VTT INFORMATION TECHNOLOGY

RESEARCH REPORT TTE1-2001-18

LOUHI

Overview of Data Mining for Customer Behavior Modeling

Version 1

29 June, 2001

by

Catherine Bounsaythip and Esa Rinta-Runsala

Version history

Version	Date	Author(s)	Reviewer	Description
0.1-1	15.3.01	C. Bounsaythip	E. Rinta-Runsala, S. Pensar	First draft
0.1-7	14-05-01	C. Bounsaythip, E. Rinta-Runsala	S. Pensar	Draft
1	29.06.01	C. Bounsaythip, E. Rinta-Runsala	J. Kiviniemi	Final

Contact information

Catherine Bounsaythip
VTT Information Technology
P.O. Box 1201, FIN-02044 VTT, Finland
Street Address: Tekniikantie 4 B, Espoo
Tel. +358 9 456 5957, fax +358 9 456 7024
Email: Catherine.Bounsaythip@vtt.fi
Web: http://www.vtt.fi/tte/

Last modified on 2 August, 2001
P:\louhi\Results\profilingSurveyFinal.doc

UDK
Key Words Data mining, customer segmentation, customer profiling, Web mining

Abstract

This report examines the problems of customer relationship management (CRM) particularly customer segmentation and customer profiling, and how data mining tools are used to support the decision making. We first describe the steps towards predicting customer's behavior, such as collecting and preparing data, segmentation and profile modeling. Then, we present a general overview of most used data mining methods including cluster discovery, decision trees, neural networks, association rule and sequential pattern discovery. The report also covers a discussion about Web mining which is treated as a separate section due to its current popularity in electronic commerce. A guideline to choose a data mining software package is also given in the last section.

Preface

This report is written within the frame of LOUHI project. The purpose of this report is to provide to our project partners a general view on data mining methodologies used in today's customer relationship management (CRM).

In today's marketing strategies, customers have a real value to the company. Therefore, it is essential to any company to be successful in acquiring new customers and retain those that have high value. For this, many companies have gathered significant numbers of large and heterogeneous databases and these data need to be analyzed and applied in order to develop new business strategies and opportunities. The problem is that "*we are rich in data and poor in information*". What are the methods that can be used to automatically extract knowledge from data? Recently, new data analysis tools have appeared using various "machine learning" techniques. It is through the use of machine learning that data mining tools emerge. The advantage of data mining is that it can handle large amount of data and "learn" inherent structures and patterns in data; it can also generate rules and models that are useful in replicating or generalizing decisions that can be applied to the future cases. Data mining tools are therefore very useful in market segmentation, customer profiling, risk analysis, and many other applications.

The growing interests in data mining tools have also fostered the growth of the data mining tool market. Nowadays, there are so many vendors offering all range of products. For a person who is new in the field and would like to use those tools, it is essential that he/she understands how each method works in order to be able to choose the adequate ones for his/her problems. This report aims tempts to make the reader familiar with the most used data mining methods (clustering, decision trees, neural networks, associations...) and also the emerging Web mining techniques. We do not intend to provide a complete overview of each technique, neither to compare systems.

Contents

1 Introduction

Nowadays market is characterized by being global, products and services are almost identical and there is an abundance of suppliers. And because of the size and complexity of the markets, mass marketing is expensive (Figure 1) and the returns on investment are frequently questioned.

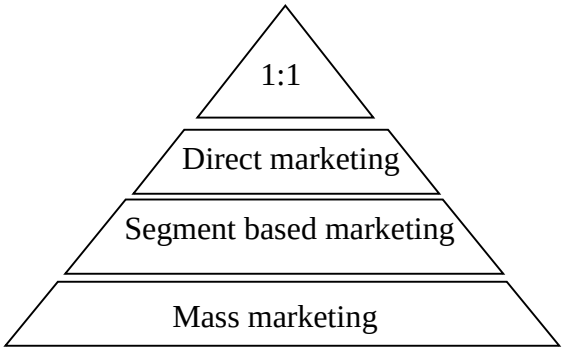


Figure 1: Communication with customers, B2B & B2C [Price99].

Instead of targeting all prospects equally or providing the same incentive offers to everyone, a firm can select only those customers who meet certain profitability criteria based on their individual needs and buying patterns [Price99]. This is achieved by building a model to **predict the future value of an individual** based on his demographic characteristics, life-style, and previous behavior. The model produces information that will focus customer retention and recruitment programs on building and keeping the most profitable customer base. This is called *customer behavior modeling* (CBM) or *customer profiling*. A *customer profile* is a tool to help target marketers better understand the characteristics of their customer base.

The long term motivation of customer profiling is to turn this understanding into an *automated* interaction with their customers [Price99, Thearling00]. For these tasks, today's marketplace needs wide ranges of processes and IT tools. These tools have been used to collect data and simplify the process of extracting knowledge about the market and planning marketing campaign. Data mining tools have been used to identify meaningful *groups* in the historical data, e.g. selection criteria for mailing lists, or to identify markets with high potential, or media or lifestyle characteristics that match target customers. In brief, data mining tools are able to find human interpretable patterns that **describe** the data; they are also able to use some variables to **predict** unknown or future values of other variables.

The way how a data mining system describes data and predicts the future value of a variable will be explained in the further sections.

After this introductory section, section 2 gives a general view on the problem of customer segmentation and profiling. The section includes data collection and preparation and general methodology for customer profiling

Section 3 gives an overview of the most used data mining methods: clustering, classification and regression, associations and sequential pattern analysis. For each technology, a brief definition and illustration are given. In order to keep the readability of the report, technical details are sometimes skipped on purpose. At the end of the section, we give a summary table gathering most characteristic features of the described data mining technologies.

Section 4 is dedicated to Web Mining. Web mining is another kind of data mining for Web data. We give definition of what constitutes Web data and the techniques used for extracting knowledge from them.

Section 5 gives guidelines for choosing the many data mining packages that are offered in the market. i.e. a list of checkpoints for purchasing a software package.
Section 6 gives some case studies for illustration of data mining application on real data (potential loan customers for a bank, churn prediction in telecommunication and online bookselling).

2 Customer segmentation and customer profiling

Customer relationship management (CRM) includes *customer segmentation* and *customer profiling*.

Customer segmentation is a term used to describe the process of dividing customers into homogeneous groups on the basis of shared or common attributes (habits, tastes etc.)

Customer profiling is describing customers by their attributes, such as age, income, and lifestyles. This is done by building a customer's behavior model and estimating its parameters. Customer profiling is a way of applying external data to a population of possible customers. Depending on data available, they can be used to prospect new customers or to "drop out" existing bad customers. The goal is to predict behavior based on the information we have on each customer [Thearling00]. Profiling is performed after customer segmentation.

Having the two components, marketers can decide which marketing actions to take for each segment and then allocate scarce resources to segments in order to meet specific business objectives (Figure 2).

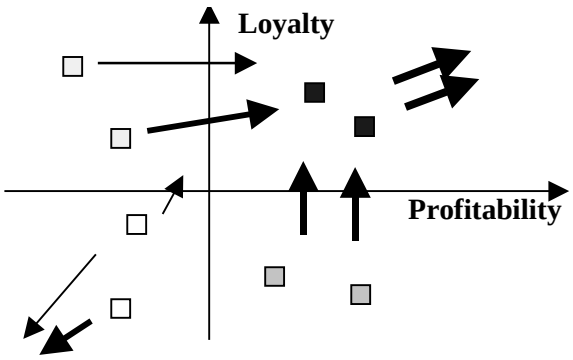


Figure 2: Segmentation offers to a company a way to know about loyalty and profitability of their customers. And knowing the profile of each customer, the company can treat the customer according to his/her individual needs in order to increase the lifetime value of the customer [Price99].

In the following, we define the methodologies to tackle the two problems.

2.1 Customer segmentation

Segmentation is a way to have more targeted communication with the customers. The process of segmentation describes the characteristics of the customer groups (called *segments* or *clusters*) within the data. Segmenting means *putting the population into segments* according to their affinity or similar characteristics. Customer segmentation is a preparation step for classifying each customer according to the customer groups that have been defined.

Segmentation is essential to cope with today's dynamically fragmenting consumer marketplace. By using segmentation, marketers are more effective in channeling

resources and discovering opportunities. Difficulties in making good segmentation are [AH98]:

Relevance and quality of data are essential to develop meaningful segments. If the company has insufficient customer data or too much data, that can lead to complex and time-consuming analysis. If the data is poorly organized (different formats, different source systems) then it is also difficult to extract interesting information. Furthermore, the resulting segmentation can be too complicated for the organization to implement effectively. In particular, the use of too many segmentation variables can be confusing, resulting in segments which are unfit for management decision-making. Alternatively, apparently effective variables may not be identifiable. Many of these problems are due to an inadequate customer database.

Intuition: Although data can be highly informative, marketers need to be continuously developing segmentation hypotheses in order to identify the 'right' data for analysis.

Continuous process: Segmentation demands continuous development and updating as new customer data is acquired. In addition, effective segmentation strategies will influence the behavior of the customers affected by them; thereby necessitating revision and reclassification of customers. Moreover, in an e-commerce environment where feedback is almost immediate, segmentation would require almost a daily update.

Over-segmentation: A segment can become too small and/or insufficiently distinct to justify treatment as separate segments.

The data mining methods used for customer segmentation belong to the category of **clustering** or **nearest-neighbors algorithms**.

2.2 Customer profiling

Customer profiling provides a basis for marketers to "communicate" with existing customers in order to offer them better services and retaining them. This is done by assembling collected information on the customer such as demographic and behavioral data. Customer profiling is also used to prospect new customers using external sources, such as demographic data purchased from various sources. This data is used to break the database into clusters of customers with shared purchasing traits [WWW1].

Depending on the goal, one has to select what is the profile that will be relevant to the project. A simple customer profile is a file containing at the least his name, address, city, state, and zip code. And if one needs profiles for specific products, the file would contain product information and/or volume of money spent.

The customer features that can be used for profiling are [Wilson00]:

- *Geographic*. Are they grouped regionally, nationally or globally?
- *Cultural and ethnic*. What languages do they speak? Does ethnicity affect their tastes or buying behaviors?
- *Economic conditions, income and/or purchasing power*. What is the average household income or purchasing power of the customers? Do they have any payment difficulty? How much or how often does a customer spend on each product?

- For acquired customer, *shopping frequency*, *frequency of complaints*, *degree of satisfaction*, *preferences* may be used to build a purchase profile.
- *Age*. What is the predominant age group of your target buyers? How many children and of what age are in the family?
- *Values, attitudes, beliefs*. What is the customers' attitude toward your kind of product or service?
- *Life cycle*. How long has the customer been regularly purchasing products?
- *Knowledge and awareness*. How much knowledge do customers have about a product or service, or industry? How much education is needed? How much brand building advertising is needed to make a pool of customers aware of the offer?
- *Lifestyle*. How many lifestyle characteristics about purchasers are useful?
- *Media used*. How do targeted customers learn? What do they read? What magazines do they subscribe to?
- *Recruitment method*. How was the customer recruited?

2.3 Data collection and preparation

There are many ways of collecting the data:

- *In-house customer database*. Names can come from direct mailers used in the past, frequent buyer programs, contest, warranty registrations, receipts and membership cards.
- *External sources.* There are software or databases that can discover lifestyle, demographic information by using for example only zip code. E.g. *SuomiCD* can find Finnish demographic data from Finnish zip code. *Mosaic boxes* [WWW2] can also provide lifestyle data on small areas (boxes) of different countries including Finland.
- *Research survey* either face-to-face, over the telephone, via a postal questionnaire or through Internet.

There are two types of information from the data that should be collected : classification variables and descriptor variables [WWW3].

2.3.1 Classification variables

Classification variables are used to classify survey respondents into segments. These variables are demographic, geographic, psychographic or behavioral variables.

- *Demographic variables* - Age, gender, income, ethnicity, marital status, education, occupation, household size, length of residence, type of residence, etc.
- *Geographic variables* - City, state, zip code, census tract, county, region, metropolitan or rural location, population density, climate, etc.
- *Psychographic variables* - Attitudes, lifestyle, hobbies, risk aversion, personality traits, leadership traits, magazines read, television programs watched, etc.
- *Behavioral variables* - Brand loyalty, usage level, benefits sought, distribution channels used, reaction to marketing factors, etc.

2.3.2 Descriptor variables

Descriptors are used to describe each segment and distinguish one group from the others.

Descriptor variables must be easily obtainable measures or linkable to easily obtainable measures that exist in or can be appended to customer files. Many of the classification variables can be considered descriptor variables. However, only a small portion of those classification/descriptor variables are readily available from external sources.

2.3.3 Data preparation

Before the data can be introduced to a data mining tool, they need to be cleaned and prepared in a required format [Kimball97]. These tasks are:

- Resolving inconsistent data formats and resolving inconsistent data encoding, geographic spellings, abbreviations, and punctuation.
- Stripping out unwanted fields. Data may contain many fields that are meaningless from an analysis point of view, such as version numbers and formatted production keys.
- Interpreting codes into text. This means to augment or replace cryptic codes with textual equivalents written in recognizable words.
- Combining data such as customer data from multiple sources under a common key.
- Finding multiple used fields. A good way to find it out is to count and perhaps list all the distinct values residing in a field.

The following data preparations may be needed in some data mining tools:

- Checking out abnormal, out of bounds, or ambiguous facts. Some measured facts may be correct but highly unusual, thus unexplainable.
- Checking missing values or if they have been replaced by a default value.
- Adding computed fields as inputs or targets.
- Mapping continuous values into ranges, e.g. for decision trees.
- Normalizing values between 0 and 1, e.g. for neural networks.
- Converting nominal data (like *yes*/*no* answers) to metric scales.
- Converting from textual to numeric or numeral category, e.g. locations into ZIP codes.

New fields can be generated through combinations, e.g. *frequencies*, *cross-tabulations*, *averages* and *minimum*/*maximum* values, *relationships* between different profiling variables etc. The number of variables can be reduced to a more manageable size while also removing correlations between each variable. Techniques used for this purpose are often referred to as *factor analysis*, *correspondence analysis* and *conjoint analyses* [WWW3].

