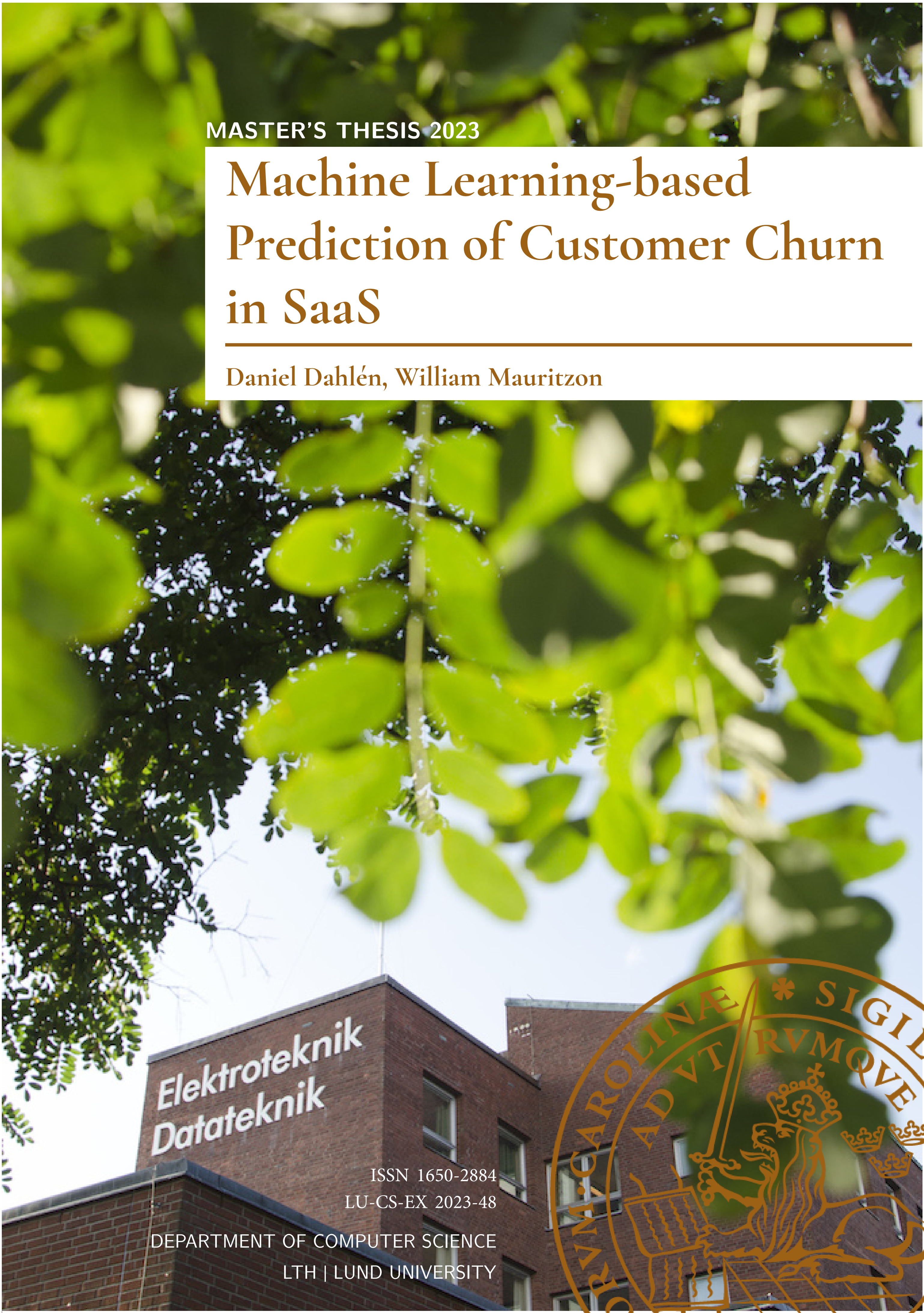
MASTER'S THESIS 2023

Machine Learning-based Prediction of Customer Churn in SaaS

Daniel Dahlén, William Mauritzon



ISSN 1650-2884
LU-CS-EX 2023-48

DEPARTMENT OF COMPUTER SCIENCE
LTH | LUND UNIVERSITY

Machine Learning-based Prediction of Customer Churn in SaaS

Daniel Dahlén
da0785da-s@student.lu.se

William Mauritzon
wi3328ma-s@student.lu.se

November 28, 2023

Master's thesis work carried out at
the Department of Computer Science, Lund University.

Supervisors: Alma Orucevic Alagic, Alma.Orucevic-Alagic@cs.lth.se
Salomeh Kiani Johnsson, Sally@fieldly.com

Examiner: Mathias Haage, Mathias.Haage@cs.lth.se

Abstract

The focuses for businesses on acquiring new customers often overshadows the equal importance of retaining their existing customer base. Customer churn, which refers to the loss of customers, presents a critical challenge with the potential to greatly impact various aspects of a business, including its revenue, profitability, and overall success.

This study delves into customer churn prediction in the B2B SaaS sector, aiming to develop a machine learning model to find churn factors and enhance customer retention. In this study, several models, including Logistic Regression, Random Forest, XGBoost, GBC, and LGBM, were evaluated using techniques like SMOTE for handling imbalanced data, RFECV for feature selection, and Gridsearch for hyperparameter tuning.

The study's results indicated that the XGBoost model outperformed all other models, achieving an AUROC score of 0.900, an accuracy of 0.848, and a recall score of 0.832.

The feature analysis identified significant factors in both short-term and long-term churn. These factors included customer tenure, project involvement, activity in expense reports, and integration-related activities. While specific features were more important, the overall finding is that predicting customer churn relies on the collective contribution of multiple features rather than any single individual feature.

Keywords: Churn Prediction, Churn Factors, Customer Churn, Machine Learning, Customer Relationship Management, SaaS, B2B

Acknowledgements

We would especially like to thank our supervisor, Alma Orucevic Alagic, for her guidance and help throughout the project. We would also like to thank Mathias Haage for providing us with initial help when discussing about architecture networks and general questions. We're also grateful to the team at the case company for their engagement in our thesis and answering all our questions. Without the help from all mentioned we would not have been able to achieve the results that we did.

Abbreviation

SaaS	Software-as-a-Service
B2B	Business-to-Business
CCP	Customer-Churn-Prediction
AUROC	Area Under the Receiver Operating Characteristic Curve
SMOTE	Synthetic Minority Oversampling Technique
RFECV	Recursive Feature Elimination, Cross-Validated
SHAP	SHapley Additive exPlanations
XGBoost	eXtreme Gradient Boosting
GBC	Gradient Boosting Classifier
LGBM	Light Gradient Boosted Machine

Contents

Chapter 1
Introduction

1.1 Background

In the world of commercial enterprises, achieving financial success is closely tied to customer management. This means not only acquiring new customers but also minimizing customer churn, as these factors together contribute to a business's overall performance. The significance of customer churn, or the rate at which customers discontinue their association with a business, cannot be overstated. Recognizing customer churn as a critical issue implies understanding that when customers leave a business at a higher rate than desired, it indicates deeper problems or challenges within the organization. A high rate of customer churn directly impacts revenue by reducing the customer base and leads to hidden costs due to increased marketing expenditure and the potential damaging of a brand's reputation.

Chang et al. [1] explain how many companies are now focusing their efforts on analytical Customer Relationship Management (CRM) to stay competitive in today's business environment. For every customer, an extensive array of data points is stored — numbering in the thousands, and sometimes even millions. With this abundance of data, there is the potential to uncover valuable business knowledge that spans the entire customer life cycle. Within CRM, customer retention has received most attention as research has showed that retaining existing customers is considerably more lucrative than continually seeking new ones [2].

This discovery among other reasons has contributed to a shift in marketing trends from a focus on individual transactions to building long-term relationships [3]. For much of the 20th century, businesses primarily engaged in transactional marketing, which emphasized sales promotions and the continuous acquisition of new customers [3]. However, in recent decades, companies have come to realize that solely relying on transactional marketing may not be enough to stay competitive [3]. Modern marketing is about more than just selling products; it's about creating and maintaining strong, long-lasting relationships with customers [3].

According to Palmatier et al. [4], relationship marketing is particularly effective in service-based settings compared to product-oriented ones, and it thrives in business-to-business

(B2B) contexts rather than business-to-consumer (B2C) environments. Unlike physical products, services are inherently more complex due to their variability and greater difficulty in assessment, which results in deeper customer involvement [4]. In that manner, businesses are coming to recognize that their most valuable asset is their customers [5]. This shift in perspective has prompted a strategic redirection which aims to gain deeper insights into their customer base.

The advancements in the fields of data analytics and machine learning have changed the ways how businesses approach the challenge of customer churn. Prior to predictive models, companies relied on customer surveys to gather feedback and enhance their services while establishing customer success teams to facilitate smoother onboarding and provide support [6]. Furthermore, the company analyzed in this study [6] primarily focused on a limited set of data features, assessing a customer's likelihood of churning based on their activity within these specific features.

Predictive modeling, particularly customer churn prediction, has proven to be effective in identifying customers on the verge of churning and, subsequently, retaining them through retention efforts [7][8][9].

The telecom industry has received a lot of attention due to its distinctive characteristics, notably its position in an intensely competitive market.

While customer churn prediction research has often centered on sectors like telecommunications within B2C [10][11][12], Software as a Service (SaaS) companies, especially within B2B bring forth distinct characteristics. This distinction arises from the fact that SaaS companies focus on providing a service that involves direct interaction between customers and the software. This interaction allows for the tracking of a wide range of user activities, which sets them apart from the data typically collected in telecommunications studies.

In telecommunications studies, the data primarily centers around basic call metrics like call volumes and missed calls [10][11]. In contrast, in SaaS scenarios, a much broader set of features is considered, including metrics related to the number of projects, logs, feature usage, user logins, files, and comments, among other factors [13].

This distinction is driven by the nature of the services provided: telecommunications companies offer communication services like calls and internet access, where customers do not interact extensively with the underlying software.

Also, after reviewing numerous churn prediction studies, customer churn prediction lacks a standardized or generic approach [8]. Instead, these approaches are customized to fit the specific characteristics, data, and resources of each company or study. This emphasizes the significance of acquiring industry specific insights, considering the distinctive features of SaaS and the wide array of customers it serves.

1.2 Context and Case Company

This thesis is done with collaboration with the Swedish construction software company, Fieldly [14]. Fieldly provides a SaaS solution for companies in several fields for instance: installation & service, construction service, ground & facility, drilling, and for contractors. Their platform simplifies project management by handling tasks from creating sales quotations to managing payroll and invoicing documentation, which can be integrated with the company's financial system. As a user, you can access a user-friendly mobile and cloud-based platform to efficiently manage projects, work orders, resources, checklists, expense reports, and other documents.

In recent years, Fieldly has started to upscale and add a lot of new features to attract new customers, and to keep current ones. As illustrated in Figure 1.1, the years between 2020 and 2022 Fieldly witnessed a surge in customer numbers. The blue line in Figure 1.1 shows the number of customers at a given year. The red line in Figure 1.1 illustrates the churn rate across various time intervals. For example, the churn rate for the year 2022 reflects the percentage of customers from 2021 who did not continue as customers in 2022, see Section 2.2 for more about churn calculation. As can be seen in the figure, Fiedly had a churn rate of approximately 16% in the year 2022, indicating that there is potential for reducing customer churn.

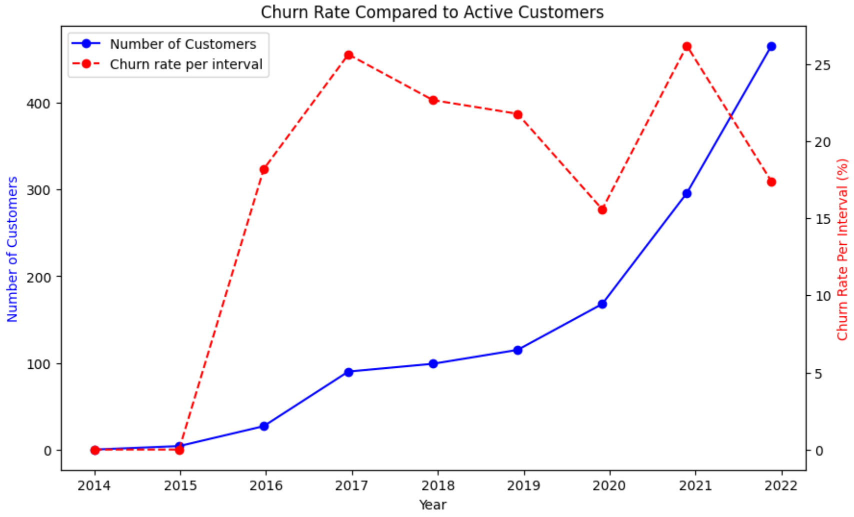


Figure 1.1: History number of customers

The cost of churn is staggering with American companies collectively losing an estimated $168 billion each year [15]. Even minor percentages accumulate into substantial financial losses. For instance, a mere 2% increase in customer churn for a business generating $5 million in annual sales results in a loss of $100,000 in revenue, not including the cost of filling the void of lost customers [15].

Currently, Fieldly assesses each customer's health status, which reflects their likelihood of churning, by the customers usage of a few functions from the service provided. From this score they determine which customer to contact to improve the usage of the service for the customer. Fieldly's inability to effectively identify churn and its contributing factors may lead to missed revenue opportunities and reduced market competitiveness.

In Figure 1.2, which displays the tenure distribution for churned customers over a period of 3 years, it becomes evident that churn rates decrease over time, with the majority of churn occurring within the first year. An observation from the figure reveals a notable annual spike in churn. Given that Fieldly often provides yearly contracts with customers, the occurrence of this churn spike is not unexpected. Focusing on the distribution of first-year tenure depicted in Figure 1.3, we observe a decline in churn rates as time progresses.

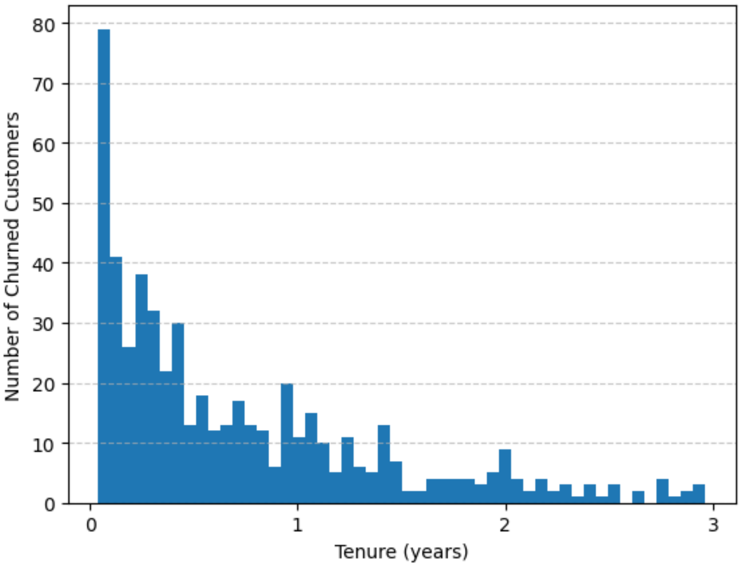


Figure 1.2: Distribution of tenure 3 years

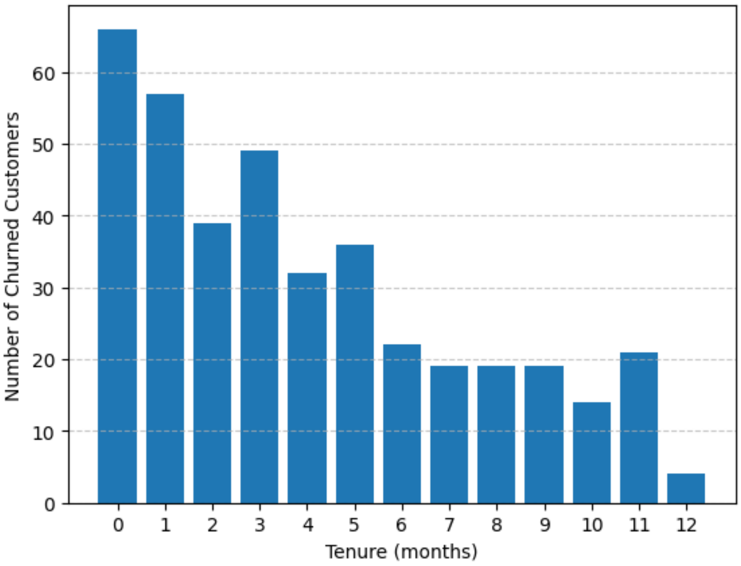


Figure 1.3: Distribution of tenure 1 year

1.3 Aim

This thesis aims to develop a model using data handling and machine learning to identify the primary factors modulating customer churn and predict instances of churn. By analyzing a company's characteristics and their activity, including the specific features they use and the extent to which they use them, the model can predict churn and pinpoint crucial indicators. This can then be used to take timely actions to prevent churn.

Given that the study was conducted in collaboration with Fieldly, a SaaS B2B company, this study will focus specifically on the SaaS B2B sector. Following this, here are our research questions:

1. What are the most important factors that affect customer churn?
2. Among the tested models, which model performs the best at predicting churn?

1.4 Delimitations

Delimitations had to be set to fit the scope of our thesis and based on resource and time constraints. Also, to make the thesis more viable some practical choices were made.

- Exclusively data that is readily available from the Fieldly's internal databases and systems will be used in the predictive analysis. The result is thereby not dependent on other unpredictable data sources that may fail. Examples of such data sources may include external market data, providing information on industry trends, competition, and economic indicators. Relying solely on internal data will improve reliability and also helps with seamless integration for future use.
- Customer churn analysis can consider different time periods, such as short-term, mid-term, and long-term churn, see Section 2.2.2 for a description of churn time periods. For this thesis, the focus will be on mid-term and long-term churn, see Section 3.7 for how churn periods were categorized.

1.5 Outline

The thesis is organized as follows. In Chapter 2, we present the literature study, including churn and churn prediction concepts that is helpful to understand to grasp the essence of the study. In Chapter 3, we give an overview of our methodology, including data extraction techniques, chosen evaluation metrics, model architecture and optimization strategies. In Chapter 4, we present the results of the study. Furthermore, in Chapter 5, the result and findings are reviewed. Finally, in Chapter 6, we describe the conclusion of the study.

Chapter 2

Literature Study

2.1 Software as a Service

SaaS is a model where a service is hosted, maintained, and updated by a provider [16]. This service is usually paid using a subscription-based structure where customers can pay on a monthly or annual basis. The SaaS model makes is possible to access services remotely over the internet, increasing flexibility and availability for customers. The primary identifiers for the SaaS model are internet accessibility, flexibility, security, scalability, and centralized feature updating [16]. The service provided is updated by the provider without the customer needing to upgrade [16]. Flexibility in terms that each customer can customize the data used, opt in, or opt out of modules provided [16].

2.2 Customer churn

One of the most important if not the most important metric for a SaaS company is the customer churn rate [17]. Customer churn, as defined in the literature [18], occurs when a customer discontinues their usage of a company's services. The churn rate refers to the percent of customers that stopped using a service or product over a certain time, for instance monthly, quarterly, or annually. The formula to calculate customer churn rate is defined below.

$$\text{Churn Rate} = \frac{\text{Number of Customers Lost during a Period}}{\text{Total Number of Customers at the Start of the Period}}$$

Keeping a low churn rate is important to ensure a steady revenue stream. A 5% decrease in customer churn rate can have a significant impact on profits, potentially leading to a 25% improvement in overall profitability [19]. Per P. Campbell [20], "Churn is the silent killer of your company. If you don't address churn early, you'll be working extremely hard just to stand still". This is because all the effort and resources a company invests in acquiring new customers would essentially be squandered if those very customers end up leaving at a significantly high rate.

