Improving Classifier Performance by Knowledge-Driven Data Preparation

Laura Welcker[1], Stephan Koch[2], and Frank Dellmann[1]

[1] Münster University of Applied Sciences, Münster, Germany
{laurawelcker,dellmann}@fh-muenster.de
[2] BBDO Proximity GmbH, Hamburg, Germany
Stephan.Koch@bbdoproximity.de

Abstract. Classification is a widely used technique in data mining. Thereby achieving a reasonable classifier performance is an increasingly important goal. This paper aims to empirically show how classifier performance can be improved by knowledge-driven data preparation using business, data and methodological know-how. To point out the variety of knowledge-driven approaches, we firstly introduce an advanced framework that breaks down the data preparation phase to four hierarchy levels within the CRISP-DM process model. The first 3 levels reflect methodological knowledge; the last level clarifies the use of business and data know-how. Furthermore, we present insights from a case study to show the effect of variable derivation as a subtask of data preparation. The impact of 9 derivation approaches and 4 combinations of them on classifier performance is assessed on a real world dataset using decision trees and gains charts as performance measure. The results indicate that our approach improves the classifier performance.

Keywords: classification, framework for data preparation, knowledge-driven data preparation, decision trees.

1 Introduction

Classification is one of the most important and widely used data mining techniques, especially in the area of marketing and Customer Relationship Management (CRM) [1,2]. Due to the widespread use of classification applications in today's highly competitive sectors, continuous improvement of its predictive power has gained importance. Approaches for improvement of classifier performance can be categorized according to the data mining process and its distinct steps. Common process models describing the data mining process are the KDD (Knowledge Discovery in Databases) process model [3], SEMMA [4], Reinartz's framework [5] and CRISP-DM [6]. All process models include a step dealing with data preparation. This step is often referred to as the most time consuming but also the most important part of the data mining process [7,8]. To outline the importance we clarify the risks that occur when accurate data preparation is lacking. First, the data might not meet the requirements of the

P. Perner (Ed.): ICDM 2012, LNAI 7377, pp. 151–165, 2012.

algorithms in use. Second, poor or no data preparation is likely to lead to an incomplete and inaccurate data representation space, which is spanned by variables and realizations used in the modeling step. Both risks may dramatically affect the classifier performance and can lead to poor prediction accuracy or even in wrong predictive models. From the authors' experience, the potential of improved data preparation often remains unused by practitioners. They fail to transform their understanding of business and data into a properly prepared representation space. Reasons for this could be a lack of time and/or methodological know-how. That is why many researchers and especially software vendors try to enforce the automation of data preparation that guarantees an ease of use and time savings. However, automation is where knowledge-driven data preparation becomes impossible because the selection and evaluation of preparation instruments in automated data preparation strongly relies on mathematical and statistical methods, instead of relying on business, data and methodological understanding. We support the assumption that the inclusion of knowledge-driven data preparation has a positive impact on classifier performance contrasted to the exclusion of it.

Many authors state that the role of the human within the data mining process is essential [9,10,11]. This approach is often described as domain knowledge. "Domain knowledge consists of information about the data that is already available either through some other discovery process or from a domain expert" [9, p. 37]. Furthermore, [9] claim that domain knowledge can affect the discovery process within the data mining system in two ways. First, it can make patterns more visible by generalizing the variable realizations, and second, it can constrain the representation space as well as the rule space. [10] analyses the question how domain knowledge can be used to evaluate and interpret classification models. The study of [11] concentrates on the use of domain knowledge in all phases of the data mining process. In this paper the knowledge-driven approach can be differentiated from domain knowledge as it is defined as business, data and methodological know-how. Moreover, the knowledge-driven approach focuses on its integration only within the data preparation phase. A study, which deals with the same topic as this paper comes from [12]. The authors compare the performance of classification models with and without domain knowledge. But they express their domain knowledge only by one categorical variable derived from an expert. In this study, we show the multitude and power of knowledge-driven approaches by applying more than one derivation approach on a large number of variables.

The paper's objective is to show that the use of knowledge-driven data preparation leads to higher classifier performance in comparison to standard preparation (see section 3). To reach this objective, the paper reports on a case study applying knowledge-driven data preparation. In this study a classifier is built on a standard dataset, which was extended through knowledge-driven derivation approaches. The resulting classifier is compared to a reference classifier, which was built only on the standard dataset (without the extension). The findings and results gained from the study are major contributions of this paper.

Furthermore, a variety of data preparation tasks are structured within a framework in order to present a compilation of methods as well as a comprehensive and procedural guideline. Most of the preparation tasks has already been mentioned in literature [8,13,14,15], but a framework, which lists and structures all of the identified approaches, does not exist to the authors' knowledge. Therefore, the proposed framework for knowledge-driven data preparation can be considered as a further contribution of this paper.

