Wrocław University
of Science and Technology

Faculty of Pure and Applied Mathematics
Field of study: Applied Mathematics
Specialty: –

Engineering Thesis

EXPLORING MACHINE LEARNING MODELS FOR CHURN PREDICTION OF MEMBERSHIP SUBSCRIPTIONS

Dawid Marczak

keywords:
machine learning, supervised learning, churn prediction, classification, membership, subscription, renewal, BNI

short summary:
The aim of this work is to test and compare various Machine Learning algorithms on BNI company data to predict if a member will be renewing their membership, thus focusing on a so-called churn prediction problem. This thesis was completed in cooperation with BNI Alberta South company, which shared its data for this thesis. The experimental part evaluated what is the performance of the machine learning algorithms fitted to the data as well as which features are most important according to tested models.

Supervisor	Dr inż. Radosław Michalski	.....	.....
	Title/degree/name and surname	grade	signature

*For the purposes of archival thesis qualified to:**

a) category A (perpetual files)

b) category BE 50 (subject to expertise after 50 years)

** delete as appropriate*

stamp of the faculty

Wrocław, 2022

Contents

List of Figures

List of Tables

Abstract

Data Science is a rapidly growing field in both technology and academia as it finds more use-cases across industries. Having in mind the broad range of machine learning capabilities, one could consider its subset of binary classification as "mundane", but it also has its use in the modern world. The Subscription Insitute states in its March 2021 report[1] that the subscription-based economy has grown six times since 2012, and subscription businesses have grown five to eight times faster than traditional businesses. This shift towards recurrent payments makes the topic of churn prediction all the more relevant. BNI Alberta South is an example of a company with a subscription business model and agreed to share its data for conducting this thesis paper. This study aims to explore different machine learning models on the BNI data, test how well the algorithms perform in the task of predicting member non-renewals, and compare which features are most critical for churn prediction, according to each of the models. This thesis also includes a description of the process of transforming the data into a machine learning-usable format, feature engineering, data exploration, and model testing.

[1] `http://info.zuora.com/rs/602-QGZ-447/images/Zuora_SEI_Report_2021.pdf`

Chapter 1

Introduction

Along with the advances of computer hardware capabilities such as graphical processing units and computing processing units, the field of machine learning which relies on computing power is also growing. The beginnings of machine learning classification task dates back to 1960[20], but this topic is still relevant today, specifically for predicting membership renewals and non-renewals. Although the topic of machine learning models for churn prediction has already been researched by Asthana[1], Lalwani et al.[12], Khan et al. [11] and Nie et al.[14], it is worth noting that this problem is usually tackled within the industry of telecommunication or banking. As companies continue to collect data about their users and shift towards a subscription business model, they unlock the potential of machine learning classification for churn prediction.

1.1 About BNI

Business Networking International (BNI) is an American networking organization established in 1985 which operates in over 70 countries and has over 200,000 members worldwide[1]. As its main service, BNI offers membership to business owners to become a part of a local networking group called "chapter" as well as gain access to the network of business professionals within the organization. Members meet within their chapters every week and, as a form of advertising, can refer other members' companies, resulting in new business opportunities. BNI subscription is usually one-year long at an estimated cost of $1000 USD for a new membership and $700 USD for every consecutive renewal, although options and prices may vary from country to country. The company's operational structure be divided into levels or units:

1. Global
2. Country
3. Region
4. Chapter
5. Member

where Global is the biggest unit containing countries within the BNI organization, countries can contain multiple regions, region multiple chapters and a chapter typically consists

[1] `https://www.bni.com/about`

of 20 to 50 members. Each unit except for Member has its own leadership team but is important to note that Region leadership may differ: some regions are franchised and some are owned by BNI Global. BNI Alberta South is an example of a Canadian BNI Region franchised by Moji Ajele, who kindly agreed to share the company data for this thesis. In 2020 BNI Alberta South had around 500 members and over 20 chapters in the Canadian province of Alberta, in cities such as Calgary, Red Deer, and Lethbridge. The only membership option available in BNI Alberta South is a one-year subscription. Throughout the rest of this thesis, the BNI Alberta South region will be referred to as "BNI ABS" for simplicity.

1.2 Data

The shared data was acquired from an internal BNI system and consists of multiple files. The key data which was used as input for machine learning models were in "PALMS reports" shared in a monthly format for January 2015 to October 2021 period and contained measures of member attendance, referrals, activity, education units, amongst others. Two other datasets which contained general information about members were needed and utilized to transform and prepare the data for machine learning modelling.

1.3 Thesis goals

One of the most important measures of success for companies with subscription-based payment models is retention rate: percentage of customers who continue to pay for a product/service over a given timeframe. Increasing this rate through convincing members to renew their membership would mean profit for the company. If one would consider this thesis as a project for developing a machine learning solution in cooperation with BNI, then its goals can be split into two categories:

Business goal

Create a solution for calculating probability of member not-renewal which will allow determining a group of members within a specific non-renewal probability, act upon this data and convert the said group to renew their memberships, ultimately increasing retention rate by 5%.

Project goals

1. Devise a machine learning model for predicting who won't be renewing their membership with a Receiver Operation Characteristic area under curve score[8] of at least 80%,

2. Determine which features are most important according to each of the algorithms that can be acted upon in the process of converting a non-renewing member to renewing.

The business goal is out of the scope of this thesis as it depends strictly on the company itself, but reaching it is a potential succession of completing the project goals, which will be the focus of this thesis.

1.4 Methodology

All programming needed for this thesis has been completed with the scripting language Python in Anaconda Jupyter Notebooks with the use of packages such as pandas[15], numpy[10] and scikit-learn[16]. The repository containing the code and results of the thesis is publicly available on GitHub[2] but it does not contain any data to preserve BNI members privacy. Below is a short overview of phases necessary to complete this thesis project:

- Initially, the PALMS data was in a monthly format and lacked any labels, which meant it was not ready for machine learning modelling. During this phase, the data was anonymized, transformed into a machine learning-usable format, and labeled. Additionally, new features were created out of the data,
- Exploratory Data Analysis – the goal of this phase was to explore the data and learn its quality using summary statistics and graphical representations,
- Modelling steps – the aim of this phase was to fit existing implementations of machine learning models to the data, test and improve their performance, and analyze which features are most important according to each algorithm.

1.5 Summary

This chapter described the topic of this thesis paper, the BNI company, and the data shared by this organization. It also informed the reader of the business and project goals set for this thesis, as well as the methodology employed to reach those goals.

[2] `https://github.com/da-veed/thesis_bachelor`

Chapter 2

Data Preparation

Even though the amount of data is increasing every day; if the data can't be used to produce valuable insights, it is useless. Specifically, the machine learning classification task requires the data to satisfy two criteria: contain appropriate class labels, and one row of data must represent one instance. The same principles apply to the BNI data, although initially, it met neither of the two mentioned criteria. The goal of this chapter was to describe both the thought process and methodology for cleaning and preparing the data as a necessary step of a machine learning project.

2.1 Methodology

As the thesis business goal is to increase BNI's member retention rate, it would be useless to create a model to predict member renewal, only to find out that the member had already left. BNI ABS membership is 12 months long, and so, the approach undertaken for this project was to aggregate 9 out of 12 months of PALMS data as input for the machine learning algorithm, thus leaving the company three months with the information of member churn probability to act on. This way, the machine learning input and output components can be denoted as follows:

- $x = [x_1, x_2, \ldots, x_n]$ reflects 9-months of PALMS data of a single subscription, where x_1, x_2, …, x_n are values in consecutive columns.
- $h(x) \in [0, 1]$ is the model probability of the member not renewing their membership.
- $\hat{y} \in \{0, 1\}$ is the label predicted by the machine learning algorithm. Usually:

$$\hat{y} = \begin{cases} 1, & h(x) \geq 0.5 \\ 0, & h(x) < 0.5 \end{cases}$$

  but the $\hat{y}$ class threshold can be changed from 0.5 to any number between 0 and 1.

To meet the second necessary criteria of classification learning, the data x had to be labeled based on members' renewal and drop dates in the following manner:

$$y = \begin{cases} 1, & \text{"Positive class" – member did not renew their membership} \\ 0, & \text{"Negative class" – member renewed their subscription} \end{cases} \tag{2.1}$$

It might seem unintuitive to label non-renewals with "1" and renewals as "0", but the reason behind it is that the business goal 1.3 is to detect non-renewals rather than renewals.

This allows focusing members within a specified range of churn probability and acting upon this information to convert the member to renew.

2.2 Data

To understand the data at hand, it is essential to get some context about the BNI organization first, which will also allow understanding the data preparation process undertaken in this thesis. Below are two of BNI's member policies[1]:

- "Only one person from each BNI classification can join a chapter of BNI. Each Member can only hold one BNI classification in a BNI Chapter."
- "A Member is allowed three absences within a continuous six-month period. If a member cannot attend, they may send a substitute; this will not count as an absence."

The first policy prevents competition for referrals within chapters. The second establishes the maximum number of permissible meeting absences, thus pointing out that meeting attendance is crucial for BNI members. Aside from the policies, another important piece of information about the data is that several members transferred chapters during their membership, which produced some duplicates in the dataset. Lastly, a member can leave BNI at any point of their membership and doesn't have to wait the full subscription period of 12 months. If such a member were to re-join BNI, the unused months could be re-added to their membership, but it is subject to the regional teams' decision.

PALMS reports

Contains measures of member attendance, referrals, performance as well as corresponding month and chapter. Those reports are filled by BNI members on a weekly basis but were exported from the system in a monthly format. The datasets timeframe spans from 2015-01 to 2021-10 inclusively which accounts for a total of 82 shared reports containing PALMS data. Dataset shape: 38713 rows × 16 columns.

	user_ID	chapter_ID	P	A	L	M	S	RGI	RGO	RRI	RRO	V	1-2-1	TYFCB	CEU	palms_date
37630	1551	8	4	1	0	0	0	0	1	2	0	0	2	14181	16	2021-09-01
30258	1759	14	3	0	0	0	0	1	0	1	1	1	6	1074	7	2020-07-01
35373	1325	8	2	0	0	2	0	1	0	2	0	0	4	89	1	2021-05-01
26450	1635	32	2	1	0	0	1	1	3	2	1	0	7	717	1	2019-11-01
17732	1663	25	3	0	0	0	0	1	0	1	0	0	1	0	3	2018-06-01
12731	1198	18	5	0	0	0	0	2	0	1	0	1	7	1557	5	2017-08-01
37811	1114	15	3	1	0	0	0	3	0	1	4	0	0	2661	3	2021-09-01
24733	2638	18	2	0	0	0	2	4	2	5	1	0	5	25	2	2019-08-01
28914	509	26	5	0	0	0	0	0	3	0	0	1	11	0	12	2020-04-01
19637	2238	24	3	1	0	0	0	1	2	0	1	0	6	2429	5	2018-10-01

Figure 2.1: Sample PALMS report data

[1] https://www2.bni.com/rs/166-SUM-744/images/BNI_Policies.pdf

Table 2.1: PALMS variable description

Variable	Full feature name	Data type	Description
user_ID	user identification	integer	Unique number identifying the member
chapter_ID	chapter identification	integer	Unique number identifying the members' chapter
P	Present	integer	Attendance: meeting presences
A	Absent	integer	Attendance: meeting absences
L	Late	integer	Attendance: meeting lates
M	Medical	integer	Attendance: meeting medical leaves
S	Substitute	integer	Attendance: meetings where a member sent in a substitute in his place
RGI	Referral Given Inside	integer	Referral: given from self for self
RGO	Referral Given Outside	integer	Referral: given from other member for self
RRI	Referral Received Inside	integer	Referral: received from other member for the member
RRO	Referral Received Outside	integer	Referral: received from member for a different business professional
V	Visitors	integer	Number of visitors brought to BNI
1-2-1	one-to-one	integer	Member 1-2-1 meetings
TYFCB	Thank You For Closed Business	integer	Amount of other members' closed business thanks to the member
CEU	Chapter Education Units	integer	One hour worth of learning registered in BNI system
palms_date	PALMS date	date	Appropriate date of PALMS report

Source: own elaboration

Region database export

Displays general information about each of the members such as name, chapter, industry, company name, address, phone number, join date, renewal date, etc. Dataset shape: 3567 rows × 6 columns.

