_years	0.18434 <.0001 1000	0.34257 <.0001 970	1.00000 1000	0.18686 <.0001 1000	0.00982 0.7565 1000	0.09498 0.0026 1000
woe_net_income	0.42057 <.0001 1000	0.30725 <.0001 970	0.18686 <.0001 1000	1.00000 1000	-0.05708 0.0712 1000	-0.01082 0.7324 1000
woe_loan_duration	0.01505 0.6346 1000	0.01841 0.5669 970	0.00982 0.7565 1000	-0.05708 0.0712 1000	1.00000 1000	-0.07034 0.0261 1000
woe_debt_inc_ratio	0.29439 <.0001 1000	0.00154 0.9619 970	0.09498 0.0026 1000	-0.01082 0.7324 1000	-0.07034 0.0261 1000	1.00000 1000

This part shows that none of the variables has a correlation > 0.7 or < -0.7, so the Variance Inflation Factor (VIF) is likely low. This ensures stable and interpretable logistic regression coefficients. The selected variables are independent enough to contribute unique information to the scorecard, improving predictive power. WoE transformations are performing well, there are no redundant predictors, which means each WoE variable is helping to separate good and bad credit behaviour.

5. Model Building: Logistic Regression

In this step, the logistic regression model is used as a foundation of credit scoring. The WoE transformed dataset is used for input data in the predictive model.

Data Splitting (Train/Test): To evaluate fairly and assess the outcome of the module and prevent overfitting, WOE-transformed data was randomly split into **70% training** and **30% testing** sets, using a random seed of **123**. This allows valuable evaluation of unseen data.

Although the WoE encoding is performed before the train-test split, this decision is applied to ensure optimal binning using all available data. The potential for slight overfitting is acknowledged, and model performance on an unseen sample supports the validity of the results.

Logistic Regression Methodology: The WoE transformed data set is used for input data in the predictive model. By WoE transformed variable the non-linear relationships in the original feature are linearized in logs odds of the target variable. Logistic regression is used because of transparency, strong performance in credit score, and modelling probability.

Variable Selection (Stepwise): Stepwise and backwards methods are usually used in scorecard modelling; here, the stepwise method is used. This method automatically adds or removes variables based on their statistical significance.

Table 15: Summary of stepwise selection

Summary of Stepwise Selection							
	Effect						
Step	Entered	Removed	DF	Number In	Score Chi-Square	Wald Chi-Square	Pr > ChiSq
1	woe_home_ownership		1	1	45.9094		<.0001
2	woe_loan_duration		1	2	12.7812		0.0004
3	woe_emp_years		1	3	5.5155		0.0188
4	woe_debt_inc_ratio		1	4	2.9998		0.0833

This table clearly illustrates the number of variables selected in each step.

Table 16: Analysis of maximum likelihood estimated

Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-1.1104	0.0974	129.9556	<.0001
woe_home_ownership	1	-0.9878	0.1777	30.9156	<.0001
woe_emp_years	1	-0.5030	0.2512	4.0112	0.0452
woe_loan_duration	1	-1.0881	0.3320	10.7409	0.0010
woe_debt_inc_ratio	1	-0.6395	0.3292	3.7726	0.0521

This is the vital table; these estimated values are used to score points further. The estimated coefficients (Betas) for intercept and selection variables are shown for each WOE-transformed predictor variable, along with their standard errors, Wald Chi-Square statistics, and p-values. The coefficient for woe_home_ownership demonstrates that a one-unit increase in its WOE (indicating lower risk) decreases the log-odds of default by 0.9286.

Model Output & Coefficients: Final model includes intercept and four transformed variables woe_home_ownership, woe_emp_years, woe_loan_duration, woe_debt_inc_ratio. The model fit statistic demonstrates goodness of fit.

Goodness-of-fit measures like **AIC, SC, and -2 Log L** are provided for both the intercept-only model and the final model with covariant. These are important for evaluating the overall fit and parsimony of the model. The R-Square values (0.1002 and 0.1480 for Max-rescaled) are also crucial to discuss the explanatory power.

Table 17: Step 4 stepwise results

Step 4. Effect woe_debt_inc_ratio entered:

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

Model Fit Statistics		
Criterion	Intercept Only	Intercept and Covariates
AIC	793.633	727.718
SC	798.184	750.473
-2 Log L	791.633	717.718

R-Square	0.1002	Max-rescaled R-Square	0.1480

Testing Global Null Hypothesis: BETA=0			
Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	73.9157	4	<.0001
Score	66.4181	4	<.0001
Wald	58.5636	4	<.0001

Residual Chi-Square Test		
Chi-Square	DF	Pr > ChiSq
0.8636	2	0.6494

6. Model Performance and Validation

Here assesses the performance (discrimination and calibration) the model on test dataset.

Table 18 provides the **C-statistic (0.713)**, which is equivalent to the Area Under the ROC Curve (AUC); this shows model reasonably discriminate between good and bad. The table also includes Somers' D, Gamma, and Tau-a, all are measures for model discrimination. This is for assessing the model's predictive power.

