Customer Retention Analysis

BY

AGYAPONG PRINCE

Table of Contents

Section I: Introduction

1.1 Background

In the world of credit risk management, predicting whether a customer will default on their loan or credit card payment is crucial for lenders to make informed decisions about lending and managing risk. To achieve this goal, various modeling techniques based on Artificial Neural Network (ANN) and MARS model can be utilized. In this analysis, we will compare the performance of these two models in predicting customer default by evaluating their ROC curves and KS plots. The aim is to identify the best performing model that can accurately predict whether a customer is likely to default and help lenders minimize credit risk while maximizing profits.

1.2 Objectives

The objective of the study is to evaluate the performance of the predictive models, in predicting whether a customer will go bad or not. The study aims to determine which model is more effective in predicting customer risk and to identify the most important features that contribute to a customer going bad.

1.3 Exploratory Data Analysis

Exploratory plots are an essential tool for understanding the distribution, relationship, and trends of variables in a dataset. In this report, we will use exploratory plots to gain insights into the characteristics of the independent variables and their relationship with the dependent variable.

Table 1. Bad Rate Summary

BAD CLIENTS	FREQUENCY
NO	5496
YES	624
TOTAL	6120

From this table, out of the 6120 customers, 624 are bad while 5496 clients are good clients. Hence, about 10% of the customers are bad.

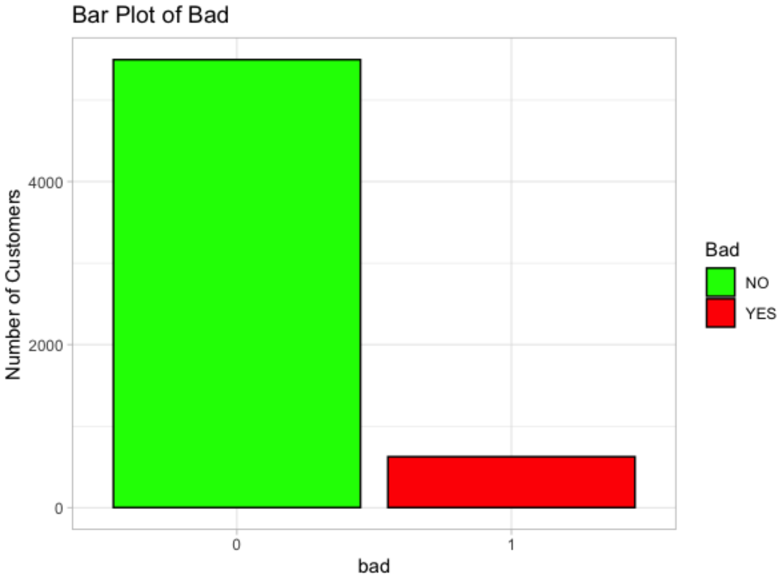

Figure 1. Bar plot of bad and good customers

This figure shows the number of bad and good customers. We can see that number of good customers are higher than that of bad customers.

Section II: Model Fitting

2.0 MARS

MARS (Multivariate Adaptive Regression Splines) is a non-parametric regression technique that can model complex relationships between predictors and a response variable. In this case, we want to build a MARS model to predict the value of "Bad" based on the following independent variables: "ID", "Months_On_Book", "Credit_Limit", "Opening_Balance", "Ending_Balance", "Over_limit_Amount", "Actual_Min_Pay_Due", "Total_Min_Pay_Due", "Net_Payments_During_Cycle", "Net_Purchases_During_Cycle", "Net_Cash_Advances_During_Cycle", "Net_Fees", "Net_Behavior_Fees_Billed_During", "Net_Concessions_Billed_During_Cy", "Score1", and "Utility".

ADVANTAGES

MARS (Multivariate Adaptive Regression Splines) is a flexible and powerful modeling technique that offers several advantages, including:

1. Non-linearity: MARS can capture non-linear relationships between the dependent and independent variables, unlike linear models such as logistic regression. This allows MARS to model complex relationships in the data, which can lead to better predictive performance.

2. Interpretability: MARS models are relatively easy to interpret compared to some other complex modeling techniques, such as neural networks. MARS produces a series of simple linear models, which can be easily understood and explained.

3. Feature selection: MARS automatically performs feature selection, identifying the most important variables in predicting the dependent variable. This can lead to simpler and more interpretable models and can also improve predictive performance by removing irrelevant variables.

4. Robustness: MARS is robust to outliers and missing data, which can be a common issue in real-world datasets. MARS can handle missing data by imputing missing values based on the available data, and outliers can be handled using robust regression methods.

DATA

To develop the MARS model, we must first prepare the data by performing cleaning, transformation, and standardization of the variables. We removed the variable "Score2" due to missing values. The model will be trained on a training dataset and evaluated using a validation dataset. The primary objective is to identify the optimal set of predictor variables and their transformations that minimize the prediction error for "Bad". Once the MARS model is trained and validated, it can be utilized to make predictions on new data.

