Retail Credit Risk Modeling Assignment

Ian Lee
Joshua Kim
Siyu Jia
Ti Zhang
Hanhua Zhang
Yanrong Huo
N'dah Ekissi

December 11, 2022

Contents

1 Introduction

1.1 Purpose

Granting credit to individual or corporate customers is the core business of retail and commercial banks. In doing so, banks need to have adequate systems to decide to whom to grant loans. Credit scoring is a key risk assessment technique to analyze and quantify the creditworthiness of the applicant. *Behavioral credit scoring* refers to credit scoring models for existing clients of a firm, as opposed to new clients, for which the scoring agent has accumulated performance history. This study proposes a behavioral scoring model to manage existing credit card customers in a bank. The report describes the process of developing a data-driven small-business behavioral score model and provides the rationale for model development, including details about data source, development procedure, model assessment, and model limitations and assumptions.

1.2 Portfolio

The attached data set has a total of 506 distinct factors, which are well detailed in the chart Business Bureau variables, describing their purposes in the process of analysing the Small Business. Among such characteristics, demographic data such as age, phone number, postal code, and address cannot be utilised in credit choices; nonetheless, the other aspects are taking into consideration, such as ”Number of active trades”, ”Worst rating on credit card trades”, ”Average balance of open installment trades verified in past 12 months”, ”Total positive balance of all mortgage trades verified in past 12 months”, ”number of mortgage trades with delinquencies in past 12 months”, ”Total monthly obligation for all accounts”, and ”Number of public record bankruptcies”. These parameters combine together with certain weights of assessment allocated to create a portfolio for the Small Business.

1.3 Model Use

A business model outlines how a company will make money. Typically if someone wants to launch a business, it is appropriate that one should take the time to identify the model that will best serve his objectives and include details about its design in the business plan and market analysis. There are certainly many types of business models. In our case, we are investigating in the use of a Small Business behavior score model.

Furthermore, the development of a business model, which is introduced specifically in *Section 1.5*, contains determining business objectives, data preparation, model development, model approval, model deployment, and model monitoring. A small business behavior score model will gather customers' (small retail business) credit application information along with their references to gener-

ate a score representing their credit reliability. Then the next step in modeling is to carefully identify the characteristics of the company and the principal owner that are linked to bad payment behavior. These connections should be transformed into statistical probabilities that rank small businesses according to their likelihood of engaging in delinquent behavior. An effective scoring model will proceed to rank risks appropriately even if the economy is struggling. It can determine whether a company is low, moderate, or high in risk by estimating its future payment performance. With the aid of these scoring systems, the user can automatically accept or reject credit. A small business behavior score model does not have to be complex; it can be simple if all the crucial components are present and are fairly weighed.

1.4 Economic and Market Outlook

Global economy has been slowed due to geopolitical uncertainties, high inflation, and high interest rates. There are still various downside factors that will maintain tightening monetary policy. This will add further downside pressure on our economy. In addition, high interest rates is causing expensive financial costs for many customers. Overall, because of the recession, many small business or customers will face extremely difficult to keep their payments, and this will eventually bring negative impact to financial institutions. Thus, credit risk models will be extremely important for the financial institutions to minimize their potential losses.

1.5 Model Development Process

There are four major model development steps in this project. Firstly, we perform variable reduction, including techniques like pre-screening, univariate screening, and mutivariate screening analysis, to remove variables that we deem not to used. This also helps with reducing potential multi-collinearity or overfitting in the model. Secondly, we fit step-wise logistic regression to obtain an optimal model for the behavioral score, which is followed by building a scorecard to obtain a more usable and readable format. Lastly, we conduct a series of scorecard assessments such as rank-ordering, population stability, and scorecard benchmarking to analyze whether the model is strong and robust.

2 Data

2.1 Data Sources

The data source given for the project in developing the retail business behavioral credit score model includes mostly external data that are composed of credit bureau information, i.e. business bureau and consumer bureau variables. These variables are explained in other given dictionaries. As for business bureau variables, business names, addresses, phone numbers, number and volume of trades, credit limits, and various public records of business operations have been listed. As for consumer bureau variables, almost 2000 factors have been provided, where majority of them have strong relationship with credit liabilities of small retail businesses that our model seeks to explore.

2.2 Timeframes

As for business bureau variables, they contain 6, 12, and 24-month historical data from the fixed observation point. For example, the number of satisfactory trades, credit cards trades, demand loan trades, and installment trades all follow 6, 12, and 24-months observation periods. As for consumer bureau variables, 6-month lookback period from the observation point is frequently used. For example, the number of revolving trades, and installment trades are recorded every six months.

2.3 Target Variable

We defined the binary variable of whether the customer defaults in the next 12 months as our target variable. Formally speaking, this is the indicator function on the event that the sum of $t1, t2, \ldots, t12$ variables is greater than 0. More specifically, we defined the default risk as the uncertainty on which a lender takes when a borrower is unable to fulfill the required payments on time.

2.4 Population Exclusions

To make our dataset more representative of real-life loan applicants, it is necessary to exclude particular demographic groups. According to the original database, we considered excluding customers that are widely held or deceased. For deceased customers, they do not generate credit records, resulting in inaccurate credit records; therefore, it will lower the quality of the dataset. For widely held customers, they may also mislead the predictions of database because they cannot provide an exact 1-on-1 standard. Overall, the data contains about 9028 customers, but there are 11 deceased customers and 5 widely held customers. Hence only 0.177% of the data is removed after the exclusions, so the impact on the sample size is minor, but with more reliable training data.