Table 18: Association of predicted probabilities and observed responses

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	70.9	Somers' D	0.427
Percent Discordant	28.2	Gamma	0.431
Percent Tied	0.9	Tau-a	0.161
Pairs	92571	c	0.713

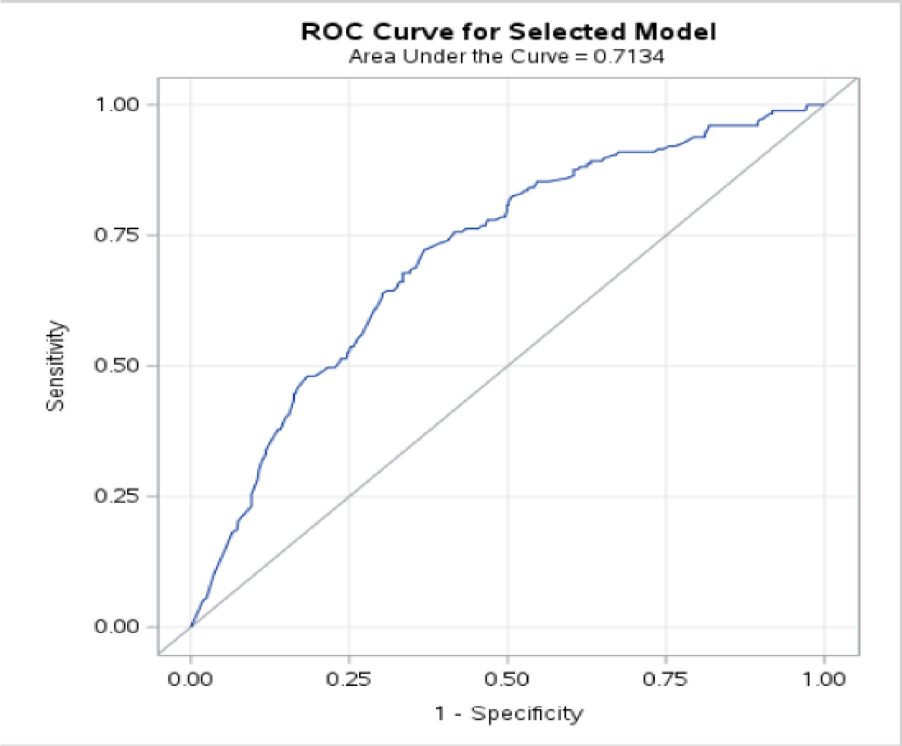


Figure6: ROC curve

The ROC curve visually illustrates model's ability to discriminate between good and bad loans. The AUC value is explicitly stated, reinforcing the numeric measure from the Association table

Goodness-of-Fit: This part evaluates calibration of model (how model's predicted probability align with real observation outcomes).

The Hosmer-Lemeshow test evaluate the overall goodness-of-fit of the model.This test (with Pr>ChiSq=0.4255) this number greater than 0.05 significant level. Therefore, the null hypothesis is not rejected, and the model fits the observed data.

Table 19: The Hosmer-Lemeshow

Hosmer and Lemeshow Goodness-of-Fit Test		
Chi-Square	DF	Pr > ChiSq
8.0823	8	0.4255

Table 20 shows the model's accuracy, sensitivity, and specificity at a given probability level (0.500). Higher specificity would mean fewer misclassified of bad loans are defined as good, this is the important evaluation in default prediction. This table could be helpful for determining the cut-off point, which would depend on the business context and the relative costs of these misclassifications.

Table 20: classification table

Classification Table									
	Correct		Incorrect		Percentages				
Prob Level	Event	Non-Event	Event	Non-Event	Correct	Sensi-tivity	Speci-ficity	Pos Pred	Neg Pred
0.500	9	513	10	168	74.6	5.1	98.1	47.4	75.3

An AUC of 0.713 on the test dataset shows reasonable predictive power, which is acceptable in credit risk. The Hosmer-Lemeshow test confirmed good calibration.

New modelling techniques or features to boost the AUC could be considered as future improvements. In addition, specific customer segment analysis and a cost-benefit analysis for different cut-off points would be beneficial.

7. Scorecard Construction

The logistic regression output is converted to the scorecard, a point-based system using a specific formula.

Credit Score = SUM [(WoE * Beta_i + (Beta_0 / N)) * Factor] + (Offset / N)

The parameters used here are Score at Odds: 600**,** Odds: 50**,** PDO: 20**,** Offset: 87.123**,** and Factor: 28.854**.**

Base Score: It comes from a nutria class where the ratio of good and bad is the same, and WoE is equal to zero. Each applicant's starting point is derived from the model's intercept and chosen odds-to-score ratio.

Points per WOE Variable: This is calculated for each characteristic using the coefficient from logistic regression. It ensures attributes with a more substantial impact on credit risk contribute more points.

Points per Characteristic Category/Bin: Points assigned to each category or bin in each characteristic. This involves adding the scaled WoE contribution of that bin to a portion of the base score, allowing for granular scoring based on an applicant's attributes within each category.