When there is a large amount of data, it is also useful to apply data reduction techniques (data cube aggregation, dimension and numerosity reduction, discretization and concept hierarchy generation). Dimension reduction means that one has to select relevant feature to a minimum set of attributes such that the resulting probability distribution of data

classes is as close as possible to the original distribution given the values of all features. For this additional tools may be needed, e.g. exhaustive, random or heuristic search, clustering, decision trees or associations (see further).

2.4 Model building

Modeling is essentially a means of determining whom to target in a marketing action. Profiling techniques serve as a useful precursor to model-building, since they help break a vast mailing population into manageable clusters. The model obtained will fine tune assumptions by providing comparative information (cluster vs. cluster performance) for a specific marketing scenario. Modeling usually involves test mailing to a sample that is representative of the company's database. The model is then constructed by analyzing the response from that mailing, determining how each demographic and behavioral variable affected the response.

In order to build a customer's behavior model, we need to :

- *Identify the variables* for inclusion in the profile - e.g. products bought, length of time a customer is loyal, etc.
- *Build the model* which segments and profiles the different customers.
- *Use the model to predict* which customers are most likely to buy, respond to a cross-selling offer or defect to the competitor, etc.
- *Identify the most discriminating data variables* - i.e. the variables which are most effective at predicting a customer's likelihood of buying, propensity to take-up a cross-selling offer or defect.

Data mining is a machine learning variation which uses a database as training set. Historical data are-- used for training and getting the model, and new data are used for prediction (*Figure 3*).

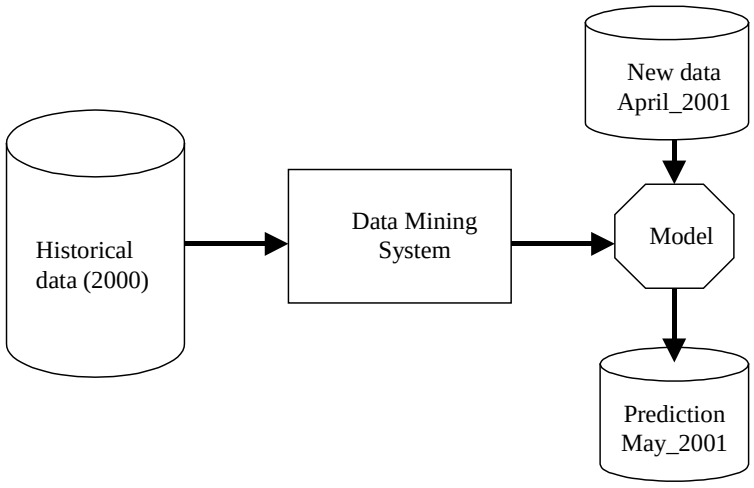


Figure 3: The information gathered into the historical database about the customer would be used to build a model of customer behavior and could be used to predict which customers would be likely to respond to the new catalog [Thearling00].

2.4.1 Data sampling

The historical data are sampled into different datasets. These datasets should be distinct from each other. *Random sampling* is the most used sampling technique. If there is a lot of data, a sample limited to a small percentage of the whole dataset will still capture enough variability to be able to generate a good model.

The datasets are:

- *Training set* is the data necessary to build a predictive model.
- *Test set* is used during the modeling phase by the model builder to evaluate the accuracy of a particular model.
- *Validation set* is often used by the data miner to evaluate the accuracy of the final model by comparing predictions from the model to known outcomes.
- *Control set* is (optionally) used in some techniques while building the model to control *over-training*.

2.4.2 Training, testing and validating the model

The first phases of building a model is *training* or *learning* process. It uses the data from the training set which is composed of cases where the outcome is known and included. The field or **variable that contains the outcome,** e.g. credit risk, is called the **dependent** or **target variable**. All of the other fields, e.g. income or marital status, are called the **independent** variables.

If the data is not very accurate (or *noisy*), the inaccuracies (i.e. *noise*) may affect the building of the predictive model and the model will make incorrect predictions to the degree that it has been affected by errors in the training data. Such a model is said to be *overtrained* (see Figure 4).

Once we get a model, we have to **validate** it on an independent dataset, the one that has not been used to create the model.

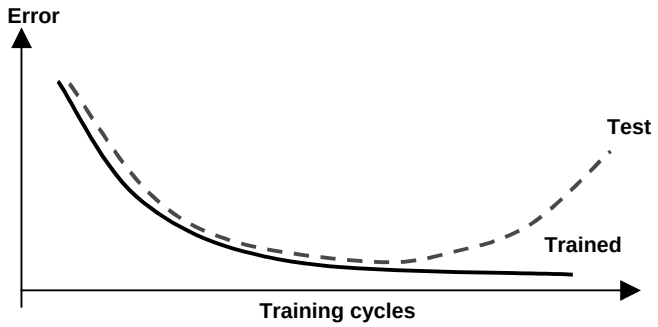


Figure 4: Example of over-training or overfitting. This phenomena happens when there are too many data in the training set or when the training phase lasts too long. One simple way to avoid this is to stop the training before the algorithm converges.

2.4.3 Model deployment

Deploying the model means that one has to **identify the most discriminating data variables** - i.e. the variables which are most effective at predicting a customer's likelihood of buying, or propensity to take-up a cross-selling offer or defect to the competitor. Deployment may require building computerized systems that capture the appropriate data and generate a prediction in real time so that a decision maker can apply the prediction. For example, a model can determine if a credit card transaction is likely to be fraudulent.

2.4.4 Updating the model

It is necessary to monitor the model because the environment changes. The economy changes, competitors introduce new products, or the news media finds a new hot topic. Any of these forces will alter customer behavior. So the model that was correct yesterday may no longer be very good tomorrow. Monitoring models requires constant revalidation of the model on new data to assess if the model is still appropriate.

Summary of different steps in model building is depicted in Figure 5.

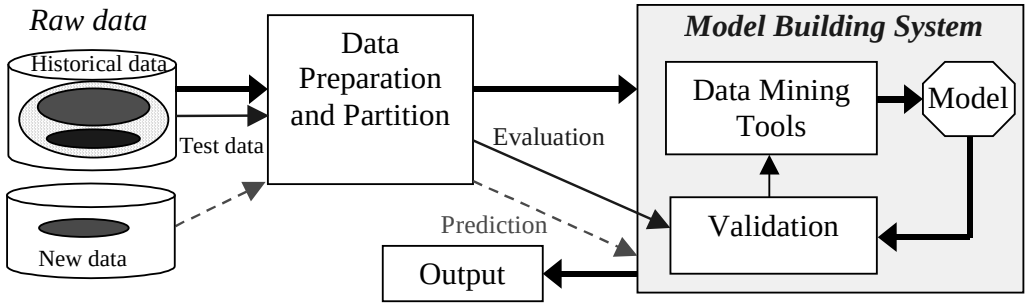


Figure 5:Building a model involves data preparation and data partition (often included in the package), then running the data mining tool to get a model. The model is validated with a dataset that was not used in creating the model. Several models can be obtained with different tools and the best one is selected to make predictions with new data.

3 Data mining techniques

In the following we give a quick overview of the most used data mining methods. Understanding how each data mining technique works can help to choose appropriate techniques to a specific business problem.

3.1 Introduction

Considering the amount of information available for each customer, it is clearly non-trivial to set up manually a query like :

If age <25 and job="worker" and credit amount < 5.000,- then include customer in mailing.

Data mining and programming tools help to extract automatically prototypical customer profile from data. They are also useful to visualize non-linear interaction of variables. There are several data mining methods (see overviews [HAS94, CHY96, GG99, Hall99]), and determination of which to apply can be decided by the quality of the data, the situation, and the objective. There are methods that create analytical models for ***prediction***, ***description*** or both [WWW5].

A *predictive model* can answer questions such as "*Is this transaction fraudulent*," or "*How much profit will this customer generate*."

Descriptive models provide information about the *relationships* in the underlying data, generating information of the form "*A customer who purchases diapers is 3 times more likely to also purchase beer*," or "*Households with incomes between $60,000 and $80,000 and two or more cars are much more similar to each other than households with no children and incomes between $40,000 and $60,000*."

Most but not all predictive models are also descriptive. Some descriptive models cannot be used for prediction.

The most known data mining methods are:

- Clustering (descriptive),
- Classification (predictive) and regression (predictive),
- Association rule discovery (descriptive) and sequential pattern discovery (predictive).

Clustering is a technique that puts similar entities into the same groups based on similar data characteristics and those with dissimilar entities are put in different groups. Similarity is measured according to a distance measure function. The meaning of the clusters is therefore dependent on the distance function used. Thus, clustering always requires significant involvement from a business or domain expert who needs to both propose an appropriate distance measure to judge whether the clusters are useful.

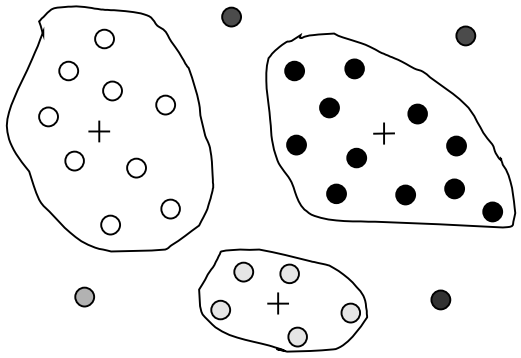


Figure 6: With clustering, data are organized into groups of "similar" values. Rare or unclassifiable values can de discarded.

Clustering supports the development of population segmentation models, such as demographic-based customer segmentation. Clustering techniques include k-means or **k-nearest neighbours** (**k-NN**), a special type of neural network called Kohonen net or self-organising maps (**SOM**).

Classification and regression represent the largest part of problems to which data mining is applied today, creating models to predict class membership (classification) or a value (regression). Classification is used to predict what group a case belongs to.

Regression is used to predict a value of a given continuous valued variable based on the values of other variables, assuming a linear or non-linear model of dependency. Logistic regression is used for predicting a binary variable. It is a generalization of **linear regression**, because the binary dependent variable cannot be modeled directly by linear regression. Logistic regression is a classification tool when used to predict categorical variables such as whether an individual is likely to purchase or not, and a regression tool when used to predict continuous variables such as the probability that an individual will make a purchase.

There are several classification and regression techniques including **decision trees**, **neural networks**, Naïve-Bayes and nearest neighbor.

Association and sequencing tools analyze data to discover rules that identify patterns of behavior, e.g. what products or services customers tend to purchase at the same time, or later on as follow-up purchases. While these approaches had their origins in the retail industry, they can be applied equally well to services that develop targeted marketing campaigns or determine common (or uncommon) practices. In the financial sector, association approaches can be used to analyze customers' account portfolios and identify sets of financial services that people often purchase together. They may be used, for example, to create a service "bundle" as part of a promotional sales campaign.

3.2 K-Nearest neighbors

3.2.1 Definition

K-nearest neighbor is a predictive technique suitable for classification models. K represents a number of similar cases or the number of items in a group. With the *k-NN* technique, the training data is the model. When a new case or instance is presented to the model, the algorithm looks at all the data to find a subset of cases that are most similar to it and uses them to predict the outcome.

There are two principal parameters in the *k*-NN algorithm:

1. the number of nearest cases to be used (*k*);
2. a metric to measure the similarity.

Each use of the *k-NN* algorithm requires that a positive integer value for *k* is specified. This determines how many existing cases are looked at when predicting a new case. For example, 4-*NN* indicates that the algorithm will use the four nearest cases to predict the outcome of a new case.

3.2.2 Algorithm

K-NN decides into which class to place a new case by examining some number (the *k*) of the most similar cases or neighbors. The algorithm computes the distance from the new case to each case in the training data. The new case is predicted to have the same outcome as the predominant outcome in the *k* closest cases in the training data. So, the new case is assigned to the same class to which most of the similar cases belong (Figure 7).

Suppose that we have *n* example feature vectors x_1, x_2, ..., x_n all from the same class, and we know that they fall into *c* compact clusters, $c < n$. Let m_i be the mean of the vectors in *Cluster i*. If the clusters are well separated, we can use a minimum-distance classifier to separate them. That is, we can say that *x* is in *Cluster i* if the distance $\| x - m_i \|$ is the minimum of all the *k* distances.

K-NN is based on a **concept of distance**, and this requires a metric to determine distances. For continuous attributes Euclidean distance can be used, for categorical variables, one has to find a suitable way to calculate the distance between attributes in the data. Choosing a suitable metric is a very delicate task because, different metrics, used on the same training data, can result in completely different predictions. This means that a business expert is needed to help determine a good metric.

3.2.3 Illustration

Suppose we have to classify a new case *N*. The algorithm will compare *N* with its *k* nearest neighbors. It will be assigned to class X because it is closer to more X's than Y's (Figure 7).

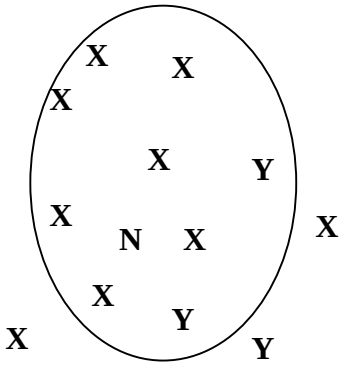


Figure 7: Example of k-NN classification. N is a new case. It would be assigned to the class X because within the ellipse the number of X's is higher than that of Y's.

3.2.4 Advantages/Disadvantages

With *k-NN*, the number of classes is usually given as an input variable, in some problems it is not always easy to guess.

K-NN is a huge model because it uses the entire training set as the model. *K-NN*'s calculation time increases as the factorial of the total number of points because k-NN requires the calculation to be made for every new case.

Distance based similarity measure cannot cope with high dimensional data, because the notion of neighborhood becomes meaningless. In this case, a more sophisticated method is used (see SOM). Another problem is related to handling data inaccuracies and cluster overlapping: how to decide in which group to put those data that are on the cluster boundary region. Fuzzy approach can be used. A fuzzy *k*-nearest neighbors (fuzzy *k-NN*) technique is simply a nearest neighbors technique in which the basic measurement technique is fuzzy [Hansen00].

