ESCUELA POLITÉCNICA NACIONAL

DEPARTMENT OF INFORMATICS AND COMPUTER SCIENCE

OPTIMIZING THE COLLECTION PROCESS IN CREDIT RISK MANAGEMENT: A COMPARISON OF MACHINE LEARNING TECHNIQUES FOR PREDICTING PAYMENT PROBABILITY AT DIFFERENT STAGES OF ARREARS

WORK PRIOR TO OBTAINING A MASTER'S DEGREE IN COMPUTER SCIENCE

ANDRÉS SEBASTIÁN CARRERA SÁNCHEZ
andres.carrera@epn.edu.ec

DIRECTOR: MARCO BENALCÁZAR, PHD
marco.benalcazar@epn.edu.ec

QUITO, JUNE 2024

CERTIFICATION

I certify that the present work was developed by ANDRÉS SEBASTIÁN CARRERA SÁNCHEZ, under my supervision

Marco Benalcázar, PhD
Project Director

STATEMENT

I, ANDRÉS SEBASTIÁN CARRERA SÁNCHEZ, declare under oath that the work here written is of my authorship; which has not previously been submitted for any degree or professional qualification; and that I have consulted the bibliographic references included in this document.

Through this declaration I hereby grant my intellectual property rights, corresponding to this work, to the Escuela Politécnica Nacional, as established by the Intellectual Property Law, by its regulations and by the institutional regulations in force.

Andrés Sebastián Carrera Sánchez

Contents

RESUMEN

En el ámbito del riesgo crediticio, se han desarrollado modelos de scoring basados en regresión logística para optimizar la evaluación del riesgo de incumplimiento. Sin embargo, estos modelos requieren ingeniería de características compleja y su precisión se ve afectada a medida que avanza la morosidad.

Este estudio propone el uso de técnicas de aprendizaje automático (XGBoost y Redes Neuronales Artificiales) para generar scores en diferentes segmentos de mora (Sin Mora, 1-30 días, 31-90 días y todos los segmentos). Se utiliza la métrica Kolmogorov-Smirnov (KS) para evaluar la eficiencia y el poder predictivo de los modelos.

Para garantizar la precisión y fiabilidad de los modelos, se emplea una metodología de cinco pasos. Comienza con la formulación del problema, seguida de la selección de una muestra de datos y definición de la variable objetivo, luego se realiza un análisis descriptivo de los datos para facilitar la limpieza. Posteriormente, se entrenan y prueban los modelos, y finalmente, se analizan los resultados y se interpretan los modelos obtenidos.

Los resultados muestran que tanto XGBoost como las Redes Neuronales Artificiales superan a la regresión logística en la mayoría de los segmentos de mora. En el segmento Sin Mora, XGBoost (63,36%) y ANN (61,84%) superan a LR (56,42%). En el segmento 1-30 días, XGBoost (51,38%) y ANN (50,35%) también superan a LR (47,32%). En el segmento 31-90 días, ANN (38,77%) supera a LR (36,62%), pero no a XGBoost (34,47%). Finalmente, en el modelo de todos los segmentos, tanto XGBoost (74,05%) como ANN (73,59%) superan a LR (71,01%).

PALABRAS CLAVE: XGboost, Artificial Neural Networks, Logistic Regression, Credit Scoring, Credit Risk Management.

ABSTRACT

In the field of credit risk, scoring models based on logistic regression have been developed to optimize the assessment of default risk. However, these models require complex feature engineering and their accuracy suffers as delinquency progresses.

This study proposes the use of machine learning techniques (XGBoost and Artificial Neural Networks) to generate scores in different delinquency segments (No Arrears, 1-30 Arrears Segment, 31-90 Arrears Segment, and All Segments). The Kolmogorov-Smirnov (KS) metric is used to assess the efficiency and predictive power of the models.

To ensure the accuracy and reliability of the models, a five-step methodology is employed. It starts with the formulation of the problem, followed by the selection of a data sample and definition of the target variable, then a descriptive analysis of the data is performed to facilitate cleaning. Subsequently, the models are trained and tested, and finally, the results are analyzed and the models obtained are interpreted.

The results show that both XGBoost and Artificial Neural Networks outperform logistic regression in most of the arrears segments. In the No Delinquency segment, XGBoost (63.36%) and ANN (61.84%) outperform LR (56.42%). In the 1-30 days segment, XGBoost (51.38%) and ANN (50.35%) also outperform LR (47.32%). In the 31-90 days segment, ANN (38.77%) outperforms LR (36.62%), but not XGBoost (34.47%). Finally, in the all-segments model, both XGBoost (74.05%) and ANN (73.59%) outperform LR (71.01%).

KEYWORDS: XGboost, Artificial Neural Networks, Logistic Regression, Credit Scoring, Credit Risk Management

Chapter 1

INTRODUCTION

1.1 Credit Scoring and Debt Collection

Over the past twenty years, banks and other financial institutions have used advanced methods to analyze data and predict borrower behavior. By using computers and mathematical tools, lenders can make more accurate predictions about loan repayment. This process, called "scoring," has helped lenders increase their market share and reduce the risk of financial loss.

In the field of finance, these models are used to determine the creditworthiness of customers. Based on their payment behavior, banks and financial institutions classify customers as either good or bad. To do this, enough information is collected from customers to score them and decide on credit approval.

Banks and financial institutions often try to encourage the use of their products by offering deals on credit cards, loans, and other financial products. They may offer promotions or cash advances to stimulate increased spending, or they may make efforts to encourage the repayment of old debts. While these offers can be tempting, it's important to be aware of the terms and conditions before signing up, as they can sometimes be quite aggressive.

Many financial institutions focus on evaluating a borrower's creditworthiness only at the time of loan origination. However, this approach neglects the importance of ongoing credit management throughout the duration of the loan. As a result, the concept of integral credit management has not been extensively developed, leaving many borrowers without the necessary support to maintain good credit behavior over time.

The credit cycle described in Fig.1.1 shows the stages that should be implemented in the administration to be able to comprehensively control the risk generated in the placement of a loan.

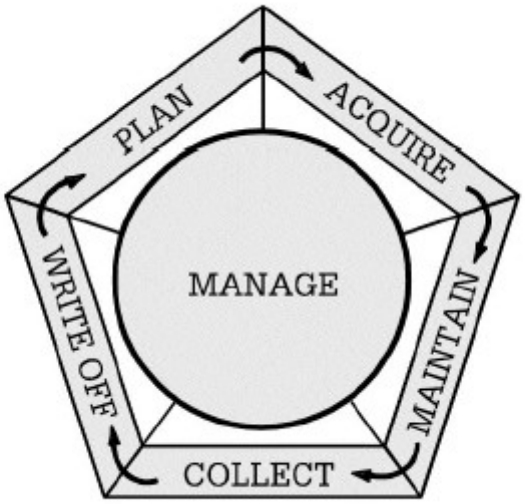

Figure 1.1: The Credit Cycle [1]

As the number of individuals taking out loans increases, it becomes crucial for companies to devise a plan for situations when borrowers are unable to repay their debts. Utilizing data analysis enables companies to determine the most effective method of collecting money from debtors. This not only aids in increasing profits and minimizing losses from non-payment but also ensures a more consistent and equitable debt collection process for all parties involved.

The integration of scoring models in collections management has improved data analysis and decision-making capabilities, especially in the management of large portfolios and the allocation of resources. This innovative approach to credit management is increasingly popular among financial institutions. Techniques such as clustering, logistic regression and artificial neural networks are used to derive profitability from portfolios with non-performing accounts. Due to its easy interpretability, logistic regression is a favourite in this field.

Most collection companies employ strategies based on segmentation with three variables: age of arrears, product, and geographic region. Recovery channels are targeted according to cost and level of impact. Additionally, a prioritization variable related to the amount owed by the customer is incorporated into the management. When analyzing contact channels, it is important to consider their impact on the customer. This includes how the message is delivered, such as through phone calls, visits, text messages, emails, or online chat. Each channel has a different impact and cost for the company (table 1.1), so it is crucial to have a proper strategy to manage the entire portfolio and avoid high costs.

The trained models are implemented by using the scoring variable as a key part of the collection strategies. This involves combining segmentation and contact channels, with the aim of achieving quicker and more cost-effective recovery [2]. The higher the score, the higher the likelihood that the customer will default on their obligations. Therefore, the

Table 1.1: Contact channel by cost, interaction and contact

Contact Channel	Cost	Interaction	Impact
Telephony Manager	Medium	High	High
Telephony IVR	Low	Low	Low
Visit	High	High	High
e-mail	Low	Medium	Low
SMS	Medium	Low	Medium
Chat	Low	Medium	High

strategy must be tailored based on the customer score and the channel of contact to achieve efficiency in both management outcomes and cost. Collection management strategies are typically planned on a weekly and monthly basis. Weekly strategies are adjusted based on the results during the month. The table 1.2 a general management strategy that is based on the results of the score by arrears ranges.

Table 1.2: Collection strategy by score

Contact Channel	0 No Arrears Segment	1 - 30 Segment	31 - 90 Segment
	Week 1	Week 1	Month
SMS	Score >= 186	Score >= 98	Score <= 91
Mail		Score < 98	Score <= 91
IVR	Score >= 186	Score <= 91	
Telephony	Score >= 400	Score >= 109	All scores
Visits		Score >= 186	All scores

This study explores the effectiveness of conventional logistic regression in comparison to two machine learning techniques, Extreme Gradient Boosting (XGBoosting) and artificial neural networks (ANN), in assessing the arrears levels of a large portfolio of retail sector credits. The success of a scoring model is measured through various metrics, including the KS test, GINI statistic, and performance tables. Our study focuses on the KS test, which is considered the most significant metric for evaluating model performance.

In order to guarantee the precision and dependability of the models, a five-step methodology is employed. It commences with the formulation of the problem statement, followed by the selection of a data sample and definition of the target variable, and then proceeds with a descriptive analysis of the data to facilitate cleaning. Subsequently, the models are trained and tested, and finally, the results are analyzed and the obtained models are interpreted.

1.2 Theoretical Framework

This section reviews some efforts to predict the probability of client default to make informed lending decisions. We explore various modeling techniques, including logistic regression, XGBoost, and artificial neural networks, and evaluate their performance using the Kolmogorov-Smirnov (KS) statistic.

1.2.1 Credit Scoring with Logistic Regression

Logistic regression is a frequently utilized statistical method in the credit rating industry because of its capability to model the likelihood of a binary event, such as a credit default [3]. This model is renowned for its interpretability and predictive accuracy. Recent research has shown that logistic regression has achieved remarkably high accuracy in forecasting credit risk, reaching up to 99% in certain instances [4].

1.2.2 Credit Scoring with XGBoost

XGBoost, a decision tree-based machine learning algorithm, is widely acclaimed for its exceptional performance in classification and regression tasks. This model excels at uncovering intricate patterns in data and has demonstrated successful applications in credit risk prediction[5] [6]. However, there have been instances where XGBoost has underestimated credit risk, indicating the necessity for further refinements and validations [7].

1.2.3 Credit Scoring with Artificial Neural Networks

Artificial neural networks are machine learning models inspired by the human brain functioning. These networks are adept at capturing non-linear and complex relationships between variables, making them well-suited for predicting credit defaults [8] [9]. In comparative studies, neural networks have demonstrated performance comparable to logistic regression, achieving an accuracy of 71% in training and 72% in testing [8].

1.2.4 Performance Evaluation

The Kolmogorov-Smirnov (KS) statistic is a metric utilized to evaluate the predictive performance of scoring models. It measures the difference between the cumulative distributions of good and bad payers scores, indicating the degree of distinction between the two sets of scores. Recent studies have incorporated KS alongside other metrics like the area under the ROC curve and the GINI test to evaluate and compare the effectiveness of various models [3] [21].

1.2.5 Comparison of Models

In comparing models for credit risk prediction, logistic regression, neural networks, and XGBoost have all been identified as suitable techniques. Each method has its own strengths and weaknesses. Logistic regression is highly interpretable and exhibits high accuracy in predicting credit extension [3] [4]. XGBoost, on the other hand, delivers high performance and can handle large amounts of data, although it may require additional adjustments to prevent underestimating risk [5] [7]. While neural networks are complex, they can capture non-linear relationships and have demonstrated performance comparable to logistic regression [8] [9].

In sum, selecting the most suitable modeling technique for credit scoring and collections depends on various factors such as interpretability, data volume, and predictive capability. Viable techniques include logistic regression, XGBoost, and artificial neural networks, each with their distinct advantages and limitations. Evaluating performance using metrics like the Kolmogorov-Smirnov statistic is essential for assessing the efficacy of each model in predicting credit risk.

1.2.6 Research goal

Find the best machine learning technique, among Logistic Regression, Extreme Gradient Boosting (XGboost) and Artificial Neural Networks (ANN), for predicting payment probability at different stages of Arrears.

1.2.7 Specific goals

- Evaluate the performance of XGBoost and ANN in capturing the different behaviors of individuals at each stage of arrears in the collections process, compared to traditional logistic regression models.
- Investigate the interpretability of XGBoost and ANN models in credit risk management, and assess their ability to provide insights into the factors that drive default behavior.
- Stablish protocols for using XGBoost and ANN models in real-world credit risk management scenarios.

1.2.8 Hypothesis

XGBoost and ANNs will outperform traditional logistic regression models in predicting payment probability, and will require less feature engineering to achieve superior results.

Chapter 2

METHODOLOGY

2.1 Problem Statement

2.1.1 Scoring Model Definition

In the field of mass credit management, scoring models have proven to be the most valuable tool for the past two decades. By analyzing historical data, these models provide predictions of future behavior, which help control portfolios with greater accuracy and less uncertainty. A scoring model takes into consideration numerous variables at the same time, which helps to establish a pattern and group members together based on their likelihood of experiencing an event. These models work best when dealing with large volumes of data that have relatively homogeneous values. It is important to note that scoring models are designed to identify patterns and groupings, rather than to provide precise predictions for individual cases.

Figure 2.1: Credit Scoring Scheme

At present, statistical techniques are used to develop scoring models that classify customers based on their behavior. This is a supervised learning problem in the context of machine learning. The objective is usually to calculate the likelihood of a customer paying off their debt, determine their level of risk in case of loan approval, or evaluate the profitability of offering multiple products to one person.

During different stages of the credit cycle, some institutions have developed models to detect potential over-indebtedness among clients, or to identify accounts that have defaulted. Scoring models are used not only in the credit business, but also in insurance companies to evaluate candidates for longevity, good or bad health, or automobile accident risk. They are also used in services such as tax payment, telephony, cable TV services, and even in matchmaking, although they are not very popular.

The main advantage of using scoring models is that they provide an objective assessment of risk, eliminating personal biases in the decision-making process. This allows decisions to be made consistently and in a standardized manner. Although the current methodology has some limitations that have been identified over time, the benefits of implementing the models far outweigh them. However, it is important to be aware of the following limitations:

- Firstly, the development and implementation process takes some time due to data collection and variable engineering.
- Secondly, the model can only identify the probability of a good or bad operation or client, rather than determining whether it is actually good or bad, according to the definition in the construction.
- Finally, all models lose predictability over time due to various factors, such as changes in the economy, the population, or the target market being evaluated.

Therefore, it is essential to constantly monitor and evaluate the models to determine whether their levels of discrimination and ordering remain optimal. Despite these limitations, implementing the models generates efficiencies in the processes. By correctly defining the good and bad characteristics, the technique separates the probability distributions efficiently, allowing better decisions to be made.

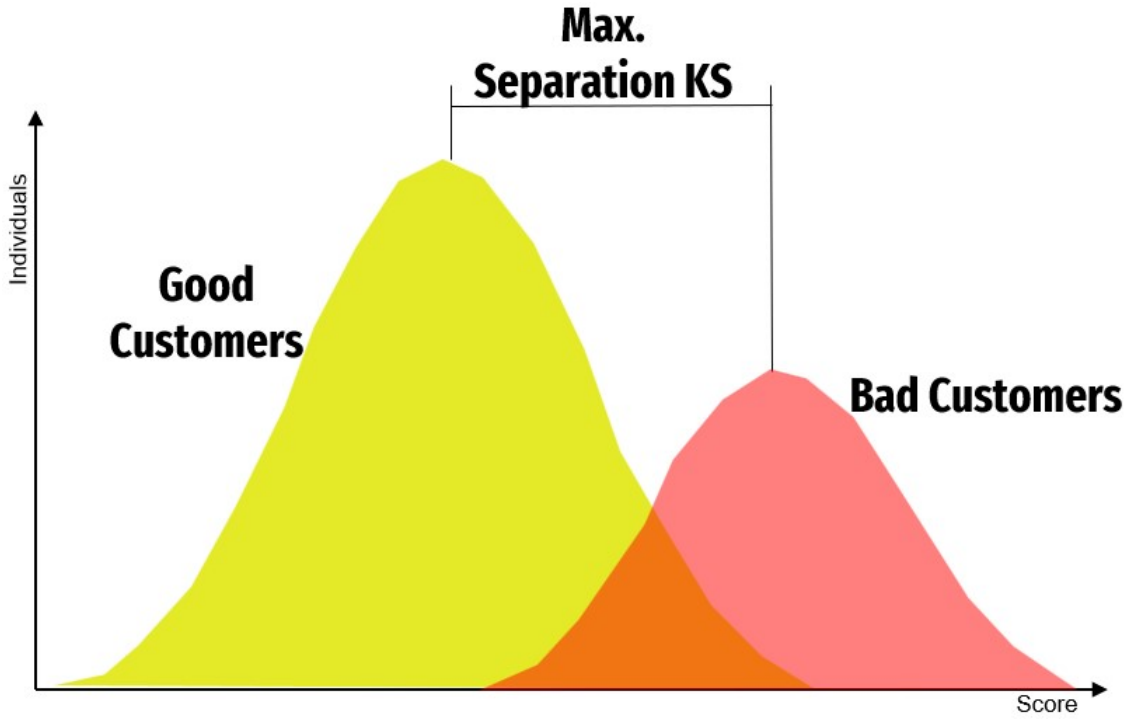

Figure 2.2: Representation of the separation or divergence of two probability distributions.

