# Data Mining: An Introduction

Ishwar K. Sethi
Intelligent Information Engineering Laboratory
Department of Computer Science and Engineering
Oakland University
Rochester, MI 48309
isethi@oakland.edu

# ABSTRACT

Data mining is concerned with finding hidden relationships present in business data to allow businesses to make predictions for future use. This report provides an introductory overview of data mining technology and products. After explaining what data mining is and establishing its need, the relationship between data warehousing and data mining is explored. Explaining the differences between database query tools and data mining follows this. The report describes in detail the different stages of the data mining process and different data mining models. Important methodologies for data mining are described with their relative strengths and weaknesses. A summary of commercial data mining tools is given in Appendix A along with a set of product evaluation factors that a prospective user can use to evaluate different data mining products. Appendix B of the report presents several real-life examples of data mining in different industries. Appendix B also discusses pitfalls of data mining and presents a set of rules for a new user. Three case studies from banking, telecom, and utilities are presented in Appendix C. A list of important web resources for data mining is also included in the report as Appendix D.

# TABLE OF CONTENTS

# 1

# INTRODUCTION

With the impending deregulation in the utility industry, staying competitive by controlling costs, providing quality service, and developing new markets is expected to gain further urgency. Based on the experiences of other industries that went through deregulation in recent years, it is expected that most utility profits after deregulation will come from commercial and residential mass consumers. Furthermore, the utilities will be able to bundle electricity sales with information, entertainment and other services and products to their consumers. Many utilities have already started acting based on such expectations and this trend is expected to continue. One key computing technology that is likely to play a significant role in the new deregulated environment is data mining. Data mining, also referred to as knowledge discovery in databases, is concerned with nontrivial extraction of implicit, previously unknown and potentially useful information from data in databases. It is a technology which is being rapidly recognized by many industrial companies as an important component of their decision support systems.

This report provides an introductory overview of data mining technology and products. After explaining what data mining is and establishing its need in this section, the relationship between data warehousing and data mining is explored in Section 2. This is followed by explaining the differences between database query tools and data mining in Section 3. The different stages of the data mining process and different data mining models are presented in Section 4. Important methodologies for data mining are described in Section 5. Sections 6 and 7 try to provide answers to the questions "How good is the mined model?" and "Which data mining methodology is best?" respectively. The report also contains a summary of commercial data mining tools in Appendix A, along with a set of product evaluation factors that a prospective user can use to evaluate different data mining products. Appendix B of the report presents several real-life examples of data mining in different industries. It also discusses various pitfalls of data mining and provides a set of rules for a new user. Three case studies from the banking, telecom, and utility industries are presented in Appendix C. A list

of important web resources for data mining is also included in the report in Appendix D.

## What is Data Mining?

In order to understand what data mining is and how a decision-maker can benefit from it let us consider first the distinction between data, information, and knowledge through an example. Suppose you are at your local grocery store in one of the checkout lanes, having completed your weekly grocery shopping. When your grocery goes past the checkout scanner, data is being captured. When your grocery store looks at the accumulated data for different customers at some point in time and finds certain grocery shopping patterns, the captured data is transformed into *information*. It is important to remember that whatever it is that converts data into information resides external to the data and in the interpretation. It is information and not the data that is used to affect a decision. When we have a very high degree of certainty or validity about information, we refer to it as *knowledge*. Continuing with our example, if the grocery shopping patterns discovered by our local grocery store are found to hold at many other grocery stores also, we have a situation where data is finally transformed into knowledge. Thus, we see that information and knowledge are both derived from data.