3.3 SOM

3.3.1 Definition

When the set of inputs is multi-dimensional, traditional clustering algorithms do not offer an easy way to visualize the "closeness" of other clusters. A self-organizing map or Kohonen feature map is a special kind of neural network architecture that provides a mapping from the multi-dimensional input space to a lower-order regular lattice of cells (typically 2 dimensional grid). Such a mapping is used to identify clusters of elements that are similar (in a Euclidean sense) in the original space.

In a SOM, the clusters are usually organized into a lattice of cells, usually a two-dimensional grid but also one-dimensional or multi-dimensional. The grid exists in a space that is separate from the input space; any number of input features may be used as long as their number is greater than the dimensionality of the grid space. A SOM tries to find clusters such that any two clusters that are close to each other in the grid space have cluster close to each other in the input space. But the converse does not hold: cluster centroids that are close to each other in the input space do not necessarily correspond to clusters that are close to each other in the grid. For more information, the reader is referred to [Kohonen95, WWW4].

Unlike other neural network approaches, the SOM network performs unsupervised training. The more common approach to neural networks requires supervised training of the network, i.e., the network is fed with a set of training cases and the generated output is compared with the known correct output. The SOM network, on the other hand, does not require the knowledge of the corresponding outputs. The nodes in the network converge to form clusters to represent groups of entities with similar properties. The number and composition of clusters can be visually determined based on the output distribution generated by the training process.

The SOM network is typically a feed-forward neural network with no hidden layer (see further section) : it has two layers of nodes, the input layer and the Kohonen layer. The input layer is fully connected to a two- or one- or multi-dimensional Kohonen layer. It performs unsupervised learning (i.e. without examples of known output), and uses Euclidean measure for distance measure.

3.3.2 Algorithm

The algorithm works in a relatively straightforward manner. Cluster centroids are assigned a location in spatially organized matrix. The data is processed with the following steps:

1. Assign a "neighborhood" function which, for a given centroid, identifies "neighboring" cluster centroids.
2. For each data point:
   3. Find the cluster centroid which is "closest" to the data point (the "*winner*");
   4. Move the winner centroid towards the data point;
   5. Use the "neighborhood" function to identify neighbor centroids and move them towards the data point;
6. Decrease the size of the neighborhood and repeat the process until the neighborhood only includes the winner centroid.

By this way, the clustering starts out as a very general process and proceeds becoming more and more localized as the neighborhood decreases (Figure 8).

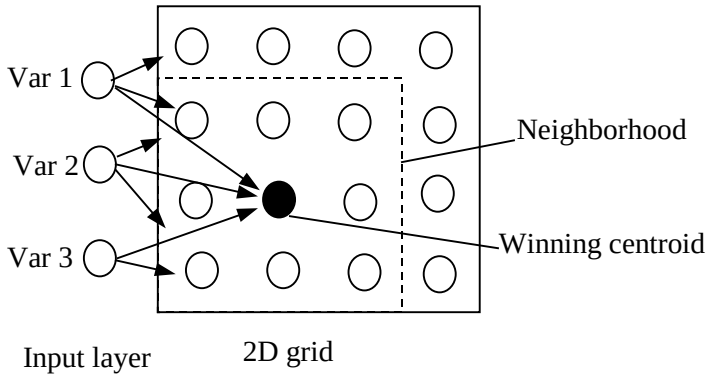


Figure 8: Example of Kohonen's Self-Organizing Feature Map. In this grid, each square corresponds to a cluster. Each customer point has its distance computed from 16 cluster points. The cluster centroid that is closest to this customer point is chosen to be the cluster center. Next all the other cluster centroids will move towards this chosen cluster center. This process is iterated until cluster centroids can hardly move.

3.3.3 Advantages/Disadvantages

SOM method is very powerful for visualizing multiple interactions between features thus offering better understanding of the data. SOM can learn from complex, multi-dimensional data and transform them into a map of fewer dimensions, such as a 2-dimensional plot. The 2-dimensional plot provides an easy-to-use graphical user interface to help the decision-maker visualize the similarities between consumer preference patterns. For example, suppose there are ten measures of customer attributes that are to be used to segment a market. It would be difficult to visually classify individuals based on all these attributes because the grouping must be done in a 10-dimensional space. By using the information contained in the 10-variable set by mapping the information into a 2-dimensional space, one can visually combine customers with similar attributes. These relationships can then be translated into an appropriate type of structure that genuinely represents the underlying relationships between market segments.

At the first approach, SOM is a technique that is quite difficult to handle and its outputs is not always easy to interpret.

3.4 Artificial neural networks

3.4.1 Definition

Artificial neural networks (ANN) are among the most complicated of the classification and regression algorithms. They are often considered as a black box. A neural network require a lot of data for training, thus consuming time, but once trained, it can make predictions for new cases very quickly, even in real time. Moreover, neural networks can provide multiple outputs representing multiple simultaneous predictions. A key feature of neural nets is that they only operate directly on numbers. As a result, any nonnumeric data in either the independent or dependent (output) columns must be converted to numbers, e.g. variables with "yes/no", "high/low" values must be replaced by "0/1".

As there exists an abundance of tutorial materials that can be found easily on the Web, see e.g. [WWW4], in the following we give a very summary presentation of ANN.

3.4.2 Terminology

Neural networks are defined by their **architecture**. The most common type of artificial neural network (ANN) consists of three *layers* of units: A layer of "**input**" units is connected to a layer of "**hidden**" units, which is connected to a layer of "**output**" units (see Figure 9).

The one depicted in Figure 9 is called **feed forward neural network**: A feed-forward NN allows signals to travel one way only; from input to output. There is no feedback (loops) i.e. the output of any layer does not affect that same layer. Feed-forward ANNs tend to be straight forward networks that associate inputs with outputs.

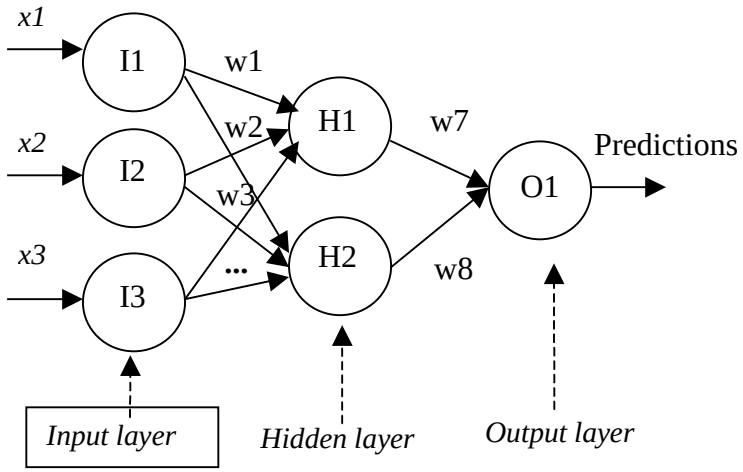


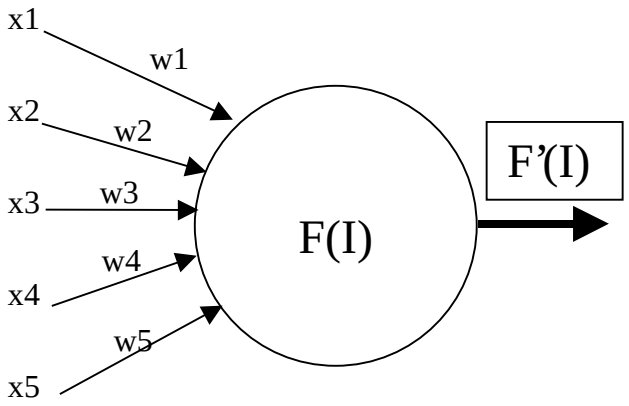
Figure 9: An example of a feed forward neural networks with six nodes.

The **weights** (w1, w2, w3, ... on Figure 9) of each arc linking one input to a node determine the degree of importance of that input.

- The activity of the **input units** represents the raw information that is fed into the network.
- The activity of each **hidden unit** is determined by the activities of the input units and the weights on the connections between the input and the hidden units (Figure 10).
- The behavior of the **output units** depends on the activity of the hidden units and the weights between the hidden and output units.

The **hidden layer** makes the network recognize more patterns, therefore the number of hidden nodes often increases with the number of inputs and the complexity of the problem. But too many hidden nodes can lead to *overfitting*, and too few hidden nodes can result in models with poor accuracy. Finding an appropriate number of hidden nodes is an important part of any data mining effort with neural nets. Several neural net products include search algorithms to find the optimum number of hidden nodes to use (e.g. using genetic algorithms). The weights between the input and hidden units determine when each hidden unit is active, and so by modifying these weights, a hidden unit can choose what it represents.

A neural network performs non-linear transformations of the inputs. Figure 10 shows what happens inside each node of the network.



F(Inputs) = F(I) = x1.w1+ x2.w2+ x3.w3+ x4.w4+x5.w5
F'(I) = Non-linear transformation of F(I) (*cf*. squash funct.)

Figure 10: What happens inside a node of a neural network.

Squashing function (Figure 11). This ensures that the values of the inputs to the next layer stay within a certain range, usually between 0 and 1. The function forbids "catastrophic" evolutions (due to feedback which increases values at each loop).

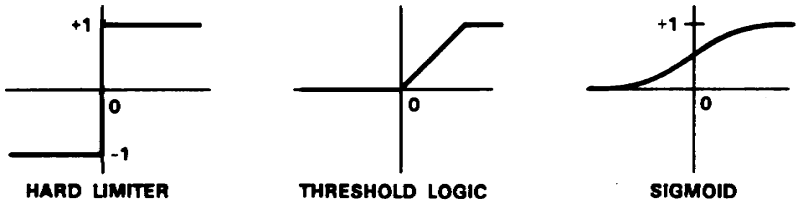


Figure 11: Example of some squashing functions.

Two most common architectures

Multilayer perceptrons (MLPs) and radial basis function (RBF) networks are the two most commonly-used types of feedforward network. The only fundamental difference is the way in which hidden units combine values coming from preceding layers in the network--MLPs use inner products, while RBFs use Euclidean distance. There are also differences in the customary methods for training MLPs and RBF networks, although most methods for training MLPs can also be applied to RBF networks. Furthermore, there are crucial differences between two broad types of RBF network--ordinary RBF networks and normalized RBF networks--that are ignored in most of the NN literature. These differences have important consequences for the generalization ability of the networks, especially when the number of inputs is large [WWW4].

3.4.3 Illustration

Suppose a bank has to predict risk related to loan applications made by customers. The bank has historical data about customers such as their age, marital status and income etc. and the final outcomes of the past loan customers. These data are used to build a model. Once the model is built, it is able to predict a new loan application. The attributes of the new case, which may contain empty fields, are put to the neural network to predict how risky it is to offer a loan to the customer (Figure 12).

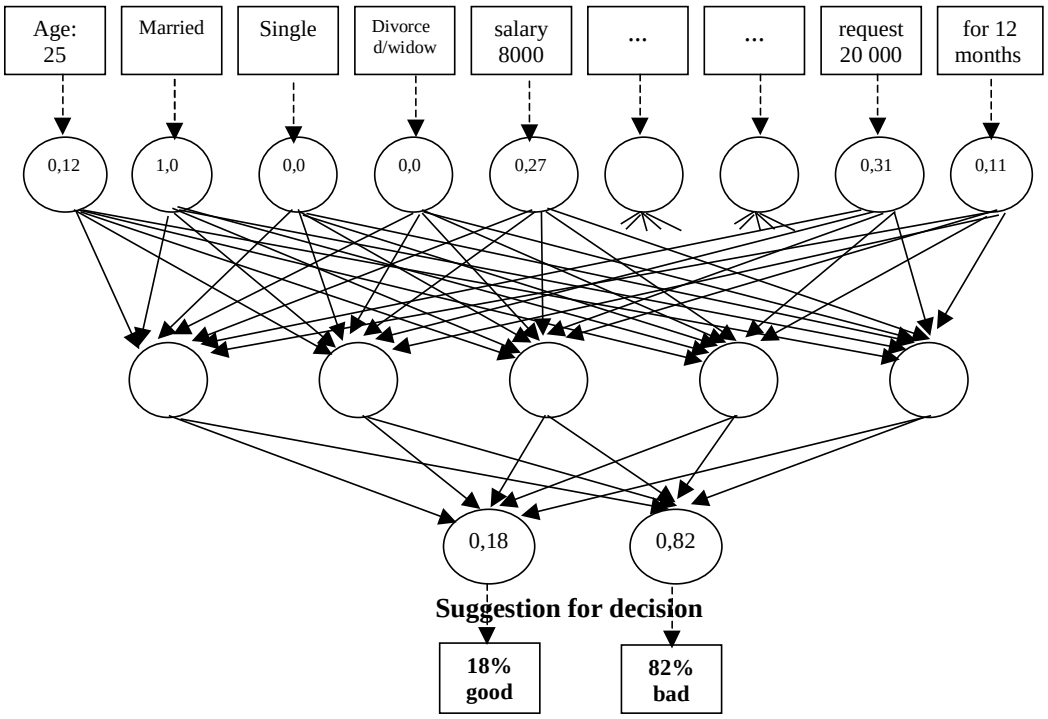


Figure 12: Example of neural network application.

3.4.4 Advantages/Disadvantages

Neural network is often considered as a black box as it is unable to explain the found relationships. It only works with numeric data, thus this means that non numerical data need to be converted. Moreover, the inputs need also to be normalized between 0 and 1.

Neural network is quick in predicting new cases if it is properly trained. The training phase is quite delicate, while one needs to choose appropriate number of data and control overfitting. In some package, this is done with help of other data mining tool (e.g. genetic algorithm). The drawback is that a neural network can never be exact (only accurate), even if it is trained for ever [BL97].

3.5 Decision trees

3.5.1 Definition and terminology

Decision tree learning is a method for approximating discrete-valued target functions, in which the learned function is represented by a decision tree. Decision trees perform many tests and then try to arrive at the best sequence for predicting the target. Each test creates branches that lead to more tests, until testing terminates in a *leaf* node (Figure 13). The path from the root to the target leaf is the *rule* that classifies the target. The rules are expressed in **if-then** form.

