Churn Prediction for the Dutch Energy Market

Author: Roxana CRISTIAN
Supervisor: Dr. Mark HOEGENDOORN

A thesis submitted in fulfillment of the requirements for the degree of MSc Artificial Intelligence in the Artificial Intelligence Computer Science Department

January 27, 2016
Abstract

Faculty of Science
Computer Science Department

MSc Artificial Intelligence

Churn Prediction for the Dutch Energy Market

by Roxana Cristian

In highly competitive markets such as a liberalized energy market, where the offered product has the same properties regardless of the provider, emerging companies have to counter customer churn in order to strive. To reach this goal, a company has to be able to identify which customers are likely to churn. Although many researchers have studied churn prediction, not many have tried to predict churn for an energy market. Analyzing the existing literature and utilizing customer historical data of a Dutch energy supplier, this thesis tends to identify the best types of data to be used for churn prediction, as well as the best churn predictive model for the specific domain of an energy market. Following the CRISP-DM methodology, the historical data was collected, processed and analyzed before being used to fit predictive models created using different machine learning techniques. The predictive models have been evaluated by computing a set of evaluation metrics and the results have shown very good performance for some of them.
# Contents

Abstract ii

1 Introduction 1

2 Methodology and Background 3
  2.1 CRISP-DM Methodology .............................................. 3
  2.2 The Dutch Energy Market ........................................... 6
  2.3 Customer Churn ...................................................... 8

3 Related Work 10
  3.1 Data Categories ...................................................... 11
  3.2 Modeling ............................................................ 12

4 The Data 14
  4.1 Data categories ...................................................... 14
  4.2 Data Preparation .................................................... 15
    4.2.1 Data Cleaning .................................................. 17
    4.2.2 Data Aggregation .............................................. 19
    4.2.3 Data Transformation ........................................... 20
  4.3 Data Enrichment .................................................... 24
  4.4 Exploratory Data Analysis ......................................... 28

5 Theoretical Fundamentals 34
  5.1 Motivation ........................................................... 34
    5.1.1 Data Mining Techniques ...................................... 34
    5.1.2 Feature Selection Techniques ................................. 35
    5.1.3 Evaluation Metrics ............................................ 35
  5.2 Theoretical Fundamentals .......................................... 36
    5.2.1 Data Mining Techniques ...................................... 36
      Regression .......................................................... 36
      Decision Trees ..................................................... 37
      Random Forests .................................................... 38
      AdaBoost ............................................................ 39
      Support Vector Machines ........................................ 40
      Neural Networks ................................................... 41
    5.2.2 Feature Selection .............................................. 43
      Principal Component Analysis .................................... 43
      Alternative Feature Selection Method .......................... 44
    5.2.3 Evaluation Metrics ............................................ 45
      Typical Evaluation Approach .................................... 45
      Sensitivity .......................................................... 46
      Specificity ........................................................... 46
6 Modeling
6.1 Modeling Setup ........................................ 49
6.2 Implementation Details ................................. 52
  6.2.1 Models .............................................. 52
  6.2.2 Feature Selection .................................. 53
  6.2.3 Evaluation Metrics ................................. 53

7 Evaluation
7.1 General considerations ............................... 55
7.2 Removal of correlated attributes .................. 56
7.3 Removal of outliers .................................. 56
7.4 Principal Components Analysis .................... 57
7.5 PCA and Removal of correlated attributes ........ 57
7.6 Algorithms .............................................. 59
7.7 General evaluation .................................... 59
7.8 Most frequently used attributes ................. 61

8 Conclusions and future work ......................... 64
8.1 Conclusions ............................................. 64
  8.1.1 What information is needed to predict customer churn of energy suppliers and how useful each type of information is for churn prediction? .............................................. 64
  8.1.2 Which of the different prediction models that were successfully used for churn prediction in other domains is the most suitable one for the Dutch energy market? ......................... 65
  8.1.3 Can customer churn of energy suppliers be predicted with an accuracy that would allow a company to rely on prediction for retaining their customers? ......................... 65
8.2 Future work ............................................. 66
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>The CRISP-DM life cycle</td>
<td>5</td>
</tr>
<tr>
<td>2.2</td>
<td>Energy market entities and their roles</td>
<td>7</td>
</tr>
<tr>
<td>4.1</td>
<td>Frequency distribution of PersonEmailDomain</td>
<td>23</td>
</tr>
<tr>
<td>4.2</td>
<td>Frequency distribution of AggregatedEmailDomain</td>
<td>24</td>
</tr>
<tr>
<td>4.3</td>
<td>Density of CustomerAge attribute</td>
<td>29</td>
</tr>
<tr>
<td>4.4</td>
<td>Dependency of consumption and socio-economic attributes</td>
<td>30</td>
</tr>
<tr>
<td>4.5</td>
<td>Density for socio-economic attributes</td>
<td>30</td>
</tr>
<tr>
<td>4.6</td>
<td>Boxplots for numeric highly correlated attributes</td>
<td>31</td>
</tr>
<tr>
<td>4.7</td>
<td>Comparison of ContractCashback and PaidCashback</td>
<td>32</td>
</tr>
<tr>
<td>5.1</td>
<td>Pseudo code for AdaBoost algorithm</td>
<td>39</td>
</tr>
<tr>
<td>5.2</td>
<td>Classification for linearly separable classes</td>
<td>40</td>
</tr>
<tr>
<td>5.3</td>
<td>Neural network with one hidden layer</td>
<td>42</td>
</tr>
<tr>
<td>7.1</td>
<td>Evaluation metrics for important models</td>
<td>60</td>
</tr>
<tr>
<td>7.2</td>
<td>ROC curves for important models</td>
<td>61</td>
</tr>
</tbody>
</table>
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Mapping between the phases of CRISP-DM model and the sections of this thesis</td>
<td>6</td>
</tr>
<tr>
<td>4.1</td>
<td>Attributes from data set 1</td>
<td>16</td>
</tr>
<tr>
<td>4.2</td>
<td>Attributes from data set 2</td>
<td>16</td>
</tr>
<tr>
<td>4.3</td>
<td>Attributes from data set 3</td>
<td>18</td>
</tr>
<tr>
<td>4.4</td>
<td>Computed attributes</td>
<td>21</td>
</tr>
<tr>
<td>4.5</td>
<td>Socio-economic attributes</td>
<td>25</td>
</tr>
<tr>
<td>4.6</td>
<td>Attributes of the final data set</td>
<td>28</td>
</tr>
<tr>
<td>4.7</td>
<td>Correlation of different attributes to the target</td>
<td>31</td>
</tr>
<tr>
<td>5.1</td>
<td>Confusion matrix for a churn predictive model</td>
<td>45</td>
</tr>
<tr>
<td>7.1</td>
<td>AUC and Computation time for each algorithm, varying the removal of</td>
<td>56</td>
</tr>
<tr>
<td></td>
<td>correlated attributes parameter</td>
<td></td>
</tr>
<tr>
<td>7.2</td>
<td>AUC and Computation time for each algorithm, varying the removal of</td>
<td>57</td>
</tr>
<tr>
<td></td>
<td>outliers parameter</td>
<td></td>
</tr>
<tr>
<td>7.3</td>
<td>AUC and Computation time for each algorithm, varying the use of PCA</td>
<td>58</td>
</tr>
<tr>
<td>7.4</td>
<td>AUC and Computation time for each algorithm, varying the use of PCA</td>
<td>58</td>
</tr>
<tr>
<td></td>
<td>and Correlated attributes</td>
<td></td>
</tr>
<tr>
<td>7.5</td>
<td>AUC and Computation time for each algorithm</td>
<td>59</td>
</tr>
<tr>
<td>7.6</td>
<td>Evaluation metrics for the most important models</td>
<td>60</td>
</tr>
<tr>
<td>7.7</td>
<td>Usage of attributes for C50, RF and ADA models</td>
<td>62</td>
</tr>
</tbody>
</table>
## List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>DM</td>
<td>Data Mining</td>
</tr>
<tr>
<td>KD</td>
<td>Knowledge Discovery</td>
</tr>
<tr>
<td>KDD</td>
<td>Knowledge Discovery in Databases</td>
</tr>
<tr>
<td>KDMM</td>
<td>Knowledge Discovery in Data Mining</td>
</tr>
<tr>
<td>CRISP-DM</td>
<td>CRoss-Industry Standard Process for Data Mining</td>
</tr>
<tr>
<td>ZIP</td>
<td>Zone Improvement Plan</td>
</tr>
<tr>
<td>CCB</td>
<td>Customer Contract and Behavior</td>
</tr>
<tr>
<td>CP</td>
<td>Customer Perception</td>
</tr>
<tr>
<td>CSD</td>
<td>Customer Socio-Demographics</td>
</tr>
<tr>
<td>CPD</td>
<td>Customer Personal Data</td>
</tr>
<tr>
<td>DT</td>
<td>Decision Trees</td>
</tr>
<tr>
<td>RF</td>
<td>Random Forests</td>
</tr>
<tr>
<td>NN</td>
<td>Neural Networks</td>
</tr>
<tr>
<td>ADA</td>
<td>ADABoost</td>
</tr>
<tr>
<td>GLM</td>
<td>Generalized Linear Model</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machines</td>
</tr>
<tr>
<td>PCA</td>
<td>Principal Components Analysis</td>
</tr>
<tr>
<td>AUC</td>
<td>Area Under the Curve</td>
</tr>
<tr>
<td>ROC</td>
<td>Receiver Operating Characteristic</td>
</tr>
<tr>
<td>CI</td>
<td>Confidence Interval</td>
</tr>
<tr>
<td>SE</td>
<td>Standard Error</td>
</tr>
<tr>
<td>TP</td>
<td>True Positives</td>
</tr>
<tr>
<td>FP</td>
<td>False Positives</td>
</tr>
<tr>
<td>TN</td>
<td>True Negatives</td>
</tr>
<tr>
<td>FN</td>
<td>False Negatives</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

In liberalized energy markets, such as the Dutch energy market, the most important aspect is competition, since large and experienced energy companies compete with emerging and therefore less experienced companies, offering the same utility [1]. Moreover, this utility displays the exact same qualities regardless of the company that is supplying it. This results into customers switching their supplier often, looking for the best offers and services [2]. As a consequence, customer acquisition is more expensive than customer retention [2, 3, 4], which results into the necessity of giving a higher priority to customer retention. This is translated into the necessity of investing effort into retaining customers, rather than into winning new ones. Hence, customer churn, which is defined as changing loyalties in favor of a different service provider [5], has to be countered. Accordingly, an accurate and effective customer churn predictive model would be a strong tool for an energy company, in its quest of keeping existing customers.

The purpose of this thesis is to evaluate commonly used data mining techniques for identification of churn in order to create a churn predictive model for an emerging Dutch energy company, using historical data about its customers. The customer churn predictive model should be able to tell if an existing customer is likely to leave the company when his contract ends or not. The company can use this information to lower their customer retention costs by investing only in those customers that are likely to leave. In terms of data mining, churn prediction is a classification problem. The model should determine if a certain customer belongs to the churner class or to the non churner one, based on information about past churners and non churners. The model is trained on historical data and classifies customers by comparing them with old churners. Therefore, churners are identified by finding similarities between customers to be classified and customers that churned.

Research has proven that liberalized markets change with age [6, 7], in the sense that in emerging liberalized markets customers switch less as compared to more mature markets. Hence, a prediction model created for a recently liberalized market might not be as accurate for predicting churn for another energy market, which was liberalized earlier. Consequently, a study on churn prediction for energy suppliers performed
for an energy market other than the Dutch one, might not be relevant for this specific energy market. To the best of our knowledge, there is only one other similar study [8], which is for the Belgian market and was performed independently in parallel with this thesis. Hence the need of creating a list of relevant data for this type of churn. For this purpose the existing literature about churn prediction in similar markets, such as telecommunications, was studied and domain experts were also consulted.

This thesis focuses on the following research questions:

- What information is needed to predict churn for customers of energy suppliers and how useful each type of information is for churn prediction?
- Which one of the different prediction models that were successfully used for churn prediction in other domains is the most suitable one for the Dutch energy market?
- Can customer churn of energy suppliers be predicted with an accuracy that would allow a company to rely on prediction for retaining their customers?

The contributions of this thesis are:

- A list of data categories that can be used for predicting customer churn of Dutch energy suppliers
- A churn prediction model for the Dutch energy market
- The performance evaluation of multiple predictive models

This thesis follows the CRISP-DM methodology and it is structured as follows. Chapter 2 introduces the reader into the context by explaining the used methodology as well as offering some details on the Dutch energy market and the concepts of customer loyalty and churn by looking at different studies, some of them having a marketing perspective, while others a data mining one. Chapter 3 offers a partial answer to the first research question by analyzing related research on churn prediction form a data perspective. Further, by looking at the related work regarding churn prediction from a modeling perspective, the most common techniques used in the field are extracted. In Chapter 4 the data is described, along with the exploratory data analysis that was performed on it, including the pre-processing step. Chapter 5 starts by listing the modeling solutions that were chosen, such as feature selection techniques and prediction algorithms, as well as the evaluation metrics. Next, it provides the theoretical fundamentals of each concept. Chapter 6 describes the modeling setup and implementation details, while Chapter 7 presents the evaluation of the performance of each model and the comparison of results. Finally, Chapter 8 offers the conclusions of this study and proposes some future work.
Chapter 2

Methodology and Background

This chapter explains concepts that will help the reader understand the methodology that this thesis follows as well as the background of its subject. First, the CRISP-DM methodology will be described, together with the way it was used while developing this thesis. The second section offers a bird’s eye view of the Dutch energy market, while the third one tries to explain the churn issue.

2.1 CRISP-DM Methodology

Prior to presenting the methodology, some key concepts have to be clarified. A good start is the term "Data Mining", which is the method usually used for performing predictions. In [9] data mining, or DM, is defined as "the extraction of implicit, previously unknown, and potentially useful information from data". The purpose would be to use the found patterns to make accurate predictions on future data. According to the authors of [9], data mining involves machine learning. One of the definitions of learning offered in [9] is changing behaviour in a way that improves future performance. The best way to understand how machine learning is used for data mining is to think that machine learning techniques look for structural descriptions of what is learned, while the purpose of data mining is discovering patterns [9]. Therefore, if the large quantities of data in which data mining is trying to find patterns is used for learning, the result is a model that describes found patterns. Moreover, the resulted model can also be used for automated prediction.

Moving a step forward, it is important to explain the term "Knowledge Discovery" or KD, which is defined as the process of seeking new knowledge about an application domain. DM, on the other hand, involves applying low-level methods in order to extract patterns and it is only a step of KD [10]. KDD, or Knowledge Discovery in Databases concerns the KD process applied to databases [10] and can be viewed as a particularization of KDDM, or Knowledge Discovery and Data Mining, which is defined as the
application of KD to any data source [10]. As stated in [10], KDDM covers the entire process of knowledge discovery and it has been proposed as the most appropriate name for the overall process of Knowledge Discovery.

Considering that the purpose of this thesis is to create a churn predictive model, while also finding new knowledge in the domain of churn prediction for the Dutch energy market, it looks like the best way to structure it is following KDDM. However, as the authors of [10] explain, a process model for KDDM is needed to ensure that the end product will be useful. Moreover, a KDDM process model should be able to assure that formal models of knowledge seeking are used, which should finally lead to discover of valuable data [10].

CRISP-DM is one of the most popular process models that were conceived for KDDM [10, 11]. Extensive research has already been done trying to create a "standard process model for data mining" [12]. The findings were in favor of the above mentioned process model and one of the most notable results was that CRISP-DM is independent of both the industry sector and the technology used. The acronym stands for CRoss-Industry Standard Process for Data Mining and the process model was developed by a consortium initially composed of DaimlerChrysler, SPSS and NCR[11].