RESULTS

Table 2. Coefficients of Mass Model

	BAD
(INTERCEPT)	0.2053778303
H(MONTHS_ON_BOOK-25)	-0.0008282260
H(25-MONTHS_ON_BOOK)	0.0032297040
H(UTILITY-0.7922)	0.4638243964
H(0.7922-UTILITY)	-0.0598780258
H(NET_FEES-18)	-0.1180426970

H(154-NET_PAYMENTS_DURING_CYCLE)	-0.0006179348
H(400-CREDIT_LIMIT)	0.0003374520
H(NET_FEES-74)	0.0147240488
H(NET_FEES-17)	0.1134185751
H(NET_PURCHASES_DURING_CYCLE--2)	-0.0354068643
H(NET_PURCHASES_DURING_CYCLE-2)	0.0353301928

The MARS model provides coefficients for each predictor variable, where each coefficient represents the change in the response variable associated with a one-unit change in the corresponding predictor variable, while holding all other variables constant.

Here are the interpretations for each coefficient in the MARS model:

- **INTERCEPT**: This is the estimated value of **Bad** when all predictor variables are equal to zero. In this case, the estimated value is 0.2053778303.
- **H(MONTHS_ON_BOOK-25)**: This is the estimated effect of the **MONTHS_ON_BOOK** variable on the response variable when **MONTHS_ON_BOOK** is less than or equal to 25. The coefficient value of -0.0008282260 suggests that **Bad** is expected to decrease by 0.0008282260 units for each unit decrease in **MONTHS_ON_BOOK** below 25.
- **H(25-MONTHS_ON_BOOK)**: This is the estimated effect of the **MONTHS_ON_BOOK** variable on the response variable when **MONTHS_ON_BOOK** is greater than 25. The coefficient value of 0.0032297040 suggests that **Bad** is expected to increase by 0.0032297040 units for each unit increase in **MONTHS_ON_BOOK** above 25.
- **H(UTILITY-0.7922)**: This is the estimated effect of the **UTILITY** variable on the response variable when **UTILITY** is less than or equal to 0.7922. The coefficient value of 0.4638243964 suggests that **Bad** is expected to increase by 0.4638243964 units for each unit increase in **UTILITY** below 0.7922.
- **H(0.7922-UTILITY):** This is the estimated effect of the **UTILITY** variable on the response variable when **UTILITY** is greater than 0.7922. The coefficient value of -0.0598780258 suggests that **Bad** is expected to decrease by 0.0598780258 units for each unit increase in **UTILITY** above 0.7922.
- **H(NET_FEES-18): This** is the estimated effect of the **NET_FEES** variable on the response variable when **NET_FEES** is less than or equal to 18. The coefficient value of -0.1180426970 suggests that **Bad** is expected to decrease by 0.1180426970 units for each unit decrease in **NET_FEES** below 18.
- **H(154-NET_PAYMENTS_DURING_CYCLE):** This is the estimated effect of the **NET_PAYMENTS_DURING_CYCLE** variable on the response variable when **NET_PAYMENTS_DURING_CYCLE** is less than or equal to 154. The coefficient value of -0.0006179348 suggests that **Bad** is expected to decrease by 0.0006179348 units for each unit decrease in **NET_PAYMENTS_DURING_CYCLE** below 154.
- **H(400-CREDIT_LIMIT):** This is the estimated effect of the **CREDIT_LIMIT** variable on the response variable when **CREDIT_LIMIT** is less than or equal to 400. The coefficient value

of 0.0003374520 suggests that **Bad** to increase by 0.0003374520 units for each unit increase in **CREDIT_LIMIT** below 400.

- **H(NET_FEES-74):** This is the estimated effect of the **NET_FEES** variable on the response variable when **NET_FEES** is greater than 74. The coefficient value of 0.0147240488 suggests that **Bad** expected to increase by 0.0147240488 units for each unit increase in **NET_FEES**
- **H(NET_FEES-17):** For values of **NET_FEES** less than or equal to 17, the model predicts an increase in **Bad** by 0.113 units, holding all other variables constant.
- **H(NET_PURCHASES_DURING_CYCLE--2):** For values of **NET_PURCHASES_DURING_CYCLE** less than -2, the model predicts a decrease in the **Bad** by 0.035 units, holding all other variables constant.
- **H(NET_PURCHASES_DURING_CYCLE-2):** For values of **NET_PURCHASES_DURING_CYCLE** greater than or equal to -2, the model predicts an increase in **Bad** by 0.035 units, holding all other variables constant.