2.2.1 Voluntary and Involuntary churn

In the context of customer churn, there are two types of churn that are important to distinguish, this involves; involuntary and voluntary churn. Involuntary churn refers to when a customer is forced to terminate a service due to events out of their control. In the context of SaaS, it often pertains to customers discontinuing their subscription due to factors like bankruptcy or insufficient funds to cover the service, or being acquired by another company that uses a different related service. It can also be less apparent reasons like the customer moving to a new location where the service is not available. Basically, all reasons that do not involve any dissatisfaction with the providing company. Involuntary churn affects lower-tier plan customers more frequently than high-tier plan customers due to a reduced likelihood of payment issues [6].

On the other hand, voluntary churn happens when customers actively decide to end their association with a business. This type of churn can happen due to many reasons, such as the product or service no longer aligning with the customer's needs or the discovery of a cheaper or superior alternative.

Voluntary churn is the primary type of churn targeted for prediction in studies, as involuntary churn is not outcome of inadequate service's performance.

2.2.2 Churn time periods

Churn is often categorized into three separate stages, that is: short-term, mid-term and long-term churn.

The short-term stage typically occurs within the initial months (usually the first month) after a customer sign up, during which they are introduced to the service and can form an initial assessment of its quality [20]. Churn rates tend to be higher during this stage because people often sign up to test out services and understand their main benefits and drawbacks and deciding whether to continue using it or not [20].

The mid-term churn stage typically happens after the initial evaluation period, usually within the first one to six months of the customer's journey [20]. By this point, customers are likely enjoying the product to some extent and have experienced its core value.

The long-term stage occurs after the customer has been using the product or service for an extended period, usually from 6 months to 12 months, and they have become loyal to the brand [20]. In this stage, the customer is much less likely to churn as they have invested time and effort into the product and most likely have had a positive relationship with the company. A good strategy here is to introduce new features or upgrades to keep the customer engaged and again experience the core value of the service [20]. In this stage however, it is important to be aware that the customer is entering the maturity stage of the product life cycle and companies may begin planning for the product's replacement [21].

These time periods fluctuate depending on the on boarding and adaptation time when opting for a new system, as it takes a different amount of time depending on the service characteristics. For example, a single user signing up for a telecom subscription might enter the mid-stage after only a week or two, while for a B2B environment, it might take one year to enter the mid-stage since it takes longer time for companies to adapt than individual customers.

2.2.3 Drivers for churn

In order to find ways to keep customers engaged, it's essential to grasp the reasons behind churn and find ways to stop customer churn. According to a case study by Nikola [22] on a data management SaaS company, 27% of customer churn is due to the customer not achieving the expected results from the service, while the second biggest group of churners at 17% were because of not having enough personnel that could be devoted to use the service. The third largest reason at 13%, churned due to dissatisfaction with the customer service. Three percent of customers churned because of financial difficulties. Two percent of customers churned with the reason of switching to another company. The remaining 24% did not identify their reason of churning [22].

Within voluntary churn the reasons for churn are often categorized as in Table 2.1. Wangenheim et al. explains that the primary churn factor is failure of the service provided [23]. More-so, the better support a company receives the more likely they are to stay and renew their contract, even small improvements in service quality could have major positive influence on the likelihood of a company upgrading [23].

One framework used to identify customer behaviour is CUSAMS (customer asset management of services) [24]. Bolton et al. introduces CUSAMS with the term's breadth, depth, and length of customer relationships [24]. The primary focus for this study lies in the length dimension of CUSAMS. The longer a customer stays with a company the more likely they are to renew their contract, the longer the length of the relationship the more positive affect it has on customer spending patterns [23].

One major factor that plays a big role in B2B relationships whether customers stay or not is the switching cost [25]. Switching costs can be defined as the costs that a customer suffers from when they switch from one product or service to another [25]. In B2B industries, several factors can influence the dimensions of switching costs. These factors include the time and resources spent on researching alternatives, learning a new service, establishing new relationships, the perceived loss of prior investments in the current relationship, and the uncertainty that switching will lead to better outcomes [25].

When switching services is difficult and expensive, people are likely to stay with what they have. On the other hand, in situations where switching is easier, customers might be more willing to change to another service, thereby affecting the churn rate.

Category	Example
Product performance	Not fulfilling promised results
Product failures	Slow performance and reliability issues
Competition	Competitive prices and features
Support experience	Contact with support and overall helpfulness received
Tenure	The length of the relationship

Table 2.1: Typical Churn Drivers

2.3 Customer Churn Prediction

In this section we will define the problem and explain what type of data that is typically used for churn prediction. Moreover, previous studies in the area are examined.

2.3.1 Defining the problem

In the context of managing customer churn, predicting which customers might leave is only part of the equation. To make our analysis more valuable, we need to understand the reasons behind the churn and identify the appropriate retention strategies. To explore this further, we present the following questions:

1. Who will leave? What characteristics do a churning customer exhibit?
2. When will they leave? How accurately in advance can we predict that a customer will stop using the service?
3. Why do they leave? What are the main causes for leaving?
4. What can be done to retain customers?

These questions are crucial before shaping solution for churn. If we know *"who"* we can personalize solutions. If we know *"when"* we can prevent churn in time. If we know *"why"* we can, if possible, improve the services or change parts of the service to keep said customers. And if we know *"what"*, then specific actions can be taken to address the churn problem.

2.3.2 Classifying Churn

Classifying a customer as a churned customer can be approached in various ways based on available data. One approach involves maintaining a database that records customer churn status. This database can be updated, for example, when a customer submits a churn ticket or requests to cancel their subscription. Another approach focuses on user activity, potentially allowing earlier detection by monitoring usage patterns. In the activity-based approach, a specific time window is chosen, which within users must show activity within the service [6]. This method does not require a subscription status database but relies solely on an activity database, assuming consistent tracking of user behavior's for all customers.

2.3.3 Data

Successfully predicting when customers might leave relies on understanding how they use the service or product, which requires access to lots of well-organized data. The four big data aspects; Volume, Velocity, Variety, and Veracity, are relevant here [26]. Volume is important as it ensures the availability of a sufficient amount of data to accurately replicate real-world complexity [26]. Veracity refers to the quality and trustworthiness of the data, and its critical as proper data handling ensures reliability and eliminates anomalies like test accounts [26]. Velocity refers to the speed at which data is collected and processed and affects real-time model usage. Lastly, Variety introduces diverse data types to be able to make a better analysis of a problem [26]. These aspects are very important, as the quality of the dataset will directly influence the predictive accuracy of a model.

Big data is often defined as data volume in the terabytes or petabytes range [26]. Even though the data size may not meet this definition, it still reasonable to validate the quality of the data fed to the model based on the four key factors associated with big data mentioned above.

Typical data categories in churn prediction models include customer demographic features, user behavior features, support features and contextual features [10] [23] [27], see Table 2.2.

Data Category	**Example**
Customer Demographic Features	Company Type, Financial State, Subscription Type
User Behavior Features	Utilized Features and Activity Patterns
Support Features	Customer Satisfaction Scores
Contextual Features	Economic Trends, Competitors' Pricing Strategies

Table 2.2: Typical Data Categories

2.3.4 Previous Studies

Churn prediction have sometimes in previous studies been categorized into different types of problems. A classification problem, such as determining whether a customer will churn or not. Regression problem, for example, calculating the probability of a customer churning. Lastly a ranking problem, is identifying the customers with the highest likelihood of churning. These classifications are sometimes very ambiguous. Making the classifications hard to differentiate between just by looking at a previous study. By setting a threshold one can utilize a regression problem then to make classifications. The threshold defines the cut off values a classification is made for. Making it both a classification, but also a regression problem in this definition above.

In Table 2.3 six studies were chosen from how well they had performed. In this list we can see combinations of using XGBoost, SMOTE-ENN and decisions tree's scoring all from 80% up to over 90% accuracy.

References	Architecture	Result
Nguyen et al. (2022) [28]	RBF kernel Support Vector Machine with parameter tuning, SFS/SBS, and SMOTE ENN.	Accuracy of 99.01% and an F1 score of 98.88%.
Gore et al. (2020) [9]	Neural network (ANN) prediction model, boosted with SMOTE-ENN to handle imbalanced data.	Achieving an 94% accuracy. In comparison, the Decision Tree model scored 92% on the same dataset.
A. Abdelrahim et al. (2019) [27]	Decision Tree, Random Forest, GBM, and XGBoost	Best results came from XGBoost, reaching a 93.3% AUC.
P. Lalwan et al. (2021) [29]	Multiple classification algorithms (logistic regression, naive bayes, support vector machine, random forest, and decision trees). Used boosting, ensembling, and K-fold cross-validation for accuracy enhancement and hyperparameter tuning.	Adaboost and XGBoost Classifier as top performers with 81.71% and 80.8% accuracy, and an 84% AUC score.
O. Pandithurai et al. (2023) [30]	Four supervised classification algorithms: Logistic Regression, Support Vector Machine (SVM), Decision Tree Classifier, and the Random Forest algorithm.	The Random Forest algorithm outperformed Logistic Regression, SVM, and decision trees, with an accuracy of 84.3%.
R. Peddarapu et al. (2023) [31]	The study employs ensemble learning with XGBoost, SVM, Logistic Regression, and Random Forest models.	The ensemble achieves an 86.3% accuracy. The Random Forest classifier stands out as the most accurate in the analysis.

Table 2.3: Churn Prediction Performance Comparison

2.4 Confusion Matrix

Confusion matrix is a summary of the predictions made of the model, containing information that can be seen in Table 2.4 [32]. The matrix is used to show the number of true positives, false positives, true negatives, and false negatives. These measures enable a detailed evaluation of the model's performance in correctly identifying positive and negative cases, as well as the extent of misclassifications.

Type	Definition
True Positive	Observation is positive, and prediction is positive
False Positive	Observation is negative, but prediction is positive
True Negative	Observation is negative, and prediction is negative
False Negative	Observation is positive, but prediction is negative

Table 2.4: Confusion Matrix Labels

2.5 Evaluation Metrics

To evaluate the performance of the model, AUROC was used as a measuring tool, as discussed in Section 2.5.2. However, other metrics were used as well to further understand how the model behaved. True positive rate (TPR) and false positive rate (FPR) were used when constructing AUROC. TPR and FPR is used to show how well a model correctly classifies cases. Accuracy, Precision, Recall, Specificity and F1 Score were also used in this study, see Table 2.5.

Metrics	Calculations
TPR / Recall	$\frac{\text{True Positives}}{\text{True Positives + False Negatives}}$
FPR	$\frac{\text{False Positives}}{\text{False Positives + True Negatives}}$
Accuracy	$\frac{\text{Correct Classifications}}{\text{Total Amount of Classifications}}$
Precision	$\frac{\text{True Positives}}{\text{True Positives + False Positives}}$
Specificity	$\frac{\text{True Negatives}}{\text{True Negatives + False Positives}}$
F1 Score	$\frac{2 \cdot (\text{Precision} \cdot \text{Recall})}{\text{Precision} + \text{Recall}}$

Table 2.5: Evaluation metrics

2.5.1 F1 Score

F1 score is the harmonic mean of precision and recall [32]. How to calculate F1 score can be seen in 2.5. This score can range between 0 and 1, where 1 is the model having a 100% accuracy of classifying each observation. F1 Macro is calculated by taking the macro scores from precision and recall and then applying the f1 score formula. F1 macro is used to get an overview of how well the classifications were done by a model as a whole. F1 macro does this by evaluating the scores for all classes and combining them into one. This scoring metric is widely used within the machine learning community and is a good indicator of how well a model performs.

2.5.2 AUROC

AUROC (Area Under the Receiver Operating Characteristic Curve) is used for measuring classifications problems [32]. The AUC is used for determining how capable the model is at distinguishing between class 1 from class 0. Where ROC is a probability curve plotted with TPR (True Positive Rate) against FPR (False Positive Rate). Together AUC and ROC explain how reliable one model is at classifying data. The score received can range from 0 to 1 where 1 is the highest score possible [32].

2.6 Correlation Matrix

A correlation matrix is a statistical tool used to show how much each feature in a dataset correlated to another [32]. There can be both positive and inverse correlation [32]. A correlation of 1 between two variables means that the two features in question is 100% correlated to each other [32]. The same is true for -1 where it instead has 100% inverse correlation with the feature compared to [32]. The optimal scenario is having uncorrelated features, a score of 0 for each comparison. The reason why we want low correlation is because then each feature would describe the target class independently resulting in more information for the model. On the contrary if each feature would be correlated with each other then it would be no difference of having hundreds of features or only one, due to their correlation all features would describe the same behaviour.

2.6.1 Spearman's Correlation

Spearman's correlation serves as a valuable tool for quantifying the degree of association between two variables. Spearman's correlation does this by assessing if there exists a monotonic relationship between the variables. Monotonic relationship is when one of the variables either always increase or decrease when the other variable increases or decreases, meaning that they both would be linear dependant of each other [33]. By using Spearman's correlation, one could then measure all variables in a dataset to construct a correlation matrix.

2.7 Loss function

The loss function is a mathematical function used to measure the difference from the actual values and the predicted values of a model [32]. A small loss would mean that the predicted values are close to the actual values.

2.8 Overfitting

Overfitting of the data happens when a model learns from minor variations [32]. The goal of creating a model is to find the "true signal" [32] and ignore the "noise". This means that it is not useful for a model to understand every minor detail of a problem, but instead it is better to for the model to get a more generalized understanding of a problem. Overfitting usually occurs when we use a very flexible and or fast model [32].

2.9 Regularization

Regularization is something utilized to reduce potential overfitting [32]. It works by adjusting the loss function and thus preventing overfitting.

2.10 Machine Learning Models

In this section, the machine learning models used in the churn prediction will be covered briefly.

2.10.1 Linear and Logistic Regression

Linear regression is a fundamental statistical method used to model the relationship between a dependent variable (y) and a set of independent variables (x_1, x_2, etc.) [32]. The objective for this method is to find the best-fitting line, represented by a linear equation, as seen below:

$$y = b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + \ldots + b_n \cdot x_n$$

The coefficients $(b_0, b_1, b_2, \ldots, b_n)$ in this equation are determined through an iterative process known as the method of least squares. This method minimizes the squared differences between the actual data points and the values predicted by the line. The goal of linear regression is basically to create a line that closely matches the data by modulating these coefficients. Initially, they are set to random values and the algorithm iteratively updates them until the model converges and to a state where further iterations do not drastically reduce the error.

Given the values of the independent variables $(x_1, x_2, \ldots, x_n)$, the model estimates the corresponding value of y, making it easy to understand the relationship between input and output, and how individual variables affect the dependant variable.

Logistic regression is another essential regression technique used for binary classification tasks. While linear regression predicts continuous values, logistic regression predicts the probability of an observation belonging to a specific class or category.

In logistic regression, we use the logistic function (often referred to as the sigmoid function) to transform a linear combination of the independent variables into a probability score:

$$P(Y = 1) = \frac{1}{1 + e^{-(b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + \ldots + b_n \cdot x_n)}}$$

Here, $P(Y = 1)$ represents the probability of the binary outcome being 1 (or "yes"), and e is the base of the natural logarithm.

2.10.2 Ensemble Learning

Ensemble learning is a machine learning technique that combines multiple machine learning models with the goal of achieving a better predictive performance compared to using an individual model [34], such as just a linear regression model.

One ensemble algorithm is boosting which reduces bias and variance, it is a greedy algorithm that fits a model by applying weak learners sequentially [32]. More weight is given to misclassified examples made by earlier rounds [32].

Gradient Boosting works similar to regular boosting, but rather than changing the weights of each iteration as in boosting, Gradient Boosting tries to make each new model adjust the new predictor based on the errors that were not correctly predicted by the previous predictor [34].

Gradient Boosting Classifier

Gradient Boosting Classifier (GBC) is an implementation of a gradient boosting algorithm used for classification tasks [35]. The GBC starts with a simple model, typically a decision tree with limited depth, and sequentially builds more complex models by focusing on the errors of the previous ones. In essence, it corrects the mistakes of its predecessors, improving predictive accuracy with each iteration.

Light Gradient Boosting Machine

Light Gradient Boosting Machine (LGBM) is a gradient boosting framework that differs from traditional GBC in its tree-building strategy [36]. Unlike GBC, which expands trees level by level, LGBM employs a histogram-based approach that partitions data into smaller bins and constructs tree's leaf-wise [36]. This strategy reduces computational overhead, makes LGBM exceptionally fast, and is particularly suitable for large datasets [36].

XGBoost

XGBoost, short for Extreme Gradient Boosting, is a powerful and widely used machine learning library [37]. It has gained popularity for its effectiveness in winning Kaggle competitions and its ability to build highly efficient and accurate models [37]. XGBoost employs gradient-based optimization and regularization techniques to prevent overfitting. It allows for parallel processing and is optimized for performance. While LightGBM may be faster and more accurate in some cases, XGBoost is often chosen for building robust models due to its level-wise growth strategy [38].

Random Forests

Random Forest is an ensemble learning method that builds a collection of decision tree's and combines their predictions [32]. The final prediction is an average of all the decision tree predictions. The key difference from GBC is that Random Forest constructs each tree independently and assigns equal weight to all trees. It introduces randomness both in selecting data subsets for training and in choosing subsets of features for each tree, which helps mitigate overfitting [32]. Random Forest is robust and versatile, with good generalization performance.

2.11 Data Handling Techniques

In this section we will go over different techniques that are used when preparing data.

2.11.1 Balancing

In churn prediction datasets, we often run into a situation where the target variable, indicating whether customers have churned or not, is noticeably imbalanced. This means that the number of customers labeled as "not churned" far exceeds the count of those labeled as "churned" [39]. This can lead to poor performance of machine learning models, as most algorithms are designed around the assumption of an equal number of examples for each class. There are some strategies that one can use to balance the data, including:

1. Collect more data: the more data you must train your model on, the better its performance will be.
2. Oversampling: adding more copies of the minority class to the dataset.
3. Undersampling: removing some samples from the majority class to balance the dataset.

One common oversampling technique is Synthetic Minority Oversampling Technique (SMOTE) [9] [28]. The SMOTE algorithm works by creating synthetic samples based on the existing minority class samples. It does this by selecting a random minority class sample and finding its k nearest neighbors [9]. The algorithm then generates new samples by interpolating between the original sample and its neighbors.

2.11.2 Imputation

It is a common issue in datasets that not all rows contain complete data, resulting in missing rows that can impact model performance. To address this challenge, imputation techniques come into play. Imputation is used to replace these missing rows with either simple or more advanced strategies with plausible data [32].

For instance, consider the case of K Nearest Neighbors (KNN) imputation, where the goal is to identify the k nearest examples in the dataset where the relevant feature is not missing [9]. Subsequently, the missing values for that feature are substituted with the most frequently occurring value within the group.

Another approach involves predicting missing values using additional machine learning models. This method determines the final imputation value for a characteristic (let's call it "x") based on other features. Here, a machine learning model is trained using the values in the remaining columns, utilizing rows in which feature "x" does not have missing values as the training set. Basic imputation methods, like using the mean value, are employed to temporarily impute missing values when multiple feature fields contain missing data [40].

2.11.3 Normalization

Normalization rescales data to a common scale and distribution. One way is to use standard scaling it center's data around zero and adjusts its scale based on standard deviation, making the mean 0 and standard deviation 1. This ensures all variables contribute equally to model training, reduces data variance for stability [32].

This step is crucial because it ensures that all variables contribute equally to model training. Without it, variables with larger ranges could disproportionately influence the model, potentially leading to biased results. It also reduces data variance, making it more stable and predictable, which is essential for effective machine learning model training.

2.11.4 Encoding Nominal Features

Nominal and categorical features cannot be used directly as model features. This is because nominal and categorical types of features cannot be ranked against each other, as neither can be ranked higher or lower than the other [32].