Table 21: Final scorecard

Characteristic	Category / Bin	Score Points
Home Ownership	Renter	47
	Other	53
	Owner	91
	Neutral class	59
Emp_Years	emp_years <= 3	52
	4 <= emp_years <= 7	59
	8 <= emp_years <= 10	64
	emp_years >= 11 or missing	67
	Neutral class	59
Debt_Inc_Ratio	debt_inc_ratio <= 0.326	55
	0.327 <= debt_inc_ratio <= 0.506	61
	0.508 <= debt_inc_ratio <= 0.68	66
	0.681 <= debt_inc_ratio <= 0.842	71
	Neutral class	59
Loan_Duration	loan_duration <= 3	71
	4 <= loan_duration <= 5	78
	loan_duration = 6	59
	7 <= loan_duration <= 8	44
	9 <= loan_duration <= 10	57
	Neutral class	59
	Cut-off	237

A score of **237** or above will be accepted.

This cut-off point has been obtained with the neutral score indicating a distribution of good and bad (50:50) clients, which is Weight of Evidence =0.

This helps to give the threshold at which the clients are more inclined to be bad clients and to set up a cut-off score. A cut-off score is the lowest possible credit score for an individual to be accepted for a loan. It is a complex process that is determined by the business value.

8. Cut-Off Point Determination and Applicant Scenario

Cut-off Point: The right cut-off point is a crucial business decision, since it directly affects the number of accepted loans and the risk of default. This is a balance that aligns the model predictions with the organisation's strategic goals.

For this model, a specific cut-off score of **237** has been chosen. This cut-off is designed to optimize the balance between accepting a sufficient number of profitable loans and minimizing the risk of defaults.

Applicant Scenario: As an application scenario, a profile with these features is considered when applying for a loan**.**

Table 22: Application Scenarios

	Home Ownership	Emp_Years	Debt_Inc_Ratio	Loan_Duration	Total Score
Applicant A	Renter 47	3 **52**	0.326 **55**	4 **78**	**232**
Applicant B	Other 53	10 **64**	0.51 **66**	6 **59**	**242**
Applicant C	Owner 91	11 **67**	0.69 **71**	5 **78**	**307**

Applicant A's profile and calculated total score are not accepted for a loan, which indicates its risk profile as the model evaluates. It is below the produced cut off, showing a higher likelihood of default.

On the other hand, applicant's B and C are accepted for the loan. Their profiles indicate scores at or above 237(cut-off point), a lower or moderate risk of default based on the model.

9. Conclusion

With WoE transformation and logistic regression this project successfully developed a credit risk scorecard**,** represents a standard and well-accepted approach in retail credit risk modelling

Showing strong predictive power for default and non-default, including achieving a C-statistic (AUC) of 0.713 on the test data. Model good calibration confirmed with the Hosmer-Lemeshow test, with a p-value of **0.4255.**

The final scorecard can use as a practical tool for loan decision making, with **a** cut-off point of 237 to balance accepting profitable loans with minimizing default risk.

For future improvement, external validation on new dataset, monitoring of performance, and exploring other modelling techniques would be suitable to develop predictive power.

10. References

Seddiqi, S.A. (2022) *Credit Risk Analysis: A Practical Approach*. 2nd ed. London: RiskPress.