2.2 Scoring Models in the Collection Stage of the Credit Cycle

Efficient collection management is a critical aspect of managing large credit portfolios. It not only affects customer interactions but also impacts collection operations. Without access to agile and effective tools, negative customer reactions towards their payment obligations are likely to occur.

With the rise of banking services for individuals, creditors are looking to integrate credit and automated collection processes for risk reduction. This involves debiting bank accounts or integrating payment points that are increasingly accessible to customers, regardless of whether they belong to the creditor's network of branches or warehouses. Despite the proliferation of these services, there is still a high level of indebtedness in the market. As a result, creditors need to focus on collecting faster and more efficiently to stay ahead of other creditors and ensure prompt payment. This requires the development of effective strategies to prioritize debt collection.

In the world of credit, it is widely accepted that a loan is deemed to be in default if the payment, as stated in the installment or account statement, is not made on time. Moreover, it is commonly understood that as the arrears age increases, the chances of recovering the funds decreases. Therefore, it is imperative to develop effective strategies to avoid this sit-

uation from occurring. Typically, collection management employs portfolio segmentation and customer contact channels to determine appropriate actions based on the number of days in arrears and product type. This approach may involve various methods such as telephone calls, field visits, text messages, emails, or letters to encourage customers to fulfill their obligations and ensure a positive outcome.

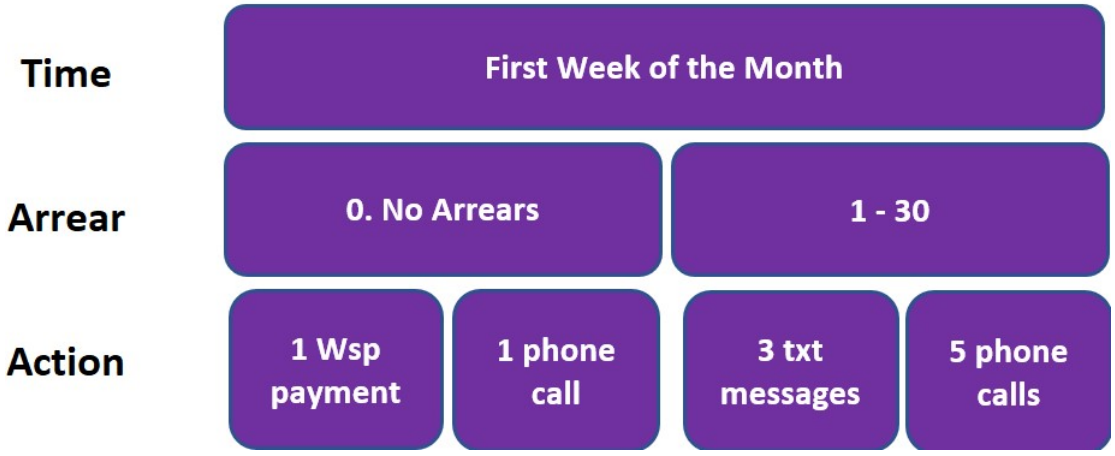

Figure 2.3: Collection Strategy Example

When implementing scoring models for collections, it is crucial to consider the number of days that have passed since the loan was due, to determine whether it is still recoverable. Hence, it is recommended to segment the portfolio into 30 or 15-day arrears ranges, based on the loan disbursement terms. Therefore, the following arrears segments are possible:

- 0 - No Arrears segment
- 1 - 30 segment
- 31 - 60 segment
- 61 - 90 segment
- 91 - 120 segment
- More than 120 segment

Where More than 120 is considered as a loss segment.

When arrears increase, it is crucial to differentiate between each segment and design a scoring model to distinguish reliable payers from those who default on payments. It is essential to keep in mind that as arrears increase, the pool of individuals decreases, which may impact the logistic regression's ability to differentiate. However, thankfully, we can measure the scoring model's ability to discriminate using the Kolmogorov-Smirnoff or KS statistic.

2.2.1 Kolmogorov-Smirnov Tests

The K-S test, or Kolmogorov-Smirnov test, is a non-parametric method utilized to assess the similarity of two distinct continuous distributions. It evaluates the hypothesis of whether or not they are identical. The KS statistic is computed by employing the cumulative empirical distribution function [10].

$$\widehat{F}(x) = \frac{1}{n}\sum_{i=1}^{n}\begin{cases} 1, & \text{if } x_i \leq x \\ 0, & \text{otherwise} \end{cases} \tag{2.1}$$

Consider two samples x_s and y_s of size n_1 and n_2 respectively, with cumulative distribution functions F_1 and F_2 of a continuous random variable X. The KS test is used to test hypotheses:

$$\begin{cases} H_0: & F_1(x) = F_2(x), \forall x \\ H_1: & F_1(x) \neq F_2(x) \end{cases} \tag{2.2}$$

Based on the use of the empirical cumulative distribution function (2.1), the KS statistic is used to test the null hypothesis H_0. Its value is obtained using the following expression:

$$KS = \max_x |\widehat{F}_1(x) - \widehat{F}_2(x)| \tag{2.3}$$

The notation $\widehat{F}_1$ represents the empirical accumulation function of x_s and $\widehat{F}_2$ represents the empirical distribution function of y_s. If the KS statistic is greater than the critical value KS_α for a given significance level α, we reject the null hypothesis H_0. In [11], you can find a table of critical values for different sample sizes.

Then, the KS statistic is a measure of divergence between the distributions of two variables. It is the maximum distance between F_1 and F_2, and its value ranges between 0 and 1. Values close to 0 indicate that the distributions of x_s and y_s are identical, while values close to 1 indicate that the distributions of x_s and y_s differ. Therefore, the KS statistic is useful for distinguishing the differences between two distributions.

In our current project, we aim to determine the classification technique that achieves the highest KS value. We will compare the results of logistic regression, Extreme Gradient Boosting, and Artificial Neural Networks. The predicted score values for individuals who are categorized as good will be represented by x_s, while the predicted score values for individuals who are categorized as bad will be represented by y_s.

2.3 Data Selection and Treatment

When a new loan is disbursed, the information available only contains socio-demographic information, the conditions of the loan, and sometimes information on payment behavior in the financial system. Therefore, it is necessary that the portfolio has a sufficient number of disbursements and that enough time has passed so that historical information can be obtained to construct the predictor variables and the dependent variables that describe the events to be predicted. To comply with the necessary premises, information from a company that has been consolidated for more than 23 years as one of the most important retail and financial services multinationals in Latin America and the Caribbean has been considered.

2.3.1 Data Sample Selection

When building a scoring model, it is important to gather as much information as possible based on the stage of the credit cycle . In the collection stage, it is particularly essential to have data on payment behavior within the institution that provided the loan, information about the social and demographic profile of the customer, and insights into the results of collection management. Sometimes, it can also be useful to include information about payment behavior in other financial institutions.

For our study, we have focused on loans provided directly to consumers in the retail sector, which includes credit for items such as televisions, computers, technology, white goods, and other consumer goods. To explain how the information needed to develop the score models, let's consider figure 2.4. The time period before the observation point is called the "behavioral window", which cannot be longer than 36 months as per the provision of the superintendence of banks and insurance companies in Ecuador.

Typically, when creating scoring models for credit decisions during the acquisition and maintenance stages (see fig. 1.1) of the credit cycle, a history of 36 months is used. However, during the collection stage, using such a long history can be counterproductive. This is because the collection stage is much more dynamic and unpredictable, and one can make mistakes by considering very old payment behaviors that may not reflect the current situation. As a result, it may become challenging to predict their next payment, then we use 12 months of history.

During this period, variables related to the individual's credit history are generated, such as their payment and indebtedness habits, maximum and average arrears, open transactions, telephone transactions, effective telephone contacts, card quotas, consumption amounts, etc. Socio-demographic variables like age, marital status, province, region, etc. are generated at the point of observation.

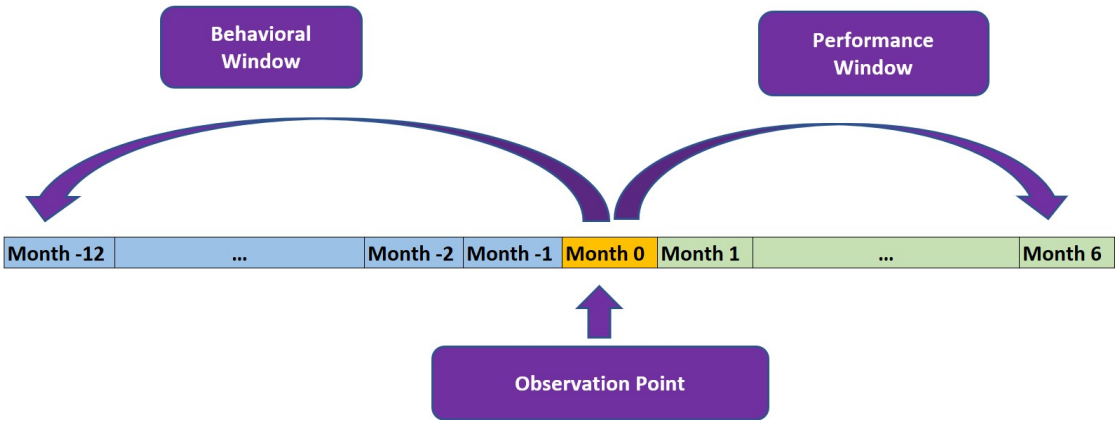

Figure 2.4: Historic Data Selection

After the observation point, we evaluate an individual's payment behavior during a period called the "performance window". This window provides crucial information that helps us define good and bad individuals (dependent variable Y). Since payments are made monthly, we use a one-month window to determine whether a payment has been made or not. After that, we evaluate the individual's payment behavior over a period of 6 months. The selected observation points and the number of customers are shown in the Table 2.1.

Table 2.1: Months of Observation Points (Month 0 - PO) Selection and Number of Customers by Month

Month 0 - PO	Customers
jul-22	26,816
ago-22	22,796
sep-22	19,267
feb-23	16,568
mar-23	17,266
may-23	22,746
Total	**125,459**

2.3.2 Dependent Variable Setting

The dependent variable Y is binary, with a value of 1 assigned to individuals marked as "Good Customer" and 0 assigned to those identified as "Bad Customer". We will be using two different definitions to evaluate performance. The first is based on payment event within a one-month performance window, while the second is based on monthly payment behavior within a 6-month performance window.

DEFINITION 2.3.1. ***Payment Event***

$$Y_1 = \begin{cases} 0: & \textit{if the customer has paid a full instalment is Good} \\ 1: & \textit{otherwise is Bad} \end{cases} \tag{2.4}$$

DEFINITION 2.3.2. ***Monthly Payment Behavior***

$$Y_2 = \begin{cases} 0: & \textit{customer who made all payments before being 30 days in arrears is Good} \\ 1: & \textit{otherwise is Bad} \end{cases} \tag{2.5}$$

With Y_1, the aim is to have as few clients as possible switch to higher arrears ranks. This definition will be used in each arrears range to discriminate between good and bad clients. On the other hand, with Y_2, the aim is to control the deterioration of the portfolio in the medium term, to avoid excessive losses.

2.3.3 Data Exploratory Analysis and Data Cleaning

Data Exploratory Analysis

For the construction of score models, it is of great importance to have large amounts of structured and good-quality data. This means that they are available in tabular format and that the data are consistent with what is known from experience from normal mass credit administration.

Four main groups of variables are generated within the credit cycle:

- **Sociodemographic Variables:** Include personal and geographic information about the client, such as age, marital status, level of education, and place of residence. It is essential to update this information periodically.
- **Operational Variables:** It provide information on the loan's disbursement conditions, such as the loan amount, term, installment value, nominal rate, etc.
- **Behavioural Variables:** Describe the client's payment behavior, including Average Quarterly Arrears, Maximum Semesterly Arrears, and other relevant factors.
- **Collection Management Variables:** Collection Management Variables contain information on the collection management process, such as the number of telephone calls made, the rate of effective contacts, the number of text messages sent, etc.

Exploratory analysis is a technique that helps in understanding the structure and quality of data. This step involves the use of position measures for quantitative variables, and descriptive statistics with proportions for categorical variables. In addition, it is essential

to describe the number of individuals who are more than 90 days delinquent at the observation point, which is also referred to as obvious bad debtors. Furthermore, the analysis should discuss those individuals who were granted a recent loan at the observation point but do not have sufficient historical information.

Data Cleaning

Data cleaning involves making decisions based on the information obtained from exploratory data analysis. One common issue that data analysts encounter are missing data, outliers and variables with too many categories.

- **Missing Data:** To handle missing data properly, it is crucial to understand the nature of each variable. If a record has a NULL mark, is blank, or has a value that does not correspond to the nature of the variable, it may be due to an error during the extraction of the information, an error in the database, or simply no data exists for that particular case.

  If the errors can be corrected at the source during the extraction process, the data extraction should be reprocessed to reduce the amount of missing data. If the number of missing data in a variable does not exceed 50%, it can be replaced by 0 for quantitative variables or assigned a category for qualitative variables. However, if the number of missing data exceeds 50% of the variable's values, the variable should be deleted, as it may not provide enough information to be useful in analysis.

- **Outliers:** In statistics, outliers are data points that are significantly different from other data points in a sample. They can have a significant impact on the results of regression models. Typically, outliers are removed from the data sample. However, sometimes it's important to keep outliers to capture all possible behavioral patterns of individuals. In such cases, we can replace the lower and upper outliers by the maximum and minimum non-outliers, respectively.

- **Variables with too many categories:** When it comes to finding patterns for classification, having too many or too few categories in a variable is not ideal. Therefore, it's essential to set a maximum number of categories. We usually group those categories with smaller proportions into a new category, keeping a maximum of 10 categories.

Details of the exploratory data analysis and data cleaning can be found in Appendix 5.1. After data cleaning, the remaining dataset consists of 125,459 retail sector consumer credit records and 86 predictor variables.

2.4 Train and Test Models

A scoring model is created by identifying patterns within the predictor variables, which can be used to classify individuals into good and bad categories based on the event to be predicted. In machine learning, this model is developed through supervised learning, where the model is trained with data that is different from the data used in the training phase.

It is crucial to have a diverse dataset during the training phase to ensure that the model is trained on a wide range of information. In order to achieve this, the data needs to be randomly split into three datasets. The first dataset, comprising 60% of the data, is used for training. The second dataset, containing 25% of the data, is used for testing. Finally, the remaining 15% of the data is used for validation.

Table 2.2 the distribution of the training, testing, and validation samples. Meanwhile, Table 2.3 indicates the distribution of customers classified as good and bad, for Y_1 and Y_2.

Table 2.2: Distribution of the training, testing, and validation samples

Train	Test	Validation
87,821	37,638	18,819
61%	26%	13%

Table 2.3: Distribution of customers classified as Good (G) and Bad (B)

Dep.Var.	Arrears	Train		Test		Validation	
		G	B	G	B	G	B
Y1	0 - no arrears	75%	25%	75%	25%	75%	25%
	1 - 30	68%	32%	68%	32%	67%	33%
	31 - 90	32%	68%	32%	68%	33%	67%
Y2	**All**	**63%**	**37%**	**63%**	**37%**	**63%**	**37%**

2.4.1 Logistic Regression Training

Logistic regression is a widely used technique for predicting a categorical variable using a set of explanatory variables. It is a parametric method that is formulated as follows.

Consider N quantitative variables $X_1, ..., X_N$. For each combination of these variables, the response variable Y follows a Bernoulli distribution [12].

$$Y|(X_1 = x_1, ..., X_N = x_N) \rightarrowtail B(1, p(x_1, ..., x_N))$$

We are interested in modelling the conditional expectation.

$$E[Y|(X_1 = x_1, ..., X_N = x_N)] = P[Y = 1|X_1 = x_1, ..., X_N = x_N] = p(x_1, ..., x_N)$$

The multiple logistic regression model for Y in terms of the values of the variables X, can be modelled as:

$$p(x_1, ..., x_N) = \frac{\exp(\alpha + \sum_{n=0}^{N} \beta_n x_n)}{1 + \exp(\alpha + \sum_{n=0}^{N} \beta_n x_n)} \tag{2.6}$$

with $\alpha = \beta_0$ and $x_0 = 1$

In matrix terms it would be

$$p(x) = \frac{\exp(\beta^t x)}{1 + \exp(\beta^t x)} \tag{2.7}$$

with $x = (1, x_1, ..., x_N)$ and $\beta = (\beta_0, ..., \beta_N)$

Finally, a linear model for the logit transformation is obtained.

$$\ln\left[\frac{p(x)}{1 - p(x)}\right] = \sum_{n=0}^{N} \beta_n x_n \tag{2.8}$$

Unbalanced Problem

In some cases where we use logit, probit or linear probability models, the number of observations in one group is much smaller than in the other. For instance, in lending, the number of bad clients is expected to be much smaller than the number of good clients because if both were equal, the financial institution would face bankruptcy. Therefore, to reach accurate predictions, we need either a large dataset or a balanced sample containing equal proportions of both groups. In this case, we would consider all bad customers and sample the good customers to achieve a 50/50 ratio.

The question arises as to how we can analyze data in such cases. We suggest using a weighted logit (or probit or linear probability) model, similar to weighted least squares. If the logit model is used for estimating the coefficients of the explanatory variables, the different sample sizes for the two groups do not affect the coefficients [13].

