Research practice I

Final report

Supervised Statistical Methods to Identify Credit Acceptance Rate

Valentina Yusty[A] Henry Laniado[B] and Marcela Gutíerrez Mejía[C]
vyustym@eafit.edu.co, hlaniado@eafit.edu.co,marcelagutierrez@celerix.com

[A]Mathematical Engineering, Universidad EAFIT
[B]Mathematical Science Department, School of Sciences, Universidad EAFIT
[C]Analitycal Deparment, Celerix S.A.S
Mathematical Modeling Research Group, Universidad EAFIT

June 9, 2020

Abstract

Since incorrect decisions can have detrimental effects on financial institutions, the possibility for these to forecast business failures becomes indispensable. In the financial domain, the focus of research problems rarely revolves around the identification of the clients who desist their credit offering, but rather on bankruptcy prediction and credit scoring. The general objective of this paper revolves around the implementation of supervised machine learning algorithms that will allow CrediOrbe, a credit company, to target customers whose profile assimilates those who desist their credit offering. Machine learning algorithms have been greatly studied as tools to aid decisions makers in the realm of finance. Performance measurements are calculated and analyzed through the use of statistical classification measurements. Suggestions for further research are provided.

Keywords: Machine learning, credit scoring , financial institutions, statistical classification measurements.

1 Introduction

According to Thomas et al. (2002) , in the field of banking and finance, credit scoring has been considered a useful application of operations research modeling. Credit scoring practices have allowed financial institutions to witness an increase in their capital by mitigating risks associated with customer credit. These practices are detrimental for these institutions since it allows them to have the ability to identify what types of credit requests fall under the category of clients that are likely to repay their debt and those who are not. These evaluation is based on a credit score associated with each client.

As mentioned in Vojtek and Kocenda (2006), a variety of factors are considered when calculating the credit score of each applicant: age, income, expenses, credit score, gender, last level of education completed, among others. The higher the score, the lower the risk associated with the applicant

and vice versa. Therefore, credit scoring is a tool used to mitigate the risk that financial institutions face when granting a credit to their customer.

For financial institutions, corporate failure predictions is a tool used to mitigate the risk of credit loan applications. For example, as examined in Gouvêa and Gonçalves (2007), these predictions forecast the possibility of a financial loss when a customer does not honor the credit contract. Hence, corporate failure predictions enable financial entities to implement proper lending practices by sustaining profitability and preventing adverse effects such as bankruptcy and economic repercussions. It is important to note that due to their promising prediction accuracy, data mining approaches are used to develop corporate failure prediction models. As stated in De Veaux (2003), the goal of data mining is to find underlying relationships between variables from a given data set. To further expand, a component of machine learning is data mining. As noted in De Veaux (2003), machine learning allows computer algorithms to automatically improve from experience by using statistics. Over the past century, many machine learning algorithms have been developed because of their performance in evaluating predictive modeling problems.

This research practice will report on the use of use supervised machine learning algorithms to identify clients who desist their credit offering in the motorcycle financing company CrediOrbe. There are two main motivations for carrying out this research practice. First, a successful implementation of supervised machine learning algorithms will allow the company to place stronger retention strategies on customers who are likely to desist the credit offering,as indicated by the algorithm. Second, by implementing market segmentation practices, especially in the final stage of the credit application process, large amounts of operational expenses could be saved.

Three main outcomes are presented in this research practice: the description and implementation of the three supervised machine learning algorithms, an statistical analysis and performance evaluation of the executed algorithms, and suggestions for future projects and improvements.

This research practice will begin by presenting the research problem in Section 2, where the problem is described, the main mathematical elements are listed, and some potential applications are explored. This section will be preceded by Section 3, where there is a description of the Support Vector Machine (SVM), Random Forest(RF) and Logistic Regression (LR) algorithms. The remainder of the research practice is divided into three sections. Section 4 refers to the CRISP-DM methodology, which served as a guideline to the development of this project. Section 5 will focus specifically on displaying and analyzing the performance results. And finally in Section 6 conclusions are discussed and suggestions for future work are presented.

2 Problem definition

2.1 Problem Description

In this research practice, machine learning is utilized to solve one of the main problems faced by the motorcycle financing company CrediOrbe. In short, CrediOrbe is a company with extensive experience in the motorcycle financing market. Its customers are those whose credit requests are usually rejected by traditional credit companies due to their credit risk. The purpose of this research practice is to use supervised machine learning algorithms to identify clients who desist their credit offering. Hence, the population to be considered is conformed by individuals whose credit score have exceeded the credit score threshold specified by the company.

2.2 Main Mathematical Elements

The main mathematical elements of the research practice can be listed as follows:

- Machine Learning was used to implement the SVM, RF, LR algorithms.
- Statistics was used to understand, evaluate and interpret the data and the performance of the algorithms.
- Probability was used to make classification decisions by the algorithms and to evaluate their performance.

2.3 Potential Applications

There are two potential applications associated to the previous problem.

- Reduction of operational cost and in turn an increase of company's revue.
- Employment of marketing strategies.

It is important to note that marketing strategies (such as market segmentation) targeting groups of individuals who desist the credit offering, will allow the company to prevent monetary losses. Since, as a whole, the evaluation of a credit application symbolizes a significant source of operational and labor costs to the company.

3 State of the art

3.1 Supervised Learning Algorithms

The algorithms implemented in this research practice fall under the category of supervised machine learning algorithms. As defined in Cristianini and Shawe-Taylor (2000), supervised machine learning is an approach in which a computer is used to understand and classify data by predicting output variables from input variables . The supervised learning algorithms that are going to be implemented are SVM, RF and LR. To assess how these perform, measurements such as the number of instances that are correctly and incorrectly classified, precision, F-Measure, accuracy and recall were calculated.

3.2 SVM

According to Cristianini and Shawe-Taylor (2000), SVM is a discriminative classifier whose classification is based on the construction, in a high or infinite dimensional space, of a hyperplane or a set of hyperplanes. By definition, the hyperplane is used as a decision boundary in which each input vector from the input space is classified. This algorithm can be used for a variety of purposes: classification, regression and outlier detection.

3.2.1 Linear Classification

According to Cristianini and Shawe-Taylor (2000), a binary classification problem involves the use of a classification decision rule to classify elements into two distinct groups. For notation purposes let $X \in \mathbb{R}^n$, denote the input space and $Y = \{-1, 1\}$ denote the output domain. Let an example from the data set be represented as the pair $(\boldsymbol{x}_i, y_i)$, where $\boldsymbol{x}_i = (x_1, .., x_n)$ represents the input vector and y_i the respective output value. Lastly, let the training data be denoted as $S = ((\boldsymbol{x}_1, y_1)), ..., (\boldsymbol{x}_l, y_l)) \in (X, Y)^l$ and let the testing data be denoted as $S' = ((\boldsymbol{x}_1, y_1), ..., (\boldsymbol{x}_k, y_k)) \in (X, Y)^k$.

Given a testing example $(\boldsymbol{x}_i, y_i) \in S'$, binary classification is performed by using a function $f : X \in \mathbb{R}^n \to \mathbb{R}$, denoted as

$$f(\boldsymbol{x}_i) = \langle \boldsymbol{w}^T \cdot \boldsymbol{x}_i \rangle + b, \tag{1}$$

where $(\boldsymbol{w}, b) \in \mathbb{R}^n * \mathbb{R}$ are parameters that control the decision rule and are learned from a S. Note that if the value returned after evaluating $\boldsymbol{x}_i$ in Eq.(1) is positive, then the testing example is assigned to the positive class and vice versa for the negative class.