Appendix

Full SAS Code

```
LIBNAME MYSAS '/home/u64149066/sasuser.v94/MM711/MYSAS';

PROC IMPORT
DATAFILE="/home/u64149066/sasuser.v94/MM711/MYSAS/loans_dataset.xlsx"

   OUT=MYSAS.loan

   DBMS=XLSX

   REPLACE;

   SHEET="CreditRisk";

   GETNAMES=YES;

RUN;



Libname mysas '/home/u64149066/sasuser.v94/MM711/MYSAS';


proc sort data=mysas.loan;
```

```
        by Default_status;
run;
/* Dataset contains exactly 1000 rows, so N=1000 does not reduce the data.
  This step applies stratified randomization only — full data is used. */
proc surveyselect data = mysas.loan out = mysas.loan_training
      method = srs N=1000 /*samprate =0.8*/ seed = 12345;
              strata Default_status / alloc=proportional;
run;
/* the same % of Bad/Good clients in both data sets*/
Title 'LOAN total data';
proc freq data=mysas.loan;
tables Default_status;
quit;
Title 'LOAN training data';
proc freq data=mysas.loan_training;
tables Default_status;
quit;
PROC MEANS DATA=MYSAS.loan_training_for_model NMISS;
   VAR _ALL_; /* Checks all variables in the dataset */
   OUTPUT OUT=Missing_Values_Report NMISS=;
RUN;
proc means data=mysas.loan_training_for_model n nmiss;
run;

PROC PRINT DATA=Missing_Values_Report;
   VAR  Age Net_Income Emp_Years Home_Ownership Default_status Debt_Inc_Ratio and
Loan_Duration;
   TITLE "Number of Missing Values Per Variable";
RUN;

PROC CONTENTS DATA=MYSAS.loan;
RUN;
```

```
PROC CONTENTS DATA=MYSAS.loan_training_for_model;
RUN;

PROC PRINT DATA=MYSAS.loan_training_for_model (OBS=5);
RUN;

PROC MEANS DATA=MYSAS.loan_training_for_model NMISS MAXDEC=0 NOPRINT;
   OUTPUT OUT=Missing_Values_Report NMISS=;
RUN;

PROC PRINT DATA=Missing_Values_Report;
RUN;

PROC MEANS DATA=MYSAS.loan_training_for_model NMISS NOPRINT;
   OUTPUT OUT=Missing_Values_Report NMISS=;
RUN;

PROC TRANSPOSE DATA=Missing_Values_Report OUT=Missing_Transposed;
RUN;

PROC PRINT DATA=Missing_Transposed;
   ID _NAME_;
   VAR COL1;
   TITLE "MISSING VALUES";
RUN;
/*  drop the IDClient variable from the training set for modeling */
data mysas.loan_training_for_model;
   set mysas.loan_training;
   drop IDClient; /* Exclude the unique identifier */
run;
```

```
/* Verification of proportions (using the _training dataset as it still has IDClient for comparison if needed) */

Title 'LOAN total data - Default Status Distribution';

proc freq data=mysas.loan;

tables Default_status;

quit;


Title 'LOAN training data - Default Status Distribution';

proc freq data=mysas.loan_training;

tables Default_status;

quit;

ods graphics on; /* ODS Graphics for plots */


/******************************Numerical Data Exploration******************************/

/* Univariate distributions for selected numerical variables */

proc univariate data=mysas.loan_training_for_model;

VAR     Net_Income;

CDFPLOT        Net_Income;

HISTOGRAM  Net_Income;

run;


proc univariate data=mysas.loan_training_for_model;

VAR     Age;

CDFPLOT        Age;

HISTOGRAM  Age;

run;


proc univariate data=mysas.loan_training_for_model;

VAR     Loan_Duration;

CDFPLOT        Loan_Duration;
```

```
HISTOGRAM  Loan_Duration;
run;

proc univariate data=mysas.loan_training_for_model;
VAR     Debt_Inc_Ratio;
CDFPLOT        Debt_Inc_Ratio;
HISTOGRAM  Debt_Inc_Ratio;
run;

/* Descriptive statistics for all numerical variables */
proc means data=mysas.loan_training_for_model
N MEAN MEDIAN MODE P1 P99 MAXDEC=4;
var Age Net_Income Emp_Years Debt_Inc_Ratio Loan_Duration;
run;

/* QQ-Plots for normality assessment */
proc univariate data=mysas.loan_training_for_model noprint;
QQPLOT Net_Income /NORMAL(MU=EST     SIGMA=EST    COLOR=LTGREY);
run;

proc univariate data=mysas.loan_training_for_model noprint;
QQPLOT Age /NORMAL(MU=EST     SIGMA=EST    COLOR=LTGREY);
run;



/******************************Categorical Data
Exploration*******************************/
/* Frequency tables for categorical variables */
proc freq data=mysas.loan_training_for_model;
tables Home_Ownership Default_status; /* Default_status is also categorical*/
quit;
```

```
/* Scatter plot for two numerical variables */

proc gplot data=mysas.loan_training_for_model;
plot Net_Income*Age; /* Example: Relationship between Net_Income and Age */
run;


/***********************************************************************
*****
* *
* STEP 4: MACROS FOR REUSABLE EDA COMPONENTS                    *
* *
************************************************************************
****/

/************* MACRO: HISTOGRAMS WITH CLASS STATEMENT
**********************/

%macro hist(var_x=);
proc univariate data=mysas.loan_training_for_model;
  class Default_status; /* Stratify histograms by Default_status */
  var &var_x.;
  histogram &var_x. / nrows=2 odstitle="PROC UNIVARIATE with CLASS statement for
&var_x.";
  ods select histogram; /* display only the histograms */
run;
%mend;

%hist(var_x=Net_Income);
%hist(var_x=Age);
%hist(var_x=Emp_Years);
```

```
%hist(var_x=Debt_Inc_Ratio);

%hist(var_x=Loan_Duration);


/************* MACRO: DESCRIPTIVE STATISTICS TABLE
***********************/


%Macro DescripStats(VarX=,n=);


PROC UNIVARIATE Noprint DATA=mysas.loan_training_for_model PLOTS;

                        VAR &VarX.;

                        Histogram / Cfill=Blue Outhist = HistOut&n.;

                        OUTPUT OUT=Stat&n. NMISS=NMISS
NOBS=NOBS PCTLPTS =2.5 97.5 PCTLPRE=P

                        MEAN=MEAN MODE=MODE
MEDIAN=MEDIAN

                        Q1=Q1 Q3=Q3 P5=P5 P10=P10 P90=P90 P95=P95
STD=STD

                        MAX=MAX MIN=MIN KURTOSIS=KURTOSIS
SKEWNESS=SKEWNESS;

            RUN;


            DATA Stat&n.;

                  FORMAT Name $32. NOBS NMISS KURTOSIS
SKEWNESS MEAN MODE STD MIN P2_5 P5 P10 Q1 MEDIAN Q3 P90 P95 P97_5 MAX
BEST12.;

                  SET Stat&n.;

