DATA MINING ON TRANSACTION DATA

Jürgen Cleve, Uwe Lämmel and Stefan Wissuwa
Wismar University, Germany

ABSTRACT

In a joint project with a German bank, Data Mining techniques are applied in order to analyse customer data. A set of data is given showing the monthly online transactions of customers. We try to answer the following questions: Is it possible to detect customers who most likely will cancel their contract? If so, additional marketing effort may avoid a cancellation. Are there any customers who do not use the online facilities efficiently? Again additional help can change their online banking behaviour, and both partners - the institution as well as the customer - can save money.

We describe a method of applying Data Mining and Neural Networks to classify customers as to their payment history. The goal is to determine a customer's specific behaviour and to create a computer model to automatically detect abnormal deviations that, for example, may indicate a customer turning to a competitor.

INTRODUCTION

In a co-operative work with a German bank we analyse data of customers' online transactions. The aim of the project can be summarized in two questions:

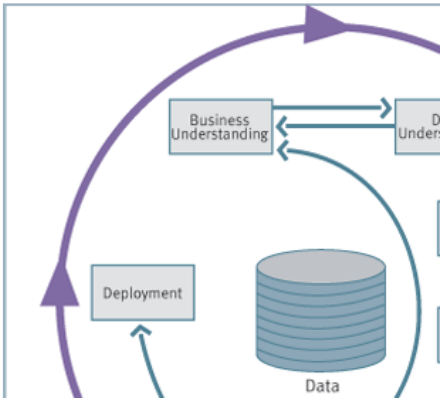
- Are there any customers who most likely will cancel their contracts within the near future?
- Some customers do not use the online facilities in a proper way provoking extra efforts and costs for both, customer and institute. Can these customers be identified?

We are going to answer the questions by applying Data Mining techniques. Our investigations are based on data describing the transaction history of customers per annum. The goal is to create a model of a customer's specific behaviour by clustering the available records and by investigating them. We expect each cluster to represent a specific kind of customer. Based on this 'customer profile' it will be possible to detect abnormal deviations in the customers' transactions. This information will help our partners to identify customers that may need specific attention.

Data Mining in general is an incremental process which includes the following phases as described by the CRISP-DM Process Model (Cross Industry Standard Process for Data Mining [6]):

1. Business Understanding: defining objectives and requirements
2. Data Understanding: data collection; checking quality and consistency
3. Data Preparation: preparing the data for the Data Mining algorithms / tools (selection, normalization etc.)
4. Modelling: applying various Data Mining techniques to the data; stepping back to 3. may be necessary due to requirements of some algorithms

Figure 1: CRISP-DM Process Model



5. Evaluation: evaluation of the model to be certain that it meets the requirements; if not, stepping back to 1. may be necessary
6. Deployment

Our work currently focuses on phase 3 to 4. A more detailed description of Data Mining and algorithms can be found in [2] [6] .

TRANSACTION DATA

The data are provided by our partner. A single record of data contains the customer id, account number, information about the type of transaction, followed by twelve values showing the monthly amount of transactions. Due to data protection regulations we can not handle other information like e.g. owner of the account, size of the enterprise the account belongs to etc. All accounts belong to business customers. The size of

Table 1: Transaction Data

knt_idx	pro	ks_idx	knd_idx	01/01	02/01	03/01	04/01	05/01	06/01	…	12/01
9848	10	6	5034	10,000,000	1,940,500	711,300	320,000	9,853,000	2,337,300	…	1,262,100
1333582	112	6	15651	147,993	129,593	147,993	157,251	482,720	205,302	…	0

the enterprises varies from small enterprises to big companies. Therefore the absolute value of a monthly transaction is less important than the sequence of the figures within a year.

We have a total of 406,000 records available over a six-year period. Approximately 75% are distributed equally over the last three years. Since purposes to have an account may vary very much, we have to address the problem that quite a lot of accounts are not used regularly month by month. From the Data Mining point of view it means that there are a lot of zero values in the data set.

