Leveraging Churn Prediction to Drive Effective Decision-Making in the Media Industry

Ricardo Emanuel Padrão Sereno Cabral

Master's Dissertation

Supervisor at FEUP: Marta Campos Ferreira

Master's in Industrial Engineering and Management

2023-06-26

Abstract

The newspaper industry is highly competitive and the increasing usage of the internet and online news consumption led to a digital disruption phenomena, reshaping the newspaper and media industry and specially the customer-firm relationship, with the current landscape displaying a highly competitive and churn-prone environment. In this context, efficient data gathering and analysis become extremely important to better support business decisions, such as customer retention.

This thesis presents a comparative analysis of machine learning techniques to assess customer churn in a Portuguese newspaper outlet. It aims at contributing to the present literature by: (1) presenting a comparative machine learning models approach in the newspaper industry – which presents, as of today, a relatively scarce literature; (2) implementing a novel community-based detection algorithm for feature selection based on multi-objective optimization for feature clustering; (3) considering the dynamics of customer behavior by using multi-time-slices and presenting its empiric impact; (4) discussing an uplift model to better distinguish different types of churners and (5) determining optimal subscription characteristics and retention measures to diminish customer turnover.

The conclusions drawn from this work are the following: (1) xGBoost demonstrated to be the most powerful algorithm to predict churn; (2) both Boruta and the community based genetic algorithm present good ability to reduce problem dimensionality without compromising significantly the predictive ability; (3) share and subscription's age are two of the most impactful variables in predicting whether or not a customer is going to cancel its subscription – indicating that the affiliation and duration of a customer's association with the service has a substantial impact on their likelihood of canceling their subscription; (4) the historical record of each customer's interaction with the company's website, as measured by the cumulative number of page views, also emerged as a reliable predictor of churn intentions and (5) price assumes to be an important indicator regarding the subscription's purchase but not its renewal.

Resumo

A indústria dos jornais é altamente competitiva e o aumento do uso da internet e o consumo de notícias online potenciaram um fenómeno de disrupção digital, reconfigurando a indústria dos jornais e dos mídia, nomeadamente a relação entre o cliente e a empresa, onde o *status quo* se carateriza por um ambiente altamente competitivo e propenso a perda de clientes. Neste contexto, a coleta e análise eficiente de dados tornam-se extremamente importantes de modo a apoiar melhor as decisões de negócio, bem como a retenção de clientes.

Esta tese apresenta uma análise comparativa de técnicas de *machine learning* para avaliar o cancelamento de subscrições online num jornal português. O presente texto visa contribuir para a literatura atual do seguinte modo: (1) apresentando uma abordagem comparativa de modelos de *machine learning* na indústria dos jornais – que apresenta, até hoje, uma literatura relativamente escassa; (2) implementando um novo algoritmo de deteção baseado na teoria dos grafos e criação de comunidades e uma função multi-objetivo para seleção de atributos como entrada no modelo; (3) considerando o comportamento dinâmico do cliente utilizando segmentos temporais bem definidos e apresentando o seu impacto empírico; (4) discutindo um modelo de $uplift$ para distinguir melhor os diferentes tipos de clientes propensos a cancelar a subscrição e (5) determinando características ótimas de assinatura e medidas de retenção para diminuir o cancelamento de assinaturas.

Este trabalho permitiu concluir o seguinte: (1) O xGBoost demonstrou ser o algoritmo mais poderoso para prever a perda de clientes; (2) tanto o Boruta quanto o algoritmo genético baseado na teoria dos grafos e criação de comunidades apresentam boa capacidade de reduzir a dimensionalidade do problema sem comprometer significativamente a capacidade preditiva; (3) a participação e a idade da assinatura são duas das variáveis mais impactantes na previsão de se um cliente vai ou não cancelar sua assinatura – indicando que a afiliação e duração da associação do cliente com o serviço têm um impacto substancial na sua probabilidade de cancelar a assinatura; (4) o registo histórico da participação online do cliente, medida pelo número de páginas vistas num dado período de tempo, apresenta também alguma capacidade preditiva e (5) o preço demonstra ser um indicador importante em relação à compra da assinatura, mas não em relação à sua renovação.

Acknowledgments

A significant milestone has been reached, prompting reflection on the common thread throughout various life events: the individuals who supported and accompanied me on my journey. This expression of gratitude serves as a tribute to all of them.

To start with, I'd like to thank LTPlabs for the chance of developing my master thesis under such good guidance. To Pedro Amorim and Bernardo Almada-Lobo, who where very much the team that accompanied me throughout these last months, a warm thank you.

Finishing my master's was probably the most enduring battle I've yet faced.To my coordinator, Marta Campos, I express my gratitude for keeping me on track and ensuring that I met the deadline. To every other professor at FEUP who has contributed to my education over the years, I am deeply thankful for the unending challenges that have shaped me into the person I am today.

Life extends beyond the realm of career pursuits, and success cannot be measured solely by objective standards. To my family, who have supported me unconditionally, I extend an eternal thank you. To my friends, whose constant presence and encouragement pushed me beyond my perceived limitations, I am grateful.

Lastly, I express my deepest appreciation to my heroine, my mother, who has shown me that life is not meant to be easy. Despite the obstacles one may encounter, it is our responsibility to persevere. Mom, I love you and thank you for everything.

"For what I lack in experience I make up for in diligence."

Marcus Cícero

Contents

Acronyms and Symbols

AUC	Area Under the Curve
AUCPR	Area Under the Precision-Recall Curve
CGC	Cumulative Gains Chart
CRM	Customer Relationship Management
DNN	Deep Neural Networks
DRF	Default Random Forest
DT	Decision Trees
ERF	Extremely Randomized Forest
ETL	Extract, Transform and Load
FN	False Negatives
FP	False Positives
GB	Gradient Boosting
GBM	Gradient Boosting Machine
GLM	Generalized Linear Models
ISCD	Iterative Search Community Detection
LTV	Lifetime Value
ML	Machine Learning
NN	Neural Networks
RF	Random Forest
SVM	Support Vector Machine
TN	True Negatives
TP	True Positives
xGBoost	Extreme Gradient Boosting

List of Figures

List of Tables

Chapter 1

Introduction

This chapter elucidates the underlying motivation of the research and its relationship with the newspaper industry. Subsequently, the project's objectives are delineated, followed by a comprehensive overview of the paper's structure. Significant notes are added to facilitate comprehension and assist the reader in following the discourse.

1.1 Motivation

In the last ten years, the news media industry has experienced a digital disruption, whereby the traditional three-sided marketplace operation, comprising of producing trusted news content, selling reader views or advertisements to brands, businesses, and individuals, and distributing through complex platforms, has been restructured, as stated by Shruti (2017). Heavy capital expenditure requirements and complex distribution platforms have been an entry barrier for new players, giving an advantage to established firms. However, the introduction of the internet and online media consumption has led to a reshaping of the supply chain, where content can be produced at zero marginal cost per view, creating a new setting that provides an edge to relatively new but popular media companies.

The Pew Research Center estimated that over 80% of Americans obtain their news from online sources, as per Shearer (2021). Additionally, a study by the Reuters Institute (Cardoso et al., 2021) indicated that nearly 73% of readers in Spain accessed news through their smartphones, with increasing trends observed in other countries.

The increasing digitalization of the news media industry requires changes in the behavior of firms. Boutetière et al. (2018) emphasizes the need for implementing digital tools to facilitate the analysis of complex information to achieve success in the digital transformation era, reinforcing the proposition that data gathering and analysis may empower business decisions.

Additionally, the decreasing overall interest in news and increasing suspicion about their trustworthiness (Cardoso et al., 2021) places the online media industry in a fragile situation. Hence, and knowing that most people only subscribe to one online media service (Fletcher, 2019), the

landscape displays a highly competitive and churn-prone environment, thus supporting the importance of customer retention strategies, specially since retaining a customer is less costly than acquiring a new one (Balle et al., 2013). All in all, given the potential benefits of using data analytics for customer retention and firm profit, there is a need for practical approaches to predict customer churn and apply retention measures.

This project aims to develop a machine learning model for predicting churn in a Portuguese news outlet. Specifically, a comparison between up-to-date machine learning classifiers will be conducted. Additionally, the impact of a community-detection feature selection and multi-objective feature clustering will be assessed. Moreover, multi-time-slices will be used to capture the dynamics of customer behavior over time and an uplift model will be discussed as an additional layer to the churn model, allowing the firm to differentiate churners. Finally, the research aims to bridge the gap between academic investigation and practical implementation by identifying a set of retention measures that can enhance the organization's capacity to retain customers.

1.2 Objectives

As previously stated, the main objective of this research is to develop and evaluate machine learning models for predicting churn in a news outlet and present possible retention measures, based on the most prominent and important features observed. Developing churn prediction models is a valuable and highly explored area of research. This paper aims to contribute by: (1) presenting a comparative machine learning models approach in the newspaper industry – which presents, as of today, a relatively scarce literature; (2) implementing a novel community-based detection algorithm for feature selection based on multi-objective optimization for feature clustering; (3) considering the dynamics of customer behavior by using multi-time-slices and present its empiric impact; (4) discussing an uplift model to better distinguish different types of churners and (5) determining optimal subscription characteristics and retention measures to diminish customer turnover.

To achieve the above presented objectives, the following steps were undertaken:

1. Collect and prepare a dataset – essentially a collection of data – of customer information for machine learning purposes. This step involves the selection of appropriate features through the Feature Selection methodology and the cleaning of the dataset (through null-values imputation, outlier treatment, etc.);

2. Develop and train machine learning models for predicting customer churn in a Portuguese news outlet. This objective involves the selection, development and implementation of the best machine learning models for the problem at hand, as well as the optimization of each model's hyperparameters. The multi-time-slice approach will refer to this step;

3. Evaluate the performance of the selected models. This step involves selecting and using appropriate (and previously discussed) metrics to select the winner model;

4. Identify the variables and factors most relevant for predicting churn. This milestone comprises mainly the interpretation of the machine learning models' results;

5. Construct an uplift model aiming at differentiating churners (thus allowing to minimize resource wastage);

6. Develop, alongside the Portuguese news outlet's teams, customer prevention and retention plans, providing valuable insights about customer behavior and tracking the effectiveness of each measure.

1.3 Dissertation Structure

This dissertation is divided into five different sections. In Chapter 2, background literature on data mining, customer relationship management, machine learning and machine learning related topics is presented. The chapter also encompasses literature review regarding both churn and uplift modelling and aims at building a common foundation in which to develop the following themes. Chapter 3 presents the problem description and the market and company-specific information regarding the firm whose data was used to deploy the models. Additionally, the methodology used is detailed, where Data Management and the Initial Setting are discussed. Finally, results are shown in Chapter 4 and conclusions are drawn and presented in Chapter 5. A comprehensive summary of the project can be seen in Figure A.1, in the annexes.

Chapter 2

Background and Literature Review

In this section, an overview of some background concepts will be presented, such as data mining, customer relationship management, machine learning, machine learning techniques, regularization techniques, performance evaluation, hyperparameter tuning and cross-validation. Subsequently, an introduction to the current state-of-the-art in both churn and uplift fields will follow.

2.1 Background

The purpose of the Background section, as mentioned earlier, is to provide a shared understanding for the reader. Consequently, a comprehensive elucidation of all crucial concepts pertaining to this objective will be addressed.

2.1.1 Data Mining

Data mining is an interdisciplinary research area aimed at extracting valuable insights and knowledge from large datasets (Fayyad et al., 1996). Its development has been fueled by the exponential increase in data availability in diverse fields such as business, science, engineering, and medicine. Consequently, data mining techniques have become a crucial tool for discovering hidden patterns and trends, predicting future outcomes, and making informed decisions.

The data mining process involves several stages, including data preparation, data exploration, modeling, evaluation, and deployment (Luo, 2008). During the data preparation stage, raw data is collected and preprocessed to ensure its suitability for analysis. In the data exploration phase, various techniques are employed to understand the data and uncover patterns and trends. The modeling stage utilizes statistical and machine learning methods to build models that can predict future outcomes or classify data. In the evaluation phase, the performance of the models is assessed, and necessary improvements are made. Finally, in the deployment phase, the models are integrated into the respective environment.

A variety of data mining techniques are available for different applications, including classification, clustering, association rule mining, and anomaly detection. Classification is a supervised learning technique that categorizes data into predefined classes. Clustering is an unsupervised

learning technique that groups data into clusters based on similarities. Association rule mining involves discovering relationships between variables, while anomaly detection identifies outliers or unusual patterns in data.

2.1.2 Customer Relationship Management

Customer Relationship Management (CRM) encompasses guidelines, procedures, processes and strategies that enable organizations to consolidate customer interactions and track all customer-related information (Khan et al., 2012). It is considered a customer-centric business strategy (Guo and Qin, 2017) and plays a crucial role in establishing effective channels and methods for managing customer-based information. In the modern business landscape, CRM is recognized as one of the most powerful tools for managing business reality, particularly when combined with technological innovations, as it has the potential to enhance firms' economic performance (Guerola-Navarro et al., 2021).

CRM can be applied to four key areas: customer identification, customer attraction, customer retention, and customer development (Guerola-Navarro et al., 2021). This thesis primarily focuses on customer retention, which involves building customer loyalty and establishing long-term fruitful relationships to improve company performance. To achieve this, it is essential to comprehend the primary factors influencing a customer's decision-making process regarding the company's products and services. By gaining a deeper understanding of customer behavior, firms can better engage customers, foster lasting relationships, and reduce customer churn.

To achieve this, churn models can be employed, which aim to predict the likelihood of customer churn based on pattern recognition and signal analysis (e.g., mapping consumer behavior). Various machine learning techniques have been developed to address this challenge, enabling organizations to proactively identify customers at risk of churn and implement targeted retention strategies. These advanced analytical approaches provide valuable insights for understanding customer preferences, anticipating their needs, and fostering customer loyalty.

2.1.3 Machine Learning

Machine learning may be described as "the technique that improves system performance by learning from experience via computational methods", as per Zhou (2021). Thus, machine learning is the ability to automatically transform data – experience – into knowledge, *i.e.*, meaningful relationships and patterns (Bishop, 2007).

Russel and Norvig (2021) state that early attempts to construct analytical models relied on manually programming recognized relationships, procedures, and decision-making logic into intelligent systems, such as expert systems used for medical diagnosis. However, with the development of new programming frameworks, easy access to data, and the widespread availability of computing power, analytical models are increasingly being built using machine learning (ML) techniques, as described by Brynjolfsson and McAfee (2017).

ML has shown to be reliable for classification, regression or clustering purposes, as shown by Janiesch et al. (2021). Based on the problem at hand, ML problems may be categorized into supervised learning, unsupervised learning and reinforcement learning. Supervised learning involves the use of a training dataset containing input examples as well as labeled target values or answers for the output. Unsupervised learning refers to learning systems that discover patterns without prior labelling or specifications. Reinforcement learning, on the other hand, does not rely on input and output pairs. Instead, the current system state is described, a goal is specified, a set of permissible actions and their environmental constraints are provided for their outcomes, and the ML model is allowed to experience the process of achieving the goal on its own using the trial-and-error principle to maximize a reward. They are mostly used in robotics or gaming and will not be a topic of discussion in this thesis.

Regarding supervised machine learning techniques, one may refer to both regression or classification problems. Regression problems involve predicting a continuous numerical value, such as the price of a house or the temperature of a city. The goal of regression is to learn a relationship between the input features and the output value, which can be used to predict the output value for new data points. In regression, the output variable is considered to be a dependent variable that depends on the values of the independent variables or input features. The output of a regression model is a continuous function that can take on any value within a given range. On the other hand, classification problems involve predicting a categorical label or class, such as whether an email is spam or not, or whether a customer is likely to buy a product or not. The goal of classification is to learn a decision boundary that separates the input features into different classes based on their attributes. In classification, the output variable is considered to be an independent variable that determines the class to which the input features belong. The output of a classification model is a discrete function that assigns a label or class to a new data point. This thesis will focus on the implementation of a classification model, aiming at predicting if a given customer will churn or not in the next k months.

With respect to unsupervised machine learning techniques, one of the most common ones is clustering. It can be seen as a method to find a structure in a collection of unlabeled data (Madhulatha, 2012) by aggregating objects that are similar between them and dissimilar when compared to the others. A wide variety of clustering techniques have been used in many applications including machine learning, data mining, pattern recognition and much more – this thesis will use its own version of clustering, carefully explained upfront.

The following section will be divided as follows:

- Firstly, the concepts of bias and variability will be explained, enabling the reader to better understand all the topics discussed ahead;

- An overview of the diverse machine learning techniques applied in the comparative analysis will be presented. Specifically, the benchmark analysis will involve Generalized Linear Model, Decision Trees, Random Forest, Neural Networks, Gradient Boosting and Extreme Gradient Boosting;

- Subsequently, the topic of regularization, one of machine learning most important concepts, will be discussed and some techniques used in the ML methods described will be presented;

- Evaluation metrics, to better assess the most accurate and tailored model to the problem at hand, will be discussed.

- Afterwards, balancing techniques, hyperparameter tuning and cross-validation, all used to improve a model's results will be further explained;

- Finally, an overview of churn and churn-related literature (namely uplift modelling) will be detailed.

2.1.3.1 Bias and Variance

Bias may be defined as the simplifying assumptions made by a model to make the target function easier to learn – thus connected to the difference between the predicted value and the true value of the output variable. Models with high bias oversimplify the model and show poor results on the training and test set. A high value is thus connected to underfitting, which states the idea that the model was not able to recognize all the existing patterns – it usually happens when less than required data is used to build the model or non-linear data is applied to a linear model.

On the other hand, variance is the variability of the model, which states the spread of the data – one may think of it as the amount that the estimate will change given different datasets. Models with high variance pay a lot of attention to the training data and tend to poorly generalize on data not yet observed – thus showing poor results on the testing data, regardless of the good results in the training set. High variance is connected to overfitting, which states the idea that the model captured noise along the underlying pattern of the data.

The prediction error for any machine learning algorithm can be broken down into three parts: bias error, variance error and irreducible error. Bias and Variance are reducible errors, in the sense that they can be reduced to improve model accuracy. However, irreducible error refers to errors that will always be present in the model and cannot be tackled.

Systematic errors refer, then, to systematic deviations that are not due to chance alone, as opposed to random error which states the idea of variability introduced in the system in unpredictable ways. In this sense, unlike random errors, averaging a large number of observations does not yield a zero net effect, as the errors from the observations follow a given direction and show a skewed distribution (Bhandari, 2023).

Thus, there is a trade-off at play between these two concepts, as increasing the variance decreases the bias and vice-versa, clearly shown in the Figure 2.1 below. It is important to state that all models should strive for low variance and low bias, although usually linear models present high bias and low variance and non-linear models present high variance and low bias.

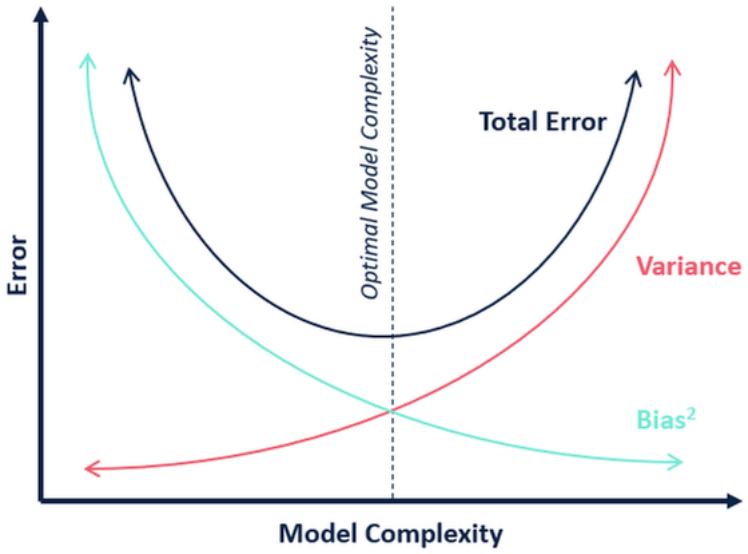


Figure 2.1: Graphical representation of the trade-off between bias and variability, adapted from Fortmann-Roe (2012).

2.1.4 Machine Learning techniques

Various machine learning techniques have been used across literature for the most various purposes with no definitive consensus on which approach is universally better. Accordingly, this section will outline several selected methods, with a corresponding summary table provided in the appendixes (B.1).

2.1.4.1 Decision Trees

Decision trees consist of nodes that form the so called root tree that presents no incoming edges. From this starting point, all the other nodes have exactly one incoming edge (these are the leaves).

The basic idea behind a decision tree is that of recursively partitioning the data based on the values of the input variables in order to minimize the entropy, *i.e.*, the uncertainty or randomness of the set of data – thus selecting, at each point, the attribute that best separates the data into two or more groups.

Decision trees are usually easy to interpret and explain and can deal well with large amounts of noisy data (Vafeiadis et al., 2015) but may be prone to overfitting and may not perform as well as other algorithms (generically). Additionally, their greedy construction at each step may lead to overall sub-optimal selection of attributes and they only deal with discrete variables (Dreiseitl and Ohno-Machado, 2002). Some argue that they may not be viable when dealing with imbalanced datasets (Japkowicz and Stephen, 2002) (Branco et al., 2017). An illustrative example of a decision tree can be seen in Figure 2.2.