Note that by substituting Eq.(12) into Eq.(1), the decision rule can also be evaluated as the inner product between a testing point $\boldsymbol{x}_t$ and the training points in S, denoted as

$$f(\boldsymbol{x}_i) = \sum_{j=1}^{l} \alpha_j y_j \langle \boldsymbol{x}_j \cdot \boldsymbol{x}_i \rangle + b, \tag{2}$$

where $\alpha_j \geq 0$ are the Lagrange multipliers. When dealing with linear classification, functional and geometric margins can be used as a tool to evaluate the performance of the algorithm. The functional margin of an example $(\boldsymbol{x_i}, y_i)$ with respect to the hyperplane $(\boldsymbol{w}, b)$ can be calculated as

$$\hat{\gamma_i} = y_i(\langle \boldsymbol{w}^T \cdot \boldsymbol{x}_i \rangle + b).$$

Note that if the value $\hat{\gamma_i}$ is greater than zero, the classification of $(\boldsymbol{x}_i, y_i)$ is correct.It is important to take into account that given S, we define the geometric margin of $(\boldsymbol{w}, b)$ with respect to S as

$$\hat{\gamma} = \min_{i=1...l} \hat{\gamma_i}.$$

The geometric margin, on the other hand, represents the Euclidean distance of an example with respect to the decision boundary and can be calculated as

$$\gamma_i = y_i \left(\left(\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|} \right)^T \cdot \boldsymbol{x}_i + \left(\frac{b}{\|\boldsymbol{w}\|} \right) \right).$$

It is important to take into account that given S, we define the geometric margin of $(\boldsymbol{w}, b)$ with respect to S as

$$\gamma = \min_{i=1...l} \gamma_i.$$

3.2.2 Kernel-Induced Feature Spaces

In some cases, the linear combination of independent variables cannot effectively classify the response variable. In this case, the idea of more abstract features of the data is considered. According to Cristianini and Shawe-Taylor (2000), by using the kernel representation of the data, linear classification learning machines can better classify data on a high dimensional feature space.

The purpose of kernel representation is to change the way data is represented in order to obtain an objective function that can fully distinguish different classes. The preprocessing method consist on changing the representation of all $\boldsymbol{x}_i \in X \; i = \{1, ..., n\}$:

$$\boldsymbol{x}_i = (x_1, ..., x_n) \rightarrow \boldsymbol{\phi}(\boldsymbol{x}_i) = (\phi_1(\boldsymbol{x}_i), ..., \phi_d(\boldsymbol{x}_i)) \quad d \leq n. \tag{3}$$

The process represented in Eq.(3) is equivalent to mapping the input space X into a feature space represented by $F = \{\boldsymbol{\phi}(\boldsymbol{x}_i) | \boldsymbol{x}_i \in X\}$. Let $\boldsymbol{\phi} : X \rightarrow F$ represent a non-linear map from the input space to a feature space.

A reason why feature mapping occurs is due to the need of machines to classify non-linear relationships. Feature mapping allows non-linearly separable data to become linearly separable by rewriting the data into a new representation. According to Cristianini and Shawe-Taylor (2000), this process consists on using a linear machine to classify the data on the feature space, which was initially applied a non-linear mapping. Given a testing example $(\boldsymbol{x}_i, y_i) \in S'$, the linear machine used to classify the testing example on feature space F is denoted as

$$f(\boldsymbol{x}_i) = \langle \boldsymbol{w}^T \cdot \boldsymbol{\phi}(\boldsymbol{x}_i) \rangle + b, \tag{4}$$

where $(\boldsymbol{w}, b) \in \mathbb{R}^d * \mathbb{R}$ are parameters that control the decision rule in the feature space and are learned from a S.

Due to the fact that linear learning machines can also be expressed in dual representation, Eq.(4) can be rewritten as the inner product between the input vector from a testing example $(\boldsymbol{x}_i, y_i) \in S'$ and the input vector from the training points in S in the feature space F:

$$f(\boldsymbol{x}_i) = \sum_{j=1}^{l} \alpha_j y_j \langle \boldsymbol{\phi}(\boldsymbol{x}_j) \cdot \boldsymbol{\phi}(\boldsymbol{x}_i) \rangle + b. \tag{5}$$

According to Cristianini and Shawe-Taylor (2000), for all pairs of examples in the input space, a kernel function, K, can be represented as:

$$K(\boldsymbol{x}, \boldsymbol{z}) = \langle \boldsymbol{\phi}(\boldsymbol{x}) \cdot \boldsymbol{\phi}(\boldsymbol{z}) \rangle. \tag{6}$$

Hence, by substituting Eq.(6) into Eq.(5), we obtain

$$f(\boldsymbol{x}_i) = \sum_{j=1}^{l} \alpha_j y_j K \langle \boldsymbol{x}_j, \boldsymbol{x}_i \rangle. \tag{7}$$

3.2.3 Support Vector Classification: Maximal Margin Classifier

A frequently used model of Support Vector classification for linearly separable data on the input space is denoted as the maximal margin classifier. According to Cristianini and Shawe-Taylor

(2000), the main idea associated to the maximal margin classifier is to correctly classify data by using a maximal margin hyperplane.

Given a linearly separable training sample S, the optimization problem associated to the maximal margin classifier is

$$\begin{aligned} \max_{\gamma,w,b} \quad & \gamma \\ \text{s.t.} \quad & y_i(\langle \boldsymbol{w}^T \cdot \boldsymbol{x_i} \rangle + b) \geq \gamma \qquad i = 1, ...l \\ & \|\boldsymbol{w}\| = 1. \end{aligned} \tag{8}$$

The objective function of Eq.(8) is to maximize the geometric margin γ. The first constraint indicates that each training example must have functional margin of at least γ. The second constraint, a scale constraint, ensures that the functional margin is equal to the geometric margin. Since Eq.(8) is a non-convex optimization problem, it can be written as a convex-optimization problem with a convex quadratic objective and a linear constraint:

$$\begin{aligned} \min_{\gamma,w,b} \quad & \frac{1}{2} \|\boldsymbol{w}\|^2 \\ \text{s.t.} \quad & y_i(\langle \boldsymbol{w}^T \cdot \boldsymbol{x_i} \rangle + b) \geq 1 \qquad i = 1, ...l. \end{aligned} \tag{9}$$

Equation (9), denoted as primal optimization problem for the maximal margin classifier, tries to maximize the distance between the hyperplanes $< \boldsymbol{w}^T, \boldsymbol{x_i} > + b = 1$ and $< \boldsymbol{w}^T, \boldsymbol{x_i} > + b = -1$, calculated as $\frac{2}{\|\boldsymbol{w}\|}$, and is subject to constraints that ensure the classes are separable.

To convert the primal optimization for the maximal margin classifier into a dual optimization problem, let Eq.(9) be rewritten as

$$\begin{aligned} \min_{\gamma,w,b} \quad & \frac{1}{2} \|\boldsymbol{w}\|^2 \\ \text{s.t.} \quad & -y_i(\langle \boldsymbol{w}^T \cdot \boldsymbol{x_i} \rangle + b) + 1 \leq 0 \qquad i = 1, ...l. \end{aligned} \tag{10}$$

Constructing the Lagrangian of Eq.(10) we obtain

$$\mathcal{L}(\boldsymbol{w}, b, \alpha) = \frac{1}{2} \|\boldsymbol{w}\|^2 - \sum_{l=1}^{l} \alpha_i [y_i(\langle \boldsymbol{w}^T \cdot \boldsymbol{x_i} \rangle + b) - 1]. \tag{11}$$

The Karush-Kush-Tucker conditions associated with Eq.(10) are:

$$\frac{\partial \mathcal{L}(\boldsymbol{w}, b, \alpha)}{\partial \boldsymbol{w}} = \boldsymbol{w} - \sum_{i=1}^{l} \alpha_i y_i \boldsymbol{x_i} = 0 \Longrightarrow \boldsymbol{w} = \sum_{l=1}^{l} \alpha_i y_i \boldsymbol{x_i}, \tag{12}$$

$$\frac{\partial \mathcal{L}(\boldsymbol{w}, b, \alpha)}{\partial b} = \sum_{l=1}^{l} \alpha_i y_i = 0, \tag{13}$$

$$\alpha_i [-y_i(\langle \boldsymbol{w}^T \cdot \boldsymbol{x_i} \rangle + b) + 1] = 0 \quad i = 1, ...l, \tag{14}$$

$$\alpha_i \geq 0, \quad i = 1, ...l.$$

Note that the training examples for which the value of $\alpha_i \neq 0$ are denoted as support vectors. According to Tax and Duin (1999), support vectors are training examples which have the smallest geometrical margin to the hyperplane and greatly affect the direction of the hyperplane.

