Tesis Doctoral

Predicción de abandono de clientes mediante modelos de aprendizaje automático de supervivencia híbridos

Pedro Nuno de Alexandre Sobreiro

Programa de Doctorado en Tecnologías Informáticas

(TIN)

Con la conformidad de los directores

Dr. José Javier Berrocal Olmeda y Dr. José Manuel García Alonso

Esta tesis cuenta con la autorización del director/a y coodirector/a de la misma y de la Comisión Académica del programa. Dichas autorizaciones constan en el Servicio de la Escuela Internacional de Doctorado de la Universidad de Extremadura

2023

"It is not the strongest of the species that survive, or the most intelligent, but the ones most responsive to change."

Charles Darwin

"Necessity, who is the mother of invention."

Plato

For my grandfather, I cannot forget you.

For my wife, for is patience.

For my son, for being who is.

For my stepsons for their support.

For Martinho, without him I wouldn't started this thesis.

For all my family, coleagues and friends that understanded the lack of availability and helped me.

For Javier and José for all their support and understanding during this process.

UNIVERSITY OF EXTREMADURA

Abstract

University of Extremadura

Department of Information and Communication Technologies

Doctor of Philosophy

Customer dropout prediction using machine learning hybrid survival models

by Pedro SOBREIRO

Customer dropout is a problem in most organizations. They usually lose money when customers stop paying monthly fees for their contractual settings. The process to understand when customers dropout or what are the factors related to the dropout seems a logical approach to developing preemptive actions before a customer churns. The existing studies address the problem as a technical problem and not as a business problem which underlies the implementation and improvement of existing approaches. The dropout prediction has been addressed using a static perspective, and using metrics related to the performance of the algorithms without considering their interpretability. This interpretability is key to support the development of action and retention plans using information about the timings at which the dropout could occur. This misalignment between existing approaches and how to use them to support business actions is a problem that needs to be addressed. The goal of this thesis is to understand how the historical data can be used to predict the customer dropout and support the development of countermeasures in contractual settings. To achieve these goals, a new approach to predict the timings related to the customer dropout is proposed. This approach uses survival trees, combined with the use of clustering techniques. Following this idea are explored two case studies in a health club and sport club, using their membership data. By combining these techniques we were able to increase the accuracy of the survival models. The defined proposal is applied in two different case studies (a health club and sport club) using their membership data.

Keywords: Machine Learning, Survival Analysis, Customer dropout, Survival Trees

UNIVERSIDAD DE EXTREMADURA

Resumen

Universidad de Extremadura

Departmento de Ingeniería de Sistemas Informáticos y Telemáticos

Doctorado en Tecnologías de la Información

Predicción de abandono de clientes mediante modelos de aprendizaje automático de supervivencia híbridos

por Pedro Sobreiro

El abandono de clientes es un problema en la mayoría de las organizaciones. Estas suelen reducir sus ingresos cuando los clientes dejan de pagar las cuotas mensuales. Comprender cuándo se produce el abandono de clientes, o cuáles son los factores relacionados, parece un enfoque lógico para desarrollar acciones preventivas antes de que un cliente se dé de baja. Los estudios existentes abordan el problema como un problema técnico y no como un problema empresarial. Hasta ahora, la predicción del abandono se ha abordado desde una perspectiva estática y utilizando métricas relacionadas con el rendimiento de los algoritmos sin tener en cuenta su interpretabilidad. Esta interpretabilidad es clave para apoyar el desarrollo de planes de acción y retención utilizando información sobre los momentos en los que podría producirse el abandono. Este desajuste entre los enfoques existentes y la forma de utilizarlos es un problema que debe abordarse. El objetivo de esta tesis es comprender cómo se pueden utilizar los datos históricos para predecir el abandono de clientes y apoyar el desarrollo de contramedidas. Para lograr estos objetivos, se propone un nuevo enfoque para predecir los tiempos y momentos relacionados con el abandono del cliente. Este enfoque utiliza árboles de supervivencia, combinados con el uso de técnicas de clustering. Esta propuesta es aplicada en dos casos de estudio: en un club de salud y un club deportivo. Los resultados de esta validación muestran que combinando estas técnicas conseguimos aumentar la precisión de los modelos.

Palabras Clave: Aprendizaje automático, análisis de supervivencia, Ingeniería del Software

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the business environment, organizations lose money when a customer stops paying monthly fees related to a contractual setting. Organizations are often pressured to develop attraction strategies to acquire new members, a pressure that increases if the organization loses more customers than the new ones that are gaining.

The process to understand when customers dropout or what are the factors related to the dropout seems a logical approach to developing preemptive actions before a customer churns. Customer churn has an impact on the organization performance, namely (Amin et al., 2017): (1) negative impact on the overall performance; (2) low sales due to short-term customers; (3) competitors gain dissatisfied customers with promotions; (4) leads to revenues losses; (5) negative impact on long-term customers; (6) increases uncertainty which reduces the ratio of new possible customers; (7) costs to attract new customers is expensive than retaining; and (8) risks the company image in a competitive market and loss of customer base. Customer dropout could be addressed by forecasting who is likely to end their contract and then offer concessions or gifts to retain (Sivasankar & Vijaya, 2019).

The development of an approach to manage customer churn is fundamental for retaining the customer (Ascarza, 2018), and a successful approach requires retention strategies which depend on the accuracy of the prediction, and also of the timing when the prediction is done (Alboukaey et al., 2020). Overall, is pivotal to identify the customer future decision to support the development of counteractions in early stages (Nie et al., 2011). To solve this problem, organizations usually address this, using historical data to train a

model to classify customers as a binary outcome (churners/non-churners) (Verbeke et al., 2012a). This common approach to addressing customer dropout is based on the analysis to detect patterns related to dropout using machine learning techniques.

Existing studies approach the problem as churn, a marketing related concept representing a customer who is going to a competitor in a near future (Glady et al., 2009). Neslin et al. (2006) adds a time perspective to the concept, considering a decision to cease doing business with a company in a given time period. Berry & Linoff (2004) considers that a customer churn can be voluntary or involuntary, where voluntary means that the customer makes a decision to terminate with the provider, or involuntary when the company withdraws the customer due to abuse of service. The main idea is that a customer ends his relationship with the organization moving to a competitor (Coussement & Van den Poel, 2009; Keramati et al., 2014), not developing a membership renewal decision. Which, according to Bhattacharya (1998) is a repeating buying decision to obtain access to the organization services. Devriendt et al. (2019) uses also the concept of customer attrition or customer defection to represent the loss of a customer from a customer base. Other authors (e.g. Jerath et al. (2011)) adopt also the concept of customer dropout, although addressing noncontractual settings where dropout represents a customer that become permanently inactive (Glady et al., 2015), meaning that the customer stops acquiring the organization services or products. Considering the business context of memberships, we use the concept of customer dropout as an event that represents one that abandons an attempt, activity, or chosen path, and could also be interpreted as a withdrawal from a membership[1]. Broadly speaking, the membership renewal decision in the case of paid memberships falls under the rubric of repeat buying.

Several approaches are employed to predict customer dropout, such as random forest, which is commonly used due to the following reasons (Coussement & Van den Poel, 2008): (1) predictive performance; (2) robust to outliers and noise; (3) reasonable computing time and (4) easy to implement. Other researchers also adopted the decision trees, because of their interpretability (Coussement & De Bock, 2013). Considering the timing when dropout occurs, the survival models were also employed to explore a dynamic perspective (Burez & Vandenpoel, 2008). Vijaya & Sivasankar (2019b) suggested also the adoption of hybrid models, combining more than one classifier to increase the performance against

[1]Merriam-Webster. (n.d.). Dropout. In Merriam-Webster.com dictionary. Retrieved July 27, 2022, from https://www.merriam-webster.com/dictionary/dropout

single classifiers, an idea also explored using clusters and predicting the customer dropout in each cluster (Hung et al., 2006).

However, to our knowledge, few studies consider when customer dropout occurs (Sobreiro, Martinho, et al., 2022), which according to Risselada et al. (2010) the parameters change over time and the prediction models should be adapted regularly, where the prediction performance deteriorates after the estimation period. Addressing dropout considering the duration of the relationship is an approach that could be considered. An organization may explore different models using time-related variables as an important feature to explore and identify patterns. Moreover, it seems there is a lack of research addressing time-related variables that could be combined with different techniques improve its performance and understanding, which entails also the model interpretability.

In this thesis, we explore other approaches to better predict the customers future behavior, to support preemptive actions. We use traditional machine learning models combined with survival models to improve and identify momentums related to customer churn. To support our research idea, we explore two cases where the customers pay a monthly fee to use their services, one addressing customer membership in a Health Club, and a second case customer membership in a sports club. These two cases show how organizations can explore these approaches (e.g. Marketing Departments or Managers) to capture the customers dropout time-related events and develop counteractions to reduce customers churns and consequently increase their profits, using the data available in the organization. The selection of random survival models is related to their easy use and interpretability without the limitations of existing models of time-related variables such as the cox-regression (Van den Poel & Larivière, 2004). To address this, we developed a new approach combining clusters with random forest survival models to better predict when customers tend to dropout, and compare its performance against models without clusters.

This chapter introduces the research context. In the section 1.1 are presented the reasons that originated this thesis. Section 1.2 describes the research context that underlies the development of this thesis. In section 1.3 we present our research questions and thesis research objectives 1.4. The contribution of the thesis is represented in section 1.5. Finally, section 1.6 is described the thesis structure.

1.1 Motivation

Customer analysis is fundamental to developing business and marketing intelligence (Sheth et al., 1998), this allows understanding historical data to identify data and patterns (Berry & Linoff, 2004). The information that is available in the organizations allows supporting decisions, using the most valuable asset they possess (Jones et al., 2000).

The underlying assumptions that originated this thesis were based on the idea that several organizations had problems resulting from customer dropout created by fierce competition due to saturated markets, dynamic market conditions, and continuous introductions of new offers (Coussement et al., 2017). These problems require a proper strategy to ensure the organization survival (Van den Poel & Larivière, 2004). Using existing data to solve the problem of dropout seems a logical approach to explore. To our knowledge, dropout is a problem that is commonly addressed using dropout prediction without considering time-varying variables (Gür Ali & Arıtürk, 2014).

Simultaneously, the professional context that motivated this thesis was developed in 2015. During these years was developed work related to Machine Learning predicting customer dropout in Sport Organizations and started exploring new approaches to address the problem considering the area of data science. The main problem in those organizations was related to customer retention and how to address the problem of using other approaches than traditional statistics models to support the development of actions in a business context using existing information in the organizations.

It seems that independently of the type of approach the problem should be targeted considering the extraction of patterns considering their interpretability and not only their performance. Existing research shows the application of different cost-sensitive performance metrics, such as the area under the receiver operating curve (AUC), sensitivity, specificity, recall, precision, and F-score (Jafari-Marandi et al., 2020). However, it should also be also considered a business perspective that requires more than a prediction if a customer will or not churn (Devriendt et al., 2019), where if we increase the model interpretability to provide better support in the development of retention strategies. We feel that the model performance was the main goal in the customer dropout prediction, but the use of the insights generated from the models could be interesting to support

time-related retention strategies than exploring models using only their prediction performance. Additionally, the existing studies addressed nontime-varying variables (Gür Ali & Arıtürk, 2014). Those ideas triggered the need to develop approaches that could consider several perspectives such as interpretability (Benoit & Van den Poel, 2012), classification performance (De Bock & Van den Poel, 2012), and business objectives such profitability (Ekinci et al., 2014).

The understanding of an overall perspective and how this can be explored to develop a better models underlie this research which aims to contribute to new perspectives and the exploration of better approaches to anticipate customer dropout considering a time-related context.

The existing studies address the problem mainly as a technical problem and not as a business problem which underlies the implementation and improvement of existing approaches. **The dropout prediction seems that is mainly addressed using metrics related to the performance of the algorithms without considering their interpretability to support the development of action plans to contribute to the organizations improvement of retention supported in timings when they could occur. This misalignment between existing approaches to predict customer dropout and how to use this supporting business actions is a problem that needs to be addressed**.

1.2 Research Context

Customer analysis is fundamental to developing business and marketing intelligence (Sheth et al., 1998), this allows the understanding of existing historical customer data to extract trends and patterns (Berry & Linoff, 2004). The development of analysis in existing data in organizations sustains a correct targeting of the customers (Nie et al., 2011) to predict customers intentions to dropout (García et al., 2017). This context is also supported by the comprehension that the organizations realized that customers data is a valuable asset (Athanassopoulos, 2000; Jones et al., 2000).

There are several approaches that could be employed. Predictive learning could be adopted but after the appearance of the domain of artificial intelligence (Friedman, 1994), machine learning was considered a modern extension of predictive analytics (Ongsulee et

al., 2018). Machine learning is interpreted as an automated process to extract patterns from the data (Kelleher et al., 2015) generalizing from the examples in the training set (Domingos, 2012). More recently, deep learning has emerged, which is considered a part of machine learning (Dargan et al., 2020), and mimic the behavior of the human brain (Agrawal et al., 2018), relying upon machine learning algorithms that model nonlinear high-level abstractions (Dargan et al., 2020). Machine learning encompasses deep learning and is understood to be a consequence of predictive analytics. However, this process could be also known as data mining, which is the extraction of knowledge from data (Han & Kamber, 2006) using fitting models to determine patterns (Fayyad et al., 1996). Generally, several approaches could be employed, using techniques such as statistics, machine learning, pattern recognition, database and data warehouse systems, information retrieval, visualization, algorithms, and high-performance computing (Han et al., 2012).

The dropout is a concept, that according to Glady et al. (2009) is a marketing-related term that represents a consumer who is switching from one company to a competitor in the near future. Verbeke et al. (2012a) considers that is also a management science problem that adopts a data mining approach trying to solve the challenge related to the lower costs for retaining customers versus the costs of attracting new customers (Alboukaey et al., 2020). The dropout event occurs in two main scenarios (Ascarza, 2018): (1) contractual settings and (2) non-contractual settings. The first is related to events that occur when the customers pay a fee, such a subscription service where the customer has to inform the end of the relationship, the second scenario the organization as to infer if the customer is still purchasing for example books, computers, where is not available in a contractual setting.

Xue et al. (2021) states that contractual scenarios encompass cases such as insurance, telecommunications, and magazine subscriptions, where firms can estimate the cash flow generated by their customers. The main characteristic of a contractual setting is the contact of the customer canceling a subscription (Fader & Hardie, 2007). Additionally, allows also access to records and usage logs (Verbeke et al., 2014). With the anticipation of possible churners is possible to develop countermeasures, such as concessions to retain many customers as possible (Sivasankar & Vijaya, 2019).

The dropout is a problem with consequences in the organization performance, leading to reduced sales, rivals gaining new customers increases the costs of attracting new

customers, and is a risk to the organization image with the lost consumer market and customer base (Amin et al., 2017). This entails a more punitive scenario than non-contractual settings, representing a well-defined end of a relationship (Risselada et al., 2010). It is also known, that the costs of retaining customers are lower when compared to the costs of attracting new ones (Edward & Sahadev, 2011a), reinforced by that the reduction of the dropout rates could represent an increase in the profits (Reichheld, 1996a).

Machine learning algorithms have been used to predict customer dropout (Bandara et al., 2013), and they could support the development of actions before the dropout event occurs. Machine learning could be used to extract knowledge to understand dropout and support the development of effective retention strategies (Verbeke et al., 2011) to identify the discovery of patterns to address dropout. Those patterns could be used to extract knowledge to understand the customer dropout and back the development of effective retention strategies (Verbeke et al., 2011), where the use of decision trees allows us to extract actionable knowledge (Pan et al., 2007; Kim et al., 2001). But, to our knowledge, there is a lack of an overview of research related to the use of machine learning techniques to target customer dropout with contractual settings considering also the timings of the dropout.

This work focuses on using existing data in organizations to anticipate customer dropout in contractual settings. The development of an approach to identify when customer dropout could occur, anticipating this scenario using machine learning.

1.3 Research Questions

Considering the context described in the previous section, the anticipation of customer dropout could be developed using machine learning where organizations could gain competitive advantages. Dropout prediction could be considered a business objective, which requires more than predicting if the customer dropout will occur or not (Devriendt et al., 2019), and considering also, the interpretability to provide better support in the development of retention strategies.

The wider adoption of decision trees and random forests (Antipov & Pokryshevskaya, 2010; Benoit & Van den Poel, 2012) and logistic regression (Coussement et al., 2010), which could be due to its interpretability. Interpretability is a relevant aspect for managers for

the extraction of valuable information to support the development of effective retention strategies (Verbeke et al., 2012a).

Considering the importance of the algorithms to predict customer dropout, based on the decision trees and random forests (Antipov & Pokryshevskaya, 2010; Benoit & Van den Poel, 2012; Burez & Van den Poel, 2007) related to its interpretability and flexibility (Keramati et al., 2014). Interpretability is an important facet for the extraction of information that could be used for the development of retention strategies (Verbeke et al., 2012a). Decision trees based algorithms allow us to extract actionable knowledge (Pan et al., 2007; Kim et al., 2001).

The comprehension when dropout occurs gives insights into when to target actions. Survival analysis consists of a family of techniques to describe the probability of surviving past a specific point in time (Van den Poel & Larivière, 2004), which allows considering a temporal factor and what features influence the length of the relation. However, those approaches encompass several challenges related to the timing of the dropout and the dynamic behavior of a customer with the intention to dropout (Alboukaey et al., 2020). The importance of understanding when dropout will occur and the risk when discarding the temporal perspective of the problem seems to be an element that should be addressed.

Few studies have considered the timings when the dropout occurs (Perianez et al., 2016; Burez & Vandenpoel, 2008; Sobreiro, Martinho, et al., 2022). This gap in the research represents an opportunity to address its influence on the efficiency of the model. There are several approaches employed using ensemble methods or combining various models with a hybrid approach, but to our knowledge, none of the existing studies integrated the survival approach to predict customer dropout, using a hybrid approach (Sobreiro, Martinho, et al., 2022).

The efficiency of the models is more complex than the performance issue of the employed algorithm using metrics, such as Area Under the Curve, sensitivity, specificity, recall, precision, and F-score. The performance of the dropout prediction should be analyzed according to the environment where is develop. In marketing retention strategies the uplift supports the development of proactive actions to tighten the investment in retention strategies (Coussement & Van den Poel, 2008). This perspective underlies in the assumption that top-decile lift allows the development of retention actions on customers that are the most likely to dropout, which allows a more proactive action (Coussement &

Van den Poel, 2008; Xie et al., 2009), considering the top 10% of customers with greater risk, and that the investments should be developed in customers more susceptible to marketing actions instead from those who will leave anyway and are suboptimal as targets Ascarza (2018). However, customers with a higher risk of churning may not be the best targets, and the investments in retention strategies should distinguish churners susceptible to marketing actions from those who will leave anyway (Coussement et al., 2017). Using these models seems to be a good strategy, as they can outperform predictive models that consider only accuracy, instead considering also a profitability business perspective.

Those perspectives imply that the customers with a higher risk of churn could not be the best targets for the development of retention strategies. The business context or the clarification of the business objective underlying the prediction of the customer dropout should be developed before employing machine learning algorithms. Following this idea, the analysis of customer dropout could be developed considering those assumptions, which were formulated the following research questions:

1. What is the state of the art being used to predict customer dropout?
2. Which approaches are being employed to predict customer dropout?
3. Propose a new approach considering existing gaps in research.

1.4 Thesis Objectives

Considering this context, this research tries to analyze the state of the art to identify Machine Learning studies to predict customer dropout to support the development of counteractions before the customer churns, taking into account also when it occurs. Using this theoretical framework we want to explore a novel approach using machine learning to predict customer dropout improving the existing ones.

This thesis addresses how can the historical data be used to predict the customer dropout and support the decision to develop countermeasures to avoid customer desertion in contractual settings.

Following those assumptions were considered the following objectives:

- Identify which approaches could be employed to predict customer dropout considering its binary outcome (dropout/non-dropout) and the time perspective related to dropout;
- Create a model to improve the predicting performance.

1.5 Thesis Contributions

The thesis goals were achieved during its development. This thesis provides several contributions addressing customer dropout. First, we explored the state of the art related to existing studies to create theoretical foundations and research gaps for the thesis development, this process leads to three publications, the first in the IEEE Access Sobreiro, Martinho, et al. (2022) and Sobreiro, Garcia-Alonso, et al. (2022) with an indexing Journal Citation Reports (JCR) publication and Sobreiro et al. (2021) with and indexing SCImago Journal Rank (SJR).

- Sobreiro, P., Garcia-Alonso, J., Martinho, D., & Berrocal, J. (2022). Hybrid Random Forest Survival Model to Predict Customer Membership Dropout. Electronics, 11(20), Art. 20. https://doi.org/10.3390/electronics11203328
- Sobreiro, P., Martinho, D. D. S., Alonso, J. G., & Berrocal, J. (2022). A SLR on Customer Dropout Prediction. IEEE Access, 10, 14529–14547. https://doi.org/10.1109/ACCESS.2022.3146397
- Sobreiro, P., Martinho, D., Berrocal, J., & Alonso, J. (2021). Dropout Prediction: A Systematic Literature Review. CAPSI 2021 Proceedings. https://aisel.aisnet.org/capsi2021/18

In order to propose a new approach to improve the performance in the prediction of customer dropout using survival analysis. A fourth study was developed, exploring the customers of a health club, addressing the research gaps identified, and creating a methodology more suitable. A fifth study was also developed using membership data of customers, to confirm the results identified previously and validate the previous results. Both studies used survival analysis that allowed them to consider timing related to the dropout events, combined with clusters to improve the performance of the model:

- Sobreiro, P., Berrocal, J., Martinho, D., García-Alonso, J. (no prelo). Health club customer dropout membership. Unpublished manuscript.
- Sobreiro, P., Berrocal, J., Martinho, D., García-Alonso, J. (no prelo). Customer dropout membership. Unpublished manuscript.

Additionally, were also developed or collaborated on other studies exploring machine learning algorithms:

- Martinho, D., Sobreiro, P., & Vardasca, R. (2021). Teaching Sentiment in Emergency Online Learning—A Conceptual Model. Education Sciences, 11(2), Art. 2. https://doi.org/10.3390/educsci11020053
- Silva, A., & Sobreiro, P. (2022). Running involvement, loyalty to running, and subjective well-being: A cluster analysis: Implicación en carrera y bienestar subjetivo en corredores. Cuadernos de Psicología Del Deporte, 22(1), Art. 1. https://doi.org/10.6018/cpd.468611
- Silva, A., Sobreiro, P., & Monteiro, D. (2021). Sports Participation and Value of Elite Sports in Predicting Well-Being. Sports, 9(12), Art. 12. https://doi.org/10.3390/sports9120173
- Sobreiro, P., Guedes-Carvalho, P., Santos, A., Pinheiro, P., & Gonçalves, C. (2021). Predicting Fitness Centre Dropout. International Journal of Environmental Research and Public Health, 18(19), Art. 19. https://doi.org/10.3390/ijerph181910465
- Sobreiro, P., Martinho, D., Pratas, A., Garcia-Alonso, J., & Berrocal, J. (2019). Predicting High-Value Customers in a Portuguese Wine Company. Journal of Reviews on Global Economics, 9, 1732–1740. https://doi.org/10.6000/1929-7092.2019.08.155
- Sobreiro, P., Silva, A., Conceição, A., Louro, H., Pinheiro, P., & Guedes de Carvalho, P. (2022). Swimmer Dropout Rate: A Survival Analysis. Apunts Educación Física y Deportes, 147, 74–83. https://doi.org/10.5672/apunts.2014-0983.es.(2022/1).147.08

1.6 Thesis structure

The thesis is organized as follows:

Chapter 1 Introduction. Corresponding to this introduction, which incorporates the thesis origins, the research context of the work developed needed to provide a background to the contents of this thesis. Are also described the main research questions and thesis objectives. Last, it contains also the thesis contributions.

Chapter 2 State of the Art. This chapter reviews existing knowledge supporting the underlying assumptions related to this thesis objectives. Namely, the context and how is being addressed the customer dropout with contractual settings, why do we address this problem? What kind of approaches are being employed? Which data is being used? How is addressed the timing of the dropout? And How is being measured the performance of machine learning algorithms?

Chapter 3 Predicting Customer Dropout. This chapter address the adopted elements used to predict customer dropout, mainly the overall idea beyond survival analysis, clusters, survival trees, the model proposal, and how is the model evaluated. The model evaluation compares the performance of the proposed hybrid survival model, against a regular survival model, without using clusters. Are explained also how the several elements are integrated into an overall approach to creating a hybrid model to predict the timing related to customer dropout.

Chapter 4 Case studies validating the customer dropout prediction. This chapter address the customer dropout prediction using survival analysis based on a random forest survival model. It also explored a hybrid random forest survival model combined with the segmentation of the customers using cluster analysis to increase the prediction performance. The analysis is developed in two datasets, one is related to a customer membership of a sports club and the second is customer data of a fitness center.

Chapter 5 Conclusions and Future Work. This chapter presents the main conclusions drawn after the development of this thesis, papers written during its development, provides some explanation about future works, and finalizes with final conclusions.

Chapter 2

State of the Art

Following this thesis objective, how can the historical data be used to predict customer dropout and support the decision to develop countermeasures to avoid customer desertion in contractual settings? This requires some underlying assumptions.

Data mining is a process of extracting knowledge from data, supporting the comprehension of historical data to identify trends and patterns (Berry & Linoff, 2004). Customer data analysis supports the development of business and marketing intelligence (Sheth et al., 1998), using models to determine patterns in the available data (Fayyad et al., 1996). Traditionally these approaches generally are time-consuming (Hung et al., 2006) and statistics, machine learning, pattern recognition, database and data warehouse systems, information retrieval, visualization, algorithms, and high-performance computing are used to extract patterns through automated and semi-automated methods (Berry & Linoff, 2004; Hung et al., 2006; Quinlan, 1986).

The process is framed in an understanding that organizations consider the customer database as the most valuable asset they possess (Athanassopoulos, 2000; Jones et al., 2000). Using this information managers can support their decisions using more insights about their customers. Allowing them to target the right customers (Nie et al., 2011) and increase their knowledge about how the loyalty mechanisms work to avoid customers intentions to dropout (García et al., 2017). This provides opportunities for organizations to explore and get a competitive advantage against their competitors.

Using existing information about the customers allows the development of retention actions exploring existing information. This means that one of the main goals is to avoid

churn, which according to Glady et al. (2009) is a marketing related concept representing a customer moving from one company to another in a near future. The main idea is to solve the problem related to the lower costs of retention against attracting new customers (Edward & Sahadev, 2011a).

The development of retention actions requires the identification of the customers that have a higher propensity to dropout to target counteractions to avoid dropout (Alboukaey et al., 2020). However, this requires the clarification of voluntary dropout, which represents the the intention of the customer to end the relationship with the company, and involuntary dropout where the customers do not fulfill their obligations (Berry & Linoff, 2004). Other factors according to Neslin et al. (2006) is that the customer dropout is triggered by an event related to a given time period. There are also two contexts, that should be considered related to the dropout (Gupta et al., 2006; Ascarza, 2018): (1) contractual setting, where the customers pay a fee, such as a subscription, and the customer informs the company that they are ending the relationship, and (2) non-contractual settings where the company has to infer if the customer is still active. The scenarios of contractual settings underlie in the assumption that customers need to renew their contracts/memberships/subscriptions to continue (Ascarza & Hardie, 2013), against the consumer ending the relationship (Fader & Hardie, 2007).

This chapter addresses the assumptions underlying the research developed, motivations, and the context used for this thesis.

2.1 What is customer dropout?

Dropout is an event that represents one that abandons an attempt, activity, or chosen path, it could represent a withdrawal from a membership[1]. The membership renewal decision in the case of paid memberships falls under the rubric of repeat buying (Bhattacharya, 1998), which could be organized into two categories: (1) access membership required to access the organization goods; (2) full-choice when the service is available to the customer independently of being member or not. This should be contextualized in a contractual settings or non-contractual settings scenario (Gupta et al., 2006; Ascarza, 2018), where

[1]Merriam-Webster. (n.d.). Dropout. In Merriam-Webster.com dictionary. Retrieved July 27, 2022, from https://www.merriam-webster.com/dictionary/dropout

in the contractual settings is possible to understand the cash flow generated by their customers (Xue et al., 2021) and have access to subscription records and usage logs (Verbeke et al., 2014).

In a contractual setting, the customer has to choose if will dropout or not, which means could renew a contract or not (Prasasti & Ohwada, 2014). This means that in contractual settings the customer dropout represents an explicit ending of a relationship which is more penalizing than non-contractual settings (Risselada et al., 2010), and have implications for the profitability of the organizations increasing marketing costs and reducing sales (Amin et al., 2017).

Several researchers working in the areas of marketing, applied statistics, and data mining has developed a number of models that attempt to either explain or predict customer churn at the next contract renewal occasion (Ascarza & Hardie, 2013), in a contractual scenario Ascarza & Hardie (2013) explored a model to address the customer dropout in membership scenarios, the problem addresses the concept of contractual settings, where the customers need to renew their contracts/memberships/subscriptions to have access to the underlying service.

However, other authors such Berry & Linoff (2004), adopted the term churn, to represent an event related to the decision of the customer to end the relationship with an organization. Berry & Linoff (2004) presents several scenarios: (1) voluntary churn when the customer decide to go stop doing business with an organization; (2) involuntary churn when the company ends the relationship, usually due to lack of payment, and (3) expected churn when the customer is not in the target market, such as families moving away or workers retiring and don't need a retirement plan. The voluntary dropout is not a scenario that is desirable for the organization. When the customer stops fulfilling its obligations, the organization stops the relationship. Lastly, when there are expected churns is possible to anticipate this scenario in some cases. Glady et al. (2009) states that is a marketing-related concept representing a customer moving from one company to another in a near future and Verbeke et al. (2012b) states that is a management science problem. This problem impacts the organizational performance causing reduced sales, and allowing the competitors to gain new customers, and creating a negative image for the organization with the loss of market and customer base (Amin et al., 2017). Customer churn is

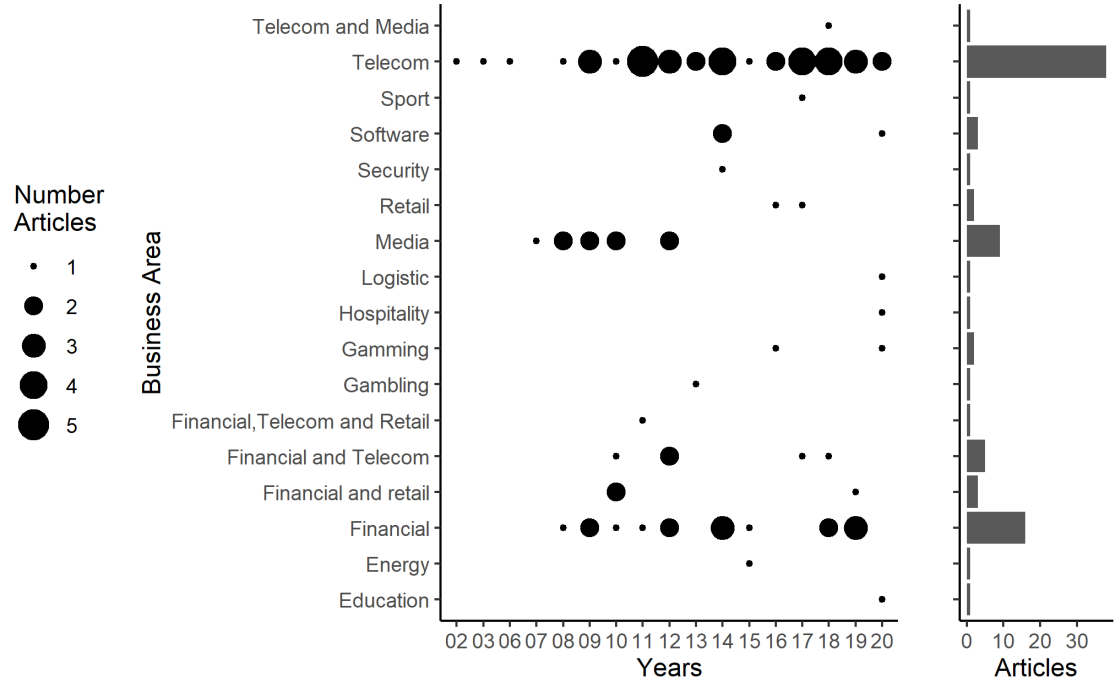


FIGURE 2.1: Studies by business area

more negative than selling fewer products in noncontractual scenarios because represents a well-defined end of a relationship (Risselada et al., 2010).

The dropout has been addressed in multiple areas such: Education (Kotsiantis et al., 2004; Dekker et al., 2009), generally for subscription models (Ascarza & Hardie, 2013), Telecom (Agrawal et al., 2018; Al-Molhem et al., 2019), Media (Ballings & Van den Poel, 2012; Ballings et al., 2012), Financial (Benoit & Van den Poel, 2012; M. A. H. Farquad et al., 2009; Martono et al., 2014; Prasasti & Ohwada, 2014), Logistics (Schaeffer & Rodriguez Sanchez, 2020), Hospitality (Routh et al., 2020), Security (Jiang et al., 2014), Gambling (Coussement & De Bock, 2013), Energy Moeyersoms & Martens (2015), and Gaming Perianez et al. (2016). However, was possible to identify several research gaps in areas under researched (Sobreiro, Martinho, et al., 2022), such as education, sport, logistic, hospitality, gaming, and gambling 2.1. Additionally, was also identified a discard of a temporal perspective when addressing customer dropout.

2.2 Why customer dropout prediction?

The advantage of developing some strategies for retention is supported by the concept that the costs of customer retention are lower than customer acquisition (Fornell & Wernerfelt,

1987; Edward & Sahadev, 2011b), where a reduction of 5% of the dropout could represent almost a duplication of the profits (Reichheld, 1996b). To address this problem the use of the customer's database could be explored, which is considered the most valuable asset that most organizations possess (Athanassopoulos, 2000). The development of a customer retention strategy could be supported in the identification of the customers that will dropout (Alboukaey et al., 2020), using churn prediction models to detect customers with a high propensity to dropout (Verbeke et al., 2011).

Why should dropout prediction be addressed? Churn dropout prediction is a problem being addressed supported by the idea that the customers database is the most valuable asset that organizations possess (Athanassopoulos, 2000), which requires determining customers that will attrite (Alboukaey et al., 2020). Dropout implies in contractual settings that the customer needs to renew their contracts to continue its use (Ascarza & Hardie, 2013). However, in contractual settings, the customer dropout represents an explicit ending of a relationship that is more punitive than in non-contractual settings (Risselada et al., 2010), which has implications for the profitability of the organizations increasing marketing costs and reducing sales (Amin et al., 2017). The main idea is that companies are more profitable in retaining more customers due to lower marketing costs and greater sales (Amin et al., 2017). The reduced costs in retention against acquiring new customers which according to Bhattacharya (1998) cost more 5 or 6 times, allowing to maintain the organization profitability (Devriendt et al., 2019). Reichheld (1996a) stated that the reduction of the dropout rates by 5% could represent an increase in the profits up to double. By anticipating the dropout, organizations can develop of countermeasures to reduce customer churn. Several studies address this problem by trying to improve profitability (Coussement & Van den Poel, 2009; García et al., 2017; Devriendt et al., 2019).

The development of analysis on the customer database available in the organizations allows targeting the right customers to retain (Nie et al., 2011), using the most valuable asset (Athanassopoulos, 2000; Jones et al., 2000). This solves the problem related to the lower costs of retaining customers against attracting new ones (Edward & Sahadev, 2011a).

How to identify customers with a higher propensity to dropout? The loss of customers could be measured as a percentage rate of the customers ending the relation (Shirazi & Mohammadi, 2019) or as a churn probability calculated for each customer using historical data to predict their future behavior (Coussement et al., 2017). This could be developed

using customer analysis for the development of business and marketing intelligence (Sheth et al., 1998), where in contractual scenarios the organization has access to subscription records and interaction logs (Verbeke et al., 2014). Data mining allows to develop an approach to solve the problem related to lower costs retaining customers against attracting new ones (Verbeke et al., 2012b). Data mining provides means to answer business questions that are time-consuming (Hung et al., 2006) using techniques such as statistics, machine learning, pattern recognition, database and data warehouse systems, information retrieval, visualization algorithms, and high-performance computing (Han et al., 2012) to extract patterns using automated or semiautomated methods (Berry & Linoff, 2004; Han & Kamber, 2006; Quinlan, 1986).

Machine Learning have being used to predict customer dropout in the telecommunications sector (Wei & Chiu, 2002; Wai-Ho Au et al., 2003; Hung et al., 2006) and media (Burez & Van den Poel, 2007). This allows the development of automated extraction patterns from data (Kelleher et al., 2015) to support the development of customer retention strategies based on existing data (Verbeke et al., 2011). Machine learning is used to develop churn prediction models generalizing the relationship between churn and historical data to support the predictions of future behavior of the customers influenced by the input data and algorithm used model (De Bock & Van den Poel, 2011). This means that machine learning could be used to predict a future dropout or nondropout, considering two outcome events.

The number of publications predicting customer dropout started to increase in 2008, initially targeting the media, financial and telecom areas (Coussement & Van den Poel, 2008; Shao et al., 2008; Xia & Jin, 2008; Burez & Vandenpoel, 2008). Certain areas only addressed the problem more recently (in 2020), such as logistics (Schaeffer & Rodriguez Sanchez, 2020), hospitality (Routh et al., 2020) and education (Delen et al., 2020) There could be used several algorithms such De Bock & Van den Poel (2011): logistic regression, artificial neural networks, survival analysis, Markov chains, support vector machines, generalized additive models, decision trees, naive Bayes classifiers, K-nearest neighbor classifiers, random forests, cost-sensitive classifiers, and evolutionary algorithms.

However, it seems there is a lack of research addressing customer dropout using contractual settings, which has been mainly developed in the telecom and financial areas, such as education, logistics, and hospitality (Sobreiro, Martinho, et al., 2022). Where

much of the existing subscription models seems to be an uncharted territory (e.g. Sport subscription services or energy sector). Additionally, the research seems to be lacking hybrid approaches integrating different algorithms to allow the extraction of actionable information and integrating time-related variables, instead of using only static models. Without those dynamic approaches, the prediction models should be adapted regularly (Risselada et al., 2010) by considering "time" as an important feature. The dynamic behavior of the customer over time assumes that decisions change over time and that the intention to dropout at the end of the month could not happen (Alboukaey et al., 2020).

2.3 Approaches being used to predict customer dropout

The dropout prediction allows the implementation of strategies to increase the life expectancy of the customers developing strategies that create value (García et al., 2017). There are two approaches to modeling customer dropout, as a binary outcome, predicting who will churn or not and the second approach to estimate the customers' remaining lifetime (Berry & Linoff, 2004). Other researchers to improve the prediction accuracy suggested also the use of hybrid approaches, combining clusters with classification algorithms (Gök et al., 2015; Vijaya & Sivasankar, 2019a). The following section addresses the prediction as a binary outcome, which according to Berry & Linoff (2004) have advantages in short temporal horizons.

2.3.1 Customer dropout binary outcome

There are several categories of algorithms that could be used (Sobreiro, Martinho, et al., 2022): (1) nature-inspired; (2) ensemble; (3) decision trees; (5) statistical; (6) rule-based; (7) clusters; (8) markov chain; (9) support vector machines (10) other approaches. Nature-inspired Fang et al. (2015), which encompass nature-inspired algorithms, including artificial neural networks, fuzzy systems, evolutionary computing, and swarm intelligence. Ensemble, which uses multiple classifiers built independently and participate in a voting procedure to obtain a final class prediction, such as random forests, bagging, and boosting Verbeke et al. (2012a). Decision trees, which are one of the most popular classification techniques Keramati et al. (2014). Statistical, such as logistic regression, naive Bayes, and Bayesian networks Verbeke et al. (2012a). Rule-based, which allows us to reduce

the amount of information to understandable statistically supported statements Keramati et al. (2014). Clusters allow classifying customers Kianmehr & Alhajj (2009). Survival analysis methods, which model the occurrence and timing of events Van den Poel & Larivière (2004). Markov chain Burez & Van den Poel (2007). SVM, which classifies the data using the maximal margin hyperplane Huang et al. (2009). Other approaches were not integrated into the previous categories.

Nature-inspired algorithms are algorithms inspired in nature such as neural networks or based on swarm intelligence (Fang et al., 2015). This category explores the idea related to biological systems using the concept of neurons (interconnected elements) working together to solve problems (Keramati et al., 2014). Ensemble methods use multiple base classifiers to improve the prediction performance, which is considered one of the best approaches to model customer churn (Idris et al., 2013). The most common ensemble method is the random forest to predict customer dropout (Sobreiro, Martinho, et al., 2022), having several advantages (Coussement & Van den Poel, 2008; Burez & Vandenpoel, 2008): (1) predictive performance; (2) robustness to outliers; (3) reasonable computing time and (4) easy to implement. Decision trees are one of the most used algorithms to predict customer dropout (Sobreiro, Martinho, et al., 2022), the reasons are related to easy use (Nie et al., 2011), interpretability (Burez & Vandenpoel, 2008), and conceptual transparency (Keramati et al., 2014). The statistical category includes algorithms such as logistic regression, a simple method that provides good performance results against other algorithms when is considered tuning the data (Coussement et al., 2017). Rule category encompasses several algorithms providing an approach to extract rules, such as association rules to reduce large amount information to small and understandable amounts of information (Keramati et al., 2014).

The cluster category allows the development of customer segmentation to support machine learning in each segment (Hung et al., 2006; Vijaya & Sivasankar, 2019a; Sivasankar & Vijaya, 2019). The common approach is to combine clustering algorithms in parallel with classification methods to have more control in the decision process of churn management optimizing the classification (Jafari-Marandi et al., 2020).

Survival analysis allows developing a dynamic approach over a period of time, considering the time when the dropout occurs, against approaches that are static. There are several algorithms such as random survival forest (Routh et al., 2020).

Markov chain explores the concept of random processes independent from the past considering the state of a previous event. This approach considers a current state, using algorithms such as multivariate time series, transfer learning, random walk, active based learning, social networks, and fuzzy classifiers (Sobreiro, Martinho, et al., 2022).

Support vector machines algorithms allow finding optimal hyperplanes maximizing the margin between positive and negative examples (e.g. dropout or nondropout). This type of approach is considered to have higher performance (Verbeke et al., 2011). Other approaches such active learning (Verbeke et al., 2011), fuzzy classifiers (Azeem et al., 2017) and random walk (Liu et al., 2020) can be also employed.

An overview of different types of algorithms that could be employed to predict customer dropout with contractual settings are presented in table 2.1, where were identified almost 80 different types of algorithms. The most common was the category of statistical (56 articles) and ensemble (56 articles), next decision tree with 54 articles, nature-inspired (38 articles), rule-based (8 articles), clusters (14 articles), survival (4 articles), Markov (4 articles), SVM (19 articles) and others (7 articles).

2.3.2 Customers remaining lifetime

The approaches normally adopted use a dependent variable representing dropout or non-dropout, without considering a dynamic perspective that the dropout risk changes over time (Alboukaey et al., 2020). The survival models try to solve this limitation (Routh et al., 2020) capturing a temporal dimension of the customer dropout (Perianez et al., 2016). Perianez et al. (2016) used survival analysis to predict also when the dropout will occur. This approach considers the survival time of a customer as a prediction, instead only considering a binary outcome (dropout/nondropout). Survival analysis allows to develop the analysis of the time until an event of interest and exploring its relationships with different factors. It is also known as time-to-event analysis, in which the roots and terminology comes from medical research and failure analysis in manufacturing, centered on the tenure of the customer, and providing a way to understand (Berry & Linoff, 2004):

- When a customer is likely to leave
- The next time the customer is likely to move to a new customer segment

TABLE 2.1: Categories of algorithms adapted from Sobreiro, Martinho, et al. (2022)

Type	Algorithm	#	Articles
nature-inspired	neural networks	20	(Agrawal et al., 2018; Esteves & Mendes-Moreira, 2016; Glady et al., 2009; Hung et al., 2006; Keramati et al., 2014; Lee & Jo, 2010; Liu et al., 2020; Mohanty & Naga Ratna Sree, 2018; Óskarsdóttir et al., 2018; Prasasti & Ohwada, 2014; Semrl & Matei, 2017; Sivasankar & Vijaya, 2019; Tsai & Chen, 2010; Verbeke et al., 2012a; Wai-Ho Au et al., 2003; Wei & Chiu, 2002; Xia & Jin, 2008; Xiao et al., 2012; Xie et al., 2009; Zhang et al., 2012)
	genetic	5	(Amin et al., 2017; Kianmehr & Alhajj, 2011a; Shao et al., 2008; Wai-Ho Au et al., 2003; Xiao et al., 2012)
	multilayer perceptron	4	(Azeem et al., 2017; Coussement et al., 2017; Huang et al., 2009; Jafari-Marandi et al., 2020; Verbeke et al., 2012a)
	extreme learning machines	2	(Agrawal et al., 2018; Mohanty & Naga Ratna Sree, 2018)
	particle swarm optimization	1	(Vijaya & Sivasankar, 2019a)
	counter propagation neural networks	1	(Mohanty & Rani, 2015)
	fuzzy ARTMAP	1	(Mohanty & Rani, 2015)
	long short term memory	1	(Alboukaey et al., 2020)
	convolutional neural network	1	(Alboukaey et al., 2020)
	self-organizing error-driven	1	(Jafari-Marandi et al., 2020)
	ant colony optimization	1	(Verbeke et al., 2011)
ensemble	random forest	30	(Alboukaey et al., 2020; Amornvetchayakul & Phumchusri, 2020; Ascarza, 2018; Azeem et al., 2017; Ballings & Van den Poel, 2012; Benoit & Van den Poel, 2012; Burez & Vandenpoel, 2008; Burez & Van den Poel, 2007, 2009; Coussement et al., 2017; Coussement & Van den Poel, 2009, 2008; De Bock & Van den Poel, 2011, 2010, 2012; Devriendt et al., 2019; Esteves & Mendes-Moreira, 2016; Idris et al., 2013; Kaya et al., 2018; Liu et al., 2020; Martono et al., 2014; Mitrović et al., 2018; Óskarsdóttir et al., 2018; Prasasti & Ohwada, 2014; Schaeffer & Rodriguez Sanchez, 2020; Semrl & Matei, 2017; Verbeke et al., 2012a, 2014; Vijaya & Sivasankar, 2019a; Zhu et al., 2018)
	bagging	6	(Azeem et al., 2017; Coussement et al., 2017; De Bock & Van den Poel, 2012; Verbeke et al., 2012a, 2014; Xiao et al., 2015)
	adaboost	6	(Azeem et al., 2017; De Bock & Van den Poel, 2010; Esteves & Mendes-Moreira, 2016; Jafari-Marandi et al., 2020; Schaeffer & Rodriguez Sanchez, 2020; Wang & Xiao, 2011)
	boosting	4	(Azeem et al., 2017; Coussement et al., 2017; Esteves & Mendes-Moreira, 2016; Verbeke et al., 2012a)
	rootboost	2	(De Bock & Van den Poel, 2011; Idris et al., 2013)
	rotation forest	2	(De Bock & Van den Poel, 2011; Idris et al., 2013)
	extreme gradient boosting	1	(Óskarsdóttir et al., 2018)
	gradient boosting	1	(Coussement et al., 2017)
	logistic model tree	1	(Verbeke et al., 2012a)
	decorate	1	(Idris et al., 2013)
	random subspace method	1	(De Bock & Van den Poel, 2012)
	feature-selection-based dynamic transfer ensemble	1	(Xiao et al., 2015)
decision tree	decision tree	52	(Amornvetchayakul & Phumchusri, 2020; Antipov & Pokryshevskaya, 2010; Azeem et al., 2017; Ballings & Van den Poel, 2012; De Bock & Van den Poel, 2011; Glady et al., 2009; Gür Ali & Arıtürk, 2014; Huang et al., 2009; Agrawal et al., 2018; Hung et al., 2006; Hutchison et al., 2010; Jafari-Marandi et al., 2020; Keramati et al., 2014; Lee & Jo, 2010; Liao & Chueh, 2011; Liu et al., 2020; Nie et al., 2011; Óskarsdóttir et al., 2018; Perianez et al., 2016; Radosavljevik & van der Putten, 2013; Risselada et al., 2010; Semrl & Matei, 2017; Sivasankar & Vijaya, 2019; Tsai & Chen, 2010; Verbeke et al., 2011, 2012a; Vijaya & Sivasankar, 2019a; Wai-Ho Au et al., 2003; Wei & Chiu, 2002; Xia & Jin, 2008; Xie et al., 2009; Zhang et al., 2012; Coussement & De Bock, 2013; Gök et al., 2015; Mohanty & Rani, 2015; Verbeke et al., 2012a; Xiao et al., 2012)
	alternating decision tree	2	(Verbeke et al., 2012a, 2014)
	Chi-Square automatic interaction detection	2	(Antipov & Pokryshevskaya, 2010; Radosavljevik & van der Putten, 2013)
statistical	logistic regression	35	(Amornvetchayakul & Phumchusri, 2020; Antipov & Pokryshevskaya, 2010; Azeem et al., 2017; Ballings & Van den Poel, 2012; Burez & Van den Poel, 2007, 2009; Coussement et al., 2010; Coussement & Van den Poel, 2009, 2008; De Bock & Van den Poel, 2012; Devriendt et al., 2019; Dierkes et al., 2011; Glady et al., 2009; Gür Ali & Arıtürk, 2014; He et al., 2014; Huang et al., 2009; Jiang et al., 2014; Liu et al., 2020; Mitrovic et al., 2017; Mitrović et al., 2018; Moeyersoms & Martens, 2015; Nie et al., 2011; Óskarsdóttir et al., 2018; Radosavljevik & van der Putten, 2013; Risselada et al., 2010; Saravanan & Vijay Raajaa, 2012; Semrl & Matei, 2017; Verbeke et al., 2011, 2012a, 2014; Verbraken et al., 2014; Sivasankar & Vijaya, 2019; Xia & Jin, 2008; Zhang et al., 2012; Zhu et al., 2018)
	Bayesian	17	(Azeem et al., 2017; M. A. H. Farquad et al., 2009; Hutchison et al., 2010; Lee & Jo, 2010; Shao et al., 2008; Sivasankar & Vijaya, 2019; Verbeke et al., 2014; Xia & Jin, 2008; Esteves & Mendes-Moreira, 2016; Perianez et al., 2016; Verbeke et al., 2012a; Verbraken et al., 2014; Vijaya & Sivasankar, 2019a; Coussement et al., 2017; M. Farquad et al., 2012; M. A. H. Farquad et al., 2014; Delen et al., 2020)
	general additive models	3	(Coussement et al., 2010; Coussement & De Bock, 2013; De Bock & Van den Poel, 2012)
	general linear model	1	(Ascarza, 2018)
rule based	decision rule	2	(Tsai & Chen, 2010; Wei & Chiu, 2002)
	ripper	2	(Verbeke et al., 2011, 2012a)
	part	1	(Verbeke et al., 2012a)
	exhaustive algorithm	1	(Amin et al., 2017)
	covering algorithm	1	(Amin et al., 2017)
	LEM2	1	(Amin et al., 2017)
clusters	k-means	8	(Esteves & Mendes-Moreira, 2016; Hung et al., 2006; Keramati et al., 2014; Schaeffer & Rodriguez Sanchez, 2020; Sivasankar & Vijaya, 2019; Ullah et al., 2019; Verbeke et al., 2012a; Vijaya & Sivasankar, 2019a)
	fuzzy c-means	2	(Sivasankar & Vijaya, 2019; Vijaya et al., 2019)
	possibility c-means	1	(Vijaya et al., 2019)
	hierarchical	1	(Kianmehr & Alhajj, 2009)
	cluster-based local outlier factors	1	Ullah et al. (2019)
	local outlier factors	1	(Ullah et al., 2019)
survival	survival analysis (Kaplan Meyer or cox regression)	3	(Perianez et al., 2016; Burez & Vandenpoel, 2008; Gür Ali & Arıtürk, 2014)
	random survival forest	1	(Routh et al., 2020)
Markov	Markov chain	2	(Burez & Van den Poel, 2007; Verbraken et al., 2014)
	Markov logic network	2	(Dierkes et al., 2011; Verbraken et al., 2014)
SVM	support vector machine	29	(Amornvetchayakul & Phumchusri, 2020; Ascarza, 2018; Azeem et al., 2017; Coussement et al., 2017; Coussement & Van den Poel, 2009, 2008; M. A. H. Farquad et al., 2009; M. Farquad et al., 2012; M. A. H. Farquad et al., 2014; Gök et al., 2015; He et al., 2014; Huang et al., 2009; Hutchison et al., 2010; Keramati et al., 2014; Kianmehr & Alhajj, 2011a; Lee & Jo, 2010; Liu et al., 2020; Moeyersoms & Martens, 2015; Óskarsdóttir et al., 2018; Perianez et al., 2016; Prasasti & Ohwada, 2014; Schaeffer & Rodriguez Sanchez, 2020; Sivasankar & Vijaya, 2019; Verbeke et al., 2011, 2012a; Vijaya & Sivasankar, 2019a; Wang & Xiao, 2011; Xia & Jin, 2008; Zhu et al., 2018)
others	multivariate time series, transfer learning, random walk, active learning, social networks, fuzzy classifier	7	(Alboukaey et al., 2020; Schaeffer & Rodriguez Sanchez, 2020; Xiao et al., 2015; Liu et al., 2020; Verbeke et al., 2011; Al-Molhem et al., 2019; Azeem et al., 2017)

- The next time a customer is likely to broaden or narrow the customer relationship
- The factors in the customer relationship that increase or decrease likely tenure
- Quantitative effects of several factors on the customer tenure

Survival analysis, which originally comes from biomedical statistics, is especially well-suited to studying the timing of events in longitudinal data (Singer & Willett, 1993). Allows examining not only if an event occurred but also how long it took to occur. However, the main advantage is related to the concept of censoring, which indicates the number of cases that are not complete to the event of dropout, for example, customers without dropout that is also incorporated in the analysis. These customers are still active and we don't know if the event of dropout has occurred, which is considered censorship. The survival models incorporate this and improve the prediction of the time to the event used for example in the regression models, that only consider the customers that already had dropout, instead, they calculate the probability of the event occurring in a particular time.

There are several challenges to identify the timing related to dropout, considering the dynamic behavior of the customer intention to dropout (Alboukaey et al., 2020), considering that could change over time. The importance of understanding when dropout will occur, and the risk of discarding the temporal perspective of the problem seems to be an element that should be addressed. Few studies considered this (Perianez et al., 2016; Burez & Vandenpoel, 2008).

There are several approaches that could be employed that are organized into three different types of statistical methods (Wang et al., 2017): non-parametric; semi-parametric and parametric (figure 2.2). More recently appeared the machine learning methods encompassing survival trees, Bayesian methods, Neural Networks, Support Vector Machines and Advanced Machine Learning.