Let m_1 and m_2 be the sample proportions of the two groups, with $m_2 > m_1$. Since m_1 is the probability that an observation belonging to the first group is selected, and m_2 is the probability that an observation belonging to the second group is selected, when the samples are disproportionate the logistic function is shifted as follows:

$$\ln\left[\frac{p(x)}{1 - p(x)} * \frac{m_2}{m_1}\right] = \sum_{n=0}^{N} \beta_n x_n \tag{2.9}$$

When $m_1 = m_2$ the logistic function cuts on the x-axis, at the value 0.5, as seen in Figure 2.5. Now, if $m_1 = 0.2$ and $m_2 = 0.8$ the curve shifts and ends up cutting on the x-axis at 0.8, as seen in Figure 2.6.

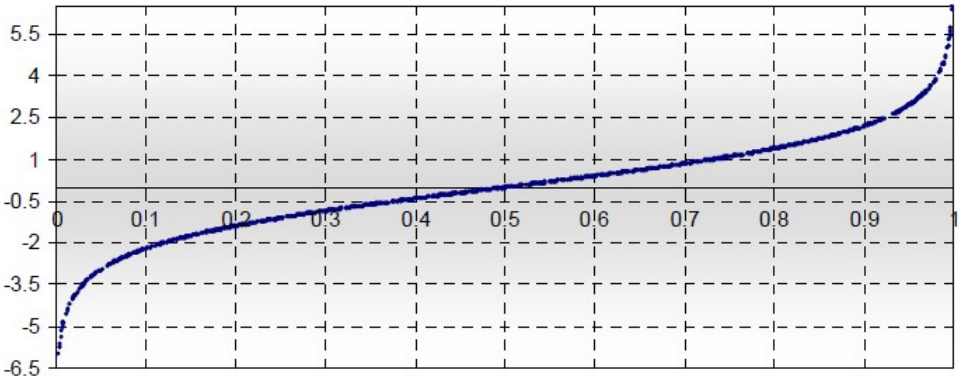

Figure 2.5: Logit Function

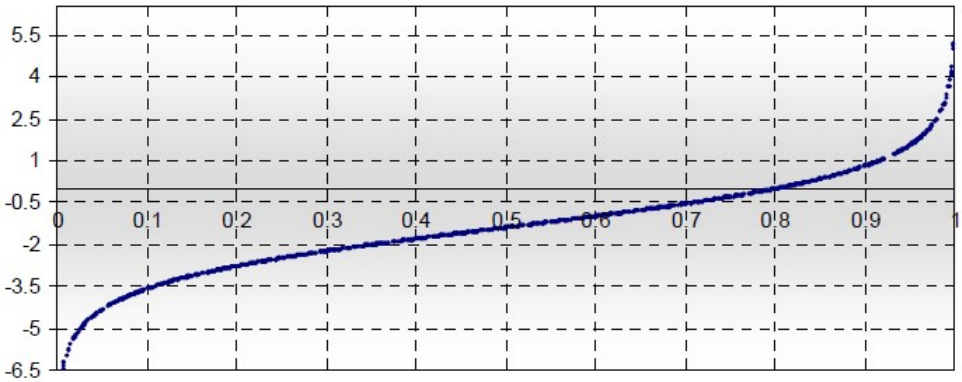

Figure 2.6: Shifted Logit Function

Therefore, the disproportionality of the samples only affects the constant term of the model and one has that

$$p(x) = \frac{\exp(\gamma + \beta^t x)}{1 + \exp(\gamma + \beta^t x)} \tag{2.10}$$

where $\gamma = -\ln(m')$, $m' = \frac{m_2}{m_1}$ [14].

Feature Engineering

In this particular section, we are focused on constructing new variables by combining the predictor variables. These new variables are dummy variables, meaning they are binary variables that take a value of either 1 or 0. The objective of the construction of the dummy variables is to generate characteristics with greater predictive power in the greater predictive power in classification of individuals into Good and Bad. To create these variables, we will be using decision trees as a tool. Decision trees allow us to identify the most important features and construct new variables that can be used to improve the accuracy of our predictive models.

To create the new dummy variables, we follow a specific process. Initially, we divide the population into two groups that are similar in characteristics. Then, we further divide each of these groups into two more similar subsets. We continue this recursive process until we reach a minimum number of individuals in each subset (stopping criterion). Finally, we determine whether each subset is Good or Bad based on the distribution of Good and Bad individuals compared to the distribution in the initial set (assignment criterion).

Below are the criteria used for executing the algorithm:

1. Partition criterion. To establish the cut-off values of the explanatory variable that will define the segments, the CHAID partitioning method is used, which is based on the χ^2 statistic to assess the dependence between the dependent variable and the categorical variable constructed based on the partitioning criteria generated.

2. Stopping criterion. A subset is partitioned only if its percentage of individuals is greater than a previously established percentage, for example, 3

3. Allocation criterion. After verifying the stop criterion, the final subsets obtained are known as terminals, the terminal subsets in our case will be of two types:

   - Good. If the percentage of Good individuals in the terminal subset is higher than the percentage of Good individuals in the initial set.
   - Bad. If the percentage of Bad individuals in the terminal subset is greater than the percentage of Bad individuals in the initial set.

In our research, we use decision trees to create distinct groups of individuals with similar characteristics. To illustrate this process, the Figure 2.7 provides an example of a decision tree that demonstrates the CHAID partitioning method and how it can be used to generate binary variables called dummies.

Consider node 29. This subset contains 9% of individuals, whose average installment payment (cp_pl) is greater than 0.93 and their Saldo_cuota_credito is less than or equal to $54,060$. Since the percentage of baddies in this subset is 99.7%, which is greater than the percentage of baddies in the initial set, then this subset is categorized as bad. We then create the dummy variable defined as

$$V_1 = \begin{cases} 1: & \text{cp\_pl } > 0.93 \wedge \text{Saldo\_cuota\_credito} <= 54,060 \\ 0: & \text{otherwise is Bad} \end{cases} \tag{2.11}$$

All dummy variables created for each arrear range model, are presented in Appendix 5.2.

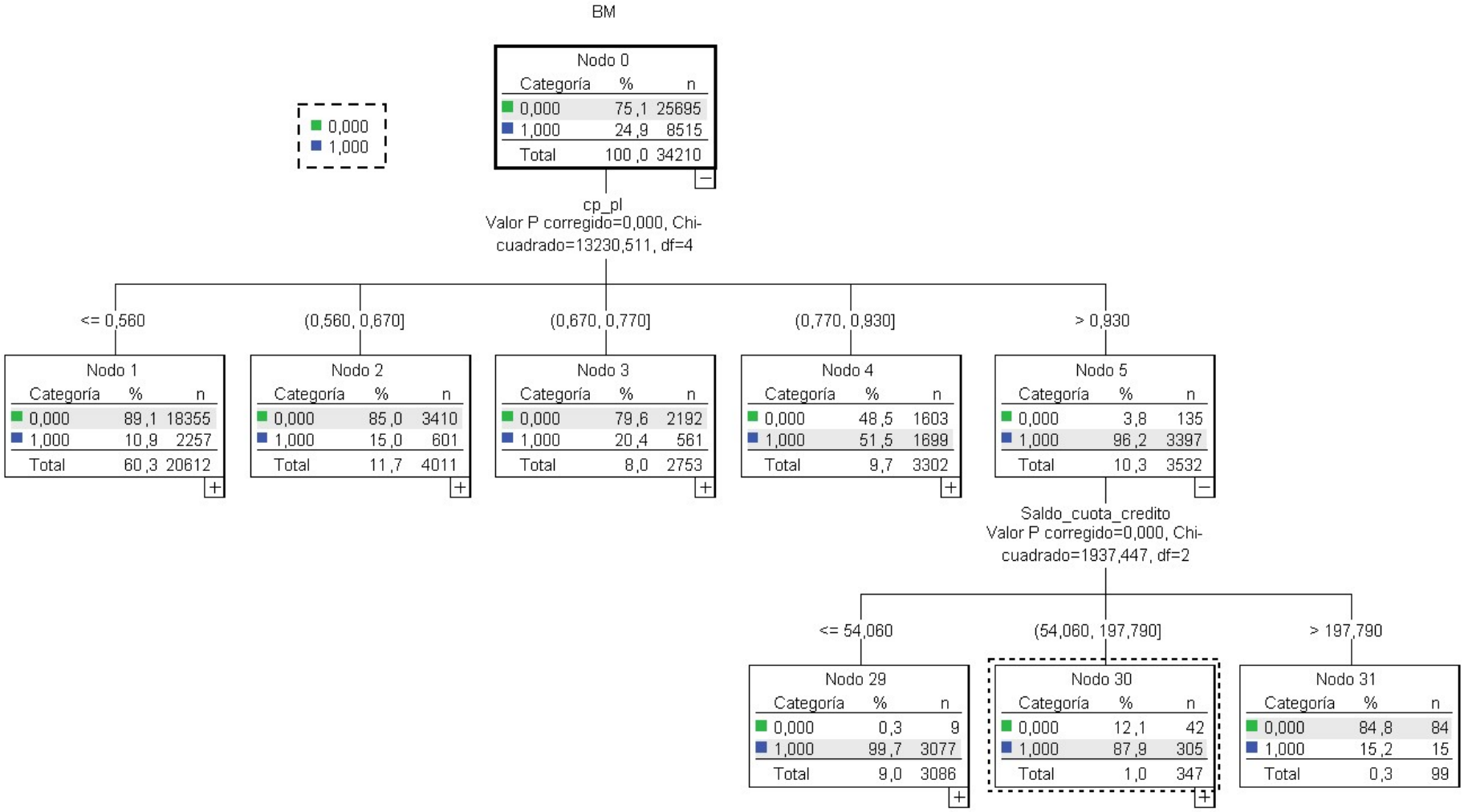

Figure 2.7: Decision Tree for Dummies Variables

Logistic Regression Training Results

The training methodology for selecting the best regression model is based on comparing the variance of the regression model between different models. Typically, the stepwise procedure is used, which involves comparing models with different variables using conditional likelihood ratio tests. In forward selection, the process starts with the simplest model and at each step, the most significant variable is added based on the conditional likelihood ratio test. The process stops when the model with the largest possible number of variables and interactions is reached, or when no model improves on the current model.

Another way to proceed is to start from the model with the most variables and interactions and eliminate variables, again using the criterion of the likelihood ratio tests.

In this paper, we use a combination of both procedures so that at each step, we test whether a new variable enters or a variable that is already in the model leaves the model.

For this purpose, a significance level 1 is set for the contrast with models that add a variable and a significance level 2 for the elimination of variables, with $2 > 1$. In each step, several contrasts are performed, both for the inclusion of variables and for the elimination, and the process continues until the contrasts cease to be significant, i.e. no more variables are included, nor are any of those that entered eliminated.[12]

After obtaining the best model through the process described above, we analyze the consistency of the signs. This means that if a variable was classified as "bad" with the decision trees, its sign in the regression model must be positive. Similarly, if it was classified as "good", its sign must be negative. Therefore, if a variable is not consistent, it is eliminated from the regression model.

The results of the best regression model obtained for each segment of arrears and their *KS* values obtained with the test samples are shown below.

The description of the variables is detailed in 5.2.

0 - no arrears segment

$KS = 56.42\%$

Table 2.4: Estimated coefficients for no arrears model

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.2987277	0.04463	-18.054	< 0.00***
V1	-1.74701	0.05871	-29.758	< 0.00 ***
V2	-1.74726	0.09534	-18.326	< 0.00 ***
V3	-0.40177	0.04948	-8.12	0.00 ***
V4	1.61231	0.06411	25.148	< 0.00 ***
V5	0.87094	0.04869	17.887	< 0.00 ***
V7	-0.85414	0.06449	-13.245	< 0.00 ***
V9	-0.27386	0.03938	-6.954	0.00 ***
V18	0.10523	0.04411	2.386	0.01705 *
V19	-0.46984	0.05082	-9.246	< 0.00 ***
V20	0.08418	0.04399	1.914	0.05567 .
V23	-0.15439	0.03595	-4.294	0.00 ***
V25	-0.11464	0.04301	-2.665	0.00769 **
V26	-0.67996	0.08161	-8.332	< 0.00 ***

1 - 30 segment

$KS = 47.32\%$

Table 2.5: Estimated coefficients for 1 - 30 segment model

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.3411896	0.044187	-8,928	< 0.00 ***
V1	1.56644	0.048238	32,473	< 0.00 ***
V2	-0.001064	0.00017	-6,258	0.00 ***
V3	-0.922552	0.063436	-14,543	< 0.00 ***
V4	-0.813158	0.058572	-13,883	< 0.00 ***
V6	-0.23837	0.065325	-3,649	0.00 ***
V7	0.704773	0.055055	12,801	< 0.00***
V8	-0.177843	0.038652	-4,601	0.00 ***
V9	-0.547729	0.046243	-11,845	< 0.00 ***
V10	-0.10689	0.039408	-2,712	0.007**
V12	0.641873	0.053738	11,945	< 0.00 ***
V13	-0.012972	0.00305	-4,253	0.00 ***
V22	-0.1124	0.04454	-2,524	0.011617 *
V26	-0.310141	0.033923	-9,143	< 0.00 ***
V30	0.011898	0.001898	6,270	0.00 ***
V38	-0.215627	0.03559	-6,059	0.00 ***

31 - 90 segment

$KS = 36.62\%$

Table 2.6: Estimated coefficients for 31 - 90 segment model

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	1.440795	0.07451	9.326	< 0.00 ***
V1	-0.52883	0.06594	-8.02	0.00 ***
V2	-0.58201	0.06415	-9.073	< 0.00 ***
V3	0.19268	0.05963	3.231	0.001234 **
V4	-0.82238	0.05487	-14.99	< 0.00 ***
V5	-0.90174	0.0629	-14.34	< 0.00 ***
V8	-0.30622	0.0617	-4.963	0.00 ***
V9	0.51679	0.06319	8.178	0.00***
V14	0.5346	0.08328	6.419	0.00 ***
V15	0.19934	0.07222	2.76	0.005780 **
V18	-0.26505	0.04547	-5.829	0.00 ***
V22	-0.13209	0.0472	-2.798	0.005135 **
V27	0.7743	0.11301	6.852	0.00 ***
V39	0.12466	0.05834	2.137	0.032633 *
V41	0.17361	0.05468	3.175	0.001497 **
V53	0.23547	0.07261	3.243	0.001182 **
V57	0.16562	0.05602	2.957	0.003111 **
V58	0.13992	0.06208	2.254	0.024212 *
V60	0.1434	0.05619	2.552	0.010712 *
V61	0.48752	0.06572	7.418	0.00 ***
V66	0.16522	0.04805	3.439	0.000585 ***
V69	-0.18121	0.06162	-2.941	0.003276 **
V74	-0.15128	0.04789	-3.159	0.001584 **
V75	-0.17226	0.05711	-3.016	0.002560 **
V78	0.19746	0.05064	3.899	0.00 ***
V79	0.18916	0.05226	3.62	0.000295 ***
V81	0.14625	0.03679	3.975	0.00 ***
V83	0.15036	0.0607	2.477	0.013245 *
V85	0.10917	0.04848	2.252	0.024343 *
V86	0.11468	0.03987	2.877	0.004021 **

All arrears

$KS = 71.01\%$

Table 2.7: Estimated coefficients for All arrears segment model

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-1.44516	0.05196	-27.812	< 0.00 ***
V2	2.22953	0.03473	64.198	< 0.00 ***
V3	-0.81571	0.06752	-12.082	< 0.00 ***
V5	-0.521	0.05834	-8.931	< 0.00 ***
V9	0.33046	0.04173	7.919	0.00 ***
V20	-0.47253	0.04065	-11.624	< 0.00 ***
V21	-0.24696	0.05266	-4.69	0.00 ***
V22	0.71427	0.04143	17.239	< 0.00 ***
V29	-0.27818	0.05805	-4.792	0.00 ***
V33	0.11321	0.04254	2.661	0.007783 **
V35	0.22049	0.03923	5.621	0.00 ***
V37	-0.52771	0.03251	-16.232	< 0.00 ***
V39	-0.2157	0.04852	-4.446	0.00 ***
V44	-0.59042	0.02875	-20.534	< 0.00 ***
V47	-0.44351	0.03725	-11.907	< 0.00 ***
V49	-0.16629	0.03376	-4.926	0.00 ***
V50	-0.15746	0.03026	-5.203	0.00 ***
V51	0.21689	0.03484	6.225	0.00 ***
V53	0.21209	0.03593	5.903	0.00 ***
V54	-0.14446	0.03772	-3.83	0.000128 ***
V55	0.10587	0.0446	2.374	0.017610 *
V57	-0.28297	0.04198	-6.741	0.00 ***
V60	0.22757	0.02673	8.515	< 0.00 ***

2.4.2 Extreme Gradient Boosting Training

XGBoost (Extreme Gradient Boosting) is a machine learning algorithm that was introduced by Chen and Guestrin in 2016. It uses the concept of tree gradient boosting to improve its performance and speed. XGBoost was designed to reduce overfitting by introducing regularization parameters. Gradient boosting trees use regression trees in a sequential learning process as weak learners. These regression trees are similar to decision trees, but they assign a continuous score to each leaf that is then summarized to provide the final prediction.

For each iteration i in which a tree t grows, scores w are computed that predict a given outcome y. The learning process aims to minimize the overall score, which is composed of the loss function at i-1 and the new tree structure of t. This allows the algorithm to sequentially grow the trees and learn from previous iterations. The gradient descent is then used

to calculate the optimal values for each leaf and the total tree score t. The score is also called the impurity of a tree's predictions.

The XGBoost algorithm employs a loss function that includes a penalty term to reduce the complexity of the regression tree functions. This penalty term is adjustable and can take values equal to or greater than 0. Setting this term to 0 results in no difference between the gradient-boosted and XGBoost trees' prediction results. Moreover, Chen and Guestrin [15] introduce shrinkage (a learning rate) and column subsampling (random forest) to this gradient tree augmentation algorithm, which allows for further reduction of overfitting.

