UNIVERSITÉ LIBRE DE BRUXELLES
Faculté des Sciences
Département d'Informatique

Churn Prediction and Causal Analysis on Telecom Customer Data

Théo Verhelst

Promotor :
Prof. Gianluca Bontempi

Mémoire présenté en vue de
l'obtention du grade de
Master en Sciences Informatiques

Academic year 2018 - 2019

Abstract

Telecommunication companies are evolving in a highly competitive market where attracting new customers is more expensive than retaining existing ones. Retention campaigns can be used to prevent customer churn, but this requires efficient churn prediction models. Such prediction is a difficult problem, involving large amount of data, non-linearity, imbalance and overlap between churners and non-churners. In this master thesis, we approach this problem with Orange Belgium customer data. A descriptive analysis of the dataset is conducted, and predictive modeling of churn is achieved with a random forest classifier. The large class imbalance between the two classes is handled with the Easy Ensemble algorithm. We assess the impact of different data preprocessing techniques such as feature selection and creation of new features. The directionality of the impact of variables on churn is estimated through a sensitivity analysis. Also, we explore the application of data-driven causal inference, which allows to infer causal relationships between variables purely from observational data. More specifically, a causal Bayesian network, two methods of Markov blanket inference, two causal filters and a supervised method are applied. We draw general conclusions on the possible causes of churn, supported by the prior knowledge of experts at Orange Belgium.

Résumé

Les compagnies de télécommunication évoluent dans un marché hautement compétitif où attirer de nouveaux clients est plus coûteux que retenir les clients déjà présents. Des campagnes de rétention peuvent être utilisées pour réduire la résiliation des clients, mais cela nécessite des modèles de prédiction efficaces. La prédiction de résiliation est un problème difficile, impliquant de grands quantités de données, des relations non-linéaires, des classes non-équilibrées et avec une large superposition inter-classe. Dans ce mémoire de master, nous abordons le problème de prédiction de résiliation avec des données client de Orange Belgium. Une analyse descriptive du jeu de données est effectuée, et une modélisation prédictive de la résiliation est obtenue en utilisant un classificateur random forest. Le non-équilibrage entre les classes est pris en compte avec l'algorithme Easy Ensemble. Nous évaluons l'impact de différentes techniques de prétraitement de données, telles que la sélection de variables et la création de nouvelles variables. La directionalité de l'impact des variables sur la résiliation est déduite avec une analyse de sensibilité. Nous explorons également l'utilisation de l'inférence causale, qui permet de comprendre les liens de causalité entre différentes variables à partir de données d'observation. Plus spécifiquement, nous utilisons un réseau bayésien causal, une méthode d'inférence de couverture de Markov, deux filtres causaux et une méthode supervisée. Nous en tirons des conclusions générales sur les possibles causes de la résiliation, en prenant en compte les connaissances à priori d'experts chez Orange Belgium

Acknowledgments

First and foremost, I would like to thank Prof. Gianluca Bontempi for allowing me to work with him on this subject and for his unfailing support and dedication. It is only after these two years that I am able to realize how much I learned on scientific research and communication with him.

In addition, I am grateful to Olivier Caelen, Jean-Christophe Dewitte, and the rest of the data science team at Orange for introducing me to their work, for providing all the resources necessary for this thesis, and for the friendly atmosphere in the Orange office. In particular, the work of Pierre Brogniet on churn prediction during his internship brought very valuable and complementary knowledge to my work.

My special thanks go to my parents, my partner and the rest of my family for their love and support during these two years and before.

I would like to thank my fellow students at the ULB for the mutual support and the uncountable science-enthusiastic discussions, fostering my motivation to pursue scientific research.

Lastly, I would like to express my gratitude to Prof. Tom Lenaerts and Prof. Bernard Manderick for accepting to evaluate my thesis. Moreover, Prof. Tom Lenaerts also allowed me to take part in an Erasmus program at Southampton University, which played a significant role in my academic formation.

Contents

Chapter 1

Introduction

1.1 Churn in the telecommunication industry

In recent years, the number of mobile phone users increased substantially, reaching more than 3 billion users worldwide. The number of mobile phone service subscriptions is above the number of residents in several countries, including Belgium (ITU, 2018). Telecommunication companies are evolving in a saturated market, where customers are exposed to competitive offers from many other companies. Hadden, Tiwari, Roy, and Ruta (2007) show that attracting new customers can be up to six times more expensive than retaining existing ones. This led companies to switch from a selling-oriented to a customer-oriented marketing approach. By building customer relationship based on trustworthiness and commitment, a telecommunication company can reduce the incentives for their client to churn, therefore increasing benefits through the subsequent customer lifetime value.

One of the various marketing processes used to improve customer relationship is to conduct retention campaigns. This traditionally consists in selecting clients according to some simple statistical criteria and offering them a promotion or advantage. Typical promotions include a reduced invoice, free calls, SMS or data volume. However, due to the limited nature of this statistical analysis, it is plausible that the customers thus reached might never have planned to churn in the first place. While this is of course not a problem for the customer, it would be far more beneficial for the telecommunication company to be able to focus the retention campaigns only on risky customers, in the hope of preventing attrition that would otherwise occur if no action is taken. The problem of detecting churn can be addressed with *data mining*, by collecting data about customers and using this information to infer typical patterns exhibited by risky clients. This data-driven approach is nowadays taken by most major telecommunication companies, and a part of the data mining literature is devoted to churn detection. We will describe this approach further in the next section.

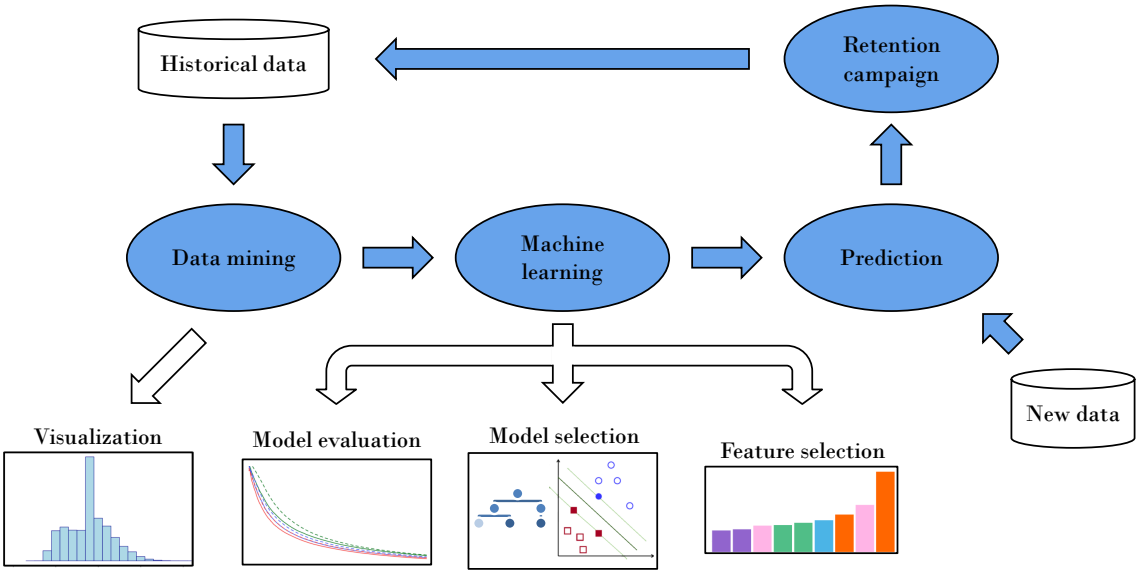


Figure 1.1 – The churn prevention process implemented at Orange.

1.2 Churn detection and prevention

The churn prevention process (depicted in figure 1.1) starts by collecting data about the customers and creating a *historical database*. This data summarizes the calls, messages, internet usage and other actions performed by the customers[1]. It also includes information about the subscription, such as the type of tariff plan, its price, the subscription date, and so on. Finally, personal data provided by the customer may as well be used, such as the age or the place of residence. *Data mining* is then used on this database to provide a quantitative understanding of the customers and their overall behavior. In particular, the difference between churners and non-churners behavior can be visualized. This is useful to decide which type of machine learning procedure should be used for churn prediction, but can also bring valuable knowledge to marketing and customer relationship teams. Once the data is sufficiently understood, a set of relevant *machine learning* models are built. These models learn from the historical data the patterns typically exhibited by clients that will churn in the near future. The types of patterns being learned, the techniques used to find them, and the underlying assumptions are dependent on the model under consideration. For example, a *naive Bayes classifier* assumes that each variable is independent of the others if we know whether the customer is a churner or not. In order to decide which model should be selected, the performances of each model are evaluated by testing them on data left out of the training phase, and different models are compared. Feature selection is also performed, and consists in evaluating the importance of each variable for predicting churn and training the model by using only the most important ones. This reduces the computation time and generally improves the performances of the model since this reduces the noise

[1]This data is limited to the metadata of the phone call, message, or internet usage. The content of the communication is never used. Moreover, the location information in the metadata is also excluded.

unimportant variables bring along. Once an efficient model has been selected, the latest customer data is submitted to the model which outputs a probability of churn for each customer. By ordering the customers by churn probability, a list of the riskiest customers is established and sent to the campaigning team. They split this set of clients into a *target group* and a *control group*. Each customer of the target group is offered an incentive either by phone call, email or message, while the control group is left untouched. This allows, a few months later, to evaluate the impact of the retention campaign by assessing the difference of churn rate in retrospective in the two groups. Also, the accuracy of the prediction model can be evaluated by comparing the control group and the rest of the customer base

1.3 Causal inference

Depending on the resources available and the techniques used, this data mining pipeline can successfully predict potential churners, therefore allowing to conduct targeted retention campaigns. But campaigners are then faced with another challenge: what should they propose to the selected customers? Indeed, the predictions given by data mining algorithms usually just consist in a probability of churn. This prediction therefore does not indicate *why* the customer is about to stop her subscription. We need different analysis tools to tackle this problem. This is the purpose of *causal inference*, which is a formal approach to find the causes of some event in a system. Causal inference is usually conducted through *controlled randomized experiments* (Fisher, 1937). In the context of customer relationship management, controlled experiments are possible through retention campaigns, where the offers made to the customers act as variable manipulations. For example, offering a discount would act on the variable "invoice amount". Such experiment is out of the scope of this master thesis, since it requires time, planning, and a collaboration with the direct marketing department. We thus focus on data-driven approaches, which are based on some properties of the statistical distribution of causally linked variable.

To give an example of how causal inference can be conducted without experiment, consider the simplistic world where two types of phone are available, an expensive one and a cheap one. The expensive phone type makes customers consume more data per month than the cheap one, because more data-hungry application are available on it. Also, it increases the probability of churn because these phones are targeted more often by concurrent advertisement. We represent such causal scenario with a *directed acyclic graph* as in figure 1.2. We assume that once we know the phone type of a customer, its data usage and its propensity to churn are independent, since these are two different consequences of the customer's choice of phone. We plotted the data usage against the probability of churn for type of phone in figure 1.3. A positive correlation can be observed between data usage and churn but disappears when considering each phone type separately. If a causal relationship also existed between data usage and churn probability, the statistical dependency would still be

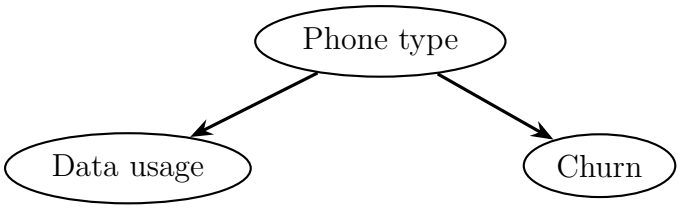


Figure 1.2 – Toy example of a causal diagram.

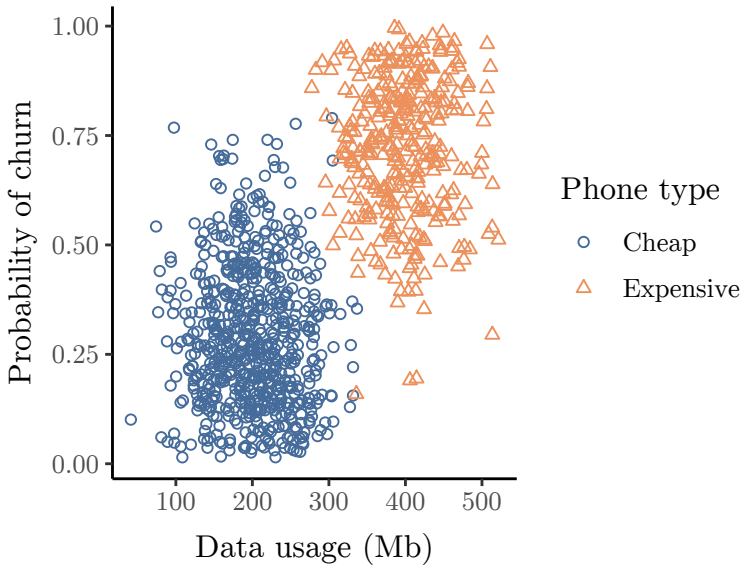


Figure 1.3 – Toy example of data usage against the probability of churn for different phone types.

visible, even when conditioning on the phone type. This idea of conditioning to discard putative causal links is at the basis of most causal inference algorithms.

Such inference methods are based on a number of modeling assumptions, such as ability to represent the underlying causal mechanism by a graph as in figure 1.2, or the absence of confounding factor. This latter assumption would be violated if, in our previous example, the age of the customer influences both its choice of phone type and its propensity to churn. The causal inference algorithm would still conclude that the expensive phones causes clients to churn, while in reality, they churn solely because of their young age. This sort of erroneous conclusions can lead to ineffective action in retention campaigns. Other inference methods imply other assumptions, and care must be exercised when using them.

1.4 Context and motivation

This master thesis is conducted in collaboration with Orange Belgium, and originated from the long-lasting scientific collaboration between Prof. Gianluca Bontempi (ULB Machine Learning Group) and Dr. Olivier Caelen (Orange Belgium). This collaboration enables us to work on real-world data, in Orange Belgium premises, and with people directly involved in the subject (data science and business intelli-

gence teams at Orange). The churn prediction problem is challenging in many regards: the dataset is large (in our case, 1.5 million entries per month, over 5 months), highly imbalanced (there are very few churners in the whole customer base), and highly overlapping (many non-churners exhibit the same behavior as churners). It is therefore an interesting subject for a master thesis in machine learning, since it requires the use of different state-of-the-art tools for overcoming these challenges.

The churn prediction task shares many similarities with the fraud detection problem, for example regarding the class imbalance and overlap, and the large amount of data. The Machine Learning Group (MLG) has a fruitful collaboration with Wordline in fraud detection since 2012 when Prof. Bontempi wrote and supervised the Doctiris project "Adaptive real-time machine learning for credit card fraud detection" (PhD student: Andrea Dal Pozzolo). The research activity in this domain continued thanks to "Brufence: Scalable machine learning for automating defense system" project and the recent TEAM-UP project DefeatFRAUD. This thorough experience in the subject is a motivation and a valuable asset to pursue research in churn prediction.

The interest for causal inference comes from the experience of the MLG in the subject. They mainly applied causal inference onto bioinformatics application, such as gene selection (Bontempi, Haibe-Kains, Desmedt, Sotiriou, & Quackenbush, 2011) or microarray data (Bontempi & Meyer, 2010). More recently, a competition on causal analysis was organized on the Kaggle website (https://www.kaggle.com/c/cause-effect-pairs), with the goal of fostering causal discovery between two variables. This led to the development of new methods, notably using machine learning (Bontempi & Flauder, 2015). Moreover, the use of causal inference is seldom explored in the literature on churn prevention. It is thus stimulating to conduct research at the intersection of these two domains, benefiting from the technical expertise of the MLG and the business knowledge of Orange.

1.5 Contributions

The main contributions of this thesis are

- Understanding of the churn prediction problem with a real-world dataset from a telecom company (section 3.1).
- Evaluation of the predictive power of a state-of-the-art churn prediction model, and the impact of several variations of the model by using different features and different type of subscription contracts (section 3.3).
- Assessment of the directionality of the impact of a variable on prediction, by shifting the value of the variable in the test set and measuring the difference in average predicted score (section 3.3).
- Study of causal analysis, and its application to churn prediction from observational data (section 4.4).

1.6 Outline

This master thesis is partitioned into 5 chapters, presented here.

1. (Chapter 1) An introduction to the problem of churn in the telecommunication industry, the current methods in use to tackle it, and the contributions of this master thesis.
2. (Chapter 2) Exposition of the state of the art in churn prediction and causal analysis.
    - Churn prediction: choice of predictive model, data preprocessing, class balancing and evaluation measure.
    - Causal analysis: Bayesian network learning, Markov blanket inference, information-theoretic filters, bivariate and supervised methods.
3. (Chapter 3) Assessment of a churn prediction model on Orange customer data. This chapter is further divided into:
    - Presentation of the dataset
    - Description of the data preparation
    - Description of the experimental setting
    - Presentation of the results
    - Comparison to other state-of-the-art methods
    - Conclusion and main outcomes of the experiments
4. (Chapter 4) Exploration of causal inference methods for churn prevention:
    - Theoretical background for causal inference
    - Scope of application
    - Prior knowledge of the possible causes of churn
    - Description of the experimental setting
    - Presentation of the results
    - Discussion and conclusion
5. (Chapter 5) A conclusion, general remarks and directions for further work.

1.7 Notation

The mathematical notation used throughout this document is presented in table 1.1. Bold font denotes vectors or sets, and uppercase letters denote random variables. A uppercase bold font thus denotes a set or vector of random variables.

n	Number of features
$\mathcal{X} \subseteq \mathbb{R}^n$	Feature space
$\boldsymbol{X} = [X_1, \ldots, X_n]$	Vector of random variables in $\mathcal{X}$
$\boldsymbol{x} = [x_1, \ldots, x_n]$	Feature vector, realization of $\boldsymbol{X}$
$\mathcal{Y} = \{0, 1\}$	Target label space
$y \in \mathcal{Y}$	Example label
$P(e)$	Probability of an event e
S	Random variable of predicted score as a function of $\boldsymbol{X}$
s	Predicted score for a given $\boldsymbol{x}$, realization of S
$t \in [0, 1]$	decision threshold
$f_y(s)$	Probability density function of S for instances labeled y
$F_y(s)$	Cumulative distribution function of S for instances labeled y
$X \perp Y$	Random variables X and Y are independent
$\boldsymbol{X} \setminus \boldsymbol{Y}$	Set difference between $\boldsymbol{Y}$ and $\boldsymbol{Y}$

Table 1.1 – Summary of the mathematical notation.

Chapter 2

State of the art

2.1 Churn prediction

We organize our presentation of the state of the art in churn prediction by considering 4 aspects of interest in a usual churn prediction process: learning algorithm, data preprocessing, class balancing and choice of an evaluation measure.

Learning algorithm

A large number of machine learning models have been applied to churn prediction in the literature. While some studies focus on simple and interpretable models such as decision tree (Keramati et al., 2014) or logistic regression (Olle & Cai, 2014), other studies prefer the use of more complex models, at the expense of direct interpretation. These methods include boosting applied on simple classifiers (Vafeiadis, Diamantaras, Sarigiannidis, & Chatzisavvas, 2015), random forests, support vector machine, neural networks, among others (Umayaparvathi & Iyakutti, 2016; Verbeke, Dejaeger, Martens, Hur, & Baesens, 2012). An extensive overview of the current trends in churn prediction modeling is given in (Kayaalp, 2017). We describe in this section the learning algorithms used most often according to this review.

A decision tree is a non-linear model that partitions the input space into mutually exclusive regions, and each of these regions is assigned a mapping between the input vector and the output value. In the case of a classification tree, each region is assigned a label. The partition of the input space is determined by a tree structure composed of internal nodes and terminal nodes. Each internal node is associated with a rule that determines which child node should be visited next, based on the value of one of the variables. Each of the terminal nodes is associated with a mapping function, a label in the case of a classification tree. The main methods of decision tree inductions are ID3 (Quinlan, 1986), C4 (Quinlan, 1987), and CART (Breiman, Friedman, Olshen, & Stone, 1984).

Logistic regression is a statistical binary classification model that assigns a probability to each point of the input space. The probability of an input vector $\boldsymbol{x} = [x_1, \ldots, x_n]$ is calculated as the application of the sigmoid function on a linear combination of the values in $\boldsymbol{x}$. More precisely, to each variable X_i is assigned a co-

efficient w_i, and an intercept coefficient w_0 is also defined. The predicted probability of a logistic regression model is thus

$$s(\boldsymbol{x}) = \frac{1}{1 + \exp\left(-(w_0 + w_1 x_1 + \cdots + w_n x_n)\right)}.$$

The weights $w_0, \ldots, w_n$ can be learned by maximizing their log-likelihood given a set of learning examples using the gradient descent algorithm.

Artificial neural networks are computational models that process input in a distributed, parallel fashion. They are initially inspired by biological neurons, but their current development has moved towards a more rigorous interpretation based on theoretical machine learning. A wide variety of neural network architectures exists, but we present here only *feed-forward neural networks* (FNN) for their prominent use in the churn prediction literature. A MLP consists of a series of layers, and each layer comprises multiple neurons. In the fully connected configuration, a neuron takes as input all outputs from the previous layer. The output of the neuron is the result of applying a non-linear activation function to the dot product of the inputs with a vector of weights. Many activation functions exist, such as the sigmoid function, the rectified linear unit (Nair & Hinton, 2010), the hyperbolic tangent, and many others (Cheng & Titterington, 1994). The weights of each inter-neuron link are learned by gradient descent on a loss function given training samples. The exact learning setup can vary widely, thanks to the numerous possibilities in the choice of the loss function and the gradient descent implementation.

A support vector machine (SVM) is a geometrical classification model based on the notion of *maximum margin hyperplane*. A SVM model splits the input space into two separate regions by using a hyperplane. This hyperplane is chosen as to maximize its distance to the nearest data point in either class. In order to handle arbitrary input space or non-separable classes, a kernel function is used to project the input space into a well-known Hilbert space. Also, slack variables can be introduced to relax the strict requirement that the margin should perfectly separate the two classes. The geometrical formulation of this model enables an exact optimization procedure based on the dual of the Lagrangian. One of the first modern formulations of the SVM model is given in (Boser, Guyon, & Vapnik, 1992).

A random forest is an ensemble model composed of multiple decision trees. One of the first descriptions of this model is given in Breiman (2001). Each tree is constructed from a subset of samples and using only a subset of the variables. In the case of a classification task, each of the individual trees votes for the output label of a given data example, whereas in the case of a regression task, the output is the average prediction of all trees. The intuition is to obtain a set of deep trees having a high variance but low biais. The average of these trees has thus a lower variance than each individual tree, as long as these trees are uncorrelated. The tree are made uncorrelated by sampling independently the set of samples and variables used to train each of them.

Data preprocessing

While the choice of model is important, it is equally important to consider a proper choice of features, data preprocessing and evaluation measure. Feature choice is limited by the available data infrastructure and usually consists of call detail records on the course of a few weeks. Huang, Kechadi, and Buckley (2012) presented however how new kinds of features, including demographics profiles, marketing segments, and complaint information, can improve prediction accuracy.

Data preprocessing refers to the process following data acquisition and consists of modifying the data in various ways before the use in a predictive model. This comprises, non-exhaustively, feature engineering, data reduction, anomalies removal, encoding of categorical variables, and data normalization (Zhang, Zhang, & Yang, 2003). Coussement, Lessmann, and Verstraeten (2017) give an overview of common preprocessing steps used in churn prediction, and how careful preprocessing can positively affect the performance of the model. They even show that a simple logistic regression model applied to optimally preprocessed data competes with complex learning algorithms such as neural network or support vector machine applied on data preprocessed with a basic preparation procedure.

Recent years have witnessed the widespread usage of network-based classification for churn prediction. Verbeke, Martens, and Baesens (2014) present some experiment in this area of research. (Óskarsdóttir et al., 2017) perform an extensive benchmark of the different techniques proposed in the literature. The approach is based on call detail records (CDR) containing the communication logs of the subscribers. This data can be organized as a graph where nodes represent customers and edges represent social ties between customers. The basic assumption of network-based classifiers, which use such graphs to classify customers as churners or non-churners, is that customers having social ties with churners are more likely to churn themselves. This assumption is purposely loosely defined, as its exact implementation implies different assumptions and modeling decisions (Óskarsdóttir et al., 2017). From that, one can either construct a predictive model that directly uses the social graph, or extract features from the network and use them as the input of a conventional classifier, or even combine the two approaches. The outcome of the two articles is that relational and classical non-relational classifiers detect different types of churners and that a combination of both types of classifiers approaches performs best.

Óskarsdóttir, Van Calster, Baesens, Lemahieu, and Vanthienen (2018) present a novel, end-to-end approach to the problem of churn detection where a time-varying social network of the customer base is constructed, and a multivariate time series is then extracted for each customer from this network. Then, different time series classifiers are used to predict churn. A novel multivariate time series classifier is proposed, an adaptation of the similarity forest classifier (Sathe & Aggarwal, 2017). Óskarsdóttir et al. (2018) conclude that their approach outperforms state-of-the-art time series classifiers and non-time-based models for early churn prediction. However, static random forest and logistic regression are better at predicting late

churn (that is, on short time scales).

