A comprehensive example of credit risk management using Logistic Regression and Weight of Evidence measure

Phan Anh Khôi*

April 2024

Abstract

Effective credit risk management is paramount for financial institutions to rate the creditworthiness of customers and make informed decisions regarding loan approval. Among the classification models utilized for this purpose, Logistic Regression using Weight of Evidence (WoE) is a widely adopted approach due to its interpretability and predictive accuracy. In this study, the author tries to implement Logistic Regression with and without WoE, alongside seven alternative machine learning models, for the credit risk management problem. It is found that Logistic Regression using Weight of Evidence is the best model for the credit risk management problem as well as Linear Discriminant Analysis and Gaussian Naive Bayes. The author also proposes feature selection using Weight of Evidence with Logistic Regression, especially for high-dimensional data, to reduce the issue of overfitting associated with the model.

Introduction

Credit scoring remains a critical concern for loan institutions, particularly in the context of loan approval processes. The ability to accurately rate the likelihood of a borrower defaulting on their obligations is fundamental to managing credit risk effectively.

The idea behind credit risk management is the fact that there is a relationship between the likelihood that a person/entity defaults on their debt and some certain feature of these customers (their income/revenue, current status of their assets and liabilities, status of their previous loans,...).

*Student ID: 11212902 - DSEB 63 - National Economic University

Various methodologies have been employed to distinguish between "good" and "bad" customers in credit risk management. Among these, Logistic Regression using Weight of Evidence (WoE) has emerged as a standard approach used by banks and financial institutions due to its interpretability, prediction accuracy, and other advantages [1]. However, the rapid advancements in Machine Learning techniques have prompted exploration into alternative classification models for credit scoring.

This paper's goal is to find the best classification model for credit risk management by comparing Logistic Regression with WoE against other machine learning classification models such as Linear Discriminant Analysis (LDA) and k-Nearest Neighbor (KNN). The primary objective is to reimplement the credit risk assessment process used by financial institutions, categorizing customers as either "good" or "bad" based on their individual data profiles, and subsequently evaluating the performance of different modeling techniques.

To accomplish this objective, we leverage the "German Credit Risk - With Target" dataset available on Kaggle [2]. This dataset consists of information on 1000 loan applicants, each described by nine variables, and categorized as either "Good" or "Bad" credit risk. In the subsequent sections, we will dive into the specifics of this dataset and outline the methodology adopted for our comparative analysis.

Previous works and Methodology

Previous works

Desai et. al. (1996) [3] suggests that Neural Network is superior in credit risk modelling compare to other techniques. Yobas et. al. (2000) [4] compare Neural Network, Decision Tree, Linear Discriminant Analysis (LDA) and genetic algorithms and show that LDA outperform other models in the problem. Ong et. al. (2005) [5] show that credit scoring could enhance the prediction and saving time and effort, and suggest genetic programming for this problem. However, Finlay (2009) [6] finds that the difference in performance between Logistic Regression and genetic programming is not statistically significant.

A way to improve models' performance is feature selection. Bernhardsen et. al. (2007) [7] suggest genetic programming (GP) to find optimal explanatory variables. Abdou et. al. (2009) [8] compare Weight of Evidence (WoE) measure to GP and probit analysis (PA) and show that WoE is the best among these models.

Information Value and Weight of Evidence

Information Value (IV) measures the strength between the dependent and independent variables and it is used for variable selection, while WoE measures the strength of each grouped attribute, in separating defaulters and non-defaulters, where high negative values are equivalent to a high risk of default and vice versa [11].