By substituting Eq.(12) into Eq.(11) and considering the result obtained from Eq.(13) the objective function for the dual function can be obtained:

$$\mathcal{L}(\boldsymbol{w}, b, \alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} y_i y_j \alpha_i \alpha_j \langle \boldsymbol{x}_i^T \cdot \boldsymbol{x}_j \rangle. \tag{15}$$

Taking into account Eq.(15) and the previous constraints, the following dual optimization problem for the maximal margin classifier is proposed:

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} y_i y_j \alpha_i \alpha_j \langle \boldsymbol{x}_i^T \cdot \boldsymbol{x}_j \rangle \\ \text{s.t.} \quad & \alpha_i \geq 0, \quad i = 1, ...l \\ & \sum_{i=1}^{l} \alpha_i y_i = 0. \end{aligned}$$

3.2.4 Support Vector Classification: Soft Margin Optimization

When the data cannot be linearly separated in the input space, Support Vector classification methods, such as soft margin optimization, are used. As mentioned in Cristianini and Shawe-Taylor (2000), unlike the maximal margin classifier, soft margin optimization can tolerate the presence of outliers and noise in the data

Taking into consideration Eq.(9) and allowing margin constraints to be violated by adding a slack variable we obtain

$$\begin{aligned} \min_{\gamma, w, b} \quad & \frac{1}{2} \|\boldsymbol{w}\|^2 + C \sum_{l=1}^{l} \xi_i \\ \text{s.t.} \quad & y_i(\langle \boldsymbol{w}^T \cdot \boldsymbol{x_i} \rangle + b) \geq 1 - \xi_i \qquad i = 1, ...l \\ & \xi_i \geq 0 \qquad\qquad\qquad\qquad\quad i = 1, ...l, \end{aligned} \tag{16}$$

where C hyperparameter and the slack variable, ξ, measures the distance of an incorrectly classified training point from its corresponding class's margin. Note that if the value of C is small, maximizing the margin is given more importance than the classification error, and vice versa, if the value of C is large. It is important to note that the objective function of the optimization problems Eq.(9) and Eq.(17) differentiate in the second term.

Similar to the process shown for the maximal margin classifier, the dual optimization problem for soft margin optimization can be written as

$$\begin{aligned} \max_{\alpha} \quad & \sum_{l=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} y_i y_j \alpha_i \alpha_j \langle \boldsymbol{x}_i^T \cdot \boldsymbol{x}_j \rangle \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq C, \quad i = 1, ...l \\ & \sum_{i=1}^{l} \alpha_i y_i = 0. \end{aligned}$$

3.3 RF

As defined in Ali et al. (2012), RF is a collection of decision tree classifiers. When testing S', each decision tree casts a unit vote for a class in which the testing example is classified based on its own criteria. Hence, the testing example will be classified to the class that has the greatest number of unit votes. It is important to note that each decision tree varies in structure and variable selection: the data and the features from which the decision trees are trained is made by randomly selected training data from the input space and randomly selected variables.

The main problem with the decision trees classifier is that they tend to produce overfitted models. According to Ali et al. (2012), overfitted models perform poorly when being tested since the classification of such models correspond very closely to the data from which they were trained. Therefore, a benefit of RF is that the classification of a testing example is based on the classification obtained from multiple decision trees. Therefore, the use of RF has a trade-off: a reduction of the variance at the expense of an increase in bias and a decrease of interpretability.

3.3.1 Decision Trees

As defined in Louppe (2014), a Decision Tree is a tree-structured model: $\varphi : X \to Y$, where X is the input space and $Y = \{0, 1\}$ the output domain. A Decision Trees is represented as a binary tree composed by branches (segments that connect nodes), nodes (the position where the branches divide), leaf nodes (the node from which the tree ends) and a root node (the node from which the tree starts). Branches are represented as segments that connect nodes while nodes are represented as circles. In a Decision Tree, each node represents a subspace X_t and has a feature test $s(t)$. To further expand, in the Decision Tree, the outcome of a set of feature questions Q determine the split s_t for every internal node t. Precisely, the split s_t for every internal node t divides the space X_t into disjoint subspaces. Each leaf node, or terminal nodes, are labeled with a guess value $y_t \in Y$. Lastly, the main objective of the Decision Tree is to find a tree-structured classifier that is able to distinguish between classes.

In a Decision Tree, each node has an impurity measure, denoted as $i(t)$. According to Louppe (2014), an impurity measure evaluates the goodness of the classification at a node. One of the most frequently used ways to calculate impurity at a node is by using the Gini impurity calculated as

$$i_G(t) = \sum_{i=1}^{C} p(c_i|t)(1 - p(c_i|t)),$$

where C is the number of classes in the output domain and $p(c_k|t)$ is a conditional probability representing the fractions of the examples labeled with class i at node t. It is important to note that the smaller the impurity measure on a given node, the purer the node.

The creation of a Decision Tree begins from a root node (representing the input space) that iteratively grows by dividing nodes into purer ones. Hence, if all X_t at a given node t belong to the same class, the pure node t will become a leaf node. On the other hand, if the X_t at a given node t contain examples that belong to more than one class, the node is divided into a binary split using a feature test. After a binary split, the node t is divided into a left node t_L and a right node t_R. In such cases, the impurity decrease at node t is calculated as:

$$\triangle i(s, t) = i(t) - p_L i(t_L) - p_R i(t_R),$$

where $p_R = \frac{N_{tR}}{N_t}$ and $p_L = \frac{N_{tL}}{N_t}$ are the fraction of training examples from X_t, with size N_t, that go to t_R and t_L, respectively. The previous statement indicates that the termination criteria for the Decision Tree is achieved once each leaf node is pure. As indicated in Ali et al. (2012), this termination criteria is what precisely makes the Decision Trees prone to overfitting: comprehensive changes in the structure of the decision tree can be caused by small changes in the data. It is important to note that when dealing with Decision Trees not all features are going to be utilized, only those that best divide the training examples into their respective classes are used.

The generalization error is used to calculate the accuracy of a Decision Tree. Given a testing data S', the generalization error of the model φ_L is calculated as:

$$Err(\varphi_L) = P(Y \neq \varphi_L(X)), \tag{17}$$

where P denotes the theoretical probability. However, since the theoretical probability is unknown, Eq.(17) is rewritten as:

$$\widehat{Err}(\varphi_L) = \frac{1}{N} \sum_{(\boldsymbol{x},y)\in S'} 1(y \neq \varphi((\boldsymbol{x})), \tag{18}$$

where the N denotes the number of examples in S' and 1 denotes the unit condition:

$$1 = \begin{cases} 1 & \text{if condition is true} \\ 0 & \text{if condition is false.} \end{cases}$$

Equation (18) measures the probability of misclassification of φ_L, the Decision Tree. Note that the purpose of the Decision Tree is not to make the most accurate prediction on the training set, but on the testing set because it will prove that the model is reliable.

