# Introduction

The telecommunication market is a highly competitive market. Customers are able to choose between a variety of suppliers and it is relatively easy to switch between them. This results in a very high churn rate and represents a serious problem for the suppliers as the costs for attracting new customers exceed the costs of customer retention (e.g., Lu, 2002). An important and necessary task for the telecommunication companies is to predict the probability of customers to cancel the contract in the near future. This information can be used then to start a loyalty initiative such as offering a discount or a new mobile phone to the customer. To gain valuable knowledge out of the data telecommunication providers have about their customers, data mining techniques can be used.

This working task deals with such a customer churn problem. The provided real-world data of 100,000 customers of an anonymous telecommunication provider are described by 173 attributes, such as customer characteristic and usage behaviour. The observed customers were with the company for at least six month and they were sampled during July, September, November of 2001 and January of 2002. At the time a customer was observed the input variables were calculated based on the previous four months. If a customer churned within a period of 31-60 days after the observation time, the data provides the information of “churn=1”, otherwise “churn=0”. The data set is divided into two parts of equal size (training and test set). One set represents the training set including the information if a customer churned as explained above. This set will be used to build our model. Our goal is then to predict the churn probabilities for each customer in the test set. The higher the probability the higher the risk that a customer will churn within the period of 31-60 days after the observation time. An accurate prediction gives the company sufficient time to react to the information and retain the customer.

In doing so we proceed in accordance with the KDD process (knowledge discovery in databases) (e.g., Fayyad, et al., 1996). The first step *selection* is already done as our target data is selected (see above). In chapter 2 we will do some exploratory data analysis which is part of the *preprocessing* to get an overview of the data. The following steps of the KDD process will be divided into two chapters (chapter 3 and 4). Our approach is to run two iterations during the model building process. Within the first iteration we will focus on rather simple methods (for example for data cleaning and reduction). We will start with data cleaning such as missing value and outlier handling. Next within *transformation* we will focus on data reduction (through correlation analysis and feature selection) as well as encoding. Within the next step of *data mining* we will apply data mining algorithms to search for patterns in our data. In our case the data mining model is classification and seven popular classification methods will be applied: Logistic Regression, Neural Network, Random Forest, Naïve Bayes, K-nearest Neighbors, Decision Tree (J48) and Support Vector Machines. At the end of chapter 3 we will interpret the results and evaluate which approaches worked well and which should be improved. In the second iteration we will handle the needs for improvement identified in the first iteration and focus on rather complex methods if required. We will see for example that during preprocessing and transformation the missing value and outlier handling, feature selection and feature extraction should be improved. During data mining we will only focus then on those models in the second iteration that performed well in the first iteration. Moreover we will also apply a heterogeneous ensemble method that makes a linear combination of the individual models. In doing so we want to further increase the predictive accuracy of the individual models. During the whole procedure we will always keep in mind that the KDD process is not a strict sequence, but rather an iterative process.

# Feature Selection first iteration

The data set consists of 172 input variables and probably not all of them are relevant for the prediction. One part of vertical data reduction is the feature or variable selection where we want to find a subset of relevant variables. Out of a variety of feature selection approaches we decide within the first iteration to use a simple Filter Approach. The approach is to pre-screen the variables prior to model building and only use those variables as predictors that pass a certain statistical criterion. In our case we use the rfe function out of the caret package in R (http://topepo.github.io/caret/filters.html). Through an underlying ANOVA model it is tested for each variable if the mean is statistically different between the two classes “churn=1” and “churn=0”. If so, the variable is regarded as relevant and therefore should be considered in the prediction model. There are several other approaches for feature selection that are more effective, for instance the Wrapper Approach or using models with built-in feature selection. The latter have the advantage that feature selection is included in the objective function that is optimized which is not given when feature selection is separated from all the other steps (like in the Filter Approach). Furthermore a disadvantage of the Filter Approach is that it is used in our case in a univariate manner, so that redundancy and interactions of variables are not taken into account. Nevertheless considering that we have high-dimensional data we decide in the first iteration for this approach because of its much lower computational costs. As a result of the filter we get 68 variables that can be viewed as relevant and thus will be used as input variables for the data mining in the following.

Additionally it makes sense to check if the selected variables had many missing values in the original data set before imputation. It would not be reasonable to include variables that have a very high missing value rate because the information content would be low. Within our selected variable subset only two variables have more than 3% missing values, namely the variables “lor” (length of residence; 30.38% missing values) and “adults” (number of adults in household; 23.84% missing values). These two variables do not seem to be very relevant for the churn problem. But as we do not want to lose information and the variables still have values for more than two third of the observations, we keep them as predictors for now.

# Feature Selection second iteration

During the first iteration we noticed that the filter approach only selected numerical variables although we assume that also some categorical variables are relevant for our predictions. To improve the feature selection process we now apply a wrapper approach. Using a prediction model the variables are here treated as inputs and model performance is the output that shall be optimized. For our problem we decide to use a Random Forest Variable Importance (RFVI) method which is included in the caret package. We expect it to have higher computational costs because a random forest has to be trained prior to importance ranking, but probably it will perform better than the filter approach. The RFVI consists of three main steps:

* Step 1: For each tree in the random forest the classes are predicted for the out-of-the-bag cases and the number of correct class votes is counted.
* Step 2: Each variable is randomly manipulated (continuous as well as categorical variables).
* Step 3: For each tree and each manipulated variable step 1 is repeated.

The assumption is that manipulating an important variable should change the accuracy whereas an unimportant variable should not have any effect. The figure from step 3 is now subtracted from the figure of step 1 and then averaged over all trees to get the raw importance score (RIS). The RIS gives the average loss in accuracy due to manipulating the variables’ values and is therefore the indicator for the variable importance ranking.

Unfortunately computational costs with all variables are extremely high and after one/two days we stopped the execution. When working with high-dimensional data it is common to use a hybrid strategy, so reducing the number of variables by firstly using a filter and afterwards a wrapper approach. As our filter only selected continuous variables, we continue our wrapper approach with using these selected continuous variables and additionally all categorical variables. After implementing this adjustment we execute the random forest training and variable importance measurement again.

Figure XX shows the result from our RFVI and we can see for instance that XXX is regarded as the most important variable. Furthermore we notice that many categorical variables are regarded as highly important. This confirms our decision to not rely on the filter approach. The categorical variables are now recoded into binary ones which we can see for example by looking at variable XXX. This is the binary variable for the original categorical variable XXX with level XXX. This leads to a large increase in dimensionality as we have XXX variables now in total.

The variable importance ranking is only the first step of feature selection. Now we have to decide how many variables should be selected.

…

Moreover we decide to include a categorical value if only one level of the variable is part of our selected variables (EXAMPLE).

Having done that we get XX selected features, out of which XX% are continuous and XX% are categorical variables. Again we checked the missing value rate of the selected features and …

# References (Introduction)

Lu, J. (2002). Predicting customer churn in the telecommunications industry––An application of survival analysis modeling using SAS. *SAS User Group International (SUGI27) Online Proceedings*, 114-27.

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). The KDD process for extracting useful knowledge from volumes of data. Communications of the ACM, 39(11), 27-34.