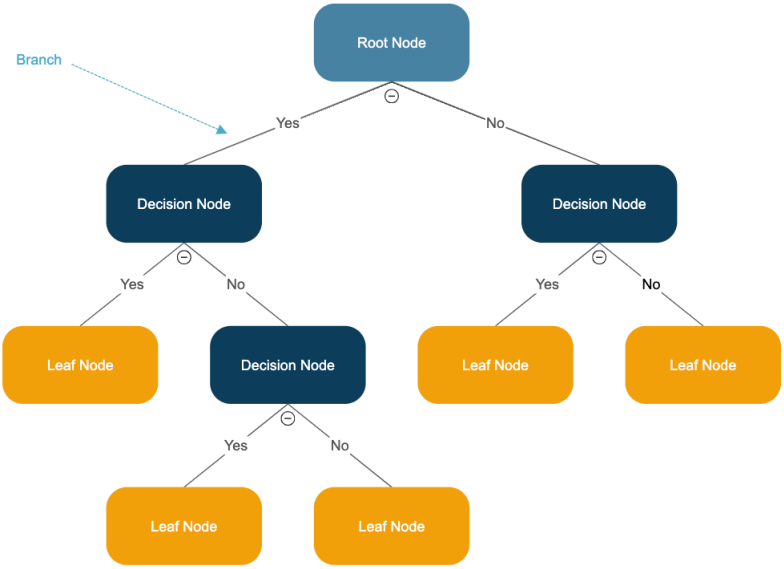


Figure 2.2: Illustrative example of a decision tree, adapted from SMARTDRAW.

2.1.4.2 Logistic Regression

In a logistic regression model, the value of the output variable is given by assigning a given coefficient to each of the input variables. In this sense, logistic regression is a discriminative classifier, *i.e.*, tries to predict the probability of the target output being a given class by fitting data into a logistic function. Despite its name, it is mostly used for binary and linear problems, as described by Subasi (2020).

The logistic function is a sigmoid function, which takes any real input t and outputs a value between 0 and 1. For the logit, this is interpreted as taking as input log-odds and having output probability – this will be clearer upfront.

The standard logistic function, $\sigma : \mathbb{R} \to [0,1]$ can be defined as:

$$\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$$

If we assume that t is a linear function of a single explanatory variable x, as:

$$t = \beta_0 + \beta_1 * x$$

The general logistic function p, where $p : \mathbb{R} \to [0,1]$ can be written as:

$$p(x) = \sigma(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 * x)}}$$

Thus, $p(x)$ can be interpreted as the probability of the dependent variable Y equaling a success rather than a failure. The logit function can be determined as the inverse of the standard logistic function, such that:

$$g(p(x)) = \sigma^{-1}(p(x)) = logit\, p(x) = ln\frac{p(x)}{1-p(x)} = \beta_0 + \beta_1 * x$$

Thus, the odds of a dependent variable equaling a success (or a case) may be given by the exponential of the last equation, as one can see below. This expression can then be generalized to a multiple explanatory case. The main drawback of logistic regression techniques is the assumption of a linear relationship between the input variables and the dependent variable – thus not being great at finding complex patterns (Dreiseitl and Ohno-Machado, 2002).

$$\frac{p(x)}{1-p(x)} = e^{\beta_0+\beta_1*x}$$

2.1.4.3 Generalized Linear Models

Generalized Linear Models (GLMs) can be regarded as an expansion of conventional linear models, including linear regression and logistic regression (Nykodym et al., 2023). GLMs have garnered significant attention as they offer the ability to relax certain constraints inherent in these models, thereby exhibiting enhanced flexibility and applicability across a wider range of problems. Specifically, GLMs allow the variance to vary as a function of the mean – whereas in the traditional models its assumed constant – and allow for non-normality of errors and a non-linear relationship between the response variables and the coefficients. This non-linearity is achieved by assuming an exponential relationship, thereby encompassing distributions such as Poisson, Gaussian, binomial, multinomial, and gamma.

2.1.4.4 Random Forest

A Random Forest classifier consists of a combination of tree classifiers using, at each iteration, a bootstrapped sample and considering only a subset of variables (Kirasich et al., 2018). Each tree casts a vote referring to the most popular class to classify an input vector. A bootstrapped sample refers to randomly selecting a sample with replacement from the N observations, where N is the total size of the original dataset. Since the method requires bootstrapping the data and using the aggregate to make a decision, the stated method is called bagging (Kirasich et al., 2018).

One of the advantages of the random forest classifier is its ability to handle high-dimensional data with many features (Ahmad et al., 2017). The algorithm is also relatively robust to noise and outliers (Breiman, 2001), as it uses a voting mechanism to make the final prediction. Moreover, the random forest can provide an estimate of the importance of each feature in the dataset, which can be useful for feature selection and feature engineering since it is able to effectively capture relationships between features.

However, the Random Forest classifier has some limitations. Trees are not good at capturing the relations between features and the response variable and the number of trees used is a critical factor, since it may overfit the training data if the number of trees in the forest is too large, and

it may underfit the data if the number of trees is too small. The algorithm also requires more computational resources and memory compared to, for instance, Decision Trees.

2.1.4.5 Neural Networks

Neural Networks are based on the idea of continuous learning from a given set of examples. Initially, the inspiration came from studies of biological nervous systems, particularly the human brain (Bishop, 1994). A feed-forward Neural Network is a non-linear mathematical function that transforms a set of input data into output data, simply put.

The process of data transformation involves a series of different layers which comprise a set of parameters, called weights – these must be calculated iteratively, which may be computationally intensive. However, after the parameter computation – which is traditionally called training – the algorithm is quite fast.

The principal disadvantage of a Neural Network stems from the need to provide a suitable set of examples for training and the lack of flexibility in generalizing in regions outside of the training space.

A simple mathematical model of a neuron was first introduced by McCulloch and Pitts (1943), stating a non-linear function that transforms a set of inputs, $x_i, i \in [1, \ldots, d]$, into an output variable z. As one may see in the equation below, the signal *xi* is first multiplied by a parameter *wi* – the weight – and then added to all other weighted signals to give a total input for the non-linear function g.

Thus, a can be given by:

$$a = \sum_{i=1}^{d} w_i * x_i + w_0 \tag{2.1}$$

Where w_0 is called bias and represents an offset parameter – typically regarded as a special case of a weight from an extra input with value permanently set to +1. Thus, one may rewrite the equation above as:

$$a = \sum_{i=0}^{d} w_i * x_i \tag{2.2}$$

Afterwards, the output z is given by applying the non-linear activation function g, such that:

$$z = g(a) \tag{2.3}$$

A visual representation of a simple neural network can be seen below, in Figure 2.3.

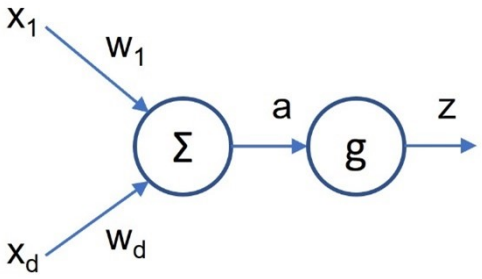


Figure 2.3: Visual representation of a simple neural network.

2.1.4.6 Gradient Boosting

Gradient boosting relies on the idea of building an ensemble of models for some particular learning task. Thus, the strategy is based on building a set of weak models and combine them to obtain one strong learner (Sharma et al., 2020).

To do so, the new models are added sequentially, where each new learner is trained with respect to the errors of the previous one. This is achieved by fitting each model to the residuals, or errors, of the previous model.

Specifically, the residuals of the previous model are used as the target variable for the next model – the models are typically Decision Trees, which introduces the problem of deciding the best feature to split each node (usually Gradient Boosting techniques consider all possible splits and then decide upon the best one).

Gradient Boosting is called such because it uses the gradient, or the slope, of the loss function to update the model parameters. The loss function measures the difference between the predicted values and the actual values, and the gradient of the loss function is used to determine the direction and magnitude of the update. The gradient is computed with respect to the model's parameters, and the update is performed in the direction that minimizes the loss function. For classification purposes, residuals are calculated using the log of the odds and transforming this value into a probability.

Initially, the boosting techniques that garnered considerable attention were solely driven by algorithms, resulting in challenges in performing an in-depth analysis of their characteristics and efficency – as seen by Schapire (2003), leading to various speculations regarding its results (Sewell, 2011).

Gradient Boosting techniques can handle a wide variety of data types and are known to deal effectively with high-dimensional and sparse data (Natekin and Knoll, 2013).

2.1.4.7 Extreme Gradient Boosting

Extreme Gradient Boosting differs from Gradient Boosting in mainly two ways: it is a more regularized form of gradient boosting and it is one of the fastest implementations of Gradient Boosting trees.

With respect to the first major difference, GB usually uses subsampling, shrinkage or early stopping as regularization techniques, whilst xGBoost uses L1 and L2 regularization – these all will be covered next (Chen and Guestrin, 2016).

With respect to the second one, xGBoost tackles one of the major inefficiencies regarding the implementation of gradient boosted trees – considering all potential splits for each leaf (Chen and Guestrin, 2016). Instead of doing so, it looks at the distributions of features across all data points in a leaf and reduces the search space of possible future splits – speeding up the process by a large amount.

2.1.5 Regularization

Regularization techniques are commonly used in machine learning to reduce overfitting (reduce variance in the estimator) by introducing a penalty term to the loss function that the model tries to optimize during the training phase – the idea is to introduce the concept of complexity cost in the model. In fact, unless instructed otherwise, the model will always fit the training data as best as possible – thus possibly leading to a model that is too biased by the training data and poorly predicts any new information. As such, these techniques help to constrain the fitting procedure and balance the predictive performance of the resulting model – as observed by Sutton (2005) and Zhang and Yu (2005).

Furthermore, it may be necessary to have a simple yet interpretable model and, thus, a measure to prevent over-complexity is needed – this follows the widely known problem-solving principle 'Occam's Razor', which states the idea that when presented with competing hypothesis about the same prediction, one should prefer the one that requires fewest assumptions.

Some regularization techniques such as subsampling, shrinkage and early stopping will be further discussed ahead.

2.1.5.1 Subsampling

Subsampling procedures have shown to improve the generalization properties of the model (Sutton, 2005). The idea behind this method is to introduce some randomness into the fitting procedure by selecting only a portion of data, in each iteration, for training (Natekin and Knoll, 2013).

Thus, a parameter called "bag fraction" needs to be determined – it states the portion of data, at each iteration, that will be used for training. It is important to wisely chose the value of this parameter, as a very low value may lead to a poorly fit model due to the lack of degrees of freedom. On the other hand, increasing this value decreases the impact of regularization – increasing the overfitting probability of the model. This parameter can be optimized using a comparative procedure. Another major advantage of using a procedure like subsampling is that there is no need to use enormous amounts of data at once, due to the sampling procedure through a bag fraction.

2.1.5.2 Shrinkage

Shrinkage is a technique used to reduce the impact of each additional learner in the final model (Natekin and Knoll, 2013). It is based on the idea that it is better to improve a model by taking many small steps rather than fewer large steps. Thus, if one of the iterations turns out to be erroneous, its impact overall is diminished.

Assuming a regression model with the following mathematical formulation:

$$Y = \beta_0 + \beta_1 * x_1 + ... + \beta_n * x_n + \varepsilon \tag{2.4}$$

Where Y is the dependent variable, $x \in [x_1, \ldots, x_n]$ represents the independent variables used to predict Y, $\beta \in [\beta_1, \ldots, \beta_n]$ are the coefficients to be determined and ε is the estimation error. In this setting, the mean squared error can be given by the following expression (bear in mind that, due to being out of the scope of this thesis, the intermediate steps are not presented here):

$$MSE[\beta_j] = E[(\hat{\beta}_j - \beta_j)^2] = \underbrace{(E[\hat{\beta}_j - \beta_j])^2}_{\text{Bias}^2} + \underbrace{Var[\beta_j]}_{\text{Variability}} \tag{2.5}$$

As such, there's a clear interplay between variance and bias, where increasing one leads to decreasing the other. The shrinkage methods draws on this conclusion and pursues the idea of adding an amount of smart bias in order to reduce its variance (García-Portugués, 2023). The two main methods utilized in machine learning are L1 and L2 regularization – briefly covered next.

2.1.5.3 L1 and L2 Regularization

An easy way to understand both L1 and L2 regularization techniques is to first consider the mathematical formulation of the residual squared errors, given by:

$$RSS(\beta) = \sum_{i=1}^{n} (Y_i - \beta_0 - \beta_1 * x_{i_1} - ... - \beta_n * x_{i_n})^2 \tag{2.6}$$

L1 regularization, also known as lasso regression, simply adds a penalty term to the cost function of the model that is proportional to the absolute values of the model coefficients. Mathematically, it can be expressed as:

$$RSS(\beta) + \lambda * \sum_{j=1}^{p} |\beta_j| = RSS(\beta) + \lambda |\beta_{-1}| \tag{2.7}$$

The main effect of lasso regularization is that it forces some coefficients to be 0, which can be particularly useful when the dataset has many irrelevant features – it is commonly used for feature selection.

On the other hand, L2 regularization, or ridge regression, adds a penalty to the cost function that is proportional to the square of the model coefficients – thus shrinking the magnitude of the coefficients towards zero, although not exactly zero. This results in using all features but assigning smaller weights to less impactful ones. Ridge regression is often used when the data

set has collinear/codependent features, since these properties tend to increase coefficient variance, making coefficients less reliable and hurting model generability.

Mathematically,

$$RSS(\beta) + \lambda * \sum_{i=1}^{p} \beta^2 j = RSS(\beta) + \lambda(\beta_{-1})^2 \tag{2.8}$$

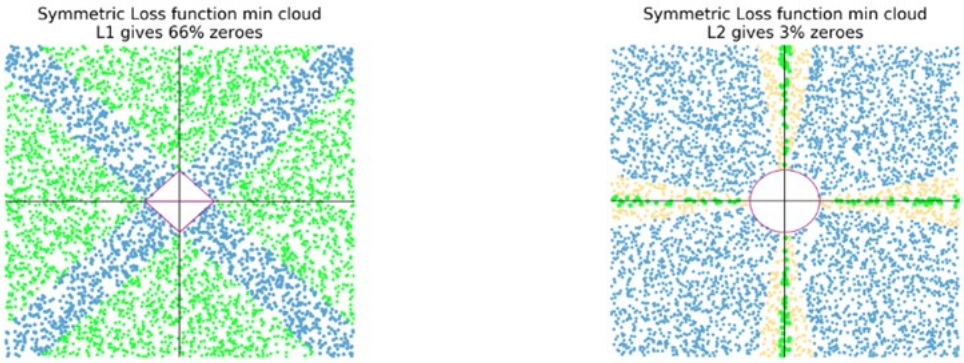


Figure 2.4: Illustrative example of the difference between L1 and L2 regularization techniques, adapted from Parr.

To better understand these differences, let's consider the Figure 2.4 above. Green dots represent a random loss function that resulted in a regularized coefficient being zero. Blue represents a random loss function where no regularized coefficient was zero. Orange represents loss functions in L2 plots that had at least one coefficient close to zero (within 10% of the maximum distance of any coefficient pair). As one may see through the illustrative examples, L1 regression tends to determine as zeros the coefficients in the zone perpendicular to the diamond edges and not to give any near zero coefficient (represented by the orange dots). On the other hand, L2 regression classifies as zeros the coefficients that are on or very close to the axis, clearly constraining coefficients close to zero as we move close to the axis.

2.1.5.4 Early Stopping

As the name indicates, early stopping refers to determining the number of validation set performance iterations, which can be different from the shrinkage-related number of iterations.

Thus, one may benefit from trimming the number of trees at the point corresponding to the validation set minima of the error curve, preventing overfitting and minimizing accuracy costs. One practical consideration is that the number of boosts at which the early stopping should be considered varies with the shrinkage parameter considered. This idea is easily understandable by observing the following Figure 2.5:

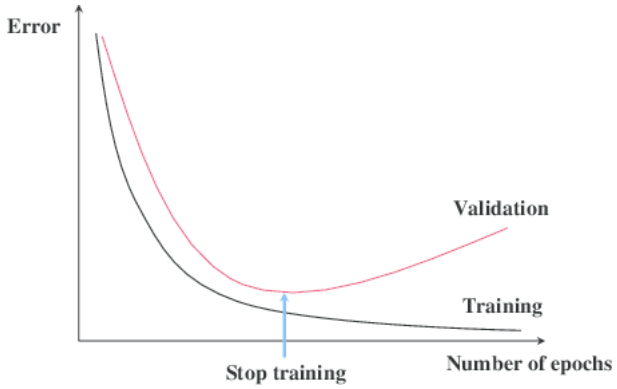


Figure 2.5: Visualization of the early stopping technique, adapted from Shin et al. (2016).

2.1.6 Performance Evaluation

The effectiveness of a predictive model can be compared using various performance metrics such as confusion matrix, accuracy, precision, recall, F1, F2, MCC or even AUC and Lift – all of them will be discussed below. It is important to understand that choosing one alternative rather than the other must be based on both the user's goals and data characteristics – this will become clearer with the detailed explanation of each.

2.1.6.1 Confusion Matrix

The confusion matrix, for a binary problem, is a table with two dimensions: actual and predicted, both divided into two different classes. Each value represents the number of instances regarding a combination of classes. They are very useful to understand whether the model is correctly attributing classes. Figure 2.6 visually represents what has been discussed.

By observing the matrix, one may conclude the following:

- TP (True Positives) – When the predicted classification was positive and corresponds to the actual classification;
- FP (False Positives) – When the predicted classification was positive but does not correspond to the actual classification;
- TN (True Negatives) – When the predicted classification was negative and corresponds to the actual classification;
- FN (False Negatives) – When the predicted classification was negative but does not correspond to the actual classification.

For anyone familiar with statistics, false positives are usually classified as type I errors, whilst false negatives are usually classified as type II errors.

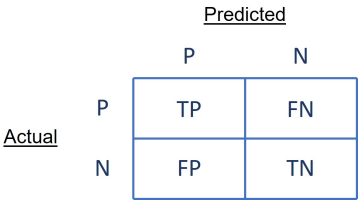


Figure 2.6: Visual representation of a confusion matrix.

2.1.6.2 Accuracy

Accuracy provides general information about the share of misclassified objects. Thus, accuracy can be given as the sum of total correct classifications over the total number of classifications. Mathematically:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{2.9}$$

2.1.6.3 Precision and Recall

Precision and Recall are metrics that are related to exactness and completeness, respectively. Precision states the fraction of relevant predictions (true positives) out of all items predicted as relevant (true positive plus false positives). Recall determines the fraction of relevant items out of all true relevant items (true positives plus false negatives). Mathematically,

$$\text{Precision} = \frac{TP}{TP+FP} \tag{2.10}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{2.11}$$

2.1.6.4 F1 Score

The F1 Score is a metric given by an harmonic mean between precision and recall. Thus, it provides a measure for how well a binary classifier can classify positive cases – with values ranging from 0 (either precision or recall, or both, are very low) and 1 (precision and recall are perfect).

$$\text{F1 Score} = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{2.12}$$

2.1.6.5 F2 Score

Unlike the F1 Score, precision and recall are not weighted the same – the F2 Score gives more weight to recall (penalizing the model more for false negatives than for false positives). If one were to think about diagnosing a patient, this metric would probably suit right, since not predicting that a patient is ill when he actually is seems to be worse than prescribing medicine for someone who

is not actually ill.

$$\text{F2 Score} = 5 * \frac{Precision * Recall}{4 * Precision + Recall} \tag{2.13}$$

2.1.6.6 Matthews Correlation Coefficient

Matthews Correlation Coefficient (MCC), first formulated by Brian Matthews in 1975, is a special metric that can be used for binary classification and has underlined a special case of a linear correlation coefficient setting.

Its value ranges from -1 (complete disagreement between the prediction and the actual outcome) and 1 (perfect prediction). A value of 0 denotes a perfectly random prediction.

Mathematically,

$$\text{MCC} = \frac{TP * TN - FP * FN}{\sqrt{(TP+TN)(TP+FN)(TN+FP)(TN+FN)}} \tag{2.14}$$

MCC is a very useful metric specially for dealing with imbalanced data, where one of the classes has a much larger sample than the other (which happens in the case described in this thesis).

2.1.6.7 Receiver Operator Characteristic Curve and AUC

The Receiver Operator Characteristic graphs are tools that measure how sensitivity (true positives rate) changes with varying specificity (false positives rate), at different classification thresholds. A random classification would see a diagonal line crossing this graph, as one may observe in Figure 2.7.

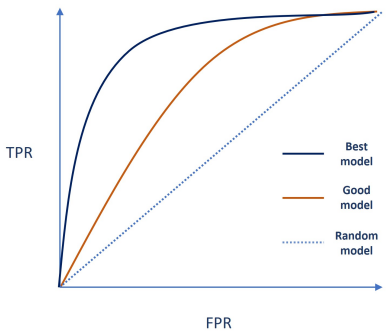


Figure 2.7: Illustrative example of the Receiver Operator Curve.

To compute the points in an ROC curve, we could perform multiple logistic regressions with different classification thresholds. However, that would be inefficient. To tackle this, the used measure is the AUC (Area Under the ROC Curve), which gives an idea of the performance of the classification model – by providing an aggregate measure of how well the model ranks a random positive example higher than a random negative example. A value of 1 means perfect predictions, whilst a value of 0 states all predictions as wrong. It is usually calculated using the trapezoidal

method – geometrical method based on linear interpolation between each point of the ROC curve – or, even simpler, using the balanced accuracy approximation (0.5 * [sensitivity + specificity]).

AUC is a strong metric since it is scale invariant – it measures how well predictions are ranked, rather than their absolute values – and is threshold invariant. It is important to note, however, that one may not want a metric to be either scale invariant (because one wishes to know how accurate is the classification) nor threshold invariant (in case of high disparity between false positives and false negatives' cost).

For imbalanced datasets AUC is also not the best metric – since a high number of true negatives can overshadow the effects of changes in other metrics like false positives.

2.1.6.8 AUCPR

Unlike AUC, the AUCPR calculates the Area Under the Precision-Recall Curve. Since this curve does not care about true negatives, the AUCPR will be much more sensitive to TP, FP and FN and is a better metric for imbalanced data.

2.1.6.9 Cumulative Gains Curve and Lift Chart

The Cumulative Gains Chart (CGC) represents the share of positive observations (Y-axis) with respect to the cumulative percentage of the sample (X-axis), as one may see in the Figure 2.8. Thus, it states the ability of the model to rank observations according to their predicted probability of belonging to the positive classes.

