Decision Support Systems 72 (2015) 72–81

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Including high-cardinality attributes in predictive models: A case study in churn prediction in the energy sector

Julie Moeyersoms *, David Martens

Faculty of Applied Economics, University of Antwerp, Belgium

ARTICLE INFO

Article history:
Received 11 August 2014
Received in revised form 15 December 2014
Accepted 6 February 2015
Available online 16 February 2015

Keywords:
Data mining
Predictive modeling
High-cardinality attributes
Churn prediction

ABSTRACT

High-cardinality attributes are categorical attributes that contain a very large number of distinct values, like for example: family names, ZIP codes or bank account numbers. Within a predictive modeling setting, such features could be highly informative as it might be useful to know that people live in the same village or pay with the same bank account number. Despite this notable and intuitive advantage, high-cardinality attributes are rarely used in predictive modeling. The main reason for this is that including these attributes by using traditional transformation methods is either impossible due to anonymization of the data (when using semantic grouping of the values) or will vastly increase the dimensionality of the data set (when using dummy encoding), thereby making it difficult or even impossible for most classification techniques to build prediction models. The main contributions of this work are (1) the introduction of several possible transformation functions coming from different domains and contexts, that allow the inclusion of high-cardinality features in predictive models. (2) Using a unique data set of a large energy company with more than 1 million customers, we show that adding such features indeed improves the predictive performance of the model significantly. Moreover, (3) we empirically demonstrate that having more data leads to better prediction models, which is not observed for "traditional" data. As such, we also contribute to the area of big data analytics.

1. Introduction

The increasingly widespread collection and processing of data enable companies to use these data assets to improve decision making. A popular way to do so is by using predictive modeling. Examples can be found in different domains like credit scoring, where predictive models are used to separate good from bad loan applications [5,26,44] and in marketing where likely adopters are identified [6,29].

Two types of data are commonly used in order to build predictive models: structured data (e.g. socio-demographic data, number of products purchased) on the one hand and relational or behavioral data (e.g. transactional invoicing data, phone call or other networked data) on the other hand. Previous work has shown that using behavioral data is extremely valuable and improves the performance of the model significantly [20,21,27]. Unfortunately, these behavioral data are very unique and are exclusively preserved and accessible to the banks, Telco operators and Googles of the world. Structured data, on the other hand, is widely available. Consider the example of an energy company: they have extensive information on customer's socio-demographics such as address, age, and family size but they do not have data on transactions between customers such as phone calls or payments.

The focus in this paper is on one specific type of structured data, namely high-cardinality attributes. These are categorical attributes with a very large number of distinct values, such as: bank account number, family names and ZIP codes. Surprisingly, high-cardinality features are rarely used in predictive modeling. The main reason is that high-cardinality attributes are difficult to handle as including them would imply that the dimensions of the data set will quickly explode. Consider the example of the energy company that has information on the family names of the customers. Including this attribute with dummy encoding implies that millions of dummies will be created: one for every family name. As a consequence, the computational effort will increase substantially, and it will even be impossible for most of the techniques to cope with such high dimensions.

This case study discusses churn prediction in an energy context, where the aim is to predict which customers are most likely to churn and thus switch to another energy supplier. The data is stemming from a large energy company in Belgium with more than 1 million customers and includes several high-cardinality attributes. The question remains whether or not it is possible to include this type of attributes without expanding the dimensionality of the data set and if it actually improves the prediction model significantly. This research is built around the following three research questions:

1. Is it useful to include high-cardinality attributes?
2. How to transform and include such high-cardinality attributes?

* Corresponding author at: Prinsstraat 13, 2000 Antwerp, Belgium.
E-mail address: Julie.Moeyersoms@uantwerp.be (J. Moeyersoms).

http://dx.doi.org/10.1016/j.dss.2015.02.007
0167-9236/© 2015 Elsevier B.V. All rights reserved.

3. Does adding more data yield a better generalization performance of the prediction model?

This last question refers to the issue of Big Data, which is defined as data that is so big that traditional data processing systems cannot cope with it [25]. In the case of behavioral or relational data, it has been shown that adding more data points effectively improves model performance [21]. For socio-demographic data on the other hand, traditional predictive analytics may not receive much benefit from increasing the amount of data beyond a certain point [32]. Whether or not the same principle applies for high-cardinality data will be shown later in the paper.

Note that although we focus on a case study in the energy sector, this research is relevant for all kinds of retailers as this high-cardinality data is largely available everywhere. An important final remark is that a sufficient amount of data is required when working with high cardinality attributes, in order to apply the techniques proposed in this work. The rest of the paper is structured as follows: Section 2 discusses churn prediction and the importance of the topic in the Belgian energy sector. In the next section, 'high-cardinality' data is explained in more detail as well as possible techniques to transform and include them in the data set. Section 4 defines the methodology of the experiments and provides information on the data set. Next, Section 5 describes the results of the experiments. Finally, the last section concludes the paper and identifies some interesting issues for future research.

2. Churn prediction in an energy setting

The market value of the Belgian electricity and gas market was estimated at $22.4 billion in 2011 [13] and is expected to grow steadily over the next years [37]. This clearly indicates the size and importance of this market for the Belgian economy. Belgium fully liberalized its energy market in July 2007, thereby allowing customers to choose their natural gas and electricity supplier. Since this liberalization, the energy landscape in Belgium is in a continuous change [53]. A recent report of the European Commission [13] shows that the rate of people who switch suppliers has increased rapidly. In 2011 the churn already reached 10% for the electricity retail market and 11.2% for the entire gas retail market. Due to these recent developments, customer churn prediction has received a large amount of attention from energy suppliers.

The goal of churn prediction is to indicate which customers are most likely to leave the company. In this way, those customers can be offered incentives to stay and the churn rate can be reduced [45]. This also allows companies to decrease the costs of customer retention campaigns, because they can target more efficiently the customers with the highest probability to churn [49]. The economic value of customer retention has received much attention in the literature [45,49]. A decrease in customer switching can lead to many benefits for the company because of the following reasons [45,51]: first, retaining existing clients is five to six times less expensive than to acquire new customers [7,9]. Second, it is shown that long-term customers are more willing to recommend the company to other people [9,16]. This positive word-of-mouth associated with increased customer retention can, in turn, lead to lower marketing costs to acquire new clients [22]. Finally, long-term customers are less sensitive to competitive pull. They pay less attention to competitor's advertising and are less likely to compare prices of their own supplier with those of other suppliers [9,42].

Many research has been done on customer churn prediction and its usage [8,30,39,50,52], which proves the importance of the topic. For an extensive overview of literature on churn prediction modeling we refer to [51]. None of the studies listed in [51] includes high-cardinality data. Yet, such data is readily available (think of the ZIP code or last name of customers) and including these data can lead to a superior predictive performance. Although in data mining research the focus is often on benchmarking different classification techniques in order to find the best performing technique in terms of predictive performance, it is argued that data quality is at least equally important [4]. Good data quality is often the best way to augment the performance of a prediction model. Therefore, the focus in this paper is on data inclusion rather than the classification technique applied.

3. Including high-cardinality features

The inclusion of attributes with a high cardinality in prediction models constitutes the subject of this case-study. First, a more detailed explanation of high-cardinality attributes is given. Next, possible transformation techniques are listed that allow us to include categorical features with or without a high cardinality.

3.1. High-cardinality versus traditional nominal data

Structured attributes can be continuous (implying that they contain real numbers and are defined over a continuous range) or nominal (meaning that they can take only a finite number of values).[1] Examples of continuous attributes are the *amount of the invoice* or *kWh used*. An example of a nominal feature is *type of contract*, that can take three different values: *electricity*, *gas* or *electricity and gas*. The cardinality of a nominal feature can be defined as the number of distinct values that attribute can take [31]. Thus, for the example of the feature *type of contract*, the cardinality is 3. The features with a small cardinality are referred to as '*traditional*' nominal attributes. The features with a very high cardinality on the other hand, are named '*high-cardinality*' attributes. The latter are generally removed from the data as including them using dummy encoding will expand the dimensions of the data set quickly, thereby impeding the model building. Based on literature [49], it can be noted that features with more than 100 different values are mostly discarded from the analysis, hence we consider this to be the threshold and name all features with more than 100 distinct values high-cardinality features. Based on this assumption, three features of the energy data set are identified as high-cardinality data: *family name*, *bank account number* and *ZIP code*.[2]

3.2. Transformation techniques

In data mining literature, nominal attributes are usually included in the data set using dummy encoding or grouping methods. These methods can also be applied on high-cardinality features and are discussed in this section. Moreover, three other methods are proposed that allow the transformation of high-cardinality features into continuous attributes. All methods are listed in this section and illustrated with an example in Table 1.