In their study, the authors of [10] compare CRISP-DM with four other KDDM process models and observe that they all consist of multiple steps which are executed in a sequence, often including loops and iterations. Two other common features are stated to be the span of covered activities and the incorporation of feedback loops, triggered by a revision process. Although covering the same activities, from understanding of the domain and the data, through data preparation, data analysis, to application of results, the number of steps differs from model to model, as well as the scope of each of the steps [10]. Among the models compared and described in [10] the one developed by Fayyad et al. can be mentioned, as it is the first reported KDDM model and it is also used for comparison by the authors of [11], together with SEMMA and CRISP-DM. The two above mentioned studies compare widespread KDDM process models motivating that a standard process is needed for good results and ease of deployment. While [10] identifies CRISP-DM as the most suitable KDDM process model for novices in the DM field, especially for industrial projects, [11] finds the same model the most complete of those compared. As a consequence, CRISP-DM was chosen as process model for this thesis.

The six stages of CRISP-DM are the following [12]:

1. Business understanding - In this phase, an understanding of the project’s objectives and requirements from a business perspective should be achieved. The result should be a data mining problem definition and a preliminary project plan.
2. Data understanding - This phase starts with data collection and its purpose is finding data quality problems as well as insights that would allow the formation of some hypotheses for hidden information.

3. Data preparation - All the activities that are necessary for obtaining a final data set.

4. Modeling - This phase covers the selection of models and their implementation and tuning, using the already constructed data set.

5. Evaluation - During this phase, the models obtained in the previous phase are evaluated in order to make sure that the objectives are achieved.

6. Deployment - After a final model has been obtained, the data might need some transformation that would make it more readable or usable for the customer. All these transformations are performed in the deployment phase.

Figure 2.1: The CRISP-DM life cycle

Figure 2.1 shows the CRISP-DM life cycle. The most important thing to observe is that the sequence of the six phases is flexible; the arrows indicate the most important and frequent dependencies, without strictly limiting the order of phases. This flexibility gives the researchers the liberty to go back to any of the previous phases in order to obtain better results in the current one. They can start over if the obtained results are not satisfactory, as well go back to data transformation while tuning the models. The
CRISP-DM phase | Thesis section
--- | ---
Business understanding | Chapter 3
Data understanding | Chapter 3
Data preparation | Chapter 4
Modeling | Chapter 6
Evaluation | Chapter 7
Deployment | Chapter 8

**Table 2.1: Mapping between the phases of CRISP-DM model and the sections of this thesis**

outer circle symbolizes the cyclic nature of DM: deployment can offer new business questions, so the process can be re-started.

According to [12], each of the phases previously described can be further split into a number of tasks and even sub-tasks. The authors also distinguishes between the "Reference Model", which represents an overview of phases and tasks, and User Guide, which gives detailed instructions for each phase and task. Therefore, though following the CRISP-MD methodology, the structure of this thesis does not follow all of the phases to the letter. The mapping between the CRISP-DM phases and the chapters of this thesis can be found in Table 2.1.

### 2.2 The Dutch Energy Market

Studies such as [13] describe the evolution of the Dutch electricity market before and after the liberalization. According to the authors of this study, the Dutch government started liberalizing the Dutch energy market in 1998, in four steps: large users in 1998, the middle segment in 2002, green energy market for all consumers in 2001 and the remaining in 2004. Before 1998, 23 distribution companies, which were owned by municipalities and provinces, were supplying electricity in designated territories. Privatization allowed three large players to emerge in the new market: Nuon, Essent and Eneco. The large players, which were created by old monopolists that had merged together after being privatized, had to face two different kinds of competitors: each other, since they were no longer geographically restricted, and new entrants. The old monopolists have the advantage of being well known companies, with very loyal customers, though the disadvantage of not being accustomed to compete.

In the previous paragraph, the changes on the Dutch energy market were discussed from the perspective of energy suppliers. Figure 2.2 shows how the energy supply companies operate between the end-consumer of energy and the companies that produce and distribute the energy.
All energy suppliers offer combinations of electricity and gas, in the form of contracts for 1 year, 3 years or for a variable period. The fixed length contracts are bounding the customer to the supplier for the designated period of time. If the customer decides to leave earlier, the company can force the customer to pay a fine. This is what the supplier companies informally call a "happy churn", since the company receives money without having to provide neither energy nor customer service to the churning customer. The last type of contract only binds the customer for one month. In order to close more long term contracts, the suppliers offer better tariffs and even "cashbacks" to their prospect customers. "Cashback" is a term used to describe an amount of money that the company pays to a client after the contractual period of one or three years has ended, therefore rewarding the client for not churning. On their quest of convincing customers that used to be loyal to the more experienced companies to switch, the younger ones have to be visible and make their offers known on different channels, such as: the website of the company, comparison websites, auction companies, door to door companies, advertisement.

An emerging energy supply company, which entered the Dutch energy market at the end of 2011, offered their data and domain knowledge for the purposes of this thesis. Its target customers are consumers and small business users. The company reached 2% market share in 2014. At about the same time it started facing the issue of customer churn, as the 3 years contracts closed in the first year were ending and the company had no retention procedure in place. More details on the churn concept will be given in the next section.
2.3 Customer Churn

In a market governed by competition, energy suppliers have become aware of the increasing need to invest in retaining existing customers. Additionally, studies [2, 3, 4] have shown that customer retention is less expensive than customer acquisition. To support this idea, the nature of the offered product has to be discussed. Gas and electricity have the same quality regardless of the supplier, especially since the distributors are still designated and limited to geographical areas. Consequently, suppliers can only acquire customers through marketing. As a result, energy suppliers are putting effort into preventing the loss of customers. More economic advantages of customer retention are offered by the authors of [14]:

- decreasing the need of customer acquisition allows focusing on the demands of existing customers
- positive word of mouth from satisfied customers results into customer acquisition, while share of negative experiences results into negative image among prospect customers
- long-term customers are less costly to serve
- long-term customers tend to buy more
- long-term customers are less sensitive to marketing strategies performed by competitors

After emphasizing the advantages of customer retention, a short presentation of the opposite term is the natural step. Although churn is defined in slightly different ways throughout the existing DM and marketing literature, such as changing loyalties in favor of a different service provider [5], discontinuation of a contract [14] and move of custom to a competing service provider [15], the main idea remains the same: a churner is a customer that chooses to use the same service from a competing provider. Even though the term in general includes both service-provider initiated churn and customer initiated churn [3], for energy suppliers in the Dutch energy market, the first case does not represent a problem. This is because, if a customer churns before the end of the contractual period, the company can fine him, therefore gaining money without having to provide energy or service. As a result, only customer initiated churn will be considered and named churn for the remaining of this thesis.

Customer switching or churn, together with its opposite, customer loyalty, make the subject of many marketing studies. As stated in [6], several studies show that customer loyalty is influenced by customer satisfaction, which is closely related to customer churn, while others go into more detail and classify determinants of loyalty in two groups: cognitive/economic and social/affective. Looking from the perspective of customer churn, [16] examines the motives of consumers to switch energy suppliers...
and identifies two main factors: dissatisfaction or relationship fatigue and "monetary-motivated curiosity", which is described as an economic factor in which recommendations of friends also plays a role. It is easy to observe that the results are similar regardless of the perspective and that they indicate that the quality of service is important, together with monetary incentives and image of the company. This conclusion will be enforced in Chapter 3, where an analysis of the data categories used for churn prediction by other researchers will show that the above mentioned factors are commonly used in this field.

Examining from yet another perspective, the authors of [6] note that liberalizing markets have a special property: the fact that many of the customers were forced to be loyal prior to the liberalization. This would be the result of the mistrust they have in the new entrants, which increases switching cost for the customer. Moreover, customers of the monopolists are not accustomed to switching and are already in a long-term relationship with the supplier company. However, once these obstacles have been overcome, there seems to be no reason for the customer to not switch again. Therefore, from the point of view of the emerging energy suppliers, the market is as competitive as any other market, which was always free. This theory is confirmed by the domain expert, who states that the market is highly competitive and customers seem to be mostly driven by economic reasons. To his support comes the fact that the main channels for customer acquisition for the young suppliers are comparison websites, which rank suppliers mostly on the price offers, and auctions. For an auction, a few thousand prospects are gathered by an external company and price offers are requested from all available suppliers. The company that offers the best price closes contracts with all the prospects that participated. This also supports the above mentioned monetary factor.

It has been highlighted in this section that customer retention has tremendous value and needs to be enforced. However, in order to keep the retention costs down, the suppliers have to be able to predict which customers are going to churn. Special price offers can then be made only to the customers that exhibit a churning risk.
Chapter 3

Related Work

To the best of our knowledge, there is only one other study [8] on churn prediction for the energy market. This study focuses on the usage of high cardinality attributes, such as bank account number, family names or postal codes, in predictive modeling, taking the energy market as a case study. One of the stated purposes of the authors of this paper is finding out how to transform high cardinality attributes in order to use them for prediction. The same authors are able to identify and apply a few methods that allow including high cardinality features when predicting, such as:

- dummy encoding, where all possible values of a nominal feature are transformed into separate boolean features,
- semantic grouping, where semantically meaningful groups are identified for the feature and transformation to continuous attribute

Different classification techniques are used for the purposes of the above mentioned study: C4.5 Decision Trees, Logistic regression and Support Vector Machines [8]. The results of this study demonstrate that including high cardinality attributes improves the performance of churn prediction: an increase of AUC from 67.7 for the base model to 74.39 for the model created with data transformed with one of the methods proposed as transformation to continuous attribute is obtained.

As mentioned above, the focus of [8] is on a certain type of attributes and the energy market is only a case study. Moreover, the study was not yet published at the moment when the current research was conducted. Therefore, the rest of this chapter will present studies that investigate either customer churn prediction, from a data mining perspective, or customer loyalty or churn from a marketing perspective. The first section provides an answer to the first of the Research questions by analyzing both data mining and marketing related work, while the second explains some modeling choices that were made while this research was conducted, by looking only at data mining studies focused on churn prediction.
3.1 Data Categories

Starting from the 1990s, many researchers from various domains have studied churn prediction. Most of the studies [1, 4, 17, 18, 19, 20] focused on or chose as case study the domain of telecommunications. Others were conducted for other subscription based domains, like internet market [21, 22], banking [23, 24, 25, 26] or insurances [22], as well as for internet related businesses like email providers [27] or e-commerce platforms [25, 28]. However, since none of the published studies had performed prediction for the customers of the energy market, a decision was made to create a list of data categories that could be used for this type of prediction. This was accomplished by reviewing churn prediction studies, such as those mentioned in the first part of this chapter, as well as marketing studies focused on the energy market.

The authors of [14] tried to identify commonly used churn prediction methods and evaluation techniques. However, they have used a list of 4 data variable sets that was initially presented in [29] and then used to describe the groups of data variables in other churn prediction studies [5, 24, 30] as well, for different domains. The list is as follows:

- customer behavior
- customer perceptions
- customer demographics
- macro-environment variables

Other studies, such as [22] use some of the data variable sets, or categories, described in [29], while also including "relationship" or contract data. Marketing studies such as [31] and [32] show that customer trust and satisfaction have a high influence over customer loyalty. It is important to mention that, while the second one of the above mentioned studies has inquired how customer retention in general, with an empirical study in the telecommunication market, is influenced by customer satisfaction and other markers, the first one has examined the customer loyalty for the residential energy market. Customer satisfaction and customer trust fall into the "customer perception" category mentioned above, therefore supporting the usage of the same list of categories for churn prediction for the energy market. Another marketing study, [16], demonstrated that, besides customer satisfaction, switching an energy supplier can also be motivated by monetary incentives, which, if considered as part of the contractual agreement, fall under the "contract data" category. Although this seems to be a new category, it can also be considered as part of the customer behavior category. Another option, which was chosen for better clarity, would be to create a combined category, called customer contract and behavior.
More recent studies [20, 28, 33] have investigated different ways of improving churn prediction by mining social network data. The idea behind this is that the decision to churn can be influenced by neighbours inside the social network that have already churned. However, the authors of [20] state that the added value of using social data is minimal and does not justify the increase of computational costs.

3.2 Modeling

Many data mining or machine learning techniques were investigated in order to increase the performance of churn prediction. Some of them are: Decision or Classification Trees [1, 15, 22, 23, 27, 28, 34], Random Forests [24, 25, 28], Logistic Regression [1, 2, 22, 34, 35], Neural Networks [23, 24, 28, 33, 34], Support Vector Machines [2, 5, 24, 26, 34, 36]. Moreover, some of the studies perform a comparison of different models. For example, [28] compares Naive Bayes, Decision Trees, Random Forests and Neural Networks and reports that Decision Trees and Random Forests have the best performance, both measuring specificity of over 0.95. However, when comparing the values of sensitivity, the Random Forests model performs better, reaching 0.544, while the model based on Decision Trees, scores 0.412. These results, however, are obtained for one of the datasets, while, for the other, the specificity is around 0.6 and the sensitivity is very high, 0.99 for the model based on Decision Trees. Despite this difference, the Decision Trees and random Forests model still perform better when compared to Naive Bayes and Neural Networks.

Decision Trees also has the best results in the tests performed by the authors of [37], where the method was compared to Neural Networks and Regression. Accuracy measured for the Decision Trees model was found to be 82%, while accuracy measured for the Neural Network model, of 72%.

Looking at other studies, [34] compares Naive Bayes, Decision Trees, Neural Networks, Logistic regression and Support Vector Machines and finds that the latter performs best, having an accuracy of 0.91 for one of the data sets and 0.6 for the other one. The other models registered accuracy values between 0.83 and 0.89 for the first data set and between 0.52 and 0.58 for the second one. Both data sets were from the domain of telecommunications: the first one for cellular, while the second one for home telecommunications.

Some of the more recent studies have tried less traditional approaches like Rotation Forests [17, 25], Evolutionary algorithms [19, 38] or Self Organizing Maps [39, 40] and have mostly shown that they outperform the more traditional models. The models based on Rotation Forests have registered accuracy values above 0.8, while those based on Self Organizing maps have registered an accuracy of 0.99. However, the first exception is [39], which have come the conclusion that, although the models based on Self
Organizing Maps have reached an accuracy of 0.93, the Neural networks models outperformed them, reaching an accuracy of 0.943. The second exception is [19], which concluded that the evolutionary algorithm has proved to be "impractical and ineffective on large dataset with high dimension".

It is common among the churn prediction researchers to use feature selection techniques to improve prediction performance as well as computation time [3, 4, 17, 23, 25, 26, 34, 35, 36, 41]. Although many feature selection techniques are available and have been used with success, the favored technique [17, 23, 25, 41] is Principal Components Analysis or PCA. This method has proved to be efficient in handling highly dimensional data [23]. Moreover, [41] has shown that PCA surpasses other feature selection methods.

Another issue that many of the researchers in the churn prediction field are trying to overcome is that of imbalanced data. While some researchers state that this issue can be overcome using PCA [23], others use dedicated methods such as random over sampling [4, 5, 26, 34, 37] or random under sampling [5, 17, 19, 25, 36]. However, not many studies have looked into which of the two sampling methods is better, especially for churn prediction. One study [42] has demonstrated that under-sampling can improve the accuracy of churn prediction when measured by AUC. Another study [43] concludes that under-sampling outperforms over-sampling by evaluating classification models based on C4.5 Decision Tree for 4 different data sets. It is also important to mention that although more sophisticated solutions for solving the class imbalance issue exist, they are used less often than random over and under sampling.

It was also observed during literature review, that authors of studies on churn prediction use different evaluation metrics for their models. A list of these evaluation metrics was extracted and is presented below. Additionally, more details on what each of the measures and the formulae will follow in Chapter 6.