The Lift Curve is given by calculating, for each percentage of the total population, the percentage of positive values predicted with respect to the outcome variable. On the other hand, the Lift Chart can be derived from the cumulative gains chart by calculating, for each percentage of the total population, the lift value. Thus, lift is a measure of the effectiveness of the predictive model, calculated as the ratio of the results obtained with and without the model.

Both the CGC and Lift Curve can be used to evaluate the performance of the model across different deciles or percentiles of the sample. The curves allow us to compare the performance of different models or different parameters of the same model, and to identify the point at which the marginal gain of adding more observations to the model begins to diminish. A steeper curve indicates a better model performance, and the point of maximum separation between the curves and the baseline indicates the optimal point at which to set the classification threshold.

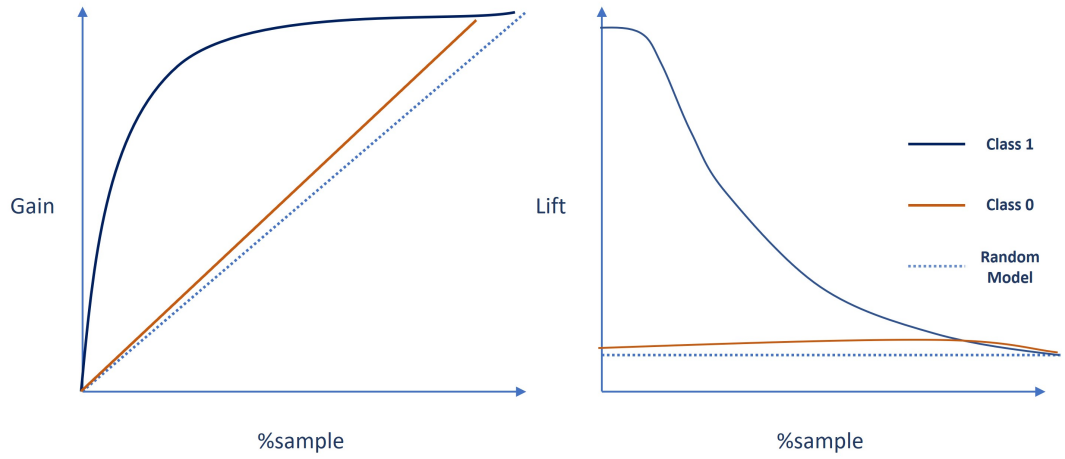


Figure 2.8: Cumulative Gains Curve (left) and Lift Chart derived from it (right).

2.1.7 Hyperparameter Tuning

Most machine learning algorithms have a set of parameters that control their inner functions, which are known as hyperparameters – the weights attributed to each input variable in the Neural Networks are one example of hyperparameters. Random forests also have hyperparameters such as the number of trees, maximum depth or even the stopping criteria, as previously discussed.

In practice, it is necessary to select these parameters such that the error of the prediction is minimized – a current technique is to optimize via MSE (Minimum Squared Error) minimization. However, despite its importance, there seems to be no efficient method for performing such task – thus leading practitioners to prefer a trial search instead, as stated per Bergstra and Bengio (2012).

One very common technique is to use grid search, which comprises the idea of training the model exhaustively for every combination of the hyperparameter values. However, according to Bergstra and Bengio it is possible to obtain similar or better results by simply using a random search rather than a grid search, minimizing the computation time.

Successive halving, a technique used to speed up the grid search process, can also be used. The idea is to utilize a subset of data early on to find the best performing parameters and increasing the data subsequentially as it narrows on the best combinations.

2.1.8 Cross-validation

For comparative purposes, it is important to have a reliable measure of the predictive performance of each model. The accuracy of each estimator should be given by the average error across a set of randomly selected samples – assuming the data distribution is the same across samples. It is also important to note that, while comparing different models, the absolute accuracy is less important and one is willing to trade bias for variance, *i.e.*, assuming bias affects all models the same, to select the best model only variance across samples matters, as per Kohavi (1995).

Literature also points to the idea that increasing computational cost is not always relevant, specially when comparing models – as concluded by Efron (1983), since the leave-one-out model, although unbiased, presented high variance, leading to unreliable results.

The holdout method simply partitions the data in two subsets – training and testing sets. The holdout method can be seen as a pessimistic one, if one were to assume that the inducer's (training set) accuracy would increase given more instances. However, increasing the size of the inducer reduces the size of the testing set, increasing the size of the confidence intervals. Usually a given dataset is distributed in a 70/30 manner – with 70% of data used for training and 30% used for testing.

Another common re-sampling method is the k-fold cross-validation, where the available learning set is partitioned into k subsets (folds) of approximately the same size. This is done by randomly selecting subsets of data without replacement. Then, the model is trained using k-1 folds (training set) and is then applied to the remaining subset (testing set). The procedure is repeated until all folds have been used as testing set.

Cross-validation often involves stratified random sampling, which states the idea of ensuring that the class proportions of the individual subsets reflect the proportions of the learning set.

2.2 Literature Review

The objective of this section is to present a comprehensive overview of the current state-of-the-art in both churn and uplift concepts. Accordingly, it will delve into the literature to highlight key findings related to each of these subjects. The inclusion of these topics is vital in establishing a shared understanding and providing a solid foundation for the research conducted in this dissertation. The conclusion of this section involves a discussion on the contribution of the current project to the existing literature. It highlights how the research conducted in this project fills a gap or adds value to the current knowledge in the field.

2.2.1 Churn

In contemporary and competitive environments, customer loyalty plays a pivotal role in determining a firm's success, as observed by Smith and Wright (2004) in the PC industry and in UK retail by Ramanathan et al. (2017). Customer retention is directly linked to customer loyalty, as demonstrated by Singh (2006). It is well known that it is more profitable for firms to retain and satisfy existing customers than to constantly attract new ones, as per Reinartz and Kumar (2003). This is due to several reasons: (1) profitable firms tend to maintain long-term relationships with customers to focus on customer service rather than acquisition (Reinartz and Kumar, 2003); (2) lost customers may influence others to do the same, as observed by Nitzan and Libai (2011) and (3) long-term customers have both profit advantages (tendency to buy more and recommend the company to others) and cost advantages (less service cost since the company already has knowledge about the customer).

Thus, predicting and preventing customer churn, defined as the premature termination of the customer-firm relationship, becomes crucial for the success of the firm. Research shows that a small change in the retention rate may lead to significant changes in contribution, as per Van den Poel and Larivière (2004). The literature also highlights three types of churn: deliberate (the customer voluntarily decides to terminate the relationship), incidental (the customer cannot sustain the relationship), and passive (the firm discontinues the contract).

Churn prediction and modeling have been tackled by various researchers, utilizing the most varied techniques, with telecommunications emerging as the primary area of research. Across literature, models such as Decision Trees, Logistic Regression or Random Forest emerge as the most common ones. For instance, Zhang et al. (2010) used Decision Trees, Logistic Regressions and Neural Networks to predict customer churn, achieving a 90% lift rate. Additionally, Owczarczuk (2010) constructed churn models for prepaid customers in a polish cellular telecommunication company, with a dataset of 1381 potential variables and dealing with pre-paid clients rather than the usual post-paid, which are deemed to be more volatile and prone to churn. He concluded that linear models are more stable than Decision Trees, since these get old quickly and their performance decreases over time. Khodabandehlou and Rahman (2017) also compared different churn prediction models, including DT, NN and SVM and pointed to Decision Trees as the worst overall estimator.

Sequential Patterns (Chiang et al., 2003) and Genetic Modeling (Eiben et al., 1998) have also been used to tackle this problem. Similarly, a study by Galar et al. (2012) used ensemble techniques to predict customer churn. The study found that combining multiple models improved the accuracy of churn prediction. More recent approaches involve machine learning-based models such as GBM, XGBoost, or deep learning models. However, there is no consensus on the performance of churn prediction modelling techniques, given that results differ between industries as well as data mining techniques applied (Verbeke et al., 2012).

Despite the significance of churn prediction in business operations, limited research has been conducted on churn prediction in the newspaper industry. Notably, Coussement and den Poel (2008) utilized SVM models to predict subscription terminations and determined that parameter optimization is essential for achieving desirable performance, although Random Forest demonstrated superior performance in both scenarios. Another noteworthy investigation by Ballings and Poel (2012) on a newspaper case revealed that nearly 69% of data added no predictive performance after five years of additional information regarding clients who churned. This indicates that, for effective churn prediction modeling, an extensive amount of data is not necessarily required and may potentially reduce data-related burdens. Finally, Das et al. (2015) implemented a maximum entropy-based model to identify the most dominant features from user complains and web data for churn prediction – with temporal, unigram and status history models being clear winners.

The lack of literature regarding churn prediction in the newspaper industry may be mitigated by assuming the business model as a subscription-based one and accessing the corresponding literature. This section of the text presents a brief overview of some key findings in this field.

Coussement and den Poel (2009) found that incorporating emotionality indicators in the model

using textual data (e-mails) enhances predictive performance. They also generalized that exploring and adding new types of customer information and choosing the appropriate classification technique can improve overall churn estimation performance. They tested three different models – Logistic Regression, Support Vector Machines, and Random Forest – and determined that the latter was the most effective.

Pendharkar (2009) used a Maximum Likelihood Neural Network (MLNN) model combined with a genetic algorithm to determine the weights for each layer to calculate churn in cellular wireless services. The results of the experiment proved the model to be more accurate than the standard z-score model.

Ascarza and Hardie (2013) proposed a model of usage and churn behaviors in contractual settings, in which both behaviors reflected a dynamic latent variable and using an underlined hidden Markov process – obtaining accurate multiperiod forecasts.

Vadakattu et al. (2015) presented a churn prediction model for large-scale subscription-based businesses and introduced a ranking customer algorithm to categorize potential churners as high risk, medium risk, or low risk, enabling marketing teams to take action and improve customer retention.

Other two important conclusions gathered from literature in the subscription business industry are the following: (1) after applying several classification methods – such as linear, non-linear, Neural Networks and ensemble methods – the general conclusion was that of ensemble methods being overall superior to the other ones (Bogaert and Delaere, 2023) and (2) in some cases deep learning may not be advantageous, as per Saias et al. (2022) – supporting the idea that there is no consensus regarding the best classification techniques for churn prediction.

Additionally, subscription-based churn modelling was also applied to online multiplayer games, with Hossain et al. (2023) suggesting that both Random Forest and Decision Tree algorithms provide over 90% accuracy and are robust to different scenarios.

Finally, new emerging trends related to churn prediction indicate the following: (1) The use of uplift models instead of churn prediction models is more advantageous, as uplift models take into account the anticipated response of specific customers to personalized marketing effort, as per Devriendt et al. (2021), and (2) retention strategies may prove ineffective for customers with the highest likelihood of churning, as some of them may be deemed as ‘lost causes’ (Ascarza, 2018).

2.2.2 Churn Prediction Related Problems

This subsection will cover two of the most common problems related to classification tasks regarding churn prediction: imbalanced datasets and the ability to capture dynamic variables. Although many other authors tried to mitigate the impact of these problems, no definite solution has been found.

2.2.2.1 Imbalanced datasets

Classification problems often encounter imbalanced datasets, where the majority of observations belong to one class, usually the less relevant one. This can lead the estimator to overfit on information about the majority class. To address this issue, various techniques have been proposed, grouped into pre-processing methods, cost-sensitive learning, algorithm-centered approaches, hybrid methods, and learning algorithms (Kaur et al., 2019).

Regarding pre-processing methods, the most common ones are random undersampling and oversampling (usually referred to as sampling methods). Undersampling refers mainly to a non-heuristic method that balances classes through randomly eliminating examples from the majority class. This method has, however, some disadvantages: (1) it can potentially discard useful data for the predictor and (2) since the main purpose of the estimator is to determine the probability distribution of the population, undersampling forces one not to consider the sample random anymore, thus violating the main assumption behind determining the population distribution through a random sample. Oversampling is another non-heuristic method that balances classes through the replication of examples from the minority class. Its main problem is related to increasing the likelihood of occurring overfitting, as many authors point out (Kubat, 2000). Another possible technique is SMOTE, which, although alike the previous one in many aspects, tackles this problem by generating minority examples using the k-nearest neighbors method.

Cost-sensitive learning assigns different costs to each class and aims to minimize the total cost when predicting the class of an observation. Defining these costs can be challenging, as per Krawczyk et al. (2014).

Algorithm-centered approaches use techniques like cluster-based undersampling, bagging, boosting or cost-sensitive SVM. We will only discuss boosting and gradient boosting since they are a pivotal part of the machine learning techniques used. Boosting is an ensemble technique that iteratively trains a sequence of weak learners on the same data, assigns weights to the training points, and combines the weak learners to form a strong learner. Gradient boosting, a specific type of boosting, uses gradient descent optimization to minimize the loss function of the ensemble. Weak learners are typically Decision Trees, and a new tree is trained to minimize the residual error of the previous tree at each iteration, until the loss function reaches a minimum. Hybrid methods combine pre-processing and algorithm-centered approaches.

Research in methods for handling imbalanced data in churn prediction point to the idea that cost-sensitive learning methods have better performance than sampling methods (Nguyen and Duong, 2021). Therefore, in the context of this dissertation, a simulation of a cost-sensitive approach was conducted by selecting performance metrics that consider different costs, namely the F2 metric.

2.2.2.2 Single-time-slice vs multiple-time-slices

One of the challenges in churn prediction is to capture the dynamics of customer behavior over time. Customer behavior is not static and can change over time due to various factors such as

changes in their needs, preferences, or external circumstances. For example, a customer who has been loyal for several years may suddenly start using the company's services less frequently or stop using them altogether due to a new competitor entering the market or a change in their personal circumstances.

Training a churn prediction model on a single time slice may not capture these dynamics of customer behavior over time. Instead, a temporal approach that considers multiple time periods in the training data is recommended. This approach involves dividing the customer data into several time slices or periods, and then training the model on each time period to capture the trends and patterns in customer behavior over time.

By using a temporal approach in churn prediction, it is possible to capture the dynamics of customer behavior over time and improve the accuracy of churn prediction. This, in turn, can help companies take proactive measures to retain customers and minimize revenue loss.

This dissertation will use a sequential multi-slicing technique, that proved good results, as per Gattermann-Itschert and Thonemann (2021). The main idea is to divide the dataset in different time slices, each representing a different instant, and then train separate models on each time slice, using the following time slice to test. Afterwards, the predictions of each model are combined, as represented in Figure 2.9. At each iteration, the data used to train cannot be the same used to test.

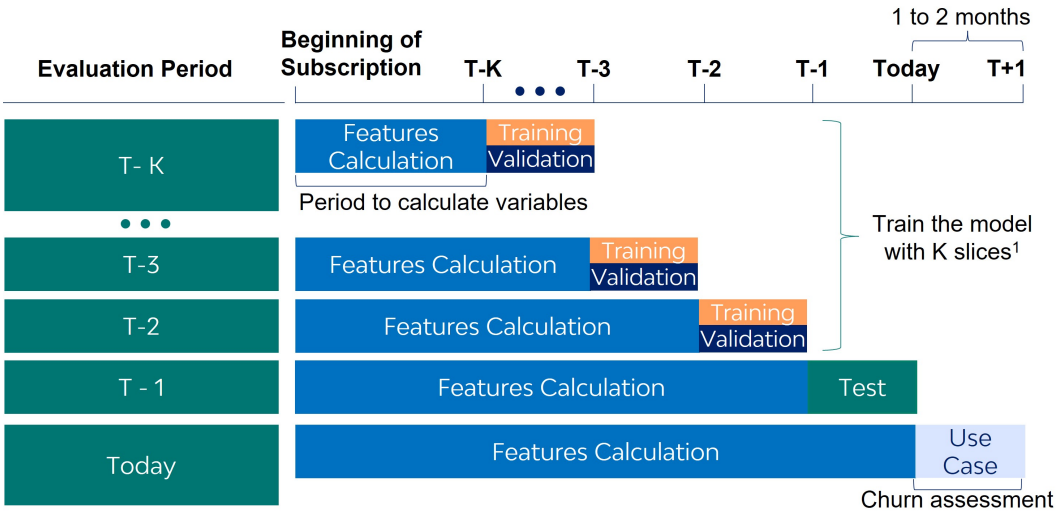


Figure 2.9: Visual explanation of the multiple-time-slices technique.

2.2.3 Uplift Models

Churn prediction modelling tries to forecast the set of customers whose likelihood of churning is the highest. Hence, these models only tackle the gross outcome, *i.e.*, they focus on determining whether a given customer will belong to the churn or non-churn class. The major problem with this type of approach is the misalignment between the model's objective and the business intent of retaining customers whilst spending the least amount of resources possible. In fact, just because

a given customer has a high chance of cancelling the subscription, it does not mean a marketing effort will change the outcome.

With that in mind, if a firm wants to be competitive and aims at achieving the maximum profit – and it is no bold assumption that every company strives to do so – choosing the right customers to tackle becomes a crucial aspect of customer retention.

An additional problem is that traditional customer churn prediction models are subject to feedback loops, as per Sculley et al. (2015). This is easily explainable if one were to imagine a scenario where a first churn prediction model is constructed and then a second one, based on the feedback of the former, is used to better target the most sensitive-to-promotion customers. In fact, and as perfectly pointed out by Devriendt et al. (2021), one of the following may happen:

- Customers observed to churn in the first model may either be: (1) customers who have been predicted to churn and the campaign was not effective or (2) customers who were not targeted in the first model and, thus, are difficult to predict. So, one can infer that the next model will try to tackle both 'lost causes' or very hard to predict customers.

- Customers observed not to churn in the first model may either be: (1) those who where never about to churn or (2) customers who where about to churn but where correctly predicted and the campaign was effective. However, because this customers are now labeled as 'non-churners', the next model will not focus on them.

Therefore, uplift models try to tackle the shortcoming of churn modelling, by aiming at the net effect rather than the gross outcome – they establish the difference in customer behavior between a control group (customers who didn't suffer from a specific treatment) and a treatment group (customers affected by a specific treatment – for instance, a promotional campaign).

There are several uplift models currently in use – this dissertation will discuss two of them. However, to grasp each of the techniques, it is important to understand the four customer categories that these type of models consider – they can be seen in Figure 2.10. Lost Causes are customers that will churn nonetheless, regardless of the treatment applied – tackling these customers will result in a waste of resources. The same conclusion can be said about Sure Things, although due to different reasons – they will always stay in the company. On the other hand, Do Not Disturbs are a class of customers whose treatment effect is negative, *i.e.*, targeting these customers will cause them to churn, despite the fact they initially where not going to cancel the subscription – this is because retention efforts may lead customers to churn, as per Lo (2002) – for example, it may cause the customer to remind himself of the imminent expiration of a contract. The aim of an uplift model is, then, to predict the Persuadables class – those whose retention strategy has the highest positive effect.

One of the most used techniques due to its simplicity is to define a model for the control group and a different one for the treatment group and then check the difference. Mathematically:

$$M_U = M_T - M_C \tag{2.15}$$

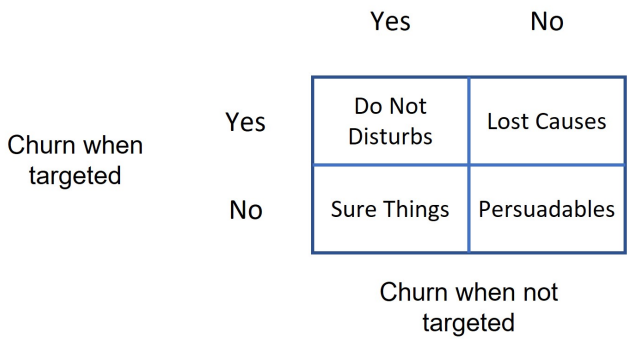


Figure 2.10: Visual representation of the uplift matrix.

The main disadvantage of this technique is that, by constructing two models independently, they are not necessarily consistent in terms of predictors and the errors of independent estimates may sum up, leading to significant resulting errors in uplift estimates.

Alternatively, one can use a single framework and model directly through the usage of tree-based approaches. This method differentiates himself due to finding the best split at each node by considering both the user's behavior and treatment effect, thus tackling the problems already discussed associated with building two models.

2.3 Contribution

Upon conducting an extensive review of the background and state-of-the-art literature, it is evident that a comprehensive assessment of the significance of this research is warranted. While it is true that the topic of churn has received considerable attention in the literature, with numerous models and implementations dedicated to understanding customer behavior in relation to subscription cancellations, this work holds particular importance. It not only analyzes diverse models within the premature media sector, shedding light on the factors influencing post-paid media subscriptions, but also contributes by evaluating the impact of two distinct feature selection approaches in this specific context. One of the methodologies employed in this study introduces a pioneering algorithm that combines the principles of Pareto-front and multi-objective approaches. This novel algorithm has been designated as Pareto-Based Multi-objective Feature Selection (PMFS), signifying its distinctiveness and unique characteristics. These approaches aim to reduce the complexity and dimensionality of the problem while minimizing implications for the results. Furthermore, this work is notable for presenting a practical application of an uplift model for churn prevention, which remains relatively under-explored in the existing literature. Lastly, the study elucidates potential retention measures and their consequential impact, bridging the gap between theoretical models and real-world implementation – an aspect that has been largely unaddressed by scholars.

Subsequent to the completion of this dissertation, two papers were submitted. The first paper pertains to the novel community-based feature selection algorithm and is titled "A Novel Feature Selection Algorithm Based on Pareto-Front and Multi-objective Approaches". It was submitted

to the IEEE Transactions on Knowledge and Data Engineering. The second paper focuses on the insights derived from customer behavior analysis and is titled "Unveiling Subscription Churn: A Machine Learning Analysis of Customer Behavior in the Media Industry". This paper was submitted to the European Journal of Marketing.