3.2.1. Dummy encoding

A common way to include nominal features is to use dummy encoding where the M categorical values are transformed into M new dichotomous variables that are coded as 1 or 0. This method allows the adding of these variables to the model and provides an easy interpretation of the output since one variable matches with one value of the original variable. The main drawback of this method is that if M becomes large, the computational effort will increase significantly.

In the case of traditional nominal features, dummy encoding is an appropriate method to use. Applying dummy encoding to high-cardinality attributes, on the other hand, could create millions of dummies. Since most of the predictive modeling techniques do not scale to such dimensions, dummy encoding cannot be used for high-cardinality data. For this reason, previous studies did not take into account this type of attributes. The first part of Table 1 shows an example where dummy encoding

[1] A third variable type is ordinal, which, like nominal variables, is categorical, but with an order in the values. For example age encoded as young, middle-aged and old. The proper way to handle such variables is through thermometer encoding.

[2] In our data set, these features include even a larger amount of distinct values, going from 1000 up to more than 1 million.

Table 1
Example of transformations: the original data set includes high-cardinality features (ZIP codes). In this example the ZIP codes from individual customers (C1, C2 etc.) are transformed using different methods: (1) dummy encoding: for each ZIP code a dummy variable is created. (2) Semantic grouping: ZIP codes are clustered into provinces which are dummy encoded afterwards. (3) Transformation to a continuous variable: a continuous score is calculated for each ZIP code and each ZIP code is replaced with its score in a second phase.

	Zip code
C1	2000
C2	1800
...	...
C100	2000
...	...
C3M	4000

→ *Dummy encoding*

	1000	1100	...	1800	...	2000	...	4000
C1	0	0		0		1		0
C2	0	0		1		0		0
...	...							
C100	0	0		0		1		0
...	...							
C3M	0	0		0		0		1

	Zip code
C1	2000
C2	1800
...	...
C100	2000
...	...
C3M	4000

→ *Semantic grouping*

	Ant	Lim	FB	EF	WF	Hain	WB	Nam	Li	Lux
C1	0	0	1	0	0	0	0	0	0	0
C2	1	0	0	0	0	0	0	0	0	0
...	...									
C100	1	0	0	0	0	0	0	0	0	0
...	...									
C3M	0	0	0	0	0	0	0	0	1	0

	Zip Code	Label
C1	2000	1
C2	1800	0
...	...	...
C100	2000	1
...	...	...
C3M	4000	0

→ *Continuous Transformation*

	Score
C1	0.83
C2	-0.64
...	...
C100	0.83
...	...
C3M	-0.04

is applied on the ZIP code of every individual customer in the data set. For each ZIP code, a dummy variable is created. In the case of Belgium this means that 589 dummies (which are equal to the number of municipalities) are created, ranging from ZIP code 1000 to 9992. In the United States there are approximately 43,000 ZIP codes.

3.2.2. Semantic grouping

In semantic grouping, the aim is to identify semantically meaningful groups. If there is a possibility to identify logical groups in the case of high-cardinality data, this technique is particularly useful since it allows the user to reduce the amount of distinct values and to create an understandable grouping. For our data set, this technique can be applied to the ZIP codes. That is, the ZIP codes will be grouped by province, thereby creating 10 (there are 10 provinces in Belgium) different values instead of a few thousands. Afterwards, dummy encoding is used to create 10 different features. This is shown in the second part of Table 1. It should be noted that grouping could imply a certain loss of information since the more fine-grained or detailed data (ZIP codes) is replaced by grouped and thus more high-level data (provinces). However, for most of the high-cardinality data no logical grouping is even possible due to the nature of the data or because the data is anonymous. In our case, the high-cardinality features *family names* and *bank account* cannot be grouped in a logical way and another method should be used in order to include this data in the prediction model.

3.2.3. Transformation to continuous attribute

The previous two methods are used to transform nominal attributes into dummy encoded attributes. Another possibility is to transform the nominal into continuous features whose values are correlated with the target label (i.e. churn). Three different transformation functions are proposed in this paper. It should be noted that these methods differ from the previous ones as information on the target label is used for the calculation. In order to avoid overfitting on the data, the scores for the continuous transformations are calculated on a separate part of the data. That is, the prediction model is built on a training set and tested on a (hold-out) test set. The calculation of the scores is done in a preprocessing step on a separate part of the training set, which is not used for model building or parameter tuning. The exact set-up of the experiments will be explained in Section 5.1.

Metric 1: weight of evidence. The first transformation method is the Weight of Evidence (WOE) [17,43]. The WOE will transform a categorical attribute with many different nominal values into a continuous attribute. For a binary classification problem, it can be defined as [17,43]:

$$\mathrm{WOE}_i^X = \ln\left(\frac{C_i^X/TC}{N_i^X/TN}\right) \tag{1}$$

where TC and TN define the total number of churners and non-churners respectively in the data set (or more generally number of instances of class c_1 versus class c_2). C_i^X and N_i^X denote the number of churners and non-churners for the ith value of attribute X. Take for example the attribute ZIP code with value 5000. Assume that the data set consists of 1000 customers of which 300 have churned ($TC = 300$ and $TN = 700$). Suppose 100 customers in the data set have ZIP code 5000, of which 20 have churned ($C_i^X = 20$ and $N_i^X = 80$), then the WOE for ZIP code 5000 would be:

$$\mathrm{WOE}_{5000}^{\mathrm{ZIP\ code}} = \ln\left(\frac{20/300}{80/700}\right) = -0.5390.$$

Eq. (1) can also be written as:

$$\mathrm{WOE}_i^X = \ln\left(\frac{C_i^X/TC}{N_i^X/TN}\right) = \ln\left(\frac{C_i^X}{N_i^X}\right) - \ln\left(\frac{TC}{TN}\right). \tag{2}$$

Or for the example:

$$\mathrm{WOE}_{5000}^{\mathrm{ZIP\ code}} = \ln\left(\frac{20/300}{80/700}\right) = \ln\left(\frac{20}{80}\right) - \ln\left(\frac{300}{700}\right) = -0.5390.$$

This illustrates that WOE consists of two parts: a variable part that depends on the individual characteristics of the customer (the ZIP code) and a fixed part that depends on the data. This last part is a constant for the data (i.e. the part of the training data that is used for calculating the WOE) and ensures that the score gets centralized around zero. That is, a score equal to zero implies that the probability of churning for this customer is equal to the overall churn rate in the data. A negative or positive score signifies a below- or above-average chance to churn respectively. As the WOE-score implies the use of a logarithmic function, there are some preconditions that need to be met. In order to comply with these conditions, a modified calculation of WOE as proposed in [54] was implemented. More specifically if C_i^X (or N_i^X) equals zero, the authors propose to add one data point so that C_i^X (or N_i^X) equals one and a certain number of data points is added to N_i^X (or C_i^X) so that the overall ratio of the added points equals the ratio of the whole data set (TC/TN). For these specific cases, the WOE can be calculated as follows [54]:

$$\mathrm{WOE}_i^X = \begin{cases} \ln\left(\dfrac{\left(C_i^X + (TC/TN)\right)/TC}{TC}\right) = \ln\left(\dfrac{\left(C_i^X \times TN\right) + TC}{TC}\right), & \text{if } N_i^X = 0 \\ & \text{and } C_i^X > 0 \\ \ln\left(\dfrac{1/TC}{(N_i^X + (TN/TC))/TN}\right) = \ln\left(\dfrac{TN}{(N_i^X \times TC) + TN}\right), & \text{if } C_i^X = 0 \\ & \text{and } N_i^X > 0. \end{cases}$$

As stated before, it is important to notice that in order to avoid overfitting, the WOE score should be calculated on a separate part of the training data. When building the predictive model in a next step, the predictive power should be evaluated in terms of out-of-sample performance for test cases that were not used to construct the scores. Therefore, it is possible that a certain value X_i in the test data has not appeared in the data used to construct the scores. In this case, a WOE-score of zero is assigned to the data point, which implies an average chance to churn.