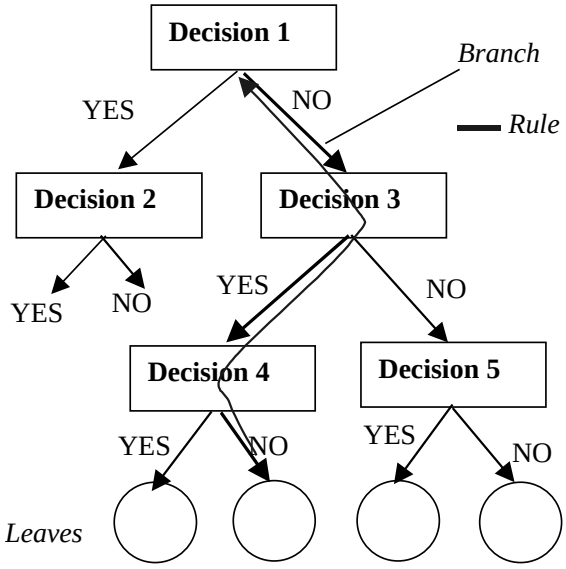


***Figure 13**: A decision tree grows from the root node, at each node the data is split to form new branches, until reaching a node that is not splittable any more (leaf node). Traversing the tree from the best leaf node to the root provides the rule that classifies the target variable.*

3.5.2 Tree induction

The training process that creates the decision tree is called **induction** and requires a small number of passes through the training set. Most decision tree algorithms go through two phases: a tree growing (**splitting**) phase followed by a **pruning** phase.

- **Splitting**: The tree growing phase is an iterative process which involves splitting the data into progressively smaller subsets. The first iteration considers the root node that contains all the data. Subsequent iterations work on derivative nodes that will contain subsets of the data. At each split, the variables are analyzed and the best split is chosen. One important characteristic of splitting is that it is *greedy* which means that the algorithm does not look forward in the tree to see if another decision would produce a better overall result.

- **Stopping criteria** : Tree-building algorithms usually have several stopping rules. These rules are usually based on several factors including maximum tree depth, minimum number of elements in a node considered for splitting, or its near equivalent, the minimum number of elements that must be in a new node. In most implementations the user can alter the parameters associated with these rules. Some algorithms, in fact, begin by building trees to their maximum depth. While such a tree can precisely predict all the instances in the training set (except conflicting records), the problem with such a tree is that, more than likely, it has *overfit* the data.

- **Pruning** : After a tree is grown, one can explore the model to find out nodes or subtrees that are undesirable because of overfitting, or rules that are judged inappropriate. Pruning removes splits and the subtrees created by them. Pruning is a common technique used to make a tree more general. Algorithms that build trees to

maximum depth will automatically invoke pruning. In some products users also have the ability to prune the tree interactively.

3.5.3 Understanding the output

Once trained, a tree can predict a new data instance by starting at the top of the tree and following a path down the branches until encountering a leaf node. The path is determined by imposing the split rules on the values of the independent variables in the new instance. A decision tree can help a decision maker identify which factors to consider and how each factor has historically been associated with different outcomes of the decision.

Decision trees have obvious value as both predictive and descriptive models. Prediction can be done on a case-by-case basis by navigating the tree. More often, prediction is accomplished by processing multiple new cases through the tree or rule set automatically and generating an output file with the predicted value or class appended to the record for each case. Many implementations offer the option of exporting the rules to be used externally or embedded in other applications.

3.5.4 Different decision algorithms

Decision tree algorithms commonly implemented include Chi-squared Automatic Interaction Detection (**CHAID**), Classification and Regression Trees (**CART**), **C4.5** and **C5.0**. All are well suited for **classification**; some are also adaptable for **regression**. The distinguishing features between tree algorithms include:

- **Target variables:** Most tree algorithms require the target (dependent) variable be categorical. Such algorithms require that continuous variables are *binned* (grouped) for use with regression.
- **Splits**: Many algorithms support only binary splits, that is, each parent node can be split into at most two child nodes. Others generate more than two splits and produce a branch for each value of a categorical variable.
- **Split measures**: help select which variable to use to split at a particular node. Common split measures include criteria based on gain, gain ratio, GINI, chi-squared, and entropy.
- **Rule generation**: Algorithms such as **C4.5** and **C5.0** include methods to generalize rules associated with a tree; this removes redundancies. Other algorithms simply accumulate all the tests between the root node and the leaf node to produce the rules.

ALGORITHM	CHARACTERISTICS
CART	Binary split based on GINI (recursive partitioning motivated by statistical prediction), exactly two branches exist from each non-terminal node. Pruning based on measure of complexity of the tree. Support classification and regression. Handles continuous target variables. Requires data preparation.
C4.5 and C5.0 (Enhanced versions of ID3)	Produce tree with multiple branches per node. The number of branches is equal to the number of categories of predictor. Combine multiple decision trees into a single classifier. Use information gain for splitting. Pruning based on error rate at each leaf.
CHAID	Multi-way splits using chi-square tests (detection of complex statistical relationships). The number of branches varies from two to the number of predictor categories.
SLIQ	Fast scalable classifier. Fast tree pruning algorithm.
SPRINT	For large dataset. Splitting based on the value of a single attribute. Removes all memory restrictions by using attribute list data structure.

Table 1: Difference between decision tree algorithms.

3.5.5 Illustration

The example is taken from [WWW7] about a company which has to consider whether to tender or not for two contracts (MS1 and MS2) on offer from a government department for the supply of certain components. The company has three options:

1. tender for contract MS1 only (cost £50,000); or
2. tender for contract MS2 only (cost £14,000); or
3. tender for both contracts MS1 and MS2 (cost £55,000).

If the tender is successful, the component supply cost would be respectively:

1. £18,000 for MS1 only
2. £12,000 for MS2 only
3. £24,000.

In addition, subjective assessments have been made of the probability of getting the contract with a particular tender price as shown in Table 2. Note here that the company can only submit one tender and cannot, for example, submit two tenders (at different prices) for the same contract.

Option	Possible tender prices (£)	Probability of getting contract
MS1 only	130,000	0.20
	115,000	0.85
MS2 only	70,000	0.15
	65,000	0.80
	60,000	0.95
MS1 and MS2	190,000	0.05
	140,000	0.65

Table 2: Related costs and probabilities of getting contracts.

In the event that the company tenders for both MS1 and MS2 it will either win both contracts (at the price shown above) or no contract at all. The risk is that f a tender is unsuccessful the company will have made a loss.

The decision tree for this example is shown in Figure 14.

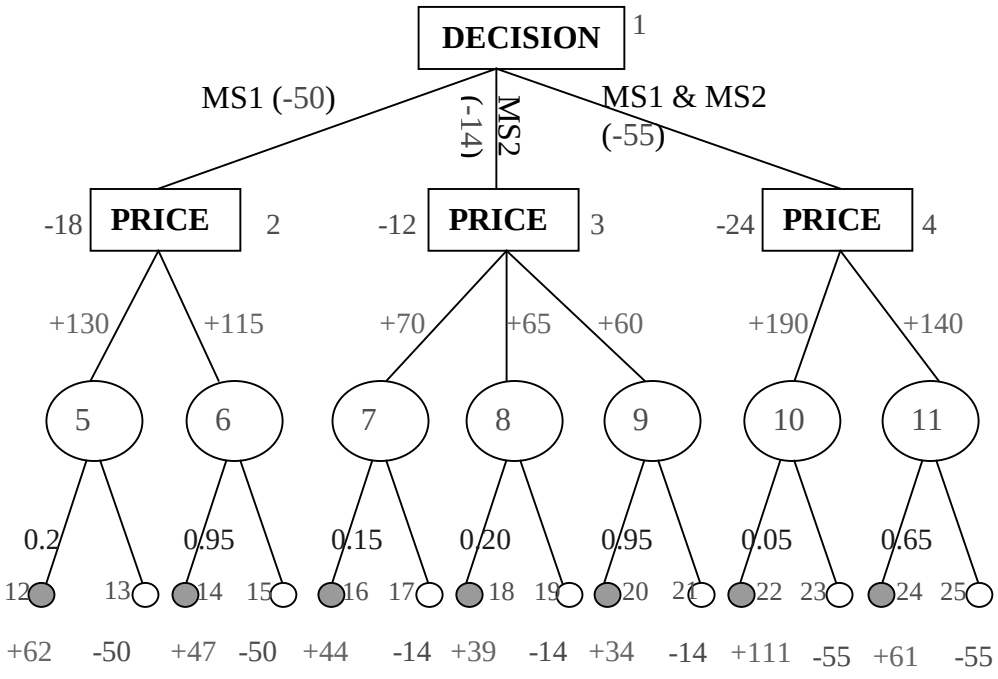


Figure 14: Decision tree for tendering for contracts. Each filled leaf node is a win [WWW7].

Path to winning nodes are for example:

- Path to terminal node 12: tender for MS1 only (cost 50), at a price of 130 , and the contract is successful so the incurring component supply costs are 18. Total profit 130-50-18 = 62.
- Path to terminal node 16: tender for MS2 only (cost 14), at a price of 70, and the contract is successful, so the incurring component supply costs are 12. Total profit 70-14-12 = 44.
- Path to terminal node 22: tender for MS1 and MS2 (cost 55), at a price of 190, and the contract is successful, so the incurring component supply costs are 24. Total profit 190-55- 24=111.

Using probabilities to make decisions at node 2 :

- Node 5: 0.2×(62) + 0.8× (-50)= -27.6
- Node 6: 0.85×(47) + 0.15×(-50) = 32.45

Thus, the best decision at node 2 is to tender at price £115.

Likewise, we can find out that the best decision at node 3 is to tender at a price 60 and at node 4 at a price of 140.

3.5.6 Advantages/Disadvantages

Decision trees have unique advantages. They produce models that are easy to understand and they are unaffected by missing values in data.

Decision trees impose certain restrictions on the data that is analyzed. First, decision trees permit only a single dependent variable. In order to predict more than one dependent variable, each variable requires a separate model. Also, most decision tree algorithms require that continuous data are grouped or converted to categorical data.

3.6 Association rules discovery

3.6.1 Definition and terminology

Association and sequencing tools analyze data to **discover rules that identify patterns of behavior**, e.g. what products or services customers tend to purchase at the same time, or later on as follow-up purchases. The process of using an association or sequencing algorithm to find such kinds of rules is frequently called **market basket analysis.** Detailed description of this method can be found in the book by M. Berry and G. Linoff [BL97].

Example of rule:

When people buy a tourism book *they also buy* ***a pocket dictionary*** *20% of the time.*

Each rule has two measures, called *confidence* and *support*.

1. **Support** (*prevalence*) indicates the frequency of a pattern, i.e. how often items occur together. In the example above, the confidence is 20%. Rules with a low value for support might simply be due to a statistical anomaly. A minimum support is

necessary if an association is going to be of some business value.
If **X** and **Y** then **Z** with support **s**.
The rule holds in *s*% of all transactions.
Support is computed as follows: $s(\mathbf{A} \Rightarrow \mathbf{B}) = P(\mathbf{A} \cup \mathbf{B})$.

2. **Confidence** (*predictability*) denotes the strength of an association, i.e. how much a specific item is dependent on another.
   If **X** and **Y** then **Z** with confidence *c*.
   If X and Y are in the basket, then Z is also in the basket in *c*% of the cases.
   $c(\mathbf{A} \Rightarrow \mathbf{B}) = P(\mathbf{B}|\mathbf{A}) = P(\mathbf{A} \cup \mathbf{B})/P(\mathbf{A})$.

Suppose **A** and **B** appear together in only 1% of the transactions but whenever **A** appears there is 80% chance that **B** also appears. The 1% presence of **A** and **B** together is the support and 80% is the confidence of the rule.

In the absence of any knowledge about what else was bought, the following assertions can be also made from the available data:

People buy gardening book 4% of the time.

People buy dictionary 6% of the time.

These numbers – 4% and 6% -- are called the **expected confidence** of buying gardening book or dictionary, regardless of what else is purchased.

Lift measures the difference between the confidence of a rule and the expected confidence. The difference can be computed either by subtracting the two values or, more commonly, by putting them a ratio. *Lift* is one measure of the strength of an effect. The negative lift (or a lift ratio of less than 1) suggest that it's less likely that people would buy the two products at the same time.

3.6.2 Data format

The data used by an association algorithm is made up of **entities** and **attributes**. E.g. The entity might be a market basket, and the attributes all the items purchased at one time.

The data needs to be in one of two formats called *horizontal* and *vertical*.

- In the **horizontal format** there is one row for each entity, and there are columns for each attribute, e.g. one row for each market basket, with columns for each product (Table 3). The problem is that the number of columns can become quite large (number of products might exceed 100,000), and similar products need to be grouped together to reduce the number of columns to a reasonable quantity.

Trans. ID	Item A	Item B	Item C	Item D	Item E
132	Y				Y
428		Y	Y	Y	

Table 3: Example of horizontal data format in association rules.

- In the **vertical format** uses multiple rows to store an entity, using one row for each attribute (Table 4). The rows for a particular entity are tied together with a common

ID. This kind of representation is more normalized in the relational sense, and it works much better when an entity can have great variability in terms of the number of attributes.

Trans. ID	Product
132	Item A
428	Item B
428	Item C
428	Item D
132	Item E

Table 4: Example of vertical data format in association rules.

Some DM product support a pivoting operation that converts a horizontal format to a vertical format.

Association algorithms **can only operate on categorical data**. If non-categorical attributes should be used, such as income, the non-categorical data must be *binned* into ranges (for example, 0 to 20,000; 20,001 to 40,000; 40,001 to 70,000; and greater than 70,001), *turning each range into an attribute*.

Another common characteristic of association rule generators is an **item hierarchy**. **Item hierarchy** can be used to reduce the number of combinations to a manageable size by grouping similar items together. Using an item hierarchy reduces the number of combinations and also helps to find more general, and probably more useful, higher-level relationships such as those between any kind of item *X* and any kind of item *Y*. Unfortunately, even if an item hierarchy is used to group items together, for a real application, combinatorial explosion will always occur. But in practice, many of the combinations will never occur. Nevertheless, some sort of dynamic memory or counter allocation and addressing scheme will be needed.