XGBoost is an algorithm that has two significant advantages over AdaBoost and other algorithms. First, it executes faster, and second, it has a regularisation parameter that helps to reduce variance. Additionally, XGBoost uses learning rate and subsample features (random forests), which allows it to generalize better. However, tuning and understanding XGBoost can be more complex than AdaBoost or only random forests. Numerous hyperparameters can be adjusted to increase performance.

Several hyperparameters are relevant when it comes to training a model. Some of them include the learning rate, column subsampling, and regularisation rate. Additionally, subsampling (which involves bootstrapping the training sample), the maximum depth of the trees, the minimum weights on the children's scores to split, and the number of estimators (trees) are also commonly used to address the bias-variance-compensation. While higher values for the number of estimators, regularisation, and weights on secondary grades are associated with reduced overfitting, learning rate, maximum depth, subsampling, and column subsampling should have lower values to achieve reduced overfitting. However, setting extreme values for any of these hyperparameters can lead to model misfits.

Hyperparameter Tuning

Hyperparameters are settings or configurations of the methods (models), which are freely selectable within a certain range and influence model performance (quality).

Grid search in XGBoost is an optimization technique that seeks to find the set of hyperparameters that yields the most accurate predictive model. It operates by defining a grid of hyperparameter values and evaluating the model's performance for each combination of these values. This process is facilitated by the use of cross-validation, typically k-fold cross-validation, to assess the performance of the model on different subsets of the training data, thereby ensuring that the model's performance is robust and not overly dependent on the particularities of one set of training data.

The hyperparameters commonly tuned in XGBoost through grid search include *max_depth*, *min_child_weight*, *gamma*, *subsample*, *colsample_bytree*, and *learning_rate* (*eta*). The grid search process evaluates the model for each combination of hyperparameters in the grid, which can be computationally intensive but is necessary for identifying the optimal parameters that minimize overfitting and maximize predictive performance.

Grid search is a brute-force approach to model tuning that can be highly effective but may also require significant computational resources, especially with large datasets and complex models. It is a critical step in the machine learning pipeline for XGBoost, ensuring that the final model is as accurate and generalizable as possible.

XGBoost Hyperparameter nrounds

The parameter *nrounds* specifies the number of boosting steps. Since a tree is created in each individual boosting step, *nrounds* also controls the number of trees that are integrated into the ensemble as a whole. Its practical meaning can be described as follows: larger values of *nrounds* mean a more complex and possibly more precise model, but also cause a longer running time.

$nrounds \in [1,\infty[$. Only integer values are valid.

XGBoost Hyperparameter eta

The parameter *eta* is a learning rate and is also called "shrinkage" parameter. It controls the lowering of the weights in each boosting step. It has the following practical meaning: lowering the weights helps to reduce the influence of individual. $eta \in [0, 1]$

XGBoost Hyperparameter max_depth

The *max_depth* hyperparameter in XGBoost refers to the maximum depth of a tree. It is used to control how deep the decision trees within the model can grow during any boosting round. A deeper tree can model more complex patterns in the data, but it also increases the risk of overfitting. The default value is typically set to 6, but it can be adjusted depending on the complexity of the task and the amount of data available.

$max\_depth \in [0, n]$. Only integer values are valid.

XGBoost Hyperparameter min_child_weight

Like gamma and maxdepth, *min_child_weight* restricts the number of splits of each tree. In the case of *min_child_weight*, this restriction is determined using the Hessian matrix of the loss function.

$min\_child\_weight \in [0, \infty[$.

XGBoost Hyperparameter subsample

In each boosting step, the new tree to be created is usually only trained on a subset of the entire data set, similar to random forest. The *subsample* parameter specifies the portion of the data approach that is randomly selected in each iteration. Its practical significance can be described as follows: an obvious effect of small *subsample* values is a shorter running time for the training of individual trees, which is proportional to the *subsample*.

$subsample \in]0, 1]$.

XGBoost Hyperparameter colsample_bytree

The parameter *colsample_bytree* is the number of features is chosen for the splits of a tree. In XGBoost this choice is made only once for each tree that is created, instead for each split. Here *colsample_bytree* is a relative factor. The number of selected features is therefore $colsample\_bytree \times n$. *colsample_bytree* enables the trees of the ensemble to have a greater diversity. The runtime is also reduced, since a smaller number of splits have to be checked each time (if $colsample\_bytree < 1$).

$colsample_bytree \in]0, 1]$.

XGBoost Hyperparameter lambda

The parameter *lambda* is used for the regularization of the model. This parameter influences the complexity of the model. Its practical significance can be described as follows: as a regularization parameter, *lambda* helps to prevent overfitting. With larger values, smoother or simpler models are to be expected.

$lambda \in [0, \infty[$.

XGBoost Training Results

The results of the best XGBoost model obtained for each segment of arrears and their *KS* values obtained with the test samples are shown below.

0 - No Arrears Segment Model

$KS = 63.36\%$

Table 2.8: No Arrears Segment XGBoost Model Best Hyperparameters Values

HYPERPARAMETER	VALUE
objective	binary:logistic
eval_metric	error
nrounds	100
eta	0.112
booster	gbtree
max_depth	6
min_child_weight	2.72
subsample	0.884
colsample_bytree	0.56
lambda	0.174

1 - 30 Segment Model

$KS = 51.38\%$

Table 2.9: 1 - 30 Segment XGBoost Model Best Hyperparameters Values

HYPERPARAMETER	VALUE
objective	binary:logistic
eval_metric	error
nrounds	100
eta	0.383
booster	gbtree
max_depth	4
min_child_weight	2
subsample	0.915
colsample_bytree	0.652
lambda	0.411

31 - 90 Segment Model

$KS = 34.47\%$

Table 2.10: 31 - 90 Segment XGBoost Model Best Hyperparameters Values

HYPERPARAMETER	VALUE
objective	binary:logistic
eval_metric	error
nrounds	100
eta	0.117
booster	gblinear
max_depth	7
min_child_weight	9.3
subsample	0.961
colsample_bytree	0.96
lambda	0.242

All Arrears Model

$KS = 74.05\%$

Table 2.11: All Segments XGBoost Model Best Hyper parameters Values

HYPERPARAMETER	VALUE
objective	binary:logistic
eval_metric	error
nrounds	100
eta	0.133
booster	gbtree
max_depth	8
min_child_weight	9.66
subsample	0.81
colsample_bytree	0.618
lambda	0.393

2.4.3 Artificial Neural Networks Training

Training a neural network revolves around the following objects [16]:

- Layers, which are combined into a network (or model)
- The input data and corresponding targets
- The loss function, which defines the feedback signal used for learning
- The optimizer, which determines how learning proceeds

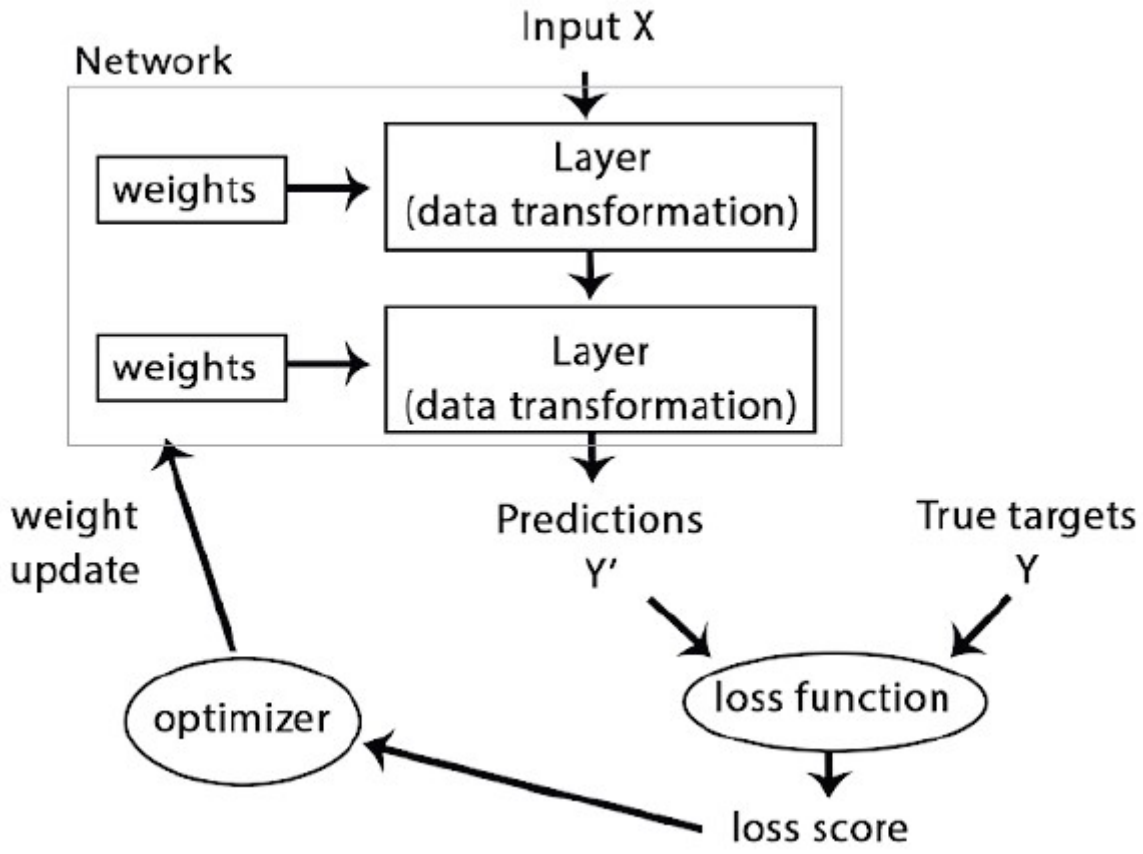

Figure 2.8: Relationship between the network, layers, loss function, and optimizer

Figure 2.8 shows the network, composed of layers that are chained together, maps the input data to predictions. The loss function then compares these predictions to the targets, producing a loss value: a measure of how well the network's predictions match what was expected. The optimizer uses this loss value to update the network's weights.

Building the Neural Networks

When feeding data into a neural network, it's important to first apply one-hot encoding to the categorical variables. This means that for a variable with n categories, you would create

$n - 1$ dummy variables of 0s and 1s. After that, it's essential to standardize the data so that all variables have the same scale. This standardized data is then used as the input for the first layer of the neural network. As for the Y variable, it is kept numerical with 1s and 0s.

A type of network that performs well on binary classification problem is a simple stack of fully connected ("dense") layers [16].

There are two key architecture decisions to be made about such stack of dense layers:

- How many layers to use
- How many hidden units to choose for each layer

The intermediate layers will use relu as activation function, and the final layer will use a sigmoid activation to output a probability (a score between 0 and 1, indicating how likely the sample is to have the target "1": that is, how likely the review is to be positive). A relu (rectified linear unit) is a function meant to zero out negative values, whereas a sigmoid "squashes" arbitrary values into the [0, 1] interval, outputting something that can be interpreted as a probability.

When setting up a neural network, it's important to select a loss function and an optimizer. For a binary classification problem with network output as probabilities, it's best to use the binary cross-entropy loss. Cross-entropy is a reliable choice for models that deal with probabilities, as it measures the distance between probability distributions or, in this case, the actual distribution and its predictions[16].

The optimizer of choice is Adam, (Adaptive Moment Estimation), Adam adjusts the neural network weights more efficiently by calculating adaptive learning rates for each parameter. It uses first and second-moment estimates of the gradients (i.e., the mean and non-centred variance) to perform parameter updates.

Adding Dropout

Dropout is a widely used regularization technique for neural networks, developed by Geoff Hinton at the University of Toronto. When dropout is applied to a layer during training, a certain number of output features are randomly set to zero [16]. For example, if a layer would normally return the vector $[0.2, 0.5, 1.3, 0.8, 1.1]$ for a given input sample during training, applying dropout might result in a vector like $[0, 0.5, 1.3, 0, 1.1]$.

The dropout rate is the fraction of features that are zeroed out, typically set between 0.2 and 0.5. During testing, no units are dropped out; instead, the layer's output values are scaled down by a factor equal to the dropout rate to balance the fact that more units are active than at training time. The technique may seem strange and arbitrary, but why would it help reduce over-adjustment? Hinton says he was inspired, among other things, by a fraud prevention mechanism used by banks.

In his own words: "I went to my bank. The tellers kept changing and I asked one of them why. He said he didn't know, but they changed them a lot. I assumed it must be

because it would take cooperation among the employees to get the bank to cheat. This made me realize that randomly removing a different subset of neurons in each example would prevent conspiracies and thus reduce over-fitting" [16]. The central idea is that by introducing noise into the output values of a neural network layer, you can break random patterns that are not meaningful (what Hinton calls conspiracies), which the network will start to memorize if there is no noise.

Neural Networks Training Results

The results of the best Neural Network model obtained for each segment of arrears and their KS values obtained with the test samples are shown below.

0 - No Arrears Segment Model

$KS = 61.84\%$

Table 2.12: No Arrears Segment Neural Network

Layer	**(type)**	**Activation Function**	**Output Shape**	**Param #**
dense_2	(Dense)	relu	(None, 128)	15616
dropout_1	(Dropout)		(None, 128)	0
dense_1	(Dense)	sigmoid	(None, 64)	8256
dropout	(Dropout)		(None, 128)	0
dense	(Dense)	sigmoid	(None,1)	65
loss:	binary_crossentropy			
optimizer:	adam			
Total params:	23937		(93.50 KB)	
Trainable params:	23937		(93.50 KB)	

1 - 30 Segment Model

$KS = 50.35\%$

Table 2.13: 1 - 30 Segment Neural Network

Layer	**(type)**	**Activation Function**	**Output Shape**	**Param #**
dense_8	(Dense)	relu	(None, 64)	7936
dropout_5	(Dropout)		(None, 64)	0
dense_7	(Dense)	sigmoid	(None, 32)	2080
dropout_4	(Dropout)		(None, 32)	0
dense_6	(Dense)	sigmoid	(None,1)	33
loss:	binary_crossentropy			
optimizer:	adam			
Total params:	10049		(39.25 KB)	
Trainable params:	10049		(39.25 KB)	

31 - 90 Segment Model

$KS = 38.77\%$

Table 2.14: 31 - 90 Segment Neural Network

Layer	**(type)**	**Activation Function**	**Output Shape**	**Param #**
dense_17	(Dense)	relu	(None, 16)	1984
dropout_11	(Dropout)		(None, 16)	0
dense_16	(Dense)	relu	(None, 16)	272
dropout_10	(Dropout)		(None, 16)	0
dense_15	(Dense)	sigmoid	(None,1)	17
loss:	binary_crossentropy			
optimizer:	adam			
Total params:	2273		(8.88 KB)	
Trainable params:	2273		(8.88 KB)	

All Arrears Model

$KS = 73.59\%$

Table 2.15: All Arrears Neural Network

Layer	**(type)**	**Activation Function**	**Output Shape**	**Param #**
dense_20	(Dense)	relu	(None, 16)	1984
dropout_13	(Dropout)		(None, 16)	0
dense_19	(Dense)	relu	(None, 16)	272
dropout_12	(Dropout)		(None, 16)	0
dense_18	(Dense)	sigmoid	(None,1)	17
loss:	binary_crossentropy			
optimizer:	adam			
Total params:	2273		(39.25 KB)	
Trainable params:	2273		(39.25 KB)	

Chapter 3

RESULTS AND DISCUSSION

This section deals with the interpretation of the results of the trained models by taking their application in practice.

3.1 Interpretation of Logistic Regression Coefficients

The estimated coefficients β_n in a regression can be better understood by considering the concept of relative risk. Relative risk is the ratio of the probability of an event occurring (p) to the probability of it not occurring ($1 - p$), also known as odds ratios. Odds ratios indicate how much the odds change per unit change in the explanatory variables[14].

The exponential of β_n, $\exp(\beta_n)$, represents the relative risk, which measures the influence of the variables x_n on the risk of the event occurring, assuming all other variables in the model remain constant. Once the values of β_n have been estimated, we can determine the probability of the event for different values of x_n.

The coefficients of logistic regression are not as easy to interpret as those of linear regression. While the β_n coefficients are useful for model validation tests, $\exp(\beta_n)$ is easier to interpret. $\exp(\beta_n)$, represents the change in the odds ratio for each one-unit change in the variable x_n.

For example, take the variable V4 ($cp\_pl \leq 0.77$) in the No Arrears Segment Model. It means *"individuals whose value of instalments paid over time is up to 0.77"*. Its estimated coefficient β_4 is 1.612, so $\exp(\beta_4)$ = 5.014 indicates that the odds ratio of *"individuals whose value of instalments paid over time is up to 0.77"*, is 5.014 times higher than other customers if all other variables are held constant. In Other words, the probability that individuals with $cp\_pl \leq 0.77$ will make a payment next month is 5.014 times higher than others.

3.2 Interpretation of XGBoost models results

XGBoost is often considered a "black box" algorithm because, while it is highly effective at making accurate predictions, it can be difficult to understand how it arrives at these predictions. This is due to the complexity of the decision tree models it creates and how these trees are combined to form the final model.

Machine learning models like XGBoost, which utilize ensemble and boosting techniques, generate multiple decision trees during the training process. Each tree is constructed to fix the errors of the previous one, leading to a final model that is a weighted sum of many trees.[15]

Because of this combination of models and their interactions, it can be challenging to precisely determine which features are influencing the predictions and how they are doing so. Nevertheless, ongoing efforts are being made to enhance the interpretability of these models, including the use of feature importance techniques, SHAP values, and tree visualizations, which can provide insight into model decisions.[17]

The significance of variables in the XGBoost algorithm pertains to the impact of each feature in the dataset on the accuracy of the model. From an interpretability standpoint, this helps in understanding which variables carry the most weight in the decisions made by the model and how each influences the final result.