	user_ID	chapter_ID	industry	sponsor_ID	join_date	renewal_date
1027	1005	26	Appearance, Cosmetics, Skin Care	NaN	2010-12-01	2018-06-01
3155	267	14	Trades, Electrician	3012.0	2018-02-01	2021-02-01
1875	1119	6	Health and Wellness, Spa	1808.0	2018-02-01	2019-02-01
1873	1029	6	Office, Office Supplies	1314.0	2018-01-01	2019-01-01
2919	781	2	z(Archived Duplicate) Associations, Non-Profit...	969.0	2016-04-01	2018-06-01
231	204	41	Embroidery, Embroidery	861.0	2007-03-01	2008-03-01
1746	656	6	Legal, Lawyer Real Estate	3049.0	2011-07-01	2011-07-01
2029	884	17	Real Estate, Property Management	2189.0	2013-07-01	2014-07-01
52	524	23	z(Archived Duplicate) Alternative Medicine, Cl...	NaN	2010-12-14	2012-11-01
3264	2755	42	Insurance, Life,Health and Disability Insurance	NaN	2018-04-01	2019-04-01

Figure 2.2: Sample Database export data

Table 2.2: Database export variable description

Variable	Data type	Description
user_ID	integer	Unique number identifying the member
chapter_ID	integer	Unique number identifying the members' chapter
industry	string	Members' business classification
sponsor_ID	integer	Unique number identyfing a different member who sposored the member
join_date	date	BNI join date
renewal_date	date	Most recent membership renewal date

Source: own elaboration

Dropped members report

Contains information about member "drops": removals from the system when a subscription ends. There can be different drop reasons such as chapter transfer, membership termination, or expiration. Dataset shape: 2223 rows × 4 columns.

	user_ID	chapter_ID	reason	drop_date
691	1454	26	Other Reason (see notes)	2018-10-05
440	2814	11	Company related (e.g. Changed Jobs, Left Compa...	2019-10-08
1614	1353	17	NaN	2014-04-01
838	1799	21	Not Enough Referrals	2018-03-16
1430	1546	11	No Reason Entered	2015-05-27
1491	2121	23	NaN	2014-12-08
2182	909	23	NaN	2008-10-24
546	2629	15	Scheduling Conflicts	2019-06-26
670	1073	6	Company related (e.g. Changed Jobs, Left Compa...	2018-11-14
1487	2666	17	NaN	2015-01-01

Figure 2.3: Sample Dropped members data

Table 2.3: Dropped members report variable description

Variable	Data type	Description
`user_ID`	integer	Unique number identifying the member
`chapter_ID`	integer	Unique number identifying the members' chapter
`reason`	string	Drop reason
`drop_date`	date	Date when the member was dropped from BNI

Source: own elaboration

2.3 Data limitations

Sample size

BNI ABS consists of roughly 500 members which implies that, upon aggregation, the PALMS data would produce around 3000 usable records at best. This number of records could be even smaller as there were instances of member drops before the 9-month subscription mark, and such members' data wasn't considered in the final dataset.

Missing values

The "database export" and "dropped member" reports contained some missing values, which weren't relevant for machine learning modelling as the algorithm input was based on PALMS data.

Human error

Some records could be duplicated or incorrect if a member was re-entered or dropped due to human error. There can be an instance where a member forgot that he needed to renew their membership and was automatically dropped from the system and then re-entered.

Pandemic data

When the COVID-19 pandemic started in 2020, BNI ABS introduced certain measures to maintain members' safety and well-being. The meetings have gone online and some modifications were introduced to both payment plans and attendance rules. The pandemic may be a source of anomalies in the data and may impact machine learning model performance.

Chapter transfers

Chapter transfers require dropping a member from the system and re-entering into a new chapter, which could result in a false notion of a member ceasing his subscription. Those transfers can be traced within the data by paying attention to the members' `chapter_ID` throughout the PALMS data.

Lack of information

Available data fails to represent relationships between members, which could be critical for churn prediction. The PALMS reports contain data for each month but lack member-specific information such as:

- With whom were `1-2-1` meetings conducted with,
- Who gave the member a referral or received a referral from him,
- How much a member earned from BNI referrals (`TYFCB` only reflects how much others have thanked a member for monetized business they closed).

2.4 Preparation steps

To transform the BNI data into machine learning usable state an extensive process was carried out, which can be broken down to the following phases:

1. Initial preparation – this step involved anonymizing the data, removing any data that did not reflect BNI member subscriptions, and adding a column with the appropriate PALMS date to each report. The jupyter notebook containing python code for this phase was not shared in the GitHub repository for privacy reasons.
2. Concatenating PALMS – creating one PALMS dataset instead of 82 separates by concatenating the monthly data,
3. Creating a master dataset – joining PALMS, dropped members and database export datasets based on `user_ID`,

4. Calculating relative renewal date – as the renewal date column contained each members' latest renewal date, an appropriate number of years had to be subtracted from it, based on how long a member has been in BNI. This task turned out to be challenging for chapter transfers as their renewal dates were dependent on when they re-joined and if they had additional months added to their membership upon transfer,

5. Calculating membership length – this was an additional step, which also turned out to be challenging for chapter transfers requiring a drop from the system and re-enter into the new chapter,

6. Aggregating data – this step required filtering out the last three months of each membership in respect to the relative renewal date calculated in step 5. Once completed, the measures were summed, resulting in a much smaller dataset of 2423 rows by 18 columns (which can be denoted as $m \times n$ shaped dataset where $m = 2423$ and $n = 18$),

7. Feature engineering – some of the data limitations were mitigated by enhancing the dataset with additional features of higher company operational level and a notion of relationship within the chapter:

   - `chapter_size` – the number of members in the members' chapter at the 9-month mark of their subscription,
   - `chapter_retention_rate` – ratio of members who stayed within the chapter during a members' 9-month period. This variable was calculated individually for each member in respect to their `relative_renewal_date`:

     $$\texttt{chapter\_retention\_rate} = \frac{s - d}{s}$$

     where:

     - s = chapter size at the beginning of membership,
     - d = no. of member drops in the chapter during the 9-month period of membership.
   - `chapter_growth_rate` – similar to `chapter_retention_rate` with the only difference of adding the number of members who joined the chapter:

     $$\texttt{chapter\_growth\_rate} = \frac{s - d + j}{s}$$

     where:

     - s = chapter size at the beginning of membership,
     - d = no. of member drops in the chapter during the 9-month period of membership,
     - j = no. of member joins in the chapter during the 9-month period of membership.
   - `seat_popularity_rate` – one of the BNI policies (2.2) states that there can be only one member with a given classification per chapter, and since BNI has a finite set of such designations (`industry` variable in PALMS data), a seat popularity rate allows to quantify this classification. This rate was calculated by summing the number of members in a given industry and dividing it by the total number of chapters within the region at a given time.

8. Labeling records – a new column `wont_renew` was added to the aggregated data for class labels. The labelling process included several steps:
   (a) Removing records with `relative_renewal_date` greater than 2021-09-01 (couldn't be labeled given the fact that such members did not pass the 12-month mark of their membership),
   (b) Label all as $y =$"0"
   (c) Creating variable `final_palms_date` with members' last PALMS date listed in the PALMS reports
   (d) For each record if `relative_renewal_date`+1 month $\geq$ `final_palms_date` $\rightarrow$ label $y =$"1".
9. Post-aggregation cleaning - since the dataset contained several indicators of member attendance, namely the `P`, `A`, `L`, `M`, and `S` columns, those could be summed up to determine the `total_meetings` count. As BNI meetings take place once a week and the aggregated data consists of 9-months, the `total_meetings` number should be within certain bounds. Upon further examination, there were several records with either too many or too few meetings, which resulted in their removal from the dataset. Another post-aggregation check was to inspect if each membership record contained a maximum of 9 rows of pre-aggregated PALMS data: one row representing one month. Some observations with more than nine rows of data weren't removed, as chapter transfers usually contained ten rows of data. The resulting dataset was shaped 2209×29 $(m \times n)$.
10. Removing unnecessary features - some features were non-numeric and thus; unnecessary for machine learning modelling. Upon removing them, the dataset column number was reduced from 29 to 23.

2.5 Summary

This chapter was key for understanding the available BNI data as well as the approach for preparing it for machine learning modelling. It also described all the steps taken and the extensive effort needed to anonymize, clean, and prepare the data. The result of this phase was an entirely new dataset shaped 2209×23 $(m \times n)$, created using aggregated PALMS data.

Chapter 3

Exploratory data analysis

Before starting machine learning modelling it's considered a good practice to explore and analyze the data beforehand using summary statistics and graphical representations. The data exploration phase is a crucial step for a machine learning project as it allows to investigate data to discover patterns, detect outliers or anomalies and verify assumptions. This phase also helps to assess the quality of data. There is a common saying within the field of data science: "garbage in, garbage out", meaning that data quality has an immense impact on the performance of machine learning algorithms. If the data quality is poor, then; no matter how sophisticated a model is, the outcome will most likely yield poor results.

Dataset information

The dataset at hand is made with aggregated PALMS data and has 2191 rows and 23 rows with no missing values. Column names and data types are depicted in Figure 3.1:

```
Data columns (total 23 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   user_ID                 2191 non-null   int64
 1   chapter_ID              2191 non-null   int64
 2   relative_renewal_date   2191 non-null   datetime64[ns]
 3   P                       2191 non-null   int64
 4   A                       2191 non-null   int64
 5   L                       2191 non-null   int64
 6   M                       2191 non-null   int64
 7   S                       2191 non-null   int64
 8   RGI                     2191 non-null   int64
 9   RGO                     2191 non-null   int64
 10  RRI                     2191 non-null   int64
 11  RRO                     2191 non-null   int64
 12  V                       2191 non-null   int64
 13  1-2-1                   2191 non-null   int64
 14  TYFCB                   2191 non-null   int64
 15  CEU                     2191 non-null   int64
 16  year_of_membership      2191 non-null   int64
 17  chapter_size            2191 non-null   int64
 18  chapter_retention_rate  2191 non-null   float64
 19  chapter_growth_rate     2191 non-null   float64
 20  seat_popularity_rate    2191 non-null   float64
 21  total_meetings          2191 non-null   int64
 22  wont_renew              2191 non-null   bool
dtypes: bool(1), datetime64[ns](1), float64(3), int64(18)
```

Figure 3.1: Dataset information

Label balance

Upon further examination of the label balance, there were 1519 “0” and 672 “1”, indicating a data imbalance of 69.3% and 30.7% in favor of “0”. The data imbalance implies that some data balancing might help improve the machine learning models later.

Chapter counts

Table 3.1 contains the number of records per each `chapter_ID` in the data:

Table 3.1: Record counts per each chapter

chapter_ID	number_of_records
32	185
9	172
6	159
23	158
26	157
19	151
18	124
12	123
11	121
31	114
10	109
17	103
15	92
25	88
24	83
21	64
8	47
14	42
4	41
2	24
29	21
28	13

Source: own elaboration

which shows that certain chapters have more members who reach the 9-month of their membership over the years; for example, chapter number 32 has the highest number of 185 records whereas chapter number 28 has the least: 13.

Feature correlation

Pearson correlation coefficient is a great measure for finding linearly correlated features within a dataset. Such correlation is unwanted as it would account for duplicated information for a machine learning model. The correlation matrix in Figure 3.2 shows positively correlated features as red, negatively - blue, and uncorrelated with grey. Upon closer examination, the matrix reveals that no two distinct features have a correlation higher than 0.8 or lower than -0.8, which would be an acceptable ground to drop one of the features. The two most negatively correlated features are `P` and `A` with the score of -0.63, whereas `RRO` and `RRI` are the most positively correlated features with the value of 0.44. Overall the correlation across the matrix is low, except for features within the same group as seen in Figure 3.2.