One advantage is the simplicity of the analysis and interpretation, the survival probabilities are presented as a survival curve. The survival curve is a representation of the survival probabilities corresponding to the time when the events are observed (Bland & Altman, 1998). The survival curves were developed using the package lifelines (Davidson-Pilon, 2021). Where, the time of dropout is represented by T, which is a non-negative random variable, indicating the time period of the event occurring for a randomly selected individual from the population, representing the probability of an event to occur each

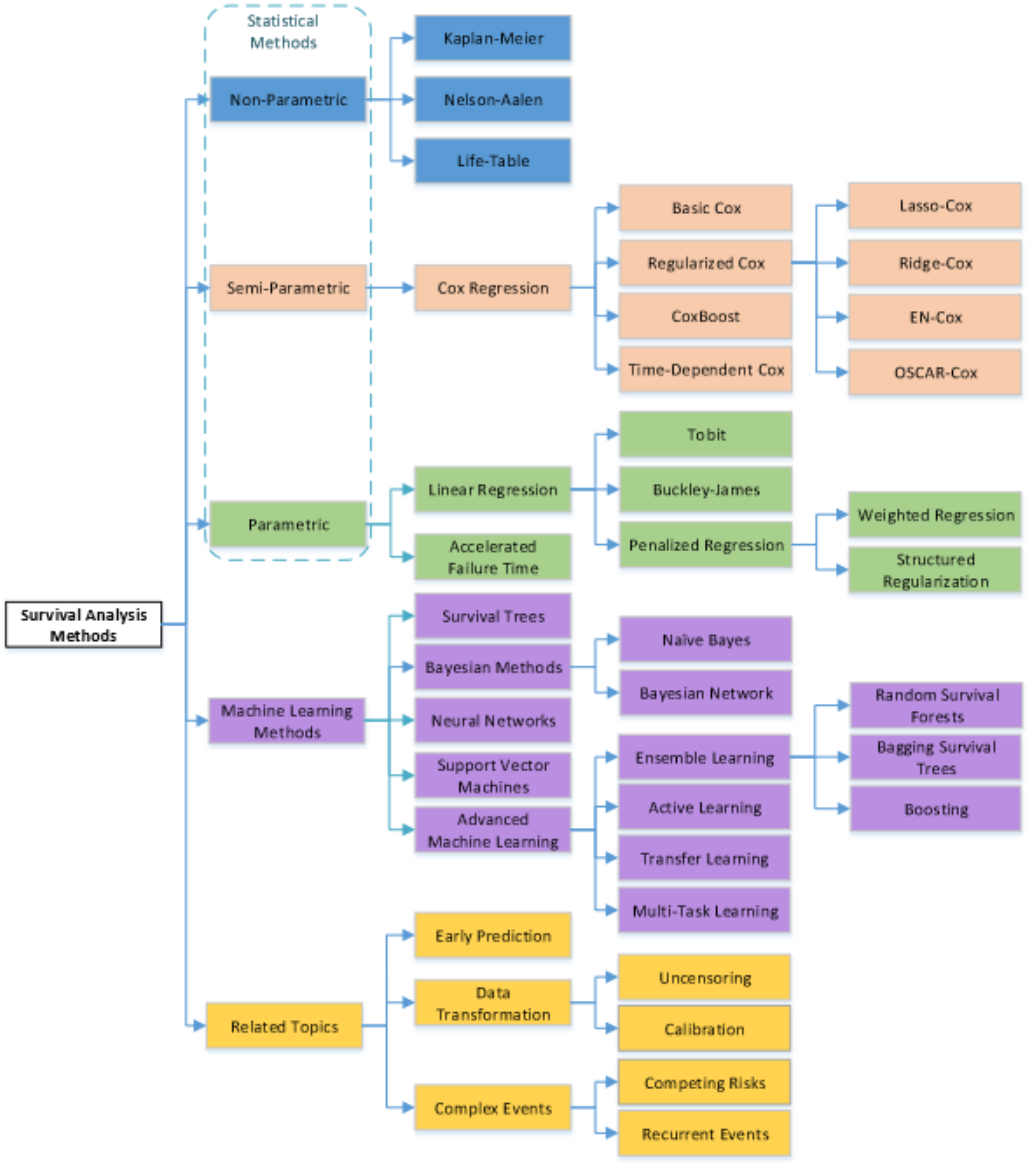


FIGURE 2.2: Existing methods to develop survival analysis adapted from Wang et al. (2017)

time period given that has not already occurred in a previous time period, known as the discrete-time hazard function (Singer & Willett, 1993). The survival function represents the probability of an individual surviving after time t, S(t) = P(T > t), t ≥ 0, with the properties S(0) = 1,S(∞) = 0. The distribution function is represented with F, defined as F(t) = P(T ≤ 0), for t ≥ 0. The function of probability density is represented with f where:

$$f(t) = \lim_{dt \to 0^+} \frac{P[t \leq T < t + dt]}{dt} \tag{2.1}$$

$f(t)dt$ represents the probability of an event occurring in the moment t. The need to represent the distribution evolution of the death probability over time, uses to the hazard function, represented as:

$$\lambda(t) = \lim_{dt \to 0^+} \frac{P[t \leq T < t + dt | T \geq t]}{dt} \tag{2.2}$$

The determination of the survival curves is based on the following elements: (1) the total value of observations removed during the time period (e.g. days, months, or years), either by dropout or by censorship; (2) observations that composed the sample of the study; (3) customers who had not yet dropped out at any given time. The survival probability until the time period ii (p_i) is calculated with:

$$p_i = \frac{r_i - d_i}{r_i} \tag{2.3}$$

Where r_i is the number of individuals that survived at the beginning of the period, d_i the number of individuals who left during the period. The survival time estimate was also taken considering the month in which it is found (estimated). Cox's allow test difference between survival times. The advantage of using survival analysis was that allows us to detect if the risk of an event differs systematically across different people, using specific predictors. The coefficients in a Cox regression were related to the hazard, where a positive value represents a worse prognosis and the opposite, negative value a better prognosis. The advantage of survival analysis was that allow us also to include information on covariates that was censored up to the censoring event.

2.3.3 Hybrid approaches

Several studies proposed also the integration of several algorithms to improve the performance in the prediction of the dropout such the usage of clusters combined with churn prediction (Hung et al., 2006; Gök et al., 2015; Vijaya & Sivasankar, 2019a) The idea relies in the assumption that combining customers in different clusters improves the prediction accuracy. Vijaya & Sivasankar (2019a) suggested the adoption hybrid models combining more than one classier to achieving increased performance compared to those using single classifiers.

The variety and exploration of different approaches to address dropout accuracy has led to the integration of several algorithms to increase performance.

The hybrid approach using clustering and classification, which segments the customers before developing a classification, could be effective (Jafari-Marandi et al., 2020). An approach that is different from ensemble methods uses several models that are built independently before developing a final step that combine different models to create a final class prediction; the approach applies a pipeline concept, where the output of one algorithm server as the input for another algorithm (M. A. H. Farquad et al., 2014).

2.4 What features are being used?

The dropout prediction is based on the use of several features. The standard adopted features in predictive models are based on structured data: (1) socio-demographic data, number of products purchased and (2) relational or behavioral data such as transactional invoicing data, phone call or other networked data (Moeyersoms & Martens, 2015). Risselada et al. (2010) used variables such as demographic information, socioeconomic information(income) and commitment and the relationship to the length (time in the relationship with the company), breadth (usage of other services), and depth (type of service purchased) to predict customer dropout. Another study explored the use of new types of data, such as customers interactions with the organization, using a survey to address service quality, dropout, and economic indicators reflecting external economic indicators that could influence dropout (Gür Ali & Arıtürk, 2014). Ballings et al. (2012) explored the use of pictorial data to represent the feelings and mindset, which is related to the sentiment of the customer, features that are not possible to extract from internal processes in the organization, such as purchase intentions, interests, satisfaction, and emotions. Benedek et al. (2014) exploited network topological properties to represent if the calls are made to customers within the same telecommunication service or to a customer using another telecommunication service. Kaya et al. (2018) adopted spatio-temporal data to explore the time and location as behavioral data. Moeyersoms & Martens (2015) explored the use of high-cardinality attributes which are categorical attributes such as family names, ZIP codes, or bank account numbers.

Other approaches, explored the dropout prediction problem using RFM (Recency, Frequency, Monetary) model generated variables (Mitrovic et al., 2017; Coussement & Van den Poel, 2009; Coussement & De Bock, 2013). Mitrovic et al. (2017) exploited both structural and interaction information from the Call Detail Records enhanced by RFM define the following measures on a customer level: 1) how recent the event occurred, i.e. how long is the time interval between the event's last occurrence and the moment of reference in time; 2) how frequently the event occurred, i.e. what is the number of event occurrences in the observed time frame; 3) what is the monetary aspect of the event, i.e. how much money the customer spent related to the event.

Al-Molhem et al. (2019) exploited social network analysis to heighten the results of churn prediction models in the telecom domain using call detail records to construct a weighted graph representing how close two subscribers are to each other to calculate the centrality. The usage of marketing-related variables, such as promotions offered to a customer, calls developed in a retention strategy, and helpdesk interactions, were used by Verbeke et al. (2012a). (Verbeke et al., 2012a) explored the idea that although retention strategies in customers in risk of dropout are a good assumption, suggesting the optimization of an optimal set of customers considering their highest predicted probabilities to churn in a retention campaign, which allows increasing profitability in targeting mainly customers with a lower risk of dropout.

The selection of the features and how they are used is very important to the dropout prediction. The selection of features is an important and critical step (Azeem et al., 2017), where can be adopted existing processes to extract data, such as the Cross Industry Standard Process for Data Mining (CRISP-DM), which is developed in six stages, i.e., business understanding, data understanding, data preprocessing, modeling, evaluation, and deployment (Coussement et al., 2017).

The data analysis uses the extracted patterns to develop retention strategies, which depend of the quality of the learning using the training set (Jafari-Marandi et al., 2020). Coussement et al. (2017) recommends investing in data preparation, which provides better results against methods that commonly have better performance, such as neural networks or ensemble methods (e.g., random forests), using decision-tree-based to remap for categorical variables, equal frequency binning for continuous variables, and weigh-of-evidence conversion as the representation method, proposing the following techniques for

data preparation:

- Transformation of categorical and continuous variables in values;
- Remapping categorical variables;
- Discretization(binning) continuous variables;
- Dummy encoding;
- Weight of evidence for calculating the strength of a category to separate the churners;
- Variable selection using heuristics, such as sub-setting variables with higher correlation with dependent variables and low inter-correlation between independent variables.

The preparation of the dataset to be used in the model prediction is a fundamental step before the development of the prediction model.

2.5 What features are being explored in survival analysis?

The problem assumption lies in the dynamic behavior of the customer, based on the premise that decisions change over time and a customer with the intention to dropout may not engage in this behavior later (Alboukaey et al., 2020). This is a less common approach, which tries to figure out how long a customer To consider this dynamic perspective survival models have been developed (Routh et al., 2020). These models solve important limitations in traditional methods, such as regressions, which would be appropriate only when all customers end the relationship and do not allow and consider censored data, i.e. observations with incomplete information about churn time (Perianez et al., 2016). Censoring represents situations where the event of interest is not observed, either due to the time limitation of the study period or losing track during the observation period. This means that there are cases where the event of interest (dropout) is experienced and in other cases, the event did not occur. In these contexts, is not appropriate to apply predictive algorithms using standard statistical and machine learning approaches to analyze the survival data (Wang et al., 2017). Survival analysis provides various mechanisms to handle such censored data (Wang et al., 2017).

Several studies addressed the time-related variables (Perianez et al., 2016; Burez & Vandenpoel, 2008; Wei & Chiu, 2002; Liu et al., 2020). Perianez et al. (2016) addressed churn predicting for high-value players in the video game industry using survival trees, considering that semi-parametric techniques present difficulties due to its assumptions that the variables should be constant and have complications to scale to big data problems. The parametric techniques, on the other hand, have to follow a distribution (e.g. Weibull, lognormal, or exponential) which according to Perianez et al. (2016) is uncommon that the data follow these distributions shapes.

This time perspective is an important perspective to address. Where survival analysis is considered an appropriate approach (Berry & Linoff, 2004), which as advantages in relation to dropout binary prediction mainly to make forecasts into the future, and provide insights into customer loyalty and customer value. Agrawal et al. (2018) states that the duration of the customer relationship with the organization allows to analyze the tenure of the relation, where long relations have lower dropout risks, against higher risk in relations lower than 18 months. The time perspective is also explored by Burez & Vandenpoel (2008) finding that one in three customers leave the company before one year, and half the customers leave within two years. Those perspectives give an additional view to identify when the customer dropout and when should be developed retention strategies considering the timings of the events.

The duration of the relationship between the customer and the organization is an important feature that allows us to understand that the decision of the customer to dropout changes over time. The time perspective allows us to identify momentums when the retention actions should be developed.

Perianez et al. (2016) explored the use of variables such: as time until dropout, player attention, player loyalty, playing intensity, and player level. One requirement in survival analysis is if the dropout occurs during a specific time window is considered dropout, all the other situations are considered censoring. Bansal et al. (2019) used demographic variables such: as age, income, and occupation to predict the usage period of using mobile phones. Zhou & McArdle (2015) examined the potential of using survival trees to predictors: personal data dichotomous variable indicating person crime (such as assault or kidnapping), property dichotomous variable indicating property-related offense, and age at the time of release. The described cases used mainly a duration (T) and other variables

such as demographics to analyze their effect on the survival time, however, most cases that deal with customer data consider demographic data (Wang et al., 2017).

2.6 Measurement of the performance of machine learning algorithms

There are several metrics such as the area under the receiver operating curve (AUC), sensitivity, specificity, recall, precision, and F-score (Jafari-Marandi et al., 2020). In the context of the churn prediction, false negatives are five times as undesirable as false positives. The most commonly employed indicator to measure performance was accuracy, which was utilized in 39 studies, followed by roc auc in 36 and sensitivity in 36, lift in 26, precision in 16, specificity in 9, and f-measure in 13 (Sobreiro, Martinho, et al., 2022). The analysis of the performance should take into account not placing more emphasis on one class over another class, this means that it is not biased against the minority class (Burez & Van den Poel, 2009) and that the performance analysis should address the imbalance in the datasets (Kaya et al., 2018).

The accuracy metric reflects the percentage of the corrected cases identified which is calculated as a ratio between the number of correct predictions against the total number of prediction (Coussement & Van den Poel, 2008). Sensitivity, recall, or true positive represents the positive cases (customers that dropout) in relation to their correct prediction and incorrect cases (false negatives). The ROC AUC (receiver operating characteristic area under the curve) is a trade-off the hit rate and the error rate. The AUC is a numerical value between 0.5 and 1 that combines the sensitivity and the specificity, which is the number of negatives (customers who did not drop out) against the total number of customers. Several studies adopted this metric (Alboukaey et al., 2020; Antipov & Pokryshevskaya, 2010; Azeem et al., 2017; Ballings & Van den Poel, 2012; Ballings et al., 2012; Benoit & Van den Poel, 2012; Burez & Vandenpoel, 2008; Burez & Van den Poel, 2007, 2009; Coussement et al., 2010, 2017; Coussement & Van den Poel, 2009, 2008; De Bock & Van den Poel, 2011, 2012; Delen et al., 2020; Dierkes et al., 2011; Esteves & Mendes-Moreira, 2016; Gür Ali & Arıtürk, 2014; Idris et al., 2013; Kaya et al., 2018; Liu et al., 2020; Mitrovic et al., 2017; Mitrović et al., 2018; Óskarsdóttir et al., 2018; Perianez et al., 2016; Semrl & Matei, 2017; Wang & Xiao, 2011).

Precision represents the fraction of the detected class members that are actually correct (Kianmehr & Alhajj, 2009) and specificity (Esteves & Mendes-Moreira, 2016; M. Farquad et al., 2012; M. A. H. Farquad et al., 2009, 2014; Hutchison et al., 2010; Idris et al., 2013; Mohanty & Rani, 2015), which is also identified as the true negative (Gök et al., 2015; Vijaya & Sivasankar, 2019a), and represent the true negative rate by measuring the proportion of non-churners that are correctly classified (Verbeke et al., 2012a). The f-measure is calculated with the harmonic mean of recall and precision (Alboukaey et al., 2020; Amin et al., 2017; Amornvetchayakul & Phumchusri, 2020; Gök et al., 2015; He et al., 2014; Jafari-Marandi et al., 2020; Keramati et al., 2014; Kianmehr & Alhajj, 2009; Martono et al., 2014; Routh et al., 2020; Ullah et al., 2019; Vijaya & Sivasankar, 2019a; Tsai & Chen, 2010).

The lift allows us to calculate the churn rate by group, and the top decile focuses on the measurement of the 10% of cases that are more likely to churn (Zhu et al., 2018; Coussement & Van den Poel, 2008; Xie et al., 2009; Nie et al., 2011; Zhang et al., 2012), while the profit approach in the uplift expands this approach to achieve the maximization amount of profit (Devriendt et al., 2019; Verbeke et al., 2012a; Neslin et al., 2006), thus providing a cost-benefit perspective. The lift metric allows us to consider customers with a level of confidence that dropout will occur, supporting the development of retention strategies for customers that will almost definitively churn. This metric has been explored in different studies (Alboukaey et al., 2020; Ascarza, 2018; Benoit & Van den Poel, 2012; Burez & Van den Poel, 2007; De Bock & Van den Poel, 2010; Gür Ali & Arıtürk, 2014; Hung et al., 2006; Mitrovic et al., 2017; Mitrović et al., 2018; Moeyersoms & Martens, 2015; Wai-Ho Au et al., 2003; Xia & Jin, 2008; Zhang et al., 2012).

The metrics being adopted to predict customer dropout should consider the following scenarios: (1) customers who have a greater risk of dropout should be targeted to provide a base for a better ROI in the retention strategies (Xie et al., 2009; Coussement & Van den Poel, 2008) and (2) retention strategies should be developed focusing on customers with higher satisfaction, or its inclusion could be a reminder of the contractual agreement nearing an end and leading to churn (Devriendt et al., 2019). The use of uplift models could outperform predictive models and contribute to greater profitability in the development of retention campaigns to reduce dropout (Devriendt et al., 2019). This idea underlies on the assumption that it should be not considered only the model accuracy but also the financial performance to maximize the retention strategies, this means that better approaches could

be provided to increase the return on investment in marketing campaigns (Devriendt et al., 2019), where the business objective is to reduce customer churn and customers who are about to churn but cannot be retained should be excluded from the campaign, as targeting them will be a waste of scarce resources. However, Ascarza (2018) states that customers with a higher risk of churning may not be the best targets. Investments in retention strategies are investments that should be developed to distinguish churners susceptible to marketing actions from those that will leave anyway (Coussement et al., 2017).

2.7 Summary

In this chapter, we have developed a review of existing research targeting customer dropout prediction. Customer dropout has two scenarios: (1) customers with contractual settings where there are periodic fees to be paid, where the customer informs when end the relationship and (2) noncontractual settings where the organization has to infer if the customer is still active. Our review address scenarios of contractual settings. Customer dropout is an important facet to be addressed in organizations, there are available several models to analyze existing data available in the organizations to identify trends and patterns related to dropout. Using patterns and trends related to customer churn is possible to develop countermeasures to avoid the dropout event.

Customer dropout prediction underlies on the following assumptions:

- Companies are more profitable targeting customer retention because the costs of retaining customers are lower than acquiring new ones;
- Machine learning has been used to predict customer dropout, using models that mainly are static (applied in a given time) and don't consider that dropout changes over time;
- The adoption of cluster analysis allows to develop customer segmentation, and develop customer dropout prediction in each segment;
- There are several metrics to measure the performance of the algorithms such as the area under the receiver operating curve (AUC), sensitivity, specificity, recall, precision, and F-score

- Lift metric allows measuring the churn rate by group, where the top decile focuses on the measurement of the 10% cases that are more likely to churn

Therefore, the development of a prediction model to identify customers that could churn, is desirable, mainly if we consider that there is a lack of research addressing the timing of the dropout. Secondly, cluster segmentation has been used but never combined with survival analysis. It would be desirable to explore those ideas to improve the models prediction performance. Finally, the metrics adopted to predict customer dropout should consider the customers who have a greater risk of dropout and should be targeted to provide greater profitability in the development of retention campaigns to reduce dropout.

Chapter 3

Predicting Customer Dropout

The variety and exploration of different approaches to address dropout accuracy have led to the integration of several algorithms to increase performance. Vijaya & Sivasankar (2019a) suggested that works that adopt hybrid models combining more than one classifier can achieve increased performance compared with those using single classifiers. Some studies have explored the combination of clusters with churn prediction (Hung et al., 2006; Gök et al., 2015; Sivasankar & Vijaya, 2019), where if the customers are grouped into clusters, the prediction accuracy can be improved within each cluster. The hybrid approach uses clustering and classification, which segments the customers before developing a classification could be effective (Jafari-Marandi et al., 2020). An approach that is different from ensemble methods uses several models that are built independently before developing a final step that combines different models to create a final class prediction; the approach applies a pipeline concept, where the output of one algorithm server as the input for another algorithm (M. A. H. Farquad et al., 2014).

In this section, to overcome the existing limitations we propose a hybrid random survival forests combined with segmentation of customers using cluster analysis which, as far as we known, never has been used in understanding factors affecting membership in a sports club and fitness center using existing data. The analysis is based on the use of random survival forests in the presence of covariates that do not necessarily satisfy the PH assumption. Additionally, we also propose a new approach combining variety and exploration of different approaches to address dropout accuracy has led to the integration of several algorithms to increase performance. Vijaya & Sivasankar (2019a) suggested

that works that adopt hybrid models combining more than one classifier can achieve increased performance compared with those using single classifiers. This idea is not new, and some studies have explored the combination of clusters with churn prediction (Hung et al., 2006; Gök et al., 2015; Sivasankar & Vijaya, 2019), where if the customers are grouped into clusters, the prediction accuracy can be improved within each cluster. The hybrid approach uses clustering and classification, which segments the customers before developing a classification could be effective (Jafari-Marandi et al., 2020). An approach that is different from ensemble methods uses several models that are built independently before developing a final step that combines different models to create a final class prediction; the approach applies a pipeline concept, where the output of one algorithm server as the input for another algorithm (M. A. H. Farquad et al., 2014).

3.1 Survival analysis

The main objective of survival analysis is to determine the probability of an outcome event happening or not, analyzing the time until an event of interest occur and exploring its relationships with different factors. The main difficulty is related to, that at a determined point in time, only some individuals have experienced the event and the others have not. Survival analysis considers the individuals that don't have experienced the dropout, which is its main advantage, using the concept of censoring. Censoring indicates observations that are not completely related to the event of interest, e.g. Customers that didn't dropout yet, which are incorporated in the analysis. This means that there are customers still active for which we don't know if the event of dropout has occurred, which is called censorship. Survival models take censoring into account and incorporate this uncertainty, instead of predicting the time of the event such in regression models, the survival models allow predicting the probability of an event happening at a particular time.

The time of dropout is represented by T, which is a non-negative random variable, this indicates the time period of the event occurring for a randomly selected individuals from the population. T represents the probability of an event to occur each time period given that has not already occurred in a previous time period, known as discrete-time hazard function (Singer & Willett, 1993). The survival function represents the probability of an individual surviving after time t, $S(t) = P(T > t)$, $t \geq 0$, with the properties $S(0) =$

1,S(∞) = 0. The distribution function is represented with F, defined as F(t) = P(T $\leq$ 0), for t $\geq$ 0. The function of probability density represented with f where:

$$f(t) = \lim_{dt \to 0} \frac{P[t \leq T < t + dt]}{dt} \tag{3.1}$$

$f(t)dt$ represents the probability of an event occurring in the moment t. The need to represent the distribution evolution of the dropout probability along the time uses to the hazard function, represented as:

$$\lambda(t) = \lim_{dt \to 0} \frac{P[t \leq T < t + dt | T \geq t]}{dt} \tag{3.2}$$

The determination of the survival curves is based on the following elements: (1) the total value of observations removed during the time period (e.g. days, months, or years), either by dropout or by censorship; (2) observations that composed the sample of the study; (3) customers who had not yet dropped out at any given time. The survival probability until the time period ii (p_i) is calculated with:

$$p_i = \frac{r_i - d_i}{r_i} \tag{3.3}$$

Where r_i is the number of individuals that survived at the beginning of the period, d_i is the number of individuals who left during the period. The survival time estimate was also taken considering the month in which it is found (estimated). Cox's allow test difference between survival times. The advantage of using survival analysis was that allows us to detect if the risk of an event differs systematically across different people, using specific predictors. The coefficients in a Cox regression were related to the hazard, where a positive value represents a worse prognosis and the opposite, negative value a better prognosis. The advantage of survival analysis was that allowed us to include information on covariates that were censored up to the censoring event.

The Cox PH model assumes the covariates to be time independent, for example, gender and age when where retrieved do not change over time (Schober & Vetter, 2018a). Because the Cox model requires the hazards in both groups to be proportional, researchers are often asked to "test" whether hazards are proportional (Stensrud & Hernán, 2020). Considering

this we explored other approaches that allow us to develop this analysis without the proportional hazard assumptions, the survival trees.

3.2 Why clusters?

Clusters is a unsupervised learning, which is a set of statistical tools intended for the setting in which we have only a set of features X_1, X_2, . . . , X_p measured on n observations, where there is no interest in prediction, due to the lack of an associated response variable Y (James et al., 2013). The main idea is to answer how can we discover subgroups in the data? Clustering is a process of partitioning data into a set of meaningful sub-classes (Kianmehr & Alhajj, 2011b).

There are several approaches that can be developed such as hierarchical clustering and K-means clustering (James et al., 2013). In hierarchical clustering is used a dendrogram (visual representation tree-like visualization) and in K-means clustering we seek to partition the observations into a predefined number of clusters (James et al., 2013)

K-means clustering is a simple and elegant approach for partitioning a data set into K distinct, non-overlapping clusters. To perform K-means clustering, we must first specify the desired number of clusters K; then the K-means algorithm will assign each observation to exactly one of the K clusters (James et al., 2013).

The main advantage of adopting hierarchical clustering is that doesn't require that the number of clusters are specified in advance using a dendrogram (James et al., 2013).

Figure 3.1 shows an example of a dendrogram representing a hierarchical clustering, the y axis is the Euclidean distance. The figure in the center shows a cut at the height of 9 resulting in two clusters, and the right figure cut at 5 resulting in three clusters.

Compare the advantages and disadvantages of the two methods. . . The main advantages of K-means are computational fast and simple and understandable, however is difficult to identify the number of clusters initially (Namratha & Prajwala, 2012). Regarding hierarchical clustering its main advantages are that don't require the number of clusters in advance, however, has high complexity and are computational slow (Namratha & Prajwala, 2012).

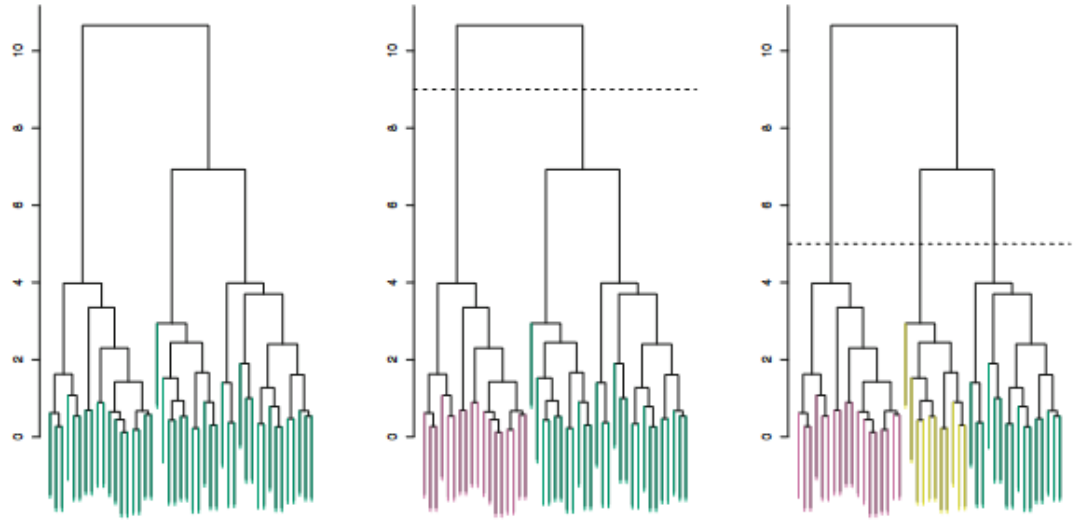


FIGURE 3.1: Dendrogram example (James et al., 2013)

The problem regarding the determination of the number of clusters in the K-Means could be solved using the Bayesian Information Criterion(BIC), where the model with the lowest score can be selected as the best model (Schwarz, 1978), however Scrucca et al. (2016) suggests the use of the higher BIC score.

3.3 Why survival trees?

Survival trees partition the covariate space into smaller regions (nodes) which contains observations with homogeneous survival outcomes (Bertsimas et al., 2022). Each node is created in a greedy manner, making a local optimal choice, which means that each node split is selected in isolation without considering its effect in the bellow splits of the tree (Breiman et al., 1984).

The node splits are created to get an understanding of what variables uncover a predictive structure of the problem. The main advantage of Random Survival Forests is that does not make the proportional hazards assumption (Ehrlinger, 2016a) and has the flexibility to model survivor curves that are of dissimilar shapes for contrasting groups of subjects. Random Survival Forest is an extension of Random Forest allowing efficient non-parametric analysis of time to event data (Breiman, 2001). These characteristics allow us to surpass the Cox Regression limitation of the proportional hazard assumption, requiring to exclude variables which not fulfill the model assumptions. It was also shown by Breiman

(2001) that ensemble learning can be further improved by injecting randomization into the base learning process - a method called Random Forests. Another advantage of survival trees is related that are tailored to handle with censored data, considering that in tree models each node has a splitting criterion and the data considering the event of interest are grouped together (Wang et al., 2017). This presents some advantages, mainly when we are considering the duration of the event until the event of dropout.

Survival trees are methods based on Random Forest models (Breiman, 2001). Random survival forests is an ensemble method for the analysis of right-censored data (Ishwaran et al., 2008), using randomization to improve the performance. Ishwaran et al. (2008) suggested this framework for survival forests:

- Draw B random samples of the same size from the original dataset with replacement. The samples that are not drawn are said to be out-of-bag (OOB). Grow a survival tree on each of the b = 1 , . . . , B samples.
- At each node, select a random subset of predictor variables and find the best predictor and splitting value that provide two subsets (the daughter nodes) which maximizes the difference in the objective function.
- Repeat step 2 recursively on each daughter node until a stopping criterion is met.
- Calculate a cumulative hazard function (CHF) for each tree and average over all CHFs for the B trees to obtain the ensemble CHF.
- Compute the prediction error for the ensemble CHF using only the OOB data.

In each node is selected a predictor x from a random selected predicted variables and split value c (one unique value of x). Each sample i if assigned to the daughter right node if $x_i \leq c$ or left if $x_i \geq c$, then is calculated the logrank such as:

$$L(x,c) = \frac{\sum_{i=1}^{N}\left(d_{i,1} - Y_{i,1}\frac{d_i}{Y_i}\right)}{\sqrt{\sum_{i=1}^{N}\frac{Y_{i,1}}{Y_i}\left(1-\frac{Y_{i,1}}{Y_i}\right)\left(\frac{Y_i-d_i}{Y_i-1}\right)d_i}} \tag{3.4}$$

Where:

- j: Daughter node, $j \in \{1, 2\}$

- $d_{i,j}$: Number of events at time t_i in daughter node j
- $Y_{i,j}$: Number of elements that had the event or are in risk at time t_i in daughter node j
- d_i: Number of events at time t_i, such $d_i = \sum_j d_{i,j}$
- Y_i: Number of elements that experienced an event or are at risk at t_i so $Y_i = \sum_j Y_{i,j}$

We loop every x and c until find x^* that satisfy $|L(x^*, c^*)| \geq |L(x, c)|$ for every x and c.

3.4 Model proposal

The survival analysis was conducted using the package Lifelines (Davidson-Pilon, 2021), which provides survival curves and Kaplan-Meier estimators, simplifying a first interpretation of the data.

Considering the limitations of the Cox Regression, we adopted the random survival forest developed using the package PySurvival (Fotso & et al., 2019). PySurvival is an open source python package for Survival Analysis modeling - the modeling concept used to analyze or predict when an event is likely to happen.

Using the package mclust (Scrucca et al., 2016) the number of clusters are calculated choosing the number of components and identifying the structure of the covariance matrix, based on the modeling with a multivariate normal distribution for each component that constitute the data set (Akogul & Erisoglu, 2016). The calculation of the number of clusters is based on the Bayesian Information Criterion (BIC). The model that gives the minimum BIC score can be selected as the best model (Schwarz, 1978) simplifying the problem related to choosing the number of components and identifying the structure of the covariance matrix, based on modeling with multivariate normal distributions for each component that forms the data set (Akogul & Erisoglu, 2016).

The approach uses a Gaussian mixture model (GMM), which assumes a (multivariate) Gaussian distribution for each component. Clusters are based on parameters such as

distribution, volume, shape, and orientation. The table 3.1 resumes the geometric characteristics. There are 14 possible models with different geometric characteristics that can be specified (Scrucca et al., 2016).

TABLE 3.1: Parameters of models to determine BIC

Model	Distribution	Volume	Shape	Orientation
EII	Spherical	Equal	Equal	—
VII	Spherical	Variable	Equal	—
EEI	Diagonal	Equal	Equal	Coordinate axes
VEI	Diagonal	Variable	Equal	Coordinate axes
EVI	Diagonal	Equal	Variable	Coordinate axes
VVI	Diagonal	Variable	Variable	Coordinate axes
EEE	Ellipsoidal	Equal	Equal	Equal
EVE	Ellipsoidal	Equal	Variable	Equal
VEE	Ellipsoidal	Variable	Equal	Equal
VVE	Ellipsoidal	Variable	Variable	Equal
EEV	Ellipsoidal	Equal	Equal	Variable
VEV	Ellipsoidal	Variable	Equal	Variable
EVV	Ellipsoidal	Equal	Variable	Variable
VVV	Ellipsoidal	Variable	Variable	Variable

Following those ideas, the hybrid approach is proposed with the following steps:

1. Identify the optimal number of clusters using Scrucca et al. (2016) mclust package.
2. Fit the model using the identified number of clusters.
3. Estimate for each element of the cluster.
4. For each cluster follow the framework proposed by Ishwaran et al. (2008) to calculate the random survival model.

3.5 Model evaluation

Before employing a model to be tested in an empirical context, we need to clarify how the performance could be evaluated. The goal is to evaluate if a proposed approach has better results than an existing one.

The model evaluation assesses if the prediction performance improves against existing data, this is operationalized using train data to create the model, and test data to evaluate the performance. Dropout is a binary value where one value represents churn and the

other not churn. In the context of survival analysis, there is a new element in the analysis, which is censoring. Considering the existence of censoring, common metrics employed for example in regression such as root mean squared error and r^2 are not suitable for measuring the performance in survival analysis (Wang et al., 2017). Therefore the model performance could be calculated with the concordance probability (C-index), Brier Score (BS) and Mean Absolute Error (MAE) (Wang et al., 2017).

The BS is used to evaluate the predicted accuracy of the survival function at a given time t. Representing the average square distance between the survival status and the predicted survival probability, where the value 0 is the best possible outcome.

$$BS(t) = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{\left(0-\hat{S}(t,\vec{x}_i)\right)^2 \cdot \mathbb{1}_{T_i\leq t,\delta_i=1}}{\hat{G}(T_i^-)} + \frac{\left(1-\hat{S}(t,\vec{x}_i)\right)^2 \cdot \mathbb{1}_{T_i>t}}{\hat{G}(t)}\right) \tag{3.5}$$

The model should have a Brier score below 0.25. Considering that if $\forall i \in [\![1, N]\!], \hat{S}(t,\vec{x}_i) = 0.5$, then $BS(t) = 0.25$.

The feature importance was determined by calculating the difference between the true class label and noised data (Breiman, 2001).

Additionally, validation tests have been performed to compare the accuracy of the hybrid approach against the random survival model without clusters. To that end, a paired Mann-Whitney test has been used to estimate whether the prediction ability is significant using a confidence interval of 95

To achieve the research goals, first to simplify the analysis, the survival probabilities are presented as a survival curve to provide an overall perspective of the dropout along time, the representation of the survival probabilities shows the time when the events are observed (Bland & Altman, 1998). Then the machine learning survival model was created following Ishwaran et al. (2008) approach using PySurvival (Fotso & et al., 2019). The model performance was determined with the Brier Score (BS) and Mean Absolute Error (MAE), considering that due to the censoring of data, standard evaluation metrics such as root mean square error are not suitable (Wang et al., 2017). The feature importance was determined by calculating the differences between the true class label and noised data (Breiman, 2001).

The hybrid model is developed with the identification of an optimal number of clusters. The calculation of the optimal number of clusters is developed based on the Bayesian Information Criterion (BIC), where the model with the lowest score can be selected as the best model (Schwartz et al., 2015), however, Scrucca et al. (2016) suggests the higher BIC score, which we followed. In addition, a visualization to increase the interpretability of the number of clusters is also provided using the elbow method.

The BS is used to evaluate the predicted accuracy of the survival function at a given time t. Representing the average square distance between the survival status and the predicted survival probability, where the value 0 is the best possible outcome, and 1 an inaccuracy model.

The BS could be calculated considering the right censoring as:

$$BS(t) = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{\left(0-\hat{S}(t,\vec{x}_i)\right)^2 \cdot \mathbb{1}_{T_i\leq t,\delta_i=1}}{\hat{G}(T_i^-)} + \frac{\left(1-\hat{S}(t,\vec{x}_i)\right)^2 \cdot \mathbb{1}_{T_i>t}}{\hat{G}(t)}\right) \tag{3.6}$$

The model should have a Brier score below 0.25, such as $\forall i \in [\![1,N]\!], \hat{S}(t,\vec{x}_i) = 0.5$, then $BS(t) = 0.25$.

3.6 Summary

In this chapter, we have proposed an hybrid approach to overcome existing limitations combining cluster analysis with survival trees. The use of the survival trees to suport the development of the survival analysis tries to overcome the existing limitations of using survival analysis, such as the proportional hazard assumptions.

The proposed approach suggests the used of the Kaplan-Meier estimators, as a first exploration of the data. To solve the problem related to the determination of the number of clusters, is proposed the calculation of the optimal number of clusters based in the Bayesian Information Criterion. Using the determined number of clusters is possible to apply the survival trees to within each cluster. The evaluation of the performance of the model is developed using the concordance probability, Brier Score and Mean Absolute Error.

Chapter 4

Case studies validating the customer dropout prediction

Following this thesis objective, how can the historical data be used to predict the customer dropout and support the decision to develop countermeasures to avoid customer desertion in contractual settings? This requires some underlying assumptions. This chapter explores those ideas previously addressed (chapter 3), exploring two cases studies. The first case study address data from a health club, where is used a random survival model combined with clusters (Analysis developed available in appendix A). The second case study, follows the same idea proposed for the data of the health club, where we test if the hybrid model improves the performance comparing to the results of a model without clusters using membership data (Analysis developed available in appendix B).

4.1 Health club hybrid survival model

In this case study, data from 5,209 health club customers was analysed (mean age = 27.88, SD = 11.80 years) from a Portuguese fitness centre. The data corresponds to the time period between 2014 and 2017. The data was collected from software esport (Cedis, Portugal) between 2014 and 2017. The information retrieved was: Age of the participants in years; Sex (0-female, 1-male); Non-attendance days before dropout; Total amount billed; Average number of visits per week; Total number of visits; Weekly contracted accesses;

TABLE 4.1: Summary statistics of features used

Characteristic	**N = 5,209**
age (age in years), Mean (SD)	28 (12)
Male or female, %	35%
dayswfreq (non-attendance days before dropout), Mean (SD)	76 (102)
tbilled (total amount billed), Mean (SD)	155 (155)
maccess (average entries by week), Mean (SD)	0.89 (0.76)
freeuse (user with freeuse (1) or with limited entries (0), %	4.9%
nentries (number total of entries), Mean (SD)	29 (41)
cfreq (weekly contracted accesses), %	
2	1.3%
4	2.4%
6	0.2%
7	96%
months (customer enrolment in months), Mean (SD)	9 (8)
dropout (customer dropout 1 non-dropout 0), %	88%

Number of registration renewals; Number of customer referrals; Registration month; Customer enrolment duration; and status (dropout/non-dropout).

Dropout event occur when customer communicate the intention to terminate the contract or did not pay the monthly fee during 60 days. Dropout is a binary value where one represent churn and zero not churn. The dropout happens when a member does not have a payment. The survival time in the dataset is represented by the number of months the customer begin affiliated.

Table 4.1 shows data's summary statistics. The average age is 27.9 ± 11.8, the entries are 29 ± 41.1 with an inscription period of 9 ± 8.2 months. Figure 4.1 shows the distribution of the dropout considering the number of years of membership, where 0 represents a non-dropout and 1 dropout.

The correlation between the variables was checked using listing 4.1 where the variable *tbilled* as an higher correlation with *nentries* (r=0.77) and was removed. After the variable removal the correlations seem aceptable figure 4.3.

4.1.1 Survival analysis

To simplify the analysis, the survival probabilities are presented as a survival curve. The survival curve is a representation of the survival probabilities corresponding to a time where

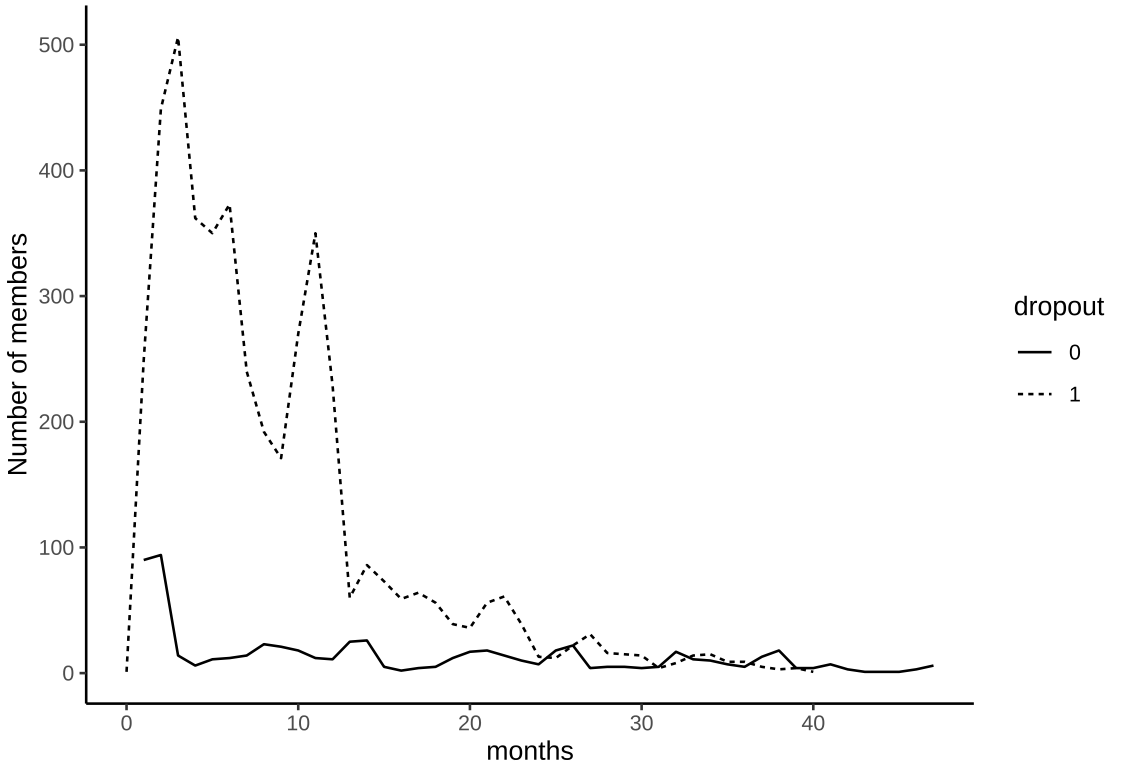


FIGURE 4.1: Number of members by month

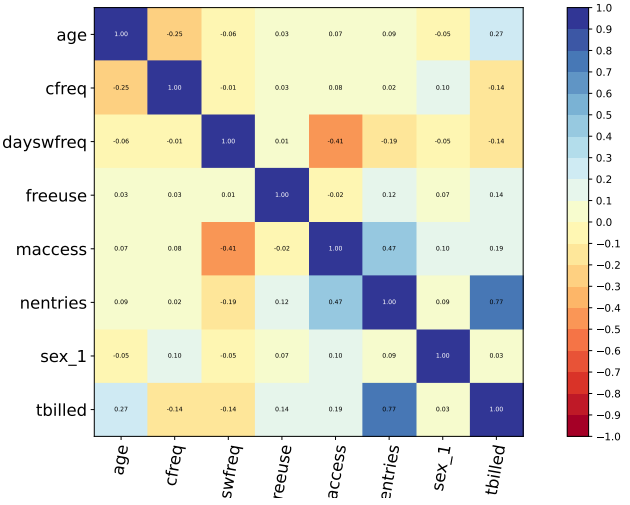


FIGURE 4.2: Correlation matrix variables used

```
from pysurvival.utils.display import correlation_matrix
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

col = ['sex']

df_members = r.df_members #copy r dataframe to python

# convert to int
df_members['age']=df_members['age'].astype(int)
df_members['dayswfreq']=df_members['dayswfreq'].astype(int)
df_members['cfreq']=df_members['cfreq'].astype(int)
df_members['months']=df_members['months'].astype(int)
df_members['dropout']=df_members['dropout'].astype(int)
df_members['sex']=df_members['sex'].astype(int)

df_members = pd.get_dummies(df_members, columns=col,drop_first=True)

# Creating the time and event columns
time_column = 'months'
event_column = 'dropout'

# Extracting the features
features = np.setdiff1d(df_members.columns, [time_column, event_column] ).
    tolist()


correlation_matrix(df_members[features], figure_size=(10,10), text_fontsize
    =6)

r.df_members = df_members #return values to R
```

Listing 4.1: Cox proportional hazard

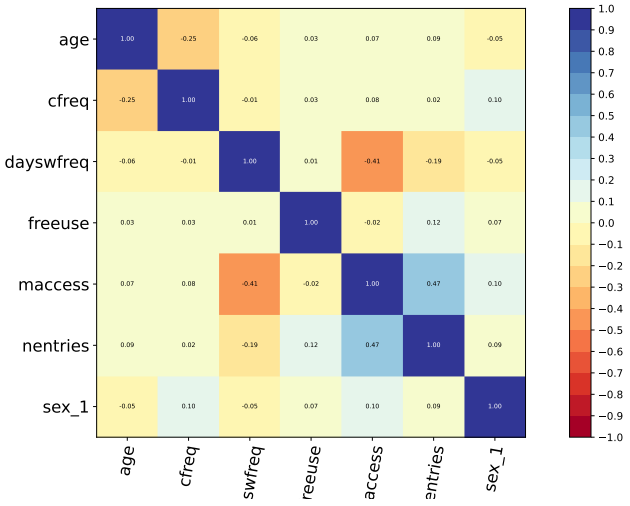


Figure 4.3: Correlation matrix after variable removal

the events are observed (Bland & Altman, 1998). The survival curves where developed using the package lifelines (Davidson-Pilon, 2021).

The time of dropout is represented by T, which is a non-negative random variable, indicating the time period of the event occurring for a randomly selected individual from the population, representing the probability of an event to occur each time period given that has not already occurred in a previous time period, known as discrete-time hazard function (Singer & Willett, 1993). The survival function represents the probability of an individual surviving after time t, $S(t) = P(T > t)$, $t \geq 0$, with the properties $S(0) = 1, S(\infty) = 0$. The distribution function is represented with F, defined as $F(t) = P(T \leq 0)$, for $t \geq 0$. The function of probability density represented with f where:

$$f\left(t\right) = \lim_{dt \to 0^{+}} \frac{P\left[t \leq T < t + dt\right]}{dt} \tag{4.1}$$

$f(t)dt$ represents the probability of an event occurring in the moment t. The need to represent the distribution evolution of the death probability along the time, uses to the hazard function, represented as:

$$\lambda\left(t\right) = \lim_{dt \to 0^{+}} \frac{P\left[t \leq T < t + dt \middle| \, T \geq t\right]}{dt} \tag{4.2}$$

The determination of the survival curves is based in the following elements: (1) the total value of observations removed during the time period (month), either by dropout or by censorship; (2) observations that composed the sample of the study (N= 5219); (3) customers who had not yet dropped out at any given time. The survival probability until the time period ii (pi) is calculated with:

$$p_i = \frac{r_i - d_i}{r_i} \tag{4.3}$$

Where r_i is the number of individuals that survived at the beginning of the period, d_i the number of individuals who left during the period. The survival time estimate was also taken considering the month in which it is found (estimated). Cox's allow test difference between survival times. The advantage in using survival analysis was that allow us to detect if the risk of an event differs systematically across different people, using specific

```
from lifelines import KaplanMeierFitter
kmf = KaplanMeierFitter()
T = df_members['months']
C = df_members['dropout']

kmf.fit(T,C,label="Customers")

kmf.event_table.reset_index()
kmf.conditional_time_to_event_

survival_table = pd.concat([kmf.event_table.reset_index(),
                            kmf.conditional_time_to_event_.reset_index
(),
                            kmf.survival_function_.reset_index()],axis
=1)

survival_table.drop(['timeline'],axis=1,inplace=True)
survival_table.columns = ['event_at', 'removed', 'observed', 'censored'
,
                          'entrance', 'at_risk', 'estimated_survival',
                          'prob']
```

Listing 4.2: Cox proportional hazard

predictors. The coefficients in a Cox regression were related to the hazard, where a positive value represents a worse prognosis and the opposite, negative value a better prognosis. The advantage of survival analysis was that allow us to include information of covariates that were censored up to the censoring event.

The survival curves where determined using the Cox proportional hazard model available in the package lifelines (listing 4.2).

The table 4.2 depicts the data of the survival time of the customers during the first months, the results showed that the customers have a survival probability of 24.44% at 12 months (column p_i - likelihood probability) with a median survival time of 10 months (column estimated_survival). The survival probability at 6 months was 54.5%, representing an risk of dropout of 45.5% with a estimated survival of 6 months.

Figure 4.4 shows the Kaplan Meier survival curve for the customers considering the number of months of membership (x axis) and survival probability (y axis). The customer dropout is very high in the first 12 months, ranging from a survival probability of 54% after the first 6 months until 24% after 12 months.

Figure 4.5 shows the survival by gender. The survival curves by gender are very similar, both types of customers present a behavior that is not very different.

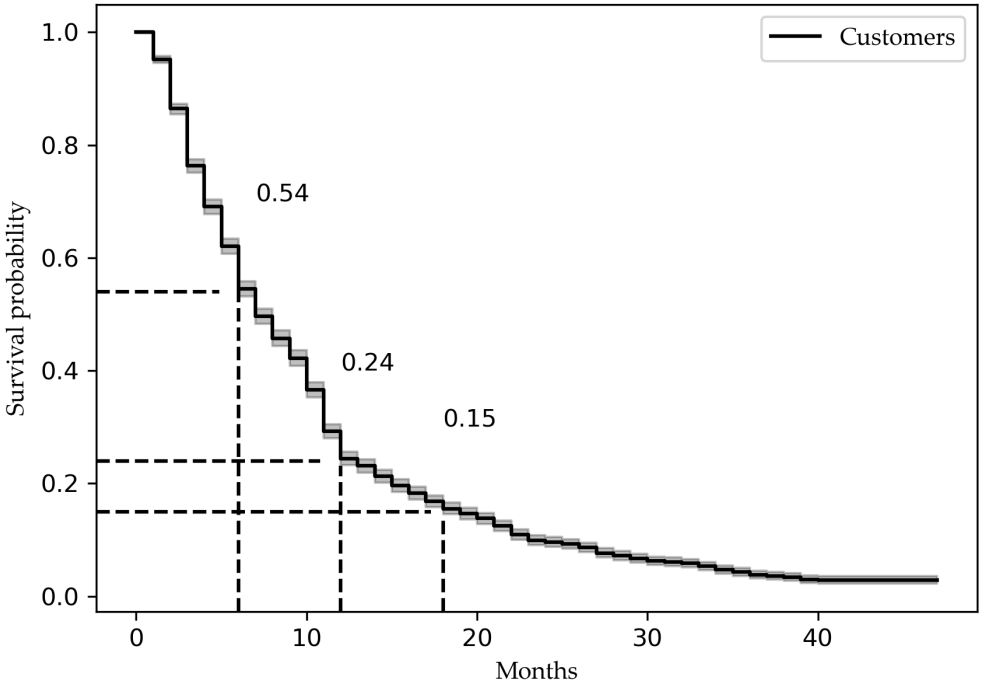


FIGURE 4.4: Survival probability

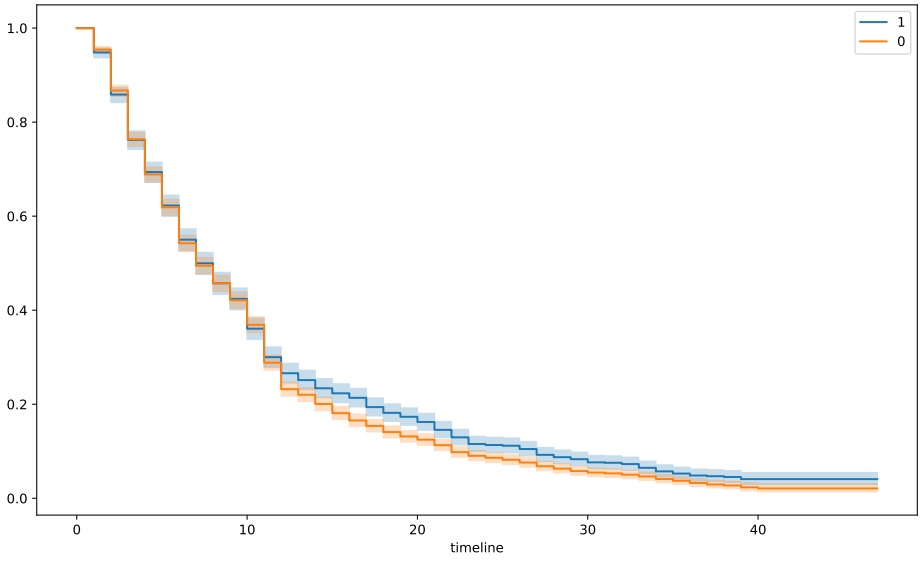


FIGURE 4.5: Survival probability by gender

TABLE 4.2: Determination of the survival time probabilities

event_at	removed	observed	censored	entrance	at_risk	estimated_survival	prob
0	1	1	0	5210	5210	7	1.000
1	339	249	90	0	5209	7	0.952
2	543	449	94	0	4870	7	0.864
3	520	506	14	0	4327	7	0.763
4	368	362	6	0	3807	7	0.691
5	361	350	11	0	3439	6	0.620
6	385	373	12	0	3078	6	0.545
7	254	240	14	0	2693	5	0.497
8	215	192	23	0	2439	6	0.457
9	192	171	21	0	2224	6	0.422
10	288	270	18	0	2032	6	0.366
11	362	350	12	0	1744	9	0.293
12	240	229	11	0	1382	10	0.244
13	85	60	25	0	1142	9	0.231
14	112	86	26	0	1057	9	0.213
15	78	73	5	0	945	9	0.196
16	61	59	2	0	867	10	0.183
17	68	64	4	0	806	10	0.168
18	61	56	5	0	738	9	0.155
19	51	39	12	0	677	9	0.147
20	53	36	17	0	626	9	0.138
21	74	56	18	0	573	10	0.125
22	75	61	14	0	499	11	0.109
23	49	39	10	0	424	11	0.099
24	20	13	7	0	375	10	0.096

Note:
Removed – the sum of customers with dropout and that are censored; Censored – the event did not occur during the period of this data, collection; Risk of Dropout – number of customers at risk of, dropout; pi – survival probability; Estimated Survival - months to survive in the sports facility.

Figure 4.6 shows the survival by contracted frequency. Customers with contracted frequency of 6 and 4 times a week have higher survival probabilities, against lower of customers with contracted frequencies of 7 and 2 times a week. Survival curves allow to explore tendencies related to survival to extract actionable knowledge.

The model assumptions was determined using listing 4.3. The proportional hazard assumptions failed in the following variables: age $p < 0.01$, cfreq $p < 0.01$, dayswfreq $p < 0.01$, tbilled $p < 0.01$, freeuse $p < 0.01$, nentries $p < 0.01$.