2.5 Modeling Population

	Population	Rate
Default	900	9.9867%
Not Default	8112	90.0133%

Table 1: *default rate from total population*

	Jan 2014	Apr 2014	July 2014	Oct 2014
Default	219	236	206	239
Not Default	1967	2004	2039	2102
Default Rate	10.0183%	10.5357%	9.1759%	10.2093%

Table 2: *default rate by observation points*

2.6 Explanatory Variables

We create a total of 11 new explanatory variables listed below, which are derived from monthly debit and credit transactions recorded over 12 months, to better capture the distributions, trends, and cross-relation of debit and credit data.

- maximum over 12 months of each debit and credit
- minimum over 12 months of each debit and credit
- mean over 12 months of each debit and credit
- standard deviation over 12 months of each debit and credit
- linear trend over 12 months of each debit and credit
- ratio between the debit and credit means

While monthly ratios of debit and credit transactions have been considered, the variance in each month is too large for any sensible modelling. For instance, some months may have larger debit transactions due to high holiday counts, so we use the average of monthly debit and credit transactions for computing the ratio for a more tightly clustered samples.

2.7 Segmentation

Segmentation refers to the process where an enterprise divides customers in the market into several customer groups according to a certain standard. Two types of segmentations we considered are experience/operational-based and statistical clustering. Given that there are a large number of explanatory variables and that we lack retail lending industry expertise, a natural choice for

us is to employ statistical clustering. Specifically, we performed k-means clustering over the 16 categorical variables, including "CUSTTYPE", "STARTUP", "VISACUST", and "DOCTOR_DENTIST_IND". Since there are over 500 features to potentially cluster populations with, one needs to carefully select a subset of them in order to avoid the curse of dimensionality. We believe the 16 categorical features are highly interpretable for business purpose and representative of customer profile at the same time.

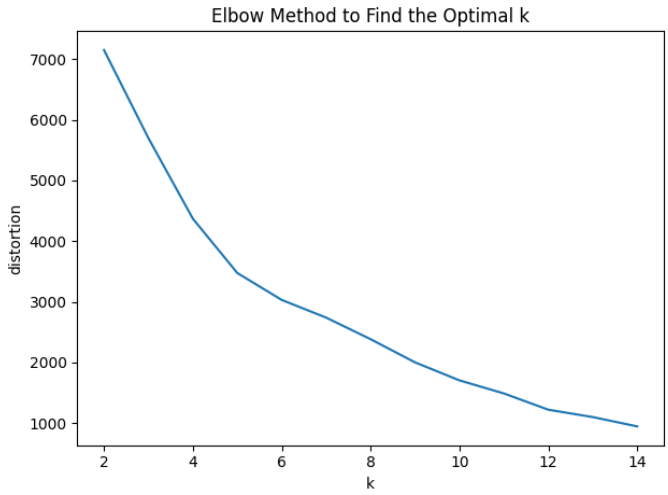


Figure 1: *elbow method for k-means clustering*

From *Figure 1*, a natural choice of the cutoff would be $k = 5$ where the slope of the plot significantly changes. Thus we segment the population into 5 distinct clusters for scorecard development purpose.

2.8 Sampling Methodology

We initially leave out data from the latest time point (Oct 2014) as the out-of-time validation (test) set. This is to prevent any look-ahead bias entering into our training data. Subsequently, training and in-time validation sets are split on the leftover data using $80 : 20$ split, a commonly used ratio in the field of machine learning. The purpose of train-test split is to obtain an unbiased evaluation of a trained model with a dataset that was unused during training. If the training set were to be used to measure model performance, the evaluation would be overly optimistic since the model has already seen the training set before.

Sample weights were also taken into account when splitting because there is a discrepancy in the number of data points between two classes. To achieve the

unbiased nature of the test set and represent this class imbalance equally during the split, we assigned the same ratio of two target classes to both training and test sets.

3 Scorecard Development

3.1 Modeling Considerations

An appropriate modeling technique should sufficiently represent the relationship between the explanatory variables and the target. Indeed a machine learning approach is suitable for the purpose of this project given the large number of provided features and the ability to incorporate additional or transformed features to enrich our credit scoring model. Moreover it can automatically capture non-linear patterns within the data, which can eventually offer efficient and concise model development and assessment process.

3.2 Variable Reduction

3.2.1 Pre-Screening

Pre-screening is a crucial initial step in the analysis of our high-dimensional database. This stage focuses on removing non-informative or redundant predictors from the model. The treatment of missing values and outliers may enhance the reliability, validity, and specificity of the model and reduce the number of input variables exclusively to those that are believed to be the most useful for predicting the target variable. In practice, financial institutions must meet standards or regulation to demonstrate the applications of data. There exists 592 explanatory variables in the provided dataset. Firstly, we remove all sparse columns because they contain elements that do not carry any information; any techniques to impute them will most likely result in carrying inaccuracies. For instance, some columns contain more than 9000 missing values in the dataset. Then, we drop variables that are not relevant to our model, e.g. certain features are constant or are unrelated to credit risk. Finally, we drop any forward-looking features or target proxies such as benchmark scores. In summary, the following has been considered during the pre-screening stage of variable reduction:

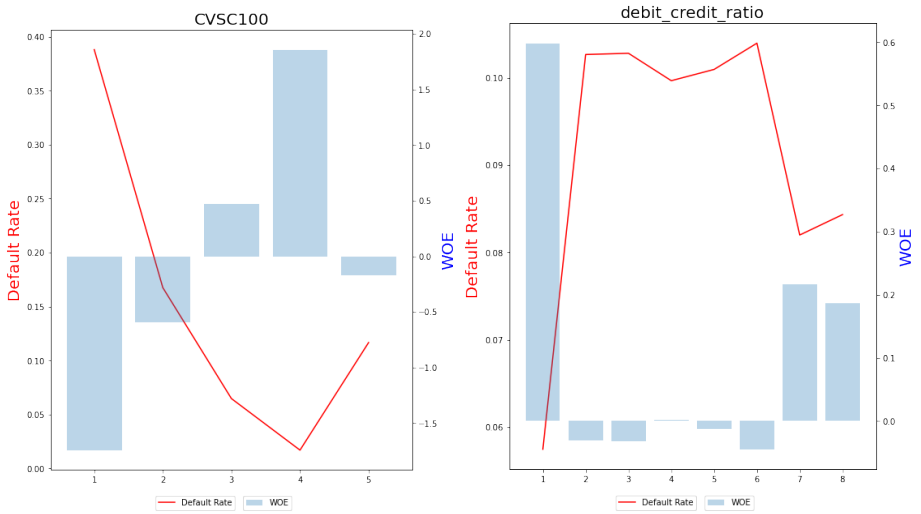
- sparse variables: columns where more than 10% of its values are missing
- population exclusions: “WIDELYHD”, “deceased”
- indices and metadata: “‘tu_seq_id”, “TIME_KEY”
- constant variables

3.2.2 Univariate Screening

The weight of evidence (WOE) demonstrates the relationship between predictive variable and target variable. In other words, WOE is a measure of the separation of good and bad customers, where the “bad customer” represents the defaults and the “good customers” the non-defaults. Moreover, information value (IV) measures how strong this relationship and its variable are, which can be utilized to select important variables for our model.

For the newly created variables, we initially bin them into 20 equally-sized buckets, followed by iterative grouping stage where bins with similar WOE, weak IV, or ill-behaved values from division by zero are collapsed.

Based on the the values of IV, we discover that "CVSC100" and "debit_credit_ratio" are significant and non significant variables, respectively. As *Figure 2a* illustrates, the significant variable demonstrates that WOE values tend to have monotonically increasing or decreasing linear trends. This shows a clear separation between groups, so we can conclude that the variable has strong predictive power for our model. On the other hand, *Figure 2b* displays that non-significant variable does not show a great separation between defaults and non-defaults. Consequently its relationship is hard to interpret; therefore, it will not provide powerful statistical insight for our model. In general, variables with low IV values tend to illustrate weak separation; therefore, we filter out variables that have low IV values, and remove 32 variables from our dataset.



(a) significant variable (b) non-significant variable

Figure 2: *comparison between significant and non-significant variables*

3.2.3 Multivariate Screening

In this step, we want to reduce collinearity based on variable clustering. A variable clustering technique is used to reduced multicollinearity by grouping sets of correlated variables. The algorithm we employed is described below:

1. A cluster is chosen for splitting. The selected cluster has the largest eigenvalue associated with the second principal component.
2. The chosen cluster is split into two clusters by finding the first two principal components, performing an orthoblique rotation, i.e. raw quartimax

rotation on the eigenvectors, and assigning each variable to the rotated component with which it has the higher squared correlation.

3. Variables are iteratively reassigned to clusters to maximize the variance accounted for by the cluster components.

We executed the clustering algorithm to obtain 19 variable clusters from our dataset, shown in *Figure 3a*, followed by computing the $1 - R^2$ ratio of each variable. For a good multivariate screening, we choose two variables – one with the lowest $1 - R^2$ ratio and one with the highest IV – that best represent each cluster with little loss of information. This leads to 10 clusters each with 2 variables, illustrated in *Figure 3b*. For example in cluster 0, variable WOE_CVPRAGG501 is the WOE of one month of the bank card balance information and has the smallest $1 - R^2$ ratio. Variable WOE_BCC5830 tells us WOE of total monthly payment on open and closed revolving bankcard trades, excluding derogatory trades, reported in the last 6 months and has the highest IV. In cluster 6, WOE_ALL2380 has WOE of total number of trades ever 90 or more days delinquent or derogatory, including external collections and the smallest $1 - R^2$ ratio. WOE_ALL6200 shows WOE of worst ever status on a trade, including external collections and the highest IV. After our multivariate screening, 19 variables are proceed to the next step.

Cluster	Variable	RS_Own	RS_NC	RS_Ratio	IV
0	WOE_BCC5830	0.616866	0.310722	0.555848	0.344788
0	WOE_CVPRAGG501	0.689049	0.272604	0.427486	0.285129
0	WOE_BCA5030	0.660815	0.317916	0.497277	0.265103
0	WOE_CVPRAGG907	0.654874	0.214634	0.439446	0.242103
0	WOE_BCC3515	0.576050	0.258073	0.571417	0.223157
0	WOE_CVPRTRV04	0.365403	0.081994	0.691278	0.207149
0	WOE_CVPRAGG512	0.546847	0.160561	0.539828	0.185708
0	WOE_CVPRAGG519	0.504497	0.138785	0.575354	0.173707
0	WOE_CVPRTRV14	0.204604	0.082835	0.867233	0.167161
0	WOE_CVPRWALSHR01	0.449659	0.159069	0.654443	0.164800
0	WOE_IQT9420	0.131379	0.064947	0.928953	0.160785
0	WOE_IQT9410	0.123423	0.056345	0.928917	0.144419
0	WOE_CVPRWALSHR02	0.344913	0.126146	0.749653	0.144345
0	WOE_BCC3510	0.434083	0.129193	0.649877	0.133617
0	WOE_REV5030	0.488275	0.152231	0.603614	0.122651
0	WOE_CVPRTRV12	0.179456	0.056366	0.869557	0.119218
0	WOE_REV5020	0.455254	0.135721	0.630290	0.114936
0	WOE_CVPRAGG102	0.340388	0.089795	0.724685	0.108292