2.12 Cross-Validation

Cross-validation is a technique used in machine learning and statistical modeling to evaluate how well a model can generalize to new data [34]. It involves a repetitive process of partitioning a dataset into subsets, training the model on one subset (the training set), and evaluating its performance on another subset (the validation set). The most common type is k-fold cross-validation, where the dataset is divided into k equally sized subsets. The model is trained and validated k times, with each subset used once for validation while the remaining k-1 subsets are for training [34].

2.13 Feature Selection

Feature selection is used to remove features that do not help with classifying the problem [32]. In the context of predicting customer churn, feature selection is about choosing the most important customer-related factors that help us predict whether a customer might leave. One common method to use is RFECV (Recursive Feature Elimination with Cross-Validation) which is a feature selection technique that uses cross-validation to iteratively determine the most important features [41]. An evaluation metric, such as AUROC, is used to evaluate the significance of the feature and use that score to progressively eliminates less influential features. This process continues until it identifies the optimal feature subset, improving model performance by focusing on the most informative attributes.

2.14 Feature Importance

Machine learning models can in many cases be hard to interpret and can be seen as a black box. Data is fed into the model, and it produces results, yet we often don't know why specific predictions are made. One way of interpreting this information is through feature importance analysis. It reveals the factors that played a role in the model's decision-making process, ultimately shaping the predictions it generated. Based on the model used, different feature importance approaches are taken. For instance, linear models, such as linear regression, provide coefficients as part of their output. In this case, feature importance is typically determined by the magnitude and sign of these coefficients of the independent variables [32]. The magnitude of the coefficient represents the relationship between the independent and the dependent variable, and the sign of the coefficient represents the direction of the relationship [32]. In tree-based models which include for instance XGBoost and Random Forest, the feature importance is instead decided based on how much each feature split point contribute to the model's performance, weighted by the number of observation or data points that pass through the node. However, for these models the sign of the feature importance is not provided but are only given an absolute importance value and thereby requires additional statistical interpretation.

2.15 SHAP

The value of a machine learning algorithm is not only determined by its performance but also by its interpretability. Complex models like XGBoost or Neural Networks may offer high predictive accuracy, but they often lack transparency which makes it challenging to understand why they make certain predictions. Simpler models like linear regression are more interpretable but may sacrifice some predictive power.

SHAP is a Python package for model interpretation that works with any machine learning model's output [42]. It derives its name from "SHapley Additive exPlanation [43]", a concept inspired by cooperative game theory, where each player or feature is assigned, a value based on their contribution to the overall game or prediction. Calculating SHAP values employs coalition game theory, creating feature coalitions to measure their impact on predictions. A coalition refers to a group of features that are used together to make a prediction. For example, if a machine learning model is using three features (A, B, and C) to make a prediction, there are eight possible coalitions of features: A, B, C, AB, AC, BC, ABC, and the empty coalition (no features). The SHAP values are calculated by measuring the contribution of each of these coalitions to the prediction for a specific instance. The SHAP values are then derived by averaging contributions across all possible coalitions for each instance in the dataset, resulting in a set of SHAP values for each feature [43].

Chapter 3
Approach

3.1 Methodology

This study employs a mixed-methods approach, the methodology used for the thesis are experimental and exploratory research [44][45]. The exploratory part of the thesis being the gathering of data, what data should be used in this study. The experiment part of the thesis is the testing of the models, with focus on predicting customer churn. The first research question is in the study domain of exploratory research whilst the second research question is inside the study domain of experimental research.

Exploratory research can be defined to two forms, a topic that has not been researched before or one that has been researched [45]. In the case of this thesis, we will conduct the latter, by trying to find both interesting and already defined important variables to study, see Section 2.2.3. Exploratory research will involve the collection of qualitative and quantitative data through the case company, allowing us to identify potential variables of interest and refine the experimental testing models.

Experimental research is used when trying to find a potential cause and effect relationship. Experimental research consists of multiple stages, the first two being the scoping and the planning of the experiment [44]. In the scope we set the goal of the experiment, the foundation of why we are doing the experiment. After defining the scope of the experiment, we need to plan it, how do we conduct the experiment. The scope is as defined in 1.3 and the planning will be discussed later, see 3.2 for an overview. The models and techniques used in this study have been chosen based on existing literature and theoretical frameworks, see Chapter 2.

By combining elements of experimental and exploratory research, this methodology aims to provide a comprehensive understanding of the research problem while rigorously testing the findings. The integration of exploratory data gathering with experimental testing models enhances the validity and robustness of the findings, contributing to a more nuanced analysis of the research objectives.

3.2 High level approach

The approach taken can be divided into plenty of subcategories all representing an individual part of the machine learning pipeline. An overall view of the steps taken for the churn prediction can be seen in Figure 3.1. Our approach was iterative, meaning we continually refined and improved our methods at each stage of the process.

The main steps in the pipeline are illustrated by the larger boxes in the figure, including Data Handling, Model Architecture, Optimization, Evaluation and Deployment. Each of these modules are then divided into smaller tasks that make up that module. Each module in the pipeline will be explored in greater detail in the following sections.

In the data handling phase, we discuss the selection of data sources and features to include in the model. For instance, we considered demographic features and user analytics features (activities). Also, it is important to during data extraction review the veracity of the data, as discussed in Section 2.3.3, to assess data reliability. This involves cleaning the data, filtering out customers deemed invalid for churn prediction. Additionally, we applied normalization, data splitting, and data balancing, following standard machine learning techniques to potentially improve the model's performance.

Once we have processed the data and undergone multiple iterations and verification's, the next step involves selecting a viable model. This decision is not easy, and we test several models to find the best fit. It intertwines with evaluation techniques to determine if a model performs well. While accuracy is commonly used as a metric, it is not be suitable for imbalanced datasets as a model predicting one class predominantly can still give a very high accuracy. That's why we need to adopt better evaluation metrics, particularly when handling imbalanced datasets.

To further improve the model, optimization techniques such as feature selection and hyper parameter optimization was used.

Finally, in the deployment, feature analysis was conducted to interpret the prediction of the best performing model.

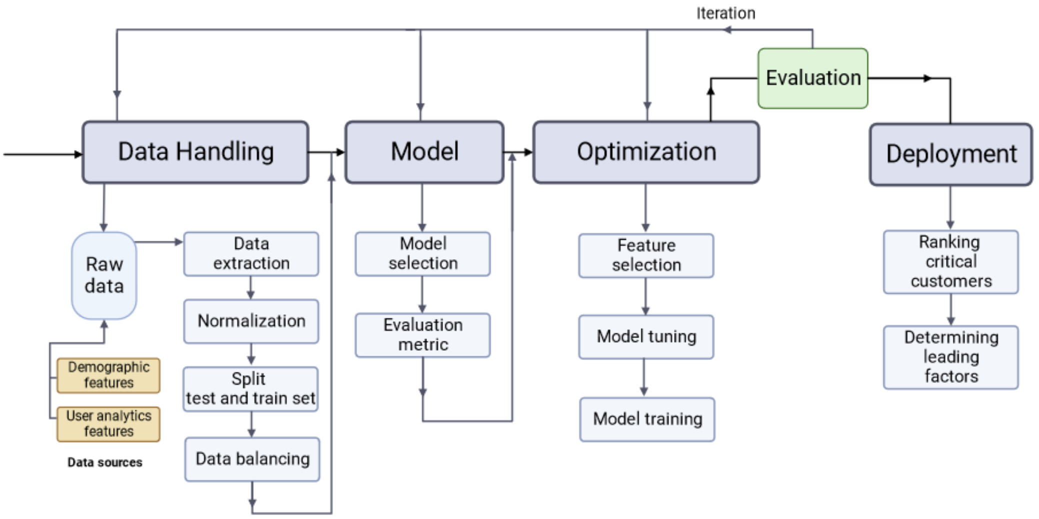


Figure 3.1: Prediction pipeline

3.3 Data Collection and Structure

In this section, we describe the steps taken when collecting data for the churn prediction and how this data is planned to be used.

3.3.1 Classifying Churn

One main data feature that needs to be collected is the target variable. The target variable is defined as whether the company has churned or not. In this study, a non-churned customer is represented as ’0,’ while a churned customer is represented as ’1.’

In Section 2.3.2, we introduced different ways to define when a customer becomes a churned one. One way to do this is by obtaining actual information from the company that they have initiated a churn process, typically by filling a churn ticket. From that moment, they are classified as churned customers. However, in our case, this information was not available for all customers. Therefore, we adopted an activity-based churn definition. This means that if a customer has not engaged in any activity related to a specific feature for ’x’ months, they are considered inactive and classified as churned customers. In our study, we applied this classification for various time periods and for different features, comparing it to the actual churn data we had. Notably, when examining the ’number of projects’ feature with a 2 month time period, we observed a very small margin of error when compared to the actual churn status. This feature is universally used by all companies, making it a reliable indicator for churn classification. We ended up selecting a 2-month period of inactivity, since customers usually had contracts with the case company for longer periods and did not always terminate these contracts in advance. Classifying by activity gave a much better indication of when a customer had decided that they no longer wanted or could continue using the service that the case company provided. As can be seen in Figure 3.2, the churn status for a customer was decided within 2 months prior to the classification date. The classification date is the date when we received access to the database.

In essence, a customer is deemed churned if they have not created any projects within the last 2 months.

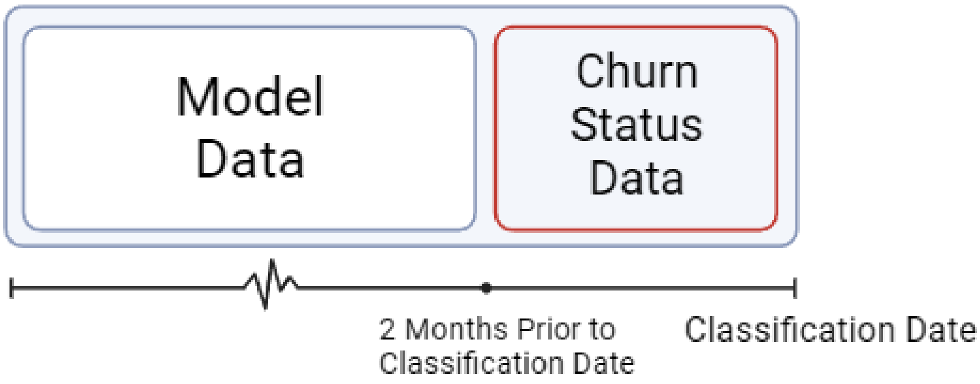


Figure 3.2: Churn Classification Overview

3.3.2 Data collection

The first crucial step is data gathering. To build an effective model, we must carefully identify which data variables will be beneficial and which ones can be ignored. Collaboration with the case company is essential, as it can provide valuable insights into the data collected from the customers. Data collection was done continuously throughout most of this thesis. Effort placed here were necessary for two reasons. First to both understand the data we had access to and how this data was structured. Second to sufficiently gather data that is rich and diverse in number of features to most precisely and accurately predict the target class chosen, relating to the Volume and Veracity concepts introduced in 2.3.3.

Furthermore, most of the data used had to be formatted or constructed to be used as features. As some of the data were not in the structure or in a state that we deemed necessary to qualify as a feature due to the quality we were aiming for. One example of not having the correct structure that we were looking for is that there were not any totals for the number of reports made or the tenure of a customer. We constructed the tenure and the total of amounts for different areas such as reports from existing data. By not having the correct state of the data this could be for example that test data was found in users marked not as test accounts, or that old functions that were no longer in use were found together with new information. In the case of not having a state we deemed necessary the data was simply removed.

By consultation with the case company and by researching prior studies data was selected. Data includes that was deemed relevant stored in the database of the case company. For more information about the data used see Appendices A and B.

3.4 Change Variables

It was decided early in this project that we wanted to try giving the model access to features that would allow it to understand patterns of the customer. The data collected was processed to be put inside timeframes. In Section 3.5 we can see Table 3.2 for variables that were measured within each timeframe. This table contains data that has been segmented into the most recent 3 months from the last activity for each account. These 3 most recent months from the last activity of each user were then constructed to show the change that had occurred from the month before the one compared to. For example, $month3 - month4$ would result in a positive number if the newer month (month 3) has had an increase in the measured feature. These change variables were done for all data variables inside Table 3.2 except the data denoted as activities. In cases where the term activities are part of the data name, instead the total count is used for the activities during these three months rather than the change between them. Basically, the four following months, 4 months before their latest activity were constructed as these change variables. Month 3 shown previous being the data between the 6th and the 7th month. These timeframes were used on all data that would change over time. Data excluded from being put inside timeframes were data that would fluctuate significantly over time or data for which we lacked information regarding updates.

3.5 Data

The data utilized in the development of our machine learning models is presented in both Table 3.1 and Table 3.2.

Table 3.1 consists mostly of the cumulative usage of respective function per user, such as the total amount of expense reports made by one account.

On the contrary Table 3.2 contains data that has been segmented into the most recent 3 months for each account.

Variable Names	**Variable Names**
account_id	number_of_users
churn_status	number_of_projects
country_id	segment
client_titles	number_of_subcontractors
total_articles	total_checklists
number_of_integrations	total_expense_reports
number_of_projects_closed	number_of_projects_completed
number_of_projects_pending	number_of_projects_activated
total_field_reports	number_of_bookings
travel_total_distance	total_time
total_break_time	total_quotes
number_of_ledgers	supplier_titles
tenure	Article_total_activities
Attachment_total_activities	CertificationType_total_activities
Checklist_total_activities	Client_total_activities
Comment_total_activities	Contact_total_activities
Department_total_activities	ExpenseReport_total_activities
ExpenseType_total_activities	FederationType_total_activities
FieldReport_total_activities	GlobalEntity_total_activities
Integration_total_activities	Invoice_total_activities
Ledger_total_activities	LedgerEntry_total_activities
Quote_total_activities	Role_total_activities
SerialNumber_total_activities	Subcontractor_total_activities
SubcontractorType_total_activities	SubStatus_total_activities
SupplierDocument_total_activities	SupplierInvoice_total_activities
TimeReport_total_activities	TimeType_total_activities
TravelReport_total_activities	TravelType_total_activities
Unit_total_activities	User_total_activities
Usergroup_total_activities	VatType_total_activities
WorkItem_total_activities	

Table 3.1: Gathered data

Variable Names	Variable Names
number_of_users_change	number_of_projects_closed_change
number_of_projects_completed_change	number_of_projects_pending_change
number_of_projects_activated_change	number_of_projects_change
total_expense_reports_change	number_of_field_reports_change
number_of_bookings_change	total_break_time_change
total_time_change	total_invoiced_change
total_quotes_change	Article_number_of_activities
Attachment_number_of_activities	CertificationType_number_of_activities
Checklist_number_of_activities	Client_number_of_activities
Comment_number_of_activities	Contact_number_of_activities
Department_number_of_activities	ExpenseReport_number_of_activities
ExpenseType_number_of_activities	FederationType_number_of_activities
FieldReport_number_of_activities	GlobalEntity_number_of_activities
Integration_number_of_activities	Invoice_number_of_activities
Ledger_number_of_activities	LedgerEntry_number_of_activities
Quote_number_of_activities	Role_number_of_activities
SerialNumber_number_of_activities	Subcontractor_number_of_activities
SubcontractorType_number_of_activities	SubStatus_number_of_activities
SupplierDocument_number_of_activities	SupplierInvoice_number_of_activities
TimeReport_number_of_activities	TimeType_number_of_activities
TravelReport_number_of_activities	TravelType_number_of_activities
Unit_number_of_activities	User_number_of_activities
Usergroup_number_of_activities	VatType_number_of_activities
WorkItem_number_of_activities	

Table 3.2: Gathered time frame data

3.6 Datasets

We decided to create two datasets for predicting customer churn in 4 months (short) and in 8 months (long). The decision to create two models was made due to the case company wanting the prediction to be 4 months ahead, giving enough time for applying retention strategies. Additionally, we were interested in exploring the model's performance when making predictions further into the future, hence the creation of the long-term model for an 8-month prediction time. Also, by having two models we can identify patterns that emerge on a shorter and a longer period.

In the short dataset, we excluded 4 months of data, while in the long dataset, we excluded 8 months of data. The reason for the removal of data was to simulate the information the real model would have of the customer prior to churning. Additionally, data was removed to prevent potential biases. These biases could occur from data that might have too high correlation to customer churn, such as customers not using the software close to the customer churn. The decision to exclude all data prior to 2020 was done to keep data that is most in line with the recent version of the software.

3.7 Selecting Churn Time Period

Based on the literature study on churn time periods (see Section 2.2.2), it is clear that there are three distinct churn periods: short-term, mid-term, and long-term churn. The distribution of the customers churning in the case company for each period is illustrated in Figure 3.3. Short-term churn typically falls in the initial month, however for our analysis we define short-term churn as churn in the initial 2 months. This is because the case company provides trial accounts to new customer to test out their service which are often 2 weeks. Its therefore reasonable to consider a longer period for the short-term definition. The orange bar, which comprises most churned customers, represents mid-term churn. The green bar representing long-term churn is nearly equivalent to short-term churn.

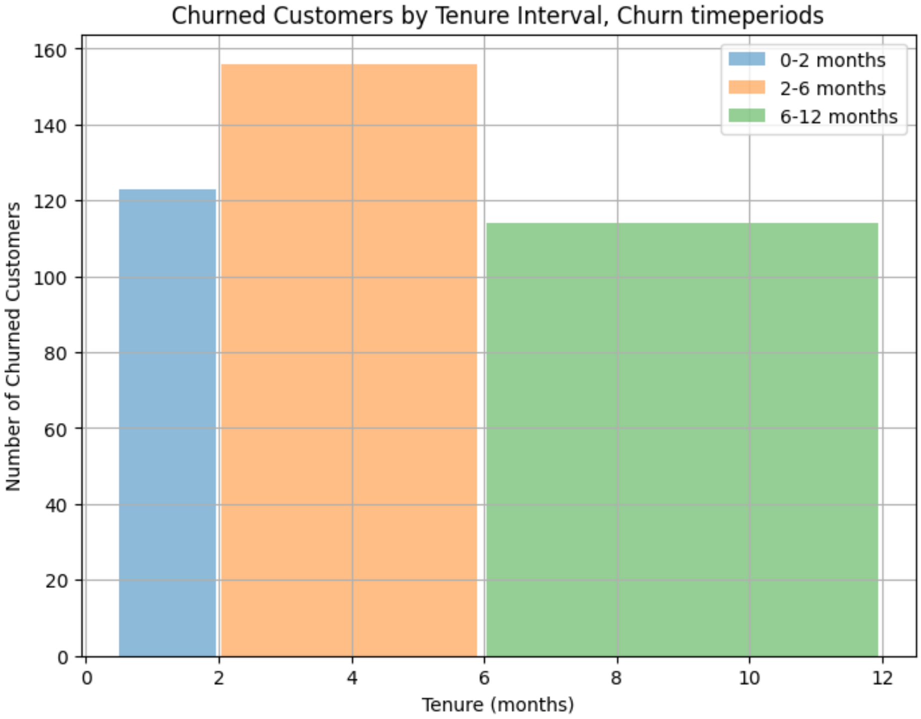


Figure 3.3: Distribution of tenure Churn Time Periods

As the goal was to predict churn at least 4 months in advance, the analysis is primarily focused on mid and long-term churn customers. Furthermore, to ensure meaningful activity tracking, a prerequisite of a minimum of 4 months of data is imposed.