As can be seen from Eq. (1), the WOE will estimate the probability that a customer will churn by taking into account the target variable (churn). An important advantage of the method is that it has an easy interpretation: the higher the score, the higher the probability of churn. Another advantage of the weight of evidence method is that you do not increase the number of variables in the data as every value of the nominal variable can be replaced by its WOE-score. This characteristic makes the WOE method extremely useful for high-cardinality data. Moreover, the WOE-score is easy to calculate, which is essential when working with huge data sets.

In the literature, the WOE method is sometimes used as a transformation technique in the credit scoring sector [17,18,43,47]. However, in the latter the WOE-transformation is applied solely on traditional nominal data. To the best of our knowledge, this method has not been applied on high-cardinality data, or in other domains despite its particular useful characteristics for the transformation of nominal features.

Metric 2: supervised ratio. A different way to transform a high-cardinality into a continuous feature is by using a supervised ratio, which is inspired from a social network perspective (see further):

$$\mathrm{SR}_i^X = \frac{C_i^X}{C_i^X + N_i^X} \tag{3}$$

where C_i^X and N_i^X are again the number of churners and the number of non-churners for the ith value of attribute X respectively. The value of the score is situated between 0 and 1, for which a higher score implies a higher chance of churn. If we take the previous example, the supervised ratio for the value 5000 of the attribute ZIP code would be:

$$\mathrm{SR}_{5000}^{\mathrm{ZIP\ code}} = \frac{20}{100} = 0.20.$$

As for the WOE-transformation, customers with the same value i for attribute X are assigned the same score. Also, the score is easy to calculate and its interpretation is straightforward. Again, the scores need to be calculated on a separate part of the data in order to evaluate the predictive power of the prediction model in terms of out-of-sample performance. An 'unseen' value for X therefore receives an SR-score equal to the average churn rate ($TC/(TC + TN)$).

It should be noted that both the WOE-score and the supervised ratio could be looked at from a social network perspective. In domains where information on explicit connections between customers is available, taking advantage of this social network among customers has shown to be remarkably effective [1,11,20,35,38]. Unfortunately, in most business domains, no information on the real, explicit social network of customers is available. Again, consider the example of the energy company, that does not have information on the explicit connections between customers such as phone calls or payments among each other. Luckily, other detailed information on the customers, in the form of high-cardinality data, is readily available. The two transformation functions described above are based on the assumption that these high-cardinality features could be used as a proxy for the real social network between the customers [2,20,23,41]. The key idea is that similarities in some high-cardinality attributes imply useful similarity between customers [28]. This is different from a real social network, since there is no reason to believe that the customers have a true social relationship with one another [28]. The only assumption that can be made in this electricity case is that customers with the same *family name*, *bank account number* or *ZIP code* are likely to be more similar than random customers. If in this case, customers with the same ZIP code are assumed to be connected in a (pseudo-) social network, the supervised ratio can be interpreted as an implementation of the weighted-vote relational neigbor classifier [24]: Of all 100 customers in the training set who have ZIP code 5000, 20 have churned which means that a new customer with ZIP code 5000 receives a score of 0.20, based on the assumption that he has the same chance to churn as his network neighbors. The same perspective can be used for the WOE-score: customers with the same value for a high-cardinality attribute are supposed to be network neighbors and thus receive the same score, which implies a similar chance to churn.

Metric 3: Perlich ratio. A related research topic is the integration of information from many-to-many or one-to-many relationships like in the work of [31]. Examples of these relationships are a person that watched many movies or called many numbers. The first part in Fig. 1 shows the example of customers buying many books (ISBNs), thereby creating a many-to-many relationship. The main challenge in this situation is to include these relationships by using aggregation operators. Simple aggregation operators such as *count* (in this case the number of books bought) could be used but do not necessarily perform well for these many-to-many relationships. Therefore, more adequate aggregation operators are proposed in the work of [31], that captures more information about the value distributions. More specifically, they include high-cardinality data (with many-to-many relationships) by constructing a reference vector per class (e.g. churn or non-churn) and calculating the distance between a new data point and the reference vector. This implies that each value receives a score according to its distance from the average number of positive or negative cases for that value, thereby including information about the value distributions.

The main difference with our study lies in the type of data that is used. As explained before, Perlich et al. [31] use information from many-to-many relationships and therefore the focus is on the introduction of more sophisticated aggregation operators for these relationships. For the high-cardinality data in our setting on the other hand, no aggregation is needed. That is, every customer has only one ZIP code, family name and bank account number and thus no many-to-many relationships need to be handled. Graphically, this is shown in the second part in Fig. 1. Nevertheless, the techniques proposed in the work of [31] (which is elaborated on by [40]), could also be applied on the high-cardinality attributes of the energy data. Yet, it can be argued that for our problem, using the WOE or supervised ratio is more convenient as these scores are very easy to calculate and the interpretation of these scores is more straightforward.

In order to support this statement, we give a small example of different customers (C1–C4) each living in a city with a certain ZIP code. The case vectors, which are basically the dummy encoded vectors, and reference vectors as described in the work of [31] are given in Table 2. The elements r_i^1 and r_i^0 for the reference vectors are equal to the average number of occurrences of value X_i related to a positive or negative target label respectively. It should be noted that for high-cardinality features, the elements of the positive reference vector are equal to the supervised ratio or numerator of the WOE-score (C_i^X/TC) whereas the elements of the negative reference vector equal the denominator of the WOE-score. Therefore, this score can be written as[3]:

$$\mathrm{Score}_{\mathrm{WOE}}^{\mathrm{ZIP}_i} = \ln\left(\frac{C_i^{\mathrm{ZIP}}/TC}{N_i^{\mathrm{ZIP}}/TN}\right) = \ln\left(\frac{r_i^1}{r_i^0}\right). \tag{4}$$

In a next step, what we name the Perlich ratio (PR) can be calculated as the cosine distance between the case vector and positive reference vector:

$$\mathrm{PR}_{cosine1}^{\mathrm{ZIP}_i} = \frac{\left(r_i^1\right) * 1 + 0 + \ldots + 0}{1 * \sqrt{\left(r_1^1\right)^2 + \left(r_2^1\right)^2 + \ldots + \left(r_m^1\right)^2}}.$$

[3] C_i^{ZIP} is the same as C_i^X from Eq. (1) but because we are taking the example of ZIP code, the X is replaced by 'ZIP'.

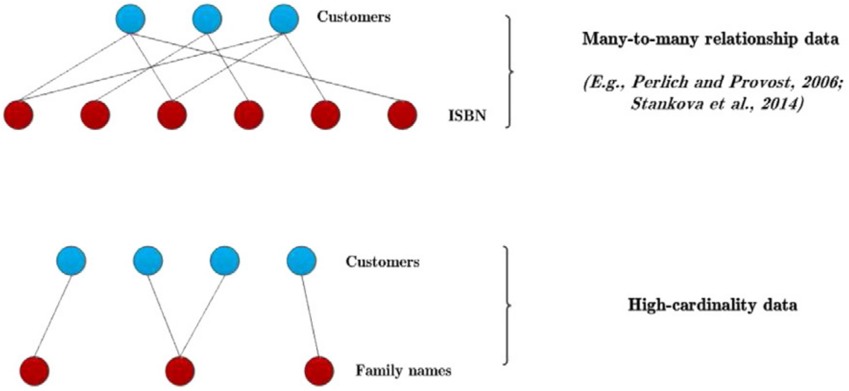


Fig. 1. Comparison of high-cardinality data with many-to-many relationship data.

Note that many other distance measures are used in the study of [31] but we chose to elaborate on the cosine distance due to its common use and suitability for our setting.