The paper is divided into four sections. After the introduction, section 2 introduces the advanced framework based on data preparation and describes the step of variable derivation in further detail, since it is the basic concept of the study. Section 3 reports on the case study by describing the experimental setup and the results. Section 4 provides a conclusion by assessing the results and showing further research fields as well as limitations.

2 Introducing an Advanced Framework Focused on Data Preparation

2.1 Advanced Framework for Data Preparation

The advanced framework for data preparation is based on the first two levels of the CRISP-DM methodology. We focus on CRISP-DM, since it is considered as the most complete [16,17] and broadly adopted data mining process model [7]. It provides a systematic overview of the life cycle of a data mining project and consists of six major phases. Even if the original aim of CRISP-DM was already to make data mining projects such as classification projects "more efficient, better organized, more reproducible, more manageable and more likely to yield business success" [18], the necessity for more specific and detailed framework was also proclaimed by [6, p. 11] in the context of "mapping for the future". However, this update of the CRISP-DM methodology (named CRISP-DM 2.0) was only initialized [18], but has not revealed any official results to this date.

Referring to the idea of CRISP-DM 2.0 we designed our advanced framework focused on data preparation because of its high impact on the quality of the classifier performance. This impact has been experienced by us in numerous practical projects and is also stated in literature [19]. In computer science for instance, the impact of low data quality on the output quality has been discussed under the "Garbage in garbage out"-principle [20,21].

To implement the advanced framework we broke down the second level of the CRISP-DM process model, the generic tasks into three additional levels: subtasks, steps, and realization options (see Figure 1). With these additional levels the user gets a deeper insight into the data preparation opportunities and can more easily decide which approach would reasonably improve the individual classification issue. The five generic tasks for data preparation represents an adequate baseline for our framework because of its completeness and discriminatory power, which is beneficial to the practitioners. The five generic tasks are distinct from each other. They are sorted in chronological order as the user will

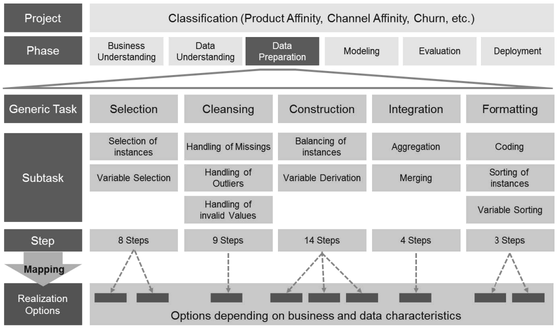


Fig. 1. Advanced Framework for Data Preparation [Source: Authors' own construct]

need them in a real project. Nevertheless, having feedback loops between the generic tasks is always required. Each generic task is divided into a complete set of subtasks as detailed below.

1. **Selection:** The generic task selection includes all subtasks to select valid and relevant instances and variables. In terms of instances, the selection contains among others a form of complexity reduction by the step of splitting (in case of very heterogeneous instances, e.g. private and business customers). Selection of variables in turn comprises for example the steps of time reference and elimination of multicollinearity.
2. **Cleansing:** This generic task contains all steps to replace, keep, bin and, if necessary, eliminate conspicuous data such as outliers, missing values and invalid instance values.
3. **Construction:** Construction is divided into two subtasks: balancing of instances and derivation of new variables. Balancing of instances can be performed by oversampling and undersampling. Derivation of new variables is categorized into univariate (e.g. normalization), multivariate (e.g. factor analysis) and hybrid approaches (e.g. segmentation with classified variables). Further information about the variable derivation gives subsection 2.2.
4. **Integration:** Integration deals with the aggregation (multiple rows to single row) and disaggregation of instances (single row to multiple rows) as well as merging of datasets.
5. **Formatting:** The last generic task contains the steps to adjust value coding and to sort instances and variables in accordance with the software and algorithms requirements.

The subtasks consist of different steps, which are only counted and not further described within this paper. Up to the level of steps the framework has a general validity. At level 4, a specialized mapping has to be conducted because the realization options depend on business and data characteristics and cannot be defined on a generic basis.

We have to consider that our framework has certain limitations. Although we derived numerous realization options for all steps, our framework cannot claim to be exhaustive. Applying this framework the practitioner only has to decide which subtasks, steps and realization options are relevant in the specific context. Compared to existing process models this deeper hierarchical structure facilitates the analyst's job. He can more easily transform his knowledge about business, data and methods into a relevant dataset.