The long model is trained to predict churn 8 months ahead, implying that it can predict churn earliest 12 months from the customer's start date. In contrast, the short model can predict churn earliest 8 months from the customer's start date. The values of 12 months and 8 months as the earliest prediction times for the long and short models, respectively, is based on the model's requirement of having at least 4 months of customer data for accurate predictions. Thereby, an additional 4 months is added to the respective prediction times of 8 months and 4 months for the long and short models. If a customer has a risk of churning in 4 months (short) or 8 months (long), they also have a greater risk of churning earlier, say 4 months before the actual predicted churn occurrence. From this, the prediction time width can be extended with 4 months prior to current prediction time. Consequently, the short data model can be used to predict churn for primarily customers in the mid-term stage (4-8 months) and the long data model can be used to predict churn for customers in the long-term stage (8-12 months).

From the Figure 3.3, we can see that there are a significant number of customers that churn during the mid-term and long-term stages, which means the case company can improve churn for these churn time periods.

3.7.1 Short Dataset, 4 Months in Advance

We decided that the best time to gather data from was from 2020 until 4 months before to the most recent date in the database for each customer, see Figure 3.4. Furthermore, it was decided to be ample amount of time when discussed with the case company to have 4 months to act and prevent a potential churner. This dataset serves to detect mid-term churn and to be a better predictor of churn compared to the long dataset, see 3.4.

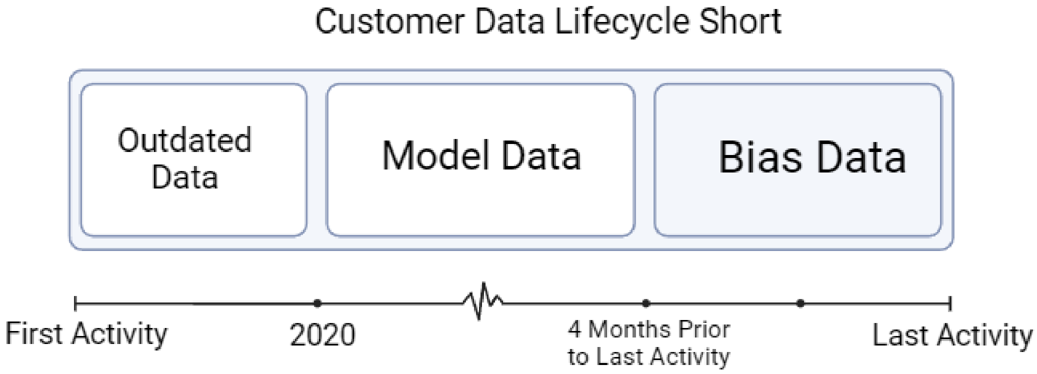


Figure 3.4: Data customer timeline, dataset: Short

3.7.2 Long Dataset, 8 Months in Advance

The dataset with information from 2020 until 8 months prior to churn was done to stop long-term churn, see Figure 3.5. As the long dataset needs more data to function properly it will not be usable as early compared to the short dataset. However, the long dataset has the potential of finding customer churn 4 months earlier than the short dataset, giving more time to stop the potential churn.

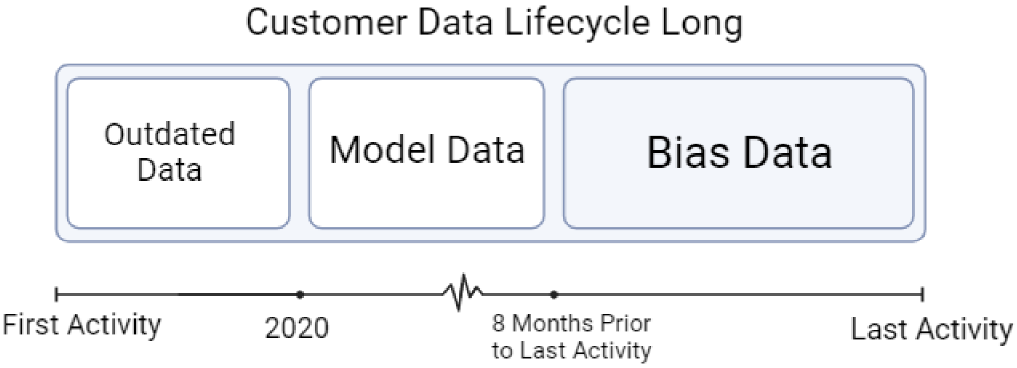


Figure 3.5: Data customer timeline, dataset: Long

3.8 Data Preprocessing

In this section, we detail the data transformation techniques employed to prepare the dataset for model input.

3.8.1 Cleaning

In the cleaning of the dataset certain rows were excluded due to several reasons. First, test accounts created by Fieldly to assess new features had unrealistic data that could negatively impact the model's results. These rows were easily filtered out using the customer segment column. Customer segment is column that describes what type of sector the customer works in, or if their account is a test account. Also, customers who didn't initiate any service usage were excluded to ensure meaningful data. These customers were identified by seeing if they have initiated more than one project within the service. Also, to avoid a short observation period for customers, we imposed a requirement of at least two weeks of tenure.

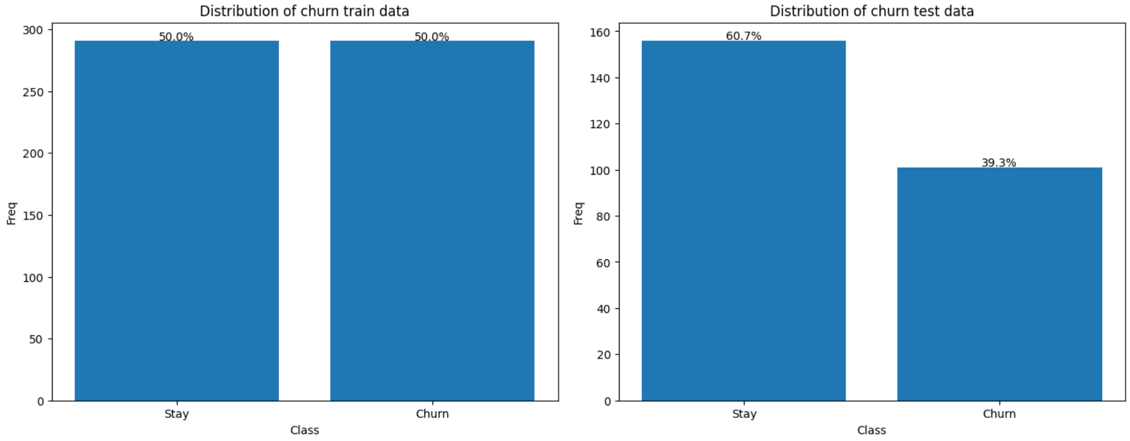
Moreover, as discussed in the Voluntary and Involuntary churn section, customers who churned involuntarily, for instance, due to bankruptcy, are not viable to include in the churn prediction. These customers were removed from the dataset and identified based on information retrieved from the case company.

3.8.2 Balancing

To address the challenge of imbalanced class distribution resampling techniques such as Synthetic Minority Over-sampling Technique combined with Edited Nearest Neighbors (SMOTE-ENN) was applied. This helps class imbalance by generating synthetic samples for the minority class while at the same time removing noisy instances. In that sense, the models can be trained on a more balanced dataset and improving their ability to accurately predict churn.

In Figure 3.6 and 3.7 the distribution for the training and test data for respective dataset can be viewed. The distribution showed for the training data is one after that SMOTE was applied to the dataset. As can be seen in the figure the training data is perfectly balanced between the classes.

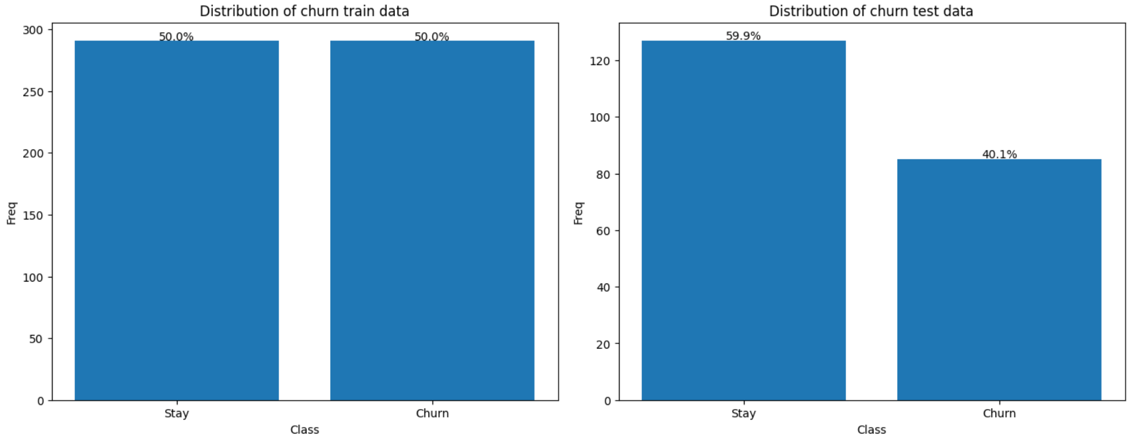
It is important to only balance the training data as the test data should mimic realistic inputs which are not balanced by nature. The balancing is only applied to the training data to improve results by ensuring the model sees an equal number of samples from both classes.



(a) Train data

(b) Test data

Figure 3.6: Data class distribution for, dataset: Short



(a) Train data

(b) Test data

Figure 3.7: Data class distribution for, dataset: Long

3.8.3 Splitting of Data

The data was split into train and test. Where 30% was split into the test set and the remainder of the data was utilized as training data. A validation set was not used in this case due to lack of data. Instead, more data was distributed between the train and the test set, to be able to achieve a better result.

3.8.4 Imputation

Because we had no missing data in our datasets no imputation was done.

3.8.5 Normalization

For normalization, standard score were used $\frac{X-\mu}{\sigma}$. Without normalization of the data done, biases might occur due to one feature being more dominant than other features. Therefore, by normalizing the data we ensure that all numerical columns will have the appropriate amount of weight on the model.

3.8.6 Encoding of Nominal Features

Lastly all nominal features were encoded. Nominal features are features such as the country a company is located in. This was done using one hot encoding.

3.9 Evaluation Metrics

The one major evaluation metric used were AUROC, its main use was for training the models. For further information regarding the metrics used when reviewing the performance of respective model see Section 2.5.

3.10 Architecture Selection

A logistic regression model was initially used as the base model due to its simplicity and ease of interpretation. Logistic regression is a straightforward linear classification technique that is particularly suitable for binary outcomes like churn prediction. Its simplicity makes it a suitable starting point serving as a benchmark against which more complex models can be evaluated.

Also, other alternative models were tested to improve the result of the base model, including the models; Random Forest, advanced gradient boosting algorithms such as XGBoost, GBC, and LightGBM. These models were chosen based on their performance in prior research within the area of churn prediction. As such, we could narrow the number of models to test to just a few based on their superiority in comparison to other models in this area.

3.11 Hyperparameter Tuning

Gridsearch was used for hyperparameter tuning optimization with cross-validation. Gridsearch tries combinations of hyperparameters, then returns the most optimal combinations of hyperparameters. The cross-validation is used to hinder overfitting the model. More so, this is especially important for smaller datasets as data variance has a tendency of being much lower.

3.12 Feature Selection With RFECV

For the feature selection backwards elimination with cross-validation was used. Backwards elimination is when we start with all the features that were gathered and remove one feature at a time. After removing each feature, we then compare the scores from all the features being removed and see which of the features gave the most increase in the metric used for scoring. This process of removing features is done until the model does no longer improve by removing features.

3.13 Deployment

The model's usage included interpreting its predictions through feature importance analysis, specifically using SHAP analysis (see 2.15). Additionally, deployment involved using the model to identify the customers at the case company with the highest probability of churning.

Chapter 4
Evaluation

4.1 Correlation Matrix

The correlation matrix constructed from one of the datasets (short) can be seen Figure 4.1. In this figure we can see major correlations both positive and inverse. However, we still see many features being red, meaning they have no correlation to another feature.

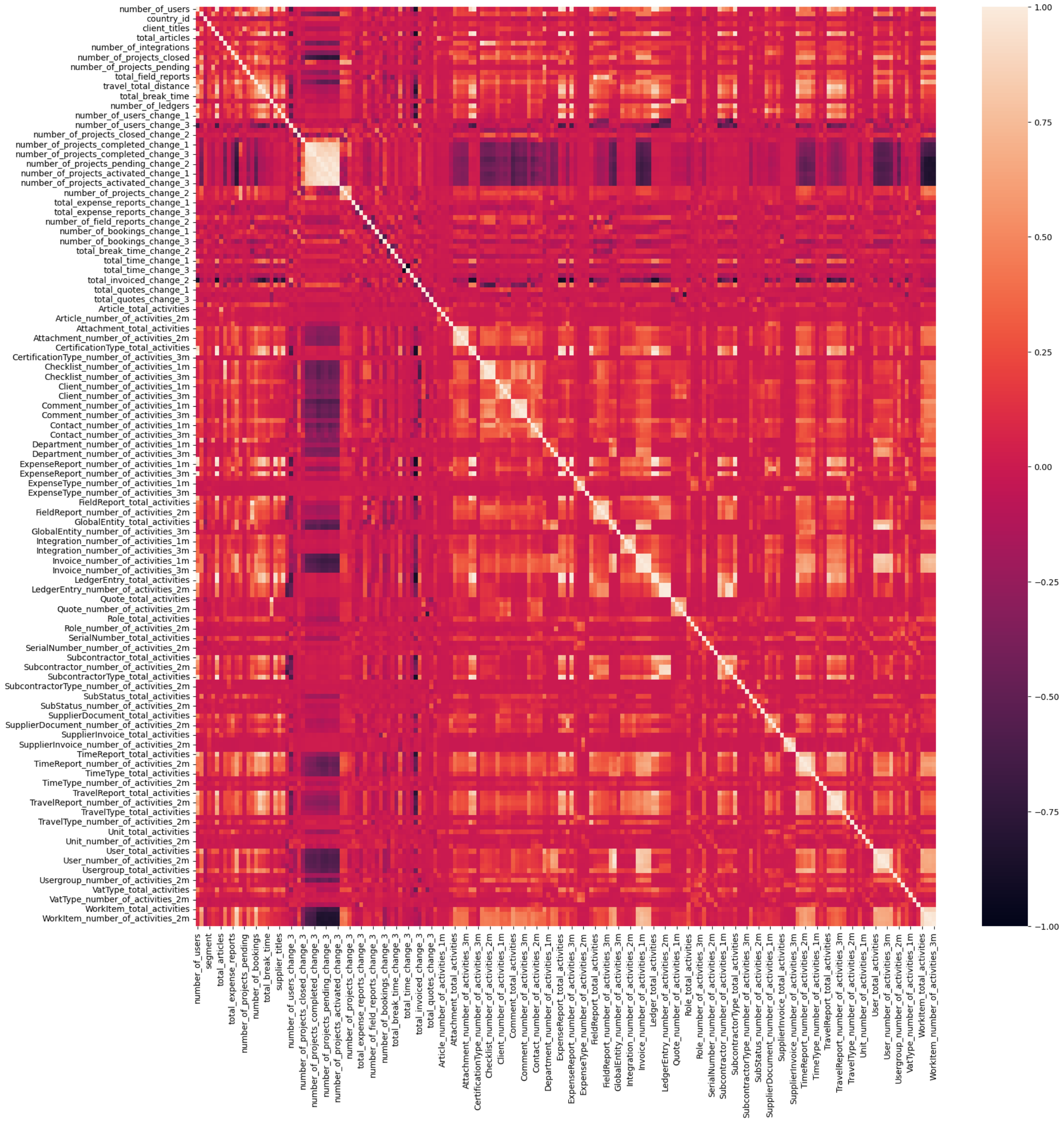


Figure 4.1: Correlation matrix

4.2 Models

The results of the models trained from the short and the long datasets are shown in Table 4.1 and Table 4.2. A higher overall performance can be seen from the short dataset. The models that consistently outperformed other models across most of the metrics in both datasets are the XGBoost models.

Model	f1_macro	AUROC	Accuracy	Recall	Precision	Specificity
Logistic Regression	0.712	0.798	0.712	**0.891**	0.588	0.596
Random Forest	0.776	0.867	0.782	0.792	0.696	0.776
XGBoost	**0.843**	**0.900**	**0.848**	0.832	**0.787**	**0.853**
LGBM	0.812	0.881	0.812	0.822	0.741	0.814
GBC	0.754	0.835	0.762	0.732	0.685	0.782

Table 4.1: Results for dataset: Short

Model	f1_macro	AUROC	Accuracy	Recall	Precision	Specificity
Logistic Regression	0.759	0.819	0.764	**0.765**	0.684	0.764
Random Forest	0.745	0.834	0.759	0.647	0.724	0.835
XGBoost	**0.770**	**0.858**	**0.783**	0.682	**0.753**	**0.850**
LGBM	0.740	0.832	0.755	0.647	0.715	0.827
GBC	0.745	0.832	0.759	0.647	0.724	0.835

Table 4.2: Results for dataset: Long

4.2.1 Result using different change variables

In Table 4.3 the results when using or not using the change variables for the short dataset are shown.

Dataset type	f1_macro	AUROC	Accuracy	Recall	Precision	Specificity
With change	**0.843**	0.900	**0.848**	**0.832**	**0.787**	0.853
Without change	0.819	**0.902**	0.829	0.762	0.794	**0.872**

Table 4.3: XGBoost, change variables, dataset: Short

4.3 AUROC

In Figure 4.2 and 4.3 the AUROC curves can be viewed for respective dataset. From these figures we can see that the short dataset achieved a higher AUROC score compared to the long dataset. Both the XGBoost models performing the best out of the models tested with Logistic Regression performing the worst. For the short dataset XGBoost got an AUROC of 0.90 and for the long dataset it scored an AUROC of 0.86.