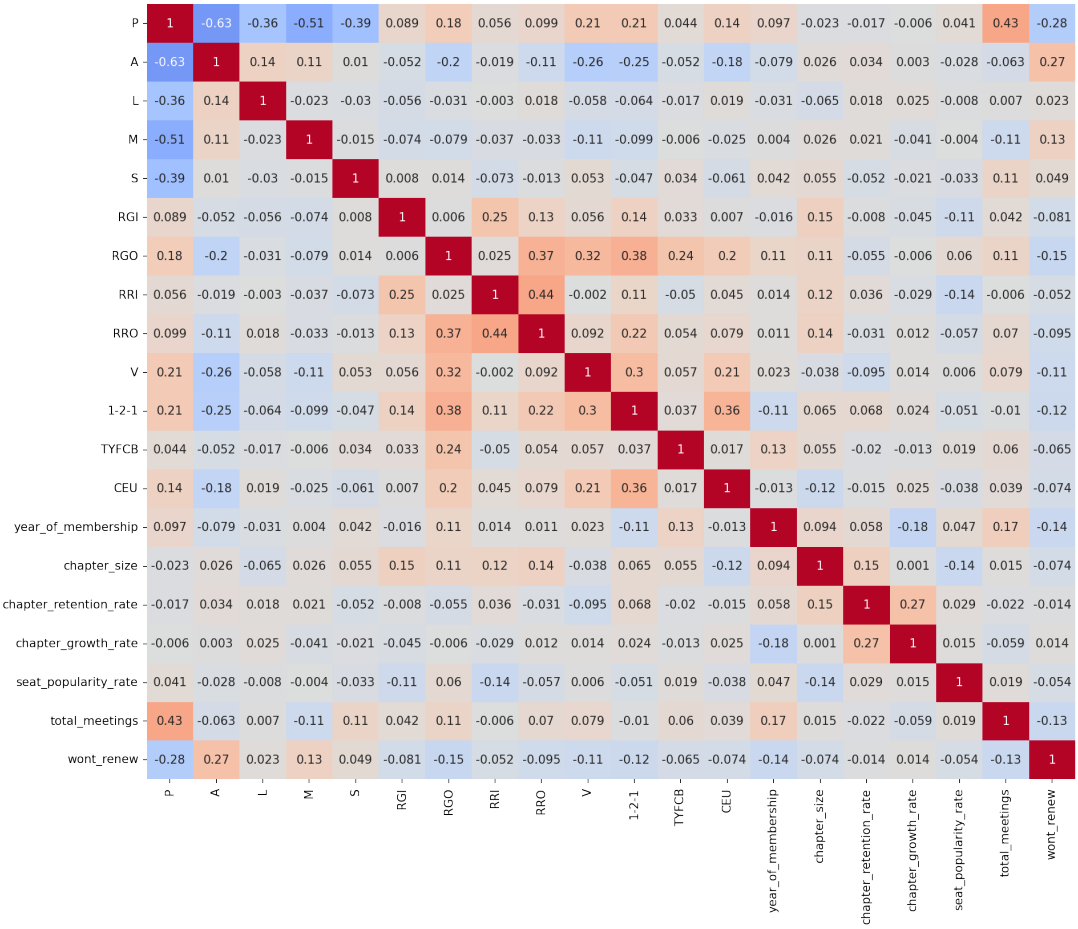

Figure 3.2: Dataset correlation matrix

3.1 Feature groups

To maintain data readability the data will be explored in categories:

- Attendance measures: `P`, `A`, `L`, `M`, `S`, and `total_meetings`;
- Referral measures: `RGI`, `RGO`, `RRI`, and `RRO`;
- Other measures: `V`, `1-2-1`, `TYFCB`, and `CEU`;
- Additional created measures:
  - `year_of_membership`,
  - `chapter_size`,
  - `chapter_retention_rate`,
  - `chapter_growth_rate`,
  - `seat_popularity_rate`.

Each category will contain a table summary statistics of mean, standard deviation, minimum, 25^{th} percentile, 50^{th} percentile, 75^{th} percentile and maximum per feature, followed by graphical representations of:

- boxplots comparing the distribution of labels “0” and “1” per each feature,

- density histogram plots (with feature mean marked as a red dashed line) comparing the distribution of labels “0” and “1” per each feature.

3.1.1 Attendance measures

Columns `P`, `A`, `L`, `M`, `S`, and `total_meetings` demonstrate member attendance activity throughout the 9-month period.

Summary statistics

Table 3.2: Attendance measures summary statistics

Feature name	mean	std	min	25%	50%	75%	max
`P`	33.1794	3.5659	14	31	34	36	40
`A`	1.5436	1.8334	0	0	1	2	22
`L`	0.3729	1.1162	0	0	0	0	15
`M`	0.4386	1.5094	0	0	0	0	21
`S`	1.607	1.5857	0	0	1	3	11
`total_meetings`	37.1415	1.4303	32	36	37	38	40

Source: own elaboration

Graphical representations

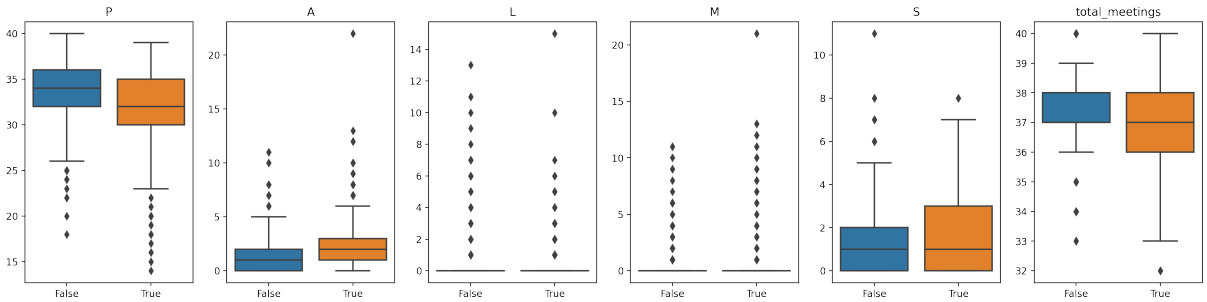

Figure 3.3: Attendance boxplots

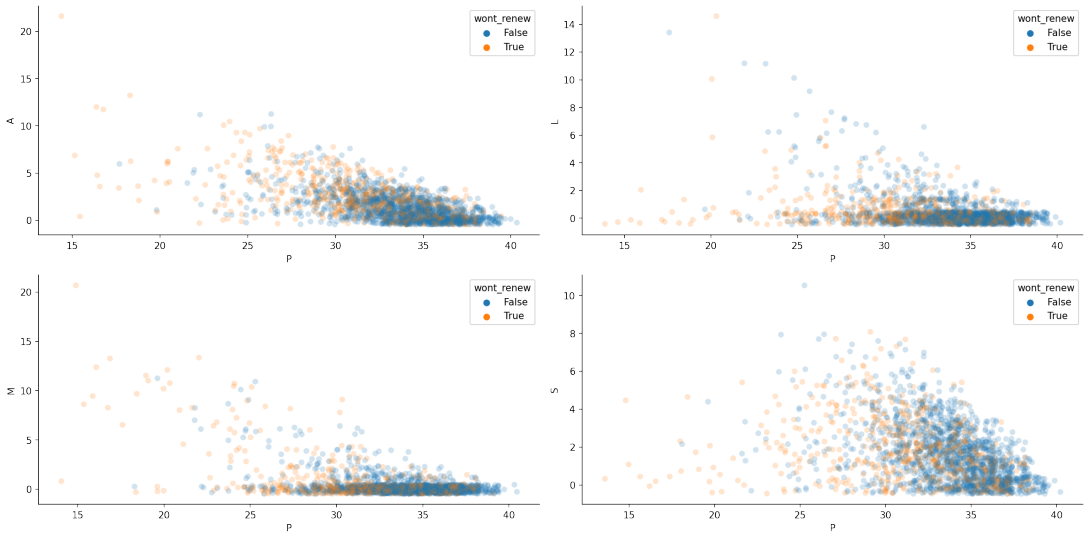

Figure 3.4: Attendance scatter plots

The attendance scatter plots in Figure 3.4 were created by adding a bit of horizontal and vertical jitter to the observations. It is clear that the classes are not linearly separable and for some features, the majority of observations are focused around 0.

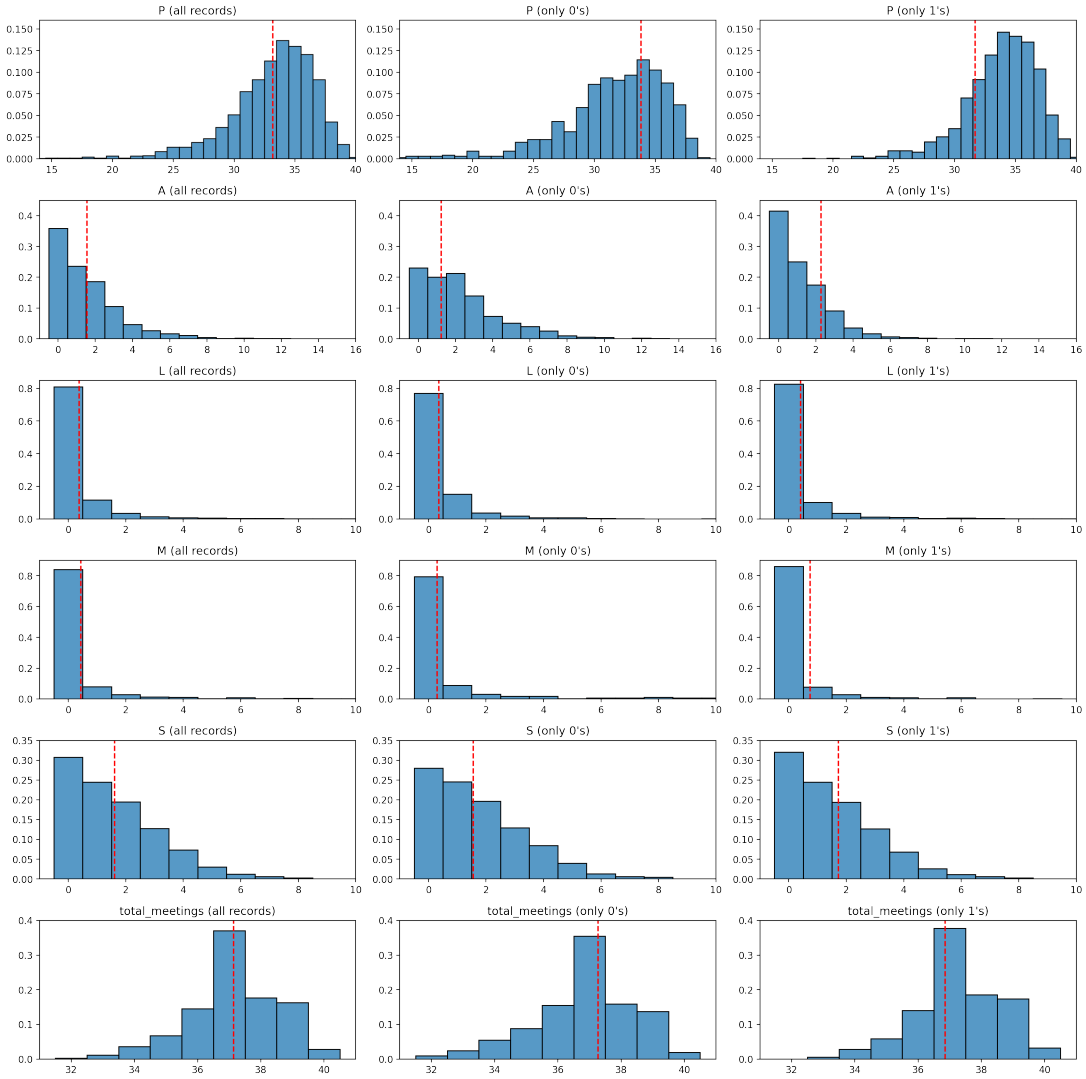

Figure 3.5: Attendance density histograms

Table 3.3: Attendance measures means

Feature name	all records	only 0's	only 1's
`P`	33.18	33.83	31.7
`A`	1.54	1.22	2.28
`L`	0.37	0.36	0.41
`M`	0.44	0.3	0.74
`S`	1.61	1.55	1.72
`total_meetings`	37.14	37.27	36.86

Source: own elaboration

The `P`, `A`, `S`, and `total_meetings` values in Figure 3.3 indicate that members who decide to not renew their membership ("1") overall attend fewer meetings than members who renew. The distributions of `L`, `M` don't differ much for renewals and non-renewals but it is worth noting that the mean of `M` is a bit higher for non-renewals which is most likely due to several outliers with higher amount of medical leaves within this class. Visually, `total_meetings` is the only feature for which histogram resembles that of normal distribution, implying that feature scaling might improve model performance in the later stage of machine learning modelling.

3.1.2 Referral measures

Columns `RGI`, `RGO`, `RRI` and `RRO` are measures of member referral activity.