Replacing the reference vectors based on Eq. (4) gives us:

$$\begin{aligned}\mathrm{PR}^{\mathrm{ZIP}_i}_{cosine1} &= \frac{C_i^{\mathrm{ZIP}}/TC}{\sqrt{\left(\frac{C_1^{\mathrm{ZIP}}}{TC}\right)^2+\left(\frac{C_2^{\mathrm{ZIP}}}{TC}\right)^2+\ldots+\left(\frac{C_m^{\mathrm{ZIP}}}{TC}\right)^2}}\\ &= \frac{C_i^{zip}/TC}{\sqrt{\frac{\left(C_1^{\mathrm{ZIP}}\right)^2+\left(C_2^{\mathrm{ZIP}}\right)^2+\ldots+\left(C_m^{\mathrm{ZIP}}\right)^2}{(TC)^2}}}\\ &= \frac{C_i^{zip}}{\sqrt{(C_1^{\mathrm{ZIP}})^2+(C_2^{\mathrm{ZIP}})^2+\ldots+(C_m^{\mathrm{ZIP}})^2}}\end{aligned}$$

which can be simply written as:

$$\mathrm{PR}^{\mathrm{ZIP}_i}_{cosine1} = \frac{C_i^{\mathrm{ZIP}}}{\alpha}.$$

In a similar way, the cosine distance between the case vectors and the negative reference vector can be written as: $\mathrm{PR}^{\mathrm{ZIP}_i}_{cosine0} = N_i^{\mathrm{ZIP}}/\beta$ where $\beta = \sqrt{\left(N_1^{\mathrm{ZIP}}\right)^2+\left(N_2^{\mathrm{ZIP}}\right)^2+\ldots+\left(N_m^{\mathrm{ZIP}}\right)^2}$. After normalization, the final Perlich ratio can be calculated as:

$$norm\mathrm{PR} = \frac{\mathrm{PR}^{X_i}_{cosine1}}{\mathrm{PR}^{X_i}_{cosine1}+\mathrm{PR}^{X_i}_{cosine0}}.$$

Again, ‘unseen’ values might appear in the test data. In this case, a score equal to $\frac{\alpha}{\alpha+\beta}$ is assigned to these data points. By simplifying the method used in Perlich et al. for high-cardinality data without many-to-many relationships, it can be seen that this method could also be used for our problem. The main drawback is that the derived score is less comprehensible than the WOE-score or supervised ratio as the α and β do not have a clear interpretation.

4. Data and methodology

4.1. Data and techniques

The data used in this study is stemming from a large energy supplier in Belgium and contains more than 1 million data points (customers).[4] It covers the customer records from August 2011 until August 2012 and consists of socio-demographic and consumption information. Table 3 gives an overview of the attributes included in the data set.

The data set contains both continuous and nominal features. As explained before, we distinguish between two types of nominal attributes: traditional features and high-cardinality features. As can be seen from Table 3 the attributes *bank account number*, *family names* and *ZIP code* are indicated as high-cardinality data, meaning that they each include more than 100 distinct values.

Different classification techniques are considered, representing various types of models. First, a C4.5 decision tree [36] is applied, which is grown in a recursive way by partitioning the training records into a purer subset by using the entropy measure. Next, we considered a statistical classifier, more specifically a logistic regression (logit). Finally, a support vector machine [48] with linear kernel was used. More details on these techniques are given in the next paragraphs.

4.1.1. C4.5

C4.5 is a popular tree induction technique based on information theoretic concepts. More specifically, it uses entropy to measure how informative an attribute is in splitting the data [36]. Entropy quantifies the order (or disorder) among observations with respect to the classes. If we consider p_1 (p_0) to be the proportion of examples of class 1 (0) in sample S, we can state that the entropy equals 1 when $p_1 = p_0 = 0.5$ (maximal disorder, minimal order) and 0 (maximal order, minimal disorder) when all observations belong to the same class, $p_1 = 0$ or $p_0 = 0$. In order to decide upon which attribute to split at a given node, the gain criterion is used [36]. Gain is defined as the expected reduction in entropy due to splitting on attribute X_i. C4.5 uses a gain ratio criterion and applies normalization in order to avoid that attributes with many distinct values will be favored [36].

4.1.2. Logit

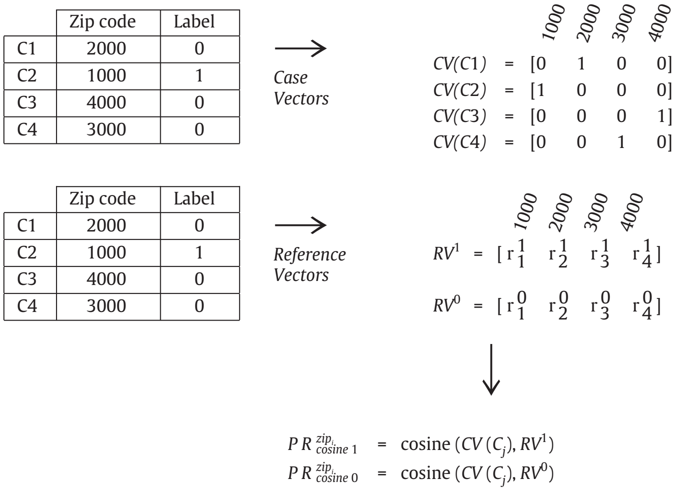
Logistic regression is one of the most widely applied data mining techniques for binary classification. It can be derived from the posterior class probabilities in the simple case of multivariate Gaussian class distributions with identical covariance matrices. In logistic regression, the posterior class probabilities are between 0 and 1 and sum up to 1. Given a data set of N data points $D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ with input data $\boldsymbol{x}_i \in \mathbb{R}^n$ and corresponding binary labels $y_i \in \{0, 1\}$, the logistic regression estimates the probability $P(y = +1|x)$ as follows [3]:

$$P(y = +1|x) = \frac{1}{1+\exp(-(b+\omega^T x))}$$

where $\boldsymbol{x}_i \in \mathbb{R}^n$ is an n-dimensional input vector, $\boldsymbol{\omega}$ is the parameter vector and b is the intercept. The parameters $\boldsymbol{\omega}$ and b are then estimated using the maximum likelihood.

[4] Note that the real size of the data set is not mentioned due to confidentiality reasons.

Table 2
Example of the Perlich ratio.



	Zip code	Label
C1	2000	0
C2	1000	1
C3	4000	0
C4	3000	0

→ *Case Vectors*

$$\begin{array}{lccccc} & & 1000 & 2000 & 3000 & 4000 \\ CV(C1) & = & [0 & 1 & 0 & 0] \\ CV(C2) & = & [1 & 0 & 0 & 0] \\ CV(C3) & = & [0 & 0 & 0 & 1] \\ CV(C4) & = & [0 & 0 & 1 & 0] \end{array}$$

	Zip code	Label
C1	2000	0
C2	1000	1
C3	4000	0
C4	3000	0

→ *Reference Vectors*

$$\begin{array}{lccccc} & & 1000 & 2000 & 3000 & 4000 \\ RV^1 & = & [r_1^1 & r_2^1 & r_3^1 & r_4^1] \\ RV^0 & = & [r_1^0 & r_2^0 & r_3^0 & r_4^0] \end{array}$$

↓

$$PR^{zip_i}_{cosine\ 1} = \text{cosine}\,(CV(C_j), RV^1)$$
$$PR^{zip_i}_{cosine\ 0} = \text{cosine}\,(CV(C_j), RV^0)$$

4.1.3. Support vector machine (SVM)

The support vector machine is a learning procedure based on the statistical learning theory [48]. Given a training set of N data points $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ with input data $\mathbf{x}_i \in \mathbb{R}^n$ and corresponding binary class labels $y_i \in \{-1, +1\}$, the SVM classifier should fulfill the following conditions (see Fig. 2): [10,26,46,48]:

$$\begin{cases} \mathbf{w}^T\varphi(\mathbf{x}_i) + b \geq +1, & \text{if } y_i = +1 \\ \mathbf{w}^T\varphi(\mathbf{x}_i) + b \leq -1, & \text{if } y_i = -1 \end{cases} \tag{5}$$

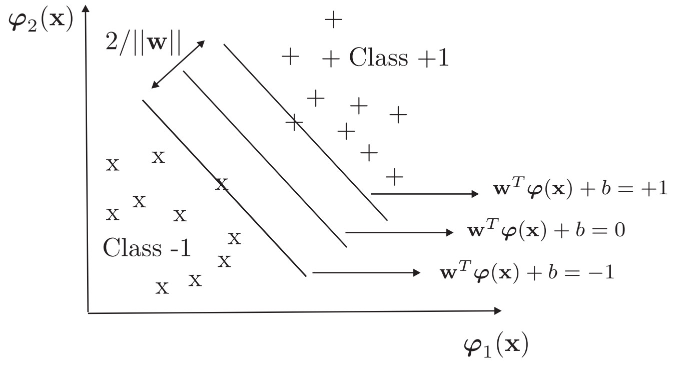
which is equivalent to

$$y_i\left[\mathbf{w}^T\varphi(\mathbf{x}_i) + b\right] \geq 1, \quad i = 1, \ldots, N. \tag{6}$$

The non-linear function $\varphi(\cdot)$ maps the input space to a high (possibly infinite) dimensional feature space. In this feature space, the above inequalities basically construct a hyperplane $\mathbf{w}^T\varphi(\mathbf{x}) + b = 0$ discriminating between the two classes. By minimizing $\mathbf{w}^T\mathbf{w}$, the margin between both classes is maximized.

