Wissuwa, Stefan; Dipl. Wirt.-Inf.
Wismar University
s.wissuwa@wi.hs-wismar.de

Cleve, Jürgen; Prof. Dr.
Wismar University
j.cleve@wi.hs-wismar.de

Lämmel, Uwe; Prof. Dr.
Wismar University
u.laemmel@wi.hs-wismar.de

DATA MINING TO SUPPORT CUSTOMER RELATIONSHIP MANAGEMENT

Analyzing Transaction Data

Abstract

In a joint project with a German bank, data mining techniques are applied on transaction data of corporate clients in order to analyze and classify their behaviour. We describe an approach to analyze time series using standard data-mining-algorithms and to build classification models to detect significant deviations automatically. This can be used to build a 'ranking' system as part of the analytical CRM to be used by account managers to focus their activities on certain clients.

Introduction

Due to the intense competitive situation of financial institutions, effective customer relationship management has become a major factor of success. As the monetary flexibility of private customers increases, which is partially caused by the rising popularity of online banking, and so the decreasing importance of physically existing offices within the mobility range of a customer, the flexiblity of business customers is lower as they tend to have more complex account structures and financial service contracts. But this makes a business customer change a more serious loss.

Because of the high computerization in the banking sector, there are a lot of detailed, quickly accessible information about customers available. To use these information to optimize customer relationship is one essential aspect for a long-term successful and stable business.

Analytical CRM provides the information that, for example, help to segment and classify customers, to identify potential enhancements within the customer relationship, and to predict customer behaviour. The information used for analytical CRM depend on the branch and purpose. For financial

2nd International Conference Baltic Business and Socio-Economic Development,
Szczecin, Poland, September 2006.

institutions these are (among others): names, adresses, branches, accounts, used products and the complete history of transactions and turnover. As the turnover reflects the business activity directly, this should be a criteria for customer classification.

Possible outcomes and focus of analysis

If a typical behaviour of a customer or a segment is known, atypical deviations can be detected. Some of the possible reasons are, but not limited to:

- market changes, in which case other customers within the same or related segment should show similar behaviour
- internal difficulties, which are customer specific
- 'unusual' usage of banking products, which may indicate a improper banking software or the need of further training.
- the customer is about to change to a competitor, resulting in new arrangements and monetary transfers.

Access to these information may help account managers to take specific, well-placed actions, such as adjusting the product portfolio or conditions of contracts, which results in better customer relation.

In this paper we describe how to analyze transaction data in order to segment customers and to detect changes in their behaviour automatically. The tasks are:

- to find typical turnover functions,
- to segment customers into groups with similar behaviour,
- and to detect changes.

Data Mining

Data Mining, also know as Knowledge Discovery in Databases (KDD) is a common and powerful technique to analyze large amounts of data, often in terms of millions of records, giga- or even terabytes. It can be described as iterative, non-trivial search process for patterns. Data Mining can be seen as bottom-up approach on data analysis to extract possible hypotheses from available data - which often has been collected originally for different purposes - whereas statistics is a top-down approach to verify a previously defined hypothesis using a (likely) relevant set of data. But most Data Mining algorithms are based on or utilize statistics.

Data Mining covers various methods for specific tasks, such as:

- classification
- clustering
- association

- prognosis

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):

1. Business Understanding: defining objectives and requirements
2. Data Understanding: data collecting, checking quality and consistency
3. Data Preparation: preparing the data for the data mining algorithms / tools (includes 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
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

Clustering algorithms calculate a partitioning of a dataset into subsets (clusters) in a way that instances within a subset are more similar to each other than to instances within another subset. This is usually done using a distance measure *D(r1,r2)*, which is specific for the data to be compared. Typical distance measures are the Euclidian and the Manhattan distance.