(a) lowest $1 - R^2$ & highest IV in Cluster 0

Cluster	Variable	RS_Ratio	IV
0	WOE_CVPRAGG501	0.427486	0.285129
0	WOE_BCC5830	0.555848	0.344788
1	WOE_ALL2327	0.382940	0.424131
1	WOE_REV2328	0.315473	0.335407
2	WOE_TBSBC104S	0.392312	0.329792
2	WOE_max_ks_max_util_3mos	0.553610	0.625246
3	WOE_dda_sum_Acc_Db_Bal	0.413178	0.710410
3	WOE_cust_max_dlq_3mos	0.552252	1.225009
4	WOE_TBSG001B	0.219789	0.705309
4	WOE_TBSG001B	0.219789	0.705309
5	WOE_CVSC100	0.561365	0.978108
5	WOE_BCC7110	0.433243	0.586483
6	WOE_ALL2380	0.267426	0.200994
6	WOE_ALL6200	0.772311	0.328312
7	WOE_dda_av_bal	0.259215	0.499720
7	WOE_dda_avg_dly_dep_amt_L90	0.249659	0.393671
8	WOE_CVPRRVLR07	0.776425	0.404502
8	WOE_CVPRTPR212	0.322869	0.307298
9	WOE_TBSSC100	0.805003	0.669620
9	WOE_TBSBR34S	0.286979	0.610942

(b) final result after multivariate screening

Figure 3: *multivariate screening*

3.3 Model Fitting

From the previous steps, we have eliminated all but 19 candidate variables for the regression model. We now fit a stepwise backward logistic regression for 5 segmented population groups using the training set as follows:

1. Fit a logistic regression with n candidate variables and calculate its BIC score.
2. From the set of all subsets with $n-1$ variables, find the subset with the lowest BIC score when fitted to a logistic regression.
3. If the new BIC score is lower than the old one, update the model using the one fitted with the optimal subset.
4. Repeat Step 1-3 until the minimum BIC score is achieved.

Table 3: *19 final variable candidates*

Parameters	Model 0	Model 1	Model 2	Model 3	Model 4
WOE_cust_max_dlq_3mos	✓	✓	✓	✓	✓
WOE_dda_avg_dly_dep_amt_L90		✓	✓	✓	✓
WOE_TBSG001B	✓	✓		✓	✓
WOE_CVSC100	✓	✓		✓	
WOE_ALL2327	✓	✓	✓		✓
WOE_dda_av_bal	✓			✓	
WOE_REV2328				✓	✓
WOE_TBSBR34S		✓	✓		
WOE_CVPRRVLR07					✓
WOE_max_ks_max_util_3mos		✓			
WOE_CVPRAGG501		✓	✓		
WOE_BCC7110	✓		✓		
WOE_TBSSC100			✓		
WOE_dda_sum_Acc_Db_Bal			✓		
WOE_CVPRTPR212				✓	
WOE_BCC5830	✓				
WOE_ALL2380				✓	
WOE_ALL6200					
WOE_TBSBC104S					

Table 4: *logistic regression summary for cluster 0*

Dep. Variable:	target	**No. Observations:**	548
Model:	Logit	**Df Residuals:**	541
Method:	MLE	**Df Model:**	6
Date:	Sun, 11 Dec 2022	**Pseudo R-squ.:**	0.01223
Time:	13:50:49	**Log-Likelihood:**	-182.81
converged:	True	**LL-Null:**	-185.07
Covariance Type:	nonrobust	**LLR p-value:**	0.6057

	coef	**std err**	**z**	**P> |z|**	**[0.025**	**0.975]**
WOE_ALL2327	-0.6772	0.246	-2.756	0.006	-1.159	-0.196
WOE_BCC5830	0.6995	0.280	2.500	0.012	0.151	1.248
WOE_BCC7110	-0.6345	0.239	-2.652	0.008	-1.104	-0.166
WOE_CVSC100	-0.6705	0.169	-3.959	0.000	-1.002	-0.339
WOE_TBSG001B	-0.8300	0.193	-4.305	0.000	-1.208	-0.452
WOE_cust_max_dlq_3mos	-1.2393	0.154	-8.057	0.000	-1.541	-0.938
WOE_dda_av_bal	-0.9638	0.191	-5.052	0.000	-1.338	-0.590

Table 5: *logistic regression summary for cluster 1*

Dep. Variable:	target	**No. Observations:**	2465
Model:	Logit	**Df Residuals:**	2457
Method:	MLE	**Df Model:**	7
Date:	Sun, 11 Dec 2022	**Pseudo R-squ.:**	-0.3106
Time:	13:50:51	**Log-Likelihood:**	-1154.2
converged:	True	**LL-Null:**	-880.69
Covariance Type:	nonrobust	**LLR p-value:**	1.000

	coef	**std err**	**z**	**P> |z|**	**[0.025**	**0.975]**
WOE_ALL2327	-0.4400	0.097	-4.518	0.000	-0.631	-0.249
WOE_CVPRAGG501	-0.4277	0.109	-3.917	0.000	-0.642	-0.214
WOE_CVSC100	-0.3588	0.065	-5.497	0.000	-0.487	-0.231
WOE_TBSBR34S	-0.2636	0.087	-3.045	0.002	-0.433	-0.094
WOE_TBSG001B	-0.5120	0.072	-7.127	0.000	-0.653	-0.371
WOE_cust_max_dlq_3mos	-1.0165	0.073	-13.886	0.000	-1.160	-0.873
WOE_dda_avg_dly_dep_amt_L90	-0.5723	0.088	-6.483	0.000	-0.745	-0.399
WOE_max_ks_max_util_3mos	-0.2615	0.084	-3.104	0.002	-0.427	-0.096