Information Value of each bin is calculated using the formula:

$$IV = \sum (\text{Proportion of "Good" risk observations in the bin} - \text{Proportion of "Bad" risk observations in the bin}) * WoE \quad (1)$$

where WoE value of each bin is given by:

$$WoE = \ln\left(\frac{\text{Proportion of "Good" risk observations in the bin}}{\text{Proportion of "Bad" risk observations in the bin}}\right) \quad (2)$$

The use of WoE requires a transformation of numerical variables to categorical variables by binning. Siddiqi (2006) [10] suggests 3 criteria for a good binning strategy:

- Missing values should be grouped in a separate bin.
- Each bin should contain more than 5 percent of observations.
- No bin should have zero good or bad loan.

Bailey (2001) [9] recommends the following guideline for feature selection using WoE:

- Less than 0.03 : poor prediction.
- From 0.03 to less than 0.10 : weak prediction.
- From 0.10 to less than 0.30 : average prediction.
- From 0.30 to less than 0.50 : strong prediction.
- Over 0.50 : very strong prediction

Rickard Persson (2021) [11] notes some drawbacks of WoE measure technique. First, there might be a loss of information when we group numerical data into bins. Second, WoE measure is very sensitive to multicollinearity between variables in the dataset.

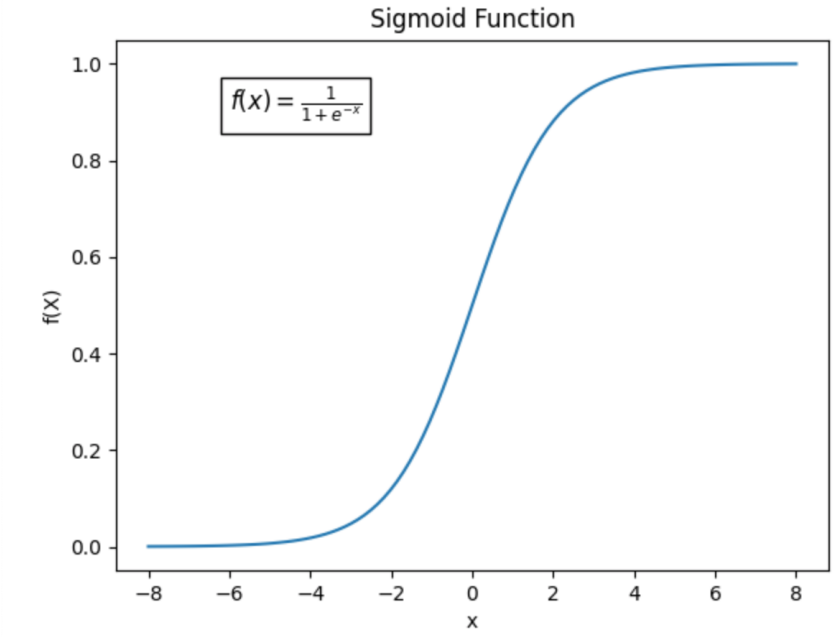

Figure 1: Graph of sigmoid function

Logistic Regression

Logistic regression can be used to predict whether the borrower has defaulted ('bad' risk) or not. The goal of Logistic Regression is to take an input vector $\mathbf{x}$ and to assign it to one of k discrete classes C_k where $k = 1, 2, ...n$.
For credit scoring problem, the simplest case, binary classification (k = 2) is used. Observations are classified as follow:

$$\begin{cases} C_1 & : \text{"Bad" risk} \\ C_0 & : \text{"Good" risk} \end{cases}$$

The posterior probability for class C_1 is given as:

$$P(C_1|\mathbf{x}) = \frac{P(\mathbf{x}|C_1)P(C_1)}{P(\mathbf{x}|C_1)P(C_1) + P(\mathbf{x}|C_0)P(C_0} = \frac{1}{1+e^{-a}} = \sigma(a)$$

where $a = \ln \frac{P(\mathbf{x}|C_1)P(C_1)}{P(\mathbf{x}|C_0)P(C_0)}$

We called $\sigma(a)$ sigmoid function, a function with value domain (0,1).