The modern technology of computers, networks, and sensors have made data collection an almost effortless task. Consequently, data is being captured and stored at a phenomenal pace. However, the captured data needs to be converted into information and knowledge to become useful. Traditionally, the task of extracting information and knowledge from recorded data has been performed by analysts; however, the increasing volume of data in modern business enterprises calls for computer-based methods for this task. Such methods have come to be known as *data mining* methods and the entire process of applying computer-based methodology is known as *knowledge discovery*. Note that this data mining viewpoint does not impose any restriction on the nature of the underlying computer data analysis tools. This is the viewpoint that is held by most of the vendors of data mining products. However, some people, especially those belonging to the artificial intelligence community, have a slightly narrower definition for data mining. According to their viewpoint, the underlying data analysis tools must be based on one or more sub-technologies of artificial intelligence, for example machine learning, neural networks, or pattern recognition, to qualify as the data mining method.

## Importance In Business Decision Making

Data mining technology is currently a hot favorite in the hands of decision-makers as it can provide valuable hidden business intelligence from historical corporate data. It should be remembered, however, that fundamentally, data mining is not a new technology. The concept of extracting information and knowledge discovery from recorded data is a well-established concept in scientific and medical studies. What is new is the convergence of several factors that have created a unique opportunity for data mining in the corporate world. Businesses are suddenly realizing that the data that they have been collecting for the past 15-20 years can give them an immense competitive edge. Due to the client-server paradigm, data warehousing technology, and the currently available immense desktop computing power, it has become very easy for an end-user to look at stored data from all sorts of perspectives and extract valuable business intelligence. Data mining is being used to perform market segmentation to launch new products and services as well as to match existing products and services to customers' needs. In the banking, healthcare, and insurance industries, data mining is being used to detect fraudulent behavior by tracking spending and claims patterns. In the context of the utility industry with its impending deregulation, data mining technology is poised to play a significant role. It is going to let utilities stay competitive by controlling costs, providing quality service, and developing new markets. By tapping into current customer databases, data mining can help utilities identify customer needs to serve them better, yield clues to hidden market segments to give superior value to utility customers and identify new electricity-oriented products and services.

## Data Classification

One can classify data into three classes: (1) *structured data*, (2) *semi-structured data*, and (3) *unstructured data*. Most business databases contain structured data consisting of well-defined fields of numeric or alphanumeric values. Semi-structured data has partial structure. Examples of semi-structured data are electronic images of business documents, medical reports, executive summaries, and repair manuals. The majority of web documents also fall in this category. An example of unstructured data is a video recorded by a surveillance camera in a departmental store. Such visual or multimedia recordings of events or processes of interests are currently gaining widespread popularity due to reducing hardware costs. This form of data generally requires an extensive amount of processing to extract contained information. Structured data is often referred to as *traditional data* while the semi and unstructured data are lumped together as *non-traditional data*. Because of the presence of structure in it, traditional data has no ambiguity. On the other hand, non-traditional data is difficult to interpret and often has multiple interpretations. Most of the current data mining methods and

commercial tools are meant for traditional data; however, development of data mining tools for non-traditional data is growing at a rapid rate.

Another way of looking at data, especially traditional data, is to look at the behavior of recorded attributes with respect to time. Certain attributes, for example a customer's social security number, do not change with time. A database containing only such kinds of records is considered to have *static data*. On the other hand, there are attributes, for example a customer's monthly utility consumption, that change with time. A database containing such records is considered to have *dynamic* or *temporal data* as well. The majority of the data mining methods are more suitable for static data and special consideration is often required to mine dynamic data.

# 2

# DATA MINING AND DATA WAREHOUSING

---

In this section, the relationship between data warehousing and data mining is addressed. Although the existence of a data warehouse is not a prerequisite for data mining, in practice the task of data mining is made a lot easier by having access to a data warehouse.

## Data Warehouse

A *data warehouse* can be viewed as a data repository for an organization set up to support strategic decision-making. The architecture and the features of a data warehouse are very different from other databases in an organization, which are operational databases designed to support day-to-day operations of an organization. The function of a data warehouse is to store historical data of an organization in an integrated manner to reflect the various facets of the organization's business. The data in a warehouse is never updated but used only to respond to queries from end-users, who are generally the decision-makers. This is in contrast with the users of operational databases, whose interaction with the database consists of either reading some records from it or updating them. Unlike operational databases, data warehouses are huge in size storing billions of records. In many instances, an organization may have several local or departmental data warehouses. Such data warehouses (see Figure 1) are often called *data marts* due to their smaller size.