Summary statistics

Table 3.4: Referral measures summary statistics

Feature name	mean	std	min	25%	50%	75%	max
`RGI`	10.2273	7.414	0	5	9	14	62
`RGO`	25.2487	19.8073	0	13	21	32	224
`RRI`	9.9772	11.343	0	3	7	12	126
`RRO`	24.8389	21.3778	0	11	20	32	296

Source: own elaboration

Graphical representations

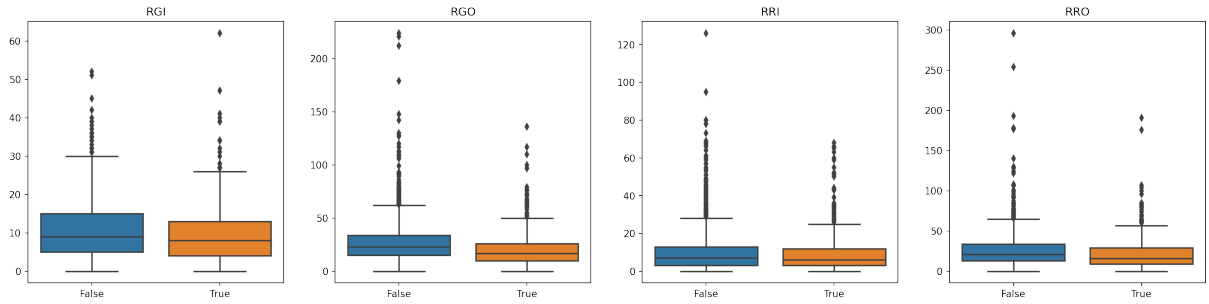

Figure 3.6: Referral boxplots

The boxplots in 3.6 show similar distribution for both "0" and "1" but zooming in on the y-axis as depicted in Figure 3.7 emphasizes their differences.

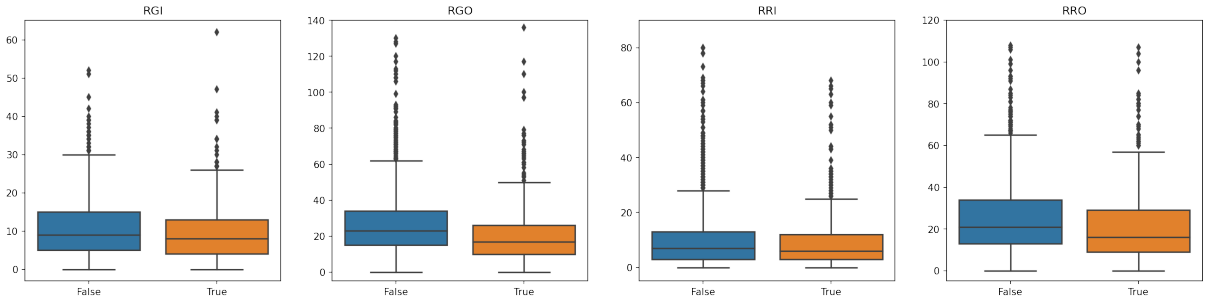

Figure 3.7: Referral boxplots (zoomed in)

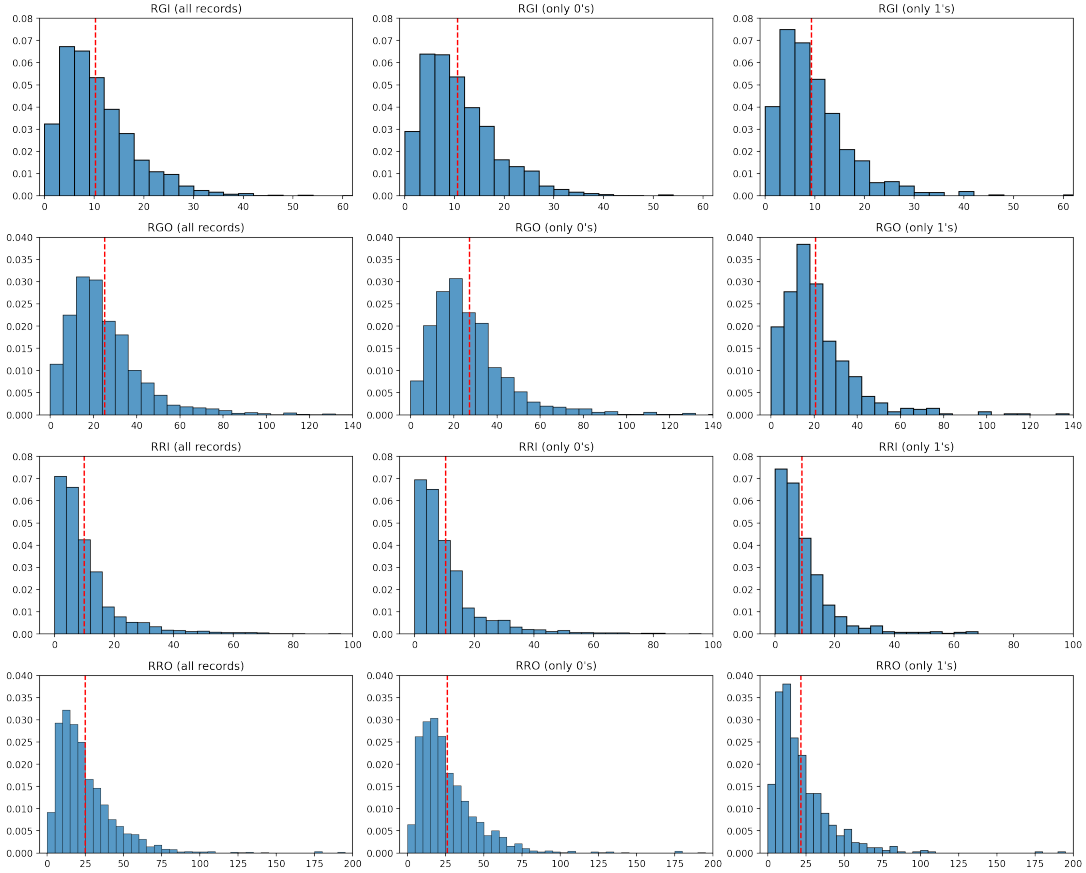

Figure 3.8: Referral density histograms

Table 3.5: Referral measures means

Feature name	all records	only 0's	only 1's
`RGI`	10.23	10.63	9.32
`RGO`	25.25	27.28	20.65
`RRI`	9.98	10.37	9.09
`RRO`	24.84	26.2	21.77

Source: own elaboration

None of the features has normal distribution as Figure 3.8 clearly shows positive skewness across all plots, which might indicate log-normal distribution instead. Similarly to attendance features, as seen in Table 3.3, all members who don't renew are less active, which can be observed by fewer given and received referrals as indicated in means in Table 3.5 and quartiles in boxplots 3.7.

3.1.3 Other measures

Features `V`, `1-2-1`, `TYFCB` and `CEU` cannot be grouped in a similar manner as attendance or referral measures as they describe different information:

- `V` is the number of visitors a member brought into the chapter,
- `1-2-1` the number of 1-2-1 meetings the member had with other BNI members,
- `TYFCB` - the amount of closed business the member has been thanked for, and
- `CEU` - the number of education units (1 hour = 1 CEU).

Summary statistics

Table 3.6: Other PALMS report measures summary statistics

Feature name	mean	std	min	25%	50%	75%	max
`V`	3.9429	4.0213	0	1	3	5	36
`1-2-1`	42.6823	22.3701	4	29	38	52	276
`TYFCB`	28641.1159	78602.5655	0	5345.5	12460	27907	1467394
`CEU`	41.3962	41.1458	0	19.5	36	50	580

Source: own elaboration

Graphical representations

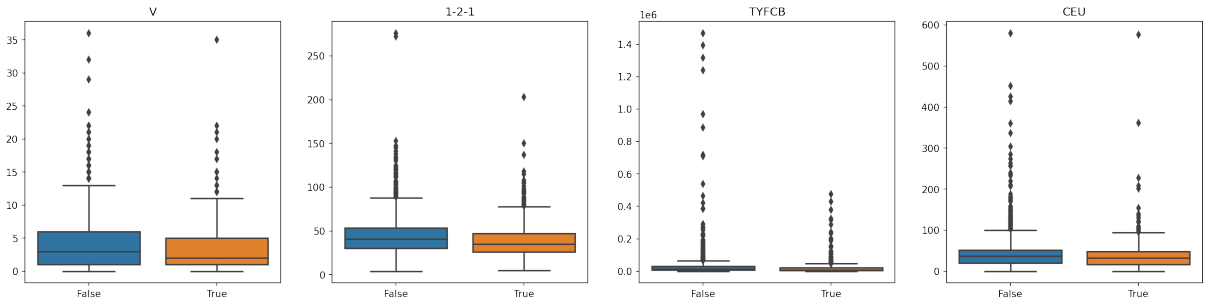

Figure 3.9: Other PALMS report measures boxplots

Zooming in on the y-axis of boxplots in Figure 3.8 as seen in 3.9 allows to see the differences between "1" and "0" more clearly.

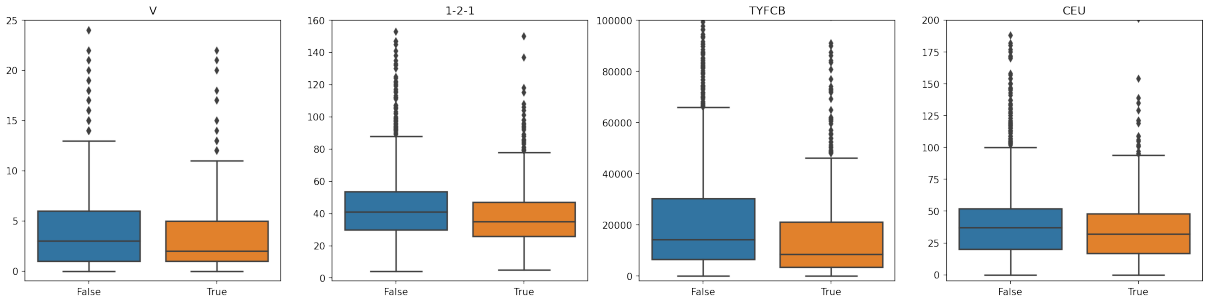

Figure 3.10: Other PALMS report measures boxplots (zoomed in)

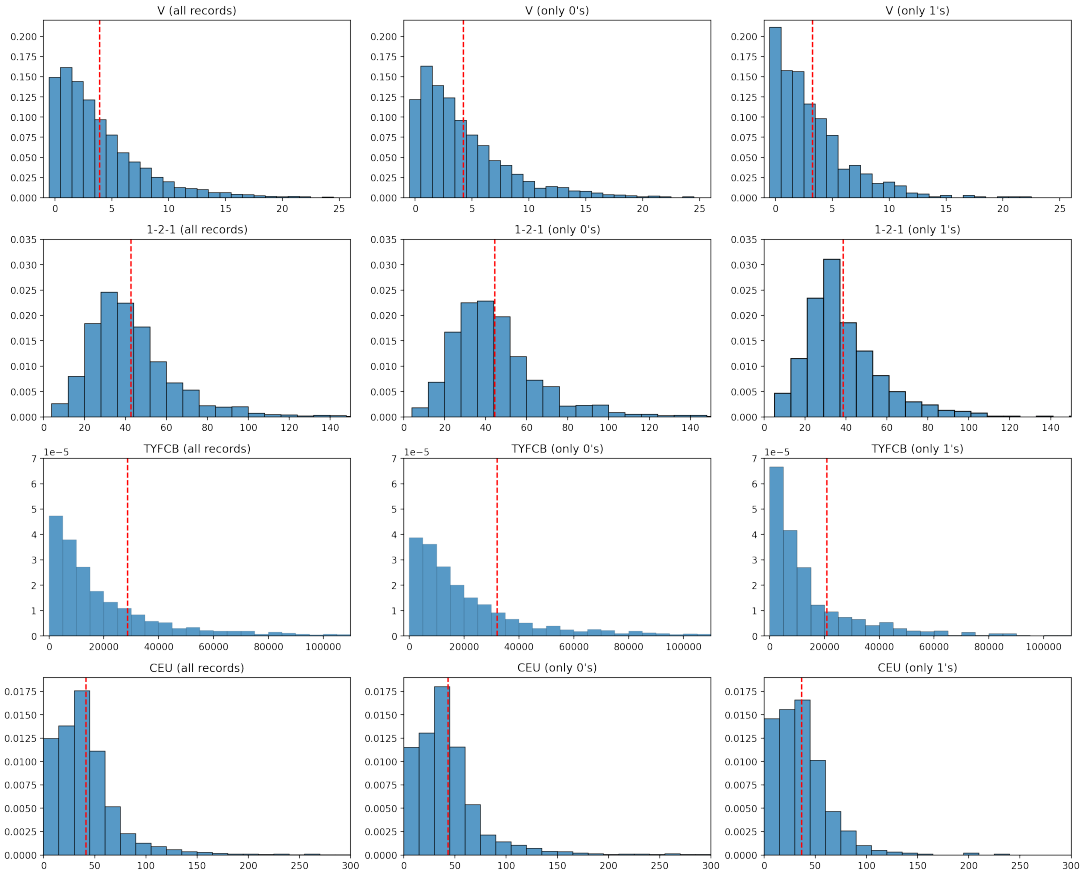

Figure 3.11: Other PALMS report measures density histograms

Table 3.7: Other measures means

Feature name	all records	only 0's	only 1's
`V`	3.94	4.24	3.27
`1-2-1`	42.68	44.46	38.66
`TYFCB`	28641.12	32045.09	20946.72
`CEU`	41.4	43.41	36.85

Source: own elaboration

All of the features have displayed positive skewness. There was a significant amount of outliers within the `TYFCB` feature, which made it hard to compare the quartiles across classes in Figure 3.9. Distributions of `V`, `1-2-1` and `CEU` look quite like log-normal whereas the distribution of `TYFCB` seems exponential. The range of values across `TYFCB` was also significantly larger than that of any other feature examined so far.