For a dataset $\phi(i)$ (denotes a transformation of x) and t_i (denotes the label) where $t_i \in \{0, 1\}$, i = 1,2,..,n. The probability that the model predict an input

belongs to a class is given by:

$$P(C_1|\phi) = y(\phi_i) = \sigma(w^T \phi_i)$$

and

$$P(C_0|\phi) = 1 - P(C_1|\phi)$$

the likelihood function can be written as

$$P(t|w) = \prod_{i=1}^{n} y_i^{t_i} (1 - y_i)^{1-t_i}$$

Assume that the log odds of the event $a = \ln \frac{P(\mathbf{x}|C_1)P(C_1)}{P(\mathbf{x}|C_0)P(C_0)} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ...$, we could estimate these parameters by maximizing the log-likelihood.

For comparison, the author also try 7 different state-of-the-art Machine Learning models, including:

- Linear Discriminant Analysis (LDA)
- K-Nearest Neighbor (KNN)
- Decision Tree
- Gaussian Naive Bayes
- Random Forest
- Support Vector Machine (SVM)
- XG Boost

Formal theoretical backgrounds of these models will not be specified in this article as they lie beyond the scope of the author's intended focus.

Evaluation metrics

In credit risk problem, predicting a customer who actually has 'bad' risk as 'good' (False Negative) is more unacceptable for loan institutions than predict a customer who actually has 'good' risk as 'bad' (False Positive), so accuracy score is not good enough to evaluate the model. To prioritize evaluating False Negative, the author use Recall score, calculated by the formula:

$$Recall = \frac{\text{True Positive}}{\text{True Positive + False Negative}}$$

Another metric used to evaluate the performance of models is Gini coefficient. Gini coefficient is given by

$$Gini = 2 * AUROC - 1$$

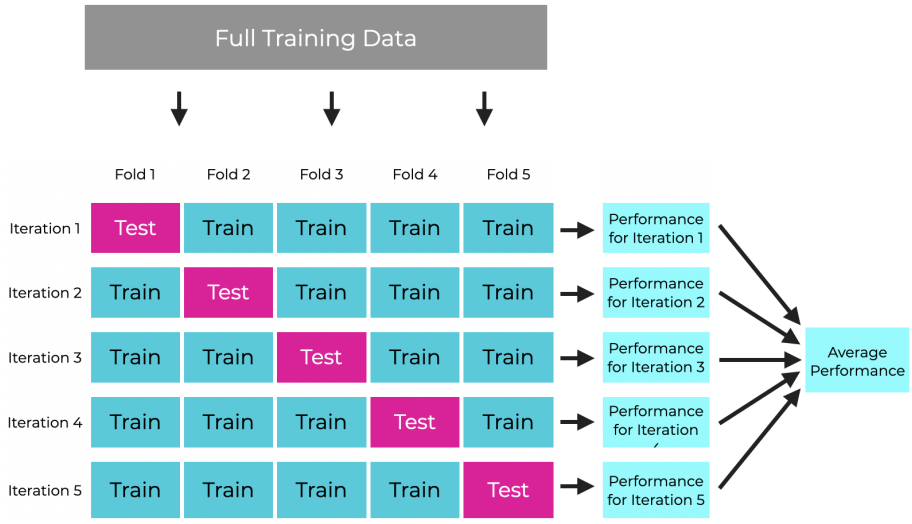

Figure 2: k-fold cross validation process with k = 5

where AUROC is Area Under Receiver Operating Characteristic curve. The Gini coefficient measures the separation power between 2 classes. The higher the Gini coefficient, the higher the separation between 'Good' and 'Bad', thus the better the classification model.

Due to the fact that Logistic Regression tends to be overfitting with high-dimensional data, the author also do the k-fold cross-validation within the train dataset to measure the level of overfitting. The train dataset is divided into 5 "folds" of approximately equal size. For each unique fold, we take the fold as test set and take the remaining groups as train set, then we fit the model to the training set and evaluate it on the test set. Overfitting exists when the model score varies significantly from the average cross-validation score.