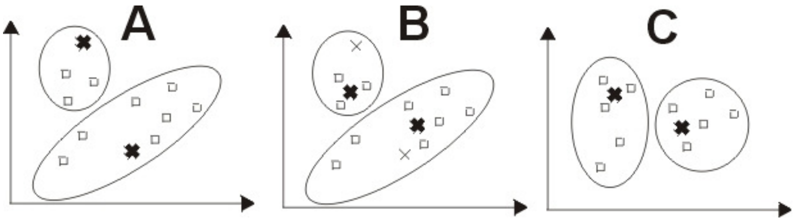

Figure 1: K-Means Algorithm / Cluster Adjustment

K-Means

The K-Means algorithm is a simple, 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:

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 1A)
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 1B)

4. Point 2. und 3. are repeated (Figure 1C) until the centroids do not change any more.

Neural Networks

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. while walking etc. These capabilities are learned from examples an can be applied to new situations, new letters, new persons or new footpaths. So, artificial neural networks can be used to face problems which cannot be solved by classical algorithms.

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 (SOM or Kohonen feature map) are widely used for data mining. A SOM consists of two neuron layers, the input layer and the Kohonen (or recognition) layer, as shown in Figure 2.

Each input neuron is connected with all neurons of the Kohonen layer. Each connection is labelled with 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 lowest distance to a certain input pattern is called the winner neuron. Initially, the winner neurons are widely spread over the Kohonen layer. Calculating the winner neuron and adapting the corresponding weights produces a distribution of the winner neurons so that nearby neurons have a similar weight vector. When training is done, similar input data will be projected onto the same or a nearby neuron, which leads to clusters that represent similar pattern (or records).

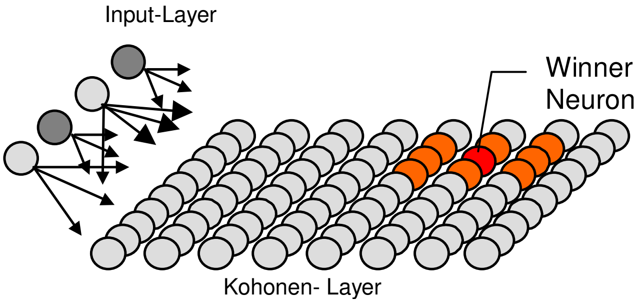

Figure 2: Self-Organizing Map

Statistics

Using statistics is a proper method for Data Mining. Four our purpose, the following statistical approaches may be considered suitable:

The classical statistic approach to describe a time series is to estimate a function of time $f(t,x_{0...n})$ in the first step and then adjust the parameters $x_{0...n}$ until the function meets the time series best, usually by minimizing the sum of squared residuals using methods like ordinare least square estimation. The problem with this method is, that this is a top-down approach, where the hypothesis is created for the data, not from the data. This means that only previously known - or at least expected – hypotheses can be verified. For our purpose – finding typical turnover functions – this means that turnover functions have to be invented first and then verified to see if there are records present which support the assumptions.

Transaction Data

The data are provided by our partner. It includes accounts of business customers, from small enterprises to big companies. The turnover is always cumulated over one month, personal information such as company name or real account numbers are anonymized, but the branch of a customer is known.

A single record holds the customer id, account number and the type of transaction as primary key, and the cumulated turnover per month including a time stamp. From this data we calculate a turnover history with variable length. The analysis shown here are based upon time series with a length of 12 months, which means that there are a total of 406,000 records available over a six-year period. Approximately 75% are distributed equally over the last three years.

Data Preparation

The success of data mining techniques depends highly on an appropriate preprocessing of the 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 relation between a certain attribute and the desired result is not obvious, the attribute has to be considered as well because 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 the data and group records according to the similarities.

Normalization

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

Transformation

As we deal with time-series, it is necessary to transform the data in a way that allows clustering algorithms to consider the order of the attributes. The distance functions used in clustering algorithms are not designed for time series data, but for a set of independent attributes.

Let there be two customers which have both a very saisonal dependent business. The time series vectors are *A* and *B*. The beginning of the business saison should be March for customer A, and May for customer B, so the transaction history of both customers should be identical, but shifted about 2 months. Common distance functions will calculate the distance between *A* and *B* using pairs of elements with the same indices [A_i,B_i]. Because *A* and *B* are pairwise different, both customers will also be classified as different.

The time series *B* is shifted against *A* about *s*, the distance should be calculated like

$$dist(A,B)=\sum_{i=0}^{n}\begin{cases} i \geq s : \Delta(A_{i-s}, B_i) \\ i < s : \Delta(A_{n-i+s}, B_i) \end{cases}$$

where *Δ* is the pairwise distance and *Σ* is the overall distance. Except in our example, *s* is initially not known, so this does not solve our problem. Of course it is possible to calculate *s* for a vector *V* and an unshifted similar reference-vector *R*, but *R* is also not known for the reason that there are a initially unknown number R_n of vectors, each representing a typical turnover function. And finding these R_n is exactly the goal of clustering, because each R_i is the center of one cluster.