3.1.4 Additional created measures

Chapter 2.4 contains detailed description of the additional features created to display information of higher operational level than member and a notion of relationship within chapter.

Summary statistics

Table 3.8: Created measures summary statistics

Feature name	mean	std	min	25%	50%	75%	max
`year_of_membership`	1.7471	2.4032	0	0	1	3	22
`chapter_size`	29.2428	6.5605	10	24	30	34	44
`chapter_retention_rate`	0.6542	0.1419	0.087	0.5714	0.6667	0.7419	1
`chapter_growth_rate`	1.1812	0.3457	0.5758	1	1.1111	1.2727	2.875
`seat_popularity_rate`	0.2618	0.2072	0	0.0909	0.2	0.4	1

Source: own elaboration

Graphical representations

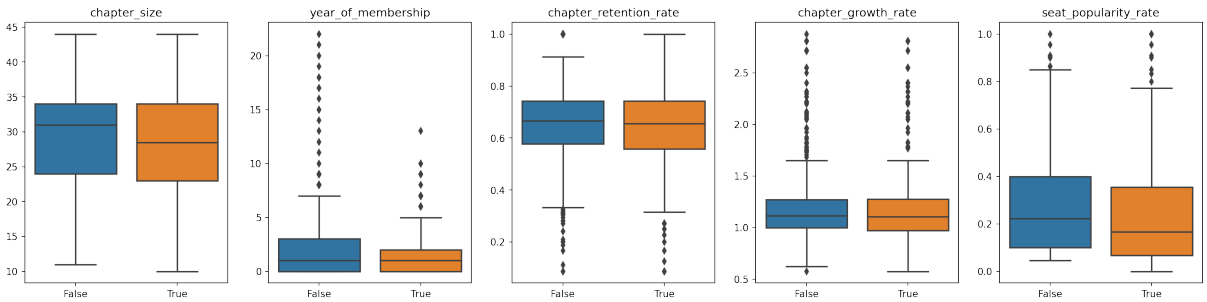

Figure 3.12: Created measures boxplots

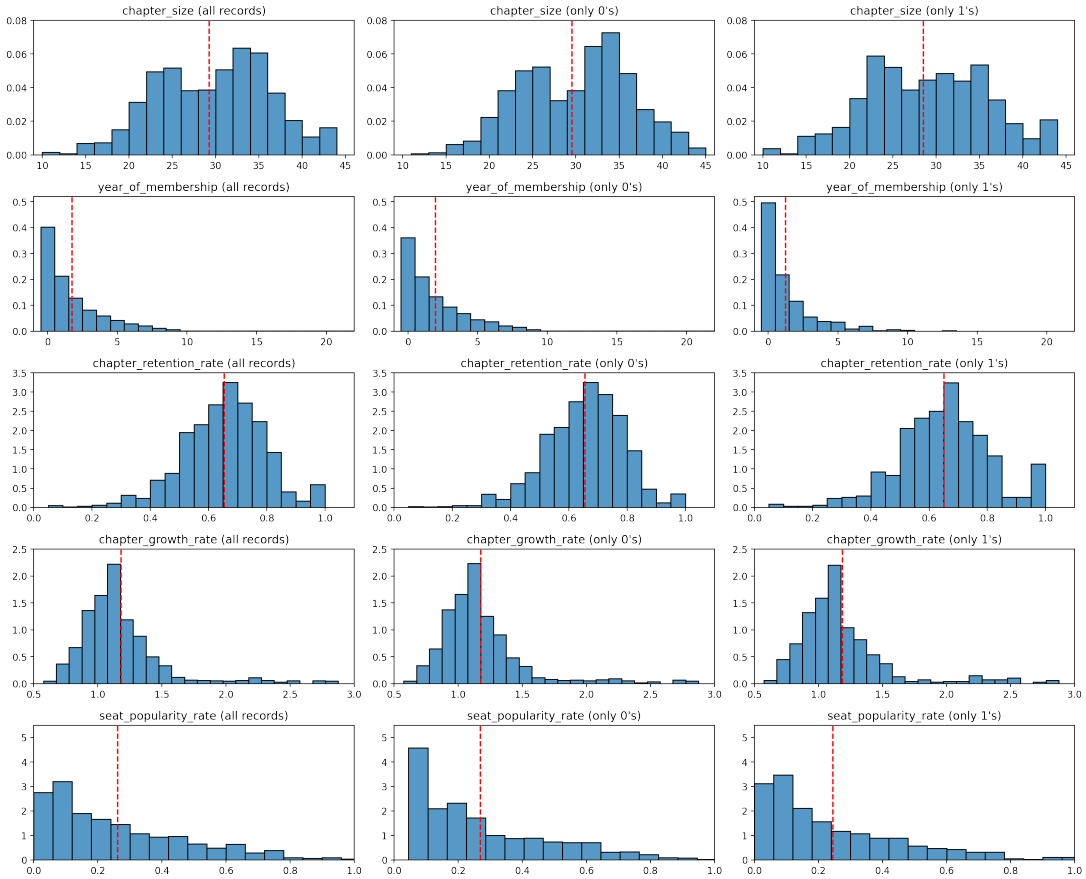

Figure 3.13: Created measures density histograms

Table 3.9: Created measures means

Feature name	all records	only 0's	only 1's
`chapter_size`	29.24	29.56	28.52
`year_of_membership`	1.75	1.97	1.25
`chapter_retention_rate`	0.65	0.66	0.65
`chapter_growth_rate`	1.18	1.18	1.19
`seat_popularity_rate`	0.26	0.27	0.24

Source: own elaboration

The distributions look similar for both classes across all features, as seen in boxplots, means, and histograms. It is worth noting that `year_of_membership` mean is significantly lower for members labeled with "1" than for "0", which means that the tendency for members who don't renew their subscription is that they resign in the first year of their membership. Notably, in histogram `year_of_membership (only 1's)` in Figure 3.13 nearly half of the members who quit BNI are first years. This number might be even higher as the histogram only shows members who have been a part of BNI for at least 9 months. There were more records for members who quit before the 9-month mark of their subscription.

3.2 Summary

An observable tendency within the dataset was that members who do not renew show less activity than members who do, which was observable both in summary statistics and graphical representation when comparing "1" and "0" classes. The differences between class distributions were usually minor, implying that the classes are not linearly separable. Very few features indicated normal distributions, as a majority of the features showed positive skewness, which means that scaling the data could result in performance improvement during later stages of machine learning modelling. There were multiple observable outliers across features, especially in the `TYFCB` column, but it was impossible to determine if those outliers were or weren't a result of an error, which is why they weren't removed from the dataset.

Chapter 4

Machine Learning modelling

Once the data preparation and exploration phases were over, it was time for the final stage of the project: creating and refining machine learning models. The machine learning classification models used for this chapter were:

1. Logistic Regression[2],
2. Naive Bayes[7],
3. K-nearest neighbors[17],
4. Decision Tree[4],
5. Random Forest[3],
6. AdaBoost[9],
7. XGBoost[6],
8. CatBoost[19],

which are publicly available in Python packages such as scikit-learn, xgboost and catboost. As the model is supposed to determine probability of a member churn, the testing metric employed for this project is Receiver Operation Characteristic area under curve (abbreviated as "AUC")[8]. This metric is especially useful for domains with unbalanced class distribution and unequal classification error costs, which is the case for BNI data as mentioned in chapter 3.

Although the best performing model allows for accurate predictions, the model resulting from this thesis might be used by the BNI company, steps were taken to avoid model overfitting (not being able to generalize results to new data). To maintain a balance; the models were compared on the following grounds:

- AUC score - model scoring method.
- Feature importance - for determining what the model considers the most important factor if a member won't renew their membership. Such information gives the company knowledge of which features are key for member renewal and could perhaps be influenced to convince a member to renew.
- Probability calibration - if a model has a great AUC score but does not accurately output probability, then it creates the risk of adding unnecessary work for the company by pointing towards incorrectly predicted renewals.

4.1 Modelling steps

A strategy undertaken in this thesis involves exploring the models in multiple steps. Each step contains the solutions adapted in previous steps, meaning step 2 contains also contains solutions from step 1, step 3 from steps 2 and 1, and so on. As a point of reference for model performance, Lalwani et al. initially achieved an AUC score of 0.7928 with a Random Forest algorithm used for churn prediction[12] but it is worth noting that they operated on a different and larger dataset. Nonetheless, the goal for this machine learning project remains to achieve an AUC score of 0.8. The data has been split into an 80/20 ratio for model evaluation. It is also worth noting that the additionally created features during the data preparation phase such as `chapter_size` or `year_of_membership` were initially removed from the dataset to be added later in one of the further modelling steps to determine how much they improve model performance and the CatBoost algorithm had an advantage of utilizing the `chapter_ID` categorical feature in contrary to other algorithms.

4.1.1 Baseline

Initially, the models were fitted to the data without any data or model tampering, except for setting model evaluation metric to AUC for XGBoost and CatBoost algorithms, as well as specifying `chapter_ID` as a categorical column for CatBoost algorithm. The scores seen in Table 4.1 set a baseline for the rest of this project.

Table 4.1: Baseline model scores

Model name	Training time [s]	Accuracy	Precision	Recall	F1	AUC
Logistic Regression	0.1000	0.7289	**0.6905**	0.2148	0.3277	0.6912
Naive Bayes	0.0000	0.6310	0.4372	**0.6963**	**0.5371**	0.6724
K-nearest neighors	0.0100	0.6401	0.3544	0.2074	0.2617	0.5340
Decision Tree	0.0200	0.6401	0.4135	0.4074	0.4104	0.5754
Random Forest	0.2600	**0.7335**	0.6731	0.2593	0.3743	0.6659
AdaBoost	0.1100	0.7267	0.6119	0.3037	0.4059	0.6947
XGBoost	0.1100	0.7039	0.5325	0.3037	0.3868	0.6468
CatBoost	25.6800	0.7198	0.6250	0.2222	0.3279	**0.6947**

Source: own elaboration

The best performing model was initially CatBoost with an AUC score of 0.694688, slightly beating AdaBoost with a score of 0.694652.

4.1.2 Feature scaling

One of the crucial steps in machine learning modelling is feature scaling, as models rarely perform well when numerical input features have diverging ranges. During this phase, all numerical features were standardized:

$$z_j := \frac{z_j - \mu_j}{\sigma_j}$$

for $j = 1, 2, \ldots, n$ where:

- μ_j - j_{th} feature mean
- σ_j - j_{th} feature standard deviation

which resulted in Logistic Regression taking the lead with an AUC score of 0.6953.