DATA PREPARATION

It is well known that the success of Data Mining techniques depends highly on an appropriate pre-processing of the given data. Pre-processing includes data selection, data normalization and transformation.

Selection

Data selection is critical for the result of a Data Mining process. Although a relationship between a certain attribute and the desired result is not obvious, the attribute has to be considered as well. Some information may be hidden in the data. Thus on the one hand all available attributes should be processed; on the other hand a reduction of input attributes can reduce the complexity of the problem enormously. In a first attempt we only left out attributes which identify accounts or customers by numbers. Such numbers are different for each customer or account respectively and therefore are a handicap for a clustering algorithm. Clustering algorithms "look" for similarities in data and group data together according to the similarities.

Normalization

Normalization is a quite obvious transformation of data in order to meet the input requirements of various Data Mining algorithms. Artificial Neural Networks should be fed with input values in the range of [0,1] or [-1,1]. Other algorithms may only handle discrete or continuous values.

Table 2: Normalized Data (between 0 and 1)

01/01	02/01	03/01	04/01	05/01	06/01	07/01	…	12/01
0.7194	0.6256	0.5440	0.8420	0.1080	0.8770	0.2043	…	0.8041
0.5458	0.5518	0.4760	0.5308	0.5401	0.5345	0.0000	…	0.4585

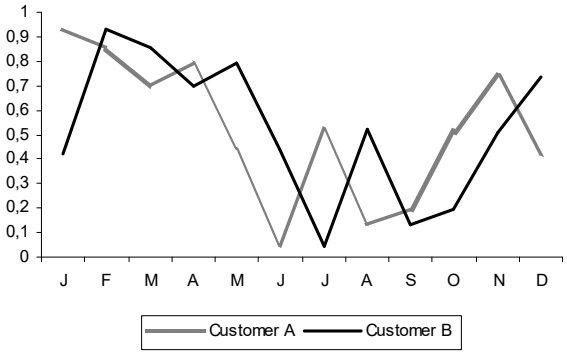
Table 2 gives an impression of the data after selection and normalization.

Transformation

Financial transactions over some years follow a cyclic pattern. In preparation of an annual closing of an account there are often more transactions than at the beginning of a year. Other peaks within a financial year are often based on business actions which repeat themselves frequently.

A problem here is that for example two different customers show the same transaction curve shifted by some months due their individual financial years (Figure 2). Usually, Data Mining algorithms can not deal with those kinds of distorted data, they will classify both curves as being different. The Fourier Transformation is quite a promising technique to prepare such data.

Figure 2: Time-dependent data



CLUSTERING THE DATA

By using various clustering algorithms we try to discover customers who show the same behaviour in their transactions. We expect each cluster to represent a specific type of customer. Based on this 'customer profile' it might be possible to detect abnormal deviations in the customers' transactions and to react in time.

Neural Networks Approach

Artificial neural networks try to copy the way a human brain works. Thousands of simple (nerve) cells are highly interconnected and work massively in parallel. Using this kind of hardware a brain is able to learn from examples and to handle new, previously unknown situations or problems. Computer programs can execute predefined algorithms very fast and efficiently. But up to now a computer can hardly solve certain problems a human being can solve very easily: recognition of another known person, recognition of especially handwritten patterns like letters or numbers, keeping the balance e.g. in walking and so on. These capabilities are learned from examples and can be applied to new situations, new letters, new persons, or new footpaths.

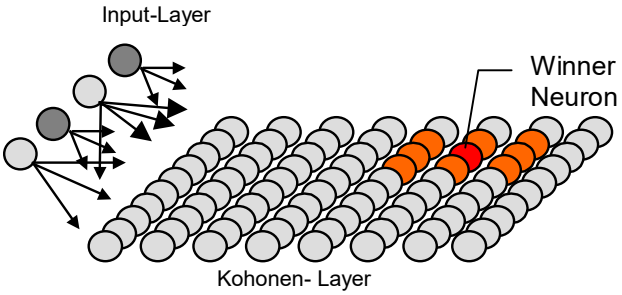
Artificial neural networks can be used to face problems which cannot be solved by algorithms. In other words: if an algorithmic solution is known, do not use neural networks at all. We use neural networks as a Data Mining technique. Up to know we have no data containing the information whether a certain customer has cancelled or not. We have no information if the customer uses the online transaction techniques in a proper way either. Thus we have no examples for training a neural network.