Chapter 3

Context and Methodology

This chapter provides a comprehensive overview of the context in which this dissertation was developed, both at a macro and micro level, and the framework that facilitated the achievement of the research objectives and subsequent results. It begins by delving into the Portuguese market and its current state, along with the positioning of the firm involved. Subsequently, the chapter outlines the structure of the project and the objectives of the newspaper. Additionally, a quantitative assessment of the potential impact of the project is presented. The Methodology section is divided into two subsections: Data Management and Churn and Uplift Modelling. Each subsection elucidates the decision-making process and provides a rationale for the choices made at each stage.

3.1 Problem Description

This study was conducted in collaboration with LTPlabs, a firm specialized in advanced analytics and business consultancy. The objective of this project is to apply customer analytics in the context of a news outlet firm, with the aim of predicting potential churners and implementing retention measures to counteract this tendency, thereby producing significant impact.

This thesis focuses on the loyalty and retention steps of the customer journey, striving to prevent churners and increase the average customer lifespan – thus increasing, consequently, customer's Lifetime Value (LTV).

3.1.1 The Portuguese Online News Market

In a market that ranks high in terms of trust in news, where more than 60% of people state to trust in the overall news they see (Cardoso et al., 2021), several newspapers emerge as prominent entities in terms of circulation, specifically Expresso, Público, Correio da Manhã, and Jornal de Notícias, among others (Statista, 2022).

News is the most sought-after type of content by Portuguese consumers, with almost 90% of people admitting to watch it. By gender, men are the most likely to consume news, whilst also consuming series or documentaries, whereas women are the largest consumers of entertainment

shows or soap operas. When accounting for the viewer's age, only the segment between 15-24 doesn't place news at the top, being surpassed by TV series.

However, when considering overall media consumption, newspapers rank third in terms of popularity, trailing behind social networks, with television being the primary source of information. Notably, Portuguese consumers tend to access information either in the early morning, early evening, or late at night, indicating alignment with the Portuguese lifestyle. Moreover, television is the most commonly used device, followed closely by the computer.

In terms of video and consumption formats, most of the Portuguese consumers (around 80%) access news via the list of headlines on news site homepages and are willing to read longer articles (65%). Impartiality is highly valued, with 75% of respondents indicating a preference for journalists to present varying viewpoints.

Another interesting trend that has been observed, and which the COVID-19 pandemic potentiated, was the increasing consumption of online news in prejudice of offline consumption. In fact, according to Reuters (Cardoso et al., 2021), across the first quarter of the pandemic, paid circulation fell by nearly 20% (and never truly recovered). However, the digital news subscriptions followed a completely inverse trend, with an overall increase of digital paid circulation, albeit still at low values.

In terms of future willingness to pay for online news access, more than three-quarters of respondents indicated a high level of skepticism – thus, one may conclude that the Portuguese consumers are not favorable to online news subscriptions or paywalls. Cardoso et al. (2021) concluded that only 17% of consumers do pay for online news. The Portuguese people have also shown to be some of the most concerned with the financial state of the news organizations and were one of the few countries to support government intervention in the media industry (Cardoso et al., 2021) to help the organizations who couldn't survive on their own.

3.1.2 The Company

The company where the project was developed is a major news outlet in Portugal and was the first Portuguese newspaper to have an online edition.

With the rise of digitalization and customers increasingly consuming news through the internet and social media, the company had to adapt to the changing landscape by increasing its online presence. This was achieved through the creation of an online edition and the adoption of new popular media channels such as podcasts and social media. To promote conversion, a paywall method was also introduced, along with techniques such as increased reader engagement through exclusive subscriber content, personalized offers based on customer characteristics, and discounts.

Currently, the company does not utilize any churn prediction methods to detect the most likely customers to discontinue their subscription early. In terms of prevention measures, the most common practice is to send a friendly e-mail reminder of the approaching payment deadline, which may not be sufficient to retain the most indecisive customers.

However, the company uses the analytics software PIANO which allegedly empowers firms to better understand customer behavior through the usage of data – thus tracking relevant metrics

such as web-site clicks or user's articles' history. This software provides an interesting metric – Likelihood to Churn (LtC) which aims at predicting the probability of each customer of ceasing the contract, *i.e.*, of churning. Nevertheless, a quick look at the algorithm allowed to conclude that the metric was off – it was not being well calculated due to some inconsistencies in data tracking and the variables used were a black box – the software did not specify most of them. Therefore, we decided not to use it as a reasonable benchmark.

3.1.3 Project Structure and Newspaper's Objectives

The primary objective of this dissertation was to provide the newspaper company with a churn prediction model and effective churn retention measures. The churn prediction model aimed to assist the commercial team in identifying customers most likely to cancel their subscription, thereby narrowing down the target audience for retention campaigns. The restructuring of the churn retention measures aimed to enhance the company's ability to retain these identified customers. Furthermore, the uplift model aimed to identify the correct churners, specifically those with a high probability of churn who would positively respond to the company's campaigns, thereby reducing resource wastage on futile efforts.

The initial proposal contemplated two main deliverables (the uplift model was an add-on): (1) churn prediction tool and (2) a list of possible new retention measures given the most important features and the outlined customer behavior.

This thesis culminates at a critical juncture, where the focus lies on comprehending the actual impact of the developed model. Methodologies have been established to assess the model's accuracy in predicting forthcoming churners and their response to retention campaigns. To facilitate this assessment, two distinct groups have been formed: (1) a control group consisting of customers deemed likely to churn but excluded from receiving any retention intervention, and (2) a treatment group comprising customers categorized as likely to churn and targeted with retention offers. Through this procedure, the organization can evaluate the performance of the churn prediction model while concurrently gathering fresh data to enhance the model's predictive accuracy. Moreover, this data will be utilized in the uplift modeling process to enhance its capability in identifying the persuadable segment, namely the actionable churners. The ultimate aim is to develop a fully operational tool by 2024 with the objective of increasing the company's anual revenue by a quarter of a million euros, as can be seen in the equation below.

$$\text{LTV} = 12 \text{ (number of months)} * 6 \text{ (subscription cost)} = 72€ \tag{3.1}$$

$$\text{Revenue} = \text{LTV} * 20000 \text{ (number of subscribers)} * 0.17 \text{ (share of churners)} = 244.800€ \tag{3.2}$$

It is important to emphasize that these calculations are intended to provide an estimation of the overall potential impact resulting from the implementation of an accurate retention plan and

churn prediction model. The following assumptions have been made:

- Each customer typically subscribes for a full year, and by successfully retaining a customer, their subscription is extended for another year.
- The monthly subscription price is fixed at 6€ and is not expected to change in the upcoming years.
- The current churn rate stands at approximately 17% (although churn data points account for only 5%) and is projected to remain unchanged in the future.
- The current subscriber count is approximately 20,000 and is anticipated to remain stable in the forthcoming years.
- The marginal cost of adding a new article is close to zero.

3.2 Methodology

This section briefly outlines the methodological approach used during the project, delving into the core stages of the process: Data Management and Churn and Uplift Modelling – both claiming their own section in this chapter. The Data Management section describes the essential techniques implemented to preprocess the dataset in order to facilitate its use in machine learning models. The Churn and Uplift section outlines the model-related decisions that were made to achieve accurate churn prediction results, whilst also elucidating about the process of transforming a churn prediction model into a business-oriented model that can effectively predict the impact of marketing efforts. A graphical representation of the implemented methodology is presented in the Figure A.3, in the appendixes.

3.2.1 Data Management

A data pipeline refers to the idea of moving data from one place (the source) to another (the destination). In between, a series of transformations may be applied, such that data is manipulated and optimized – thus making it easier to be used at the final stage.

ETL – which stands for Extract, Transform and Load – is a kind of data pipeline that sees data being gathered from the sources, modified in the intermediate processing steps and then shared at the destination – the process can be seen in the Figure 3.1.

The methodology followed to preprocess the dataset in this thesis followed a similar guideline to that just presented. Initially, the information was retrieved from a database and underwent a two-phase transformation process that included Data Manipulation and Feature Selection. Data Manipulation involved cleaning, transforming, and reformatting the raw data to make it more suitable for analysis. The Feature Selection phase involved selecting a subset of relevant variables from the cleaned dataset that would be used for analysis. This was done to reduce dimensionality and improve model performance. This section briefly explains all these steps and the results are presented in Section 4.

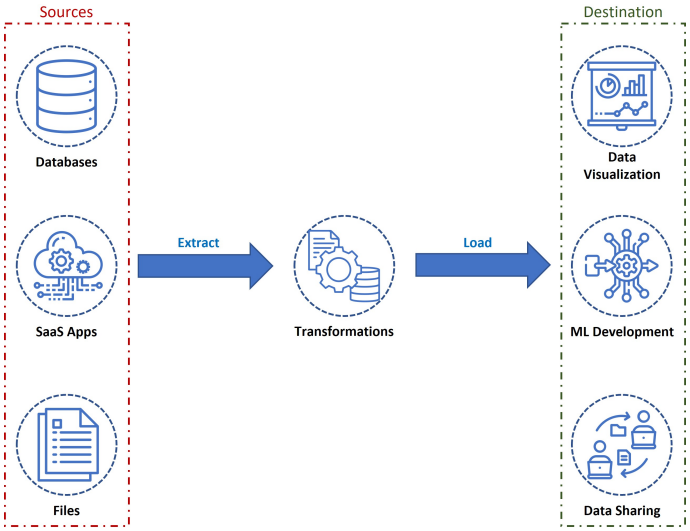


Figure 3.1: Visual representation of a ETL pipeline.

3.2.1.1 Data Manipulation

One of the most prominent issues about working with data is that it is usually disordered and untidy. Specifically, the prevalence of missing values, erroneous values requiring imputation, sub-optimal data types, and other similar issues are commonplace. The data employed in this dissertation was not immune to such problems, requiring modifications to certain variables or even their exclusion altogether.

The initial data and its business meaning can be seen in Table B.3, in the appendixes (although some less relevant variables are not displayed). A quick analysis shows data divided into three big groups:

- Subscription Data – data related to each customer's various subscriptions, their timing and payment;
- User Data – data related to each user's characteristics and statistics;
- Demographics Data – data related to customer's demographics, such as age, localization, gender, etc.

In order to address a large volume of information, a framework was designed, starting with the selection of a core dataframe. Among the available input tables, the 'SUBSCRIPTIONS' table was chosen due to its capacity for displaying the greatest amount of information. Subsequently, the information contained within the remaining dataframes was merged with the core dataframe, creating a unified information source that could be utilized for machine learning purposes. The final dataframe utilized for Data Manipulation consisted of a considerable dataset, containing over 1 million rows. Among these rows, churn entries constituted only approximately 5% of the total. The dataframe was comprised of 38 independent variables, providing a comprehensive set of features for analysis and modeling purposes. The variables included both categorical and numerical

data. The dataset covered information from 2016 onwards and the categorical variables exhibited a significant amount of information, as they often represented more than 20 distinct values. Additionally, some binary variables were present, indicating the activation status of certain attributes. Overall, the initial dataset encompassed a diverse range of information types and refered to over 20000 different subscribers.

Following this, data cleaning procedures were employed, primarily to address NaN values – the columns exhibiting the highest occurrence of missing values[1] are depicted in Figure 3.2. Some rows were removed, whilst other rows were imputed with numerical values in order to maintain a similar distribution of data points (*i.e.*, the imputation was such that the proportion of the dataset belonging to each class remained the same). The specific approach to imputation varied depending on the proportion of NaN's present in each column. Additionally, column's data types were addressed and converted – this is a common step which is required for ML purposes.

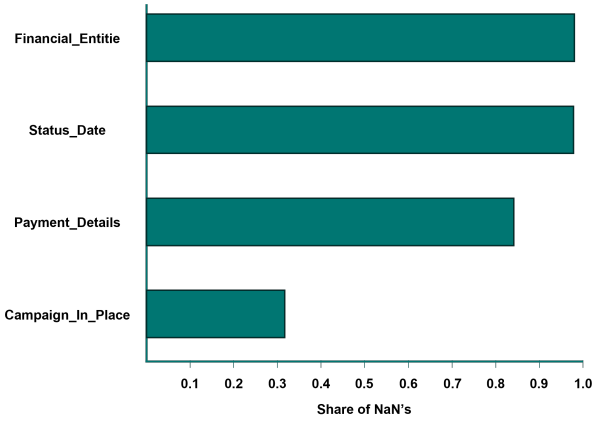
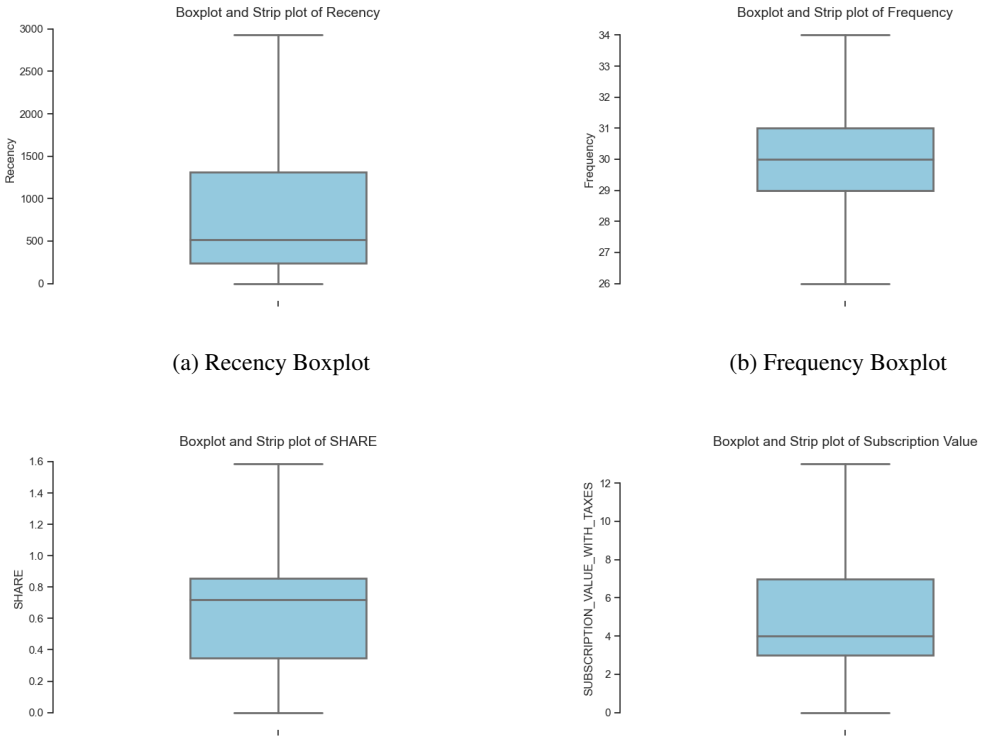


Figure 3.2: Visual representation of the columns with the highest share of missing values.

Outliers, observations that lie an abnormal distance from other values in a random sample of the population, where treated following the methodology stated in Leys et al. (2019) – some can be observed in the boxplots presented in Figure 3.3. Thus, instead of using the mean value of the variable as a reference point to decide whether or not a given instance is extreme, the median absolute deviation (MAD) was used. This approach is based on the following two premises: (1) outlier detection methods based on either the mean or standard deviation suffer from the fact that these two are affected by outliers themselves and (2) given the concept of breakdown point – defined as the proportion of values set to infinity that can be part of the dataset without compromising the estimator (Donoho and Huber, 1983) – this value is 0 for both the mean and standard deviation, but the same doesn't necessarily hold true for the median.

[1] Although initially presenting a challenge with a considerable number of NaN values, Status_Date and Payment_Details were eventually removed and did not pertain to the most significant variables in the analysis. Consequently, Table B.3 does not encompass these variables, as these variables were solely depicted in Figure 3.2 to illustrate the significant occurrence of NaN values within specific columns.



(a) Recency Boxplot

(b) Frequency Boxplot

(c) SHARE Boxplot

(d) Subscription Value Boxplot

Figure 3.3: Visual representation of some quantitative variables.

Finally, feature engineering was employed, by both manipulating certain variables to better retrieve the intended insights (*e.g.*, cluster creation regarding demographic variables) and designing new variables (*e.g.*, subscription's age). The whole process can be visually assessed in the Figure 3.4 below.

Figure 3.4: Visual representation of the Data Manipulation pipeline.

3.2.1.2 Feature Selection

One of the most recurring problems in machine learning is to define, out of all data available, the sections to be used. Selecting the right features to input into the ML model requires both analytical skills and good business understanding of what each variable represents and its impact. The amount of information used to feed the model directly impacts the fitting of the same to the data provided – this means that using a lot of noise may lead to overfitting. Additionally, feature selection improves accuracy and reduces training times by eliminating the redundant and irrelevant variables.

There are several ways to perform feature selection, usually divided in two categories: unsupervised and supervised. Unsupervised measures refer to eliminating redundant information – they only consider the relationships between independent variables. Alternatively, supervised measures aim at eliminating irrelevant information – thus considering the relationships between the independent variables and the dependent variable.

One of the most common method is to use statistical measures to predict the relationships between the input variables and the outcome variable – can be deemed as unsupervised feature selection. Alternatively, one can simply run a classifier (like a decision tree or GBM model) to understand the variables that most impact the output – thus being categorized as supervised feature selection.

This project used two distinct approaches to feature selection, representing both unsupervised and supervised methods, and compared their performance in the context of the specific problem at hand – a summary of the findings can be seen in Chapter 4. Specifically, the study utilized a genetic algorithm based on graph theory and the Boruta Algorithm to identify the most relevant features. The former approach involves a search strategy based on the principles of evolutionary biology, while the latter relies on Random Forest classifiers to select features – and both will be detailed next. Furthermore, to facilitate a comparative framework, a baseline scenario was established in which neither the genetic algorithm nor Boruta were utilized, and instead the model's feature importance determined the selection of features.

Community-Based Genetic Algorithm The community-based genetic algorithm employed in this study draws from the work of Rostami et al. (2021). The algorithm comprises several distinct stages, such as (1) Fisher scores calculation; (2) feature clustering; (3) population initialization; (4) fitness calculation; (5) crossover and mutations and (6) repair operation, until finding the final set of features to use. These steps may be seen in the Figure 3.5 below, retrieved directly from the previously mentioned paper.

To compute feature clustering, a community detection algorithm called Iterative Search for Community Detection (ISCD+) was used (Bai et al., 2017). This algorithm is based on three fundamental concepts: (1) internal compactness; (2) importance concentration and (3) representability. Internal compactness measures the similarity between nodes (*i.e.*, features) in the same cluster based on the number of common neighbors these vertices have. Importance concentration measures how concentrated the importance of a feature is in a cluster – the higher the value, the smaller the number of clusters to which the feature is essential. Representability in a community is based on the idea that if a vertex is surrounded by vertices with high representability in a community, it is likely that the vertex also belongs to that community. The local search aims to find a near-optimal solution that maximizes internal compactness and importance concentration objectives.

Since the ISCD algorithm is based on the creation of a graph, composed by nodes (features) and edges (links between them), feature similarity must be computed. Whilst several approaches may be used, due to having to handle both categorical and numerical variables, a combination of Pearson's correlation – for numeric variables – and Cramer's V – for categorical variables –

was used. Pearson's correlation measures the degree of linear association between two variables, with values ranging from -1 to 1, where -1 indicates a perfect negative correlation, 0 signifies no correlation, and 1 denotes a perfect positive correlation. It assumes a linear relationship between variables and a normal distribution of variables. Cramer's V, on the other hand, is an effect size measurement for the chi-square test of independence, and states whether two categorical fields are associated. Its value ranges from 0 to 1, with values less than 0.2 displaying little to no relation between the variables, values up until 0.6 displaying moderate association and any value above displaying strong linkage, as per IBM (2023).

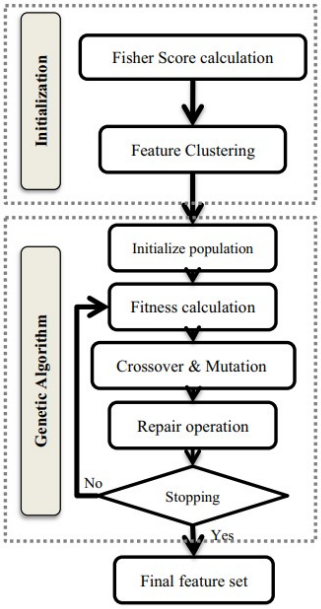


Figure 3.5: Visual representation of the Community Based Genetic Algorithm, adapted from Rostami et al. (2021).

The next step comprises the creation of k communities, with k defined by the user – albeit with help from a graph and the elbow method – which then represents the number of clusters needed as input for the community based genetic algorithm. Then, an intrinsic genetic algorithm is applied (with a similar framework than that of the community-based one) to find the best solution, *i.e*, the best set of clusters, regarding the three main ideas previously stated (internal compactness, importance concentration and representability).

The original idea comprised a single objective function given by the product of both important concentration and internal compactness across all nodes. However, using a simplified objective function that represents two distinct, yet connected, goals may not be the best alternative, since: (1) trade-off between objectives may not be clear; (2) combining both objectives into a single function may not result in a non-convex function, which states the idea of existing multiple local optima, making it challenging to reach (approximately) the optimal solution and (3) the scaling of both functions may be different, causing one to override the other.

In order to address the challenges posed by multi-objective searches, the research employed a methodology based on the Pareto front concept. Specifically, the Non-dominated Sorting Genetic Algorithm 2 (NSGA-2) was selected as a prevalent approach for effectively handling this type of problems (Deb et al., 2002). The algorithm will be explained further ahead. It becomes important to state that this algorithm tackles the three main problems stated previously, namely by: (1) NSGA-2 can identify the Pareto front, which represents the set of solutions not dominated by any other solution in terms of both objectives – allowing for the decision maker to visually understand the trade-off; (2) the usage of a non-dominated sorting algorithm and crowding distance increases diversity and prevents premature convergence, allowing the algorithm to cover the entire pareto front, including non-convex regions, and (3) by using a normalization technique, the algorithm prevents one solution from dominating the other. This clustering technique aims at grouping the most correlated features into the same clusters, whilst also increasing the dissimilarity with features in other clusters.