3.6.3 Algorithm

To discover associations, we assume that we have a set of transactions, each transaction being a list of items (e.g. list of books). A user might be interested in finding all associations which have *s*% support with *c*% confidence such that :

- all associations satisfying user constraints are found,
- associations are found efficiently from large databases.

To find such associations, the following algorithm is given. This algorithm is also known as "*APrioriAll algorithm*" from IBM's group [PZOD99, WWW8]: here are the steps to follow :

1. Discover all frequent items that have support higher than the minimum support required;
2. Use the set of frequent items to generate the association rules that have high enough confidence level;
3. Scan all transactions and find all items that have transaction support above *s*%. Let these be L_1;

4. Build item pairs from L_1. This is the candidate set C_2;
5. Scan all transactions and find all frequent pairs in C_2. Let this be L_2;

The general rule for steps 4 and 5 is:

1. Build sets of *k* items from L_{k-1}. This is set C_k.
2. Scan all transactions and find all frequent sets in C_k. Let this be L_k.

Step 1 is computationally expensive because it has to find all the possible associations.

3.6.4 Illustration

Consider an example [WWW8] with the following set of transactions:

Trans. ID	Product
001	B, M, T, Y
002	B, M
003	T, S, P
004	A, B, C, D
005	A, B
006	T, Y, E
007	A, B, M

Assume that we wish to find associations with at least 30% support and 60% confidence.

The list of frequent items is now computed. Only the following three items are qualified as frequent since they appear in more than 30% of the transactions. This is set L_1.

Item	Frequency
A	3
B	5
M	3

These three items form three pairs {A, B}, {B, M}, and {A, M}. This set is C_2. Now we find the frequency of these pairs, which is :

Pair	Frequency
{A, B}	3
{B, M}	3
{A, M}	1

The first two pairs have more than 30% support. Their confidence level is:

$c(\mathbf{A} \Rightarrow \mathbf{B}) = 100\%$

$c(\mathbf{B} \Rightarrow \mathbf{A}) = 60\%$

$c(\mathbf{B} \Rightarrow \mathbf{M}) = 60\%$

$c(\mathbf{M} \Rightarrow \mathbf{B}) = 100\%$

All are therefore acceptable.

The frequent item pairs (that is L_2) are:

Pair	Frequency
{A, B}	3
{B, M}	3

These pairs are now used to generate a set of three items (i.e. C_3). In this example only one such set is possible which is {A, B, M}. The frequency of this set is only 1 which is below 30% support and therefore this set of three items is not qualified.

The algorithm to construct the candidate set for large item sets is crucial to the performance of the algorithm. It is the generation of the large 2-item sets that is the key to improving the performance of the algorithm.

3.7 Sequential patterns discovery

3.7.1 Definition

Sequential patterns discovery (or sequencing for short) extends association by **adding time comparisons between transactions**. This means that in sequencing, not only the coexistence of items within a transaction may be important, but **also the order in which those items appear across ordered transactions and the amount of time between transactions**. To handle this, each time series is converted into a multi-item transaction, duplicate items are removed and then association rule discovery can be applied to the transactions.

The input data is a set of *sequences*, called *customer-sequences*. Each data sequence is a ordered list of transactions(or itemsets), where each transaction is a sets of items (literals). Typically there is a transaction-time associated with each transaction.

A customer *supports* a sequence if that sequence is contained in the customer-sequence. Given a database of customer transactions, the problem of mining sequential patterns is to find all sequential patterns that have a certain user-specified minimum support.

The length of a sequence is the number of itemsets in the sequence. A sequence of length *k* is called a *k*-sequence.

3.7.2 Time series

Analyzing data containing time related data provides patterns such that the presence of a set of items is followed by another item in a time-ordered set of *episodes* [SCDT00, MTV97]. By analyzing the sequence, one can find frequent episodes, i.e., collections of

events occurring frequently together (see Figure 15). Having such information, marketers can predict purchase patterns which can be helpful in advertising targeting to certain customers.

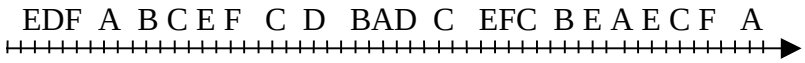


Figure 15: Example of sequence of events happening within a time window. A, B, C, D etc. are event types (i.e. user's action). Here we can observe that whenever A and B occur, in either order, C occurs soon [MTV97].

Because time differences are continuous rather than categorical values, the continuous values must be grouped into categorical values based on the time-related objectives set by the user. Examples of time-related objectives are "*within three months*," "*next visit*," or a set of mutually exclusive ranges such as next day, next week, next month, and next year. Some products only support "*at any later time*." In any case, an **appropriate time-related objective** can (and should) be used to reduce the number of combinations. And, as we saw above, the association algorithm is already laced with combinatorial issues; the time-related pairing just adds another layer of combinations!

3.7.3 Algorithm

The algorithm for sequential pattern discovery works like in association rule discovery. This algorithm is also known as IBM's sequential pattern mining algorithm [PZOD99, WWW9]:

1. **Sort phase**. Sort the database on customer-id and transaction-id.
2. **Itemset phase**. Find all large sequences of length 1.
3. **Transformation phase**. Map each transaction to the set of all large itemset contained in the transaction. Here, each large itemset is mapped to an integer. Sequential patterns are now represented by a list of litterals, rather than by a list of sets of items (see Table 5).
4. **Sequence phase**. Find all large sequences
5. **Maximal phase**. Delete all non-maximal sequences.

Customer Id	**Original customer sequence**	**Transformed customer sequence**	**After mapping**
1	<(Shirt Tie) (Shoes) (Socks Coat Hat)>	<{(Shoes)} {(Sock), (Hat), (Socks Hat)}>	<{1} {2, 3, 4}>
2	<(Shoes) (Belt)>	<{(Shoes)} {(Belt)}>	<{1} {5}>

Table 5: Database transformation with mapping items to integers.

3.7.4 Illustration 1

Sequential patterns example from [WWW8]. Consider the database shown in the table hereafter (Table 6). Table 7 shows the same database sorted by customer and date. From this table, we can extract a set of customer sequences shown in Table 8.

Transaction Time	Customer	Items Bought
June 20, 1994 10:13 am	Brown	Milk, Eggs
June 20, 1994 11:02 am	Zappa	Chocolate
June 20, 1994 11:47 am	Brown	Juice
June 20, 1994 2:32 pm	Moore	Juice
June 21, 1994 9:22 am	Brown	Chips, Cider, Peanuts
June 21, 1994 3:19 pm	Mitchell	Juice, Beer, Peanuts
June 21, 1994 5:27 pm	Adams	Juice
June 21, 1994 6:17 pm	Moore	Chips, Peanuts
June 22, 1994 10:34 am	Adams	Chocolate
June 22, 1994 5:03 pm	Moore	Chocolate

Table 6: Original database sorted by transaction time and customer Id.

Adams	June 21, 1994 5:27 pm	Juice
Adams	June 22, 1994 10:34 am	Chocolate
Brown	June 20, 1994 10:13 am	Milk, Eggs
Brown	June 20, 1994 11:47 am	Juice
Brown	June 21, 1994 9:22 am	Chips, Cider, Peanuts
Mitchell	June 21, 1994 3:19 pm	Juice, Beer, Peanuts
Moore	June 20, 1994 2:32 pm	Juice
Moore	June 21, 1994 6:17 pm	Chips, Peanuts
Moore	June 22, 1994 5:03 pm	Chocolate
Zappa	June 20, 1994 11:02 am	Chocolate

Table 7 : Database sorted by customer and date.

Customer	Customer Sequence
Adams	(Juice) (Chocolate)
Brown	(Milk, Eggs) (Juice) (Chips, Cider, Peanuts)
Mitchell	(Juice, Beer, Peanuts)
Moore	(Juice) (Chips, Peanuts) (Chocolate)
Zappa	(Chocolate)

Table 8 : Customer-sequence database.

With a minimal support set to 40%, we find two sequences that are maximal supported by three customers (see Table 9). The sequential pattern <((Juice) (Chocolate)> is supported by Adams and Moore. Moore bought items (Chips, Peanuts) in between Juice and Chocolate but support the pattern <((Juice) (Chocolate)> since we are looking for pattern that are not necessarily contiguous. The pattern <(Juice) (Chips, Peanuts)> is supported by Brown and Moore. Brown has bought (Milk, Eggs) and Cider along with (Juice) and (Chips, Peanuts) but support this pattern since {(Juice) (Chips, Peanuts)} is subset of {(Milk, B) (Juice) (Chips, Cider, Peanuts)}. Other sequences do not have minimum support required.

Sequential Patterns with Support > 40%	**Customers Supporting it**
(Juice) (Chocolate)	Adams, Moore
(Juice) (Chips, Peanuts)	Brown, Moore

Table 9: Sequential patterns in the database.

From the sequential patterns, we can extract sequential rules (Table 10).

Sequential Rule with Support>40%	**Customers Supporting the Rule Body**	**Confidence Value**
(Juice) => (Chocolate)	Adams, Moore	50%
(Juice) => (Chips, Peanuts)	Brown, Moore	50%

Table 10: Derived sequential rules.

3.7.5 Illustration 2

This example is taken in [Joshi97] with a database of the customer-sequences shown below in Table 1. Here, the customer sequences are in transformed form where each transaction has been replaced by the set of itemsets contained in the transaction and the itemsets have been mapped to integers. The minimum support has been specified to be 40% (i.e. 2 customer sequences).

The first pass over the database made in the itemset phase, has produced sequences of length 1 (contain one item) shown in Table 16.

The large sequences together with their support at the end of the second, third, and fourth passes are also shown in the following tables. No candidate is generated for the fifth pass because there is no sequence that satisfies the minimum support.

Customer Id	Customer sequences
1	<{1 5 2 3 4 }>
2	<{1 3 4 3 5 }>
3	<{1} {2} {3} {4}>
4	<{1} {3} {5} >
5	<{4} {5}>

Table 11: Customer sequences after mapping.

1-Sequence	Support
{1}	4
{2}	2
{3}	4
{4}	4
{5}	4

Table 12: Sequences of length 1 (or 1-sequences) found in the list in Table 11.

2-Sequence	Support
{1 2}	2
{1 3}	4
{1 4}	3
{1 5}	3
{2 3}	2
{2 4}	2
{3 4}	3
{3 5}	2
{4 5}	2

Table 13: Sequences of length 2 (or 2-sequences)after the second pass.

3-Sequence	Support
{1 2 3}	2
{1 2 4}	2
{1 3 4}	3
{1 3 5}	2
{2 3 4}	2

Table 14: Sequences of length 3 (or 3-sequences) after the third pass.

4-Sequence	Support
{1 2 3 4}	2

Table 15: Sequences of length 4 (or 4-sequences) the fourth.

The maximal large sequences would be the three sequences {1 2 3 4 }, {1 3 5} and {4 5}.

3.7.6 Advantages/Disadvantages

Interesting association rules that capture relationships between bought items can be used, to identify a typical set of precursor purchases that might predict the subsequent purchase of a specific item. For example, a business manager can use the analysis to plan:

- *Couponing and discounting* e.g. to offer simultaneous discounts on products that are usually bought together. Instead, discount one to pull in sales of the other.
- *Product placement*. Place products that have a strong purchasing relationship close together to take advantage of the natural correlation between the products. In e-commerce, if a customer puts product X in the shopping cart, pop up an offer of Y product.
- *Timing and cross-marketing*. This can be useful for marketing a new product on the right time based on the rules found by sequential analysis.

The problem that the algorithm has to face is the combinatorial explosion. The more the number of items increase, the more possibilities to pair. And in practical problem, there are hundreds of thousands of different items and millions of transactions, this means many gigabytes of data. Moreover, many of the sought associations and found relations are not interesting and may never occur in reality.

3.8 Other data mining methods

Additional approaches used in conjunction with these and other analytical techniques include, **genetic algorithms, fuzzy logic, Bayesian belief networks** and **discriminant analysis**.

- **Genetic algorithms** (GA) are optimization methods based on the concept of evolutionary genetics where the fittest individuals survive and evolve through crossovers and mutations. GA are mostly used in optimization problems and very little in data mining. Usually they are incorporated into neural network packages to increase the performance of the latter. In this case GA work without much intervention of the user. If GA must be used alone, they require a clever coding of the problem into chromosomes and a meaningful fitness function. GA are included in few packages and offered by small amount of vendors.
- **Fuzzy logic** can rank results based on closeness to the desired result. The approach is used for clustering and classification. The technique is useful in Web mining because clusters and associations do not have crisp boundaries. The overlapping are best described by fuzzy sets [NFJK99]. Small number of applications and vendors.
- **Bayesian belief networks (BBN)** is a model for representing uncertainty in knowledge in a certain domain. To handle uncertainty, probability theory is used to explicitly represent the conditional dependencies between the different knowledge components. This provides an intuitive graphical visualization of the knowledge including the interactions among the various sources of uncertainty. BBN can be used for clustering customer (or web users).
- **Discriminant analysis** finds hyper-planes that separate the classes. The resultant model is very easy to interpret because all the user has to do is determine on which side of the line (or hyper-plane) a point falls. Training is simple and scalable [CC98].

3.9 Which DM technology to use?

We have seen that there are many DM techniques that can extract information from data and perform clustering, association or prediction. Before choosing a technology, one has to first consider which problem is to be addressed and which results to obtain. Secondly, how much data need to be processed and from which predictions are needed.

Summary of most characteristic features of the most popular DM technologies are presented in Table 16 and in [Hall99].