In XGBoost, the importance of variables can be assessed in various ways, including information gain, coverage, or frequency of occurrence of a feature in the decision trees. These metrics offer a clear understanding of the relevance of each variable and enable data scientists and analysts to make well-informed decisions regarding feature selection and model optimization.

Gain represents the average contribution of a feature to model improvement each time it is utilized in a tree. A higher value signifies the feature's greater importance in making splits that enhance model performance.

Weight refers to the frequency of a feature's appearance in all trees of the model. A feature with a higher weight is deemed more significant.

Cover measures the frequency of a feature's utilization in the trees, weighted by the amount of data passing through those splits. A high coverage suggests that the feature has a substantial impact on the model's predictions.

The top 10 variables with the highest gain for each range of arrears are listed below. In simpler terms, these are the 10 variables that, in terms of gain, contribute to divisions that improve the model's performance.

0 - No Arrears Segment Model

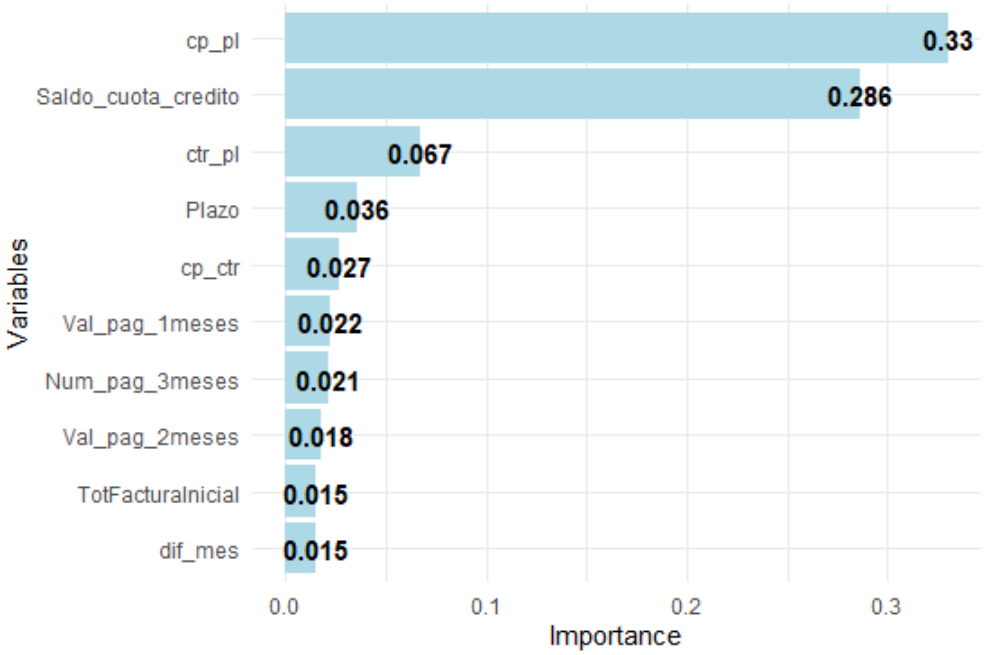

Figure 3.1: Top 10 Important Variables in XGBoost No Arrears Segment Model

1 - 30 Segment Model

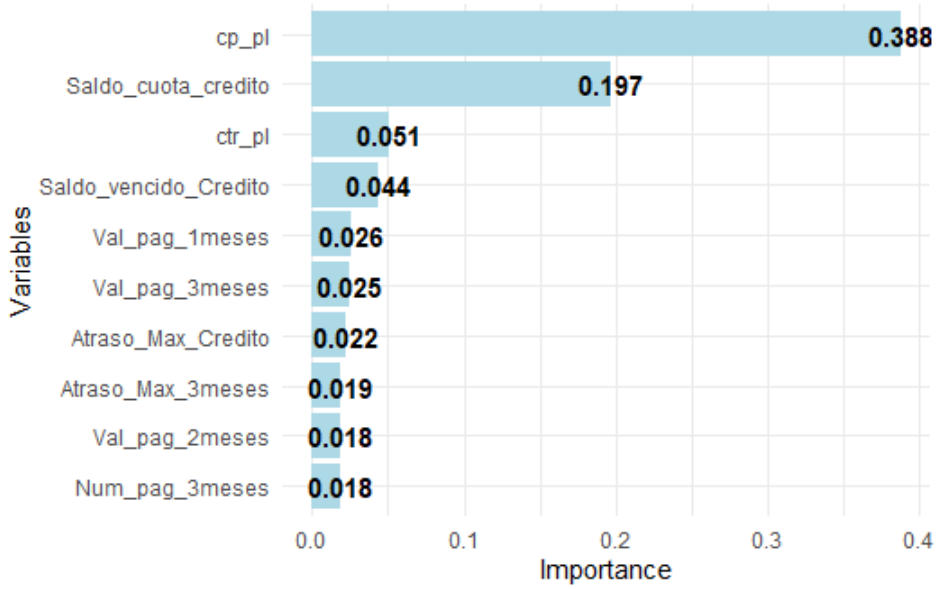

Figure 3.2: Top 10 Important Variables in XGBoost 1-30 Segment Model

31 - 90 Segment Model

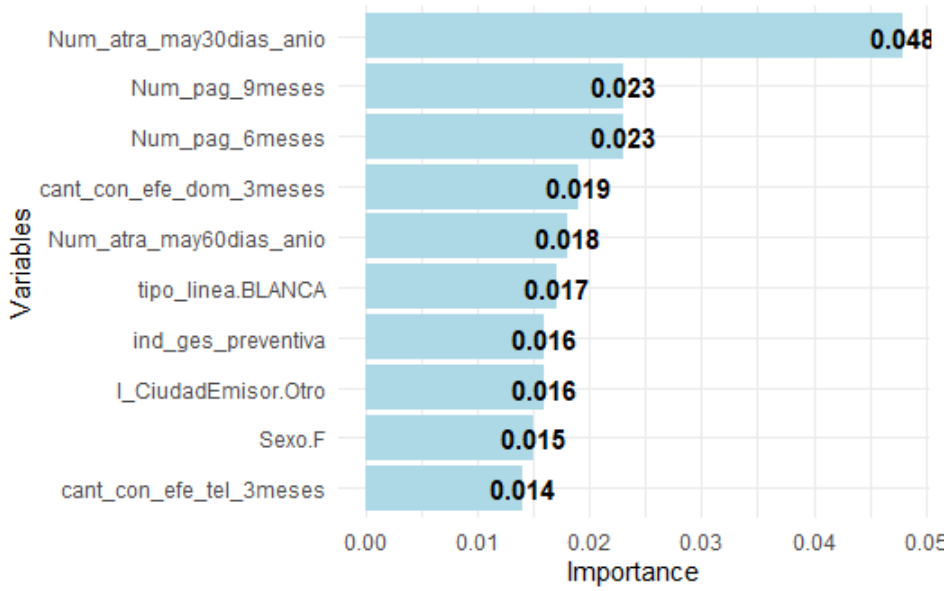

Figure 3.3: Top 10 Important Variables in XGBoost 31-90 Segment Model

All Segments Model

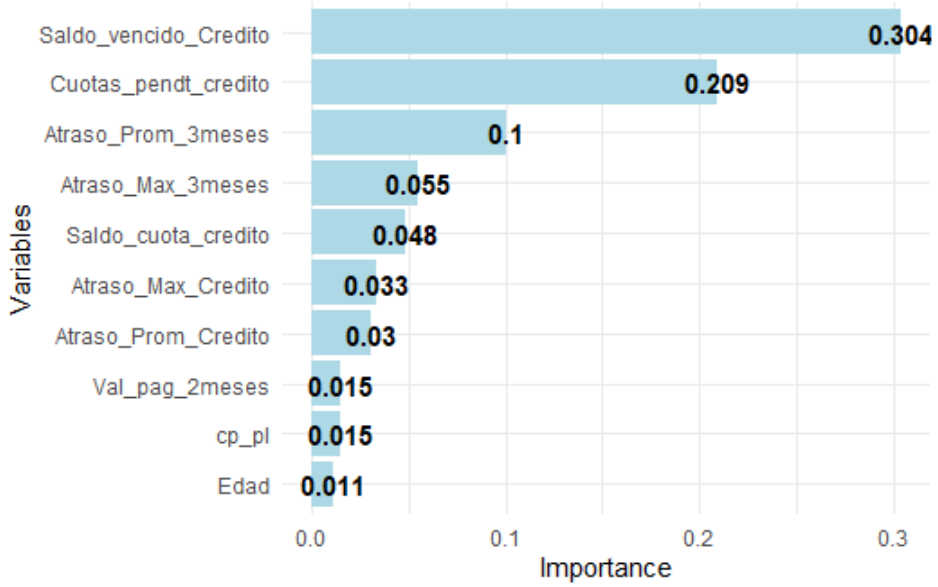

Figure 3.4: Top 10 Important Variables in XGBoost All Segments Model

It has been noted that for the 0 - No Arrears Segment (Figure 3.1) and 1-30 segment(Figure 3.2), the variables with the most significant influence on the calculation of the probability

of payment in the next month are cp_pl, saldo_cuota_credito, and ctr_pl. In contrast, for the 31-90 arrears segment, the variables that most impact the calculation of the probability of payment in the next month are Num_atra_may30dias_anio, Num_pag_9meses, and Num_pag_6meses.

Moreover, in predicting the likelihood of an individual making suspensions before reaching 30 days in arrears in the next six months, the variables with the most influence are Saldo_vencido_credito, Cuotas_pendientes_credito, and Atraso_prom_3meses.

Therefore, it is possible to create collection strategies or stablish protocols for using XGBoost results for credit risk management scenarios. For the ranges of arrears No Arrears and 1-30 that include the variables:

- **cp_pl:** installments paid over installments.
- **credit_quota_balance:** Client's installment balance at the observation point, including principal and interest.
- **ctr_pl:** outstanding installments over installment

For higher arrears ranges, variable-focused strategies:

- **Num_atra_may30dias_anio:** Number of arrears greater than 30 days in a year.
- **Num_pag_9meses:** Number of payments in the last nine months.
- **Num_pag_6meses:** Number of payments made in last six months.

To prevent default in the medium term, use the variables:

- **Saldo_vencido_credito:** Balance overdue on the loan. **Cuotas_pendientes_credito:** Outstanding installments of the loan. **Atraso_prom_3meses:** Average arrears in last three months.

About Neural Networks Interpretation, like XGBoost, are often considered "black boxes" due to their complexity and the way they process information. Despite this, researchers have developed various techniques to help interpret how neural networks make decisions[18].

In this work we do not address interpretation in neural networks and leave it as an open topic for future work.

3.3 KS Test Results Summary

Based on the data in the Table 3.1, it appears that both XGBoost and Artificial Neural Networks (ANN) tend to outperform Logistic Regression (LR) in some segments. However, the superiority of one model over the other may depend on the specific segment.

- In Segment 0 Models, XGBoost (63.36%) and ANN (61.84%) outperform LR (56.42%).
- In Segment Model 1 - 30, XGBoost (51.38%) and ANN (50.35%) also outperform LR (47.32%). However, in the 31 - 90 Segment Model, ANN (38.77%) outperforms LR (36.62%), but XGBoost (34.47%) does not.
- Finally, in the All Delays Model, both XGBoost (74.05%) and ANN (73.59%) outperform LR (71.01%).

These findings support the hypothesis that XGBoost and ANN outperform LR in predicting events. It's essential to consider that these results can vary based on the data's characteristics, the quality of feature engineering, and the model's hyperparameters, among other factors. Additionally, while XGBoost and ANN offer higher accuracy, they may also be more intricate and computationally demanding compared to LR. Hence, the choice of model might depend on balancing accuracy and computational efficiency, as well as the specific requirements of the prediction task.

Lastly, it's crucial to note that these results are specific to this dataset and cannot be generalized to other datasets or prediction tasks. Therefore, it's good practice to cross-validate and fine-tune the hyperparameters for each model and dataset.

Table 3.1: All Models KS Results by Arrears Segment

0 Segment Models

LR	XGB	ANN
56.42%	63.36%	61.84%

1 - 30 Segment Model

LR	XGB	ANN
47.32%	51.38%	50.35%

31 - 90 Segment Model

LR	XGB	ANN
36.62%	34.47%	38.77%

All Arrears Model

LR	XGB	ANN
71.01%	74.05%	73.59%

Chapter 4

CONCLUSIONS

1. To reach the specific goal of evaluating the performance of the Xgboost and ANN compared to logistic regression we use the Kolmogorov-Smirnov statistic. The XGBoost algorithm consistently showed better performance in the 0-no arrears, 1-30 arrears segments and when all segments were considered together with 63.36%, 51.38% and 74.05% respectively. This suggests that XGBoost is more effective in binary classification compared to logistic regression and neural networks.

2. Although logistic regression requires more time for preprocessing and training due to feature engineering and sample balancing, its performance in terms of KS did not outperform XGBoost and neural networks in any of the arrears segments.

3. In the 31-90 arrears segment, neural networks outperformed XGBoost with a 38.77% KS value, indicating that the complexity and adaptability of neural networks can be advantageous in certain scenarios, despite the longer training time required.

4. The importance of feature engineering and data balancing for logistic regression highlights the substantial influence these steps can have on its performance, whereas machine learning algorithms, XGBoost and neural networks do not need as much work to give far superior results.

5. Regarding the specific objective on the interpretation of the results, it can be seen that although logistic regression has inferior results to XGBoost and neural networks, it is easy to interpret, which makes it one of the favourites in the practice of implementing scoring models.

   The interpretation of the XGboost results, on the other hand, is subject to the contribution of the variable in the splitting of the random trees, and neural networks are still under investigation to create a satisfactory interpretation.

   The results suggest that, when creating a scoring model, you have to decide what you want to sacrifice: interpretability or predictive

6. In compliance with the research goal, If you want to be clear about what happens to

individuals in order for them to default, the best option is to use logistic regression. This methodology is used in the planning and acquisition stages of the credit cycle, and sometimes to describe the probability that a customer will decide to terminate all business with the lender.

On the other hand, if more predictive or discriminatory power is desired, XGBoost or neural networks are undoubtedly the best option. These algorithms can be used in the servicing and collection stages of the credit cycle, as there is much more data to analyse and collection results can change in a very short time.

7. With the results obtained, protocols can be established according to the best variables obtained, or with the probability values obtained, as shown in section 3.2.

8. Explore the combination of models to leverage the individual strengths of each algorithm, such as XGBoost and neural networks, to improve prediction accuracy in different segments or scenarios.

9. Making a detailed analysis of the specific characteristics of the 31-90 arrears segment that favoured the superior performance of the neural networks, and in order to interpret the results of neural networks, it is recommended to keep an eye on new research on this topic and try to implement it in future work.

Chapter 5

APPENDIX

5.1 Data Exploratory Analisis

N	TYPE	VARIABLE	% NA / NULL	ACTION
1	Demographics	CodigoEmisor	0%	Keep
2	Demographics	NombreEmisor	0%	Keep
3	Demographics	CodigoAbrev	0%	Keep
4	Demographics	Num_SolCre	0%	Keep
5	Demographics	Num_Factura	0%	Keep
6	Demographics	Num_Operacion	0%	Keep
7	Demographics	FechaCortePO	0%	Keep
8	Demographics	FechaGeneracion	0%	Keep
9	Demographics	FechaFactura	0%	Keep
10	Demographics	FechaColocacion	0%	Keep
11	Demographics	FechaCorteAnexo	0%	Keep
12	Demographics	FechaCreacion	0%	Keep
13	Demographics	EstadoCivil	0%	Keep
14	Demographics	N_EstadoCivil	0%	Attach NULL to SOLTERO
15	Demographics	Sexo	0%	Keep
16	Demographics	Nacionalidad	0%	Keep
17	Demographics	N_Nacionalidad	0%	Keep
18	Demographics	FechaNac	0%	Keep
19	Demographics	CantonCiudad	0%	Keep

20	Demographics	N_CantonCiudad	0%	Keep the NULL category
21	Demographics	UbicacionDomicilio	0%	Keep
22	Demographics	N_UbicacionDomicilio	0%	Keep
23	Demographics	Vivienda	0%	Keep
24	Demographics	N_Vivienda	0%	Keep
25	Demographics	RelacionTrabajo	0%	Keep
26	Demographics	ClasificacionCliente	0%	Keep
27	Demographics	UbicacionTrabajo	0%	Keep
28	Demographics	N_UbicacionTrabajo	0%	Keep
29	Demographics	ActividadEconomicaEmpresa	0%	Keep
30	Demographics	N_ActividadEconomicaEmpresa	30%	Keep NULL Category
31	Demographics	ActividadEconomicaCliente	0%	Keep
32	Demographics	N_ActividadEconomicaCliente	0%	Delete
33	Demographics	CargoActual	0%	Keep
34	Demographics	N_CargoActual	8%	Keep NULL Category
35	Demographics	AntiguedadEmpresaAnios	0%	Keep
36	Demographics	AntiguedadEmpresaMeses	0%	Keep
37	Demographics	IngresosPropios	0%	Keeping the Outliers
38	Demographics	IngresosConyuge	0%	Keeping the Outliers
39	Demographics	OtrosIngresos	0%	Keeping the Outliers
40	Demographics	CargasFamiliares	1%	Replace the NO with 0 and keep the outliers
41	Demographics	NivelEducacionCliente	0%	Delete
42	Demographics	N_NivelEducacionCliente	0%	Join NULL to PRIMARIA
43	Demographics	Num_Telef_Celular	0%	Delete
44	Demographics	Num_Telef_Empresa	0%	Delete
45	Demographics	Num_Telef_Particular1	0%	Delete
46	Demographics	ID_Num_Telef_Celular	0%	Keep
47	Demographics	ID_Num_Telef_Empresa	0%	Keep
48	Demographics	ID_Num_Telef_Particular1	0%	Keep
49	Demographics	Numero_Ruc	0%	Delete
50	Demographics	ID_Numero_Ruc	0%	Keep
51	Demographics	Cant_Refer_Personales	0%	Replace the NO with 0 and keep the outliers