4.1.3 Dataset balancing

One of the challenges posed by the dataset is that it is imbalanced: only 537 observations were labeled positively and the remaining 1215 - negatively, thus splitting the dataset in the ratio of 30.7% to 69.3%. Although this imbalance was not severe, the goal of this step was to explore if data balancing techniques would improve model performance. Three different strategies were examined:

1. downsample majority class using Tomek Links[21],
2. upsample minority class using SMOTE[5],
3. combine minority upsampling using SMOTE and majority downsampling using Tomek Links.

Other methods were also tested, as the upsampling and downsampling strategies, but the best results were achieved with Tomek Links and SMOTE. The top results of each approach are found in Table 4.2:

Table 4.2: Model results after data balancing

Model name	Strategy number	Negative class size	Positive class size	AUC
Logistic Regression	1	1005	537	**0.7011**
Naive Bayes	2	1215	1215	0.6901
Naive Bayes	3	1034	850	0.6899

Source: own elaboration

The strategy that yielded the best results was Tomek Links majority downsampling which allowed to break the AUC threshold of 0.7 by achieving 0.7011 with Logistic Regression. It is worth noting that CatBoost, which initially had the best results, suffered a major performance loss due to data balancing, so in the following steps of machine learning modelling, the data used for CatBoost training was not resampled.

4.1.4 Adding custom features

Before the first stage of machine learning modelling, the additional features created in the data preparation phase were removed, which allowed comparing if they make any difference in model performance. Indeed, adding them resulted in the biggest AUC score gain so far, increasing the previously achieved Logistic Regression score of 0.7011 by 0.0169, summing up to an AUC score of 0.718 by CatBoost.

4.1.5 Feature multiplication

A common strategy for increasing model complexity is to create new features out of existing ones with Hadamard Products (also known as element-wise multiplication) denoted as:

$$A \odot B = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix} \odot \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} = \begin{bmatrix} a_1 \cdot b_1 \\ a_2 \cdot b_2 \\ \vdots \\ a_m \cdot b_m \end{bmatrix}$$

and

$$A \div B = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix} \div \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} = \begin{bmatrix} \frac{a_1}{b_1} \\ \frac{a_2}{b_2} \\ \vdots \\ \frac{a_m}{b_m} \end{bmatrix}$$

where A and B represent a single feature or column from the dataset shaped $m \times 1$. Thus, new features were created:

- `P_by_A` = `P` $\div$ (`A` + 1)
- `P_by_M` = `P` $\div$ (`M` + 1)
- `P_by_S` = `P` $\div$ (`S` + 1)
- `P_by_total` = `P` $\div$ `total_meetings`
- `A_by_total` = `A` $\div$ `total_meetings`
- `M_by_total` = `M` $\div$ `total_meetings`
- `P_tim_retention` = `P` $\odot$ `chapter_retention_rate`
- `P_tim_growth` = `P` $\odot$ `chapter_growth_rate`
- `P_tim_popularity` = `P` $\odot$ `seat_popularity_rate`
- `P_tim_V` = `P` $\odot$ `V`
- `P2_tim_V` = `P` $\odot$ `P` $\odot$ `V`
- `P_tim_TYFCB` = `P` $\odot$ `TYFCB`
- `year_tim_retention` = `year_of_membership` $\odot$ `chapter_retention_rate`
- `P2` = `P` $\odot$ `P`

This raised the number of features from 20 to 34 (excluding `chapter_ID`) which, surprisingly, resulted in model score deterioration. The best scoring algorithm in this phase was CatBoost with AUC score of 0.7163. The lower score might be associated with duplicate information contained in the newly created features which could be resolved with feature selection.

4.1.6 Feature selection

In order to avoid model overfitting, a standard procedure is to reduce the dataset dimension while preserving as much information as possible. There are several algorithms for dimensionality reduction, for example, t-SNE or PCA but, the downside of said methods is that the output features cannot be mapped to the original dataset. As one of the goals of this thesis is to find out which features are most important for member non-renewals, a different approach was taken. The majority of machine learning model implementations have built methods for calculating the importance of each feature which allows discarding the least important ones in an iterative way as described in algorithm 1:

Algorithm 1 Recursive feature selection algorithm

Require: Machine Learning implementation has built-in feature importance attribute
  for $i = n, n-1, \ldots, 3$ **do**
    Fit the model to X_{train} $(m_{\text{train}} \times i)$ and y_{train} $(m_{\text{train}} \times 1)$ data
    Get model probability output $h(X_{\text{test}})$ $(m_{\text{test}} \times 1)$
    Calculate AUC score
    Get model feature importance
    $X_{\text{train}} \leftarrow X_{\text{train}}$ subset with $i-1$ features with highest importance
    $X_{\text{test}} \leftarrow X_{\text{test}}$ subset with $i-1$ features with highest importance
  end for

Before running this algorithm, another method was used to discard features with a high Pearson correlation coefficient: `SmartCorrelatedSelection` available in Python `feature-engine` package. The `SmartCorrelatedSelection` method grouped features with a Pearson correlation coefficient of more than 0.8 (or less than -0.8), leaving one with the highest variance from each group. Upon completing this selection process, the dataset used for algorithm 1 was consisted of21 features. Finally, the recursive feature selection algorithm was utilized, which results can be seen in Figure 4.1:

Figure 4.1: Feature selection scores

Naive Bayes and K-nearest neighbors are not included in the figure as they do not have methods for calculating feature importance. Some key observations from this Figure are

that the mean, displayed with a black dashed line on the chart, is steady until ten selected features, at which it consistently starts to decrease. For this reason, the final models were built with ten features according to their feature importance selection. The CatBoost algorithm initially showed the best results for 22 features, but its performance dropped faster compared to other models, which is observable around the mark of 12 selected features. Random Forest and Logistic Regression show high and consistent AUC scores from the beginning of recursive selection until 10 to 8 features, followed by a significant score increase for Logistic Regression upon selecting four features. This score improvement could be specific to the dataset split and selecting such a low number of features would create a risk of underfitting. The Decision Tree shows the lowest results across the chart compared to all other models, which is why this model isn't a good indicator of what features are critical for member non-renewals. For this reason, the Decision Tree algorithm is excluded from table 4.3 which displays model feature importances:

Table 4.3: Top 10 features according to model feature importance

Ranking	LogReg	RF	XGBoost	CatBoost	AdaBoost
1	A	P_tim_TYFCB	P2	P_tim_popularity	P_tim_TYFCB
2	year_of_membership	P_tim_retention	RGO	P_tim_retention	P_tim_popularity
3	M	P_tim_popularity	P_tim_TYFCB	P_by_A	P_tim_retention
4	P_by_S	P_by_A	P_tim_popularity	P_tim_TYFCB	P_by_M
5	P_tim_popularity	1-2-1	RGI	chapter_ID	RRI
6	total_meetings	P_tim_growth	1-2-1	1-2-1	1-2-1
7	RGO	RGO	P_by_A	P2_tim_V	P2
8	chapter_size	RRO	chapter_size	RGO	year_of_membership
9	1-2-1	CEU	P_tim_retention	RGI	P_by_S
10	RGI	chapter_size	year_of_membership	P_tim_growth	P2_tim_V

Source: own elaboration

Two features are listed by all of the models: `P_tim_popularity` and `1-2-1`. The `P_tim_popularity` is even more critical as it is listed in the top 3 for three separate models. Logistic Regression, which yielded the best results during this stage, breaks out in comparison to other models by determining `A`, `year_of_membership` and `M` as its top 3 features. The AUC scores outputted by models with ten selected features as input are in Table 4.4:

Table 4.4: AUC scores after 10 selected features

Model name	AUC
Logistic Regression	**0.7021**
Random Forest	0.691
XGBoost	0.6728
CatBoost	0.6684
AdaBoost	0.6674
Decision Tree	0.5890

Source: own elaboration

4.1.7 Hyperparameter tuning

Hyperparameter tuning requires longer to finish computing as the models have to be fitted to the data multiple times with different hyperparameters. For this reason, this step has was left to be one of the last ones. The method taken for this step was to fit the models to the data across a hyperparameter grid with 3-fold cross-validation. The results can be seen in Table 4.5:

Table 4.5: Tuned models results

Model name	Training time [s]	Accuracy	Precision	Recall	F1	AUC
XGBoost	397.57	0.7198	0.7500	0.1333	0.2264	**0.7073**
Logistic Regression	2.04	0.7084	**1.0000**	0.0519	0.0986	0.7023
Random Forest	58.89	**0.7289**	0.6176	0.3111	0.4138	0.6906
AdaBoost	5.17	0.7062	0.5341	0.3481	0.4215	0.6801
K-nearest neighbors	4.11	0.7016	0.5294	0.2667	0.3547	0.6609
Naive Bayes	0	0.4351	0.3313	**0.8222**	**0.4723**	0.6569
Decision Tree	5.56	0.7175	0.6170	0.2148	0.3187	0.6494
CatBoost	360.07	0.7039	0.5301	0.3259	0.4037	0.6358

Source: own elaboration

4.1.8 Probability calibration

As the main goal of this thesis is to create a model which outputs probability, it is desired for this probability to be as accurate as possible. A well calibrated machine learning classifier ought to classify records in a way that from among the samples for which it outputs a probability of around 0.8, approximately 80% belongs to the positive class. To check how well a model outputs probability reliability curves can be utilized in which the black dotted line represents a perfectly calibrated classifier:

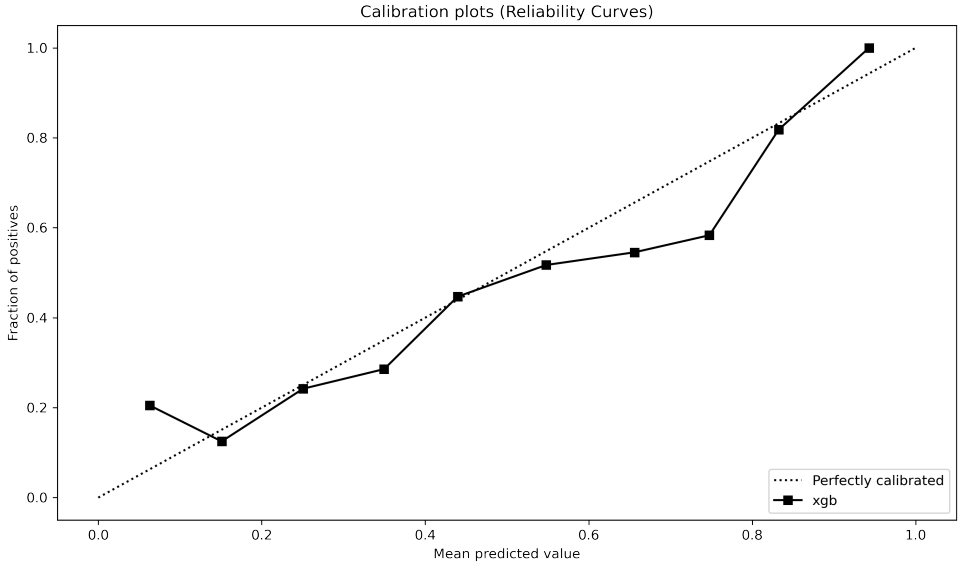

Figure 4.2: XGBoost reliability curve with no calibration

Niculescu-Mizil and Caruana propose two methods for calibrating probability output of

a machine learning classifier in their paper "Predicting Good Probabilities With Supervised Learning"[13]:

- Platt scaling devised by Platt[18],
- Isotonic Regression used by Zadrozny and Elkan[22].