Unsupervised Learning

If no teaching examples are available a supervised training is impossible. Unsupervised learning is an approach in order to come around the problem. Self-organizing maps (or Kohonen feature map) are widely used within Data Mining. A self-organizing map (SOM) consists of two neuron layers, the input layer and the Kohonen layer.

Each input neuron is connected with all neurons of the Kohonen layer. Each connection is labelled by a parameter, the weight of the connection. The resemblance of the input vector X and the weight vector W is measured by the Euclidian distance. The neuron having the smallest distance to a certain input pattern is called the winner neuron. Winner neurons are widely spread over the Kohonen layer. Calculating the winner neuron and adapting the corresponding weights produces a distribution of winner neurons so that clusters can be detected. Roughly spoken a neuron is a representation of all input data for which this neuron is the winner neuron or the winner neuron lies within a small circle of that neuron. Figure 3 shows a self-organizing map. We use a SOM to cluster the transaction data.

Figure 3: Self-Organizing Map



The SOM-approach applied to our transaction data gave us the following results: all data sets, or all customers respectively, belonging to one cluster (the winner neuron) have a similar behaviour according to the artificial neural network. Unfortunately, it is not obvious why the neural network clusters the data in this way. Thus, we have to perform some more experiments.

An ART network can be used for clustering data as well. Whereas in a SOM it is not obvious how many different clusters will occur in an ART net we are able to define an upper limit for the number of different clusters. Although this seems to be very attractive to a data miner, ART networks have not been used so often. We take ART networks into consideration and will compare their behaviour with SOM behaviour. We expect better results using an ART network.

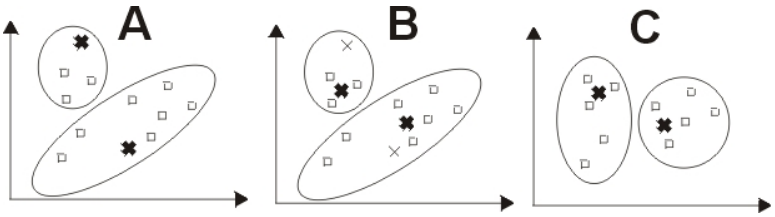
Clustering Algorithms

A lot of experiments have been performed using a clustering algorithm. Such algorithms are quite time consuming. The results we have gained have not been convincing yet. The algorithm has produced clusters we have not yet found an interpretation for.

K-Means

The K-Means algorithm is a well known and widely used clustering algorithm. The algorithm is based on the idea that objects (input vectors, records) are grouped into clusters according to a distance function, for example the Euclidian distance. The resulting clusters contain objects with a minimum within-cluster distance. The algorithm is performed as follows:

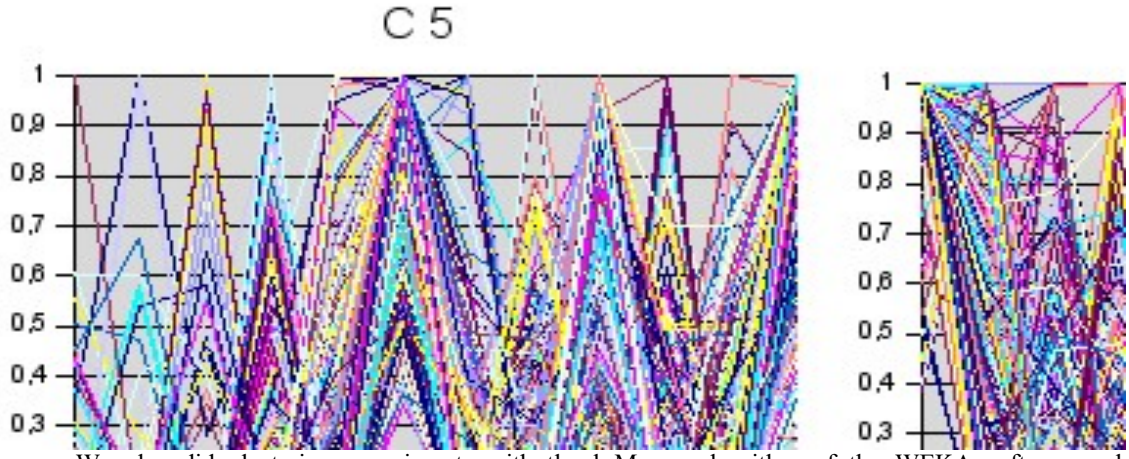
Figure 4: K-Means Algorithm / Cluster Adjustment



1. A number of so called centroids are randomly spread among the input range. For each cluster one centroid will be calculated.
2. Each object is assigned to the cluster represented by the closest centroid according to the distance function. (Figure 4A)
3. The new position of the centroid is found by calculating the centre of all objects that are assigned to that cluster. This will cause the centroids to 'move around'. Figure 4B)
4. Point 2. und 3. are repeated Figure 4C) until the centroids do not change any more.

Figure 5 shows two of twelve clusters as a result of one of our experiments. They show normalized records of approximately 2000 customers each during a time period of one year. The clustering was done using a 3x4 neuron matrix and 98,000 patterns. It can be seen that each cluster represents customers that have a peak in their transactions in one specific month. A further investigation is difficult because of the high number of records to be displayed.

Figure 5: Visualization of two clusters



We also did clustering experiments with the k-Means algorithm of the WEKA software, which produces less clusters, but a similar mixture of records within a cluster. We can say, clustering the original normalized data does not lead to any results, mainly because the clustering algorithms did not consider the time-dependent order of the values.

FOURIER TRANSFORMATION

As a result of our clustering experiments using the original normalized data, we found that it is necessary to transform the data in a way that allows the clustering algorithms to consider the order of the attributes.

The Fourier Transformation has its origin in signal processing. In theory, every signal can be described by superimposing periodic harmonic waves (such as sine or cosine) of defined frequency, phase and amplitude. For a signal consisting of *N* frequencies, the amplitude at the time of *t* is defined as:

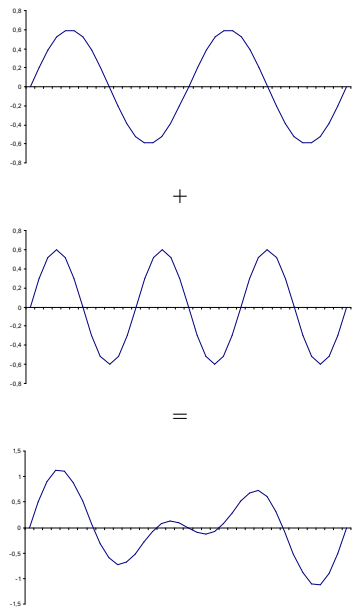
$$f(t)=\sum_{n=0}^{N} A_n \cos(n\omega t+\varphi_n)$$

where *A* is the amplitude, ω is the angular frequency and φ is the phase. This is also shown in Figure 6.

The Discrete Fourier Transformation is a special algorithm for the case that the signal is not continuous, as in our case. It decomposes an input vector of discrete values over time into its components: frequency, amplitude, and phase. Using this transformation, no information is lost, as the original data can be recalculated from the Fourier vector using the Inverse Fourier Transformation.

The length of the Fourier vector as well as the highest detectable frequency is determined by the number of values to transform. Let the period for ω be *T*. The first wave has the frequency 0, which is equal to a shift along the y-axis by the value of the amplitude. The other waves have frequencies from 1/*T* down to *N*/*T*, where *N* is half the number of elements in the input vector.