2.2 Variable Derivation

In this paper we examine in depth the step of deriving new variables because we are of the opinion that a knowledge-driven variation of the representation space has a greater potential to improve classifier performance than e.g. the reduction of it, caused by selection or cleansing. The relevance of the other generic tasks is due to further research. Moreover, the subtask of variable derivation is closely linked to the business and data understanding of the user. [14] state that the most effective derived variables are those, that express additional information (beyond the database), such as a description of some underlying customer behavior. For the creation of useful derivations it is important to use background or domain knowledge and not to randomly combine a large number of variables. That is the reason why automated tools are not well-suited to produce valuable results by creating derived variables.

In the area of Machine Learning, derivation of new variables is referred to constructive induction, which was introduced by [22]. Subsequently, [23] presented three different constructive induction strategies:

- **Hypothesis-driven:** Changes are based on hypothesis, generated in each data analysis-iteration and the discovery of patterns.
- **Data-driven:** Data characteristics are used to generate new data representation spaces.
- **Knowledge-driven:** Expert domain knowledge is applied.

The combination of two or more of these strategies is denominated as multi-strategy constructive induction [23]. Our approach of variable derivation can be classified as multi-strategy constructive induction as we combine data-driven and knowledge-driven strategies.

A considerable amount of literature has been published on variable derivation [24,25,26]. In order to develop a complete framework we collected various approaches and put them into a separate framework shown in Figure 2. The subtask of variable derivation consists of eleven distinct process steps. Corresponding to the number of variables involved in a step, they are categorized in univariate and multivariate approaches. Approaches combining multiple options are categorized as hybrid approaches. All approaches are broken down into various realization options, which are represented on the last level of our framework.

- **Univariate Approaches:** The first step called *development* includes all derivations representing the development of a variable in time. Ratios as well as absolute values could serve this purpose. On the next level of our framework we split these options up by scale and operator. *Normalization* is

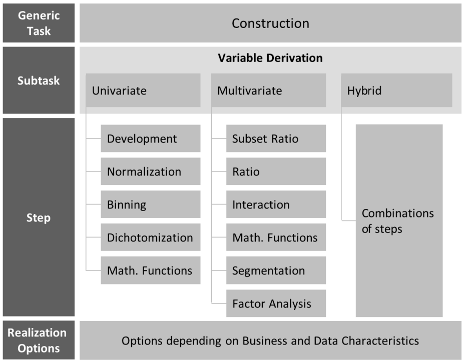


Fig. 2. Breakdown of the subtask variable derivation [Source: Authors' own construct]

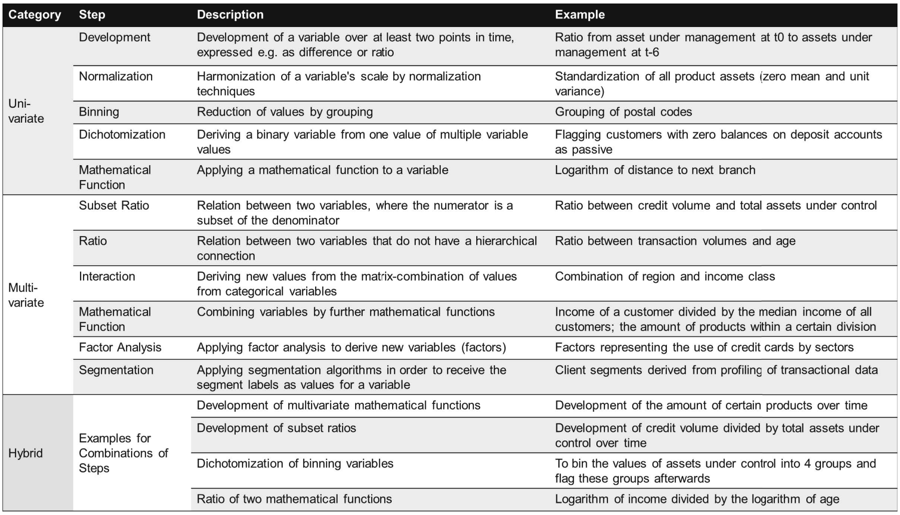
the next step and deals with all kinds of scale harmonization. The most commonly used form of normalization is the input standardized to zero mean and unit variance. In our framework the *binning* step comprises all groupings of values irrespective of the variable's scale. Realization options break it down to options for each scale. The step of *dichotomization* might be essential in data mining projects even without knowledge-driven acting. Due to algorithms' requirements, dichotomization is often used to transform categorical variables into binary variables, where each derived variable represents one value of the former variable. From the business point of view dichotomization can be used to emphasize important variables and values. Derivations realized by applying a *mathematical function* to an input variable are consolidated in the last step of univariate approaches. Logarithmic transformation is a well known realization option within this step.