- accuracy [2, 4, 5, 14, 15, 21, 24, 25, 26, 27, 34, 37, 38, 39, 40]
- AUC [1, 2, 17, 19, 25, 26, 35, 36]
- specificity [4, 5, 17, 21, 28, 14, 36, 40]
- sensitivity [17, 4, 21, 28, 14, 5, 36, 39, 40]
- precision [21, 39]
- F-measure [39]
- computation time [28, 38]
Chapter 4

The Data

As explained in Chapter 2, the sequence of the phases of the CRISP-DM process is flexible. Business understanding, for example, does not happen only at the beginning, especially since questions arise more often during the data preparation phase. Some data duplicates or missing values appear or are relevant only after data aggregation. And the list can go on. The sections and subsections of this chapter are organized in the ideal sequence, even though the actions were actually performed in a less ideal order.

Emerging energy supply companies do not always start with the best software or database structure in place. They also often fail to consider from the very beginning that the data gathered from customers will ever be used for marketing purposes. For this reason, the data that was provided for churn prediction, especially the part that was collected for early customers, is very inconsistent. In addition, it came in three different data sets.

This chapter is going to start by presenting the data categories as they were extracted from the related work studies. Next, it will offer insight into the data that was used for modeling starting from the initial data sets and presenting each of the data preparation steps, which include cleaning, transformation, aggregation and enrichment. It is worth mentioning that the data categories used throughout this entire chapter are those listed in the first subsection. However, abbreviations were sometimes employed for better use of space. At the end of the data preparation phase, the data set consisted of 42 attributes and 25433 rows. This final data set was used for modeling.

4.1 Data categories

Additionally to the categories extracted from the studies presented in Chapter 3, the domain expert thought that some patterns could also be found in the email domain of the customer, in the sense that the churners might prove to be using the same domains for their email addresses. An extra category was consequently added, customer personal data.
As a result, the following list of data categories was assumed to be relevant for churn prediction for the Dutch energy market:

- Customer contract and behavior: purchased services, duration of subscription, yearly consumption, payment behavior, offer discounts, customer source
- Customer perception: customer satisfaction
- Customer socio-demographics: gender, social status
- Macro environment variables: rank of company at specific moments, number of big market changes that the company overcame well while the customer had an ongoing contract
- Customer personal data: email domain

The above presented list constitutes half of the answer to the first of the Research Questions mentioned in Chapter 1. However, not all data categories were available to the author of this thesis. Details on what data was actually used will follow in the next sections.

### 4.2 Data Preparation

One of the provided data sets contained information about customers and contacts that the customer had with the Customer Service department, the second one contained information about contracts and the last one mixed information about the contracts and addresses, including information about the meters. The first data set had more than 250,000 rows, the second one about 235,000, while the third only had 200,000 rows. This is because a person can request an offer, which translates into an address entry. However, if one does not decide to become a customer, there will be no contract. Furthermore, an address entry can be the same for 2 different contracts and even for two different customers if two customers have lived at the same address at different moments of time. Tables 4.1, 4.2 and 4.3 contain the most important of the attributes that were present in the initial data sets, leaving out identification attributes like ContractId, CustomerId and so on. All the attributes were initially of type String, so they had to be transformed.

It is easy to observe by looking at Table 4.1 that some of the provided data could not be used in the initial format. PersonBirthDate for example, had to be transformed to PersonAge as no data mining algorithms can handle dates. The attributes of this table are from most of the categories and offer information about the satisfaction of the customer, as the number of contacts with the company for help or complaints, as well as information about the behavior of the customer, in terms of number of contracts and previous churn.
### Chapter 4. The Data

<table>
<thead>
<tr>
<th>Attribute name</th>
<th>Data category</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>PersonGender</td>
<td>CSD</td>
<td></td>
</tr>
<tr>
<td>PersonEmail</td>
<td>CPD</td>
<td></td>
</tr>
<tr>
<td>PersonBirthDate</td>
<td>CPD</td>
<td></td>
</tr>
<tr>
<td>PersonPhoneNumber</td>
<td>CPD</td>
<td></td>
</tr>
<tr>
<td>PersonContractsNumber</td>
<td>CCB</td>
<td>Number of contracts owned by the same customer</td>
</tr>
<tr>
<td>PersonAwayContractsNumber</td>
<td>CCB</td>
<td>Number of terminated contracts owned by the same customer</td>
</tr>
<tr>
<td>PersonAddressesNumber</td>
<td>CCB</td>
<td>Number of addresses that are serviced for the same customer</td>
</tr>
<tr>
<td>PersonEmailContactsNumber</td>
<td>CP</td>
<td>Number of email contacts with the Customer Service department for the same customer</td>
</tr>
<tr>
<td>PersonPhoneContactsNumber</td>
<td>CP</td>
<td>Number of phone contacts with the Customer Service department for the same customer</td>
</tr>
</tbody>
</table>

#### Table 4.1: Attributes from data set 1

<table>
<thead>
<tr>
<th>Attribute name</th>
<th>Data category</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ContractAddressesNumber</td>
<td>CCB</td>
<td>Number of addresses serviced under the same contract</td>
</tr>
<tr>
<td>PaymentMethod</td>
<td>CCB</td>
<td></td>
</tr>
<tr>
<td>HadToPayDeposit</td>
<td>CCB</td>
<td>Customers that are known to have payment issues have to pay a deposit at the start of the contract</td>
</tr>
<tr>
<td>Source</td>
<td>CCB</td>
<td>The sales channel used to acquire the customer</td>
</tr>
<tr>
<td>ContractCashback</td>
<td>CCB</td>
<td>The amount of money that the customer should receive when the contract expires</td>
</tr>
<tr>
<td>ContractCurrentBalance</td>
<td>CCB</td>
<td>The amount of money that the customer had to pay at the moment of data gathering</td>
</tr>
<tr>
<td>ContractPaymentsNumber</td>
<td>CCB</td>
<td>Number of payments performed by the customer by the moment of data gathering</td>
</tr>
<tr>
<td>ContractLatePaymentsNumber</td>
<td>CCB</td>
<td>Number of late payments performed by the customer by the moment of data gathering</td>
</tr>
<tr>
<td>ContractTerminationFee</td>
<td>CCB</td>
<td>Amount paid by the customer as fine for terminating the contract before the agreed end date</td>
</tr>
</tbody>
</table>

#### Table 4.2: Attributes from data set 2
Table 4.2 contains only attributes that belong to the Customer contract and behavior category. These attributes offer insight on some of the choices of the customer, such as the chosen payment method, but mostly on the behavior of the customer, like the mean that was chosen for signing up for a contract with the company, if one was paying the invoices on time and if the contract was terminated before the agreed end date.

The third data set, as shown in Table 4.3, offers more information about the contract, as well as behavior, in terms of energy consumption or the current status of the contract. The first attribute is Postcode and holds the postal or ZIP code. In the Netherlands the format of the postal codes is 4 digits followed by 2 letters. The ForcedLeft attribute tells if a customer churned or the contract was ended by the company, for missing or delayed payments for example. A few of the attributes in the third data set are redundant, since the energy providers have designated areas in which they operate. Therefore, the value of the Provider attribute can always be determined from the value of the Postcode attribute. Likewise, GasRegion can be determined from the postal code.

For a proper data analysis to be performed, the three data sets had to be aggregated. However, before proceeding to aggregating the data sets, some sanitization could already be performed.

4.2.1 Data Cleaning

First, and prior to data aggregation, each data set was cleaned by removing entries that were missing identity values, entries that were identified as duplicates or that had too many important missing values, such as PersonBirthDate, PersonEmail, PersonGender and PersonPhoneNumber. Secondly, one attribute that had only 0 values was removed, namely ContractLatePaymentsNumber. Secondly, even though the provided data was already partially anonymised, meaning that no names were present in the data sets, the email addresses were kept. For prediction, however, only the domain was necessary, as the domain expert thought that it would be an interesting attribute. Therefore, the PersonEmail attribute was replaced by the PersonEmailDomain attribute. Another attribute, namely ContractCashback, had values that contained an unnecessary currency character, which was removed. Similarly, the percentage symbol was removed from a few attributes. The PaidCashback attribute, which represents the amount of money that was paid to the customer after his contract expired, had positive as well as negative values. This was considered as being a saving mistake, so the absolute value was computed and stored for each value of the attribute. The PersonGender attribute also presented some inconsistencies: it had 4 possible values, as both upper case and lower case letters have been used. Since gender has only 2 possible values and working with binary attributes has a positive influence on the run time, the attribute was replaced by a binary one, called IsGenderMale. Although the identity attributes were, of course, not useful for modeling, they could not have been removed before aggregation.
<table>
<thead>
<tr>
<th>Attribute name</th>
<th>Data category</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Postcode</td>
<td>CSD</td>
<td></td>
</tr>
<tr>
<td>ContractStatus</td>
<td>CCB</td>
<td>It can be active or terminated</td>
</tr>
<tr>
<td>ForcedLeft</td>
<td>CCB</td>
<td>If the contract was terminated by the company (because the customer did not pay invoices)</td>
</tr>
<tr>
<td>ElectricityIsSingleMeter</td>
<td>CCB</td>
<td></td>
</tr>
<tr>
<td>ElectricityIsSmartMeter</td>
<td>CCB</td>
<td></td>
</tr>
<tr>
<td>ElectricityIsGivingBack</td>
<td>CCB</td>
<td>If the customer has solar panels which produce some electricity that is returned to the grid</td>
</tr>
<tr>
<td>ElectricitySJV</td>
<td>CCB</td>
<td>Yearly consumption of electricity</td>
</tr>
<tr>
<td>GasIsSmartMeter</td>
<td>CCB</td>
<td>Number of addresses that are serviced for the same customer</td>
</tr>
<tr>
<td>GasIsLargeConsumer</td>
<td>CCB</td>
<td></td>
</tr>
<tr>
<td>GasSJV</td>
<td>CCB</td>
<td>Yearly consumption of gas</td>
</tr>
<tr>
<td>GasRegion</td>
<td>CSD</td>
<td></td>
</tr>
<tr>
<td>StartContractDate</td>
<td>CCB</td>
<td>The agreed duration of the contract</td>
</tr>
<tr>
<td>EndContractDate</td>
<td>CCB</td>
<td></td>
</tr>
<tr>
<td>ContractDuration</td>
<td>CCB</td>
<td>Date of the start of delivery period</td>
</tr>
<tr>
<td>StartDeliveryDate</td>
<td>CCB</td>
<td>Date of the end of delivery period</td>
</tr>
<tr>
<td>EndDeliveryDate</td>
<td>CCB</td>
<td></td>
</tr>
<tr>
<td>PaidCashback</td>
<td>CCB</td>
<td>Amount of money already paid to the customer at the end of the contractual period</td>
</tr>
<tr>
<td>Provider</td>
<td>CSD</td>
<td>Tightly related to GasRegion and Postcode</td>
</tr>
</tbody>
</table>

Table 4.3: Attributes from data set 3

In order to make this section easier to follow and considering that no cleaning was necessary for the attributes obtained during aggregation, the aggregation step is skipped at this point and the cleaning that was performed afterwards is described next. This way the reader is able to find all the information about data cleaning in one place. The following subsection will describe the aggregation procedure in detail.

After the data was aggregated and a few new attributes were added, entries that had missing StartContractDate and ContractDuration were removed. This decision was made considering that these two fields contain important contract information and entries that lack this kind of information should not be taken into account. Following the same reasoning, entries that had values of CustomerAge lower than 18 were considered bad data and removed. Afterwards, the identification attributes were removed, together with date attributes. Finally, attributes referring to address data were removed, including Street, City, HouseNumber and Addition, since they were nominal and/or had a very high amount of possible values. Additionally, the information that the City attribute was offering, for example, was considered redundant, since the Postcode attribute was offering similar information and could have been transformed.
to numeric by keeping only the first 4 digits. More details about this transformation will be offered in subsection 4.2.3.

An important step in data cleaning is handling of missing values. For some numeric attributes, such as ContractCashback, the missing values were replaced with 0. This was decided because, according to the domain expert, for this attribute, a missing value meant that no cashback was offered. For other numeric attributes, such as GasSJ, which represents the yearly gas consumption, the missing values were replaced with NA during type transformation. The reason is that empty GasSJ does not mean that the customer had no yearly consumption, but that it was not recorded. For nominal attributes, missing values were transformed from one or more different possible values, such as empty, NULL or a dash, to "unknown". However, when a nominal attribute was replaced by some binary ones, an attribute named "IsUnknown" was created for the "unknown" possible values. More details about when and how this replacement was performed, see subsection 4.2.3. Regardless of these details, if the new "IsUnknown" field only had less than 5 entries, the attribute was removed as well as the entries that had unknown values for the particular attribute. An example is the ContractInitialType attribute, which only had 3 missing values, hence causing 3 entries to be removed from the data set.

Finally, entries that represented contracts that were not expired and still active at the moment when the modeling was performed were also removed, as not expired and active contracts could not be considered neither churned or not churned, hence being irrelevant for churn prediction. To deeply explain the reason, a customer that did not churn yet, but is not yet close to contract expiration, can be considered to not have yet had the chance to churn.

In order to make the data suitable for modeling, some attributes have suffered type transformations. However, type transformation is not considered cleaning, therefore it is discussed in a different subsection.

4.2.2 Data Aggregation

By looking at the structure of the data it is easy to observe that there were two options of aggregating it: by contract and by customer. While it would make sense to try to predict if a person is going to switch the company since the person makes the actual decision, there are many properties of the contract that would have been too hard to represent at a customer level for those customers that have more than one contract. Moreover, the domain experts have explained that sometimes the customer can have contracts that are not created at the same time, which means that they do not expire at the same time. Therefore, one contract might be switched when it expires if the customer had found a better offer, or was not happy with the customer service, but might change his or her mind before the next contract expires. In the light of these
arguments, a decision was made on aggregating the data by contract. However, a few steps were necessary for doing so.

Initially, the first 2 data sets were aggregated by a common field, PersonId. Next, the resulting data set was aggregated with the third one by a common field, ContractId. A new field was added during aggregation, NumberOfAddresses, and populated with the number of addresses that were found for the same contract, while the first one of these addresses was used for the actual entry. Although for some contracts some data was lost together with the ignored addresses, the domain expert considered that it was not important data, since a singular address can not be removed from the contract, meaning that churn can only happen at a contract level. Considering that contracts that have more than one address are usually owned by people who administrate the houses of relatives or companies that own apartment buildings, it is safe to assume that any of the addresses would offer the same indication of social status. Moreover, the number of addresses should offer some extra indication of the social status of the customer.

The NumberOfAddresses field proved to have the same value as an already existing field, ContractAddressesNumber. Some entries had 1 up to 3 extra addresses in the NumberOfAddresses field, and a closer analysis has shown that it was duplicated data. A decision was made to further consider ContractAddressesNumber as the correct number of addresses for each entry. The aggregated data set had 188661 rows and 53 attributes, including 3 identification fields.

### 4.2.3 Data Transformation

Owing to the fact that date is a complex type, which would introduce unnecessary complexity to the models a decision was made to compute age and durations from the provided date attributes. Unnecessary complexity is also introduced by nominal attributes with many possible values, such as postal codes, which led to the decision of computing a short postal code, of numerical type. This way, all the relevant information was kept while the problematic attributes could be removed.

Table 4.4 shows the attributes that were added and explains how each of them was computed and what it represents.