Processing data

German Credit Risk Dataset

German Credit Risk Dataset contains information about 1000 observations, representing 1000 different loan applicants in 9 variables:

- Age (numeric).
- Sex (Male / Female).

- Job (0 - unskilled and non-resident, 1 - unskilled and resident, 2 - skilled, 3 - highly skilled).
- Housing (own, rent, or free).
- Saving accounts (little, moderate, quite rich, rich).
- Checking account (little, moderate, rich).
- Credit amount (numeric, in Deutsch Mark - DM).
- Duration (numeric, in month).
- Purpose (car, furniture/equipment, radio/TV, domestic appliances, repairs, education, business, vacation/others).

Observations are labeled as "Good" or "Bad" risk. The dataset is publicly available on Kaggle [2]. In total, there were 70.00% of observations labeled as "Good" and 30.00% labeled as "Bad", a slight imbalance but not significant to consider in preprocessing.

Exploratory Data Analysis & Preprocessing Data

Segmentation is the process of classifying observations into retail and non-retail customers based on the "Purpose" variable in the dataset. There are 8 values accepted in the Purpose variable: car, furniture/equipment, radio/TV, domestic appliances, repairs, education, business, vacation/others. All of these values represent retail customers, except for 'business'. Since only 9.70% of observations specified as "business" purpose, that is 97 observations, too little for further modeling and require special treatment in reality, the author just treats them as outliers, drops them, and assumes that we are working only with retail customers.

Since the target variable "Risk" in our dataset is distributed in a 7:3 ratio (29.46% "bad" after dropping "business" purpose, that is, more than 5%), sampling process is not required and we could remain this rate in the sample data.

After Exploratory Data Analysis (EDA), it can be seen that only 2 columns contain missing values: Saving accounts and Checking account.

Based on the nature of the columns, the author chose to keep these missing values to avoid information loss. It will be grouped in a separate bin later following the guidelines in [10].

The sample is divided into two parts: train data for building models and test data (validation data) for validating models. The proportions of train and test data are chosen at 80/20.

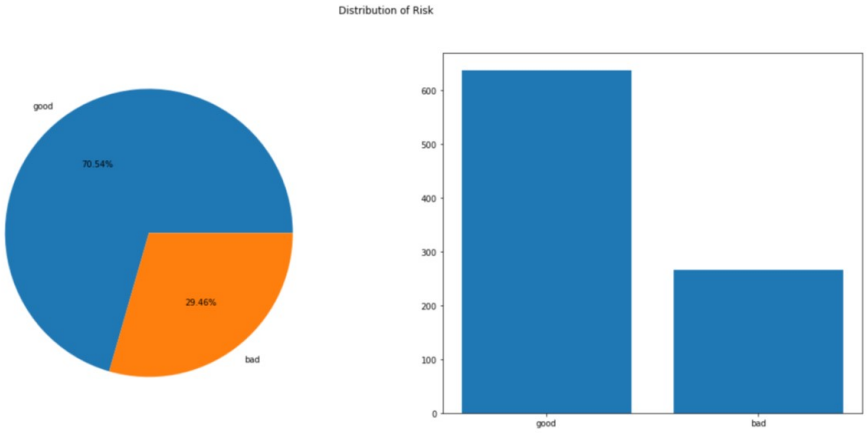

Figure 3: Distribution of target variable "Risk"