2.4 Artificial Neural Network

Artificial Neural Networks (ANNs) are machine learning models that are designed to mimic the structure and function of biological neurons in the human brain. ANNs have been used in various applications, including image and speech recognition, natural language processing, and prediction models. We will explore the use of ANN to predict the likelihood of a credit card account holder being classified as 'Bad' based on several independent variables.

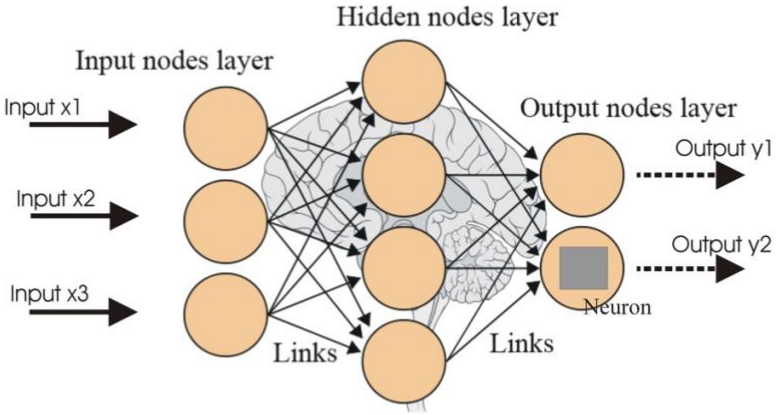

Figure 2. Artificial Neural Network

DATA

The data consists of 16 variables including the dependent variable 'Bad', which is binary indicating whether the customer is a bad debtor or not. There are 2 continuous variables and 14 categorical variables in the data.

The variable 'Score2' was removed from the data as it contains missing values. The variable 'ID' was also removed from the data as it is a unique identifier and does not contribute to the analysis.

Stepwise variable selection was used to select the important variables for the model. The final model includes the following variables: **'Months_On_Book', 'Credit_Limit', 'Opening_Balance', 'Ending_Balance', 'Over_limit_Amount', 'Actual_Min_Pay_Due', 'Net_Payments_During_Cycle', 'Net_Purchases_During_Cycle', 'Net_Cash_Advances_During_Cycle', 'Net_Fees', 'Net_Behavior_Fees_Billed_During', 'Net_Concessions_Billed_During_Cy', 'Utility'**.

The data was then divided into training and testing sets with 60% of the data used for training and 40% used for testing. This was done to evaluate the performance of the model on new data. The selected variables were used to fit an Artificial Neural Network model on the training set, and the performance was evaluated on the testing set.

MODEL ARCHITECTURE

The model architecture of the neural network was built using H2O's deep learning function, which is a type of artificial neural network. The goal of the model was to predict the value of the 'Bad' variable, which had two unique values (0/1), making this a binary classification problem.

The activation function used was 'Rectifier' and two hidden layers, each with 6 neurons, were specified in the model architecture. The model was trained for 100 epochs, with the 'train_samples_per_iteration' parameter set to -2, which means that H2O will use all the training data in each iteration.

ADVANTAGES of THE ANN

Here are some advantages of using an Artificial Neural Network (ANN) model for binary classification of the "Bad" outcome:

1. Non-linearity: ANNs are capable of modeling complex, non-linear relationships between input variables and the output. This makes them more powerful than linear models, which are often limited in their ability to capture complex patterns in the data.
2. Flexibility: ANNs can be configured in many ways, allowing for flexibility in the choice of network architecture and optimization algorithms. This makes it possible to tailor the model to the specific problem at hand, resulting in better performance and more accurate predictions.
3. Robustness: ANNs are robust to noise and missing data. They can generalize well to new data and are not as prone to overfitting as other models, such as decision trees.

4. Automatic feature extraction: ANNs can automatically extract features from the input data, which can be useful in situations where the most relevant features are not immediately obvious. This can improve the accuracy of the model and reduce the need for manual feature engineering.
5. Scalability: ANNs can handle large amounts of data and are highly scalable. They can be trained on large datasets using parallel processing, which can significantly reduce the training time.

Section III: MODEL COMPARISON

3.1 ROC

ROC curve is a widely used graphical representation of the performance of a binary classifier. It plots the true positive rate (sensitivity) against the false positive rate (1-specificity) for different threshold values. The area under the ROC curve (AUC) is a commonly used metric to compare the performance of different classifiers. A perfect classifier has an AUC of 1, while a random classifier has an AUC of 0.5.