While the first 5 attributes are easy to understand, ContractInitialType needs some extra clarification. This clarification should start with the mention that the EndDeliveryDate (EDD) attribute is able to tell if a contract was terminated, as in the customer no longer uses the companies services, or not. If the contract was terminated, the EDD must have a value different than NULL. The EndContractDate (ECD) represents the date when the contract expires. The duration between StartContractDate (SCD), which represents the date when the contract was signed, and EndContractDate (ECD) is the
<table>
<thead>
<tr>
<th>Attribute name</th>
<th>Type</th>
<th>Computation formula</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomerAge</td>
<td>Integer</td>
<td>Number of years since PersonBirthDate</td>
<td></td>
</tr>
<tr>
<td>ShortPostcode</td>
<td>Integer</td>
<td>First 4 digits of Postcode</td>
<td></td>
</tr>
<tr>
<td>ContractDeliveryDuration</td>
<td>Integer</td>
<td>Number of months between SDD and EDD</td>
<td>Duration of service</td>
</tr>
<tr>
<td>BeforeDeliveryDuration</td>
<td>Integer</td>
<td>Number of months between SCD and SDD</td>
<td>Duration from making the contract to starting using the services</td>
</tr>
<tr>
<td>WaitingPlusDeliveryDuration</td>
<td>Integer</td>
<td>Number of months between SCD and EDD</td>
<td>Duration from making the contract to the moment when the service ended</td>
</tr>
<tr>
<td>ContractInitialType</td>
<td>Integer</td>
<td>If EDD is NULL or (EDD not NULL and ECD before EDD) set ContractDeliveryDuration if it had one of the values: 12, 24 and 36 or 99 if ContractDeliveryDuration &gt; 36; Otherwise set ContractDuration</td>
<td>In months, possible values are 12, 24, 36 and 99</td>
</tr>
<tr>
<td>MovedToVariable</td>
<td>Binary</td>
<td>TRUE if EDD is NULL, but ECD is in the past; TRUE if EDD is not NULL and ECD before EDD; FALSE otherwise</td>
<td></td>
</tr>
<tr>
<td>IsStillACustomer</td>
<td>Binary</td>
<td>TRUE if EDD is NULL; FALSE otherwise</td>
<td></td>
</tr>
<tr>
<td>RealChurnType</td>
<td>Binary</td>
<td>TRUE if EDD is not NULL and EDD less than 30 days before ECD, or any time after</td>
<td>It is not considered to be churn if the contract was terminated too early (see discussion about &quot;happy churn&quot; in Chapter 2)</td>
</tr>
</tbody>
</table>

Table 4.4: Computed attributes

duration that the contract was meant to have when it was first signed. Since the expiration of a contract does not assume the termination of delivery, we have to distinguish
between initial duration, which ContractInitialType attribute represents, and delivery duration, which is represented by the ContractDeliveryDuration attribute. After a contract expires, its type is replaced by Variable. For this type of contract the end date is no longer of interest, the contract can last indefinitely.

Secondly, although the name of the ContractInitialType suggests a contract type, the values represent a duration in months. However, there is a limited number of possible values and each of them represents a type of contract. 12 stands for a 1 year contract, 24 for a 2 years contract, 36 for a 3 years contract and 1 for a variable contract. It was mentioned in Chapter 2 that energy suppliers offer contracts for 1 or 3 years or variable. However, a few years ago, 2 years contracts were offered as well.

Thirdly, the ContractDuration attribute, which also offers the default values for the ContractInitialType attribute, has, in most of the cases the same value. Since there was no way of knowing how the ContractDuration was computed and if the value represented the initial or the actual/delivery duration, a decision was made to compute both durations separately.

The MovedToVariable attribute tells if a contract has expired and was moved to variable. This could have happened for contracts that were already terminated, or for contracts that are still active. The RealChurnType attribute was obtained as explained in the table, from the date attributes and it represents the churn prediction target. As mentioned in the Explanation row, the "happy churns" were not considered to be Churns. This decision was based on the statement made by the domain expert, which made clear that, if the customer leaves before the expiration of the contract, the company is able to apply a fine, though earning money without having to offer any service in return. Therefore, "happy churners" are not of interest for Churn prediction in the specific domain of the Dutch energy market.

It is also worth mentioning that the computation of the ShortPostcode attribute allowed the removal of the nominal Postcode attribute, which had a very high amount of possible values. However, this removal was only performed after the data enrichment phase, which will be presented in detail in the next section. The new ShortPostcode attribute is numeric, but still offers enough details about the geographic region of the address. Furthermore, part of the data enrichment was performed using this field instead of the Postcode one.

Moving on to the next attribute, namely PersonAwayContractsNumber, which represents the number of terminated contracts that the owner of the current contract has, 80% of the people that have terminated contracts are in the Churn class. It can be assumed at this point that the PersonAwayContractsNumber will have a strong influence on the churn prediction. However, this is probably happening only because the attribute also counts the current contract. In other words, the attribute in this form is redundant. In order to remove the redundancy, value 1 was extracted from the value
of PersonAwayContractsNumber for each entry that had a value greater than 0 for this attribute and was also in class Churn.

Besides adding new attributes and replacing a few with newly computed ones, some nominal attributes, such as Source and PersonEmailDomain, suffered an aggregation of possible values. The Source attribute had 77 possible values which represented the channel that had brought the contract, such as a website or specific door to door company level. After consulting the domain expert, a decision was made to group the possible values by higher level, such as door to door, auction, price comparison or internal. The grouping was performed following a sources tree that was made available by the domain expert. For the PersonEmailDomain attribute, which initially had over 14000 possible values, aggregation was also performed, although in a different manner. By analyzing the frequency distribution of the possible values, presented in Figure 4.1, a few cutoff points were chosen and the possible values that had the number of entries between the cutoff points were aggregated. An exception was made for the possible values which had the number of entries above the first cutoff point, 5000, since they were only 7. Therefore, all the possible values that had more than 5000 entries were kept. The other cutoff points were 200 and 20. Consequently, the possible values having more than 200 entries were grouped into one called "OtherPopular", those having between 200 and 20 were grouped under the name of "OtherLessPopular" and the rest, with less than 20 entries, under the name of "OtherNotPopular". Since the number of possible values for the PersonEmailDomain attribute was so high, making the values of the x axis of the graph unreadable, the actual names were removed from Figure 4.1.

![Figure 4.1: Frequency distribution of PersonEmailDomain](image)

The Source and PersonEmailDomain attributes were also renamed, to reflect the aggregation, into AggregatedSource and AggregatedEmailDomain. Figure 4.2 shows the frequency distribution of the possible values of the AggregatedEmailDomain attribute.
4.3 Data Enrichment

As mentioned in Chapter 3, previous studies have found that an important type of data for churn prediction can be the social and economic. As a result a solution for enriching the existing data with social and economic data was searched and found. Statistics Netherlands\(^1\), or CBS, as they mention on their public website, is responsible for collecting and processing data in order to publish statistics. A data set containing socio-economic data about the population of the Netherlands was found: "Postcodegebieden". The data was collected in 2008 and 2010 and, and it was offered per postcode and short postcode. Since the data was in Dutch, the available attributes had to be translated. Secondly, the data came in 2 different data sets, one containing data about population and housing and the second one containing income data. Consequently, the data had to be aggregated by the common attribute, namely Postcode. The last step was to aggregate the new data set to the contracts data set. The matching was also done by Postcode and a few attributes were added. The complete list can be found in Table 4.5. More details about how the data was collected and the specific thresholds for each attribute can be found in the documentation of the original data set, at Statistics Netherlands.

As already mentioned at the beginning of this chapter, after aggregation, cleaning and enrichment of the initial data sets, the final one had 43 attributes and 25433 rows. All these attributes, together with their types and the data category to which they belong can be found in Table 4.6.

\(^1\text{Statistics Netherlands}\)
## Chapter 4. The Data

### Attribute name | Type | Explanation
--- | --- | ---
Urbanity | Integer | Measure, on a scale from 1 to 10, of the level of urbanity of an area
NoOfNonWesterners | Integer | Number of non-western immigrants per postcode; has a value only if the number is higher than 10
AverageHouseholdSize | Integer | Has a value only if the number of people is higher than 10
Population | Integer | Has a value only if the number of people is higher than 10
AverageHouseValue | Integer | PercentageOfIncomeRecipients | Double | What percentage of the population has an income
PercentageOfLowIncomeRecipients | Integer | What percentage of the population has a low income
PercentageOfHighIncomeRecipients | Integer | What percentage of the population has a high income
AverageMonthlyIncomeTax | Integer | PercentageOfIncomeRecipients | Double | What percentage of the population has an income

<table>
<thead>
<tr>
<th>Attribute name</th>
<th>Type</th>
<th>Data category</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>PersonAwayContractsNumber</td>
<td>Integer</td>
<td>CCB</td>
<td>Number of closed contracts that belong to the same customer</td>
</tr>
<tr>
<td>PersonEmailContactsNumber</td>
<td>Integer</td>
<td>CP</td>
<td>Number of email contacts with the Customer Service department for the same customer</td>
</tr>
<tr>
<td>PersonPhoneContactsNumber</td>
<td>Integer</td>
<td>CP</td>
<td>Number of phone contacts with the Customer Service department for the same customer</td>
</tr>
<tr>
<td>ContractAddressesNumber</td>
<td>Integer</td>
<td>CCB</td>
<td>Number of addresses serviced under the same contract</td>
</tr>
<tr>
<td>HadToPayDeposit</td>
<td>Binary</td>
<td>CCB</td>
<td>Customers that are known to have payment issues have to pay a deposit at the start of the contract</td>
</tr>
<tr>
<td>Attribute name</td>
<td>Type</td>
<td>Data category</td>
<td>Explanation</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-----------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ContractCashback</td>
<td>Integer</td>
<td>CCB</td>
<td>The amount of money that the customer should receive when the contract expires</td>
</tr>
<tr>
<td>ContractCurrentBalance</td>
<td>Integer</td>
<td>CCB</td>
<td>The amount of money that the customer had to pay at the moment of data gathering</td>
</tr>
<tr>
<td>ContractLatePaymentsPercentage</td>
<td>Integer</td>
<td>CCB</td>
<td>Percentage of late payments out of all the payments performed by the customer before the moment of data gathering</td>
</tr>
<tr>
<td>ContractTerminationFee</td>
<td>Integer</td>
<td>CCB</td>
<td>Amount paid by the customer as fine for terminating the contract before the agreed end date</td>
</tr>
<tr>
<td>ForcedLeft</td>
<td>Binary</td>
<td>CCB</td>
<td>If the contract was terminated by the company (because the customer did not pay invoices)</td>
</tr>
<tr>
<td>ElectricityIsSingleMeter</td>
<td>Binary</td>
<td>CCB</td>
<td>Electricity meters can be either single or double</td>
</tr>
<tr>
<td>ElectricityIsSmartMeter</td>
<td>Binary</td>
<td>CCB</td>
<td></td>
</tr>
<tr>
<td>ElectricitySJV</td>
<td>Integer</td>
<td>CCB</td>
<td>Yearly consumption of electricity</td>
</tr>
<tr>
<td>ElectricityIsGivingBack</td>
<td>Binary</td>
<td>CCB</td>
<td>If the customer has solar panels which produce some electricity that is returned to the grid</td>
</tr>
<tr>
<td>GasIsLargeConsumer</td>
<td>Binary</td>
<td>CCB</td>
<td></td>
</tr>
<tr>
<td>GasIsSmartMeter</td>
<td>Binary</td>
<td>CCB</td>
<td></td>
</tr>
<tr>
<td>GasRegion</td>
<td>Integer</td>
<td>CCB</td>
<td></td>
</tr>
<tr>
<td>Attribute name</td>
<td>Type</td>
<td>Data category</td>
<td>Explanation</td>
</tr>
<tr>
<td>------------------------</td>
<td>----------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>GasSJV</td>
<td>Integer</td>
<td>CCB</td>
<td>Yearly consumption of gas</td>
</tr>
<tr>
<td>ContractDuration</td>
<td>Integer</td>
<td>CCB</td>
<td>The agreed duration of the contract</td>
</tr>
<tr>
<td>PaidCashback</td>
<td>Integer</td>
<td>CCB</td>
<td>Amount of money already paid to the customer at the end of the contractual period</td>
</tr>
<tr>
<td>Provider</td>
<td>Nominal</td>
<td>CSD</td>
<td>Tightly connected to GasRegion and Postcode</td>
</tr>
<tr>
<td>CustomerAge</td>
<td>Integer</td>
<td>CSD</td>
<td></td>
</tr>
<tr>
<td>ContractDeliveryDuration</td>
<td>Integer</td>
<td>CCB</td>
<td>Duration of service</td>
</tr>
<tr>
<td>ShortPostcode</td>
<td>Integer</td>
<td>CSD</td>
<td>The first 4 digits of the postal or ZIP code (without the last two letters)</td>
</tr>
<tr>
<td>Urbanity</td>
<td>Integer</td>
<td>CSD</td>
<td>Measure, on a scale from 1 to 10, of the level of urbanity of an area</td>
</tr>
<tr>
<td>NoOfNonWesterners</td>
<td>Integer</td>
<td>CSD</td>
<td>Number of non-western immigrants per postal code; has a value only if the number is higher than 10</td>
</tr>
<tr>
<td>AverageHouseholdSize</td>
<td>Integer</td>
<td>CSD</td>
<td></td>
</tr>
<tr>
<td>Population</td>
<td>Integer</td>
<td>CSD</td>
<td>Has a value only if the number of people is higher than 10</td>
</tr>
<tr>
<td>PercentageOfIncomeRecipients</td>
<td>Integer</td>
<td>CSD</td>
<td>What percentage of the population has an income</td>
</tr>
<tr>
<td>PercentageOfLowIncomeRecipients</td>
<td>Integer</td>
<td>CSD</td>
<td>What percentage of the population has a low income</td>
</tr>
</tbody>
</table>
### Chapter 4. The Data

<table>
<thead>
<tr>
<th>Attribute name</th>
<th>Type</th>
<th>Data category</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>PercentageOfHighIncomeRecipients</td>
<td>Integer</td>
<td>CSD</td>
<td>What percentage of the population has a high income</td>
</tr>
<tr>
<td>BeforeDeliveryDuration</td>
<td>Integer</td>
<td>CCB</td>
<td>Duration from making the contract to starting using the services</td>
</tr>
<tr>
<td>WaitingPlusDeliveryDuration</td>
<td>Integer</td>
<td>CCB</td>
<td>Duration from making the contract to the moment when the service ended</td>
</tr>
<tr>
<td>ContractInitialType</td>
<td>Integer</td>
<td>CCB</td>
<td>In months, possible values are 12, 24, 36 and 99</td>
</tr>
<tr>
<td>MovedToVariable</td>
<td>Binary</td>
<td>CCB</td>
<td>If the contract expired and its type was changed to Variable</td>
</tr>
<tr>
<td>IsGenderMale</td>
<td>Binary</td>
<td>CSD</td>
<td></td>
</tr>
<tr>
<td>AggregatedSource</td>
<td>Nominal</td>
<td>CCB</td>
<td>The sales channel used to acquire the customer, aggregated</td>
</tr>
<tr>
<td>AggregatedEmailDomain</td>
<td>Integer</td>
<td>CPD</td>
<td></td>
</tr>
<tr>
<td>RealChurnType</td>
<td>Binary</td>
<td>Prediction Target</td>
<td></td>
</tr>
</tbody>
</table>

**Table 4.6: Attributes of the final data set**

### 4.4 Exploratory Data Analysis

Although the customer raw data provided by an energy supplier counted more than 254000 entries, as explained in the first part of this chapter, cleaning had to be performed on it as well as aggregation and data transformation. At the end, 25433 rows were left, of which 9553 have RealChurnType TRUE, meaning that the entry belongs to the Churn class. Since the Churn class entries represent about 37% of the total, which means that data imbalance will have to be handled before actual modeling.