In primal weight space the classifier then takes the form

$$y(\mathbf{x}) = \text{sign}\left[\mathbf{w}^T\varphi(\mathbf{x}) + b\right], \tag{7}$$



$\varphi_2(\mathbf{x})$
$2/||\mathbf{w}||$
+ Class +1
$\mathbf{w}^T\varphi(\mathbf{x}) + b = +1$
$\mathbf{w}^T\varphi(\mathbf{x}) + b = 0$
Class -1
$\mathbf{w}^T\varphi(\mathbf{x}) + b = -1$
$\varphi_1(\mathbf{x})$

Fig. 2. Illustration of SVM optimization of the margin in the feature space.

Table 3
Attributes included in the energy data.

Continuous	Traditional nominal	High-cardinality nominal
Age	Gender	ZIP-code
Average amount of bill	Type of contract	Family names
Contacts with company	Package	Bank account
	Payment method	

but, on the other hand, is never evaluated in this form. The convex optimization problem can be defined as:

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} \mathcal{J}(\mathbf{w}, b, \boldsymbol{\xi}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^N \xi_i \tag{8}$$

subject to

$$\begin{cases} y_i\left[\mathbf{w}^T\varphi(\mathbf{x}_i) + b\right] \geq 1 - \xi_i, & i = 1, \ldots, N \\ \xi_i \geq 0, & i = 1, \ldots, N. \end{cases} \tag{9}$$

The variables ξ_i are slack variables which are needed to allow misclassifications in the set of inequalities (e.g. due to overlapping distributions). The first part of the objective function tries to maximize the margin between both classes in the feature space and is a regularization mechanism that penalizes for large weights, whereas the second part minimizes the misclassification error. The positive real constant C is the regularization coefficient and should be considered as a tuning parameter in the algorithm.

This leads to the following classifier:

$$y(\mathbf{x}) = \text{sign}\left[\sum_{i=1}^N \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b\right], \tag{10}$$

whereby $K(\mathbf{x}_i, \mathbf{x}) = \varphi(\mathbf{x}_i)^T\varphi(\mathbf{x})$ is taken with a positive definite kernel satisfying the Mercer theorem. The Lagrange multipliers α_i are then determined by optimizing the dual problem. For the kernel function $K(\cdot,\cdot)$, this study opts to use the linear kernel:

$$K(\mathbf{x}, \mathbf{x}_i) = \mathbf{x}_i^T\mathbf{x}.$$

For low-noise problems, many of the α_i will typically be equal to zero (sparseness property). The training observations corresponding to non-zero α_i are called support vectors and are located close to the decision boundary. Due to regularization, the SVM is able to achieve a good overall performance, even when the dimensionality of the data is high. The LibLinear Library [14] is used for the linear SVM, allowing it to work with sparse matrices. As only the non-zero elements are stored, the linear SVM is suitable for very high-dimensional data. The majority of the classification techniques however do no support sparse matrices, which implies that it will not fit in memory on a normal computer when the dimensionality grows.

4.2. Performance measures

Different measures are used to test the results of the experiments in an appropriate way. As the focus lies on identifying likely churners, we chose to use true positive rate and precision to check how many of the actual or predicted churners are correctly identified. Also, the lift is calculated to show the improvement that the model provides over random guessing. Finally, the AUC is shown as it provides a simple figure-of-merit for the performance of the classifier.

4.2.1. Lift

The lift of a prediction model represents the improvement that a prediction model provides as compared to a random guess. Therefore, the data is first ranked according to the score that is given by the prediction model (where a high score implies a higher estimated likelihood of

Table 4
Confusion matrix for binary classification of churn.

		Actual	
		Churner	Non-churner
Predicted	Churner	True positive (TP)	False positive (FP)
	Non-churner	False negative (FN)	True negative (TN)

churn) with the most likely churner ranked on the top. The lift is the degree to which positive instances are "pushed up" above the negative instances in the list [34]. For example, consider a data set of 100 customers of whom 50 have churned. If the list is sorted randomly, one expects to find only half of the churners, giving a lift of 1 (=0.50/0.50). If the list is ordered by a ranking classifier, more than half of the churners should appear in the top half of the list, producing a lift greater than 1. A perfect classifier would rank all the churners in the first half of the list so at half of the list, we saw all churners and the lift would be 2 (=1/0.5) [34].

4.2.2. TPR and precision

Class-wise decomposition of the classification of cases yields a confusion matrix as specified in Table 4. Recall or true positive rate (TPR) can be defined as the proportion of positive examples which are predicted to be positive (TP/(TP + FN)). In other words, this is the percentage of churners that is correctly classified by the model. In other research domains, alternative terms are employed for this concept, such as sensitivity, detection probability or hit rate. Precision on the other hand, measures how many of the instances classified as positive, are actual churners. Just as lift, these numbers are dependent on a single threshold to define the predicted class. To obtain a metric over all possible cutoff values, the AUC metric is commonly used.

4.2.3. Area under the ROC-curve (AUC)

The receiving operating characteristic curve (ROC) is a 2-dimensional graphical illustration of the sensitivity on the Y-axis versus the X-axis for various values of the classification threshold. The ROC-curve illustrates the behavior of a classifier without taking into account the class distribution, thereby decoupling classification performance from this factor [12,15,33]. In order to compare the ROC curves of different classifiers, the *area under the receiver operating characteristics curve* (AUC or AUROC) is calculated. An intuitive interpretation for AUC is that it is equivalent to the probability that the classifier will rank a randomly chosen positive instance (in this case, a churner) higher than a randomly chosen negative instance (in this case a non-churner) [15]. Its values range between 0 and 1, where a higher value implies better model performance. Note that since the area under the diagonal corresponding to a pure random classification model is equal to 0.50, a good classifier should yield an AUC much larger than 0.50.

Table 6
Results of linear SVM for: the base (BA) model where no high-cardinality features are included, the model where high-cardinality features are grouped semantically ($BA + HC_{group}$), and the models where the high-cardinality attributes are encoded with dummies ($BA + HC_{dummy}$), weight of evidence ($BA + HC_{\mathrm{WOE}}$), supervised ratio ($BA + HC_{\mathrm{SR}}$) and Perlich ratio ($BA + HC_{\mathrm{PR}}$).

	BA	$BA +$ HC_{group}	$BA +$ HC_{dummy}	$BA +$ HC_{WOE}	$BA +$ HC_{SR}	$BA +$ HC_{PR}
AUC	67.70	72.50	72.83	**74.39**	74.10	74.25
	(±0.26)	(±0.12)	(±0.14)	(±0.13)	(±0.12)	(±0.13)
TPR (1%)	2.60	3.84	**4.14**	3.80	3.95	3.94
TPR (5%)	12.62	15.92	16.41	17.44	17.45	**17.70**
Precision (1%)	21.20	31.26	**33.74**	30.95	32.18	32.08
Precision (5%)	20.56	25.95	26.73	28.42	28.44	**28.84**
Lift (0.1%)	1.92	2.86	**6.19**	5.28	4.77	4.15
Lift (1%)	2.60	3.84	**4.14**	3.80	3.95	3.94

5. Experiments

5.1. Set-up of the experiments

Before applying the different classification techniques on the data, preprocessing needs to be performed in order to make the data suitable for analysis. Transformation of the different features is done as a preprocessing step. Continuous features are normalized and traditional nominal features are included after dummy encoding. For the high-cardinality attributes, multiple transformations are explored in this experiment. As mentioned in the Introduction, the first research question that is posed is whether or not it is useful to include high-cardinality features. In order to answer this question, a benchmarking model that does not contain the high-cardinality features is built and compared to different models that do include high-cardinality features. The benchmarking model therefore only includes the traditional nominal attributes (dummy encoded) and continuous attributes (normalized). For simplicity we name the benchmarking model the Base (BA) model.