- **Multivariate Approaches:** *Subset ratios* are calculated within the first multivariate step in order to reflect the structure of a given entity subset. Consequently, the numerator is always a subset of the denominator. Ratios between hierarchy-independent variables are separately categorized in the step named *ratio*. Whereas the above mentioned multivariate approaches focus on the metric scale, *interactions* allow the practitioner to combine categorical variables. New values are derived from the matrix of the original value pairs. Combinations of two or more variables by means of other *mathematical functions* are summarized in a further step. *Segmentation* and *factor analysis* are quite complex approaches to derive new variables as they represent small data mining projects on their own. However, these two steps can be useful especially when dealing with huge numbers of variables or observing multicollinearity. Depending on the input data there are different realization options for both analyses. The study of [27] shows that a knowledge-driven derivation of segmentation variables can outperform a random selection of variables with regard to performance gain.
- **Hybrid Approaches:** In the category of hybrid approaches, deriving new variables by using more than one of the steps mentioned above is considered

as another powerful way to improve predictive accuracy by widening the representation space. This last step enables the practitioner to transfer more complex business know-how into the data.

Finally, Table 1 summarizes all steps by giving a short description and an example. The examples mostly refer to the financial sector because they are adopted from our case study, which is described in the next section.

Table 1. Derivation Approaches [Source: Authors' own construct]



Category	Step	Description	Example
Uni-variate	Development	Development of a variable over at least two points in time, expressed e.g. as difference or ratio	Ratio from asset under management at t0 to assets under management at t-6
	Normalization	Harmonization of a variable's scale by normalization techniques	Standardization of all product assets (zero mean and unit variance)
	Binning	Reduction of values by grouping	Grouping of postal codes
	Dichotomization	Deriving a binary variable from one value of multiple variable values	Flagging customers with zero balances on deposit accounts as passive
	Mathematical Function	Applying a mathematical function to a variable	Logarithm of distance to next branch
Multi-variate	Subset Ratio	Relation between two variables, where the numerator is a subset of the denominator	Ratio between credit volume and total assets under control
	Ratio	Relation between two variables that do not have a hierarchical connection	Ratio between transaction volumes and age
	Interaction	Deriving new values from the matrix-combination of values from categorical variables	Combination of region and income class
	Mathematical Function	Combining variables by further mathematical functions	Income of a customer divided by the median income of all customers; the amount of products within a certain division
	Factor Analysis	Applying factor analysis to derive new variables (factors)	Factors representing the use of credit cards by sectors
	Segmentation	Applying segmentation algorithms in order to receive the segment labels as values for a variable	Client segments derived from profiling of transactional data
Hybrid	Examples for Combinations of Steps	Development of multivariate mathematical functions	Development of the amount of certain products over time
		Development of subset ratios	Development of credit volume divided by total assets under control over time
		Dichotomization of binning variables	To bin the values of assets under control into 4 groups and flag these groups afterwards
		Ratio of two mathematical functions	Logarithm of income divided by the logarithm of age

3 Case Study on Variable Derivation

After presenting our advanced framework for data preparation in general and the subtask of variable derivation in more detail, this section deals with the practical application of these approaches.

3.1 Experimental Setup

Research Domain and Data. The conducted study was realized with customer data from the German financial retail sector. The raw dataset comprised about 1.7 million instances and 541 potential predictor variables. The underlying direct marketing issue was defined as building an affinity score for closed funds in order to support the selection of relevant target groups for direct mail. In this context buying a closed fund as a reaction of a direct mailing was designed as binary target variable with "1" for customer reaction and "0" for non-reaction. In order to keep data free of any causal reference between product sale and predictor values we set up a time-dependent selection of potential predictor variables based on the individual acquisition date of each customer.

Research Design. Figure 3 displays the test setting, which was defined to evaluate the impact of knowledge-driven derivation on classifier performance. The methodology is basically marked by a comparison of two classifiers that only differ in the composition of their input data. Based on the raw dataset (541 variables), measures of standard data preparation are conducted. Consequently, the raw dataset gets reduced by x variables. The dataset created by this means (541 - x variables) is the basis for the reference model, the REF classifier. For the other classifier the dataset is extended through knowledge-driven derivation by y variables, so that the classifier ALL has 541 - x + y potential input variables. The two classifiers are both built the same way by the same classification technique with the same parameters. The only difference between the two are the y derived variables, so that every difference in performance can clearly be assigned to the impact of variable derivation.