After feature clustering, the initial population for the genetic algorithm is created, corresponding to a set of chromosomes (vectors with binary values that indicate which features are active). Then, the fitness of each individual of the population is calculated using a specific function that not only considers the classification accuracy of the active features but also the similarity between them. Also here, an innovative approach was taken, with the k-nearest-neighbors classifier being substituted by a GBM model, which stems to provide better results. As in any genetic algorithm, crossover and mutation operations are performed and the fitness of the best individuals is reassessed. The most fit chromosomes advance on to the next phase and repair operations are performed to re-adjust the number of features selected in each group. This step becomes essential as the mutation phase is done randomly following a right skewed probability distribution, with larger number of genes being mutated at the same time having increasingly less probability (simply put, a mutation of one gene is more likely than the mutation of two genes). The repair operations reinstate the expected number of features from each cluster that should be active, w, and are based on the features' Fischer scores. Thus, when a given cluster has a different number of active features than w, the repair operation either adds or removes features by giving higher probability to those with higher Fisher scores.

The genetic algorithm keeps performing crossovers, mutations and repair operations until a given stopping criterion is given – defined in our methodology as a 1000 iterations. The used framework can be seen in Figure 3.6. Additionally, the pseudo-codes related to the main procedures of this algorithm are presented in the Annexes 1, 2, 3, 4 and 5. The final selected variables can be seen in Table B.7. It is important to highlight that the Iterative Search Community Detection algorithm and the Community Based Genetic Algorithm were implemented entirely through manual programming, without the utilization of any pre-existing software packages. This approach was chosen to accommodate the various modifications made during the research process. Considering the substantial alterations implemented, resulting in a notably distinct Feature Selection algorithm, this research proposes the name Pareto-Based Multi-objective Feature Selection (PBFS) for the deployed approach.

Figure 3.6: Visual representation of the genetic algorithm framework used in this project.

Boruta Algorithm The Boruta Algorithm is based on two key ideas: shadow features and binomial distribution. Its aim is to identify relevant features for a given problem by distinguishing them from irrelevant ones. This algorithm is based on the Random Forest algorithm, which is a decision tree-based ensemble method.

This algorithm works by first generating a set of random variables that are permuted copies of the original variables (shadow features). The Random Forest algorithm is then applied to this set of variables, and the importance score of each variable is computed. The importance score is a measure of how well the variable separates the classes in the dataset. After computing the importance scores of the variables, the algorithm applies a feature selection process based on the comparison between the original variables and their permuted copies. The key idea is to identify variables that have a higher importance score than their corresponding permuted copy.

However, it is intuitive that one trial is not enough, as that one hundred trials are more reliable than twenty trials. Thus, one must consider the probability of keeping a feature. Given the fact that the maximum level of uncertainty about the feature is 50% (purely random) and knowing that each independent experiment can only have two outcomes (relevant or not), a series of n trials follows a binomial distribution.

Variables that have a higher importance score than their corresponding permuted copy are called confirmed variables. Variables that have a lower importance score than their corresponding permuted copy are called rejected variables. The remaining variables are called tentative variables and require further evaluation.

In order to evaluate the tentative variables, the algorithm performs additional iterations of the Random Forest algorithm, using only the confirmed variables and the tentative variables. This process is repeated until all variables are either confirmed or rejected. Similarly to the inclusion of the pseudo-code for the Community Based Genetic Algorithm in the appendices, the pseudo-code for the Boruta Algorithm can be found in Appendix 6.

The Boruta algorithm has several advantages over other feature selection methods. First, it can handle a large number of variables without requiring any prior knowledge of the data. Second, it can handle both continuous and categorical variables. Third, it is robust to noise and outliers in the data.

3.2.2 Churn and Uplift Modelling

In this section, a comprehensive overview of churn and uplift modelling is presented. Initially, the focus is on the variables employed as inputs for both models, along with the necessary adjustments made to the implemented methodology. Subsequently, attention is given to hyperparameter tuning,

with an exposition of the specific hyperparameters utilized in each of the employed models. Lastly, the underlying assumptions related to the uplift model are delineated.

3.2.2.1 Initial Variables and Setting

After completing the Data Cleaning phase, the processed data that serves as input for both the churn prediction model and uplift model is summarized in the Tables B.4, B.5 and B.6, in the annexes. It is noteworthy to mention that the effect of Feature Selection has not been addressed at this stage. However, a dedicated section will be provided in Chapter 4 to thoroughly examine its impact and implications.

By comparing both the initial dataframe and the final table, one can find important differences. For once, the columns CAMPAIN_IN_PLACE and SUBSCRIPTION_FINANCIAL_ENTITIE were dropped, the reason being the high number of NaN's they presented, which diminished their ability to be helpful features in the model. Another change is related to the columns SUBSCRIPTION_CLASSIFICATION and SUBSCRIPTION_IS_ACTIVE which, being futuristic variables (*i.e.*, they would only be registered after the customer had already decided whether or not to proceed with the subscription), would potentially damage the model's ability to display reality.

The categorical variables where then mapped automatically by the model – it chooses the best alternative among one-hot encoding, binary encoding, eigen-value encoding or no encoding at all. Continuous variables, on the other hand, where mapped into categorical ones through binning, as it proved to reduce overfitting of the model (concluded by looking at the variables' partial plots).

Additionally, developing a machine learning model assumes training it by utilizing appropriate datasets. Considering what has been discussed previously regarding time-slices, the dataset was, then, divided accordingly. Nonetheless, the existence of varied subscription types with diverse time intervals, such as monthly, bi-monthly, semestral, and yearly, resulted in the creation of diverse time-slices, thereby leading to the division of the dataset in a heterogeneous manner. This thesis only focused on the monthly model, mainly due to its representativeness across the dataset (accounted for almost 2/3 of the whole data points) and to the lack of relevance of considering all possible time intervals regarding this dissertation's purpose. It is relevant to state that, although time slices vary, the methodology used for all models is alike.

The monthly model slices displayed a 30-day period and the testing set displayed a 15-day gap to account for a practical operational period, as seen in the appendixes (A.2). After rightly dividing the dataset, the same had to be divided into training, testing and validation sets. It is important to state that all of these folds have different purposes. Therefore, the initial dataset was divided using the 75-25 rule into both the training set (aiming at training the model) and testing set (aiming at assessing the performance of the model regarding unseen data points). Furthermore, out of the data reserved for training, 15% was used for validation purposes, whilst only 85% were really used for training. The importance of the validation set lies in the idea of fine-tuning the model's parameters by comparing the adjusted model and evaluating its accuracy. This study utilized five years of information – following the insights gathered from Ballings and Poel (2012) to train the models, with the testing set comprising the March 2023 time slices.

Finally, the dependent variable (CHURN) was mapped. To do so, churn needed to be correctly defined. As such, it was considered that if a customer would fail to meet the deadline payment for more than one month, he would have churned. This definition was based on two premises: (1) the newspaper addressed had a high rate of the churn-and-return phenomena – a big chunk of subscribers would (willingly, as far as one could tell) churn and then return the next month/period and (2) most of the payment methods were non-recurrent, which meant that the subscriber himself had to go online on the website the required day to proceed to pay the subscription fee in order to keep their subscription active – many failed and returned a few days after.

3.2.2.2 Churn Prediction Model Selection

A crucial step in every churn prediction modelling approach is that of deciding which model to be used for prediction. This thesis focused on a comparative approach – essentially running a set of algorithms and picking the best one. However, whilst aiming at choosing the most suitable model for the problem at hand, one must first decide which performance metrics to use to assess the model's performance and decide upon it. The ideas discussed in the Performance Evaluation section in Chapter 2 constitute the basis for the selection of the most appropriate metrics. As such, AUC, AUCPR, F1, F2, MCC and Lift where chosen to support this decision.

After appropriately preparing the input data for predictive models and utilizing h2O's python package, an automatic comparative selection process, commonly referred to as AutoML (LeDell and Poirier, 2020), was conducted. Although this method provides values for each model based on the chosen evaluation metrics, these values pertain to the training data used for model training. Hence, it is not immediately evident that the best-performing model on the training set will also yield the best predictions for future observations. To ensure a more robust decision-making process, cross-validation was employed, as discussed in Chapter 2. Additionally, following the approach proposed by Gattermann-Itschert and Thonemann (2021) regarding time-slices, a two-fold cross-validation was performed for each model. This way, a two-step cross validation was deployed, whereas the first one refers to the automatic cross-validation done by the AutoML solver and the second one comprises the idea of combining two different runs and averaging the scores on each metric.

The AutoML tool facilitates the training of a predetermined collection of models, encompassing diverse variations such as Gradient Boosting Machines, Default Random Forest, Generalized Linear Model, Extremely Randomized Forest, and Deep Neural Networks. Moreover, a series of Stacked Ensemble models, constructed upon the aforementioned models, are generated, and their outcomes are presented. Consequently, the comparative analysis primarily concentrates on these models and presents xGBoost as an add-on (since this model is not available in Windows, a Linux virtual machine was deployed).

3.2.2.3 Hyperparameter Tuning

As previously discussed, GBM, DRF, GLM, ERF, DNN and xGBoost were selected as the predictive models to undergo cross-validation and subsequent evaluation and comparison. The hyperparameters of these models were then modified in an attempt to identify configurations that converged favorable outcomes. Determining the values of the hyperparameters and the bounds for the grid-search posed a challenge, since it was not clear which direction to go for better convergence. Ultimately, a 'trial-and-error' procedure was used, with every run deemed an adjustment from the previous one – this can be seen in the Chapter 4.

Lastly, the pertinent hyperparameters to consider for each of the aforementioned models are as follows:

Table 3.1: Models' hyperparameters

Model	Hyperparameters
GBM	minimum leaf size, sample rate, number of trees
DRF	minimum leaf size, sample rate, number of trees
GLM	alpha, lambda
ERF	minimum leaf size, sample rate, number of trees
DNN	epochs
xGBoost	gamma, number of trees

3.2.2.4 Uplift Model

To construct an uplift model, aiming at differentiating churners and preventing wastage of firm's resources on lost causes, an experiment is needed – control and treatment groups must be determined and the results assessed. Due to the impracticality of such for the purpose of this document, the predisposition of customers to receive marketing offers was used as a proxy of the customer's reaction to retention campaigns – one must consider this when discussing potential results.

For this purpose, this thesis used a transformed outcome tree-based uplift model built in the h2O's package, whose model is based on the implementation of a Distributed Random Forest. Consequently, given a set of data, a series of classification uplift trees are generated, each acting as a weak learner trained on a subset of the data. The final outcome refers to estimating the probability of change in user behavior (churn/not churn) regarding the treatment applied (marketing susceptibility). It is noteworthy to state that, although not perfect, this metric does convey important information about the potential behavior of a customer regarding retention strategies (which may be deemed as similar to marketing campaigns).

Chapter 4

Results

This chapter presents the findings of both the Feature Selection process and the models discussed in the preceding chapter. The initial section focuses on the churn prediction results obtained during the initial comparative analysis, with and without the impact of the Feature Selection procedure. Subsequently, a separate table showcases the outcomes of hyperparameter tuning for the most promising model. The chapter concludes by summarizing the results of the uplift model and by determining the impact of creating separate models with respect to the difference between recurrent and non-recurrent payments. Each section of this chapter includes a concise discussion about the findings and their relation to the domain-specific business knowledge acquired.

4.1 Benchmark Model

To ensure a robust evaluation of the model's performance, it is necessary to establish a reliable benchmark that can provide insights into the quality of the presented results. However, selecting an appropriate benchmark for churn analysis poses a challenge due to the dataset-specific nature of the problem. Generalizing a specific threshold for performance metrics to classify models as good or bad is not feasible.

Therefore, in order to gain a comprehensive understanding of the implications of the findings and evaluate the quality of the obtained results, it was crucial to construct an initial model using the raw, unprocessed data. Known to be one of the most used techniques in literature, a Random Forest model was chosen as a preliminary approach. The ensuing outcomes of this model are presented in the subsequent Table 4.1.

Table 4.1: Benchmark model's results regarding the different evaluation metrics

Model	AUC	AUCPR	F1	F2	Lift	MCC
DRF	0.753	0.973	0.958	0.982	1.031	0.176

4.2 Initial Comparative Analysis

As previously discussed, the comparative analysis was performed by using h2O's AutoML package. Since the impact of the Feature Selection techniques has not been addressed, three different runs were performed: (1) AutoML with the total information retrieved after the Data Cleaning procedure; (2) AutoML with the partial information retrieved after both Data Cleaning and Feature Selection (Genetic Algorithm) procedures, and (3) AutoML with the partial information retrieved after both Data Cleaning and Feature Selection (Boruta Algorithm) procedures. Results are shown below. Nevertheless, it is important to highlight two crucial points: Firstly, the disparities observed among the models' outcomes, as presented in Table 4.2, are of minimal significance. Secondly, the reported values correspond to the optimal iterations of each model, wherein h2O's package employs a collection of predefined models, some of which possess minor variations but fundamentally share the same core model.

Table 4.2: Models' results regarding the different evaluation metrics

Scenario	Metrics					
	AUC	**AUCPR**	**F1**	**F2**	**Lift**	**MCC**
Total Information						
SE	0.868 (+ 15.3%)	0.993 (+ 2.1%)	0.977 (+ 2.0%)	0.990 (+ 0.8%)	1.048 (+ 1.6%)	0.261 (+ 48.3%)
GBM	0.867 (+ 15.1%)	0.993 (+ 2.1%)	0.977 (+ 2.0%)	0.990 (+ 0.8%)	1.048 (+ 1.6%)	0.257 (+ 42.6%)
DRF	0.873 (+ 15.9%)	0.995 (+ 2.3%)	0.983 (+ 2.6%)	0.993 (+ 1.1%)	1.036 (+ 0.5%)	0.251 (+ 44.9%)
GLM	0.861 (+ 14.3%)	0.994 (+ 2.2%)	0.983 (+ 2.6%)	0.993 (+ 1.1%)	1.035 (+ 0.4%)	0.255 (+ 42.6%)
DNN	0.851 (+ 13.0%)	0.993 (+ 2.1%)	0.983 (+ 2.6%)	0.993 (+ 1.1%)	1.030 (- 0.1%)	0.247 (+ 44.8%)
XRT	0.873 (+ 15.9%)	0.995 (+ 2.3%)	0.983 (+ 2.6%)	0.993 (+ 1.1%)	1.036 (+ 0.5%)	0.238 (+ 35.2%)
xGBoost	0.893 (+ 18.6%)	0.995 (+ 2.3%)	0.978 (+ 2.1%)	0.991 (+ 0.9%)	1.044 (+ 1.3%)	0.260 (+ 47.7%)
Genetic Algorithm						
SE	0.859 (+ 14.1%)	0.989 (+ 1.6%)	0.972 (+ 1.5%)	0.982 (+ 0.0%)	1.039 (+ 0.8%)	0.230 (+ 30.7%)
GBM	0.857 (+ 13.8%)	0.989 (+ 1.6%)	0.973 (+ 1.6%)	0.983 (+ 0.1%)	1.040 (+ 0.9%)	0.228 (+ 29.5%)
DRF	0.857 (+ 13.8%)	0.990 (+ 1.7%)	0.975 (+ 1.8%)	0.985 (+ 0.3%)	1.033 (+ 0.2%)	0.228 (+ 29.5%)
GLM	0.856 (+ 13.7%)	0.990 (+ 1.7%)	0.975 (+ 1.8%)	0.985 (+ 0.3%)	1.033 (+ 0.2%)	0.230 (+ 30.7%)
DNN	0.850 (+ 12.9%)	0.988 (+ 1.5%)	0.975 (+ 1.8%)	0.985 (+ 0.3%)	1.031 (+ 0.0%)	0.226 (+ 28.4%)
XRT	0.857 (+ 13.8%)	0.990 (+ 1.7%)	0.974 (+ 1.7%)	0.986 (+ 0.4%)	1.029 (- 0.2%)	0.224 (+ 27.3%)
xGBoost	0.868 (+ 15.3%)	0.992 (+ 2.0%)	0.975 (+ 1.7%)	0.984 (+ 0.2%)	1.042 (+ 1.0%)	0.232 (+ 31.8%)
Boruta Algorithm						
SE	0.865 (+ 14.9%)	0.991 (+ 1.9%)	0.974 (+ 1.7%)	0.985 (+ 0.3%)	1.048 (+ 1.6%)	0.238 (+ 35.2%)
GBM	0.864 (+ 14.7%)	0.991 (+ 1.9%)	0.975 (+ 1.7%)	0.986 (+ 0.4%)	1.045 (+ 1.4%)	0.234 (+ 32.9%)
DRF	0.862 (+ 14.5%)	0.995 (+ 2.3%)	0.977 (+ 2.0%)	0.990 (+ 0.8%)	1.035 (+ 0.4%)	0.233 (+ 32.4%)
GLM	0.861 (+ 14.3%)	0.992 (+ 2.0%)	0.977 (+ 2.0%)	0.990 (+ 0.8%)	1.034 (+ 0.3%)	0.239 (+ 35.8%)
DNN	0.854 (+ 13.4%)	0.990 (+ 1.7%)	0.978 (+ 2.1%)	0.990 (+ 0.8%)	1.030 (- 0.1%)	0.234 (+ 32.9%)
XRT	0.862 (+ 14.5%)	0.995 (+ 2.3%)	0.977 (+ 2.0%)	0.990 (+ 0.8%)	1.032 (+ 0.1%)	0.232 (+ 31.8%)
xGBoost	0.875 (+ 16.2%)	0.994 (+ 2.2%)	0.976 (+ 1.9%)	0.991 (+ 0.9%)	1.041 (+ 1.0%)	0.245 (+ 39.2%)

After assessing the results, the following conclusions may be drawn:

- The xGBoost model emerged as the most effective among the models evaluated;
- The results exhibit promise overall, as the models demonstrate a relatively strong capacity to predict individuals who are likely to churn;
- There's room for improvement, as certain explanatory variables may currently be excluded from the model, as inferred from the Lift and Matthews Correlation Coefficient (MCC) values;
- The modifications employed to the initial dataset allowed for a significant improvement of both AUC and MCC metrics when comparing to the benchmark model, thus inferring a much better ability to detect churners;
- Employing all available information proved to be the most favorable approach, although both the Boruta and Genetic Algorithm methods facilitated significant dimensionality reduction at the cost of minimal losses.

It is crucial to emphasize that the conclusions drawn in this study are contingent upon the particular context and dataset that were analyzed. Despite the fact that all models demonstrated satisfactory predictive abilities, it is important to note that the dataset exhibited a significant class imbalance, with only 5% of the data points identified as churn entries. Consequently, even a random model would yield a reasonably good overall performance (though inferior to the achieved values). Thus, in order to thoroughly assess the potential impact of the model and facilitate more reliable decision-making, both the confusion matrix and variable importance will be presented. The latter aims to elucidate the implications of the findings in terms of business knowledge.

In consideration of the objectives of this thesis, it was deemed impractical and unproductive to display the results for each individual model. This decision was based on the substantial effort required and the lack of meaningful benefits due to the similarities observed in the comparison of values obtained. Consequently, only the results pertaining to the GBM model will be showcased. This selection was made because, although xGBoost demonstrated slightly superior results, the GBM model provided more comprehensible outcomes and ranked second in terms of single models. The rationale behind this choice lies in the GBM model's ability to offer clearer insights for subsequent prescriptive measures. Additionally, the xGBoost variable importance and SHAP will be presented in the appendixes, specifically in Figures A.4 and A.5. Furthermore, considering the aforementioned conclusions, and despite the intriguing results related to the utilization of partial information through feature selection techniques, the entire dataset was utilized for the subsequent sections.

4.3 Confusion Matrix

As discussed in the Chapter 2, the confusion matrix provides important information about the model's ability to correctly attribute classes. The churn model results can be seen below, in the Figure 4.1.

		Predicted to churn		
		Yes	No	Error (%)
Churned	Yes	1446	3176	68.7
	No	5552	98307	5.4
	Total	6998	101483	8.1

Figure 4.1: Confusion matrix results regarding churn prediction.

The results demonstrate that the model is able to predict approximately 31% of the total churners. This finding suggests that there are other variables influencing customer behavior that have not been incorporated into the model. Further details regarding these variables will be presented in the Next Steps section of Chapter 5. Additionally, it is essential to determine whether there are any actionable insights regarding the predicted churners.

4.4 Precision for Top Churners

To gain a deeper understanding of the model's practical implications, and determine, out of all of those predicted to churn, how many of them did the model correctly predict, precision was plotted over the top k subscriptions most likely to churn, as presented in Figure 4.2.

The model achieves 100% precision in predicting the top 21% of churners (*i.e.*, all of the top 21% most likely to churn actually churned). This implies that, while there may still be room for improvement, the current methodology can effectively identify a significant portion of churners. Consequently, it enables the accurate targeting of customers for retention strategies. It is worth noting that churn prediction is a complex task, and being able to successfully identify 21% of churners without missing a single one is a notable achievement.

Moreover, it is worth mentioning that the literature lacks comprehensive exploration of this concept. Typically, performance metrics such as AUC, accuracy or F1 Score are emphasized, which may not accurately reflect the model's impact in real-world contexts. As explained in Chapter 2, many of these metrics do not account for dataset imbalances and can provide misleading insights into the predictive model's utility. Consequently, no reliable benchmark regarding this metric was found, and the results presented here may serve as one for future research in this field of study. Furthermore, in order to provide readers with a more comprehensible understanding of

the work accomplished, a similar analysis was performed on the benchmark model (as one can see in the same figure), despite the lack of comparative results.