The Cox PH model assumes the covariates to be time independent, for example gender

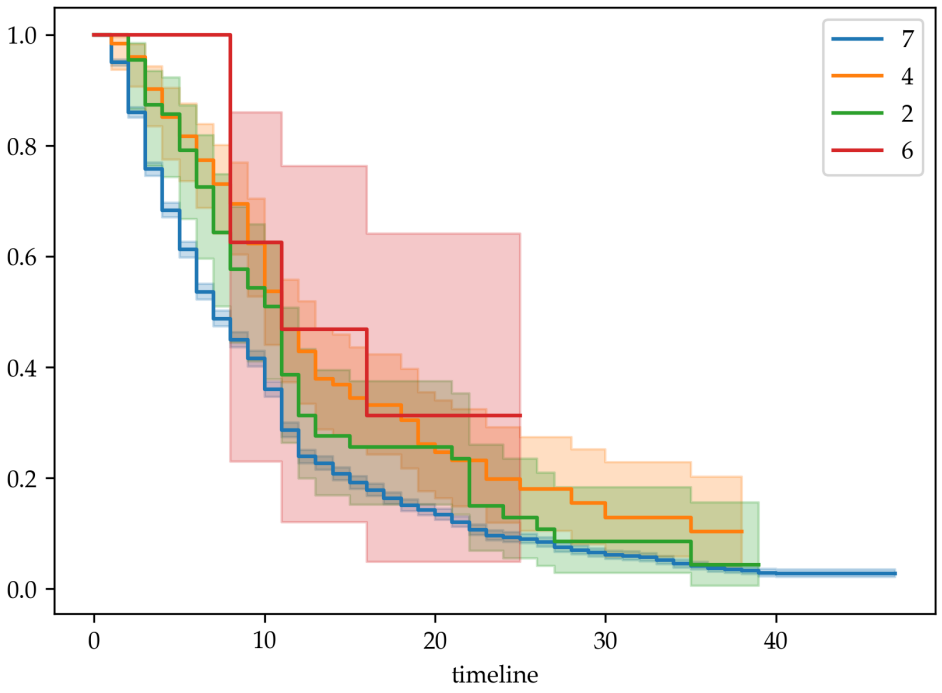


FIGURE 4.6: Survival probability by Contracted Frequency

```
vars= ['age', 'dayswfreq', 'tbilled', 'maccess', 'freeuse', 'nentries',
       'cfreq', 'months', 'sex_1', 'dropout']
df_regression = df_members[vars].copy()

from lifelines import CoxPHFitter
cph = CoxPHFitter()
cph.fit(df_regression,duration_col='months',event_col='dropout')

cph.check_assumptions(df_regression)
```

LISTING 4.3: Checking model assumptions

when where retrieved do not change over time (Schober & Vetter, 2018b). Because the Cox model requires the hazards in both groups to be proportional, researchers are often asked to "test" whether hazards are proportional (Stensrud & Hernán, 2020). Considering this we explored other approach that allow us to develop this analysis without the proportional hazard assumptions, the survival trees.

Random Survival Forests does not make the proportional hazards assumption (Ehrlinger, 2016b) and has the flexibility to model survivor curves that are of dissimilar shapes for contrasting groups of subjects. Random Survival Forest is an extension of Random Forest allowing efficient non-parametric analysis of time to event data (Breiman, 2001). This

```
from pysurvival.models.survival_forest import RandomSurvivalForestModel
from sklearn.model_selection import train_test_split
from pysurvival.utils.metrics import concordance_index
from pysurvival.utils.display import integrated_brier_score
from pysurvival.utils.display import compare_to_actual

X = df_members.copy()
t = df_members['months']
e = df_members['dropout']
X.drop(axis=1, columns=['months','dropout'],inplace=True)

X_train, X_test, t_train, t_test, e_train, e_test = train_test_split(X, t,
    e, test_size=0.3, random_state=0)

# Fitting the model
csf = RandomSurvivalForestModel(num_trees=20)
csf.fit(X_train, t_train, e_train, max_features='sqrt', max_depth=5,
        min_node_size=20, seed = 1)

c_index = concordance_index(csf, X_test, t_test, e_test)
ibs = integrated_brier_score(csf, X_test, t_test, e_test, t_max=12,
                             figure_size=(12,5))
```

LISTING 4.4: Creating the survival model using the package PySurvival

characteristics allow us to surpass the Cox Regression limitation of the proportional hazard assumption, requiring to exclude variables which not fulfill the model assumption. It was shown by (Breiman, 2001) that ensemble learning can be further improved by injecting randomization into the base learning process - a method called Random Forests.

4.1.2 Random Survival Forest

The random survival forest was developed using the package PySurvival (Fotso & et al., 2019). The most relevant variables predicting the dropout are analysed using the log-rank test. The metric variables are transformed to categorical using the quartiles to provide a statistical comparison of groups. The survival analysis was conducted using the package Lifelines (Davidson-Pilon, 2021). The model was built with with 70% of the data for training and 30% for testing. The survival model parameters where:

PySurvival is an open source python package for Survival Analysis modeling - the modeling concept used to analyze or predict when an event is likely to happen. It is built upon the most commonly used machine learning packages such NumPy, SciPy and PyTorch. The model is created (Listing 4.4) using the duration of the survival "months" and the dropout event "dropout" variables fitting the train set and the accuracy of the model is created using the test set.

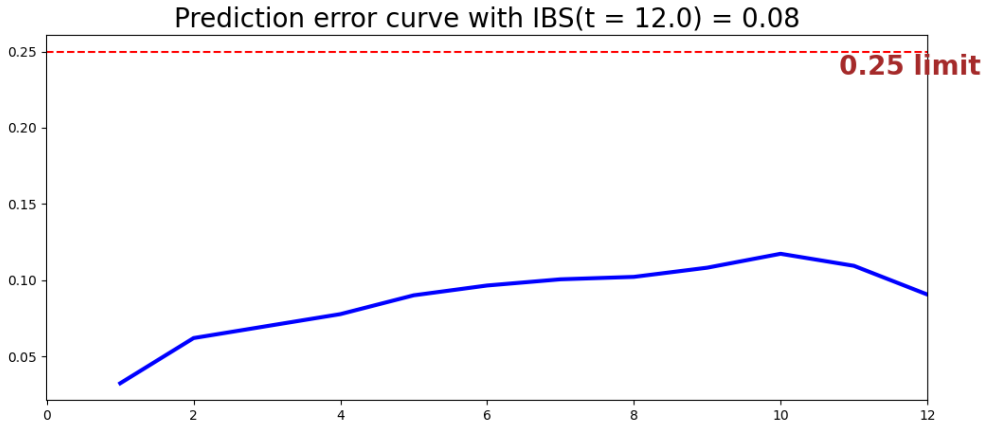


FIGURE 4.7: Model global performance

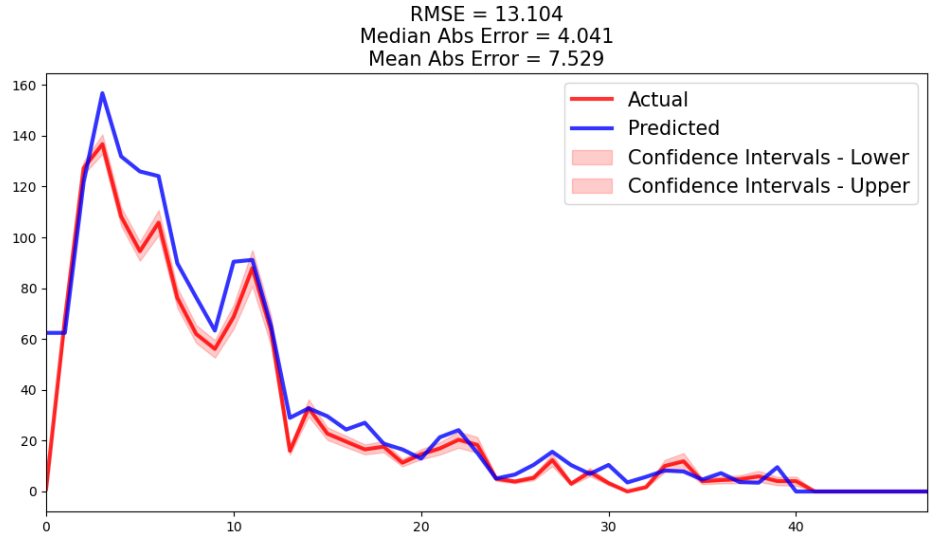


FIGURE 4.8: Global performance versus predicted

The model gives the accuracy with IBS and concordance with the metrics RMSE, Median Absolute Error and Mean Absolute Error. The prediction is very similar to the actual value 4.8. The model accuracy is very high with a root mean square error of 13. The mean absolute error mean was 7.53 customers, and the median absolute error was 4.04.

Table 4.3 shows variables importance calculated according (Breiman, 2001) where the percent increase in misclassification rate as compared to the out-of-bag rate (with all variables intact), out-of-bag is a bootstrap aggregating (subsampling with replacement to create training samples for the model to learn from) where two independent sets are created. One set, the bootstrap sample, data chosen to be in-the-bag by sampling with replacement

```
tbl <- py$csf$variable_importance_table
kbl(tbl, booktabs = T, caption = "Features importance in the survival model
    ")
```

LISTING 4.5: Getting feature importance in R from a python object using Reticulate

TABLE 4.3: Features importance in the survival model

feature	importance	pct_importance
dayswfreq	8.8737185	0.3091196
tbilled	5.4427383	0.1896000
nentries	5.2469602	0.1827800
maccess	3.2807358	0.1142858
freeuse	3.1634538	0.1102002
cfreq	1.4725277	0.0512961
age	0.6708896	0.0233707
sex_1	0.5553969	0.0193475

and the out-of-bag is all data not chosen in the sampling process. The importance of the variables use the fitted in the train dataset (Listing 4.5)

The most important variable is the $dayswfreq$, followed by $tbilled$ and $nentries$. Less important variables are $creq$, age and sex.

4.1.3 Survival trees based model with clusters

In this approach we have created clusters and applied the survival trees within each cluster. The calculation of the number of clusters used the package mclust (Scrucca et al., 2016) using the Bayesian Information Criterion (BIC). The model that gives the minimum BIC score can be selected as the best model (Schwarz, 1978) simplifying the problem related to choosing the number of components and identifying the structure of the covariance matrix, based on modelling with multivariate normal distributions for each component that forms the data set (Akogul & Erisoglu, 2016).

The model which gives the lowest BIC score, is the EEV model (figure 4.9). The top 3 models based on the BIC criterion where the EEV model: 9 clusters 6990.944; 7 clusters -30105.79; and 6 clusters -44616.30. Figure 4.10 shows also the elbow analysis. An optimal number of clusters was considered of five. Considering that was the value after the average distortion was flattened.

```
library(mclust)
y <- scale(py$dfmembers)

set.seed(0) #make it reproducible

bic <- mclustBIC(y)
# Best model using the BIC criteria
plot(bic, what="BIC")
summary(bic, what="BIC")
```

Listing 4.6: Calculation of the optimal number of clusters using BIC

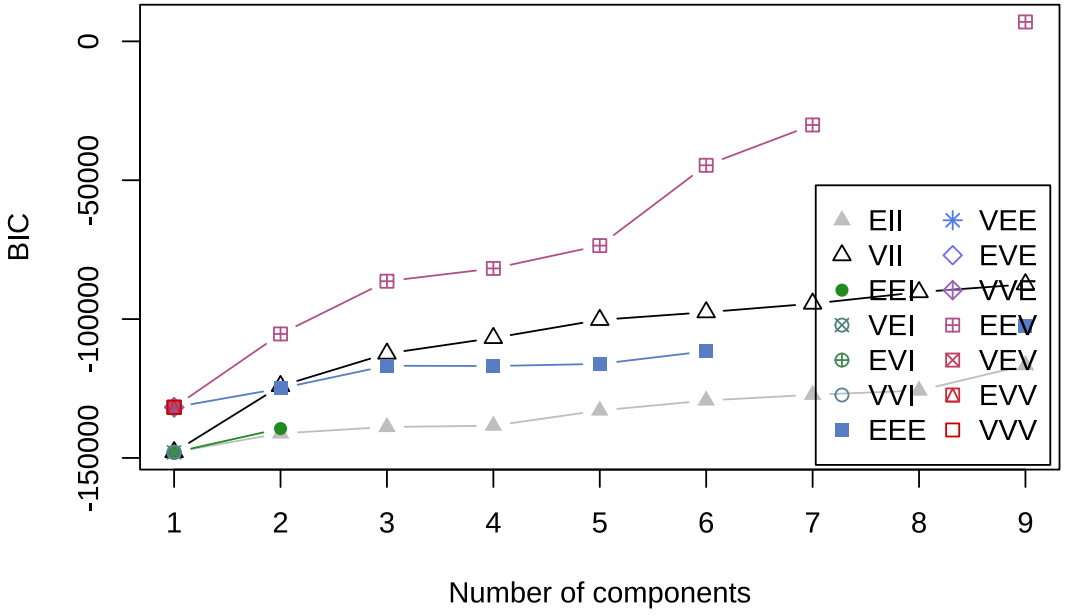


Figure 4.9: Best number of clusters according to BIC

In this approach we have created clusters and applied the survival trees within each cluster. The determination of the clusters using the BIC criterion where the EEV model: 9 clusters 6990.94; 7 clusters -30105.59; and 6 clusters -44616.29. The elbow analysis available in Figure 4.10 shows that the curve is flattened after 7 clusters. Therefore, it was considered an optimal number of seven clusters, which was used to partition the customers. The calculation of the clusters to each member in the dataset was developed considering, 7 clusters (listing 4.7). The model was created for each cluster fitting the train set and calculating the accuracy using the test set (listing 4.8). The overall descriptive statistics of each cluster is available at table 4.5.

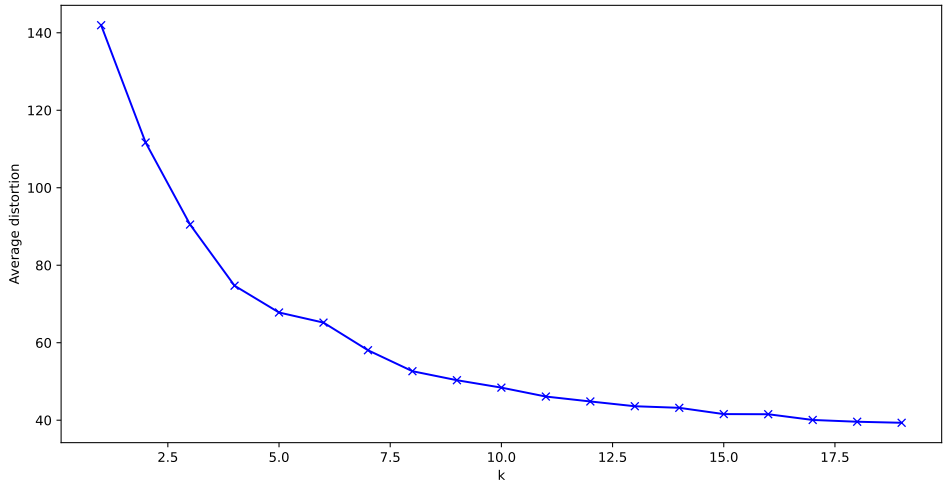


FIGURE 4.10: Elbow analysis fitness customers

```
# we are going to use random_state = 0 for the centroid
# initialization being deterministic allowing a better
# reproducibility
cluster = KMeans(n_clusters=7, random_state=0)

cluster.fit(df_members)
df_members['cluster']=cluster.predict(df_members)
```

LISTING 4.7: KMeans clustering

```
# Building training and testing sets for the clusters
df_resultados = pd.DataFrame(columns=['cluster','rmse','mean','median'])

for cluster in df_members.cluster.unique():
    # Number of samples in the dataset
    df_members_cluster = df_members[df_members.cluster == cluster].copy()
    X = df_members_cluster.copy()
    t = df_members_cluster['months']
    e = df_members_cluster['dropout']
    X.drop(axis=1, columns=['months','dropout'], inplace=True)
    X_train, X_test, t_train, t_test, e_train, e_test = train_test_split(X,
     t, e, random_state=0)
    # Fitting the model
    csf = RandomSurvivalForestModel(num_trees=20)
    csf.fit(X_train, t_train, e_train, max_features='sqrt', max_depth=5,
            min_node_size=20, seed = 1)

    results_cluster = compare_to_actual(csf, X_test, t_test, e_test,
                                is_at_risk = False, figure_size=(12, 6),
                                metrics = ['rmse', 'mean', 'median'])


    df_resultados = df_resultados.append({'cluster' : cluster, 'rmse' :
   results_cluster['root_mean_squared_error'], 'mean' : results_cluster['
   median_absolute_error'], 'median' : results_cluster['
   mean_absolute_error']}, ignore_index = True)
```

LISTING 4.8: Creating the survival model for each cluster

Table 4.4: Summary statistics of features used

Clusters	EII	VII	EEI	EEE	EEV
1	-147909.2	-147909.16	-147986.2	-131747.0	-131747.01
2	-141159.2	-124106.47	-139423.1	-124961.9	-105347.59
3	-113847.4	-112419.03	NA	-116796.3	-86361.73
4	-138388.0	-106804.02	NA	-116891.4	-81762.20
5	-132923.0	-100286.69	NA	-116141.7	-73540.95
6	-129317.3	-97531.03	NA	-111579.3	-44616.29
7	-127247.0	-94434.32	NA	NA	-30105.79
8	-125740.5	-90293.40	NA	NA	NA
9	-116596.0	-87473.09	NA	-102552.8	6990.94

Note: the models with one cluster where omited from the table: VEI -148014.6; EVI -148014.6; VVI -148014.6; VEE -132972.8; EVE -132972.8; VVE -132972.8; VEV -132972.8; EVV -132972.8 and VVV -132972.8

TABLE 4.5: Summary statistics of each cluster

Characteristic	**Cluster 0**, N = 2,473	**Cluster 1**, N = 332	**Cluster 2**, N = 430	**Cluster 3**, N = 84	**Cluster 4**, N = 1,033	**Cluster 5**, N = 817	**Cluster 6**, N = 40
Age in years, Mean (SD)	27 (11)	34 (15)	27 (10)	41 (15)	30 (13)	25 (9)	29 (10)
dayswfreq, Mean (SD)	29 (23)	37 (59)	298 (40)	23 (74)	39 (40)	142 (35)	679 (149)
tbilled, Mean (SD)	83 (35)	471 (90)	91 (58)	927 (199)	241 (54)	87 (45)	183 (109)
maccess, Mean (SD)	1.01 (0.81)	1.18 (0.70)	0.25 (0.21)	1.59 (0.91)	1.07 (0.73)	0.47 (0.35)	0.16 (0.13)
freeuse, %	2.3%	11%	7.9%	15%	7.8%	4.0%	0%
nentries, Mean (SD)	14 (12)	98 (57)	12 (12)	189 (115)	50 (31)	14 (12)	18 (15)
cfreq, %							
2	1.0%	2.7%	2.1%	2.4%	1.6%	0.7%	0%
4	1.1%	8.4%	0.9%	8.3%	4.5%	1.2%	5.0%
6	¡0.1%	1.5%	0%	1.2%	0.3%	0%	0%
7	98%	87%	97%	88%	94%	98%	95%
months, Mean (SD)	5 (5)	23 (8)	12 (4)	32 (7)	14 (7)	8 (3)	27 (6)
dropout, %	90%	62%	99%	14%	85%	98%	72%
Male or female, %	36%	42%	28%	36%	38%	31%	38%

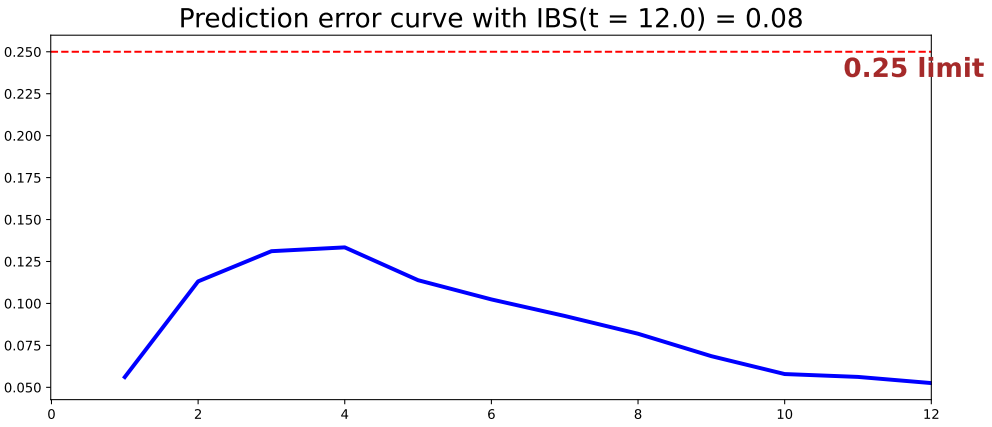


FIGURE 4.11: cluster 0

The performance in the prediction (IBS score), Mean Absolute Score and Median Absolute Score in the clusters 0,1,2,3,4,5,6 and 7 is presented next. The performance of the cluster 0 the IBS presents an accuracy of 0.08 (figure 4.11) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 10.27 customers, the median absolute error was 2.129 and the Root Mean Square Error of 5.204 (figure 4.12). The features importance in the survival model cluster 0 (table 4.6) identifies the three most relevant features to predict survival $maccess$, $freeuse$, and $cfreq$. The features with lower relevance were sex, $tbilled$, and $dayswfreq$.

The performance of the cluster 1 the IBS presents an accuracy of 0.04 (figure 4.14) along

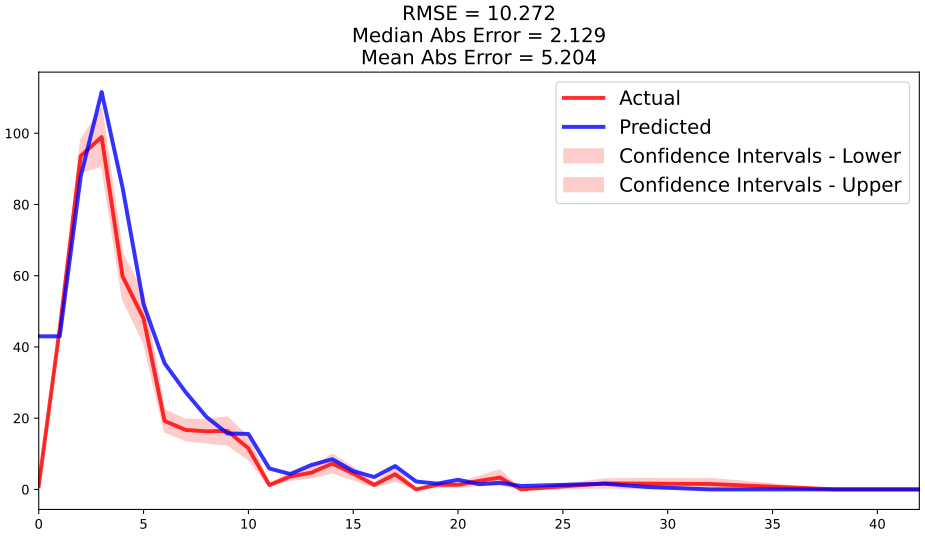


FIGURE 4.12: cluster 0

TABLE 4.6: Features importance in the survival model with cluster 0

feature	importance	pct_importance
maccess	4.467	0.532
freeuse	2.152	0.256
cfreq	1.026	0.122
age	0.638	0.076
nentries	0.120	0.014
sex_1	-0.475	0.000
tbilled	-1.177	0.000
dayswfreq	-1.402	0.000

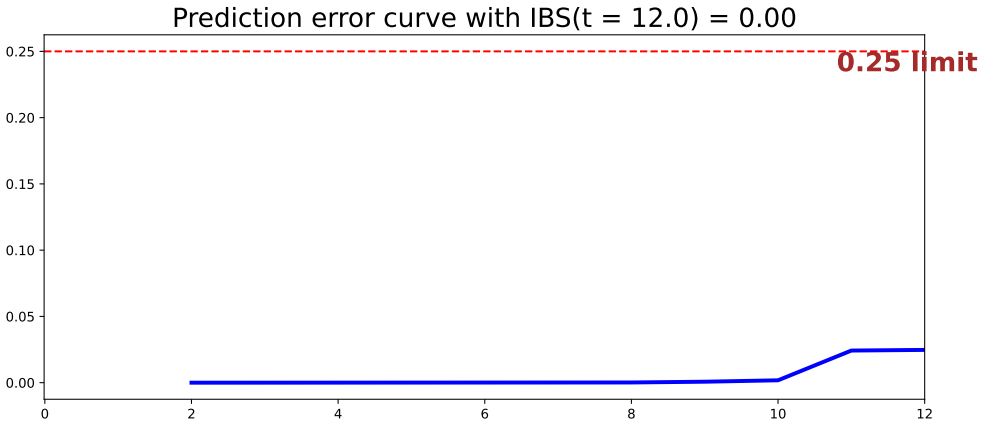


FIGURE 4.13: Conditional survival forest cluster 1

all time. The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 2.149 customers, the median absolute error was 0.698 and the Root Mean Square Error of 3.999 (figure 4.13).

The features importance in the survival model cluster 1 (table 4.7) identifies the three most relevant features to predict survival *maccess*, *nentries*, and *tbilled*. The features with lower relevance were *sex*, $cfreq$, and $freeuse$.

The performance of the cluster 2 the IBS presents an accuracy along time 0.04 (figure 4.15) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 3.501 customers, the median absolute error was 1.018 and the Root Mean Square Error of 2.08 (figure 4.16). The features importance in the survival model cluster 2 (table **??**) identifies

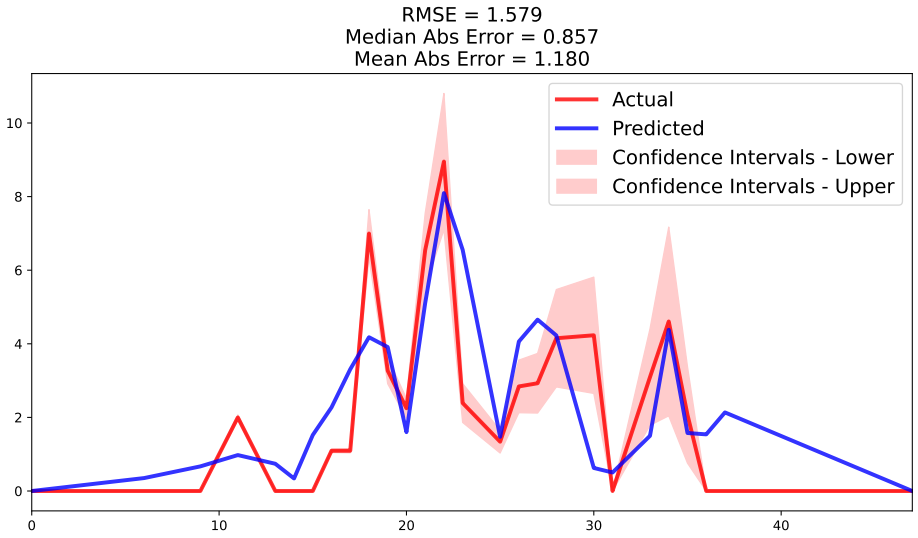


FIGURE 4.14: Model performance cluster 1

TABLE 4.7: Features importance in the survival model with cluster 1

feature	importance	pct_importance
maccess	5.453	0.305
nentries	3.853	0.216
tbilled	3.232	0.181
dayswfreq	2.481	0.139
age	1.646	0.092
sex_1	0.658	0.037
cfreq	0.536	0.030
freeuse	0.000	0.000

the three most relevant features to predict survival *tbilled*, *nentries*, and $daysw freq$. The least relevant were $freeuse$, $cfreq$, and *age*.

The performance of the cluster 3 the IBS was not possible to calculate due to the smaller size of the number of elements (n=84). The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 2.086 customers, the median absolute error was 1.018 and the Root Mean Square Error of 3.501 (figure 4.17). The features importance in the survival model cluster 3 (table 4.9) identifies the three most relevant features to predict survival $daysw freq$, *maccess*, and *nentries*. The least relevant were $freeuse$, $cfreq$, and *sex*.

The performance of the cluster 4 the IBS presents an accuracy along time 0.06. (figure 4.18) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 1.884

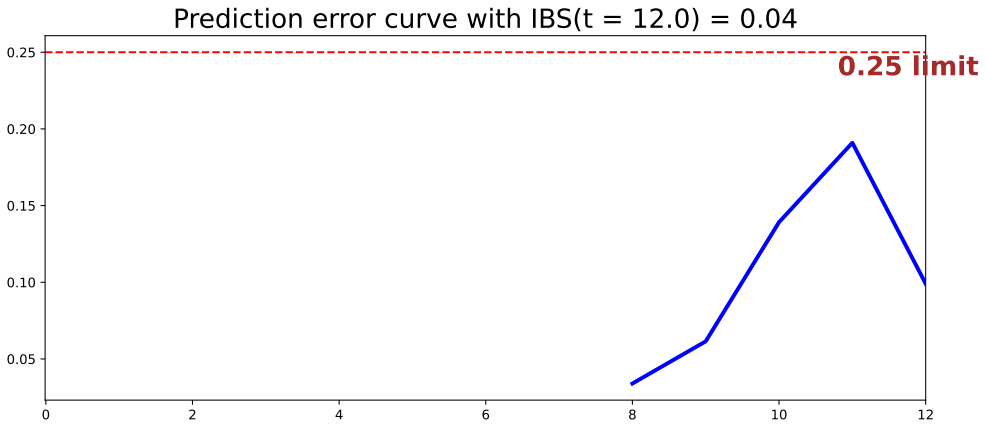


FIGURE 4.15: Conditional survival forest cluster 2

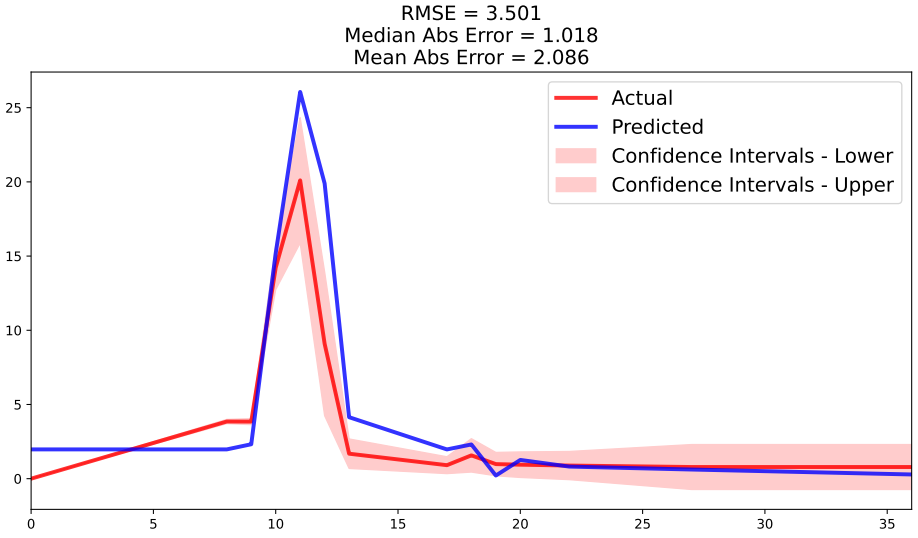


FIGURE 4.16: Model performance cluster 2

TABLE 4.8: Features importance in the survival model with cluster 2

feature	importance	pct_importance
tbilled	2.793	0.548
nentries	1.592	0.312
dayswfreq	0.565	0.111
sex_1	0.125	0.024
maccess	0.022	0.004
freeuse	0.000	0.000
cfreq	0.000	0.000
age	-0.689	0.000

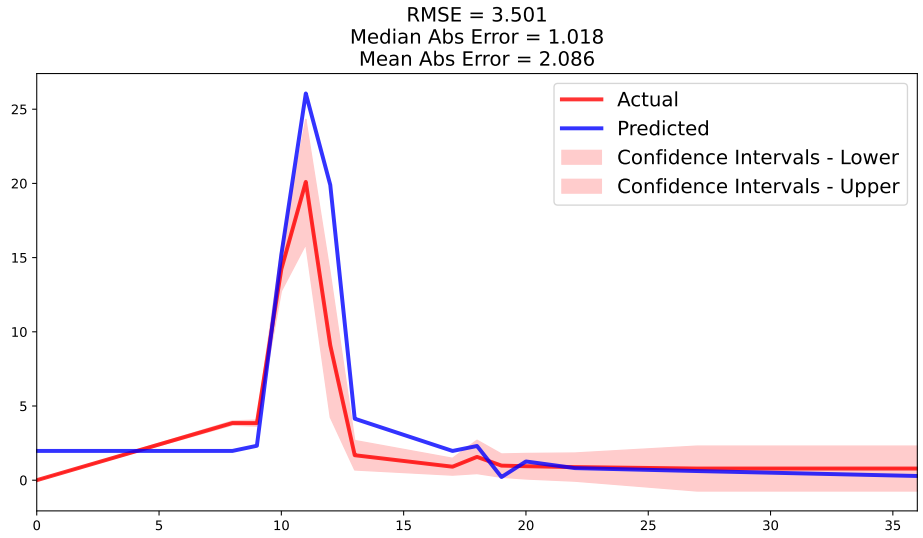


FIGURE 4.17: Model performance cluster 3

TABLE 4.9: Features importance in the survival model with cluster 3

feature	importance	pct_importance
tbilled	2.793	0.548
nentries	1.592	0.312
dayswfreq	0.565	0.111
sex_1	0.125	0.024
maccess	0.022	0.004
freeuse	0.000	0.000
cfreq	0.000	0.000
age	-0.689	0.000

customers, the median absolute error was 1.465 and the Root Mean Square Error of 2.680 (figure 4.19). The features importance in the survival model cluster 4 (table 4.10), identifies the three most relevant features to predict survival *maccess*, $daysw freq$, and *nentries*. The least relevant were $freeuse$, *sex*, and $cfreq$.

The performance of the cluster 5 the IBS presents an accuracy along time 0.08 (figure 4.20) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 3.341 customers, the median absolute error was 1.859 and the Root Mean Square Error of 4.594 (figure 4.21). The features importance in the survival model cluster 5 (table 4.11) identifies the three most relevant features to predict survival *nentries*, *maccess*, and *age*. The least relevant were $freeuse$, $cfreq$, and *sex*.

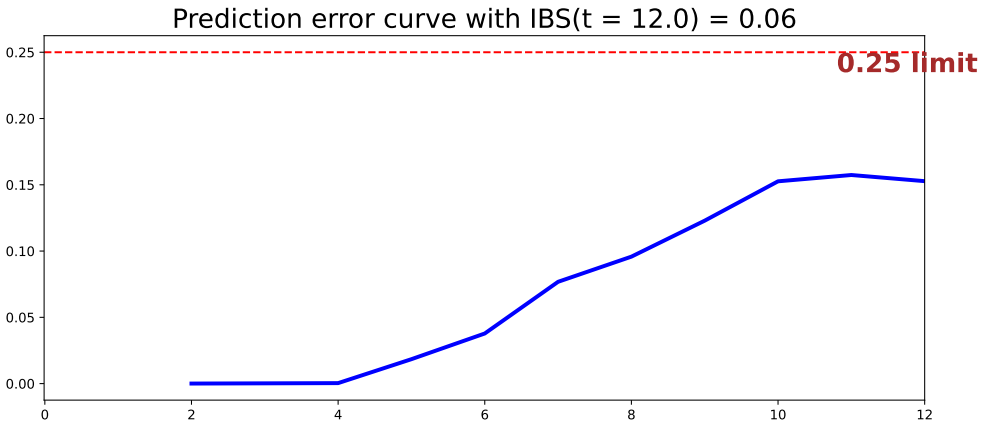


FIGURE 4.18: Conditional survival forest cluster 4

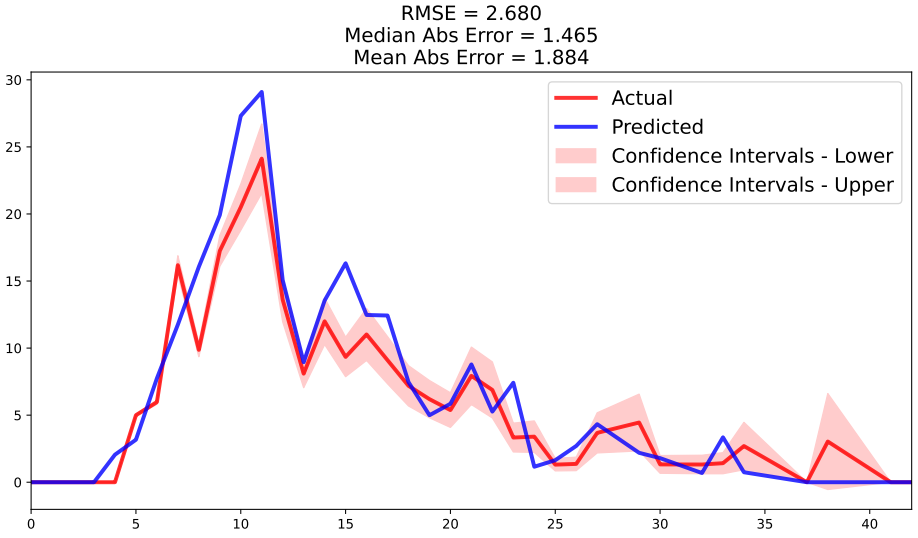


FIGURE 4.19: Model performance cluster 4

TABLE 4.10: Features importance in the survival model with cluster 4

feature	importance	pct_importance
maccess	6.718	0.257
dayswfreq	5.076	0.194
nentries	4.362	0.167
tbilled	3.593	0.138
age	2.706	0.104
freeuse	2.565	0.098
sex_1	1.086	0.042
cfreq	-1.039	0.000

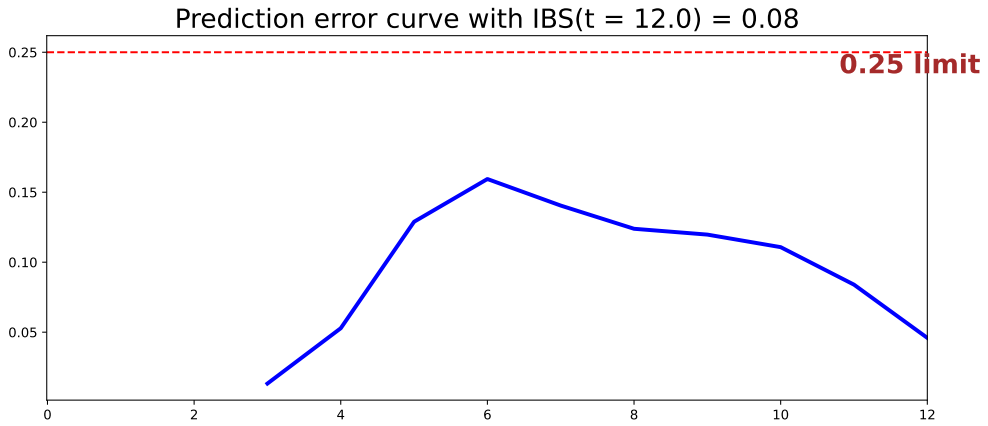


FIGURE 4.20: cluster 5

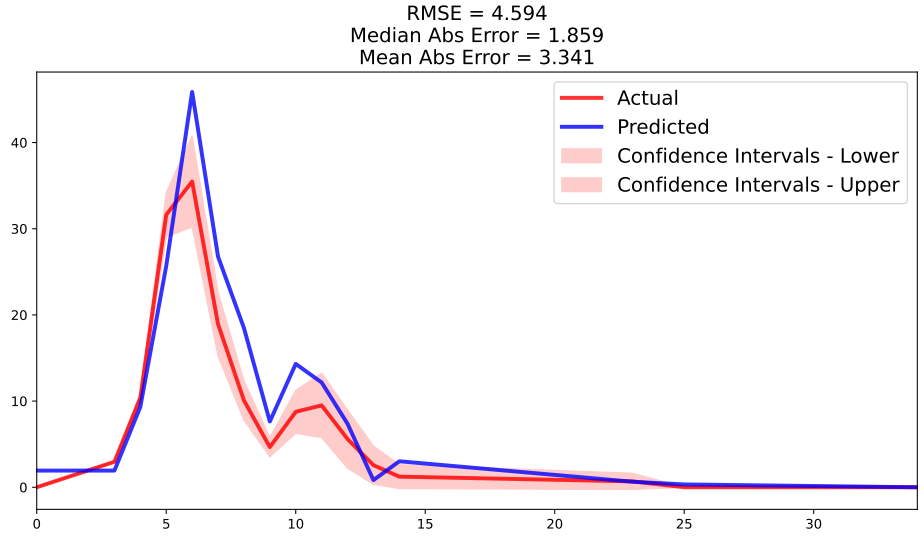


FIGURE 4.21: cluster 5

TABLE 4.11: Features importance in the survival model with cluster 5

feature	importance	pct_importance
nentries	2.794	0.269
tbilled	2.589	0.250
maccess	2.011	0.194
dayswfreq	1.755	0.169
age	1.227	0.118
freeuse	0.000	0.000
cfreq	0.000	0.000
sex_1	-1.015	0.000

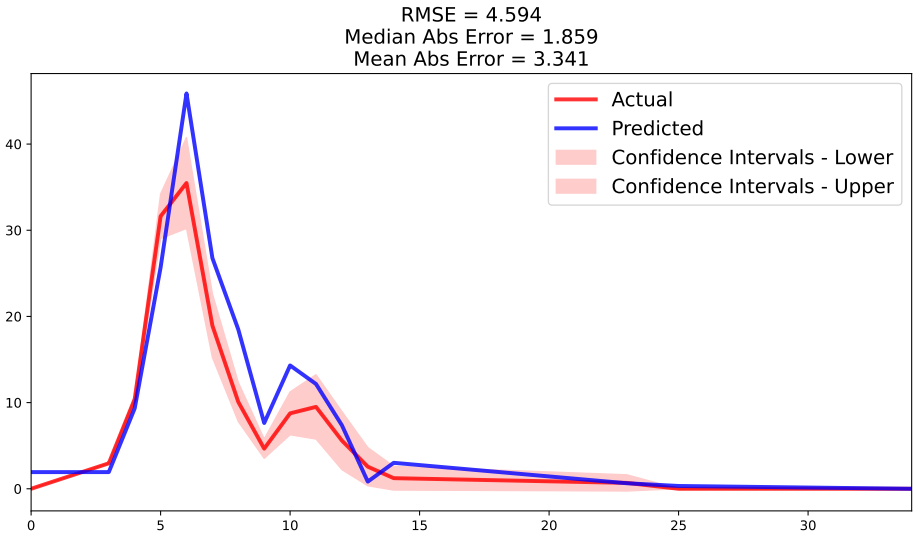


FIGURE 4.22: cluster 6

TABLE 4.12: Features importance in the survival model with cluster 6

feature	importance	pct_importance
nentries	2.794	0.269
tbilled	2.589	0.250
maccess	2.011	0.194
dayswfreq	1.755	0.169
age	1.227	0.118
freeuse	0.000	0.000
cfreq	0.000	0.000
sex_1	-1.015	0.000

The performance of the cluster 6 the IBS was not possible to calculate due to the smaller size of the number of elements (n=40). The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 3.641 customers, the median absolute error was 1.859 and the Root Mean Square Error of 4.594 (figure 4.22). The features importance in the survival model cluster 6 (table 4.12) identifies the three most relevant features to predict survival *nentries*, *tbilled*, and *maccess*. The least relevant were $freeuse$, $cfreq$, and sex.

TABLE 4.13: Brier Score performance prediction in each cluster

cluster	rmse	mean	median	ibs	n	ntrain	ntest
0	9.547	1.791	4.968	0.084	2473	1854	619
1	1.620	1.043	1.235	0.003	332	249	83
2	3.242	1.060	1.923	0.038	430	322	108
3	0.453	0.000	0.256	NaN	84	63	21
4	2.655	0.990	1.812	0.062	1033	774	259
5	4.048	1.709	3.071	0.079	817	612	205
6	0.689	0.400	0.521	NaN	40	30	10
w/cluster	13.888	3.756	7.226	0.084	5209	3646	1563

Note:
ibs NaN value not possible to calculate; n represents the number of elements in each cluster. ntrain and ntest are the number of elements used to train and test the model

4.1.4 Model comparison

The majority of the members where integrated in the cluster 0 (2473 members), followed by cluster 4 (1033 members), and cluster 5 with 817 members (table 4.13). Table 4.13 shows the performance of both approaches, with or without clusters. The model accuracy without clusters had a RMSE of 13.888, the mean absolute error mean was 3.756 customers, the median absolute error was 7.226, and IBS 0.084. The cluster in the model using clusters with the worse performance (cluster 5) had a RMSE 4.048, mean absolute error 1.709 and median absolute error 3.071. The model with the best performance (cluster 1) had a RMSE of 1.62, mean absolute error 1.043 and median absolute error 1.235. The overall performance of the model improved using clusters, the IBS 0.084 was only surpassed by cluster 0 with an IBS 0.084

The model using clusters allowed to combine the customers in different clusters, an hybrid approach. Based on this performance the proposed model using clusters improved the accuracy on the survival model allowing to target approaches considering the timing when the dropout occurs, considering the clusters where the customer is. Is very important for managers use this information to improve their retention strategies.

4.2 Sport club hybrid survival model

In this case, data from a sport club was analysed. This sport club is one of the biggest soccer clubs in Portugal. The club stores information about customers transactions, stadium entries and payments made, and other demographic information. The information retrieved was: Age of the participants in years; Sex (F-female, M-male); Marital status (Single, Married and other); Monthly fee member; Total payed amount until the data was retrieved; Match attendance and Months since last payment.

Considering the sport club policies all the customers with payments less than 24 months where considered active. The variables extracted from the software corresponded to the time interval of becoming a customer until the end of observation (censoring on 31 May 2019) or the end of the customer relationship (dropout). The survival time in the dataset is represented by the number of years the customer begin affiliated. We extracted records of 25316 customers (male n=17246, female n=8070); data corresponded to the time period between October 1, 1944 and May 31, 2019.

Table 4.14 shows data's summary statistics. The average age is 27.3 $\pm$ 20.1, the members have an attendance of 27 $\pm$ 45.8 with a membership of 11 $\pm$ 10.9 years. Figure 4.23 shows the distribution of the dropout considering the number of years of membership. The dropout is greater in the first 10 years.

4.2.1 Survival analysis

As described previously in the survival analysis of the health club (section 4.1), the survival probabilities are presented as a survival curve, representation of the survival probabilities corresponding to a time where the events are observed (Bland & Altman, 1998). The survival curves where developed using the package lifelines (Davidson-Pilon, 2021).

The time of dropout is represented by T, which is a non-negative random variable, indicating the time period of the event occurring for a randomly selected individual from the population, representing the probability of an event to occur each time period given that has not already occurred in a previous time period, known as discrete-time hazard function (Singer & Willett, 1993). The survival function represents the probability of an individual surviving after time t, $S(t) = P(T > t)$, $t \geq 0$, with the properties $S(0) =$

TABLE 4.14: Summary statistics of features used

Characteristic	N = 25,316
Age in years, Mean (SD)	27 (20)
Male or female, %	
F	32%
M	68%
Single, married and other., %	
Married	20%
Not defined	30%
Other	2.0%
Single	48%
monthly_fee, %	
0	¡0.1%
1	32%
2.5	28%
5	3.4%
6	12%
10	24%
total_amount, Mean (SD)	316 (494)
total_matches, Mean (SD)	27 (46)
season_matches, Mean (SD)	2.2 (4.1)
months_since_last_payment, Mean (SD)	19 (32)
dropout, %	22%
years_membership, Mean (SD)	11 (11)
stadium_access, %	40%
quart_stadium_entries, %	
1 a 21	10%
21 a 56	9.8%
56 a 105	10.0%
ate 1	60%
mais 105	10.0%
inscription_month, Mean (SD)	6.9 (3.4)

$1, S(\infty) = 0$. The distribution function is represented with F, defined as $F(t) = P(T \leq 0)$, for $t \geq 0$. The function of probability density represented with f where:

$$f(t) = \lim_{dt \to 0^+} \frac{P[t \leq T < t + dt]}{dt} \tag{4.4}$$

$f(t)dt$ represents the probability of an event occurring in the moment t. The need to represent the distribution evolution of the death probability along the time, uses to the hazard function, represented as:

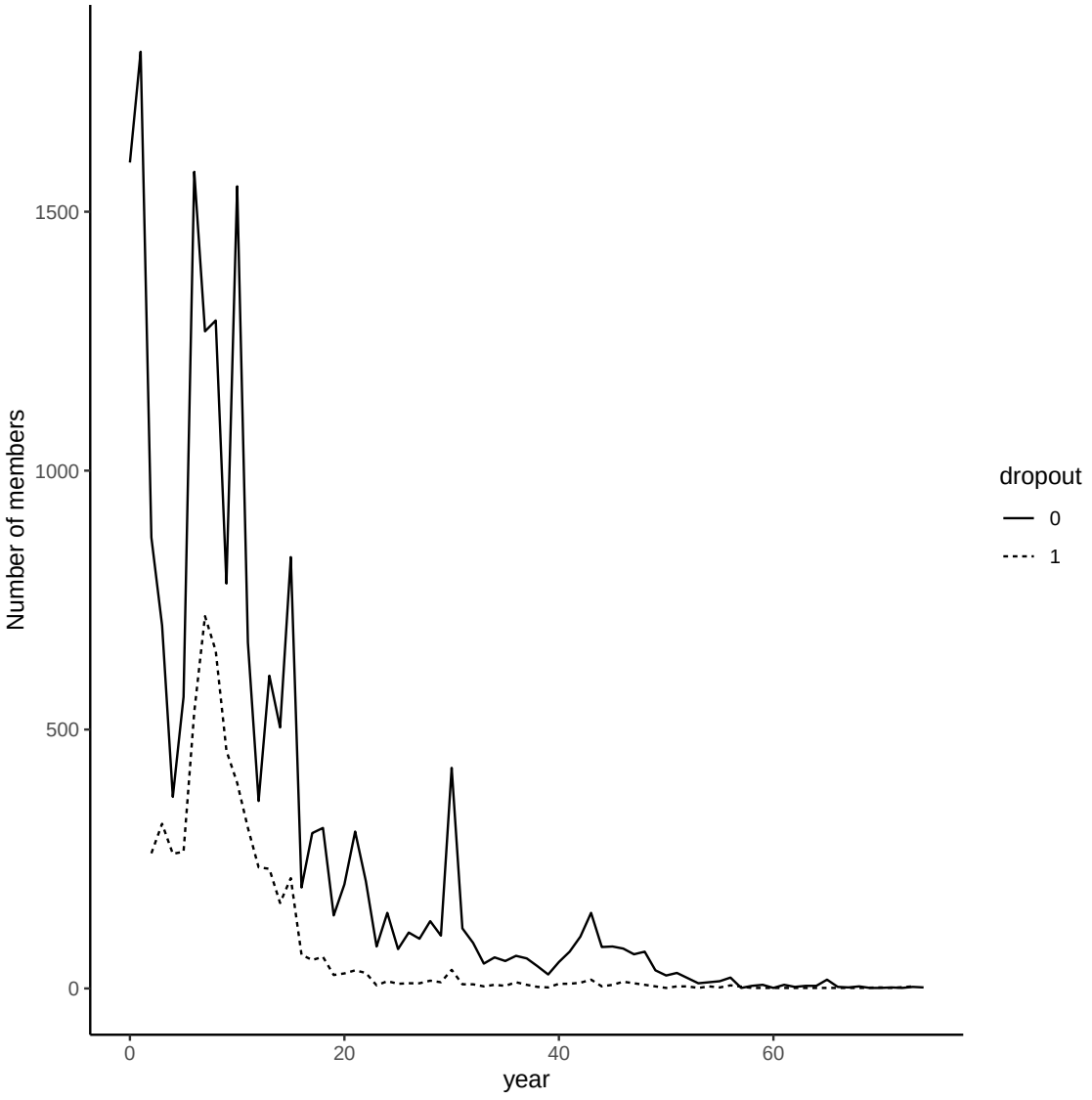


FIGURE 4.23: Number of members by year

$$\lambda(t) = \lim_{dt \to 0^+} \frac{P\left[t \leq T < t + dt | \; T \geq t\right]}{dt} \tag{4.5}$$

The determination of the survival curves is based in the following elements: (1) the total value of observations removed during the time period (month), either by dropout or by censorship; (2) observations that composed the sample of the study (N= 5219); (3) customers who had not yet dropped out at any given time. The survival probability until the time period ii (pi) is calculated with:

$$p_i = \frac{r_i - d_i}{r_i} \tag{4.6}$$

```
from lifelines import KaplanMeierFitter
kmf = KaplanMeierFitter()
T = df_members_sport['years_membership']
C = df_members_sport['dropout']

kmf.fit(T,C,label="Customers")

kmf.event_table.reset_index()
kmf.conditional_time_to_event_

survival_table = pd.concat([kmf.event_table.reset_index(),
                            kmf.conditional_time_to_event_.reset_index
(),
                            kmf.survival_function_.reset_index()],axis
=1)

survival_table.drop(['timeline'],axis=1,inplace=True)
survival_table.columns = ['event_at', 'removed', 'observed', 'censored'
,
                          'entrance', 'at_risk', 'estimated_survival',
                          'prob']
```

LISTING 4.9: Cox proportional hazard

Where r_i is the number of individuals that survived at the beginning of the period, d_i the number of individuals who left during the period. The survival time estimate was also taken considering the month in which it is found (estimated). Cox's allow test difference between survival times. The advantage in using survival analysis was that allow us to detect if the risk of an event differs systematically across different people, using specific predictors. The coefficients in a Cox regression were related to the hazard, where a positive value represents a worse prognosis and the opposite, negative value a better prognosis. The advantage of survival analysis was that allow us to include information of covariates that were censored up to the censoring event.

The survival curves where determined using the Cox proportional hazard model available in the package lifelines (listing 4.9).

The table 4.15 depicts the data of the survival time of the customers during the first months, the results showed that the customers have a survival probability of 77.8% at 10 years membership (column p_i - likelihood probability). The survival probability at 6 months was 54.5%, representing an risk of dropout of 45.5% with a estimated survival of 6 months.

Figure 4.24 shows the Kaplan Meier survival curve customers considering the number of years of membership (x axis) and survival probability (y axis). The customer membership

TABLE 4.15: Determination of the survival time probabilities

event_at	removed	observed	censored	entrance	at_risk	estimated_survival	prob
0	1595	0	1595	25316	25316	48	1.000
1	1809	0	1809	0	23721	47	1.000
2	1132	261	871	0	21912	47	0.988
3	1019	318	701	0	20780	48	0.973
4	630	260	370	0	19761	47	0.960
5	827	264	563	0	19131	47	0.947
6	2111	534	1577	0	18304	48	0.919
7	1988	719	1269	0	16193	49	0.878
8	1942	652	1290	0	14205	48	0.838
9	1241	459	782	0	12263	55	0.807
10	1946	397	1549	0	11022	58	0.778
11	978	310	668	0	9076	57	0.751
12	596	234	362	0	8098	58	0.729
13	835	231	604	0	7502	57	0.707
14	669	165	504	0	6667	56	0.689
15	1046	213	833	0	5998	55	0.665
16	260	65	195	0	4952	54	0.656
17	355	55	300	0	4692	54	0.649
18	371	61	310	0	4337	53	0.639
19	167	26	141	0	3966	52	0.635
20	230	29	201	0	3799	51	0.630
21	338	35	303	0	3569	50	0.624
22	237	30	207	0	3231	49	0.618
23	87	6	81	0	2994	48	0.617
24	160	14	146	0	2907	47	0.614

Note: Removed – the sum of customers with dropout and that are censored; Censored – the event did not occur during the period of this data, collection; Risk of Dropout – number of customers at risk of, dropout; pi – survival probability; Estimated Survival - months to survive in the sports facility.

dropout after the first 13 years is about 70%, representing 30% of dropout. In the next 10 years is approximately of 10%.