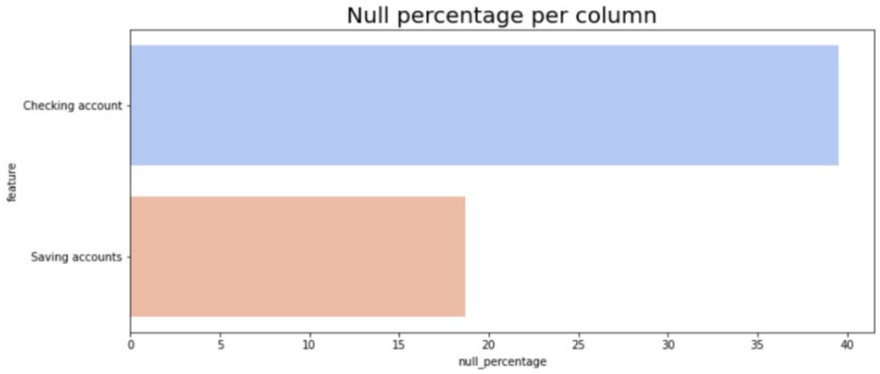

Figure 4: Null percentage of each column

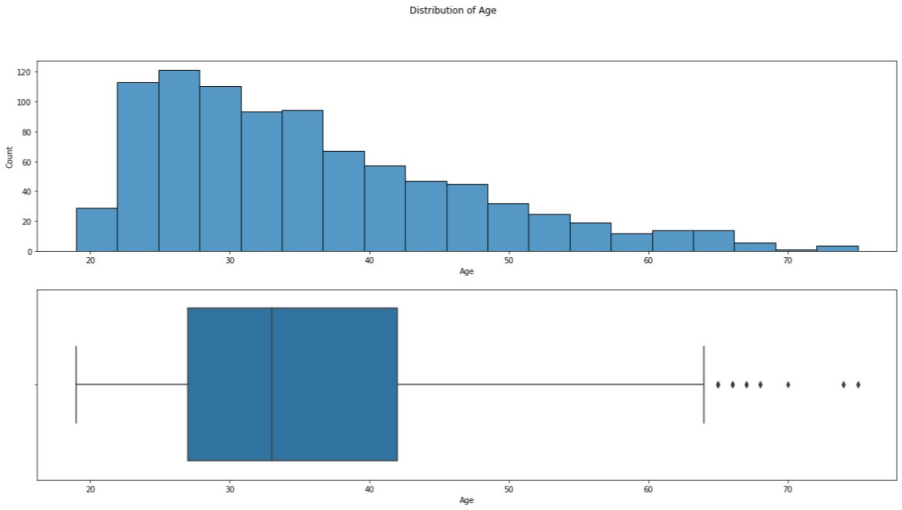

Figure 5: Distribution of "Age" variable

Outliers are special values which are very different from the rest of the population. Appropriate techniques, such as histogram and boxplot, is necessary to detect outliers. It is clear from plots of numerical variables that it is not necessary to remove outliers and there is no value that abnormal from the nature of the data.

Variable selection

Variables are selected by calculating Information Value (IV) and WoE following guidelines in [9]. Values in numerical variables (Age, Credit amount, Duration) are grouped into bins with the number of bins specified based on EDA. While both 'Age' and 'Credit amount' values could be grouped into 10 bins, 'Duration' could only be grouped in 6 bins to ensure that each bin has at least 5% of observations.

Based on Information Value, the author chooses medium and above predictors ($IV \geq 0.1$) and replaces bins by their WoE value to fit the models. The preprocessing procedure finished when we divided all numerical variables in the test set into bins with bin edges calculated previously in the train set, then one more time replaced these bins by WoE value of them in the train set.