Table 6: *logistic regression summary for cluster 2*

Dep. Variable:	target	**No. Observations:**	1120
Model:	Logit	**Df Residuals:**	1112
Method:	MLE	**Df Model:**	7
Date:	Sun, 11 Dec 2022	**Pseudo R-squ.:**	0.3871
Time:	13:50:52	**Log-Likelihood:**	-146.90
converged:	True	**LL-Null:**	-239.68
Covariance Type:	nonrobust	**LLR p-value:**	1.309e-36

	coef	**std err**	**z**	**P> |z|**	**[0.025**	**0.975]**
WOE_ALL2327	-0.9680	0.229	-4.225	0.000	-1.417	-0.519
WOE_BCC7110	-1.0230	0.269	-3.810	0.000	-1.549	-0.497
WOE_CVPRAGG501	-0.9387	0.299	-3.145	0.002	-1.524	-0.354
WOE_TBSBR34S	0.6871	0.262	2.626	0.009	0.174	1.200
WOE_TBSSC100	-0.9944	0.259	-3.833	0.000	-1.503	-0.486
WOE_cust_max_dlq_3mos	-2.1299	0.387	-5.508	0.000	-2.888	-1.372
WOE_dda_avg_dly_dep_amt_L90	-0.8669	0.249	-3.476	0.001	-1.356	-0.378
WOE_dda_sum_Acc_Db_Bal	-2.3990	0.309	-7.768	0.000	-3.004	-1.794

Table 7: *logistic regression summary for cluster 3*

Dep. Variable:	target	**No. Observations:**	604
Model:	Logit	**Df Residuals:**	596
Method:	MLE	**Df Model:**	7
Date:	Sun, 11 Dec 2022	**Pseudo R-squ.:**	0.2003
Time:	13:50:53	**Log-Likelihood:**	-139.82
converged:	True	**LL-Null:**	-174.84
Covariance Type:	nonrobust	**LLR p-value:**	1.442e-12

	coef	**std err**	**z**	**P> |z|**	**[0.025**	**0.975]**
WOE_ALL2380	1.2456	0.456	2.733	0.006	0.352	2.139
WOE_CVPRTPR212	-1.8095	0.325	-5.565	0.000	-2.447	-1.172
WOE_CVSC100	-0.7059	0.208	-3.398	0.001	-1.113	-0.299
WOE_REV2328	-1.1454	0.330	-3.469	0.001	-1.792	-0.498
WOE_TBSG001B	-1.0427	0.213	-4.900	0.000	-1.460	-0.626
WOE_cust_max_dlq_3mos	-0.6201	0.143	-4.348	0.000	-0.900	-0.341
WOE_dda_av_bal	-0.9978	0.233	-4.280	0.000	-1.455	-0.541
WOE_dda_avg_dly_dep_amt_L90	-1.6434	0.299	-5.495	0.000	-2.230	-1.057

Table 8: *logistic regression summary for cluster 4*

Dep. Variable:	target	**No. Observations:**	596
Model:	Logit	**Df Residuals:**	590
Method:	MLE	**Df Model:**	5
Date:	Sun, 11 Dec 2022	**Pseudo R-squ.:**	0.03759
Time:	13:50:55	**Log-Likelihood:**	-215.17
converged:	True	**LL-Null:**	-223.58
Covariance Type:	nonrobust	**LLR p-value:**	0.004873

	coef	**std err**	**z**	**P$>\|$z$\|$**	**[0.025**	**0.975]**
WOE_ALL2327	-1.1776	0.280	-4.211	0.000	-1.726	-0.630
WOE_CVPRRVLR07	-0.6918	0.198	-3.493	0.000	-1.080	-0.304
WOE_REV2328	0.9859	0.333	2.965	0.003	0.334	1.638
WOE_TBSG001B	-1.3279	0.174	-7.644	0.000	-1.668	-0.987
WOE_cust_max_dlq_3mos	-0.8741	0.140	-6.239	0.000	-1.149	-0.599
WOE_dda_avg_dly_dep_amt_L90	-1.6897	0.201	-8.406	0.000	-2.084	-1.296

As an example, the cluster 0 summary indicates that the WOE of total monthly payment on open and closed revolving bankcard trades (BCC5830) has an positive effect, meanwhile the WOEs of total number of trades ever 30 or more days delinquent or derogatory (ALL2327), the overall balance to credit amount ratio on open revolving bankcard trades reported (BCC7110), the Credit Vision Risk Score (CVSC100), the number of 30+ dpd ratings in past 12 months (TBSG001B), the max delinquencies in last 3 months (cust_max_dlq_3mos), and the average outstanding balance (dda_av_bal) all result in a negative effect. Among them, the max delinquencies in last 3 months has the most significant negative effect. This aligns with our business understanding since a high rate of delinquencies would be a strong indicator of financial instability, as well as all other negatively correlated variables. On the other hand, large monthly bankcard trades would suggest the customer is financially active and stable, agreeing with the positive correlation.