Figure 4.25 shows the survival by gender. The males have an increased survival probability with values higher than the females , both types of customers present a behavior that is not very different. The table 4.16 depicts the data of the survival time of the male customers during the first years, the results showed that the male customers have a survival probability of 79.7% at 10 years membership (column p_i). The survival probability at 20 years was 67%, representing an risk of dropout of 33%. The table 4.17 depicts the data of the survival time of the female customers during the first years, the results showed that

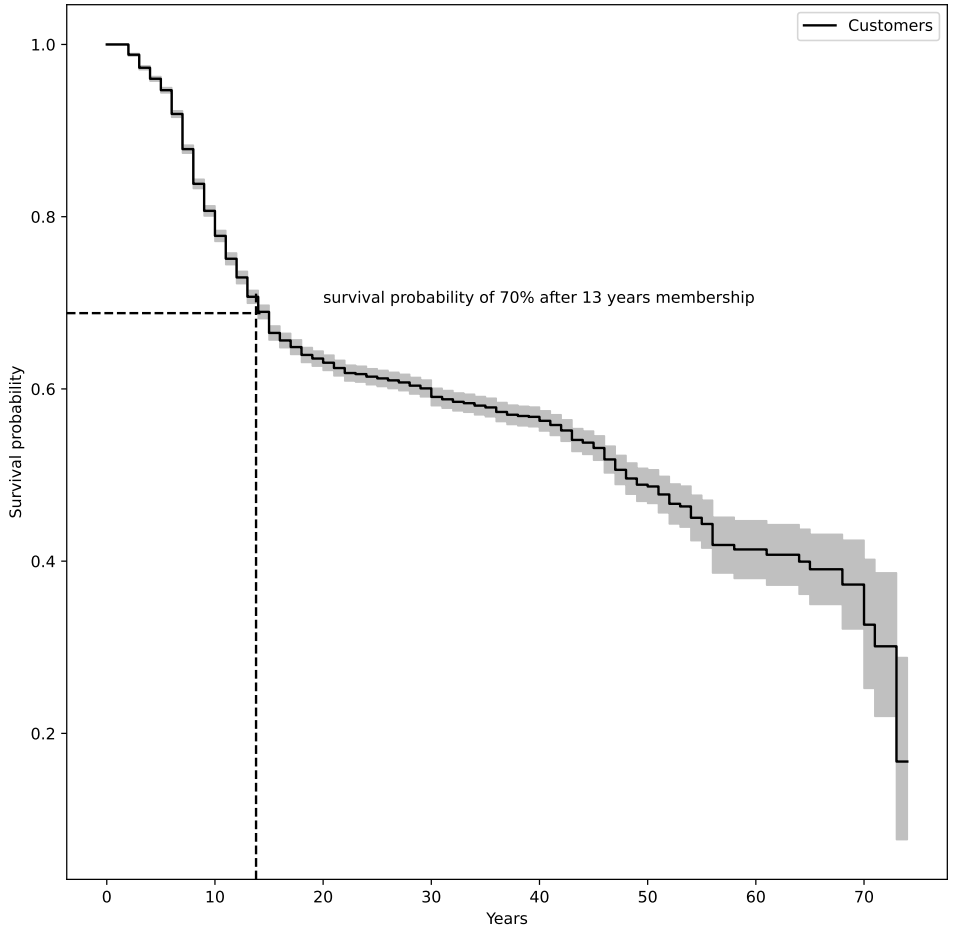


FIGURE 4.24: Survival probability

the male customers have a survival probability of 73.3% at 10 years membership (column p_i). The survival probability at 20 years was 51.9%, representing an risk of dropout of 48.1%. It seems that females have a bigger risk of dropout against males.

The model assumptions was determined using listing 4.10. The proportional hazard assumptions failed in the following variables: monthly_fee $p < 0.01$, total_amount $p < 0.01$, season_matches $p < 0.01$, months_since_last_payment $p < 0.01$, inscription_month $p < 0.01$, sex_M $p < 0.01$, marital_status_nao $p < 0.01$, marital_status_solteiro $p < 0.01$, quart_stadium_entries_21to56 $p < 0.01$, and quart_stadium_entries_56to105 $p < 0.01$.

The Cox PH model assumes the covariates to be time independent, for example gender when where retrieved do not change over time (Schober & Vetter, 2018b). Because the Cox

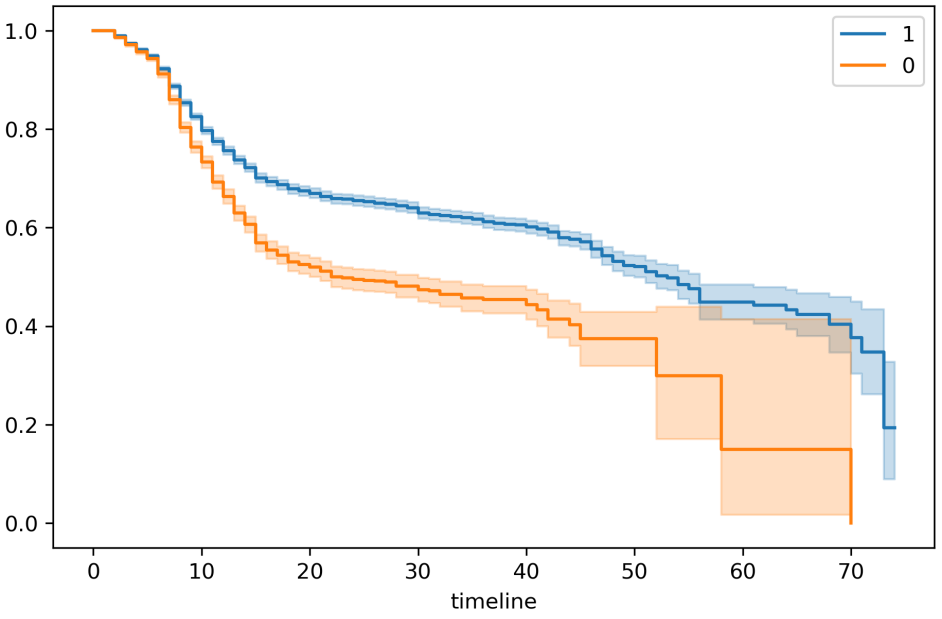


FIGURE 4.25: Survival probability by gender

```
vars= ["age","monthly_fee","total_amount","season_matches",
       "months_since_last_payment","dropout","years_membership",
       "stadium_access", "inscription_month","sex_M",
       "marital_status_nao definido","marital_status_outro",
       "marital_status_solteiro","quart_stadium_entries_21 a 56",
       "quart_stadium_entries_56 a 105",
       "quart_stadium_entries_mais 105"]

df_regression = df_members[vars].copy()

from lifelines import CoxPHFitter
cph = CoxPHFitter()
cph.fit(df_regression,duration_col='months',event_col='dropout')

cph.check_assumptions(df_regression)
```

LISTING 4.10: Checking model assumptions

model requires the hazards in both groups to be proportional, researchers are often asked to "test" whether hazards are proportional (Stensrud & Hernán, 2020). Considering this we explored other approach that allow us to develop this analysis without the proportional hazard assumptions, the survival trees.

TABLE 4.16: Determination of the survival by gender male

event_at	removed	observed	censored	entrance	at_risk	estimated_survival	prob
0	1120	0	1120	17246	17246	53	1.000
1	1200	0	1200	0	16126	52	1.000
2	739	162	577	0	14926	52	0.989
3	685	224	461	0	14187	51	0.974
4	410	164	246	0	13502	51	0.962
5	574	182	392	0	13092	51	0.948
6	1266	340	926	0	12518	50	0.923
7	1187	435	752	0	11252	54	0.887
8	1163	380	783	0	10065	57	0.853
9	831	293	538	0	8902	59	0.825
10	1268	278	990	0	8071	60	0.797
11	650	186	464	0	6803	59	0.775
12	404	151	253	0	6153	58	0.756
13	539	142	397	0	5749	58	0.737
14	452	111	341	0	5210	57	0.722
15	701	136	565	0	4758	56	0.701
16	184	43	141	0	4057	57	0.694
17	239	39	200	0	3873	56	0.687
18	262	44	218	0	3634	55	0.678
19	115	20	95	0	3372	54	0.674
20	166	23	143	0	3257	53	0.670
21	260	28	232	0	3091	52	0.664
22	186	21	165	0	2831	51	0.659
23	65	4	61	0	2645	50	0.658
24	131	12	119	0	2580	49	0.655

Note:
Removed – the sum of customers with dropout and that are censored; Censored – the event did not occur during the period of this data, collection; Risk of Dropout – number of customers at risk of, dropout; pi – survival probability; Estimated Survival - months to survive in the sports facility.

4.2.2 Random Survival Forest

Random Survival Forests does not make the proportional hazards assumption (Ehrlinger, 2016b) and has the flexibility to model survivor curves that are of dissimilar shapes for contrasting groups of subjects. Random Survival Forest is an extension of Random Forest allowing efficient non-parametric analysis of time to event data (Breiman, 2001). This characteristics allow us to surpass the Cox Regression limitation of the proportional hazard assumption, requiring to exclude variables which not fulfill the model assumption. It was

TABLE 4.17: Determination of the survival time probabilities by gender Female

event_at	removed	observed	censored	entrance	at_risk	estimated_survival	prob
0	475	0	475	8070	8070	23	1.000
1	609	0	609	0	7595	22	1.000
2	393	99	294	0	6986	23	0.986
3	334	94	240	0	6593	25	0.972
4	220	96	124	0	6259	26	0.957
5	253	82	171	0	6039	26	0.944
6	845	194	651	0	5786	30	0.912
7	801	284	517	0	4941	35	0.860
8	779	272	507	0	4140	37	0.803
9	410	166	244	0	3361	36	0.764
10	678	119	559	0	2951	42	0.733
11	328	124	204	0	2273	41	0.693
12	192	83	109	0	1945	40	0.663
13	296	89	207	0	1753	39	0.630
14	217	54	163	0	1457	38	0.606
15	345	77	268	0	1240	43	0.569
16	76	22	54	0	895	42	0.555
17	116	16	100	0	819	41	0.544
18	109	17	92	0	703	40	0.531
19	52	6	46	0	594	39	0.525
20	64	6	58	0	542	38	0.519
21	78	7	71	0	478	37	0.512
22	51	9	42	0	400	36	0.500
23	22	2	20	0	349	35	0.498
24	29	2	27	0	327	34	0.494

Note:
Removed – the sum of customers with dropout and that are censored; Censored – the event did not occur during the period of this data, collection; Risk of Dropout – number of customers at risk of, dropout; pi – survival probability; Estimated Survival - months to survive in the sports facility.

shown by (Breiman, 2001) that ensemble learning can be further improved by injecting randomization into the base learning process - a method called Random Forests.

The random survival forest was developed using the package PySurvival (Fotso & et al., 2019). The most relevant variables predicting the dropout are analysed using the log-rank test. The metric variables are transformed to categorical using the quartiles to provide a statistical comparison of groups. The survival analysis was conducted using the package Lifelines (Davidson-Pilon, 2021). The model was built with with 70% of the data for training and 30% for testing. The survival model parameters where:

```
1
2  from pysurvival.models.survival_forest import RandomSurvivalForestModel
3  from sklearn.model_selection import train_test_split
4  from pysurvival.utils.metrics import concordance_index
5  from pysurvival.utils.display import integrated_brier_score
6  from pysurvival.utils.display import compare_to_actual
7
8  X = df_members_sport.copy()
9  t = df_members_sport['years_membership']
10 e = df_members_sport['dropout']
11 X.drop(axis=1, columns=['years_membership','dropout'])
12
13 X_train, X_test, t_train, t_test, e_train, e_test = train_test_split(X, t,
       e, test_size=0.3, random_state=0)
14
15 # Fitting the model
16 csf = RandomSurvivalForestModel(num_trees=20)
17 csf.fit(X_train, t_train, e_train, max_features='sqrt', max_depth=5,
       min_node_size=20, seed = 1)
18
19 c_index = concordance_index(csf, X_test, t_test, e_test)
20 ibs = integrated_brier_score(csf, X_test, t_test, e_test, t_max=12,
       figure_size=(12,5))
21 results = compare_to_actual(csf, X_test, t_test, e_test, is_at_risk = False
       ,  figure_size=(12, 6), metrics = ['rmse', 'mean', 'median'])
```

Listing 4.11: Creating the survival model for each cluster

PySurvival is an open source python package for Survival Analysis modeling - the modeling concept used to analyze or predict when an event is likely to happen. It is built upon the most commonly used machine learning packages such NumPy, SciPy and PyTorch. The model is created (Listing 4.11) using the duration of the survival "months" and the dropout event "dropout" variables fitting the train set and the accuracy of the model is created using the test set.

The model gives the accuracy with IBS and concordance with the metrics RMSE, Median Absolute Error and Mean Absolute Error. The prediction is very similar to the actual value 4.26. The model accuracy is very high with a root mean square error of 13. The mean absolute error mean was 7.53 customers, and the median absolute error was 4.04.

The model gives the accuracy with IBS and concordance with the metrics RMSE, Median Absolute Error and Mean Absolute Error. The prediction is very similar to the actual value 4.27. The model accuracy is very high with a root mean square error of 13. The mean absolute error mean was 7.53 customers, and the median absolute error was 4.04.

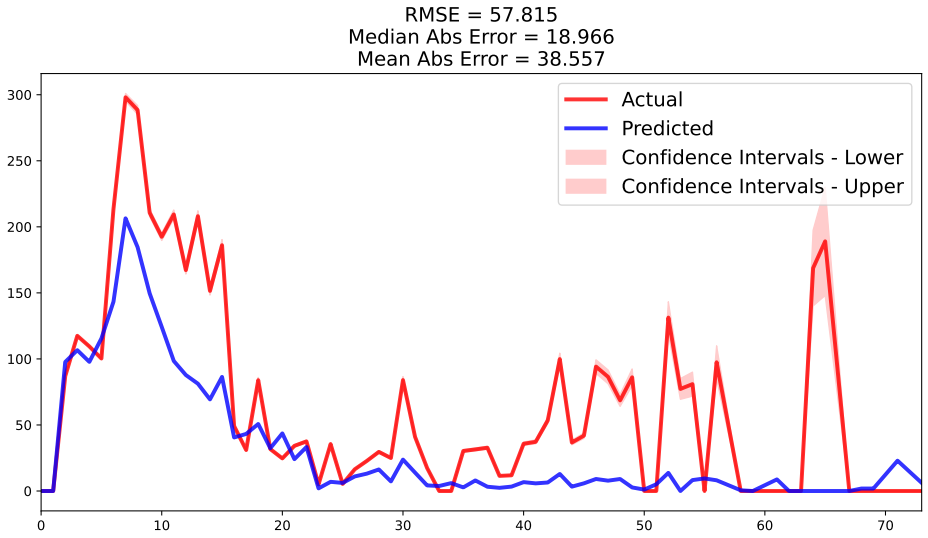


FIGURE 4.26: Model global performance

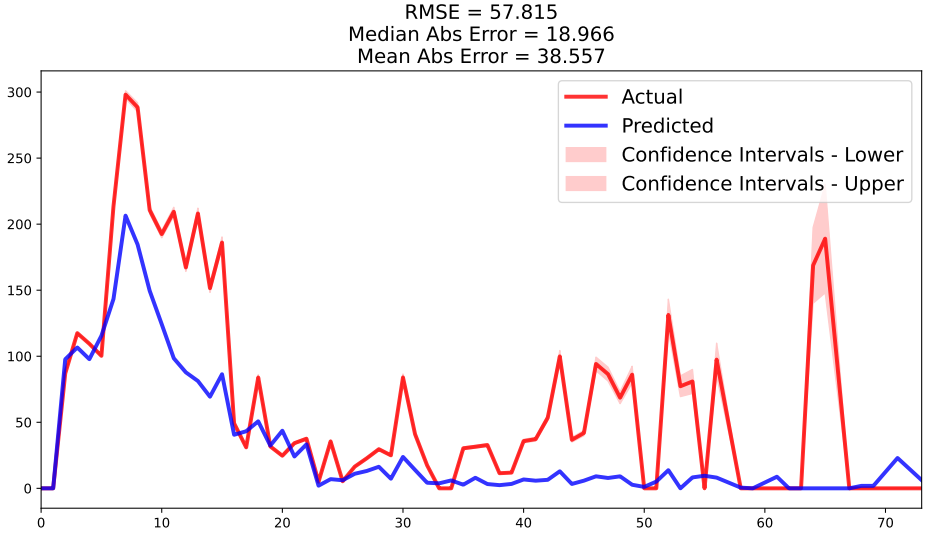


FIGURE 4.27: Global performance versus predicted

The table B.3 shows the variable importance according to Breiman (2001) where the percent increase in misclassification rate as compared to the out-of-bag rate (with all variables intact), out-of-bag is a bootstrap aggregating (subsampling with replacement to create training samples for the model to learn from) where two independent sets are created. One set, the bootstrap sample, data chosen to be in-the-bag by sampling with replacement and the out-of-bag is all data not chosen in the sampling process. The feature importance was determined using the fitted model (Listing 4.12).

```
tbl <- py$csf$variable_importance_table
kbl(tbl, booktabs = T, caption = "Features importance in the survival model
    ")
```

Listing 4.12: Getting feature importance in R from a python object using Reticulate

Table 4.18: Features importance in the survival model

feature	importance	pct_importance
months_since_last_payment	9.5547860	0.3034228
total_amount	4.4419030	0.1410576
season_matches	3.4720831	0.1102598
stadium_access	2.6559085	0.0843413
marital_status_solteiro	2.2583599	0.0717167
monthly_fee	1.7899917	0.0568432
quart_stadium_entries_mais 105	1.5944932	0.0506349
inscription_month	1.5940396	0.0506205
age	1.3756597	0.0436856
quart_stadium_entries_21 a 56	0.9715332	0.0308521
sex_M	0.8753704	0.0277984
quart_stadium_entries_56 a 105	0.3698806	0.0117460
marital_status_nao definido	0.3088305	0.0098073
marital_status_outro	0.2271649	0.0072139

The most important variable is *months_since_last_payment*, *total_amount* and *season_matches*. Less important variables where *quart_stadium_entries_56a105*, *marital_status_naodefinido*

4.2.3 Survival trees based model with clusters

In this model we have created clusters and applied the survival trees within each cluster. The calculation of the number of clusters used the package mclust (Scrucca et al., 2016) using the Bayesian Information Criterion (BIC). The model that gives the minimum BIC score can be selected as the best model (Schwarz, 1978) simplifying the problem related to choosing the number of components and identifying the structure of the covariance matrix, based on modelling with multivariate normal distributions for each component that forms the data set (Akogul & Erisoglu, 2016).

The model which gives the lowest BIC score, is the VEV model (figure 4.28). The top 3 models based on the BIC criterion where the VEV model: 5 clusters -388409.4 and 4 clusters -405189.77. The third best model was EEV with 9 clusters (-431849.90).

```r
library(mclust)
y <- scale(py$dfmemberssport)

set.seed(0) #make it reproducible

bic <- mclustBIC(y)
# Best model using the BIC criteria
plot(bic, what="BIC")
summary(bic, what="BIC")
```

Listing 4.13: Calculation of the optimal number of clusters using BIC

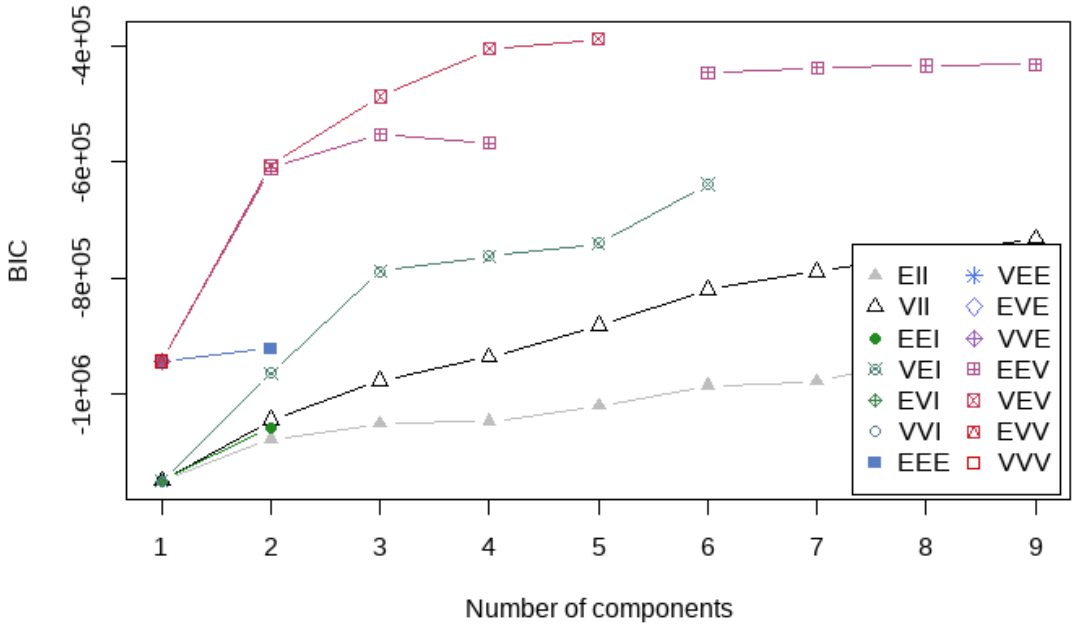


Figure 4.28: Best number of clusters according to BIC

The clusters where calculated to each member considering 5 clusters 4.14. The majority of the members where integrated in the cluster 1 (17069 members), followed by cluster 5 (2420 members), cluster 3 (2817 members), cluster 2 (2080 members) and cluster 4 with only 930 members. The overall descriptive statistics of each cluster is available at table 4.20.

Figure 4.29 shows also the elbow analysis, where is possible to confirm the curve flattening arround five clusters Thorndike (1953) considering the average distortion was flattened. Increasing the number of clusters improves the explanation of the variation.

Table 4.19: Summary statistics of features used

Clusters	EII	VII	EEI	EEE	EEV
1	-1149655.5	-1149655.5	-1149808	-943276.0	-943276.0
2	-1077552.1	-1044319.4	-1057064	-611248.9	-605297.4
3	-1050592.1	-977318.1	NA	-552363.1	-487496.2
4	-1046711.3	-935718.8	NA	-567915.2	-405189.8
5	-1020086.6	-881634.4	NA	NA	-388409.4
6	-985198.0	-819578.9	NA	-446158.3	NA
7	-977577.0	-788765.1	NA	-437428.4	NA
8	-939733.4	-759172.1	NA	-433846.4	NA
9	-935093.9	-731379.2	NA	-431849.9	NA

Note: the models with one cluster where omited from the table: EVI -1149808; VVI -1149808; VEE -943276.8; EVE -943276; VVE -943276 and EVV -943276

Table 4.20: Summary statistics of each cluster

Characteristic	**Cluster 1**, N = 17,069	**Cluster 2**, N = 2,080	**Cluster 3**, N = 2,817	**Cluster 4**, N = 930	**Cluster 5**, N = 2,420
Age in years, Mean (SD)	17 (12)	54 (13)	41 (18)	60 (13)	49 (16)
monthly_fee, %					
0	¡0.1%	0%	¡0.1%	0%	0%
1	47%	0%	¡0.1%	0.1%	¡0.1%
2.5	39%	0%	15%	0%	0%
5	0.6%	6.7%	11%	2.5%	12%
6	5.1%	2.1%	39%	1.9%	45%
10	7.8%	91%	35%	95%	42%
total_amount, Mean (SD)	35 (46)	1,292 (112)	383 (115)	1,823 (134)	801 (123)
season_matches, Mean (SD)	0.9 (2.7)	6.5 (5.1)	3.7 (4.7)	5.1 (4.9)	4.8 (4.9)
months_since_last_payment, Mean (SD)	23 (38)	5 (8)	14 (17)	5 (5)	9 (13)
dropout, %	27%	4.8%	23%	2.3%	11%
years_membership, Mean (SD)	7 (5)	26 (13)	12 (10)	34 (14)	19 (12)
stadium_access, %	20%	88%	76%	84%	84%
inscription_month, Mean (SD)	6.7 (3.5)	7.5 (3.2)	7.0 (3.1)	7.9 (3.1)	7.2 (3.2)
sex_M, %	65%	100%	62%	99%	58%
marital_status_nao definido, %	38%	12%	19%	13%	13%
marital_status_outro, %	0.8%	3.8%	4.8%	4.5%	4.7%
marital_status_solteiro, %	58%	14%	43%	6.1%	28%
quart_stadium_entries_21 a 56, %	5.4%	11%	26%	15%	18%
quart_stadium_entries_56 a 105, %	2.3%	26%	23%	23%	29%
quart_stadium_entries_mais 105, %	1.0%	46%	14%	36%	28%
Male or female, %	65%	100%	62%	99%	58%

```
cluster = KMeans(n_clusters=5)

cluster.fit(df_members_sport)
df_members_sport['cluster']=cluster.predict(df_members_sport)
print(df_members_sport.cluster.value_counts());
```

LISTING 4.14: KMeans clustering

The model was created for each cluster fitting the train set and calculating the accuracy using the test set (listing 4.15).

The performance of the cluster 1 in the IBS presents and accuracy along the time of 0.06 (figure 4.30). The size of cluster 1 is 17069, the root mean squared error was 41.87, median absolute error 20.89, and mean absolute error 31.04 (figure 4.31).

The features importance in the survival model cluster 1 (table 3.17) identify the three most relevant features to predict survival maccess, tbilled, and dayswfreq.

The features importance in the survival model cluster 1 identified the three most relevant features (table 4.21) *months_since_last_payment* representing 25.06% in the

```
# Building training and testing sets for the clusters
for cluster in df_members_sport.cluster.unique():
    # Number of samples in the dataset
    df_members_sport_cluster = df_members_sport[df_members_sport.cluster ==
     cluster]
    X = df_members_sport_cluster.copy()
    t = df_members_sport_cluster['years_membership']
    e = df_members_sport_cluster['dropout']
    X.drop(axis=1, columns=['years_membership','dropout'])
    X_train, X_test, t_train, t_test, e_train, e_test = train_test_split(X,
     t, e, random_state=0)
    print(f"The cluster {cluster} as a size of {X.shape[0]}")
    # Fitting the model
    csf = RandomSurvivalForestModel(num_trees=20)
    csf.fit(X_train, t_train, e_train, max_features='sqrt',
        max_depth=5, min_node_size=20, seed = 1)

    c_index = concordance_index(csf, X_test, t_test, e_test)

    ibs = integrated_brier_score(csf, X_test, t_test, e_test, t_max=12,
    figure_size=(12,5))

    results = compare_to_actual(csf, X_test, t_test, e_test, is_at_risk =
    False,
                                figure_size=(12, 6), metrics = ['rmse', '
    mean', 'median'])
    print(f"Cluster {cluster} RMSE {results['root_mean_squared_error']}")
    print(f"Cluster {cluster} MAE {results['median_absolute_error']}")
    print(f"Cluster {cluster} MAError {results['mean_absolute_error']}")
```

LISTING 4.15: Creating the survival model for each cluster

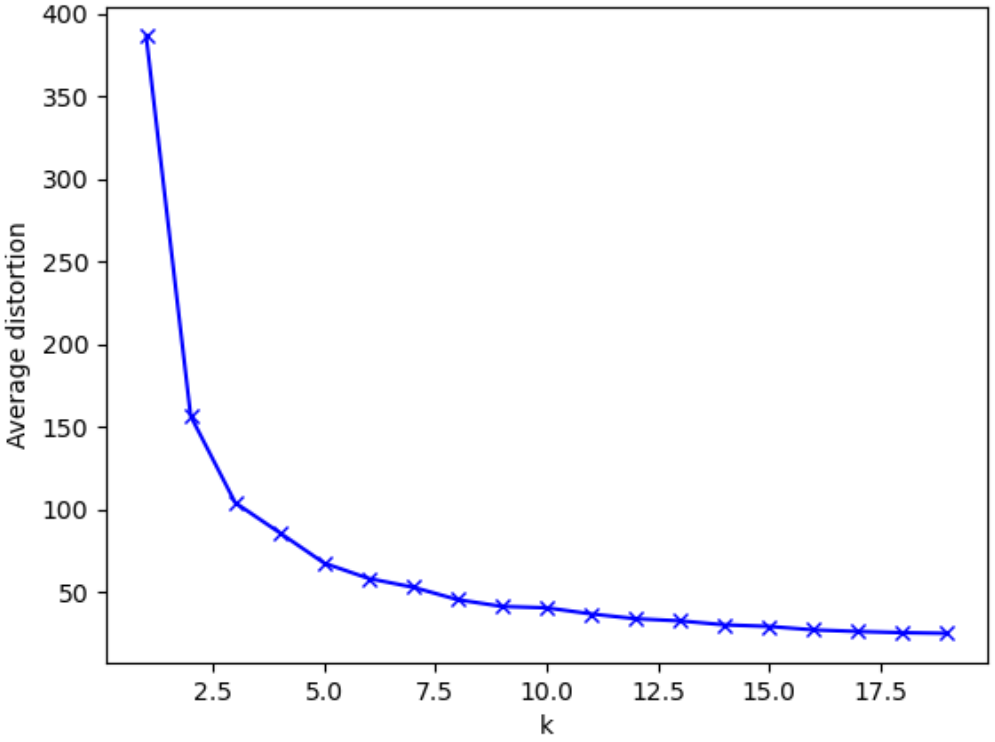


FIGURE 4.29: Elbow analysis fitness customers

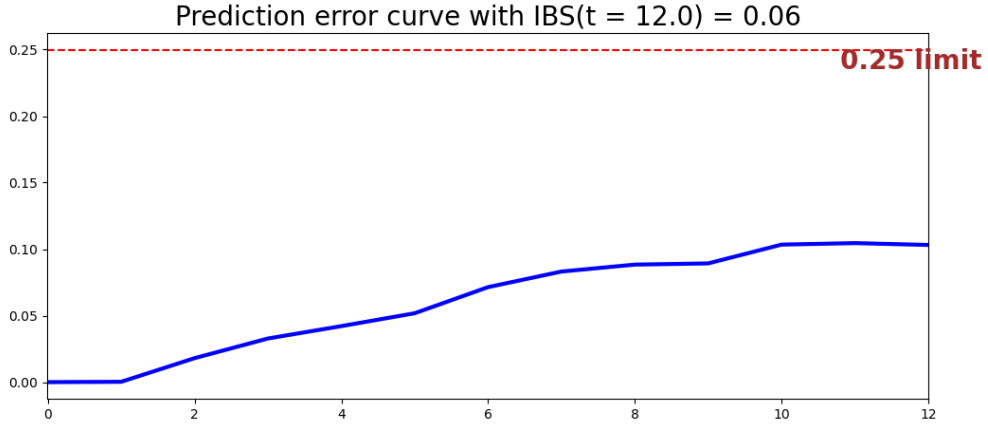


FIGURE 4.30: Model performance cluster 1

prediction importance, followed by *total_amount* with 10.97% and *season_matches* with 10%

The performance of the cluster 2 in the IBS presents and accuracy along the time of

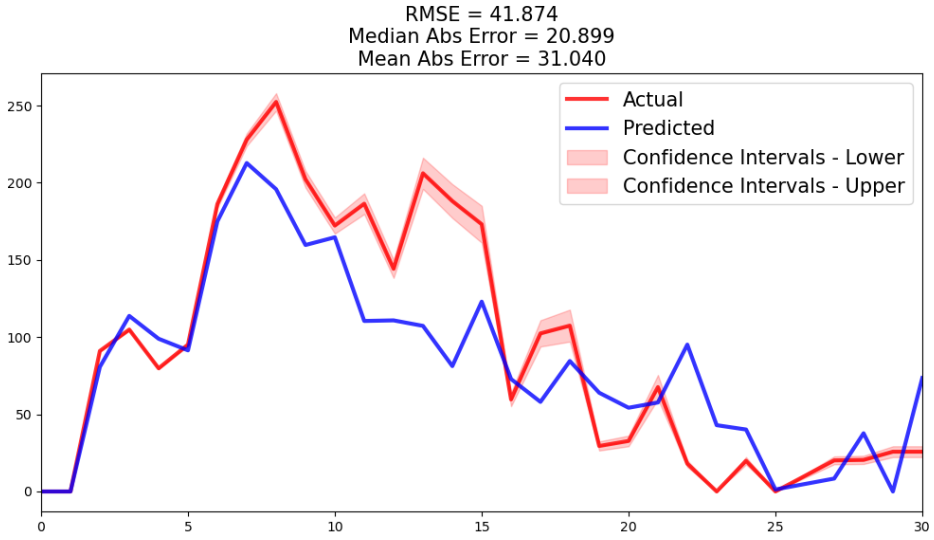


FIGURE 4.31: Performance cluster 1 actual versus predicted

TABLE 4.21: Features importance in the survival model with cluster 1

feature	importance	pct_importance
months_since_last_payment	7.9718880	0.2506004
total_amount	3.4904383	0.1097237
quart_stadium_entries_21 a 56	3.0461302	0.0957567
season_matches	3.0189963	0.0949037
monthly_fee	2.8196026	0.0886357
inscription_month	2.2867690	0.0718858
marital_status_solteiro	2.1419768	0.0673341
quart_stadium_entries_mais 105	1.7729598	0.0557339
stadium_access	1.7154423	0.0539258
marital_status_nao definido	1.3549447	0.0425934
quart_stadium_entries_56 a 105	1.2276591	0.0385921
age	0.9643497	0.0303148
sex_M	-0.7487548	0.0000000
marital_status_outro	-1.8563364	0.0000000

0.00 (figure 4.32). The size of cluster 2 is 2080, RMSE 8.805, MAE 1.878, and MAError 3.866 (figure 4.31). The features importance in the survival model cluster 2 (table 4.22) identify the three most relevant features *months_since_last_payment* representing 24.91% in the prediction importance, followed by $season_matches$ with 20.76% and $quartstadiumentriesmais105$ with 9.48%.

The performance of the cluster 3 in the IBS presents and accuracy along the time of 0.03 (figure 4.34). The size of cluster is 2817, RMSE 8.529, MAE 5.404, and MAError 6.268 (figure 4.35). The features importance in the survival model cluster 3 (table 4.23)

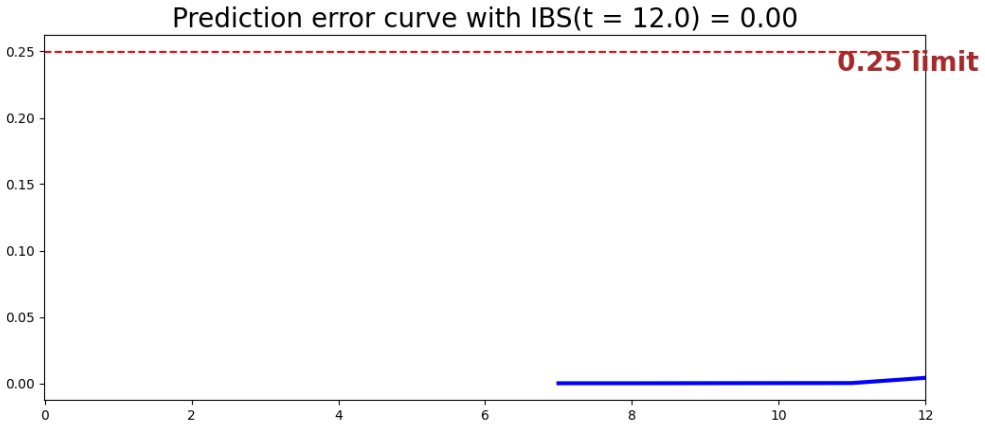


FIGURE 4.32: Model performance cluster 2

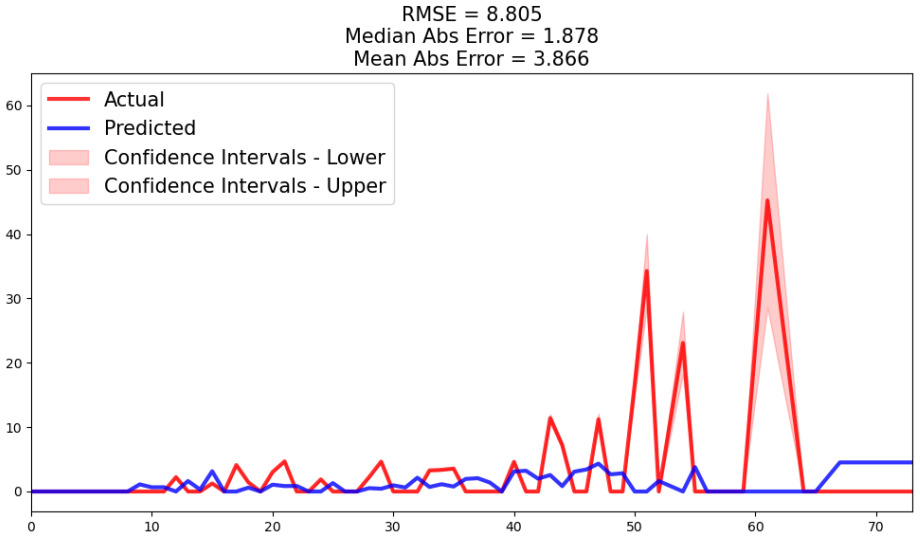


FIGURE 4.33: Performance cluster 2 actual versus predicted

identify the three most relevant features *months_since_last_payment* representing 24.59% in the prediction importance, followed by *monthly_fee* with 12.77% and *season_matches* with 10%.

The performance of the cluster 4 in the IBS presents and accuracy along the time of 0.0 (figure 4.36). The size of cluster is 930, RMSE 2.761, MAE 0.000, and MAError 0.840 (figure 4.37). The features importance in the survival model cluster 4 (table 4.24) identify the three most relevant features *months_since_last_payment* representing 28.73% in the prediction importance, followed by *total_amount* with 21.47% and *age* with 8.76%.

The performance of the cluster 5 in the IBS presents and accuracy along the time of 0.01 (figure 4.38). The size of cluster is 2420, RMSE 15.401, MAE 2.508, and MAError 6.914.

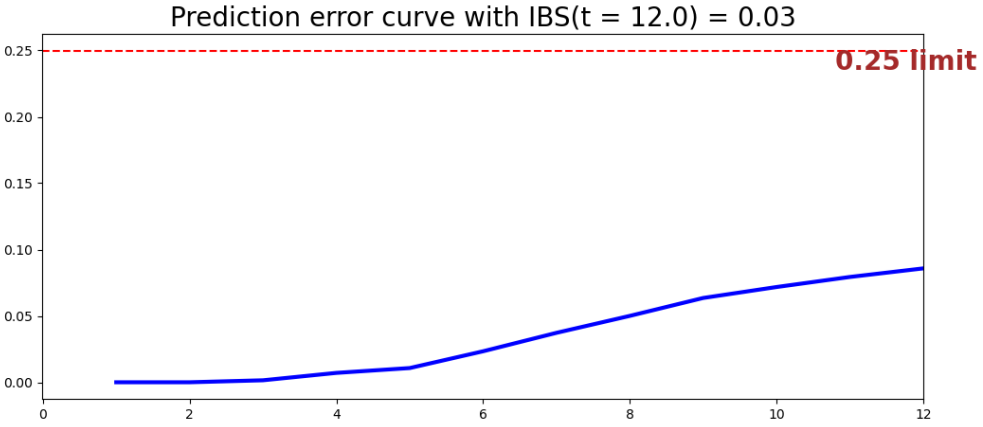


FIGURE 4.34: Model performance cluster 3

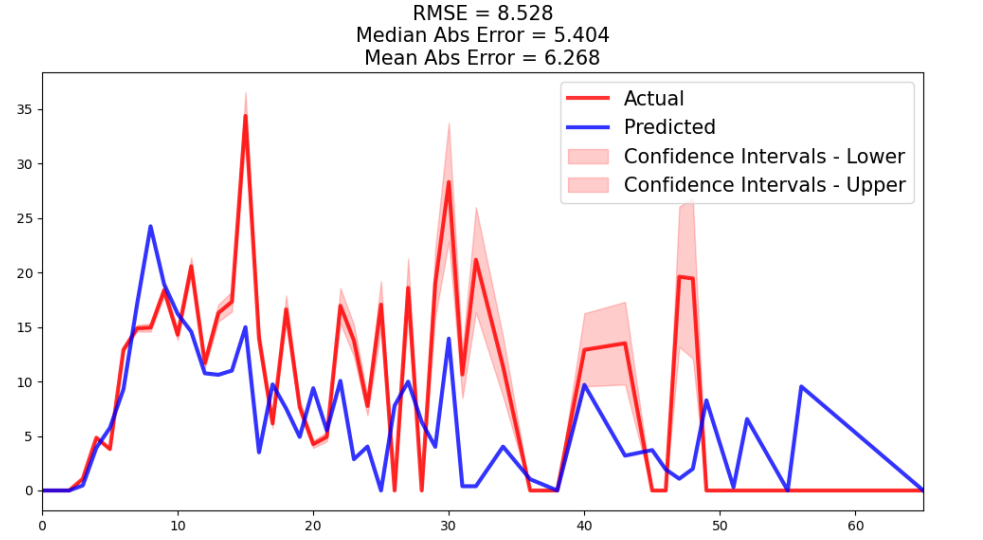


FIGURE 4.35: Performance cluster 3 actual versus predicted

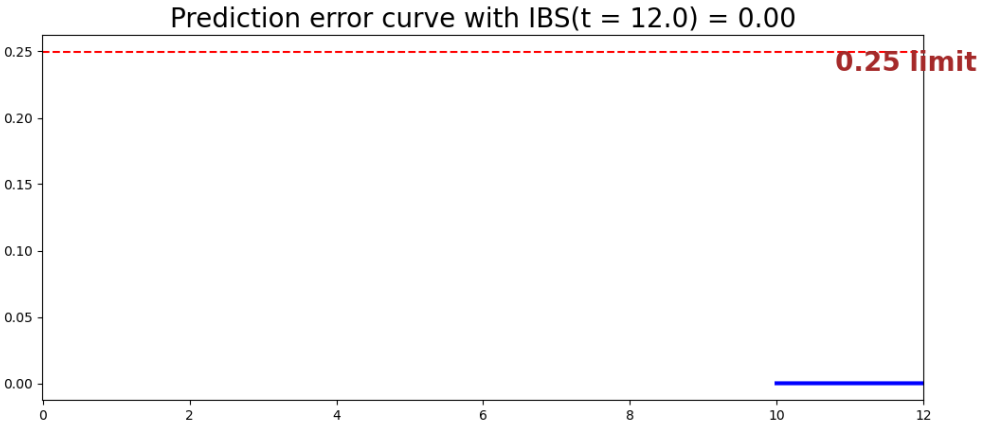


FIGURE 4.36: Model performance cluster 4

TABLE 4.22: Features importance in the survival model with cluster 2

feature	importance	pct_importance
months_since_last_payment	5.9872066	0.2491094
season_matches	4.9914874	0.2076806
quart_stadium_entries_mais 105	2.2792423	0.0948323
stadium_access	2.2573120	0.0939199
age	1.9097863	0.0794604
marital_status_solteiro	1.6355833	0.0680516
marital_status_nao definido	1.3333022	0.0554746
monthly_fee	1.0488715	0.0436403
quart_stadium_entries_21 a 56	1.0286933	0.0428008
total_amount	0.9596941	0.0399299
quart_stadium_entries_56 a 105	0.6032673	0.0251001
sex_M	0.0000000	0.0000000
marital_status_outro	0.0000000	0.0000000
inscription_month	-1.3324701	0.0000000

TABLE 4.23: Features importance in the survival model with cluster 3

feature	importance	pct_importance
months_since_last_payment	8.2219201	0.2458512
monthly_fee	4.2692367	0.1276584
season_matches	3.3464787	0.1000662
total_amount	2.5500942	0.0762527
age	2.5107861	0.0750773
quart_stadium_entries_mais 105	2.4517413	0.0733118
stadium_access	1.8968781	0.0567203
quart_stadium_entries_56 a 105	1.8554828	0.0554825
marital_status_solteiro	1.7158599	0.0513075
quart_stadium_entries_21 a 56	1.6516070	0.0493862
inscription_month	1.4075370	0.0420881
sex_M	1.1621292	0.0347499
marital_status_nao definido	0.4029124	0.0120479
marital_status_outro	-1.0259784	0.0000000

The features importance in the survival model cluster 5 (table 4.24) identify the three most relevant features *months_since_last_payment* representing 28.46% in the prediction importance, followed by *season_matches* with 11.03% and *monthly_fee* with 8.00%.

The features importance in the survival model cluster 5 (table 4.25)

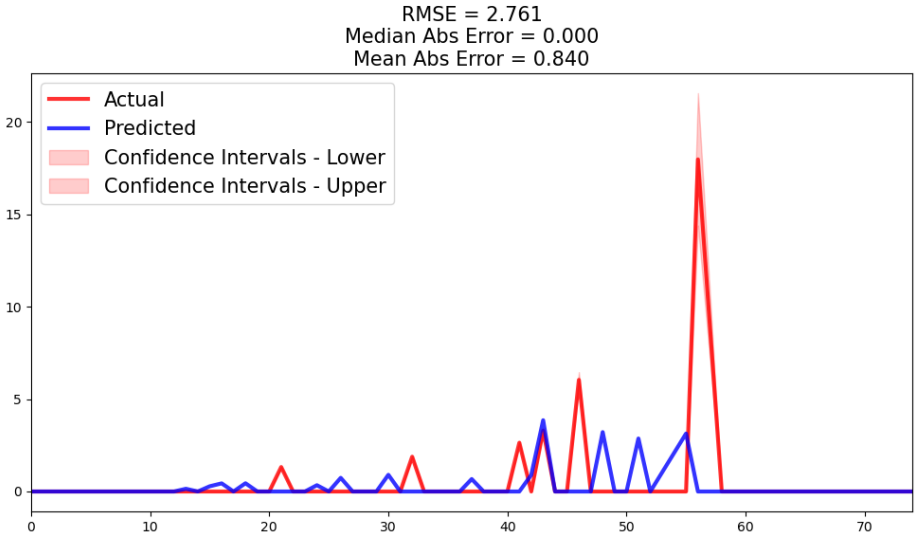


FIGURE 4.37: Performance cluster 4 actual versus predicted

TABLE 4.24: Features importance in the survival model with cluster 4

feature	importance	pct_importance
months_since_last_payment	4.8835861	0.2873432
total_amount	3.6484226	0.2146679
age	1.4886006	0.0875871
quart_stadium_entries_mais 105	1.3693405	0.0805700
inscription_month	1.1782424	0.0693261
quart_stadium_entries_56 a 105	1.1495877	0.0676401
marital_status_solteiro	1.0893559	0.0640961
season_matches	1.0259784	0.0603671
quart_stadium_entries_21 a 56	0.6879737	0.0404794
stadium_access	0.4745695	0.0279230
monthly_fee	0.0000000	0.0000000
sex_M	0.0000000	0.0000000
marital_status_outro	0.0000000	0.0000000
marital_status_nao definido	-0.6536132	0.0000000

4.2.4 Model comparison

Table 4.26 shows the performance of both approaches, with or without clusters. The model accuracy without clusters is very high with a root mean square error of 57, the mean absolute error mean was 38.557 customers, and the median absolute error was 18.966. The model using clusters improved the performance significantly with a RMSE in cluster 1 of 41.874, cluster 2 28.805, cluster 3 8.528, cluster 4 2.761, and cluster 5. The performance using clusters improved significantly.

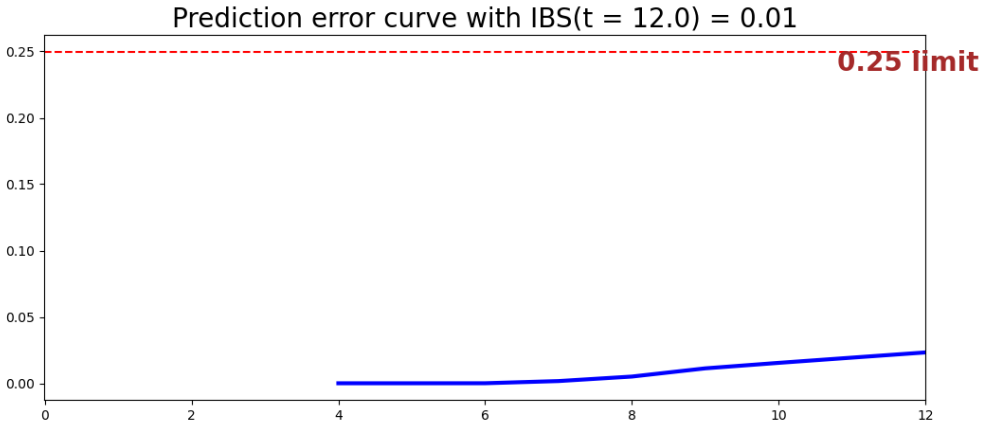


FIGURE 4.38: Model performance cluster 5

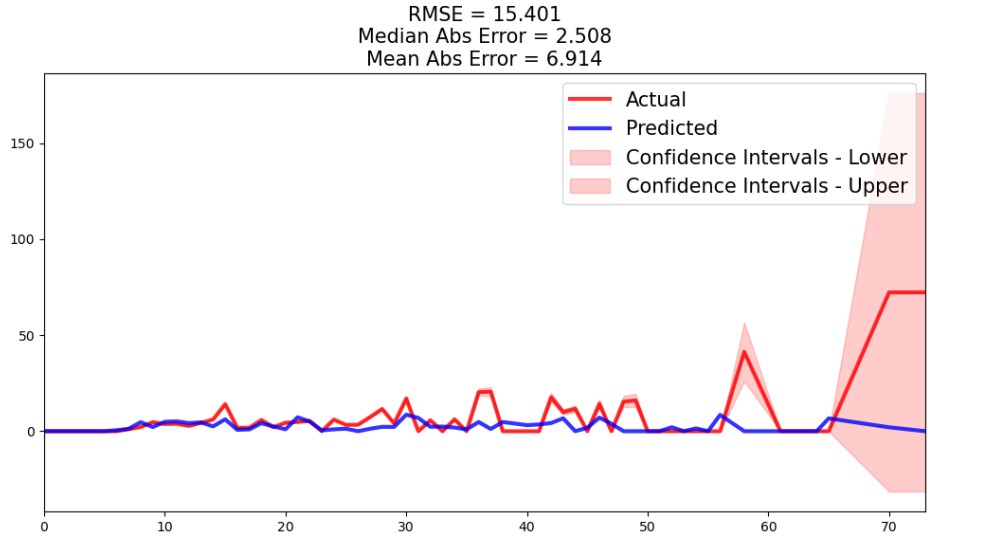


FIGURE 4.39: Performance cluster 5 actual versus predicted

The model using clusters allowed to combine the customers in different clusters, an hybrid approach. Based on this performance the proposed model using clusters improved the accuracy on the survival model allowing to target approaches considering the timing when the dropout occurs, considering the clusters where the customer is. Is very important for managers use this information to improve their retention strategies.

4.3 Discussion

We have evaluated the performance of random survival forests using membership data from a health club and a sport club. In both cases the survival model was created to

TABLE 4.25: Features importance in the survival model with cluster 5

feature	importance	pct_importance
months_since_last_payment	9.648520	0.2846462
season_matches	3.738756	0.1102991
monthly_fee	2.711904	0.0800053
total_amount	2.557423	0.0754479
quart_stadium_entries_mais 105	2.387826	0.0704445
quart_stadium_entries_21 a 56	2.299669	0.0678438
age	2.296840	0.0677603
stadium_access	2.066592	0.0609676
sex_M	1.871691	0.0552178
inscription_month	1.821203	0.0537283
marital_status_solteiro	1.389542	0.0409936
quart_stadium_entries_56 a 105	1.106573	0.0326456
marital_status_outro	0.000000	0.0000000
marital_status_nao definido	-1.025978	0.0000000

TABLE 4.26: Performance of prediction in each cluster

cluster	rmse	mean	median	n
w/cluster	57.815	38.557	18.966	25316
Cluster 1	41.874	20.899	31.040	17069
Cluster 2	8.805	1.878	3.866	2080
Cluster 3	8.528	5.404	6.268	2817
Cluster 4	2.761	0.000	0.840	930
Cluster 5	15.401	2.508	6.914	2420

determine the prediction accuracy of the relationship duration. This approach provides a way to identify when the customer dropout will occur.

In the health club case study more than 70% of the customers were predicted to dropout in the first 12 months, and in the sport club 30% of the customers had a trend of dropout in the first 12 years. The results very different, and the contexts are also. In the sport club we had to infer the dropout, using the time since last payment, whether in the health was possible to consider the information in the analysed data. The health club high dropout, which is very high, and has not been identified in other studies. Burez & Vandenpoel (2008), in a study of pay TV users, have found that one out of three customers leave the company before one year, and half the customers leave within two years.

The accuracy calculated using the actual and predicted customers who dropped out during the 40 months showed a mean absolute error of 7.5 customers. Using the hybrid model, the mean absolute errors were 1.56, 6.52, and 1.56 customers in clusters 1,2, and 3,

respectively. The features daysw freq, tbilled and nentries represented more than 66% of the importance predicting the survival model without clusters. Accordingly, in the hybrid approach, the most relevant features were nentries, tbilled, and daysw freq, representing 67% of the importance in the cluster 1; in Cluster 2, daysw freq, tbilled, and nentries represented 70% in prediction importance; and, in Cluster 3, tbilled, daysw freq, and nentries representing 69% in the prediction importance.

Regarding to the sport club case study during the first 12 years 30% of the members dropout, which is a value very different from the previous case, here the mean absolute error showed an higher value, with 38.5, however the sample size is much bigger, considering that the n in the health club is $5,209$ and the sport club is $25,316$, which is approximately 5 times greather. In this case we have a membership of a sport club which as another context, however the values are similar in terms of accuracy considering the sample size. Using the hybrid model, the mean was 20.8, 1.87, 5.4, 0.0 and 2.5 in clusters 1, 2, 3, 4, and 5, respectively. The most relevant features were *months_since_lastpayment*, *total_amount*, *season_matches*, representing 55% of the importance predicting the survival model without clusters. Using the hybrid approach, the most relevant features in cluster 1 were *months_since_lastpayment*, *total_amount*, *quart_stadium_entries* 21 *a* 56, cluster 2 *months_since_last payment*, *season_matches*, and *quart_stadium_entries* 21 *a* 56, cluster 3 *months_since_last payment*, $monthly\_fee$, and *season_matches*, cluster 4 *months_since_last payment*, *total_amount*, and *age*, cluster 5 *months_since_last payment*, *season_matches*, and $monthly\_fee$.

The exploration of clustering to develop customer segmentation to improve the performance of machine learning techniques is not new. Jafari-Marandi et al. (2020) used also clusters to improve the prediction accuracy. However, this approach combined with the use of survival models, for our knowledge, has not been previously attempted or reported. The better performance of the hybrid model in predicting when customers will dropout, using existing data, supports the development of management counter-measures to reduce dropout. The duration of the relationship between the customer and the organi- zation is an important aspect, allowing us to understand that the decision of the customer to dropout changes over time, which implies that existing models predicting customer dropout may be only correct at a specific point in time, after which their decision may change (Alboukaey et al., 2020).

The time perspective allows us to identify the period in which retention actions should be developed; therefore, the prediction should be as accurate as possible. In general, we can consider two dimensions in the prediction of the dropout, one with a static perspective and the other with a dynamic perspective, considering that the risk varies over time. From a static perspective we have predictions of customer dropout, which are carried out at a given time, here we essentially use algorithms that are trained with test data (for learning the models, hence the concept of machine learning) to predict the outcome of a customer's dropout or not. The algorithms generally used are in most cases nature-inspired (e.g., neural networks or swarm intelligence), decision trees (the algorithms preferred by the interpretability that they have), among others, such as Logistic Regression, Random Forests or support vectors machines. These studies that we have done with this static perspective in predicting dropout, allows us to help quantify the risk of churning that a customer has at a certain point in, without considering, however, that the risk of dropout varies over time.