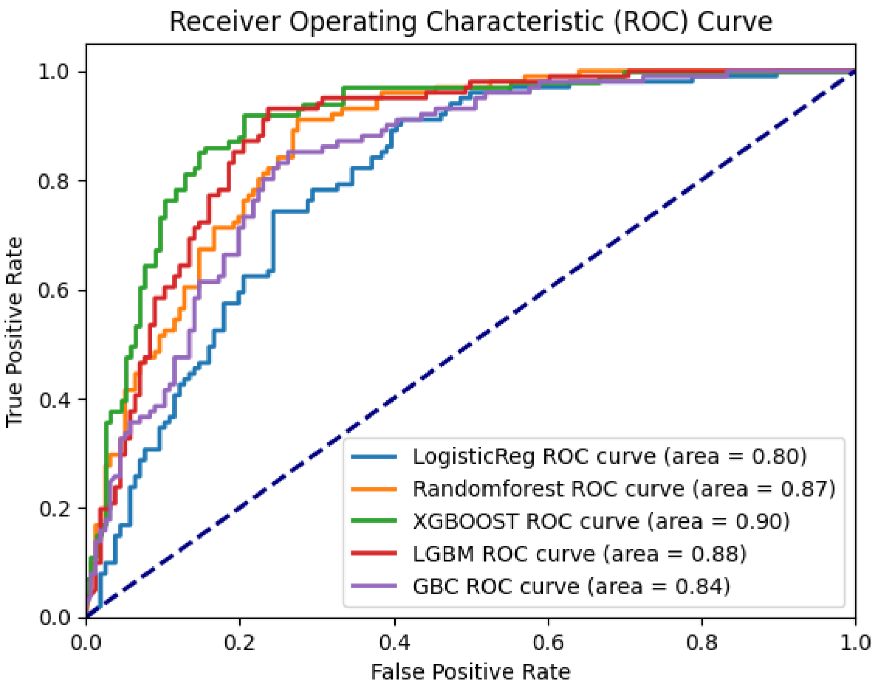


Figure 4.2: AUROC, dataset: Short

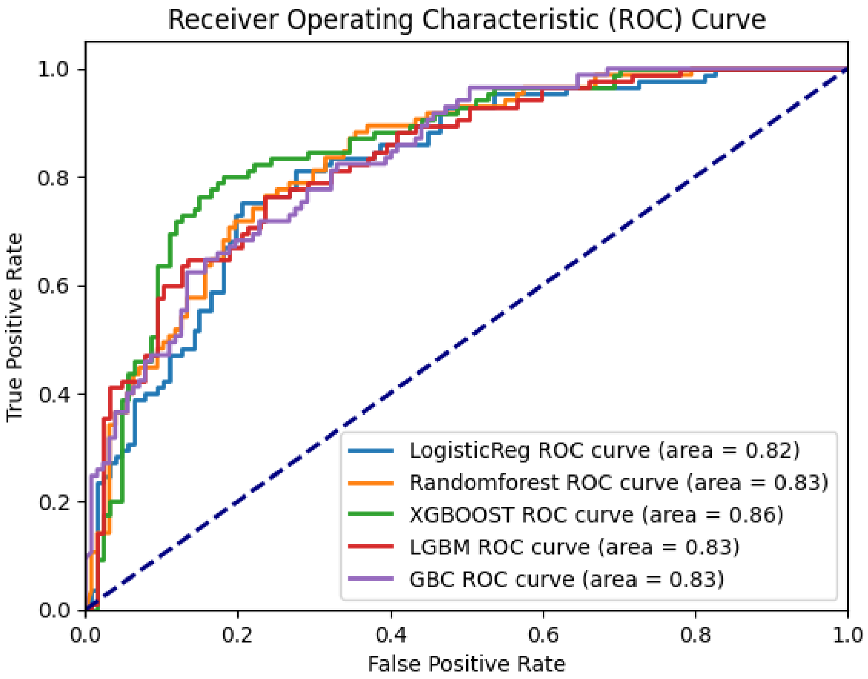


Figure 4.3: AUROC, dataset: Long

4.4 Result for XGBoost

Based on the collected results, it becomes evident that XGBoost outperformed other models, see Table 4.1 and Table 4.2. Upon further investigation into feature importance, which included basic analysis as well as more advanced analysis using SHAP (SHapley Additive exPlanations), we identified several features that played a significant role in the predictive performance. The accuracy, recall and AUROC metrics were primarily used to determine it as the top performing model.

4.4.1 Dataset Short

Here results specific for the short dataset will be shown. After feature selection, the long dataset comprised of 50 features, with 760 customers in the training data and 257 in the test data.

Confusion Matrix

In Figure 4.4 and 4.5 each classification made by XGBoost are shown. The top-left and bottom-right values show the classifications the model predicted correct, while top-right and bottom-left show the classifications the model predicted incorrect. We find by looking at the confusion matrix for the test set, that the model has more faulty False Positive predictions rather than False Negatives.

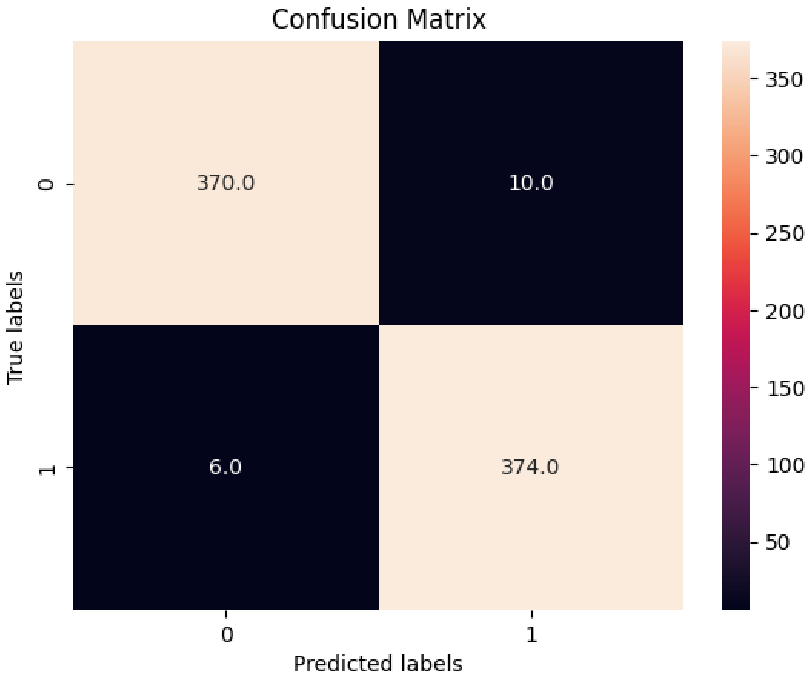


Figure 4.4: Confusion Matrix: Train, dataset: Short

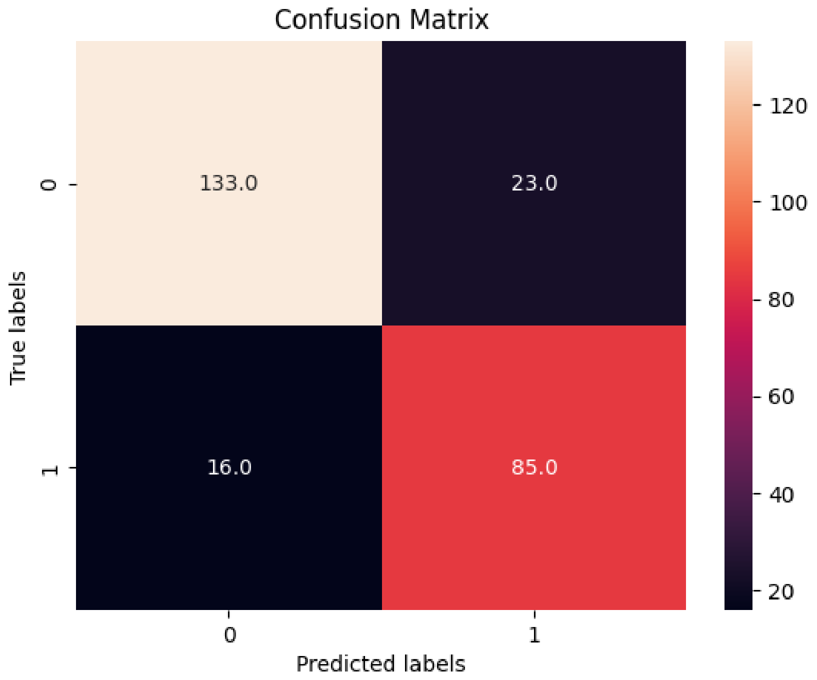


Figure 4.5: Confusion Matrix: Test, dataset: Short

Learning Curve

For the learning curve, see Figure 4.6. The loss of the test dataset stops to improve around a logarithmic loss of 0.4 and the train dataset a bit below 0.2. We can see that the training data starts to improve much faster than the test set and continues to improve while the test data stops on improving around the 200 mark.

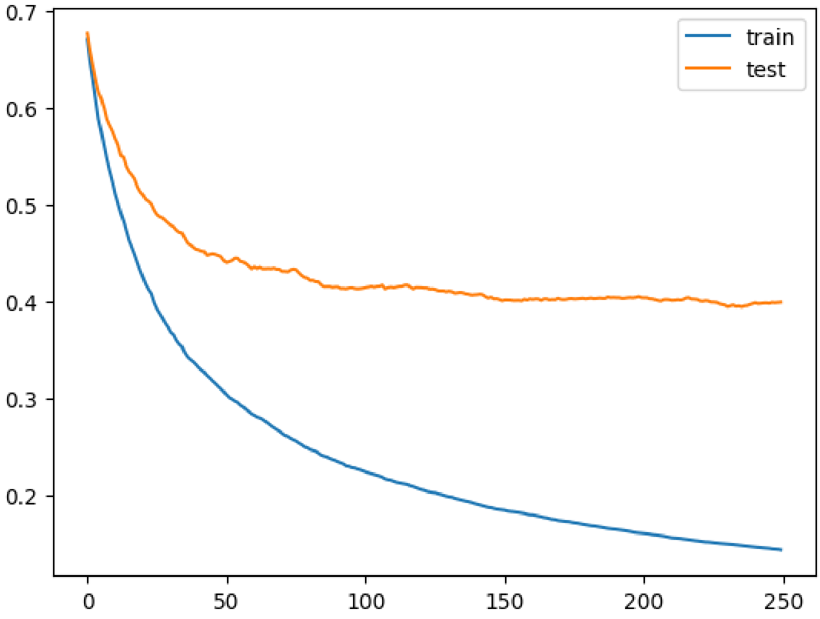


Figure 4.6: Learning Curve, dataset: Short

4.4.2 Dataset Long

Here results specific for the long dataset will be shown. After feature selection by the XGBoost model, the long dataset comprised of 105 features, with 668 customers in the training data and 212 in the test data.

Confusion Matrix

Figure 4.7 and 4.8 show each classification made by XGBoost. By inspecting the confusion matrix for the test dataset, we find that the model has more wrong predictions in the FN rather than the FP.

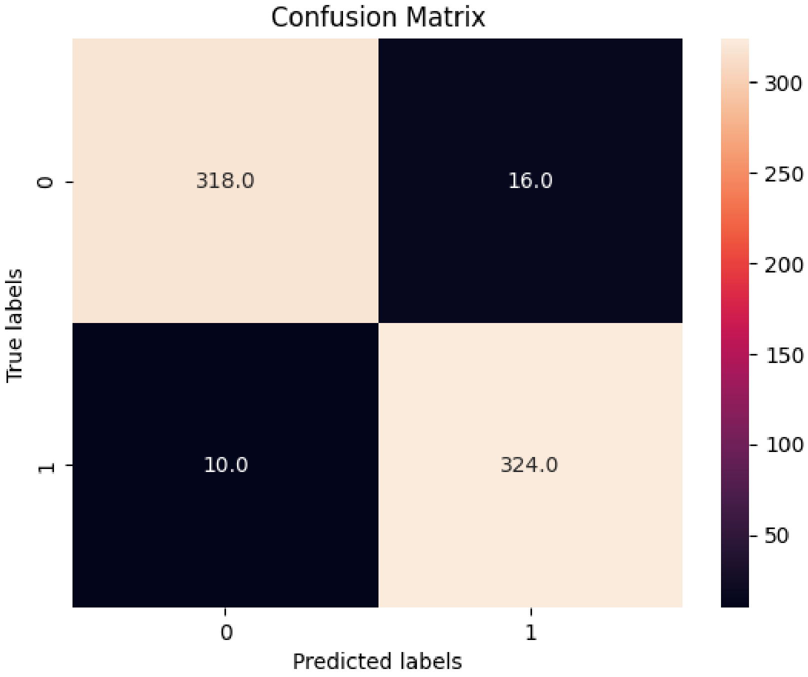


Figure 4.7: Confusion Matrix: Train, dataset: Long

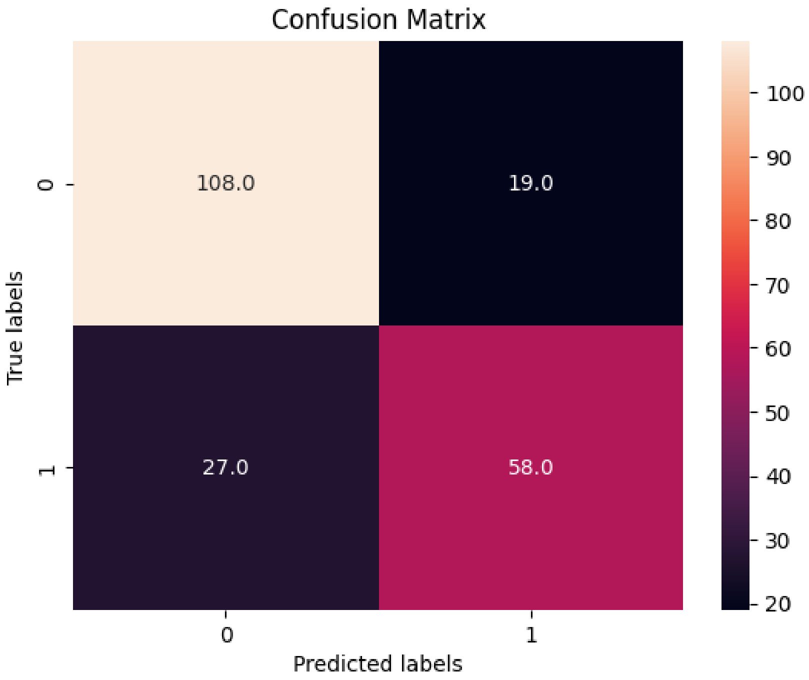


Figure 4.8: Confusion Matrix: Test, dataset: Long

Learning Curve

The learning curve can be viewed in Figure 4.9. We see that the curve for the test dataset stops improving around a logarithmic loss of 0.5 and the train dataset a bit below 0.2. We can see that the training data starts to improve much faster than the test set and continues to improve while the test data stops on improving after around the 100 mark.

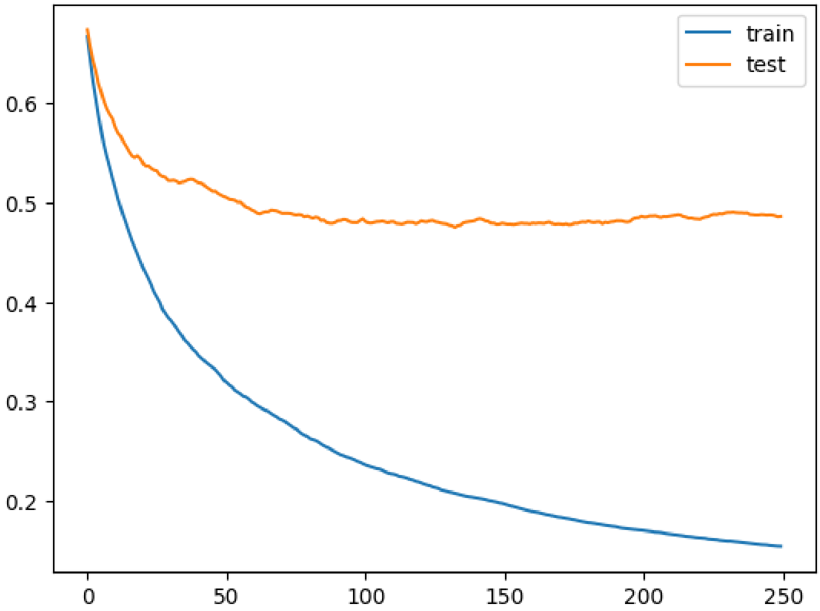


Figure 4.9: Learning Curve, dataset: Long

4.4.3 Feature Importance

The feature importance for the top 45 features in the XGBoost model can be viewed in Figure 4.10 and Figure 4.11. The most impactful features that affect churn are shown in these figures. We can see a good mix of variables in the feature importance figures. Total amounts, change between months and information such as tenure and how many clients a company has. In the short dataset, noteworthy features included the change in total break time from the previous month, the number of integration activities in the prior 2 months, and total expense report activities. Conversely, in the long dataset, key features comprised the number of expense reports from the prior month, total expense report activities, and the number of integration activities in the prior 2 months.

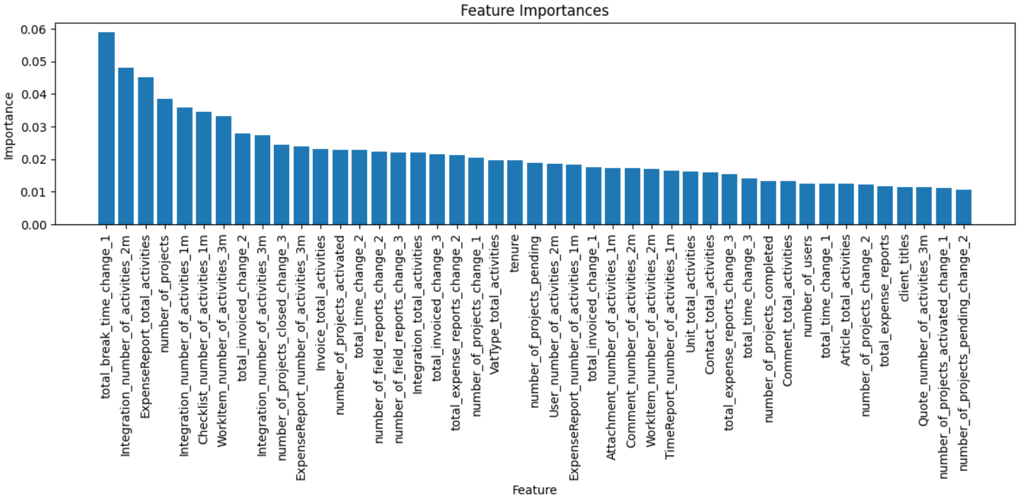


Figure 4.10: Feature importance, dataset: Short

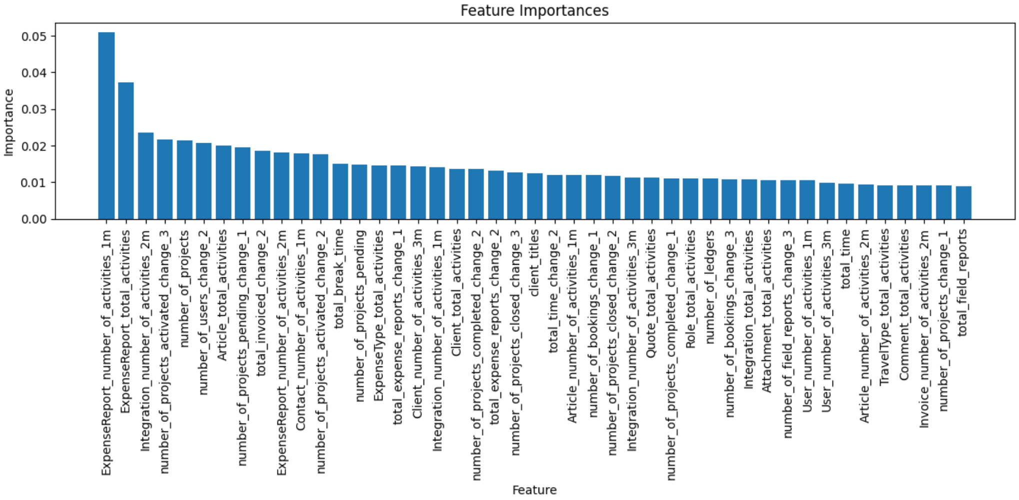


Figure 4.11: Feature importance, dataset: Long

4.5 SHAP

Following the creation and evaluation of the tested models, the XGBoost model was chosen as the top performer for both datasets based, primarily based on its result in accuracy, AU-ROC and recall. The SHAP method was employed to interpret the model's decision-making process.

We can see in Figure 4.12 and 4.13 the SHAP values for the feature importance for each of the datasets. These will be discussed more thoroughly in 5.2. This figure displays the most important features from top to bottom, giving information on how high and low sample values are linked to churn or no churn. A SHAP-value below zero indicates a connection to 'no churn,' while a SHAP-value above zero is associated with 'churn.' The color-coding uses blue for low values and red for high values of each feature. For the short dataset model, number of integration activities the prior month, total number of projects and integration total activities was the most important features. For the long dataset model, number of projects, tenure, and total expense report activities was the most important features.