3.4 Scorecard Scaling

Adhering to the business convention, we aim to scale the scores into three-digit integers for ease of implementation and high interpretability. The scores are calibrated such that a customer with (non-defaulted : defaulted) odds of 50 : 1 is assigned a score of 600, with odds doubling every 20 points. Hence the scaling components are calculated as follows:

- $factor = \frac{PDO}{\ln(2)} = \frac{20}{\ln(2)} \approx 28.8539$
- $offset = score_0 - factor \cdot \ln(odds) = 600 - \frac{20}{\ln(2)} \cdot \ln(50) \approx 487.1229$

Subsequently, each score on the scorecard is determined as

$$score_{i,j} = \frac{offset}{n} - factor \cdot \beta_i \cdot WOE_{i,j}$$

where β_i is the logistic regression coefficient for the explanatory variable i, $WOE_{i,j}$ is the weight of evidence of the group j for explanatory variable i, and n is the number of explanatory variable in the logistic regression model. To obtain the final score for a customer, one simply needs to sum the scores corresponding to each variable. See *Table 9*, *10*, *11*, *12*, and *13* for the final scorecard of each cluster.

Table 9: *final scorecard for cluster 0*

	variable	GRP	WOE	score
0	ALL2327	1	0.413286	78
1	ALL2327	2	-0.447208	61
2	ALL2327	3	-1.463340	41
3	ALL2327	4	-0.109909	67
4	BCC5830	1	0.872137	52
5	BCC5830	2	0.244914	65
6	BCC5830	3	-0.693143	84
7	BCC5830	4	-0.155451	73
8	BCC5830	5	-0.109909	72
9	BCC7110	1	0.751054	83
10	BCC7110	2	-0.483722	61
11	BCC7110	3	-1.502516	42
12	BCC7110	4	-0.222646	66
13	BCC7110	5	-0.109909	68
14	CVSC100	1	-1.742633	36
15	CVSC100	2	-0.597481	58
16	CVSC100	3	0.469315	79
17	CVSC100	4	1.859388	106
18	CVSC100	5	-0.118791	67
19	TBSG001B	1	0.248217	76
20	TBSG001B	2	0.022387	70
21	TBSG001B	3	0.642846	85
22	TBSG001B	4	-1.555931	32
23	cust_max_dlq_3mos	1	0.579079	90
24	cust_max_dlq_3mos	2	-1.906722	1
25	cust_max_dlq_3mos	3	-2.584367	-23
26	cust_max_dlq_3mos	4	-2.671452	-26
27	dda_av_bal	1	-1.000382	42
28	dda_av_bal	2	-0.485122	56
29	dda_av_bal	3	0.025116	70
30	dda_av_bal	4	0.931760	95
31	dda_av_bal	5	0.777061	91

Table 10: *final scorecard for cluster 1*

	variable	GRP	WOE	score
0	ALL2327	1	0.413286	66
1	ALL2327	2	-0.447208	55
2	ALL2327	3	-1.463340	42
3	ALL2327	4	-0.109909	59
4	CVPRAGG501	1	0.459755	67
5	CVPRAGG501	2	0.033720	61
6	CVPRAGG501	3	-0.428041	56
7	CVPRAGG501	4	-1.208184	46
8	CVPRAGG501	5	-0.118791	59
9	CVSC100	1	-1.742633	43
10	CVSC100	2	-0.597481	55
11	CVSC100	3	0.469315	66
12	CVSC100	4	1.859388	80
13	CVSC100	5	-0.118791	60
14	TBSBR34S	1	0.080274	62
15	TBSBR34S	2	1.151736	70
16	TBSBR34S	3	-0.045780	61
17	TBSBR34S	4	-0.632657	56
18	TBSBR34S	5	-1.450989	50
19	TBSG001B	1	0.248217	65
20	TBSG001B	2	0.022387	61
21	TBSG001B	3	0.642846	70
22	TBSG001B	4	-1.555931	38
23	cust_max_dlq_3mos	1	0.579079	78
24	cust_max_dlq_3mos	2	-1.906722	5
25	cust_max_dlq_3mos	3	-2.584367	-15
26	cust_max_dlq_3mos	4	-2.671452	-17
27	dda_avg_dly_dep_amt_L90	1	-0.771559	48
28	dda_avg_dly_dep_amt_L90	2	-0.239723	57
29	dda_avg_dly_dep_amt_L90	3	0.482896	69
30	dda_avg_dly_dep_amt_L90	4	1.049503	78
31	dda_avg_dly_dep_amt_L90	5	0.777061	74
32	max_ks_max_util_3mos	1	1.087368	69
33	max_ks_max_util_3mos	2	0.616829	66
34	max_ks_max_util_3mos	3	0.017097	61
35	max_ks_max_util_3mos	4	-1.287173	51
36	max_ks_max_util_3mos	5	-0.354041	58

Table 11: *final scorecard for cluster 2*

	variable	GRP	WOE	score
0	ALL2327	1	0.413286	72
1	ALL2327	2	-0.447208	48
2	ALL2327	3	-1.463340	20
3	ALL2327	4	-0.109909	58
4	BCC7110	1	0.751054	83
5	BCC7110	2	-0.483722	47
6	BCC7110	3	-1.502516	17
7	BCC7110	4	-0.222646	54
8	BCC7110	5	-0.109909	58
9	CVPRAGG501	1	0.459755	73
10	CVPRAGG501	2	0.033720	62
11	CVPRAGG501	3	-0.428041	49
12	CVPRAGG501	4	-1.208184	28
13	CVPRAGG501	5	-0.118791	58
14	TBSBR34S	1	0.080274	59
15	TBSBR34S	2	1.151736	38
16	TBSBR34S	3	-0.045780	62
17	TBSBR34S	4	-0.632657	73
18	TBSBR34S	5	-1.450989	90
19	TBSSC100	1	-0.555370	45
20	TBSSC100	2	-0.780669	38
21	TBSSC100	3	0.233117	68
22	TBSSC100	4	1.041280	91
23	TBSSC100	5	1.801821	113
24	cust_max_dlq_3mos	1	0.579079	96
25	cust_max_dlq_3mos	2	-1.906722	-56
26	cust_max_dlq_3mos	3	-2.584367	-98
27	cust_max_dlq_3mos	4	-2.671452	-103
28	dda_avg_dly_dep_amt_L90	1	-0.771559	42
29	dda_avg_dly_dep_amt_L90	2	-0.239723	55
30	dda_avg_dly_dep_amt_L90	3	0.482896	73
31	dda_avg_dly_dep_amt_L90	4	1.049503	87
32	dda_avg_dly_dep_amt_L90	5	0.777061	80
33	dda_sum_Acc_Db_Bal	1	-0.972497	-6
34	dda_sum_Acc_Db_Bal	2	-0.694147	13
35	dda_sum_Acc_Db_Bal	3	-0.258289	43
36	dda_sum_Acc_Db_Bal	4	0.924821	125
37	dda_sum_Acc_Db_Bal	5	0.777061	115