TECHNOLOGY	ADVANTAGES	LIMITATIONS	WHEN TO USE IT
Rule-based analysis (association, sequencing, market basket analysis)	Easy to understand. Good for data that is "complete" with data relationships that can be modeled with *if ... then* rules. Rules are readable. Handle continuous and categorical data.	Data may not have strong rules-based relationships. Combinatorial explosion because it seeks for every possible relationship between all fields in a database.	For well defined items that can group together in an interesting way. For retail industries, and time-series problems.
Decision trees	Among the easiest to understand. Tend to excel when a particular target attribute value is based on a complex, set of attributes with particular values. Generate also rules which are mutually exclusive.	Less appropriate for estimation tasks where the goal is to predict the value of a continuous variable. Not for time-series data unless represented in a way that trends and sequential patterns are made visible.	For classification of records or predictions of outcomes. To generate rules to be exported in other tools or translated in natural language.
Neural networks (except SOM)	Versatile. Good results even with complicated (non-linear, noisy) data. Can process large amount of data. Predictions with continuous data, classification and clustering with discrete data. Can work well with noisy data and data with missing some values.	Inability to explain the found relationships. Work with numeric data, otherwise conversion is needed. Inputs need to be in [0,1]. May converge too early. If too many input features, NN cannot find patterns and can never converge.	Good tool for predictions and classification. When there are lots of input features, select only important features for the training phase by using other methods (e.g. associations).
Cluster detection (K-means, SOM...)	Easy to apply. Good start of a data mining process. No prior knowledge of the internal structure of a database is needed. Works well with categorical, numeric and textual data.	Requires a lot of memory. Difficult to find right distance measures and weights. K-means is not good for clustering closely matching records. May be hard to interpret resulting clusters (SOM).	When having a large complex dataset with many variables and lots of internal structures. Often used to find outliers or records that don't fit the predictive model.

Table 16: Feature summary of data mining technologies.

As data mining is a process of knowledge discovery, even when tackling only one problem, it might be advantageous to have multiple algorithms at hand. One may need to compare a decision tree model to a neural net model, increasing confidence in the results when predictions from the two models are identical, and appropriately raising a flag when the two models disagree. In addition, one may need different techniques in different phases of the data mining process.

Some packages may propose a combination of DM techniques. In this case the software will choose the best technology for the problem or can compare the results of the different technologies. Users do not require to learn several tools. But this may not provide best-in-class techniques for each technology.

4 Web mining

"*Today, 30 million users are active on the Net. Each month, 1 million to 2 million new people become Internet explorers. Over 40,000 public and private computer networks are now linked to the Net with business and information services. Citizens from 140 nations are using the Net*" [WWW10]. Internet offers to nowadays' companies huge business opportunities. The traditional marketing strategies and techniques need to be revisited in this context. This has promoted the rapid development of Web mining. Web mining is the process of discovering and analyzing information from the World Wide Web [CMS97]. Web mining is divided into *Web Content Mining* and *Web Usage Mining*. *Web Content Mining* deals with discovering and organizing Web-based information (e.g. electronic library) and *Web usage mining* addresses the problem of analyzing the behavioral patterns from the data collected about the Internet users. Thus, Web usage mining is more relevant for customer profiling. In the following when we refer to Web mining it is the same as Web usage mining.

The state of the art of Web mining is still primitive. Most of publications date from the second half of the 90s. According to our knowledge, the book "*Data Mining Your Website*" by J. Mena [Mena99] is the first one published in the domain.

4.1 Internet marketing

The major challenges for most of e-commerce traders concern the transition from vast amounts of Internet server and transaction data to actionable marketing. Because the online channel provides rapid availability of new information about customer, data mining systems must be able to discover new behavioral and attitudinal differences among customers by allowing newly observed behavioral data to immediately participate in the existing data pool [CDH01]. For e-commerce traders, Web mining techniques are becoming essential for maintaining "e-CRM" strategies: These techniques discover hidden pattern and relationships within Web data for the three main marketing actions [MJHS96, SZAS97, BM98, SCDT00]:

1. Discovering association rules for *customer attraction,*
2. discovering sequential patterns for *customer retention* and
3. discovering classification rules and data clusters for *cross-selling*.

4.1.1 Customer attraction with association

The two essential parts of attraction are the *selection of new prospective customers* and the *acquisition of the selected potential candidates*. One marketing strategy to perform this exercise, among others, is to find common characteristics in already existing visitors' information and behavior for the classes of profitable and non-profitable customers. These groups (e.g. "*no customer"*, for browsers who have logged in, but did not purchase, "*visitor once"* and "*visitor regular*") are then used as labels for a classifier to discover Internet marketing rules, which are applied online on new site visitors.

Depending on the outcome, a dynamically created page is displayed, with contents depending on found **associations** between browser information and offered products/services.

4.1.2 Customer retention with sequential patterns

Customer retention is the step of attempting to keep the online shopper as loyal as possible. Due to the non-existence of distances between providers, this is an extremely challenging task in electronic commerce scenarios. The strategy is similar to that of acquisition, which is dynamically creating web offers based on associations but by considering associations **across time**, also known as **sequential patterns**.

The discovered sequence can then be used **to dynamically display special offers** after a certain page sequence has been visited.

4.1.3 Cross-sales and attribute-orientated induction

The objective of cross-sales is to horizontally and/or vertically diversify selling activities to an existing customer base. In order to discover potential customers, characteristic rules of existing cross-sellers has to be discovered, which is performed through the application of **attribute-oriented induction** [HCC92]. In attribute-oriented induction, a simple Web page (leaf page in the web page hierarchy) is replaced by its corresponding general page based on the page hierarchy. Duplicate pages are removed with their times added together.

The entire set of discovered interesting rules can then be used as the model to be applied at run-time on incoming actions and requests from existing customers.

4.2 Web data collection

Internet offers a variety of tools to collect information about Internet users and potential or actual customers. *Cookies*, *server log files*, in addition to *online questionnaire* and information acquired through *transactions* can be used to define customer profiles. Data that can be collected are explicated in [Kock00]. They are:

- **Http-protocols** contain information about the users' browser, browser version and operating system.
- **Cookies** contain preferences and information about the Internet user (e.g. preferred language, content or interests, age, gender, income etc.). A cookie is a file containing information about what an Internet user does on a web site. Thus it is the most effective way to identify Internet users.
- **Server log files** contain all Internet activities of a specific Internet user. The log files contain the IP address of the Internet user, date and time of the web page requests, the elements of the web page that were sent to the Internet user and the currently opened web page.
- **Query data** to a web server are for example **on-line search** for products, or information. The logged query data must be linked to the access log through cookie data and/or registration information.
- Other data that can be collected are: number of *hits*, *page views*, *visits*, *ad clicks*, *view time* and *ad view time*.
  - The number of **hits** is related to how often web site elements are requested on the server. As a single web page contains a variety of elements (graphics, links, pictures etc.) and as every element is counted as a hit, the absolute number of hits may not be

meaningful in measuring the effectiveness of a web page for placing marketing tools e.g. placing advertisements.

- **Page view** is the number of requests of a whole web page independent of the number of elements on that page. Note that a page with frame is made of several single web pages put together.
- A **visit** refers to a server request coming from outside to the web site. Visits can be determined through the different request addresses written in the log file.
- **Ad clicks** count the number of the Internet users that visit the advertising company's web site by clicking on the advertisement object. Click through rate correspond to response rate in traditional marketing.
- **View time** measure the time spent by the Internet user on a web site (measured by a JavaScript code placed on the web page). **Ad view time** measure how long and advertisement is in the viewable part of the browser window, thus can be seen by the Internet user.
- **Error logs** store data of failed requests, such as missing links, authentication failures, or timeout problems. Apart from detecting erroneous links or server capacity problems, error logs are not so useful for actionable marketing.

While most of the data are generated by the way how Internet works (e.g. access log files), or by analyzing the generated data (e.g. number of hits), some other are acquired using some specifically designed tools, e.g. view time, and recently **Web bugs** [Weise00].

- A **Web bug** [WWW11] is like a tiny "spying agent" represented by a graphic of size 1-by-1 pixel hidden on a Web page or in an Email message. It is designed to monitor who is reading the Web page or Email message. It collects the IP address of the Internet user's computer, the type of browser s/he has, the URL of the visited page on which the Web bug is located, the time the Web bug was viewed and a previously set cookie value. Ad networks can use Web bugs to add information to a personal profile of what sites a person is visiting. The personal profile is identified by the browser cookie of an ad network. At some later time, this personal profile which is stored in a data base server belonging to the ad network, determines what banner ad is shown. Another use of Web bugs is to provide an independent accounting of how many people have visited a particular Web site. Web bugs are also used to gather statistics about Web browser usage at different places on the Internet. Web bugs are also known as "1-by-1 GIFs", "invisible GIFs", and "beacon GIFs."

4.3 Web data processing

Before any knowledge discovery takes place from the Web data, the data goes through a preprocessing phase to clean the data from irrelevant or redundant entries. Then the data is formatted appropriately according to the application (Association Rules and Sequential Patterns require the input data to be in different forms).

A Web server access log contains a complete history of file accesses by clients. Each log entry consists of :

User's IP address,

Access time,

Request method ("GET", "POST", etc),

URL of the page accessed,

Data transmission protocol (typically HTTP/1.0),

Return code,

Number of bytes transmitted.

The data that may be discarded include a variety of image, sound, and video files; executable *cgi* files; coordinates of clickable regions in image map files; and HTML files.

After the log entries are cleaned, the fields that are left are generally: *IP address*, *user id*, *access time* and the *URL*. In e-commerce, typical fields may contain: (*CustomerKey*, *ProductKey*, *LocationKey*, *DateKey*, *SessionKey*) as well as some statistical summarization information (*Quantity*, *TotalPrice*, *ClickThroughRate*) [BM98].

After that, the log data is converted into a form suitable for a specific data mining task. This transformation is accomplished according to the transaction model for that particular task (e.g., association rule or sequential pattern discovery).

Another preprocessing tasks is discrepancy resolution. The most typical discrepancies encountered are representation mismatches of marketing information (for instance customer numbers) and log files (cookie identifiers as well as the identification of a specific customer through standard log file information). Most inconsistencies are resolved through mapping tables and conversion functions.

In order to reduce the data dimension, log data can be segmented into *user sessions* [FSS99]. "*A user session is defined as a sequence of temporally compact accesses by a user*".

4.4 Discovering association rules

In the context of Web mining, discovering association rules turns to be discovering the *correlations among accesses to various files available on the server by a given client*.

For example, using association rule discovery techniques we can find correlations such as the following:

60% of clients who accessed the page with `URL /company/products/`, *also accessed the page* `/company/products/product1.html`; or

40% of clients who accessed `/company/products/product1.html`, *also accessed* `/company/products/product2.html`; or

30% of clients who accessed `/company/announcements/special-offer.html`, *placed an online order in* `/company/products/product1`.

Since usually such transaction-based databases contain extremely large amounts of data, current association rule discovery techniques try to prune the search space according to support for items under consideration. In Web mining, the hierarchical organization of the files can be also used for pruning [FSS99].

For example, if the support for `/company/products/` is low, one may conclude that the search for association between the two secondary pages with URLs

/company/products/product1 and /company1/products/product2 should be pruned since neither are likely to have adequate support.

In e-commerce, discovery of association rules can help in the development of effective marketing strategies as well as an indication of how to best organize the organization's Web space.

For example, if 80% of the clients accessing /company/Page1.html and /company/Page2.html also accessed /company/file1.html, but only 30% of those who accessed /company/Page3.html also accessed /company/products.html. There is some information Page1 or Page2 that seems to attract people to file1. This observation might suggest that this information should be moved to a higher level (e.g. company.html) to increase access to file2.html (Figure 16).

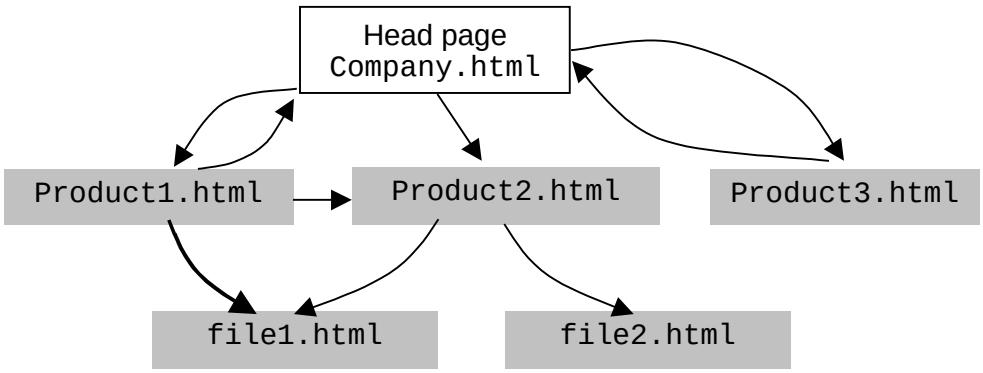


Figure 16: Navigational patterns of customer inform company about the efficiency of their Web page hierarchy.

4.5 Discovering time sequences and sequential patterns

Generally, transaction-based databases collect data over a period of time, and the time-stamp for each transaction is explicitly available. Given such a database of transactions, the problem of discovering sequential patterns is to find inter-transaction patterns such that the presence of a set of items is followed by another item in the time-stamp ordered transaction set. Usually, analyses are made using data taken within a certain time gap [MPT00].

In Web server transaction logs, a visit by a client is recorded over a period of time. The time stamp associated with a transaction in this case will be a time interval which is determined and attached to the transaction during the data cleansing process.

The techniques used in sequential pattern discovery are similar to those used in association rule discovery or classification, except in this case the discovered rules have to be further classified based on the time stamp information [MJHS96, CMS97, MPT00].

For better decision making, non typical patterns have to be discarded. To do so, less frequent sequences are removed based on a minimum support threshold. The sequence is said frequent if its support is higher than the threshold. In order to find frequent sequences, one needs to find all data sequences satisfying the minimum support.

The discovery of sequential patterns in Web server access logs allows Web-based organizations to predict user visit patterns and helps in targeting advertising aimed at groups of users based on these patterns.

4.6 Classification and clustering

After the discovery of hidden common patterns among data items, classification is used to develop a profile for items belonging to a particular group according to their common attributes. This profile can then be used to classify new data items that are added to the database. In Web mining, a profile is built for clients who access particular server files based on demographic information available on those clients.

Classification on WWW access logs allows one to discover relationships such as the following:

clients who often access `/company/products/product3` *tend to be from educational institutions*; or

clients who placed an online order in `/company/products/product2`, *tend to have previously visited the site for* Company X; or

50% *of clients who placed an online order in* `/company/products/product2`, *were in the 20-25 age group and lived on Downtown areas.*

In some cases, valuable information about the customers can be gathered automatically from the client browsers by the server. This includes information available on the client side in the history files, cookie files, etc.