The second research question that was put forward is how to include high-cardinality features. As stated before, the nature of this data entails additional challenges and not all transformation methods are suitable. In order to answer our second question, four different models are built, each using a different technique to transform high-cardinality features as mentioned in Section 3.2. A detailed overview of all transformations for the different models is given in Table 5. The first model applies semantic grouping in case a logical grouping is possible for the feature. That is, in our case only the attribute ZIP code can be grouped in a logical way (provinces) and, after being dummy encoded, added to the data of the base model. The other high-cardinality features, *family names* and *bank account number* are excluded from the data set since no grouping is possible for these features. This model is denoted as $BA + HC_{group}$.

Table 5
Transformation of the features in the different models.

Type of attribute	Transformations for different models					
	BA	$BA + HC_{group}$	$BA + HC_{dummy}$	$BA + HC_{\mathrm{WOE}}$	$BA + HC_{\mathrm{SR}}$	$BA + HC_{\mathrm{PR}}$
Continuous						
Age	Normalized	Normalized	Normalized	Normalized	Normalized	Normalized
Amount of bill	Normalized	Normalized	Normalized	Normalized	Normalized	Normalized
Contacts with company	Normalized	Normalized	Normalized	Normalized	Normalized	Normalized
Traditional nominal						
Gender	Dummy	Dummy	Dummy	Dummy	Dummy	Dummy
Type of contract	Dummy	Dummy	Dummy	Dummy	Dummy	Dummy
Package	Dummy	Dummy	Dummy	Dummy	Dummy	Dummy
Payment method	Dummy	Dummy	Dummy	Dummy	Dummy	Dummy
High-cardinality nominal						
ZIP code	/	Grouped (provinces)	Dummy	WOE	SR	PR
Family names	/	/	Dummy	WOE	SR	PR
Bank account	/	/	Dummy	WOE	SR	PR

In a next model, the high-cardinality features are all dummy encoded and added to the data of the base model. This implies that for every value of *ZIP code*, *bank account number* and *family names*, a separate feature is created. The model is named $BA + HC_{dummy}$. Note that this leads to a huge feature explosion, which only the SVM technique was able to handle. Next, we assess two models where the high-cardinality features are included by using the transformation functions explained in Section 3.2.3. The model where the high-cardinality attributes are added to the data set using a WOE-transformation is referred to as $BA + HC_{\mathrm{WOE}}$. In case the transformation is performed using the supervised ratio, we name the resulting model $BA + HC_{\mathrm{SR}}$. Finally, the model where the Perlich ratio is used to transform the high-cardinality attributes is included and is indicated as $BA + HC_{\mathrm{PR}}$.

The last research question handles the relationship between adding more data and the predictive performance of the model. In order to verify whether or not adding more data improves the generalization behavior for the different models, various sample sizes were tested.

A 10 fold cross-validation is applied on the data, thereby splitting the data each time in a training and test (hold-out) set. All customers in the training data that churned are labeled as the known churners and all the customers in the test data are scored (concealing the true customers' status for the experiment until the time of evaluation). In order to avoid overfitting, the predictive power has to be evaluated in terms of out-of-sample performance for test cases that were not used to construct the scores. Therefore, the WOE-score, supervised ratio and Perlich ratios are calculated on a separate part of the training set. The remaining part of the training set is again divided in a training part (3/4) on which the model is trained and a validation part (1/4) to optimize the hyperparameters using a gridsearch. Finally, the predictive power is evaluated on the separate test set. The results are shown in the next section and aim to answer the research questions.

5.2. Results and discussion

Table 6 shows the average results (for the whole data set) over the 10 folds of the linear SVM for the different models. The best performance for each measure is denoted in boldface and underlined. Performances that are not significantly worse at a 5% confidence level (according to a Wilcoxon signed rank test) are tabulated in bold face. Differences at the 5% but not at the 1% level are reported in normal script. Significant underperformances at the 1% level are emphasized in italics. The research questions can now be answered based on these results.

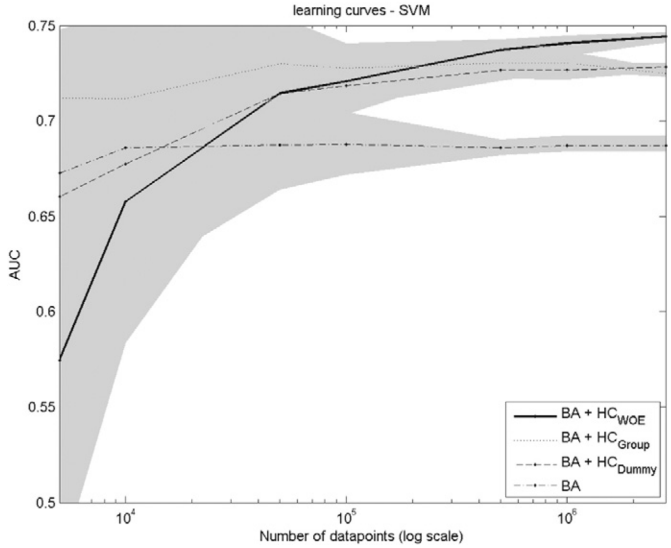


Fig. 3. Learning curves for the different models (SVM).

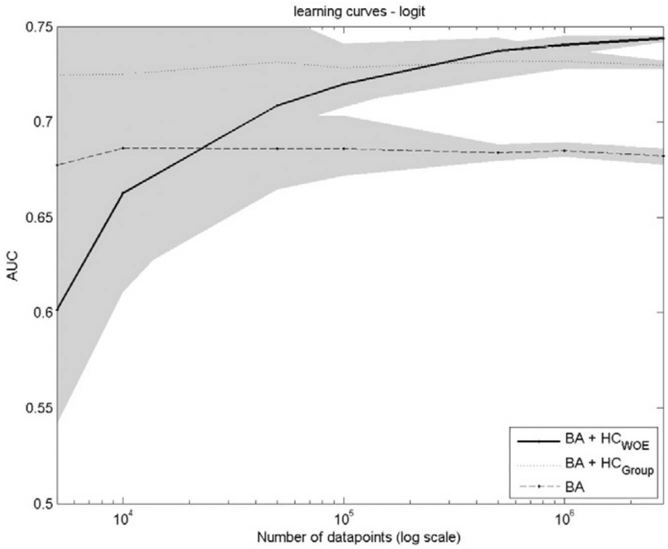


Fig. 4. Learning curves for the different models (logit).

Question 1: is it useful to include high-cardinality features?

Table 6 reports the average results over the 10 folds (with standard deviation between brackets) of the different models. It is shown that in terms of AUC, the model without high-cardinality features (*BA*) performs worse than all other models, which do include this type of data. Also in terms of true positive rate, precision and lift, the *BA* model performs significantly worse than the other models. This clearly indicates that the inclusion of high-cardinality attributes indeed gives extra information and improves the model performance.

Question 2: how to transform and include high-cardinality features?

From Table 6, it can be noticed that in terms of AUC, the WOE-method performs best with an AUC of 74.39. In terms of true positive rate and precision however, other methods such as dummy encoding and the Perlich ratio perform better. The grouping method is in this case always inferior to the other models that include high-cardinality features. A possible explanation is that the more detailed information gets lost. Moreover, the high-cardinality features bank account number and family name are not included in this model as no logical grouping is possible in these cases. This means that less data is used in this model.