A good starting point for data exploration are the demographic attributes, such as IsGenderMale. For more than 70% of the entries, this attribute has a TRUE value, which means that the customers are males. 7% of the males are also churners and about 6% of the women. The difference is probably too small to influence prediction on its own.
Figure 4.3: Density of CustomerAge attribute

Figure 4.3 presents the density of the CustomerAge attribute for each Churn class. It can be observed that most of the customers are between 25 and 70 years old. By looking at age groups, a small difference between the rate of churners for older people than for younger ones can be observed, for males, 4.6% versus 6%, as well as females, 4.3% versus 5.5%. This seems to indicate that young people are more likely to churn. However, since the difference is not statistically relevant, CustomerAge could eventually prove to not have a big influence in the final predictive model.

The consumption attributes, GasSJv and ElectricitySJv, are linearly dependent, as it can be observed in Figure 4.4. Furthermore, GasSJv is linearly dependent to one of the socio-economic attributes, namely AverageHouseholdSize. In addition, although not visible in the previously mentioned figure, all the socio-economic attributes exhibit some sort of linear dependency to one another.

The socio-economic attributes keep the general trend, in the sense that they also exhibit outliers and that the distribution of their values in the Churn and non Churn classes follows the proportion of the data set. Plots of densities for 4 of these attributes can be seen in Figure 4.5.

To determine the most important attributes for prediction, the Pearson correlation coefficient was computed for all attributes and the target. The highest coefficient values and the attributes for which they were computed can be observed in Table 4.7.

Although the correlation coefficients are not high, these attributes are the most correlated with the target. Therefore, some statistics will be studied for each of them.
Chapter 4. The Data

Figure 4.4: Dependency of consumption and socio-economic attributes

Figure 4.5: Density for socio-economic attributes
Table 4.7: Correlation of different attributes to the target

<table>
<thead>
<tr>
<th>Attribute name</th>
<th>Pearson correlation coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>MovedToVariable</td>
<td>0.3884176</td>
</tr>
<tr>
<td>PaidCashback</td>
<td>0.3825637</td>
</tr>
<tr>
<td>ContractPaymentsNumber</td>
<td>0.2522944</td>
</tr>
<tr>
<td>ContractCashback</td>
<td>0.1730106</td>
</tr>
<tr>
<td>HasContractDurationVariable</td>
<td>0.1700777</td>
</tr>
<tr>
<td>PersonAwayContractsNumber</td>
<td>0.1384231</td>
</tr>
</tbody>
</table>

In addition, the same exploratory study will be performed for the socio-economic attributes. As visible in Figure 4.6, the highest correlation coefficient was computed for the MovedToVariable attribute. All the contracts that were moved to variable are in the non Churn class and less about 80% of those that did not move to variable are also in the non Churn class. This suggests that only a small portion of the churners wait one more month after the contract expiration date before switching. Furthermore, this implies that renewal offers have to be made before the contract expiration date. The values of the attribute also indicate that 60% of all the contracts were moved to variable. However, since the contracts for which the expiration date is in the future were removed from the data set, this proportion was anticipated. Figure 4.6 shows boxplots for the numeric highly correlated attributes. The easiest thing to observe is that 3 out of 4 attributes have many outlier values. Consequently, the prediction value of the observed outliers should be investigated in the modeling phase.

Figure 4.6: Boxplots for numeric highly correlated attributes
PaidCashback has more than 87% values equal to 0, and the rest are mostly lower than 500. This is expected, as cashback is usually offered only for 3 year contracts and it is around 100 euros. Since this attribute shows which of the customers have already received a cashback it has been compared to ContractCashback, which is the cashback that the customer is or was supposed to receive at contract expiration. The plot in Figure 4.7, which shows more values of 0 for PaidCashback indicates that for some contracts that have cashback, the amount was not paid. This is expected and can mean either that the contract did not expire yet, or that the contract was terminated before the expiration date. By looking at Figure 4.6 one can also see that the highest promised cashback values are in the non churn class.

The PersonAwayContractsNumber attribute has 75% of the values equal to 0, and less than 0.5% of the values are greater than 1. Less than 10% of the values equal to 0 are in the Churn class, while only 0.1% of the values equal to 1 are in the same Churn class. In other words, not many contracts have a relation to already terminated contracts. However, 10% of those that do not have the above mentioned relation are in the Churn class, while only a very low percentage of those have one relation are Churn.

Finally, according to the domain expert’s intuition, the PersonEmailDomain attribute should have exhibited influence on churn prediction. However, the correlation coefficients for the binary attributes that were derived from this attribute and the target are very close to 0. Moreover, one of the theories of the domain expert, which stated
that there would be a relation between churn and a certain domain, "hotmail" is contradicted by the percentage of churners out of all contract entries that have PersonEmail-DomainHotmail true, which is only 7%. The proportion is similar for PersonEmailDomainGmail as well as for PersonEmailDomainOtherPopular.
Chapter 5

Theoretical Fundamentals

According to the CRISP-DM methodology, after having prepared the data, the modeling phase should commence, with feature selection, followed by the creation of some predictive models. Next, once the models are ready, they should be evaluated and compared. However, many of the concepts that have to be used to describe the modeling and evaluation phases might not be familiar to the reader. This chapter is going to first list the machine learning algorithms, feature selection methods and evaluation metrics which were chosen for the purposes of this thesis and offer a short motivation of the solution. Next, the theoretical fundamentals of each of the concepts will be offered. For the readers acquainted with Data Mining research, only the first section of this chapter should be read, while the rest is optional.

5.1 Motivation

In order to select the right techniques for feature selection, prediction and evaluation, the information presented in Chapter 3 was used. This section is going to present the list of chosen techniques and shortly motivate the decisions.

5.1.1 Data Mining Techniques

The selection of the data mining techniques to be used for creating predictive models for the customers of a Dutch energy supplier was made based on the discussion in the second section of Chapter 3, where it was shown that most of the presented studies were using about the same prediction algorithms. The list is the following:

- Logistic Regression
- Decision Trees
- Random Forests
- AdaBoost
• Support Vector Machines
• Neural Networks

5.1.2 Feature Selection Techniques

As referenced in Chapter 3, feature selection is an important step in churn prediction, as reducing the dimensionality of the data set improves the training time as well as the performance of the predictive models. The first part of this section offers a short motivation on performing Principal Component Analysis or PCA for feature selection before modeling. Next, a presentation of PCA and a few details on the implementation choice are given. Finally, an alternative feature selection method is presented, as well as its implementation details.

The authors of [23], a study that has focused on churn prediction, have successfully combined Principal Component Analysis, a popular feature selection method and SVM. In addition to being popular, Principal Component Analysis or PCA has also been proved to outperform other feature selection methods for rotations algorithms [25]. PCA reduces data dimensionality and therefore noise, which the previous section has shown to be able to badly influence some of the chosen data mining algorithms. Moreover it overcomes over fitting and decreases the computation time for SVM by converting the quadratic optimization problem into computation of linear equations[23].

Considering all the above mentioned reasons it has been decided that performing Principal Component Analysis on the data set before modeling would improve the performance of at least some of the predictive models. However, since the result of PCA is a data set of principal components and not attributes, making it hard to determine which attributes mostly contributed to the performance of the model, a feature selection method that was not already validated by other studies, but would keep the most important attributes in the data set was considered. This method is described after PCA.

5.1.3 Evaluation Metrics

Starting from the evaluation metrics used throughout the related studies discussed in Chapter 3, a list of evaluation metrics to be computed and used for assessing the performance of the models that will be described in Chapter 6 was created and it is going to be presented in this section. It was stated in Chapter 3 that the above mentioned studies use different evaluation metrics, making it hard to actually compare the results between each other. Therefore, for the purposes of this thesis a decision was made to compute all the below listed metrics.

• Sensitivity
Chapter 5. *Theoretical Fundamentals*

- Specificity
- Accuracy
- F-measure
- Area Under the Curve
- Confidence Intervals
- Computation Time

## 5.2 Theoretical Fundamentals

This section is going to offer the information about each of the techniques and metrics introduced in the previous section that is crucial for understanding the following chapters. However, if the reader is already familiar with the concepts presented in the previous section, this section can be skipped.

### 5.2.1 Data Mining Techniques

This section offers the theoretical fundamentals of each of the techniques and algorithms that will be used to create the predictive models described in the following chapter.

**Regression**

According to [9], one of the data mining methods that work naturally with numeric attributes is Linear Regression. This method is used when the outcome of the prediction is numeric. The main idea is to express the prediction target or class as a linear combination of the attributes, with weights calculated from the training data. An example is equation 5.1, where $x$ is the target, $a_1$ to $a_n$ are the values of each attribute and $w_0$ to $w_n$ are the weights. When the model is applied to a new instance, the predicted values are obtained. The best model is chosen by identifying the weights that minimize the sum of the differences between predicted and actual values for the majority of training instances. However, if the pattern in the data is not linear, regression can still be used. Furthermore, regression can also be used for classification, either by performing regression on both classes of the target, or by building a linear model based on a transformed target variable[9]. According to the authors of [9], the latter option, also called
Logistic Regression or logit, holds a few advantages over the first one when used for classification.

\[ x = w_0 + w_1a_1 + w_2a_2 + \ldots + x_na_n \]  

(5.1)

As mentioned in Chapter 3, logistic regression was recently used for churn prediction by authors of [35], [22], [1] and [34].

### Decision Trees

Decision Trees (DT) classification is a popular and relatively new data mining technique[44] that involves partitioning the data into predefined classes. When it is following a "divide-and-conquer" approach, this technique is sometimes called "top-down induction of decision trees"[9]. A specific method of this technique, improved over many years, finally resulted into the algorithm called C5.0, which was used for the purposes of this thesis.

As simply stated by the authors of [44], a decision tree is a predictive model that can be viewed as a tree of which each node or branch is a classification question, while the leaves are the partitions of the data set with their classification. A Decision Tree is constructed top-down, from a root node and involves partitioning the data into subsets that contain instances with similar properties. At each node the data set is split into subsets based on the value of the attribute at the node. To choose the attribute to test at each node, the attribute that splits the target into the purest possible children nodes must be identified. In other words, nodes that do not contain a mix of both churners and non churners. This measure of purity is called information and it represents the expected amount of information that would be needed to specify whether a new instance should be classified as churned or non churner, given that the example reached that node [44]. Information is calculated based on the number of churner and non churner classes at the node. A measure of impurity, the opposite of what it was discussed up to this point, is entropy. For a binary target with values \(a\) and \(b\), Entropy is defined as shown in Equation 5.2. \(p(a)\) represents the probability that class \(a\) is chosen. The base of the logarithm is usually 2 [44].

\[ E = -p(a)\log(p(a)) - p(b)\log(p(b)) \]  

(5.2)

The value of the entropy can be computed before and after the split in order to obtain a measure of information gain, or how much information was gained by doing the split for the chosen attribute. It is important to mention that, to compute the entropy after the split, two values of entropy must be computed, for the right and for the left node. These two values are then combined using a weight factor equal to the number
of instances that went down on each branch. Finally, by performing the computation of information gain at each new node for every remaining attribute, the one that provides most information gain can be chosen.

Decision trees can be built for any type of attributes due to the fact that each node tests an attribute against a constant value. However, the number of the splitting branches depends on the type of the attribute: for a nominal one, there is usually a branch for each possible value; for a numeric attribute, the test at a node usually determines if a value is greater than a predetermined constant\cite{9}. However, missing values pose a problem for decision trees, as a test on a node cannot determine on which branch the instance should go down to reach a leaf.

It is important to mention that a decision tree can also be used to predict numerical values. In this case, the leaves are numbers representing the average outcome of all the instances that reached the leaf\cite{9}. This type of tree is called a regression tree. An important advantage of a Decision Trees model is that it can be read and used by a person to extract some rules that can be directly applied, without using the actual prediction model. Moreover, decision trees are able to handle nominal attributes, so not much pre-processing of the data is needed. However, an important disadvantage is their lack of robustness and suboptimal performance \cite{45}.

**Random Forests**

Random Forests (RF) is an ensemble learning method based on bagging and decision trees and introduced by Breiman in 2001. A simple definition for ensemble learning is offered by the authors of \cite{46}, a set of methods that generate a multitude of classifiers and aggregate their results. According to the same authors, one of the most common of the ensemble learning methods is bagging, which creates classifiers that are independent from the previous ones using a bootstrap sample of the training data set and finally taking a majority vote for prediction \cite{46}.

Random Forests, as proposed by Breiman in \cite{47}, is a bagging algorithm that generates decision trees. In addition to creating independent trees by using a bootstrap sample of the training data set, Random forests also change the way the trees are created: each node is split using the best of a set of randomly chosen predictors\cite{46}. Being an optimization of the Decision Trees algorithm, Random Forests exhibits all the advantages of the base technique, plus the one of being robust against overfitting \cite{45}. Moreover, it performs well when compared to other very popular classification algorithms like Support Vector Machines or Neural networks \cite{46}. However, a disadvantage of the technique is the fact that it has low performance for imbalanced training data sets \cite{17}.
AdaBoost

Boosting is a machine learning technique that combines the performance of a set of classifiers in order to obtain better classification performance \[48\]. According to the authors of \[48\], the most popular version of boosting is AdaBoost (ADA), introduced by \[49\] in 1997. The same authors explain that the original boosting techniques were able to improve performance of a classifier by producing a "majority vote" of similar classifiers. Further, the algorithms evolved into adaptive versions, in the sense of sequentially tweaking "weak" classifiers in favor of those instances miss-classified by previous classifiers. AdaBoost, or Adaptive Boosting is one of them.

As described in \[48\], using a set of weighted versions of the training sample, AdaBoost trains the "weak" classifiers, giving higher weights to the currently miss-classified cases. This training is performed sequentially on weighted samples and the final classifier is a linear combination of the classifier for each stage. In more detail, the pseudo code for the AdaBoost algorithm is the one from Figure 5.1 \[50\].

\[
\begin{align*}
\text{Given: } & (x_1, y_1), \ldots, (x_m, y_m) \text{ where } x_i \in X, y_i \in \{-1, +1\}. \\
\text{Initialize: } & D_1(i) = 1/m \text{ for } i = 1, \ldots, m. \\
\text{For } & t = 1, \ldots, T: \\
& \quad \text{Train weak learner using distribution } D_t. \\
& \quad \text{Get weak hypothesis } h_t : X \rightarrow \{-1, +1\}. \\
& \quad \text{Aim: select } h_t \text{ with low weighted error:} \\
& \quad \quad \quad \epsilon_t = \Pr_{i \sim D_t} [h_t(x_i) \neq y_i]. \\
& \quad \quad \text{Choose } \\
& \quad \quad \quad \alpha_t = \frac{1}{2} \ln \left( \frac{1 - \epsilon_t}{\epsilon_t} \right). \\
& \quad \text{Update, for } i = 1, \ldots, m: \\
& \quad \quad D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t} \\
& \quad \quad \text{where } Z_t \text{ is a normalization factor (chosen so that } D_{t+1} \text{ will be a distribution).} \\
\text{Output the final hypothesis:} \\
& \quad H(x) = \text{sign} \left( \sum_{t=1}^T \alpha_t h_t(x) \right).
\end{align*}
\]

\textbf{Figure 5.1: Pseudo code for AdaBoost algorithm}

Weighted re-sampling can be replaced by a weighted tree-growing algorithm and the performance is comparable.