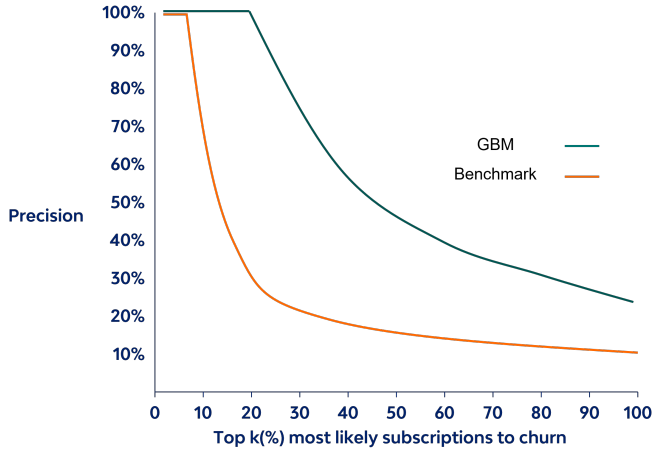


Figure 4.2: Evolution of precision with share of most likely subscriptions to be canceled.

4.5 Variable Importance

As previously discussed in Chapter 2, any algorithm that is based on Decision Trees (as it happens with GBM) recursively divides the training data based on certain splitting criteria whilst minimizing a given impurity measurement – for the package at hand the decision is based on minimizing the squared error. As such, feature importance is calculated by analyzing the relative influence of each variable, *i.e.*, whether that variable was used to split a given node during the construction phase and its impact on the impurity metric. The variable importance of the GBM model can be seen in the Figure 4.3 below.

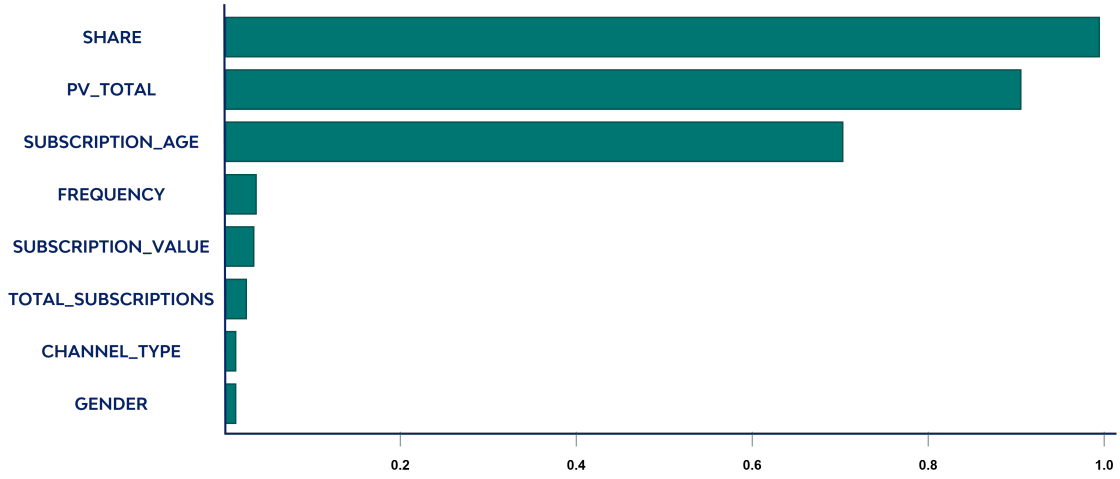


Figure 4.3: Relative importance of the GBM variables.

After analysing the figure above, it becomes evident that SHARE, PV_TOTAL and SUBSCRIPTION_AGE have a significant impact on the outcome variable. Consequently, it is imperative to examine the partial plots of each factor, which assess the incremental effect of a specific attribute on the response variable by averaging the alteration in its value. These can be seen in the Figures 4.4, 4.5 and 4.6.

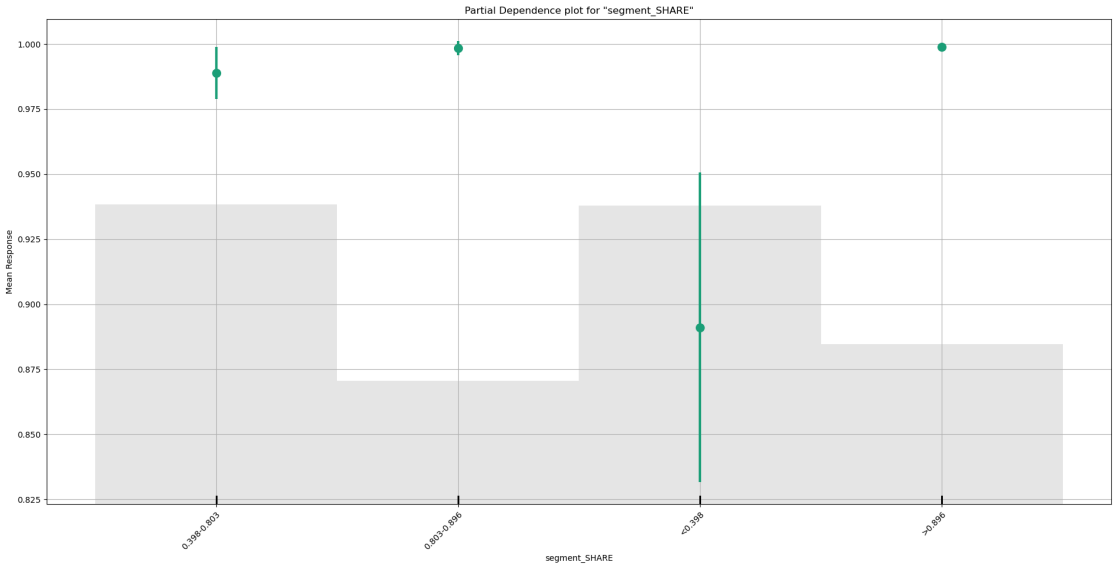


Figure 4.4: Partial plot of the SHARE variable.

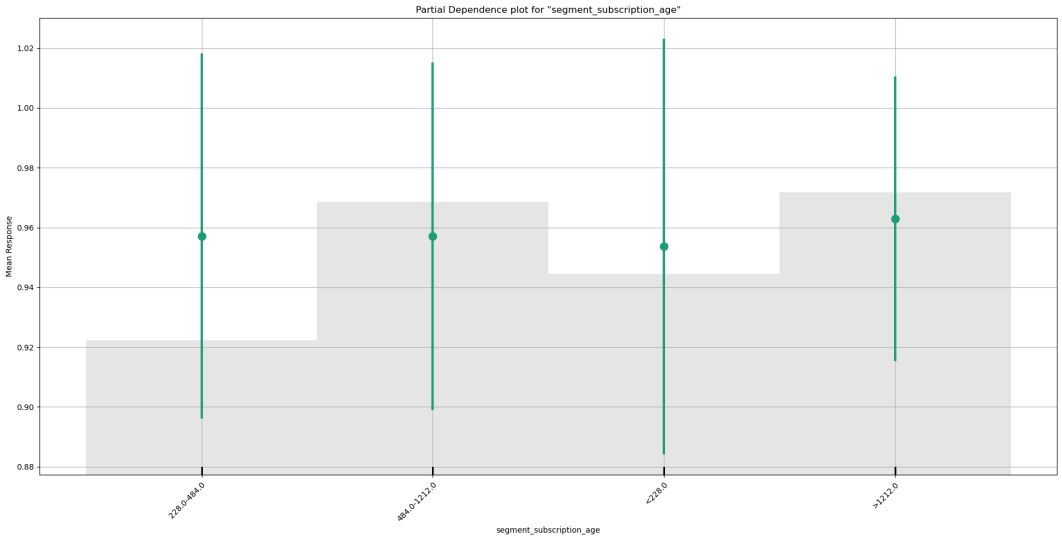


Figure 4.5: Partial plot of the SUBSCRIPTION_AGE variable.

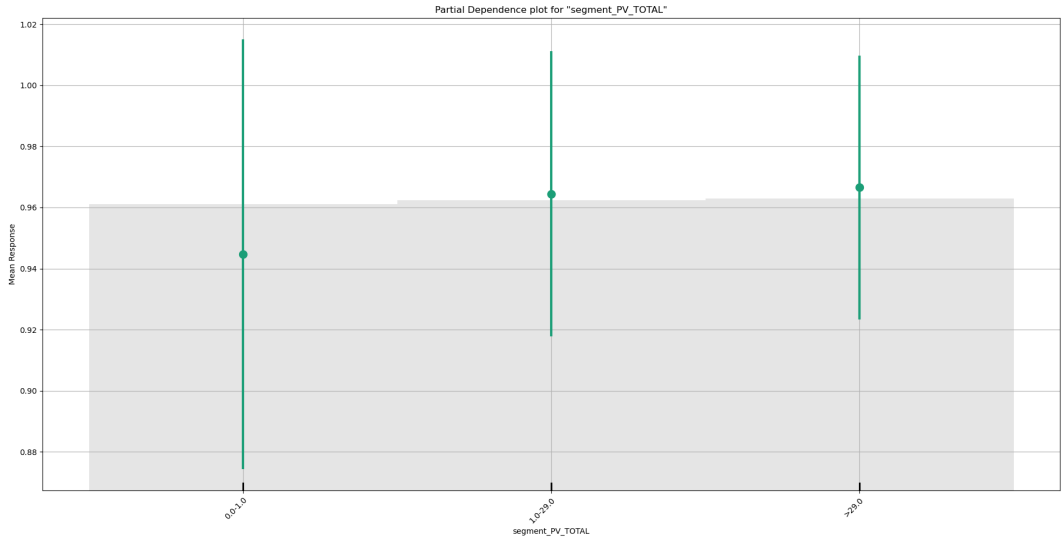


Figure 4.6: Partial plot of the PV_TOTAL variable.

Finally, one may draw the following conclusions:

- The variable SHARE exhibits a positive correlation with the Class 1 variable, indicating that customers who have a higher proportion of total subscribed time with the company are less likely to churn. To illustrate this further, consider an example: if customer A initially subscribed on January 1, 2022, and discontinued their subscription after 3 months in March 2022, their share would be 0.25 (representing 3 months out of a total period of 12 months) as of January 1, 2023;

- Similarly, the variable SUBSCRIPTION_AGE demonstrates a positive association with the Class 1 variable. Consequently, customers who have been subscribed for more than 3 years are less likely to churn compared to those who have been subscribed for less than half a year;

- The variable PV_TOTAL displays, alike the previous ones, a positive correlation with the Class 1 variable. This suggests that an increase in website engagement is associated with a less churn likelihood;

- Contrary to initial intuitions, the analysis reveals that the price of the subscription does not have a substantial impact on consumer behavior. This finding suggests that while price may influence the initial decision to purchase a subscription, it does not significantly affect the subsequent decision to retain that subscription.

Additionaly, the Shap Summary is presented below in Figure 4.7. SHAP stands for Shapley Additive exPlanations and is a machine learning technique that is used to explain a given model. It works by determining the contribution of each feature towards the prediction. To do so, it uses the concept of Shapley value (Shapley, 1953), which bases the idea on game theory and determines

the payout to each 'player' depending on their contribution to the total payout, which explains the difference between the actual prediction and the average prediction. Therefore, the Shapley value is the average of all the marginal contributions of that feature to all possible coalitions – which is different from computing the impact of removing the feature from the model. An illustrative example, gathered from Molnar (2022) is that of thinking about a room full of features (coalition) and a given feature entering that room. The Shapley value is the average change that the coalition in the room receives when the feature value joins them. Each point on the Shap Summary is, therefore, a Shapley value for a given feature in a given instance. Thus, analyzing this graph provides the user with an overall understanding of not only the feature importance ranking but the average marginal effects of each feature's value on the prediction itself. The results affirm the previously presented findings regarding feature importance and lead to an additional conclusion: features that have a higher explanatory power exhibit a greater disparity in instances where their results differ. In other words, there is a relationship between the predictive capability of a variable and its capacity to identify distinct behaviors associated with different values of that feature. Put simply, the more easily distinct behaviors can be identified, the higher the likelihood that the variable possesses predictive power.

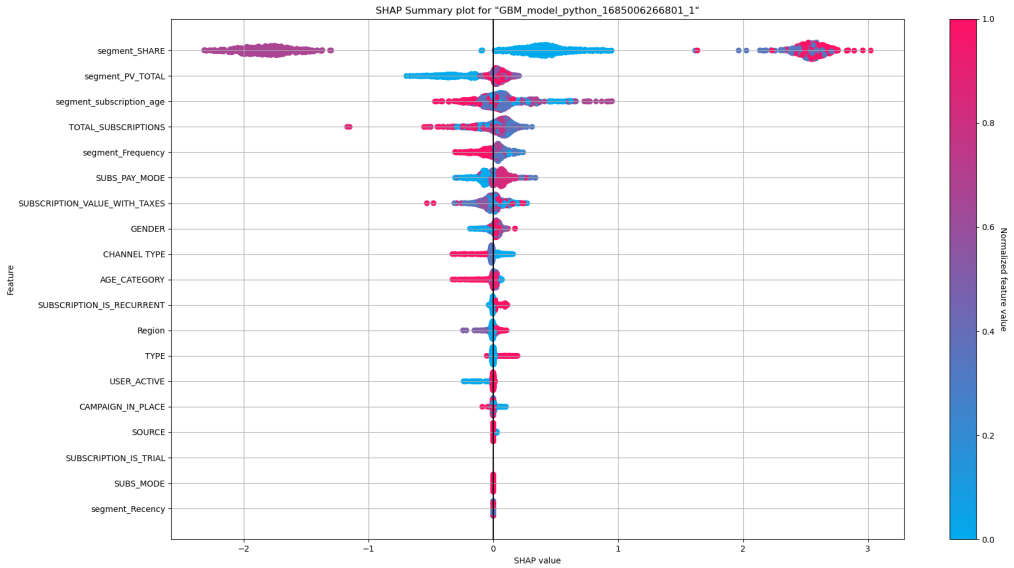


Figure 4.7: SHAP results regarding the GBM model.

4.6 Hyperparameter Tuning

As mentioned earlier, the purpose of the hyperparameter tuning phase is to optimize the selection of parameters that best fit the model to the training data. The objective is to identify the combination of parameter values that yield the highest-performing model within a feasible time frame. To achieve this, a random search grid methodology, inspired by the findings of Bergstra and Bengio

(2012), was employed. The results of this hyperparameter optimization process are presented in Table 4.3.

Table 4.3: Hyperparameter tuning results

Leaf Size	Sample Rate	Number of Trees	AUC
9	0.5	90	0.8679
12	0.5	65	0.8674
9	0.5	75	0.8672
9	0.5	60	0.8672
10	0.6	70	0.8669

The results indicate that a smaller leaf size and sample rate, along with a larger number of trees, are the most favorable configurations for achieving optimal convergence. It is important to note that the differences in performance, as measured by the AUC metric, among the various models are negligible. This observation becomes even more apparent when considering that the AUC metric exhibited the highest degree of variability among all metrics. Furthermore, the presentation focused on showcasing only the top 5 combinations, as the aim was to provide a concise subset for drawing conclusions.

4.7 Uplift Results

The results presented in this analysis regarding the uplift modeling primarily focus on the potential impact of the model based on the initial dataset, rather than the direct implications for the firm's current business model. This approach is justified by the absence of any implemented initiatives by the newspaper at the present time, making it impossible to ascertain the actual effects of the proposed model. Consequently, to gain a deeper understanding of the algorithm's potential outcomes, this discussion will present both the cumulative uplift curve and the proportion of predicted churners expected to respond positively to interventions.

Similar to the Lift Curve presented in Chapter 2, this metric displays the performance of the model across different deciles or percentiles of the sample and allow one to compare them with a random model, as one may see in the Figure 4.8. The findings suggest that the model exhibits a slight advantage in capturing the behavioral disparity between the control and treatment groups when compared to a random model. This outcome can be considered favorable, as it indicates that the model is performing better than chance in distinguishing the effects of the treatment, meaning it can capture whether or not a given churner is likely to be a persuadable.

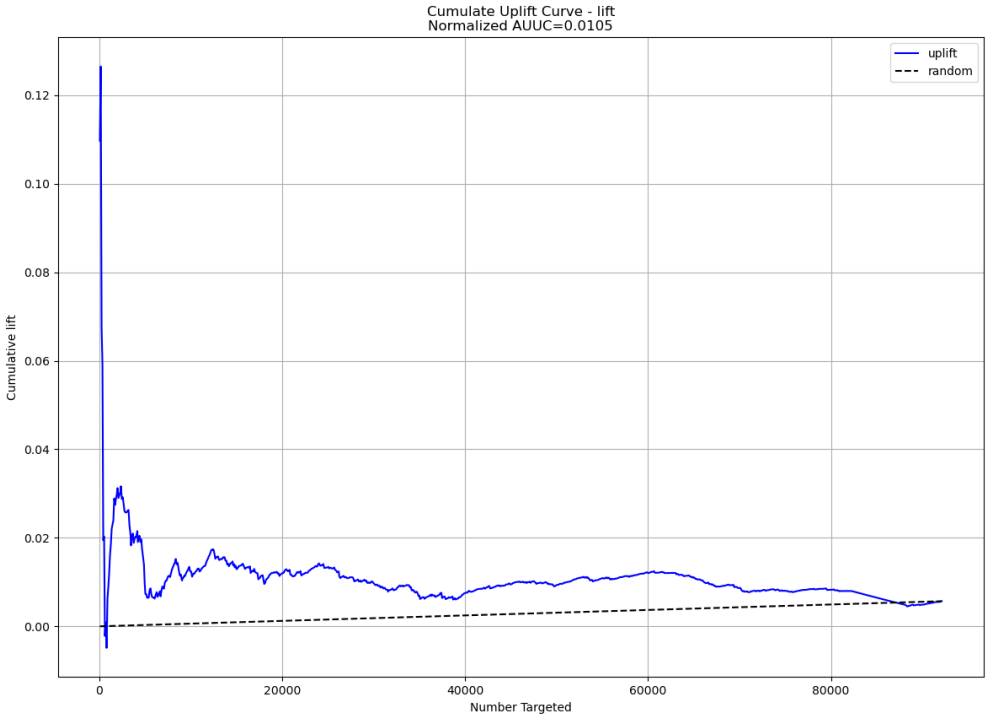


Figure 4.8: Visual representation of the uplift curve regarding the uplift model.

The metric measuring the proportion of predicted churners anticipated to respond positively to interventions serves the purpose of providing an intuitive understanding of the potential impact of implementing an uplift model following churn prediction (*i.e.*, the potential savings in retention efforts). It is important to note that this metric does not evaluate the quality of the current uplift, as that has already been assessed when analyzing the uplift curve. The results may be seen in Figure 4.9. It's important to state that a churner was considered to be receptive to a retention campaign if the model predicted a positive difference in behavior. Additionally, the figure is a mere visual representation, as the absolute values of predicted positive cases were 450 out of 652.

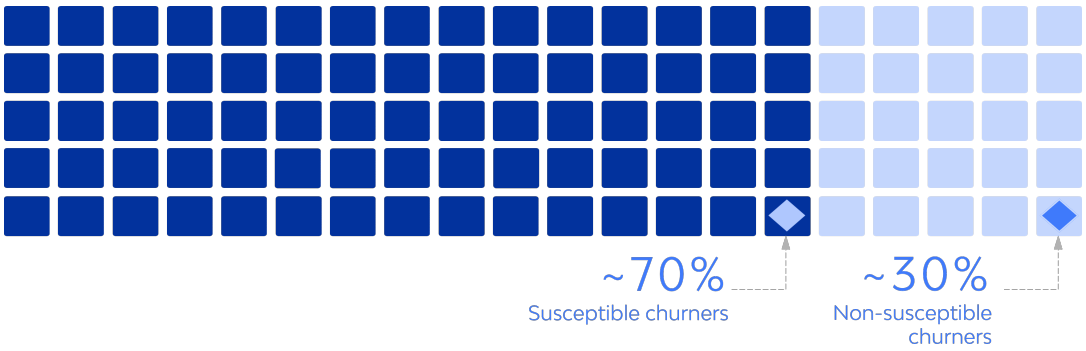


Figure 4.9: Proportion of predicted churners.

4.8 Payment Mode Differentiation

As outlined in Chapter 3, the examination of the CHURN variable revealed that customer behavior differed between recurrent and non-recurrent subscriptions. Non-recurrent subscriptions necessitated active renewal by the customers, while recurrent subscriptions did not require any additional effort. In order to assess the potential impact of this distinction, two separate models were developed, one for recurrent subscriptions and another for non-recurrent subscriptions, despite the variable not demonstrating significant influence as a predictor. Results can be seen in Table 4.4.

Table 4.4: Monthly models' results by payment mode

Model	AUC	AUCPR	F1	F2	Lift	MCC
Recurrent	0.849	0.992	0.978	0.991	1.045	0.217
Non-Recurrent	0.875	0.994	0.977	0.991	1.046	0.274

Based on the analysis of the table, it can be inferred that dividing the monthly model by payment mode did not yield a satisfactory outcome – supporting the idea of it not being an important variable. Consequently, the final model was determined to be a combination of both payment modes, as it had been previously.

Chapter 5

Conclusion and Recommendations

This chapter encompasses the key findings and insights derived from the conducted research. It also includes prudent recommendations regarding the practical implementation of the obtained conclusions. Additionally, a self-critical assessment is provided, followed by suggestions for future work and proposed next steps, concluding this thesis.

5.1 Summary of Findings

As mentioned in Chapter 1, the objective of this thesis was to present a practical application of machine learning models for predicting customer churn in the media industry, which is currently an area of research in its early stages. Furthermore, the study explored the implications of an uplift modelling technique and is yet to derive potential retention measures that the firm could implement to reduce customer turnover.

The analysis of the churn models yielded intriguing insights. Contrary to intuitive expectations, the price of the newspaper subscription was not found to be a significant factor in determining whether a customer would cancel their subscription at a given point in time. This suggests that price may have greater importance in attracting new customers rather than in retaining existing ones. Additionally, customer loyalty metrics demonstrated a strong ability to accurately predict churners. The historical record of each customer's interaction with the company's website, as measured by the cumulative number of page views, also emerged as a reliable predictor of churn intentions. Another noteworthy finding was that customer-specific characteristics, rather than subscription-based factors, did not exhibit high importance in the decision to churn. In other words, customer demographics appeared to play no role in their churn decision. Lastly, despite the inclusion of a wide range of variables in the model, the predictive ability was concentrated on a few features. This may suggest that while multiple variables may be considered, ultimately the decision to churn may be driven by one or two key considerations.