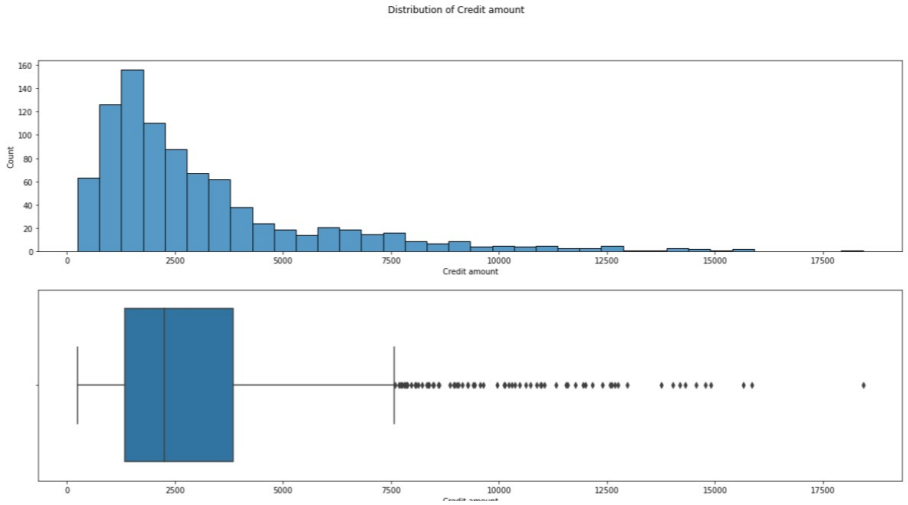

Figure 6: Distribution of "Credit amount" variable

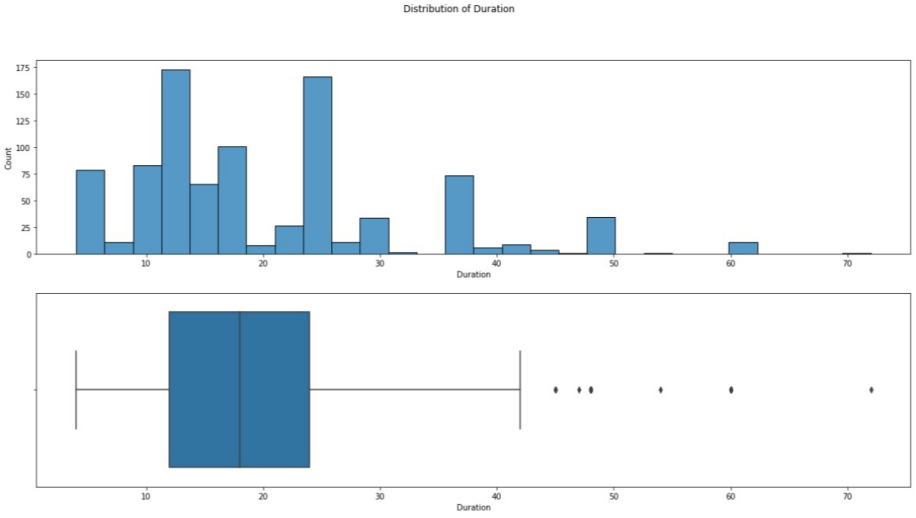

Figure 7: Distribution of "Duration" variable

```
Sex: Weak predictor
Job: Unless for prediction
Housing: Weak predictor
Saving accounts: Medium predictor
Checking account: Suspicious or too good predictor
Purpose: Weak predictor
Age_bin: Medium predictor
Credit_AMT_bin: Medium predictor
Duration_bin: Medium predictor
```

Figure 8: Variable selection using WoE value

Result

Model	Accuracy	Recall	Gini coef.	Avg. cross-validation Gini coef.
Logistic Regression	0.761	0.646	0.649	0.475
LDA	0.778	0.687	0.661	0.461
KNN	0.650	0.542	0.071	0.093
Decision Tree	0.683	0.627	0.254	0.235
Gaussian NB	0.750	0.732	0.591	0.426
Random Forest	0.750	0.657	0.593	0.458
SVM	0.678	0.496	-0.104	-0.008
XG Boost	0.739	0.658	0.521	0.463

Table 1: Model performances without using Weight of Evidence

To verify the effectiveness of using WoE, the author simply creates new dummy variables from categorical variables by using one-hot encoder, then fits the models and evaluates as described in the Methodology section. The result can be seen in Table 1.