For the generalization error of the model φ_L to be the smallest, the generalization error at each leaf node t has to be minimized as:

$$y_t = \arg\max_{c\in Y} P(Y = c|X \in X_t). \tag{19}$$

However, since the theoretical probability is unknown, (19) is rewritten as:

$$\hat{y}_t = \arg\max_{c\in Y} p(c|t).$$

3.3.2 RF Algorithm

One of the benefits associated with a ensemble of Decision Trees is that, generally speaking, the expected generalization error of a set of Decision Trees is less than the generalization error of each individual Decision Tree. According to Louppe (2014), the RF algorithm uses an ensemble method based on randomization. The idea behind this method is to generate different models from the training data by introducing random perturbations in order to make a prediction based on the set of predictions.

Let M denote a set of randomized models $\{\varphi_{S,\theta_m}|m = 1, .., M\}$ that were built from different random seeds θ_m and where learned from different random samples of training data S. When evaluating the RF algorithm with S', the result obtained is computed by:

$$\Psi_{S,\theta_1,...,\theta_n} = \arg\max_{y \in c} \sum_{l=1}^{C} 1(\varphi_{S,\theta_m}(\boldsymbol{x_i}) = c). \tag{20}$$

As shown in Eq.(20), the label prediction for each $(\boldsymbol{x_i}, y_i) \in S'$ is obtained by considering the classification from the ensemble of Decision Trees. Given that each Decision Tree provides a unit vote for a label prediction, the testing example will be classified on the label prediction that has the majority of votes.

3.4 LR

As mentioned in Barasa and Muchwanju (2015), models such as LR have been regarded as a primal tool to describe the relationship between a dependent variable and a collection of independent variables. The dependent variable, the output domain, is represented as $Y = \{0, 1\}$. The advantages associated with the LR model are many: flexible, practical, and the estimates of effects of the model parameters can be evidenced.

3.4.1 Simple LR Model

The purpose of the simple LR model is to describe the relationship between a dependent variable and a single independent variable. According to Hosmer Jr et al. (2013), a measurement that is of great importance when working with the LR model is the conditional mean, denoted as $E(Y|x) = P(Y = 1|x)$. The classification of a testing example is determined by the value of the conditional mean, and the estimate of the conditional mean must satisfy $0 \leq E(Y|x) \leq 1$.

Let the conditional mean be represented as $\pi(x) = E(Y|x)$ and let the simple LR model be denoted as:

$$\pi(x) = \frac{e^{\beta_0 + \beta_1 x}}{1 + e^{\beta_0 + \beta_1 x}}, \tag{21}$$

where β_0 and β_1 are the parameters of the model. When evaluating $(x_i, y_i) \in S'$, given S' is of the form $S' = ((x_1, y_1), ..., (x_k, y_k)) \in (X, Y)^k$, if the value returned by Eq.(21) is greater than or equal to 0.5, the data point will be classified to the outcome class Y=1. Alternatively, if the value is less than 0.5, then data point will be classified to the outcome class Y=0.

Another representation of the simple LR model is called the logit transformation:

$$g(x) = \ln\left[\frac{\pi(x)}{1 + \pi(x)}\right] = \beta_0 + \beta_1 x_1.$$

The logit transformation model and the simple linear regression model have some common properties: $-\infty \leq g(x) \leq \infty$ and continuity.

In order to test the accuracy of S' in the simple LR model, the model parameters β_0 and β_1 from Eq.(21) have to be estimated. To estimate the previous parameters, the maximum likelihood method is used. As noted in Hosmer Jr et al. (2013), the maximum likelihood estimators of the parameters, obtained from the maximum likelihood method, are those that classify the training set more accurately. However, in order to apply the maximum likelihood method, the likelihood function has to be constructed.

To obtain the likelihood function, let us consider the following considerations. Due to the fact that $Y = \{0, 1\}$, the conditional probability $P(Y = 1|x)$ is the quantity $\pi(x)$ and the conditional

probability $P(Y=0|x)$ is the quantity $1-\pi(x)$. Therefore, when considering a $(x_i, y_i) \in S$, the contribution to the likelihood function of the pair when $y_i = 1$ and when $y_i = 0$ is $\pi(x_i)$ and $1-\pi(x_i)$, respectively. Hence, the contribution of the pair to the likelihood function can be expressed as:

$$\pi(x_i)^{y_i} \left[1-\pi(x_i)\right]^{1-y_i}.$$

Since each pair in S is independent, the likelihood function for the training set S is denoted as the product:

$$l(\boldsymbol{\beta}) = \prod_{i=1}^{l} \pi(x_i)^{y_i} \left[1-\pi(x_i)\right]^{1-y_i}. \tag{22}$$

By applying the log to Eq.(22), the log-likelihood for the training set S can be defined as:

$$L(\boldsymbol{\beta}) = ln\left[l(\boldsymbol{\beta})\right] = \sum_{i=1}^{l} \left(y_i ln\left[\pi(x_i)\right] + (1-y_i) ln\left[1-\pi(x_i)\right]\right). \tag{23}$$

In order to obtain the maximum likelihood estimators of the parameters ($\hat{\boldsymbol{\beta}}$), Eq.(23) has to be differentiated with respect to β_0 and β_1 and the resulting expressions have to be equated to zero to obtain the likelihood equations:

$$\sum_{i=1}^{l} \left[y_i - \pi(x_i)\right] = 0$$

and

$$\sum_{i=1}^{l} x_i \left[y_i - \pi(x_i)\right] = 0.$$

For the simple LR model, hypothesis testing is used to determine which $\beta_i \in \boldsymbol{\beta}$ for $i = \{0, 1\}$ are statistically significant:

$$\boldsymbol{H_0} = \beta_i = 0$$
$$\boldsymbol{H_1} = \beta_i \neq 0,$$

where $\boldsymbol{H_0}$ indicates the null hypothesis and $\boldsymbol{H_1}$ indicates the alternative hypothesis. To determine which parameters are statistically significant, the respective p-value of the maximum likelihood estimators of the parameters have to be considered. If the p-value associated to an $\hat{\beta}_i \in \hat{\boldsymbol{\beta}}$ is greater than a level of significance α, the null hypothesis is accepted. The previous statement suggests that the independent variable is not statistically significant. Conversely, if the p-value associated is less than α, the null hypothesis is rejected. The previous statement suggests that the independent variable is statistically significant.

3.4.2 Multiple LR Model

The purpose of the multiple LR model is to describe the relationship between a dependent variable and a series of p independent variables, denoted as $\boldsymbol{x} = (x_1, x_2, \ldots, x_p)$. As with the simple LR model, a measurement that is of great importance when working with the multiple LR model is the conditional mean, denoted as $E(Y|\boldsymbol{x}) = P(Y = 1|\boldsymbol{x})$.

For simplification purposes, let the conditional mean be represented as $\pi(\boldsymbol{x}) = E(Y|\boldsymbol{x})$ and let the multiple LR model be denoted as:

$$\pi(\boldsymbol{x}) = \frac{e^{\beta_0+\beta_1 x_1+\cdots+\beta_p x_p}}{1 + e^{\beta_0+\beta_1 x_1+\cdots+\beta_p x_p}}, \tag{24}$$

where $\boldsymbol{\beta} = \{\beta_0, \beta_1, ..., \beta_p\}$ are the model parameters. An alternate representation of Eq.(24) is called the logit transformation:

$$g(\boldsymbol{x}) = \ln\left[\frac{\pi(x)}{1 + \pi(x)}\right] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p.$$

Note that when evaluating $(\boldsymbol{x}_i, y_i) \in S'$, if the value returned by Eq.(24) is greater than or equal to 0.5, the data point will be classified to the outcome class Y=1. Alternatively, if the value is less than 0.5, then data point will be classified to the outcome class Y=0.