      Name="&VarX";


    RUN;


    Proc Append base=Summary_STAT data=Stat&n. force; Run;

%mend DescripStats;
```

```
/* Call macro for all numerical variables */

%DescripStats(VarX=Age,n=1);
%DescripStats(VarX=Net_Income,n=2);
%DescripStats(VarX=Emp_Years,n=3);
%DescripStats(VarX=Debt_Inc_Ratio,n=4);
%DescripStats(VarX=Loan_Duration,n=5);



/************* MACRO: FREQUENCY TABLES FOR CATEGORICAL VARIABLES (USING PROC SQL) ***********************/



%macro freq_categorical (VarX=,n=);

	/* Use PROC FREQ to get counts and percentages */
	proc freq data=mysas.loan_training_for_model noprint;
		tables &VarX. / out=FreqRaw&n NOCUM;
	run;


	data FreqTemp&n.;
		set FreqRaw&n;

		length Var $50 Class $100;

		Var = "&VarX.";

		Class = vvaluex("&VarX.");
```

```
            N = COUNT;
            Total = sum(COUNT);
            percentage = PERCENT/100;



            keep Var Class N Total percentage;
            format Var $50. Class $100. N Best10. Total Best10. percentage percent10.2;
        run;



        proc sql;
            %if %sysfunc(exist(UNIVAR_CLASS_1)) %then %do;
                insert into UNIVAR_CLASS_1
                select Var, Class, N, Total, percentage
                from FreqTemp&n.;
            %end;
            %else %do;
                create table UNIVAR_CLASS_1 as
                select Var, Class, N, Total, percentage
                from FreqTemp&n.;
            %end;
        quit;

        /* Clean up temporary datasets */
        proc datasets library=work nodetails nolist;
            delete FreqRaw&n FreqTemp&n;
        quit;

    %mend freq_categorical;
```

```
/* Call macro for categorical variables */

%freq_categorical(VarX=Home_Ownership,n=1);
%freq_categorical(VarX=Default_status,n=2);

Data mySAS.LOAN_training_WOE;
  Set mySAS.LOAN_training;

  /* Age */
  if age <= 31 then woe_age = -0.199415990153255;
  else if 32 <= age <= 45 then woe_age = 0.12721302409545;
  else if 46 <= age <= 75 then woe_age = 0.19074012725955;

  /* Net Income */
  if net_income <= 24171 then woe_net_income = -0.238173634719848;
  else if 24234 <= net_income <= 32758 then woe_net_income = 0.266020575670601;
  else if 32786 <= net_income <= 223300 then woe_net_income = 0.543615446588982;

  /* Employment Years */
  if emp_years <= 3 then woe_emp_years = -0.418480476220235;
  else if 4 <= emp_years <= 7 then woe_emp_years = -0.0224728558520586;
  else if 8 <= emp_years <= 10 then woe_emp_years = 0.312374685042152;
  else if emp_years >= 11 or emp_years = 9999 or missing(emp_years) then woe_emp_years
= 0.484557696945381;

  /* Debt-to-Income Ratio */
  if debt_inc_ratio <= 0.326 then woe_debt_inc_ratio = -0.259338543550954;
  else if 0.327 <= debt_inc_ratio <= 0.506 then woe_debt_inc_ratio = 0.12350476778123;
  else if 0.508 <= debt_inc_ratio <= 0.68 then woe_debt_inc_ratio = 0.421213465076303;
  else if 0.681 <= debt_inc_ratio <= 0.842 then woe_debt_inc_ratio = 0.733969175080201;
```

```
  /* Loan Duration */
  if loan_duration <= 3 then woe_loan_duration = 0.27286698666664;
  else if 4 <= loan_duration <= 5 then woe_loan_duration = 0.422601390351152;
  else if loan_duration = 6 then woe_loan_duration = 0.00921665510492405;
  else if 7 <= loan_duration <= 8 then woe_loan_duration = -0.339283201226863;
  else if 9 <= loan_duration <= 10 then woe_loan_duration = -0.0357180826020792;

  /* Home Ownership */
  if home_ownership = 'Renter' then woe_home_ownership = -0.430245137;
  else if home_ownership = 'Other' then woe_home_ownership = -0.221036069;
  else if home_ownership = 'Owner' then woe_home_ownership = 1.2075378705;

RUN;

%Let input_table=mysas.loan_training;

/*Continuous variables*/
%Macro BivariateCont(VarX=,n=);

/* Handle missing values for the continuous variable by replacing them with 9999. */
/* This before clustering to ensure all observations are included. */
data &input_table.;
set &input_table.;
if &VarX.=. then &VarX.=9999;
run;



proc sort data=&input_table.;
by &VarX.;
run;
```

```
proc fastclus noprint data=&input_table. out=cont_clust_&n. /*converge=0*/ maxclusters=12
/*MAXITER=200 REPLACE=FULL*/ nomiss;
      var &VarX.;
run;



data cont_clust_&n.;
 set cont_clust_&n. (keep=&VarX. Cluster Default_status);
run;

Proc Sql NOPRINT;create table Report_ContClust_&n. as
                                                    Select &VarX. as Var,
                                                     Cluster,
                                                            Default_status,
                                                            Count(*)   as Total,
                                                              sum(Default_status=1)              as
Total_Bads,
                                        sum(Default_status=0 ) as Total_Goods
                                                  from  cont_clust_&n.
;
Quit;

/* Create the final summary report for the continuous variable's clusters. */
Proc Sql NOPRINT;create table SUM_Report_ContFinal_&n. as
                                                    Select Cluster,
                                                    Total,
                                                    Total_Bads,
                                                    Total_Goods,
                                                            Count(*)   as N_Class,
```

```
Min(Var) as Min , /* Minimum value of the continuous variable in the cluster */
Max(Var) as Max , /* Maximum value of the continuous variable in the cluster */
sum(Default_status=1)/Total_Bads*100 as PCT_B format=5.2 ,
sum(Default_status=0)/Total_Goods*100 as PCT_G format=5.2,
sum(Default_status=1) as Bads,
sum(Default_status=0 ) as Goods,

log((sum(Default_status=0)/Total_Goods)/(sum(Default_status=1)/Total_Bads)) as WOE,
(calculated PCT_G - calculated PCT_B)*(calculated Woe) as IVi,