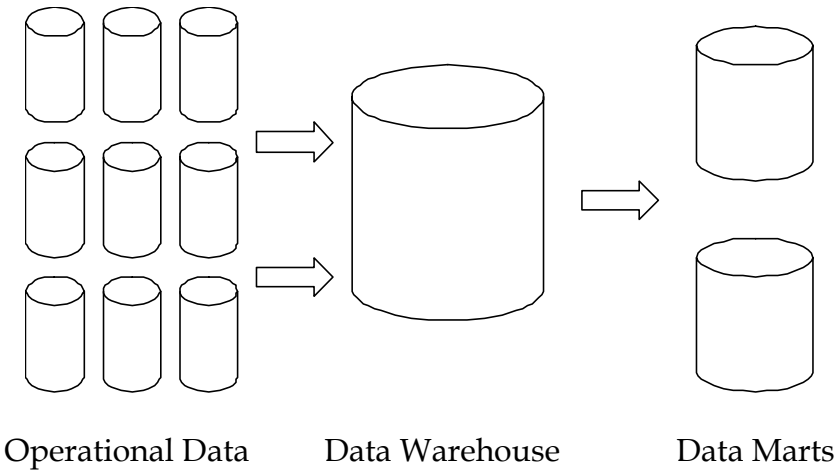



**Figure 1. Operational databases, a data warehouse, and data marts**

## Relationship Between Data Mining and Data Warehousing

The link between data mining and data warehousing is a mutually reinforcing link. It is difficult to say whether the prospects of an informed and focussed decision-making through data mining are responsible for a surge in industry-wide interest in data warehousing; or whether the availability of clean, well-formatted historical data in warehouses is the cause of current boom in data mining. In any case, data mining is one of the major applications for data warehousing since the sole function of a data warehouse is to provide information to end-users for decision support. Unlike other query tools as explained in the next section, the data mining tools provide an end-user with a capability to extract hidden information. Such information, although more difficult to extract, can provide a bigger business advantage and yield higher returns on data warehousing investments for a business. Thus, the data mining tools, although one of the many tools to extract information from a data warehouse, are very important for the success of a data warehouse. To put a perspective on data warehousing and data mining activities in the corporate world, one should note that more than 90% of Fortune 1000 companies have a data warehouse. The remaining companies are in the process of implementing one within the very near future. Similarly, many computer industry vendors – both hardware and software, either have a data mining product or are in the process of developing one. According to a report from Meta Group, a consulting firm in Burlingame, California, the current market for data mining is estimated to be $300M and is expected to reach $800M by the year 2000.

# 3

# DATA MINING AND QUERY TOOLS

All databases provide a variety of query tools for users to access information stored in the database. For ease of operation, these query tools generally provide a graphical interface to users to express their queries. In the case of relational databases, the query tools are known as *SQL* tools because of the use of *Structured Query Language* to query the database. In the case of dimensional databases, the query tools are popularly known as *on-line analytical processing* (*OLAP*) tools. Following the viewpoint that data mining is concerned with the extraction of information and knowledge from databases, one obvious question to raise is "How is data mining different from structured query language (SQL) and on-line analytical processing (OLAP) tools?" In this section, we try to provide an answer to this question.