In order to obtain the maximum likelihood estimators for the multiple LR model, similar to the procedure done for the simple LR model, the model parameters from Eq.(24) have to be estimated with the maximum likelihood method. Consequently, the likelihood equations are constructed:

$$\sum_{i=1}^{l} [y_i - \pi(\boldsymbol{x}_i)] = 0$$

and

$$\sum_{i=1}^{l} x_{ij} [y_i - \pi(\boldsymbol{x}_i)] = 0 \qquad j = 1, \ldots, p.$$

For the multiple LR model, hypothesis testing is used to determine which $\beta_i \in \boldsymbol{\beta}$ for $i = \{0, 1, \ldots, p\}$ are statistically significant using the same criteria as the simple LR model.

4 Solution method / Methodology

The project was conducted using a data mining methodology called CRISP-DM (Cross Industry Standard Process for Data Mining). As described in Niaksu (2015), this methodology presents an organized and iterative process model to tackle a data mining projects by generic-to-specific approach. The CRISP-DM methodology consists of six phases, with cyclic iteration between them: business understanding, data understanding, data preparation, modeling, evaluation, and deployment.

4.1 Business Understanding

The step of business understanding consists on clarifying the objectives of the project and conducting activities such as expert meetings and specific field learning. In regard to the project, various meetings were held with members of the CrediOrbe team: credit analysts, marketing operations group and call center employees. In these meetings, insights such as credit scoring practices and application evaluation processes were discussed.

A credit request can arrive to CrediOrbe in three different manners: through a customer request in a motorcycle dealer company, by filling an online questionnaire on the company's website, and by contacting the company's call center. It is also important to note that once a credit request arrives to CrediOrbe, for its approval, it has to pass through a series of stages. After the credit request arrives to the company, the request is analyzed by a credit analyst and the documentation provided by the customer is verified. Once the customer's request passes this stage, the credit request is automatically approved and, if the customer accepts the credit offer, the disbursement stage will begin. Note that in each of the stages previously described, besides the customers whose credit request is approved, there are customers whose credit request is denied and customers who decide to desist the credit application process.

Note that the customer group in the last stage of the credit application process is conformed by customers who decide to accept their credit offering, and those who desist their credit offering. Specifically, this project focuses on this population since it it that which brings the greatest source of monetary income and loss to the company. Therefore, the objective of the project was clearly established: identify and anticipate the customers who will desist their credit offering by using supervised machine learning algorithms.

4.2 Data Understanding

The second step is data understanding and its objective is to describe and explore the data. This process involves the identification of the dependent variable and the independent variables, data description, bi-variate analysis as well as aggregated data exploration.

4.2.1 Dependent and Independent Variable Identification

Table 1 shows the identification of the continuous independent variables, categorical independent variables and dependent variable, which were considered when evaluating the supervised machine learning algorithms.

Dependent Variable	Continuous Independent Variable	Categorical Independent Variable
Accept/Desist Credit Offering	Number of People in Charge Age Income Expenses Monthly Quota Value Months on Current Job Credit Score	Gender Martial Status Housing Stratum Job Formality Level of Education

Table 1: Variable identification

Let Table 2 describe the variable representation for the conduction of the statistical analysis.

Representation	Variable
x_1	Months on Current Job
x_2	Number of People in Charge
x_3	Age
x_4	Monthly Quota Value
x_5	Income
x_6	Expenses
x_7	Credit Score

Table 2: Variable description

4.2.2 Univariate Analysis

Table 3, which was modified due to confidentiality agreements, describes the univariate analysis for the variables described in Table 2.

Variable	Mean	Median	Mode
x_1	30	18	6
x_2	5	3	4
x_3	39	55	40
x_4	712.3894	301.0570	999.7180
x_5	$5.717x10^6$	$7.2160x10^6$	$4.2000x10^6$
x_6	566	460	700
x_7	190	130	140

Table 3: Univariate analysis

4.2.3 Correlation Matrix

Table 4 describe the correlation matrix for the variables described in Table 2.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
x_1	1	0.1385	0.4425	0.0497	0.2138	0.1756	0.2023
x_2	0.1384	1	0.2749	-0.0693	0.0790	0.0930	0.0455
x_3	0.4424	0.2749	1	-0.0105	0.1993	0.1621	0.3318
x_4	0.0497	-0.0693	-0.0105	1	0.2639	0.1757	0.0667
x_5	0.2137	0.0790	0.1994	0.2639	1	0.9337	0.0862
x_6	0.1756	0.0930	0.1621	0.1757	0.9337	1	0.0121
x_7	0.2022	0.0455	0.3318	0.0667	0.0862	0.0120	1

Table 4: Correlation matrix

4.2.4 Bi-Variate Analysis

In order to conduct the bi-variate analysis, simple linear regression was implemented to establish a relationship between two variables: an independent and dependent variable. The underlying goal of simple linear regression is to establish a relationship of the form

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x. \tag{25}$$

To evaluate how close the data fits the relationship established in (25), measurements such as the R-Squared were considered. Note that the closer the value of the coefficient of determination(R-Squared) is to one, the better the data fits the model.

Table 5 shows examples of simple linear regression models along with their corresponding R-Squared and the p-values of the model parameters are in parentheses.

y	x	β_0	β_1	R-Squared
x_5	x_6	$8.4230\text{x}10^{6}_{(0.000)}$	$1.1452_{(0.000)}$	0.8880
x_5	x_4	$4.1970\text{x}10^{5}_{(0.000)}$	$6.1081\text{x}10^{4}_{(0.000)}$	0.070
x_7	x_5	-	$1.689\text{e-}05_{(0.000)}$	0.551

Table 5: Simple Linear Regression

4.2.5 Aggregated Data Exploration

In order to conduct aggregated data exploration, multiple linear regression was implemented to establish a relationship between a dependent and multiple independent variables. The underlying goal of multiple linear regression is to be able to establish a relationship between the variables of the form

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_k x_k.$$

Table 6 shows examples of multiple linear regression models along with their corresponding R-Squared and the p-values of the model parameters are in parentheses.

y	x_1	x_2	β_0	β_1	β_2	R-Squared
x_7	x_5	x_6	-	$4.111\text{x}10^{-5}_{(0.000)}$	$\text{-}4.035\text{x}10^{-5}_{(0.000)}$	0.706
x_4	x_5	x_6	-	$0.0002_{(0.000)}$	$\text{-}0.0002_{(0.000)}$	0.718
x_5	x_1	x_6	$7.745\text{x}105_{(0.000)}$	$2503.5225_{(0.000)}$	$1.1321_{(0.000)}$	0.874

Table 6: Multiple Linear Regression

4.3 Data Preparation

In this step, the data set is cleaned by using techniques such as NA processing and outlier detection. In addition, principal component analysis (PCA) for dimensionality reduction is also considered

4.3.1 Data Cleaning

As previously stated, the data cleaning process involved a series of steps: dropping missing entries, selecting the problem population as well as outlier detection using Mahalanobis distance. Table 7, shows how the number of data points gradually diminishes in the data cleaning process.

Stage 1	Number of Data Points
Initial data set	171.118
Dropping missing entries	107.797
Selecting the problem population	27.786
Outlier detection	25.835

Table 7: Stage 1

Table 10 shows the division of the data set into a training and testing set. Note that seventy percent of the data is used for testing and thirty percent is used for training.

Stage 2	Number of Data Points
Training data	18.084
Testing data	7.751

Table 8: Stage 2

4.3.2 Principal Component Analysis

The principal component analysis (PCA) was also conducted in order to reduce the dataset's dimension. The goal of this technique is to avoid information loss and improve interpretability. Table 9 shows the eigenvalues of the feature space and Table 10 shows the eigenvectors associated to the principal components.