Table 12: *final scorecard for cluster 3*

	variable	GRP	WOE	score
0	ALL2380	1	0.221593	53
1	ALL2380	2	-0.933247	94
2	ALL2380	3	-0.109909	65
3	CVPRTPR212	1	0.300540	77
4	CVPRTPR212	2	-1.044706	6
5	CVPRTPR212	3	-0.211593	50
6	CVPRTPR212	4	0.569830	91
7	CVPRTPR212	5	-0.118791	55
8	CVSC100	1	-1.742633	25
9	CVSC100	2	-0.597481	49
10	CVSC100	3	0.469315	70
11	CVSC100	4	1.859388	99
12	CVSC100	5	-0.118791	58
13	REV2328	1	0.397237	74
14	REV2328	2	-0.367427	49
15	REV2328	3	-1.184236	22
16	REV2328	4	-0.109909	57
17	TBSG001B	1	0.248217	68
18	TBSG001B	2	0.022387	62
19	TBSG001B	3	0.642846	80
20	TBSG001B	4	-1.555931	14
21	cust_max_dlq_3mos	1	0.579079	71
22	cust_max_dlq_3mos	2	-1.906722	27
23	cust_max_dlq_3mos	3	-2.584367	15
24	cust_max_dlq_3mos	4	-2.671452	13
25	dda_av_bal	1	-1.000382	32
26	dda_av_bal	2	-0.485122	47
27	dda_av_bal	3	0.025116	62
28	dda_av_bal	4	0.931760	88
29	dda_av_bal	5	0.777061	83
30	dda_avg_dly_dep_amt_L90	1	-0.771559	24
31	dda_avg_dly_dep_amt_L90	2	-0.239723	50
32	dda_avg_dly_dep_amt_L90	3	0.482896	84
33	dda_avg_dly_dep_amt_L90	4	1.049503	111
34	dda_avg_dly_dep_amt_L90	5	0.777061	98

Table 13: *final scorecard for cluster 4*

	variable	GRP	WOE	score
0	ALL2327	1	0.413286	95
1	ALL2327	2	-0.447208	66
2	ALL2327	3	-1.463340	31
3	ALL2327	4	-0.109909	77
4	CVPRRVLR07	1	0.473415	91
5	CVPRRVLR07	2	-0.510659	71
6	CVPRRVLR07	3	-1.188507	57
7	CVPRRVLR07	4	-0.118791	79
8	REV2328	1	0.397237	70
9	REV2328	2	-0.367427	92
10	REV2328	3	-1.184236	115
11	REV2328	4	-0.109909	84
12	TBSG001B	1	0.248217	91
13	TBSG001B	2	0.022387	82
14	TBSG001B	3	0.642846	106
15	TBSG001B	4	-1.555931	22
16	cust_max_dlq_3mos	1	0.579079	96
17	cust_max_dlq_3mos	2	-1.906722	33
18	cust_max_dlq_3mos	3	-2.584367	16
19	cust_max_dlq_3mos	4	-2.671452	14
20	dda_avg_dly_dep_amt_L90	1	-0.771559	44
21	dda_avg_dly_dep_amt_L90	2	-0.239723	69
22	dda_avg_dly_dep_amt_L90	3	0.482896	105
23	dda_avg_dly_dep_amt_L90	4	1.049503	132
24	dda_avg_dly_dep_amt_L90	5	0.777061	119

3.5 Scorecard Assessment

3.5.1 Rank-Ordering

We employ three techniques to assess how effective our scorecards rank order: Kolmogorov-Smirnov (KS) statistic, accuracy ratio (AR), and lift curve.

Kolmogorov-Smirnov Statistic The KS statistic measures the maximum difference between two cumulative distributions, i.e. distribution of non-defaulted and defaulted customers. More formally, it can be expressed as

$$KS = sup_s |F_D(s) - F_{ND}(s)|$$

where $F_D(s)$ is the cumulative distribution of defaults by score and $F_{ND}(s)$ for non-defaults. The Kolmogorov-Smirnov test, derived from the KS statistic as the name suggests, essentially answers the question "what is the probability that these two sets of samples were drawn from the same probability distribution?".

Table 14 summarizes the KS statistics and their corresponding scores calculated from six different datasets generated from train-val-test splits and three distinct time points (Oct 2014 is omitted as it is represented as the test set). We can observe that KS statistics, as well as the corresponding scores, are fairly consistent and high across all datasets, implying the model is capable of separating risky clients from those with low retail risk.