We have explored perspective with the use of "dynamic" algorithms, such as the survival analysis. The word "dynamic" is used because it allows to obtain the risk of dropout that customers have over time, bringing a temporal perspective. Combining the use of clusters to input the survival model, by using the output of one model as input from another, we implement the algorithms in pipeline, and we were able to increase the accuracy of the survival models being used.

Chapter 5

Conclusions and Future Work

This chapter presents the main conclusions after the development of this thesis. It summarizes the main contributions presented as publications and future work to be developed. Finally, is presented a final reflection.

5.1 Conclusions

This thesis development started during 2015. Through those years it was being developed the work related to Machine Learning predicting customer dropout in Sport Organizations and started explore new approaches to solve the problem how to target the problem of customer dropout. The main problem in those organizations was related to the customer retention and how to address the problem using other approaches than traditional statistics models to supported the development of actions in a business context using existing information in the organizations.

This thesis was developed to address those issues, which required the exploration how the problem was being addressed and how could be developed an approach to tackle them. This research was able to identify some gaps, namely the development of an dynamic approach using survival models, instead of considering only as a static perspective.

The main problem identified reviewing the state of the art was that considering the dropout of a customer based in its dropout risk, which does not consider that the customer dropout risk varies along time. In order to tackle those issue, we have explored perspective

with the use of "dynamic" algorithms, such as the survival analysis. Obtaining the risk of dropout that customers have over time, bringing a temporal perspective. Combining those idea with use of clusters to input the survival model, by using the output of one model as input from another, implementing the algorithms in pipeline Using this approach we were able to increase the accuracy of the survival models.

The advantage of this approach have been validated in two case studies. One was published in a JCR indexing and the other case was developed to confirm the results achieved. The outcomes of this thesis allows the development of targeted approaches taking into account the timing of when the dropout occurs, managers can use the resulting information for improvement of their retention strategies.

5.2 Publications

The results of this thesis have been published in scientific journals.

- Sobreiro, P., Martinho, D., Berrocal, J., & Alonso, J. (2021). Dropout Prediction: A Systematic Literature Review. CAPSI 2021 Proceedings. https://aisel.aisnet.org/capsi2021/18
- Sobreiro, P., Martinho, D. D. S., Alonso, J. G., & Berrocal, J. (2022). A SLR on Customer Dropout Prediction. IEEE Access, 10, 14529–14547. doi: 10.1109/ACCESS.2022.3146397 (JCR, 3.476)
- Sobreiro, P., Garcia-Alonso, J., Martinho, D., & Berrocal, J. (2022). Hybrid Random Forest Survival Model to Predict Customer Membership Dropout. Electronics, 11(20), Art. 20. doi: 10.3390/electronics11203328 (JCR, 2.690)
- Sobreiro, P., Martinho, D., Pratas, A., Garcia-Alonso, J., & Berrocal, J. (2019). Predicting High-Value Customers in a Portuguese Wine Company. Journal of Reviews on Global Economics, 9, 1732–1740. doi: 1929-7092.2019.08.155 (SJR, 0.227 - discontinued in Scopus in 2019)

5.3 Future work

Through the development of this thesis, and especially in production models, that could be employed in existing organizations and provide greater benefits, it would be interesting address the following problems:

- Customers who have greater risk of dropout should be targeted to provide a base for a better ROI in the retention strategies (Xie et al., 2009; Coussement & Van den Poel, 2008);
- Some assumptions that underlie the adoption of uplift metrics consider that customers with a higher risk of churning could not be the best targets, as suggested by Ascarza (2018)
- The retention strategies should be developed focused on customers with higher satisfaction, or its inclusion could be a reminder of the contractual agreement nearing an end and could lead to churn (Devriendt et al., 2019).

This requires implementations and compare the performance of using retention strategies based in the information retrieved with the proposed approaches, which entail the optimization of the retention efforts to customers that could give optimal returns and not customers that independently of the retention actions they would dropout anyway.

The business objective is to reduce customer churn, but it should be considered that customers that will churn and cannot be retained should be excluded from the retention actions reasoning that targeting them will be a waste of scarce resources (Devriendt et al., 2019).

However, customers who will churn but cannot be retained should be excluded from the countermeasures to avoid dropout, considering that targeting them may constitute a waste of scarce resources.

Additionally, should be considered the use of Big Data that allow the exploration of opportunities through the analysis of high volume data, using additional sources, such, structured data and social networks.

5.4 Final reflection

The idea for the development of this thesis started in 2015. During this period I have profound the knowledge in this research area. This allowed to improve my skills in data analysis and supporting the process of decision making.

Organizations have large amount of data that can be used to provide leads of customer dropout prediction. This area could improve the research being developed and increase their results.

The development of this thesis allowed to improve my skills to support research and teaching this subjects. This is very useful for my current research and teaching and will also in the future.

References

Agrawal, S., Das, A., Gaikwad, A., & Dhage, S. (2018, July). Customer Churn Prediction Modelling Based on Behavioural Patterns Analysis using Deep Learning. In *2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)* (pp. 1–6). Shah Alam: IEEE. doi: 10.1109/ICSCEE.2018.8538420

Akogul, S., & Erisoglu, M. (2016, 09). A comparison of information criteria in clustering based on mixture of multivariate normal distributions. *Mathematical and Computational Applications*, *21*(3), 34. Retrieved from https://www.mdpi.com/2297-8747/21/3/34 doi: 10.3390/mca21030034

Alboukaey, N., Joukhadar, A., & Ghneim, N. (2020, December). Dynamic behavior based churn prediction in mobile telecom. *Expert Systems with Applications*, *162*, 113779. doi: 10.1016/j.eswa.2020.113779

Al-Molhem, N. R., Rahal, Y., & Dakkak, M. (2019, December). Social network analysis in Telecom data. *Journal of Big Data*, *6*(1), 99. doi: 10.1186/s40537-019-0264-6

Amin, A., Anwar, S., Adnan, A., Nawaz, M., Alawfi, K., Hussain, A., & Huang, K. (2017, May). Customer churn prediction in the telecommunication sector using a rough set approach. *Neurocomputing*, *237*, 242–254. doi: 10.1016/j.neucom.2016.12.009

Amornvetchayakul, P., & Phumchusri, N. (2020, April). Customer Churn Prediction for a Software-as-a-Service Inventory Management Software Company: A Case Study in Thailand. In *2020 IEEE 7th International Conference on Industrial Engineering and Applications (ICIEA)* (pp. 514–518). Bangkok, Thailand: IEEE. doi: 10.1109/ICIEA49774.2020.9102099

Antipov, E., & Pokryshevskaya, E. (2010, June). Applying CHAID for logistic regression diagnostics and classification accuracy improvement. *Journal of Targeting, Measurement and Analysis for Marketing*, *18*(2), 109–117. doi: 10.1057/jt.2010.3

Ascarza, E. (2018, February). Retention Futility: Targeting High-Risk Customers Might be Ineffective. *Journal of Marketing Research*, *55*(1), 80–98. doi: 10.1509/jmr.16.0163

Ascarza, E., & Hardie, B. G. S. (2013, May). A joint model of usage and churn in contractual settings. *Marketing Science*, *32*(4), 570–590. doi: 10.1287/mksc.2013.0786

Athanassopoulos, A. D. (2000, Mar). Customer satisfaction cues to support market segmentation and explain switching behavior. *Journal of Business Research*, *47*(3), 191–207. (291 citations (Crossref) [2021-03-30]) doi: 10.1016/S0148-2963(98)00060-5

Azeem, M., Usman, M., & Fong, A. C. M. (2017, December). A churn prediction model for prepaid customers in telecom using fuzzy classifiers. *Telecommunication Systems*, *66*(4), 603–614. doi: 10.1007/s11235-017-0310-7

Ballings, M., & Van den Poel, D. (2012, December). Customer event history for churn prediction: How long is long enough? *Expert Systems with Applications*, *39*(18), 13517–13522. doi: 10.1016/j.eswa.2012.07.006

Ballings, M., Van den Poel, D., & Verhagen, E. (2012). Improving Customer Churn Prediction by Data Augmentation Using Pictorial Stimulus-Choice Data. In J. Casillas, F. J. Martínez-López, & J. M. Corchado Rodríguez (Eds.), *Management Intelligent Systems* (Vol. 171, pp. 217–226). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-642-30864-2_21

Bandara, W. M. C., Perera, A. S., & Alahakoon, D. (2013, 12). Churn prediction methodologies in the telecommunications sector: A survey. In *2013 international conference on advances in ict for emerging regions (icter)* (pp. 172–176). Colombo, Sri Lanka: IEEE. doi: 10.1109/icter.2013.6761174

Bansal, G., Anand, A., & Yadavalli, V. S. S. (2019, Feb). Predicting effective customer lifetime: an application of survival analysis for telecommunication industry. *Communications in Statistics - Theory and Methods*, *0*(0), 1–16. doi: 10.1080/03610926.2019.1570264

Benedek, G., Lubloy, A., & Vastag, G. (2014, February). The Importance of Social Embeddedness: Churn Models at Mobile Providers: Social Embeddedness and Churn. *Decision Sciences*, *45*(1), 175–201. doi: 10.1111/deci.12057

Benoit, D. F., & Van den Poel, D. (2012, October). Improving customer retention in financial services using kinship network information. *Expert Systems with Applications*, *39*(13), 11435–11442. doi: 10.1016/j.eswa.2012.04.016

Berry, M. J. A., & Linoff, G. (2004). *Data mining techniques: for marketing, sales, and customer relationship management* (2nd ed ed.). Indianapolis, Ind: Wiley Pub.

Bertsimas, D., Dunn, J., Gibson, E., & Orfanoudaki, A. (2022, Aug). Optimal survival trees. *Machine Learning*, *111*(8), 2951–3023. doi: 10.1007/s10994-021-06117-0

Bhattacharya, C. B. (1998, Dec). When customers are members: Customer retention in paid membership contexts. *Journal of the Academy of Marketing Science*, *26*(1), 31. doi: 10.1177/0092070398261004

Bland, J. M., & Altman, D. G. (1998, Dec). Survival probabilities (the kaplan-meier method). *BMJ (Clinical research ed.)*, *317*(7172), 1572. (00553)

Breiman, L. (2001, Oct). Random forests. *Machine Learning*, *45*(1), 5–32. doi: 10.1023/A:1010933404324

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). *Classification and regression trees*. New York: Routledge. doi: 10.1201/9781315139470

Burez, J., & Van den Poel, D. (2007, February). CRM at a pay-TV company: Using analytical models to reduce customer attrition by targeted marketing for subscription services. *Expert Systems with Applications*, *32*(2), 277–288. doi: 10.1016/j.eswa.2005.11.037

Burez, J., & Vandenpoel, D. (2008, July). Separating financial from commercial customer churn: A modeling step towards resolving the conflict between the sales and credit department. *Expert Systems with Applications*, *35*(1-2), 497–514. doi: 10.1016/j.eswa.2007.07.036

Burez, J., & Van den Poel, D. (2009, April). Handling class imbalance in customer churn prediction. *Expert Systems with Applications*, *36*(3), 4626–4636. doi: 10.1016/j.eswa.2008.05.027

Coussement, K., Benoit, D. F., & Van den Poel, D. (2010, March). Improved marketing decision making in a customer churn prediction context using generalized additive models. *Expert Systems with Applications*, *37*(3), 2132–2143. doi: 10.1016/j.eswa.2009.07.029

Coussement, K., & De Bock, K. W. (2013, September). Customer churn prediction in the online gambling industry: The beneficial effect of ensemble learning. *Journal of Business Research*, *66*(9), 1629–1636. doi: 10.1016/j.jbusres.2012.12.008

Coussement, K., Lessmann, S., & Verstraeten, G. (2017, March). A comparative analysis of data preparation algorithms for customer churn prediction: A case study in the telecommunication industry. *Decision Support Systems*, *95*, 27–36. doi: 10.1016/j.dss.2016.11.007

Coussement, K., & Van den Poel, D. (2008, January). Churn prediction in subscription services: An application of support vector machines while comparing two parameter-selection techniques. *Expert Systems with Applications*, *34*(1), 313–327. doi: 10.1016/j.eswa.2006.09.038

Coussement, K., & Van den Poel, D. (2009, April). Improving customer attrition prediction by integrating emotions from client/company interaction emails and evaluating multiple classifiers. *Expert Systems with Applications*, *36*(3, Part 2), 6127–6134. doi: 10.1016/j.eswa.2008.07.021

Dargan, S., Kumar, M., Ayyagari, M. R., & Kumar, G. (2020, Sep). A survey of deep learning and its applications: A new paradigm to machine learning. *Archives of Computational Methods in Engineering*, *27*(4), 1071–1092. (110 citations (Semantic Scholar/DOI) [2021-10-24]) doi: 10.1007/s11831-019-09344-w

Davidson-Pilon, C. (2021). *Camdavidsonpilon/lifelines*. Retrieved from https://github.com/CamDavidsonPilon/lifelines

De Bock, K. W., & Van den Poel, D. (2010). Ensembles of Probability Estimation Trees for Customer Churn Prediction. In N. García-Pedrajas, F. Herrera, C. Fyfe, J. M. Benítez, & M. Ali (Eds.), *Trends in Applied Intelligent Systems* (Vol. 6097, pp. 57–66). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-642-13025-0_7

De Bock, K. W., & Van den Poel, D. (2011, September). An empirical evaluation of rotation-based ensemble classifiers for customer churn prediction. *Expert Systems with Applications*, *38*(10), 12293–12301. doi: 10.1016/j.eswa.2011.04.007

De Bock, K. W., & Van den Poel, D. (2012, June). Reconciling performance and interpretability in customer churn prediction using ensemble learning based on generalized additive models. *Expert Systems with Applications*, *39*(8), 6816–6826. doi: 10.1016/j.eswa.2012.01.014

Dekker, G., Pechenizkiy, M., & Vleeshouwers, J. (2009). Predicting students drop out: A case study. In *Educational data mining 2009.* Retrieved from http://www.educationaldatamining.org/conferences/index.php/EDM/2009/paper/download/1467/1433 (00244)

Delen, D., Topuz, K., & Eryarsoy, E. (2020, March). Development of a Bayesian Belief Network-based DSS for predicting and understanding freshmen student attrition. *European Journal of Operational Research*, *281*(3), 575–587. doi: 10.1016/j.ejor.2019.03.037

Devriendt, F., Berrevoets, J., & Verbeke, W. (2019, December). Why you should stop predicting customer churn and start using uplift models. *Information Sciences*. doi: 10.1016/j.ins.2019.12.075

Dierkes, T., Bichler, M., & Krishnan, R. (2011, June). Estimating the effect of word of mouth on churn and cross-buying in the mobile phone market with Markov logic networks. *Decision Support Systems*, *51*(3), 361–371. doi: 10.1016/j.dss.2011.01.002

Domingos, P. (2012, 10). A few useful things to know about machine learning. *Commun. ACM*, *55*(10), 78–87. doi: 10.1145/2347736.2347755

Edward, M., & Sahadev, S. (2011a, 6 14). Role of switching costs in the service quality, perceived value, customer satisfaction and customer retention linkage. *Asia Pacific Journal of Marketing and Logistics*, *23*(3), 327–345. doi: 10.1108/13555851111143240

Edward, M., & Sahadev, S. (2011b, Jun). Role of switching costs in the service quality, perceived value, customer satisfaction and customer retention linkage. *Asia Pacific Journal of Marketing and Logistics*, *23*(3), 327–345. (00132) doi: 10.1108/13555851111143240

Ehrlinger, J. (2016a, December). ggrandomforests: Exploring random forest survival. *arXiv:1612.08974 [stat]*. Retrieved from http://arxiv.org/abs/1612.08974

Ehrlinger, J. (2016b, 12 28). ggrandomforests: Exploring random forest survival. *arXiv:1612.08974 [stat]*. Retrieved from http://arxiv.org/abs/1612.08974

Ekinci, Y., Uray, N., & Ülengin, F. (2014, Jan). A customer lifetime value model for the banking industry: a guide to marketing actions. *European Journal of Marketing*, *48*(3/4), 761–784. doi: 10.1108/EJM-12-2011-0714

Esteves, G., & Mendes-Moreira, J. (2016, September). Churn perdiction in the telecom business. In *2016 Eleventh International Conference on Digital Information Management (ICDIM)* (pp. 254–259). Porto, Portugal: IEEE. doi: 10.1109/ICDIM.2016.7829775

Fader, P. S., & Hardie, B. G. (2007, 1). How to project customer retention. *Journal of Interactive Marketing*, *21*(1), 76–90. doi: 10.1002/dir.20074

Fang, W., Li, X., Zhang, M., & Hu, M. (2015, Oct). Nature-inspired algorithms for real-world optimization problems. *Journal of Applied Mathematics*, *2015*, e359203. doi: 10.1155/2015/359203

Farquad, M., Ravi, V., & Raju, S. B. (2012). Analytical CRM in banking and finance using SVM: a modified active learning-based rule extraction approach. *International Journal of Electronic Customer Relationship Management*, *6*(1), 48. doi: 10.1504/IJECRM.2012.046470

Farquad, M. A. H., Ravi, V., & Raju, S. B. (2009). Data Mining Using Rules Extracted from SVM: An Application to Churn Prediction in Bank Credit Cards. In D. Hutchison et al. (Eds.), *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing* (Vol. 5908, pp. 390–397). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-642-10646-0_47

Farquad, M. A. H., Ravi, V., & Raju, S. B. (2014, June). Churn prediction using comprehensible support vector machine: An analytical CRM application. *Applied Soft Computing*, *19*, 31–40. doi: 10.1016/j.asoc.2014.01.031

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996, Mar). From data mining to knowledge discovery in databases. *AI Magazine*, *17*(33), 37–37. doi: 10.1609/aimag.v17i3.1230

Fornell, C., & Wernerfelt, B. (1987). Defensive marketing strategy by customer complaint management: A theoretical analysis. *Journal of Marketing Research*, *24*(4), 337–346. (01918) doi: 10.2307/3151381

Fotso, S., & et al. (2019). *Pysurvival: Open source package for survival analysis modeling.* Retrieved from https://www.pysurvival.io/

Friedman, J. H. (1994). An overview of predictive learning and function approximation. In V. Cherkassky, J. H. Friedman, & H. Wechsler (Eds.), *From statistics to neural networks* (p. 1–61). Springer. doi: 10.1007/978-3-642-79119-2_1

García, D. L., Nebot, A., & Vellido, A. (2017, June). Intelligent data analysis approaches to churn as a business problem: a survey. *Knowledge and Information Systems*, *51*(3), 719–774. doi: 10.1007/s10115-016-0995-z

Glady, N., Baesens, B., & Croux, C. (2009, August). Modeling churn using customer lifetime value. *European Journal of Operational Research*, *197*(1), 402–411. doi: 10.1016/j.ejor.2008.06.027

Glady, N., Lemmens, A., & Croux, C. (2015, Mar). Unveiling the relationship between the transaction timing, spending and dropout behavior of customers. *International Journal of Research in Marketing*, *32*(1), 78–93. (12 citations (Semantic Scholar/DOI) [2022-11-29]) doi: 10.1016/j.ijresmar.2014.09.005

Gupta, S., Hanssens, D., Hardie, B., Kahn, W., Kumar, V., Lin, N., ... Sriram, S. (2006, 11 1). Modeling customer lifetime value. *Journal of Service Research*, *9*(2), 139–155. doi: 10.1177/1094670506293810

Gök, M., Özyer, T., & Jida, J. (2015). A Case Study for the Churn Prediction in Turksat Internet Service Subscription. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015 - ASONAM '15* (pp. 1220–1224). Paris, France: ACM Press. doi: 10.1145/2808797.2808821

Gür Ali, O., & Arıtürk, U. (2014, December). Dynamic churn prediction framework with more effective use of rare event data: The case of private banking. *Expert Systems with Applications*, *41*(17), 7889–7903. doi: 10.1016/j.eswa.2014.06.018

Han, J., & Kamber, M. (2006). *Data mining: concepts and techniques* (2nd ed ed.). Amsterdam ; Boston : San Francisco, CA: Elsevier ; Morgan Kaufmann.

Han, J., Kamber, M., & Pei, J. (2012). *Data mining: concepts and techniques* (3. ed ed.). Elsevier; Morgan Kaufmann.

He, B., Shi, Y., Wan, Q., & Zhao, X. (2014, January). Prediction of Customer Attrition of Commercial Banks based on SVM Model. *Procedia Computer Science*, *31*, 423–430. doi: 10.1016/j.procs.2014.05.286

Huang, B. Q., Kechadi, M.-T., & Buckley, B. (2009). Customer Churn Prediction for Broadband Internet Services. In T. B. Pedersen, M. K. Mohania, & A. M. Tjoa (Eds.), *Data Warehousing and Knowledge Discovery* (Vol. 5691, pp. 229–243). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-642-03730-6_19

Hung, S.-Y., Yen, D. C., & Wang, H.-Y. (2006, October). Applying data mining to telecom churn management. *Expert Systems with Applications*, *31*(3), 515–524. doi: 10.1016/j.eswa.2005.09.080

Hutchison, D., Kanade, T., Kittler, J., Kleinberg, J. M., Mattern, F., Mitchell, J. C., ... Raju, S. B. (2010). Rule Extraction from Support Vector Machine Using Modified Active Learning Based Approach: An Application to CRM. In R. Setchi, I. Jordanov, R. J. Howlett, & L. C. Jain (Eds.), *Knowledge-Based and Intelligent Information and Engineering Systems* (Vol. 6276, pp. 461–470). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-642-15387-7_50

Idris, A., Khan, A., & Lee, Y. S. (2013, October). Intelligent churn prediction in telecom: employing mRMR feature selection and RotBoost based ensemble classification. *Applied Intelligence*, *39*(3), 659–672. doi: 10.1007/s10489-013-0440-x

Ishwaran, H., Kogalur, U. B., Blackstone, E. H., & Lauer, M. S. (2008, Sep). Random survival forests. *The Annals of Applied Statistics*, *2*(3), 841–860. (arXiv: 0811.1645) doi: 10.1214/08-AOAS169

Jafari-Marandi, R., Denton, J., Idris, A., Smith, B. K., & Keramati, A. (2020, September). Optimum profit-driven churn decision making: innovative artificial neural networks in telecom industry. *Neural Computing and Applications*, *32*(18), 14929–14962. doi: 10.1007/s00521-020-04850-6

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (Vol. 103). New York, NY: Springer New York. Retrieved from http://link.springer.com/10.1007/978-1-4614-7138-7 doi: 10.1007/978-1-4614-7138-7

Jerath, K., Fader, P. S., & Hardie, B. G. S. (2011, Sep). New perspectives on customer "death" using a generalization of the pareto/nbd model. *Marketing Science*, *30*(5), 866–880. doi: 10.1287/mksc.1110.0654

Jiang, M., Chu, N., & Bi, X. M. (2014, July). Research on Customers Churn Prediction Model Based on Logistic. *Advanced Materials Research*, *989-994*, 1517–1521. doi: 10.4028/www.scientific.net/AMR.989-994.1517

Jones, M. A., Mothersbaugh, D. L., & Beatty, S. E. (2000, Jun). Switching barriers and repurchase intentions in services. *Journal of Retailing*, *76*(2), 259–274. doi: 10.1016/S0022-4359(00)00024-5

Kaya, E., Dong, X., Suhara, Y., Balcisoy, S., Bozkaya, B., & Pentland, A. S. (2018, December). Behavioral attributes and financial churn prediction. *EPJ Data Science*, *7*(1), 41. doi: 10.1140/epjds/s13688-018-0165-5

Kelleher, J. D., Namee, B. M., & D'Arcy, A. (2015). *Fundamentals of machine learning for predictive data analytics: Algorithms, worked examples, and case studies* (1edition ed.). Cambridge, Massachusetts: The MIT Press.

Keramati, A., Jafari-Marandi, R., Aliannejadi, M., Ahmadian, I., Mozaffari, M., & Abbasi, U. (2014, November). Improved churn prediction in telecommunication industry using data mining techniques. *Applied Soft Computing*, *24*, 994–1012. doi: 10.1016/j.asoc.2014.08.041

Kianmehr, K., & Alhajj, R. (2009, April). Calling communities analysis and identification using machine learning techniques. *Expert Systems with Applications*, *36*(3, Part 2), 6218–6226. doi: 10.1016/j.eswa.2008.07.072

Kianmehr, K., & Alhajj, R. (2011a). A fuzzy prediction model for calling communities. *International Journal of Networking and Virtual Organisations*, *8*(1/2), 75. doi: 10.1504/IJNVO.2011.037162

Kianmehr, K., & Alhajj, R. (2011b). A fuzzy prediction model for calling communities. *International Journal of Networking and Virtual Organisations*, *8*(1/2), 75. doi: 10.1504/IJNVO.2011.037162

Kim, J. W., Lee, B. H., Shaw, M. J., Chang, H.-L., & Nelson, M. (2001, Mar). Application of decision-tree induction techniques to personalized advertisements on internet

storefronts. *International Journal of Electronic Commerce*, *5*(3), 45–62. (00204) doi: 10.1080/10864415.2001.11044215

Kotsiantis, S., Pierrakeas, C., & Pintelas, P. (2004, May). Predicting students' performance in distance learning using machine learning techniques. *Applied Artificial Intelligence*, *18*(5), 411–426. doi: 10.1080/08839510490442058

Lee, K. C., & Jo, N. Y. (2010). Bayesian Network Approach to Predict Mobile Churn Motivations: Emphasis on General Bayesian Network, Markov Blanket, and What-If Simulation. In T.-h. Kim, Y.-h. Lee, B.-H. Kang, & D. Slezak (Eds.), *Future Generation Information Technology* (Vol. 6485, pp. 304–313). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-642-17569-5_30

Liao, K.-H., & Chueh, H.-E. (2011). Applying Fuzzy Data Mining to Telecom Churn Management. In R. Chen (Ed.), *Intelligent Computing and Information Science* (Vol. 134, pp. 259–264). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-642-18129-0_41

Liu, X., Xie, M., Wen, X., Chen, R., Ge, Y., Duffield, N., & Wang, N. (2020, April). Micro- and macro-level churn analysis of large-scale mobile games. *Knowledge and Information Systems*, *62*(4), 1465–1496. doi: 10.1007/s10115-019-01394-7

Martono, N. P., Kanamori, K., & Ohwada, H. (2014). Utilizing Customers' Purchase and Contract Renewal Details to Predict Defection in the Cloud Software Industry. In Y. S. Kim, B. H. Kang, & D. Richards (Eds.), *Knowledge Management and Acquisition for Smart Systems and Services* (Vol. 8863, pp. 138–149). Cham: Springer International Publishing. doi: 10.1007/978-3-319-13332-4_12

Mitrovic, S., Singh, G., Baesens, B., Lemahieu, W., & de Weerdt, J. (2017, October). Scalable RFM-enriched Representation Learning for Churn Prediction. In *2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA)* (pp. 79–88). Tokyo, Japan: IEEE. doi: 10.1109/DSAA.2017.42

Mitrović, S., Baesens, B., Lemahieu, W., & De Weerdt, J. (2018, June). On the operational efficiency of different feature types for telco Churn prediction. *European Journal of Operational Research*, *267*(3), 1141–1155. doi: 10.1016/j.ejor.2017.12.015

Moeyersoms, J., & Martens, D. (2015, April). Including high-cardinality attributes in predictive models: A case study in churn prediction in the energy sector. *Decision Support Systems*, *72*, 72–81. doi: 10.1016/j.dss.2015.02.007

Mohanty, R., & Naga Ratna Sree, C. (2018). Churn and Non-churn of Customers in Banking Sector Using Extreme Learning Machine. In V. Bhateja, J. M. R. Tavares, B. P. Rani, V. K. Prasad, & K. S. Raju (Eds.), *Proceedings of the Second International Conference on Computational Intelligence and Informatics* (Vol. 712, pp. 51–58). Singapore: Springer Singapore. doi: 10.1007/978-981-10-8228-3_6

Mohanty, R., & Rani, K. J. (2015, December). Application of Computational Intelligence to Predict Churn and Non-Churn of Customers in Indian Telecommunication. In *2015 International Conference on Computational Intelligence and Communication Networks (CICN)* (pp. 598–603). Jabalpur, India: IEEE. doi: 10.1109/CICN.2015.123

Namratha, M., & Prajwala, T. (2012). A comprehensive overview of clustering algorithms in pattern recognition. *IOSR Journal of Computer Engineering*, *4*(6), 23–30. doi: 10.9790/0661-0462330

Neslin, S. A., Gupta, S., Kamakura, W., Lu, J., & Mason, C. H. (2006, May). Defection detection: Measuring and understanding the predictive accuracy of customer churn models. *Journal of Marketing Research*, *43*(2), 204–211. doi: 10.1509/jmkr.43.2.204

Nie, G., Rowe, W., Zhang, L., Tian, Y., & Shi, Y. (2011, November). Credit card churn forecasting by logistic regression and decision tree. *Expert Systems with Applications*, *38*(12), 15273–15285. doi: 10.1016/j.eswa.2011.06.028

Ongsulee, P., Chotchaung, V., Bamrungsi, E., & Rodcheewit, T. (2018, Nov). Big data, predictive analytics and machine learning. In *2018 16th international conference on ict and knowledge engineering (ict ke)* (p. 1–6). doi: 10.1109/ICTKE.2018.8612393

Pan, R., Yang, Q., Ling, C., & Yin, J. (2007, Jan). Extracting actionable knowledge from decision trees. *IEEE Transactions on Knowledge & Data Engineering*, *19*(01), 43–56. (00097) doi: 10.1109/TKDE.2007.10

Perianez, A., Saas, A., Guitart, A., & Magne, C. (2016, October). Churn Prediction in Mobile Social Games: Towards a Complete Assessment Using Survival Ensembles. In *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)* (pp. 564–573). Montreal, QC, Canada: IEEE. doi: 10.1109/DSAA.2016.84

Prasasti, N., & Ohwada, H. (2014, May). Applicability of machine-learning techniques in predicting customer defection. In *2014 International Symposium on Technology Management and Emerging Technologies* (pp. 157–162). Bandung, Indonesia: IEEE. doi: 10.1109/ISTMET.2014.6936498

Quinlan, J. R. (1986, Mar). Induction of decision trees. *Machine Learning*, *1*(1), 81–106. doi: 10.1007/BF00116251

Radosavljevik, D., & van der Putten, P. (2013). Preventing Churn in Telecommunications: The Forgotten Network. In D. Hutchison et al. (Eds.), *Advances in Intelligent Data Analysis XII* (Vol. 8207, pp. 357–368). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-642-41398-8_31

Reichheld, F. F. (1996a, 3 1). Learning from customer defections. *Harvard Business Review*(March–April 1996). Retrieved from https://hbr.org/1996/03/learning-from-customer-defections

Reichheld, F. F. (1996b, 3 1). Learning from customer defections. *Harvard Business Review*(March–April 1996). Retrieved from https://hbr.org/1996/03/learning-from-customer-defections

Risselada, H., Verhoef, P. C., & Bijmolt, T. H. A. (2010, August). Staying Power of Churn Prediction Models. *Journal of Interactive Marketing*, *24*(3), 198–208. doi: 10.1016/j.intmar.2010.04.002

Routh, P., Roy, A., & Meyer, J. (2020, August). Estimating customer churn under competing risks. *Journal of the Operational Research Society*, 1–18. doi: 10.1080/01605682.2020.1776166

Saravanan, M., & Vijay Raajaa, G. S. (2012). A Graph-Based Churn Prediction Model for Mobile Telecom Networks. In D. Hutchison et al. (Eds.), *Advanced Data Mining and Applications* (Vol. 7713, pp. 367–382). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-642-35527-1_31

Schaeffer, S. E., & Rodriguez Sanchez, S. V. (2020, January). Forecasting client retention — A machine-learning approach. *Journal of Retailing and Consumer Services*, *52*, 101918. doi: 10.1016/j.jretconser.2019.101918

Schober, P., & Vetter, T. R. (2018a, 9). Survival analysis and interpretation of time-to-event data: The tortoise and the hare. *Anesthesia and Analgesia*, *127*(3), 792–798. doi: 10.1213/ANE.0000000000003653

Schober, P., & Vetter, T. R. (2018b, 9). Survival analysis and interpretation of time-to-event data: The tortoise and the hare. *Anesthesia and Analgesia*, *127*(3), 792–798. doi: 10.1213/ANE.0000000000003653

Schwartz, H., Sap, M., Kern, M., Eichstaedt, J., Kapelner, A., Agrawal, M., ... Ungar, L. (2015, 11). Predicting individual well-being through the language of social media. In (pp. 516–527). WORLD SCIENTIFIC. Retrieved from https://www.worldscientific.com/doi/abs/10.1142/9789814749411_0047

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, *6*(2), 461–464. Retrieved from https://www.jstor.org/stable/2958889

Scrucca, L., Fop, M., Murphy, T., & Raftery, A. (2016). mclust 5: Clustering, classification and density estimation using gaussian finite mixture models. *The R Journal*, *8*(1), 289. Retrieved from https://journal.r-project.org/archive/2016/RJ-2016-021/index.html doi: 10.32614/RJ-2016-021

Semrl, J., & Matei, A. (2017, October). Churn prediction model for effective gym customer retention. In *2017 International Conference on Behavioral, Economic, Socio-cultural Computing (BESC)* (pp. 1–3). Krakow: IEEE. doi: 10.1109/BESC.2017.8256385

Shao, H., Zheng, G., & An, F. (2008). Construction of Bayesian Classifiers with GA for Predicting Customer Retention. In *2008 Fourth International Conference on Natural Computation* (pp. 181–185). Jinan, Shandong, China: IEEE. doi: 10.1109/ICNC.2008.724

Sheth, J. N., Mittal, B., & Newman, B. (1998). *Customer behavior: Consumer behavior and beyond* (1edition ed.). Fort Worth, TX: South-Western College Pub.

Shirazi, F., & Mohammadi, M. (2019, October). A big data analytics model for customer churn prediction in the retiree segment. *International Journal of Information Management*, *48*, 238–253. doi: 10.1016/j.ijinfomgt.2018.10.005

Singer, J. D., & Willett, J. B. (1993, Jun). It's about time: Using discrete-time survival analysis to study duration and the timing of events. *Journal of Educational Statistics*, *18*(2), 155–195. (00865) doi: 10.3102/10769986018002155

Sivasankar, E., & Vijaya, J. (2019, November). Hybrid PPFCM-ANN model: an efficient system for customer churn prediction through probabilistic possibilistic fuzzy clustering and artificial neural network. *Neural Computing and Applications*, *31*(11), 7181–7200. doi: 10.1007/s00521-018-3548-4

Sobreiro, P., Garcia-Alonso, J., Martinho, D., & Berrocal, J. (2022, Jan). Hybrid random forest survival model to predict customer membership dropout. *Electronics*, *11*(2020), 3328. doi: 10.3390/electronics11203328

Sobreiro, P., Martinho, D., Berrocal, J., & Alonso, J. (2021, Oct). Dropout prediction: A systematic literature review. *CAPSI 2021 Proceedings*. Retrieved from https://aisel.aisnet.org/capsi2021/18

Sobreiro, P., Martinho, D. D. S., Alonso, J. G., & Berrocal, J. (2022). A slr on customer dropout prediction. *IEEE Access*, *10*, 14529–14547. (0 citations (Semantic Scholar/-DOI) [2022-02-10]) doi: 10.1109/ACCESS.2022.3146397

Stensrud, M. J., & Hernán, M. A. (2020, Apr). Why test for proportional hazards? *JAMA*, *323*(14), 1401–1402. doi: 10.1001/jama.2020.1267

Thorndike, R. L. (1953, Dec). Who belongs in the family? *Psychometrika*, *18*(4), 267–276. doi: 10.1007/BF02289263

Tsai, C.-F., & Chen, M.-Y. (2010, March). Variable selection by association rules for customer churn prediction of multimedia on demand. *Expert Systems with Applications*, *37*(3), 2006–2015. doi: 10.1016/j.eswa.2009.06.076

Ullah, I., Hussain, H., Ali, I., & Liaquat, A. (2019, July). Churn Prediction in Banking System using K-Means, LOF, and CBLOF. In *2019 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)* (pp. 1–6). Swat, Pakistan: IEEE. doi: 10.1109/ICECCE47252.2019.8940667

Van den Poel, D., & Larivière, B. (2004, Aug). Customer attrition analysis for financial services using proportional hazard models. *European Journal of Operational Research*, *157*(1), 196–217. doi: 10.1016/S0377-2217(03)00069-9

Verbeke, W., Dejaeger, K., Martens, D., Hur, J., & Baesens, B. (2012a, April). New insights into churn prediction in the telecommunication sector: A profit driven data mining approach. *European Journal of Operational Research*, *218*(1), 211–229. doi: 10.1016/j.ejor.2011.09.031

Verbeke, W., Dejaeger, K., Martens, D., Hur, J., & Baesens, B. (2012b, Apr). New insights into churn prediction in the telecommunication sector: A profit driven data mining approach. *European Journal of Operational Research*, *218*(1), 211–229. doi: 10.1016/j.ejor.2011.09.031

Verbeke, W., Martens, D., & Baesens, B. (2014, January). Social network analysis for customer churn prediction. *Applied Soft Computing*, *14*, 431–446. doi: 10.1016/j.asoc .2013.09.017

Verbeke, W., Martens, D., Mues, C., & Baesens, B. (2011, March). Building comprehensible customer churn prediction models with advanced rule induction techniques. *Expert Systems with Applications*, *38*(3), 2354–2364. doi: 10.1016/j.eswa.2010.08.023

Verbraken, T., Verbeke, W., & Baesens, B. (2014, January). Profit optimizing customer churn prediction with Bayesian network classifiers. *Intelligent Data Analysis*, *18*(1), 3–24. doi: 10.3233/IDA-130625

Vijaya, J., & Sivasankar, E. (2019a, September). An efficient system for customer churn prediction through particle swarm optimization based feature selection model with simulated annealing. *Cluster Computing*, *22*(S5), 10757–10768. doi: 10.1007/ s10586-017-1172-1

Vijaya, J., & Sivasankar, E. (2019b, Sep). An efficient system for customer churn prediction through particle swarm optimization based feature selection model with simulated annealing. *Cluster Computing*, *22*(S5), 10757–10768. (10 citations (Semantic Scholar/-DOI) [2021-03-26]) doi: 10.1007/s10586-017-1172-1

Vijaya, J., Sivasankar, E., & Gayathri, S. (2019). Fuzzy Clustering with Ensemble Classification Techniques to Improve the Customer Churn Prediction in Telecommunication Sector. In J. Kalita, V. E. Balas, S. Borah, & R. Pradhan (Eds.), *Recent Developments in Machine Learning and Data Analytics* (Vol. 740, pp. 261–274). Singapore: Springer Singapore. doi: 10.1007/978-981-13-1280-9_25

Wai-Ho Au, Chan, K., & Xin Yao. (2003, December). A novel evolutionary data mining algorithm with applications to churn prediction. *IEEE Transactions on Evolutionary Computation*, *7*(6), 532–545. doi: 10.1109/TEVC.2003.819264

Wang, P., Li, Y., & Reddy, C. (2017, 08). Machine learning for survival analysis: A survey. *arXiv:1708.04649 [cs, stat]*. Retrieved from http://arxiv.org/abs/1708.04649

Wang, Y., & Xiao, J. (2011, September). Transfer Ensemble Model for Customer Churn Prediction with Imbalanced Class Distribution. In *2011 International Conference of Information Technology, Computer Engineering and Management Sciences* (pp. 177–181). Nanjing, Jiangsu, China: IEEE. doi: 10.1109/ICM.2011.397

Wei, C.-P., & Chiu, I.-T. (2002, August). Turning telecommunications call details to churn prediction: a data mining approach. *Expert Systems with Applications*, *23*(2), 103–112. doi: 10.1016/S0957-4174(02)00030-1

Xia, G.-e., & Jin, W.-d. (2008, January). Model of Customer Churn Prediction on Support Vector Machine. *Systems Engineering - Theory & Practice*, *28*(1), 71–77. doi: 10.1016/S1874-8651(09)60003-X

Xiao, J., Xiao, Y., Huang, A., Liu, D., & Wang, S. (2015, April). Feature-selection-based dynamic transfer ensemble model for customer churn prediction. *Knowledge and Information Systems*, *43*(1), 29–51. doi: 10.1007/s10115-013-0722-y

Xiao, J., Xie, L., He, C., & Jiang, X. (2012, February). Dynamic classifier ensemble model for customer classification with imbalanced class distribution. *Expert Systems with Applications*, *39*(3), 3668–3675. doi: 10.1016/j.eswa.2011.09.059

Xie, Y., Li, X., Ngai, E. W. T., & Ying, W. (2009, April). Customer churn prediction using improved balanced random forests. *Expert Systems with Applications*, *36*(3, Part 1), 5445–5449. doi: 10.1016/j.eswa.2008.06.121

Xue, W., Sun, Y., Bandyopadhyay, S., & Cheng, D. (2021, Jun). Measuring customer equity in noncontractual settings using a diffusion model: An empirical study of mobile payments aggregator. *Journal of Theoretical and Applied Electronic Commerce Research*, *16*(33), 409–431. doi: 10.3390/jtaer16030026

Zhang, X., Zhu, J., Xu, S., & Wan, Y. (2012, April). Predicting customer churn through interpersonal influence. *Knowledge-Based Systems*, *28*, 97–104. doi: 10.1016/j.knosys .2011.12.005

Zhou, Y., & McArdle, J. J. (2015, Sep). Rationale and applications of survival tree and survival ensemble methods. *Psychometrika*, *80*(3), 811–833. (00017) doi: 10.1007/ s11336-014-9413-1

Zhu, B., Baesens, B., Backiel, A., & vanden Broucke, S. K. L. M. (2018, January). Benchmarking sampling techniques for imbalance learning in churn prediction. *Journal of the Operational Research Society*, *69*(1), 49–65. doi: 10.1057/s41274-016-0176-1

Óskarsdóttir, M., Baesens, B., & Vanthienen, J. (2018, March). Profit-Based Model Selection for Customer Retention Using Individual Customer Lifetime Values. *Big Data*, *6*(1), 53–65. doi: 10.1089/big.2018.0015

Appendix A

Data analysis sport club

A.1 Introduction

```
knitr::opts_chunk$set(cache = FALSE)
knitr::opts_chunk$set(echo = TRUE)
# Use cache = TRUE if you want to speed up compilation
# set path
# get rmarkdown directory
caminho <- getwd()
# set working directory
setwd(caminho)
print(caminho)
```

```
## [1] "C:/nuvem/Dropbox/doutoramento/tese/3.case_study/customerDropoutMembership/article"
```

```
# A function to allow for showing some of the inline code
rinline <- function(code) {
  html <- '<code  class = "r">``` `r CODE` ```</code>'
  sub("CODE", code, html)
}
```

A.2 Reticulate configuration

Reticulate allows the interoperability between Python and R

```
# load essential libraries
library(dplyr)
library(dlookr)
library(ggplot2)
library(reticulate)
#Replace by your environment, usually which python solves the problem
#conda env list also is a good option
#set environment first then call reticulate library
path_python_windows <- "C:\\Users\\sobre\\AppData\\Local\\r-miniconda\\envs\\rsurvival\\python.exe"
path_python_linux <- "/home/sobreiro/miniconda3/envs/survival/bin/python"
switch(Sys.info()[["sysname"]],
       Windows = {
            Sys.setenv(RETICULATE_PYTHON = path_python_windows)
            #call reticulate
            library(reticulate)
            #activate environment
            use_condaenv("rsurvival", required = TRUE)
       },
       Linux = {
            Sys.setenv(RETICULATE_PYTHON = path_python_linux)
            library(reticulate)
            use_condaenv("survival", required = TRUE)
       }
)
```

A.3 Dataset

In this case, data from a sport club was analysed. The information retrieved was: Age of the participants in years; Sex (F-female, M-male); Marital status (Single, Married and other); Monthly fee member; Total payed amound until the data was retrieved; Match attendance and Months since last payment.

Dropout is a binary value where one represent churn and zero not churn. The dropout happens when a member does not have a payment. Considering the sport club policies all the customers with payments less than 24 months where considered active
The variables extracted from the software correspond to the time interval of becoming a customer until the end of observation (censoring on 31 Maio 2019) or the end of the customer relationship

(dropout). The survival time in the dataset is represented by the number of years the customer begin affilated. We extracted records of 25316 customers (male n=17246, female n=8070) from a sport club; data corresponded to the time period between October 1, 1944 and May 31, 2019.

```
library(stargazer)
library(readxl)
library(dplyr)
library(visdat)

df_members_sport <- read_excel("../data/membershipData.xlsx")



# rename column labels
names(df_members_sport) <- c("num_socio", "dt_inscription", "year_inscription",
                             "birth_date", "age", "sex", "marital_status",
                             "category", "monthly_fee", "occupation",
                             "zip_code", "dt_last_invoice", "dt_last_payment",
                             "total_amount", "total_matches", "season_matches",
                             "days_since_last_payment",
                             "months_since_last_payment", "dropout",
                             "years_membership", "stadium_access",
                             "quart_stadium_entries", "inscription_month")

names(df_members_sport)
```

```
##  [1] "num_socio"                "dt_inscription"
##  [3] "year_inscription"         "birth_date"
##  [5] "age"                      "sex"
##  [7] "marital_status"           "category"
##  [9] "monthly_fee"              "occupation"
## [11] "zip_code"                 "dt_last_invoice"
## [13] "dt_last_payment"          "total_amount"
## [15] "total_matches"            "season_matches"
## [17] "days_since_last_payment"  "months_since_last_payment"
## [19] "dropout"                  "years_membership"
## [21] "stadium_access"           "quart_stadium_entries"
## [23] "inscription_month"
```

```
# select relevant variables
df_members_sport <- df_members_sport %>%
select(age, sex, marital_status, monthly_fee, total_amount, total_matches,
       season_matches, months_since_last_payment, dropout, years_membership,
       stadium_access, quart_stadium_entries,inscription_month)

str(df_members_sport)
```

```
## tibble [25,316 x 13] (S3: tbl_df/tbl/data.frame)
##  $ age                      : num [1:25316] 83 88 73 97 97 91 88 95 88 78 ...
##  $ sex                      : chr [1:25316] "M" "M" "M" "M" ...
##  $ marital_status           : chr [1:25316] "casado" "solteiro" "nao definido" "casado" ...
##  $ monthly_fee              : num [1:25316] 10 10 10 5 10 5 5 5 10 10 ...
##  $ total_amount             : num [1:25316] 1906 1906 1553 790 1466 ...
##  $ total_matches            : num [1:25316] 0 0 0 0 0 20 74 0 154 0 ...
##  $ season_matches           : num [1:25316] 0 0 0 0 0 0 0 0 6 0 ...
##  $ months_since_last_payment: num [1:25316] 3 3 36 8 35 4 41 40 4 2 ...
##  $ dropout                  : num [1:25316] 0 0 1 0 1 0 1 1 0 0 ...
##  $ years_membership         : num [1:25316] 74 74 73 73 73 73 73 73 73 72 ...
##  $ stadium_access           : num [1:25316] 0 0 0 0 0 1 1 0 1 0 ...
##  $ quart_stadium_entries    : chr [1:25316] "ate 1" "ate 1" "ate 1" "ate 1" ...
##  $ inscription_month        : num [1:25316] 10 10 8 9 9 12 1 1 2 4 ...
```

```
vis_dat(df_members_sport) #check
```

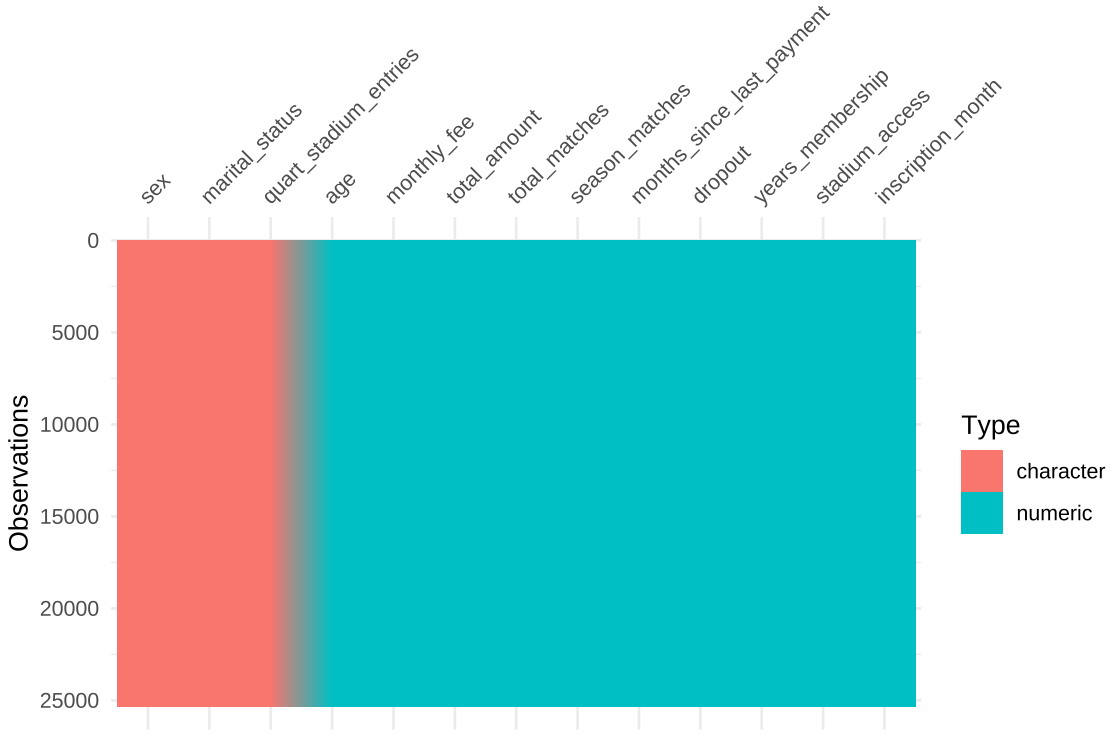


Table B.1 shows data's summary statistics. The average age is 27.3 ± 20.1, the members have an attendance of 27 ± 45.8 with a membership of 11 ± 10.9 years.

```
library(stargazer)
df_summary <- summary(df_members_sport)
stargazer(df_members_sport,
          title = "Summary statistics",
          label = "tab1cars",
          table.placement = "h",
          header = FALSE,
          summary = TRUE,
          summary.stat = c("min", "p25", "median", "p75", "max", "median", "sd")
         )
```

```
library(gtsummary)
library(kableExtra)
library(labelled)

var_label(df_members_sport$age) <- "Age in years"
var_label(df_members_sport$sex) <- "Male or female"
var_label(df_members_sport$marital_status) = "Single, married and other."
```

Table A.1: Summary statistics of features used

Characteristic	**N = 25,316**
Age in years, Mean (SD)	27 (20)
Male or female, %	
F	32%
M	68%
Single, married and other., %	
casado	20%
nao definido	30%
outro	2.0%
solteiro	48%
monthly_fee, %	
0	¡0.1%
1	32%
2.5	28%
5	3.4%
6	12%
10	24%
total_amount, Mean (SD)	316 (494)
total_matches, Mean (SD)	27 (46)
season_matches, Mean (SD)	2.2 (4.1)
months_since_last_payment, Mean (SD)	19 (32)
dropout, %	22%
years_membership, Mean (SD)	11 (11)
stadium_access, %	40%
quart_stadium_entries, %	
1 a 21	10%
21 a 56	9.8%
56 a 105	10.0%
ate 1	60%
mais 105	10.0%
inscription_month, Mean (SD)	6.9 (3.4)

```
tbl <- df_members_sport %>%
    tbl_summary(statistic = list(all_continuous() ~ "{mean} ({sd})",
                                 all_categorical() ~ "{p}%"),
                type = list(age ~ "continuous")) %>%
    add_stat_label()


as_kable_extra(tbl, booktabs = T,
               caption = "Summary statistics of features used")
```

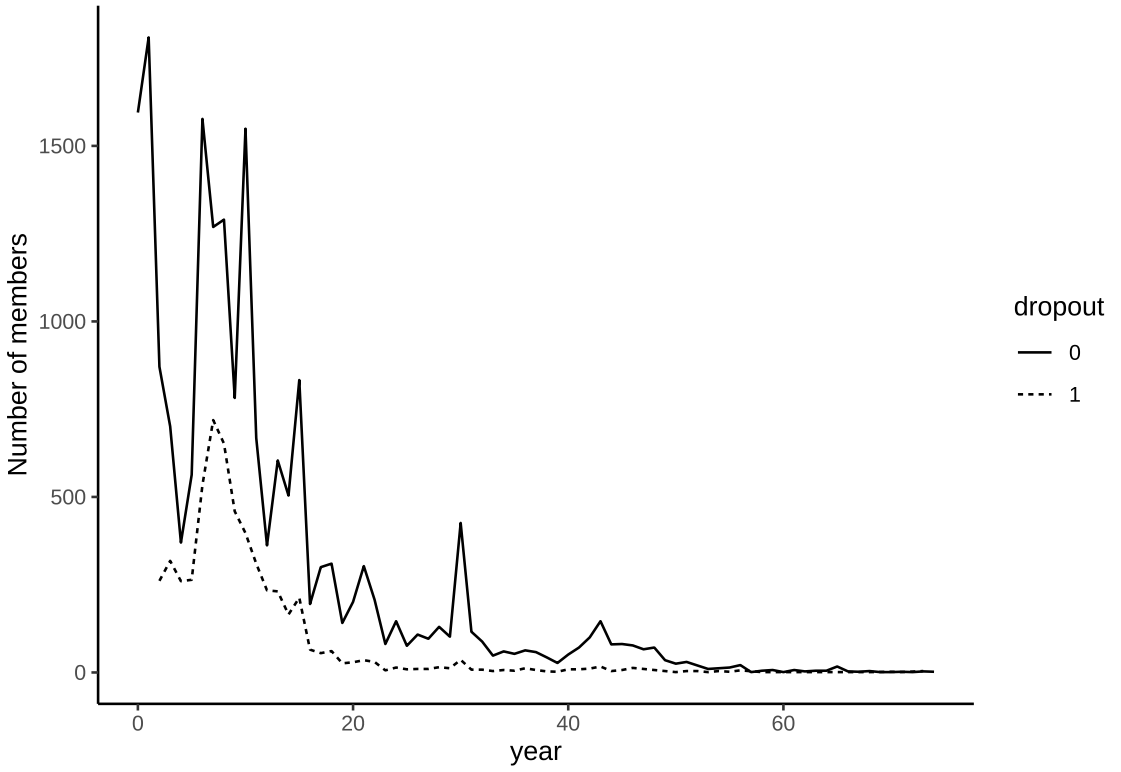


FIGURE A.1: Number of members by year

Figure A.1 shows the distribution of the dropout considering the number of years of membership.

```
members_year <- df_members_sport %>%
  select(years_membership, dropout) %>%
  group_by(years_membership, dropout) %>%
  summarize(count = n())

members_year$dropout <- factor(members_year$dropout)
g <- ggplot(data = members_year,
                mapping = aes(x = years_membership, y = count,
                              linetype = dropout))
g <- g + geom_line() + labs(x = "year", y = "Number of members") +
    theme_classic()
g
```

A.3.1 Model construction

The categorical variables sex, marital_status and quart_stadium_entries where converted to dummy variables.

The random survival forest was developed using the package PySurvival (Fotso et al. 2019). The most relevant variables predicting the dropout are analysed using the log-rank test. The metric variables are transformed to categorical using the quartiles to provide a statistical comparison of groups. The survival analysis was conducted using the package Lifelines (Davidson-Pilon 2021).