Eigenvalue
$1.6383 \text{x} 10^5$
$4.3597 \text{x} 10^4$
$1.8577 \text{x} 10^4$
$4.3950 \text{x} 10^3$
$2.2750 \text{x} 10^3$
210.7790
796.4190

Table 9: Eigenvalues

1^{st}	2^{nd}	3^{rd}	4^{th}	5^{th}	6^{th}	7^{th}
0.0782	0.019	-0.9756	0.089	-0.1829	-0.0015	0.0020
0.0030	0.0075	-0.0152	0.0048	0.0764	0.9957	-0.0496
0.0262	0.0212	-0.1836	-0.1020	0.9411	-0.0871	-0.2478
0.1753	-0.9807	-0.0027	0.0818	0.0244	0.0043	-0.0055
0.8198	0.0982	0.0333	-0.55427	-0.0896	0.0047	-0.0431
0.5387	0.1658	0.1015	0.8098	0.1060	-0.0099	0.0691
0.0058	-0.0073	-0.0517	-0.1086	0.2347	0.0298	0.9640

Table 10: Eigenvectors

Table 11 shows the percentage of variability explained by each component of the PCA and Table 12 shows respective the accumulated variability.

Explained Variability
0.7011
0.1866
0.0790
0.0188
0.0097
0.0009
0.0003

Table 11: Explained variability

Accumulated Variability
0.7011
0.8876
0.9671
0.9860
0.9956
0.9965
1

Table 12: Accumulated variability

From Table 12, it can be concluded that the three principal components with the highest variability can explain around 96.71% of the total variability.

4.4 Modeling

As stated in Wirth and Hipp 2000, the modeling phase includes the selection of the modeling techniques (SVM, LR and RF) and the building and description of the models.

4.5 Evaluation

The evaluation and discussion of the algorithm's performance are discussed in Section 5.

4.6 Deployment

The deployment strategies are discussed in Section 6.

5 Results

In this section, the performance and evaluation of the LR, SVM and RF algorithm are shown. Two alternatives were proposed. The first alternative, Alternative 1, considered the categorical and continuous independent variables from Table 1. On the other hand, Alternative 2 was proposed by

considering Table 10. The formulation of this alternative was based on the principal component analysis, were the variables with the largest weights from the first two principal components were taken into consideration. Therefore, variables Z_1 and Z_2 were proposed:

$$Z_1 = 0.18x_4 + 0.82x_5 + 0.54x_6,$$

and

$$Z_2 = 0.17x_6 - 0.98x_4.$$

5.1 Alternative 1 : SVM

Radial based function Kernel(RBF)

Table 13 shows the confusion matrix, Table 14 shows the statistical measurements and Table 15 shows the accuracy for the SVM algorithm with a RBF kernel.

	Predicted Desist	Predicted Accept
Actual Desist	4	2366
Actual Accept	4	5377

Table 13: Confusion matrix

	Precision	Recall	F1-Score
Desist	0.50	0.00	0.00
Accept	0.69	1.00	0.82

Table 14: Statistical measurements

	Accuracy
Training set	0.99
Testing set	0.69

Table 15: Accuracy

Figure 1 shows ROC Curve and Figure 2 shows the heat map obtained from the evaluation of this algorithm.

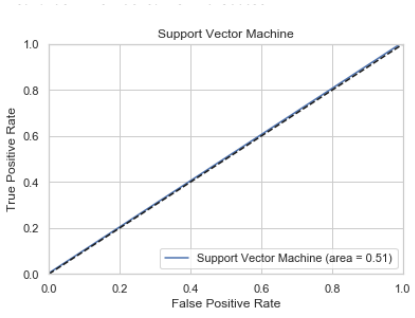

Figure 1: ROC curve

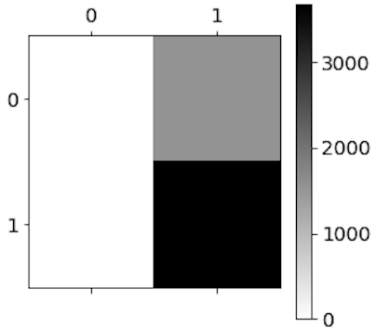

Figure 2: Heat map

From the previous measurement results, it can be concluded that, because the accuracy of the model on the testing data set is very low, the SVM algorithm with RBF kernel cannot distinguish these two classes. It can be concluded from Table 14 and Table 15 that the algorithm classifies the vast majority of data points on the accept class. This problem makes the performance results for the desist class to be greatly deficient, as shown in Table 14.

Polynomial Kernel

Table 16 shows the confusion matrix, Table 17 shows the statistical measurements and Table 18 shows the accuracy for the SVM algorithm with the polynomial kernel.

	Predicted Desist	Predicted Accept
Actual Desist	1453	971
Actual Accept	3203	2124

Table 16: Confusion matrix

	Precision	Recall	F1-Score
Desist	0.31	0.60	0.41
Accept	0.69	0.40	0.51

Table 17: Statistical measurements

	Accuracy
Training set	0.96
Testing set	0.47

Table 18: Accuracy

Figure 3 shows ROC Curve and Figure 4 shows the heat map obtained from the evaluation of this algorithm.

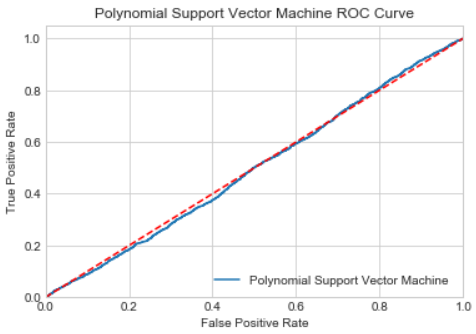

Figure 3: ROC curve

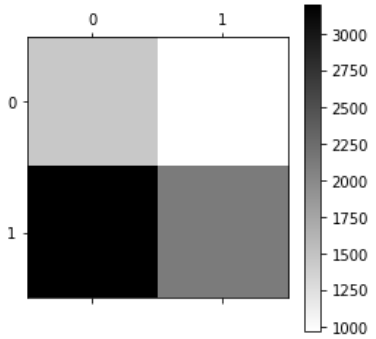

Figure 4: Heat map

It can be concluded from the from Table 16, Table 17 and, Table 18, that the SVM algorithm with polynomial kernel cannot distinguish between the two classes. If the results obtained are compared to the SVM algorithm with a RBF kernel, it can be said that this algorithm classifies a greater number of data points in the desist class. However, the algorithm is not successful when classifying the classes.

5.2 Alternative 1 : LR

Table 19 shows the confusion matrix, Table 20 shows the statistical measurements and Table 21 shows the accuracy for the LR algorithm.

	Predicted Desist	Predicted Accept
Actual Desist	1174	1093
Actual Accept	2212	3272

Table 19: Confusion matrix

	Precision	Recall	F1-Score
Desist	0.35	0.52	0.42
Accept	0.75	0.60	0.66

Table 20: Statistical measurements

	Accuracy
Training set	0.70
Testing set	0.57

Table 21: Accuracy

Table 22 shows the variables that are statistically significant along with the coefficient of the variables and their respective p-value in parenthesis.

Variable	Model Parameter Coefficient
Intercept	$0.0124_{(0.0000)}$
Age	$0.0392_{(0.0066)}$
Credit Score	$0.1008_{(0.0000)}$
Gender	$-0.0993_{(0.0000)}$
Job Formality	$0.1604_{(0.0000)}$

Table 22: Model parameters

Figure 5 shows ROC Curve and Figure 6 shows the heat map obtained from the evaluation of this algorithm.