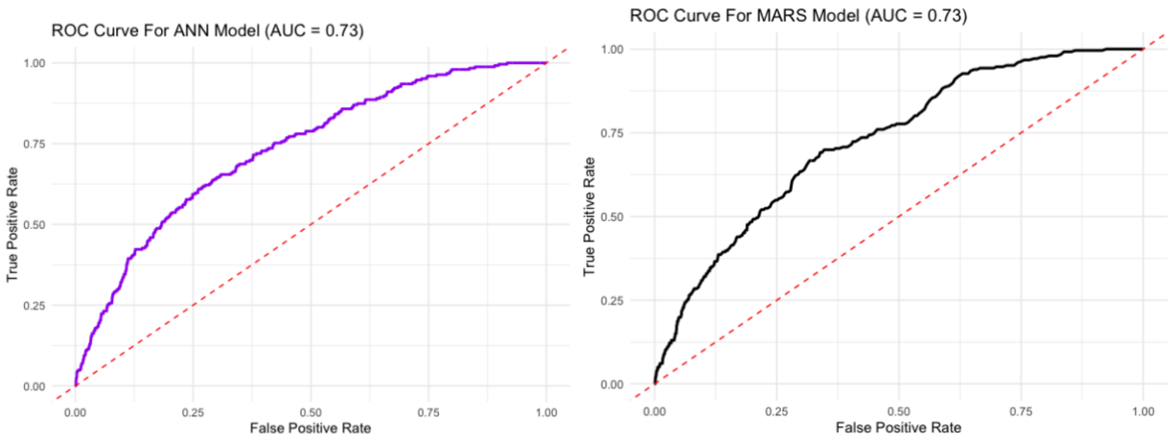

Figure 3. ROC curve

The AUC values of the Mars and ANN models can be compared to determine which model performs better in terms of overall classification performance. The AUC value of the Mars model is 0.73, and the AUC value of the ANN model is also 0.73. This suggests that both models perform better than a random classifier.

3.2 K-S

KS plot and statistic are commonly used to evaluate the accuracy of binary classification models, where the predicted and actual values are either 0 or 1. The KS statistic is a useful metric to compare the performance of different models and can help to identify which model is better suited for a particular task.

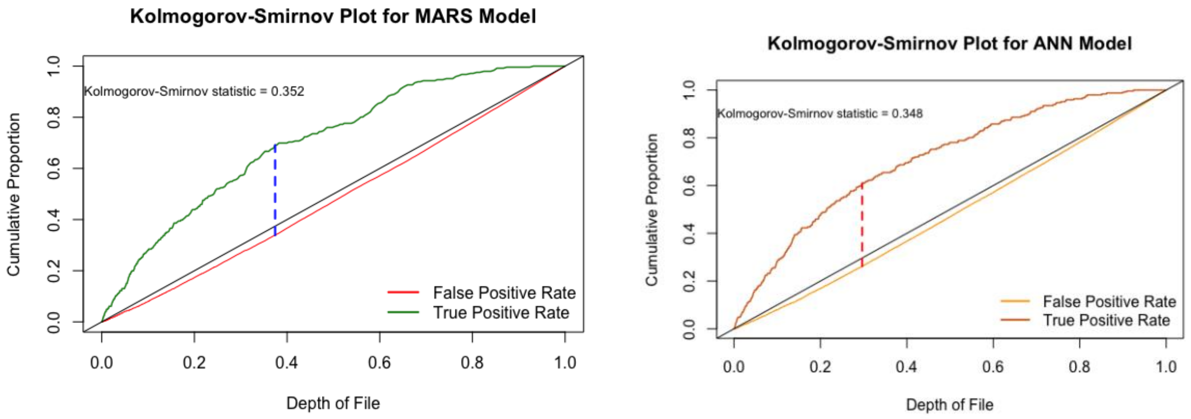

Figure 4: K-S plots

The KS plot reveals that the Logistic Model based on WOE outperforms the rest of the models, as indicated by the larger vertical distance between the two curves. This suggests that the Logistic Regression Model based on WOE is better at accurately classifying the target variable.

Section IV: Findings and Conclusion

By examining the ROC curve and its corresponding area under the curve (AUC), we can evaluate the performance of a classification model. A higher AUC indicates better performance. In this case, both the Mars and ANN models have similar AUC values, suggesting that they have comparable predictive ability.

Similarly, the KS statistic can also be used to evaluate the performance of a classification model. The KS statistic measures the distance between the cumulative distribution functions of the positive and negative classes. A higher KS value indicates better performance. Once again, the Mars and ANN models have comparable KS values, indicating that they perform similarly in terms of separating the positive and negative classes.

Overall, both the Mars and ANN models exhibit similar performance based on their ROC and KS values.

References

Chen, Y., & Luo, Y. (2017). Credit risk modeling using WOE analysis and machine learning. International Journal of Computer Applications, 175(12), 9-14.

Gupta, S., & Singh, S. (2018). Artificial Neural Networks: A Review of Applications in Predictive Modeling. Journal of Artificial Intelligence and Systems, 1(1), 1-10.

Hanley, J. A., & McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology, 143(1), 29-36.

García, S., & Herrera, F. (2012). An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. Journal of Machine Learning Research, 13(1), 1379-1394.