Class balancing

It is important to consider class imbalance when designing models for churn prediction. Indeed, the number of churners is usually very low compared to the total number of customers, and most machine learning models are usually not suited to handle highly imbalanced data (Batista, Prati, & Monard, 2004). Class balancing techniques have to be used to tackle this problem. These techniques can roughly be divided into three categories: data-level balancing, model-level balancing, and ensemble methods.

Data-level balancing consists in modifying the dataset by either decreasing the number of majority instances, increasing the number of minority instances, or both. Random oversampling (ROS) consists in randomly replicating instances of the minority class, whereas random undersampling (RUS) randomly eliminates instances of the majority class. ROS increases the risk of overfitting, by replicating exactly existing examples, whereas RUS loses information by removing potentially useful examples, thus increasing variance. Synthetic minority oversampling technique (SMOTE) (Chawla, Bowyer, Hall, & Kegelmeyer, 2002) addresses the overfitting issue by creating new instances of the minority class with linear interpolations between minority example and their nearest neighbors. Several variations and improvements exist, such as ADASYN (He, Bai, Garcia, & Li, 2008) and MWMOTE (Barua, Islam, Yao, & Murase, 2014). Intelligent undersampling techniques also exist, such as the one-sided selection (OSS) (Kubat, Matwin, et al., 1997) that removes redundant or borderline majority instances, or the cluster-based undersampling (CLUS) (Yen & Lee, 2009) that groups majority instances into clusters, and reduce the number of instances in each cluster. Dal Pozzolo, Caelen, and Bontempi (2015) provide a condition under which undersampling improves the ranking of test instances, which depends on the class balance rate, the nonseparability of the data, and the variance of the learning algorithm. This highlights the fact that no unique undersampling strategy is adapted to all problems.

Model-level balancing consists in modifying the learning algorithm in such a way that minority instances are given more importance, often through the use of asymmetric misclassification costs. Ting (2002) applied this methodology to classification trees, and Veropoulos, Campbell, and Cristianini (1999) to support vector machines.

Ensemble methods for class imbalance combine multiple models to obtain a better overall model while taking into account class imbalance. For example, Roughly Balanced Bagging (Hido, Kashima, & Takahashi, 2009) uses bagging to create multiple classifiers, each being trained by undersampling using a number of majority instances determined by a negative binomial distribution. Chen and Breiman (2004) proposes integrating balancing by random undersampling or misclassification weighting into the random forest algorithm.

Since we use the ensemble balancing method *EasyEnsemble* (Liu, Wu, & Zhou, 2009) in this thesis, we present it here in more details. Given the set of positive

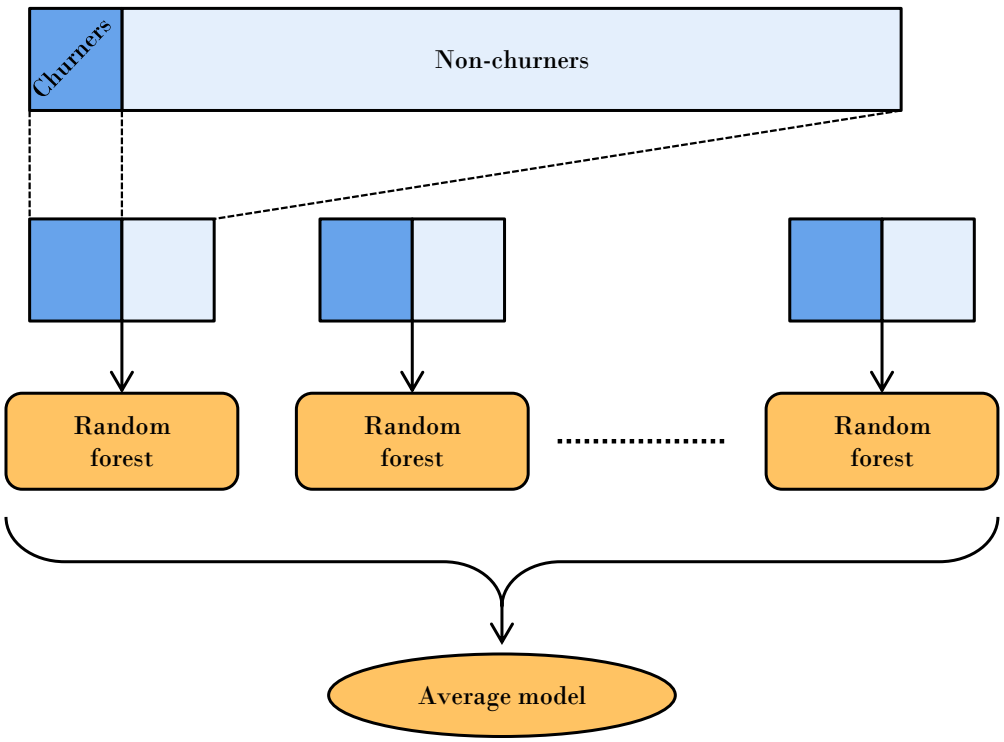


Figure 2.1 – Easy ensemble methodology for unbalanced data. A set of random forests is trained each on the whole set of positive instances, and on a randomly chosen subset of negative instances. The final predictions are the average of all the random forests individual predictions.

instances $\mathcal{P}$ and the set of negative instances $\mathcal{N}$, data-level undersampling methods consist in choosing a subset $\mathcal{N}' \subset \mathcal{N}$ such that, by training on $|\mathcal{N}'| \cup |\mathcal{P}|$, the ratio between positive and negative instances is suitable for a given learning algorithm. It is common to choose a ratio of 1. The main issue with this approach is that a large number of positive instances (i.e. all of $\mathcal{N} \setminus \mathcal{N}'$) is ignored, thus increasing the variance of the resulting trained model. EasyEnsemble overcomes this issue by independently sampling $T > 1$ subsets $\mathcal{N}_1, \ldots, \mathcal{N}_T$ from $\mathcal{N}$. Then, T predictive models are trained on all of $\mathcal{P}$ and individually on each $\mathcal{N}_i$. The final predictions are the average of the predictions of the T models. This process is pictured in figure 2.1. In our case, the predictive models are random forests.

Dal Pozzolo, Caelen, Waterschoot, and Bontempi (2013) showed that no unique class balancing method works best in all situations, and the they should be evaluated and selected on a case by case basis. An extensive overview and comparison of class balancing techniques in churn prediction is given in (Zhu, Baesens, & vanden Broucke, 2017). Dal Pozzolo, Caelen, Le Borgne, Waterschoot, and Bontempi (2014) compared several balancing techniques in the context of credit card fraud detection, as well as solutions for other aspects of the fraud detection problem (online learning and concept drift).

	Predicted P ($s > t$)	Predicted N ($s \leq t$)
Actual P ($y = 1$)	True positives (TP)	False negatives (FN)
Actual N ($y = 0$)	False positives (FP)	True negatives (TN)

Table 2.1 – Confusion matrix.

Evaluation measure

The last step of a predictive model assessment is the evaluation of classification performance. Several performance measures exist for this purpose, such as precision, recall, F-score, area under the receiver operating characteristic curve (AUC), lift, maximum profit criterion, and expected maximum profit criterion. We briefly explain here these measures and their current use in the literature.

We first introduce the notion of confusion matrix, as shown in table 2.1. The fact that a customer is a churner or not is labeled by the value of y, and corresponds the first column. We also write a churner as an "actual P" (for *positive* instance), and similarly a non-churner as "actual N" (for *negative* instance). The prediction of a classification algorithm for a customer is a score s. If this score is larger than a chosen threshold t, then the customer is predicted to be a churner, written "predicted P". Similarly, if the score s is below the threshold t, the customer is predicted not to be a churner, written "predicted N". The four possible combinations that these two conditions yield are presented in the confusion matrix in table 2.1. The value of each cell (TP, FP, FN, and TN) denotes the number of instances which meet both of the corresponding criteria.

Precision is the fraction of true churners among all those we predicted to be churners, and recall is the fraction of predicted churners among all the true churners in the population.

$$precision = P(y = 1|S > t) = \frac{TP}{TP + FP}$$

$$recall = P(S > t|y = 1) = \frac{TP}{TP + FN}$$

In order to optimize both of these scores at the same time, one can use the F-score (also called F1 score, or F-measure), which is the harmonic mean of precision and recall.

$$F1 = \frac{2 \times precision \times recall}{precision + recall}$$

By using the harmonic mean, we punish low values in any of the two values. It is sometimes used in the churn prediction literature (Ahmed & Maheswari, 2017; Keramati et al., 2014).

Receiver operating characteristic (ROC) curve (Krzanowski & Hand, 2009) is a plot of all possible compromises between true positive rate (TPR) and false positive rate (FPR). TPR is another name for recall, and FPR is the fraction of non-churners falsely predicted to be churners, among all non-churners.

$$FPR = P(S > t|y = 0) = \frac{FP}{FP + TN}$$

In order to compare the ROC curve of different models, we use the area under the curve (AUC) as a measure of performance.

$$AUC = \int_{-\infty}^{+\infty} F_1(s) f_0(s) ds$$

AUC is very often used in churn prediction (Coussement et al., 2017; Mitrović, Baesens, Lemahieu, & De Weerdt, 2018) because it is less sensitive to class imbalance (few churners in a large population of customers) and misclassification cost (Verbeke et al., 2012).

An important drawback of these performance measures is that they do not represent the real objective of churn prediction: given that we are certainly not able to offer an incentive to all customers likely to churn, we have to restrict the retention campaign to a limited number of customers. From that, we wish to minimize the number of misclassifications occurring in this small subset of customers. This is the definition of the lift: given a certain set of customers (usually the subset of the customers with the highest predicted probability of churn), the lift is the ratio between the churn rate in this set and the churn rate in the whole customer base.

$$lift(t) = \frac{P(y = 1|S > t)}{P(y = 1)}$$

The value of the lift indicates how much we do better than choosing customers at random. For example, a lift of 3 indicates that the model will give a set of customers with 3 times as much churners as if we picked that many customers at random. The number of customers to consider is dependent on the application (it should ideally be the number of customers reachable by the retention campaign), but a lift at 10% is sometimes used as a baseline (Verbeke et al., 2014; Zhu et al., 2017).

In order to determine formally how many customers should be included in the retention campaign, and therefore in the lift measure, Verbeke et al. (2012) developed the maximum profit criterion (MPC) and the expected maximum profit criterion (EMPC) (Verbeke et al., 2012; Verbraken, Verbeke, & Baesens, 2013). These two measures consist of evaluating the expected costs and benefits of conducting a retention campaign and selecting the optimal number of customers to call. The difference between MPC and EMPC is that MPC considers the cost and benefits to be known and fixed, while EMPC assigns a probability density function to these parameters. MPC and EMPC are often used in churn prediction (Óskarsdóttir et al., 2018; Stripling, vanden Broucke, Antonio, Baesens, & Snoeck, 2018; Zhu et al., 2017).

2.2 Causal analysis

Finding and using causes is crucial in human reasoning and decision making. While a predictive model returns the probability of a target value given that we observe a certain input vector, a causal model is supposed to return the target probability given that we set (e.g. by intervention) that input. The aim of causal analysis is to determine the consequences of manipulations and is opposed to the process of making predictions from observations. When used as a feature selection criterion, it enables increased robustness to violation of the assumption of independent and identical distributions (e.g. concept drift) (Guyon, Aliferis, et al., 2007) and an enhanced understanding of the mechanism underlying the data. The gold standard for causal modeling is to carry out *randomized controlled experiments* (Fisher, 1937). For example, in order to assess the influence of moderate wine consumption on heart disease, it is not enough to measure wine consumption and heart disease in the population. This may lead to erroneous conclusions, such as a socio-economic factor that causes both increased wine consumption and risks of heart disease. In order to avoid such a problem, one may assign to a randomly chosen group of people a moderate wine consumption. If this group then shows a significantly different risk of heart disease, then we may conclude that a causal relationship indeed exists, and the apparent correlation is not due to a confounding factor. However, such experiments may be expensive, unethical, difficult to implement or unfeasible. This, along with advances in computation, data storage capabilities and new machine learning techniques, led to the development of causal inference based on observational data. We review here 5 approaches to data-driven causal inference:

- Bayesian network learning
- Markov blanket inference
- Information-theoretic filters
- Bivariate methods
- Supervised methods

All the approaches model causal relationships between random variables. These random variables represent, for example, the different features available about a customer in the churn prevention problem.

Bayesian network learning

A causal Bayesian network is a discrete acyclic graph where the set of nodes correspond to the set of random variables, and a directed link indicates a causal relationship between two variables. This graph is also accompanied by the joint probability distribution of the set of random variables. Two conditions are usually imposed on the graph and the probability distribution (the causal Markov condition and the causal faithfulness condition, explained in section 4.1) to ensure that it respects the

semantics of a causal model. Notably, these conditions also allow predicting the effect of a manipulation (Spirtes, 2010). Causal Bayesian network can be learned from observational data, and two types of procedures have been developed for that purpose. The first one consists in a search in the graph structure space, and optimization a fitness function. This approach is detailed in (Heckerman, 1998). The second one is based on independence tests between pairs of variables, and iteratively construct and orient edges until a valid class of causal graphs is found. The PC and FCI algorithms use this approach (Spirtes, Glymour, & Scheines, 1993).

Markov blanket inference

The Markov blanket of a given target variable in a Bayesian network is a minimal set of variables that are shielding the target variable from the influence of other variables in the network. A formal definition is given in section 4.1. This subset of variables contains all the information needed to predict the target variable (that is, any additional variable would be redundant). Moreover, if the causal Markov and faithfulness assumptions hold, then this Markov blanket is the set of direct causes (parents), direct effect (children), and also the direct causes of the direct effects (spouses). Inferring the Markov blanket of a variable, as opposed to inferring the complete Bayesian network, is beneficial when the number of variables is large, such as in microarray data. Several algorithms exist for Markov blanket inference, notably KS (Koller & Sahami, 1996), GS (Margaritis & Thrun, 2000) IAMB (Tsamardinos, Aliferis, & Statnikov, 2003), HITON (Aliferis, Tsamardinos, & Statnikov, 2003), and MMPB (Tsamardinos, Aliferis, & Statnikov, 2003). These algorithms start with an empty Markov blanket and search for parents, children, and spouses of the target variable by using conditional independence tests. A number of heuristics are used to speed up the search, and most of these algorithms also include a second phase where false positives are removed from the result.

Information-theoretic filters

A filter algorithm is a feature selection method that provides a ranking of the input variables, based on some measure of statistical association with the target variable. The best ranked variables are then selected in order to reduce the computational requirements of the learning algorithm. Usual filter variable selection methods suffer from several drawbacks. For example, multiple variables may all provide the same information about the target (the extreme case occurring when a variable is actually a copy of another). In a univariate ranking approach, all of these variables would be included in the result, even though one is sufficient. Another typical problem occurs when two or more variables are not very predictive when taken individually, but on the contrary, are predictive once used conjointly. These notions can be formalized using information theory, and information-theoretic filters are designed to avoid the exposed pitfalls. Another advantage of such filters is that they do not make any assumption about the statistical distribution on the variables, thanks to the use of

mutual information as a dependency measure. State-of-the-art information-theoretic filters include mRMR (Peng, Long, & Ding, 2005), DISR (Meyer, Schretter, & Bontempi, 2008), REL (D. A. Bell & Wang, 2000), CMIM (Fleuret, 2004) and FCBF (Yu & Liu, 2004). These algorithms vary on whether or not they take complementarity into account, if they avoid the estimation of multivariate density and if they return a ranking of the variable (as opposed to an unordered set of relevant variables). These filters can also be designed to explicitly favor variables having a direct causal influence on the target, thanks to dependence relationships uniquely exhibited by children, spouses and parents of the target. This is the case of the mIMR (Bontempi & Meyer, 2010) and the MIMO (Bontempi et al., 2011) filters.

Bivariate methods

In the last decade, there is an increased interest in telling cause from effect from observational data on just two variables. This task is based on asymmetric properties of the joint distribution of the two variables, and because of the limited number of possible causal configurations, usual classification metrics can be used to assess the performance of inference methods. Moreover, the nature of the problem allows using classical machine learning algorithms for cause/effect classification. This approach gained interest through the organization of public a public challenge on Kaggle (https://www.kaggle.com/c/cause-effect-pairs). This competition resulted in several novel solutions for cause-effect detection, and a general advance in the state of the art. The winner of the challenge, the team *ProtoML*, extracts a large number of features from the variable distributions and uses a large number of models for classification. The second-ranked participant, *jarfo* Fonollosa, 2016, uses conditional distributions and other information-theoretic quantities to infer features that are used for prediction. The general outcome of this challenge is that asymmetries in causal patterns enable the development of bivariate causal distinguishers, with an accuracy significantly better than random.

Supervised methods

In the context of the aforementioned cause-effect pair competition, Bontempi and Flauder (2015) proposed an algorithm using asymmetries in the conditional distributions of the variables. They extended their method to a setting with more than two variables, by also extracting distribution features from other variables and using them to infer the existence and direction of a causal link in the two initial variables. This benefit is striking in the case of a collider configuration $x_1 \rightarrow x_2 \leftarrow x_3$: in this case, the dependency (or independence) between x_1 and x_3 tells us more about the link $x_1 \rightarrow x_2$ than the dependency between x_1 and x_2. By using a learning machine such as random forest, these features can be used to successfully infer causal links, competing with and often outperforming state-of-the-art Bayesian network inference methods.

Chapter 3

Churn prediction

3.1 Data

The data used throughout this work is a monthly summary of customers' activity, mobile data usage in MB, number of calls and messages, along with information about the type of subscription, hardware, and socio-demographic information. This amounts to a total of 73 variables. The dataset comprises one entry per customer and per month, and a total of 5 months are present from the year 2018. About 1.5 million entries are present per month, for a total of about 7.6 million entries in the entire dataset. The target variable, churn, is represented as a date if the client is known to have churned, or an empty value otherwise.

Two kinds of contracts are present in this dataset, that we will call *SIM only* and *loyalty*. The first type refers to a subscription where the customer can churn freely at any time. This differs from the second type where the customer receives a large discount on the purchase of a mobile phone but agrees not to churn for a certain period of time, usually 24 months. If the customer decides nonetheless to stop his subscription before the term of the contract, she has to pay back the remaining discount. After the data preprocessing step (presented in section 3.2), we are left with a total of about 5 millions entries corresponding to SIM only contracts, and about 250,000 entries corresponding to loyalty contracts. Therefore, we will mainly focus on SIM only contracts, for its broader impact on the customer base and its increased statistical significance. Some of the experiments will nevertheless be conducted on both types of contracts, in order to understand the differences in the churn dynamic.

Note that the number of entries in the loyalty dataset does not correspond to the real number of customers having this type of contract. This is due to the way the loyalty data entries are filtered. We focus only on customers approaching the end of the mandatory period of the contract, since it is at this point that the churn rate raises. Before this period of time, the churn is almost non-existent, due to the remaining discount to be paid back by the customer. We consider a time frame of 2 months, explaining the low number of data entries under consideration.

Due to commercial reasons, a non-disclosure agreement prevents us from communicating precise details on the variables being used or the rate of churn in the

customer base of Orange. The churn prediction problem is highly imbalanced, but we cannot disclose the exact ratio between churners and non-churners. The name of the different variables is limited to a letter corresponding to one of the 6 aforementioned categories and a number to differentiate variables among a category. We are able to disclose the exact name of a variable when its meaning is relevant for the discussion and its disclosure would not have a negative impact on confidentiality.

The 73 variables are grouped into 6 categories. We denote each variable by a letter corresponding to its category, and a number to differentiate it from others in the same category. The different categories are

- Subscription (17 variables, initial S)
- Calls and messages metadata (11 variables, initial C)
- Mobile data usage (16 variables, initial U)
- Revenue (14 variables, initial R)
- Customer hardware (6 variables, initial H)
- Socio-demographic (5 variables, initial D)

The 4 remaining variables are the churn date, the timestamp of the data entry and two customer identifier columns.

Most variables are continuous, such as the duration of phone calls, or the amount paid on the last bill. There are however a few discrete variables, either taking integer or categorical values. Such variables include the province of residence of the customer or the number of active contracts. We present in this section a descriptive analysis of some variables, in order to understand, prior to any machine learning modeling, how they are distributed, and how they interact with each other. General patterns are qualitatively discussed in the text, and the figures support the discussion whenever possible.

One of the categorical variables represents the proportion of customers having a cable subscription besides their mobile phone subscription, such as for television or landline phone. There are fewer cable subscriptions among churners, and this fact is well known for Orange Belgium: a client having the cable will be less willing to churn, since this represents a significant investment in money and time. Another variable corresponds to the payment type for the bill. The two main types are automatic debit and bank transfer. We observe that more people among the churners chose the bank transfer. Although this is only speculation, this might be caused by the "bill shock" effect: when a client consumed more calls, messages or data than provisioned by its tariff plan, she has to pay an extra amount of money called out-of-bundle (OOB). When this amount is large, the client is more likely to get upset, and therefore is more likely to churn. But if her invoice is paid with an automatic debit, she may not notice this fact straight away, and this can thus be a reducing factor of churn.

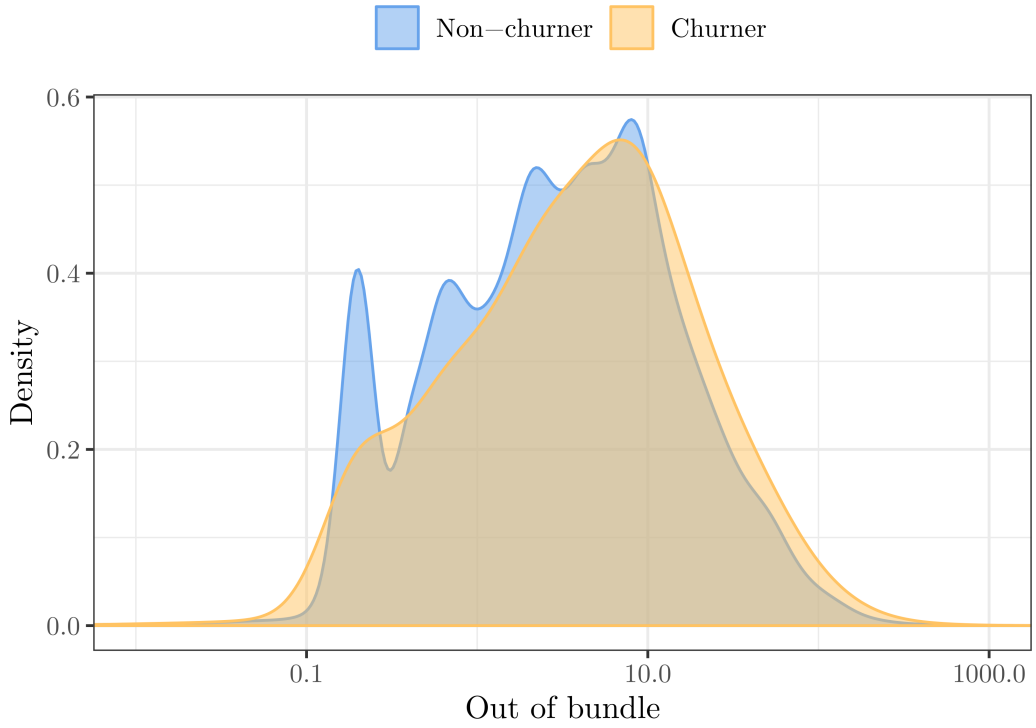


Figure 3.1 – Out-of-bundle amount (extra amount to pay on top of the usual invoice), in logarithmic scale.

The bill shock effect is demonstrated in a more straightforward way in figure 3.1. The distribution of OOB is plotted for churners and non-churners on a logarithmic scale. Note that the probability density estimation function used in this figure and in figure 3.2 uses a Gaussian kernel. This results in a distribution resembling a mixture of Gaussian, even though the underlying probability density may not be Gaussian. There is a clear discrepancy between the distributions of churners and non-churners in figure 3.1, with churners having more often a large OOB. This fact is often used to establish expert rules when conducting churn retention campaign: if a customer is likely to churn and has a large OOB, then a tariff plan more adapted to her usage profile is proposed. Another possible action to be taken in such case would be to offer a discount on the invoice in order to directly mitigate the bill shock. However, this approach is a short term solution, and the customer is likely to have a large OOB on the following invoices.

The importance of the tenure (the duration of the current subscription) is shown in figure 3.2. This graph displays the density of clients as a function of the tenure. For confidentiality reasons, the scale of the x-axis is hidden. The curve can be divided into two components: new customers and long-term customers. One can clearly observe that proportionally more churners are present in the first component than in the second. This indicates that long-term clients tend to churn less, whereas new clients are much riskier.