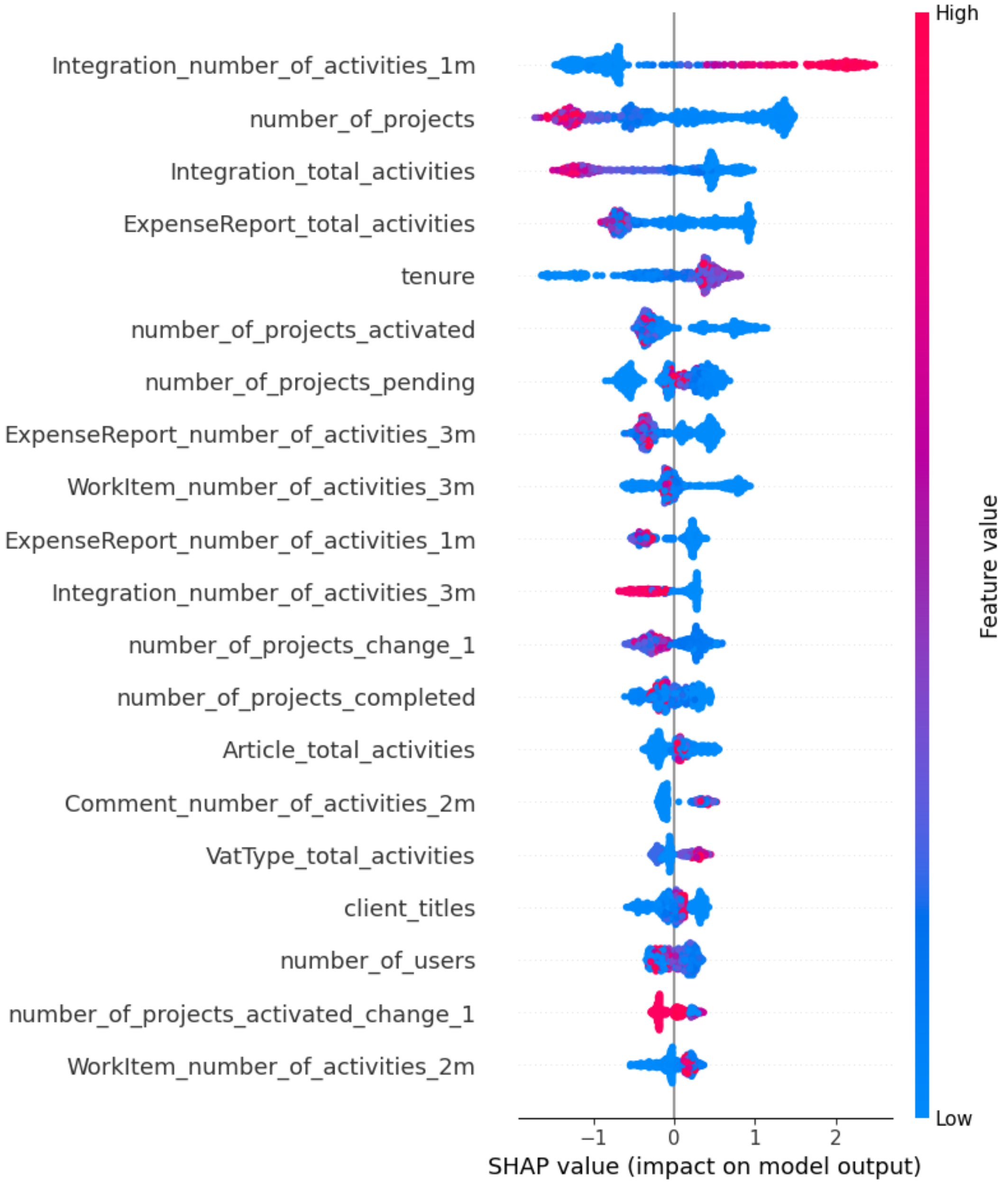


Figure 4.12: SHAP, dataset: Short

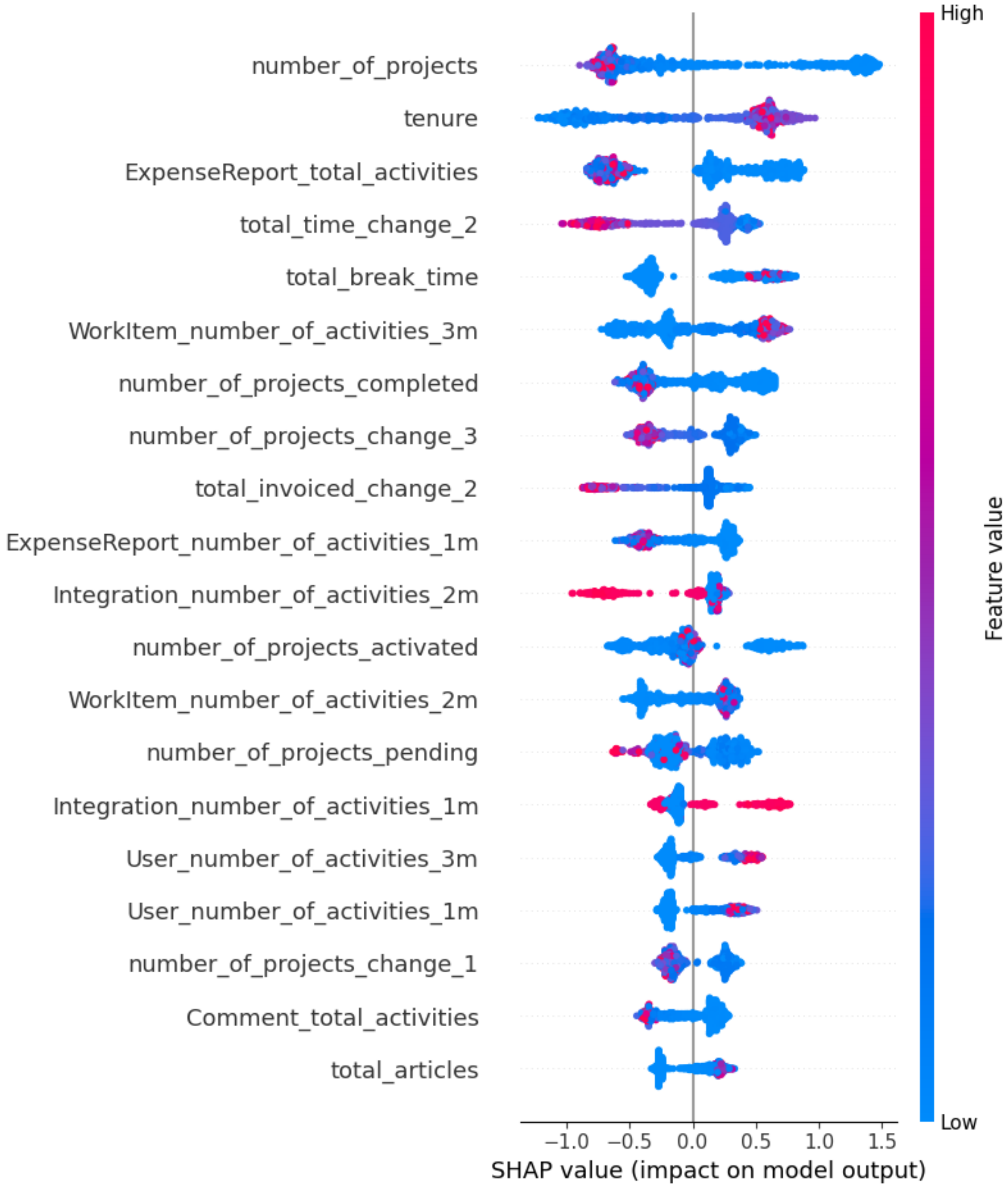


Figure 4.13: SHAP, dataset: Long

Chapter 5

Discussion

5.1 Model Prediction Performance

Based on the metrics gathered for each model prediction, the XGBoost model outperformed all other models for both the short and long datasets, as shown in Tables 4.1 and 4.2. In the short dataset, the XGBoost model excelled in all metrics, achieving an AUROC score of 0.900, accuracy of 0.848, and a recall score of 0.832. However, in the longer prediction dataset, as expected, performance declined due to having less information closer to the actual churn date.

Inspecting the confusion matrices for the test data for both the long and short datasets (see Figure 4.5 and Figure 4.8), we can identify where false classifications occur.

Analysis of the short dataset reveals that out of the customers the model predicted to churn (class 1), 85 were classified correctly, while 23 were falsely classified.

It's important to know that the model's predictions will impact the allocation of retention resources by the case company. For those customers incorrectly classified in class 1, it means applying retention strategies to customers who might not need it. However, this still helps improve the relationship with these customers.

The customers that were classified incorrectly in class 0 (no churn), are more critical than the ones classified incorrectly in class 1 (churn). These are customers in which the model predicted would not churn but actually did. In this situation, 16 customers who could have been retained will not receive retention efforts, which is a more critical issue and is defined by the recall score.

An important consideration is the integration of this model, particularly using it to predict only those customers with the highest probabilities of churning. This targeted approach can lead to more accurate customer retention efforts. Figure 5.1 illustrates the confusion matrix for the model with a churn probability threshold of 95%.

This threshold determines how the model classifies a customer as a churner based on their churn probability. In this case, with a threshold of 95%, a customer must have a churn probability higher than 95% to be classified as a churner. From the figure, we observe that only 5 out of 38 customers classified as churners did not actually churn, therefore receiving a very high accuracy within the customers predicted to churn. As expected, many customers that churn are miss-classified as non churners because of the high threshold. This strategy may not be entirely practical because customers with lower churn probabilities, even if not extremely high, could still be relatively easier to retain compared to those with probabilities exceeding 95%.

Focusing on the top critical customers predicted by the model can significantly improve retention efforts within the company.

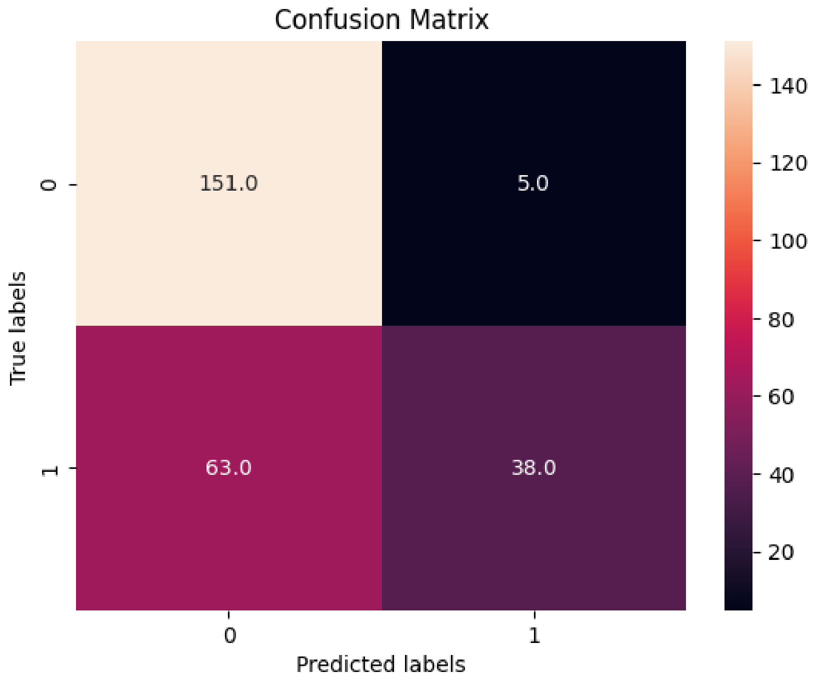


Figure 5.1: Confusion Matrix: Test, 95th percentile, dataset: Short

5.1.1 Change Variables

One approach we explored to improve model performance was to handle changes in activities during the months leading up to the churn date differently. In Figure 4.3, you can observe the results of three distinct approaches, all of which yielded similar outcomes across all three datasets. However, considering that the 'change' approach resulted in the highest recall score, we opted for it. Recall as mentioned above, is one of the main metrics to look for in churn prediction.

5.1.2 Data Size Validation

The dataset in the study was categorized into one with a short time frame consisting of 880 data rows and 50 features, and another with a long time frame consisting of 1017 data rows and 105 features. To evaluate whether these sizes are suitable for machine learning, we can apply the '10 times the number of rows as features' rule.

This rule is commonly used to ensure that there are an adequate number of data points relative to the number of features [46]. The goal of this is to strike a balance that allows the model to learn meaningful patterns in the data, avoid overfitting (where the model fits the noise rather than the signal), and improve its ability to generalize to unseen data.

For the shorter dataset, with 50 features, we would ideally want at least 500 rows. Since it contains a total of 880 rows it exceeds the minimum requirement. On the other hand, the longer dataset with 105 features falls slightly short with 1017 rows. One thing to note is that this study deals with a mid size B2B service and therefore the number of customers and therefore data rows are somewhat limited comparing to data size gathered from a B2C business. Having about 800 to 1000 customers, consisting of both churners and non churners should be able to recognize patterns within the customers to predict churn, but as most datasets in machine learning, more data rows would of course improve the credibility of the model.

In the case of the shorter dataset, we applied recursive feature elimination (RFE) to remove redundant features. But for the longer dataset this process was not as effective and resulted in a higher feature count. We believe this is because the longer prediction time, which requires the model to rely on a broader set of data for accurate forecasts and this then leading to a to higher ratio between the number of features and data rows.

Due to the limited size of our data it is important to limit the potential overfitting that could occur. We changed several parameters for our model and used regularization to prevent overfitting. Some examples of the parameters changed being the depth of the tree, the learning rate, the weight of the leafs, and the number of the estimators.

5.1.3 Comparison to Previous Studies

When comparing our results to other studies, we saw that other studies had higher scores comparatively to our result of 90.5% AUC and 84.4% accuracy for the short dataset. Gore et al. had an accuracy of 94% for their best model [9] and Abdelrahim et al. received and AUC score of 93.3% for their best model.

Gore et al. used SMOTE-ENN which they showed in their paper had a better score than SMOTE that we used [9]. The reason for not using SMOTE-ENN was because we deemed that the ENN would not be feasible to utilize due to limited data size. Furthermore Gore et al. had a better score when using ANN compared to a decision tree, 94% compared to 92% [9]. Due to time constraints, we did not test ANN, we might have seen an increase in the score of the best model if we did.

Abdelrahim et al. study proved that for its use case big data had an improvement in score, with access to upwards of 70 Terabytes of data in their study [27]. If we had more data to use in this thesis, we most likely would have seen an improvement in the scores of the models.

Furthermore, we purposefully removed data close to the last activity of all customers to simulate what a customer would have had for data prior to churn, or none churn. Thereby, limiting both the amount of data and potential indicators the model could have utilized, by limiting the model in this way, there will be a reduction in the potential accuracy of the models. When inspecting the result and comparing it between the short and the long dataset, we see that with more removed data, the model will perform worse.

5.2 Dissecting Feature Importance

In this section we will discuss features from the best model of both datasets.

5.2.1 SHAP Values, Dataset: Short

When inspecting the values in the SHAP graph we caught some interesting behaviour from the model. Two notable features were the tenure of a customer and the number of activities the most recent month (Integration_number_of_activities_1m).

The number of activities from the most recent month shows us that the model classifies customers with a very high number of activities to be much more likely to churn compared to those with a fewer of activities. This increase in the risk of customer churn can be due to a multitude of reasons, such as the customer not understanding the module and testing it frantically. Additionally, it could be that the customer in question only utilizes this one module of the service, thus having a very large amount of activity. Thereby, the customer could feel that they rather switch to using the other service which there is an integration to instead of having to use the integration.

The model scoring an increase of risk for those with high tenure could be a symptom of the contract length. It is very common to sell the service with a contract length of one year, which implies that the majority of customers are less likely to terminate the service before their contract term expires. This pattern can be corroborated by the information presented in Figure 1.2, where we notice a surge in customer churn at the end of each year. The contract length would then explain why a low tenure has a decrease in the risk of customer churn as well as an increased risk of customer churn for those of high tenure.

Three features that show more of an instinctive natural correlation between churn are the total number of projects (number_of_projects), total amount of activities for expense reports (ExpenseReports_total_activities) and the total amount of activities for integration's (Integration_total_activities). A higher total amount of projects made, activities in expense reports, and activities in integration's show that a decreased risk of customer churn. Contrary a lower amount would have an increase of customer churn. These two conclusions are obvious at only first glance, if a customer uses the service provided, they are more likely to stay with the service.

5.2.2 SHAP Values, Dataset: Long

We can see similarities between the SHAP values in the long dataset when compared to that in the short. For example, both datasets scored tenure, number of projects and the total amount of activity in expense reports the same.

However, two outliers for the long dataset is the amount of activity in projects the third most recent month (Workitem_number_of _activities_3m) and the total amount of break time (total_break_time). By having a lot of activity in projects the 3rd most recent month the model gives an increased risk of customer churn. As explained in the SHAP values for the short dataset a lot of activity can be correlated with a customer not understanding the system they use. For example, creating a project then changing the description multiple times. The total amount of break time giving an increased risk of churn for high values could be a symptom of users taking too many breaks or taking breaks too often. By taking too much time for breaks a company could be considered less efficient; making less money, then having less resources to spend on services such as the one the case company is providing.

5.2.3 Magnitude of Feature Importance

When looking at the feature importance for the two models in Figure 4.10 and Figure 4.11 we have a great number of features. The short dataset consists of 50 features, while the long dataset has 105 features. The model used numerous features for both datasets, however, the individual importance of these features tended to be relatively low. The most important feature for short is total break time, and this feature only gets importance score of 0.06, followed by other features with scores of 0.05, and 0.04. Most other features receive a score between 0.01 to 0.03. For the long dataset only two features receive an importance higher than 0.025, while the rest is around 0.01 to 0.02. This suggests that the predictive power for churn is not heavily reliant on a single feature but rather on the interplay of several features. Proving how important it is for a customer to use a variety of service features, as a single feature alone may not give a clear indication of churn.

5.3 Future Work

Future work in finding churn factors for B2B SaaS companies is mainly gathering a larger dataset and more companies to participate in a study. By increasing the data size and the variety of data in terms of being able to analyze more companies will enable a more generalized idea of why a customer would churn.

From the perspective of Fieldly, future work would concise of creating a larger dataset to train the models with. However, this can only be done over time when the case company grows and gathers more customers to analyze. Furthermore, adding important variables to the existing dataset, such as indicators of service failure and the quality of support customers receive would potentially increase the performance of the models.

While our predictive models can help pinpoint which features are influential in identifying a customer as a potential churner, addressing the "how" aspect of customer retention, as mentioned in 2.3.1, remains a more complex challenge. To determine what can be done to retain a customer once they've been identified as at-risk for churning, a deeper analysis is needed.

Furthermore, it's worth considering that exploring a variety of machine learning models beyond just ensemble methods could provide valuable insights. While our study predominantly utilized ensemble models, it might be beneficial to experiment with other techniques such as Neural Networks (ANN), as suggested in [9].

5.4 Threats of Validity

In this section we will discuss possible issues that could have skewed the data we gathered or influenced the study in a way that could have created biases.

5.4.1 Construct Validity

The construct validity is used to measure how well the concepts and theories are in relationship with what it was designed to evaluate [44]. The problem defined by us was to find potential factors that correlated to those who churn. The theories used were both for gathering data and analyzing the data. These two points are potential factors that could affect the construct validity.

Constructing a model with only internal data most likely have affected the outcome of this study. Without taking the recent pandemic and other external factors in to account the model might miss some crucial information. However, due to the removal of involuntary churn we believe that most cases were external factors that might have led to churn were removed. This is because external factors that would have led to churn probably are closely correlated to what we defined to be involuntary churn, such as bankruptcy.

We see that the gathering of data could result in issues, due to us missing some valuable information such as the amount of support tickets created or service failures that occur. By not being able to analyze these very important factors the measuring of potential churn factors could have looked different.

However, issues that could have occurred were minimized by following well established procedures and techniques for selecting, analyzing data.

5.4.2 Conclusion Validity

The conclusion validity is related to which degree the conclusions reached are correct. As seen by the correlation matrix produced, we had many dependant variables [44]. To get a good result, we needed to find models that could handle data with high correlation within it. All models except the logistic regression can handle this type of data. Furthermore, we also had feature selection that removed most of this bad data.