(calculated N_Class)/total*100 as perct_obs
from Report_ContClust_&n.
Group by Cluster
;
Quit;


/* Sort the final report by the minimum value of the cluster for ordered analysis. */
proc sort data=SUM_Report_ContFinal_&n. /*out=Report_clust_nd&n.*/ nodupkey;
by Min;
quit;



%mend BivariateCont;



%BivariateCont(VarX=Age,n=1);
%BivariateCont(VarX=Net_Income,n=2);
%BivariateCont(VarX=Emp_Years,n=3);
%BivariateCont(VarX=Debt_Inc_Ratio,n=4);
%BivariateCont(VarX=Loan_Duration,n=5);
```

```
%Let input_table=mysas.loan_training;

/*Categorial variables*/
%Macro  BivariateCategorical(VarX=,n=);

Proc Sql NOPRINT;create table Report_Bivar_&n. as
                                                            Select &VarX. as Var,
                                                             Default_status,
                                                                    Count(* )   as Total,
                                                                      sum(Default_status=1)             as
Total_Bads,
                                              sum(Default_status=0 ) as Total_Goods
                                                         from  &input_table.;
;
Quit;

Proc Sql NOPRINT;create table SUM_ReportCat_Final_&n. as
                                                            Select Var,
                                                            Total,
                                                            Total_Bads,
                                                            Total_Goods,
                                                                    Count(* )   as N_Class,
                                                                      sum(Default_status=1)/Total_Bads*100  as
PCT_B   format=5.2 ,
                       sum(Default_status=0)/Total_Goods*100  as PCT_G   format=5.2,
                                                               sum(Default_status=1)         as Bads,
                                              sum(Default_status=0 ) as Goods,

log((sum(Default_status=0)/Total_Goods)/(sum(Default_status=1)/Total_Bads))  as WOE,
```

```
                         ((calculated PCT_G/100) - (calculated PCT_B/100))*(calculated Woe)   as
IVi,
                                                            (calculated N_Class)/total*100 as perct_obs
                                                    from   Report_Bivar_&n.
                                                    group by Var
                                                    ;
                                                    Quit;

proc sort data=SUM_ReportCat_Final_&n. /*out=Report_clust_nd&n.*/ nodupkey;
by WoE;
quit;

%mend BivariateCategorical;
%BivariateCategorical(VarX=home_ownership,n=1);

Data mySAS.LOAN_training_WOE;
  Set mySAS.LOAN_training;

  /* Age */
  if age <= 30 then woe_age = -0.199415990153255;
  else if 31 < age <= 45 then woe_age = 0.12721302409545;
  else if 46 <= age <= 75 then woe_age = 0.19074012725955;

  /* Net Income */
  if net_income <= 24171 then woe_net_income = -0.238173634719848;
  else if 24234 <= net_income <= 32758 then woe_net_income = 0.266020575670601;
  else if 32786 <= net_income <= 223300 then woe_net_income = 0.543615446588982;

  /* Employment Years */
  if emp_years <= 3 then woe_emp_years = -0.418480476220235;
  else if 4 <= emp_years <= 7 then woe_emp_years = -0.0224728558520586;
```

```
  else if 8 <= emp_years <= 10 then woe_emp_years = 0.312374685042152;
  else if emp_years >= 11 or emp_years = 9999 or missing(emp_years) then woe_emp_years
= 0.484557696945381;

  /* Debt-to-Income Ratio */
  if debt_inc_ratio <= 0.326 then woe_debt_inc_ratio = -0.259338543550954;
  else if 0.327 <= debt_inc_ratio <= 0.506 then woe_debt_inc_ratio = 0.12350476778123;
  else if 0.508 <= debt_inc_ratio <= 0.68 then woe_debt_inc_ratio = 0.421213465076303;
  else if 0.681 <= debt_inc_ratio <= 0.842 then woe_debt_inc_ratio = 0.733969175080201;

  /* Loan Duration */
  if loan_duration <= 3 then woe_loan_duration = 0.27286698666664;
  else if 4 <= loan_duration <= 5 then woe_loan_duration = 0.422601390351152;
  else if loan_duration = 6 then woe_loan_duration = 0.00921665510492405;
  else if 7 <= loan_duration <= 8 then woe_loan_duration = -0.339283201226863;
  else if 9 <= loan_duration <= 10 then woe_loan_duration = -0.0357180826020792;

  /* Home Ownership */
  if home_ownership = 'Renter' then woe_home_ownership = -0.430245137;
  else if home_ownership = 'Other' then woe_home_ownership = -0.221036069;
  else if home_ownership = 'Owner' then woe_home_ownership = 1.2075378705;

RUN;

TITLE;
TITLE1 "Correlation Analysis";
FOOTNOTE;
FOOTNOTE1 ;

PROC CORR DATA=mysas.loan_training_WOE
   PLOTS=NONE
```

```
    PEARSON
    OUTP=Corr_logit
    VARDEF=DF;

    VAR WoE_home_ownership WoE_age WoE_emp_years WoE_net_income
WoE_loan_duration WoE_debt_inc_ratio;
RUN;



proc surveyselect data=mysas.loan_training_woe
    out=loan_woe_split
    samprate=0.7
    outall
    seed=123;
run;

data train test;
    set loan_woe_split;
    if selected=1 then output train;
    else output test;
run;
ODS GRAPHICS ON;

TITLE;
TITLE1 "Logistic Regression - Train";
FOOTNOTE;
FOOTNOTE1 "scoring models";

PROC LOGISTIC DATA=train DESCENDING
    PLOTS(ONLY)=ALL
    OUTMODEL=logit_model;
```

```
MODEL Default_status =
   WoE_home_ownership
   WoE_age
   WoE_emp_years
   WoE_net_income
   WoE_loan_duration
   WoE_debt_inc_ratio
   /
   OUTROC=ROC
   SELECTION=STEPWISE
   SLE=0.1
   SLS=0.1
   INCLUDE=0
   CORRB
   CTABLE
   PPROB=(0.5)
   SCALE=PEARSON
   RSQUARE
   LACKFIT
   LINK=LOGIT
   CLPARM=WALD
   CLODDS=WALD
   ALPHA=0.05;

ODS OUTPUT ParameterEstimates=Beta;
ODS OUTPUT Association=STAT_TABLE;

OUTPUT OUT=Train_Predict
   PREDPROB=(INDIVIDUAL)
   XBETA=xbeta__Target;
```