Table 14: *summary of KS statistics for various datasets*

	Train	Validation	Test (Oct 2014)	Jan 2014	Apr 2014	July 2014
KS statistic	0.6039	0.6248	0.5998	0.6365	0.5869	0.6155
score	499	503	505	500	492	499

Accuracy Ratio A cumulative accuracy profile (CAP), often utilized to visualize discrimination power of a model, is the cumulative number of positive outcomes along the y-axis versus the corresponding cumulative number of classifying parameter along the x-axis. The accuracy ratio (AR) is defined as the ratio of the area between the model CAP and random CAP and the area between the perfect CAP and random CAP. It ranges between 0 and 1, and the higher the value is, the stronger the model.

Similar to KS statistics, *Table 15* summarizes the accuracy ratios calculated from six different datasets generated from train-val-test splits and three distinct time points (Oct 2014 is omitted as it is represented as the test set). We can observe that AR are fairly consistent and high across all datasets, implying the discriminatory ability of the rating system is robust.

Table 15: *summary of accuracy ratio for various datasets*

	Train	Validation	Test (Oct 2014)	Jan 2014	Apr 2014	July 2014
AR	0.7423	0.7269	0.7011	0.7613	0.7333	0.7238

Lift Curve The lift is the ratio between the cumulative percentage of positive targets and the overall population percentage. It helps us determine how effectively the model can classify a relatively large pool of positive targets by selecting a relatively small number of samples.

Table 16 summarizes the lift values at 10% calculated from six different datasets generated from train-val-test splits and three distinct time points (Oct 2014 is omitted as it is represented as the test set). We can observe that the lift values are fairly consistent and high across all datasets, implying the model possess the capacity to extract high-risk customers immediately.

Table 16: *summary of lift value at 10% for various datasets*

	Train	Validation	Test (Oct 2014)	Jan 2014	Apr 2014	July 2014
$\text{lift}_{10\%}$	5.3338	5.5636	5.1068	5.4945	5.4661	5.4977

3.5.2 Population Stability

To verify how our model performance compares between different time points, we measure population stability using population stability index (PSI), an industry standard indicator in the field of credit scoring. PSI quantifies population differences by determining the shift between two sample distributions. More formally, it can be expressed as

$$PSI = \sum_{i=1}^{k} (N_i - B_i) \cdot \ln(\frac{N_i}{B_i})$$

where N_i is the percentage of the population in score range i for the new population, B_i for the base population, and k is the number of score bins. PSI less than 0.1 are generally considered to indicates no significant shift in distribution while PSI greater than 0.25 indicates significant shift.

Table 17 summarizes the comparisons between four distinct time points based on PSI. Given that all of them yields PSI less than 0.1, we deem that the model to be robust across time and that the scorecards are easily generalized on unseen data.

Table 17: *population stability index values between various time points*

	Jan 2014	Apr 2014	July 2014	Oct 2014
Jan 2014	0	0.0048	0.0062	0.0085
Apr 2014		0	0.0070	0.0039
July 2014			0	0.0102
Oct 2014				0

3.5.3 Scorecard Benchmarking

We conduct a benchmark analysis on the same data used to develop the logistic regression scorecards. CHAID (Chi-square automatic interaction detection) tree classifier is employed as the benchmark algorithm for its decent performance on binary classification tasks with categorical variables and high interpretability. Given that the purpose of the analysis is to provide a comparison to the original scorecards, we simplify the problem by not taking segmentation into account. The tree classifier is trained to have maximum depth of 5 layers. From the probability induced from the tree model, we determined the new score as follows:

$$score = offset + factor \cdot \ln(\frac{1}{prob} - 1)$$

where *prob* is the probability distribution obtained from the CHAID classifier and $offset$ and $factor$ are defined same as above for logistic regression. Following the benchmark training, we performed a series of identical rank-ordering analysis with KS statistic, AR, and lift curve.

Table 18 summarizes all three metrics derived from the CHAID tree classifier for six different datasets generated from train-val-test splits and three distinct time points (Oct 2014 is omitted as it is represented as the test set). To begin with, we can observe that the all three metrics are fairly consistent and high across the datasets, implying the model is reasonably strong and robust. More importantly, the CHAID classifier has higher AR on training set is worse on validation and test sets. This can be explained by the increased model complexity that the tree-based classifier offers, resulting in a small degree of overfitting. Furthermore, it is demonstrated that the KS statistics is generally better using the benchmark algorithm over the base logistic regression model.

Table 18: *summary of benchmark (CHAID tree classifier) performance*

	Train	Validation	Test (Oct 2014)	Jan 2014	Apr 2014	July 2014
KS statistic	0.6547	0.6246	0.6016	0.6617	0.6748	0.6383
score	549	554	554	554	547	535
AR	0.7964	0.7081	0.6854	0.7751	0.7992	0.6854
$\text{lift}_{10\%}$	5.5419	5.0301	5.2324	5.0367	5.5932	5.8382

4 Model Limitations and Assumptions

We make the assumption that no value has to be taken into account as an outlier while preparing the data. We have also presume that there is some multicollinearity among the explanatory factors when we cluster the variables. In addition, when fitting our model, it is assumed that there is a link between the characteristics and the target variable. Additionally, we have made the assumption that past conduct predicts future behaviour in customers and that all of the goods, rules, and processes would not change dramatically over time.

Apart from the assumptions, there are certainly some limitations for a logistic regression model. First, non-linear datasets are not suitable for the logistic regression; therefore, transformation from non linear to linear is a challenging process. Secondly, The interpretation of the model is relatively difficult because the weights is multiplicative rather than additive. In addition, logistic regression can be completely separated, implying that if there are features that perfectly separate the two classes, the logistic regression model can be no longer trained. The reason is that the weights for the features do not converge because the optimal weights are infinite.