5.4.3 Internal Validity

The internal validity is the connection between how unknown factors might have changed observed variables without the knowledge of the researcher [44]. When being introduced to a database that has evolved during many years, there might be confusion of both deprecated and miss-used features taken from this data. By having a continuously updated database without any comprehensive guide to show what each part of the database is used for it is very possible some erroneous data have gone into the datasets created. Erroneous data both in the term of being faulty in what has been saved or by having just being data that has been used as tests. More-so we encountered test accounts that had not been labeled as test accounts, when using this database. However, most of these disappeared when removing accounts with less activity than 15 days.

5.4.4 External Validity

External validity is concerned with how easily the results of the findings are generalised to a real-world setting [44]. The scope of this study was to find potential churn factors for B2B

SaaS companies. By this definition the results would be able to be generalised to some degree. However, due to only having limited data to work with and that we only had access to use data from one company we might miss larger factors of churn for other companies. The methods used and theories applied to create models and analyze data can be used by others, but the conclusion might differ in what factors contribute the most to churn.

There are ethical aspects to be considered around data-sharing. We had access to data that consisted mostly of information that could be found be any person. However, we also had access to information such as how they used or interacted with the system provided. To avoid issues regarding GPDR and other laws, we did not disclose any information regarding the customers in this thesis. Furthermore, we signed NDA's and only used information that had a direct relationship with the service provided.

Chapter 6
Conclusion

The main objectives of the thesis included creating an effective customer churn prediction model to detect churn as early as feasible and exploring various machine learning models and strategies for our dataset. This involved tasks as selecting specific timeframes for analysis, addressing data imbalances, optimizing features and model parameters. For the churn prediction, a total of five models was trained and used: XGBoost, Gradient Boosting Classifier (GBC), Logistic Regression, Random Forest, and LightGBM.

The model that performed the best in the churn prediction was the XGBoost model. For the short dataset we scored an AUROC score of 90.0%, an accuracy of 84.8%, and a recall score of 83.2%. The long dataset had an AUROC score of 85.8%, an accuracy of 78.3%, and a recall score of 68.2%.

Regarding what factors had the most importance when predicting customer churn, we found for both datasets that tenure, number of projects and the total amount of activity in expense reports were important. Our analysis of short-term churn revealed that number of activities for integration the prior month and the overall activity level for integration's was important. In the long-term churn analysis, we observed the significance of the amount of activity in projects during the third most recent month and the total amount of break time. However, even though some features stuck out more than others, the general conclusion was that many features had low importance and that it is important to use all features to get a good overall churn prediction for a customer.

From the analysis of the datasets, we find that it is important for customers to engage in the features provided by the service. It matters less how small the engagement is. However, too large of an engagement for some features seem to correlate with an increased risk of churn. Moreover, when providing a service, it seems important that the customer understands and utilizes its features. A good start could be to make sure that the company providing the service, have a clear and easy interface with possibility of guidance when using the service.

Important factors such as product failure and support experience as described in Section 2.2.3 was missing from the data analyzed. Thus, leaving room for improvement for future models. Without the information of product failure and support experience we were still able to achieve a satisfactory result.

For future work for B2B customer churn prediction, it would be beneficial to gather a larger dataset involving more companies. An expanded dataset would provide more information to understand user behaviors and factors contributing to customer churn. Future work should delve deeper into addressing how to retain at-risk customers once they have been identified and testing alternative machine learning models. Additionally, for the specific case company, including new factors like product failure and support experience indicators in future analyses could enhance the accuracy of churn prediction.

References

[1] Chang Woojung, Park Jeong Eun, and Chaiy Seoil. How does crm technology transform into organizational performance? a mediating role of marketing capability. *Journal of Business Research*, 63(8):849–855, 2010.

[2] F F Reichheld and Jr Sasser, W E. Zero defections: quality comes to services. *Harvard business review*, 68(5):105 – 111, 1990.

[3] Tereza Šonková and Monika Grabowska. Customer engagement: transactional vs. relationship marketing. *The Journal of international studies*, 8:196–207, 2015.

[4] Robert W Palmatier et al. *Relationship marketing*. Marketing Science Institute Cambridge, MA, 2008.

[5] W. Bleuel. Cultivating the customer asset. volume 2, 1999.

[6] C.S. Gold. *Fighting Churn with Data: The science and strategy of customer retention*. Manning Publications, 2020.

[7] I Nyoman Mahayasa Adiputra and Paweena Wanchai. *Customer Churn Prediction Using Weight Average Ensemble Machine Learning Model*. IEEE / Institute of Electrical and Electronics Engineers Incorporated, 2023.

[8] Aryan Raj and D Vetrithangam. *Machine Learning and Deep Learning technique used in Customer Churn Prediction: - A Review*. IEEE / Institute of Electrical and Electronics Engineers Incorporated, 2023.

[9] Shubham Gore, Yuvraj Chibber, Manan Bhasin, Shyam Mehta, and S Suchitra. *Customer Churn Prediction using Neural Networks and SMOTE-ENN for Data Sampling*. IEEE / Institute of Electrical and Electronics Engineers Incorporated, 2023.

[10] Muhammad Joolfoo, Rameshwar Jugurnauth, and Khalid Joolfoo. Customer churn prediction in telecom using machine learning in big data platform. *Journal of Critical Reviews*, 7, 2020.

[11] Bingquan Huang, Mohand Tahar Kechadi, and Brian Buckley. Customer churn prediction in telecommunications. *Expert Systems with Applications*, 39(1):1414 – 1425, 2012.

[12] Pushkar Bhuse, Aayushi Gandhi, Parth Meswani, Riya Muni, and Neha Katre. *Machine Learning Based Telecom-Customer Churn Prediction.* IEEE / Institute of Electrical and Electronics Engineers Incorporated, 2020.

[13] Yizhe Ge, Shan He, Jingyue Xiong, and Donald Brown. Customer churn analysis for a software-as-a-service company. pages 106–111, 04 2017.

[14] Fieldly.com. Available: `https://sv.fieldly.com/`. [Accessed: Oct. 3, 2023].

[15] M. Amori. How to address customer churn with ai-driven data analysis. 2023.

[16] Kristina Kolic, Sasko Ristov, and Marjan Gusev. *A model of SaaS e-Business solution.* IEEE / Institute of Electrical and Electronics Engineers Incorporated, 2014.

[17] R. Singh, A. Bhagat, and N. Kumar. Generalization of software metrics on software as a service (saas). *2012 International Conference on Computing Sciences, Computing Sciences (ICCS), 2012 International Conference on, Communication Systems, International Conference on*, pages 267–270, 2012.

[18] Qualtrics. Customer churn: how to calculate, measure, and stop losing customers. 2023.

[19] F. Reichheld. Prescription for cutting costs. 2001.

[20] Price Intelliently. The comprehensive guide to churn. 2023.

[21] Investopedia. Product life cycle explained: Stage and examples. 2023.

[22] Nikola Apostolov. Reducing churn in data management saas companies: a case study, June 2020.

[23] Florian v. Wangenheim, Nancy V. Wünderlich, and Jan H. Schumann. Renew or cancel? drivers of customer renewal decisions for it-based service contracts. *Journal of Business Research*, 79:181–188, 2017.

[24] Ruth N Bolton, Katherine N Lemon, and Peter C Verhoef. The theoretical underpinnings of customer asset management: A framework and propositions for future research. *Journal of the academy of marketing science*, 32(3):271–292, 2004.

[25] Markus Blut, Heiner Evanschitzky, Christof Backhaus, John Rudd, and Michael Marck. Securing business-to-business relationships: The impact of switching costs. *Industrial Marketing Management*, 52:82–90, 2016.

[26] Dave Beulke. Big data impacts data management: The 5 vs of big data. *Available from: Big Data Impacts Data Management: The 5Vs of Big Data, accessed*, 21, 2011.

[27] Ahmad Abdelrahim Kasem, Jafar Assef, and Aljoumaa Kadan. Customer churn prediction in telecom using machine learning in big data platform. *Journal of Big Data*, 6(1):1 – 24, 2019.

[28] Nguyen Nhu Y., Tran Van Ly, and Dao Vu Truong Son. Churn prediction in telecommunication industry using kernel support vector machines. *PLoS ONE*, 17(5), 2022.

[29] Praveen Lalwani, Manas Kumar Mishra, Jasroop Singh Chadha, and Pratyush Sethi. Customer churn prediction system: a machine learning approach. *Computing*, 104(2):271 – 294, 2022.

[30] O. Pandithurai, H. Humaid Ahmed, Hrudhai Narayan. S, B. Sriman, and Seetha R. Telecom customer churn prediction using supervised machine learning techniques. *2023 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI), Advances in Computing, Communication and Applied Informatics (ACCAI), 2023 International Conference on*, pages 1 – 7, 2023.

[31] R. Peddarapu et al. Customer churn prediction using machine learning. 2023.

[32] Kevin P. Murphy. *Machine learning : a probabilistic perspective.* MIT Press, Cambridge, Mass. [u.a.], 2013.

[33] Laerd Statistics. Spearman's rank-order correlation (statistical guide). Available: `https://statistics.laerd.com/statistical-guides/spearmans-rank-order-correlation-statistical-guide.php`. [Accessed: Oct. 8, 2023].

[34] Aurlien Gron. *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems.* O'Reilly Media, Inc., 1st edition, 2017.

[35] Scikit Learn. Gradient boosting classifier. Available: `https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html`. [Accessed: Oct. 7, 2023].

[36] Lightgbm.readthedocs.io. Lgbm documentation. Available: `https://lightgbm.readthedocs.io/en/latest/Features.html`. [Accessed: Oct. 7, 2023].

[37] NVIDIA. Xgboost. Available: `https://www.nvidia.com/en-us/glossary/data-science/xgboost/`. [Accessed: Oct. 7, 2023].

[38] Neptune.ai. Xgboost vs lightgbm: How are they different. Available: `https://neptune.ai/blog/xgboost-vs-lightgbm`. [Accessed: Oct. 7, 2023].

[39] J. Burez and D. Van den Poel. Handling class imbalance in customer churn prediction. *EXPERT SYSTEMS WITH APPLICATIONS*, 36(3):4626 – 4636, 2009.

[40] simplilearn. Introduction to data imputation). Available: `https://www.simplilearn.com/data-imputation-article`. [Accessed: Oct. 8, 2023].

[41] Adi Zaenul Mustaqim, Sumarni Adi, Yoga Pristyanto, and Yuli Astuti. The effect of recursive feature elimination with cross-validation (rfecv) feature selection algorithm toward classifier performance on credit card fraud detection. *2021 International Conference on Artificial Intelligence and Computer Science Technology (ICAICST), Artificial Intelligence and Computer Science Technology (ICAICST), 2021 International Conference on*, pages 270 – 275, 2021.

[42] shap.readthedocs.io. Welcome to the shap documentation. Available: `https://shap.readthedocs.io/en/latest/`. [Accessed: Oct. 7, 2023].

[43] Hanae Errousso, El Arbi Abdellaoui Alaoui, Siham Benhadou, and Hicham Medromi. Exploring how independent variables influence parking occupancy prediction: toward a model results explanation with shap values. *Progress in Artificial Intelligence*, 11:367–396, 2022.

[44] Claes Wohlin, Per Runeson, Martin Höst, Magnus C. Ohlsson, Björn Regnell, and Anders Wesslén. *Experimentation in Software Engineering*. Springer Berlin Heidelberg, 2012.

[45] *The Production of Knowledge: Enhancing Progress in Social Science*. Cambridge University Press, 2020.

[46] Hrvoje Smolic. How much data is needed for machine learning? Available: `https://graphite-note.com/how-much-data-is-needed-for-machine-learning`. [Accessed: Nov. 11, 2023].

Appendices

Appendix A

Data Source Overview, Dataset: Short

Feature	Mean	Median	Stddev
account_id	3709.6	3867.5	1658.9
number_of_users	39.7	7.0	684.5
churn_status	0.4	0.0	0.5
number_of_projects	1006.5	187.5	4403.2
segment	0.3	0.0	0.8
client_titles	594.1	111.0	3218.1
number_of_subcontractors	7.2	0.0	136.5
total_articles	3340.3	48.5	19756.5
total_checklists	210.1	3.0	1187.9
number_of_integrations	0.2	0.0	0.9
total_expense_reports	1630.2	30.0	4063.0
number_of_projects_closed	650.8	38.0	2420.6
number_of_projects_completed	162.5	11.0	1018.5
number_of_projects_pending	139.5	2.0	2815.2
number_of_projects_activated	44.7	8.0	167.2
total_field_reports	354.3	5.0	1997.5
number_of_bookings	1167.3	199.0	3873.9
travel_total_distance	34875894.6	559000.0	156128547.0
total_time	88570687.6	15108892.0	387579427.4
total_break_time	943336.0	0.0	5011524.2
total_quotes	22.7	0.0	101.2
number_of_ledgers	4.4	0.0	43.3
supplier_titles	58.0	2.0	128.6
number_of_users_change_1	-0.0	0.0	4.5
number_of_users_change_2	-0.2	0.0	3.4

Feature	Mean	Median	Stddev
number_of_users_change_3	-0.1	0.0	4.4
number_of_projects_closed_change_1	-0.1	0.0	21.5
number_of_projects_closed_change_2	0.7	0.0	23.9
number_of_projects_closed_change_3	-2.8	0.0	27.0
number_of_projects_completed_change_1	-17.1	0.0	98.8
number_of_projects_completed_change_2	-17.6	0.0	90.0
number_of_projects_completed_change_3	-20.1	0.0	101.0
number_of_projects_pending_change_1	-22.1	0.0	81.9
number_of_projects_pending_change_2	-22.0	0.0	82.2
number_of_projects_pending_change_3	-25.1	0.0	93.2
number_of_projects_activated_change_1	-21.2	0.0	86.3
number_of_projects_activated_change_2	-21.4	0.0	80.9
number_of_projects_activated_change_3	-24.8	0.0	94.5
number_of_projects_change_1	16.5	4.0	65.5
number_of_projects_change_2	14.7	3.0	55.4
number_of_projects_change_3	10.5	2.0	48.0
total_expense_reports_change_1	16.4	0.0	67.8
total_expense_reports_change_2	-13.9	0.0	87.0
total_expense_reports_change_3	-1.8	0.0	80.2
number_of_field_reports_change_1	1.5	0.0	25.5
number_of_field_reports_change_2	-0.8	0.0	22.9
number_of_field_reports_change_3	-0.2	0.0	21.2
number_of_bookings_change_1	4.7	0.0	37.5
number_of_bookings_change_2	-0.3	0.0	49.6
number_of_bookings_change_3	-2.3	0.0	52.6
total_break_time_change_1	1035.2	0.0	48253.6
total_break_time_change_2	252.2	0.0	56519.3
total_break_time_change_3	-2089.7	0.0	33197.6
total_time_change_1	317110.2	0.0	2137041.0
total_time_change_2	-520329.9	0.0	9092266.4
total_time_change_3	230766.7	0.0	8757876.3
total_invoiced_change_1	1285420.4	0.0	47918986.1
total_invoiced_change_2	-19065178.8	0.0	178338169.9
total_invoiced_change_3	620319.8	0.0	131082409.2
total_quotes_change_1	0.2	0.0	3.3
total_quotes_change_2	-0.3	0.0	9.7
total_quotes_change_3	-0.1	0.0	3.8
tenure	919.8	796.5	590.3
Article_total_activities	5654.2	49.0	39091.8
Article_number_of_activities_1m	201.0	0.0	2182.3
Article_number_of_activities_2m	198.6	0.0	2986.7
Article_number_of_activities_3m	114.9	0.0	482.9
Attachment_total_activities	352.3	16.0	1403.3
Attachment_number_of_activities_1m	18.4	0.0	62.9
Attachment_number_of_activities_2m	16.9	0.0	61.2

Feature	Mean	Median	Stddev
Attachment_number_of_activities_3m	16.9	0.0	61.6
CertificationType_total_activities	0.1	0.0	1.3
CertificationType_number_of_activities_1m	0.0	0.0	0.0
CertificationType_number_of_activities_2m	0.0	0.0	0.1
CertificationType_number_of_activities_3m	0.0	0.0	0.1
Checklist_total_activities	192.2	1.0	1100.5
Checklist_number_of_activities_1m	11.3	0.0	75.8
Checklist_number_of_activities_2m	10.8	0.0	79.6
Checklist_number_of_activities_3m	9.6	0.0	66.8
Client_total_activities	190.1	49.0	427.6
Client_number_of_activities_1m	8.5	2.0	23.1
Client_number_of_activities_2m	9.2	1.0	28.3
Client_number_of_activities_3m	9.6	1.0	27.2
Comment_total_activities	131.3	4.0	763.9
Comment_number_of_activities_1m	6.3	0.0	36.1
Comment_number_of_activities_2m	6.8	0.0	46.2
Comment_number_of_activities_3m	7.2	0.0	46.6
Contact_total_activities	52.8	7.0	159.0
Contact_number_of_activities_1m	2.6	0.0	8.6
Contact_number_of_activities_2m	2.4	0.0	7.9
Contact_number_of_activities_3m	2.7	0.0	9.2
Department_total_activities	1.2	0.0	12.7
Department_number_of_activities_1m	0.0	0.0	0.2
Department_number_of_activities_2m	0.0	0.0	0.3
Department_number_of_activities_3m	0.0	0.0	0.2
ExpenseReport_total_activities	30604.1	975.0	600639.5
ExpenseReport_number_of_activities_1m	427.4	65.0	2509.2
ExpenseReport_number_of_activities_2m	287.6	55.5	690.9
ExpenseReport_number_of_activities_3m	1094.8	55.0	20324.8
ExpenseType_total_activities	1.8	0.0	5.2
ExpenseType_number_of_activities_1m	0.1	0.0	1.7
ExpenseType_number_of_activities_2m	0.1	0.0	0.8
ExpenseType_number_of_activities_3m	0.1	0.0	0.8
FederationType_total_activities	0.1	0.0	1.6
FederationType_number_of_activities_1m	0.0	0.0	0.0
FederationType_number_of_activities_2m	0.0	0.0	0.0
FederationType_number_of_activities_3m	0.0	0.0	0.0
FieldReport_total_activities	489.3	9.0	2580.6
FieldReport_number_of_activities_1m	21.5	0.0	93.8
FieldReport_number_of_activities_2m	18.9	0.0	83.3
FieldReport_number_of_activities_3m	19.5	0.0	85.4
GlobalEntity_total_activities	0.5	0.0	7.4
GlobalEntity_number_of_activities_1m	0.0	0.0	0.0
GlobalEntity_number_of_activities_2m	0.0	0.0	0.0
GlobalEntity_number_of_activities_3m	0.0	0.0	0.3