Upon testing both of those methods on all of the machine learning algorithms used in this thesis, XGBoost yielded the best results and its updated calibration curves can be seen in Figures 4.3 and 4.4:

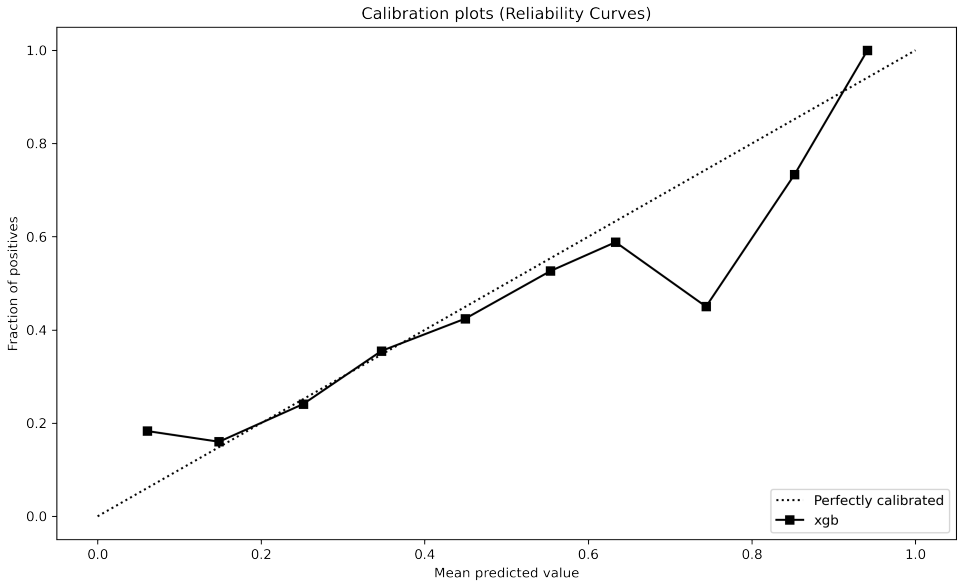

Figure 4.3: XGBoost reliability curve after Platt scaling

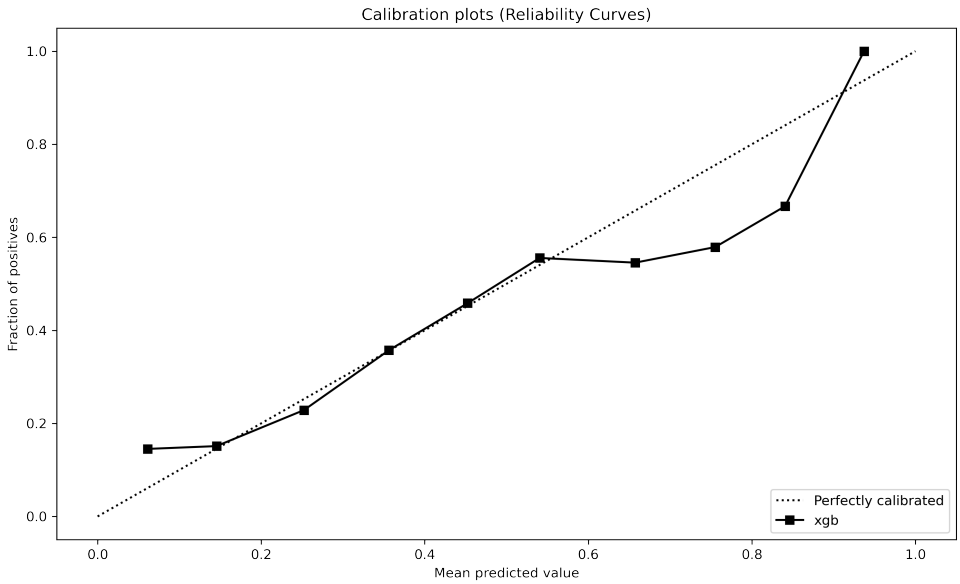

Figure 4.4: XGBoost reliability curve after Isotonic regression

The AUC scores and squared error of calibration for all models are shown in bar plots 4.5 and 4.6 respectively:

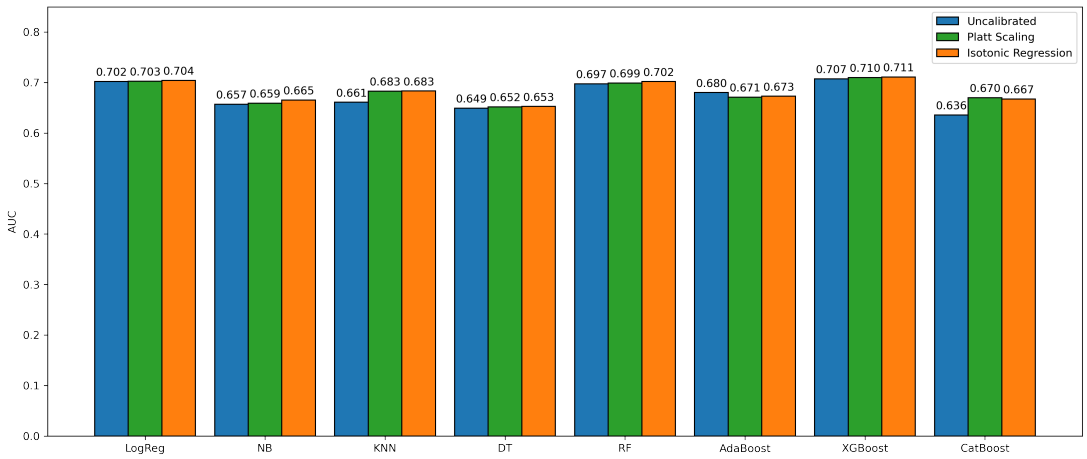

Figure 4.5: AUC scores across calibrations

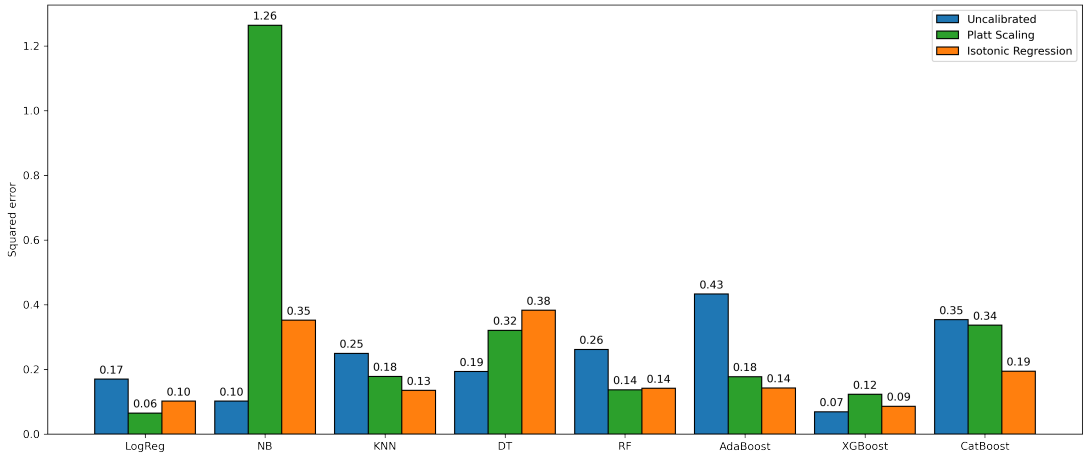

Figure 4.6: Squared calibration errors

All models except for AdaBoost yielded some AUC score improvements after calibration. The top 3 performing models remain to be XGBoost, Logistic Regression, and Random Forest, with their respective AUC scores of 0.711, 0.704, and 0.702. In nearly every case except for CatBoost, Isotonic Regression outperformed Platt Scaling, which is more suited for smaller datasets as it can overfit on datasets that have 1000 or fewer observations[13]. Surprisingly the scaling increased the squared calibration error for three of the models, amongst which is XGBoost. Naive Bayes showed some unusual results as the Platt Scaling squared error seen in figure 4.6 is significantly higher than that of other algorithms.

4.2 Model comparison

The final AUC scores for the top 3 performing models are as follows:

Table 4.6: Final AUC scores of top 3 models

Model name	Calibration method	AUC
XGBoost	Isotonic	**0.7111**
Logistic Regression	Isotonic	0.7042
Random Forest	Isotonic	0.7019

Source: own elaboration

And their history across the 8 phases of machine learning described in this chapter can be seen in Figure 4.7

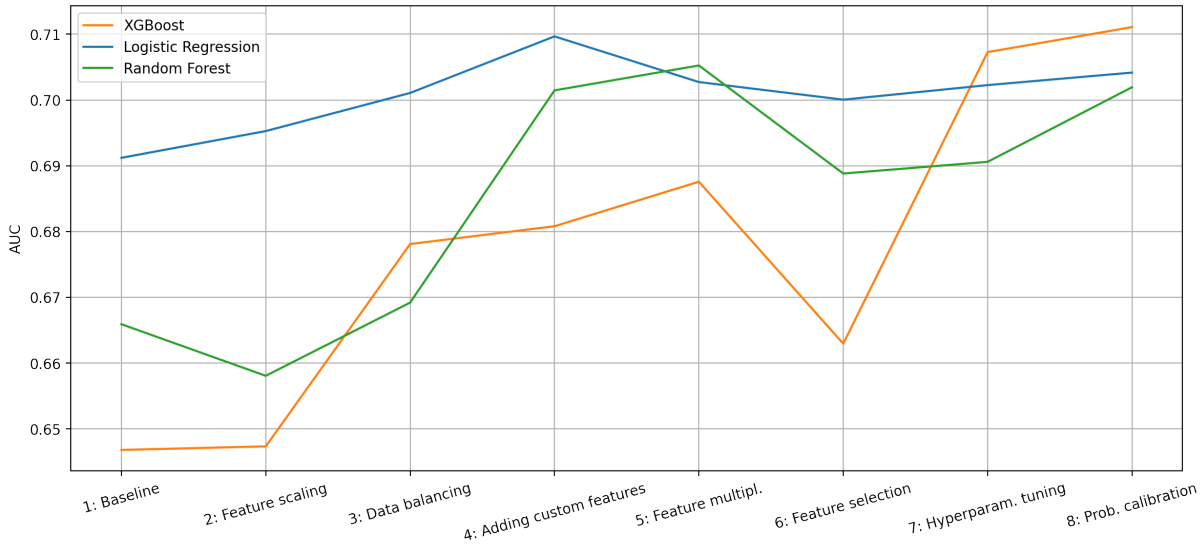

Figure 4.7: Top 3 models AUC history acrros modelling phases

It is clear that Logistic Regression showed good performance since the first phase and was the top-scoring algorithm among the majority of all stages, eventually outperformed by XGBoost after hyperparameter tuning, which increased its AUC score by over 0.04. As expected, the models suffered some performance loss after feature selection, but it is more observable for the two boosting algorithms than Logistic Regression. It is also worth noting that the AUC score of Logistic Regression decreased in phase "5: Feature multip." contrary to the other two models.

In order to ensure that the amount of data at hand is sufficient for machine learning modelling, a diagnostic tool such as learning curves can be examined. If the training curve does not converge to the same value as the validation curve, then the amount of data is insufficient. The learning curves for XGBoost, Logistic Regression can be found in Figures 4.8, 4.9 and 4.10 respectively:

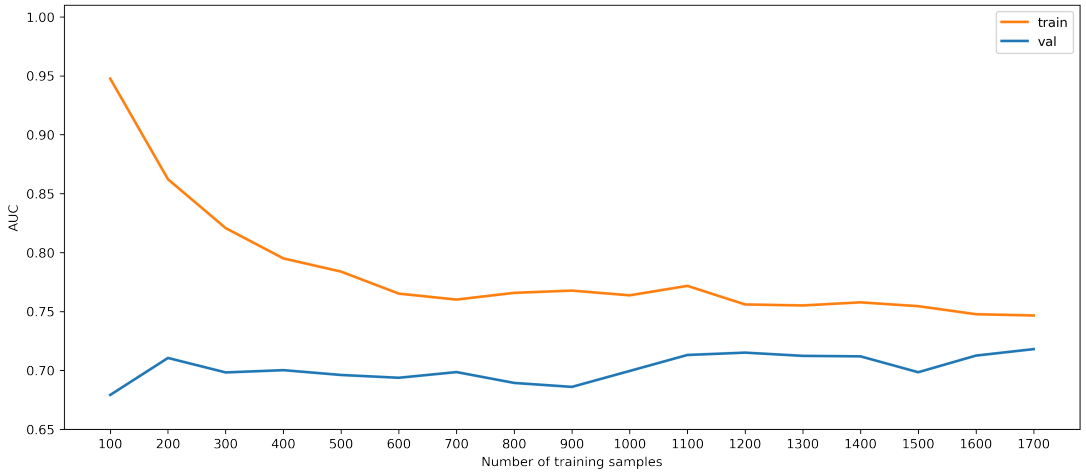

Figure 4.8: XGBoost learning curves

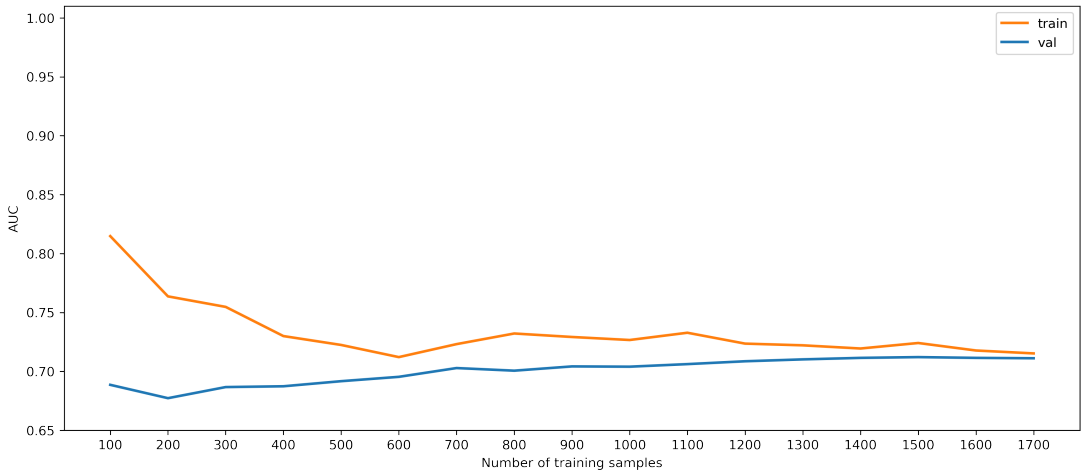

Figure 4.9: Logistic Regression learning curves

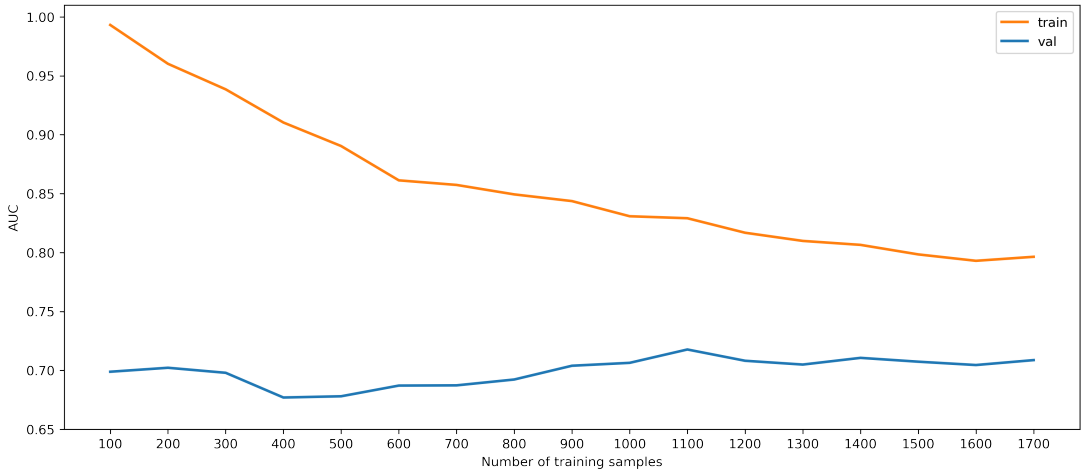

Figure 4.10: Random Forest learning curves

The best results were achieved by Logistic Regression as the training curve and validation curve in Figure 4.9 steadily converged at 1700 training samples. XGBoosts

curves show a bit more distance from each other on the right side of 4.8 which indicates that more data would improve the algorithm's performance, which is even more evident for Random Forest as seen in Figure 4.10. The differences between AUC training and validation scores at 1700 training samples are 0.0285 for XGBoost, 0.0041 for Logistic Regression, and 0.0877 for Random Forest.

4.3 Summary

This chapter described the steps taken to create, test, and compare eight machine learning models used for predicting probability of BNI members' non-renewals. Unfortunately, the thesis goal of a model surpassing an AUC score of 0.8 was not met as the final AUC score achieved by XGBoost was 0.7111. Flowchart 4.11 shows a summary of steps undertaken in the process to create the final version of the model:

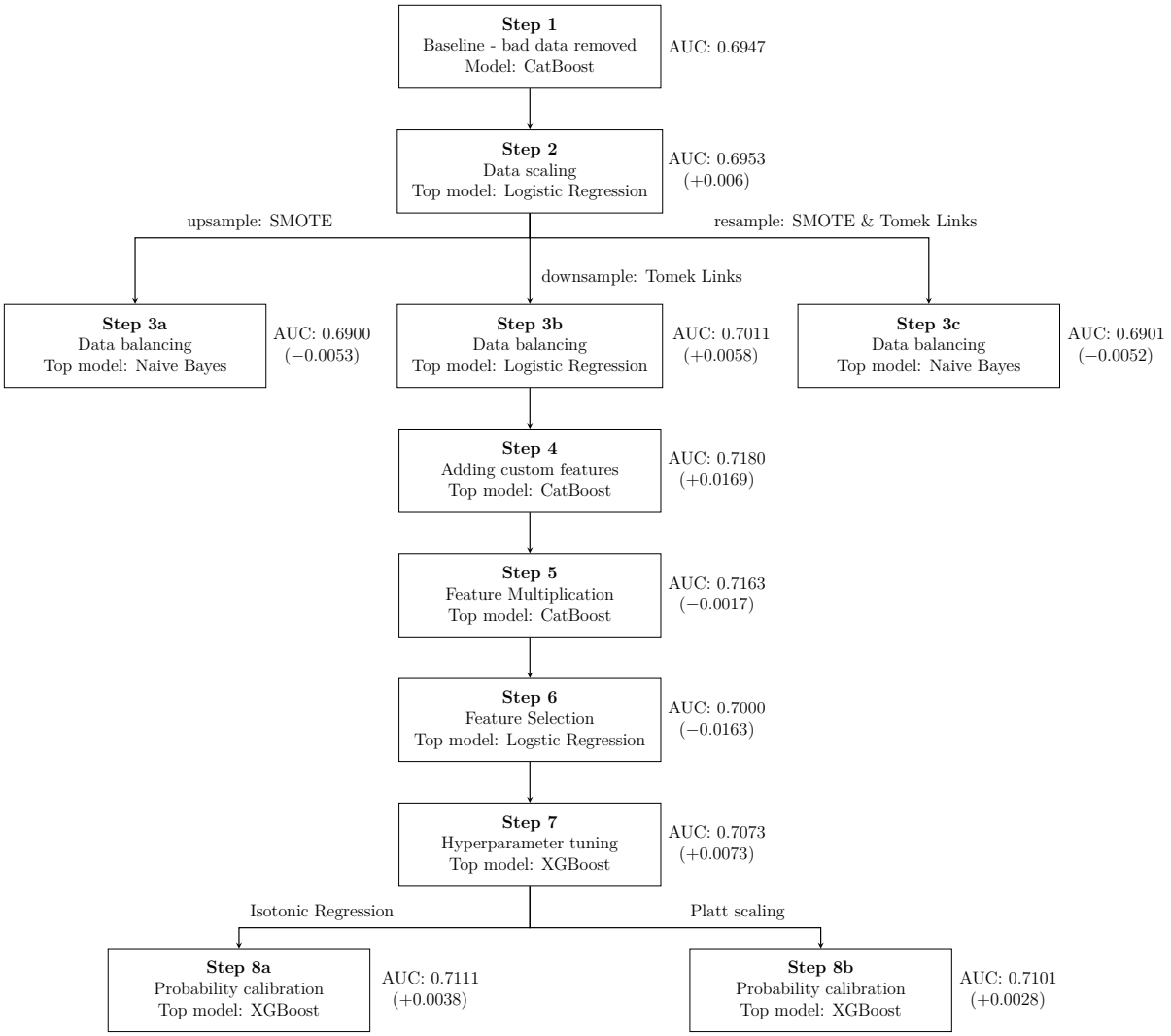

Figure 4.11: Diagram of machine learning modelling steps

Furthermore, XGBoost, Logistic Regression, and Random Forest learning curves yielded that the algorithm that converged best was Logistic Regression, followed by XGBoost, then Random Forest.

Conclusion

This thesis began by introducing the reader to the BNI company to provide a context for available data and BNI membership. Two project goals were set for this thesis:

1. create a machine learning solution for BNI ABS with an AUC score of at least 0.8 for predicting member non-renewals, and

2. determine the most important features according to each model.

To meet these goals, the data was prepared for machine learning classification, and thus it was cleaned, aggregated, and labeled. Additionally, new features were created, which indicated more than member-level performance initially included in the dataset. Upon completing an extensive data cleaning process, the data was explored using summary statistics and graphical representations. The machine learning modelling consisted of 8 steps involving data transformations and model tampering, throughout which the baseline AUC score of 0.6947 increased to 0.7111 which, unfortunately, fell short of the goals set for this thesis. Furthermore, a recursive feature selection algorithm was used to lower the dimension of the dataset and determine the most important factors for member non-renewals according to each of the models.

Future works could include reaching out to the BNI company to inquire about more data lacking in the available dataset. Such data would include but would not be limited to members' return on investment from being a member of BNI, with whom was a 1-2-1 meeting conducted, and to whom did a member give and receive each referral. This information would allow establishing relationship levels within and across chapters, resulting in a possibility of testing different methods for churn prediction, such as social graphs. Additionally, other data scaling methods could be examined for machine learning performance and an ensemble model made out of models used throughout this thesis could be tested.

Acknowledgements

Special thanks to Moji Ajele, M Eng. who was kind enough to employ me during my studies and willing to extend our cooperation by sharing BNI data for this thesis paper. Also, thanks to dr inż. Radosław Michalski for the supervision, guidance, and necessary paperwork. This thesis paper would not have come into existence without both of you.

Bibliography

[1] Asthana, P. A comparison of machine learning techniques for customer churn prediction. *International Journal of Pure and Applied Mathematics* (2018), 1149–1169.

[2] Berkson, J. Application of the logistic function to bio-assay. *Journal of the American Statistical Association 39*, 227 (1944), 357–365.

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

[4] Breiman, L., Friedman, J., Olshen, R., Stone, C. *Classification and Regression Trees*. 1984.

[5] Chawla, N. V., Bowyer, K. W., Hall, L. O., Kegelmeyer, W. P. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research 16* (Jun 2002), 321–357.

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

[7] F., C. T., Golub, G. H., LeVeque, R. J. Updating formulae and a pairwise algorithm for computing sample variances.

[8] Fawcett, T. An introduction to roc analysis. *Pattern Recognition Letters 27*, 8 (2006), 861–874.

[9] Freund, Y., Schapire, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences 55*, 1 (1997), 119–139.

[10] Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., Oliphant, T. E. Array programming with NumPy. *Nature 585*, 7825 (Sept. 2020), 357–362.

[11] Khan, M. R., Manoj, J., Singh, A., Blumenstock, J. Behavioral modeling for churn prediction: Early indicators and accurate predictors of custom defection and loyalty. *015 IEEE International Congress on Big Data* (2015), 1149–1169.

[12] Lalwani, P., Mishra, M. K., Chadha, J. S., Sethi, P. Customer churn prediction system: A machine learning approach. *Computing* (2021).

[13] Niculescu-Mizil, A., Caruana, R. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd International Conference on Machine Learning* (New York, NY, USA, 2005), ICML '05, Association for Computing Machinery, p. 625–632.

[14] Nie, G., Rowe, W., Zhang, L., Tian, Y., Shi, Y. Credit card churn forecasting by logistic regression and decision tree. *Expert Systems with Applications* (2011).

[15] pandas development team, T. pandas-dev/pandas: Pandas, Feb. 2020.

[16] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12* (2011), 2825–2830.

[17] Piegl, L. A., Tiller, W. Algorithm for finding all k nearest neighbors. *Computer-Aided Design 34*, 2 (2002), 167–172.

[18] Platt, J. C. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS* (1999), MIT Press, pp. 61–74.

[19] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., Gulin, A. Catboost: unbiased boosting with categorical features, 2019.

[20] Sebestyen, G. S. Classification decisions in pattern recognition, 1960.

[21] Tomek, I. Two modifications of cnn. *IEEE Transactions on Systems, Man, and Cybernetics 6* (1976), 769–772.

[22] Zadrozny, B., Elkan, C. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Proceedings of the Eighteenth International Conference on Machine Learning* (San Francisco, CA, USA, 2001), ICML '01, Morgan Kaufmann Publishers Inc., p. 609–616.