For the above reasons we need a way to overcome the limitations of only having independent attributes. Therefore we need a representation of our time series where each attribute describes one aspect of the whole time series, so they are independent from each other.

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

The Discrete Fourier Transformation is a special algorithm in case the signal is not continuous, as for transaction data. It decomposes an input vector of discrete values over time into its components: frequency, amplitude, and phase. The complexity is usually $O(N^2)$. If *N* is a power of 2, a special algorithm called Fast Fourier Transformation can be used which reduces the complexity to *O(N·log N).* Both are a lossless transformations, the original data can be recalculated from the Fourier vector using the inverse Fourier-Transformation, also know as Fourier Synthesis. Figure 4 shows an example for a turnover history and its frequency- and phase spectrum.

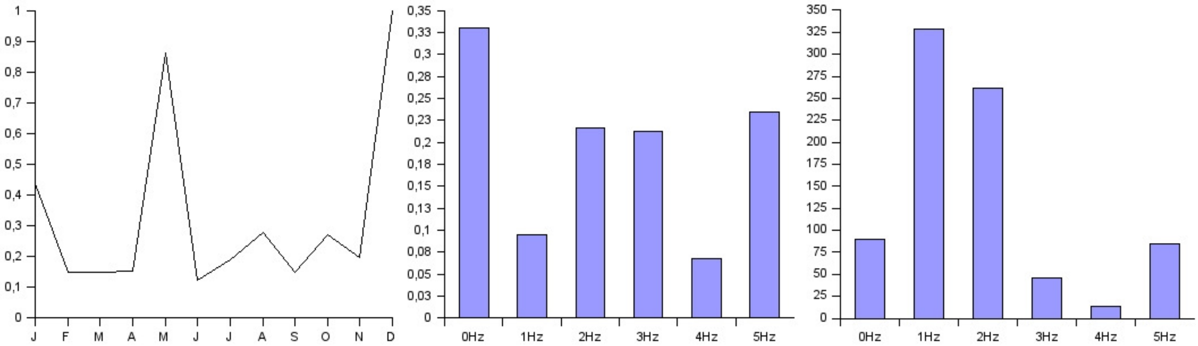

Figure 3: turnover, frequency- and phase-spectrum

The length of the Fourier vector as well as the highest detectable frequency is determined by the length of the input vector. 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* to *N*/*T*, where *N* is half the number of elements in the input vector. The discrete Fourier transformation is a complex multiplication of the input vector *V* with a signal vector for each frequency *f*. The real part of the complex Fourier vector is calculated by:

$$F_r[f] = \frac{1}{N}\sum_{n=0}^{N}[V_n \sin(\omega f)]$$

Analogous, the imaginary part is calculated by:

$$F_i[f] = \frac{1}{N}\sum_{n=0}^{N}[V_n \cos(\omega f)]$$

This can easily be transformed into pairs [amplitude, phase] using trigonometry. For visualization purposes, the shift of the time series along the time axis can be calculated from the phase spectrum, which allows a justified visualization. A shift along the time axis about +*n* results in a phase shift of $+n\omega$.

$$\varphi_{t+n} = \varphi_t + n\omega$$

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 also 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