Feature	Mean	Median	Stddev
Integration_total_activities	1171.1	350.5	2062.4
Integration_number_of_activities_1m	35.1	0.0	74.8
Integration_number_of_activities_2m	44.9	20.0	74.3
Integration_number_of_activities_3m	66.4	70.5	85.6
Invoice_total_activities	1264.1	126.5	6908.5
Invoice_number_of_activities_1m	62.6	7.0	244.3
Invoice_number_of_activities_2m	61.0	6.0	293.8
Invoice_number_of_activities_3m	73.6	8.0	333.0
Ledger_total_activities	0.4	0.0	12.2
Ledger_number_of_activities_1m	0.0	0.0	0.0
Ledger_number_of_activities_2m	0.0	0.0	0.0
Ledger_number_of_activities_3m	0.0	0.0	0.0
LedgerEntry_total_activities	765.1	0.0	16140.8
LedgerEntry_number_of_activities_1m	9.6	0.0	131.1
LedgerEntry_number_of_activities_2m	8.5	0.0	124.2
LedgerEntry_number_of_activities_3m	9.7	0.0	151.8
Quote_total_activities	169.6	0.0	704.6
Quote_number_of_activities_1m	11.0	0.0	44.2
Quote_number_of_activities_2m	9.7	0.0	41.4
Quote_number_of_activities_3m	10.7	0.0	57.0
Role_total_activities	3.1	0.0	8.0
Role_number_of_activities_1m	0.1	0.0	0.9
Role_number_of_activities_2m	0.1	0.0	0.5
Role_number_of_activities_3m	0.1	0.0	0.6
SerialNumber_total_activities	5.3	3.0	9.4
SerialNumber_number_of_activities_1m	0.1	0.0	0.9
SerialNumber_number_of_activities_2m	0.2	0.0	1.2
SerialNumber_number_of_activities_3m	0.2	0.0	0.9
Subcontractor_total_activities	2.2	0.0	20.8
Subcontractor_number_of_activities_1m	0.1	0.0	0.4
Subcontractor_number_of_activities_2m	0.1	0.0	0.7
Subcontractor_number_of_activities_3m	0.1	0.0	1.3
SubcontractorType_total_activities	0.4	0.0	6.6
SubcontractorType_number_of_activities_1m	0.0	0.0	0.1
SubcontractorType_number_of_activities_2m	0.0	0.0	0.0
SubcontractorType_number_of_activities_3m	0.0	0.0	0.2
SubStatus_total_activities	2.2	0.0	13.7
SubStatus_number_of_activities_1m	0.1	0.0	0.7
SubStatus_number_of_activities_2m	0.1	0.0	0.5
SubStatus_number_of_activities_3m	0.1	0.0	0.6
SupplierDocument_total_activities	1510.9	89.0	4145.2
SupplierDocument_number_of_activities_1m	64.3	4.5	117.7
SupplierDocument_number_of_activities_2m	58.7	2.0	141.1
SupplierDocument_number_of_activities_3m	73.4	0.0	227.1
SupplierInvoice_total_activities	46.9	0.0	607.5

Feature	Mean	Median	Stddev
SupplierInvoice_number_of_activities_1m	0.6	0.0	13.7
SupplierInvoice_number_of_activities_2m	1.0	0.0	18.8
SupplierInvoice_number_of_activities_3m	1.5	0.0	31.8
TimeReport_total_activities	6281.4	1184.0	23331.4
TimeReport_number_of_activities_1m	292.4	106.0	606.0
TimeReport_number_of_activities_2m	259.7	93.0	523.6
TimeReport_number_of_activities_3m	281.6	93.0	645.3
TimeType_total_activities	16.7	3.0	48.6
TimeType_number_of_activities_1m	0.4	0.0	2.1
TimeType_number_of_activities_2m	0.5	0.0	2.6
TimeType_number_of_activities_3m	0.7	0.0	3.9
TravelReport_total_activities	1006.5	20.5	5549.7
TravelReport_number_of_activities_1m	40.3	0.0	141.9
TravelReport_number_of_activities_2m	35.3	0.0	121.7
TravelReport_number_of_activities_3m	40.9	0.0	149.4
TravelType_total_activities	1.4	0.0	5.6
TravelType_number_of_activities_1m	0.0	0.0	0.3
TravelType_number_of_activities_2m	0.1	0.0	0.8
TravelType_number_of_activities_3m	0.0	0.0	0.4
Unit_total_activities	5.0	2.0	7.7
Unit_number_of_activities_1m	0.1	0.0	0.9
Unit_number_of_activities_2m	0.1	0.0	0.7
Unit_number_of_activities_3m	0.2	0.0	0.9
User_total_activities	155.2	35.0	1099.2
User_number_of_activities_1m	4.4	0.0	39.5
User_number_of_activities_2m	5.6	0.0	58.2
User_number_of_activities_3m	5.5	0.0	50.9
Usergroup_total_activities	8.1	2.0	27.6
Usergroup_number_of_activities_1m	0.2	0.0	1.4
Usergroup_number_of_activities_2m	0.2	0.0	1.2
Usergroup_number_of_activities_3m	0.3	0.0	1.4
VatType_total_activities	2.9	2.0	5.0
VatType_number_of_activities_1m	0.1	0.0	0.4
VatType_number_of_activities_2m	0.1	0.0	0.6
VatType_number_of_activities_3m	0.1	0.0	1.0
WorkItem_total_activities	8742.1	1262.5	33145.0
WorkItem_number_of_activities_1m	340.2	76.0	1030.5
WorkItem_number_of_activities_2m	326.2	57.5	1049.8
WorkItem_number_of_activities_3m	326.0	62.0	986.9

Appendix B

Data Source Overview, Dataset: Long

Feature	Mean	Median	Stddev
account_id	3431.3	3549.0	1542.1
number_of_users	42.7	7.0	712.1
churn_status	0.3	0.0	0.5
number_of_projects	1020.0	198.0	4573.1
segment	0.3	0.0	0.8
client_titles	613.4	110.5	3465.9
number_of_subcontractors	8.0	0.0	143.7
total_articles	3028.7	49.5	19451.2
total_checklists	202.6	3.0	1070.0
number_of_integrations	0.2	0.0	0.8
total_expense_reports	1541.5	38.0	3720.4
number_of_projects_closed	661.2	46.0	2351.2
number_of_projects_completed	157.6	9.0	968.9
number_of_projects_pending	155.3	1.0	3093.6
number_of_projects_activated	37.9	6.0	148.7
total_field_reports	363.0	5.0	1992.0
number_of_bookings	1167.8	198.0	3821.7
travel_total_distance	34923814.4	781000.0	154341318.6
total_time	89382054.3	16500600.0	397958550.9
total_break_time	1008720.6	0.0	5217066.2
total_quotes	19.4	0.0	90.8
number_of_ledgers	5.0	0.0	46.7
supplier_titles	59.7	3.0	129.1
number_of_users_change_1	0.1	0.0	3.4
number_of_users_change_2	0.2	0.0	3.1

Feature	Mean	Median	Stddev
number_of_users_change_3	-0.2	0.0	3.5
number_of_projects_closed_change_1	1.3	0.0	22.6
number_of_projects_closed_change_2	7.5	0.0	31.7
number_of_projects_closed_change_3	1.5	0.0	19.2
number_of_projects_completed_change_1	-20.8	0.0	106.1
number_of_projects_completed_change_2	-13.2	0.0	86.4
number_of_projects_completed_change_3	-13.6	0.0	86.7
number_of_projects_pending_change_1	-27.0	-1.0	94.8
number_of_projects_pending_change_2	-20.5	0.0	79.6
number_of_projects_pending_change_3	-19.0	-0.5	79.5
number_of_projects_activated_change_1	-26.7	0.0	92.1
number_of_projects_activated_change_2	-19.5	0.0	77.7
number_of_projects_activated_change_3	-18.8	0.0	76.5
number_of_projects_change_1	17.6	4.0	72.0
number_of_projects_change_2	22.4	7.0	59.4
number_of_projects_change_3	13.5	2.0	66.1
total_expense_reports_change_1	10.6	0.0	71.8
total_expense_reports_change_2	30.8	0.0	93.7
total_expense_reports_change_3	-4.4	0.0	83.1
number_of_field_reports_change_1	2.0	0.0	17.5
number_of_field_reports_change_2	3.6	0.0	21.6
number_of_field_reports_change_3	-0.5	0.0	16.3
number_of_bookings_change_1	-1.6	0.0	144.2
number_of_bookings_change_2	17.4	1.0	151.6
number_of_bookings_change_3	5.7	0.0	53.3
total_break_time_change_1	1015.9	0.0	34718.9
total_break_time_change_2	4351.2	0.0	61240.3
total_break_time_change_3	2701.4	0.0	52416.7
total_time_change_1	116740.6	0.0	3632648.7
total_time_change_2	644462.5	0.0	3714572.3
total_time_change_3	-75263.3	0.0	3058173.9
total_invoiced_change_1	-10452693.4	0.0	316158570.1
total_invoiced_change_2	28983350.2	0.0	214197421.7
total_invoiced_change_3	-20541308.3	0.0	202422493.9
total_quotes_change_1	0.3	0.0	7.0
total_quotes_change_2	0.6	0.0	3.8
total_quotes_change_3	0.2	0.0	2.9
tenure	1021.6	875.5	576.9
Article_total_activities	5231.7	46.5	39775.5
Article_number_of_activities_1m	997.2	0.0	21886.9
Article_number_of_activities_2m	137.9	0.0	455.9
Article_number_of_activities_3m	91.4	0.0	278.9
Attachment_total_activities	340.5	15.5	1318.2
Attachment_number_of_activities_1m	21.6	0.0	78.9
Attachment_number_of_activities_2m	20.0	0.0	70.4

Feature	Mean	Median	Stddev
Attachment_number_of_activities_3m	17.9	0.0	75.2
CertificationType_total_activities	0.1	0.0	1.4
CertificationType_number_of_activities_1m	0.0	0.0	0.0
CertificationType_number_of_activities_2m	0.0	0.0	0.0
CertificationType_number_of_activities_3m	0.0	0.0	0.0
Checklist_total_activities	180.8	1.0	952.3
Checklist_number_of_activities_1m	12.3	0.0	75.2
Checklist_number_of_activities_2m	9.4	0.0	49.3
Checklist_number_of_activities_3m	6.5	0.0	37.7
Client_total_activities	182.7	47.5	401.0
Client_number_of_activities_1m	11.0	2.0	28.8
Client_number_of_activities_2m	10.3	2.0	24.4
Client_number_of_activities_3m	7.6	1.0	22.0
Comment_total_activities	124.6	4.0	670.8
Comment_number_of_activities_1m	9.2	0.0	63.7
Comment_number_of_activities_2m	7.6	0.0	47.6
Comment_number_of_activities_3m	6.2	0.0	40.7
Contact_total_activities	50.8	6.0	158.3
Contact_number_of_activities_1m	3.7	0.0	19.0
Contact_number_of_activities_2m	3.1	0.0	14.0
Contact_number_of_activities_3m	2.0	0.0	7.5
Department_total_activities	1.4	0.0	13.7
Department_number_of_activities_1m	0.0	0.0	0.2
Department_number_of_activities_2m	0.0	0.0	0.3
Department_number_of_activities_3m	0.0	0.0	0.1
ExpenseReport_total_activities	32945.5	1177.5	599152.5
ExpenseReport_number_of_activities_1m	1901.1	90.0	36971.4
ExpenseReport_number_of_activities_2m	965.1	73.5	14953.6
ExpenseReport_number_of_activities_3m	1085.6	50.5	21174.2
ExpenseType_total_activities	1.6	0.0	4.7
ExpenseType_number_of_activities_1m	0.1	0.0	0.9
ExpenseType_number_of_activities_2m	0.1	0.0	0.7
ExpenseType_number_of_activities_3m	0.1	0.0	1.2
FederationType_total_activities	0.1	0.0	1.7
FederationType_number_of_activities_1m	0.0	0.0	0.1
FederationType_number_of_activities_2m	0.0	0.0	0.0
FederationType_number_of_activities_3m	0.0	0.0	0.0
FieldReport_total_activities	494.3	8.0	2531.1
FieldReport_number_of_activities_1m	24.0	0.0	90.6
FieldReport_number_of_activities_2m	20.7	0.0	80.7
FieldReport_number_of_activities_3m	15.1	0.0	61.4
GlobalEntity_total_activities	0.6	0.0	8.0
GlobalEntity_number_of_activities_1m	0.0	0.0	0.0
GlobalEntity_number_of_activities_2m	0.0	0.0	0.1
GlobalEntity_number_of_activities_3m	0.0	0.0	0.0

Feature	Mean	Median	Stddev
Integration_total_activities	1152.9	413.5	1967.7
Integration_number_of_activities_1m	71.1	116.0	82.5
Integration_number_of_activities_2m	69.2	82.0	79.2
Integration_number_of_activities_3m	66.0	35.0	79.3
Invoice_total_activities	1209.0	109.0	6340.7
Invoice_number_of_activities_1m	78.5	10.5	416.7
Invoice_number_of_activities_2m	70.9	6.0	387.7
Invoice_number_of_activities_3m	48.7	3.0	264.8
Ledger_total_activities	0.5	0.0	13.4
Ledger_number_of_activities_1m	0.0	0.0	0.0
Ledger_number_of_activities_2m	0.0	0.0	0.0
Ledger_number_of_activities_3m	0.0	0.0	0.0
LedgerEntry_total_activities	878.8	0.0	17410.4
LedgerEntry_number_of_activities_1m	12.3	0.0	172.4
LedgerEntry_number_of_activities_2m	10.0	0.0	123.4
LedgerEntry_number_of_activities_3m	7.9	0.0	86.7
Quote_total_activities	152.3	0.0	660.6
Quote_number_of_activities_1m	16.7	0.0	97.3
Quote_number_of_activities_2m	9.7	0.0	38.8
Quote_number_of_activities_3m	6.4	0.0	36.1
Role_total_activities	3.1	0.0	8.0
Role_number_of_activities_1m	0.1	0.0	0.8
Role_number_of_activities_2m	0.2	0.0	2.2
Role_number_of_activities_3m	0.1	0.0	1.1
SerialNumber_total_activities	5.3	3.0	9.6
SerialNumber_number_of_activities_1m	0.2	0.0	2.2
SerialNumber_number_of_activities_2m	0.2	0.0	0.8
SerialNumber_number_of_activities_3m	0.2	0.0	1.3
Subcontractor_total_activities	2.3	0.0	20.3
Subcontractor_number_of_activities_1m	0.1	0.0	1.6
Subcontractor_number_of_activities_2m	0.1	0.0	0.5
Subcontractor_number_of_activities_3m	0.0	0.0	0.4
SubcontractorType_total_activities	0.5	0.0	7.2
SubcontractorType_number_of_activities_1m	0.0	0.0	0.1
SubcontractorType_number_of_activities_2m	0.0	0.0	0.0
SubcontractorType_number_of_activities_3m	0.0	0.0	0.3
SubStatus_total_activities	2.2	0.0	14.7
SubStatus_number_of_activities_1m	0.2	0.0	2.6
SubStatus_number_of_activities_2m	0.0	0.0	0.4
SubStatus_number_of_activities_3m	0.0	0.0	0.5
SupplierDocument_total_activities	1502.3	95.0	4139.0
SupplierDocument_number_of_activities_1m	84.5	8.0	202.0
SupplierDocument_number_of_activities_2m	73.9	3.5	158.4
SupplierDocument_number_of_activities_3m	55.9	1.0	162.1
SupplierInvoice_total_activities	51.0	0.0	660.9

Feature	Mean	Median	Stddev
SupplierInvoice_number_of_activities_1m	0.5	0.0	12.3
SupplierInvoice_number_of_activities_2m	0.0	0.0	1.0
SupplierInvoice_number_of_activities_3m	0.3	0.0	7.1
TimeReport_total_activities	6234.7	1245.0	23545.6
TimeReport_number_of_activities_1m	319.4	138.5	636.6
TimeReport_number_of_activities_2m	298.0	109.5	620.9
TimeReport_number_of_activities_3m	236.5	80.5	514.0
TimeType_total_activities	17.3	4.0	50.8
TimeType_number_of_activities_1m	0.7	0.0	4.0
TimeType_number_of_activities_2m	0.5	0.0	2.3
TimeType_number_of_activities_3m	0.6	0.0	4.3
TravelReport_total_activities	1026.2	19.0	5545.0
TravelReport_number_of_activities_1m	46.6	0.0	165.6
TravelReport_number_of_activities_2m	44.2	0.0	149.9
TravelReport_number_of_activities_3m	30.0	0.0	98.7
TravelType_total_activities	1.4	0.0	5.6
TravelType_number_of_activities_1m	0.0	0.0	0.3
TravelType_number_of_activities_2m	0.0	0.0	0.3
TravelType_number_of_activities_3m	0.0	0.0	0.3
Unit_total_activities	4.8	1.0	7.4
Unit_number_of_activities_1m	0.1	0.0	0.8
Unit_number_of_activities_2m	0.2	0.0	0.8
Unit_number_of_activities_3m	0.2	0.0	1.4
User_total_activities	160.1	36.5	1006.4
User_number_of_activities_1m	6.1	0.0	45.5
User_number_of_activities_2m	5.8	0.0	33.2
User_number_of_activities_3m	5.1	0.0	39.8
Usergroup_total_activities	8.4	2.0	28.6
Usergroup_number_of_activities_1m	0.3	0.0	1.8
Usergroup_number_of_activities_2m	0.2	0.0	1.3
Usergroup_number_of_activities_3m	0.3	0.0	1.6
VatType_total_activities	2.9	2.0	5.2
VatType_number_of_activities_1m	0.0	0.0	0.4
VatType_number_of_activities_2m	0.1	0.0	1.1
VatType_number_of_activities_3m	0.1	0.0	0.5
WorkItem_total_activities	8958.1	1462.5	32400.1
WorkItem_number_of_activities_1m	388.2	91.0	1228.4
WorkItem_number_of_activities_2m	376.0	80.0	1213.8
WorkItem_number_of_activities_3m	278.4	50.0	1001.7

INSTITUTIONEN FÖR DATAVETENSKAP | LUNDS TEKNISKA HÖGSKOLA | PRESENTERAD 2023-10-26

EXAMENSARBETE Machine Learning-based Prediction of Customer Churn in SaaS
Enhancing Customer Retention with Predictive Analytics
STUDENTER Daniel Dahlén, William Mauritzon
HANDLEDARE Alma Orucevic Alagic (LTH), Salomeh Kiani Johnsson (Fieldly AB)
EXAMINATOR Mathias Haage (LTH)

Prevent Churn Before It Strikes, Thanks to Machine Learning

POPULÄRVETENSKAPLIG SAMMANFATTNING **Daniel Dahlén, William Mauritzon**

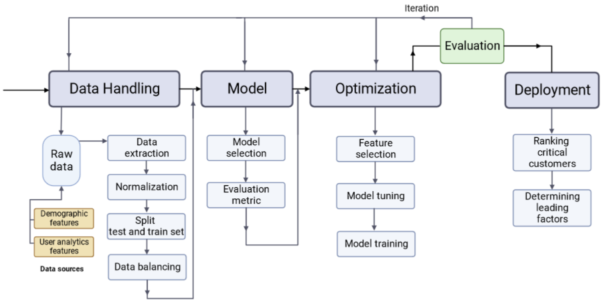
Imagine having the power to predict when a valuable customer might slip away from your business. This is precisely what our thesis aimed to achieve - the ability to predict customer churn and enable companies to take suitable preventive measures in time.

Customer churn is a pressing issue for businesses because it represents the rate at which customers leave. When customers decide to leave, companies suffer considerable revenue losses and their investments in acquiring and serving those customers go to waste. Addressing churn is not only crucial for maintaining financial stability and long-term growth but also because retaining existing customers is often more cost-effective than constantly trying to attract new ones. As a result, we have decided to provide a solution for companies to tackle the challenge of high customer churn.

Our solution includes using customer data to train a machine learning model to determine the likelihood of a customer churning. We use this probability to classify customers into churners and non-churners. The customers classified as churners are then ranked from most critical to least critical.

Our best model achieved an AUROC score of 0.905, an accuracy of 0.844, and a recall score of 0.832, demonstrating its effectiveness in identifying potential churn.

The machine learning process consisted of the stages: data handling and extraction, model selection, optimization, evaluation, and lastly deployment. In the data handling stage, data is excluded from each customer before their most recent activity to simulate the customer's state at a point in time before potential churn.



Our feature analysis revealed significant features correlated to churn including; project involvement, activity in expense reports, and integration-related activities. Surprisingly, excessive engagement with specific features is linked to a higher churn risk. This risk could result from factors like customer confusion or overuse of a single module, potentially leading them to consider switching to a specialized service.

The primary application of our model is to utilize it to make it easier for companies to prioritize retention efforts to individual customers. Additionally, by examining the feature analysis, a company can increase their efforts in specific areas of their service to improve upon.