## Data Mining and SQL

SQL is the standard language for relational database management systems. SQL is good for queries that impose some kind of constraint on data in the database in order to extract an answer. In contrast, data mining is good for queries that are exploratory in nature. For example, if we want to obtain a list of all utility customers whose monthly utility bill is greater than some specified dollar amount, we can get this information from our database using SQL. However, SQL is not a very convenient tool if we want to obtain the differences in customers whose monthly bills are always paid on time with those customers who are usually late in their payments. It is not that this question cannot possibly be answered by SQL. By several trial and error steps, perhaps one can arrive at the answer through SQL. Data mining is, on the other hand, very good at finding answers to the latter types of questions. Thus, we can say that SQL is useful for extracting obvious information, i.e. shallow knowledge, from a database but data mining is needed to extract not so obvious, i.e. hidden, information from a database. In other words, SQL is useful when we know exactly what we are looking for; but we need data mining when we know only vaguely what we are

looking for. Thus, SQL and data mining are complementary and both are needed to extract information from databases.

## Data Mining and OLAP

OLAP tools have become very popular in recent years as they let users play with data stored in a warehouse by providing multiple views of the data. In these views, different dimensions correspond to different business characteristics, e.g. sales, geographic locations, product types etc. OLAP tools make it very easy to look at dimensional data from any angle or to "slice-and-dice" it. For example, it is easy to answer questions like "How have increased advertising expenditures impacted sales in a particular territory?" with OLAP tools. To provide such answers, OLAP tools store data in a special format which corresponds to a multi-dimensional hyper-box structure. Although OLAP tools like data mining tools provide answers that are derived from data, the similarity between the two sets of tools ends here. The derivation of answers from data in OLAP is analogous to calculations in a spreadsheet; OLAP tools do not learn from data; nor do they create new knowledge. They are simply special-purpose visualization tools that can help an end-user learn patterns in the data. In contrast, data mining tools obtain their answers via learning the relationships between different attributes of database records. Often, these discovered relationships lead to creation of new knowledge by providing new insights to business. Thus, OLAP tools are useful for data mining because of their capabilities to visualize relationships between different data dimensions; however, they are not a substitute for data mining.

# 4

# THE DATA MINING PROCESS

The data mining process consists of several stages and the overall process is inherently interactive and iterative. The main stages of the data mining process, shown in Figure 2, are: (1) domain understanding; (2) data selection; (3) cleaning and preprocessing; (4) discovering patterns; (5) interpretation; and (6) reporting and using discovered knowledge. The bi-directional arrows in Figure 2 imply that some or all of the data mining stages are executed more than once in an iterative manner.

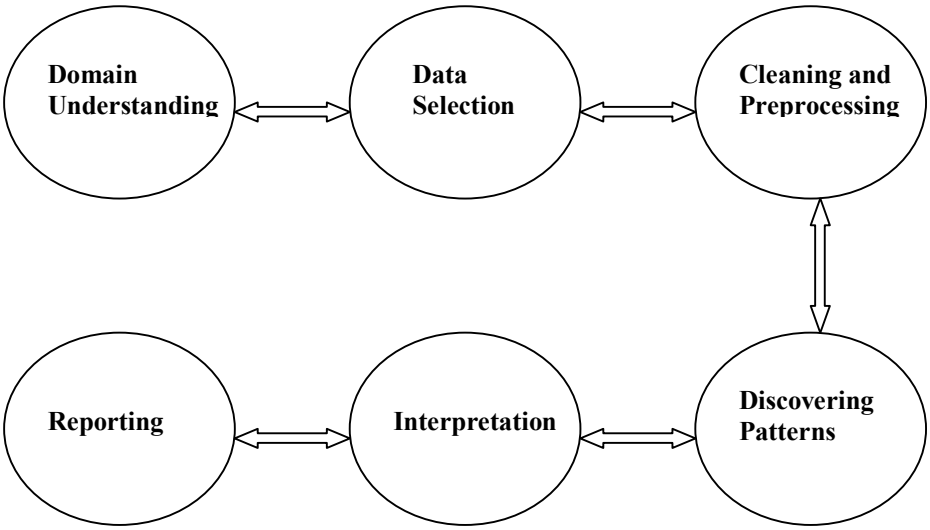



**Figure 2. Different stages of the data mining process**

## Domain Understanding

The domain understanding stage requires learning the business goals of the data mining application as well as gathering relevant prior knowledge. Blind application of data mining techniques without the requisite domain knowledge often leads to the discovery of irrelevant or meaningless patterns. This stage is best executed by a team of business and information technology persons to develop an all-around understanding of the data mining task being undertaken.