52	Demographics	Cant_Direcciones	0%	Replace the NO with 0 and keep the outliers
53	Demographics	Cant_Num_Telef_Trabajo	0%	Replace the NO with 0 and keep the outliers
54	Demographics	Cant_Num_Telef_Particular	0%	Replace the NO with 0 and keep the outliers
55	Demographics	Cant_Num_Telef_Referen	0%	Replace the NO with 0 and keep the outliers
56	Demographics	Cant_Dir_Trabajo	0%	Replace the NO with 0 and keep the outliers
57	Demographics	Cant_Dir_Particular	0%	Replace the NO with 0 and keep the outliers
58	Demographics	Direccion_Domicilio	0%	Delete
59	Demographics	Direccion_Trabajo	0%	Delete
60	Demographics	Pto_Facturacion	0%	Delete
61	Demographics	ValorCuotaGratis	0%	Keeping the Outliers
62	Demographics	fechafacturafull	0%	Delete
63	Demographics	identidad	0%	Keep
64	Operational	CodigoEmisor	0%	Delete
65	Operational	NombreEmisor	0%	Delete
66	Operational	CodigoAbrev	0%	Delete
67	Operational	Num_SolCre	0%	Delete
68	Operational	Num_Factura	0%	Delete
69	Operational	Num_Operacion	0%	Delete
70	Operational	FechaCortePO	0%	Keep
71	Operational	FechaGeneracion	0%	Delete
72	Operational	FechaFactura	0%	Delete
73	Operational	FechaAprobacion	0%	Delete
74	Operational	FechaCorteAnexo	0%	Delete
75	Operational	CapitalInteres	0%	Replace the NO with 0 and keep the outliers. Analyze by rows
76	Operational	Inicial	0%	Keep

77	Operational	TotFacturaInicial	0%	Keep
78	Operational	Cant_Productos	0%	Keep
79	Operational	Linea	0%	Keep NULL Category/Parse by Rows
80	Operational	SubLinea	0%	Delete
81	Operational	Cadena	0%	Keep
82	Operational	CiudadEmisor	0%	Keep
83	Operational	Plazo	0%	Keep
84	Operational	TasaCredito	0%	Keeping the Outliers
85	Operational	TipoAmortizacion	0%	Include NULL in Fija.
86	Operational	CodigoVendedor	0%	Delete
87	Operational	MesesGracia	0%	Replace the NO with 0 and keep the outliers.
88	Operational	CuotasGratis	20%	Keep NULL Category
89	Operational	ValorCuota	0%	Keep
90	Operational	CodigoCredito	0%	Delete
91	Operational	TipoOperacion	0%	Delete
92	Operational	Pto_Facturacion	0%	Keep
93	Operational	incialbono	83%	Replace AN with 0
94	Operational	tipoinicialbono	83%	Mantener CategorÃa NULL
95	Operational	documentofull	0%	Delete
96	Operational	identidad	0%	Delete
97	Behavioural	CodigoEmisor	0%	Delete
98	Behavioural	NombreEmisor	0%	Delete
99	Behavioural	CodigoAbrev	0%	Delete
100	Behavioural	Num_SolCre	0%	Delete
101	Behavioural	Num_Factura	0%	Delete
102	Behavioural	Num_Operacion	0%	Delete
103	Behavioural	FechaCortePO	0%	Delete
104	Behavioural	FechaGeneracion	0%	Delete
105	Behavioural	FechaFactura	0%	Delete
106	Behavioural	FechaColocacion	0%	Delete
107	Behavioural	FechaCorteAnexo	0%	Delete
108	Behavioural	FechaCreacion	0%	Delete
109	Behavioural	Pto_Facturacion	0%	Delete
110	Behavioural	Atraso_Max_Credito	0%	Replace the NA with 0 and keep the outliers

111	Behavioural	Atraso_Max_3meses	0%	Keeping the Outliers
112	Behavioural	Atraso_Max_6meses	0%	Keeping the Outliers
113	Behavioural	Atraso_Max_9meses	0%	Keeping the Outliers
114	Behavioural	Atraso_Max_12meses	0%	Keeping the Outliers
115	Behavioural	Atraso_Prom_Credito	0%	Keeping the Outliers
116	Behavioural	Atraso_Prom_3meses	0%	Keeping the Outliers
117	Behavioural	Atraso_Prom_6meses	0%	Keeping the Outliers
118	Behavioural	Atraso_Prom_9meses	0%	Keeping the Outliers
119	Behavioural	Atraso_Prom_12meses	0%	Keeping the Outliers
120	Behavioural	Rango_mora_mesact	0%	Keep
121	Behavioural	Rango_mora_max_mesant	0%	Join NULL to 000-Al Dia
122	Behavioural	Rango_mora_max_3meses	0%	Downgrade to more than 120 days
123	Behavioural	Rango_mora_max_6meses	0%	Downgrade to more than 120 days
124	Behavioural	Rango_mora_max_9meses	0%	Downgrade to more than 120 days
125	Behavioural	Rango_mora_max_12meses	0%	Downgrade to more than 120 days
126	Behavioural	Num_atra_may30dias_anio	89%	Replace the NA with 0 and keep the outliers
127	Behavioural	Num_atra_may60dias_anio	89%	Replace the NA with 0 and keep the outliers
128	Behavioural	Num_atra_may90dias_anio	100%	Delete
129	Behavioural	Val_pag_3meses	5%	Replace the NA with 0 and keep the outliers
130	Behavioural	Val_pag_2meses	8%	Replace the NA with 0 and keep the outliers
131	Behavioural	Val_pag_1meses	21%	Replace the NA with 0 and keep the outliers
132	Behavioural	Num_pag_3meses	5%	Replace the NA with 0 and keep the outliers

133	Behavioural	Num_pag_6meses	1%	Replace the NA with 0 and keep the outliers
134	Behavioural	Num_pag_9meses	1%	Replace the NA with 0 and keep the outliers
135	Behavioural	Num_pag_12meses	1%	Replace the NA with 0 and keep the outliers
136	Behavioural	Dia_mora_sig_1mes	0%	To define Y
137	Behavioural	Dia_mora_sig_2mes	0%	
138	Behavioural	Dia_mora_sig_3mes	0%	
139	Behavioural	Dia_mora_sig_4mes	0%	
140	Behavioural	Dia_mora_sig_5mes	0%	
141	Behavioural	Dia_mora_sig_6mes	0%	
142	Behavioural	Saldo_cuota_credito	8%	Replace the NA with 0 and keep the outliers
143	Behavioural	Saldo_cuota_sig_1mes	0%	Delete
144	Behavioural	Saldo_cuota_sig_2mes	0%	Delete
145	Behavioural	Saldo_cuota_sig_3mes	0%	Delete
146	Behavioural	Saldo_cuota_sig_4mes	0%	Delete
147	Behavioural	Saldo_cuota_sig_5mes	0%	Delete
148	Behavioural	Saldo_cuota_sig_6mes	0%	Delete
149	Behavioural	Saldo_vencido_Credito	8%	Delete
150	Behavioural	Saldo_vencido_sig_1mes	0%	Delete
151	Behavioural	Saldo_vencido_sig_2mes	0%	Delete
152	Behavioural	Saldo_vencido_sig_3mes	0%	Delete
153	Behavioural	Saldo_vencido_sig_4mes	0%	Delete
154	Behavioural	Saldo_vencido_sig_5mes	0%	Delete
155	Behavioural	Saldo_vencido_sig_6mes	0%	Delete
156	Behavioural	Pago_efec_1mes	28%	Para definir Y
157	Behavioural	Pago_efec_2mes	0%	Delete
158	Behavioural	Pago_efec_3mes	0%	Delete
159	Behavioural	Pago_efec_4mes	0%	Delete
160	Behavioural	Pago_efec_5mes	0%	Delete
161	Behavioural	Pago_efec_6mes	0%	Delete

162	Behavioural	Cuotas_pagad_credito	0%	Replace the NA with 0 and keep the outliers
163	Behavioural	Cuotas_pendt_credito	8%	Replace the NO with 0 and keep them outliers. Negative values place 0
164	Behavioural	Estado_credito	0%	Delete
165	Behavioural	ValorCuotaGratis	0%	Mantener los atÃ-picos
166	Behavioural	identidad	0%	Delete
167	Collection	fechaCortePO	0%	Delete
168	Collection	fechaGeneracion	0%	Delete
169	Collection	fechaDesdeUniverso	0%	Delete
170	Collection	fechaHastaUniverso	0%	Delete
171	Collection	ult_resp_ges_tel	68%	Recategorize
172	Collection	tipos_tel_vigentes	0%	Keep NULL Category
173	Collection	ind_ges_preventiva	0%	Keep
174	Collection	cant_ges_tel_1mes	86%	Replace NA with 0
175	Collection	cant_ges_dom_1mes	80%	Replace NA with 0
176	Collection	cant_ges_tel_3meses	80%	Replace NA with 0
177	Collection	cant_ges_dom_3meses	74%	Replace NA with 0
178	Collection	cant_ges_tel_6meses	73%	Replace NA with 0
179	Collection	cant_ges_dom_6meses	69%	Replace NA with 0
180	Collection	cant_ges_tel_9meses	70%	Replace NA with 0
181	Collection	cant_ges_dom_9meses	66%	Replace NA with 0
182	Collection	cant_ges_tel_12meses	68%	Replace NA with 0
183	Collection	cant_ges_dom_12meses	64%	Replace NA with 0
184	Collection	cant_ges_efe_tel_mesant	88%	Replace NA with 0
185	Collection	cant_ges_efe_dom_mesant	84%	Replace NA with 0
186	Collection	cant_ges_efe_tel_3meses	81%	Replace NA with 0
187	Collection	cant_ges_efe_dom_3meses	74%	Replace NA with 0
188	Collection	cant_ges_efe_tel_6meses	74%	Replace NA with 0
189	Collection	cant_ges_efe_dom_6meses	69%	Replace NA with 0
190	Collection	cant_ges_efe_tel_9meses	71%	Replace NA with 0
191	Collection	cant_ges_efe_dom_9meses	66%	Replace NA with 0
192	Collection	cant_ges_efe_tel_12meses	69%	Replace NA with 0
193	Collection	cant_ges_efe_dom_12meses	64%	Replace NA with 0
194	Collection	cant_con_efe_tel_mesant	95%	Replace NA with 0
195	Collection	cant_con_efe_dom_mesant	96%	Replace NA with 0

196	Collection	cant_con_efe_tel_3meses	89%	Replace NA with 0
197	Collection	cant_con_efe_dom_3meses	91%	Replace NA with 0
198	Collection	cant_con_efe_tel_6meses	84%	Replace NA with 0
199	Collection	cant_con_efe_dom_6meses	88%	Replace NA with 0
200	Collection	cant_con_efe_tel_9meses	81%	Replace NA with 0
201	Collection	cant_con_efe_dom_9meses	86%	Replace NA with 0
202	Collection	cant_con_efe_tel_12meses	79%	Replace NA with 0
203	Collection	cant_con_efe_dom_12meses	84%	Replace NA with 0
204	Collection	val_mejor_resp_tel_3meses	0%	Delete
205	Collection	desc_mejor_resp_tel_3meses	80%	Keep NULL and Re-categorize (Keep up to the second level of response, if it is NO UBICADO keep the first level)
206	Collection	val_mejor_resp_dom_3meses	0%	Delete
207	Collection	desc_mejor_resp_dom_3meses	74%	Keep NULL and Re-categorize (Keep up to the second level of response, if it is NO UBICADO keep the first level)
208	Collection	val_mejor_resp_tel_6meses	0%	Delete
209	Collection	desc_mejor_resp_tel_6meses	73%	Keep NULL and Re-categorize (Keep up to the second level of response, if it is NO UBICADO keep the first level)
210	Collection	val_mejor_resp_dom_6meses	0%	Delete
211	Collection	desc_mejor_resp_dom_6meses	69%	Keep NULL and Re-categorize (Keep up to the second level of response, if it is NO UBICADO keep the first level)
212	Collection	val_mejor_resp_tel_9meses	0%	Delete

213	Collection	desc_mejor_resp_tel_9meses	70%	Keep NULL and Re-categorize (Keep up to the second level of response, if it is NO UBICADO keep the first level)
214	Collection	val_mejor_resp_dom_9meses	0%	Keep
215	Collection	desc_mejor_resp_dom_9meses	66%	Keep NULL and Re-categorize (Keep up to the second level of response, if it is NO UBICADO keep the first level)
216	Collection	val_mejor_resp_tel_12meses	0%	Delete
217	Collection	desc_mejor_resp_tel_12meses	68%	Keep NULL and Re-categorize (Keep up to the second level of response, if it is NO UBICADO keep the first level)
218	Collection	val_mejor_resp_dom_12meses	0%	Delete
219	Collection	desc_mejor_resp_dom_12meses	80%	Keep NULL and Re-categorize (Keep up to the second level of response, if it is NO UBICADO keep the first level)
220	Collection	identidad	0%	Delete

5.2 Dummy Variables

5.2.1 0 - No Arrears Segment

- **V1**: Good
  cp_pl≤0.56 and Pago_efec_1mes(0; 77.62] and Num_pag_3meses(2;3]
- **V2**: Good
  cp_pl≤0.56 and Pago_efec_1mes(77.62; 102.88] and Val_pag_3meses(159.2;293.94]
- **V3**: Good
  cp_pl(0.56;0.77]
- **V4**: Bad
  cp_pl>0.77
- **V5**: Bad
  Saldo_cuota_credito≤197.79
- **V6**: Good
  Saldo_cuota_credito(197.79;1121.02]
- **V7**: Good
  Saldo_cuota_credito>1121.02 and cp_ctr(0.8;1] and Val_pag_1mes>24.61
- **V8**: Good
  ctr_pl≤6.5 and Num_pag_6meses(3;5]
- **V9**: Good
  ctr_pl≤6.5 and Num_pag_6meses(5;6] and Cuotas_pagad_credito≤11
- **V10**: Good
  Num_pag_12meses(3;9] and Plazo>11
- **V11**: Bad
  Num_pag_12meses>10
- **V12**: Good
  dif_mes≤4 and Num_pag_9meses≤4 and CapitalInteres>891.86
- **V13**: Good
  dif_mes(7;9] and (canal_vta==Artefacta | canal_vta==Tropimotors)
- **V14**: Bad
  dif_mes>10
- **V15**: Good
  (CuotasGratis==DESCUENTO EN CUOTAS |
  CuotasGratis==MEDIAS CUOTAS) and TotFacturaInicial(705.72;1282.43]

- **V16**: Good
(CuotasGratis==DESCUENTO EN CUOTAS |
CuotasGratis==MEDIAS CUOTAS) and TotFacturaInicial>1772.12

- **V17**: Good
(CuotasGratis==BONO INICIAL + N CUOTAS GRATIS |
CuotasGratis==CUOTAS GRATIS)
and inicialbono≤0
and (region==REGIONAL 1 | region==REGIONAL 2 | region==REGIONAL 3 |
region==REGIONAL 5 | region==REGIONAL 6 | region==REGIONAL 7)

- **V18**: Bad
(CuotasGratis==BONO INICIAL + N CUOTAS GRATIS | CuotasGratis==CUOTAS GRATIS)
and inicialbono≤0
and (region==QUITO | region==GUAYAQUIL)

- **V19**: Good
CuotasGratis==NULL

- **V20**: Bad
TasaCredito(0;15] and Inicial>59

- **V21**: Good
TasaCredito>15

- **V22**: Good
(tipoinicialbono==CUADRUPLICA | tipoinicialbono==BONO INICIAL BARATODO)
and ind_ges_preventiva==1

- **V23**: Good
(tipoinicialbono==NULL and Val_pag_2meses(102.58;136.09]
and ValorCuota(47.18;77.8]) | (tipoinicialbono==NULL and Val_pag_2meses(153.54;236.16]
and ValorCuota>77.8)

- **V24**: Bad
Cant_Productos≤1 and Cant_Num_Telef_Trabajo≤1

- **V25**: Good
Cant_Productos>2 and (linea==Tienda | linea==Recojo) and RelacionTrabajo==NO

- **V26**: Good ctr_pl≤0.75

5.2.2 1 - 30 Segment

- **V1**: Bad
Saldo_cuota_credito≤123.25

- **V2**: Good
  Saldo_cuota_credito(123.25;311.02] (cont)
- **V3**: Good
  Saldo_cuota_credito(311.02;922.58]
  and Pago_efec_1mes(0;94.46]
  and Atraso_Prom_Credito≤1
- **V4**: Good
  Saldo_cuota_credito>922.58
  and Cuotas_pendt_credito≤0
  and Pago_efec_1mes(58.87;117.52]
- **V5**: Good
  cp_pl≤0.21 | (cp_pl(0.65;0.83])
- **V6**: Good
  cp_pl(0.28;0.65] and Num_pag_3meses(2;3] and Atraso_Max_Credito≤5
- **V7**: Bad
  cp_pl>0.83 (cont)
- **V8**: Good
  Num_pag_3meses(2;3] and Atraso_Max_Credito≤5
- **V9**: Good
  ctr_pl≤3.2 and Saldo_vencido_Credito≤0
- **V10**: Good
  ctr_pl(0.32;0.77] and Atraso_Max_3meses≤0 and cp_ctr(0.77;1]
- **V11**: Good
  ctr_pl(0.77;0.87]
- **V12**: Bad
  ctr_pl>0.87 (cont)
- **V13**: Good
  Cuotas_pagad_credito(3;14] (cont)
- **V14**: Bad
  Cuotas_pagad_credito≤3 | Cuotas_pagad_credito>14
- **V15**: Good
  dif_mes≤5 and Atraso_max_12Meses≤0
- **V16**: Good
  dif_mes(5;14]