4.6.1 Clustering web data

To make the problem tractable, many approaches proposed to group individual log entries together into user **sessions** (or sessions) [JK98, FSS99]. A *session* consists then of a set of web pages accessed by a user in a certain amount of time. In fact, clusters are found in the sessions rather than in users' entire histories.

A session is identified by scanning the log file at each new IP address met. Subsequent requests from the IP address are added to its session as long as the elapse of time between two consecutive request does not exceed a predefined parameter. Otherwise, the current session is closed and a new session is created. After this process, the web data becomes a set of sessions in the form of (*session_id*, {[*page_id*, *time*]}). This method works well for a small number of pages.

Another method is to group pages according to their **hierarchy** [FSS99]. Thus, the home page of the Web server is considered as the root page and other pages that are related are either node page or leaf page. For clustering such a data format, BIRCH algorithm [ZRL97] was used. BIRCH is a hierarchical and incremental clustering algorithm for a large dataset. In order to apply this algorithm, web data have to be structured in a hierarchical way, in addition to the grouping into sessions.

Clustering of client information or data items on Web transaction logs, can facilitate the development and execution of future marketing strategies. Both online and off-line

marketing can benefit from this process, e.g. automated return mail to clients falling within a certain cluster, dynamically change a particular site for a client on a return visit, based on past classification of that client.

4.7 Illustration

The example taken from [WWW12], applies market basket analysis on an Internet portal (*Eternity.com*) in order to discover actionable information about the visitors.

Eternity.com is a dot com company that runs a wedding website on the Internet. The website has many sections related to weddings to cater to a wide group of Internet users. The objective of the website is to provide information on all things related to weddings and wedding preparation.

For the sake of simplicity, only five sections of the website are considered in the example:

1. Planning
2. Fashion & Beauty
3. Food & Venue
4. Gifts
5. Travel (Holiday locations)

The portal makes money from selling banner and advertising space on its website. It also profits from business arrangements with other companies or affiliates that have a presence on its website. The marketing problem for *Eternity* is to determine the effectiveness of its website. It needs information on what pages on the website are visited to get an insight into who its visitors are and why they visit certain pages. The information obtained will be actionable information:

- It can improve the navigation in the website by suggesting better designs and layouts e.g. changing the order of certain pages or putting links in some pages to others. Better navigation will give the visitors a better browsing experience that would make them stay at the website longer.
- Knowing which sections are visited the most will prompt the company to develop those pages better and know what services or products to have on promotion.

The Data set

Data on the site visitors and their surfing habits can be gathered using server logs. Every time someone visits the site, a record is made. The most interesting data is the path a visitor takes through the website. The traffic log actually 'follows' a visitor while he/she browses along and we can know which pages were visited and in what sequence. This will tell how the visitor came in, what interests him/her, and where he/she left the page. The data from the traffic logfiles of five visitors are gathered in Table 17.

Visitor	Page (Items)
1	Planning, Fashion & Beauty
2	Planning, Fashion & Beauty, Food & Venue
3	Food & Venue, Planning
4	Planning, Travel, Fashion & Beauty
5	Gifts, Travel

Table 17: Data gathered from traffic logfiles of five visitors.

The next step is to create a co-occurrence table (Table 18).

	Plan-ning	**Fashion& Beauty**	**Food & Venue**	**Gifts**	**Travel**
Planning	4	3	2	0	1
Fashion & Beauty	3	4	1	1	1
Food & Venue	2	1	2	0	0
Gifts	0	1	0	1	1
Travel	1	1	0	1	2

Table 18 : The co-occurrence table tells us the number of times two items co-occur in a visit. For instance, we can check at the intersection box between Fashion & Beauty *column and the* Planning *row that there were 3 visits where these pages were viewed with each other.*

By analyzing the data from the logfiles, we can generate rules like:

> *When visitors browse through the* Planning *section, they also visit the* Food & Venue *section 50 percent of the time.*

From the co-occurrence table, we can find associations rules like:

> Planning *and* Fashion & Beauty *pages are more often visited together compared to any other two items.*
>
> Gifts *pages are never visited together with* Planning *or* Food & Venue *pages.*

From these associations, we may form rules like,

> *If a person visits* Fashion & Beauty*, then the person also visits* Planning.

This holds true for *3 out of the 5 number of visits* so the support for this rule is 60 percent.

Then we can look at the confidence of the rule. The rule

> "*If* Fashion & Beauty*, then* Planning*"* is true for all visits involving the *Fashion & Beauty* pages so the confidence is 100 percent.

This could be because those who visit the *Fashion & Beauty* pages will always go to the *Planning* section to find out how far in advance of the wedding they should order a dress, or when they should make a booking with the makeup artist.

However, if we look at the inverse of the rule, *If* Planning, *then* Fashion & Beauty ,the confidence is lower. This is because of 4 the visits involving *Planning*, only 3 included *Fashion & Beauty*. The confidence of this rule is thus only 75 percent.

With the rules generated by the market basket analysis, *Eternity* can take steps to improve on the website and it contents. Some of the changes can include:

- Ensuring sufficient links between the Fashion & Beauty pages with the Planning pages.
- Make the Planning link more visible and easily located since Planning is the most visited section.
- Revamp of the Gifts section to make it more interesting and useful to those looking for gifts either for the bridal couple or for the spouse-to-be so that there will be more hits on that section.

5 How to choose a software package?

Different tools require varying levels of analyst resources, skills, and time to implement. When choosing a product, decisions must be based on both business and technical advantages. The top 10 questions to check are suggested in [WWW13] and partly reported in Table 19 and Table 20.

We don't give here a list of possible software to be used, but we refer to a report on data mining and CRM that has been already made at VTT [Käpylä00]. There are also publications [GG99, Hall99] on survey of data mining tools in general, as well as Web sites [WWW14, WWW15].

BUSINESS ORIENTED CHECKPOINTS	
1: Business Benefits *a)How will this system help us?* *b)How well does this system work for our industry-specific applications?* *c)What information can we get that we do not already have?* ***2: Technical Know-how*** *a)How technically sophisticated do we need to be to use it?* *b)Can business users operate it without calling the IS group all the time?* *c)Is it as easy to use as an internet browser?* ***3: Understandability and Explanations*** *a)Are the results intuitive or difficult to understand?* *b)Do we get clear explanations for any information item presented?* *c)Will the explanations be in technical statistical terms or in a form that we can understand?* ***4: Follow-up Questions*** *a)What kinds of follow-up questions can we ask from the system?* *b)Do we need to go to an analyst for further question answering?* *c)How fast can we drill-down on the fly to see more patterns?* ***5: Business Users*** *a)How many business users can this system support?* *b)Can the business users tailor their own questions for the system?* *c)Can users utilize the knowledge for day-to-day decision making?*	***6: Accuracy, Completeness and Consistency*** *a)How accurate are the results the system delivers?* *b)Can some patterns be missed by the system?* *c)Are the results always consistent or can 100 users get 100 different answers?* ***7: Incremental Analysis*** *a)Can we automatically analyze weekly / monthly data as it becomes available?* *b)Can the system compare the "month to month" results and patterns by itself?* *c)Can we get automatic pattern detection over time, every week or month?* ***8: Data Handling*** *a)How much data can the system deal with?* *b)Can it work directly on our database, or do we need to extract data?* *c)If it works on extracts, how do we know that some patterns are not missed?* ***9: Integration*** *a)How will it integrate into our computing environment?* *b)Will it just work on our existing SQL database?* *c)How easily will the system work on our intranet?* ***10: Support Staff*** *a)What staff do I need to keep this system installed and running?* *b)How do we get support and training to get started?* c)*What happens after we install the system?*

Table 19: Top ten questions from for choosing a data mining product [WWW13].

TECHNICAL ORIENTED CHECKPOINTS	
1: Architecture *a)How are computations distributed between the client and the server?* *b)Is any data brought from the server to the client?* *c)Can the system run in a three tiered architecture?* ***2: Access to Real Data*** *a)Does the system work on the real SQL database or on samples and extracts?* *b)If it samples or extracts, how do we know that it is accurate?* *c)If it builds flat files, who manages this activity and cleans up for on-going analyses, and how can it sample across several tables?* ***3: Performance and Scalability*** *a)How large of a database can the system analyze?* *b) How long does it take to perform discovery on a large database?* *c)Can the system run in parallel on a multi-processor server?* ***4: Multi-Table Databases*** *a)Does the system work on a single table only or can it analyze multiple tables?* *b)Does the system need to perform a huge join to access all of our tables?* *c)If it works on a single table, how can we feed it our existing data schema?* ***5: Multi-Dimensional Analysis*** *a)Does the system analyze data along a single dimension only?* *b)How are multi-dimensional patterns discovered and expressed by the system?* *c)How do we specify the dimensional structure of our data to the system?*	***6: Types and Classes of Patterns Discovered*** *a)How powerful and general are the patterns the system can discover and express?* *b)Can the system mix different pattern types, e.g. influence and affinity patterns?* *c)Can the system discover time-based patterns and trends?* ***7: System Initiative*** *a)Does the system use its own initiative to perform discovery or is it guided by the user?* *b)Can the system discover unexpected patterns by itself?* *c)Can the system start-up by itself on a weekly or monthly basis and perform discovery?* ***8: Treatment of Data Types*** *a)Are all data types handled in their own form or translated to other types?* *b)Can the system find numeric ranges in data by itself?* *c)Do a large number of non-numeric values cause problems for the system?* ***9: Data Dependencies and Hierarchies*** *a)Can the system be told about the functional dependencies in our database?* *b)Does the system understand the concept of data hierarchy?* *c)How does the system use dependencies and/or hierarchies for discovery?* ***10: Flexibility and Noise Sensitivity*** *a)How brittle is the system when dealing with noisy data?* *b)How well does the system cope with data exceptions and low quality data?* *c)Can the system provide statements with flexible numeric ranges discovered by itself in the data?*

Table 20: Top ten questions from the technical point of view for choosing a data mining product [WWW13].

6 Case studies

Customer profiling is a sensitive matter both due to the privacy issues and the competitive advantage it gives to the company conducting profiling. Due to these facts there are few public examples of profiling and the companies involved often like to stay anonymous. Hence, this chapter presents three well-known examples of customer profiling from the data mining literature.

6.1 New segment of potential loan customers

This case study is an example of the benefits of customer segmentation. The case is from Bank of America, one of the biggest banks in United States. The information of this case study is from [BL97].

In the case the National Consumer Assets Group of Bank of America wanted to use data mining to improve its acceptance rate of home equity loans. To achieve this goal they decided to further analyze their home equity loan customers. Before data mining the bank knew two existing groups of home equity loan customers.

In the first group there were people with college-age children. These people wanted to borrow for the tuition bills. The second group consisted of people with high and variable incomes. They wanted to stabilize their incomes by getting a loan against their home equity. However, data mining led to finding of a new and important group of home equity loan customers.

The basis for the data mining in the bank was good. Enough data was available for mining: the bank had nine million retail customers and plenty of historical data. The bank also had a corporate data warehouse, where the data was already cleaned and transformed. The corporate data warehouse also included all the relationships each customer had with the bank. The fields in the database consisted of conventional banking data as well as demographic information such as number of children, income, type of home, etc.

Three different data mining methods were used during the process. Decision tree was used to classify the customers into two groups, those who were likely to respond to a home equity loan offer and those who were not. The training set consisted of both thousands of customers who obtained the loan and thousands of customers who didn't. With the decision tree model the customers were flagged according to their membership in the group of likely loan customers.

Sequential pattern finding was used to find out the frequent sequence of events that preceded the positive loan decision.

Clustering was finally used to cluster the customers into groups with similar characteristics. The clustering found 14 clusters of customers, one of that was especially interesting. This cluster had two important characteristics:

- 39 % of the people in the cluster had both business and personal accounts with the bank.
- The cluster accounted for 27 % of the 11 % of the customers classified by the decision tree as likely responders to a home equity loan offer.

This evidence suggested the people might use home equity loans to start up business. After more analysis Bank of America produced new marketing materials for a campaign and resulted in more than double the original acceptance rate for home equity offers.

6.2 Predicting customer churn

This case study is within telecommunications industry. The case study is presented in [BT99]. A wireless phone provider in United Kingdom wanted to decrease its churn rate, i.e. annual loss of customers. To achieve this goal the company decided to build a model for predicting customer churn and to launch a direct marketing campaign to customers likely to quit.

The study was limited to non-corporate customers resulting to about 260,000 subscribers. The data consisted of both demographic and customer data. Also some data about the customer's contacts to the service center were available.

The data miners ended up using classification and regression tree (CART) for prediction. The training data was from March 1998 and it was used to classify the customers according to their connected or disconnected status at the end of April 1998. After the training the tree had 29 leaf nodes, which were considered as separate customer segments. Each segment was described by the rules produced by CART. An example of the rules is below.

- Contract type is equal to "N", which indicates "no contract"
- Length of service is less than 23.02 months
- Length of service is greater than 9.22 months
- One of 39 tariffs to which the mobile is connected

The maximum predicted churn rate in the leaves was 84 % and the cumulative lift chart can be seen in Figure 17.