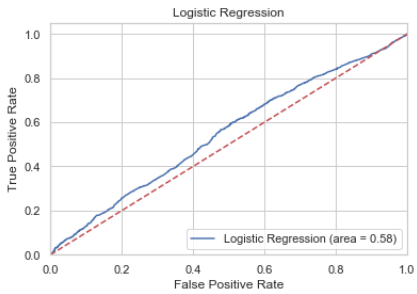

Figure 5: ROC curve

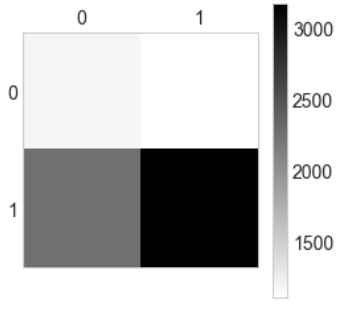

Figure 6: Heat map

It can be concluded from the previous measurement results that the LR algorithm cannot completely distinguish the two classes due to the low accuracy of the model on the testing data set, as shown in Table 21. Further, from Table 19 and Table 20, it can be concluded that the group that accepts credit offering has better performance measurements than the group that desists credit offering.

5.3 Alternative 1 : RF

Table 23 shows the confusion matrix, Table 24 shows the statistical measurements and Table 25 shows the accuracy for the RF algorithm.

	Predicted Desist	Predicted Accept
Actual Desist	1617	688
Actual Accept	1132	4314

Table 23: Confusion matrix

	Precision	Recall	F1-Score
Desist	0.59	0.70	0.64
Accept	0.86	0.79	0.83

Table 24: Statistical measurements

	Accuracy
Training set	0.99
Testing set	0.77

Table 25: Accuracy

Figure 7 shows ROC Curve and Figure 8 shows the heat map obtained from the evaluation of this algorithm.

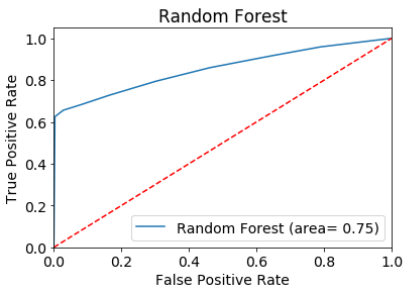

Figure 7: ROC curve

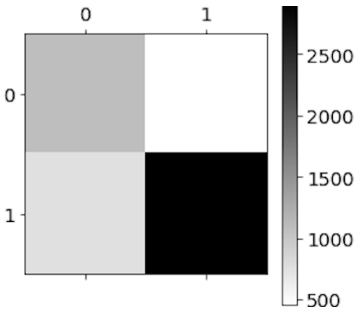

Figure 8: Heat map

It can be concluded from the previous measurement results that due to the accuracy of the model on the testing data set, the RF algorithm can adequately distinguish the two classes. In addition, from Table 24, it can be stated that the group that accepts the credit offering is better classified than the group that desists the credit offering. However, compared to the other algorithms presented, the RF algorithm is that which overall better classifies the individuals who desist the credit offering.

5.4 Alternative 2 : SVM

RBF kernal

Table 26 shows the confusion matrix, Table 27 shows the statistical measurements and Table 28 shows the accuracy for the SVM algorithm with a RBF kernel.

	Predicted Desist	Predicted Accept
Actual Desist	135	2139
Actual Accept	331	5146

Table 26: Confusion matrix

	Precision	Recall	F1-Score
Desist	0.29	0.06	0.10
Accept	0.71	0.94	0.81

Table 27: Statistical measurements

	Accuracy
Training set	0.92
Testing set	0.69

Table 28: Accuracy

Figure 9 shows ROC Curve and Figure 10 shows the heat map obtained from the evaluation of this algorithm.

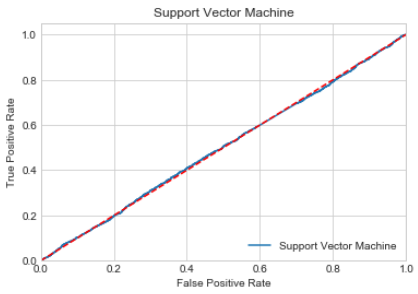

Figure 9: ROC curve

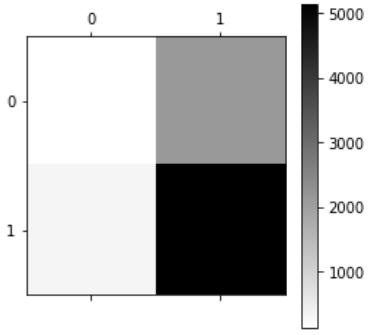

Figure 10: Heat map

It can be concluded from previous measurements that the SVM algorithm with a RBF kernel cannot distinguish between the two categories due to the low accuracy of the model on the testing data set. It can be concluded from Table 26 and Table 27 that the algorithm classifies the vast majority of data points of the accept class.

Polynomial Kernel

Table 29 shows the confusion matrix, Table 30 shows the statistical measurements and Table 31 shows the accuracy for the SVM algorithm with a polynomial kernel.

	Predicted Desist	Predicted Accept
Actual Desist	2409	15
Actual Accept	5302	25

Table 29: Confusion matrix

	Precision	Recall	F1-Score
Desist	0.63	0.00	0.01
Accept	0.32	0.99	0.48

Table 30: Statistical measurements

	Accuracy
Training set	0.72
Testing set	0.31

Table 31: Accuracy

Figure 11 shows ROC Curve and Figure 12 shows the heat map obtained from the evaluation of this algorithm.

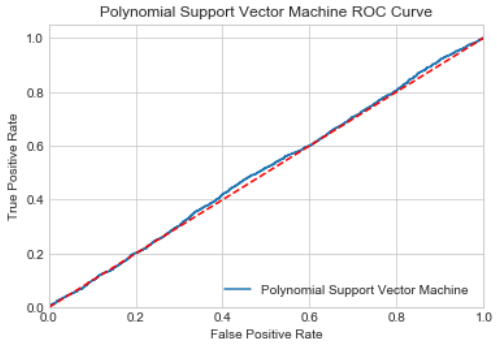

Figure 11: ROC curve

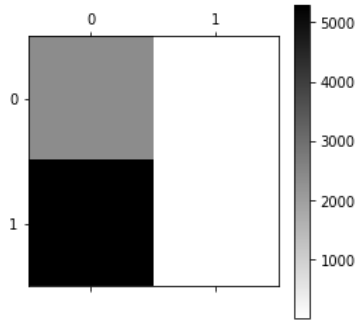

Figure 12: Heat map

It can be concluded from previous measurements that the SVM algorithm with a polynomial kernel cannot distinguish between the two categories due to the low accuracy of the model on the testing data set. Additionally, from Table 29 and Table 30 it can be concluded that the algorithm classifies the vast majority of data points of the desist class. As compared to the results obtained from Alternative 1 with a polynomial kernel, Alternative 2 classifies a significantly lower number in the accept class, which causes a decrease in the testing accuracy (Table 31).

5.5 Alternative 2: LR

Table 32 shows the confusion matrix, Table 33 shows the statistical measurements and Table 34 shows the accuracy for the LR algorithm.

	Predicted Desist	Predicted Accept
Actual Desist	893	1459
Actual Accept	2098	3301

Table 32: Confusion matrix

	Precision	Recall	F1-Score
Desist	0.30	0.38	0.33
Accept	0.69	0.61	0.65

Table 33: Statistical measurements

	Accuracy
Training set	0.68
Testing set	0.54

Table 34: Accuracy

Table 35 shows the variables that are statistically significant along with the coefficient of the variables and their respective p-value in parenthesis.