Figures 3.3 and 3.4 show the distribution of two discrete categorical variables. The first variable, R14, is a binary flag related to revenues and is slightly less often true among churners. The second variable, H8, is a categorical variable related to

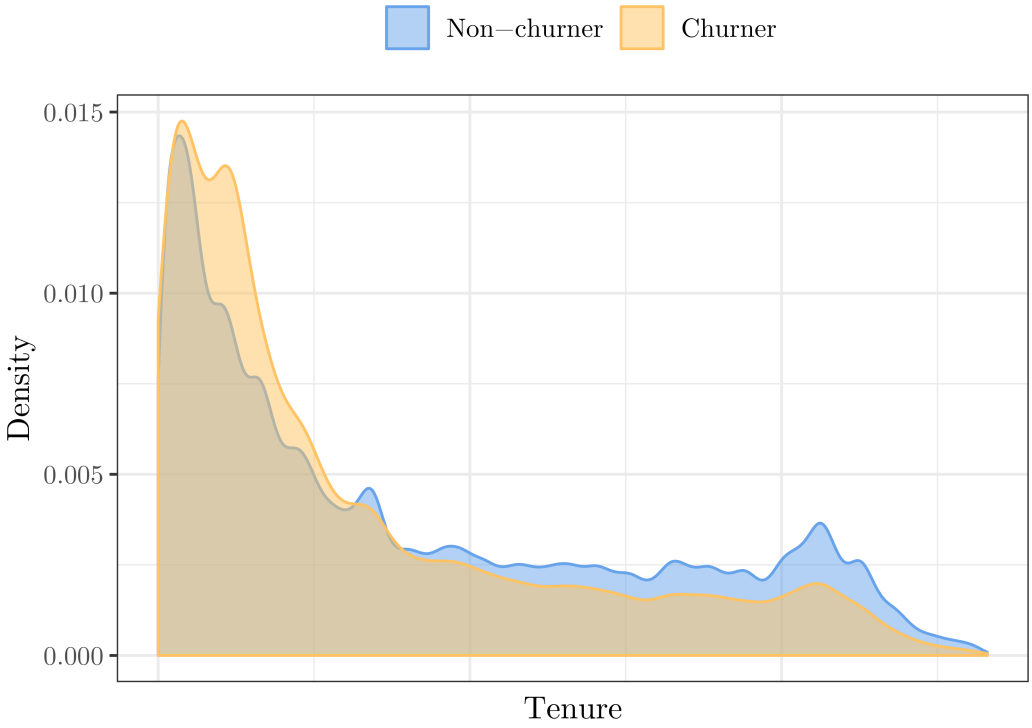


Figure 3.2 – Tenure (time spent without churning so far) for churners and non-churners.

hardware and takes 8 different values. The churn rate varies moderately depending on the value of this variable.

We demonstrate the interaction between two categorical variables in figure 3.5. The horizontal axis indicates whether a customer has a cable connection, and the vertical axis denotes the payment responsible flag. This flag is set to false only when someone else pays the bill of the customer, such as a parent. Most customers of Orange Belgium do not have a cable connection, and are responsible for the payment, as indicated by the radius of the spots. The color of the spots indicates the churn rate, with a lighter color denoting a higher probability of churn. The area is proportional to the number of clients in each category. The impact of both binary variables appears clearly, with a significant difference of churn rate between the two extrema. Once again, the precise value of churn rate cannot be disclosed.

A principal component analysis (PCA) demonstrates the important overlap between churners and non-churners (figures 3.6 and 3.6). The blue corresponds to non-churners, while the orange and yellow represent the churners respectively in the validation and the test set. We explain how the test set and the validation set are partitioned in section 3.3. The ellipses represent the contour lines of covariance, that is, the set of points at a Mahalanobis distance of 1 from the mean in each set. The mean of each set is pictured as a dot in the center of the figure. A large overlap between the population of churners and non-churners appears clearly. Also, the standard deviation is larger in the population of churners, reflecting the interpretation that churn is associated with larger values for the out-of-bundle amount, number of calls, data usage, etc.

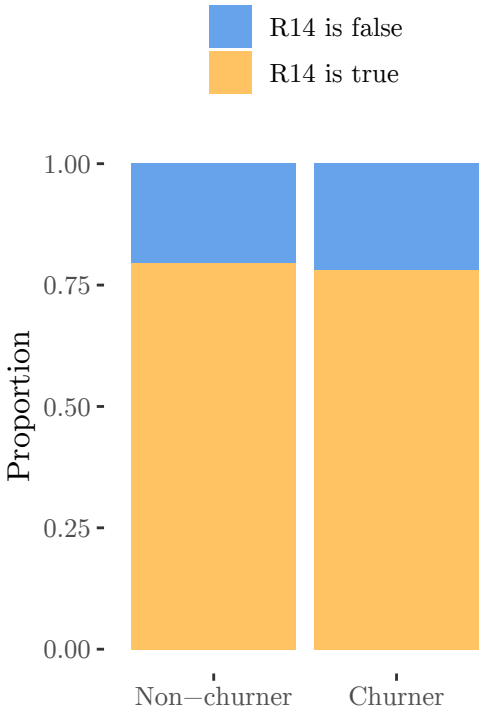


Figure 3.3 – Distribution of a binary variable related to revenues, R14, for churners and non-churners.

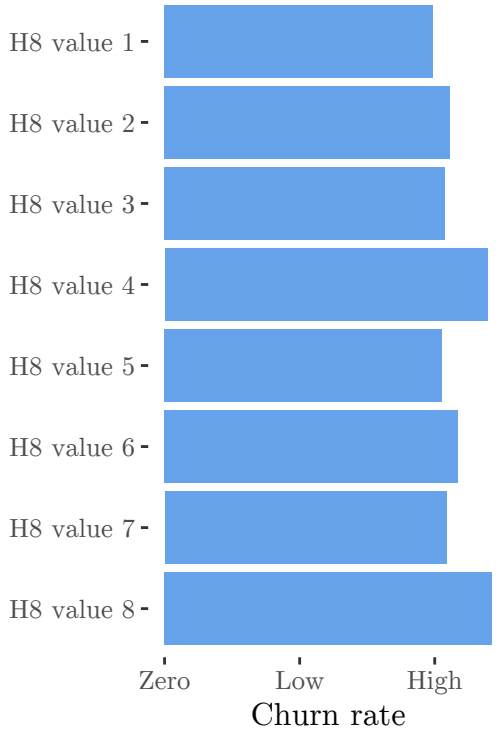


Figure 3.4 – Distribution of churn rate depending on a categorical variable related to hardware, H8.

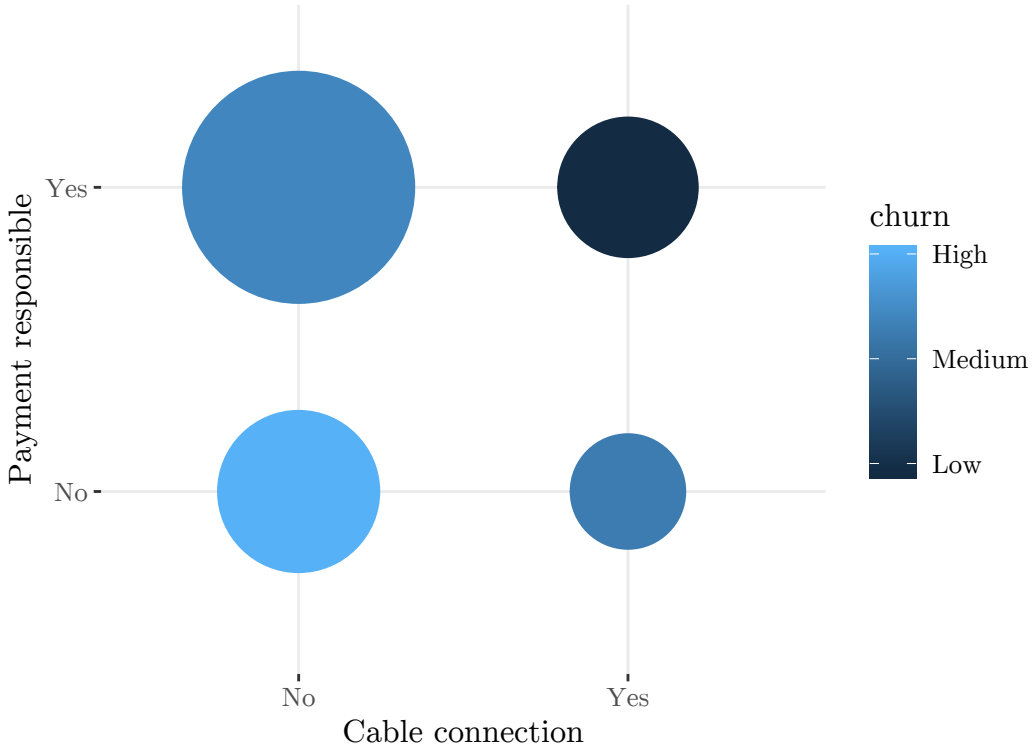


Figure 3.5 – Interaction between cable connection and payment responsible. A customer is not responsible for payment if someone else (e.g. a parent) pays the invoice in her stead. The color of the spots denotes the churn rate, whereas its area denotes the number of customers.

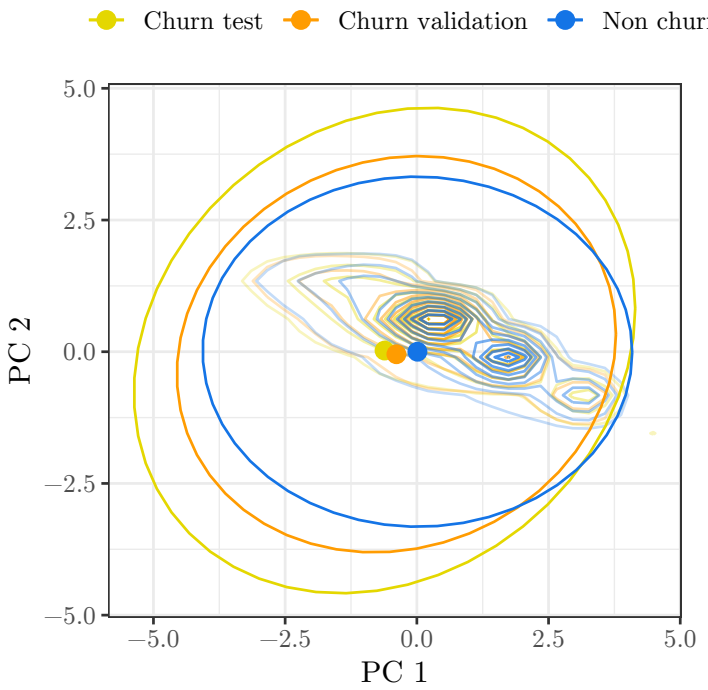


Figure 3.6 – Projection of the dataset onto the first two principal components. The ellipses show the set of points at a Mahalanobis distance of 1 from the mean of each group, which are represented by a dot.

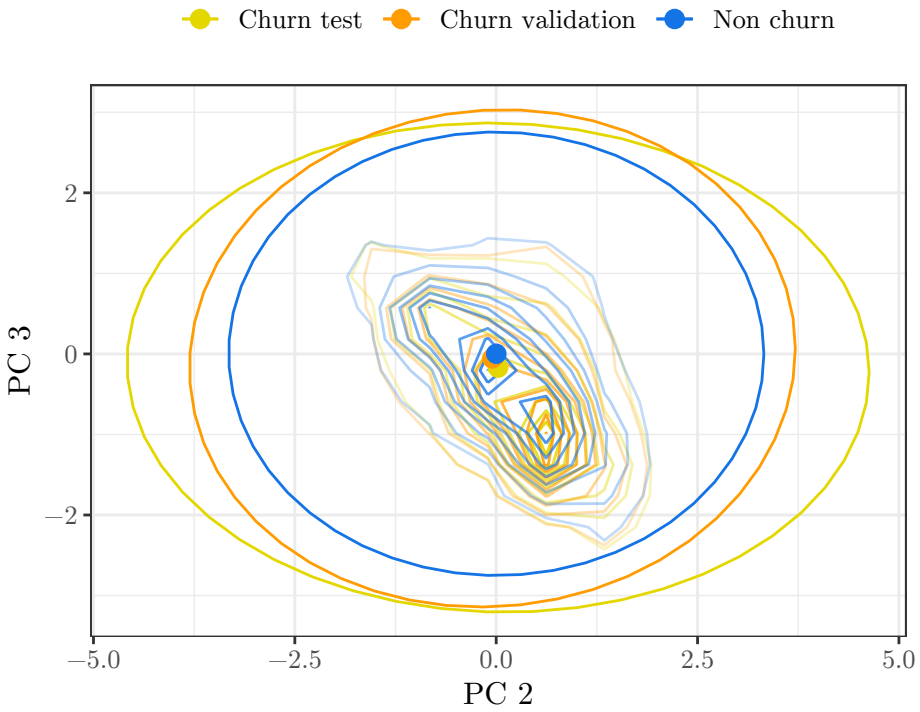


Figure 3.7 – Projection of the dataset onto the second and third principal components. The ellipses show the set of points at a Mahalanobis distance of 1 from the mean of each group, which are represented by a dot.

This data analysis demonstrates the high complexity of the churn prediction problem. No unique variable allows to unambiguously predict churn since there is a significant overlap between the population of churners and non-churners. Informative variables such as the tenue (figure 3.2) or the out-of-bundle amount (figure 3.1) allow to increase or decrease the confidence into churn only marginally. A large number of variables must be used in conjunction in order to achieve decent predictive performances. Moreover, we have no guarantee that the set of available variables are sufficient, it is easily conceivable that unknown factors, such as a promotion launched by a concurrent company, play a significant role.

3.2 Data preparation

The data preparation is of a sequence of steps.

Unknown values preprocessing The original data uses various means to specify an unknown categorical value. For example, an unknown previous tariff plan is either represented by an empty string, the string `"u"`, the string `"null"` or a null value. This preprocessing step replaces these various encodings by a unique one. Also, missing values in continuous variables are either replaced by zero or by the mean of the variable, depending on the semantics (e.g. null data usage is replaced by zero, whereas missing age is replaced by the average age in the dataset).

Date encoding Some fields are represented by a date, such as the last contract change, the date of contract activation, or the churn date. These fields are converted to the number of days between the first day of the month of the dataset entry and the field value. For example, let us consider an entry about the activity of a customer in January 2019. Say this entry contains a field with the contract activation date, with value `"20 December 2018"`. This field is converted to an integer value of 12, since there are 12 days between 1st January 2019 and 20 December 2018.

Clustering of character strings There are three character string variables, representing the current and the previous tariff plan, and the manufacturer of the customer's device. These three variables could be considered as categorical variables, but the high number of different values would make this difficult to implement. We alleviate this difficulty by clustering all the different values into a small number of groups. In the case of the two tariff plan variables, this corresponds to the different tariff plan options (*Hummingbird*, *Koala*, *Eagle*, etc). For the device manufacturer, we keep the 7 most common values, and we replace all the less frequent values by "Other".

Difference and ratio columns For each numerical field representing a quantity that can change from month to month (such as the total duration of calls, or the mobile data usage), we create 2 additional fields. They contain the

difference and the ratio of the value of the field with that of the same field the previous month. This hopefully gives the model an indication of the customer's behavior evolution over the course of the last month. 41 variables are suitable for this operation, therefore increasing the number of variables up to 155. If no data is available for the previous month (such as for the first month of data), the differences are set to 0 and the ratios are set to 1. In order to reduce computation time, not all experiments use these new columns, as discussed in the next section. This augmented dataset is named "SIM only Δ" thereafter.

Normalization The data is normalized to obtain zero mean and unit variance. Even though the only models being used are random forests, which are not sensitive to linear scaling of its input variables, this step is kept in the event that we would have tried another model requiring normalization (such as support vector machine or neural networks). This preprocessing step is also useful for sensitivity analysis, where we add a small value to a variable and observe the difference in the predictions of the model. In this case, a normalized dataset allows to use the same difference value for all variables.

Target variable The last step is to create a binary target variable from the churn date. It is defined to be true if and only if the date of churn is in the two months following the current data entry. If the churn date is in the current month or before, then the entry is discarded, for two reasons. Firstly, the information contained in these entries is incomplete. Secondly, this data is not relevant for churn prediction, as we wish to predict churn at least a few days in advance. It is not interesting to learn patterns exhibited by clients that will churn the next day, as there is probably no longer any hope of successful retention. If the churn date is not given, or if it is more than two months after the month of the data entry, the churn variable is set to false. This process is pictured in figure 3.8. The choice of the time threshold is dependent on the business application. A lower threshold focuses on short-term churn, whereas a larger threshold enables to predict churn from further in the future. Models already in use at Orange Belgium consider a time period of two months, we therefore use this value in order to enable the comparison of our results with production models.

3.3 Experiments

Scope

The experiments on predictive modeling consist in the training of predictive models on the data described in the previous sections, and an assessment of their performance. Three datasets are derived from the output of the preprocessing step: one containing the loyalty contracts, one containing the SIM only contracts, and one

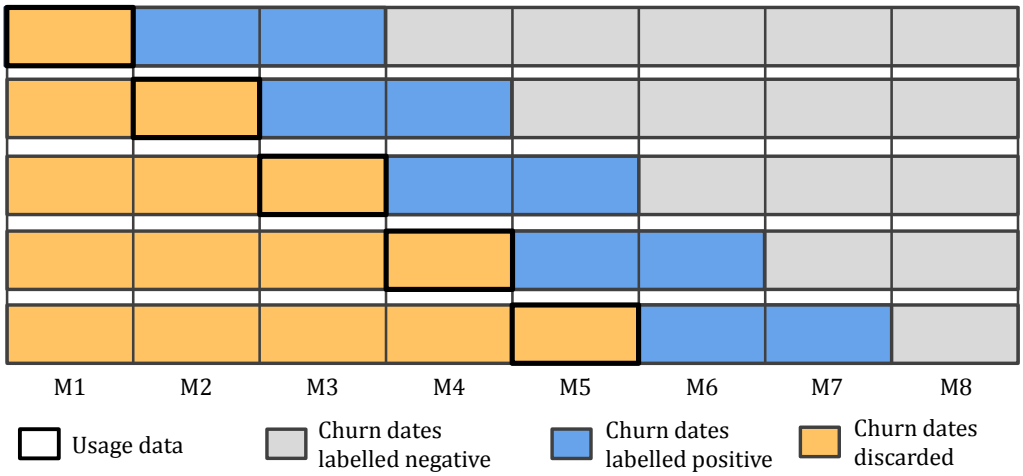


Figure 3.8 – Outline of the target variable assignment. The dataset is separated in 5 months and each data entry in each month is labeled as churner if the churn date is given and less than two months ahead.

containing the SIM only contracts with difference and ratio variables (called SIM only Δ). We evaluate the impact of

1. variable selection, based on the feature importance provided by trained random forest models;
2. the addition of difference and ratio variables;
3. the type of contract (SIM only vs. loyalty).

The high computational cost of the model training on such a large dataset does not allow to test all the possible configurations of these three parameters. We limited the number of selected variables to 20, 30 or all variables. Also, we do not explore the difference variables for loyalty contracts. These combinations of parameters yield 9 different experiment configurations.

Data segmentation

In each configuration, the corresponding dataset is split into a training and a test set. The training set comprises the first 4 months of data and the test set comprises only the last month, as pictured in figure 3.9. Separating by months allows for a potential concept drift from the training set to the test set (i.e. a change in the typical behavior exhibited by churners). We perform a k-fold cross-validation on the training set in order to provide an indication of the performance of our model on the training data. We set $k = 3$, as a compromise between statistical significance and computation time. The difference in prediction accuracy between the validation set and the test set indicates how much patterns learned on the training set are still relevant for the next month. This indicates whether training has to be repeated each month as new data arrives from the customers. Note that when testing the model on the test set, a new model is trained on the whole training set.

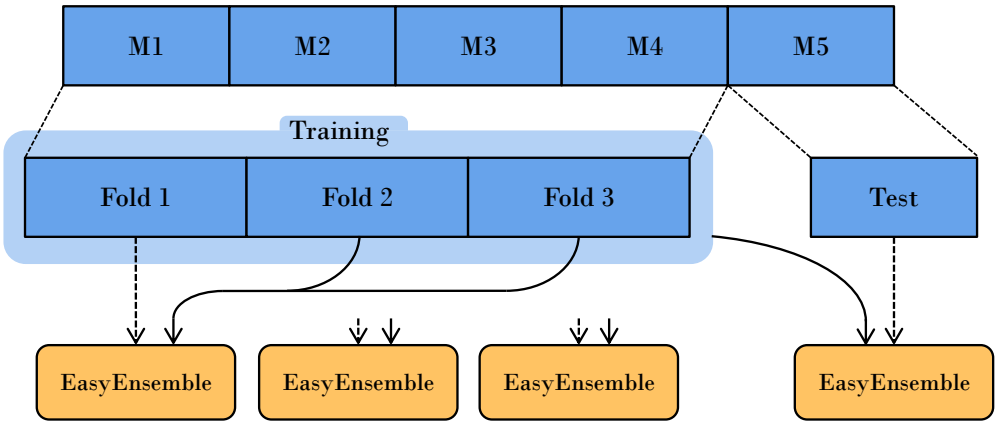


Figure 3.9 – Outline of the data repartition between training and test set, with a 3-fold cross-validation on the training set. Dotted arrows indicate testing and solid arrows indicate training. Arrows for two of the three validation models are not shown.

Class balancing

In order to counteract the sheer prominence of non-churners in the dataset, we need to use class balancing. We use the EasyEnsemble algorithm, presented in section 2.1. It consists in training different models on the whole set of positive instances, and on a randomly selected set of negative instances. The number of negative instances is chosen so that the ratio between the two classes is even. In all experiments, the EasyEnsemble is set to use 10 random forest models.

Evaluation measure

The performance of the different models is evaluated using three different measures: the lift curve, the receiver operating characteristic (ROC) curve, and the precision-recall (PR) curve. While the ROC curve and the PR curve are widely used in the machine learning literature, the lift curve is of more practical interest in evaluating churn prediction. Since a customer churn retention campaign focuses on a limited amount of customers, the lift curve allows observing the expected performance of the model as the number of customers included in the campaign varies. From the ROC and PR curves, we derive the area under the ROC curve (AUROC), the area under the PR curve (AUPRC) and the lift at different thresholds (1%, 5%, and 10%).

We argue that the most sensible cost evaluation functions are the maximum profit criterion (MPC) and the expected maximum profit criterion (EMPC) (Verbeke et al., 2012; Verbraken et al., 2013). These take into account the different costs and benefits yielded by a retention campaign and provide the decision threshold that should be applied to maximize the profit given the probability distribution output by a prediction algorithm. This process formalizes the intuition behind the lift criterion that the prediction algorithm should focus on reducing false positives, since the retention campaign is not able to reach each and every potential churner.

Despite the relevance of this approach under a profit-centric point of view, we are not able to use this evaluation measure in our study. This is caused by the necessity to evaluate different costs and benefits parameters, such as the cost of reaching a customer, the probability that a customer accepts an incentive, the benefit if the customer accepts it, etc. The evaluation of these parameters is a time-consuming process and is outside of the scope of our work.

Sensitivity analysis

The impact of variables on churn prediction is derived in two different ways. The first corresponds to the variable importance output by the random forest models. Each random forest calculates a score for each variable by measuring how much the prediction accuracy decreases when all the values of this variable are randomly permuted. The decrease in accuracy is calculated over the out-of-bag samples in each tree. This permutation cancels out any statistical dependency between this variable and the target variable, giving an estimate of the importance of the variable in the trained model. Note that if two variables share the same information about the target (for example by being highly correlated), the importance of both of these variables will be less than if only one were present. This is due to the fact that the two variables are equally likely to be chosen when splitting nodes in a tree, therefore reducing the impact of the removing one of the two variables when computing the importance.

This measure of importance allows to understand the predictive power of each variable but does not indicate the directionality of its impact on the predictions. We address this issue by constructing, for each variable X_i, an alternate training set identical to the original one, but where a value equal to one standard deviation σ_{X_i} is added to each instance of the variable X_i. A second shifted dataset is also constructed by subtracting instead of adding the standard deviation. Then, the average predicted probability of churn is computed for both the original training set and the shifted one, and the difference between the two average probabilities is taken. This difference indicates the impact of the variable on the predictions. For example, a predicted churn probability lower for the training set where a standard deviation is added to the tenure variable indicates that longer-standing customers are associated with less churn. Note that we use in this experiment the normalized dataset so that adding a standard deviation amounts to adding 1 to the instances of the variable.

3.4 Results

This section shows the results of the predictive experiments. Figures 3.10 to 3.18 are performance curves for the three different datasets. Each plot contains a curve for both validation and test sets, and for the configurations where we select 20, 30 or all of the variables. This amounts to a total of 6 curves per plot. As explained in section 3.3, the validation is done on the 4 first months of data, whereas the test

	SIM only			SIM only Δ			Loyalty		
	20	**30**	**All**	**20**	**30**	**All**	**20**	**30**	**All**
AUROC	0.66	0.73	0.73	0.72	0.73	0.69	0.74	0.76	0.76
AUPRC	0.05	0.10	0.10	0.10	0.10	0.08	0.15	0.19	0.18
Lift at 10%	2.25	3.34	3.41	3.27	3.42	3.03	2.96	3.40	3.30
Lift at 5%	2.64	4.49	4.68	4.48	4.67	4.09	3.51	4.22	4.02
Lift at 1%	4.29	9.20	9.53	10.09	9.95	7.67	4.66	6.65	6.16

Table 3.1 – Summary of the results of prediction experiments on the test set. Highest values for each type of contract and for each evaluation measure are underlined for the test set.

set corresponds to the last month. Figures 3.10 to 3.12 show the lift curves, figures 3.13 to 3.15 show the ROC curves, and figures 3.16 to 3.18 show the precision-recall curves. A summary of the results is given in table 3.1. The lift at different thresholds, the area under the ROC curve, and the area under the precision-recall curves are reported for the test set. Table 3.2 provides the same information for the predictions on the validation set. The impact of the different experimental parameters on predictive performance is discussed in the next sections.