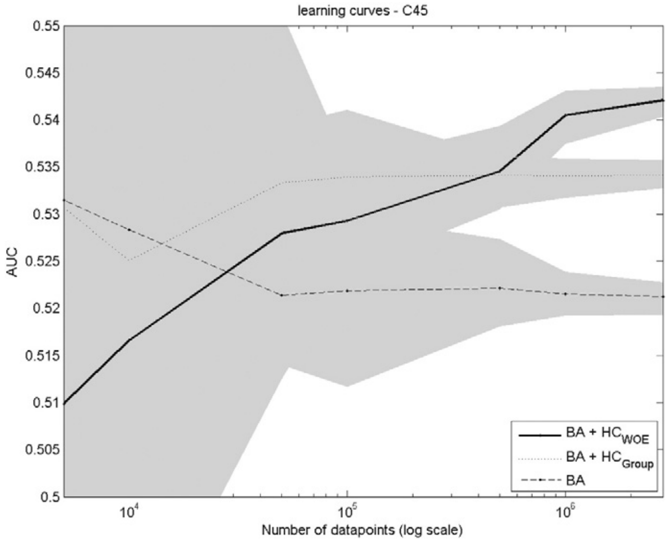


Fig. 5. Learning curves for the different models (C4.5).

The dummy model performs best in terms of true positive rate (1%), precision (1%) and lift. Unfortunately, as stated before, techniques that can handle large dimensions are required for this method. All other techniques, like for example logit and C4.5 for our experiments, were not able to handle this dimensionality and therefore the dummy model cannot be used. When only looking at the last three models that include high-cardinality attributes (WOE, SR and PR), one can notice that these models perform well but that neither of them has the best results for all performance measures. That is, in terms of AUC and lift (0.1%), the WOE gives the best results whereas SR performs best in terms of TPR (1%), precision (1%) and lift (1%). The PR has the highest TPR (5%) and precision (5%). This shows that these models provide a good (and sometimes even better) alternative to dummy encoding. Moreover, they offer a valid solution for the dimensionality problem which is crucial when working with large data sets. In addition to a good performance, the WOE and supervised ratio methods have a clear interpretation and are easy to calculate. The Perlich ratio on the other hand, is somewhat more difficult to interpret.

Question 3: does adding more data improve the generalization performance of the prediction model?

Figs. 3, 4 and 5 show the learning curves for the linear SVM, logit and C4.5 respectively. The X-axis shows the size of the sample size (in log-scale) and the Y-axis denotes the AUC. The extreme-value-boundaries of the 10 folds are represented by the gray areas (where for each sample size, the lower boundary represents the minimum AUC and the upper boundary the maximum AUC of the 10 folds). The basis model as well as the grouping and dummy method are shown in the graphs. Moreover, one of the models where the high-cardinality features are included by transforming the data into continuous variables is shown as well (we chose to include the WOE-model because of its AUC and interpretability). The learning curves for the model with dummy encoding are not shown for logit and C4.5 because these techniques could not handle the larger sample sizes. From all three graphs, it is clear that for the base model (*BA*), the maximum AUC is reached quickly and the model does not improve by adding more data. The same conclusion can be drawn for the grouping model, and in the case of the SVM also for the dummy model, although the maximum AUC is achieved at a higher sample size. This is consistent with previous work (see e.g. [32]) and further validates the common practice of sampling such data when building predictive models. For the WOE model on the other hand, the AUC keeps on increasing with higher sample sizes. The same results are obtained when using the SR or PR model. This implies that adding more data in case of high-cardinality data, indeed improves the predictive performance of the model.

6. Conclusion and future research

This paper describes several metrics that can be used to include high-cardinality attributes. Methods from different domains and contexts are brought together and applied on this problem. The results in a churn prediction setting demonstrate that including such data can lead to substantial improvements in predictive performance. What is interesting to notice is that having more data actually leads to better predictive models. Although this is rather obvious in hindsight (how can there be valuable information in e.g. ZIP code unless you have ample of data on all ZIP codes), it is in contradiction to the common practice of sampling.

On the flip side of this point: it is very hard to replicate our experiments on publicly available data sets (such as data from the popular UCI data repository [19]). These data sets are typically samples where high-cardinality attributes are not present. Only when big data is at hand, can such analyses be done. This of course is a broader issue for research in the big data arena in general. With a real-life data set from one of the largest electricity providers in Belgium, we are able to show the use and effectiveness of the proposed transformations, which allow the inclusion of features with a very high cardinality. This adds useful information to the prediction model, inducing better churn prediction. In this way, the energy company is able to identify the likely churners more accurately, allowing the company to improve the efficiency of customer retention campaigns.

Not only churn applications can benefit: clearly in any domain where predictive modeling is currently being done on a person's level, where for each data point/person we know at least the person's last name or ZIP code, can the same metrics be applied. Such applications include customer acquisition, response modeling, default prediction, and fraud detection. On top of that, even when the data is not related to a person, high-cardinality attributes are often available. Just to demonstrate its widespread use, two examples: the 2014 KDD Cup challenge[5] aims at predicting whether a project proposal will be funded or not, where high-cardinality attributes are available in the form of school ID and district ID. Imagine the explosion in dimensions if one would add a dummy for each of the 30,000 school IDs or 7000 district IDs. The proposed metrics on the other hand are fast to calculate and can easily be included in any subsequent classification algorithm. In a corporate fraud detection setting, the last name of the CEO or the specific NACE (activity) code is another potentially useful high-cardinality attribute.

To conclude, research in predictive modeling has a tendency to lead to a continuous flow of new and complex algorithms, which are often applicable only in a very specific setting (if an implementation is made available in the first place). The rather intuitive transformation techniques that we set forth to include attributes that are already available, try to answer the call for better data rather than better modeling techniques. As such, they have a wide applicability and can lead to substantial performance improvements.

Acknowledgments

The authors would like to thank the Belgian energy company and the Flemish Research Council (FWO) for the financial support (Grant G.0827.12N). We are very grateful to the Belgian energy company for allowing us to write about our work on this problem. Special thanks also go to the anonymous reviewers and Professor Cornelissens from the University of Antwerp for the valuable comments and suggestions to improve the manuscript.

References

[1] S. Aral, L. Muchnik, A. Sundararajan, Distinguishing influence-based contagion from homophily-driven diffusion in dynamic networks, Proceedings of the National Academy of Sciences 106 (51) (2009) 21544–21549.

[2] P. Baecke, D.V. den Poel, Including spatial interdependence in customer acquisition models: a cross-category comparison, Expert Systems with Applications 39 (15) (2012) 12105–12113.

[3] B. Baesens, T.V. Gestel, S. Viaene, M. Stepanova, J. Suykens, J. Vanthienen, Benchmarking state-of-the-art classification algorithms for credit scoring, The Journal of the Operational Research Society 54 (6) (2003) 627–635.

[4] B. Baesens, C. Mues, D. Martens, J. Vanthienen, 50 years of data mining and or: upcoming trends and challenges, JORS 60 (S1) (2009) s16–s23.

[5] B. Baesens, T. Van Gestel, S. Viaene, M. Stepanova, J. Suykens, J. Vanthienen, Benchmarking state-of-the-art classification algorithms for credit scoring, The Journal of the Operational Research Society 54 (6) (2003) 627–635.

[6] B. Baesens, S. Viaene, D.V. den Poel, J. Vanthienen, G. Dedene, Bayesian neural network learning for repeat purchase modelling in direct marketing, European Journal of Operational Research 138 (1) (2002) 191–211.

[7] C.B. Bhattacharya, When customers are members: customer retention in paid membership contexts, Journal of the Academy of Marketing Science 26 (1998) 31–44.

[8] J. Burez, D.V. den Poel, {CRM} at a pay-tv company: using analytical models to reduce customer attrition by targeted marketing for subscription services, Expert Systems with Applications 32 (2) (2007) 277–288.

[9] M. Colgate, K. Stewart, R. Kinsella, Customer defection: a study of the student market in Ireland, The International Journal of Bank Marketing 14 (3) (1996) 23–29.

[10] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, Cambridge University Press, New York, NY, USA, 2000.

[5] https://www.kaggle.com/c/kdd-cup-2014-predicting-excitement-at-donors-choose.