## Data Selection

The data selection stage requires the user to target a database or select a subset of fields or data records to be used for data mining. Having a proper domain understanding at this stage helps in the identification of useful data. Sometimes, a business may not have all the requisite data in-house. In such instances, data is purchased from outside sources. Examples of data often purchased from outside vendors include demographic data and life-style data. Some applications of data mining also require data to be obtained via surveys.

## Cleaning and Preprocessing

This is the most time-consuming stage of the entire data mining process. Data is never clean and in the form suitable for data mining. The following are typical of data corruption problems in business databases:

- Duplication - This kind of data corruption occurs when a record, for example a customer's purchases, appears several times in a database. It is one of the most common data corruption problems found in databases of businesses, such as direct mailers and credit card companies, dealing with individual customers. This kind of corruption is generally caused by misspelling due to typing/entry errors. Sometimes customers are known to misspell deliberately to avoid linkage with their own past records.

- Missing Data Fields - Missing fields are present in a database due to a variety of reasons. For example, a customer may simply get tired of filling in the desired information; or a missing field may be caused by a data entry error with an improper entry for a field. Filling in the missing values is generally a non-trivial task. Often, the records with missing fields are ignored for further processing.

- Outliers - An outlier is a data value in a field, which is very different from the rest of the data values in the same field. As an example, consider the field of monthly energy consumption for residential customers of a utility. If this field

typically ranges from 0-1000KW and we have an entry of 10,000KW in this field for some customer, then this entry is considered an outlier. The presence of outliers in a database is generally due to incorrect recordings of outlier fields. In many instances, outliers are easy to spot by considering domain consistency constraints. Sometimes, an outlier may be present due to exceptional circumstances such as a stolen credit card, when considering the monthly expenditure field in a credit card database. Detection of such outliers requires a considerable effort and often such a detection step itself is called the data discovery step.

### *Preprocessing*

The preprocessing step of the third stage of data mining involves integrating data from different sources and making choices about representing or coding certain data fields that serve as inputs to the data discovery stage. Such representation choices are needed because certain fields may contain data at levels of details not considered suitable for the data discovery stage. For example, it may be counter-productive to represent the actual date of birth of each customer to the data discovery stage. Instead, it may be better to group customers into different age groups. Similarly, the data discovery stage may get overwhelmed by looking at each customer's address and may not generate useful patterns. On the other hand, grouping customers on a geographical basis may produce better results. It is important to remember that the preprocessing step is a crucial step. The representation choices made at this stage have a great bearing on the kinds of the patterns that will be discovered by the next stage of data discovery.

## Discovering Patterns

The data-pattern-discovery stage is the heart of the entire data mining process. It is the stage where the hidden patterns and trends in the data are actually uncovered. In the academic or research literature, it is only this stage that is referred to as data mining with the entire process of Figure 2 being termed as KDD (Knowledge Discovery in Databases). However, no such distinction is made in industry where the term data mining applies to the entire process of Figure 2. There are several approaches to the data discovery stage. These include association, classification, clustering, regression, sequence analysis, and visualization. Each of these approaches can be implemented through one of several competing methodologies, such as statistical data analysis, machine learning, neural networks, and pattern recognition. It is because of the use of methodologies from several disciplines that data mining is often viewed as a multidisciplinary field (see Figure 3). Here, we will provide details about

different approaches to data discovery. The details about different methodologies will be presented in the next section.