In this section, numerical scores associated with variables are displayed as horizontal bar plots. The colors of the bars correspond to the categories of variables presented in section 3.1:

- Subscription
- Calls and messages
- Mobile data usage
- Revenue
- Customer hardware
- Socio-demographic

Number of variables

The number of variables has an influence on the prediction accuracy, and this impact depends on the dataset under consideration. Recall that in each configuration, the selected variables are chosen according to the variable importance given by the random forests trained on the whole training set. In the case of the SIM only dataset (figures 3.10, 3.13 and 3.16), selecting only 20 variables decreases drastically the performance. Selecting 30 variables achieves performances almost as good as selecting the whole set of 73 variables.

For the SIM only Δ dataset, a lower number of variables is beneficial for performance. As shown in figures 3.11, 3.14 and 3.17, selecting all variables appears to

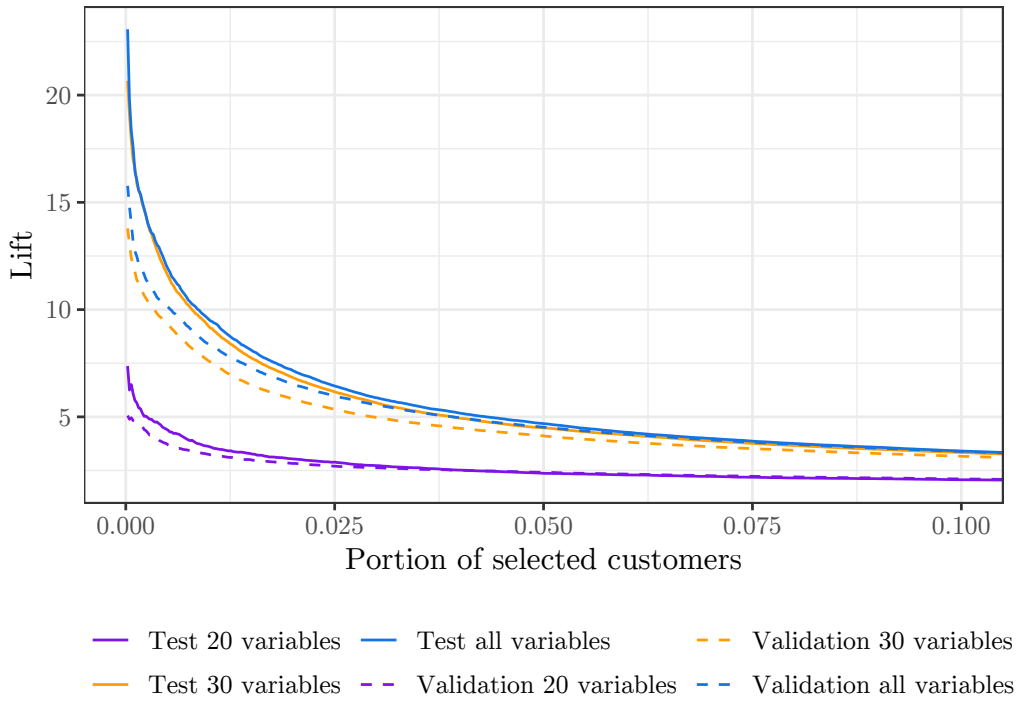


Figure 3.10 – Lift curve for SIM only

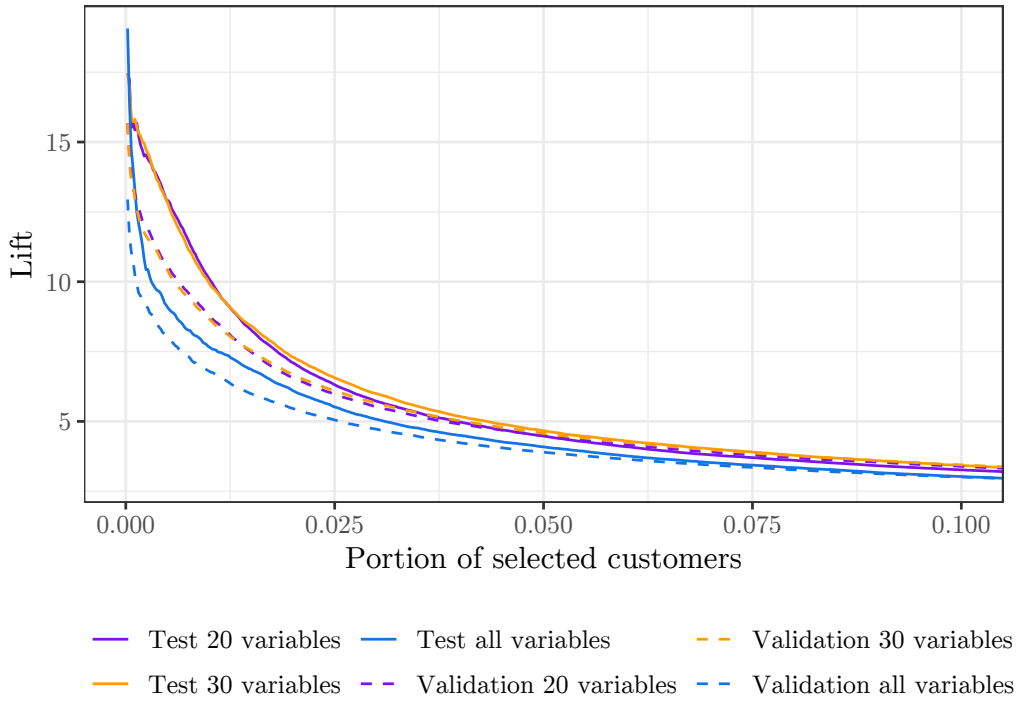


Figure 3.11 – Lift curve for SIM only with difference and ratio variables

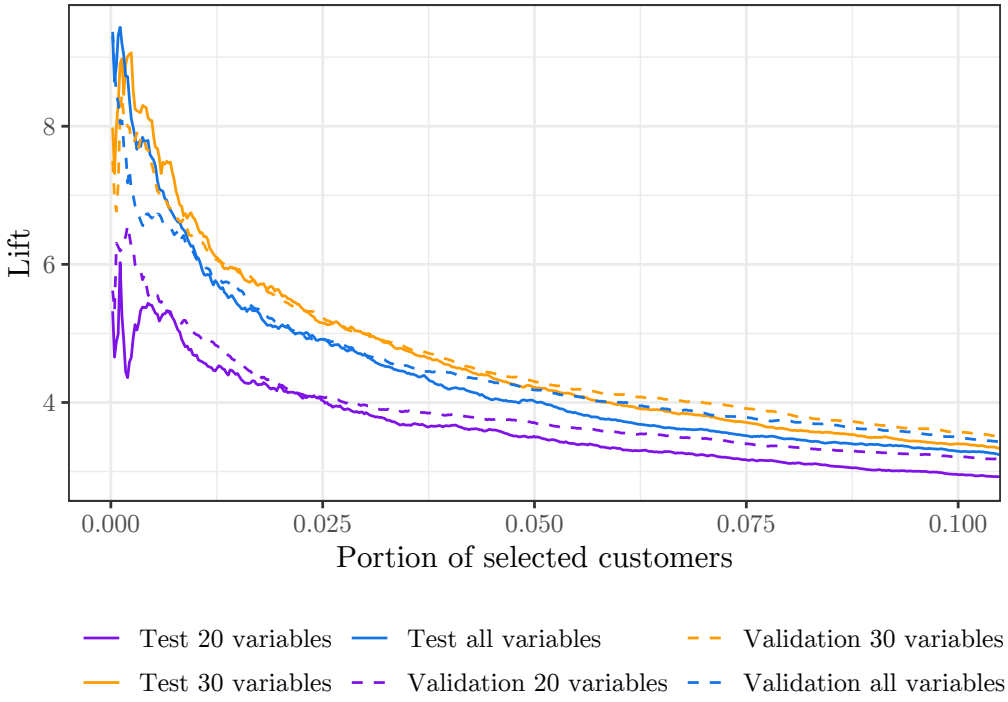


Figure 3.12 – Lift curve for loyalty

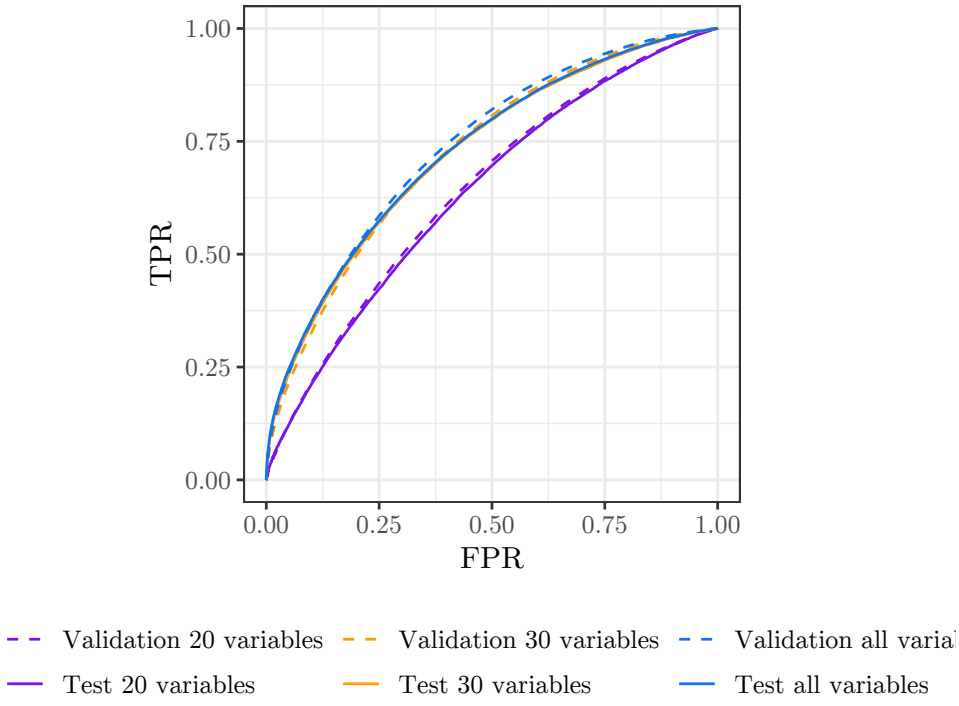


Figure 3.13 – ROC curve for SIM only.

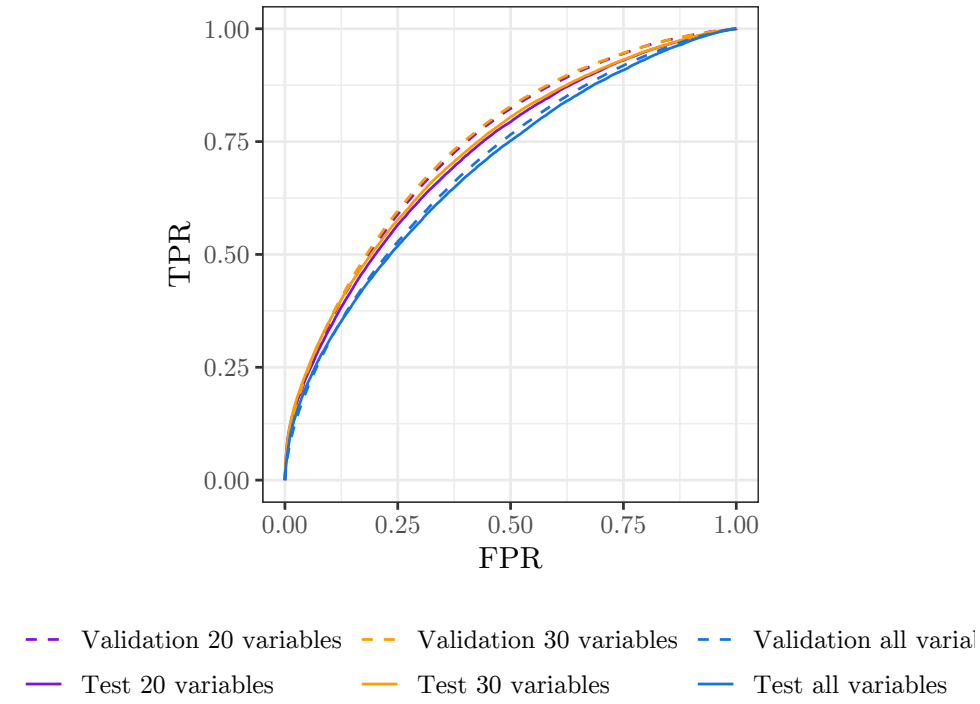


Figure 3.14 – ROC curves for SIM only with difference and ratio variables.

be detrimental to the accuracy, whereas choosing between 20 or 30 variables does not make a significant difference. This might be caused by the additional variables that add more noise than useful information for the random forests. It is interesting to note that selecting the top 20 variables when considering the SIM only Δ dataset provides much better accuracy than selecting the top 20 variables in the original dataset. As shown in figure 3.20, the 20 most important variables in the SIM only Δ dataset do not even include any difference or ratio variable. The difference between the top 20 most important variables is as follow:

- SIM only Δ includes the number of contracts, the age, and a variable on data usage, whereas SIM only does not;
- SIM only includes the device manufacturer, the previous tariff plan, and two other variables on data usage, whereas SIM only Δ does not.

The large difference in accuracy between these two configurations must be caused by this difference in the selected variables, since all other variables and all other experiment parameters are identical.

When considering the loyalty dataset, selecting 30 variables instead of the whole set of 73 variables is marginally beneficial on low thresholds (less than 0.1). On the overall space of thresholds, the difference is not significant, as indicated in table 3.1 with the AUROC and the AUPRC. However, selecting only 20 variables is clearly detrimental.

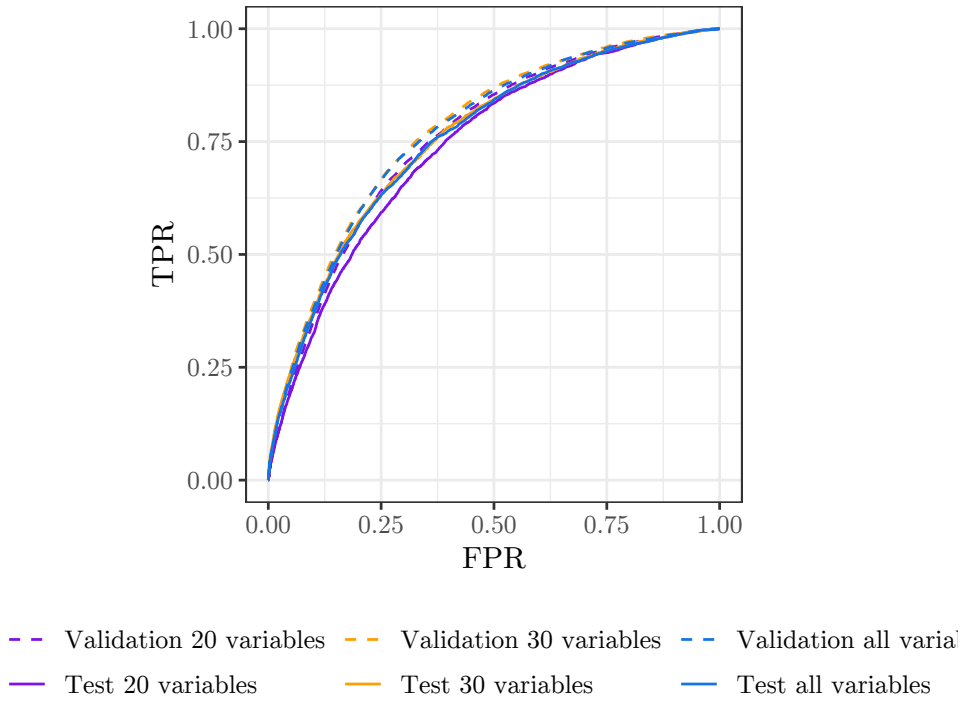


Figure 3.15 – ROC curves for loyalty.

Difference and ratio variables

The difference and ratio variables do not have a positive impact on performance. Indeed, the best performance achieved on the SIM only Δ dataset are obtained by limiting the number of variables to 20 or 30. The variables being selected in these cases do not include any of the difference and ratio variables. If all variables are used, the performance of the random forests decreases significantly, as shown in figures 3.11, 3.14, 3.17, and table 3.1. Moreover, the memory usage of this dataset is much higher than that of the original dataset, thus complicating the training process.

Generalization performances

The generalization abilities of the trained models are evaluated by comparing the accuracy on the validation set and on the test set. On the lift curves and on the precision-recall curves, it appears that for all configurations, the performance on the test set is better than on the validation set. Bear in mind that on these curves, only a small fraction of the threshold space is represented, whereas the ROC curves show all possible decision thresholds. This observation implies that customers with a very high probability of churn are proportionally more numerous in the test set than in the validation set. It is illustrated in figures 3.6 and 3.7, where the ellipse of covariance of the test set is larger than that of the validation set. This suggests that, in the test set, there are more churners with very high values for variables having a large standard deviation. This probably corresponds to the bill shock effect discussed in section 3.1: a large out of bundle amount, caused by large data

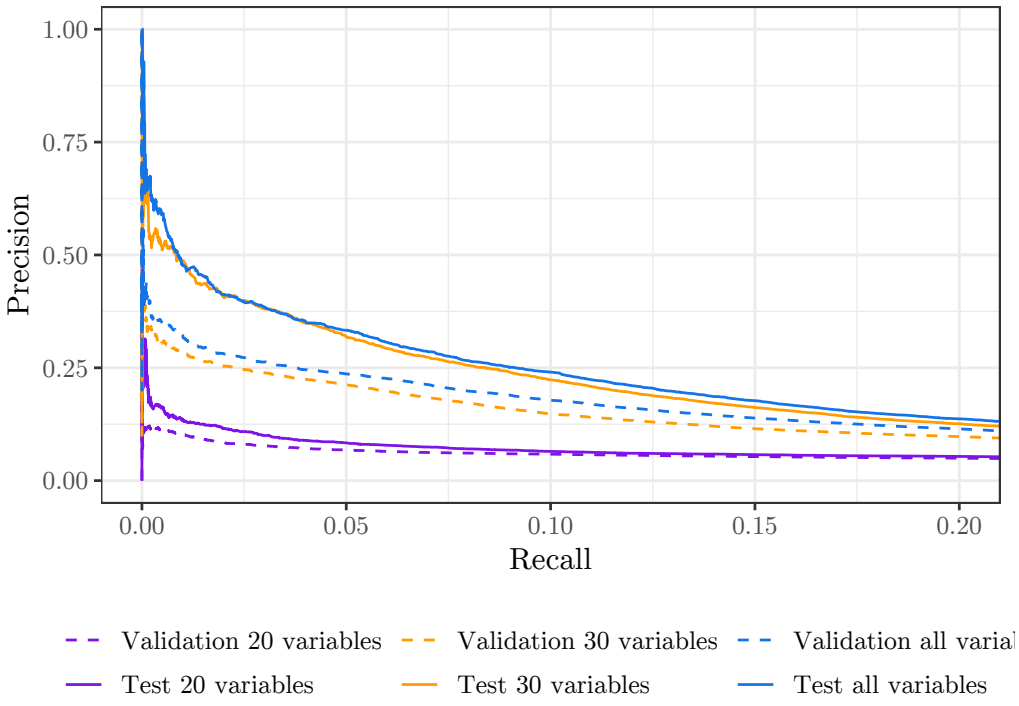


Figure 3.16 – Precision-recall curves for SIM only.

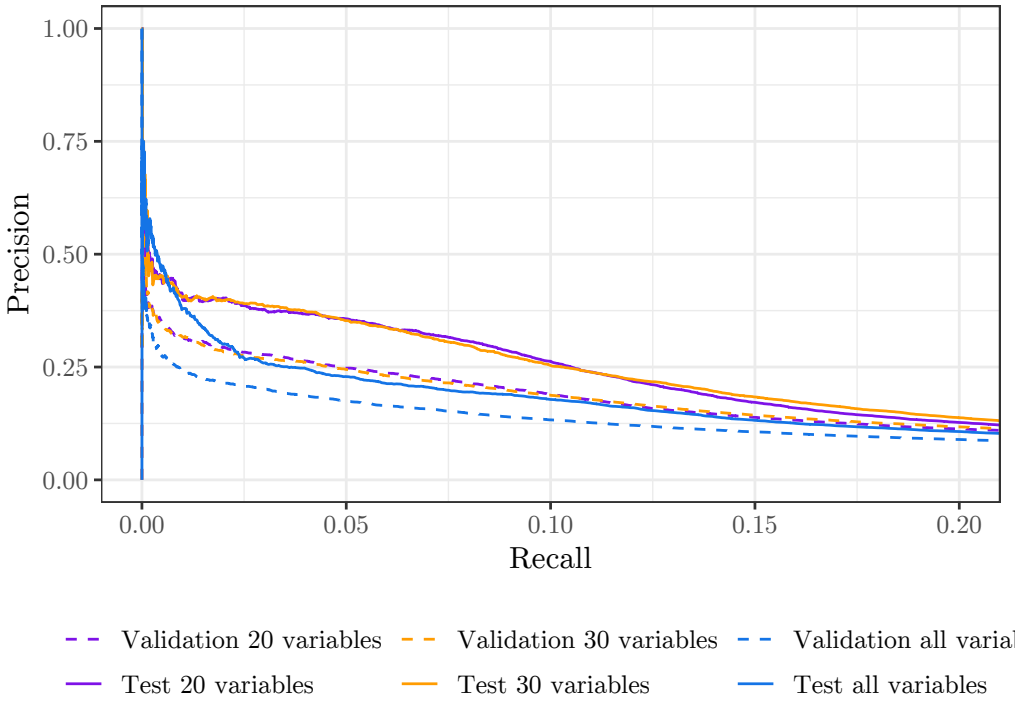


Figure 3.17 – Precision-recall curves for SIM only with difference and ratio variables.

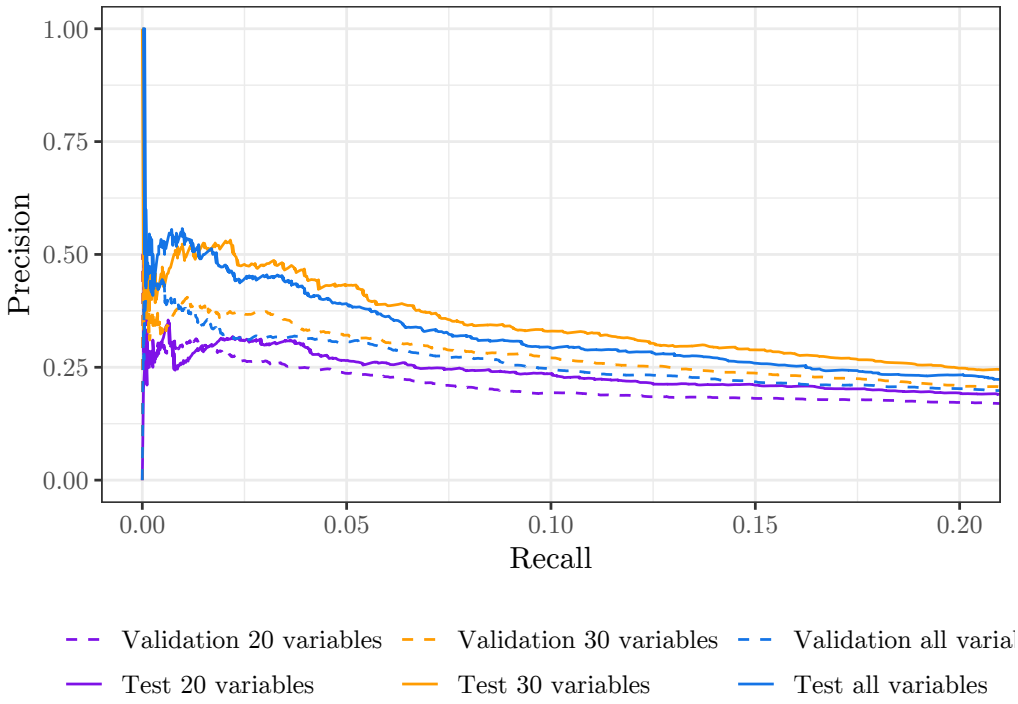


Figure 3.18 – Precision-recall curves for loyalty.

	SIM only			**SIM only Δ**			**Loyalty**		
	20	**30**	**All**	**20**	**30**	**All**	**20**	**30**	**All**
AUROC	0.64	0.73	0.74	0.74	0.74	0.70	0.76	0.78	0.77
AUPRC	0.04	0.08	0.08	0.09	0.09	0.07	0.13	0.16	0.15
Lift at 10%	2.10	3.16	3.39	3.39	3.44	3.01	3.22	3.57	3.50
Lift at 5%	2.41	4.11	4.52	4.49	4.57	3.90	3.71	4.30	4.18
Lift at 1%	3.24	7.58	8.36	8.80	8.67	6.79	5.00	6.37	6.11

Table 3.2 – Summary of the results of prediction experiments on the validation set.

consumption, increases the probability of churn. In this case, the bill shock is more pronounced in the test set, explaining the improvement in predictions.

The performance measures on the validation set are summarized in table 3.2. When we compare the lift at low thresholds in table 3.1, it is clear that the model manifests worse performance on the validation set than on the test set. However, it is not the case for the AUROC and the AUPRC, which take into account the whole space of decision threshold, and not only the riskiest customers. We can conclude that our model has been trained on a training set where the overlap between churners and non-churners is more important than on the test set. The model thus generalizes well, and even perform better on unseen data in our case due to a lucky domain shift.