5.2 Prescriptive Retention Measures

The findings gathered from the churn predictive model suggested several potential retention metrics that the firm could implement to improve customer retention and increase their expected lifetime value (LTV). The findings indicate that customers who exhibit greater levels of interaction with the website are more likely to exhibit lower churn rates. Consequently, the newspaper organization could consider implementing marketing offers aiming at increasing customer's recurrence in the website. Furthermore, the analysis establishes that customer loyalty plays a significant role in influencing the decision to renew subscriptions. Therefore, the firm could explore the development of a reward system based on account metrics related to website engagement, or even improve offers based on the number of consecutive subscription payments. Additionally, automated personalized engagement methods such as email notifications or website reminders could prove beneficial. However, it is important to note that these proposed retention strategies carry inherent risks, which are detailed in Table 5.1.

Table 5.1: Possible retention measures and associated risks

Strategy	Risk
Marketing offers aiming at increasing customer's recurrence in the newspaper's website	Customers may take advantage of the situation and wait for the promotions to subscribe
Reward system based on account metrics related to website engagement	May increase elderly customer's confusion regarding website utilization and increase churn in this segment
Improved offers based on the number of consecutive months with an active subscription	Inaccurately identifying the threshold at which a customer's loyalty is sufficiently strong that an improved offer is unnecessary, thus leading to a decrease in the customer's lifetime value
Automated personalized engagement methods such as email notifications or website reminders	May remind some customers of a subscription they forgot to cancel if the model misses to correcly detect possible churners

5.3 Prescriptive Measures Implementation

Upon thorough examination of potential prescriptive strategies for customer retention, considering the related risks, the subsequent stage entails outlining a structured plan for their successful implementation. This section presents, for each of the strategies described earlier, a possible implementation blueprint.

5.3.1 Recurrent Marketing Offers

The primary objective of recurring marketing offers is to foster customer loyalty and encourage repeated visits to the company's online platform. This can be achieved through various means, such as providing discounts on the monthly subscription fee or granting access to exclusive content, contingent upon the customer's level of engagement with the website. For instance, a feasible approach could involve offering a 15% discount on the subscription fee if the customer consistently accesses the website every day over a 30-day period. Alternatively, the company could incentivize regular visits by granting access to limited content reserved for customers who return to the website on a weekly basis, thereby providing a preview of the upcoming week's content.

5.3.2 Reward System

The construction of a reward system based on website engagement metrics aims at increasing the loyalty of customers and fostering active participation with the newspaper's content. An illustrative implementation entails the establishment of customer ranks determined by criteria such as the number of monthly website visits, engagement with articles through actions like comments and likes, as well as the diversity of articles accessed. Subsequently, the top-ranking individuals, such as the top 100 subscribers, would be granted access to exclusive content or the opportunity to win coveted prizes such as a complimentary year-long subscription.

5.3.3 Improved Offers Based On Consecutive Subscription Time

One potential approach is to establish milestones corresponding to each complete year of uninterrupted monthly subscription, accompanied by progressively decreasing discounts. For example, a 15% discount could be offered in the first year, followed by 10% in the second year, and 5% in the third year. This approach recognizes that customers who reach higher milestones are likely to exhibit greater loyalty and may require fewer incentives to maintain their connection with the company.

5.3.4 Automated Personalized Engagement

The conventional practice of employing automated email notifications or website reminders a few days before a subscription's renewal date is a widely-used strategy in the business realm to mitigate customer churn. However, its effectiveness is somewhat restricted. To enhance its impact, a more personalized approach can be adopted. By tailoring email engagements to incorporate loyalty metrics such as the duration of the customer's subscription, the total number of articles accessed by the user, their preferred author, and complementing it with their rank within the reward system, greater motivation can be instilled in the customer to retain their subscription. This approach seeks to differentiate customers by acknowledging their individual preferences and engagement history, thereby strengthening their incentive to remain loyal.

5.4 Future Work and Next Steps

In Chapter 4, it was observed that although the model successfully predicted the top 21% of customers most likely to churn, there is still room for improvement. This finding implies that either customers base their churn decisions on one or two key factors or the input variables used in the model are insufficient. For instance, it would be valuable to include information related to customers' overall satisfaction with the published articles or their preferences for specific writers, as well as their engagement levels within the articles. It would also be important to have access to data relative to service provided and resulting customer satisfaction. Additionally, incorporating customer-specific information, such as credit score, income, family structure, and incident score, would enhance the granularity of the findings and facilitate a more comprehensive analysis. Currently, these aspects are not taken into account due to the lack of stored data, which limits the model's ability to make informed predictions. By incorporating such relevant data, the model can capture additional factors that may influence customers' churn decisions and enable appropriate actions to be taken. Furthermore, in order to evaluate the effectiveness of the churn model in predicting subscription cancellations, predictions were made for the upcoming months. However, it is important to note that the results of these predictions are yet to be stored and analyzed for further assessment and that the prioritized objective lies in the development of a deployable tool that can effectively execute the selected models and generate predictions for the most probable churners. Additionally, it is important to emphasize that the execution of the suggested modifications should adhere to a feedback-driven methodology, wherein the model is consistently refined and adjusted based on real-time data gathered.

Regarding the uplift model, it is worth reiterating that the customer's responsiveness to marketing campaigns was utilized as a proxy for their behavior in relation to retention measures. However, to establish a robust model, an experiment incorporating both a treatment group and a control group would be essential. In this experiment, a subset of customers would be designated as the treatment group, receiving targeted marketing or retention offers, while another subset would serve as the control group, without any specific interventions. By comparing the variations in customer churn between these groups, a more accurate understanding of expected customer behavior could be derived. A visual representation of such procedure can be seen in Figure A.6, in the appendixes. Once the model is fed with information regarding customers' susceptibility to retention strategies, the final outcome should be evaluated, focusing on the reduction of wastage and resource utilization brought about by the model's impact. To achieve this, another experiment can be conducted, wherein the control group consists of individuals predicted by the churn model to cancel their subscription, while the experiment group comprises individuals who not only have a high probability of canceling their subscription but also a high probability of accepting a retention offer (*i.e.*, persuadables). By calculating the marginal impact of investing one euro in a retention campaign, the final impact can be assessed. If the uplift model performs as intended, the marginal impact of the experiment group should exhibit a significant increase compared to the control group, indicating the effectiveness of targeted retention efforts.

Bibliography

Ahmad, M. W., Mourshed, M., and Rezgui, Y. (2017). Trees vs neurons: Comparison between random forest and ann for high-resolution prediction of building energy consumption. *Energy and Buildings*, 147:77–89.

Ascarza, E. (2018). Retention futility: Targeting high-risk customers might be ineffective. *Journal of Marketing Research*, 55:80–98.

Ascarza, E. and Hardie, B. G. S. (2013). A joint model of usage and churn in contractual settings. *Marketing Science*, 32:570–590.

Bai, L., Cheng, X., Liang, J., and Guo, Y. (2017). Fast graph clustering with a new description model for community detection. *Information Sciences*, 388-389:37–47.

Balle, B., Casas, B., Catarineu, A., Gavaldà, R., and Manzano-Macho, D. (2013). The architecture of a churn prediction system based on stream mining. *Artificial Intelligence Research and Development*, 256:157–166.

Ballings, M. and Poel, D. V. D. (2012). Customer event history for churn prediction: How long is long enough? *Expert Systems with Applications*, 39:13517–13522.

Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2).

Bhandari, P. (2023). Random vs. systematic error | definition & examples. `https://www.scribbr.com/methodology/random-vs-systematic-error/`.

Bishop, C. M. (1994). Neural networks and their applications. *Review of Scientific Instruments*, 65:1803–1832.

Bishop, C. M. (2007). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 1 edition.

Bogaert, M. and Delaere, L. (2023). Ensemble methods in customer churn prediction: A comparative analysis of the state-of-the-art. *Mathematics*, 11:1137.

Boutetière, H., Montagner, A., and Reich, A. (2018). Unlocking success in digital transformations. `https://www.mckinsey.com/capabilities/people-and-organizational-performance/our-insights/unlocking-success-in-digital-transformations`.

Branco, P., Torgo, L., and Ribeiro, R. P. (2017). A survey of predictive modeling on imbalanced domains. *ACM Computing Surveys*, 49:1–50.

Breiman, L. (2001). Random forests. *Machine Learning*, 45:5–32.

Brynjolfsson, E. and McAfee, A. (2017). The business of artificial intelligence. https://hbr.org/2017/07/the-business-of-artificial-intelligence.

Cardoso, G., Paisana, M., and Pinto-Martinho, A. (2021). Digital news report 2021: Portugal. https://reutersinstitute.politics.ox.ac.uk/digital-news-report/2021/portugal.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794, New York, NY, USA. Association for Computing Machinery.

Chiang, D. A., Wang, Y. F., Lee, S. L., and Lin, C. J. (2003). Goal-oriented sequential pattern for network banking churn analysis. *Expert Systems with Applications*, 25:293–302.

Coussement, K. and den Poel, D. V. (2008). Churn prediction in subscription services: An application of support vector machines while comparing two parameter-selection techniques. *Expert Systems with Applications*, 34:313–327.

Coussement, K. and den Poel, D. V. (2009). Improving customer attrition prediction by integrating emotions from client/company interaction emails and evaluating multiple classifiers. *Expert Systems with Applications*, 36:6127–6134.

Das, M., Elsner, M., Nandi, A., and Ramnath, R. (2015). Topchurn: Maximum entropy churn prediction using topic models over heterogeneous signals. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15 Companion, page 291–297, New York, NY, USA. Association for Computing Machinery.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.

Devriendt, F., Berrevoets, J., and Verbeke, W. (2021). Why you should stop predicting customer churn and start using uplift models. *Information Sciences*, 548:497–515.

Donoho, D. L. and Huber, P. J. (1983). The notion of breakdown point. *A festschrift for Erich L. Lehmann*, 157184.

Dreiseitl, S. and Ohno-Machado, L. (2002). Logistic regression and artificial neural network classification models: a methodology review. *Journal of Biomedical Informatics*, 35:352–359.

Efron, B. (1983). Estimating the error rate of a prediction rule: Improvement on cross-validation. *Journal of the American Statistical Association*, 78:316–331.

Eiben, A. E., Koudijs, A. E., and Slisser, F. (1998). Genetic modelling of customer retention. In Banzhaf, W., Poli, R., Schoenauer, M., and Fogarty, T. C., editors, *Genetic Programming*, pages 178–186, Berlin, Heidelberg. Springer Berlin Heidelberg.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996). The kdd process for extracting useful knowledge from volumes of data. *Communications of the ACM*, 39:27–34.

Fletcher, R. (2019). Skip to explore the report paying for news and the limits of subscription. https://www.digitalnewsreport.org/survey/2019/paying-for-news-and-the-limits-of-subscription/.

Fortmann-Roe, S. (2012). Understanding the bias-variance tradeoff. `http://scott.fortmann-roe.com/docs/BiasVariance.html`. Accessed June 22, 2023.

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., and Herrera, F. (2012). A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4):463–484.

García-Portugués, E. (2023). Notes for predictive modeling. `https://example.com/notes-for-predictive-modeling`.

Gattermann-Itschert, T. and Thonemann, U. W. (2021). How training on multiple time slices improves performance in churn prediction. *European Journal of Operational Research*, 295:664–674.

Guerola-Navarro, V., Gil-Gomez, H., Oltra-Badenes, R., and Sendra-García, J. (2021). Customer relationship management and its impact on innovation: A literature review. *Journal of Business Research*, 129:83–87.

Guo, F. and Qin, H. (2017). Data mining techniques for customer relationship management. *Journal of Physics: Conference Series*, 910:012021.

Hossain, M. Y., Azizi, E., and Zaman, L. (2023). Predicting subscription renewal using binary classification in world of warcraft. *Entertainment Computing*, 44:100522.

IBM (2023). Cramér's v. `https://www.ibm.com/docs/en/cognos-analytics/11.1.0?topic=terms-cramrs-v`.

Janiesch, C., Zschech, P., and Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, 31:685–695.

Japkowicz, N. and Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent Data Analysis*, 6:429–449.

Kaur, H., Pannu, H. S., and Malhi, A. K. (2019). A systematic review on imbalanced data challenges in machine learning: Applications and solutions. *ACM Comput. Surv.*, 52(4).

Khan, A., Ehsan, N., Mirza, E., and Sarwar, S. Z. (2012). Integration between customer relationship management (crm) and data warehousing. *Procedia Technology*, 1:239–249.

Khodabandehlou, S. and Rahman, M. Z. (2017). Comparison of supervised machine learning techniques for customer churn prediction based on analysis of customer behavior. *Journal of Systems and Information Technology*, 19:65–93.

Kirasich, K. ., Smith, T. ., and Sadler, B. (2018). Random forest vs logistic regression: Binary classification for heterogeneous datasets. *SMU Data Science Review*, 1.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'95, page 1137–1143, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Krawczyk, B., Woźniak, M., and Schaefer, G. (2014). Cost-sensitive decision tree ensembles for effective imbalanced classification. *Applied Soft Computing Journal*, 14:554–562.

Kubat, M. (2000). Addressing the curse of imbalanced training sets: One-sided selection. *Fourteenth International Conference on Machine Learning*.

LeDell, E. and Poirier, S. (2020). H2O AutoML: Scalable automatic machine learning. *7th ICML Workshop on Automated Machine Learning (AutoML)*.

Leys, C., Delacre, M., Mora, Y. L., Lakens, D., and Ley, C. (2019). How to classify, detect, and manage univariate and multivariate outliers, with emphasis on pre-registration. *International Review of Social Psychology*, 32.

Lo, V. S. Y. (2002). The true lift model: A novel data mining approach to response modeling in database marketing. *SIGKDD Explor. Newsl.*, 4(2):78–86.

Luo, Q. (2008). Advancing knowledge discovery and data mining. In *First International Workshop on Knowledge Discovery and Data Mining (WKDD 2008)*, pages 3–5.

Madhulatha, T. S. (2012). An overview on clustering methods.

McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5:115–133.

Molnar, C. (2022). *Interpretable Machine Learning: A Guide For Making Black Box Models Explainable*. Independently published.

Natekin, A. and Knoll, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics*, 7.

Nguyen, N. N. and Duong, A. T. (2021). Comparison of two main approaches for handling imbalanced data in churn prediction problem. *Journal of Advances in Information Technology*, 12:29–35.

Nitzan, I. and Libai, B. (2011). Social effects on customer retention. *Journal of Marketing*, 75:24–38.

Nykodym, T., Kraljevic, T., Wang, A., Wong, W., and Bartz, A. (2023). Generalized linear modeling with h2o: Seventh edition generalized linear modeling with h2o.

Owczarczuk, M. (2010). Churn models for prepaid customers in the cellular telecommunication industry using large data marts. *Expert Systems with Applications*, 37:4710–4712.

Parr, T. (n.d.). The difference between l1 and l2 regularization. `https://explained.ai/regularization/L1vsL2.html`. Accessed June 22, 2023.

Pendharkar, P. C. (2009). Genetic algorithm based neural network approaches for predicting churn in cellular wireless network services. *Expert Systems with Applications*, 36:6714–6720.

Ramanathan, U., Subramanian, N., Yu, W., and Vijaygopal, R. (2017). Impact of customer loyalty and service operations on customer behaviour and firm performance: empirical evidence from uk retail sector. *Production Planning and Control*, 28:478–488.

Reinartz, W. J. and Kumar, V. (2003). The impact of customer relationship characteristics on profitable lifetime duration. *Journal of Marketing*, 67(1):77–99.

Rostami, M., Berahmand, K., and Forouzandeh, S. (2021). A novel community detection based genetic algorithm for feature selection. *Journal of Big Data*, 8.

Russel, S. and Norvig, P. (2021). *Artificial Intelligence: A Modern Approach*. Pearson, 4th edition.

Saias, J., Rato, L., and Gonçalves, T. (2022). An approach to churn prediction for cloud services recommendation and user retention. *Information (Switzerland)*, 13.

Schapire, R. E. (2003). *The Boosting Approach to Machine Learning: An Overview*, pages 149–171. Springer New York, New York, NY.

Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.-F., and Dennison, D. (2015). Hidden technical debt in machine learning systems. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, page 2503–2511, Cambridge, MA, USA. MIT Press.

Sewell, M. (2011). Ensemble learning. Technical report, University College London.

Shapley, L. S. (1953). *17. A Value for n-Person Games*, pages 307–318. Princeton University Press, Princeton.

Sharma, T., Gupta, P., Nigam, V., and Goel, M. (2020). Customer churn prediction in telecommunications using gradient boosted trees. In Khanna, A., Gupta, D., Bhattacharyya, S., Snasel, V., Platos, J., and Hassanien, A. E., editors, *International Conference on Innovative Computing and Communications*, pages 235–246, Singapore. Springer Singapore.

Shearer, E. (2021). More than eight-in-ten americans get news from digital devices. *PewResearch*.

Shin, S., Hoang, T.-T., Le, T.-H., and Lee, M.-Y. (2016). A new robust design method using neural network. *Journal of Nanoelectronics and Optoelectronics*, 11:68–78.

Shruti (2017). Re-inventing nyt: Impact of digitalization on news media. *Harvard EDU*.

Singh, H. (2006). The importance of customer satisfaction in relation to customer loyalty and retention. Working Paper.

SMARTDRAW (n.d.). Decision tree - learn everything about decision trees. `https://www.smartdraw.com/decision-tree/`. Accessed June 22, 2023.

Smith, R. E. and Wright, W. F. (2004). Determinants of customer loyalty and financial performance. *Journal of Management Accounting Research*, 16:183–205.

Statista (2022). Portugal: Leading consumer newspapers by circulation. `https://www.statista.com/statistics/1333877/portugal-leading-consumer-newspapers-by-circulation/`.

Subasi, A. (2020). *Machine learning techniques*, pages 91–202.

Sutton, C. D. (2005). 11 - classification and regression trees, bagging, and boosting. In Rao, C., Wegman, E., and Solka, J., editors, *Data Mining and Data Visualization*, volume 24 of *Handbook of Statistics*, pages 303–329. Elsevier.

Vadakattu, R., Panda, B., Narayan, S., and Godhia, H. (2015). Enterprise subscription churn prediction. pages 1317–1321. IEEE.

Vafeiadis, T., Diamantaras, K., Sarigiannidis, G., and Chatzisavvas, K. (2015). A comparison of machine learning techniques for customer churn prediction. *Simulation Modelling Practice and Theory*, 55:1–9.

Van den Poel, D. and Larivière, B. (2004). Customer attrition analysis for financial services using proportional hazard models. *European Journal of Operational Research*, 157(1):196–217. Smooth and Nonsmooth Optimization.

Verbeke, W., Dejaeger, K., Martens, D., Hur, J., and Baesens, B. (2012). New insights into churn prediction in the telecommunication sector: A profit driven data mining approach. *European Journal of Operational Research*, 218:211–229.

Zhang, T. and Yu, B. (2005). Boosting with early stopping: Convergence and consistency. *Annals of Statistics*, 33:1538–1579.

Zhang, X., Liu, Z., Yang, X., Shi, W., and Wang, Q. (2010). Predicting customer churn by integrating the effect of the customer contact network. In *Proceedings of 2010 IEEE International Conference on Service Operations and Logistics, and Informatics*, pages 392–397.

Zhou, Z.-H. (2021). *Machine Learning*. Springer Singapore.

Appendix A

Figures

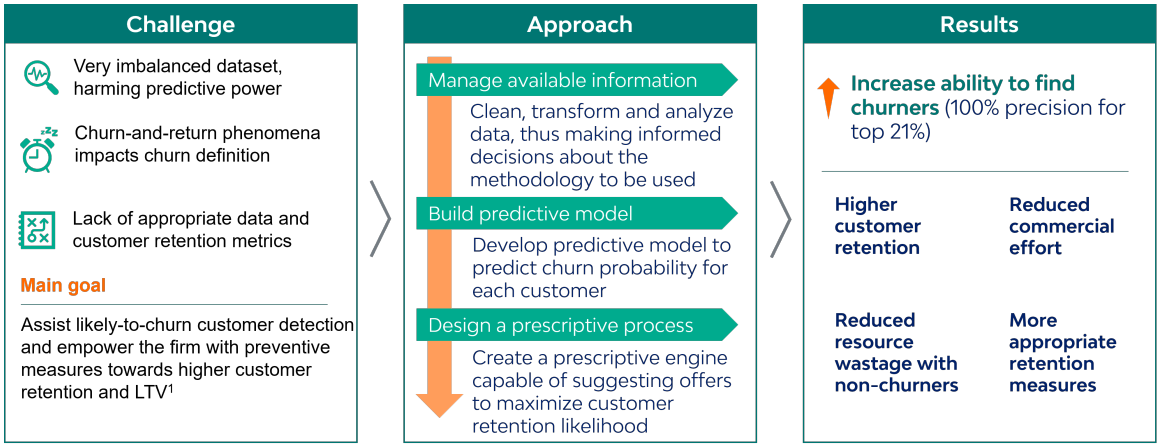


Figure A.1: Graphical representation of this thesis' framework.