PySurvival is an open source python package for Survival Analysis modeling - the modeling concept used to analyze or predict when an event is likely to happen. It is built upon the most commonly used machine learning packages such NumPy, SciPy and PyTorch. PySurvival is compatible with Python 2.7-3.7

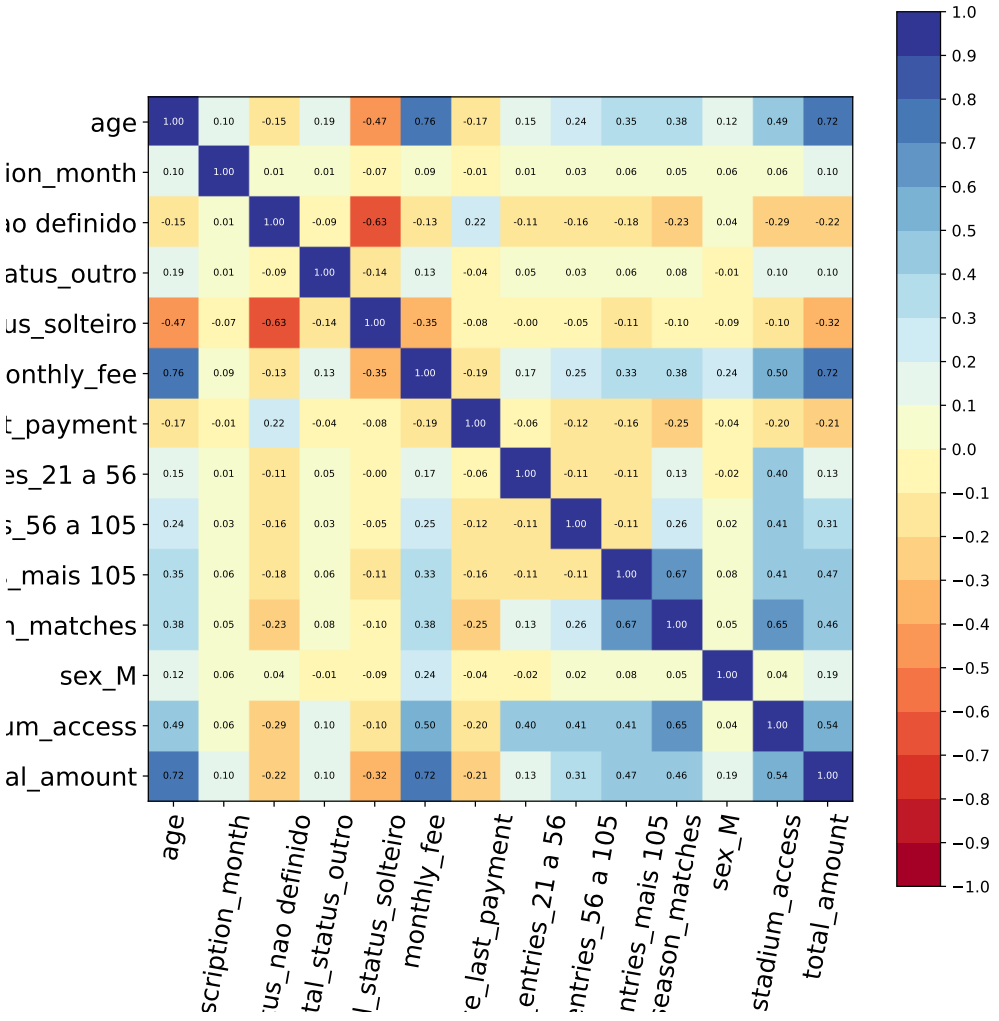
The survival trees based model uses pysurvival random forest.

```
from pysurvival.utils.display import correlation_matrix
import pandas as pd
import numpy as np

col = ['sex','marital_status','quart_stadium_entries']

df_members_sport = r.df_members_sport #copy r dataframe to python

df_members_sport = pd.get_dummies(df_members_sport, columns=col,
                                  drop_first=True)

# Creating the time and event columns
time_column = 'years_membership'
event_column = 'dropout'

# Extracting the features
features = np.setdiff1d(df_members_sport.columns,
                        [time_column, event_column] ).tolist()



correlation_matrix(df_members_sport[features],
                   figure_size=(10,10),
                   text_fontsize=6)
```

```
r.df_members_sport = df_members_sport
```

Removed the variables with greater correlations *total_matches* and *quart_stadium_entries*

```
to_remove = ['total_matches', 'quart_stadium_entries_ate 1']

#consider also the features previously removed
features = np.setdiff1d(df_members_sport[features].columns,
                        to_remove).tolist()

df_members_sport.drop(columns = to_remove, inplace=True)

correlation_matrix(df_members_sport[features],
                   figure_size=(10,10),
                   text_fontsize=6)
```



The model performance was determined with the concordance probability (C-index), Brier Score (BS) and Mean Absolute Error (MAE) (Wang, Li, and Reddy 2017). The feature importance was determined calculating the difference between the true class label and noised data (Breiman 2001).

The BS is used to evaluate the predicted accuracy of the survival function at a given time t. Representing the average square distance between the survival status and the predicted survival probability, where the value 0 is the best possible outcome.

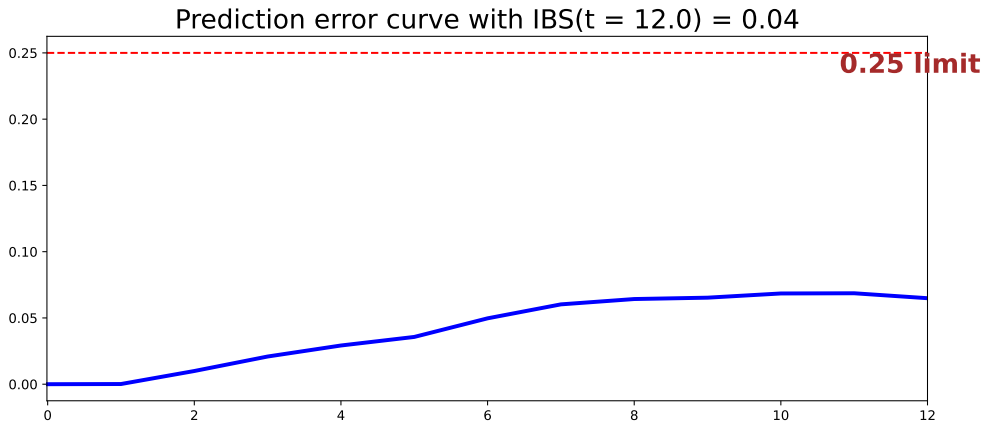
$$BS(t) = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{\left(0-\hat{S}(t,\vec{x}_i)\right)^2 \cdot \mathbb{1}_{T_i\leq t,\delta_i=1}}{\hat{G}(T_i^-)} + \frac{\left(1-\hat{S}(t,\vec{x}_i)\right)^2 \cdot \mathbb{1}_{T_i>t}}{\hat{G}(t)}\right) \tag{A.1}$$

The model should have a Brier score below 0.25. Considering that if $\forall i \in [\![1, N]\!], \hat{S}(t, \vec{x}_i) = 0.5$, then $BS(t) = 0.25$.

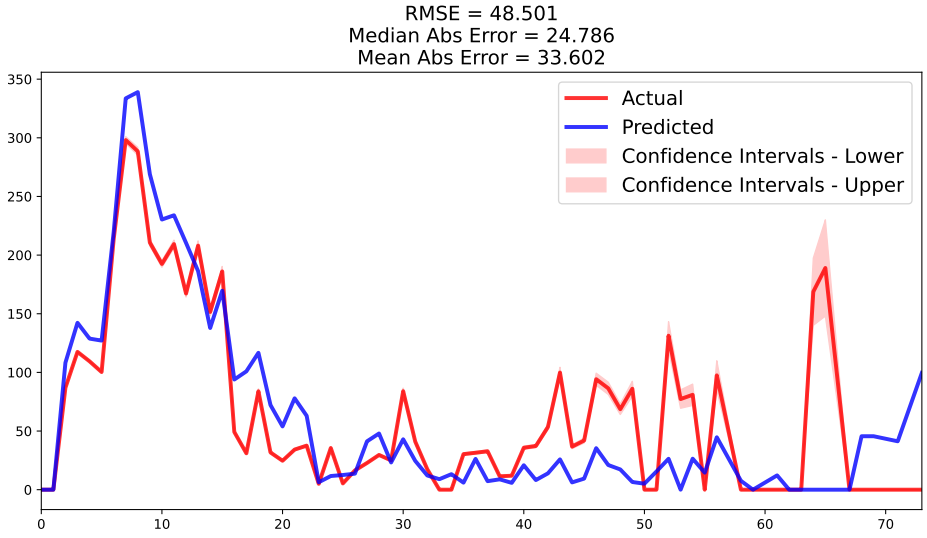
```
from pysurvival.models.survival_forest import RandomSurvivalForestModel
from sklearn.model_selection import train_test_split
from pysurvival.utils.metrics import concordance_index
from pysurvival.utils.display import integrated_brier_score
from pysurvival.utils.display import compare_to_actual

X = df_members_sport.copy()
t = df_members_sport['years_membership']
e = df_members_sport['dropout']
X.drop(axis=1, columns=['years_membership','dropout'], inplace=True)

X_train, X_test, t_train, t_test, e_train, e_test = train_test_split(X, t, e,
                                                    test_size=0.3, random_state=0)

# Fitting the model
csf = RandomSurvivalForestModel(num_trees=20)
csf.fit(X_train, t_train, e_train, max_features='sqrt', max_depth=5,
                                                    min_node_size=20, seed = 1)
```

```
## RandomSurvivalForestModel
```

```
c_index = concordance_index(csf, X_test, t_test, e_test)
ibs = integrated_brier_score(csf, X_test, t_test, e_test,
                             t_max=12, figure_size=(12,5))
```



```
results = compare_to_actual(csf, X_test, t_test, e_test,
                            is_at_risk = False,  figure_size=(12, 6),
                            metrics = ['rmse', 'mean', 'median'])
```




```
print(results)
```

```
{'root_mean_squared_error': 48.501131660621546, 'median_absolute_error': 24.785851010976415,
'mean_absolute_error': 33.601529151899356}
```

```
head(py$X)
```

```
##   age monthly_fee total_amount season_matches months_since_last_payment
## 1  83          10         1906              0                         3
## 2  88          10         1906              0                         3
## 3  73          10         1553              0                        36
## 4  97           5          790              0                         8
## 5  97          10         1466              0                        35
## 6  91           5          805              0                         4
##   stadium_access inscription_month sex_M marital_status_nao definido
## 1              0                10     1                           0
## 2              0                10     1                           0
## 3              0                 8     1                           1
## 4              0                 9     1                           0
## 5              0                 9     1                           0
```

```
## 6              1                12     1                           0
##   marital_status_outro marital_status_solteiro quart_stadium_entries_21 a 56
## 1                    0                       0                             0
## 2                    0                       1                             0
## 3                    0                       0                             0
## 4                    0                       0                             0
## 5                    1                       0                             0
## 6                    0                       0                             0
##   quart_stadium_entries_56 a 105 quart_stadium_entries_mais 105
## 1                              0                              0
## 2                              0                              0
## 3                              0                              0
## 4                              0                              0
## 5                              0                              0
## 6                              0                              0
```

```
names(py$X)
```

```
##  [1] "age"                              "monthly_fee"
##  [3] "total_amount"                     "season_matches"
##  [5] "months_since_last_payment"        "stadium_access"
##  [7] "inscription_month"                "sex_M"
##  [9] "marital_status_nao definido"      "marital_status_outro"
## [11] "marital_status_solteiro"          "quart_stadium_entries_21 a 56"
## [13] "quart_stadium_entries_56 a 105"   "quart_stadium_entries_mais 105"
```

Table B.3 shows variables importance.

```
tbl <- py$csf$variable_importance_table
kbl(tbl, booktabs = T, caption = "Features importance in the survival model")
```

The model was built with with 70% of the data for training and 30% for testing. The survival model parameters where:

A.3.2 Survival trees based model with clusters

Here we are will create clusters and developed the optimization within each cluster...

TABLE A.2: Features importance in the survival model

feature	importance	pct_importance
months_since_last_payment	9.5547860	0.3034228
total_amount	4.4419030	0.1410576
season_matches	3.4720831	0.1102598
stadium_access	2.6559085	0.0843413
marital_status_solteiro	2.2583599	0.0717167
monthly_fee	1.7899917	0.0568432
quart_stadium_entries_mais 105	1.5944932	0.0506349
inscription_month	1.5940396	0.0506205
age	1.3756597	0.0436856
quart_stadium_entries_21 a 56	0.9715332	0.0308521
sex_M	0.8753704	0.0277984
quart_stadium_entries_56 a 105	0.3698806	0.0117460
marital_status_nao definido	0.3088305	0.0098073
marital_status_outro	0.2271649	0.0072139

The calculation of he number of clusters used the package mclust (Scrucca et al. 2016) using the Bayesian Information Criterion (BIC). The model that gives the minimum BIC score can be selected as the best model (Schwarz 1978) simplifying the problem related to choosing the number of components and identifying the structure of the covariance matrix, based on modelling with multivariate normal distributions for each component that forms the data set (Akogul and Erisoglu 2016).

```
library(mclust)
y <- scale(py$df_members_sport)

set.seed(0) # to make reproducible

bic <- mclustBIC(y)
# Best model using the BIC criteria
plot(bic, what = "BIC")
summary(bic)

bic
```

Elbow calculation

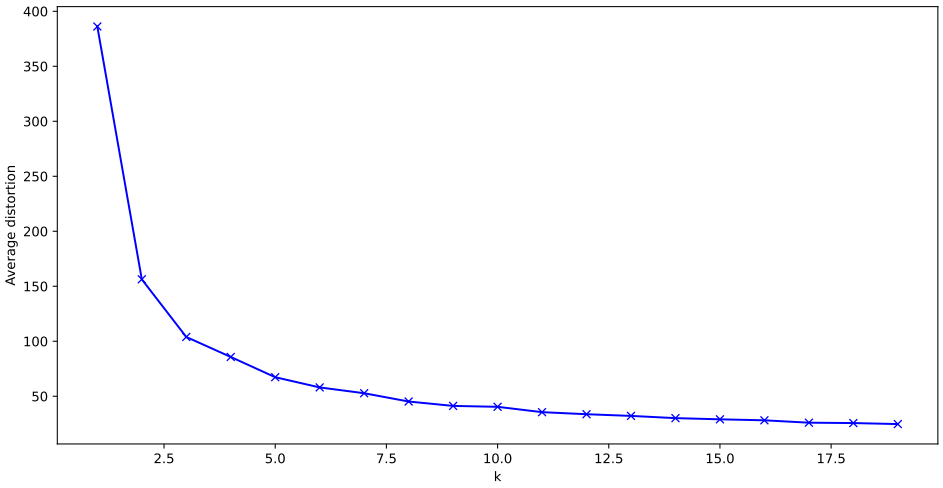
```
from sklearn.cluster import KMeans
from sklearn import preprocessing
from scipy.spatial.distance import cdist
```

```
import matplotlib.pyplot as plt


import numpy as np
#Finding optimal no. of clusters
clusters=range(1,20)
meanDistortions=[]

for k in clusters:
    model=KMeans(n_clusters=k)
    model.fit(df_members_sport)
    prediction=model.predict(df_members_sport)
    meanDistortions.append(sum(np.min(cdist(df_members_sport,
                           model.cluster_centers_, 'euclidean'), axis=1)) / df_members_sport.sha
```

```
## KMeans(n_clusters=1)
## KMeans(n_clusters=2)
## KMeans(n_clusters=3)
## KMeans(n_clusters=4)
## KMeans(n_clusters=5)
## KMeans(n_clusters=6)
## KMeans(n_clusters=7)
## KMeans()
## KMeans(n_clusters=9)
## KMeans(n_clusters=10)
## KMeans(n_clusters=11)
## KMeans(n_clusters=12)
## KMeans(n_clusters=13)
## KMeans(n_clusters=14)
## KMeans(n_clusters=15)
## KMeans(n_clusters=16)
## KMeans(n_clusters=17)
## KMeans(n_clusters=18)
## KMeans(n_clusters=19)
```

```
plt.plot(clusters, meanDistortions, 'bx-')
plt.xlabel('k')
```

```
plt.ylabel('Average distortion')
plt.show();
```



We are going to consider five clusters

```
cluster = KMeans(n_clusters=5, random_state=0)


cluster.fit(df_members_sport)
```

```
## KMeans(n_clusters=5, random_state=0)
```

```
df_members_sport['cluster']=cluster.predict(df_members_sport)
print(df_members_sport.cluster.value_counts());
```

```
## 0    17069
## 2     2817
## 4     2420
## 1     2080
## 3      930
## Name: cluster, dtype: int64
```

Summary statistics each cluster

```
library(gtsummary)
library(kableExtra)
library(labelled)

df_summary_clusters <- py$df_members_sport

var_label(df_summary_clusters$age) <- "Age in years"
var_label(df_summary_clusters$sex) <- "Male or female"

df_summary_clusters$cluster[df_summary_clusters$cluster == 4] <- "Cluster 5"
df_summary_clusters$cluster[df_summary_clusters$cluster == 3] <- "Cluster 4"
df_summary_clusters$cluster[df_summary_clusters$cluster == 2] <- "Cluster 3"
df_summary_clusters$cluster[df_summary_clusters$cluster == 1] <- "Cluster 2"
df_summary_clusters$cluster[df_summary_clusters$cluster == 0] <- "Cluster 1"

tbl <- df_summary_clusters %>%
    tbl_summary(by='cluster',statistic = list(all_continuous() ~ "{mean} ({sd})",
                                all_categorical() ~ "{p}%"),
                type = list(age ~ "continuous")) %>%
    add_stat_label()

as_kable_extra(tbl, booktabs = T,
               caption = "Summary statistics of each cluster",) %>%
  kable_styling(font_size = 7)
```

Cluster representation with PCA reducing to two variables to allow visualization.

```
from sklearn.decomposition import PCA

plt.rcParams['figure.figsize'] = [13, 4]

pca=PCA(n_components=2)
#todas as linhas da primeira coluna com redução
df_members_sport['X_pca']=pca.fit_transform(df_members_sport)[:,0]
#todas as linhas da segunda coluna com redução
df_members_sport['Y_pca']=pca.fit_transform(df_members_sport)[:,1]

fig, ax = plt.subplots()
```

TABLE A.3: Summary statistics of each cluster

Characteristic	**Cluster 1**, N = 17,069	**Cluster 2**, N = 2,080	**Cluster 3**, N = 2,817	**Cluster 4**, N = 930	**Clus**
Age in years, Mean (SD)	17 (12)	54 (13)	41 (18)	60 (13)	
monthly_fee, %					
0	¡0.1%	0%	¡0.1%	0%	
1	47%	0%	¡0.1%	0.1%	
2.5	39%	0%	15%	0%	
5	0.6%	6.7%	11%	2.5%	
6	5.1%	2.1%	39%	1.9%	
10	7.8%	91%	35%	95%	
total_amount, Mean (SD)	35 (46)	1,292 (112)	383 (115)	1,823 (134)	
season_matches, Mean (SD)	0.9 (2.7)	6.5 (5.1)	3.7 (4.7)	5.1 (4.9)	
months_since_last_payment, Mean (SD)	23 (38)	5 (8)	14 (17)	5 (5)	
dropout, %	27%	4.8%	23%	2.3%	
years_membership, Mean (SD)	7 (5)	26 (13)	12 (10)	34 (14)	
stadium_access, %	20%	88%	76%	84%	
inscription_month, Mean (SD)	6.7 (3.5)	7.5 (3.2)	7.0 (3.1)	7.9 (3.1)	
sex_M, %	65%	100%	62%	99%	
marital_status_nao definido, %	38%	12%	19%	13%	
marital_status_outro, %	0.8%	3.8%	4.8%	4.5%	
marital_status_solteiro, %	58%	14%	43%	6.1%	
quart_stadium_entries_21 a 56, %	5.4%	11%	26%	15%	
quart_stadium_entries_56 a 105, %	2.3%	26%	23%	23%	
quart_stadium_entries_mais 105, %	1.0%	46%	14%	36%	
Male or female, %	65%	100%	62%	99%	

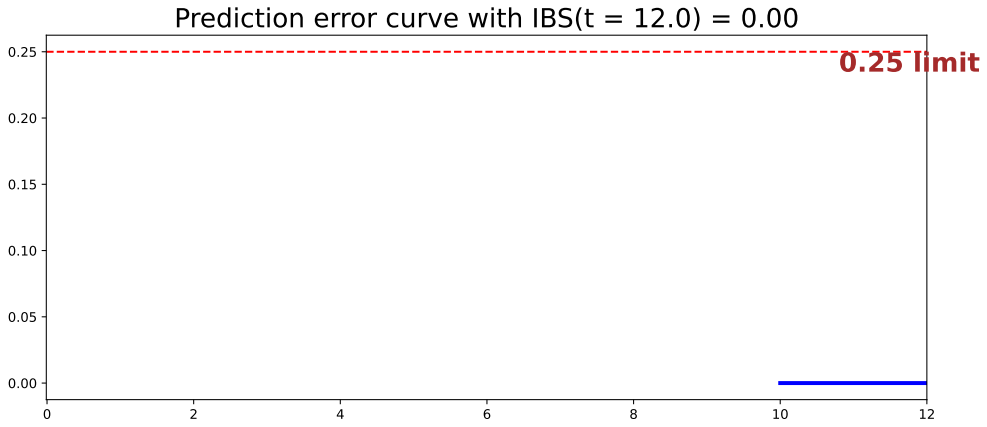
```
for cluster in df_members_sport.cluster.unique():
    x = df_members_sport['X_pca'].loc[df_members_sport.cluster == cluster]
    y = df_members_sport['Y_pca'].loc[df_members_sport.cluster == cluster]
    ax.scatter(x, y, label=cluster, alpha=1, edgecolors='none');

ax.legend()
ax.grid(True)
ax.set_title('Clusters clientes');
```

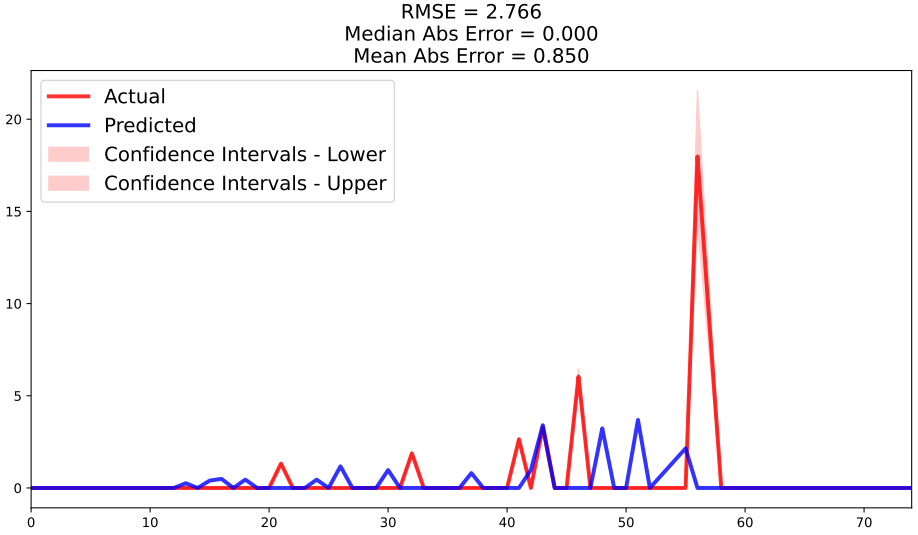
Clusters

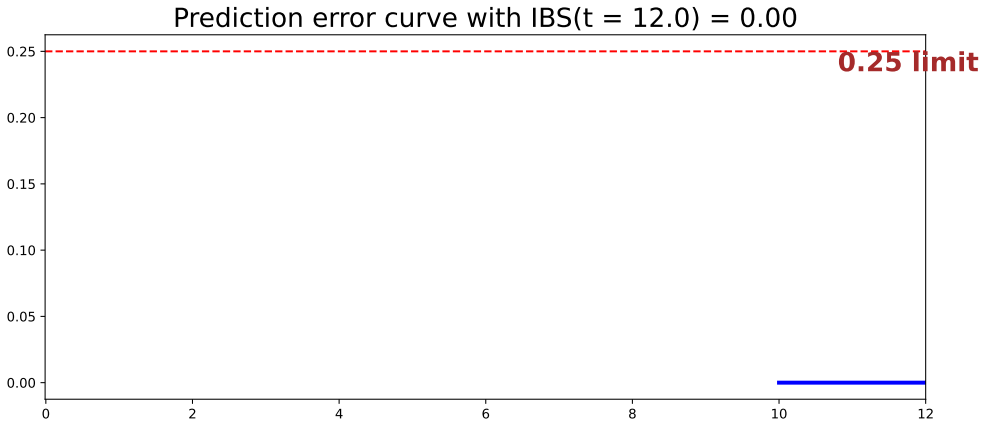
```
# Building training and testing sets for the clusters

df_resultados = pd.DataFrame(columns=['cluster','rmse','mean','median'])

for cluster in df_members_sport.cluster.unique():
    # Number of samples in the dataset
    df_members_sport_cluster = df_members_sport[df_members_sport.cluster == cluster]
    X = df_members_sport_cluster.copy()
    t = df_members_sport_cluster['years_membership']
    e = df_members_sport_cluster['dropout']
    X.drop(axis=1, columns=['years_membership','dropout'],inplace=True)
```

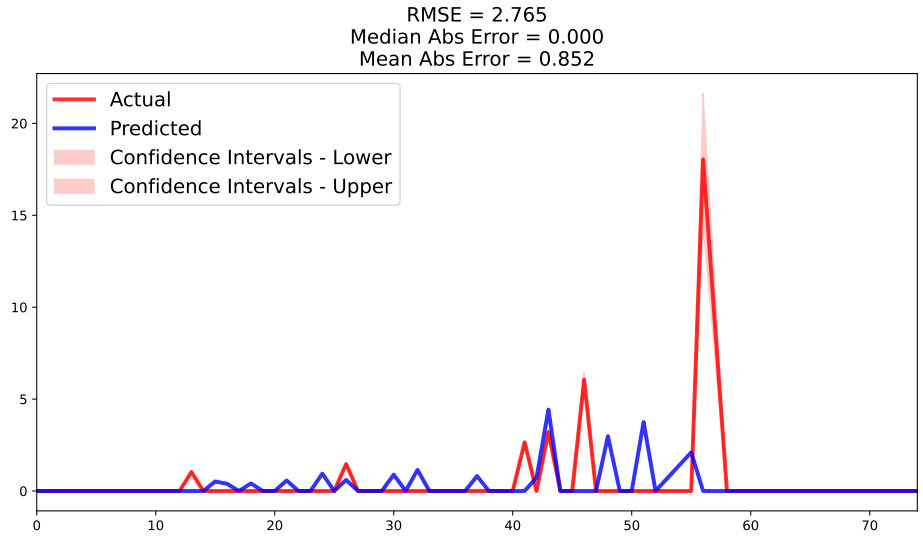
```
    X_train, X_test, t_train, t_test, e_train, e_test = train_test_split(X, t, e, random_state=0)
    print(f"The cluster {cluster} as a size of {X.shape[0]}")
    # Fitting the model
    csf = RandomSurvivalForestModel(num_trees=20)
    csf.fit(X_train, t_train, e_train, max_features='sqrt',
        max_depth=5, min_node_size=20, seed = 1)

    c_index = concordance_index(csf, X_test, t_test, e_test)

    ibs = integrated_brier_score(csf, X_test, t_test, e_test, t_max=12, figure_size=(12,5))

    results_cluster = compare_to_actual(csf, X_test, t_test, e_test, is_at_risk = False,
                                figure_size=(12, 6), metrics = ['rmse', 'mean', 'median'])

    df_resultados = df_resultados.append({'cluster' : cluster,
                                          'rmse' : results_cluster['root_mean_squared_error'],
                                          'mean' : results_cluster['median_absolute_error'],
                                          'median' : results_cluster['mean_absolute_error']},
                                          ignore_index = True)
```

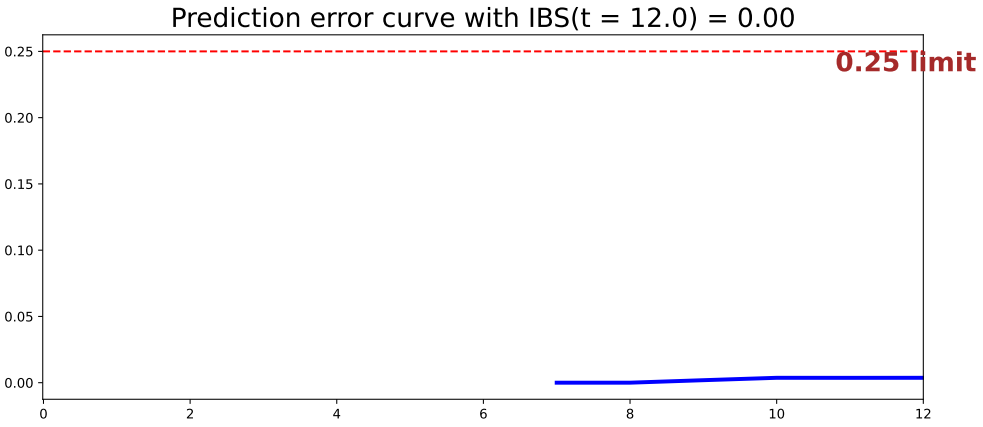
```
## The cluster 3 as a size of 930
## RandomSurvivalForestModel
## The cluster 1 as a size of 2080
## RandomSurvivalForestModel
## The cluster 4 as a size of 2420
## RandomSurvivalForestModel
## The cluster 2 as a size of 2817
## RandomSurvivalForestModel
## The cluster 0 as a size of 17069
## RandomSurvivalForestModel
```

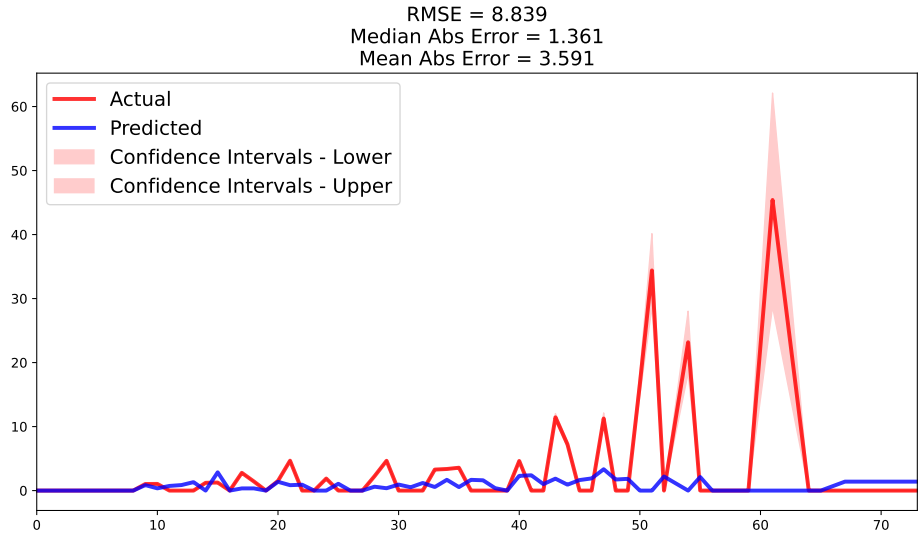

Prediction error curve with IBS(t = 12.0) = 0.00
0.25 limit
0.25
0.20
0.15
0.10
0.05
0.00
0
2
4
6
8
10
12

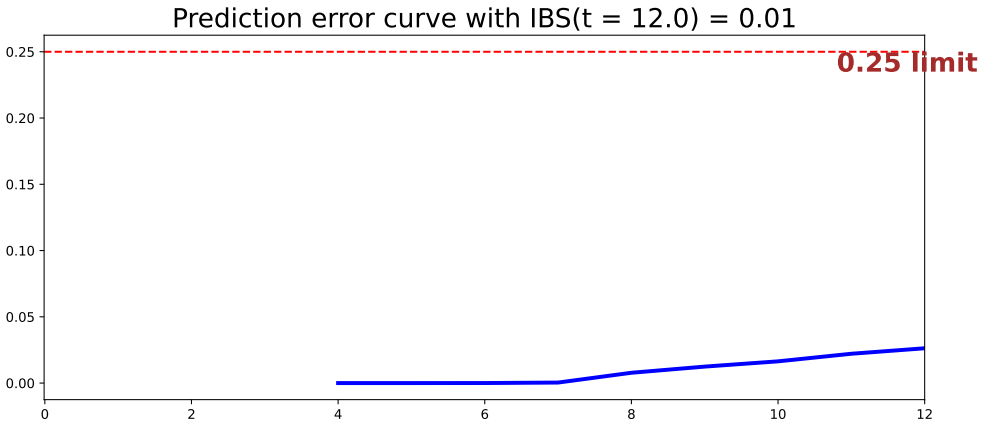

RMSE = 2.766
Median Abs Error = 0.000
Mean Abs Error = 0.850
Actual
Predicted
Confidence Intervals - Lower
Confidence Intervals - Upper
20
15
10
5
0
0
10
20
30
40
50
60
70

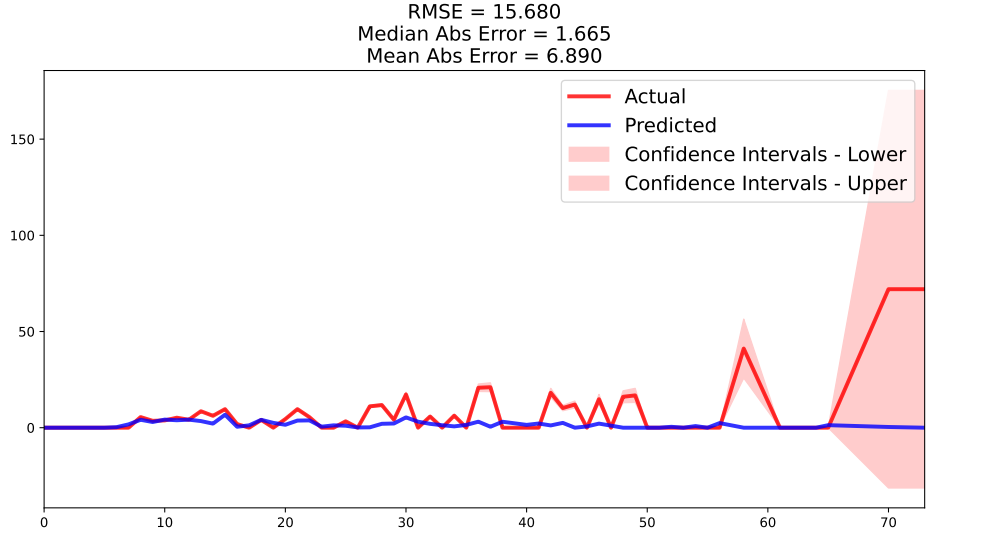

Prediction error curve with IBS(t = 12.0) = 0.00
0.25 limit
0.25
0.20
0.15
0.10
0.05
0.00
0
2
4
6
8
10
12

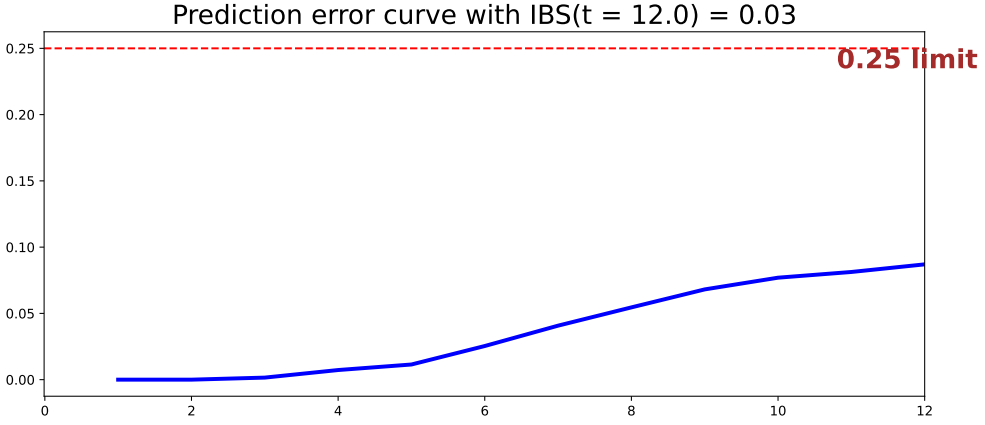

RMSE = 2.765
Median Abs Error = 0.000
Mean Abs Error = 0.852
Actual
Predicted
Confidence Intervals - Lower
Confidence Intervals - Upper

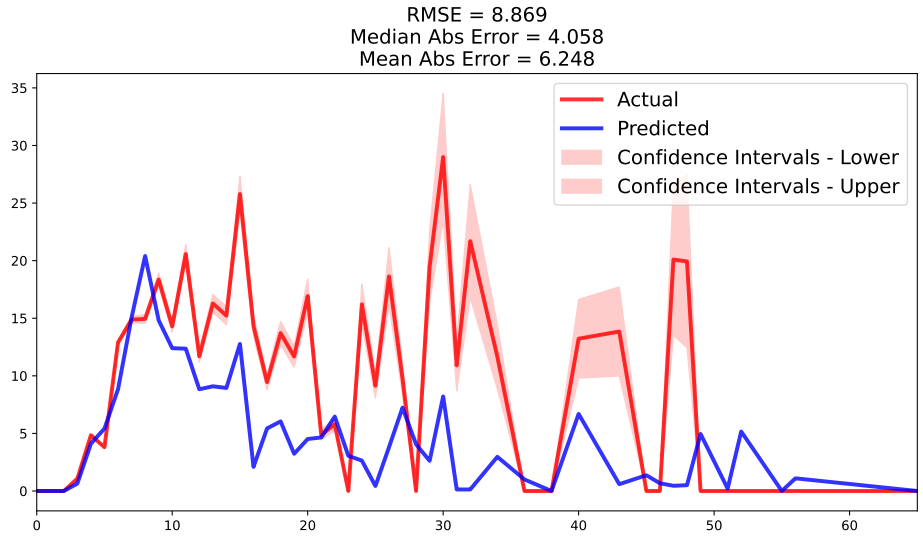

Prediction error curve with IBS(t = 12.0) = 0.00
0.25 limit


RMSE = 8.839
Median Abs Error = 1.361
Mean Abs Error = 3.591
Actual
Predicted
Confidence Intervals - Lower
Confidence Intervals - Upper


Prediction error curve with IBS(t = 12.0) = 0.01
0.25 limit
0.25
0.20
0.15
0.10
0.05
0.00
0
2
4
6
8
10
12


RMSE = 15.680
Median Abs Error = 1.665
Mean Abs Error = 6.890
Actual
Predicted
Confidence Intervals - Lower
Confidence Intervals - Upper
150
100
50
0
0
10
20
30
40
50
60
70


Prediction error curve with IBS(t = 12.0) = 0.03
0.25 limit
0.25
0.20
0.15
0.10
0.05
0.00
0
2
4
6
8
10
12



The performance of the cluster 1 the IBS presents an accuracy of 0.06 (figure B.11) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as an mean absolute error of 1.5 customers, the mean median absolute error was 0.61 and the Root Mean Square Error of 2.8 (figure B.12).

The features importance in the survival model cluster 1 (table B.5) identify the three most relevant features to predict survival *maccess*, *tbilled*, and $dayswfreq$. The features with lower relevance were $freeuse$, *sex* and $cfreq$.

```
df_members_sport_cluster = df_members_sport[df_members_sport.cluster == 0].copy()
X = df_members_sport_cluster.copy()
t = df_members_sport_cluster['years_membership']
e = df_members_sport_cluster['dropout']
X.drop(axis=1,
       columns=['years_membership','dropout','X_pca','Y_pca','cluster'],
       inplace=True)

X_train, X_test, t_train, t_test, e_train, e_test = train_test_split(X, t,
                                                    e, random_state=0)
# Fitting the model
csf = RandomSurvivalForestModel(num_trees=20)
csf.fit(X_train, t_train, e_train, max_features='sqrt', max_depth=5,
        min_node_size=20, seed = 1)
```

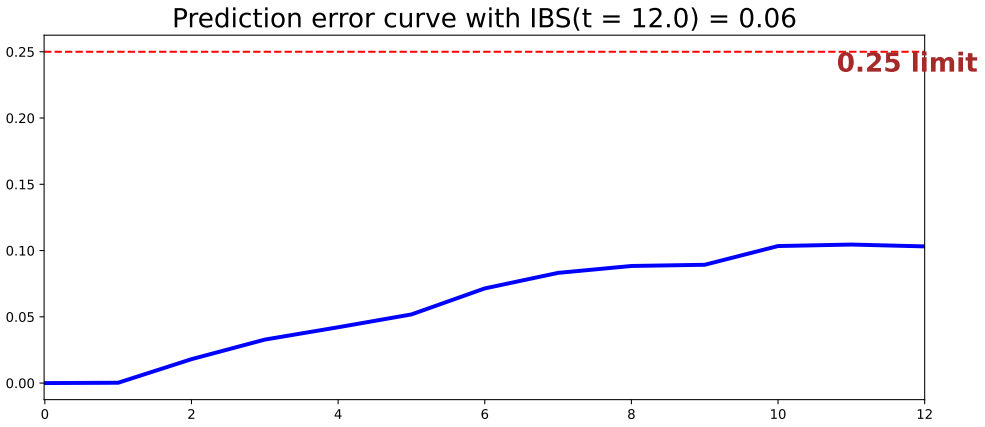


FIGURE A.2: Model performance cluster 1

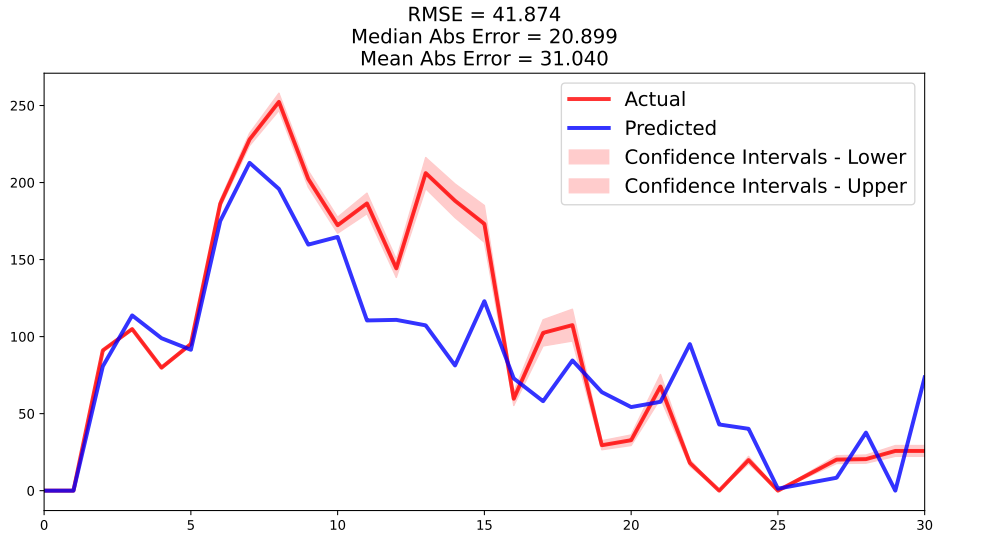


FIGURE A.3: Conditional survival forest cluster 1

```
c_index = concordance_index(csf, X_test, t_test, e_test)
ibs = integrated_brier_score(csf, X_test, t_test, e_test, t_max=12,
                             figure_size=(12,5))
```

```
compare_to_actual(csf, X_test, t_test, e_test,
                  is_at_risk = False,
                  figure_size=(12, 6),
                  metrics = ['rmse', 'mean', 'median'])
```

TABLE A.4: Features importance in the survival model with cluster 1

feature	importance	pct_importance
months_since_last_payment	7.9718880	0.2506004
total_amount	3.4904383	0.1097237
quart_stadium_entries_21 a 56	3.0461302	0.0957567
season_matches	3.0189963	0.0949037
monthly_fee	2.8196026	0.0886357
inscription_month	2.2867690	0.0718858
marital_status_solteiro	2.1419768	0.0673341
quart_stadium_entries_mais 105	1.7729598	0.0557339
stadium_access	1.7154423	0.0539258
marital_status_nao definido	1.3549447	0.0425934
quart_stadium_entries_56 a 105	1.2276591	0.0385921
age	0.9643497	0.0303148
sex_M	-0.7487548	0.0000000
marital_status_outro	-1.8563364	0.0000000

```
tbl <- py$csf$variable_importance_table
kbl(tbl, booktabs = T,
    caption = "Features importance in the survival model with cluster 1")
```

The performance of the cluster 2 the IBS presents an accuracy of xxxxx (figure B.13) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the xxxx months, which as an mean absolute error of xxxx customers, the mean median absolute error was xxxx and the Root Mean Square Error of xxxx (figure B.14).

The features importance in the survival model cluster 1 (table B.6) identify the three most relevant features to predict survival *maccess*, *tbilled*, and $daysw freq$. The features with lower relevance were $freeuse$, *sex* and $cfreq$.

```
df_members_sport_cluster = df_members_sport[df_members_sport.cluster == 1].copy()
X = df_members_sport_cluster.copy()
t = df_members_sport_cluster['years_membership']
e = df_members_sport_cluster['dropout']
X.drop(axis=1,
       columns=['years_membership','dropout','X_pca','Y_pca','cluster'],
       inplace=True)

X_train, X_test, t_train, t_test, e_train, e_test = train_test_split(X, t,
                                                    e, random_state=0)
```

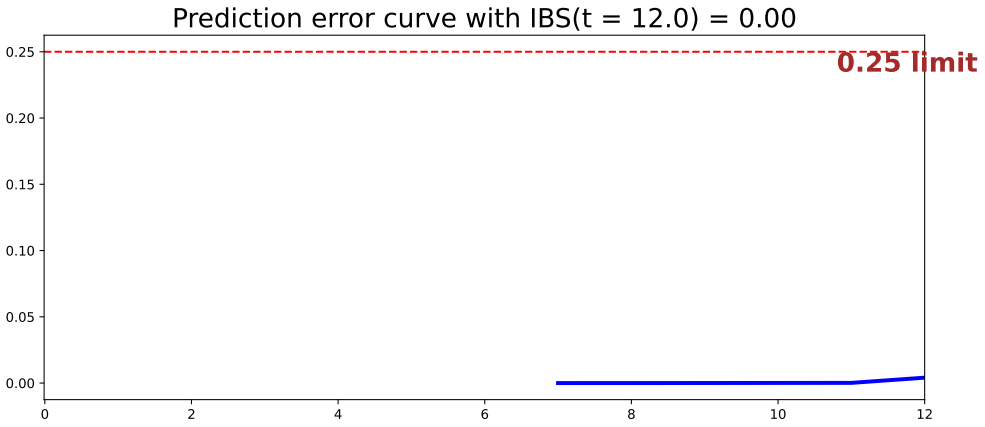


FIGURE A.4: Model performance cluster 2

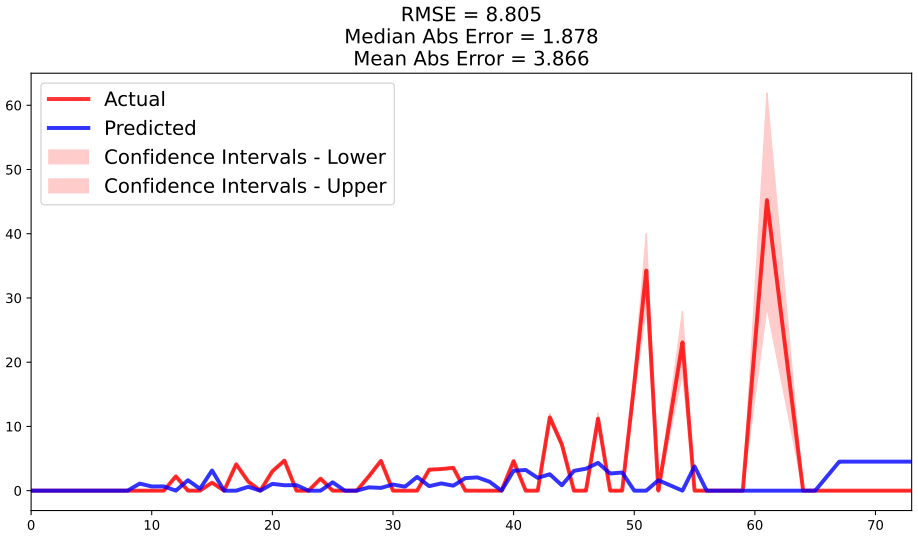


FIGURE A.5: Conditional survival forest cluster 2

```
# Fitting the model
csf = RandomSurvivalForestModel(num_trees=20)
csf.fit(X_train, t_train, e_train, max_features='sqrt', max_depth=5,
        min_node_size=20, seed = 1)
```

```
c_index = concordance_index(csf, X_test, t_test, e_test)
ibs = integrated_brier_score(csf, X_test, t_test, e_test, t_max=12,
                             figure_size=(12,5))
```

```
compare_to_actual(csf, X_test, t_test, e_test, is_at_risk = False, figure_size=(12, 6),
                  metrics = ['rmse', 'mean', 'median'])
```

TABLE A.5: Features importance in the survival model with cluster 2

feature	importance	pct_importance
months_since_last_payment	5.9872066	0.2491094
season_matches	4.9914874	0.2076806
quart_stadium_entries_mais 105	2.2792423	0.0948323
stadium_access	2.2573120	0.0939199
age	1.9097863	0.0794604
marital_status_solteiro	1.6355833	0.0680516
marital_status_nao definido	1.3333022	0.0554746
monthly_fee	1.0488715	0.0436403
quart_stadium_entries_21 a 56	1.0286933	0.0428008
total_amount	0.9596941	0.0399299
quart_stadium_entries_56 a 105	0.6032673	0.0251001
sex_M	0.0000000	0.0000000
marital_status_outro	0.0000000	0.0000000
inscription_month	-1.3324701	0.0000000

```
tbl <- py$csf$variable_importance_table
kbl(tbl, booktabs = T,
    caption = "Features importance in the survival model with cluster 2")
```

The performance of the cluster 3 the IBS presents an accuracy of xxxxx (figure B.15) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the xxxx months, which as an mean absolute error of xxxx customers, the mean median absolute error was xxxx and the Root Mean Square Error of xxxx (figure B.16).

The features importance in the survival model cluster 1 (table B.7) identify the three most relevant features to predict survival *maccess*, *tbilled*, and $daysw freq$. The features with lower relevance were $freeuse$, *sex* and $cfreq$.

```
df_members_sport_cluster = df_members_sport[df_members_sport.cluster == 2].copy()
X = df_members_sport_cluster.copy()
t = df_members_sport_cluster['years_membership']
e = df_members_sport_cluster['dropout']
X.drop(axis=1,
       columns=['years_membership','dropout','X_pca','Y_pca','cluster'],
       inplace=True)

X_train, X_test, t_train, t_test, e_train, e_test = train_test_split(X, t,
                                                    e, random_state=0)
```

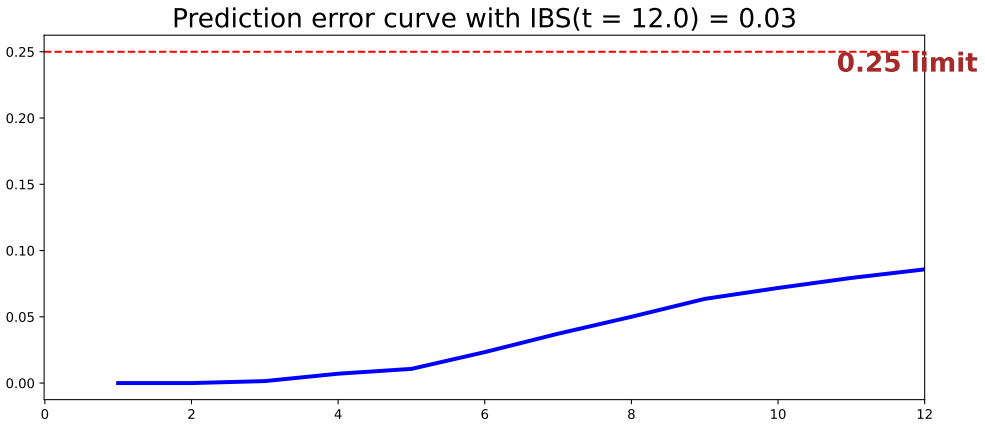


FIGURE A.6: Model performance cluster 3

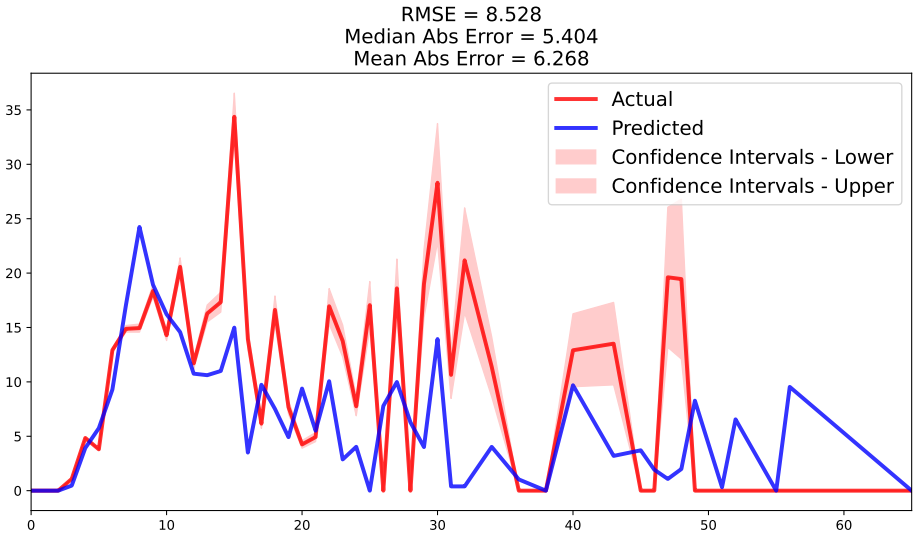


FIGURE A.7: Conditional survival forest cluster 3

```
# Fitting the model
csf = RandomSurvivalForestModel(num_trees=20)
csf.fit(X_train, t_train, e_train, max_features='sqrt', max_depth=5,
        min_node_size=20, seed = 1)
```

```
c_index = concordance_index(csf, X_test, t_test, e_test)
ibs = integrated_brier_score(csf, X_test, t_test, e_test, t_max=12,
                             figure_size=(12,5))
```

```
compare_to_actual(csf, X_test, t_test, e_test, is_at_risk = False, figure_size=(12, 6),
                  metrics = ['rmse', 'mean', 'median'])
```

TABLE A.6: Features importance in the survival model with cluster 3

feature	importance	pct_importance
months_since_last_payment	8.2219201	0.2458512
monthly_fee	4.2692367	0.1276584
season_matches	3.3464787	0.1000662
total_amount	2.5500942	0.0762527
age	2.5107861	0.0750773
quart_stadium_entries_mais 105	2.4517413	0.0733118
stadium_access	1.8968781	0.0567203
quart_stadium_entries_56 a 105	1.8554828	0.0554825
marital_status_solteiro	1.7158599	0.0513075
quart_stadium_entries_21 a 56	1.6516070	0.0493862
inscription_month	1.4075370	0.0420881
sex_M	1.1621292	0.0347499
marital_status_nao definido	0.4029124	0.0120479
marital_status_outro	-1.0259784	0.0000000

```
tbl <- py$csf$variable_importance_table
kbl(tbl, booktabs = T,
    caption = "Features importance in the survival model with cluster 3")
```

The performance of the cluster 4 the IBS presents an accuracy of xxxxx (figure A.8) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the xxxx months, which as an mean absolute error of xxxx customers, the mean median absolute error was xxxx and the Root Mean Square Error of xxxx (figure B.17).