Figure 6: Superimposing Waves



From the Data-Mining point of view, each element of the Fourier vector describes an attribute of the original curve in its whole length, while the order of the attributes becomes irrelevant. This representation makes it easy to apply various pre-processing filters like:

- smooth the curve by filtering high frequencies
- remove frequencies with low amplitudes
- detect time-dependent shifts

Using a frequency spectrum or a Fourier vector for classification and clustering allows a more accurate handling of time-dependent data.

Figure 7 and Figure 8 show a transaction history (lines) and the corresponding frequency spectrum (bars). Figure 8 shows that similar transaction curves of two customers have a similar frequency spectrum. Because the curves are not shifted over time, the similarity can also be detected by just comparing both curves value by value. Most clustering algorithms will group them correctly.

Figure 7 shows three different customers. Each of them has a single peak in the transaction history, but at a different position. That means, although the transaction data is different, they behave in the same way, and that is what we want to discover. When using the original transaction data for clustering, most Data Mining algorithms will classify them as being different. But when using Fourier-transformed representation of the data, they will be grouped into one cluster.

Figure 7: Transaction History / Frequency Spectrum

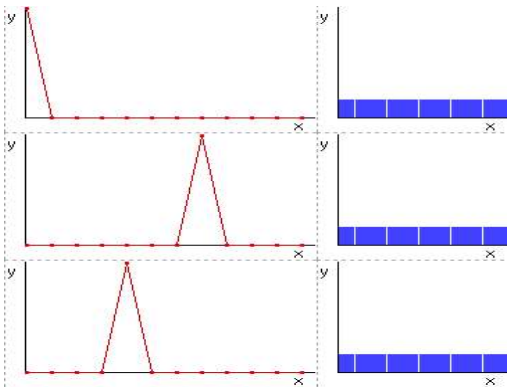
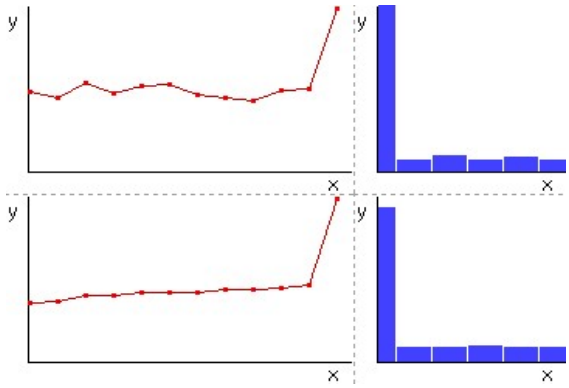