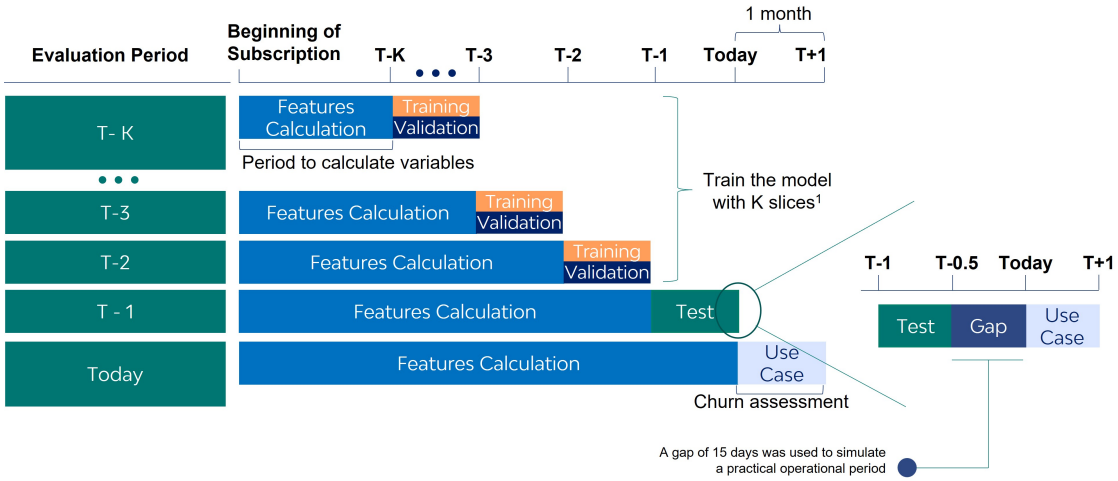


Figure A.2: Visual representation of the monthly model time slices.

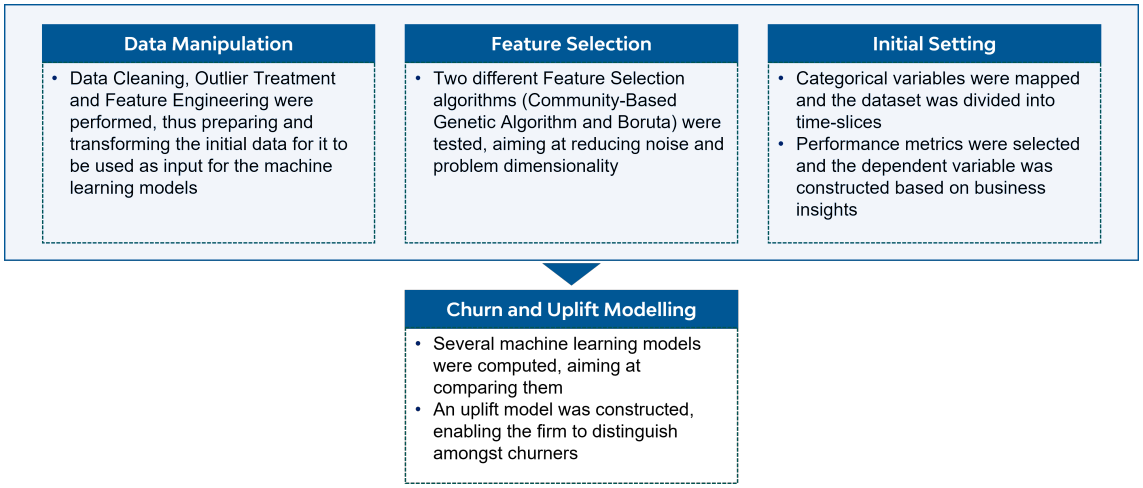


Figure A.3: Graphical representation of the applied methodology.

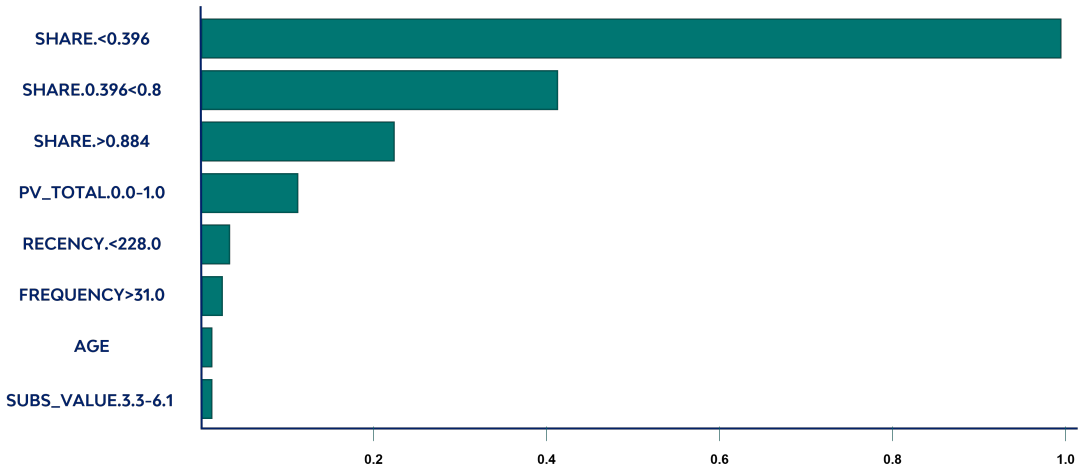


Figure A.4: xGBoost's results regarding variable importance.

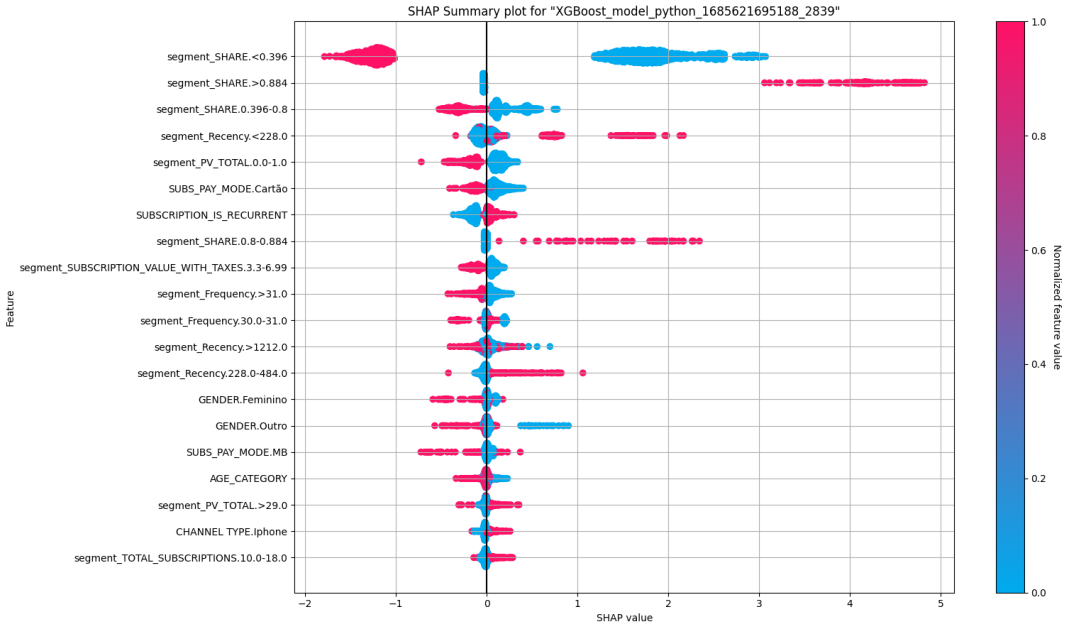


Figure A.5: xGBoost's results regarding SHAP.

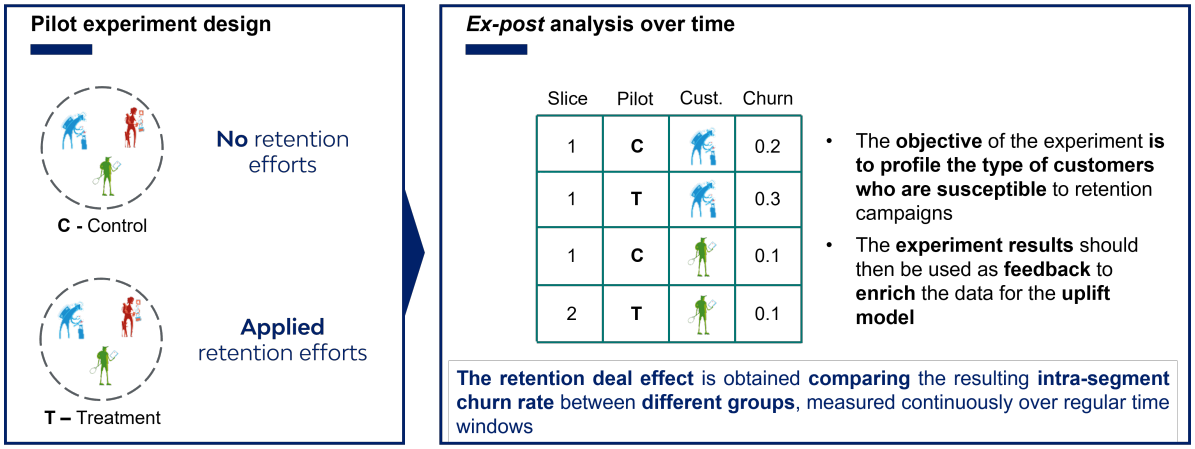


Figure A.6: Experiment framework regarding customer's responsiveness to retention offers.

Appendix B

Tables

Table B.1: Pros and cons of different ML models

Model	Characteristics	Pros	Cons
Decision Trees	Recursively partitions the data based on the value of the input variables in order to minimize the entropy	Easy to interpret and explain	Prone to overfitting; Arguably non-robust to imbalanced datasets
Logistic Regression	The value of the output variable is given by assigning a weight to each of the input variables	Efficient for binary and linear problems	Low predictive power if relationship between independent and dependent variables is not linear
Random Forest	Combination of tree classifiers	Ability to handle high-dimensional data; Relatively robust to noise and outliers; Effectively captures relationships between features	Not so robust at capturing relationships between the features and the dependent variable; Choosing the wrong number of trees may lead to either overfitting or underfitting
Neural Networks	Learns from a given set of examples and is inspired in the human nervous system	Ability to handle non-linear relationships; After training the algorithm is deemed to be very fast	Need to provide suitable set of examples for training; Lack of flexibility in generalizing outside of training region
GBM	Relies on the idea of building an ensemble of models by combining weaker models into a strong learner	Can handle wide variety of data types; Ability to handle high-dimensional and sparse data	Prone to overfitting
XGBoost	A more regularized form than GBM	Can handle wide variety of data types; Ability to handle high-dimensional and sparse data	Prone to overfitting (although less than GBM due to regularization)

Table B.2: Summary of churn techniques and findings

Study	Techniques	Findings
Zhang et al. (2010)	DT, LR, NN	Network attributes greatly improve models accuracy
Owczarczuk (2010)	DT, LR	Logistic Regression is a very good choice regarding pre-paid clients, while Decision Trees are unstable
Khodabandehlou and Rahman (2017)	DT, NN, SVM	Neural Networks are the best estimator, while Decision Trees are the worst
Chiang et al. (2003)	Sequential Patterns	The main cause for banking clients to churn was incorrectly entering their PIN twice
Eiben et al. (1998)	Genetic Programming, DT, LR	The Genetic Algorithm offers better performance
Galar et al. (2012)	Ensemble techniques	The use of ensemble classifiers outperforms the use of a single classifier
Verbeke et al. (2012)	DT, RF, NN, k-NN, Naive Bayes	A small number of variables is sufficient to predict churn with high accuracy, and there's no consensus regarding the best techniques
Coussement and den Poel (2008)	SVM, LR, RF	Random Forests outperform both SVM and LR
Ballings and Poel (2012)	LR, DT	After five years, about 69% of data adds no predictive performance
Das et al. (2015)	Maximum Entropy-based Model	Temporal, unigram, and status history are the most predictive variables
Coussement and den Poel (2009)	LR, SVM, RF	Adding emotions expressed in emails increases predictive performance; LR is a better estimator than SVM and RF
Pendharkar (2009)	GANN, MLGANN	Both GA based NN models outperform the statistical z-score test
Ascarza and Hardie (2013)	Hidden Markov Process	The model provided both accurate forecasts and interesting insights regarding the prior stages the customer goes through before churning
Vadakattu et al. (2015)	DT	Ranking customers by Customer Satisfaction Index proved to be useful for result assessment
Bogaert and Delaere (2023)	33 classifiers (simple, homogeneous and ensembles)	Ensemble methods with higher predictive power overall
Saias et al. (2022)	NN, RF	Neural Networks may not be advantageous depending on the situation

Table B.3: Initial dataset

Name	Group	Description
SUBSCRIPTION_CUSTOMER_ID	Subscription	Subscription's ID
SUBSCRIPTION_FINANCIAL_ENTITIE	Subscription	Registered financial entity which owned the subscription
SUBSCRIPTION_CLASSIFICATION	Subscription	If that subscription was classified as Canceled, Paid or on Hold
SUBSCRIPTION_ORIGINAL_DAYS	Subscription	Subscription's duration
SUBSCRIPTION_IS_ACTIVE	Subscription	If subscription is active
SUBSCRIPTION_IS_RECURRENT	Subscription	If subscription is recurrent
SUBSCRIPTION_DATE	Subscription	Date of subscription's instance
SUBSCRIPTION_START_DATE	Subscription	Date of subscription's initialization
SUBSCRIPTION_END_DATE	Subscription	Date of subscription's deadline
SUBSCRIPTION_IS_TRIAL	Subscription	If subscription is a trial
SUBSCRIPTION_VALUE_WITH_TAXES	Subscription	Subscription's price
SUBSCRIPTION_VAT_TAX	Subscription	Subscription's tax value
CAMPAIGN_IN_PLACE	Subscription	If subscription was acquired during a campaign
PRODUCT	Subscription	Type of Subscription (monthly vs yearly etc.)
SOURCE	Subscription	If subscription was bought through Piano, Store or is Corporate
SUBS_TYPE	Subscription	Either paid or offered
SUBS_MODE	Subscription	If it is a digital subscription
SUBS_PAY_MODE	Subscription	Payment method used
USER_IS_EDITOR	User	If user is editor
USER_ACTIVE	User	If user is active in that moment
TOTAL_PAGEVIEWS	User	Total number of page-views of that user until that point in time
TOTAL_NEWSLETTERS	User	Total number of different newsletters that customer has visited
REGION	Demographic	User's region (north, center, south)
GENDER	Demographic	User's gender
DATE_OF_BIRTH	Demographic	Date of birth

Table B.4: Dataset after Data Management – subscription

Name	Group	Description
SUBSCRIPTION_CUSTOMER_ID	Subscription	Subscription's ID
SUBSCRIPTION_IS_RECURRENT	Subscription	If subscription is recurrent
SUBSCRIPTION_DATE	Subscription	Date of subscription's instance
SUBSCRIPTION_IS_TRIAL	Subscription	If subscription is a trial
SUBSCRIPTION_VALUE_WITH_TAXES	Subscription	Subscription's price
CAMPAIGN_IN_PLACE	Subscription	If subscription was acquired during a campaign
PRODUCT	Subscription	Type of Subscription (monthly vs yearly etc.)
SOURCE	Subscription	If subscription was bought through Piano, Store or is Corporate
SUBS_TYPE	Subscription	Either paid or offered
SUBS_MODE	Subscription	If it is a digital subscription
SUBS_PAY_MODE	Subscription	Payment method used
SUBS_AGE	Subscription	Total number of days since subscription started

Table B.5: Dataset after Data Management – user

Name	Group	Description
USER_IS_EDITOR	User	If user is editor
USER_ACTIVE	User	If user is active at that point in time
TOTAL_PAGEVIEWS	User	Total number of page-views of that user until that point in time
TOTAL_NEWSLETTERS	User	Total number of different newsletters that customer has visited
FAV_TOPIC	User	User's favorite topic online (most clicks)
TOPIC_MEDIA	User	How much mediatic exposure the favorite topic had during last month (low, medium, high)
USER_AGE	User	Total number of days since creating user
SHARE	User	Share of time user was subscribed since joining
RECENCY	User	Interval of time (days) between first and last subscriptions
FREQUENCY	User	Interval of time (days) between consecutive purchases
VOLUME	User	Number of subscriptions of that user in the past 60 days
TOTAL_ARTICLES_CLOSED	User	Total number of closed articles accessed in the past year

Table B.6: Dataset after Data Management – demographic

Name	Group	Description
REGION	Demographic	User's region (north, center, south)
GENDER	Demographic	User's gender
AGE_CATEGORY	Demographic	Age group (children, young-adult, adult, elderly)

Table B.7: Dataset after applying the genetic algorithm

Name	Group	Description
SUBSCRIPTION_IS_TRIAL	Subscription	If subscription is a trial
SUBSCRIPTION_VALUE_WITH_TAXES	Subscription	Subscription's price
CAMPAIGN_IN_PLACE	Subscription	If subscription was acquired during a campaign
PRODUCT	Subscription	Type of Subscription (monthly vs yearly etc.)
SOURCE	Subscription	If subscription was bought through Piano, Store or is Corporate
SUBS_TYPE	Subscription	Either paid or offered
SUBS_MODE	Subscription	If it is a digital subscription
SUBS_PAY_MODE	Subscription	Payment method used
USER_IS_EDITOR	User	If user is editor
USER_ACTIVE	User	If user is active at that point in time
TOTAL_PAGEVIEWS	User	Total number of page-views of that user until that point in time
SHARE	User	Share of time user was subscribed since joining
RECENCY	User	Interval of time (days) between first and last subscriptions
FREQUENCY	User	Interval of time (days) between consecutive purchases
VOLUME	User	Number of subscriptions of that user in the past 60 days

Table B.8: Dataset after applying the Boruta algorithm

Name	Group	Description
SUBSCRIPTION_VALUE_WITH_TAXES	Subscription	Subscription's price
SOURCE	Subscription	If subscription was bought through Piano, Store or is Corporate
SUBS_AGE	Subscription	Total number of days since subscription started
TOTAL_PAGEVIEWS	User	Total number of page-views of that user until that point in time
SHARE	User	Share of time user was subscribed since joining
RECENCY	User	Number of periods (days) between first and last subscriptions
FREQUENCY	User	Number of periods (days) after a purchase that the customer buys again

Appendix C

Algorithms

Data: exemplars
Result: prob_next_exemplar
for each *node* in *graph* **do**
  if *node* not in *exemplars* **then**
    prob_next_exemplar(*node*) ← `N_neighbors` (*node*) / (`Max_neighbors` (*node*) +1);
  end
end

Algorithm 1: Pseudo-code regarding the probability of being the next exemplar's procedure

Data: N_clusters, prob_next_exemplar, exemplars
Result: init_clusters
sorted_prob_next_exemplar ← `sorted`(*prob_next_exemplar*);
for *i* ← 1 **to** *N_clusters* **do**
  add *sorted_prob_next_exemplar*[*i*] to exemplars;
end
init_clusters ← exemplars;
for each *node* in *graph* and not in exemplars **do**
  cluster ← `best_cluster`(*node*);
  add *node* to *init_clusters*[*cluster*];
end

Algorithm 2: Pseudo-code regarding clusters' initialization procedure

```
Data: N_mutations, num_solutions, init_clusters
Result: pareto_frontier
solutions ← {};
for i ← 1 to N_mutations do
    add mutation(init_clusters) to solutions;
end
for each solution in solutions do
    if domination_count(solution) = 0 then
        add solution to pareto_frontier;
    end
end
while len(pareto_frontier) < num_solutions do
    solutions ← solutions not in pareto_frontier;
    n ← len(pareto_frontier);
    if n > 0 then
        next_front ← {};
        for each solution in solutions do
            if domination_count(solution) = 0 then
                add solution to next_front;
            end
        end
        if len(pareto_frontier) + len(next_front) ≤ num_solutions then
            add next_front to pareto_frontier;
        end
        else
            remaining ← num_solutions − len(pareto_frontier);
            add next_front[: remaining] to pareto_frontier;
        end
    end
end
```

Algorithm 3: Pseudo-code regarding the pareto frontier's procedure

```
Data: N_generations, N_crossovers, population, n
Result: population
solutions ← {};
for generation ← 1 to N_generations do
    population ← pareto_frontier(population);
    top_solutions ← get_top_n_solutions (population, n);
    new_population ← {};
    for i ← 1 to N_crossovers do
        sol_1 ← random_choice (solutions);
        sol_2 ← random_choice (solutions);
        new_solution ← crossover (sol_1, sol_2);
        add new_solution to new_population;
        new_mutation ← mutation (sol_1, sol_2);
        add new_mutation to new_population;
    end
    population ← pareto_frontier(population + new_population)
end
```

Algorithm 4: Pseudo-code regarding NSGA2's method

```
Data: chromosome, max_iter
Result: chromosome_final
n ← 0;
chromosome_inicial ← chromosome;
chromosome_final ← chromosome_inicial;
eval ← 0;
best_eval ← classifier_final (chromosome);
while n < max_iter do
    k ← np_random_triangular;
    chromosome_inicial ← mutation (chromosome, k);
    eval ← classifier_final (chromosome);
    if eval > best_eval then
        chromosome_inicial ← repair (chromosome);
        eval ← classifier_final (chromosome_inicial);
        if eval > best_eval then
            best_eval ← eval;
            chromosome_final ← chromosome_inicial;
        end
    end
    n ← n + 1;
end
```

Algorithm 5: Pseudo-code regarding genetic algorithm's local search

```
Data: original_attributes, N_runs ← 0, Max_runs, MIRA ← 0
Result: extended_attributes, randomized_attributes, important_attributes,
        non_important_attributes
  while N_runs < Max_runs or len(original_attributes) > 1 do
    for each attribute in original_attributes do
      randomized_self ← compute_randomized_self (attribute);
      add randomized_self to extended_attributes;
      add randomized_self to randomized_attributes;
    end
    for each attribute in extended_attributes do
      importance ← compute_importance (attribute);
      if attribute is in randomized_attributes then
        add randomized_self to extended_attributes;
        if importance > MIRA then
          MIRA ← importance;
        end
        if importance statistically higher than MIRA then
          add attribute to important_attributes;
        end
        if importance statistically lower than MIRA then
          add attribute to non_important_attributes;
          remove attribute from original_attributes;
          remove attribute from extended_attributes;
          remove randomized_self from extended_attributes;
        end
      end
    end
  end
```

Algorithm 6: Pseudo-code regarding Boruta's procedure