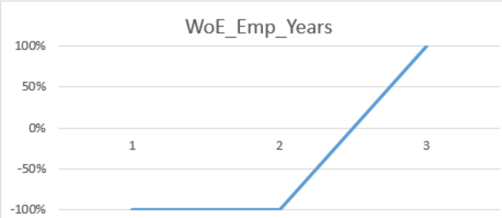
```
RUN;

PROC LOGISTIC INMODEL=logit_model;
   SCORE DATA=test OUT=Test_Predict;
RUN;
```

Detailed Output Tables

Table 23: Binning in Excel for Emp_years

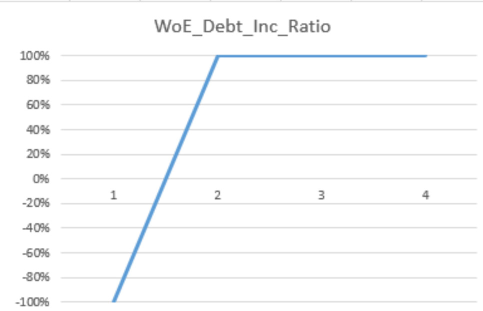
CLUSTER	Total	Total_Bad	Total_Goc	N_Class	Min	Max	PCT_B	PCT_G	Bads	Goods	WOE	IVi	perct_obs	CLASS
1	1000	250	750	151	0	1	21.2	13.07	53	98	-0.48394	3.93602	15.1	1
3	1000	250	750	194	2	3	25.2	17.47	63	131	-0.36655	2.83465	19.4	1
5	1000	250	750	168	4	5	15.2	17.33	38	130	0.13134	0.28018	16.8	2
7	1000	250	750	68	6	7	8.8	6.13	22	46	-0.36101	0.9627	6.8	2
9	1000	250	750	34	8	8	3.2	3.47	8	26	0.08004	0.02134	3.4	3
10	1000	250	750	68	9	10	4.8	7.47	12	56	0.44183	1.17822	6.8	3
11	1000	250	750	69	11	16	2	8.53	5	64	1.45083	9.47877	6.9	4
12	1000	250	750	40	17	25	2.4	4.53	6	34	0.63599	1.35678	4	4
2	1000	250	750	11	27	30	0.8	1.2	2	9	0.40547	0.16219	1.1	4
4	1000	250	750	1	32	32	0	0.13	0	1			0.1	4
6	1000	250	750	1	40	40	0	0.13	0	1			0.1	4
8	1000	250	750	195	9999	9999	16.4	20.53	41	154	0.22477	0.92904	19.5	4