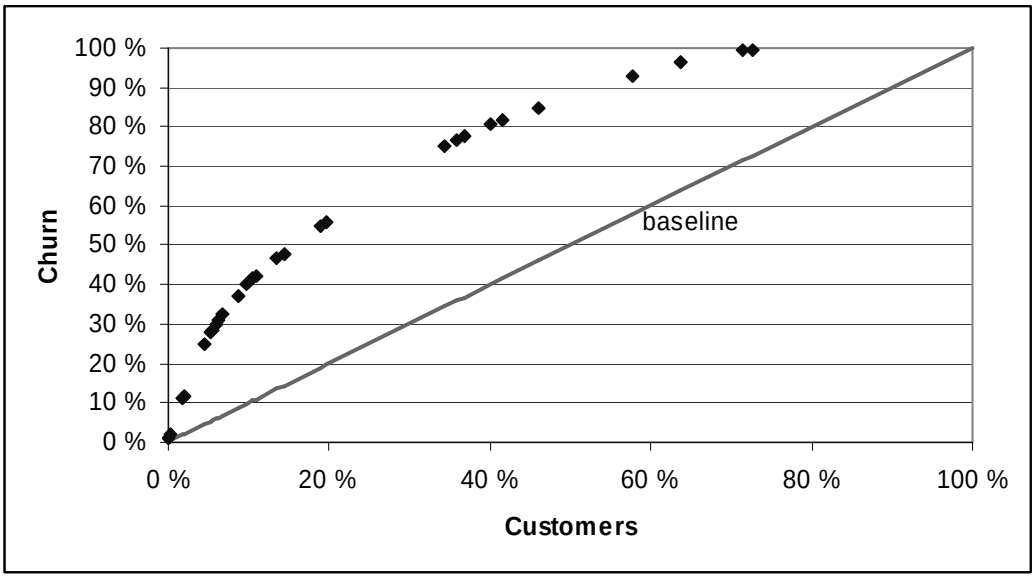


Figure 17. Cumulative lift chart for May 1998. The dots show the predicted amount of customers likely to churn, reached with aid of CART model. If no model is used the amount of likely quitters to be reached is told by the baseline.

When the model was applied to data from April 1998, it was noticed that the number of customers in each segment often changed compared to the data from March 1998. This may come from the fact that the data changes rapidly or the model degenerates quickly.

The model was used in direct marketing campaign in June 1998. One-third of the customers most likely to churn was selected as a target group. The customers in the campaign were offered three months of free line rental in exchange for a contractual commitment for twelve months beyond the end of their existing contract.

As a result of the marketing program the churn rate was reduced during the first months. However, after three months there was a surprising phenomenon. The churn rate of the customers who did respond to the promotion was very low (1.6% for a certain network), but the churn rate of the non-respondent customers was much higher (30.1%) than in a control group (17.1%). Thus, the campaign actually accelerated churn among the non-respondents. A change in the data mining strategy was proposed because of this surprising observation.

Since the marketing campaign was successful for reducing the churn within the respondents, data mining was planned to be used in predicting the likelihoods of both the churn and response to the offer. After this the campaign would be directed to the likely respondents only.

6.3 The most profitable customers of online bookseller

In the third example customer profiling is applied to a test case of an online bookseller. The example is adopted from [Mena99].

The bookseller wanted to get answers to questions like:

- What books are sold to which visitors?
- Which visitors are the most profitable?
- What factors impact its online sales?
- Who is likely to buy what books?

The bookseller had information about the visitors from the registration forms from the website, transaction data, and external information from demographers. The data set was small at size of 2000 customers.

The first phase in answering the questions was visualizing of the data. This was done by histograms and link analysis. This revealed e.g. relationships between certain book categories and age groups.

Next, the customers were both clustered and segmented. Clustering means grouping of the data according to the similarity of the customers without a predefined objective. On the other hand, segmentation is division of the data according to a clear objective, such as a certain output value. Clustering resulted in groups with similar characteristics, e.g. a group with people of age 40-44, income more than $26,976 and living in NY.

In the segmentation the goal was to find the most profitable customers. This was done by a decision tree, which segmented the data set according to the projected total sales of the customers. The decision tree found a segment with average projected online sales of $244.77 when the average of all the customers was $206.85. The tree also gave the rules for recognizing the people who belong to this group.

Prediction of customer behavior is the next phase in profiling of the online bookseller customers. Neural networks are good tools for this kind of tasks and they were trained with online activity of the customers to discover patterns in their behavior. In addition to give a better knowledge of the existing customers, neural networks can be used to target more efficiently new visitors of the website.

Finally, a sensitivity analysis of the neural network gives information about the factors affecting the online sales. In this particular case the most important factors were visitor's home state, book category and age. Thus, the findings of the sensitivity analysis agree with the previous analyses, like link analysis and segmentation.

7 Conclusions

This report is about reviewing some data mining methods that can be used for customer segmentation and profiling. We have defined some requirements for segmentation and profiling. Then some methods on data mining are described. Data mining is a vast field, we had to limit the scope of this report to the most common methods like k-nearest neighbor, neural networks, association rule and sequential pattern discovery. Likewise, each method is described in a very summary way for a quick reading. We have to mention that within the same technology, there may be still some variations in the data mining techniques, sometimes this may due to the specificity of the problems that the tools are tailored to. Web mining is a good example for this remark: e.g. it uses associations or clustering tools which are adapted to Web data. Web mining is discussed in this report as a separate part. As Web mining is gaining a great popularity due to the development of e-commerce, we feel that the Web mining part would need to be more extended if time and resource allow.

References

[AH98] Abraham, Hawks, plc. "Market Segmentation: Can you really divide and conquer?", 1998, at http://www.abramhawkes.plc.uk/pub/mktseg.htm

[BL97] M. Berry and G. Linoff, *Data Mining Techniques : For Marketing, Sales, and Customer Support*, John Wiley & Sons 1997.

[BM98] Alex G. Büchner and Maurice D. Mulvenna, "Discovering Internet Marketing Intelligence through Online Analytical Web Usage Mining", *SIGMOD Record*, vol. 27,no. 4, pp.54-61, 1998, url = http://citeseer.nj.nec.com/buchner98discovering.html

[Brooks97] P. Brooks, "Data Mining Today", *DBMS Online*, Feb. 1997, http://www.dbmsmag.com/9702d16.html

[BT99] S. Berson, K. Thearling, S. J. Smith, *Building Data Mining Applications for CRM*, McGraw-Hill Professional Publishing 1999.

[CC98] T. A. Ciriani and V. Ciriani. Data Mining Methods and Applications http://www.airo.org/aironews/notiz/html/1998/4/bridge.htm

[CDH01] H. P. Crowder, J. Dinkelacker, M. Hsu. "Predictive Customer Relationship Management: Gaining Insights About Customers in the Electronic Economy", in *DM Review* in February 2001. http://www.dmreview.com/master.cfm?NavID=198&EdID=3020

[CDH99] Q. Chen, U. Dayal, M. Hsu, "OLAP-Based Scalable Profiling of Customer Behavior", *Proc. 1st International Conference on Data Warehousing and Knowledge Discovery* (DAWAK '99), 1999, Italy.

[CHY96] M-S Chen, J. Han and P. S. Yu. "Data Mining: An Overview from Database Perspective". *IEEE Trans. On Knowledge And Data Engineering*, vol. 8, pp. 866-883, 1996.

[CMS97] R. Cooley, B. Mobasher, J. Srivastava. "Web Mining: Information and Pattern Discovery on the World Wide Web", *Proceedings of the 9th IEEE International Conference on Tools with Artificial Intelligence* (ICTAI'97), November 1997. URL: http://citeseer.nj.nec.com/cooley97web.html

[FSS99] Y. Fu and K. Sandhu and M. Shih ,"Clustering of Web Users Based on Access Patterns", In *Proceedings of the 1999 KDD Workshop on Web Mining*, San Diego, CA. Springer-Verlag, URL: www.umr.edu/~yongjian/pub/webkdd99.ps

[GG99] M. Goebel and Le Gruenwald, "A Survey of Data Mining and Knowledge Discovery Software Tools", *SIGKDD Explorations* 1999, Vol.1, Issue 1,pp. 20-33.

[Hall99] C. Hall. "Data Mining Tools, Techniques, and Services", in monthly newsletter *Data Management Strategies.* Cutter Information Corp., April 1999. http://www.cutter.com/itgroup/reports/datatool.html

[Hansen00] B.K. Hansen. *Weather Prediction Using Case-Based Reasoning and Fuzzy Set Theory*. Master of Computer Science Thesis, Technical University of Nova Scotia, Halifax, Nova Scotia, Canada. 2000. http://www.cs.dal.ca/~bjarne/thesis/Table_of_Contents.htm

[HAS94] M. Holsheimer and A. P. J. M. Siebes. "Data Mining: The Search for Knowledge in Databases", Technical report CS-R9406, P.O. Box 94079, 1090 GB Amsterdam, 1994. http://citeseer.nj.nec.com/holsheimer91data.html

[HCC92] J. Han, Y. Cai and N. Cercone. Knowledge discovery in databases: an attribute-oriented approach. In *Proc. 18th Int. Conf. Very Large Data Bases*, pp. 547-559, Vancouver, Canada, 1992. Cited in [FSS99].

[JK98] A. Joshi and R. Krishnapuram, "Robust Fuzzy Clustering Methods to Support Web Mining",Proc. Workshop in Data Mining and knowledge Discovery, SIGMOD, pp. 15-1 -- 15-8, 1998. http://citeseer.nj.nec.com/joshi98robust.html

[Joshi97] K. P. Joshi. Analysis of Data Mining Algorithms, http://userpages.umbc.edu/~kjoshi1/data-mine/proj_rpt.htm

[Kimball97] R. Kimball. " Preparing For Data Mining", in *DBMS Online*, Nov. 1997. http://www.dbmsmag.com/9711d05.html

[KJ00] V. Kumar and M. Joshi. "Tutorial on High Performance Data Mining". (1/10/00). http://www-users.cs.umn.edu/~mjoshi/hpdmtut/

[Kock00] A. Kock. " Innovative Strategic Options in Marketing using Information Technology: The Case of the Eccon-Soft AG". European Business School. Seminararbeit, Oct. 2000. URL = http://citeseer.nj.nec.com/401252.html

[Käpylä00] T. Käpylä, "Tiedon louhinta ja asiakkuudenhallinta –tuoteselvitys", VTT research report TTE1- 2000-32, Nov. 2000.

[Mena99] J. Mena. *Data Mining Your Website*, Digital Press, July 1999, ISBN: 1-55558-2222.

[MJHS96] B. Mobasher and N. Jain and E. Han and J. Srivastava, "Web mining: Pattern discovery from world wide web transactions", Technical Report TR-96050, Dep. of Computer Science, University of Minnesota, M inneapolis, 1996.

[MPT00] F. Masseglia, P. Poncelet, and M. Teisseire. "Web Usage Mining: How to Efficiently Manage New transactions and New Customers". research report of LIRMM, Feb. 2000. Short version in *Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery* (*PKDD'00*), Lyon, France, September 2000.

[MTV97] H. Mannila, H. Toivonen, and A. Inkeri Verkamo: Discovery of frequent episodes in event sequences. *Report C-1997-15*, University of Helsinki, Department of Computer Science, February 1997.

[NFJK99] O. Nasraoui and H. Frigui and A. Joshi and R. Krishnapuram, "Mining Web Access Logs Using Relational Competitive Fuzzy Clustering", *8th International Fuzzy Systems Association World Congress - IFSA 99*, Taipei, August 99, url = citeseer.nj.nec.com/286359.html

[Price99] PriceWaterHouseCoopers, *The CRM Handbook: from Group to multiindividual*, PriceWaterHouseCoopers, July 1999.

[PZOD99] S. Parthasarathy, M.J. Zaki, M. Ogihara, S. Dwarkadas, Incremental and interactive sequence mining, *ACM International Conference on Information and Knowedge Management* (CIKM99), Nov 1999.

[SCDT2000] Jaideep Srivastava,Robert Cooley, Mukund Deshpande,Pang-Ning Tan,Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data, Appear in *SIGKDD Explorations*, Vol. 1, Issue 2, 2000.

[SZAS97] C. Shahabi, A. Zarkesh, J. Adibi, V. Shah, Knowledge Discovery from Users Web-Page Navigation, In *Proceedings of the IEEE RIDE97 Workshop*, April 1997.

[Thearling00] K. Thearling, "Data Mining and Customer Relationships", http://www3.shore.net/~kht/index.htm

[Weise00] E. Weise, "A new wrinkle in surfing the Net",*USA Today*, 06/07/00. http://www.usatoday.com/life/cyber/tech/

[Wilson 00] R. F. Wilson, "Preparing a Customer Profile for Your Internet Marketing Plan", *Web Marketing Today*, April 1, 2000, http://www.wilsonweb.com/wmt5/customers-profile.htm

[ZRL97] T. Zhang, R. Ramakrishnan and M. Livny, "BIRCH: A New Data Clustering Algorithm and Its Applications", *Data Mining and Knowledge Discovery*, vol. 1, no. 2, pp. 141-182, 1997, url = citeseer.nj.nec.com/zhang97birch.html

[WWW1] P. Holmes, "Customer Profiling and Modeling", in *DMG Direct*, *Direct marketing Association*, http://www.dirmarketing.com/dmginc/page15.html

[WWW2] Mosaic lifestyle segmentation database, at *Spatial Insight* Inc., http://www.spatialinsights.com/data/attribute/mosaic/

[WWW3] Understanding Market Segmentation by *DSS Research*. http://www.dssresearch.com/Library/Segment/understanding.asp

[WWW4] FAQ on Neural Networks. http://fangorn.ci.tuwien.ac.at/docs/services/nnfaq/FAQ.html

[WWW5] The DBMS Guide to Data Mining Solutions, http://www.dbmsmag.com/

[WWW6] Viscovery SOMine, by Eudaptics, http://www.eudaptics.com/

[WWW7] OR-Notes, J. Beasley, http://www.ms.ic.ac.uk/jeb/or/decmore.html

[WWW8] Data Mining. http://mis.postech.ac.kr/topic/dm_e.html

[WWW9] Li Yang, Data mining courses: http://www.cs.wmich.edu/~yang/teach/cs595/slides/Association.pdf

[WWW10] Successful Marketing on the Internet, Part I: A user's Guide. At *DMG Direct*/Internet marketing, by A. F.-Cassorla. http://www.teleport.com/%7edmginc/Internet-Marketing-Articles/

[WWW11] What is a Web Bug? http://teets.cba.ufl.edu/ism6223f00/knapptk/page2.html

[WWW12] Managing marketing information on the Internet. "Data mining technologies", http://members.tripodasia.com.my/keryng/assign3-1.html

[WWW13] "The Top 10 Data Mining Questions" (May 2001) , by: *Information Discovery*, *Inc*., at *DM Review*, http://www.dmreview.com/,

[WWW14]An overview of data mining methods and products. http://www.cs.chalmers.se/ComputingScience/Education/Courses/xjobb/GUrapporter/MagnusBjornsson/appendixD.html

[WWW15] List of data mining software, http://www.kdnuggets.com/software/