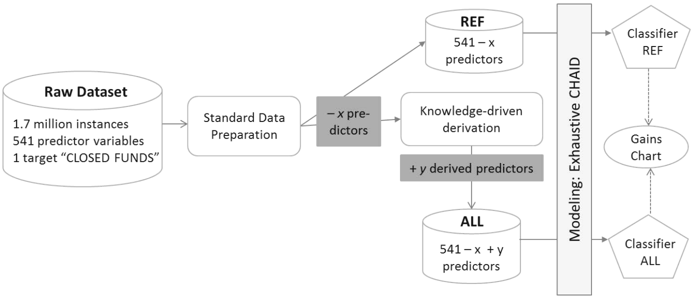


Fig. 3. Test Setting [Source: Authors' own construct]

Classification Technique. In this study, we used PASW Modeler 14 as data mining software to run the classification. Classification models within this project were created by a decision tree algorithm. According to [1,2] decision trees and regression analysis are the most commonly used data mining techniques. Furthermore, research results do not show a systematical outperformance of other techniques [28,29]. Due to their ease of use and the ability of some tree algorithm to handle missing values as separate categories, we preferred decision trees to regression analysis. In many cases the separate handling represents best the real situation. Missing values for instance were created when a development of the variable "product volume" could not be calculated due to a very recent acquisition. For using decision trees the applied software provides four different algorithms: CHAID, C5.0, QUEST and CART. Due to data characteristics we preferred tree algorithms that theoretically allow multiway splits like CHAID and C5.0. We expected them to perform better on the given data than CART or QUEST with regard to quality and clearness. The CHAID algorithm was finally chosen for this project because it tends to produce shallower trees than C5.0 with a higher level of multiway splits.

The CHAID algorithm is originally proposed by [30] and its application is popular in marketing. It uses an attribute selection measure that is based on the statistical Chi-squared test for independence [15]. The software also offers "Exhaustive CHAID", which is a refinement of CHAID proposed by [31]. Both algorithms CHAID and Exhaustive CHAID allow multiple splits of a node rather than binary or c-way splits (where c is the original number of categories of the predictor variable). The difference lies in the merging step, where Exhaustive CHAID uses an exhaustive search procedure to merge any similar pair until only a single pair remains [31].

The Exhaustive CHAID algorithm was used with Bonferroni correction to build trees of five levels. Chi-square was calculated based on Pearson and the split of merged nodes was permitted. Further parameters were kept as default in PASW Modeler 14.

Performance Measure. The performance measure in this project was a gains chart. The gains chart plots the "Gains (%)" on the y-axis and percentiles on the x-axis. Gains are defined as the proportion of hits in each increment relative to the total number of hits in the classifier. Gains charts effectively illustrate how widely the practitioner needs to cast the net to capture a given percentage of all of the hits in the tree [14,17]. This performance measure has been chosen because it is really suitable to compare two classifiers in a marketing context. Moreover, gains charts can handle imbalance class problems better than accuracy or classification error [14].

Standard Data Preparation. In course of the standard data preparation the raw dataset has been divided into a training (80%) and a test (20%) set. In consequence 345,417 instances were kept as test partition. Due to a skewed distribution of the target classes (only 0.12% for the 1-class), a balancing of instances was necessary. Several ratios of downsizing were tested to examine the best weighting of positive instances (1-class) in the training set. A ratio of 10% of positive instances clearly outperformed the other tested ratios (5% and 15%) and was subsequently applied. The balancing led to a distribution of 15,147 negative instances and 1,683 positive instances in the training set.

As a first step the given raw dataset was prepared by the adjustment of scale levels and by filtering variables. Variables were filtered out if they meet one of the following filter criteria:

1. Quality criteria: Dominance of missings or single values/categories, multicollinearity etc.
2. Target-specific criteria: Contextual irrelevance for the target (e.g. the name of the customer)

The raw dataset comprises with 541 variables all available company information, but the classification task is only focused on closed funds. Therefore, most of the variables were filtered out due to contextual irrelevance for the classification problem. To decide if a variable is relevant or irrelevant for the underlying task the importance of domain knowledge and the integration of a domain expert into the selection process were again emphasized. No variable had to be excluded due to dominating the tree construction in the first branches. In course of the standard data preparation 270 predictor variables were filtered out (with reference to Figure 3 is x = 270).

Variable Derivation. In course of knowledge-driven derivation, 50 variables were directly transformed by normalization and 201 derived variables were added to the dataset (thus named ALL) by knowledge-driven derivation (with reference to Figure 3 is y = 201). Table 2 lists all approaches that were conducted within this study. We only accepted derivations with contextual relevance. Due to restrictions in time and data limitations the implementation of further derivation approaches, such as factor analysis and segmentation was excluded from this study. Within the step *development* 41 new variables were derived as trends over one, three and six months. Changes on longer terms could not be calculated due to structural changes in the database. Besides, we did not expect long term changes to have extra predictive power based on our business experience. The calculated trends included assets under management, account balances, transaction volumes, ratings and other variables. *Normalization* was conducted for 50 metric variables. The standardized variables were the only derivations, which have directly replaced the original variables as we did not want to emphasize their information disproportionally. *Binning* was used to make the information of variables with many categories more accessible to the algorithm. We grouped for example postal codes in order to present this regional information on a higher level. As passive customers often show very special behaviors, we flagged those with zero balances on deposit accounts within the *dichotomization* step. The univariate application of a *mathematical function* was only used for one variable (distance to next branch).