Type of contract

As indicated in table 3.1, the models trained on loyalty perform slightly worse than that of the SIM only datasets on small thresholds, but better on larger thresholds. The AUROC is equal to 0.76 for loyalty, whereas the best performing configuration for SIM only achieves an AUROC of 0.73. The AUPRC is almost double, and this can be seen in figure 3.18. The precision is similar to that of the SIM only for low recall, but decreases much more slowly. It is still at approximately 0.25 when the recall is 0.2, whereas, in the SIM only PR curves, the precision is already at 0.12 at this threshold.

Recall that there are fewer loyalty customers than SIM only, less confidence can thus be given to the statistical results for loyalty, especially at low thresholds. According to these results, the models trained on the loyalty customers are slightly less efficient on small thresholds, but this is made up for on larger thresholds. The increase in performance is probably due to the more obvious churn patterns exhibited by loyalty customers. Indeed, most of the churn in this population is due to the end of the mandatory period of the subscription. This is reflected in figure 3.21, where time-related variables are prominent in variable importance. Given that we know when the mandatory part of the customer's contract ends, the confidence of the model in the probability of churn is increased compared to the SIM only case.

Sensitivity analysis

The results of sensitivity analysis are shown in figures 3.19 to 3.23. Figures 3.19 to 3.21 show the variable importance given by the random forest models. There is one plot per dataset, and in each plot the importance of each variable is averaged over the 10 models underlying the Easy Ensemble meta-model. As discussed in the previous section, the most important variables for the SIM only and the SIM only Δ datasets are almost identical. They consist in a mix of socio-demographic variables (e.g. the province), information about the tenure and the tariff plan of the customer, and aggregate variables related to phone calls. The difference and ratio variables are ranked fairly low for SIM only Δ, the first one having a rank of 40 (not shown in figure 3.20). On the other hand, the selected variables for the loyalty dataset (figure 3.21) are quite different. The tenure (first and third variables) and other time-related variables on the subscription are all important variables. Information relative to the type and time of subscription is therefore important for predicting churn in loyalty contracts. This is illustrated by the yellow color dominating the graph. Also, the age is more important than for the SIM only dataset, as well as the variables U1, U2, and U3, corresponding to data usage. This is consistent with an interpretation of a younger and fickler customer base, consuming more data and more prone to churn.

Figures 3.22 and 3.23 display the shift in predicted churn probability when a variable is offset by one standard deviation. Figure 3.22 corresponds to an increase in the value of each variable, whereas figure 3.23 corresponds to a decrease. The tenure and the number of contracts have a symmetric effect: an observed increase

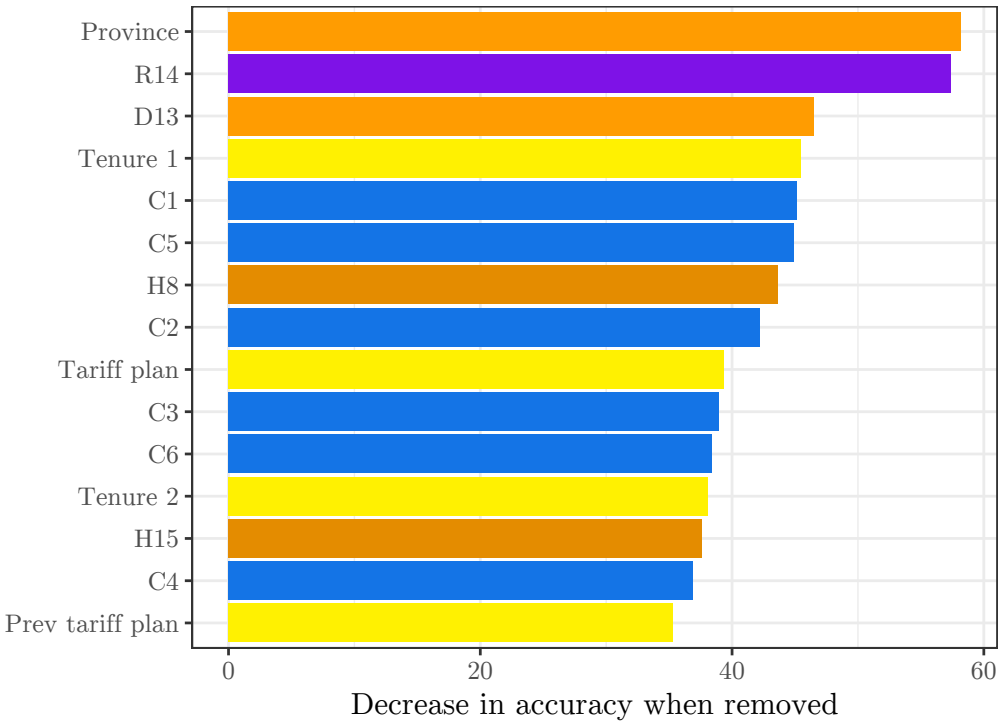


Figure 3.19 – Mean decrease in accuracy when a feature is removed for SIM only.

in their value reduces the probability of churn, and conversely. For the tenure, this corresponds to the intuition that a newer customer is more prone to churn. The number of contracts is a marker of commitment of a customer to Orange, and the presence of this variable in both figures 3.22 and 3.23 expresses that it has also a monotonic relation to churn probability.

On the other hand, all other variables in this sensitivity analysis display a non-linear relationship to predicted churn probability. This is remarkably illustrated by the prominence of variables related to revenues in figure 3.22, colored in purple. These variables increase the predicted probability of churn when they are increased, as expected by the bill shock effect. However, they are absent from figure 3.23, indicating that a reduced bill is not associated with a reduced churn. It is worth mentioning that the age is associated in both graphs with an increased risk of churn. Consequently, any age far from the average age is associated with an increased risk of churn.

Bear in mind that this analysis is solely indicating statistical associations between the values of different variables and the predicted probability of churn. It is by no means an indication of causality. For example, the number of contracts is inversely associated with the risk of churn. It is tempting to conclude that selling new contracts to customers will therefore reduce their risk of stopping their subscription. Nevertheless, the analysis does not confirm this hypothesis: it might be the case that a satisfied customer is typically not likely not churn and is also more prone to buying new services. In this case, the churn and the number of contracts have a common cause (customer satisfaction), and manipulating the number of contracts will not modify the risk of churn. Also, the magnitudes of the difference in churn

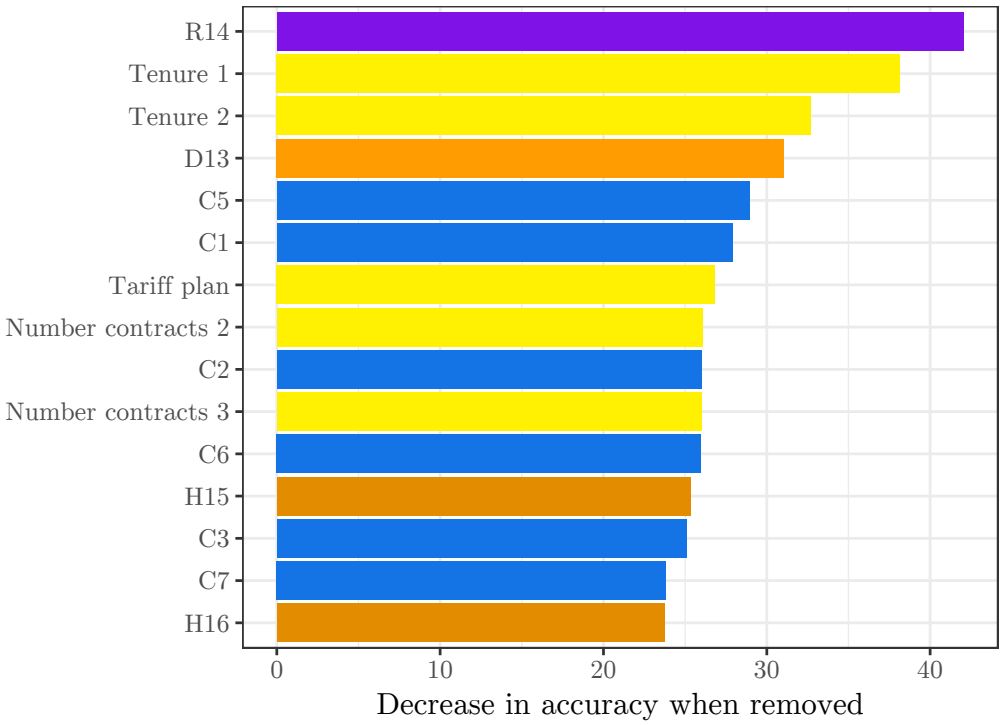


Figure 3.20 – Mean decrease in accuracy when a feature is removed for SIM only with difference and ratio variables.

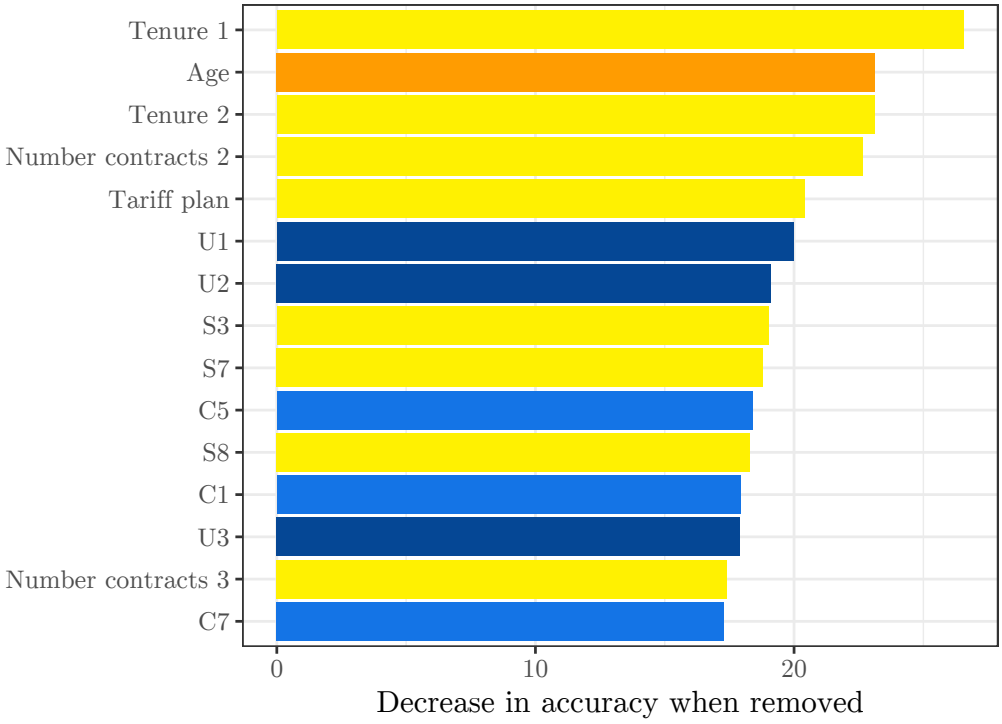


Figure 3.21 – Mean decrease in accuracy when a feature is removed for loyalty.

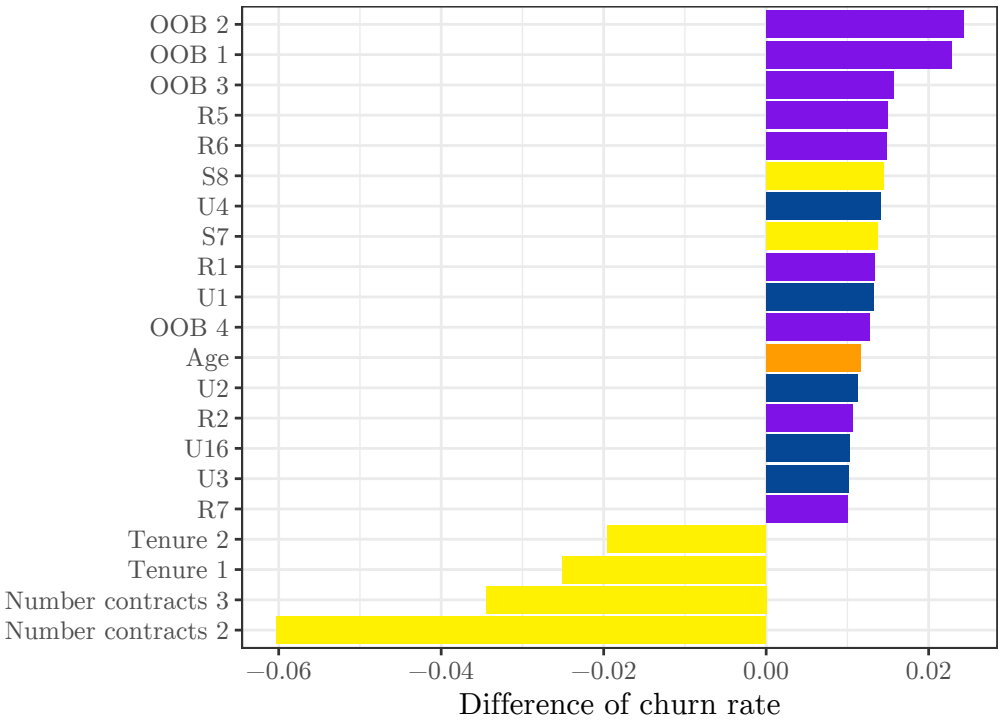


Figure 3.22 – Difference in the predicted probability of churn when a standard deviation is added separately to each variable. Run on the SIM only dataset. Only variables inducing a difference having an absolute value greater than 0.01 are shown.

rates should not be considered as realistic, meaningful values. We added a standard deviation to each variable, regardless of whether a standard deviation makes sense for each variable. For example, the number of contracts is typically not distributed according to a Gaussian distribution, since it takes as value only small positive integers. We can expect most of the decision trees composing our model to choose a split point close to 1 for this variable, in order to categorize clients as either having 1 contract or more. Adding a standard deviation to the variable would change the path of each sample in these trees. This explains the disproportionate 15% of difference in churn rate in figure 3.23.

3.5 Comparison to the state of the art

In this section, we compare our results to other studies in churn prediction. Of the 20 articles in our bibliography related to churn prediction, 15 are empirical studies either suggesting a new method or comparing existing methods. 7 of these 15 articles use precision, recall, accuracy, and F-measure as evaluation measures. These evaluation measures are applicable when the output of the prediction model is a hard label, such as for a support vector machine or a decision tree. However, our experiment uses an ensemble model composed of random forests and the predictions take the form of a score between 0 and 1. This implies that a decision threshold has to be chosen when classifying customers as churners or non-churners. The precision,

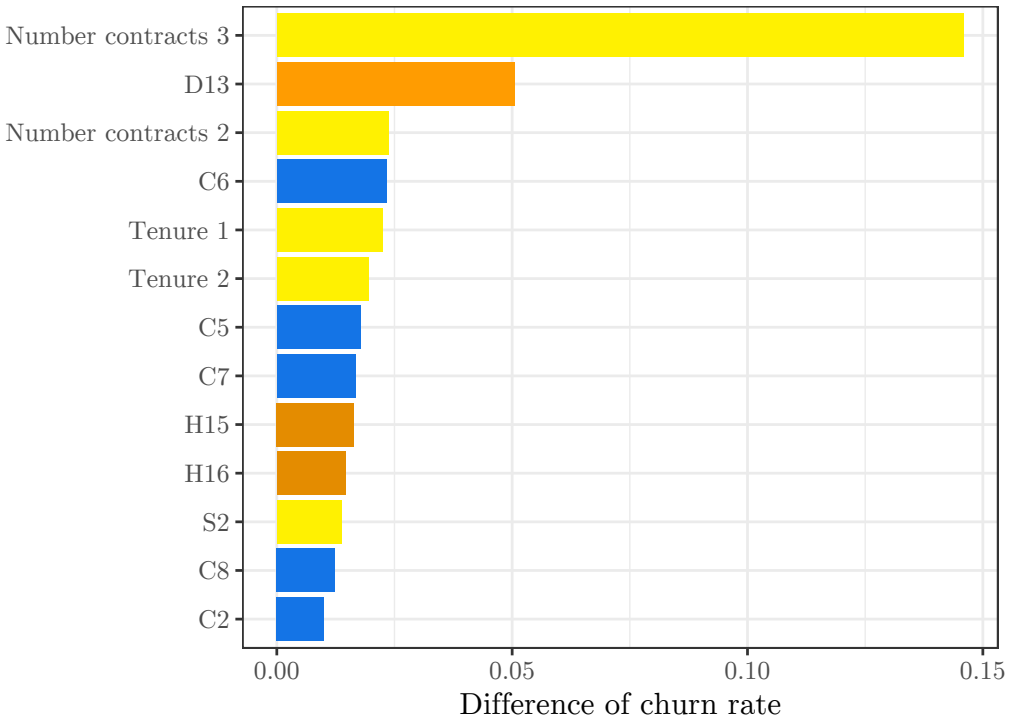


Figure 3.23 – Difference in the predicted probability of churn when a standard deviation is subtracted separately from each variable. Run on the SIM only dataset. Only variables inducing a difference having an absolute value greater than 0.005 are shown.

Paper	Best method	AUROC	Lift 10%	Lift 5%
Coussement et al., 2017	Logistic regression	0.63	2.19	—
De Caigny et al., 2018	Logit Leaf model	0.87	5.34	—
Óskarsdóttir et al., 2018	Similarity forests	0.87	6.05	—
Zhu et al., 2017	C4.5 with UnderBagging	0.80	4.54	—
Mitrović et al., 2018	Random forest	0.74	2.35	—
Idris et al., 2014	Ensemble with mRMR	0.75	—	—
Óskarsdóttir et al., 2017	Logistic Regression	0.89	—	6.16
Verbeke et al., 2014	Relational classifier	—	3.11	3.92
Our results	Easy Ensemble	0.73	3.41	4.68

Table 3.3 – Comparison of our results to other studies in churn prediction.

recall, F-measure and accuracy are thus functions of this threshold, and this does not allow a direct comparison with these 7 empirical studies. We are left with 8 other studies which use either the lift at different thresholds (most often 10%, also named top decile lift), the expected maximum profit and the area under the ROC curve (AUROC). The results of these studies are compiled in table 3.3, along with our results in the last row.

We consider our results on the SIM only test set, taking the results of one of the best performing configurations (no variable selection, no difference or ratio

variables). It is important to notice that the studies mentioned in table 3.3 obviously do not provide a unique numerical result. In each of these studies, we considered, whenever possible, the results pertaining to a dataset similar to ours in terms of churn rate and type of contracts. Also, when multiple methods are compared in a study, we retained the evaluation measure of the method performing best. The name of the method retained in each study is given in the second column of the table.

In terms of area under the ROC curve, we achieve results similar to 2 studies, both using random forests (the ensemble proposed by Idris and Khan (2014) contains a random forest, a KNN, and a rotation forest). Also, we perform better than the logistic regression proposed by Coussement et al. (2017), but the 4 remaining papers outperform our model by a clear margin. In terms of lift, we outperform the logistic regression in the first row, the random forest used in Mitrović et al. (2018) and the combined relational classifier proposed by Verbeke et al. (2014). All other studies yield a superior lift, both at 5% and 10% threshold.

3.6 Conclusion

We summarize here the main findings of our experiments on churn prediction. All those conclusions are inevitably strongly related to the limited dataset we considered. We expect that further validation could help in better supporting such conclusions. Also, the variable selection performed in these experiments give a useful, restricted scope for the causal analysis in the next chapter by discarding irrelevant variables.

- Feature selection does not reduce performance if at least 30 of the most important variables are selected.
- Adding difference and ratio variables reduces the performance if no feature selection is conducted beforehand.
- Due to a lucky domain shift, the trained models actually perform better on the test set than on the validation set.
- Churn is slightly easier to predict in the loyalty dataset, due to the importance of time-related variables.
- Important variables include non-exhaustively: the tenure, the province, the tariff plan, the number of calls, and the data usage.
- The tenure and the number of contracts are associated monotonically to the churn probability
- Variables related to the amount paid by the customer are associated to more churn when they are increased, but the opposite is not true.

Chapter 4

Causal analysis

4.1 Introduction

In this chapter, the application of causal inference to customer data is explored, in the hope of shedding light on the reasons for customer churn. A predictive experiment as conducted in the previous chapter indicates which variables are indicative of a client about to churn, but there is no guarantee that an intervention on any of these variables will have a positive effect. For example, the number of contracts registered by a customer has a strong predictive power as shown in figure 3.20. However, a hypothetical churn retention action that would sell additional contracts will maybe fail, if satisfied clients are more prone to buy new contracts but not dissatisfied ones. In this case, the predictive variable (number of contracts) and the churn have a common cause (customer satisfaction). Manipulating the number of contracts will therefore have no effect on churn. Different tools are needed to discover true causal relationships between variables. In this chapter, we focus on three types of models: causal Bayesian networks, information-theoretic filters, and supervised causal inference. An overview of state-of-the-art methods for each of these models is given in section 2.2, but we describe them here in more details.

We begin by introducing *Bayesian networks*, which are graphical models used to represent probabilistic dependencies between random variables. They are represented by a *directed acyclic graph* (DAG) where the nodes are random variables, and a joint probability density is assigned to these variables. We will use the terms *nodes* and *variable* interchangeably. In a directed acyclic graph, a node A is a parent of B if there is a direct edge from A to B, A is an ancestor of B if there is a directed path from A to B. We can define in a similar way the notions of child, descendant, and spouse in a directed acyclic graph. Bayesian networks come in a causal variant, which is defined here (Guyon, Aliferis, et al., 2007).

Definition 1 (Causal Bayesian network)**.** Let $\boldsymbol{X}$ be a set of random variables and P a joint probability density over $\boldsymbol{X}$. Let Γ be a DAG in which the vertices are $\boldsymbol{X}$. It is required that

(i) for every edge from a node $X \in \boldsymbol{X}$ to a node $Y \in \boldsymbol{X}$, X is a direct cause of Y, and

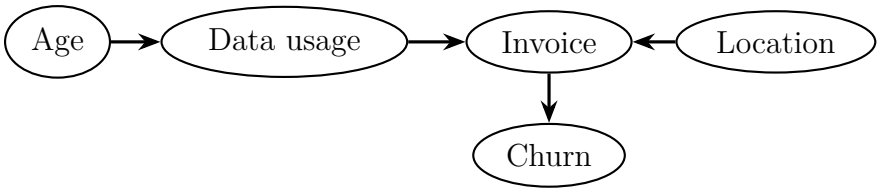


Figure 4.1 – An example of causal network illustrating the notion of d-separation. Age and invoice are d-separated given data usage, whereas data usage and location are no longer d-separated once we know the invoice amount or the churn variable.

(ii) for every node $X \in \boldsymbol{X}$, X is probabilistically independent of all its non-descendants given its parents, according to P:

$$X \perp_P (\boldsymbol{X} \setminus \text{Descendents}(X))|\text{Parents}(X) \tag{4.1}$$

where Descendents(X) is the set of descendent nodes of X and Parents(X) is the set of parent nodes of X.

The first condition is required for a Bayesian network to be causal, and the second condition is called the Markov condition. The tuple $(\boldsymbol{X}, P, \Gamma)$ is a causal Bayesian network iff both conditions are satisfied.

We denote independence between two variables X and Y according to the probability density P as $X \perp_P Y$. Similarly, the conditional independence between two variables X and Y given a set of variables $\boldsymbol{Z}$ is written $X \perp_P Y|\boldsymbol{Z}$.

The notion of d-separation, as introduced by Pearl (*e.g.* in Pearl, 2002), gives a graphical criterion to evaluate the conditional dependence between two sets of nodes entailed by the Markov property. This notion uses the terms *chain* (a causal pattern of the form $X \to Y \to Z$), *fork* ($X \leftarrow Y \to Z$), and *collider* ($X \to Y \leftarrow Z$).

Definition 2 (d-separation)**.** Let X and Y be two variables in $\boldsymbol{X}$, and Π an undirected path between X and Y in the DAG Γ. The path Π is blocked by a set of nodes $\boldsymbol{Z} \subset \boldsymbol{X} \setminus \{X, Y\}$ if either of the two following statements are true:

(i) Π contains a chain $I \to Z \to J$ or a fork $I \leftarrow Z \to J$ such that Z is in $\boldsymbol{Z}$

(ii) Π does not contain a collider $I \to Z \leftarrow J$ such that Z or any of its descendent nodes are in $\boldsymbol{Z}$

If all undirected paths Π between X and Y are blocked by $\boldsymbol{Z}$, then $\boldsymbol{Z}$ d-separates X and Y.

Using the notion of *d-separation*, we define the conditional independence between two variables X and Y given $\boldsymbol{Z}$ entailed by a graph Γ as $X \perp_\Gamma Y|\boldsymbol{Z}$. To illustrate this notion, consider the figure 4.1. In this fictional exemple, the churn is caused solely by the amount to pay on the invoice. This invoice is in turn determined by the data usage (and indirectly by the age of the customer), and also by the location: a customer in a less populated area pays more, because establishing the connectivity is more expensive for the operator. In this configuration, the age and the invoice

are d-separated by the data usage, since knowing the data usage of a client removes any information the age may bring about the invoice amount. Also, data usage and location are d-separated given the empty set, since there is a collider between them. However, if we know the invoice variable, data usage and location are no longer d-separated, as the knowledge of the invoice allows to *explain away* one variable with the other. If the client has a large invoice amount, and is living in a populated area, that must mean that its data usage was probably high. Note that knowing the churn variable instead of the invoice would have the same effect, since churn is a direct effect of invoice.