Figure 8: Data of similar customers



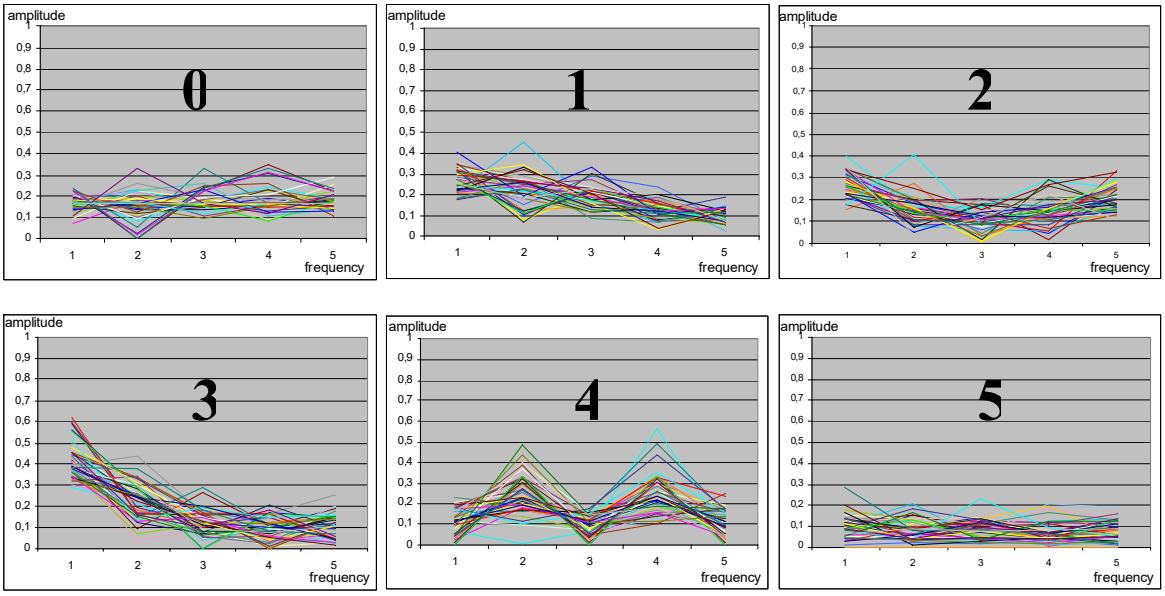
We have applied various Data Mining Algorithms to the Fourier-transformed data. Although we only used the amplitude part of the Fourier vector in our first experiments, we have found some interesting results. Figure 11 shows how the frequency spectrums are distributed over six clusters. Typical shapes and differences can be seen clearly.

The data is grouped into clusters using the frequency spectrum, but analysis of each cluster is based on the corresponding original transaction data. Each cluster contains accounts that show similar and specific transactions over the year. Some typical transaction curves are:

- one or two single transactions per year, resulting in peaks in the transaction curve
- seasonality; months with low transactions followed by months with high transactions
- relatively stable transactions with low deviations over the whole year

Further analysis included the distribution of other attributes for each customer/account, such as payment type, among the clusters. This work is still in progress, but we have already found that some clusters and therefore specific transaction curves seem to be typical for specific payment types as well as for the destination of the transaction, whether it is cross-border or not.

Figure 9: Frequency distribution on clusters



CONCLUSION

We have performed first experiments in order to analyse data of financial transactions. Our new approach to use Fourier-transformed data has shown a much better clustering, so we will be able to get better results in the near future.

A problem in predicting customer change is that we have no information about customers who already cancelled their contract with our partner, so we cannot apply supervised learning. Unsupervised learning, however, has the disadvantage that the results are often hard to understand. The interpretation of the results is difficult and time consuming. In co-operation with our partner we will try to interpret the results of the clustering experiments. Our aim is to get some more information so that we can apply supervised learning algorithms.

Actually, a complete analysis of the Data Mining results has not been made due to lack of proper visualization methods, which also restricts the maximum size of the SOM that can be inspected. The first problem is the very high number of records to be visualized using standard methods, e.g. diagrams. Second, the transaction histories are shifted over time, so we need to find a way to visualize the phase-element of the Fourier vector in a suitable way. We are currently developing visualization methods to address those problems.

The next steps include further analysis of the clustering results as well as developing the necessary tools for visualization and interactive investigation. A main point is the integration of original data into the clustering results which allows a faster and more detailed investigation. Then we will also be able to use bigger neuronal networks that allow a better granularity for clustering.

REFERENCES

[1] Füser, Karsten: Neuronale Netze in der Finanzwirtschaft; Gabler Verlag Wiesbaden, 1995.

[2] Lämmel, Uwe; Cleve, Jürgen: Lehr- und Übungsbuch Künstliche Intelligenz; Fachbuchverlag Leipzig, 2004.

[3] Witten, Ian H.; Frank, Eibe: Data Mining; Morgan Kaufmann Publishers, 2000.

[4] Zell, Andreas: Simulation Neuronaler Netze; Addison-Wesley, 1997.

[5] WEKA: http://www.cs.waikato.ac.nz/~ml/weka/ , last visited 2005-03-29.

[6] CRISP-DM: http://www.crisp-dm.org; last visited 2005-05-03.