- **V17**: Bad
  dif_mes>14
- **V18**: Good
  Num_pag_12meses(3;10]
- **V19**: Bad
  Num_pag_12meses≤3 | Num_pag_12meses>10
- **V20**: Good
  Plazo(11;15] and Num_pag_9meses(3;7]
- **V21**: Bad
  Plazo(16;18] and Num_pag_9meses(7;9]
- **V22**: Good
  Plazo(18;19] and Atraso_Prom_3meses≤0
- **V23**: Good
  Plazo>23 and Atraso_Prom_3meses≤0
  and (CuotasGratis==NULL | CuotasGratis==DESCUENTO EN CUOTAS)
- **V24**: Good
  Atraso_Prom_12meses≤0 and Num_pag_6meses≤5
- **V25**: Bad
  Atraso_Prom_12meses≤0 and Num_pag_6meses5;6] and CapitalInteres≤804.39
- **V26**: Good
  Atraso_Prom_12meses≤0 and Num_pag_6meses5;6] and CapitalInteres>10.9746
- **V27**: Good
  Atraso_Max_9meses≤0 and IngresosPropios(339;340] and RelacionTrabajo==NO
- **V28**: Good
  Atraso_Max_9meses≤0 and IngresosPropios(352;354]
- **V29**: Good
  Atraso_Max_9meses≤0 and IngresosPropios>366 and TasaCredito≤15
- **V30**: Bad
  Atraso_Max_9meses>19 (cont)
- **V31**: Good
  Atraso_Max_6meses≤0 and incialbono≤0
- **V32**: Bad
  Atraso_Max_9meses>0

- **V33**: Good
  Atraso_Prom_9meses≤0 and TotFacturaInicial(470.74;1029.66]
  and Val_pag_2meses(55.62;182.74]

- **V34**: Good
  Atraso_Prom_9meses≤0 and TotFacturaInicial(1029.66;2068.54]
  and (tipoinicialbono==NULL | tipoinicialbono==B. INICIAL CAMPAIGN)

- **V35**: Good
  Atraso_Prom_6meses≤0
  and (region==REGIONAL 1 | region==REGIONAL 2 | region==REGIONAL 6 | region==GUAYAQUI
  and Val_pag_3meses(93.04;323.04]

- **V36**: Good
  Atraso_Prom_6meses≤0
  and (region==QUITO | region==REGIONAL 3 | region==REGIONAL 5)
  and cant_ges_efe_tel_3meses≤0

- **V37**: Bad
  Atraso_Prom_6meses>0 and cant_ges_tel_6meses>0

- **V38**: Good
  Atraso_Max_3meses≤0 and Val_pag_1meses(31.74;62.59] and Inicial≤54

- **V39**: Good
  Atraso_Max_3meses≤0 and Val_pag_1meses(62.59;115.67] and cant_ges_tel_3meses≤0

- **V40**: Bad
  Linea==COMUNICACIONES

- **V41**: Good
  (Linea==BLANCA NACIONAL | Linea==VIDEO |
  Linea==BLANCA IMPORTADA | Linea==MUEBLES)
  and (ult_resp_ges_tel==NULL | ult_resp_ges_tel==MENSAJE A TERCEROS |
  ult_resp_ges_tel==CONTACTO SIN COMPROMISO |
  ult_resp_ges_tel==COMPROMISO DE PAGO FAMILIAR |
  ult_resp_ges_tel==COMPROMISO DE PAGO |
  ult_resp_ges_tel==CANCELADO |
  ult_resp_ges_tel==PROMOCIONES |
  ult_resp_ges_tel==RECOJO | CLTE FALLECIDO)
  and ind_ges_preventiva==1

- **V42**: Bad
  (Linea==ELECTRODOMESTICO | Linea==AUDIO | Linea==COMPUTO) and Cant_Productos≤2

- **V43**: Good
  cant_ges_dom_1mes≤0 and ValorCuota(58.54;106.73] and cant_ges_tel_12meses≤3

- **V44**: Good
  cant_ges_dom_12meses≤1 and (canal_vta==Artefacta | canal_vta==Oferton) and cant_ges_tel_9mese
- **V45**: Bad
  cant_ges_dom_12meses>2 and cant_ges_efe_tel_6meses>0
- **V46**: Good
  cant_ges_efe_dom_12meses≤1
- **V47**: Bad
  cant_ges_efe_dom_12meses>1
- **V48**: Good
  cant_ges_dom_3meses≤1 and Cadena==BARATODO
- **V49**: Good
  cant_ges_efe_dom_3meses≤1 and cant_ges_efe_tel_12meses≤1
- **V50**: Bad
  cant_ges_efe_dom_3meses>2
- **V51**: Good
  cant_ges_dom_9meses≤1 and cant_ges_efe_tel_9meses≤2
- **V52**: Bad
  cant_ges_dom_9meses>2
- **V53**: Good
  cant_ges_efe_dom_9meses≤1
  and cant_ges_tel_1mes≤0
  and (desc_mejor_resp_tel_12meses==NULL |
  desc_mejor_resp_tel_12meses==CONTACTO SIN COMPROMISO |
  desc_mejor_resp_tel_12meses==COMPROMISO DE PAGO FAMILIAR)
- **V54**: Bad
  cant_ges_dom_6meses>2
- **V55**: Bad
  (desc_mejor_resp_dom_9meses==MENSAJE A TERCEROS |
  desc_mejor_resp_dom_9meses==COMPROMISO DE PAGO |
  desc_mejor_resp_dom_9meses==COMPROMISO DE PAGO FAMILIAR |
  desc_mejor_resp_dom_9meses==NO UBICADOS (380) |
  desc_mejor_resp_dom_9meses==CONTACTO SIN COMPROMISO |
  desc_mejor_resp_dom_9meses==PROMOCIONES |
  desc_mejor_resp_dom_9meses==RECOJO |
  desc_mejor_resp_dom_9meses==CLIENTE SIN EMPLEO |
  desc_mejor_resp_dom_9meses==SERVICIO TECNICO)

and (desc_mejor_resp_tel_3meses==NULL |
desc_mejor_resp_tel_3meses==NO UBICADOS |
desc_mejor_resp_tel_3meses==RECOJO |
desc_mejor_resp_tel_3meses==SERVICIO TECNICO)
and (desc_mejor_resp_dom_3meses==MENSAJE A TERCEROS |
desc_mejor_resp_dom_3meses==COMPROMISO DE PAGO |
desc_mejor_resp_dom_3meses==COMPROMISO DE PAGO FAMILIAR |
desc_mejor_resp_dom_3meses==CONTACTO SIN COMPROMISO |
NO UBICADOS (380) | desc_mejor_resp_dom_3meses==RECOJO |
desc_mejor_resp_dom_3meses==CLIENTE SIN EMPLEO)

5.2.3 31 - 90 Segment

- **V1**: Good
  Atraso_Max_3meses≤11
- **V2**: Good
  Atraso_Max_3meses(11;27] and Saldo_cuota_credito>297.19
- **V3**: Bad
  Atraso_Max_3meses>42
- **V4**: Good
  Atraso_Max_6meses≤29 and cp_pl(0.32;0.84]
- **V5**: Good
  Atraso_Max_6meses(44;55] and Rango_mora_max_mesant==031-060 Dias and cp_pl≤0.93
- **V6**: Bad
  Atraso_Max_6meses>55
- **V7**: Good
  Atraso_Max_9meses≤29 and ctr_pl(0.43;0.77]
- **V8**: Good
  Atraso_Max_9meses(29;45]
- **V9**: Bad
  Atraso_Max_9meses(45;55] and Pago_efec_1mes≤0
- **V10**: Bad
  Atraso_Max_9meses>55
- **V11**: Good
  Atraso_Max_12meses≤29 and Cuotas_pagad_credito(3;11]
- **V12**: Good
  Atraso_Max_12meses(29;45]
- **V13**: Bad
  Atraso_Max_12meses>45
- **V14**: Bad
  Num_pag_3meses≤0
- **V15**: Bad
  Num_pag_3meses(0;1] and Cuotas_pendt_credito(0;2] and Val_pag_2meses≤0
- **V16**: Good
  Num_pag_3meses(1;2] and Cuotas_pendt_credito(0;1]

- **V17**: Bad
  Num_pag_3meses(1;2] and Cuotas_pendt_credito>1
- **V18**: Good
  Num_pag_3meses(2;3] and Atraso_Max_Credito≤79
- **V19**: Bad
  Val_pag_3meses≤0
- **V20**: Bad
  Val_pag_3meses(0;79.74] and cp_ctr(0.7;0.97]
- **V21**: Bad
  Val_pag_3meses(79.74;102.04]
- **V22**: Good
  Val_pag_3meses>102.04
- **V23**: Good
  Atraso_Prom_3meses≤5
- **V24**: Good
  Atraso_Prom_3meses(5;17] and cant_ges_tel_3meses>0
- **V25**: Good
  Atraso_Prom_3meses(17;21.67]
- **V26**: Bad
  Atraso_Prom_3meses(21.67;49.67]
- **V27**: Bad
  Atraso_Prom_3meses>49.67 and Num_atra_may30dias_anio≤2
- **V28**: Bad
  Num_pag_6meses(2;3] and Val_pag_1meses≤0
- **V29**: Bad
  Num_pag_6meses(3;4] and dif_mes>6 and Saldo_vencido_Credito>0
- **V30**: Good
  Num_pag_6meses>4
- **V31**: Good
  Atraso_Prom_6meses≤13.8
- **V32**: Bad
  Atraso_Prom_6meses>13.8
- **V33**: Good
  Atraso_Prom_9meses≤14.11

- **V34**: Bad
  Atraso_Prom_9meses>14.11
- **V35**: Good
  Atraso_Prom_12meses≤9.89
- **V36**: Bad
  Atraso_Prom_12meses(12.17;21.25] and Num_pag_9meses≤6
- **V37**: Bad
  Atraso_Prom_12meses>21.25
- **V38**: Good
  Num_atra_may60dias_anio≤0 and Saldo_vencido_Credito≤108.23
- **V39**: Bad
  Num_atra_may_60dias_anio≤0 and Saldo_vencido_Credito>108.23
- **V40**: Bad
  Num_atra_may_60dias_anio>0
- **V41**: Bad
  Num_pag_12meses≤3
- **V42**: Good
  Num_pag_12meses(3;5] | Num_pag_12meses>10
- **V43**: Good
  Num_pag_12meses(5;8] and cant_ges_dom_1mes>0
- **V44**: Bad
  Num_pag_12meses(8;9] and ind_ges_preventiva==0 and incialbono≤30
- **V45**: Bad
  Num_pag_12meses(9,10] and ind_ges_preventiva==0 and cant_con_efe_tel_3meses≤0
- **V46**: Good
  Atraso_Prom_Credito(6;14] and cant_ges_efe_tel_3meses>0
- **V47**: Bad
  Atraso_Prom_Credito(14;24]
  and (desc_mejor_resp_tel_3meses==NULL |
  desc_mejor_resp_tel_3meses==NO UBICADOS |
  desc_mejor_resp_tel_3meses==RECOJO |
  desc_mejor_resp_tel_3meses==SERVICIO TECNICO |
  desc_mejor_resp_tel_3meses==MENSAJE A TERCERO)

- **V48**: Good
  Atraso_Prom_Credito(14;24]
  and (desc_mejor_resp_tel_3meses==CONTACTO SIN COMPROMISO |
  desc_mejor_resp_tel_3meses==COMPROMISO DE PAGO |
  desc_mejor_resp_tel_3meses==COMPROMISO DE PAGO FAMILIAR)

- **V49**: Bad
  Atraso_Prom_Credito>24

- **V50**: Bad
  Plazo$\leq$16 and cant_ges_tel_1mes$\leq$3 and cant_ges_efe_tel_6meses$\leq$0

- **V51**: Bad
  Rango_mora_mesact==061-090 Dias
  and cant_ges_dom_12meses$\leq$8
  and cant_ges_dom_3meses(0;5]

- **V52**: Bad
  Rango_mora_mesact==061-090 Dias
  and cant_ges_dom_12meses(8;20]
  and cant_ges_dom_3meses>0

- **V53**: Bad
  Rango_mora_mesact==031-060 Dias
  and (desc_mejor_resp_tel_6meses==NULL |
  desc_mejor_resp_tel_6meses==MENSAJE A TERCERO |
  desc_mejor_resp_tel_6meses==NO UBICADOS |
  desc_mejor_resp_tel_6meses==CLIENTE SIN EMPLEO |
  desc_mejor_resp_tel_6meses==RECOJO |
  desc_mejor_resp_tel_6meses==SERVICIO TECNICO |
  desc_mejor_resp_tel_6meses==CLTE FALLECIDO)
  and cant_con_efe_dom_12meses$\leq$0

- **V54**: Good
  Rango_mora_mesact==031-060 Dias
  and (desc_mejor_resp_tel_6meses==CONTACTO SIN COMPROMISO |
  desc_mejor_resp_tel_6meses==COMPROMISO DE PAGO |
  desc_mejor_resp_tel_6meses==COMPROMISO DE PAGO FAMILIAR)
  and tipoinicialbono==NULL

- **V55**: Bad
  (ult_resp_ges_tel==NULL |
  desc_mejor_resp_tel_9meses==NO UBICADOS |
  desc_mejor_resp_tel_9meses==CLIENTE SIN EMPLEO |
  desc_mejor_resp_tel_9meses==CLTE FALLECIDO |

desc_mejor_resp_tel_9meses==SERVICIO TECNICO |
desc_mejor_resp_tel_9meses==RECOJO)
and (region==GUAYAQUIL | region==QUITO)
and cant_ges_efe_dom_mesant≤3

- **V56**: Good
(ult_resp_ges_tel==CONTACTO SIN COMPROMISO |
desc_mejor_resp_tel_9meses==COMPROMISO DE PAGO |
desc_mejor_resp_tel_9meses==COMPROMISO DE PAGO FAMILIAR |
desc_mejor_resp_tel_9meses==PROMOCIONES)
and CapitalInteres(1043.52;2297.91]

- **V57**: Bad
(Linea==VIDEO | Linea==AUDIO | Linea==CONSTRUCCION)
and cant_con_efe_tel_6meses≤0
and TotFacturaInicial≤2046.82

- **V58**: Bad
(Linea==BLANCA NACIONAL |
Linea==BLANCA IMPORTADA |
Linea==ELECTRODOMESTICO)
and cant_con_efe_tel_6meses≤0

- **V59**: Good
(Linea==BLANCA NACIONAL |
Linea==BLANCA IMPORTADA |
Linea==ELECTRODOMESTICO)
and cant_con_efe_tel_6meses>0

- **V60**: Bad
(Linea==COMUNICACIONES | Linea==FERRETERIA | Linea==NULL)
and Inicial≤0

- **V61**: Bad
(Linea==COMUNICACIONES | Linea==FERRETERIA | Linea==NULL)
and Inicial>0
and cant_con_efe_tel_12meses≤5

- **V62**: Bad
(desc_mejor_resp_dom_12meses==MENSAJE A TERCEROS |
desc_mejor_resp_dom_12meses==NULL |
desc_mejor_resp_dom_12meses==CONTACTO SIN COMPROMISO |
desc_mejor_resp_dom_12meses==CLIENTE SIN EMPLEO)
and cant_ges_efe_dom_3meses(3;6]

- **V63**: Bad
  (desc_mejor_resp_dom_12meses==COMPROMISO DE PAGO |
  desc_mejor_resp_dom_12meses==COMPROMISO DE PAGO FAMILIAR)
  and cant_ges_efe_dom_3meses(2;10]
  and cant_con_efe_tel_12meses$\leq$1

- **V64**: Good
  (desc_mejor_resp_dom_12meses==COMPROMISO DE PAGO |
  desc_mejor_resp_dom_12meses==COMPROMISO DE PAGO FAMILIAR)
  and cant_ges_efe_dom_3meses(2;10]
  and cant_con_efe_tel_12meses>1

- **V65**: Good
  (desc_mejor_resp_dom_3meses==MENSAJE A TERCEROS |
  desc_mejor_resp_dom_3meses==CONTACTO SIN COMPROMISO)
  and cant_ges_tel_6meses(0;7]

- **V66**: Bad
  (desc_mejor_resp_dom_3meses==MENSAJE A TERCEROS |
  desc_mejor_resp_dom_3meses==CONTACTO SIN COMPROMISO)
  and cant_ges_tel_6meses>10

- **V67**: Bad
  (desc_mejor_resp_dom_3meses==COMPROMISO DE PAGO |
  desc_mejor_resp_dom_3meses==COMPROMISO DE PAGO FAMILIAR)
  and cant_con_efe_tel_9meses$\leq$1

- **V68**: Good
  (desc_mejor_resp_dom_3meses==COMPROMISO DE PAGO |
  desc_mejor_resp_dom_3meses==COMPROMISO DE PAGO FAMILIAR)
  and cant_con_efe_tel_9meses>1

- **V69**: Good
  (desc_mejor_resp_dom_9meses==MENSAJE A TERCEROS |
  desc_mejor_resp_dom_9meses==NULL |
  desc_mejor_resp_dom_9meses==CONTACTO SIN COMPROMISO |
  desc_mejor_resp_dom_9meses==CLIENTE SIN EMPLEO)
  and cant_ges_tel_9meses(0;7]

- **V70**: Bad
  (desc_mejor_resp_dom_9meses==MENSAJE A TERCEROS |
  desc_mejor_resp_dom_9meses==NULL |
  desc_mejor_resp_dom_9meses==CONTACTO SIN COMPROMISO |
  desc_mejor_resp_dom_9meses==CLIENTE SIN EMPLEO)
  and cant_ges_tel_9meses(7;14]

- **V71**: Good
  (desc_mejor_resp_dom_9meses==COMPROMISO DE PAGO |
  desc_mejor_resp_dom_9meses==COMPROMISO DE PAGO FAMILIAR)
  and ValorCuota>62.6