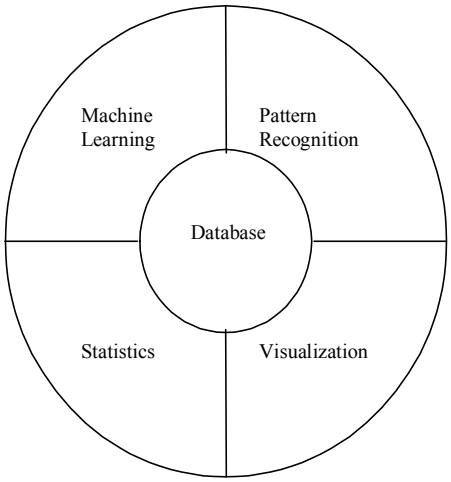



**Figure 3. Core technologies for data mining**

- Association - This approach to data discovery seeks to establish associative relationships between different items in database records. The association approach is very popular among marketing managers and retailers who find associative patterns like "90% of customers who buy product X also buy product Y" extremely helpful for market targeting and product placement in stores. The association approach to data discovery is successful when one has an idea of different associations that are being sought out. This is because one can find all kinds of correlations in a large database. Statistics, machine learning, and neural networks are popular methodologies for the association approach to data discovery.

- Classification - The classification approach to data discovery is perhaps the most widely used approach. It consists of classifying records into two or more pre-determined classes. As an example, consider a utility company planning to offer an appliance service plan to its customers. To get maximum response for a planned telemarketing effort, the utility may want to classify its customers into two classes – customers likely to respond and customers not likely to respond. Once such a classification is done, the telemarketers can concentrate on only those customers that fall in the first category. The application of the classification approach requires a classification rule, which is generally extracted from an existing set of pre-classified records. Such a set of records is often termed as a *training set* and the process of extracting the classification rule is commonly known as *learning*. Decision tree classifiers and neural network classifiers are two of the most popular methodologies for implementing the classification approach to data discovery.

- Clustering - Clustering implies data grouping or partitioning. This approach to data discovery is used in those situations where a training set of pre-classified records is unavailable. The major applications of clustering are in market segmentation and mining of customers' response data. Clustering is also known as *exploratory data analysis* (EDA). Other terms for clustering are *unsupervised learning* and *self-organization*. Performing clustering with a known number of groupings is relatively easy in comparison with those situations when the number of groups is not known a-priori and must be determined by the clustering process itself. The example of Figure 4 shows the difficulties that one encounters during clustering. As this example shows, there is no unique answer to the question of the number of clusters present in this data. Similarly, there is no single cluster shape. Some clusters are circular while others are elongated. This generally causes difficulties in having a meaningful criterion for clustering. Despite these difficulties, clustering, however, is a very popular data analysis tool. Statistical pattern recognition, neural networks, and fuzzy logic offer a variety of clustering algorithms.

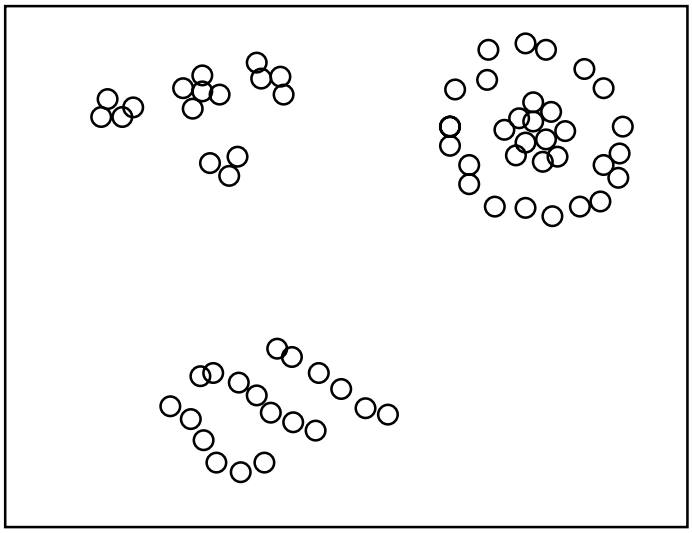


**Figure 4. Examples of clusters in two-dimensions**

- Sequence Analysis - This approach is used for discovering patterns in time-series data. For example, we could use this approach to determine the buying patterns of credit-card customers to be able to predict their future purchases. Such predictive information can be used for identifying stolen credit cards. Sequence analysis is also used to establish associations over time. For example, it can be used to find patterns like “80% of customers who buy