The condition of *faithfulness* is often set onto a causal Bayesian network:

Definition 3 (Faithfulness)**.** A DAG Γ is *faithful* to a joint probability density P iff every dependency entailed by the Markov condition on Γ is also entailed by P. That is,

$$\forall X, Y \in \boldsymbol{X}, \forall \boldsymbol{Z} \subset \boldsymbol{X}, \quad X \not\perp_{\Gamma} Y | \boldsymbol{Z} \Rightarrow X \not\perp_{P} Y | \boldsymbol{Z}. \tag{4.2}$$

Both the Markov conditions and the faithfulness conditions ensure that a given graph and a given probability density represent accurately the same set of dependencies and independencies. When both conditions are met, we write (in)dependence relations without specifying whether it is entailed by P or Γ. The faithfulness condition, in particular, ensures that the influence of a cause onto an effect by multiple causal routes does not cancel itself out. To demonstrate a violation of this assumption, assume a gene codes both for the production of a particular protein and for the suppression of another gene that codes this protein as well. In this case, when the first gene is removed, the protein is still produced by the other gene, and therefore the presence of the first gene seems to be independent of the production of the protein. The probability density P postulates the independence between the two variables, whereas the causal structure of the problem indicates a causal link(Hitchcock, 1997).

Another important concept in Bayesian networks is the notion of *Markov blanket* of a variable. Informally, this is a set of variables that shields a given target variable from the influence of the rest of the network. This notion is useful, for example, in feature selection for machine learning. The Markov blanket of a target variable is a subset that brings the maximum information about the target, among all possible sets of variables. Adding any other variable will bring some information that is already contained in other members, or in the interaction thereof, of the Markov blanket.

Definition 4 (Markov blanket)**.** For a set of random variables $\boldsymbol{X}$, a subset $\boldsymbol{M} \subset \boldsymbol{X}$, and a variable $Y \in \boldsymbol{X}$, $\boldsymbol{M}$ is the Markov blanket of Y iff for any subset $\boldsymbol{V} \subset \boldsymbol{X}$, Y is conditionally independent of $\boldsymbol{V} \setminus \boldsymbol{M}$ given $\boldsymbol{M}$.

In our case, where causal Bayesian networks are represented by a directed acyclic graph, and where both the Markov and the faithfulness conditions are met, the

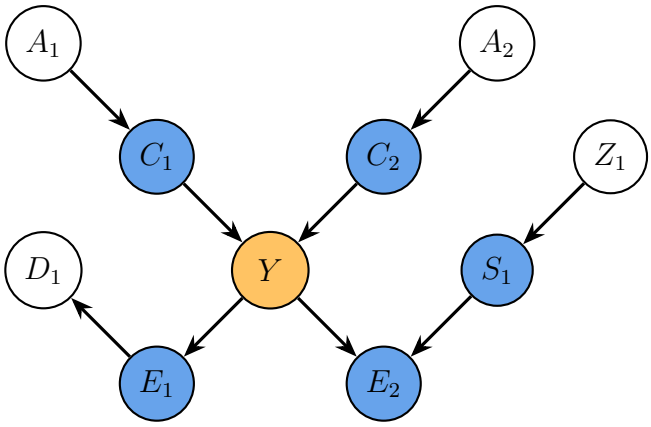


Figure 4.2 – A causal Bayesian network, with the Markov blanket of Y highlighted in blue.

Markov blanket of a variable is unique and consists of its direct causes, its direct effects, and the direct causes of its direct effects. An example is given in figure 4.2.

These definitions lay the theoretical background for graphical causal models, but methods for inference from observational data still need to be derived. The methods used in this experiment are described in section 4.4. A common requirement of some of these methods is an accurate test of statistical independence between two variables in $\boldsymbol{X}$. Many statistical independence tests exist, but in our case, we need an estimator that is able to handle a mix of categorical and continuous variables. We use the *mutual information*, which is an information-theoretic measure of statistical dependency. It is more general than the Pearson or the Spearman correlation coefficient, as it encompasses any type of dependency, and not only linear or monotonic relationships. Also, it is defined for any two random variables, regardless of their type or domain. In the case of two discrete variables X and Y taking values respectively in $\mathcal{X}$ and $\mathcal{Y}$, with a joint probability distribution $P(X, Y)$ and marginal probability distributions $P(X)$ and $P(Y)$, the mutual information between X and Y is defined as (Cover & Thomas, 2012)

$$I(X,Y) = \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} P(x,y)\log\left(\frac{P(x,y)}{P(x)P(y)}\right) \tag{4.3}$$

$$= H(X) - H(X|Y) \tag{4.4}$$

$$= H(Y) - H(Y|X) \tag{4.5}$$

$$= H(X) + H(Y) - H(X,Y) \tag{4.6}$$

where $H(X)$ is the entropy of X and $H(X|Y)$ is the conditional entropy of X given Y(Shannon, 1948). The two last equalities indicate that the mutual information is a symmetric, positive quantity, and that it can be viewed as the reduction in uncertainty that a variable brings about the other. A schematic view of these formulae is given in figure 4.3. In the case of two continuous variables, an analogous definition exist (Kolmogorov, 1956)

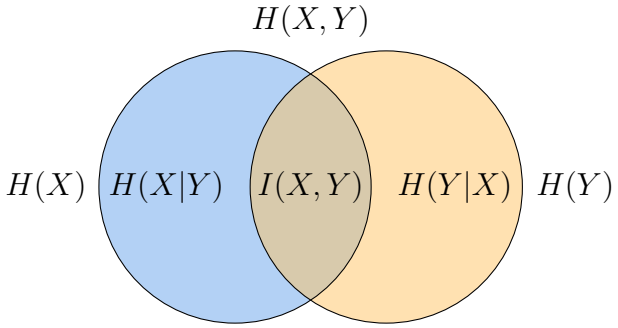


Figure 4.3 – Schematic representation of the relationship between entropy, conditional entropy, joint entropy, and mutual information of two discrete random variables. The blue circle represents $H(X)$ and the orange circle represents $H(Y)$.

$$
\begin{aligned}
I(X,Y) &= \int_{\mathcal{X}} \int_{\mathcal{Y}} P(x,y) \log \left(\frac{P(x,y)}{P(x)P(y)} \right) dx\, dy && (4.7)\\
&= h(X) - h(X|Y) && (4.8)\\
&= h(Y) - h(Y|X) && (4.9)\\
&= h(X) + h(Y) - h(X,Y) && (4.10)
\end{aligned}
$$

where $h(X)$ is the differential entropy of X and $h(X|Y)$ is the conditional differential entropy of X given Y. In the case of two normally distributed variables, we have

$$I(X,Y) = -\frac{1}{2} \log(1 - \rho^2) \tag{4.11}$$

where ρ is the Pearson correlation coefficient between X and Y. Even though the assumption of normal distribution does not hold in general, empirical results show that it is still a decent estimator for non-linear dependencies (Olsen, Meyer, & Bontempi, 2008). Finally, the mutual information between a continuous variable X and a discrete variable Y taking values in a finite set $\mathcal{Y}$ can be computed as

$$I(X,Y) = h(X) - h(X|Y) = h(X) - \sum_{y \in \mathcal{Y}} h(X|Y=y) P(Y=y) \tag{4.12}$$

which means that the mutual information in the mixed case can be computed with an estimator of the differential entropy of a continuous variable. Following the discretization method described in (Olsen et al., 2008), let N be the number of values sampled independently and identically from X. We divide the domain $\mathcal{X}$ of X into k bins of equal size Δ, and we write $\mathrm{nb}(i)$ the number of samples present in the ith bin, $\forall i \in \{1, \ldots, k\}$. The differential entropy of X is estimated with the *Miller-Madow estimator*

$$\hat{h}(X) = \sum_{i=1}^{k} \frac{\mathrm{nb}(i)}{N} \log \left(\frac{\mathrm{nb}(i)}{N} \right) + \log \Delta + \frac{k-1}{2N} \tag{4.13}$$

It follows that we can compute the mutual information in all possible configurations of variable types:

(i) Between two discrete variables, using equation 4.3

(ii) Between a continuous variable X and a discrete variable Y with equation 4.12 and the differential entropy estimator of equation 4.13

(iii) Between two continuous variables assuming normal distributions with equation 4.11

The mutual information can be generalized for any n variables, and is then named n-way *interaction* or *co-information* (A. J. Bell, 2003). A general formula exists, but we are only interested in the case $n = 3$ since it is connected with causal configurations composed of three variables. All causal configurations of three variables can be easily enumerated, allowing to develop causal discovery algorithms based solely on the 3-way interaction. In this case, the 3-way interaction (that we simply name interaction) between three random variables X_1, X_2, and Y is defined in (McGill, 1954) as

$$I(X_1, X_2, Y) = I(X_1, X_2) - I(X_1, X_2|Y) \tag{4.14}$$

where, in the case of a discrete Y taking values in a finite set $\mathcal{Y}$, $I(X_1, X_2|Y)$ can be computed as

$$I(X_1, X_2|Y) = \sum_{y \in \mathcal{Y}} P(Y = y) I(X_1, X_2|Y = y) \tag{4.15}$$

A more general definition of conditional mutual information is given in (Cover & Thomas, 2012). But since we consider a classification problem with $\mathcal{Y} = \{0, 1\}$, this restricted definition is sufficient for our purposes.

4.2 Scope

In this chapter, the same dataset as in the predictive modeling chapter is used. We restrict ourselves to SIM only contracts since it is supposed that the causes of churn are at least partially different between loyalty and SIM only contracts. All 5 months of data are used. In order to decrease computation time, only the first 30 variables in the ranking of the random forest trained in chapter 3 are used. Depending on the algorithm being used, a random subsampling has been applied, also to reach decent computation times. In all cases, the positive class (churners) is untouched, and a random subset of the negative class is sampled so that the class ratio is even.

4.3 Prior knowledge of churn

Bayesian reasoning indicates that we should always take into account our prior knowledge on a problem when considering the outcome of an experiment designed

to test some hypotheses on this problem. An unlikely theory according to our priors needs strong evidence to be proven true. Ideally, we should assign a probability (in the Bayesian sense, that is, a degree of belief) to every theory, and calculate the probability of the experiment results according to each theory. We will obviously not conduct this procedure formally in the context of this study, but the Bayesian methodology is nonetheless useful to keep in mind when interpreting the results of the causal inference experiments. This section describes the prior knowledge we have on the causes of churn. It is the result of discussions and interviews with the data science and business intelligence teams at Orange Belgium. This prior knowledge is highly valuable since it incorporates years of experience in churn prevention and customer relationship in general. We summarize the main causes of churn in four different settings.

Bill shock This setting has been already evoked multiple times in chapter 3. It occurs when a customer has an unusually large service usage, which results in an important out of bundle amount (i.e. the client is charged much more than usual). This triggers a reaction from the customer inducing an increased risk of churn. This scenario is well understood and verified in practice. It is believed to be the most important cause of churn.

Customer dissatisfaction Multiple factors influence customer satisfaction, including the quality of service and the network quality. A customer having numerous cuts of network connection during phone calls, or unable to use properly Orange online services, will be more likely to seek better alternatives elsewhere.

Wrong positioning Each customer has different service usage habits. Some people make a few minutes of phone calls per month, whereas this can be counted in hours for others. This is why multiple tariff plans are proposed by the service provider. However, choosing the right tariff plan is sometimes difficult. On the one hand, if not enough call time is provisioned, an out of bundle amount is likely to be charged at the end of the month. On the other hand, an expensive tariff plan results in a high fixed cost for the customer. When the tariff plan of the customer does not correspond to her needs, we say that the customer is wrongly positioned. A wrong positioning results in most cases to a higher bill that expected, and is a significant cause of churn.

Churn due to a move It is common to choose a product bundle from a telecommunication company comprising a subscription for mobile phone, landline phone, television, and internet connection. In this case, the subscription is tied to the particular place of domicile of the customer. When the client moves to another place, it is quite common to also change for another telecommunication service provider. Therefore, this is a significant cause of churn, albeit of a different nature from the other settings exposed above.

These different settings are described informally, and their translation to the formal definitions of causality presented in section 4.1 is not straightforward. This touches upon philosophical debates on the definition of causality, which are well beyond the scope of this thesis. More practically, we wish to find a mapping between the events believed to be causes of churn and specific instantiations of measurable random variables. In the case of the first setting, we can reasonably assume that variables measuring the out of bundle amount of the customer is a faithful proxy for bill shock. Similarly, customer satisfaction can be estimated using, for example, the number of network cuts during phone calls, or the number of calls to the customer service. The wrong positioning can also be numerically estimated, given the tariff plan of the client and its average service usage. The last setting, churn due to a move, is much more difficult to estimate, as it is not directly related to the interaction between the client and the telecommunication services.

In the dataset available for this study, the only measured variables that translate to potential causes of churn are the out of bundle, the tariff plan and service usage (phone calls, messages, mobile data). We have no measure for network quality, customer satisfaction, or propensity to move in the near future. Also, the wrong positioning is not explicitly encoded and has to be inferred by the causal inference model from the average service usage and the current tariff plan.

4.4 Experiments

The overall scheme of this experiment consists of running several causal inference techniques, which give different types of results in various forms, and extract a general consensus, if any, in the light of the different assumptions each model put on the data. Indeed, all causal inference methods are based on different assumptions, and the ability of a given method to infer causal patterns from observational data lies upon these assumptions.

More specifically, we use 5 different causal inference algorithms:

- PC
- Grow-shrink (GS)
- Incremental Association Markov Blanket (IAMB)
- Minimum interaction maximum relevance (mIMR)
- D2C

For the first three methods, we use the R package *bnlearn* (Scutari, 2009) for independence tests using mutual information and asymptotic χ^2 test (Good, 2013). For mIMR and D2C, we use the R package *D2C* (Bontempi & Flauder, 2015), along with another implementation of mIMR using the mutual information estimator given in section 4.1. The sample size used in the experiment is given after the description of each algorithm. In all cases, a false positive rate of 0.05 is chosen for statistical tests of independence.

PC

Description The PC algorithm (Spirtes & Glymour, 1991) returns the set of directed acyclic graphs that are faithful to a given probability distribution. It is based on independence tests between two variables, conditioned on a set of other variables. It uses the notion of d-separation to eliminate or find the direction of putative causal links. The PC algorithm is given in algorithm 1, where **Adjacencies**(X) is the set of nodes that are adjacent to X in the current version of the result Γ. Therefore, it evolves as Γ is modified in the algorithm. When two directions for a causal link are faithful, Γ contains an undirected edge to represent the two directions at once. The result is thus a graph having directed and undirected edges, but which represent an equivalence class of undirected acyclic graphs. The idea underlying PC is to A) start with a full graph, B) remove edges using independence tests with conditioning sets of increasing size, C) orient colliders using the d-separation property, and D) find remaining orientations using two more rules. The assumptions underlying this algorithm are

(i) There is no unmeasured confounder

(ii) The statistical tests are correct

(iii) The causal relationships between variables are the same for all samples

(iv) There exists a DAG Γ representing the causal structure that is faithful to the underlying joint probability density P (definition 3)

If these assumptions hold, then the result of the algorithm is a set of DAGs that are all faithful to the density probability P. The assumption (iii) is reasonable in our case, but the three others less so.

Experimental setting The PC algorithm is slow when the number of samples is large since the whole Bayesian network is inferred. Therefore, we restrict the dataset to 10,000 samples. The implementation given in the package *bnlearn* is used. The results are given under the form of a directed acyclic graph.

Grow-Shrink

Description The GS algorithm (Margaritis & Thrun, 2000) is a Markov Blanket discovery algorithm efficient even for a large number of variables. It is based on an estimator that returns a numerical value for the statistical dependency between two variables, potentially conditioned on a set of other variables. In our case, we use the mutual information. Consider a target variable Y and a set of predictor variables $\boldsymbol{X}$. The GS algorithm constructs the Markov blanket of Y, denoted $MB(Y)$, in three phases performed in sequence:

A) All variables $X \in \boldsymbol{X}$ are ordered in decreasing order according to $I(X, Y)$

Algorithm 1 The PC algorithm

A) Let Γ be a complete undirected graph on the set of vertices $\boldsymbol{X}$.
B) $n \leftarrow 0$
repeat
  repeat
    Select a pair of vertices X and Y such that
      • X and Y are adjacent
      • there exists a set $\boldsymbol{Z} \subseteq$ **Adjacencies**$(X) \setminus \{Y\}$ that has n elements
      • $X \perp Y | \boldsymbol{Z}$.
    Remove the edge X—Y from Γ
    Add $\boldsymbol{Z}$ to **Sepset**(X, Y) and to **Sepset**(Y, X)
  until no X, Y pairs satisfying above conditions can be found
  $n \leftarrow n + 1$
until for each pair X and Y, $|$**Adjacencies**$(X) \setminus \{Y\}| < n$
C) For all triplets X—Y—Z where there is no edge between X and Z, orient it as $X \rightarrow Y \leftarrow Z$ if Y is not in **Sepset**(X, Z)
D)
repeat
  Orient all $X \rightarrow Y$—Z as $X \rightarrow Y \rightarrow Z$
  Orient all X—Y as $X \rightarrow Y$ if there is a directed path from X to Y
until no more edges can be oriented

B) Each variable $X \in \boldsymbol{X}$ is added to $MB(Y)$ iff it is conditionally dependent on Y, given $MB(Y)$. That is, iff $I(X, Y|MB(Y)) > 0$.

C) Each variable $X \in MB(Y)$ is removed from $MB(Y)$ iff it is conditionally independent of the rest of $MB(Y)$. That is, iff $I(X, Y|MB(Y) \setminus \{X\}) = 0$.

The phase A) is a heuristic to speed up the search, however Tsamardinos, Aliferis, Statnikov, and Statnikov (2003) pointed out that this delays the inclusion of spouses, since those have small unconditional relevance to Y. Therefore, more false positives are included before spouses get into $MB(Y)$.

Experimental setting The entire set of positive samples is used, along with a subset of the same size of negative samples. This amounts to a total of 240,168 samples. The GS algorithm has been implemented using the independence test in the package *bnlearn*. The results are given as a list of members of the Markov blanket.

Incremental Association Markov Blanket

Description The IAMB algorithm (Tsamardinos, Aliferis, et al., 2003) is essentially similar to the GS algorithm, but does not include the sorting heuristic as a first phase:

A) The variable $X \in \boldsymbol{X}$ maximizing $I(X, Y|MB(Y))$ is added to $MB(Y)$, repeatedly until all remaining variables are independent of Y given $MB(Y)$.

B) Each variable $X \in MB(Y)$ is removed from $MB(Y)$ iff it is conditionally independent of the rest of $MB(Y)$. That is, iff $I(X, Y|MB(Y) \setminus \{X\}) = 0$.

Experimental setting The experimental setting is the same as for the GS algorithm.

Minimum Interaction Maximum Relevance

Description The mIMR filter (Bontempi & Meyer, 2010) is a feature selection algorithm that has similarities with the mRMR algorithm (Peng et al., 2005). By using the interaction instead of the redundancy between the candidate variable and the set of selected variables, causes and spouses of the target are favored. In order to select only causes, spouses are eliminated beforehand on the basis of their null unconditional mutual information with the target. More formally, the main objective of feature selection is to find a subset $\boldsymbol{X}^*$ of a set of variables $\boldsymbol{X}$ that maximizes the mutual information with the target Y:

$$\boldsymbol{X}^* = \underset{\boldsymbol{X}_S \subseteq \boldsymbol{X}}{\arg\max}\, I(\boldsymbol{X}_S, Y) \tag{4.16}$$

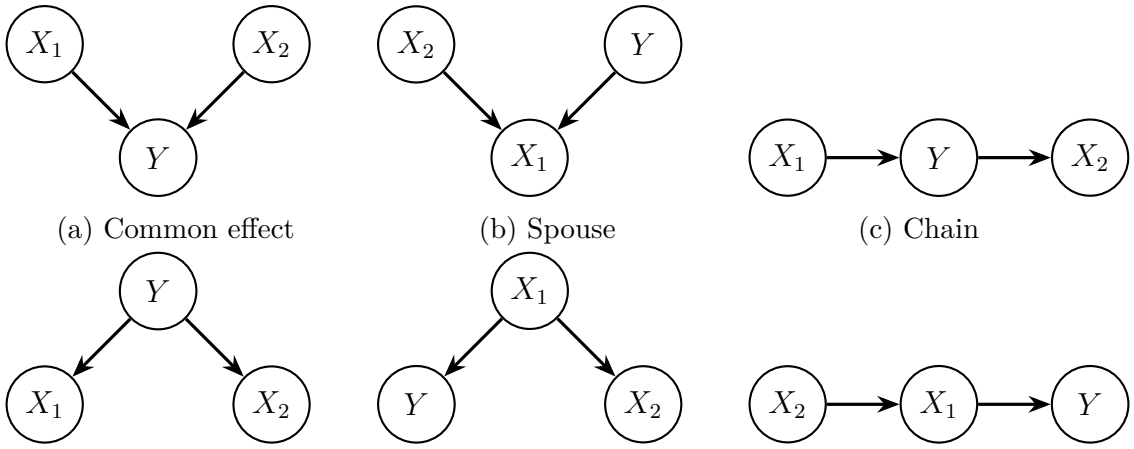
Evaluating all possible subsets $\boldsymbol{X}_S$ is computationally infeasible, the forward selection scheme is therefore adopted. It consists in repetitively selecting the variable X^*_{d+1} that bring the most improvement given the set $\boldsymbol{X}_S$ containing the d variables already selected, until a fixed number v of variables is attained. The improvement is determined in a way that favors direct causes of Y. Consider the interaction equation 4.14. The interaction $I(X_1, X_2, Y)$ can be viewed as the reduction in statistical dependency between X_1 and X_2 brought by the knowledge of Y. On the one hand, a positive interaction occurs in 4 types of causal patterns, shown in figures 4.4c, d, e and f. On the other hand, a negative interaction occurs only in the common effect configuration (figure 4.4a) and the spouse configuration (figure 4.4b). Furthermore, one can differentiate between common effect and spouse configurations by noticing that a spouse of the target has null unconditional relevance with the target (a spouse is relevant only when the common effect is known). In practice, a statistical test is used to restrict the set of considered variables to those having non-null mutual information with the target, written $\boldsymbol{X}_+$. This leads to the update criterion of mIMR that is a linear combination of relevance and interaction:

$$X^*_{d+1} = \underset{X_k \in \boldsymbol{X}_+ \setminus \boldsymbol{X}_S}{\arg\max}\, \left[I(X_k, Y) - I(\boldsymbol{X}_S, X_k, Y) \right]. \tag{4.17}$$

Using the approximation

$$I(\boldsymbol{X}_S, X_k, Y) \approx \frac{1}{d} \sum_{X_i \in \boldsymbol{X}_S} I(X_i, X_k, Y), \tag{4.18}$$

the forward step can be written as



(a) Common effect (b) Spouse (c) Chain

(d) Common cause (e) Brotherhood (f) Grandparent

Figure 4.4 – Possible causal patterns between two variables X_1 and X_2 and a target variable Y having non-null interaction.

$$X^*_{d+1} = \underset{X_k \in \boldsymbol{X}_+ \setminus \boldsymbol{X}_S}{\arg\max} \left[I(X_k, Y) - \frac{1}{d} \sum_{X_i \in \boldsymbol{X}_S} I(X_i, X_k, Y) \right] \tag{4.19}$$

$$= \underset{X_k \in \boldsymbol{X}_+ \setminus \boldsymbol{X}_S}{\arg\max} \left[I(X_k, Y) - \frac{1}{d} \sum_{X_i \in \boldsymbol{X}_S} \left(I(X_i, X_k) - I(X_i, X_k, |Y) \right) \right] \tag{4.20}$$

$$= \underset{X_k \in \boldsymbol{X}_+ \setminus \boldsymbol{X}_S}{\arg\max} \left[I(X_k, Y) + \frac{1}{d} \sum_{X_i \in \boldsymbol{X}_S} \sum_{y \in \mathcal{Y}} P(Y = y) \left(I(X_i, X_k, |Y = y) - I(X_i, X_k) \right) \right] \tag{4.21}$$

where equation 4.20 is derived using equation 4.14 and similarly 4.21 is derived from 4.15. The first two variables are selected as

$$X^*_1, X^*_2 = \underset{X_i, X_k \in \boldsymbol{X}_+}{\arg\max}\, I([X_i, X_k], Y). \tag{4.22}$$

The mIMR filter is based on some underlying estimator of mutual information between two variables, and a statistical test of independence for selecting the set of unconditionally relevant variables $\boldsymbol{X}_+$.

Experimental setting In this experiment, two implementations are used:

- One using the mutual information estimator described in section 4.1 and the test of independence in the package *bnlearn*.
- One assuming normally-distributed continuous variables, allowing to compute the mutual information (with equation 4.11) and to test for independence using the Pearson correlation coefficient. Discrete variables are converted to numerical values using one-hot encoding.

The drawback of the first method is that the mutual information estimator is ad-hoc: it assumes a monotonic relationship between two continuous variables but

uses a histogram-based entropy estimator in the mixed case. This may lead to inconsistencies in the measure of mutual information. On the other hand, the second method sets the linear assumption on all variables, even on one-hot encoded categorical variables. For the first implementation, the dataset is restricted to 10,000 samples, due to the computational cost of the entropy estimator. In the second implementation, 100,000 samples are used. The results are provided as a list of the first 15 selected variables, accompanied with the gain provided by each variable at each iteration of the algorithm.