According to Freund, the creator of the algorithm, AdaBoost has become very popular due to its simplicity and adaptivity \[51\], while also exhibiting an important disadvantage: susceptibility to noise.
Support Vector Machines

Support Vector Machine (SVM) is a relatively recent machine learning technique based on two key ideas, the maximum margin solution for a linear classifier and the "kernel trick", which is a method of expanding from a linear classifier to a non-linear one efficiently[52]. The margin is defined by equally spaced boundaries on each side of the line that separates the data into the two classes of the target. SVM is looking for the optimal separating hyperplane between the two classes by maximizing the margin [53]. The points that are lying on the boundaries are called support vectors. Figure 5.2 [53] illustrates two classes separated by a hyperplane, the concept of margin and the support vectors.

When the classes are not linearly separable, the data points are projected into a higher-dimensional space where they become linearly separable, via kernel techniques [53]. For details on the kernel techniques, or "kernel trick", the reader can refer to [52].

An SVM is an algorithm that is able to apply kernel techniques to solve non-linearity and then separate the two classes by maximizing the margin, while also penalizing data points that are on the wrong side of the linear separator[53]. The most important disadvantage of the Support Vector Machine technique is that the algorithm scales badly with data size[53]. Moreover, many experiments are needed in order to determine the optimal kernel parameters. However, the wide use of the technique and its good results suggest that using Support Vector Machines could result into high performance
predictive models for churn.

Neural Networks

A clarification on the terminology is needed before explaining what a neural network is and how it works: the correct name for what neural network refers to throughout this thesis is Artificial Neural Network. Artificial neural networks are inspired from biological neural networks, or brains, which have the capability of learning and derive their name from the idea that, if a machine mimics the structure of the brain, it could “think” [44].

Neural Networks (NN) is a machine learning algorithm used for prediction in a wide range of domains. Its most important advantage is that it can be used to create highly accurate predictive models [44]. Among the disadvantages of this algorithm lies the fact that the data needs to be pre-processed before being fed to the algorithm, as, when using neural networks with nominal attributes, for example, the complexity is very high. Therefore, the nominal attributes have to be transformed before the modeling phase. Moreover, the models created with this algorithm are almost always impossible to understand [44].

A neural network consists of nodes, which correspond to neurons and links, which are connecting the nodes. The values of each predictor attribute are inputs for the neural network, so the number of input nodes is usually equal to the number of attributes. The links have weights assigned, which are added together in the output node. For classification a threshold value is needed to transform the numeric output into a class. The NN model is created by training it with many examples and the class of each. By comparing the computed answer to the correct one, the model is adjusted by changing the weights [44]. Besides the input nodes and the output node, a neural network can also contain one or more layers of hidden nodes. The values of the hidden nodes are not visible to the end user and are determined by the neural network as it trains [44]. Figure 5.3 illustrates a simple neural network with 4 inputs and 5 neurons in the hidden layer.

Although there are many types of neural networks, the most common one among the studies presented as related work is the back-propagation feed-forward neural network. "Back-propagation" refers to the propagation of the error from the output nodes through the hidden node and to the input nodes, allowing the adjustment of weights [44]. The error is computed by comparing the obtained output to the real output and calculating mean-squared error signal. The error value is then propagated backwards through the network, and small changes are made to the weights in each layer. The weight changes are calculated to reduce the error signal for the current example, using gradient descent. The same steps are repeated for each example, then back to the first
one, until the overall error value drops below some pre-determined threshold. At this point it is considered that the network has learned the problem.

To offer some more details on how the weights are adjusted, after the error for the output layer was obtained, an error for each neuron in the hidden layers, going backwards, layer by layer has to be computed. The error for a neuron in a hidden layer is the sum of the products between the errors of the neurons in the next layer and the weights of the connections to those neurons, multiplied by the derivative of the activation function, as shown in Equation 5.3, where \( i \) is the current node and \( j \) is the neuron in the next layer. The errors of the hidden nodes will be used to calculate the variation of the weights as a result of the current input pattern and ideal outputs. As shown in Equation 5.4, the variation (delta) of a weight is the product between the output value of the input neuron and the error of the output neuron for that connection. This process is repeated for all input patterns and the variations (deltas) are accumulated. At the end of a learning iteration the actual weights are replaced by the accumulated deltas for all the training patterns multiplied with a learning rate. The learning rate is a number typically between 0 and 1 which states how fast a network converges to a result.

\[
\text{HiddenError}_i = \sum (\text{OutputError}_j \times w_{ij}) \times \frac{\delta F}{\delta \text{HiddenOutput}_i} \quad (5.3)
\]

\[
\delta w_{ij} = \text{Output}_i \times \text{Error}_j \quad (5.4)
\]
5.2.2 Feature Selection

This section is going to offer the theoretical fundamentals of the two feature selection methods that were chosen and will be used in the modeling phase.

Principal Component Analysis

Principal Component Analysis (PCA) is a statistical technique used for reducing data dimensionality by analyzing a data table representing observations described by several dependent variables, extracting important information and finally express this information as a set of new orthogonal variables called principal components [54]. In a simplified manner, presented by the authors of [54], the goals of PCA are the following:

- extract the most important information from the data table
- compress the size of the data set by keeping only this important information
- simplify the description of the data set
- analyze the structure of the observations and the variables

In order to understand how PCA works, a brief introduction of a few key concepts is required. The explanations were extracted from [55]. The first important concept for PCA is variance, which, just as standard deviation, is a measure of the spread of the data in a data set. The formula of the variance of the values of an attribute, \( X \), is shown in Equation 5.5, where \( \bar{X} \) is the mean of the same values.

\[
s^2 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{n-1}
\]  

(5.5)

Furthermore, to measure how much the dimensions vary from the mean with respect to each other, we can use covariance. Its formula is in Equation 5.6. In order to perform PCA, the covariance matrix of the data set will have to be created. This means computing the covariance of all the attributes in the data set, 2 by 2 and putting them in a matrix.

\[
cov(X, Y) = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{n-1}
\]  

(5.6)

The last two things that should be explained before describing how PCA actually works are eigen vectors and eigen values. According to [55], given a transformation matrix, which is a square matrix, if it is multiplied on the left of a vector, the result is another vector that is a transformation of the first one from its original position. If the matrix transforms the vector without changing the direction of the vector, then the vector is an eigen vector of the transformation matrix. In other words, if \( v \) is a vector that is not
zero, then it is an eigen vector of a square matrix $A$ if $Av$ is a scalar multiple of $v$. The eigen value associated with $v$ is the scalar number $\lambda$ from Equation 5.7.

$$Av = \lambda v$$  \hspace{1cm} (5.7)

PCA computes principal components as linear combinations of the original variable. The first principal component is required to have the largest possible variance, while the next ones are required to be orthogonal to the previous one and having the highest possible variance\[54\]. The steps performed by PCA are the following:

- for each attribute, subtract the mean of the values from each value; at the end, the data set has mean equal to 0
- calculate covariance matrix of the data set
- calculate the eigen vectors and eigen values of the covariance matrix
- choose the principal components by ordering by eigen values, descending
- derive the new data set from the chosen eigen vectors by taking the transpose of each vector and multiplying it on the left of the original data set, transposed.

The data set obtained in the last step of the above described algorithm was further used to create part of the final predictive models.

**Alternative Feature Selection Method**

During the exploratory data analysis, it was observed that an important number of attributes were strongly correlated to others. An algorithm that allowed the redundant attributes to be removed was developed, based on Pearson’s correlation coefficient computed for the target and each attribute, as well as for every 2 attributes. The formula for Pearson’s correlation coefficient is the one from Equation 5.8, where $\sigma$ represents the standard deviation of the values of an attribute.

$$p(X, Y) = \frac{cov(X, Y)}{\sigma_X \sigma_Y}$$  \hspace{1cm} (5.8)

The algorithm consists of the following steps, applied recursively on the data set:

- Compute correlations of each attribute to the target
- Get the attribute that has the highest correlation coefficient
- Compute correlation coefficients for the chosen attribute and all the others, excluding the target
• Remove all attributes that have a correlation coefficient to the chosen attribute higher than a previously set threshold

The expectation was that the removal of redundant attributes would, at least, decrease the computation time of the training phase for each predictive model.

5.2.3 Evaluation Metrics

This section offers the theoretical fundamentals of each of the evaluation metrics introduced in the first part of this chapter.

Typical Evaluation Approach

According to the authors of [14], the quality of the output of a classifier or prediction model is measured in terms of specificity, sensitivity and accuracy. However, each of these metrics are computed starting from the confusion matrix of the predicted output. A confusion matrix is a way of separating the output into True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN), as shown in Table 5.1.

<table>
<thead>
<tr>
<th>Predicted Churners</th>
<th>Actual Churners</th>
<th>Actual Non-Churners</th>
</tr>
</thead>
<tbody>
<tr>
<td>True Positives</td>
<td>False Positives</td>
<td></td>
</tr>
<tr>
<td>False Negatives</td>
<td>True Negatives</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.1: Confusion matrix for a churn predictive model

Moving a step forward, the Predicted Positives (PP), Predicted Negatives (PN), Real Positives (RP) and Real Negatives (RN) can be computed, as described by Equations 5.9 to 5.12. The following subsections will show how these values are used to compute evaluation metrics for predictive models.

\[
PP = TP + FP \tag{5.9}
\]

\[
PN = TN + FN \tag{5.10}
\]

\[
RP = TP + FN \tag{5.11}
\]

\[
RN = TN + FP \tag{5.12}
\]
Sensitivity

The sensitivity of a predictive model is the probability of correctly predicting a positive example and it is computed using the formula presented by Equation 5.13 [56]. This quality measure is also called recall.

\[ \text{sensitivity} = \frac{TP}{TP + FN} \]  
(5.13)

Specificity

As mentioned in the previous subsection, throughout this thesis, the term "specificity" will be used as explained in [56], to refer to the probability that a positive prediction is correct. The formula is the one from Equation 5.14. Specificity is also called precision.

\[ \text{specificity} = \frac{TP}{TP + FP} \]  
(5.14)

Accuracy

The accuracy of a predictive model is defined as the percentage of correct predictions and can be computed as shown in Equation 5.15 [14].

\[ \text{accuracy} = \text{sensitivity} \times \frac{PP}{PP + PN} + \text{specificity} \times \frac{PN}{PP + PN} \]  
(5.15)

F-measure

The F-measure, or F-score is another measure of the accuracy of a classifier and it is computed using the formula from Equation 5.16[15].

\[ F_{\text{measure}} = \frac{2 \times \text{specificity} \times \text{sensitivity}}{\text{specificity} + \text{sensitivity}} \]  
(5.16)

The general formula is the one from Equation 5.17, and the F-measure is the particularization for \( \beta = 1 \).

\[ F_{\beta} = (1 + \beta^2) \times \frac{\text{specificity} \times \text{sensitivity}}{\beta^2 \times \text{specificity} + \text{sensitivity}} \]  
(5.17)

Area Under the Curve

Another evaluation metric that was employed by many of the authors of the related studies presented in Chapter 3 is Area Under the Curve (AUC), computed for a Receiver Operating Characteristic (ROC) curve. To obtain a ROC curve, one must first
Chapter 5. Theoretical Fundamentals

construct the confusion matrix from the predicted outputs, as explained at the beginning of this section. Next, the True Positive Rate (TPR) or sensitivity has to be computed, together with False Positive Rate (FPR) or (1-specificity). These metrics have to be computed with many different thresholds for prediction and plotted together. The curve that is obtained is the ROC curve and it is used to summarize the trade-off between the amount of FP and the amount of FN produced by the model [56].

The main advantage of AUC, according to the authors of [2], is that, unlike accuracy, AUC takes into account the individual class performance. In other words, in case of imbalanced data, a model that predicts always the most common class, while neglecting the minority class, will evaluate as having high accuracy, but not high AUC[2].

Confidence Intervals

According to [57], a Confidence Interval (CI) is as a range of plausible values for a population parameter, calculated from sample data. A CI with a 95 percent confidence level has a 95 percent chance of capturing the real value of the population parameter. The center of the CI (the sample mean) is the most plausible value for the population parameter, while the ends of the CI are less plausible values for the population parameter [57]. Therefore, computing AUC on a sample could give a value that is not very close to the real AUC value, while computing a CI for AUC will offer 95% confidence that AUC is in that interval, very close to the median bound. A $100(1 - \alpha)$ Confidence Interval for AUC can be computed using the formula depicted in Equation 5.18, where [58]:

- $SE(AUC)$ is the standard error of AUC, computed using the formula from Equation 5.20; $Q_1$ and $Q_2$ can be computed as shown in Equation 5.21 and $N_1$ and $N_2$ represent the number of examples in the samples used
- $z$, the SE statistic, follows from the cumulative normal distribution function, as described in Equation 5.22

Furthermore, the width of the confidence interval is the one from Equation 5.19[58].

$$CI_{AUC} = AUC \pm z_{\frac{\alpha}{2}} SE(AUC) \quad (5.18)$$

$$2z_{\frac{\alpha}{2}} SE(AUC) \quad (5.19)$$

$$SE(AUC) = \sqrt{\frac{AUC(1 - AUC) + (N_1 - 1)(Q_1 - AUC^2) + (N - 2 - 1)(Q_2 - AUC^2)}{N_1 - N_2}} \quad (5.20)$$
\[ Q_1 = \frac{AUC}{2 - AUC} \]
\[ Q_2 = \frac{2AUC^2}{1 + AUC} \]  
(5.21)

\[ \Phi(z) = 1 - \frac{\alpha}{2} \]
\[ z = \Phi^{-1}(\Phi(z)) \]  
(5.22)

**Computation Time**

For the purposes of this thesis, the computation time of every model is only measured for the training step. Therefore, while this measure of quality is not relevant for the actual prediction, it might be useful to know how long the training takes for each model, especially if the model will be re-built after some time, to adapt it to new data, as advised by the authors of [22]. The computation time also offers a good indication of the actual worth of the performance improvement offered by a feature selection technique, for example. In other words, if applying PCA, for example, improves accuracy or AUC with 0.05, while increasing the computation time, it could be decided in favor of not using PCA.

All the quality or evaluation metrics described in this section, except for the computation time, have values between 0 and 1, with 1 being the best. The computation time is measured in seconds.
Chapter 6

Modeling

Following the CRIPS-DM methodology, when the data was prepared, the modeling phase started. A number of machine learning algorithms was chosen to create predictive models that would be able to predict if a customer of a Dutch energy supplier is likely to churn or not. This chapter is going to present the modeling setup and implementation details for the created models.

6.1 Modeling Setup

After choosing the prediction algorithms, the actual modeling had to be performed. However, a few issues that were identified in Chapter 4 had to be addressed first, using solutions that were already employed by other researchers in their studies. The most important issue was that of imbalanced data. As explained in the last subsection of Chapter 4, Churn class entries represent about 37% of the total. As demonstrated in Chapter 3, a good solution for this problem is under-sampling. This was performed using native R methods, which allowed creating a subset of the final data set, containing all the entries in Churn class and the same number of entries from non Churn class, chosen randomly. Another issue found during Exploratory analysis, was that of outliers. However, since other researchers have not paid attention to this aspect in their studies on churn prediction, the removal of outliers was implemented as an optional step, meaning that some models use it, while others do not. Moreover, since removing outliers for all the attributes has proven to reduce the data set drastically, a decision was made to only perform it for the 10 most important attributes, where importance is measured by the Pearson correlation coefficient between the attribute and the target. In order to remove outliers for each of the important attributes, the quantiles for 0.25 and 0.75 were calculated and used to compute the minimum and maximum value for each attribute. Everything that is higher than maximum or lower than minimum is removed. According to the authors of [59], the boxplot graph, introduced by Tukey in 1977, uses the same approach, called Inter Quartile Range (IQR) for separating outliers.
Having in mind the fact that churn prediction for the Dutch energy market had not been studied before the moment when this thesis debuted and, therefore, the results of other studies on churn prediction could only be considered for offering possible solutions, the two feature selection methods were also made optional. The purpose of this was to evaluate models built with and without each of them and to be able to establish if they bring any improvement in prediction performance in this particular case.