product X are likely to buy product Y in the next six months." This allows marketers to target specific products and services that the customers are more likely to buy. The popular methodologies for sequence analysis are rooted in statistics and neural networks.

- Visualization - The visualization approach to data mining is based on an assumption that human beings are very good at perceiving structure in visual forms. This approach thus consists of providing the user with a set of visualization tools to display data in various forms. The actual discovery of the patterns in the data is made by the user while viewing the visualized data. An extreme of the visualization approach to data mining consists of creating an immersive virtual reality (VR) environment so that a user can move through this environment discovering hidden relations.

## Interpretation

The interpretation stage of the data mining process is used by the user to evaluate the quality of discovery and its value to determine whether previous stages should be revisited or not. Proper domain understanding is crucial at this stage to put a value on discovered patterns.

## Reporting

The final stage of the data mining process consists of reporting and putting to use the discovered knowledge to generate new actions or products and services or marketing strategies. Without this step, the full benefits from data mining cannot be realized. Reporting can take many forms, including detailed graphical presentation of the patterns discovered and the transfer of the mined knowledge or model to the appropriate business application.

# 5

# DATA MINING METHODOLOGIES

This section of the report presents basic concepts of different data mining methodologies. We present a few selected techniques from different methodologies. From statistical data analysis methods, we describe linear regression, logistic regression, linear discriminant analysis, and clustering techniques. From pattern recognition, we focus mainly on nearest neighbor classification. After presenting the basic neuron model, we describe briefly single-layer perceptron network, multiple-layer feed-forward network, and self-organization feature map methods from neural network methodology of data mining. From machine learning, we describe decision tree methods and genetic algorithms in detail.

Many different types of variables or attributes, i.e. fields in a database record, are common in data mining. Not all of the data mining methods are equally good at dealing with different types of variables. Therefore, we first explain the different types of variables and attributes to help readers in determining the most suitable methodology for a given data mining application.

## Types of Variables

There are several ways of characterizing variables. One way of looking at a variable is to see whether it is an *independent* variable or a *dependent* variable, i.e. a variable whose value depends upon values of other variables. Another way of looking at a variable is to see whether it is a *discrete* variable or a *continuous* variable. Discrete variables are also called *qualitative* variables. Such variables are measured or defined using two kinds of non-metric scales – *nominal* and *ordinal*. A nominal scale is an order-less scale, which uses different symbols, characters or numbers, to represent the different states of the variable being measured. An example of a nominal variable is the customer type identifier, which might represent three types of utility customers – residential, commercial, and industrial, using digits 1, 2, and 3, respectively. Another example of a nominal attribute is the zip-code field of a customer's record. In each of these two

examples, numbers designating different attribute values have no particular order and no necessary relation to one another. An ordinal scale consists of ordered discrete gradations, e.g. rankings. An example of an ordinal attribute is the preference ordering by customers, say of their favorite pizza. An ordered scale need not be necessarily linear, e.g. the difference in rank orders 3 and 4 need not be identical to the difference in rank orders 6 and 7. All that can be established from an ordered scale is the greater-than or less-than relations. The continuous variables are also known as *quantitative* or *metric* variables. These variables are measured using either an *interval* scale or a *ratio* scale. Both of these scales allow the underlying variable to be defined or measured with infinite precision. The difference between the interval and ratio scales lies in how the zero point is defined in the scale. The zero point in the interval scale is placed arbitrarily and thus it does not indicate the complete absence of whatever is being measured. The best example of an interval scale is the temperature scale, where zero degrees Fahrenheit does not mean total absence of temperature. Because of the arbitrary placement of the zero point, the ratio relation does not hold true for variables measured using interval scales. For example, 80 degrees Fahrenheit does not imply twice as much heat as 40 degrees Fahrenheit. In contrast, a ratio scale has an absolute zero point and consequently the ratio relation holds true for variables measured using this scale. This is the scale that we use to measure such quantities as height, length, energy consumption, and salary.