The multivariate approach of *subset ratios* led to 33 new variables. Some of them were built to represent the overall asset structure, for example ratio of credit volume to total assets under control, ratio of deposit volume to total assets under control, and ratio of daily allowance to total assets under control. Further variables were calculated to reproduce the structure of a certain product, for example the customer account: Ratios of investments by region, ratios of investments by paper types, and ratio of investments by investment sector. The

Table 2. Applied Derivation Approaches [Source: Authors' own construct]

Univariate Steps	Nb. of derived variables	Multivariate Steps	Nb. of derived variables	Hybrid Steps	Nb. of derived variables
Development	41	Subset Ratio	33	Development & MMF	21
Normalization	50	Ratio	17	Development & Subset Ratio	46
Binning	4	Interaction	7	Dichotomization & Binning	8
Dichotomization	1	Mathematical Functions (MMF)	15	Ratio & UMF	7
Mathematical Functions (UMF)	1				
Total	**97**	**Total**	**72**	**Total**	**82**

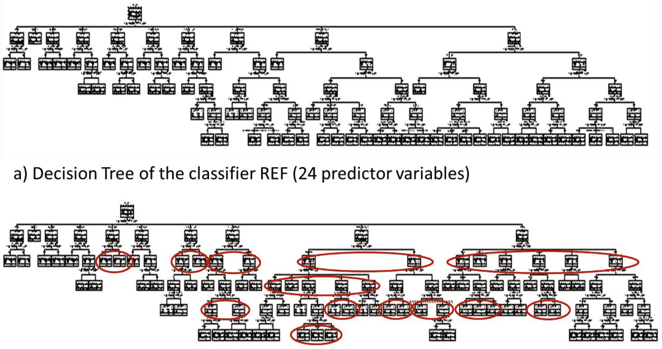
step *ratio* resulted in 17 variables. We used financial, ratio-scaled variables to derive new variables, for example ratio of transaction volumes to assets, ratio of transaction volumes to age, and ratio of electronic and written transfers. Seven variables were derived as *interactions*. Interactions included among others, combinations of region and income classes or age classes and profession. Within the step of *multivariate mathematical functions* we derived 15 new variables, e.g. the amount of products within a division. Another application was the calculation of the ratio between customers' assets and the median of assets over all customers. In this way, we could identify customers with special behavior.

Finally, we derived 82 *hybrid* variables as combination of steps. Most of the hybrid variables are developments of subset ratios (46 variables) or multivariate mathematical functions (21 variables) to illustrate structural changes in the customers' assets. These variations show especially well the changes in the customers' life and needs. Eight metric variables were firstly put into groups and got flagged afterwards. This combination of binning and dichotomization can be promising if only a subset of the original values is relevant for the target classification. The ratio of two mathematical functions was applied for 7 variables. In that context, the logarithmic function was used to make two different dimensions comparable. The logarithm of income divided by the logarithm of age for instance is a good predictor when it comes to target group characterization as it combines two important information about the customers. As the combination of steps is the most complex way to reveal underlying customer behavior, we expect these hybrid variables to be the most powerful in improving classifier performance.

3.2 Results

Finally, 11 derived predictor variables were integrated within the decision tree. These variables can assigned to the following approaches: 7 univariate or hybrid developments, 2 MMF, 1 interaction and 1 ratio & MMF. Figure 4 displays both, the decision tree of the REF and the ALL classifier. The circles in b) show the nodes, where a derived variable is responsible for the split. Four derived variables are making the second split, which indicates a great importance for the overall model. These variables are: changes in the amount of products over 1 and 3 months, changes of bonds in the deposit in the last month and trend of the amount of debit transactions within the last 6 months.

The comparison of gain performance of the two classifiers is displayed by a gains chart (see Figure 5). The reference model (REF) is represented by the lower curve; the test model (ALL) refers to the upper curve. The diagonal line plots the expected response for the entire sample if the classifier is not used. In reference to Figure 5 it can be stated, that the use of derivation methods improves classifier performance as the test model clearly outperforms the reference model for the first decile. The steeper the curve, the higher the gain. The curved line indicates how much one can improve the response by including only those customer who rank in the higher percentiles based on gain. In this study, including the top 10% customers might net more than 95% of the positive responses for the classifier



a) Decision Tree of the classifier REF (24 predictor variables)

b) Decision Tree of the classifier ALL (29 predictor variables incl. 11 derived variables)

Fig. 4. Decision Trees for Test Set [Source: PASW Modeler 14]

ALL in contrast to about 90% for the classifier REF. This result reveals, that a direct mailing campaign based on the customer classification of the test model (ALL) reaches more "reactors", which means more potential "buyers" of closed funds than the classification of the standard model (REF). The results gained from the decision tree support the following statements:

- The effectiveness of each derivation step depends on sector and target selection. For our target variable closed funds, especially *univariate or hybrid developments* lead to very powerful predictors for classification. Most likely this is due to the fact that they express best the customers' asset structure and financial situation as the following insight outlines. The derived variable

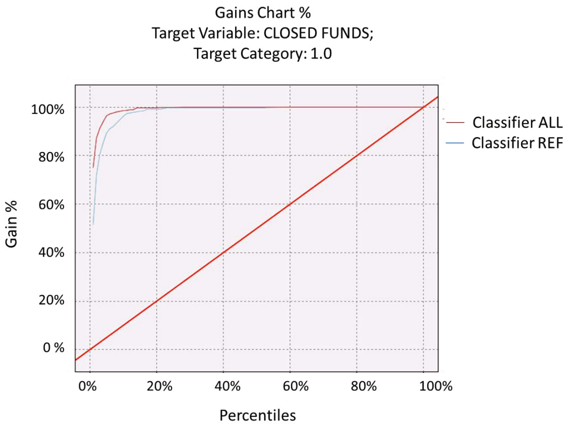


Fig. 5. Gains Chart for Test Set [Source: PASW Modeler 14]

"trend of the amount of debit transactions within the last 6 months" represents the most important variable for the classifier as it discriminates well between reactors and non-reactors.

- The amount of products bought by the customer and the change of this amount over 1 and 3 months are important predictors for the target variable. High values of these variables indicate a high buying activity, possibly due to customer's satisfaction.
- Young customers with relatively high income holding bonds are potential buyers of closed funds.
- In this study we applied 9 different derivation steps and 4 combinations of steps (see Table 2), from which finally 11 derived variables were integrated into the test model. This shows that applying various derivation approaches is a good way to find the best classifier. Nevertheless, one should bear in mind that too many variables may affect other algorithms more than it happened to the decision trees in our study.

4 Conclusion, Limitations and Future Directions

By setting up a detailed framework we intended to facilitate knowledge-driven data preparation in classification (and other data mining) projects. This hierarchical structured toolkit gives a good methodological overview and turned out to be a powerful guideline when mapping it with business and data know-how. A wide knowledge of statistical approaches becomes less important. However, the principal aim of this study was to empirically show how classifier performance can be enhanced by knowledge-driven data preparation. Therefore, approaches for variable derivation as a subtask of data preparation were conducted and tested by a specific research design. By comparing the gains curve of two classifiers (with and without derived variables), it can be stated that derivation of new variables clearly improves classifier performance. With regard to variable derivation knowledge-driven acting reduces the risk of creating a vast amount of variables, which potentially affect algorithm's efficiency and accuracy without providing added value of classifier performance. Automated derivation would increase this risk.

This study has also certain limitations. The presented framework can only perform as a guideline and needs to be specified by the user for individual application. With regard to the implementation of derivation methods, it has to be considered that only one dataset from a specific sector has been tested. Additional or other effects on another data structure are possible. Thus, these findings cannot be extrapolated to all datasets. Further limitations exist in terms of applied software and classification algorithm. Our results refer to an application with decision trees employed by PASW Modeler 14. The applied data preparation methods have possibly great potential using neural networks as well. However, more research on this topic needs to be undertaken to stabilize and specify the hypothesis that knowledge-driven data preparation is worthwhile.

Nevertheless, deriving new variables should always lead to great attention in variable selection. As the algorithm's speed and accuracy can suffer from a large amount of input variables, further research in the area of variable selection and its combination with variable derivation is necessary. Further research is needed to identify influential factors on the procedure as a first step and to test their influence on the classifier performance as a second step. Possible research areas could be the role of the target variable or the influence of the test design, i.e. do we get a different outcome by applying separate, all or only a specific combination of derivation approaches?