The features importance in the survival model cluster 1 (table B.8) identify the three most relevant features to predict survival *maccess*, *tbilled*, and $dayswfreq$. The features with lower relevance were $freeuse$, *sex* and $cfreq$.

```
df_members_sport_cluster = df_members_sport[df_members_sport.cluster == 3].copy()
X = df_members_sport_cluster.copy()
t = df_members_sport_cluster['years_membership']
e = df_members_sport_cluster['dropout']
X.drop(axis=1,
       columns=['years_membership','dropout','X_pca','Y_pca','cluster'],
       inplace=True)

X_train, X_test, t_train, t_test, e_train, e_test = train_test_split(X, t,
                                                    e, random_state=0)
```

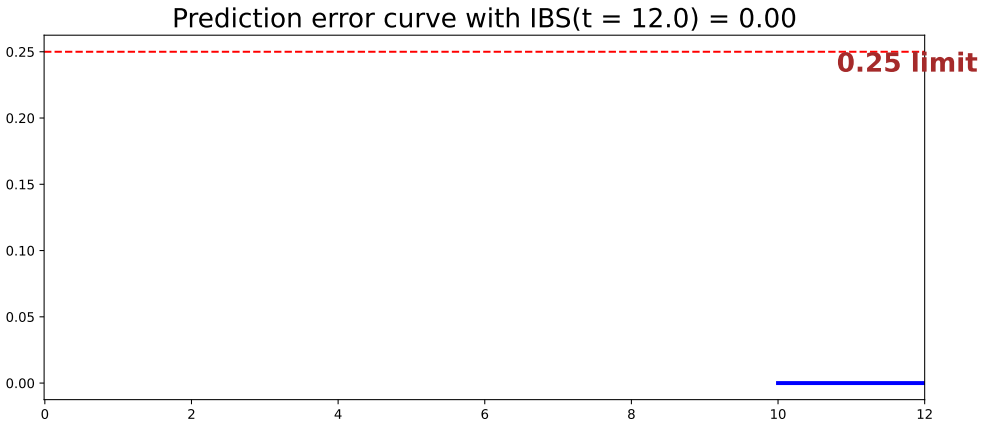


FIGURE A.8: Model performance cluster 4

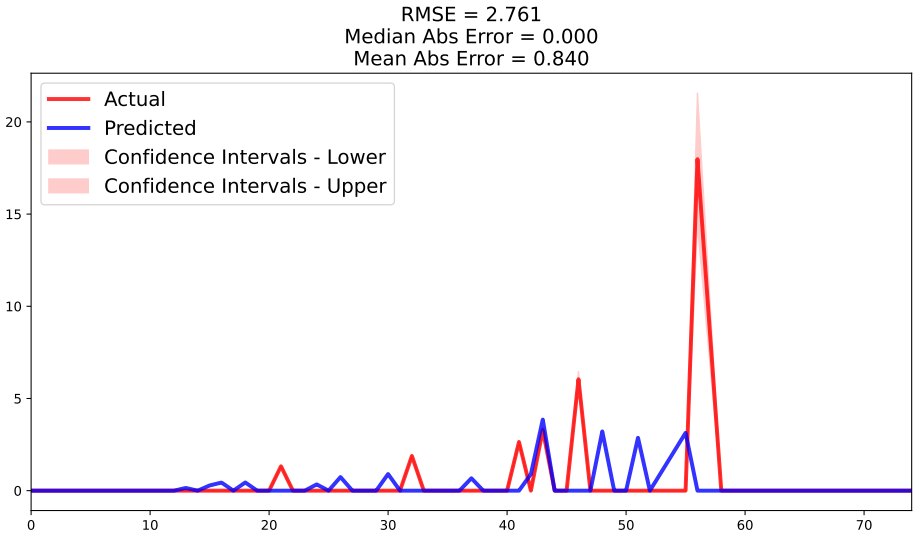


FIGURE A.9: Conditional survival forest cluster 4

```
# Fitting the model
csf = RandomSurvivalForestModel(num_trees=20)
csf.fit(X_train, t_train, e_train, max_features='sqrt', max_depth=5,
        min_node_size=20, seed = 1)
```

```
c_index = concordance_index(csf, X_test, t_test, e_test)
ibs = integrated_brier_score(csf, X_test, t_test, e_test, t_max=12,
                             figure_size=(12,5))
```

```
compare_to_actual(csf, X_test, t_test, e_test, is_at_risk = False, figure_size=(12, 6),
                  metrics = ['rmse', 'mean', 'median'])
```

TABLE A.7: Features importance in the survival model with cluster 4

feature	importance	pct_importance
months_since_last_payment	4.8835861	0.2873432
total_amount	3.6484226	0.2146679
age	1.4886006	0.0875871
quart_stadium_entries_mais 105	1.3693405	0.0805700
inscription_month	1.1782424	0.0693261
quart_stadium_entries_56 a 105	1.1495877	0.0676401
marital_status_solteiro	1.0893559	0.0640961
season_matches	1.0259784	0.0603671
quart_stadium_entries_21 a 56	0.6879737	0.0404794
stadium_access	0.4745695	0.0279230
monthly_fee	0.0000000	0.0000000
sex_M	0.0000000	0.0000000
marital_status_outro	0.0000000	0.0000000
marital_status_nao definido	-0.6536132	0.0000000

```
tbl <- py$csf$variable_importance_table
kbl(tbl, booktabs = T,
    caption = "Features importance in the survival model with cluster 4")
```

The performance of the cluster 5 the IBS presents an accuracy of xxxxx (figure A.8) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the xxxx months, which as an mean absolute error of xxxx customers, the mean median absolute error was xxxx and the Root Mean Square Error of xxxx (figure B.17).

The features importance in the survival model cluster 1 (table B.8) identify the three most relevant features to predict survival *maccess*, *tbilled*, and $daysw freq$. The features with lower relevance were $freeuse$, *sex* and $cfreq$.

```
df_members_sport_cluster = df_members_sport[df_members_sport.cluster == 4].copy()
X = df_members_sport_cluster.copy()
t = df_members_sport_cluster['years_membership']
e = df_members_sport_cluster['dropout']
X.drop(axis=1,
       columns=['years_membership','dropout','X_pca','Y_pca','cluster'],
       inplace=True)

X_train, X_test, t_train, t_test, e_train, e_test = train_test_split(X, t,
                                                    e, random_state=0)
```

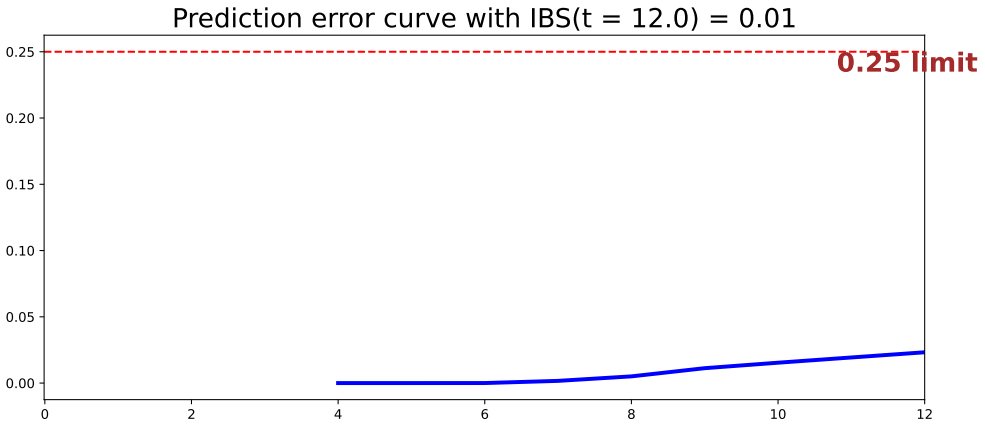


FIGURE A.10: Model performance cluster 5

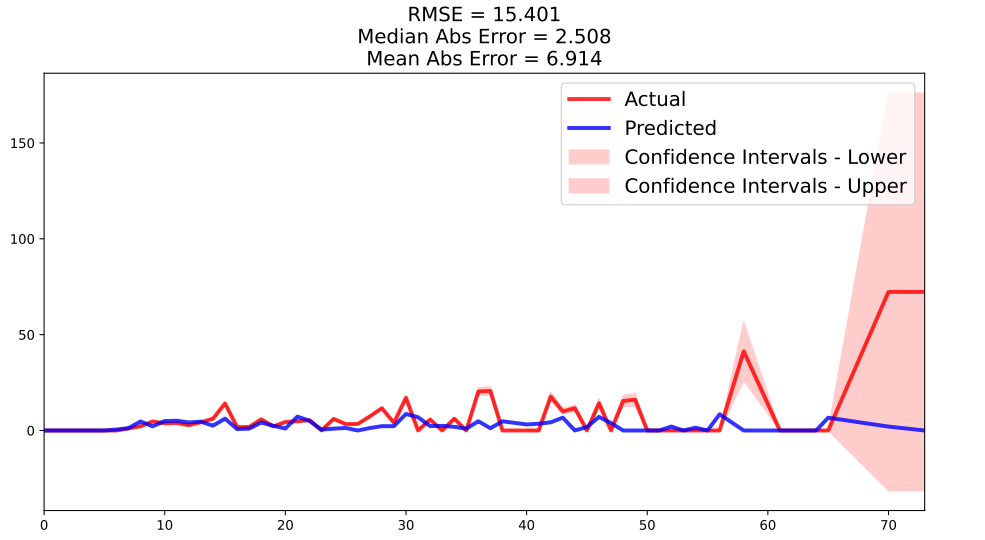


FIGURE A.11: Conditional survival forest cluster 5

```
# Fitting the model
csf = RandomSurvivalForestModel(num_trees=20)
csf.fit(X_train, t_train, e_train, max_features='sqrt', max_depth=5,
        min_node_size=20, seed = 1)
```

```
c_index = concordance_index(csf, X_test, t_test, e_test)
ibs = integrated_brier_score(csf, X_test, t_test, e_test, t_max=12,
                             figure_size=(12,5))
```

```
compare_to_actual(csf, X_test, t_test, e_test, is_at_risk = False, figure_size=(12, 6),
                  metrics = ['rmse', 'mean', 'median'])
```

TABLE A.8: Features importance in the survival model with cluster 5

feature	importance	pct_importance
months_since_last_payment	9.648520	0.2846462
season_matches	3.738756	0.1102991
monthly_fee	2.711904	0.0800053
total_amount	2.557423	0.0754479
quart_stadium_entries_mais 105	2.387826	0.0704445
quart_stadium_entries_21 a 56	2.299669	0.0678438
age	2.296840	0.0677603
stadium_access	2.066592	0.0609676
sex_M	1.871691	0.0552178
inscription_month	1.821203	0.0537283
marital_status_solteiro	1.389542	0.0409936
quart_stadium_entries_56 a 105	1.106573	0.0326456
marital_status_outro	0.000000	0.0000000
marital_status_nao definido	-1.025978	0.0000000

```
tbl <- py$csf$variable_importance_table

kbl(tbl, booktabs = T,
    caption = "Features importance in the survival model with cluster 5")
```

References

Akogul, Serkan and Murat Erisoglu. 2016. "A Comparison of Information Criteria in Clustering Based on Mixture of Multivariate Normal Distributions." *Mathematical and Computational Applications* 21(3):34.

Breiman, Leo. 2001. "Random Forests." *Machine Learning* 45(1):5–32.

Davidson-Pilon, Cameron. 2021. *CamDavidsonPilon/Lifelines*.

Fotso, Stephane and others. 2019. *PySurvival: Open Source Package for Survival Analysis Modeling*.

Schwarz, Gideon. 1978. "Estimating the Dimension of a Model." *The Annals of Statistics* 6(2):461–64.

Scrucca, Luca, Michael Fop, T. ,Brendan Murphy, and Adrian,E. Raftery. 2016. "Mclust 5: Clustering, Classification and Density Estimation Using Gaussian Finite Mixture Models." *The R Journal* 8(1):289.

Wang, Ping, Yan Li, and Chandan K. Reddy. 2017. "Machine Learning for Survival Analysis: A Survey." *arXiv:1708.04649 [Cs, Stat]*.

Appendix: Chunk options

A.3.3 Software versioning

A.3.3.1 R

```
cat(paste("#", capture.output(sessionInfo()), "\n", collapse  = ""))
```

```
## # R version 4.1.3 (2022-03-10)
## # Platform: x86_64-w64-mingw32/x64 (64-bit)
## # Running under: Windows 10 x64 (build 22621)
## #
## # Matrix products: default
## #
## # locale:
## # [1] LC_COLLATE=Portuguese_Portugal.1252  LC_CTYPE=Portuguese_Portugal.1252
```

```
## # [3] LC_MONETARY=Portuguese_Portugal.1252 LC_NUMERIC=C
## # [5] LC_TIME=Portuguese_Portugal.1252
## #
## # attached base packages:
## # [1] stats     graphics  grDevices utils     datasets  methods   base
## #
## # other attached packages:
## #  [1] labelled_2.9.1   kableExtra_1.3.4 gtsummary_1.6.0  visdat_0.5.3
## #  [5] readxl_1.4.0     stargazer_5.2.3  reticulate_1.25  ggplot2_3.3.6
## #  [9] dlookr_0.5.6     dplyr_1.0.9
## #
## # loaded via a namespace (and not attached):
## #  [1] reactable_0.2.3     webshot_0.5.3       httr_1.4.3
## #  [4] tools_4.1.3         utf8_1.2.2          R6_2.5.1
## #  [7] rpart_4.1.16        colorspace_2.0-3    withr_2.5.0
## # [10] tidyselect_1.1.2    gridExtra_2.3       curl_4.3.2
## # [13] compiler_4.1.3      extrafontdb_1.0     cli_3.3.0
## # [16] rvest_1.0.2         gt_0.6.0            xml2_1.3.3
## # [19] labeling_0.4.2      bookdown_0.26       scales_1.2.0
## # [22] mvtnorm_1.1-3       rappdirs_0.3.3      systemfonts_1.0.4
## # [25] stringr_1.4.0       digest_0.6.29       rmarkdown_2.14
## # [28] svglite_2.1.0       pkgconfig_2.0.3     htmltools_0.5.2
## # [31] showtext_0.9-5      extrafont_0.18      fastmap_1.1.0
## # [34] highr_0.9           htmlwidgets_1.5.4   rlang_1.0.2
## # [37] rstudioapi_0.13     sysfonts_0.8.8      shiny_1.7.1
## # [40] generics_0.1.2      farver_2.1.0        jsonlite_1.8.0
## # [43] magrittr_2.0.3      Formula_1.2-4       Matrix_1.4-1
## # [46] Rcpp_1.0.8.3        munsell_0.5.0       fansi_1.0.3
## # [49] gdtools_0.2.4       partykit_1.2-15     lifecycle_1.0.1
## # [52] stringi_1.7.6       yaml_2.3.5          inum_1.0-4
## # [55] grid_4.1.3          hrbrthemes_0.8.0    promises_1.2.0.1
## # [58] forcats_0.5.1       crayon_1.5.1        lattice_0.20-45
## # [61] haven_2.5.0         splines_4.1.3       hms_1.1.1
## # [64] knitr_1.39          pillar_1.7.0        glue_1.6.2
## # [67] evaluate_0.15       pagedown_0.18       broom.helpers_1.7.0
## # [70] vctrs_0.4.1         png_0.1-7           httpuv_1.6.5
## # [73] Rttf2pt1_1.3.10     cellranger_1.1.0    gtable_0.3.0
## # [76] purrr_0.3.4         tidyr_1.2.0         xfun_0.31
```

```
## # [79] mime_0.12           libcoin_1.0-9       xtable_1.8-4
## # [82] later_1.3.0         survival_3.3-1      viridisLite_0.4.0
## # [85] tibble_3.1.7        showtextdb_3.0      ellipsis_0.3.2
```

```
  # or use message() instead of cat()
```

A.3.3.2 Other used tools

Appendix B

Data analysis health club

B.1 Environment configuration

```
knitr::opts_chunk$set(cache = FALSE)
knitr::opts_chunk$set(echo = TRUE)
# Use cache = TRUE if you want to speed up compilation
# set path
# get rmarkdown directory
caminho <- getwd()
# set working directory
setwd(caminho)
print(caminho)
```

```
## [1] "/mnt/c/nuvem/Dropbox/doutoramento/tese/3.case_study/customer_fitness"
```

```
# A function to allow for showing some of the inline code
rinline <- function(code) {
  html <- '<code  class = "r">``` `r CODE` ```</code>'
  sub("CODE", code, html)
}
```

B.2 Reticulate configuration

Reticulate allows the interoperability between Python and R

```
# load essential libraries
library(dplyr)
library(dlookr)
```

```
library(ggplot2)
library(reticulate)
#Replace by your environment, usually which python solves the problem
#conda env list also is a good option
#set environment first then call reticulate library
path_python_windows <- "C:\\Users\\sobre\\AppData\\Local\\r-miniconda\\envs\\rsurvival\\python.exe"
path_python_linux <- "/home/sobreiro/miniconda3/envs/survival/bin/python"
switch(Sys.info()[["sysname"]],
       Windows = {
            Sys.setenv(RETICULATE_PYTHON = path_python_windows)
            #call reticulate
            library(reticulate)
            #activate environment
            use_condaenv("rsurvival", required = TRUE)
                caminho_figuras <- "c:/nuvem/Dropbox/doutoramento/tese/9.tese_documento/PhD-Pedro-Sob
       },
       Linux = {
            Sys.setenv(RETICULATE_PYTHON = path_python_linux)
            library(reticulate)
            use_condaenv("survival", required = TRUE)
            caminho_figuras <- "/mnt/c/nuvem/Dropbox/doutoramento/tese/9.tese_documento/PhD-Pedro-Sob
       }
)
```

B.3 Dataset

In this case, data from 5,209 fitness customers was analysed (mean age = 27.88, SD=11.80 years) from a Portuguese fitness centre. The data was collected from software e@sport (Cedis, Portugal) between 2014 and 2017. The information retrieved was: Age of the participants in years; Sex (0-female, 1-male); Non-attendance days before dropout; Total amount billed; Average number of visits per week; Total number of visits; Weekly contracted accesses; Number of registration renewals; Number of customer referrals; Registration month; Customer enrolment duration; and status (dropout/non-dropout). Dropout event occur when customer communicate the intention to terminate the contract or did not pay the monthly fee during 60 days.

Dropout is a binary value where one represent churn and zero not churn. The dropout happens when a member does not have a payment. The survival time in the dataset is represented by the number of months the customer begin affiliated. We extracted records of 5216 customers (male n=3382, female n=1834) corresponding to the time period between 2014 and 2017.

```
library(stargazer)
library(readxl)
library(dplyr)
library(visdat)

df_members <- read_excel("data/fitness_customers.xlsx")
```

```
# rename column labels
names(df_members) <- c("id", "age", "sex", "dayswfreq", "tbilled",
                       "maccess", "freeuse", "nentries", "cfreq", "nrenewals",
                       "cref", "start_date", "months", "dropout")

# remove null values
df_members <- df_members[complete.cases(df_members), ]

names(df_members)
```

```
##  [1] "id"         "age"        "sex"        "dayswfreq"  "tbilled"
##  [6] "maccess"    "freeuse"    "nentries"   "cfreq"      "nrenewals"
## [11] "cref"       "start_date" "months"     "dropout"
```

```
# select relevant variables
df_members <- df_members %>%
select(age, sex, dayswfreq, tbilled, maccess, freeuse,
       nentries, cfreq, months, dropout)

str(df_members)
```

```
## tibble [5,210 x 10] (S3: tbl_df/tbl/data.frame)
##  $ age      : num [1:5210] 23 34 24 20 21 20 26 44 20 21 ...
##  $ sex      : num [1:5210] 1 1 0 1 1 0 1 0 0 0 ...
##  $ dayswfreq: num [1:5210] 7 328 3 41 18 38 279 45 56 4 ...
##  $ tbilled  : num [1:5210] 37.6 205.6 140 71.6 113.2 ...
##  $ maccess  : num [1:5210] 1.35 0.54 0.8 1 0.08 0.33 0.16 0.93 0.31 2.26 ...
##  $ freeuse  : num [1:5210] 0 0 0 0 0 0 0 0 0 0 ...
##  $ nentries : num [1:5210] 6 39 28 13 7 11 6 52 27 21 ...
##  $ cfreq    : num [1:5210] 7 7 7 7 7 7 7 7 7 7 ...
##  $ months   : num [1:5210] 1 19 8 3 24 10 9 15 22 3 ...
##  $ dropout  : num [1:5210] 1 0 1 1 1 1 1 1 1 1 ...
```

```
vis_dat(df_members) #check
```

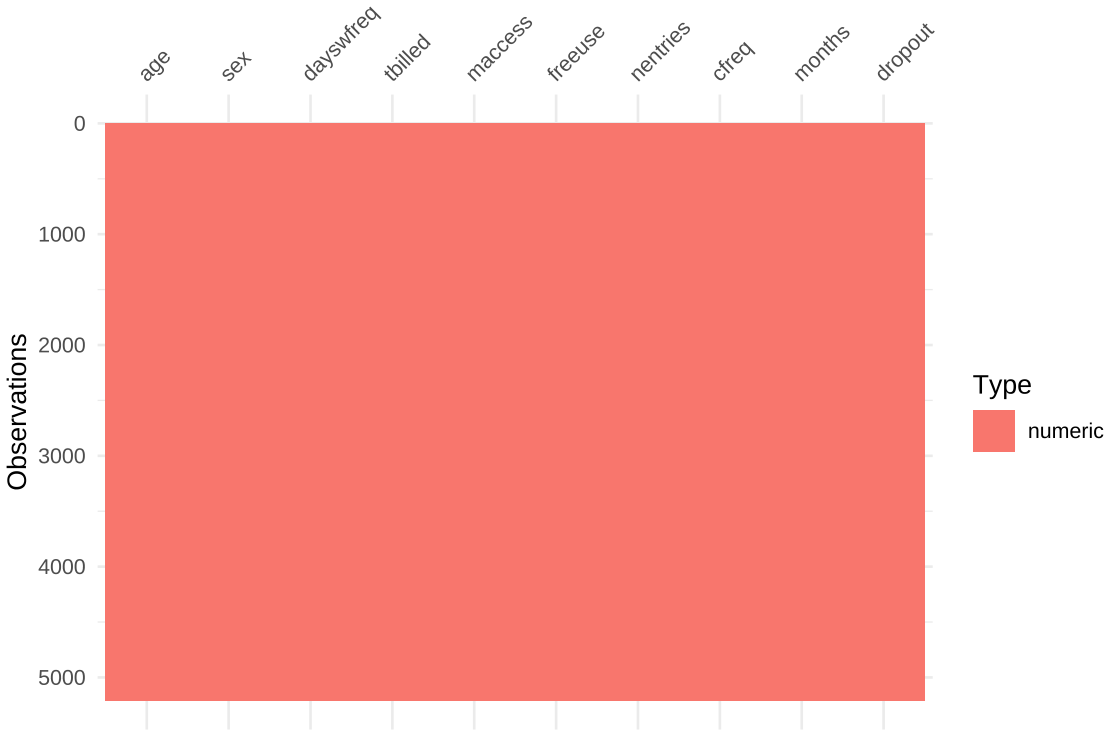


Table B.1 shows data's summary statistics. The average age is 27.9 ± 11.8, the entries are 29 ± 41.1 with an inscription period of 9 ± 8.2 months.

```
library(gtsummary)
library(kableExtra)
library(labelled)

var_label(df_members$age) <- "Age in years"
var_label(df_members$sex) <- "Male or female"

tbl <- df_members %>%
    tbl_summary(statistic = list(all_continuous() ~ "{mean} ({sd})",
                                 all_categorical() ~ "{p}%"),
                type = list(age ~ "continuous")) %>%
    add_stat_label()

as_kable_extra(tbl, booktabs = T,
               caption = "Summary statistics of features used")
```

Figure B.1 shows the distribution of the dropout considering the number of years of membership.

```
members_month <- df_members %>%
  select(months, dropout) %>%
  group_by(months, dropout) %>%
  summarize(count = n())
```

TABLE B.1: Summary statistics of features used

Characteristic	**N = 5,210**
Age in years, Mean (SD)	28 (12)
Male or female, %	35%
dayswfreq, Mean (SD)	76 (102)
tbilled, Mean (SD)	155 (162)
maccess, Mean (SD)	0.89 (0.76)
freeuse, %	4.9%
nentries, Mean (SD)	29 (41)
cfreq, %	
2	1.3%
4	2.4%
6	0.2%
7	96%
months, Mean (SD)	9 (8)
dropout, %	88%

```
members_month$dropout <- factor(members_month$dropout)
g <- ggplot(data = members_month,
            mapping = aes(x = months, y = count, linetype = dropout))
g <- g + geom_line() + labs(x = "months", y = "Number of members") +
    theme_classic()
g
```

```
# Export the file
# ggsave(file="/mnt/c/nuvem/Dropbox/doutoramento/tese/9.tese_documento/PhD-Pedro-Sobreiro/figur
```

B.3.1 Model construction

Categorical variable *sex* was converted to a dummy variables.

The random survival forest was developed using the package PySurvival (Fotso et al. 2019). The most relevant variables predicting the dropout are analysed using the log-rank test. The metric variables are transformed to categorical using the quartiles to provide a statistical comparison of groups. The survival analysis was conducted using the package Lifelines (Davidson-Pilon 2021).

PySurvival is an open source python package for Survival Analysis modeling - the modeling concept used to analyze or predict when an event is likely to happen. It is built upon the most commonly used machine learning packages such NumPy, SciPy and PyTorch. PySurvival is compatible with Python 2.7-3.7

The survival trees based model uses pysurvival random forest.

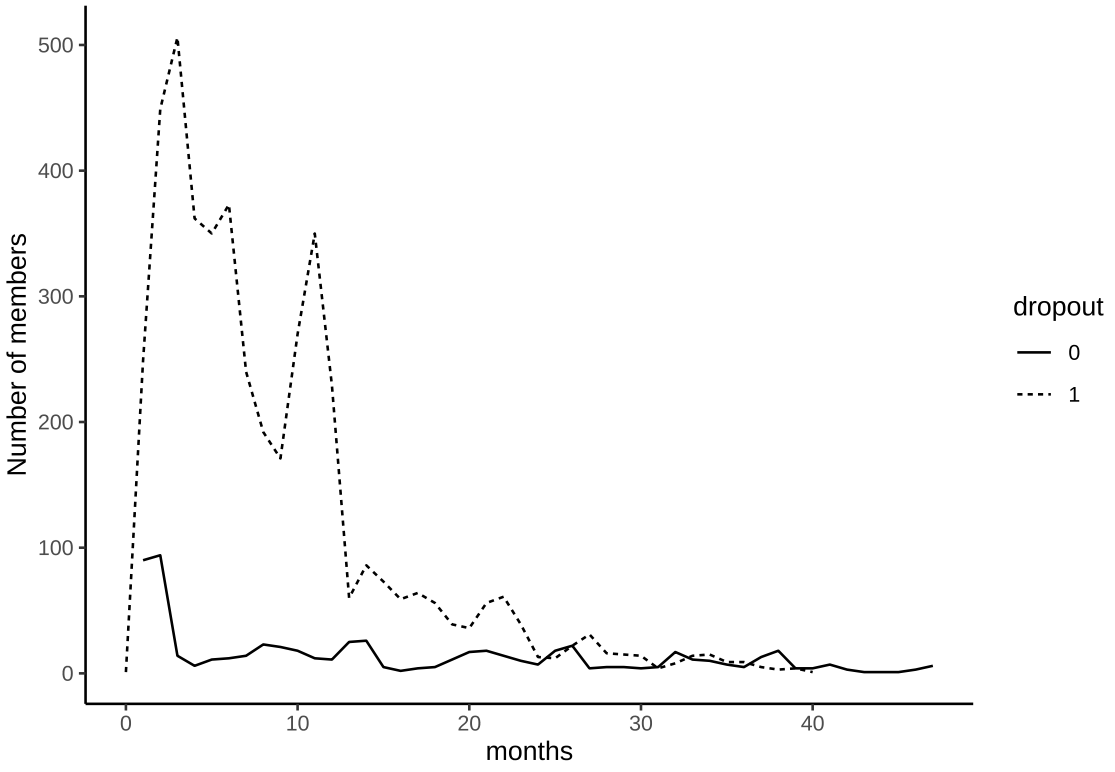


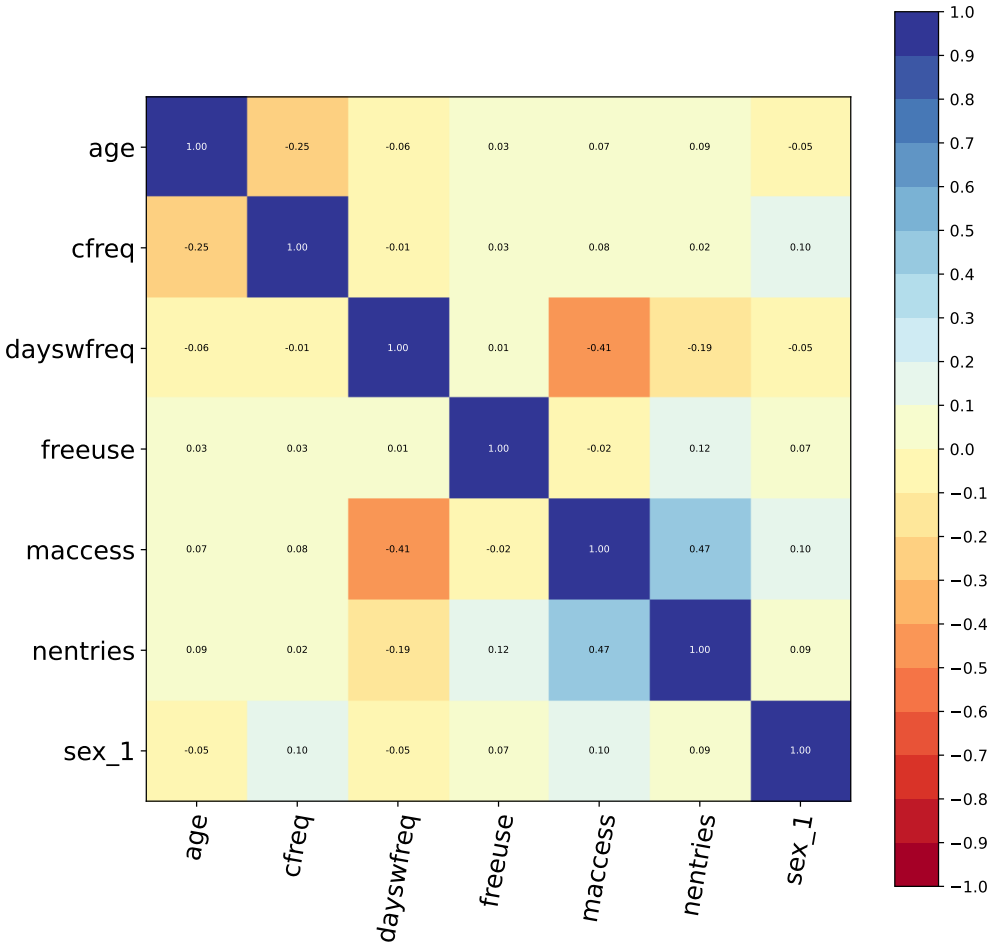
FIGURE B.1: Number of members by month

```
from pysurvival.utils.display import correlation_matrix
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

col = ['sex']

df_members = r.df_members #copy r dataframe to python

# convert to int
df_members['age']=df_members['age'].astype(int)
df_members['dayswfreq']=df_members['dayswfreq'].astype(int)
df_members['cfreq']=df_members['cfreq'].astype(int)
df_members['months']=df_members['months'].astype(int)
df_members['dropout']=df_members['dropout'].astype(int)
df_members['sex']=df_members['sex'].astype(int)

df_members = pd.get_dummies(df_members, columns=col,drop_first=True)

# Creating the time and event columns
time_column = 'months'
event_column = 'dropout'
```

```
# Extracting the features
features = np.setdiff1d(df_members.columns, [time_column, event_column] ).tolist()


correlation_matrix(df_members[features], figure_size=(10,10), text_fontsize=6)

r.df_members = df_members

#plt.plot()
```

The variable *tbilled* as an higher correlation with *nentries* (r=0.73) and was removed.

```
to_remove = ['tbilled']

#consider also the features previously removed
features = np.setdiff1d(df_members[features].columns,
                        to_remove).tolist()

df_members.drop(columns = to_remove, inplace=True)

correlation_matrix(df_members[features],
                   figure_size=(10,10),
                   text_fontsize=6)
```



The model performance was determined with the concordance probability (C-index), Brier Score (BS) and Mean Absolute Error (MAE) (Wang, Li, and Reddy 2017). The feature importance was determined calculating the difference between the true class label and noised data (Breiman 2001).

The BS is used to evaluate the predicted accuracy of the survival function at a given time t. Representing the average square distance between the survival status and the predicted survival probability, where the value 0 is the best possible outcome.

$$BS(t) = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{\left(0-\hat{S}(t,\vec{x}_i)\right)^2 \cdot \mathbb{1}_{T_i\leq t,\delta_i=1}}{\hat{G}(T_i^-)} + \frac{\left(1-\hat{S}(t,\vec{x}_i)\right)^2 \cdot \mathbb{1}_{T_i>t}}{\hat{G}(t)}\right) \tag{B.1}$$

The model should have a Brier score below 0.25. Considering that if $\forall i \in [\![1,N]\!], \hat{S}(t,\vec{x}_i) = 0.5$, then $BS(t) = 0.25$.

The table B.2 depicts the data of the survival time of the customers during the first months, the result showed that the customers of the sport club have a survival probability of 24.44% at 12 months (column p_i - likelihood probability) with a median survival time of 6 months (column estimated_survival). The survival probability at 6 months was 54.5%, representing an risk of dropout of 45.5% with a estimated survival of 6 months.

```
from lifelines import KaplanMeierFitter
kmf = KaplanMeierFitter()
T = df_members['months']
C = df_members['dropout']

kmf.fit(T,C,label="Customers")
```

```
## <lifelines.KaplanMeierFitter:"Customers", fitted with 5210 total observations, 644 right-cen
```

```
kmf.event_table.reset_index()
```

	event_at	removed	observed	censored	entrance	at_risk
0	0	1	1	0	5210	5210
1	1	339	249	90	0	5209
2	2	543	449	94	0	4870
3	3	520	506	14	0	4327
4	4	368	362	6	0	3807
5	5	361	350	11	0	3439
6	6	385	373	12	0	3078
7	7	254	240	14	0	2693
8	8	215	192	23	0	2439
9	9	192	171	21	0	2224
10	10	288	270	18	0	2032
11	11	362	350	12	0	1744
12	12	240	229	11	0	1382
13	13	85	60	25	0	1142
14	14	112	86	26	0	1057
15	15	78	73	5	0	945
16	16	61	59	2	0	867
17	17	68	64	4	0	806
18	18	61	56	5	0	738
19	19	51	39	12	0	677
20	20	53	36	17	0	626
21	21	74	56	18	0	573
22	22	75	61	14	0	499
23	23	49	39	10	0	424
24	24	20	13	7	0	375
25	25	30	12	18	0	355
26	26	44	22	22	0	325
27	27	35	31	4	0	281

```
## 28        28        21        16         5         0       246
## 29        29        20        15         5         0       225
## 30        30        18        14         4         0       205
## 31        31         9         4         5         0       187
## 32        32        25         8        17         0       178
## 33        33        25        14        11         0       153
## 34        34        25        15        10         0       128
## 35        35        16         9         7         0       103
## 36        36        14         9         5         0        87
## 37        37        18         5        13         0        73
## 38        38        21         3        18         0        55
## 39        39         8         4         4         0        34
## 40        40         5         1         4         0        26
## 41        41         7         0         7         0        21
## 42        42         3         0         3         0        14
## 43        43         1         0         1         0        11
## 44        45         1         0         1         0        10
## 45        46         3         0         3         0         9
## 46        47         6         0         6         0         6
```

```
kmf.conditional_time_to_event_
```

```
##           Customers - Conditional median duration remaining to event
## timeline
## 0.0                                                             7.0
## 1.0                                                             7.0
## 2.0                                                             7.0
## 3.0                                                             7.0
## 4.0                                                             7.0
## 5.0                                                             6.0
## 6.0                                                             6.0
## 7.0                                                             5.0
## 8.0                                                             6.0
## 9.0                                                             6.0
## 10.0                                                            6.0
## 11.0                                                            9.0
## 12.0                                                           10.0
## 13.0                                                            9.0
## 14.0                                                            9.0
## 15.0                                                            9.0
## 16.0                                                           10.0
## 17.0                                                           10.0
## 18.0                                                            9.0
## 19.0                                                            9.0
## 20.0                                                            9.0
## 21.0                                                           10.0
## 22.0                                                           11.0
```

```
## 23.0                                                    11.0
## 24.0                                                    10.0
## 25.0                                                    10.0
## 26.0                                                     9.0
## 27.0                                                     9.0
## 28.0                                                     9.0
## 29.0                                                    10.0
## 30.0                                                     9.0
## 31.0                                                     8.0
## 32.0                                                     8.0
## 33.0                                                     inf
## 34.0                                                     inf
## 35.0                                                     inf
## 36.0                                                     inf
## 37.0                                                     inf
## 38.0                                                     inf
## 39.0                                                     inf
## 40.0                                                     inf
## 41.0                                                     inf
## 42.0                                                     inf
## 43.0                                                     inf
## 45.0                                                     inf
## 46.0                                                     inf
## 47.0                                                     inf
```

```
survival_table = pd.concat([kmf.event_table.reset_index(),
                            kmf.conditional_time_to_event_.reset_index(),
                            kmf.survival_function_.reset_index()],axis=1)

survival_table.drop(['timeline'],axis=1,inplace=True)
survival_table.columns = ['event_at', 'removed', 'observed', 'censored',
                          'entrance', 'at_risk', 'estimated_survival',
                          'prob']
```

```
library(gtsummary)
library(kableExtra)
library(labelled)
survival_table <- py$survival_table
var_label(survival_table$event_at) <- "Event Month"
var_label(survival_table$removed) <- "Removed"
var_label(survival_table$observed) <- "Dropout"
var_label(survival_table$censored) <- "Censored"
var_label(survival_table$at_risk) <- "Risk of dropout"
var_label(survival_table$estimated_survival) <- "Estimated survival (months)"
var_label(survival_table$prob) <- "Survival Probability"

tbl <- head(survival_table, 25)
```

```
kbl(tbl, booktabs = T,
    format = "latex",
    caption = "Determination of the survival time probabilities",
    digits = 3) %>%
    footnote(general = paste("Removed { the sum of customers with dropout and ",
                             "that are censored; Censored { the event did not ",
                             "occur during the period of this data, ",
                             "collection; Risk of Dropout { number of",
                             "customers at risk of, dropout; pi { survival",
                             "probability; Estimated Survival - months to ",
                             "survive in the sports facility.", sep = " "),
                              threeparttable = TRUE)
```

Figure B.2 shows the Kaplan Meier survival curve customers considering the number of months of membership (x axis) and survival probability (y axis). The customer dropout is very high in the first 12 months, ranging from a survival probability of 54% after the first 6 months until 24% after 12 months.

```
import matplotlib.pyplot as plt
import matplotlib.font_manager as font_manager

plt.rcParams['figure.figsize'] = [12, 7]

fontL = font_manager.FontProperties(family='Times New Roman',weight=None,style='normal', size=10)

# fontAxis = {'family': 'Times New Roman'}

ax = kmf.plot(color='black')
ax.legend(prop=fontL)
#ax.set_xlabel('Months',fontdict=fontAxis,fontsize=12)
ax.set_xlabel('Months')
#ax.set_ylabel('Survival probability',fontdict=fontAxis,fontsize=12)
ax.set_ylabel('Survival probability')

ax.axvline(x=6,ymax=0.54,linestyle='--',color='black');ax.axhline(y=0.54,xmax=0.14,linestyle='--',col
ax.annotate("0.54",xy=(6, 0.54), xytext=(7, 0.7))

ax.axvline(x=12,ymax=0.24,linestyle='--',color='black');ax.axhline(y=0.24,xmax=0.254,linestyle='--',c
ax.annotate("0.24",xy=(12, 0.24), xytext=(12, 0.40))

ax.axvline(x=18,ymax=0.15,linestyle='--',color='black');ax.axhline(y=0.15,xmax=0.38,linestyle='--',co
ax.annotate("0.15",xy=(18, 0.15), xytext=(18, 0.30))

plt.savefig(r.caminho_figuras+'fitnessLifelinesPlot.png', dpi=300)
```

```
## findfont: Font family ['Times New Roman'] not found. Falling back to DejaVu Sans.
```

TABLE B.2: Determination of the survival time probabilities

event_at	removed	observed	censored	entrance	at_risk	estimated_survival	prob
0	1	1	0	5210	5210	7	1.000
1	339	249	90	0	5209	7	0.952
2	543	449	94	0	4870	7	0.864
3	520	506	14	0	4327	7	0.763
4	368	362	6	0	3807	7	0.691
5	361	350	11	0	3439	6	0.620
6	385	373	12	0	3078	6	0.545
7	254	240	14	0	2693	5	0.497
8	215	192	23	0	2439	6	0.457
9	192	171	21	0	2224	6	0.422
10	288	270	18	0	2032	6	0.366
11	362	350	12	0	1744	9	0.293
12	240	229	11	0	1382	10	0.244
13	85	60	25	0	1142	9	0.231
14	112	86	26	0	1057	9	0.213
15	78	73	5	0	945	9	0.196
16	61	59	2	0	867	10	0.183
17	68	64	4	0	806	10	0.168
18	61	56	5	0	738	9	0.155
19	51	39	12	0	677	9	0.147
20	53	36	17	0	626	9	0.138
21	74	56	18	0	573	10	0.125
22	75	61	14	0	499	11	0.109
23	49	39	10	0	424	11	0.099
24	20	13	7	0	375	10	0.096

Note:
Removed – the sum of customers with dropout and that are censored; Censored – the event did not occur during the period of this data, collection; Risk of Dropout – number of customers at risk of, dropout; pi – survival probability; Estimated Survival - months to survive in the sports facility.

```
plt.show()
```

```
plt.close()
```

Figure B.3 shows the survival by gender. The survival curves by gender are very similar, both types of customers present a behavior that is not very different.

```
import matplotlib.pyplot as plt

ax = plt.subplot(111)
```

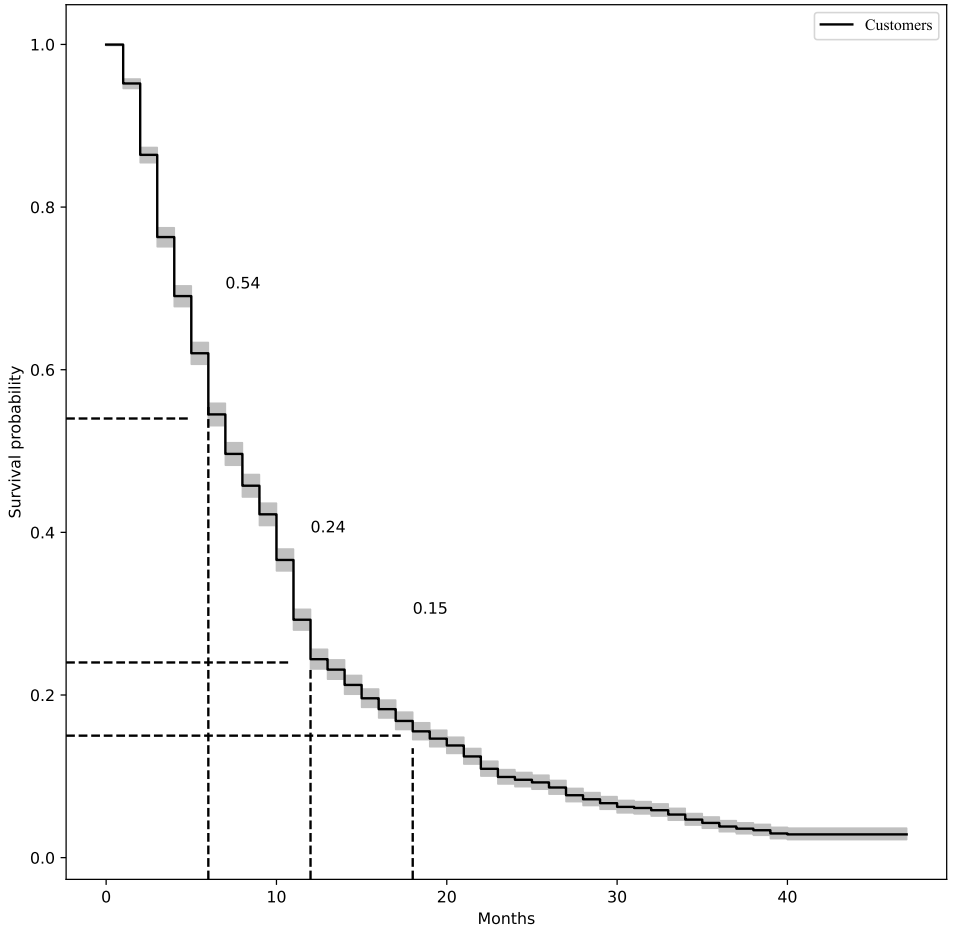


FIGURE B.2: Survival probabilities

```
plt.rcParams['figure.figsize'] = [12, 7]

for gen in df_members['sex_1'].unique():
    ix = df_members['sex_1'] == gen
    kmf.fit(T.loc[ix], C.loc[ix], label=str(gen))
    ax = kmf.plot(ax=ax)
```

```
## <lifelines.KaplanMeierFitter:"1", fitted with 1833 total observations, 234 right-censored observat
## <lifelines.KaplanMeierFitter:"0", fitted with 3377 total observations, 410 right-censored observat
```

```
plt.savefig(r.caminho_figuras+'fitnessLogrankGender.png', dpi=300)

plt.show()
```

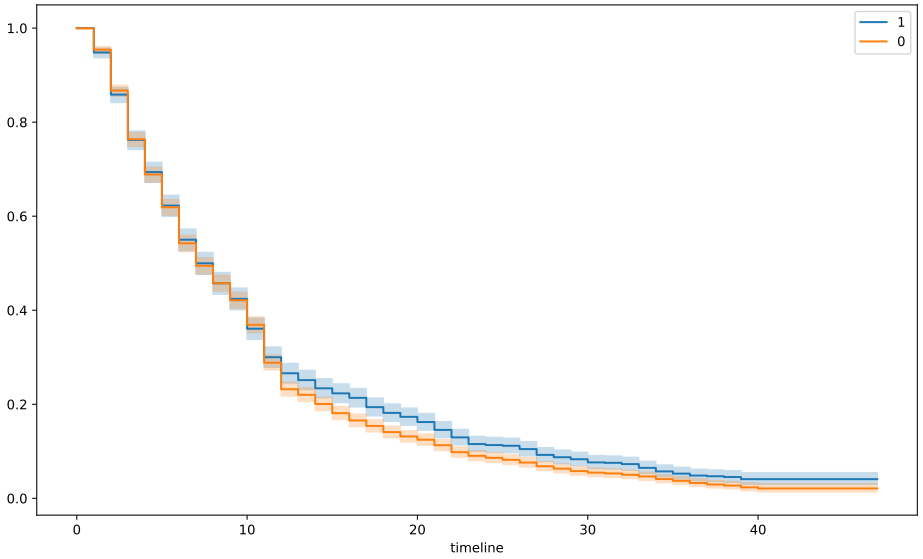


FIGURE B.3: Survival by gender

```
plt.close()
```

Figure B.4 shows the survival by contracted frequency. Customers with contracted frequency of 6 and 4 times a week have higher survival probabilities, against lower of customers with contracted frequencies of 7 and 2 times a week. Survival curves allow to explore tendencies related to survival to extract actionable knowledge.

```
import matplotlib.pyplot as plt

ax = plt.subplot(111)

plt.rcParams['figure.figsize'] = [12, 7]

for item in df_members['cfreq'].unique():
    ix = df_members['cfreq'] == item
    kmf.fit(T.loc[ix], C.loc[ix], label=str(item))
    ax = kmf.plot(ax=ax)
```

```
## <lifelines.KaplanMeierFitter:"7", fitted with 5008 total observations, 586 right-censored ob
## <lifelines.KaplanMeierFitter:"4", fitted with 125 total observations, 39 right-censored obse
## <lifelines.KaplanMeierFitter:"2", fitted with 67 total observations, 14 right-censored obser
## <lifelines.KaplanMeierFitter:"6", fitted with 10 total observations, 5 right-censored observ
```

```
plt.savefig(r.caminho_figuras+'fitnessLogrankCfreq.png', dpi=300)

plt.show()
```

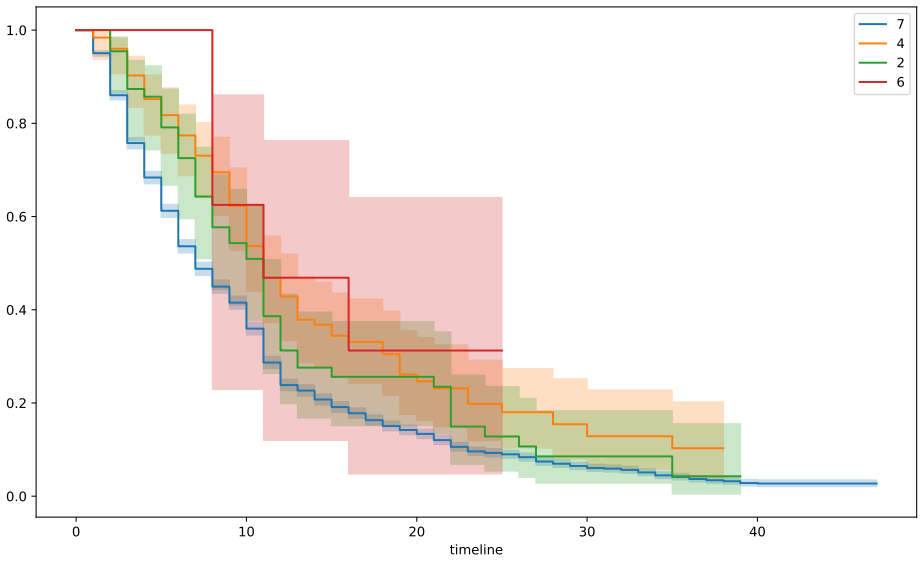


FIGURE B.4: Survival by contracted frequency

```
plt.close()
```

```
vars= ['age', 'dayswfreq', 'maccess', 'freeuse', 'nentries',
           'cfreq', 'months', 'sex_1', 'dropout']
df_regression = df_members[vars].copy()

from lifelines import CoxPHFitter
cph = CoxPHFitter()
cph.fit(df_regression,duration_col='months',event_col='dropout')
```

```
## <lifelines.CoxPHFitter: fitted with 5210 total observations, 644 right-censored observations>
```

```
cph.check_assumptions(df_regression)
```

```
## The ``p_value_threshold`` is set at 0.01. Even under the null hypothesis of no violations, some
## covariates will be below the threshold by chance. This is compounded when there are many covariate
## Similarly, when there are lots of observations, even minor deviances from the proportional hazard
## assumption will be flagged.
##
## With that in mind, it's best to use a combination of statistical tests and visual tests to determi
## the most serious violations. Produce visual plots using ``check_assumptions(..., show_plots=True)'
## and looking for non-constant lines. See link [A] below for a full example.
##
## <lifelines.StatisticalResult: proportional_hazard_test>
##  null_distribution = chi squared
```

```
## degrees_of_freedom = 1
##              model = <lifelines.CoxPHFitter: fitted with 5210 total observations, 644 right-
##          test_name = proportional_hazard_test
##
## ---
##                 test_statistic      p  -log2(p)
## age       km             23.11 <0.005     19.32
##           rank           24.14 <0.005     20.09
## cfreq     km             42.64 <0.005     33.82
##           rank           44.39 <0.005     35.11
## dayswfreq km            912.89 <0.005    663.75
##           rank          891.96 <0.005    648.64
## freeuse   km            128.79 <0.005     96.75
##           rank          125.06 <0.005     94.04
## maccess   km              9.63 <0.005      9.03
##           rank            8.62 <0.005      8.23
## nentries  km           1538.23 <0.005       inf
##           rank         1516.97 <0.005       inf
## sex_1     km              3.93   0.05      4.39
##           rank            3.35   0.07      3.90
##
##
## 1. Variable 'age' failed the non-proportional test: p-value is <5e-05.
##
##    Advice 1: the functional form of the variable 'age' might be incorrect. That is, there ma
## non-linear terms missing. The proportional hazard test used is very sensitive to incorrect
## functional forms. See documentation in link [D] below on how to specify a functional form.
##
##    Advice 2: try binning the variable 'age' using pd.cut, and then specify it in `strata=['a
## ...]` in the call in `.fit`. See documentation in link [B] below.
##
##    Advice 3: try adding an interaction term with your time variable. See documentation in li
## below.
##
##
## 2. Variable 'dayswfreq' failed the non-proportional test: p-value is <5e-05.
##
##    Advice 1: the functional form of the variable 'dayswfreq' might be incorrect. That is, th
## be non-linear terms missing. The proportional hazard test used is very sensitive to incorrec
## functional forms. See documentation in link [D] below on how to specify a functional form.
##
##    Advice 2: try binning the variable 'dayswfreq' using pd.cut, and then specify it in
## `strata=['dayswfreq', ...]` in the call in `.fit`. See documentation in link [B] below.
##
##    Advice 3: try adding an interaction term with your time variable. See documentation in li
## below.
##
```

```
##
## 3. Variable 'maccess' failed the non-proportional test: p-value is 0.0019.
##
##    Advice 1: the functional form of the variable 'maccess' might be incorrect. That is, there may
## non-linear terms missing. The proportional hazard test used is very sensitive to incorrect
## functional forms. See documentation in link [D] below on how to specify a functional form.
##
##    Advice 2: try binning the variable 'maccess' using pd.cut, and then specify it in
## `strata=['maccess', ...]` in the call in `.fit`. See documentation in link [B] below.
##
##    Advice 3: try adding an interaction term with your time variable. See documentation in link [C]
## below.
##
##
## 4. Variable 'freeuse' failed the non-proportional test: p-value is <5e-05.
##
##    Advice: with so few unique values (only 2), you can include `strata=['freeuse', ...]` in the ca
## in `.fit`. See documentation in link [E] below.
##
## 5. Variable 'nentries' failed the non-proportional test: p-value is <5e-05.
##
##    Advice 1: the functional form of the variable 'nentries' might be incorrect. That is, there may
## be non-linear terms missing. The proportional hazard test used is very sensitive to incorrect
## functional forms. See documentation in link [D] below on how to specify a functional form.
##
##    Advice 2: try binning the variable 'nentries' using pd.cut, and then specify it in
## `strata=['nentries', ...]` in the call in `.fit`. See documentation in link [B] below.
##
##    Advice 3: try adding an interaction term with your time variable. See documentation in link [C]
## below.
##
##
## 6. Variable 'cfreq' failed the non-proportional test: p-value is <5e-05.
##
##    Advice: with so few unique values (only 4), you can include `strata=['cfreq', ...]` in the call
## in `.fit`. See documentation in link [E] below.
##
## ---
## [A]  https://lifelines.readthedocs.io/en/latest/jupyter_notebooks/Proportional%20hazard%20assumpti
## [B]  https://lifelines.readthedocs.io/en/latest/jupyter_notebooks/Proportional%20hazard%20assumpti
## [C]  https://lifelines.readthedocs.io/en/latest/jupyter_notebooks/Proportional%20hazard%20assumpti
## [D]  https://lifelines.readthedocs.io/en/latest/jupyter_notebooks/Proportional%20hazard%20assumpti
## [E]  https://lifelines.readthedocs.io/en/latest/jupyter_notebooks/Proportional%20hazard%20assumpti
##
## []
##
## /home/sobreiro/miniconda3/envs/survival/lib/python3.7/site-packages/lifelines/fitters/mixins.py:10
```

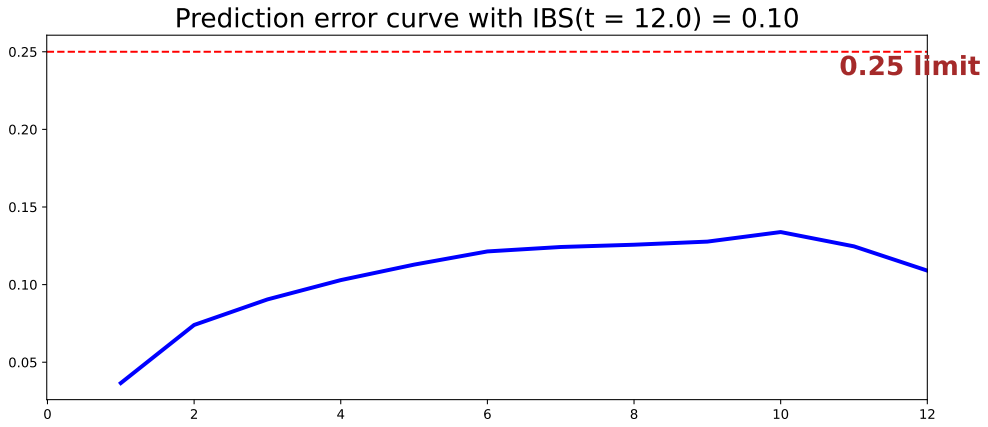


FIGURE B.5: Model performance

```
##   for variable in self.params_.index & (columns or self.params_.index):
```

The proportional hazard assumptions failed in the following variables: age p<0.01, cfreq p<0.01, dayswfreq p<0.01, tbilled p<0.01, freeuse p<0.01, nentries p<0.01.