D2C

Description The first three causal inference algorithms used in this section are solely based on statistical independence tests, and therefore are unable to differentiate between indistinguishable causal patterns, such as the two variables configuration or the fully-connected three variables configuration. Since the probability density P is reduced to a set of (in)dependence relations, any fully connected graph is faithful to P in these two cases. Asymmetrical patterns exist however in the joint probability density of a cause and its effect, as demonstrated by the results of the Kaggle competition Cause-effect pairs (https://www.kaggle.com/c/cause-effect-pairs). The D2C algorithm (Bontempi & Flauder, 2015) is based on the asymmetry of descriptors extracted from the Markov blanket of two causally linked variables. Consider two random variables X_1 and X_2 such that X_1 is a cause of X_2, and their respective Markov blanket $MB(X_1)$ and $MB(X_2)$. This setting is pictured in figure 4.5. We consider only one cause, effect and spouse per Markov blanket for the sake of the presentation, but the principles generalize obviously to any Markov blanket. Two assumptions are however made: i) the only path between $X_1 \cup MB(X_1)$ and $X_2 \cup MB(X_2)$ is the edge $X_1 \to X_2$, and ii) X_1 and X_2 have no common ancestor with their respective spouse. Failure to satisfy these conditions is expected to be compensated for by the predictive model. Even though we cannot distinguish between causes, effects and spouses among $MB(X_1)$ and $MB(X_2)$, we can derive several inequalities using d-separation. Consider the variables M_1 and M_2, which are members of respectively $MB(X_1)$ and $MB(X_2)$, but whose relation to X_1 and X_2 is unknown (that is, M_1 is either C_1, E_1 or S_1). We have

$$\begin{cases} I(X_1, M_2|X_2) > I(X_2, M_1|X_1) & \text{if } M_2 = C_2 \\ I(X_1, M_2|X_2) = I(X_2, M_1|X_1) & \text{otherwise,} \end{cases} \tag{4.23}$$

since the only collider configuration between one of X_1 and X_2 and a member of the Markov blanket of the other variable is $X_1 \to X_2 \leftarrow C_1$. By computing a population of descriptors

$$D(1,2) = \{I(X_1, M_2|X_2)|\ \forall M_2 \in MB(X_2)\} \tag{4.24}$$

$$D(2,1) = \{I(X_2, M_1|X_1)|\ \forall M_1 \in MB(X_1),\}, \tag{4.25}$$

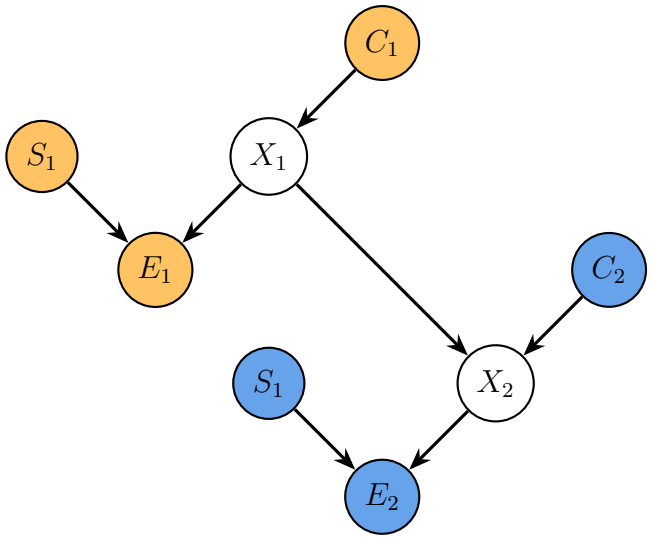


Figure 4.5 – Two causally linked variables and their Markov blanket.

equation 4.23 indicates that the distribution of $D(1,2)$ differs from $D(2,1)$. Other similar inequalities are used to compute 2 other population of descriptors. The rank of each M_1 in $MB(X_2)$ is also computed, along with the rank of each M_2 in $MB(X_1)$. The quartiles of the population of these various descriptors are computed, along with the mutual information between X_1 and X_2, and the mutual information conditioned on $MB(X_1)$ or $MB(X_2)$. All these quantities are then used as features on a machine learning algorithm, whose task is to predict the probability of a causal link between X_1 and X_2. The default implementation of D2C uses a random forest classifier.

Experimental setting The D2C model is trained using randomly generated DAGs, as described in (Bontempi & Flauder, 2015) and implemented in the R package *D2C*. We use 50 DAGs having each a number of nodes sampled uniformly between 10 and 20, and each DAG generated 50 to 200 data samples. The function underlying the edge between two nodes is randomly chosen to be either linear, quadratic or a sigmoid. A Gaussian additive noise of standard deviation chosen randomly from 0.2 to 1 is added to each directed edge. The feature extraction phase uses the lazy learning approach (Bontempi, Birattari, & Bersini, 1999) to estimate mutual information, thus avoiding the linear assumption. We assume a Markov blanket of 4 variables when constructing the asymmetrical features. Given the high computational cost of feature extraction, 2,000 samples are used from the customer dataset. The results are provided as the predicted probability for each variable to be a cause of churn.

4.5 Results

In this section, the colors of the bars in graphs correspond to the variables categories presented in section 3.1:

- Subscription
- Calls and messages
- Mobile data usage
- Revenue
- Customer hardware
- Socio-demographic

PC

The output of the PC algorithm is a dense graph linking most of the variables, but unfortunately, the churn variable is completely disconnected from the rest of the graph. Note that it also the case for the province, the tariff plan, and other variables. All of these variables have been ruled out of the Markov blanket of the churn variable in this algorithm by conditional independence tests.

Grow-Shrink

The Markov blanket output by the GS algorithm contains 21 variables:

- 4 variables related to voice calls (C5, C6, C7, and C8)
- 6 variables related to data usage (U1, U2, U3, U4, U10, and U19)
- 4 variables related to messages (C1, C2, C3, C4)
- Out of bundle amount
- Age
- Tenure
- Number of contracts
- A socio-demographic variable (D13)
- A hardware-related variable (H15)
- A subscription-related variable (S7)

As explained by Tsamardinos, Aliferis, et al. (2003), the heuristic used in GS that favors variables having high unconditional relevance to the target increases the probability of false positive, since spouses of the target are not included in the beginning. Therefore, indirect causes or effects are included instead. This is particularly obvious in our case since the 4 variables related to voice calls are highly correlated, and some of them are known to be causes of the others. The same applies to the groups of variables related to data usage and messages.

Incremental Association Markov Blanket

The Markov blanket output by the IAMB algorithm contains 2 variables:

- The current tariff plan
- The previous tariff plan

According to IAMB, the churn is therefore independent of all other variables when we know the current and the previous tariff plan of the customer. This result is surprising but is stable for different values of false positive rates: a p-value of up to 0.2 was considered significant for independence tests, always giving the same result. It is noticeable that IAMB returns a Markov blanket comprising only categorical variables, while the GS algorithm gives only numerical variables. While the conditional independence tests are identical in the two methods, the different order of tests may have favored one type of variable over the other.

Minimum Interaction Maximum Relevance

Prior to the results of the mIMR algorithm, we show in figures 4.6 and 4.7 the mutual information and the interaction matrices of the 30 most predictive variables. These values are computed using our estimator presented in section 4.1. Let X_i and X_j be two variables in our dataset, and let Y be the churn variable. Figure 4.6 shows the mutual information $I(X_i, X_j)$ in the cell of position (i, j). A line is also added for the churn variable. On the one hand, one can see that no single variable seems to have a high mutual information with the churn. This could explain why the PC and IAMB algorithms fail to find a satisfactory Markov blanket for the churn variable. On the other hand, 3 clusters of high mutual information can clearly be noticed, corresponding to the summary of voice calls, data usage and messages over the last 3 months. Given that these variables do not vary randomly from month to month, it is expected that they are strongly informative about each other, and even more about the average of the 3 months.

The matrix in figure 4.7 shows at position (i, j) the interaction between two variables X_i and X_j and the churn variable Y, that is, $I(X_i, X_j, Y)$. The row corresponding to the churn is not relevant in this figure. Recall that the interaction between two variables and a target is the reduction in statistical dependence that the knowledge of the target brings:

$$I(X_1, X_2, Y) = I(X_1, X_2) - I(X_1, X_2|Y).$$

It is also the amount of mutual dependence to the target that cannot be explained by bivariate interactions:

$$I(X_1, X_2, Y) = I(X_1, Y) + I(X_2, Y) - I([X_1, X_2], Y)$$

We thus seek couples of variables having a negative mutual information with the churn, since that means that those variables are complementary. Complementary variables are more likely to either be in a common effect or spouse configuration

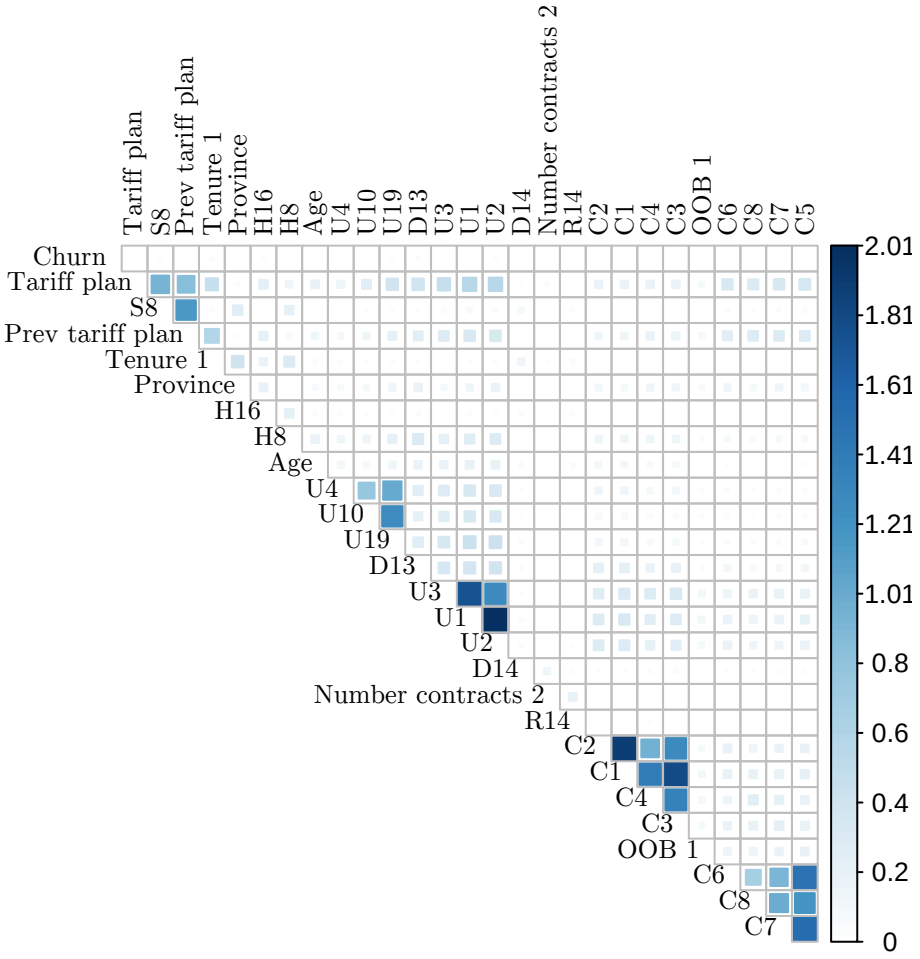


Figure 4.6 – Mutual information matrix. The color of the cell indicates the statistical dependency between the row and column variables. Clusters of high mutual information can be observed for calls, data usage, and messages variables. The churn variable has a very low mutual information with other variables.

(figures 4.4a and b) with the churn. One couple stands out clearly in figure 4.7, the tenure and the province. The province negatively interacts with most other variables, meaning that it brings information about the churn only when considering it conjointly with other variables. The clusters of strongly correlated variables in figure 4.6 have a near-zero interaction since the knowledge of the churn does not change their distributions.

Figures 4.8 and 4.9 show the sequence of variables selected by the mIMR algorithm, for both our mutual information estimator (figure 4.8) and the estimator assuming Gaussian distributions (figure 4.9). Each row corresponds to one iteration of the algorithm. The width of the bar correspond to the value of the mIMR criterion at this step of the algorithm, that is, the approximated value of $I(X_k, Y) - I(\boldsymbol{X}_S, X_k, Y)$ where Y is the churn variable, X_k is the variable under consideration and $\boldsymbol{X}_S$ is the set of variables selected before X_k (i.e. above X_k on the plot). The two first variables have no gain since they are directly selected as the pair of variables having the highest interaction with the target. Unsurprisingly, the first two variables in figure 4.8 are the tenure and the province (this couple of variable has the highest nega-

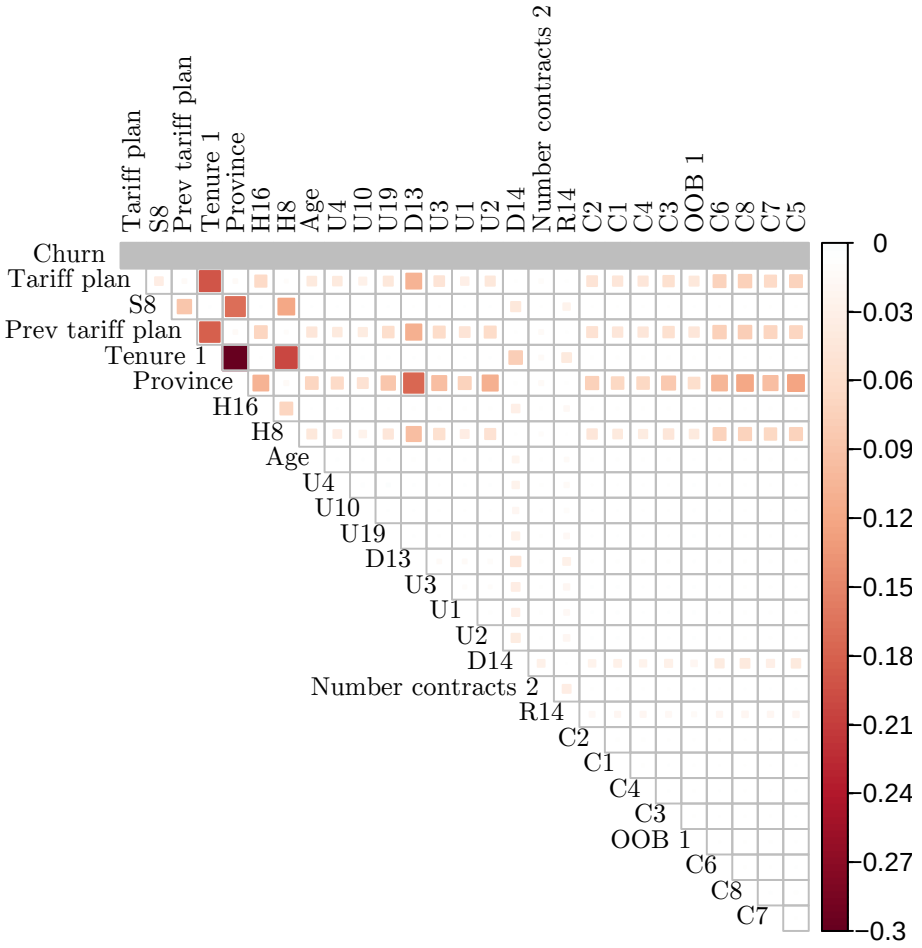


Figure 4.7 – Matrix of interaction with the churn. A large negative value indicates that the row and column variables brings more information on churn when considered together than when considered separately. The province variable has a high interaction with most other variables, as well as the tariff plan to a lesser extent.

tive interaction in figure 4.7). Background knowledge, as well as figure 4.6, indicate that the following selected variables are not redundant with one another, up to the 9th and 10th rows. At these rows, the two variables are both related to the data usage. That probably indicates that the relevance term $I(X_k, Y)$ is prevailing over the interaction term $I(\boldsymbol{X}_S, X_k, Y)$.

The selected variables in figure 4.9 are mostly similar to those in 4.8, except that all categorical variables are not in the first ranks. Since those are converted to as many numerical variables as there are categorical levels, the information is spread across multiple variables. Moreover, each of these new variables is considered to be Gaussian distributed, therefore not allowing to estimate optimally the mutual information. Another important difference between 4.8 and 4.9 is that the age and number of contracts are prevalent in the latter. The importance of age is most probably due to the Gaussian assumption, which is verified in this case, allowing efficient estimation of its mutual information with other variables.

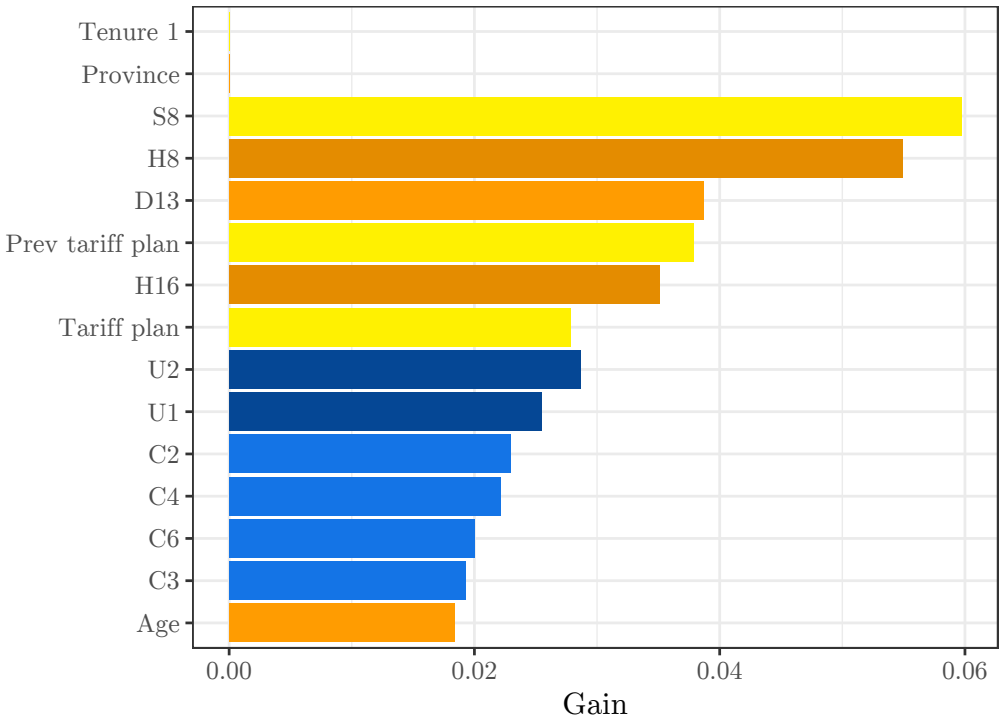


Figure 4.8 – Ranking of variables selected by mIMR with their respective gains, using the mutual information estimator of section 4.1. There is no gain for the first two variables.

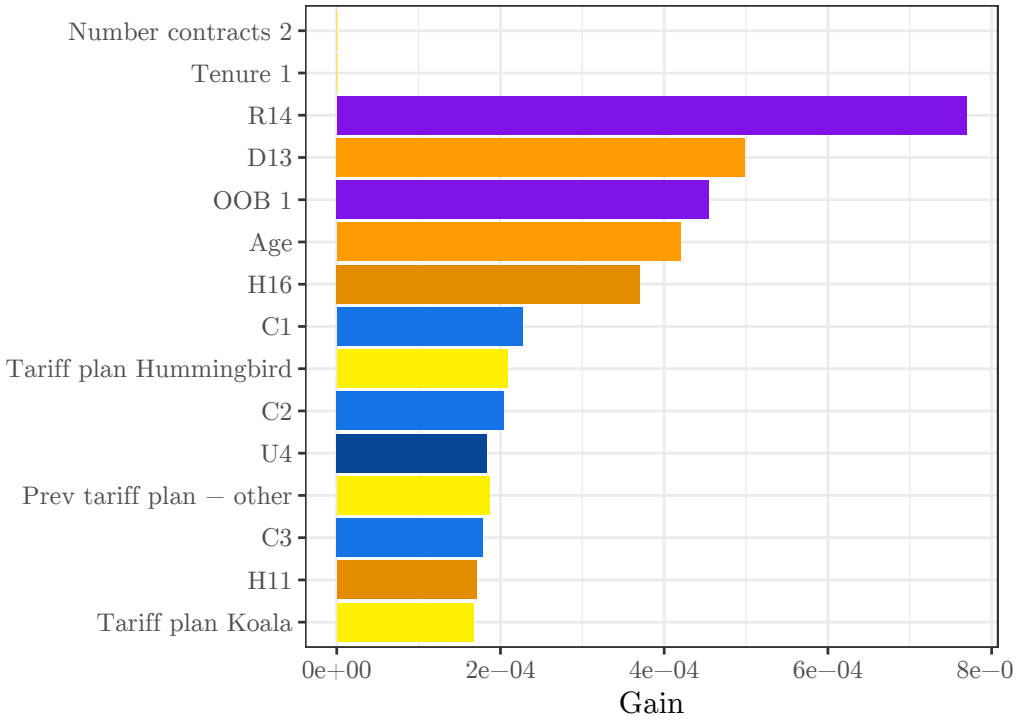


Figure 4.9 – Ranking of variables selected by mIMR with their respective gains, using one-hot encoding for categorical variables and assuming Gaussian distributions. There is no gain for the first two variables.

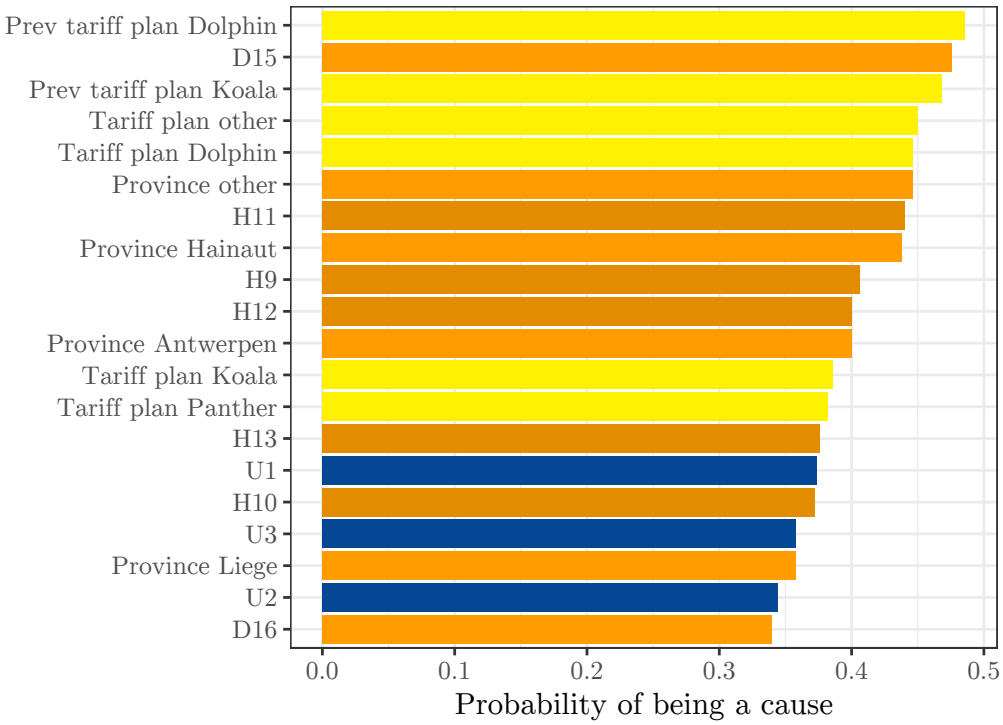


Figure 4.10 – Probability of causal link predicted by D2C

D2C

The results from the D2C algorithm are shown in figure 4.10. To each variable correspond a probability of being a cause of churn predicted by the trained random forest. Since the implementation we use for D2C is designed for numerical variables, one-hot encoding is used. All the variables present in this plot are related to the tariff plan, previous tariff plan, province of residence and hardware-related variables. This is consistent with results of mIMR in figure 4.8, except for the second and last rows, related to a categorical socio-demogrpahic variable. Mobile data usage variables are also present and are the only variables related to service usage in this graph.

We do not believe that the socio-demographic variables D15 and D16 have a causal relationship with churn, and this rank is most probably due to an artifact in the encoding of variables. Indeed, in the data preparation process, we noticed that data entries labeled as churners tend to have more often missing values in categorical variables. This issue was caused by the inclusion in the dataset of customers that already churned, even though some variables has no longer a valid value for these customers. The solution was to remove such entries from the dataset, but a related, unforeseen problem may have persisted for these categorical variables.

4.6 Discussion

The output of the GS and IAMB algorithm correspond to the Markov blanket, indistinguishably causes, effects and spouses of churn. On the other hand, mIMR and D2C focus explicitly on direct causes, but a numerical score is provided for each

variable. A choice of threshold has to be made on which variables we consider to be predicted as causes by these algorithms.

For the mIMR in figure 4.8, we include all variables up to the 9th and do not consider relevant the following ones. As discussed earlier, the 9th and 10th variables are mostly redundant with one another. Therefore, we can assume that at this threshold, the relevance term is becoming more important than the interaction term, and the causal property of the mIMR filter is not maintained anymore. In the case of mIMR with Gaussian assumption, the threshold is fixed at the 7th variable, since the following variables show a clear and distinct decrease in gain.