[11] K. Dasgupta, R. Singh, B. Viswanathan, D. Chakraborty, S. Mukherjea, A.A. Nanavati, A. Joshi, Social ties and their relevance to churn in mobile telecom networks, Proceedings of the 11th International Conference on Extending Database Technology: Advances in Database Technology. EDBT '08. ACM, New York, NY, USA, 2008, pp. 668–677.
[12] J.P. Egan, Signal Detection Theory and ROC Analysis. Series in Cognition and Perception, Academic Press, New York, NY, 1975.
[13] European Commission, Energy Markets in the European Union in 2011, November 2012.
[14] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, C.-J. Lin, LIBLINEAR: a library for large linear classification, Journal of Machine Learning Research 9 (2008) 1871–1874.
[15] T. Fawcett, Roc graphs: notes and practical considerations for researchers, Machine Learning 31 (2004) 1–38.
[16] J. Ganesh, M.J. Arnold, K.E. Reynolds, Understanding the customer base of service providers: an examination of the differences between switchers and stayers, The Journal of Marketing (2000) 65–87.
[17] D.J. Hand, W.E. Henley, Statistical classification methods in consumer credit scoring: a review, Journal of the Royal Statistical Society: Series A (Statistics in Society) 160 (3) (1997) 523–541.
[18] W.E. Henley, D.J. Hand, A *k*-nearest-neighbour classifier for assessing consumer credit risk, Journal of the Royal Statistical Society: Series A (Statistics in Society) 45 (1) (1996) 77–95.
[19] S. Hettich, S.D. Bay, The UCI KDD Archive, http://kdd.ics.uci.edu1996.
[20] S. Hill, F. Provost, C. Volinsky, Network-based marketing: identifying likely adopters via consumer networks, Statistical Science 21 (2) (05 2006) 256–276.
[21] E. Junqué de Fortuny, D. Martens, F. Provost, Predictive modeling with big data: is bigger really better? Big Data 1 (4) (2013) 215–226.
[22] S.M. Keaveney, M. Parthasarathy, Customer switching behavior in online services: an exploratory study of the role of selected attitudinal, behavioral, and demographic factors, Journal of the Academy of Marketing Science 29 (4) (2001) 374–390.
[23] J.Y. Lee, D.R. Bell, Neighborhood social capital and social learning for experience attributes of products, Marketing Science 32 (6) (2013) 960–976.
[24] S.A. Macskassy, F. Provost, Classification in networked data: a toolkit and a univariate case study, Journal of Machine Learning Research 8 (May 2007) 935–983.
[25] J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, A.H. Byers, Big Data: The Next Frontier for Innovation, Competition, and Productivity, May 2011.
[26] D. Martens, B. Baesens, T. Van Gestel, J. Vanthienen, Comprehensible credit scoring models using rule extraction from support vector machines, European Journal of Operational Research 183 (3) (2007) 1466–1476.
[27] D. Martens, E. Junqué de Fortuny, M. Stankova, Data mining for fraud detection using invoicing data. A case study in fiscal residence fraud, Working Papers 2013026, University of Antwerp, Faculty of Applied Economics, Oct. 2013.
[28] D. Martens, F. Provost, Pseudo-social network targeting from consumer transaction data, Tech. Rep., NYU Working Paper CEDER-11-05, 2011.
[29] M. Mizuno, A. Saji, U. Sumita, H. Suzuki, Optimal threshold analysis of segmentation methods for identifying target customers, European Journal of Operational Research 186 (1) (2008) 358–379.
[30] M. Naldi, A. Pacifici, Optimal sequence of free traffic offers in mixed fee-consumption pricing packages, Decision Support Systems 50 (1) (2010) 281–291.
[31] C. Perlich, F. Provost, Distribution-based aggregation for relational learning with identifier attributes, Machine Learning 62 (1–2) (2006) 65–105.
[32] C. Perlich, F. Provost, J.S. Simonoff, Tree induction vs. logistic regression: a learning-curve analysis, Journal of Machine Learning Research 4 (Dec. 2003) 211–255.
[33] F. Provost, T. Fawcett, Robust classification for imprecise environments, Machine Learning 42 (3) (2001) 203–231.
[34] F. Provost, T. Fawcett, Data Science for Business, O'Reilly, 2013.
[35] F. Provost, X. Zhang, B. Dalessandro, A. Murray, R. Hook, Audience selection for online brand advertising: privacy-friendly social network targeting, In KDD 09: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2009, pp. 707–716.
[36] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.
[37] Research, Markets, Electricity in Belgium, October 2012.
[38] Y. Richter, E. Yom-Tov, N. Slonim, Predicting Customer Churn in Mobile Networks Through Analysis of Social Groups, 2010. 732–741 (Ch. 63).
[39] S. Rosset, E. Neumann, U. Eick, N. Vatnik, Customer lifetime value models for decision support, Data Mining and Knowledge Discovery 7 (3) (2003) 321–339.
[40] M. Stankova, D. Martens, F. Provost, Classification over Bipartite Graphs Through Projection, 2014.
[41] T.J. Steenburgh, A. Ainslie, P.H. Engebretson, Massively categorical variables: revealing the information in ZIP codes, Marketing Science 22 (1) (2003) 40–57.
[42] T. Strandvik, V. Liljander, The nature of customer relationships in services, Advances in Services Marketing and Management 4 (1995) 141–167.
[43] L. Thomas, Consumer Credit Models: Pricing, Profit and Portfolios: Pricing, Profit and Portfolios, OUP Oxford, 2009.
[44] L.C. Thomas, D.B. Edelman, J.N. Crook, Credit Scoring and Its Applications, Siam, 2002.
[45] D. Van den Poel, B. Larivière, Customer attrition analysis for financial services using proportional hazard models, European Journal of Operational Research 157 (1) (2004) 196–217.
[46] T. Van Gestel, B. Baesens, D. Martens, Predictive Analytics: Techniques and Applications in Credit Risk Modelling, OUP Oxford, 2015. (forthcoming).
[47] J. Van Gool, W. Verbeke, P. Sercu, B. Baesens, Credit scoring for microfinance: is it worth it? International Journal of Finance and Economics 17 (2) (2012) 103–123.
[48] V.N. Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag New York, Inc., New York, NY, USA, 1995.
[49] W. Verbeke, K. Dejaeger, D. Martens, J. Hur, B. Baesens, New insights into churn prediction in the telecommunication sector: a profit driven data mining approach, European Journal of Operational Research 218 (1) (2012) 211–229.
[50] W. Verbeke, D. Martens, B. Baesens, Social network analysis for customer churn prediction, Applied Soft Computing 14, Part C (0) (2014) 431–446.
[51] W. Verbeke, D. Martens, C. Mues, B. Baesens, Building comprehensible customer churn prediction models with advanced rule induction techniques, Expert Systems with Applications 38 (3) (2011) 2354–2364.
[52] T. Verbraken, W. Verbeke, B. Baesens, A novel profit maximizing metric for measuring classification performance of customer churn prediction models, IEEE Transactions on Knowledge and Data Engineering 25 (5) (May 2013) 961–973.
[53] VREG, Market Monitor 2010, 2010.
[54] E. Zdravevski, P. Lameski, A. Kulakov, Weight of evidence as a tool for attribute transformation in the preprocessing stage of supervised learning algorithms, The 2011 International Joint Conference on Neural Networks (IJCNN), 2011, pp. 181–188.

Julie Moeyersoms graduated in 2012 as business engineer at the Faculty of Applied Economics at the University of Antwerp (Belgium). She is currently working as a doctoral researcher at the Department of Engineering Management at the University of Antwerp. Her research is mainly focused on social network analysis and customer analytics as well as the issue of comprehensibility in a data mining context.

David Martens is an assistant professor at the University of Antwerp, where he heads the Applied Data Mining research group. His research focuses on the development and application of data mining techniques that lead to improved understanding of human behavior, and the use thereof in marketing and finance. His work has been published in high impact journals, such as MIS Quarterly, Machine Learning, Journal of Machine Learning Research, IEEE Transactions on Neural Networks, IEEE Transactions on Knowledge and Data Engineering and IEEE Transactions on Evolutionary Computation. In 2014, David won the “Best EJOR Application Award” (European Journal of Operational Research) for his work on churn prediction. In 2008 David was finalist of the prestigious international KDD doctoral dissertation award. Together with Prof. Foster Provost, he is an inventor of four pending patent applications on data mining methods.