Survival curves allow to explore tendencies related to survival to extract actionable knowledge.

```
from pysurvival.models.survival_forest import RandomSurvivalForestModel
from sklearn.model_selection import train_test_split
from pysurvival.utils.metrics import concordance_index
from pysurvival.utils.display import integrated_brier_score
from pysurvival.utils.display import compare_to_actual

X = df_members.copy()
t = df_members['months']
e = df_members['dropout']
X.drop(axis=1, columns=['months','dropout'],inplace=True)

X_train, X_test, t_train, t_test, e_train, e_test = train_test_split(X, t, e, test_size=0.3, rando

# Fitting the model
csf = RandomSurvivalForestModel(num_trees=20)
csf.fit(X_train, t_train, e_train, max_features='sqrt', max_depth=5, min_node_size=20, seed = 1)
```

```
## RandomSurvivalForestModel
```

```
c_index = concordance_index(csf, X_test, t_test, e_test)
ibs = integrated_brier_score(csf, X_test, t_test, e_test, t_max=12, figure_size=(12,5))
#plt.savefig(r.caminho_figuras+'create_model1_fitness.png', dpi=300)
```

The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as an average absolute error of 7.5 customers (figure B.6).

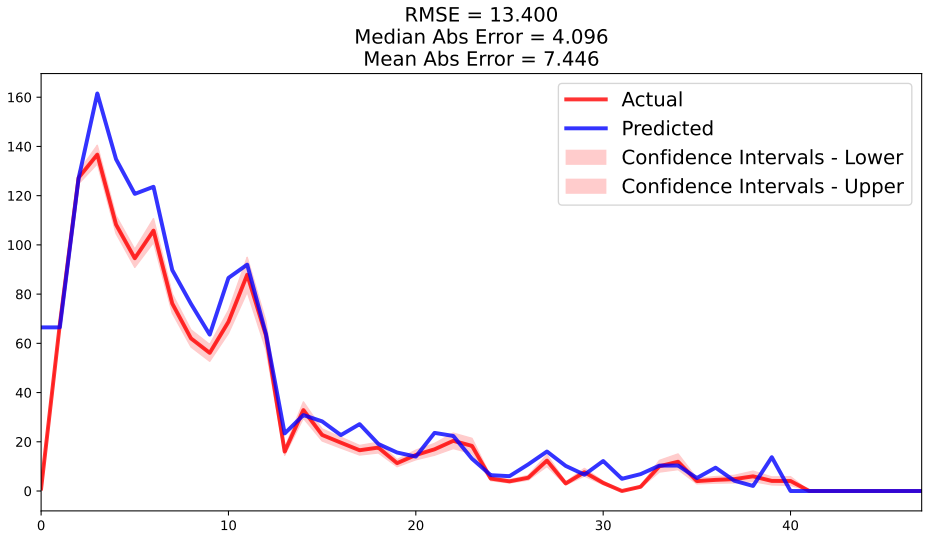


FIGURE B.6: Conditional survival forest

TABLE B.3: Features importance in the survival model

feature	importance	pct_importance
nentries	9.3451300	0.3001667
dayswfreq	8.2191218	0.2639992
freeuse	4.6434617	0.1491486
maccess	3.6848084	0.1183565
age	2.7878378	0.0895457
cfreq	1.8425762	0.0591838
sex_1	0.6101968	0.0195996

```
results = compare_to_actual(csf, X_test, t_test, e_test, is_at_risk = False,  figure_size=(12, 6), metr
```

Table B.3 shows features importance calculated according (Breiman 2001), where the percent increase in misclassification rate as compared to the out-of-bag rate (with all variables intact), out-of-bag is a bootstrap aggregating (subsampling with replacement to create training samples for the model to learn from) where two independent sets are created. One set, the bootstrap sample, data chosen to be "in-the-bag" by sampling with replacement and the out-of-bag is all data not chosen in the sampling process. The most important variable is the $dayswfreq$, followed by *tbilled* and *nentries*, compared with the $cfreq$, age, and sex.

```
tbl <- py$csf$variable_importance_table
kbl(tbl, booktabs = T, caption = "Features importance in the survival model")
```

The prediction is very similar to the actual value. The model accuracy is very high with a root mean square error of 13. The mean absolute error mean was 7.45 customers, and the median absolute error was 4.1.

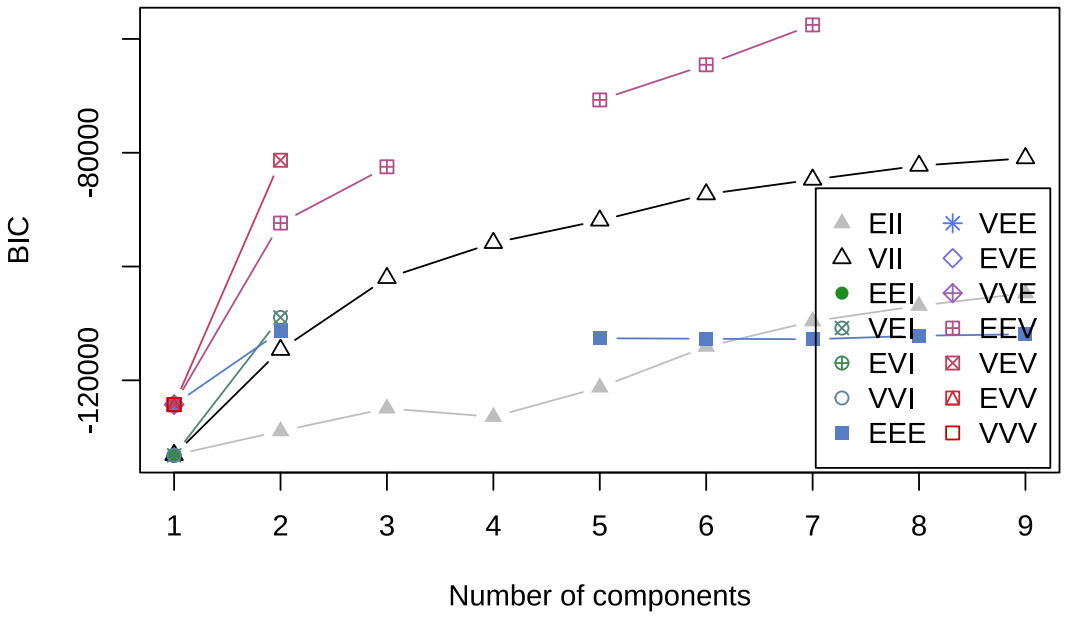


FIGURE B.7: Analysis number of clusters

B.3.2 Survival trees based model with clusters

In this approach we have created clusters and applied the survival trees within each cluster. The determination of the clusters using the BIC criterion where the EEV model: 7 clusters -57530.45; 6 clusters -65539.314; and 5 clusters -70716.38 figure B.7 shows the determination of the number of clusters using BIC, also the elbow analysis available in figure B.8 when the curve flattened after 7 clusters. Therefore, an optimal number of clusters was considered of nine.

```
library(mclust)
y <- scale(py$df_members)

set.seed(0) # to make reproducible

bic <- mclustBIC(y)
# Best model using the BIC criteria
#bic
plot(bic, what = "BIC")
```

```
summary(bic)
```

```
## Best BIC values:
##              EEV,7      EEV,6     EEV,5
## BIC      -57530.45 -64539.314 -70716.38
```

```
## BIC diff      0.00  -7008.864 -13185.93
```

```
library(NbClust)
nb <- NbClust(y, diss = NULL, distance = "euclidean",
              min.nc = 2, max.nc = 5, method = "kmeans",
              index = "all", alphaBeale = 0.1)
hist(nb$Best.nc[1, ], breaks = max(na.omit(nb$Best.nc[1, ])))
```

```
from sklearn.cluster import KMeans
from sklearn import preprocessing
from scipy.spatial.distance import cdist
import matplotlib.pyplot as plt
import numpy as np

#Finding optimal no. of clusters
clusters=range(1,20)
meanDistortions=[]

for k in clusters:
    model=KMeans(n_clusters=k)
    model.fit(df_members)
    prediction=model.predict(df_members)
    meanDistortions.append(sum(np.min(cdist(df_members, model.cluster_centers_, 'euclidean'), axis=1)
```

```
## KMeans(n_clusters=1)
## KMeans(n_clusters=2)
## KMeans(n_clusters=3)
## KMeans(n_clusters=4)
## KMeans(n_clusters=5)
## KMeans(n_clusters=6)
## KMeans(n_clusters=7)
## KMeans()
## KMeans(n_clusters=9)
## KMeans(n_clusters=10)
## KMeans(n_clusters=11)
## KMeans(n_clusters=12)
## KMeans(n_clusters=13)
## KMeans(n_clusters=14)
## KMeans(n_clusters=15)
## KMeans(n_clusters=16)
## KMeans(n_clusters=17)
## KMeans(n_clusters=18)
## KMeans(n_clusters=19)
```

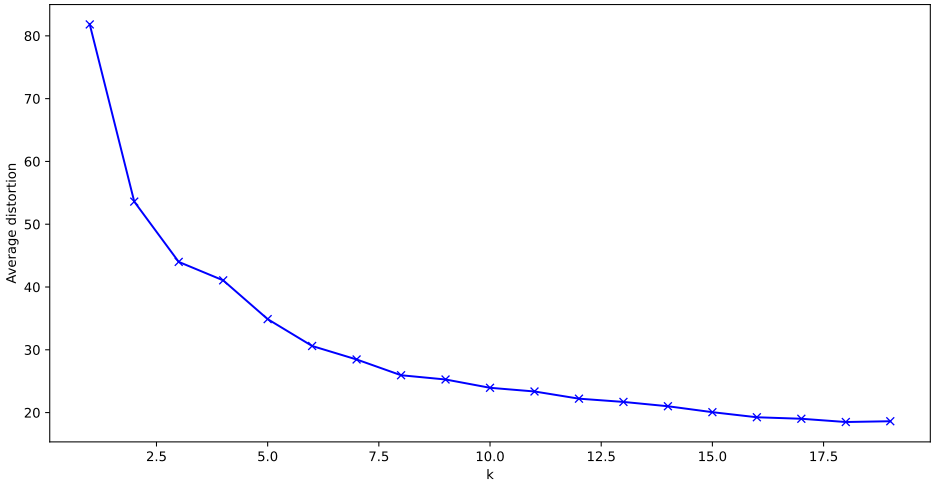


FIGURE B.8: Elbow analysis

```
plt.plot(clusters, meanDistortions, 'bx-')
plt.xlabel('k')
plt.ylabel('Average distortion')
plt.show();
```

```
# we are going to use random_state = 0 for the centroid
# initialization being deterministic allowing a better
# reproducibility
cluster = KMeans(n_clusters=7, random_state=0)

cluster.fit(df_members)
```

```
## KMeans(n_clusters=7, random_state=0)
```

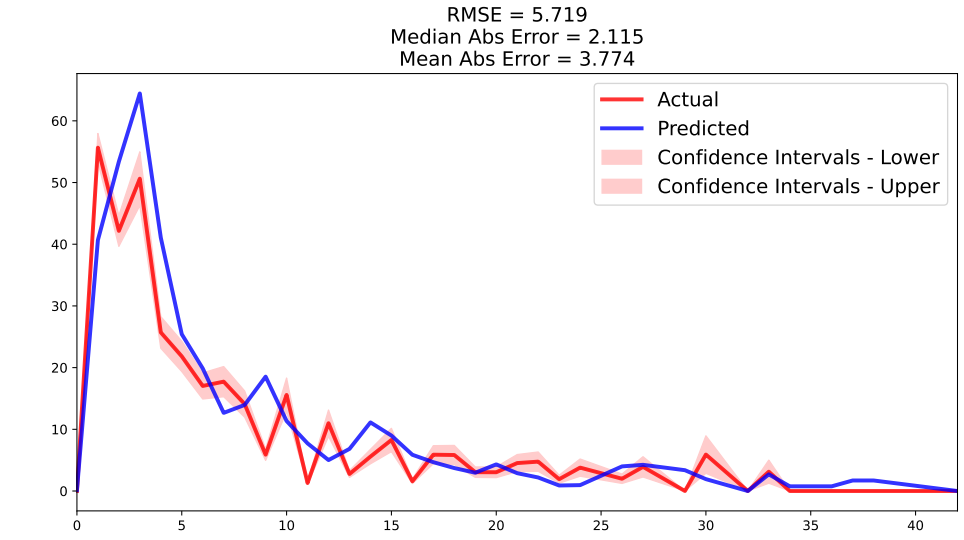
```
df_members['cluster']=cluster.predict(df_members)
print(df_members.cluster.value_counts());
```

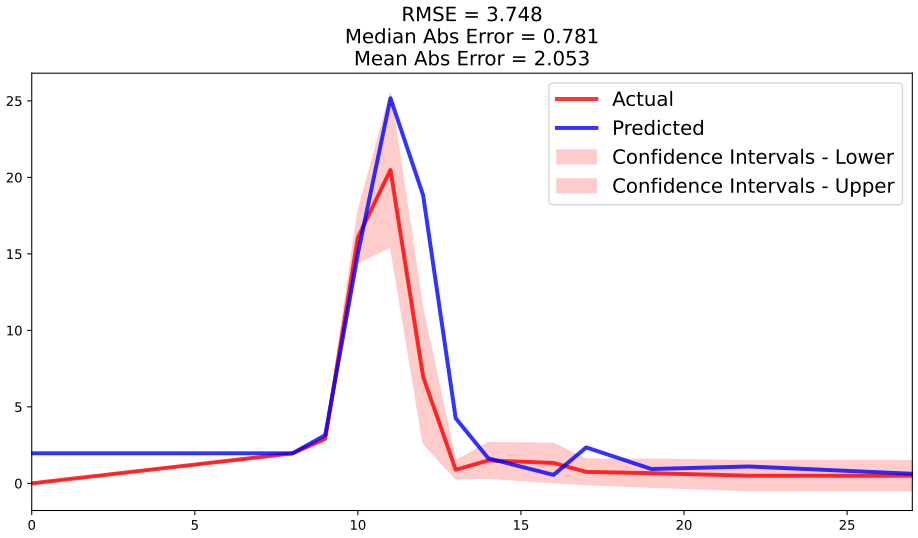
```
## 0    1683
## 5    1590
## 2     800
## 6     576
## 1     427
## 3      93
## 4      40
## Name: cluster, dtype: int64
```

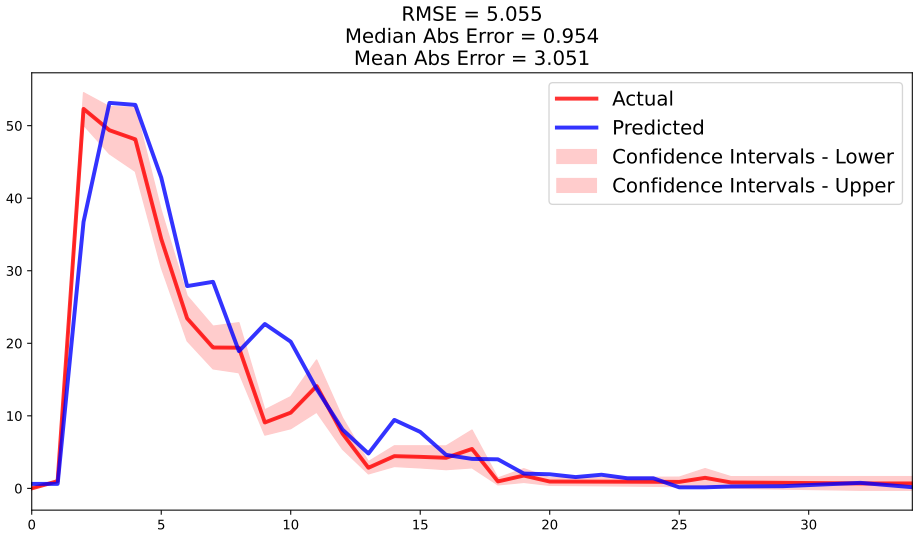
```
from sklearn.decomposition import PCA

plt.rcParams['figure.figsize'] = [13, 4]

pca=PCA(n_components=2)
df_members['X_pca']=pca.fit_transform(df_members)[:,0] #todas as linhas da primeira coluna com redução
df_members['Y_pca']=pca.fit_transform(df_members)[:,1] #todas as linhas da segunda coluna com redução

fig, ax = plt.subplots()
for cluster in df_members.cluster.unique():
    x = df_members['X_pca'].loc[df_members.cluster == cluster]
    y = df_members['Y_pca'].loc[df_members.cluster == cluster]
    ax.scatter(x, y, label=cluster, alpha=1, edgecolors='none');

ax.legend()
ax.grid(True)
ax.set_title('Customers clusters');
plt.show();
```

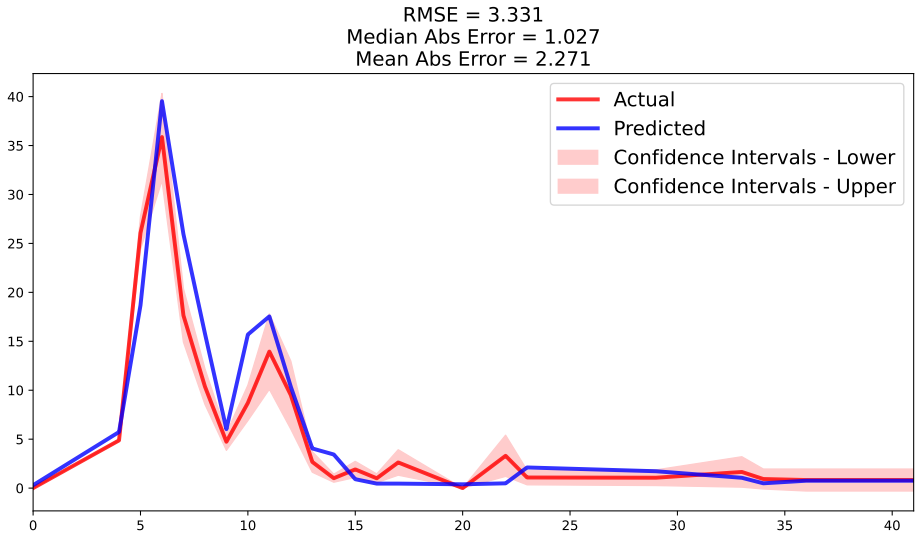
The model accuracy is very high in the first years. The prediction is very similar to the actual value. The absolute error mean of 6 customers.

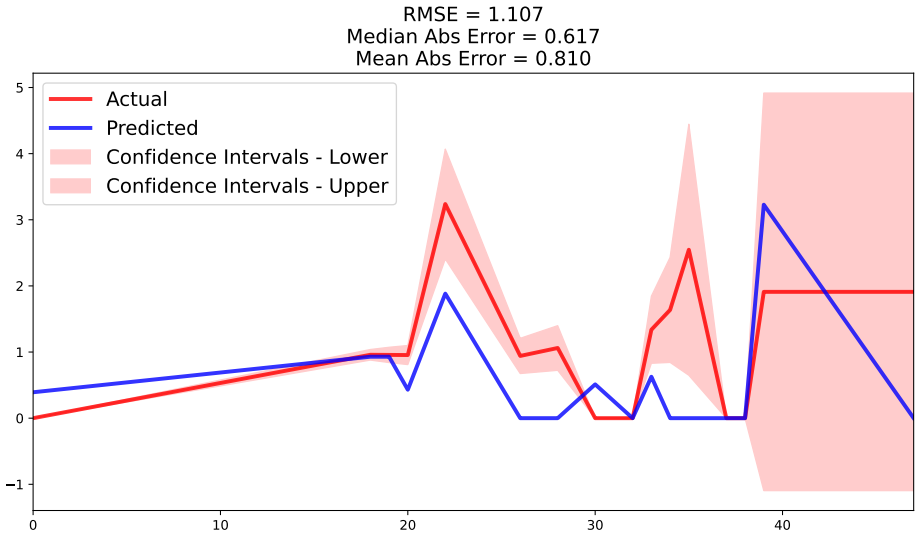
```
## RandomSurvivalForestModel
## RandomSurvivalForestModel
## RandomSurvivalForestModel
## RandomSurvivalForestModel
## RandomSurvivalForestModel
## RandomSurvivalForestModel
## RandomSurvivalForestModel
```

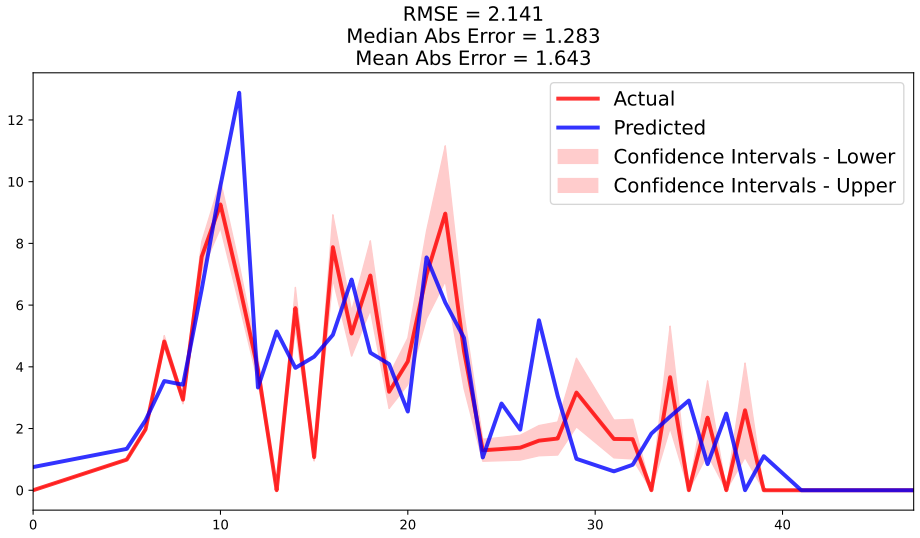

RMSE = 5.719
Median Abs Error = 2.115
Mean Abs Error = 3.774
Actual
Predicted
Confidence Intervals - Lower
Confidence Intervals - Upper


RMSE = 3.748
Median Abs Error = 0.781
Mean Abs Error = 2.053
Actual
Predicted
Confidence Intervals - Lower
Confidence Intervals - Upper


RMSE = 5.055
Median Abs Error = 0.954
Mean Abs Error = 3.051
Actual
Predicted
Confidence Intervals - Lower
Confidence Intervals - Upper


RMSE = 3.331
Median Abs Error = 1.027
Mean Abs Error = 2.271
Actual
Predicted
Confidence Intervals - Lower
Confidence Intervals - Upper


RMSE = 1.107
Median Abs Error = 0.617
Mean Abs Error = 0.810
Actual
Predicted
Confidence Intervals - Lower
Confidence Intervals - Upper


RMSE = 2.141
Median Abs Error = 1.283
Mean Abs Error = 1.643
Actual
Predicted
Confidence Intervals - Lower
Confidence Intervals - Upper

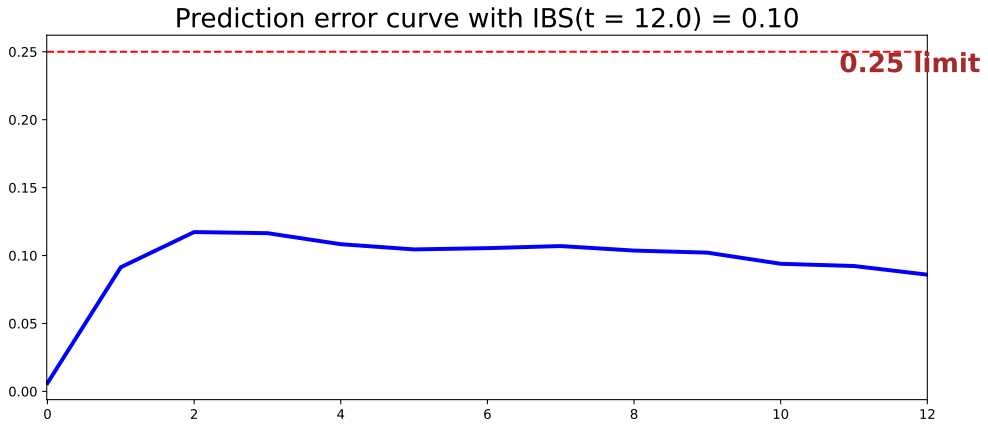
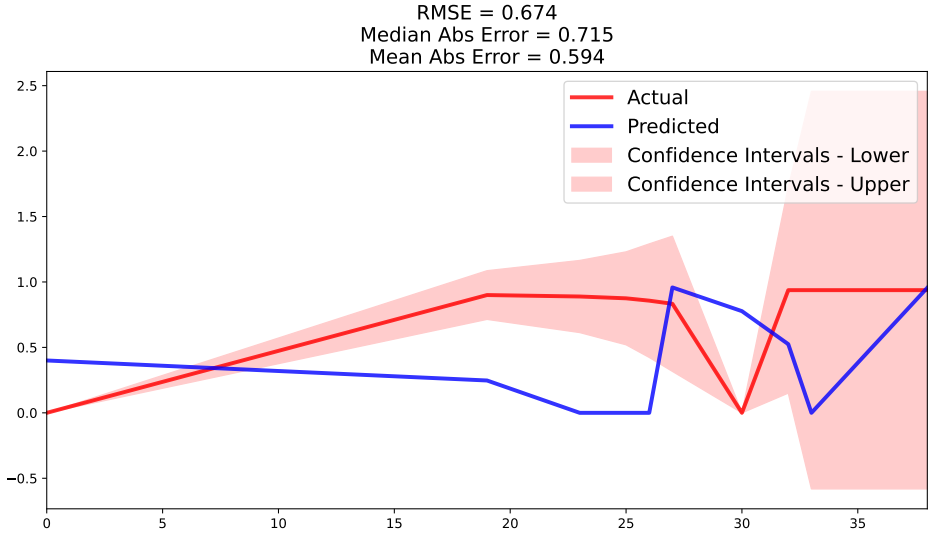


FIGURE B.9: Model performance cluster 0



The performance of the cluster 0 the IBS presents an accuracy of 0.10 (figure B.11) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 3.5 customers, the median absolute error was 1.655 and the Root Mean Square Error of 5.285 (figure B.12).

The features importance in the survival model cluster 0 (table B.5) identifies the three most relevant features to predict survival *maccess*, *nentries*, and $daysw freq$. The features with lower relevance were $freeuse$, $cfreq$, and *sex*.

The performance of the cluster 1 the IBS presents an accuracy of 0.04 (figure B.11) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 2.149 customers, the median absolute error was 0.698 and the Root Mean Square Error of 3.999 (figure B.12).

The features importance in the survival model cluster 1 (table B.5) identifies the three most relevant features to predict survival *nentries*, $daysw freq$, and *age*. The features with lower relevance were *maccess*, $cfreq$, and *sex*.

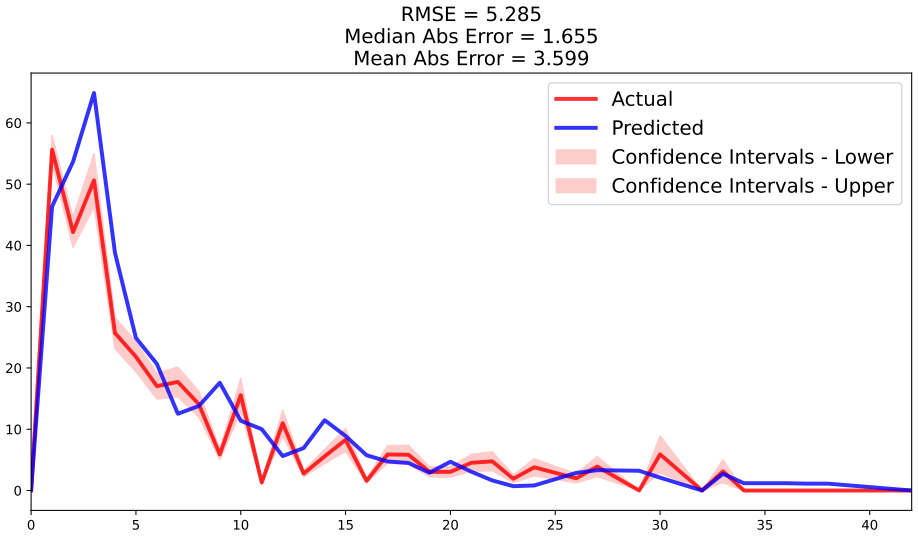


FIGURE B.10: Conditional survival forest cluster 0

TABLE B.4: Features importance in the survival model with cluster 0

feature	importance	pct_importance
maccess	11.108	0.387
nentries	6.078	0.212
dayswfreq	4.558	0.159
age	2.599	0.091
freeuse	2.498	0.087
cfreq	1.848	0.064
sex_1	-0.505	0.000

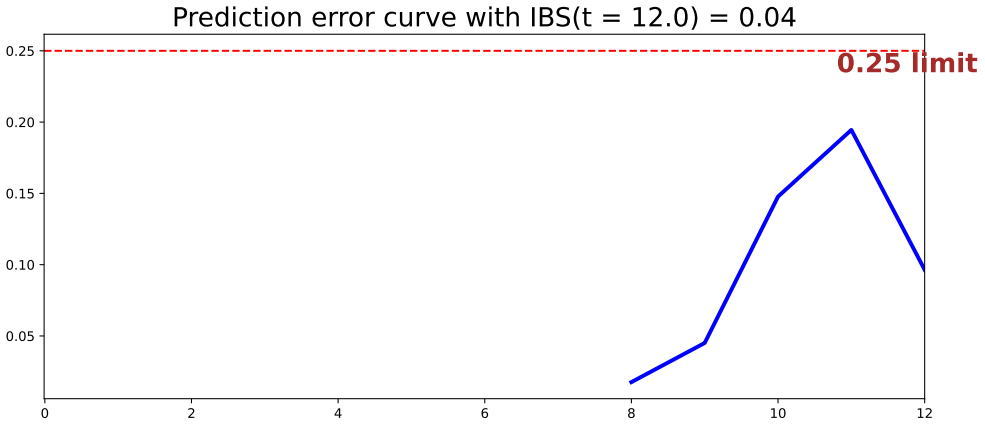


FIGURE B.11: Model performance cluster 1

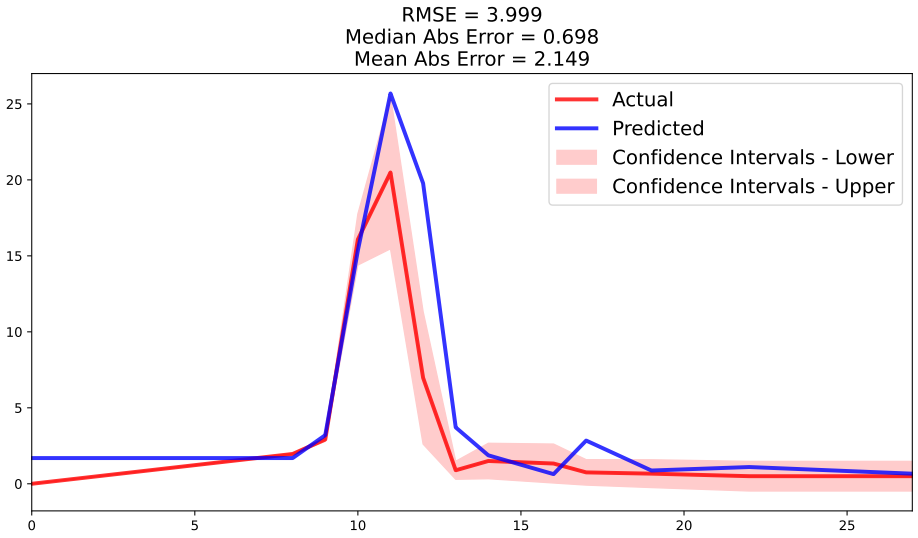


FIGURE B.12: Conditional survival forest cluster 1

TABLE B.5: Features importance in the survival model with cluster 1

feature	importance	pct_importance
nentries	5.126	0.446
dayswfreq	2.490	0.217
age	1.538	0.134
freeuse	1.210	0.105
maccess	1.123	0.098
cfreq	0.000	0.000
sex_1	-1.516	0.000

TABLE B.6: Features importance in the survival model with cluster 2

feature	importance	pct_importance
nentries	8.008	0.415
dayswfreq	3.630	0.188
maccess	3.302	0.171
age	2.546	0.132
freeuse	1.824	0.094
cfreq	0.000	0.000
sex_1	-1.457	0.000

The performance of the cluster 2 the IBS presents an accuracy along time 0.09 (figure B.13) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 2.229 customers, the median absolute error was 1.077 and the Root Mean Square Error of 3.24 (figure B.14). The features importance in the survival model cluster 2 (table B.6) identifies the three most relevant features to predict survival *nentries*, $daysw freq$, and *maccess*. The least relevant were $freeuse$, $cfreq$, and sex.

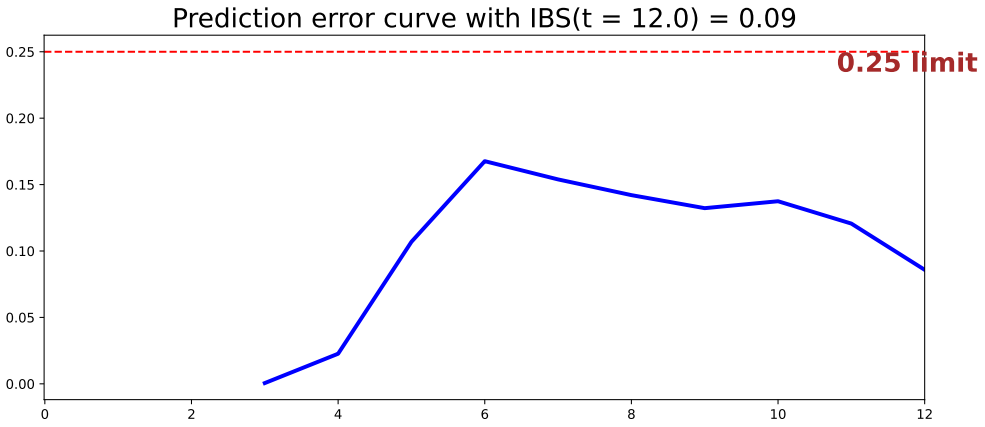


FIGURE B.13: Model performance cluster 2

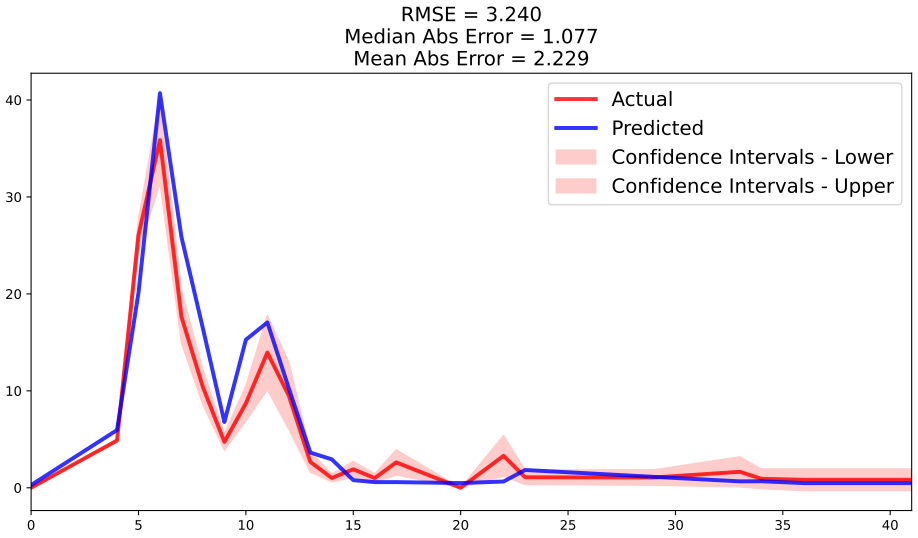


FIGURE B.14: Conditional survival forest cluster 2

The performance of the cluster 3 the IBS presents an accuracy along time 0.00 (figure B.15) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 0.821 customers, the median absolute error was 0.699 and the Root Mean Square Error of 1.117 (figure B.16). The features importance in the survival model cluster 3 (table B.7) identifies the three most relevant features to predict survival $dayswfreq$, $maccess$, and $nentries$. The least relevant were $freeuse$, $cfreq$, and sex.

The performance of the cluster 4 the IBS presents an accuracy along time NA (cluster with few members wasn't possible to calculate) (figure A.8) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 0.594 customers, the median absolute error was 0.715 and the Root Mean Square Error of 0.674 (figure B.17). The features importance in the survival model cluster 4 (table B.8).

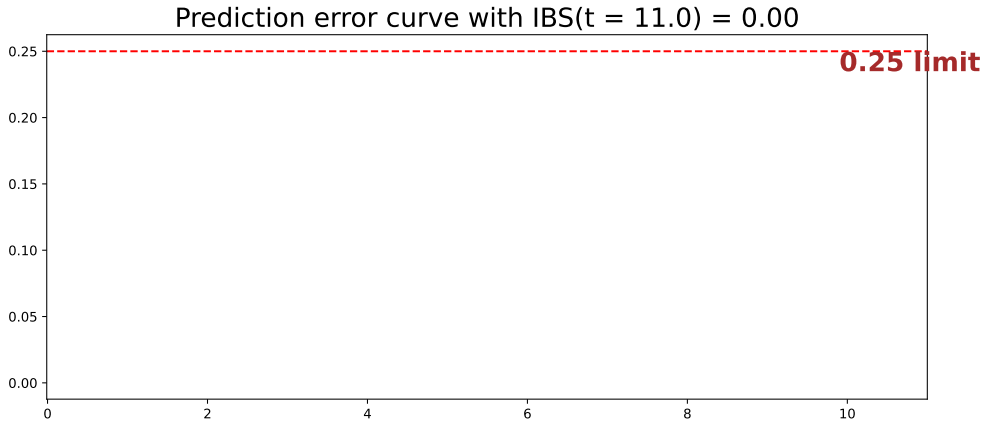


FIGURE B.15: Model performance cluster 3

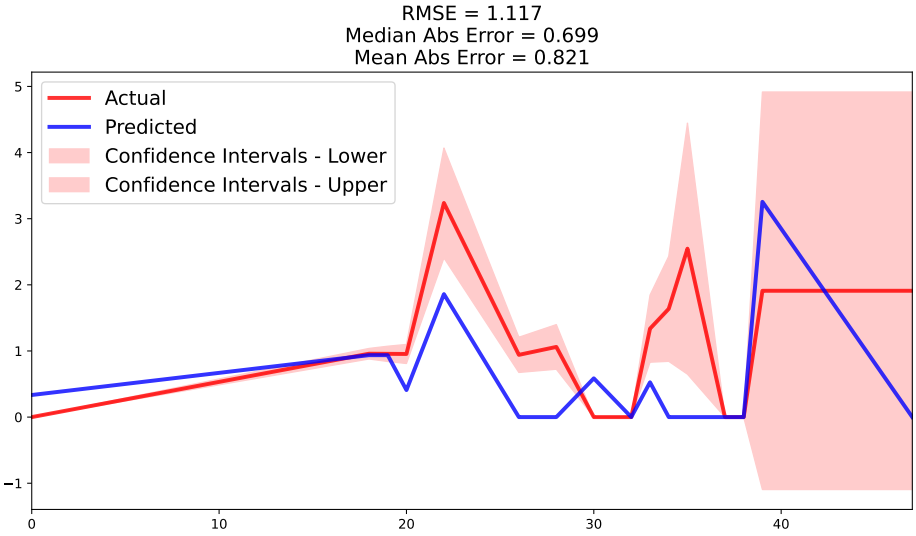


FIGURE B.16: Conditional survival forest cluster 3

TABLE B.7: Features importance in the survival model with cluster 3

feature	importance	pct_importance
dayswfreq	2.181	0.312
maccess	1.759	0.252
nentries	1.579	0.226
age	1.470	0.210
freeuse	0.000	0.000
cfreq	0.000	0.000
sex_1	-1.026	0.000

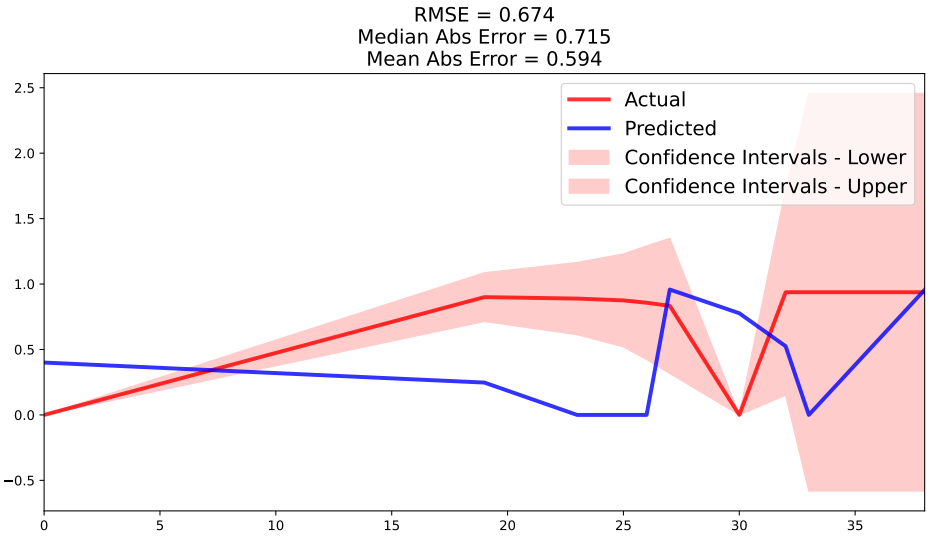


FIGURE B.17: Conditional survival forest cluster 4

TABLE B.8: Features importance in the survival model with cluster 4

feature	importance	pct_importance
age	0	NaN
dayswfreq	0	NaN
maccess	0	NaN
freeuse	0	NaN
nentries	0	NaN
cfreq	0	NaN
sex_1	0	NaN

TABLE B.9: Features importance in the survival model with cluster 5

feature	importance	pct_importance
nentries	6.234	0.305
maccess	5.408	0.265
age	3.074	0.150
dayswfreq	2.934	0.143
freeuse	2.230	0.109
cfreq	0.567	0.028
sex_1	-1.009	0.000

The performance of the cluster 5 the IBS presents an accuracy along time 0.10 (figure B.18) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 3.085 customers, the median absolute error was 1.064 and the Root Mean Square Error of 4.482 (figure B.19). The features importance in the survival model cluster 5 (table B.9) identifies the three most relevant features to predict survival *nentries*, *maccess*, and *age*. The least relevant were $freeuse$, $cfreq$, and *sex*.

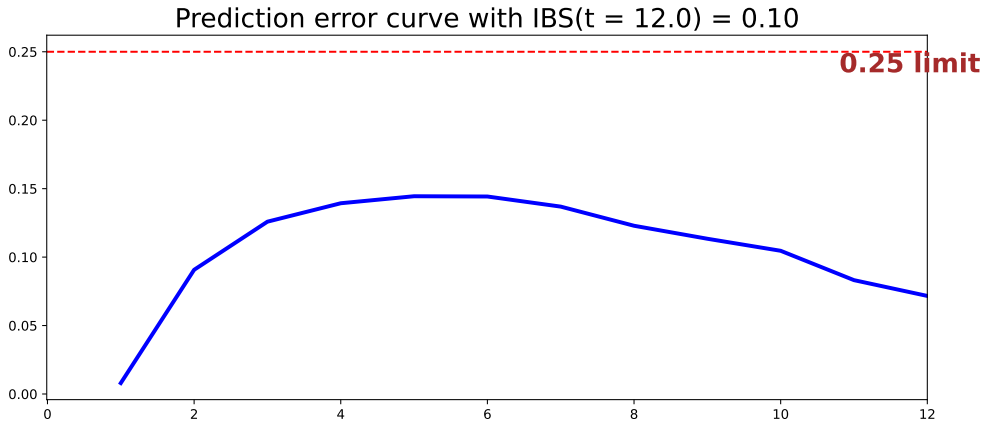


FIGURE B.18: Model performance cluster 5

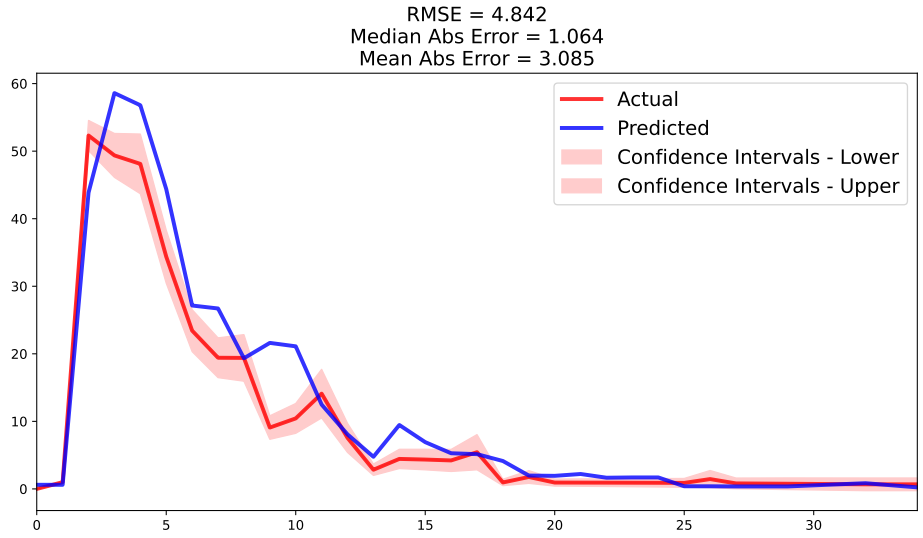


FIGURE B.19: Conditional survival forest cluster 5

The performance of the cluster 6 the IBS presents an accuracy along time 0.04 (figure B.20) along all time. The actual versus predicted model presents the actual and predicted customers which dropout during the 40 months, which as a mean absolute error of 1.6 customers, the median absolute error was 1.306 and the Root Mean Square Error of 2.075 (figure B.21). The features importance in the survival model cluster 6 (table B.10) identifies the three most relevant features to predict survival *maccess*, $daysw freq$, and *nentries*. The least relevant were $freeuse$, $cfreq$, and *sex*.

B.3.2.1 Model Comparison

Table B.11 shows the performance of both approaches, with and without clusters. The RMSE, mean and median in the clusters is lower than not using clusters to predict the survival time until dropout. Overall, the performance improved. The performance is also better using mean and median.

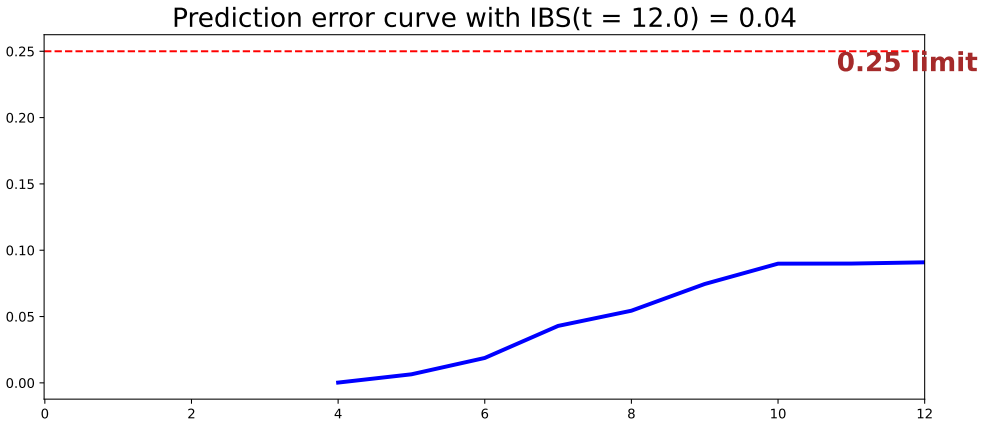


FIGURE B.20: Model performance cluster 6

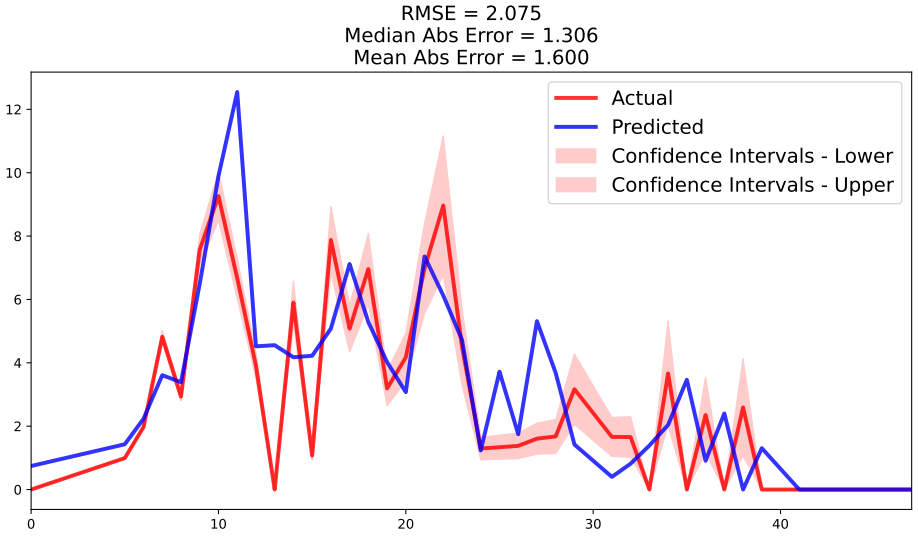


FIGURE B.21: Conditional survival forest cluster 6

TABLE B.10: Features importance in the survival model with cluster 6

feature	importance	pct_importance
maccess	7.360	0.310
dayswfreq	4.894	0.206
nentries	4.778	0.201
age	4.538	0.191
freeuse	2.177	0.092
cfreq	0.000	0.000
sex_1	-1.806	0.000

Table B.11: Performance of prediction in each cluster

cluster	rmse	mean	median
0	5.719	2.115	3.774
1	3.748	0.781	2.053
2	3.331	1.027	2.271
3	1.107	0.617	0.810
4	0.674	0.715	0.594
5	5.055	0.954	3.051
6	2.141	1.283	1.643
w/cluster	14.854	3.632	7.860

The model accuracy without clusters is very high with RMSE of 14.854, the mean absolute error mean was 3.632 customers, and the median absolute error was 7.86. The model using clusters with the worse performance (cluster 0) had a RMSE 5.719, mean absolute error 2.115 and median absolute error of 3.774. The model with the best overall performance (cluster 4) had a RMSE of 0.674, mean absolute error 0.715 and median absolute error 0.594. However, the lowest value in the mean absolute error was in the cluster 3.

References

Appendix: Chunk options

B.3.3 Software versioning

B.3.3.1 R

```
cat(paste("#", capture.output(sessionInfo()), "\n", collapse  = ""))
```

```
## # R version 4.2.1 (2022-06-23)
## # Platform: x86_64-pc-linux-gnu (64-bit)
## # Running under: Ubuntu 20.04.4 LTS
## #
## # Matrix products: default
## # BLAS:   /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
## # LAPACK: /home/sobreiro/miniconda3/envs/survival/lib/libmkl_intel_lp64.so
## #
## # locale:
## # [1] en_US.UTF-8
## #
## # attached base packages:
## # [1] stats     graphics  grDevices utils     datasets  methods   base
## #
## # other attached packages:
## #  [1] mclust_5.4.10    labelled_2.9.1   kableExtra_1.3.4 gtsummary_1.6.0
## #  [5] visdat_0.5.3     readxl_1.4.0     stargazer_5.2.3  reticulate_1.25
## #  [9] ggplot2_3.3.6    dlookr_0.5.6     dplyr_1.0.9
## #
## # loaded via a namespace (and not attached):
## #  [1] reactable_0.2.3     webshot_0.5.3       httr_1.4.3
## #  [4] tools_4.2.1         utf8_1.2.2          R6_2.5.1
## #  [7] rpart_4.1.16        DBI_1.1.3           colorspace_2.0-3
## # [10] withr_2.5.0         tidyselect_1.1.2    gridExtra_2.3
## # [13] curl_4.3.2          compiler_4.2.1      extrafontdb_1.0
## # [16] cli_3.3.0           rvest_1.0.2         gt_0.5.0
## # [19] xml2_1.3.3          labeling_0.4.2      bookdown_0.26
## # [22] scales_1.2.0        mvtnorm_1.1-3       rappdirs_0.3.3
## # [25] systemfonts_1.0.4   stringr_1.4.0       digest_0.6.29
## # [28] rmarkdown_2.14      svglite_2.1.0       pkgconfig_2.0.3
## # [31] htmltools_0.5.2     showtext_0.9-5      extrafont_0.18
## # [34] fastmap_1.1.0       highr_0.9           htmlwidgets_1.5.4
## # [37] rlang_1.0.2         rstudioapi_0.13     sysfonts_0.8.8
## # [40] shiny_1.7.1         generics_0.1.2      farver_2.1.0
```

```
## # [43] jsonlite_1.8.0      magrittr_2.0.3      Formula_1.2-4
## # [46] Matrix_1.4-1        Rcpp_1.0.8.3        munsell_0.5.0
## # [49] fansi_1.0.3         gdtools_0.2.4       partykit_1.2-15
## # [52] lifecycle_1.0.1     stringi_1.7.6       yaml_2.3.5
## # [55] inum_1.0-4          grid_4.2.1          hrbrthemes_0.8.0
## # [58] promises_1.2.0.1    forcats_0.5.1       crayon_1.5.1
## # [61] lattice_0.20-45     haven_2.5.0         splines_4.2.1
## # [64] hms_1.1.1           knitr_1.39          pillar_1.7.0
## # [67] glue_1.6.2          evaluate_0.15       pagedown_0.18
## # [70] broom.helpers_1.7.0 vctrs_0.4.1         png_0.1-7
## # [73] httpuv_1.6.5        Rttf2pt1_1.3.10     cellranger_1.1.0
## # [76] gtable_0.3.0        purrr_0.3.4         tidyr_1.2.0
## # [79] assertthat_0.2.1    xfun_0.31           mime_0.12
## # [82] libcoin_1.0-9       xtable_1.8-4        later_1.3.0
## # [85] survival_3.3-1      viridisLite_0.4.0   tibble_3.1.7
## # [88] showtextdb_3.0      ellipsis_0.3.2
```

```
  # or use message() instead of cat()
```