Another important thing to mention at this point is that, since one of the prediction algorithms to be evaluated was Neural Networks, for which handle nominal attributes introduce very high complexity, a special data set had to be obtained for training Neural Networks models. For this particular case, all the nominal attributes from what was referred to as the final data set had to be transformed to binary or numeric ones, resulting in a final data set of 93 attributes. The most important ones were the consumption attributes: GasSJV and ElectricitySJV, as well as AggregatedSource and AggregatedEmailDomain. For consumption attributes, intervals were created by looking at the distribution of values. Then a binary attribute was created for each interval. For the aggregated attributes, a binary attribute was created for each possible value.

The last issue that had an influence of the final modeling setup was the existence of a set of attributes that were identified as directly related to the target. These attributes are:

- MovedToVariable
- ContractPaymentsNumber
- ContractDeliveryDuration
- ContractStartToEndDeliveryDuration
- ContractTerminationFee
- ForcedLeft
- CustomerDuration

The first 5 attributes of the list above are a direct indication of the fact that the customer has churned, especially since their values are computed after the churn has already taken place. When a contract expires, it is automatically switched to variable. And, since most of the customers leave right after the expiration of the contract, MovedToVariable and Churn are strongly related. Similarly, the number of payments, represented in ContractPaymentsNumber and the duration of delivery, represented in 2 different attributes, are usually an indication of Churn, since the payments are made monthly and contracts have a fixed duration of 12, 24 or 36 months. So, whenever ContractPaymentsNumber, ContractDeliveryDuration or ContractStartToEndDeliveryDuration have a value of 12, 24 or 36, it means that the customer has churned. On the
other hand, until the very last month of contract, these attributes will indicate a value that will push prediction towards non churn, just to switch to churn when the contract has expired. ContractTerminationFee only has a value if the customer churned, as well as ForcedLeft. CustomerDuration always indicates variable for entries that are in class Churn.

Assuming that these attributes will cause the predictive models to change prediction based on the moment when it is used, so predicting non Churn up to the last month and Churn after the contract expiration, a decision was made to evaluate models created with and without them.

Taking into consideration all the above mentioned issues, 4 data sets were created from the so called final data set:

1. With all attributes and nominal attributes allowed – 42 attributes
2. Without attributes directly related to the target class and nominal attributes allowed – 35 attributes
3. With all attributes, no nominal attributes allowed – 93 attributes
4. Without attributes directly related to the target class, no nominal attributes allowed – 79 attributes

The last 2 were used for training the Neural Networks models, while the first two were used for the rest of the models.

Moving forward, by also varying the removal of outliers, the use of PCA and the use of the alternative feature selection method, or removal of correlated attributes, 42 data sets were created. It is important to mention that, when PCA was not applied, the data was normalized. Finally, the 42 data sets were used to train and validate predictive models, resulting into a number of 96 models to be evaluated. In order to easily differentiate among them in the Evaluation section, a naming convention was used. Each name of a model starts with the short name of the algorithm that it is based on followed by a 5 bits sequence, built using the following rules:

1. The first digit is 1 if the data set contained the above mentioned attributes that are directly correlated to the target and 0 otherwise
2. The second digit is 1 if the correlated attributes were removed from the data set and 0 otherwise
3. The third digit is 1 if the outliers were removed from the data set and 0 otherwise
4. The fourth digit is 1 if the extra processing for NN was performed on data set and 0 otherwise - always 1 for NN and 0 for the rest of the models
5. The fifth digit is 1 if PCA is used and 0 otherwise
To clarify, the model identified as ADA 00000, for example, is a model based on the AdaBoost algorithm, fit with training data from the 00000 data set. The 00000 data set has no attributes directly correlated to the target, no correlated attributes were removed, no outliers were removed, no extra processing was performed on it and PCA was not applied.

For all the above mentioned models, validation was performed by splitting the chosen data set into a training data set, containing 70% of the entries, and a validation data set, containing the rest of 30% of the entries.

For evaluation and comparison, all the evaluation metrics presented in Chapter 5 were computed for each model.

### 6.2 Implementation Details

All the modeling and evaluation steps were performed using R. More details on what packages or native functions were used for creating the models, performing feature selection and computing the evaluation metrics will be offered in this section.

#### 6.2.1 Models

It is important to mention that, in order to select the right parameters for each of the algorithms, experiments were performed on the 00000 data set. For each algorithm, the parameters that offered the highest accuracy on this model were used to create all the models of the same type.

For the first type of models, those based on logistic regression, the R implementation of Generalized Linear Model (GLM) was used. The "glm" method \(^1\) of the stats package \(^2\) was called with the "family" parameter set to "binomial(link='logit')", which translates into the algorithm using logistic regression. By setting the "family" parameter to the value mentioned above the trained model is created based on the logit algorithm. Moving forward, as already mentioned in the previous chapter, the Decision Trees models were created using an implementation of the C5.0 algorithm. Therefore, the "C5.0" method of the "C50" R package \(^3\) was employed. No special parameters were used and no tuning was performed.

Next, for the models based on Random Forests, an R implementation which is included in the "randomForest" R package \(^4\) was chosen. The "randomForest" method was called

---

\(^1\) glm R package  
\(^2\) stats R package  
\(^3\) c50 R package  
\(^4\) randomForest R package
with the default parameters: classification type, 500 trees and 4 variables tried at each split.

The AdaBoost models were built using an R implementation of the generalized version of the algorithm, proposed by [48]. This implementation is included in the "ada" R package:\footnote{5} The best results were obtained when the "ada" method was called with the number of iterations set to 50 and the "control" parameter set to and "rpart.control" object, with the following parameters: max depth, which controls the depth of each tree set to 30, minimum number of observations per split set to 20 and complexity factor of trees set to 0.01.

The SVM predictive models were created using an R implementation of the algorithm, from the "kernlab" package:\footnote{6} The name of the method is "ksvm" and it uses "C-classification". Highest performance for the models created with this algorithms was obtained when setting the "kernel" parameter to "polydot" and the cost of constraints violation, "C", to 5.

Finally, the Neural Networks algorithm used for the purposes of this thesis was an R implementation of the fast-forward back-propagation neural network, part of the "nnet" R package:\footnote{7} The number of hidden layers for the "nnet" method was 1, while the number of hidden nodes was 10 and the maximum number of iterations, 200.

### 6.2.2 Feature Selection

For the purposes of this thesis, Principal Component Analysis was performed using an R implementation of the algorithm described in Chapter 5, called "prcomp". After the principal components were computed, those that covered 80% of the covariance were saved as the new data set. For the alternative feature selection method, the algorithm presented in Chapter 5 was implemented in R. The correlation coefficients were computed using the "cor" method of the "stats" R package:\footnote{8} The threshold for the correlation coefficients was set to 0.5.

### 6.2.3 Evaluation Metrics

AUC and CI for AUC were computed using an R package, pROC:\footnote{9} The "auc" and "ci" methods were used in cascade to obtain the values, while the "roc" method was used to obtain objects which could be plotted to create the graph in Figure 7.2. For the purpose of this thesis, the level of confidence, \((1 - \alpha)\) was chosen to be 0.95. Therefore,
95% confidence interval was computed for AUC and expressed as lower, median and upper bound.
Chapter 7

Evaluation

In order to be able to answer the remaining of the research questions, the models described in the previous chapter were evaluated and compared. In this chapter a comparison of the most relevant models will be performed, with an emphasis on how each of the optional techniques presented in Chapter 5 have influenced the performance of the models.

7.1 General considerations

As mentioned in Chapter 6, 96 predictive models were created for predicting churn. The purpose was to investigate which of the most popular data mining or machine learning algorithms are more suitable for predicting churn for the customers of a young energy supplier active on the Dutch energy market. Additionally, an evaluation of performance improvements offered by feature selection and other data mining techniques used by researchers in the churn prediction field for different domains was targeted. Therefore, in order to compare the predictive models, different popular evaluation metrics were computed for each of the models. This section offers a comparison between the different models, trying to find the best one, while also emphasizing the usefulness of each data mining technique that was proven to improve performance for churn prediction in other domains, for the particular domain of the Dutch energy market.

However, before the actual comparison, it has been observed that the models for which the attributes directly correlated to the target were not removed, all the evaluation metrics, except for the computation time, had extremely high values (above 0.98). Additionally, it has been observed that the most used attributes for these models were the directly correlated attributes. Since, as explained in the previous chapter, this result was expected, those specific models were not further evaluated.

As stated in the previous section, a very good evaluation metric for imbalanced data would be AUC, while accuracy might not indicate that the model is biased towards one of the classes. Moreover, this evaluation metric seems to be one of the most used measures for prediction performance, even when data imbalance is solved by sampling
techniques. However, since no cross validation is performed for the predictive models and, as a consequence, AUC is only computed for one sample, the mean bound of the CI of AUC should be preferred. Nevertheless, the R method that was used to compute AUC actually returns the mean bound CI of AUC. In conclusion, the AUC value computed for the models will be the the most used metric while comparing them.

### 7.2 Removal of correlated attributes

Table 7.1 shows the value of AUC and computation time for the models created with each of the proposed algorithms, with and without correlated attributes. It can be observed that the performance of each model is slightly or not at all improved. An improvement can be noticed for the computation time, up to 19% for Neural Networks. However, although the purpose of the removal of correlated attributes was to eliminate redundancy, at the cost of losing some information, therefore a negative influence on the performance of the models being expected, the performance was only slightly reduced. Moreover, the model based on the Support Vector Machines algorithm registered a slight improvement. The second expectation, regarding the positive influence towards the computation time, was met. This indicates that removing the correlated attributes brings an overall slight improvement of the predictive models.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Parameters</th>
<th>AUC (CI)</th>
<th>Computation time</th>
</tr>
</thead>
<tbody>
<tr>
<td>C50</td>
<td>00000</td>
<td>0.89 (0.888-0.904)</td>
<td>2.30</td>
</tr>
<tr>
<td>C50</td>
<td>01000</td>
<td>0.89 (0.884-0.899)</td>
<td>2.02</td>
</tr>
<tr>
<td>RF</td>
<td>00000</td>
<td>0.90 (0.889-0.904)</td>
<td>35.01</td>
</tr>
<tr>
<td>RF</td>
<td>01000</td>
<td>0.89 (0.886-0.902)</td>
<td>36.25</td>
</tr>
<tr>
<td>SVM</td>
<td>00000</td>
<td>0.70 (0.690-0.714)</td>
<td>2893.80</td>
</tr>
<tr>
<td>SVM</td>
<td>01000</td>
<td>0.73 (0.721-0.743)</td>
<td>2175.00</td>
</tr>
<tr>
<td>ADA</td>
<td>00000</td>
<td>0.89 (0.885-0.901)</td>
<td>39.04</td>
</tr>
<tr>
<td>ADA</td>
<td>01000</td>
<td>0.89 (0.882-0.897)</td>
<td>34.59</td>
</tr>
<tr>
<td>GLM</td>
<td>00000</td>
<td>0.87 (0.859-0.876)</td>
<td>1.56</td>
</tr>
<tr>
<td>GLM</td>
<td>01000</td>
<td>0.86 (0.851-0.869)</td>
<td>1.60</td>
</tr>
<tr>
<td>NN</td>
<td>00010</td>
<td>0.80 (0.798-0.817)</td>
<td>37.01</td>
</tr>
<tr>
<td>NN</td>
<td>01010</td>
<td>0.80 (0.795-0.815)</td>
<td>25.18</td>
</tr>
</tbody>
</table>

**Table 7.1**: AUC and Computation time for each algorithm, varying the removal of correlated attributes parameter

### 7.3 Removal of outliers

The removal of outliers, as shown in Table 7.2, has displayed even worse performance improvement than the removal of correlated attributes. No improvement for AUC and
a slight increase of computation time. Since none of the cited related studies had mentioned outliers removal as part of the data mining approach, it was somehow expected that this practice will not bring any value to the churn prediction. However, the confirmation of this supposition could be valuable to other researchers.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Parameters</th>
<th>AUC (CI)</th>
<th>Computation time</th>
</tr>
</thead>
<tbody>
<tr>
<td>C50</td>
<td>00000</td>
<td>0.89 (0.888-0.904)</td>
<td>2.30</td>
</tr>
<tr>
<td>C50</td>
<td>00100</td>
<td>0.89 (0.889-0.904)</td>
<td>2.16</td>
</tr>
<tr>
<td>RF</td>
<td>00000</td>
<td>0.90 (0.889-0.904)</td>
<td>35.01</td>
</tr>
<tr>
<td>RF</td>
<td>00100</td>
<td>0.90 (0.888-0.904)</td>
<td>36.00</td>
</tr>
<tr>
<td>SVM</td>
<td>00000</td>
<td>0.70 (0.690-0.714)</td>
<td>2893.80</td>
</tr>
<tr>
<td>SVM</td>
<td>00100</td>
<td>0.70 (0.691-0.714)</td>
<td>2952.00</td>
</tr>
<tr>
<td>ADA</td>
<td>00000</td>
<td>0.89 (0.885-0.901)</td>
<td>39.04</td>
</tr>
<tr>
<td>ADA</td>
<td>00100</td>
<td>0.89 (0.887-0.903)</td>
<td>39.00</td>
</tr>
<tr>
<td>GLM</td>
<td>00000</td>
<td>0.87 (0.859-0.876)</td>
<td>1.56</td>
</tr>
<tr>
<td>GLM</td>
<td>00100</td>
<td>0.87 (0.859-0.877)</td>
<td>1.70</td>
</tr>
<tr>
<td>NN</td>
<td>00010</td>
<td>0.80 (0.798-0.817)</td>
<td>37.01</td>
</tr>
<tr>
<td>NN</td>
<td>00110</td>
<td>0.80 (0.798-0.817)</td>
<td>31.00</td>
</tr>
</tbody>
</table>

Table 7.2: AUC and Computation time for each algorithm, varying the removal of outliers parameter

7.4 Principal Components Analysis

Based on the related research studies cited in Chapter 3, the use of Principal Components Analysis was expected to improve the performance of the predictive models. The values displayed in Table 7.3 indicate that these expectations were confirmed. It is easy to observe by looking at the values for AUC that great improvement was achieved by applying this feature selection technique. AUC exhibits increase from 10% for the model based on ADA, to 30% for the model based on SVM. Moreover, the computation time was remarkably decreased, from 62% for the ADA, 80% for the GLM model and up to 99.5% for the model based on SVM. For the SVM model, the computation time has decreased from 48 minutes to under a minute. Considering that PCA is used to reduce the complexity of the data and eliminate the redundancy, improvement in both performance and speed were expected. However, the rates of improvement have overcome the expectations in this case.