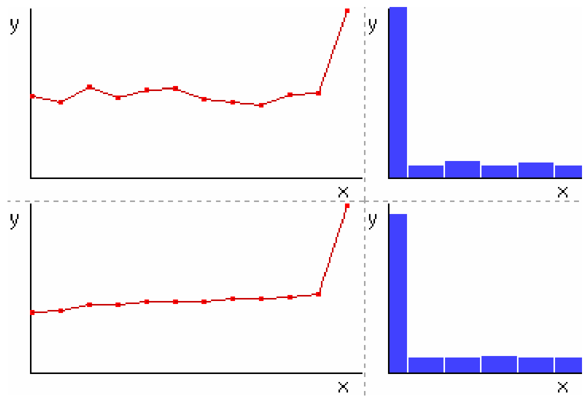

Figure 4: frequency and phase spectrum

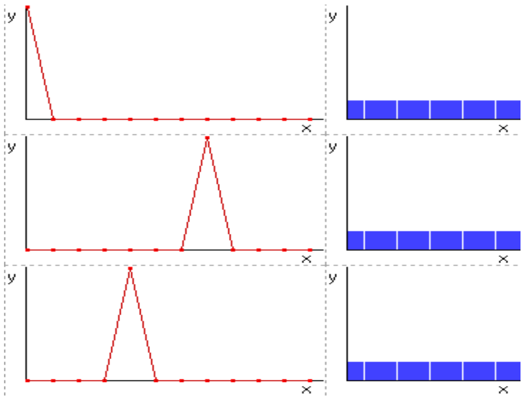

Figure 5: frequency and phase spectrum

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

Figure 4 and Figure 5 show a set of transaction histories (lines) and the corresponding frequency spectrum (bars).

Figure 4 shows that similar transaction curves of two customers have a similar frequency spectrum. Because the curves are not shifted along the time axis, the similarity can also be detected by just comparing both curves value by value. Most clustering algorithms will group them correctly.

Figure 5 shows transactions of 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: each of them uses its account exactly once per year. This kind of similar behaviour 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 the frequency part of the Fourier-transformed data, the accounts will be assigned to the same – or at least very similar – cluster.

Advantages / Disadvantages

The main advantage of the described method is that common Data Mining algorithms can be used to analyze time series data without modification, which in most cases is not even possible for the researcher due to limited access to the source code. The adaption to the specific requirements of handling time series is completely performed in the preprocessing stage.

On a AMD Athlon XP 2800+ with 1GB of RAM, MySQL 5.0 Database and SuSE Linux 8.1 Operating System, the whole process from getting the data from the database, normalization and fourier transformation takes about 1 seconds per 10,000 records.

Results & Conclusion

We did a clustering using the frequency spectrum of 140,000 records (1 record per account) and a Self-organizing map with 6x5 neurons. This configurations allows a maximum of 30 clusters. After training the network with a 12-element time series from $t_0 - t_{11}$, we applied the trained SOM as classification model to a time series from $t_1 - t_{12}$, expecting a account with significantly changed behaviour should result in a different cluster assignment, which was the case for about 19% of all records. We found that the consistency seems to correlate with the customers branch. The most significant branches with the highest respectively lowest number of changed cluster assignments are shown in tables 1 and 2.

In general, our results show that using Fourier-transformed data for clustering is a promising approach. But we have notice a relatively high number of changed cluster assignments which we did not expect. This may be accidentially caused by the following factors, which needs to be analysed:

- A unsuitable size of the recognition layer of the used SOM may cause a too high or to low number of clusters. If the number of clusters is to high, a relatively small deviation will result in a different winner neuron, which we interpreted as cluster change. In this case, a cluster will consist of a number of nearby neurons. On the other hand, if the number of clusters is to small, instances are forced into a cluster which represents different characteristics.
- The chosen length of the time period may have been to small. If a account shows a cycle which lasts longer than the chosen period, it is likely to be reported as changed although it is still within its typical behaviour.
- For our experiments, we omitted the phase spectrum, which results in a loss of 50% of available information. This was intended due to the reasons explained before. But some turnover functions may heavily depend on the phase spectrum, which then cannot be clustered correctly. Therefore we need to find a measure for the importance of the phase spectrum and a way to integrate it into our clustering experiments.

Table 1: Branches with high variability

Changes in %	Branche	Changed/total records
22.3	Taxi business	239/1070
22.3	Ship Broker Offices	64/471
20.9	Churches	228/1091
20.2	Trucking	1010/5008

Table 2: Branches with low variability

Changes in %	Branche	Changed/total records
7.7	Sewage	3/39
8.8	Container Shipping	67/755
10.0	General Shipping	26/255
11.0	Passenger Shipping	9/85

References

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

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

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

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

[5] Callan, Robert: The Essence of Neural Networks, Pearson Education, 2002.

[6] WEKA: http://www.cs.waikato.ac.nz/~ml/weka/ , last visited 2006-09-29.

[7] CRISP-DM: http://www.crisp-dm.org; last visited 2006-09-29.