- **V72**: Bad
  (desc_mejor_resp_dom_6meses==MENSAJE A TERCEROS |
  desc_mejor_resp_dom_6meses==COMPROMISO DE PAGO FAMILIAR |
  desc_mejor_resp_dom_6meses==CONTACTO SIN COMPROMISO |
  desc_mejor_resp_dom_6meses==SERVICIO TECNICO |
  desc_mejor_resp_dom_6meses==CLIENTE SIN EMPLEO)
  and cant_ges_tel_12meses>5

- **V73**: Bad
  (desc_mejor_resp_dom_6meses==COMPROMISO DE PAGO |
  desc_mejor_resp_dom_6meses==NULL)
  and cant_ges_efe_tel_9meses$\leq$0

- **V74**: Good
  (desc_mejor_resp_dom_6meses==COMPROMISO DE PAGO |
  desc_mejor_resp_dom_6meses==NULL)
  and cant_ges_efe_tel_9meses>0 and Edad>29

- **V75**: Good
  desc_mejor_resp_dom_6meses==CANCELADO |
  desc_mejor_resp_dom_6meses==PROMOCIONES

- **V76**: Bad
  cant_con_efe_dom_9meses$\leq$0
  and cant_ges_dom_6meses$\leq$9
  and cant_ges_efe_tel_12meses$\leq$0

- **V77**: Good
  cant_con_efe_dom_9meses(0;4]
  and (CuotasGratis==DESCUENTO EN CUOTAS | CuotasGratis==NULL)

- **V78**: Bad
  cant_con_efe_dom_9meses(0;4]
  and (CuotasGratis==BONO INICIAL + N CUOTAS GRATIS | CuotasGratis==CUOTAS
  GRATIS)

- **V79**: Bad
  cant_con_efe_dom_6meses$\leq$0
  and cant_ges_efe_tel_mesant$\leq$0
  and IngresosPropios$\leq$353

- **V80**: Good
  cant_con_efe_dom_6meses>0
- **V81**: Bad
  cant_con_efe_tel_mesant≤0 and Cadena==ARTEFACTA and Sexo==M
- **V82**: Good
  cant_con_efe_tel_mesant>0 and ID_Num_Telef_Particular1==NO
- **V83**: Bad
  cant_ges_efe_dom_12meses≤3 and TasaCredito≤15 and cant_con_efe_dom_3meses≤0
- **V84**: Good
  cant_ges_efe_dom_12meses(3;20] and cant_ges_efe_dom_6meses(2;9]
  and cant_ges_efe_dom_12meses>9
- **V85**: Bad
  cant_ges_efe_dom_12meses(3;20] and cant_ges_efe_dom_6meses>9
- **V86**: Bad
  (canal_vta==Artefacta |
  canal_vta==Baratodo |
  canal_vta==Credito a |
  canal_vta==Oferton)
  and (cant_ges_efe_dom_9meses≤2 |
  cant_ges_efe_dom_9meses(8;13])
- **V87**: Good
  (canal_vta==Artefacta |
  canal_vta==Baratodo |
  canal_vta==Credito a |
  canal_vta==Oferton)
  and cant_ges_efe_dom_9meses(2;8]
  and RelacionTrabajo==SI
- **V88**: Good
  (canal_vta==Artefacta |
  canal_vta==Baratodo |
  canal_vta==Credito a |
  canal_vta==Oferton)
  and cant_ges_efe_dom_9meses>13 and MesesGracia≤30
- **V89**: Bad
  cant_ges_dom_9meses≤1 | cant_ges_dom_9meses(9;13]
- **V90**: Good
  cant_con_efe_dom_mesant>0

5.2.4 All Segments

- **V1**: Good
  Cuotas_pendt_credito≤0 and Atraso_Prom_Credito≤8
- **V2**: Bad
  Cuotas_pendt_credito>0
- **V3**: Good
  Saldo_vencido_Credito≤0 and Atraso_Max_Credito≤0 and Pago_efec_1mes(57.63;124.63]
- **V4**: Good
  Saldo_vencido_Credito≤0 and Atraso_Max_Credito(0;29]
- **V5**: Good
  Atraso_Prom_3meses≤0 and cant_ges_tel_6meses≤0 and cant_ges_dom_3meses≤0
- **V6**: Good
  Atraso_Max_3meses≤0 and cant_ges_tel_3meses≤0
  and (Rango_mora_mesact==000-Al Dia | Rango_mora_mesact==031-060 Dias)
- **V7**: Bad
  Atraso_Max_3meses>18
- **V8**: Good
  Atraso_Max_6meses≤0
  and (desc_mejor_resp_tel_3meses==NULL |
  desc_mejor_resp_tel_3meses==PROMOCIONES)
  and cant_ges_efe_dom_3meses≤0
- **V9**: Bad
  Atraso_Max_6meses>20
- **V10**: Good
  Atraso_Prom_9meses≤0
  and (desc_mejor_resp_tel_6meses==NULL |
  desc_mejor_resp_tel_6meses==COMPROMISO DE PAGO FAMILIAR |
  desc_mejor_resp_tel_6meses==PROMOCIONES)
- **V11**: Good
  Atraso_Prom_6meses≤0
  and cant_ges_efe_tel_6meses≤0
  and cant_ges_dom_6meses≤0
- **V12**: Good
  Atraso_Prom_12meses≤0
  and cant_ges_efe_tel_3meses≤0

and (desc_mejor_resp_dom_3meses==NULL |
desc_mejor_resp_dom_3meses==SERVICIO TECNICO |
desc_mejor_resp_dom_3meses==PROMOCIONES |
desc_mejor_resp_dom_3meses==RECOJO |
desc_mejor_resp_dom_3meses==CLTE FALLECIDO)

- **V13**: Bad
  Atraso_Prom_12meses>3.83
- **V14**: Good
  Atraso_Max_9meses≤0
  and cant_ges_tel_1mes≤0
  and cant_ges_efe_dom_6meses≤0
- **V15**: Bad
  Atraso_Max_9meses>20
- **V16**: Good
  Atraso_Max_12meses≤0 and cant_ges_tel_9meses≤0 and cant_ges_dom_1mes≤0
- **V17**: Good
  Atraso_Max_12meses≤0 and cant_ges_tel_9meses>0
- **V18**: Bad
  Atraso_Max_12meses>21
- **V19**: Good
  Num_atra_may30dias_anio≤0 and cp_ctr(0.74;0.87] and cant_ges_dom_9meses≤0
- **V20**: Good
  Num_atra_may30dias_anio≤0 and (cp_ctr(0.87;0.95] | cp_ctr>1)
- **V21**: Good
  Num_atra_may30dias_anio≤0 and cp_ctr(0.95;1] and cant_ges_dom_9meses≤0
- **V22**: Bad
  Num_atra_may30dias_anio>0
- **V23**: Bad
  Rango_mora_max_mesant==031-060 Dias
- **V24**: Good
  Rango_mora_max_mesant==000-Al Dia
  and (desc_mejor_resp_dom_6meses==NULL |
  desc_mejor_resp_dom_6meses==SERVICIO TECNICO | desc_mejor_resp_dom_6meses==PROMOC
  desc_mejor_resp_dom_6meses==RECOJO |
  desc_mejor_resp_dom_6meses==CLIENTE SIN EMPLEO) and Num_pag_3meses(2;3]

- **V25**: Good
  Rango_mora_max_mesant==001-030 Dias
  and Num_pag_3meses(2;3] and cant_ges_efe_tel_9meses≤7

- **V26**: Good
  cant_ges_efe_dom_9meses≤0
  and (Val_pag_2meses(35.95;119.57] | Val_pag_2meses>222.26)

- **V27**: Good
  cant_ges_efe_dom_9meses≤0 and Val_pag_2meses(119.57;222.26]
  and (desc_mejor_resp_tel_9meses==NULL |
  desc_mejor_resp_tel_9meses==MENSAJE A TERCERO |
  desc_mejor_resp_tel_9meses==MENSAJE A TERCEROS |
  desc_mejor_resp_tel_9meses==COMPROMISO DE PAGO FAMILIAR |
  desc_mejor_resp_tel_9meses==RECOJO)

- **V28**: Bad
  cant_ges_efe_dom_9meses>1

- **V29**: Good
  cant_ges_dom_12meses≤0 and cant_ges_tel_12meses≤0 and Val_pag_1meses>0

- **V30**: Bad
  cant_ges_dom_12meses>1

- **V31**: Good
  cant_ges_efe_dom_12meses≤0
  and (ult_resp_ges_tel==NULL |
  ult_resp_ges_tel==CANCELADO |
  ult_resp_ges_tel==RECOJO)
  and Val_pag_3meses>70.88

- **V32**: Bad
  cant_ges_efe_dom_12meses>1

- **V33**: Bad
  cant_ges_efe_dom_mesant≤0
  and desc_mejor_resp_dom_9meses==MENSAJE A TERCEROS

- **V34**: Good
  cant_ges_efe_dom_mesant≤0
  and desc_mejor_resp_dom_9meses==NULL
  and (desc_mejor_resp_tel_12meses==NULL | (desc_mejor_resp_tel_12meses==RECOJO)

- **V35**: Bad
  cant_ges_efe_dom_mesant>0

- **V36**: Good
  cant_ges_efe_dom_mesant≤0
  and cant_ges_efe_tel_12meses≤0
  and (desc_mejor_resp_dom_12meses==NULL |
  desc_mejor_resp_dom_12meses==PROMOCIONES |
  desc_mejor_resp_dom_12meses==SERVICIO TECNICO)

- **V37**: Good
  Num_atra_may_60dias_anio≤0 and cant_ges_efe_tel_mesant≤0 and Num_pag_6meses>3

- **V38**: Good
  cant_con_efe_tel_9meses≤0 and (Num_pag_9meses(3;8] | Num_pag_9meses>9)

- **V39**: Good
  cant_con_efe_tel_9meses≤0 and Num_pag_9meses(8;9]
  and cant_con_efe_dom_12meses≤0

- **V40**: Good
  cant_con_efe_tel_6meses≤0 and cant_con_efe_dom_6meses≤0 and Num_pag_12meses>3

- **V41**: Good
  cant_con_efe_tel_12meses≤0 and cant_con_efe_dom_9meses≤0
  and Cuotas_pagad_credito>3

- **V42**: Bad
  cant_con_efe_tel_12meses>0

- **V43**: Good
  cant_con_efe_tel_3meses≤0 and cant_con_efe_dom_3meses≤0 and cp_pl>0.19

- **V44**: Good
  cant_con_efe_tel_mesant≤0
  and (Saldo_cuota_credito≤140.03 | Saldo_cuota_credito(602.09;993.64])

- **V45**: Good
  cant_con_efe_tel_mesant≤0 and Saldo_cuota_credito(140.03;347.66] and ctr_pl≤0.83

- **V46**: Good
  cant_con_efe_tel_mesant≤0 and Saldo_cuota_credito>993.64
  and CapitalInteres>1540.4

- **V47**: Good
  ind_ges_preventiva==0 and Inicial≤0 and Edad>42

- **V48**: Good
  ind_ges_preventiva==1
  and (Linea==DEL HOGAR | Linea==BLANCA NACIONAL |
  Linea==ELECTRODOMESTICO | Linea==BLANCA IMPORTADA | Linea==MUEBLES |

Linea==FERRETERIA |
Linea==SERVICIOS | Linea==VARIOS |
Linea==VIDEO | Linea==TRANSPORTE)

- **V49**: Good
incialbono≤0 and ValorCuota>66.89
- **V50**: Good
tipoinicialbono==NULL and (IngresosPropios(353;366] | IngresosPropios>420)
- **V51**: Bad
Plazo(12;16]
- **V52**: Good
Plazo(16;18] and Cadena==ARTEFACTA
- **V53**: Bad
TotFacturaInicial(493.71;920.99] and RelacionTrabajo==NO
- **V54**: Good
TotFacturaInicial>1416.6
- **V55**: Bad
(region==REGIONAL 3 | region==QUITO | region==REGIONAL 2) and CuotasGratis==BONO INICIAL + N CUOTAS GRATIS
- **V56**: Bad
(region==REGIONAL 5 | region==GUAYAQUIL)
and (linea==Tienda | linea==Satelite | linea==Recojo | linea==Terceros | linea==Televent)
and Sexo==M
- **V57**: Good
(region==REGIONAL 7 | region==REGIONAL 6) and ID_Num_Telef_Particular1==NO
- **V58**: Good
TasaCredito≤15.1 and dif_mes(7;12]
and (canal_vta==Artefacta | canal_vta==Oferton |
canal_vta==Tropimotors | canal_vta==AKT)
- **V59**: Good
TasaCredito>15.1 and Cant_Productos>1
- **V60**: Bad
Sexo==M and MesesGracia≤30 and Cant_Num_Telef_Referen≤1
- **V61**: Good
Sexo==F and Cant_Num_Telef_Referen≤1 and MesesGracia≤30

Bibliography

[1] D. B. Lawrence and A. Solomon, "Managing a consumer lending business," *(No Title)*, 2002.

[2] D. O. Vargas Lara, "Metodología para la obtención de un modelo de cobranza de créditos masivos. desarrollo y obtención de un modelo de score." Master's thesis, Quito, 2016., 2015.

[3] J. A. Suquillo Llumiquinga, "Credit scoring: aplicando técnicas de regresión logística y modelos aditivos generalizados para una cartera de crédito en una entidad financiera." B.S. thesis, Quito, 2021, 2021.

[4] Y. S. Sanchez Farfan, "Aplicación del modelo credit scoring y regresión logística en la predicción del crédito, en una entidad financiera de la ciudad del cusco 2022," 2023.

[5] Anǿnimo, "Credit scoring using scorecardpy with xgboost," 2024. [Online]. Available: https://datascience.stackexchange.com/questions/38817/credit-scoring-using-scorecardpy-with-xgboost

[6] R. Loffredo, "Building a predictive credit risk analysis model using xgboost," 2023. [Online]. Available: https://medium.com/@loffredo.ds/building-a-predictive-credit-risk-analysis-model-using-xgboost

[7] N. del Autor, "Comparing predictive models at the feature level," *FICO Community Blog*, 2024. [Online]. Available: https://community.fico.com/s/blog-post/a5Q80000000DsL6EAK/fico1331

[8] L. Thomas, J. Crook, and D. Edelman, *Credit scoring and its applications*. SIAM, 2017.

[9] N. Cifuentes Baquero and L. Gutiérrez Murcia, "Modelo predictivo de la probabilidad de aumento de los días de mora para usuarios de tarjeta de crédito," 2022.

[10] T. B. Arnold and J. W. Emerson, "Nonparametric goodness-of-fit tests for discrete null distributions." *R Journal*, vol. 3, no. 2, 2011.

[11] F. J. Massey Jr, "The kolmogorov-smirnov test for goodness of fit," *Journal of the American statistical Association*, vol. 46, no. 253, pp. 68–78, 1951.

[12] J. L. C. Reche, "Regresión logística. tratamiento computacional con r," *Universidad de Granada*, 2013.

[13] G. S. Maddala, J. Contreras García, V. Lozano López, A. García Ferrer *et al.*, "Econometría," 1985.

[14] C. Iñiguez and M. Morales, "Selección de perfiles de clientes mediante regresión logística para muestras desproporcionadas, validación, monitoreo y aplicación en la proyección de provisiones," *Escuela Politécnica Nacional, Ecuador*, 2009.

[15] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[16] F. Chollet, "Deep learning with r/françois chollet; with jj allaire," *Deep learn. R*, 2018.

[17] Z. Li, "Extracting spatial effects from machine learning model using local interpretation method: An example of shap and xgboost," *Computers, Environment and Urban Systems*, vol. 96, p. 101845, 2022.

[18] C. Molnar, "Interpretable machine learning," 2021. [Online]. Available: https://fedefliguer.github.io/AAI/redes-neuronales.html

[19] E. Bartz, T. Bartz-Beielstein, M. Zaefferer, and O. Mersmann, *Hyperparameter Tuning for Machine and Deep Learning with R: A Practical Guide*. Springer Nature, 2023.

[20] M. S. Jácome Jara, "Construcción de un modelo estadístico para calcular el riesgo de deterioro de una cartera de microcréditos y propuesta de un sistema de gestión para la recuperación de la cartera en una empresa de cobranzas," B.S. thesis, Quito: EPN, 2014, 2014.

[21] A. E. Pérez Tatamués, "Modelo de activación de tarjetas de crédito en el mercado crediticio ecuatoriano a través de una metodología analítica y automatizada en r," B.S. thesis, Quito, 2014., 2014.

[22] J. A. Capelo Vinza, "Modelo de aprobación de tarjetas de crédito en la población ecuatoriana bancarizada a través de una metodología analítica," B.S. thesis, Quito, 2012., 2012.

[23] N. Bussmann, P. Giudici, D. Marinelli, and J. Papenbrock, "Explainable machine learning in credit risk management," *Computational Economics*, vol. 57, no. 1, pp. 203–216, 2021.

[24] J. Galindo and P. Tamayo, "Credit risk assessment using statistical and machine learning: basic methodology and risk modeling applications," *Computational economics*, vol. 15, pp. 107–143, 2000.

[25] M. P. Deisenroth, A. A. Faisal, and C. S. Ong, *Mathematics for machine learning*. Cambridge University Press, 2020.

[26] W. Ertel, *Introduction to artificial intelligence*. Springer, 2018.

[27] R. Hernández, C. Fernández, P. Baptista *et al.*, *Metodología de la investigación*. méxico: mcGraw-Hill, 2014, vol. 6.

[28] A. L. Támara-Ayús, H. Vargas-Ramírez, J. J. Cuartas, and I. E. Chica-Arrieta, "Regresión logística y redes neuronales como herramientas para realizar un modelo scoring," *Revista Lasallista de Investigación*, vol. 16, no. 1, pp. 187–200, 2019.