References

1. Rexer, K.: 5th Annual Data Miner Survey - 2011 Survey Summary Report. Rexer Analytics, Winchester (2011)
2. KDnuggets, Which methods/algorithms did you use for data analysis in 2011?, `http://www.kdnuggets.com/polls/2011/algorithms-analytics-data-mining.html`
3. Fayyad, U., Piatesky-Shapiro, G., Smyth, P., Uthurusamy, R. (eds.): Advances in Knowledge Discovery and Data Mining. AAAI Press, California (1996)
4. SAS: From Data to Business Advantage: Data Mining, SEMMA Methodology and the SAS System. White Paper, SAS Institute Inc. (1997)
5. Reinartz, T.: Focusing Solutions for Data Mining: Analytical Studies and Experimental Results in Real-World Domains. Springer, Heidelberg (1999)
6. Chapman, P., Clinton, J., Kerber, R., Khabaza, R.T., Reinartz, T., Shearer, C., Wirth, R.: CRISP-DM 1.0: step-by-step data mining guide. SPSS Inc. (2000)
7. Kurgan, L.A., Musilek, P.: A survey of knowledge discovery and data mining process models. The Knowledge Engineering Review 21(1), 1–24 (2006)
8. Refaat, M.: Data Preparation for Data Mining Using SAS. Morgan Kaufmann, San Francisco (2007)
9. Anand, S.S., Bell, D.A., Hughes, J.G.: The role of domain knowledge in data mining. In: 4th Int'l ACM Conference on Information and Knowledge Management, pp. 37–43. ACM, New York (1995)
10. de Oliveira Lima, E.: Domain Knowledge Integration in data mining for churn and customer lifetime value modelling: new approaches and applications. Dissertation, University of Southhampton (2009)
11. Kopanas, I., Avouris, N.M., Daskalaki, S.: The Role of Domain Knowledge in a Large Scale Data Mining Project. In: Vlahavas, I.P., Spyropoulos, C.D. (eds.) SETN 2002. LNCS (LNAI), vol. 2308, pp. 288–299. Springer, Heidelberg (2002)
12. Sinha, A.P., Zhao, H.: Incorporating domain knowledge into data mining classifiers: An application in indirect lending. Decision Support Systems 46, 287–299 (2008)
13. Pyle, D.: Business Modeling and Data Mining. Morgan Kaufmann Publishers, Amsterdam (2003)
14. Linoff, G.S., Berry, M.J.A.: Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management. Wiley Publishing, Indianapolis (2011)
15. Han, J., Kamber, M., Pei, J.: Data Mining, Concepts and Techniques. Morgan Kaufmann, Waltham (2012)
16. Azevedo, A., Santos, M.F.: KDD, SEMMA and CRISP-DM: A Parallel Overview. In: Proceedings of the IADIS European Conference Data Mining, pp. 182–185 (2008)
17. Nisbet, R., Elder, J.F., Miner, G.: Handbook of Statistical Analysis and Data Mining Applications. Academic Press, Elsevier, Amsterdam, Boston (2009)

18. CRISP-DM 2.0 Special Interest Group (SIG), `http://www.crisp-dm.org/new.htm`
19. Hernández, M.A., Stolfo, S.J.: Real-world data is dirty: Data cleansing and the merge/purge problem. Data Mining and Knowledge Discovery 2(1), 9–37 (1998)
20. Wang, R.Y., Strong, D.M.: Beyond Accuracy: What Data Quality Means to Data Consumers. Journal of Management Information Systems 12(4), 5–33 (1996)
21. Rahm, E., Do, H.H.: Data Cleaning: Problems and Current Approaches. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering 23(4), 3–26 (2000)
22. Michalski, R.S.: Pattern Recognition as Knowledge-Guided Computer Induction. Technical Report No. 927. Department of Computer Science, University of Illinois, Urbana-Champaign, IL (1978)
23. Wnek, J., Michalski, R.S.: Hypothesis-driven constructive induction in AQ17: A method and experiments. In: Proceedings of the International Joint Conference on Artificial Intelligence, Workshop on Evaluating and Changing Representations in Machine Learning, pp. 13–22 (1991)
24. Hammer, M., McLeod, D.: The semantic data model: a modelling mechanism for data base applications. In: Lowenthal, E.I., Nell, B.D. (eds.) Proceedings of the 1978 ACM SIGMOD International Conference on Management of Data, Austin, Texas, pp. 26–36 (1978)
25. Matheus, C.J., Rendell, L.A.: Constructive Induction on Decision Trees. In: Sridharan, N.S. (ed.) 11th International Joint Conference on Artificial Intelligence, pp. 645–650. Morgan Kaufmann (1989)
26. Zheng, Z.: Constructing New Attributes for Decision Tree Learning. Dissertation, Basser Department of Computer Science (1996)
27. Welcker, L.: Segmentierungsansätze zur Variablenreduktion im Rahmen der Optimierung von Scoring-Ergebnissen. Master Thesis, unpublished, Münster University of Applied Sciences (2010)
28. Michie, D., Spiegelhalter, D.J., Taylor, C.C.: Machine Learning, Neural and Statistical Classification. Englewood Cliffs, Ellis Horwood (1994)
29. Lim, T.-J., Loh, W.-Y., Shih, Y.-S.: A Comparison of Prediction Acuracy, Complexity, and Training Time of Thirty-three Old and New Classification Algorithms. Machine Learning 40, 203–229 (2000)
30. Kass, G.V.: An exploratory technique for investigating large quantities of categorical data. Journal of Applied Statistics 29(2), 119–127 (1980)
31. Biggs, D., de Ville, B., Suen, E.: A method of choosing multiway partitions for classification and decision trees. Journal of Applied Statistics 18(1), 49–62 (1991)