Model	Accuracy	Recall	Gini coef.	Avg. cross-validation Gini coef.
Logistic Regression	0.750	0.635	0.626	0.528
LDA	0.767	0.661	0.613	0.529
KNN	0.739	0.654	0.452	0.407
Decision Tree	0.728	0.668	0.389	0.129
Gaussian NB	0.756	0.716	0.637	0.510
Random Forest	0.728	0.645	0.490	0.331
SVM	0.717	0.597	0.431	0.478
XG Boost	0.756	0.689	0.434	0.299

Table 2: Model performances using Weight of Evidence for feature selection

It is clear that feature selection using WoE significantly reduces the overfitting issue of Logistic Regression model. Average cross-validation Gini coefficient of Logistic Regression increases from 0.475 to 0.528, closer to the model performance at around 0.6, as WoE is applied. However, the overall Gini coefficient of the model slightly drops from 0.649 to 0.626 as a result of information loss due to removing insignificant variables.

LDA and Gaussian NB outperform other models to be the best classifiers among the remaining models. Their overall performance is even slightly better than Logistic Regression in 3/4 criteria, making them good alternatives to Logistic Regression in credit risk management problem.

Tree-based classification models, including Decision Tree and Random Forest, have lower Average cross-validation Gini coefficient after variable selection. This because Tree models tend to have very high variance and suffer the most from information loss. However, the overall Gini of Decision Tree improved after variable selection since Decision Tree is very likely to be overfitting as the number of features increases.

Conclusion

Logistic Regression using Weight of Evidence (WoE) has been used for decades in credit risk management by loan institutions. By employing a comprehensive comparative analysis, it is revealed that among 8 classification models surveyed, Logistic Regression with WoE, Linear Discriminant Analysis and Gaussian Naive Bayes emerge as superior models for credit risk management. Furthermore, the study advocates for the integration of feature selection using WoE with Logistic Regression, particularly in scenarios involving high-dimensional data, to mitigate the issue of overfitting associated with Logistic Regression models.

The study also successfully described the statistical-based credit risk management procedure used by loan institutions and introduced reasonable metrics to evaluate the performances of Machine Learning classification models for this problem. In the future, the author would like to extend this work to more complex, state-of-the-art models such as Neural Networks to achieve higher performance.

References

1 Thomas L. C., Consumer credit models: pricing, profit and portfolios, Oxford University Press, 2009.

2 German Credit Risk - With Target, Kaggle, https://www.kaggle.com/datasets/kabure/german-credit-data-with-risk.

3 Desai V. S., Crook, J. N., Overstreet, G. A., A comparison of neural networks and linear scoring models in the credit union environment, European Journal of Operational Research, 95(1), 24–37, 1996.

4 Yobas M.B., Crook, J.N., Ross, P., Credit scoring usingneural and evolutionary techniques, IMA Journal of Mathematics Applied in Business and Industry 11, 111–125, 2000.

5 Ong C., Huang J. Tzeng G., Building credit scoring models using genetic programming, Expert Systems With Applications, vol. 29, no. 1, pp. 41-47, 2005.

6 Finlay S.M., Are we modelling the right thing? The impact of incorrect problem specification in credit scoring, Expert Systems with Applications 36(5): 9065-907, 2009.

7 Bernhardsen E. Larsen, K., Modelling credit risk in the enterprise sector-further development of the SEBRA model, Economic Bulletin, vol. 78, no. 3, pp.102, 2007.

8 Abdou H.A., Genetic programming for credit scoring: The case of Egyptian public sector banks, Expert systems with applications, vol. 36, no. 9, pp. 11402-11417, 2009.

9 Bailey M., Credit scoring: The principles and practicalities, Kingswood, Bristol: White Box Publishing, 2001.

10 Siddiqi N., Credit Risk Scorecards: Developing and Implementing Intelligent Credit Scoring, John Wiley Sons, New York, NY, USA, 2006.

11 Rickard Persson, Weight of evidence transformation in credit scoring models: How does it affect the discriminatory power?, Lund University, Department of Statistics, 2021.