As for the output of the D2C algorithm, although there seems to be a large number of variables, most of them are different one-hot encodings of the same original variable. The most probable causes as predicted by D2C are solely the tariff plan, previous tariff plan, province of residence, data usage, and a hardware-related variable (H8). It seems reasonable to consider these 5 variables as predicted causes.

We summarize the results of the 5 algorithms (with the two implementations of mIMR) in table 4.1. For each variable, we indicate by which algorithm this variable was output, using the thresholds discussed above. There is no clear-cut consensus on which variables are causes. It is however reasonable to consider the number of messages and the duration of voice calls as *not* being inferred causes of churn, since only the GS algorithm outputs these variables. On the other hand, the tariff plan and previous tariff plan are given by all algorithms except for PC, GS, and mIMR with Gaussian assumption. We do not expect a categorical variable to be correctly predicted as a cause or not using the Gaussian assumption, the theory that the tariff plan and previous tariff plan are causes of churn are thus consistent with the observations.

In the absence of a formal procedure to assess these results, we summarize the results of causal inference from observational data to the following statements:

- The messages, the voice calls, as well as all variables not represented in table 4.1, are considered as most probably *not inferred causes of churn.*
- The tariff plan and previous tariff plan are considered as most probably *inferred causes of churn.*
- The inferred causal relationship of other variables in the table 4.1 is undecided.

In the light of our prior knowledge on the causes of churn exposed in section 4.3, we would expect the out of bundle variable to stand out more explicitly, but it is only given by mIMR with Gaussian assumption. However, recall that the distribution of the out of bundle can roughly be modeled as the exponential of a Gaussian (see figure 3.1). It is thus easy to understand why the other inference methods fail to report the causal link to the churn expected by our priors, due to the different estimators of mutual information. If it is true that the bill shock is a true cause of churn, then the results we observe in table 4.1 are not extraordinary. In other words, the credence in this theory is slightly, but not significantly, undermined.

	PC	GS	IAMB	mIMR 1	mIMR 2	D2C
Tenure	✗	✔	✗	✔	✔	✗
H14	✗	✔	✗	✔	✔	✗
D13	✗	✔	✗	✔	✔	✗
Data usage	✗	✔	✗	✔	✗	✔
Tariff plan	✗	✗	✔	✔	✗	✔
Prev tariff plan	✗	✗	✔	✔	✗	✔
Out of bundle	✗	✔	✗	✗	✔	✗
Number contracts	✗	✔	✗	✗	✔	✗
S7	✗	✔	✗	✔	✗	✗
H8	✗	✗	✗	✔	✗	✔
Province	✗	✗	✗	✔	✗	✔
Age	✗	✗	✗	✗	✔	✗
R14	✗	✗	✗	✗	✔	✗
Messages	✗	✔	✗	✗	✗	✗
Voice calls	✗	✔	✗	✗	✗	✗

Table 4.1 – Summary of the results of causal analysis. A green arrow indicates which variables are output by each algorithm. The output is the Markov blanket for PC, GS and IAMB, and direct causes for mIMR and D2C.

Two of the other causes of churn according to our prior knowledge are customer satisfaction and churn due to a move. As explained in section 4.3, none of the measured variables are direct proxies for these two putative explanations of churn. The interaction of variables present in table 4.1 might be the most direct display of the missing causal proxies. This would also explain the presence of variables seemingly unrelated to our prior knowledge. However, this hypothesis is somewhat far-fetched, and conforming to the faithfulness assumption by engineering relevant variables would be much more precise and productive.

The last of the four expected causes of churn correspond to a wrong positioning of the customer's tariff plan. This cause is supported by the results in a more obvious way. The tariff plan and previous tariff plan are reported as causes of churn by D2C and mIMR, and is also considered part of the Markov blanket by IAMB. This indicates that the wrong positioning theory is not unlikely.

Chapter 5

Conclusion

5.1 Churn prediction

We summarize in this section the main results of the experiments on churn prediction. The numerical results are reproduced in table 5.1. We observe that feature selection does not reduce performance if at least 30 of the most important variables are selected. Also, adding difference and ratio variables reduces the performance if no feature selection is conducted beforehand. Due to non-stationarity over the course of the 5 months of data, the trained models actually perform better on the test set than on the validation set. Principal component analysis indicates that in the test set (i.e. the last month of data) the churner population displays a larger magnitude of variations than in previous months of data. Regarding the type of contracts, churn is slightly easier to predict in the loyalty dataset than SIM only, due to the importance of time-related variables. Indeed, the churn is significantly higher at the end of the mandatory period of a loyalty contract, facilitating the prediction process. Irrespective of the contract type, important variables include, non-exhaustively: the tenure, the province, the tariff plan, the number of calls, and the data usage. On the one hand, the tenure and the number of contracts are observed to be monotonically associated with the churn probability. On the other hand, variables related to the amount paid by the customer are associated to more churn when they are increased, but the opposite is not true.

5.2 Causal analysis

The 5 different models used for causal analysis display different results. The PC algorithm shows no association between churn and the other variables. Of the two Markov blanket inference algorithms, GS algorithm provides a large Markov blanket for the churn variable, while the output of IAMB consists solely of the tariff plan and the previous tariff plan. The first implementation of mIMR, using a different mutual information estimator for each type of variable, infers mostly categorical variables as causes of churn. The D2C algorithm also returns categorical variables as likely causes. The second implementation of mIMR, by considering all variables as

	SIM only			SIM only Δ			Loyalty		
	20	**30**	**All**	**20**	**30**	**All**	**20**	**30**	**All**
AUROC	0.66	0.73	0.73	0.72	0.73	0.69	0.74	0.76	0.76
AUPRC	0.05	0.10	0.10	0.10	0.10	0.08	0.15	0.19	0.18
Lift at 10%	2.25	3.34	3.41	3.27	3.42	3.03	2.96	3.40	3.30
Lift at 5%	2.64	4.49	4.68	4.48	4.67	4.09	3.51	4.22	4.02
Lift at 1%	4.29	9.20	9.53	10.09	9.95	7.67	4.66	6.65	6.16

Table 5.1 – Summary of the results of prediction experiments on the test set. Highest values for each type of contract and for each evaluation measure are underlined for the test set.

continuous and Gaussian-distributed, selects only numerical variables as output. All these results differ from each other, but this is partly due to the different assumptions on the variable distributions laid by each of these methods. By considering prior knowledge on the causes of churn, we can conclude that the bill shock and the wrong positioning are likely hypotheses of churn. These two explanations, however, do not explain fully the results we obtain, and further investigations are needed to interpret and understand the inferred causal link between churn and other variables.

5.3 Internal validity

Internal validity refers to the confidence we can put into our results. More precisely, it represents the extent to which our results may be warranted, due to systematic errors. We list here threats to internal validity.

- The numerical implementation of causal inference methods relies on assumptions on the distribution of variables that does not always hold for our dataset. In particular, the mutual information between continuous variables is based on the linear assumption, whereas some variables are inherently exponentially distributed.
- The training phase of D2C is based on synthetic causal graphs, and we did not investigate alternative options for this phase. It is possible to create graphs that resemble more the data at hand, therefore reducing bias.
- In the causal inference experiments, sub-sampling with even class balancing is used to reduce computational load. When the number of selected samples is low, we may miss causal results not represented in the sample, or infer spurious ones.
- The dataset used in this study was created recently. Some problems occurred during the coding phase, leading to multiple iterations of the computation of the results. While all issues known to us have been solved in the dataset, some may have remained unnoticed.

- Not all parameters of the experimental setting for churn prediction have been assessed. In particular, we considered only three number of selected variables (20, 30 or all), and one ratio for class balancing (1 to 1). We thus have limited guarantees on the optimality of the experimental setting.

5.4 External validity

External validity is the extent to which our results can be generalized to other contexts. We list here threats to external validity.

- We use a dataset provided by Orange Belgium, which cannot be disclosed for confidential reasons. This is an obvious limit to the reproducibility, since it is impossible to verify our results using the same data.

- We can disclose the name of only a limited number of variables. Our conclusions on the impact of variables on churn are thus inherently limited to these variables.

- We do not use the maximum profit criterion, which is used in most recent studies on churn prediction. Although this requires some time investment to evaluate the relevant parameters, using this evaluation measure would improve our ability to compare our results to other studies.

- The principal component analysis, as well as the results on the test set, hints on the non-stationarity of the churner class. Our results pertain to 5 months in the year 2018, and our conclusions on variable importance and causality may less relevant today.

5.5 Added value for Orange

This thesis, conducted in collaboration with Orange, was also beneficial for them. The most obvious added value is that all conclusions we drawn from chapter 3 and 4 can directly be used to perform better churn prevention in production. This thesis displays an analysis work that would be difficult to perform for a data scientist at Orange due to time constraints. Also, this work is the start of a scientific collaboration between the Machine Learning Group from ULB and Orange Belgium. Future work is planned on causal inference applied to churn prevention, as presented in the next section. From a more practical point of view, the dataset used in this study has never been used before and this thesis showed that the variables included in this new dataset have a strong predictive power. Also, our code makes use of some R functions implemented originally for production models. We were thus able to assess the robustness and quality of these functions.

5.6 Future work

The limitations of this thesis have been discussed in sections 5.3 and 5.4. Some of them suggest improvements that can lead to more thorough and unbiased conclusions on churn prediction. In particular, we did not investigate the use of different prediction models or hyper-parameters. We believe that the prediction performance can be further improved through the exploration of alternative and optimized models, such as gradient boosting, support vector machine or others.

The evaluation of churn prediction models is based on the precision on a small subset of the test set (e.g. the lift at 5%). A possible extension of our Easy Ensemble model could consist in weighting each model based on the false positive rate calculated on the leftover portion of the training set. This procedure would favor models being less prone to consider non-churners as churners on unseen data.

A funding request has been introduced to the Innoviris organism for an "Applied PhD" program. This program consists in funding for doctoral research with an industrial partner, Orange in our case. The aim of this Ph.D. is to take further the work of this master thesis by conducting a more thorough evaluation of churn prediction, developing data visualization and understanding tools, and performing a more extensive causal inference research. In particular, the retention campaigns as they are conducted at Orange will be used as a way of verifying causal hypotheses, which is a rather unique opportunity in causal inference research. If the funding is granted, this Ph.D. will span on 4 years. A schematic overview of the research process is pictured in figure 5.1. The first year will be dedicated to churn prediction much like the third chapter of this master thesis, and the remaining 3 years will be dedicated to causal inference. At first, causal analysis from observational data will be used, as in the fourth chapter of this master thesis. Then, retention campaigns will be used to verify and define more precisely causal hypotheses formed from previous analyses. Predictions models will indicate on which customer the retention campaign should be focused.

5.7 Conclusion

We approached the churn prediction problem in the telecommunication industry with Orange Belgium customer data. A descriptive analysis of the dataset has been conducted, showing the non-linearity of variables, large overlap between churners and non-churners, non-stationarity and class imbalance. Predictive modeling of churn was achieved with a random forest classifier and the Easy Ensemble algorithm. In a series of experiments on churn prediction, we assessed the impact of variable selection, type of contract and use of engineered features. The results show that variable selection helps reducing computation time, but can decrease performance. Also, the addition of features consisting in the difference and the ratio of numerical variable seems to reduce the performance. The directionality of the impact of variables on churn is estimated through a sensitivity analysis. This shows that some variables are associated with the churn probability in a non-monotonic way. We ex-

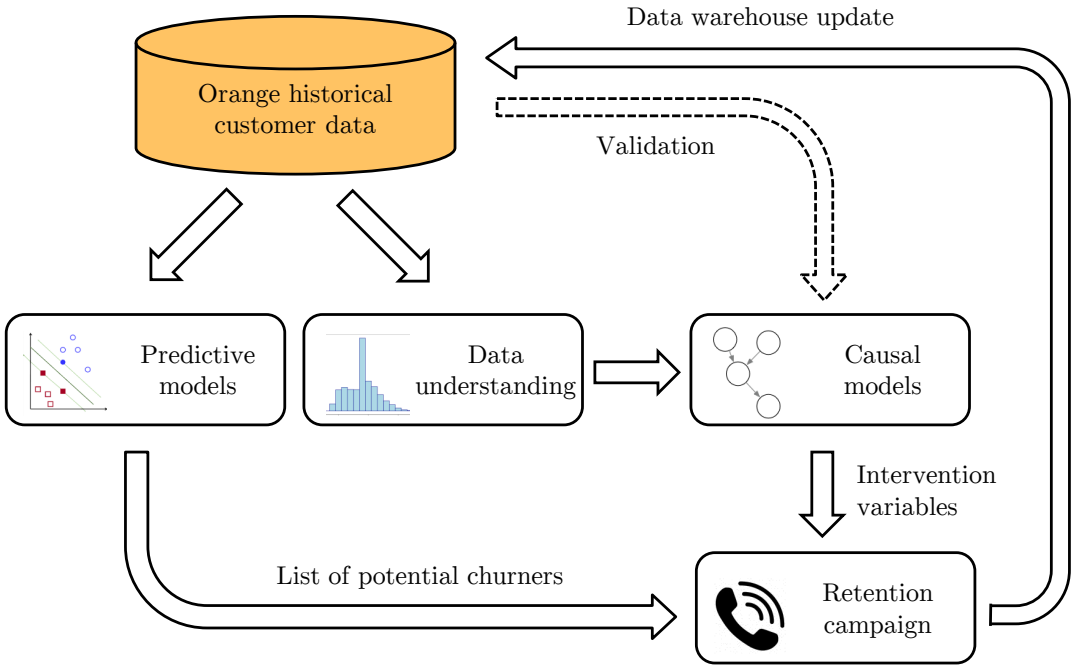


Figure 5.1 – Closed-loop validation process planned for the applied Ph.D. project.

plored the application of causal inference from observational data. More specifically, we applied 5 different causal inference methods, namely PC, Grow-Shrink (GS), Incremental Association Markov Blanket (IAMB), minimum Interaction Maximum Relevance (mRMR), and D2C. The results of these algorithms are varied and are partly consistent with prior knowledge of the causes of churn. This research highlights the difficulty of carrying out causal inference in a realistic setting, due to the large variety of variable types and distributions.

Bibliography

Ahmed, A. A., & Maheswari, D. (2017). Churn prediction on huge telecom data using hybrid firefly based classification. *journal abbreviation*, *18*(3), 215–220. doi:10.1016/j.eij.2017.02.002

Aliferis, C. F. [Constantin F], Tsamardinos, I., & Statnikov, A. (2003). Hiton: A novel markov blanket algorithm for optimal variable selection. In *Amia annual symposium proceedings* (Vol. 2003, p. 21). American Medical Informatics Association.

Barua, S., Islam, M. M., Yao, X., & Murase, K. (2014). Mwmote–majority weighted minority oversampling technique for imbalanced data set learning. *IEEE Transactions on Knowledge and Data Engineering*, *26*(2), 405–425.

Batista, G. E., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD explorations newsletter*, *6*(1), 20–29. doi:10.1145/1007730.1007735

Bell, A. J. (2003). The co-information lattice. In *Proceedings of the fifth international workshop on independent component analysis and blind signal separation: Ica* (Vol. 2003).

Bell, D. A., & Wang, H. (2000). A formalism for relevance and its application in feature subset selection. *Machine learning*, *41*(2), 175–195.

Bontempi, G., Birattari, M., & Bersini, H. (1999). Lazy learning for local modelling and control design. *International Journal of Control*, *72*(7-8), 643–658.

Bontempi, G., & Flauder, M. (2015). From dependency to causality: a machine learning approach. *The Journal of Machine Learning Research*, *16*(1), 2437–2457.

Bontempi, G., Haibe-Kains, B., Desmedt, C., Sotiriou, C., & Quackenbush, J. (2011). Multiple-input multiple-output causal strategies for gene selection. *BMC bioinformatics*, *12*(1), 458.

Bontempi, G., & Meyer, P. E. [Patrick E.]. (2010). Causal filter selection in microarray data. In *Proceedings of the 27th international conference on machine learning (icml-10)* (pp. 95–102).

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on computational learning theory* (pp. 144–152). ACM.

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5–32.

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). Classification and regression trees. wadsworth & brooks. *Cole Statistics/Probability Series*.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). Smote: Synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, 321–357.

Chen, C., & Breiman, L. (2004). Using random forest to learn imbalanced data. *University of California, Berkeley*.

Cheng, B., & Titterington, D. M. (1994). Neural networks: A review from a statistical perspective. *Statistical science*, 2–30.

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

Cover, T. M., & Thomas, J. A. (2012). *Elements of information theory*. John Wiley & Sons.

Dal Pozzolo, A., Caelen, O., & Bontempi, G. (2015). When is undersampling effective in unbalanced classification tasks? In *Joint european conference on machine learning and knowledge discovery in databases* (pp. 200–215). Springer.

Dal Pozzolo, A., Caelen, O., Le Borgne, Y.-A., Waterschoot, S., & Bontempi, G. (2014). Learned lessons in credit card fraud detection from a practitioner perspective. *Expert systems with applications*, *41*(10), 4915–4928.

Dal Pozzolo, A., Caelen, O., Waterschoot, S., & Bontempi, G. (2013). Racing for unbalanced methods selection. In *International conference on intelligent data engineering and automated learning* (pp. 24–31). Springer.

De Caigny, A., Coussement, K., & De Bock, K. W. (2018). A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees. *European Journal of Operational Research*, *269*(2), 760–772. doi:10.1016/j.ejor.2018.02.009

Fisher, R. A. (1937). *The design of experiments*. Oliver and Boyd; Edinburgh; London.

Fleuret, F. (2004). Fast binary feature selection with conditional mutual information. *Journal of Machine learning research*, *5*(Nov), 1531–1555.

Fonollosa, J. A. (2016). Conditional distribution variability measures for causality detection. *arXiv preprint arXiv:1601.06680*.

Good, P. (2013). *Permutation tests: A practical guide to resampling methods for testing hypotheses*. Springer Science & Business Media.

Guyon, I., Aliferis, C. et al. (2007). Causal feature selection. In *Computational methods of feature selection* (pp. 75–97). doi:10.1201/9781584888796.ch4

Hadden, J., Tiwari, A., Roy, R., & Ruta, D. (2007). Computer assisted customer churn management: State-of-the-art and future trends. *Computers & Operations Research*, *34*(10), 2902–2917.

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *2008 ieee international joint conference on neural networks (ieee world congress on computational intelligence)* (pp. 1322–1328). IEEE.

Heckerman, D. (1998). A tutorial on learning with bayesian networks. In *Learning in graphical models* (pp. 301–354). Springer.

Hido, S., Kashima, H., & Takahashi, Y. (2009). Roughly balanced bagging for imbalanced data. *Statistical Analysis and Data Mining*, *2*, 412–426. doi:10.1002/sam.10061

Hitchcock, C. (1997). Probabilistic causation.

Huang, B., Kechadi, M. T., & Buckley, B. (2012). Customer churn prediction in telecommunications. *Expert Systems with Applications*, *39*(1), 1414–1425.

Idris, A., & Khan, A. (2014). Ensemble based efficient churn prediction model for telecom. In *Frontiers of Information Technology (FIT), 2014 12th International Conference on* (pp. 238–244). doi:10.1109/fit.2014.52

ITU. (2018). Itu releases 2018 global and regional ict estimates. Retrieved April 13, 2019, from https://www.itu.int/en/ITU-D/Statistics/Pages/stat/

Kayaalp, F. (2017). Review of customer churn analysis studies in telecommunications industry. *Karaelmas Fen ve Mühendislik Dergisi*, *7*(2), 696–705.

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

Koller, D., & Sahami, M. (1996). Towards optimal feature selection (1996) proc. 13th int'l. conf. *Machine Learning*, 284–292.

Kolmogorov, A. (1956). On the shannon theory of information transmission in the case of continuous signals. *IRE Transactions on Information Theory*, *2*(4), 102–108.

Krzanowski, W. J., & Hand, D. J. (2009). *Roc curves for continuous data*. Chapman and Hall/CRC.

Kubat, M., Matwin, S. et al. (1997). Addressing the curse of imbalanced training sets: One-sided selection. In *Icml* (Vol. 97, pp. 179–186). Nashville, USA.

Liu, X.-Y., Wu, J., & Zhou, Z.-H. (2009). Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *39*(2), 539–550. doi:10.1109/tsmcb.2008.2007853

Margaritis, D., & Thrun, S. (2000). Bayesian network induction via local neighborhoods. In *Advances in neural information processing systems* (pp. 505–511).

McGill, W. (1954). Multivariate information transmission. *Transactions of the IRE Professional Group on Information Theory*, *4*(4), 93–111.

Meyer, P. E. [Patrick Emmanuel], Schretter, C., & Bontempi, G. (2008). Information-theoretic feature selection in microarray data using variable complementarity. *IEEE Journal of Selected Topics in Signal Processing*, *2*(3), 261–274.

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

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (icml-10)* (pp. 807–814).

Olle, G. D. O., & Cai, S. (2014). A hybrid churn prediction model in mobile telecommunication industry. *International Journal of e-Education, e-Business, e-Management and e-Learning*, *4*(1), 55.

Olsen, C., Meyer, P. E., & Bontempi, G. (2008). On the impact of entropy estimation on transcriptional regulatory network inference based on mutual information. *EURASIP Journal on Bioinformatics and Systems Biology*, *2009*(1), 308959.

Óskarsdóttir, M., Bravo, C., Verbeke, W., Sarraute, C., Baesens, B., & Vanthienen, J. (2017). Social network analytics for churn prediction in telco: Model building, evaluation and network architecture. *Expert Systems with Applications*, *85*, 204–220. doi:10.1016/j.eswa.2017.05.028

Óskarsdóttir, M., Van Calster, T., Baesens, B., Lemahieu, W., & Vanthienen, J. (2018). Time series for early churn detection: Using similarity based classification for dynamic networks. *Expert Systems with Applications*, *106*, 55–65. doi:10.1016/j.eswa.2018.04.003

Pearl, J. (2002). Causality: Models, reasoning, and inference. *IIE Transactions*, *34*(6), 583–589.

Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (8), 1226–1238.

Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, *1*(1), 81–106.

Quinlan, J. R. (1987). Simplifying decision trees. *International journal of man-machine studies*, *27*(3), 221–234.

Sathe, S., & Aggarwal, C. C. (2017). Similarity Forests. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 395–403). KDD '17. doi:{10.1145/3097983.3098046}

Scutari, M. (2009). Learning bayesian networks with the bnlearn r package. *arXiv preprint arXiv:0908.3817*.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell system technical journal*, *27*(3), 379–423.

Spirtes, P. (2010). Introduction to causal inference. *Journal of Machine Learning Research*, *11*(May), 1643–1662.

Spirtes, P., & Glymour, C. (1991). An algorithm for fast recovery of sparse causal graphs. *Social science computer review*, *9*(1), 62–72.

Spirtes, P., Glymour, C., & Scheines, R. (1993). *Causation, prediction, and search*. doi:10.1007/978-1-4612-2748-9

Stripling, E., vanden Broucke, S., Antonio, K., Baesens, B., & Snoeck, M. (2018). Profit maximizing logistic model for customer churn prediction using genetic algorithms. *Swarm and Evolutionary Computation*, *40*, 116–130. doi:10.1016/j.swevo.2017.10.010

Ting, K. M. (2002). An instance-weighting method to induce cost-sensitive trees. *IEEE Transactions on Knowledge & Data Engineering*, (3), 659–665.

Tsamardinos, I., Aliferis, C. F. [Constantin F.], & Statnikov, A. (2003). Time and sample efficient discovery of Markov blankets and direct causal relations. In

Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 673–678). doi:10.1145/956804.956838

Tsamardinos, I., Aliferis, C. F. [Constantin F], Statnikov, A. R., & Statnikov, E. (2003). Algorithms for large scale markov blanket discovery. In *Flairs conference* (Vol. 2, pp. 376–380).

Umayaparvathi, V., & Iyakutti, K. (2016). Attribute selection and customer churn prediction in telecom industry. In *2016 international conference on data mining and advanced computing (sapience)* (pp. 84–90). IEEE.

Vafeiadis, T., Diamantaras, K. I., Sarigiannidis, G., & Chatzisavvas, K. C. (2015). A comparison of machine learning techniques for customer churn prediction. *Simulation Modelling Practice and Theory*, *55*, 1–9. doi:10.1016/j.simpat.2015.03.003

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

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

Verbraken, T., Verbeke, W., & Baesens, B. (2013). A novel profit maximizing metric for measuring classification performance of customer churn prediction models. *IEEE Transactions on Knowledge and Data Engineering*. doi:10.1109/tkde.2012.50

Veropoulos, K., Campbell, C., & Cristianini, N. (1999). Controlling the sensitivity of support vector machines. *Proceedings of International Joint Conference Artificial Intelligence*.

Yen, S.-J., & Lee, Y.-S. (2009). Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Systems with Applications*, *36*(3), 5718–5727.

Yu, L., & Liu, H. (2004). Efficient feature selection via analysis of relevance and redundancy. *Journal of machine learning research*, *5*(Oct), 1205–1224.

Zhang, S., Zhang, C., & Yang, Q. (2003). Data preparation for data mining. *Applied artificial intelligence*, *17*(5-6), 375–381. doi:10.1080/713827180

Zhu, B., Baesens, B., & vanden Broucke, S. K. (2017). An empirical comparison of techniques for the class imbalance problem in churn prediction. *Information sciences*, *408*, 84–99. doi:10.1016/j.ins.2017.04.015