Row Lab	Sum of Bads	Sum of Goods	%GOOD	%BAD	WoE	Ivi
1	116	229	0.30533	0.464	-0.41848	0.0664
2	60	176	0.23467	0.24	-0.02247	0.00012
3	20	82	0.10933	0.08	0.31237	0.00916
4	54	263	0.35067	0.216	0.48456	0.06525
Grand Tota	**250**	**750**	1	1	0	0
			0	0		
						0.14094

Table 24: Binning in Excel for Debt_inc_Ratio

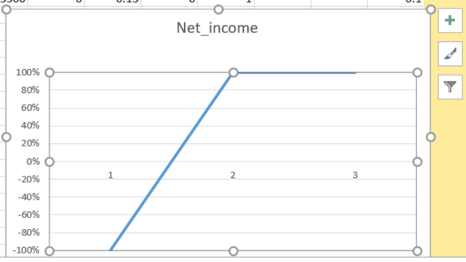
CLUSTER	Total	Total_Bac	Total_Goc	N_Class	Min	Max	PCT_B	PCT_G	Bads	Goods	WOE	IVi	perct_obs	class
1	1000	250	750	2	0.034	0.042	0	0.27	0	2			0.2	1
5	1000	250	750	3	0.054	0.061	0	0.4	0	3			0.3	1
7	1000	250	750	4	0.076	0.089	0.8	0.27	2	2	-1.0986123	0.58593	0.4	1
8	1000	250	750	68	0.094	0.166	5.2	7.33	13	55	0.3437715	0.73338	6.8	1
11	1000	250	750	397	0.168	0.326	51.2	35.87	128	269	-0.3559312	5.45761	39.7	1
12	1000	250	750	312	0.327	0.506	28.4	32.13	71	241	0.1235048	0.46108	31.2	2
4	1000	250	750	123	0.508	0.624	7.6	13.87	19	104	0.6013396	3.7684	12.3	3
9	1000	250	750	33	0.63	0.68	3.6	3.2	9	24	-0.117783	0.04711	3.3	3
3	1000	250	750	19	0.681	0.725	0.8	2.27	2	17	1.0414539	1.52747	1.9	4
10	1000	250	750	13	0.728	0.764	0.8	1.47	2	11	0.6061358	0.40409	1.3	4
2	1000	250	750	14	0.766	0.803	0.8	1.6	2	12	0.6931472	0.55452	1.4	4
6	1000	250	750	12	0.809	0.842	0.8	1.33	2	10	0.5108256	0.27244	1.2	4



Row Labels	Sum of Goods	Sum of Bads	%Good	%Bad	WoE	Ivi
1	331	143	0.44133333	0.572	-0.2593	0.03389
2	241	71	0.32133333	0.284	0.1235	0.00461
3	128	28	0.17066667	0.112	0.42121	0.02471
4	50	8	0.06666667	0.032	0.73397	0.02544
Grand Total	**750**	**250**	1	1	0	
			0	0		
						0.08865

Table 25: Binning in Excel for Net_income

Total_Goo	N_Class	Min	Max	PCT_B	PCT_G	Bads	Goods	WOE	IVi	perct_obs	class
750	44	10332	16092	7.6	3.33	19	25	-0.82418	3.516482	4.4	1
750	501	16118	24171	57.2	47.73	143	358	-0.18092	1.712747	50.1	1
750	344	24234	32758	28	36.53	70	274	0.266021	2.270042	34.4	2
750	74	32786	39845	4.4	8.4	11	63	0.646627	2.586509	7.4	3
750	18	40168	45716	2	1.73	5	13	-0.1431	0.03816	1.8	3
750	6	45880	51882	0.4	0.67	1	5	0.510826	0.13622	0.6	3
750	7	53351	61078	0	0.93	0	7			0.7	3
750	2	65341	65813	0.4	0.13	1	1	-1.09861	0.292963	0.2	3
750	1	69577	69577	0	0.13	0	1			0.1	3
750	1	88073	88073	0	0.13	0	1			0.1	3
750	1	97210	97210	0	0.13	0	1			0.1	3
750	1	223300	223300	0	0.13	0	1			0.1	3



Row Labels	Sum of Goods	Sum of Bads	%good	%bad	WoE	IVI
1	383	162	0.510667	0.648	-0.23817	0.032709
2	274	70	0.365333	0.28	0.266021	0.0227
3	93	18	0.124	0.072	0.543615	0.028268
Grand Total	750	250	1	1	0	0
			0	0	#DIV/0!	
						0.083678

Plots