Variable	Model Parameter Coefficient
Intercept	$0.0002_{(0.0000)}$
Z_1	$-0.0224_{(0.0078)}$
Z_2	$0.0162_{(0.0000)}$

Table 35: Model parameters

Figure 13 shows ROC Curve and Figure 14 shows the heat map obtained from the evaluation of this algorithm.

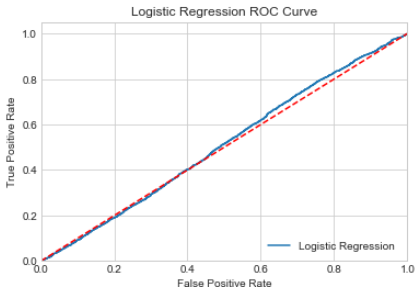

Figure 13: ROC curve

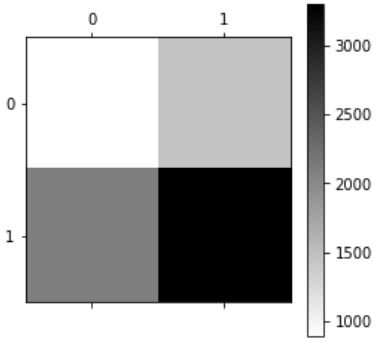

Figure 14: Heat map

By comparing the model's accuracy, obtained when evaluating the LR algorithm for the Alternative 2 (Table 34) to that of the Alternative 1 (Table 21), a slight variation in the testing set accuracy can be seen. To further expand, it can be evidenced that the testing accuracy decreased by three percent. This change is due to the fact that, by considering the principal components for the algorithm evaluation, a loss of variability occurs. Hence, in this scenario, a reduction in dimension results in a loss in model's accuracy.

5.6 Alternative 2: RF

Table 36 shows the confusion matrix, Table 37 shows the statistical measurements and Table 38 shows the accuracy for the RF algorithm.

	Predicted Desist	Predicted Accept
Actual Desist	1587	731
Actual Accept	1223	4210

Table 36: Confusion matrix

	Precision	Recall	F1-Score
Desist	0.56	0.68	0.62
Accept	0.85	0.77	0.81

Table 37: Statistical measurements

	Accuracy
Training set	0.99
Testing set	0.74

Table 38: Accuracy

Figure 15 shows ROC Curve and Figure 16 shows the heat map obtained from the evaluation of this algorithm.

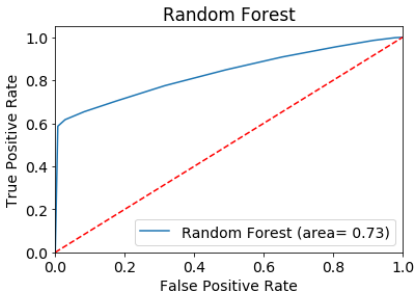

Figure 15: ROC curve

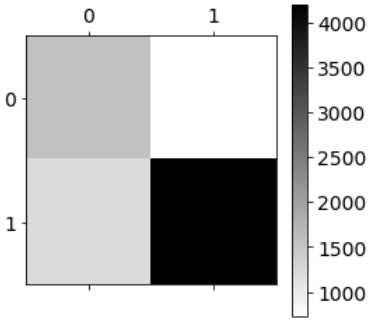

Figure 16: Heat map

By comparing the model's accuracy, obtained when evaluating the RF algorithm for the Alternative 2 (Table 38) to that of the Alternative 1 (Table 25), a slight variation in the testing set accuracy can be seen. To further expand, it can be evidenced that the testing accuracy decreased by three percent. Analogous to the explanation offered for the LR algorithm, this change is due to the fact that, by considering the principal components for the algorithm evaluation, a loss of variability results in a loss in model's accuracy. However, compared to the other algorithms from Alternative 2, the RF algorithm is also that which best classified overall both classes.

6 Conclusions and future research

From the performance measurements, it can be concluded that the RF algorithm performs significantly better than the SVM and LR algorithm. Not only is the accuracy of this algorithm the highest, but also the overall precision, recall and F1-Score for the desist and accept class. In terms of applicability, the deployment strategy consists on implementing this algorithm once the client's credit request is accepted. Implementing this algorithm at this stage of the credit application process will allow the application of strategies such as marketing segmentation practices. According to Tsai and Chiu (2004), due to the fact that the market consists of a diverse group of customers, mass market approaches have been proven to be ineffective. Furthermore, market segmentation has proven to be potent strategy when tackling problems of this nature. In practice, the market is divided into customer clusters where members of the same customer cluster share similar needs and underlying characteristics, making these more likely to exhibit a homogeneous response when presented to marketing programs.

As stated in Tsai and Chiu (2004), when a financial entity decides to select a specific customer cluster, by implementing suitable market segmentation practices, it will have the ability to establish a close relationship with the individuals from this group by implementing and offering a targeted services. In the terms of this project, implementing market segmentation on the groups of individuals who desist the credit offering, after surpassing CrediOrbe's minimum credit scoring threshold, will allow the company to build a relationship with the clients so that these become less prone to desist their credit offering. Implementing this marketing practice, specifically at the last stages of the credit application, could potentially save the company large sums of money. Note that when the credit is granted to a client, the client would have already passed through a series of stages in which the application is evaluated. As a whole, the evaluation of a credit application symbolize a significant source of cost to the company since it involves operational and labor costs. Once the company has the ability to identify the clients who will desist at the last stages, a mitigation of cost could be witnessed. Ultimately, as denoted in Tsai and Chiu (2004), during the past decades, market segmentation practices have allowed companies to establish a closer relationship with their customers.

Acknowledgements

We would like to thank Universidad EAFIT for the opportunity to conduct this research practice as well as CrediOrbe for providing us the data set for the project's elaboration.

References

Ali, Jehad et al. (2012). "Random forests and decision trees". In: *International Journal of Computer Science Issues (IJCSI)* 9.5, p. 272.

Barasa, Kennedy Sakaya and Chris Muchwanju (2015). "Incorporating Survey Weights into Binary and Multinomial Logistic Regression Models". In: *Science Journal of Applied Mathematics and Statistics* 3.6, p. 243.

Cristianini, Nello and John Shawe-Taylor (2000). *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods.* Cambridge University Press. ISBN: 9780511801389.

De Veaux, R. D. (2003). "Data mining: where do we start?" In: *Proceedings of the 25th International Conference on Information Technology Interfaces, 2003. ITI 2003.* Pp. 19–.

Gouvêa, Maria Aparecida and Eric Bacconi Gonçalves (2007). "Credit risk analysis applying logistic regression, neural networks and genetic algorithms models". In: *POMS 18th Annual Conference.*

Hosmer Jr, David W, Stanley Lemeshow, and Rodney X Sturdivant (2013). *Applied logistic regression.* Vol. 398. John Wiley & Sons.

Louppe, Gilles (2014). "Understanding random forests: From theory to practice". In: *arXiv preprint arXiv:1407.7502.*

Niaksu, Olegas (2015). "CRISP data mining methodology extension for medical domain". In: *Baltic Journal of Modern Computing* 3.2, p. 92.

Tax, David MJ and Robert PW Duin (1999). "Data domain description using support vectors." In: *ESANN.* Vol. 99, pp. 251–256.

Thomas, Lyn C, David B Edelman, and Jonathan N Crook (2002). *Credit scoring and its applications.* SIAM.

Tsai, C-Y and C-C Chiu (2004). "A purchase-based market segmentation methodology". In: *Expert Systems with Applications* 27.2, pp. 265–276.

Vojtek, Martin and Evzen Kocenda (Jan. 2006). "Credit scoring methods". In: *Finance a Uver - Czech Journal of Economics and Finance* 56, pp. 152–167.

Wirth, Rüdiger and Jochen Hipp (2000). "CRISP-DM: Towards a standard process model for data mining". In: *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining.* Springer-Verlag London, UK, pp. 29–39.