7.5 PCA and Removal of correlated attributes

After assessing the results from the previous subsections, the models for which the correlated attributes were removed and PCA was applied before modeling were expected
to exhibit even better results than those where only one of the methods was applied. However, when both methods are applied, the results are worse than those of the models where only PCA was applied. The negative influence of the combination of the two methods can be seen in Table 7.4. Both evaluation metrics presented in the table display lower values for the models built with both methods, than for those created only with PCA. The negative influence of the combined methods is, however not very strong, but noticeable. The explanation could be that some significant information is lost when removing the correlated attributes. In conclusion, applying only PCA as feature selection method is better than combining PCA and removal of correlated attributes.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Parameters</th>
<th>AUC (CI)</th>
<th>Computation time</th>
</tr>
</thead>
<tbody>
<tr>
<td>C50</td>
<td>00000</td>
<td>0.89 (0.888-0.904)</td>
<td>2.30</td>
</tr>
<tr>
<td>C50</td>
<td>00001</td>
<td>0.98 (0.972-0.980)</td>
<td>1.72</td>
</tr>
<tr>
<td>RF</td>
<td>00000</td>
<td>0.90 (0.889-0.904)</td>
<td>35.01</td>
</tr>
<tr>
<td>RF</td>
<td>00001</td>
<td>0.98 (0.978-0.985)</td>
<td>17.46</td>
</tr>
<tr>
<td>SVM</td>
<td>00000</td>
<td>0.70 (0.690-0.714)</td>
<td>2893.80</td>
</tr>
<tr>
<td>SVM</td>
<td>00001</td>
<td>0.98 (0.975-0.983)</td>
<td>18.26</td>
</tr>
<tr>
<td>ADA</td>
<td>00000</td>
<td>0.89 (0.885-0.901)</td>
<td>39.04</td>
</tr>
<tr>
<td>ADA</td>
<td>00001</td>
<td>0.985 (0.982-0.989)</td>
<td>24.36</td>
</tr>
<tr>
<td>GLM</td>
<td>00000</td>
<td>0.87 (0.859-0.876)</td>
<td>1.56</td>
</tr>
<tr>
<td>GLM</td>
<td>00001</td>
<td>0.983 (0.980-0.987)</td>
<td>0.29</td>
</tr>
<tr>
<td>NN</td>
<td>00010</td>
<td>0.80 (0.798-0.817)</td>
<td>37.01</td>
</tr>
<tr>
<td>NN</td>
<td>00011</td>
<td>0.92 (0.916-0.929)</td>
<td>11.00</td>
</tr>
</tbody>
</table>

Table 7.3: AUC and Computation time for each algorithm, varying the use of PCA

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Parameters</th>
<th>AUC (CI)</th>
<th>Computation time</th>
</tr>
</thead>
<tbody>
<tr>
<td>C50</td>
<td>01001</td>
<td>0.98 (0.972-0.980)</td>
<td>1.72</td>
</tr>
<tr>
<td>RF</td>
<td>01001</td>
<td>0.98 (0.978-0.985)</td>
<td>17.46</td>
</tr>
<tr>
<td>SVM</td>
<td>01001</td>
<td>0.97 (0.9961-0.9970)</td>
<td>18.14</td>
</tr>
<tr>
<td>SVM</td>
<td>01001</td>
<td>0.965 (0.961-0.970)</td>
<td>25.9</td>
</tr>
<tr>
<td>ADA</td>
<td>01001</td>
<td>0.985 (0.982-0.989)</td>
<td>24.36</td>
</tr>
<tr>
<td>ADA</td>
<td>01001</td>
<td>0.96 (0.9955-0.9965)</td>
<td>25.26</td>
</tr>
<tr>
<td>GLM</td>
<td>00001</td>
<td>0.983 (0.980-0.987)</td>
<td>0.30</td>
</tr>
<tr>
<td>GLM</td>
<td>01001</td>
<td>0.97 (0.963-0.972)</td>
<td>0.30</td>
</tr>
<tr>
<td>NN</td>
<td>00011</td>
<td>0.92 (0.916-0.929)</td>
<td>11.00</td>
</tr>
<tr>
<td>NN</td>
<td>01011</td>
<td>0.93 (0.928-0.941)</td>
<td>14.92</td>
</tr>
</tbody>
</table>

Table 7.4: AUC and Computation time for each algorithm, varying the use of PCA and Correlated attributes
7.6 Algorithms

As already mentioned in the previous tables, by varying only the algorithms that the models are based on, the results presented in Table 7.5 were obtained. Best performance seems to be offered by C50, ADA and RF, values around 0.90 for AUC. The lowest performance is obtained by the SVM model, AUC of 0.70. The best computation time is obtained by GLM, under 2 seconds, closely followed by C50, with a value a bit higher than 2. The highest computation time is also found for the SVM model, around 48 minutes. The results in Table 7.5 indicate that the best model, when no feature selection method is applied and the outliers are not removed is Random Forests, while the worst is SVM.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Parameters</th>
<th>AUC</th>
<th>Computation time</th>
</tr>
</thead>
<tbody>
<tr>
<td>C50</td>
<td>00000</td>
<td>0.89 (0.888-0.904)</td>
<td>2.30</td>
</tr>
<tr>
<td>RF</td>
<td>00000</td>
<td>0.90 (0.889-0.904)</td>
<td>35.01</td>
</tr>
<tr>
<td>SVM</td>
<td>00000</td>
<td>0.70 (0.690-0.714)</td>
<td>2893.80</td>
</tr>
<tr>
<td>ADA</td>
<td>00000</td>
<td>0.89 (0.885-0.901)</td>
<td>39.04</td>
</tr>
<tr>
<td>GLM</td>
<td>00000</td>
<td>0.87 (0.859-0.876)</td>
<td>1.56</td>
</tr>
<tr>
<td>NN</td>
<td>00010</td>
<td>0.80 (0.798-0.817)</td>
<td>37.01</td>
</tr>
</tbody>
</table>

Table 7.5: AUC and Computation time for each algorithm

7.7 General evaluation

It is worth mentioning that all obtained models have all evaluation metrics, except computation time, higher than 0.7, with a slightly lower sensitivity for SVM and NN without PCA, removal of correlated attributes or removal of outliers. Computation time varies from over 48 minutes for SVM without PCA, removal of correlated attributes or removal of outliers to under one second for GLM with PCA. Some of these noticeable values can be seen in Table 7.6. However, the purpose of Table 7.6 is to present the values for all evaluation metrics for the best models, as well as for some weak models.

Figure 7.1 shows how, especially for the first two models, the values of the sensitivity, specificity, F-measure and AUC are very close. For NN without PCA, however, sensitivity is low compared to the others and specificity has the highest value among the evaluation metrics. It is also easy to observe that the best performance was obtained by the first two models. The best model is ADA with PCA, closely followed by RF with PCA. For the ADA with PCA model, all quality measures have very good values: 0.989 for sensitivity, 0.982 for specificity and 0.985 for AUC. RF with PCA has obtained a slightly better sensitivity, but a lower specificity than ADA with PCA. AUC, however is comparable for the two models. Computation time is also comparable, though lower for RF. Among the models without PCA, RF and C50 are the best, obtaining AUC value
The ROC curves of the models discussed above are plotted in Figure 7.2 for comparison. It is easy to observe that the ROC curves for the first two models and those for the next two are almost overlapping, which suggests that the models are very similar in terms of prediction performance. The last one, NN without PCA covers a much smaller area under the ROC curve, which is translated into a small value for AUC, as already mentioned and observed in Table 7.6.

Considering the results of related research papers, the very good results obtained for Random Forests were expected, while those of the AdaBoost model were not. However, from the two studies that mentioned AdaBoost, one [25] was investigating the
performance of Rotation Forests, an ensemble model based on Random Forests and AdaBoost, while the other [36] was comparing AdaBoost to an advanced ensemble model. The first one had good results for the Rotation Forests, while the second had good results for the newly proposed ensemble model. Therefore, the results obtained for the AdaBoost model represent a valuable contribution for the churn prediction research field.

7.8 Most frequently used attributes

In order to be able to determine which attributes were used more frequently by the models, 3 measures of usage were extracted and compared. First, from each of the models which offered a measure of usage of attributes as output, as well as an ordered list of the used attributes, the most used attributes were extracted. For the models based on the C5.0 algorithm, the usage measure is the usage percentage computed as the percentage of cases in the training data set for which the value of that attribute is known and is used in predicting a class [60]. Furthermore, the models based on the Random Forests algorithm measured the importance of the attributes as the total decrease in node impurities from splitting on the variable, averaged over all trees \(^1\), while for the models based on AdaBoost, a measure was computed by counting in how many of the created trees an attribute was used. Next, for each attribute, the value of each usage measure was averaged over all the models of the same type. Finally, the results were combined into what it is presented in Table 7.7. It is also important to emphasize that all the attributes from the above mentioned table were used by all of

\(^1\)randomForest R package
the 3 model types: C50, RF and ADA, while those left out registered 0 for at least one of the usage measures discussed above.

The attributes were ordered based on the 3 measures. When comparing 2 attributes, it was chosen as highest in rank the attribute that displayed greater values for at least 2 of the usage measures. However, it is easy to observe that, if normalizing the 3 values, the results would be comparable for the first 7 attributes, although not for the last 5 attributes. As a result, the sum of the 3 values was also taken into account when ordering the last 5 attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Average usage percentage in C50</th>
<th>Average importance in RF</th>
<th>Average usage frequency in ADA</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>PaidCashback</td>
<td>100</td>
<td>1500</td>
<td>45.5</td>
<td>CCB</td>
</tr>
<tr>
<td>PersonAwayContractsNumber</td>
<td>98</td>
<td>643.5</td>
<td>50</td>
<td>CCB</td>
</tr>
<tr>
<td>ContractCashback</td>
<td>59.75</td>
<td>724.5</td>
<td>25.5</td>
<td>CCB</td>
</tr>
<tr>
<td>ContractInitialType</td>
<td>92.5</td>
<td>544.5</td>
<td>15</td>
<td>CCB</td>
</tr>
<tr>
<td>ElectricitySJV</td>
<td>60.5</td>
<td>316</td>
<td>35</td>
<td>CCB</td>
</tr>
<tr>
<td>AggregatedSource</td>
<td>60</td>
<td>264.5</td>
<td>16</td>
<td>CCB</td>
</tr>
<tr>
<td>PersonPhoneContactsNumber</td>
<td>21.5</td>
<td>175.5</td>
<td>18</td>
<td>CP</td>
</tr>
<tr>
<td>ShortPostcode</td>
<td>0.2</td>
<td>199</td>
<td>8</td>
<td>CSD</td>
</tr>
<tr>
<td>CustomerAge</td>
<td>2</td>
<td>190</td>
<td>7</td>
<td>CSD</td>
</tr>
<tr>
<td>ContractAddressesNumber</td>
<td>46</td>
<td>84</td>
<td>25</td>
<td>CCB</td>
</tr>
<tr>
<td>ContractCurrentBalance</td>
<td>43</td>
<td>56</td>
<td>13</td>
<td>CCB</td>
</tr>
<tr>
<td>AggregatedEmailDomain</td>
<td>6.5</td>
<td>163.5</td>
<td>8.5</td>
<td>CPD</td>
</tr>
</tbody>
</table>

Table 7.7: Usage of attributes for C50, RF and ADA models

Another important thing to observe in Table 7.7 is that almost all of the attributes fall under the Customer contract and behavior category, which suggests that this is the most relevant data category for churn prediction for the Dutch energy market. However, one attribute from the Customer perception category, namely PersonPhoneContactsNumber, 2 from the Customer socio-demographics category, namely ShortPostcode and CustomerAge, and the only one from the Customer personal data category, namely AggregatedEmailDomain, were also present. Moreover, on the long list of used attributes more from the Customer socio-demographics category were present, such as PercentageOfIncomeRecipients and GasRegion. It can, therefore be stated that all the data categories proposed by this thesis have proven relevant for churn prediction. If a ranking had to be performed for the usefulness of each data category, the data in the above mentioned table suggests the following one:

- Customer contract and behavior
- Customer socio-demographics
- Customer perception
• Customer personal data

Furthermore, the usage measures also indicate that 67% of the useful data falls into the Customer contract and behavior category, 25% under the Customer socio-demographics category and only 4% under each of the Customer perception and Customer personal data categories. By examining only one type of models, C50, it can be observed that 3 of the attributes from the Customer contract and behavior category are used close 100% and 3 other, of the same category, for around 60%. The attribute from the Customer perception category is 21.5% used, while the other 2 categories are used in very small proportions. In conclusion, analyzing the usage of the attributes per model indicates the same ranking of categories.
Chapter 8

Conclusions and future work

8.1 Conclusions

As shown in Chapter 3 many of the studies that are related to the domain of this thesis use different performance measures to evaluate predictive models for churn. However, by examining the 2 measures that are used in most of the studies mentioned in Chapter 3, namely accuracy and AUC, the models proposed by this thesis offer very good results. For some of the models presented in Chapter 6 and evaluated in 7, the computed accuracy had values close to 0.9, even without PCA. Furthermore, the best of them display great prediction performance, with accuracy and AUC greater than 0.98. It is, therefore safe to state that the purpose of this thesis, as described in Chapter 1 has been met and the research questions can be answered. This section is going to provide a clear and concise answer to each of the research questions.

8.1.1 What information is needed to predict customer churn of energy suppliers and how useful each type of information is for churn prediction?

This research question was partially answered in Chapter 3, where a list of data categories was created based on existing literature:

- Customer contract and behavior: purchased services, duration of subscription, yearly consumption, payment behavior, offer discounts, customer source
- Customer perception: customer satisfaction
- Customer socio-demographics: gender, social status
- Macro environment variables: rank of company at specific moments, number of big market changes that the company overcame well while the customer had an ongoing contract - unavailable

It was therefore assumed that the listed data categories were relevant for churn prediction for the Dutch energy market and, for the 4 categories that were present among
the data available for this study, the hypothesis was also confirmed in the Evaluation section. The created predictive models used mostly data from the first category of the above mentioned list. Next, one attribute from the second category and one from the fifth were used and ranked among the first ten attributes used by most of the models. Finally, the socio-demographic attributes were less relevant, but still used by the predictive models.

8.1.2 Which of the different prediction models that were successfully used for churn prediction in other domains is the most suitable one for the Dutch energy market?

Six prediction algorithms were investigated during this study, in order to find the most suitable one for the Dutch energy market. It is important to mention at this point that, since this was the first study on churn prediction for an energy market when it was started, only models that were simple and had already been found as suitable for churn prediction in other domains were proposed for investigation, while more complex models were left for future work. The proposed algorithms were:

- Logistic Regression
- Decision Trees
- Random Forests
- AdaBoost
- Support Vector Machines
- Neural Networks

Nevertheless, an extensive number of models was created using the above listed algorithms, 2 feature selection methods and one extra data mining technique. Two of these models have exhibited very good prediction performance, one based on AdaBoost and the second one based on Random Forests. Both of the models were created after applying Principal Components Analysis on the final data set and both have measured AUC above 0.98. While Random Forests have been successfully used for churn prediction before, the results obtained for AdaBoost were surprising and are considered an unexpected contribution of this thesis.

8.1.3 Can customer churn of energy suppliers be predicted with an accuracy that would allow a company to rely on prediction for retaining their customers?

As already mentioned in the previous subsection, two of the proposed and evaluated models have shown great prediction performance. As a result, one of the conclusions of
this thesis is that an energy supplier should be able to rely on the proposed predictive models their retention campaigns. The company can use the predictive model to create a list of customers that are likely to churn and only offer special deals to the customers on the list. This is expected to result in a decrease of the retention costs.

8.2 Future work

While the results of this study were better than expected, there is always space for improvement. Most importantly, while following the CRISP-DM approach, this study lacked an actual deployment phase. Although the obtained predictive models could be used for churn prediction at the moment, validating them against a newly connected data set would be advised. Moreover, since the company that offered the data has evolved, reading and storing data from smart energy meters and collecting more data about customer complaints, while this study was performed, more data is available at the moment and whether the current models would perform as good with the new data as they did with the old data should be investigated.

Another development idea would be to explore the possibility of collecting data as time series. This kind of data should improve prediction performance and allow for it to be performed earlier in the contractual period.

Finally, one of the data categories that was unavailable for this study, the macro-environmental variables, should be searched further and added to the data set. However, if this specific category of data would prove to be too costly to collect, one could also investigate the possibility of collecting and using data from social networks.
Bibliography


[57] Pav Kalinowski. Understanding confidence intervals (cis) and effect size estimation.