## Statistical Data Analysis

Statistical data analysis is the most well established methodology for data mining. Ranging from 1-dimensional analysis, e.g. mean, median, and mode of a qualitative variable, to multivariate data analysis simultaneously using many variables in analysis, statistics offers a variety of data analysis methods. These data analysis methods can be grouped into two categories. The methods in the first category are known as *dependence* methods. These methods use one or more independent variables to predict one or more dependent variables. Examples of this category of methods include multiple regression and discriminant analysis. The second category of statistical data analysis methods is known as *interdependence* methods. These methods are used when all of the variables involved are independent variables. Examples of interdependence methods are different types of clustering methods and multidimensional scaling.

### *Dependence Methods*

#### Multiple Linear Regression

Multiple regression method is used when a single dependent quantitative variable (also called the outcome variable) is considered related to one or more quantitative independent variables, also known as *predictors*. The objective of regression analysis is to determine the best model that can relate the dependent variable to various independent variables. Linear regression implies the use of a general linear statistical model of the following form

$$y = a_0 + a_1 x_1 + a_2 x_2 + \cdots + a_k x_k + \varepsilon$$

where $y$ is the dependent variable and $x_1, x_2 \ldots, x_k$ are the independent variables. The quantities, $a_0, a_1 \ldots, a_k$, are called unknown parameters and $\varepsilon$ represents the random error. The unknown parameters are determined by minimizing the sum of squared error (SSE). It is defined as

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

where $y_i$ and $\hat{y}_i$, respectively, are the observed and predicted values of the dependent variable for i-th the record in the database of $n$ records. When only one independent variable is involved, the process of model fitting is equivalent to best straight-line fitting in least square sense. The term *simple regression* is often used in that situation.

It should be noted that the term *linear* in the general linear model applies to the dependent variable being a linear function of the unknown parameters. Thus, a general linear model might also include some higher order terms of independent variables, e.g. terms such as $x_1^2, x_1 x_2,$ or $x_2^3$. The major effort on the part of a user in using the multiple regression technique lies in identifying the relevant independent variables and in selecting the regression model terms. Two approaches are common for this task: (1) sequential search approach, and (2) combinatorial approach. The sequential search approach consists primarily of building a regression model with a set of variables, and then selectively adding or deleting variables until some overall criterion is satisfied. The combinatorial approach is a brute force approach, which searches across all possible combinations of independent variables to determine the overall best regression model. Irrespective of whether the sequential or combinatorial approach is used,

the most benefit to model building occurs from a proper understanding of the application domain.

## Logistic Regression

In many applications, the dependent variable is a qualitative variable, e.g. the credit rating of a customer, which can be good or bad. In such cases, either logistic regression or discriminant analysis is used for prediction. Rather than predicting the state of the dependent variable, the logistic regression method tries to estimate the probability $p$ that the dependent variable will be in a given state. Thus, in place of predicting whether a customer has a good or bad credit rating, the logistic regression approach tries to estimate the probability of a good credit rating. The actual state of the dependent variable is determined by looking at the estimated probability. If the estimated probability is greater than 0.50, then the prediction is yes (good credit rating), otherwise no (bad credit rating). In logistic regression, the probability $p$ is called the *success probability* and is estimated using the following model:

$$\log(p/(1-p)) = a_0 + a_1x_1 + a_2x_2 + \cdots + a_kx_k$$

where $a_0, a_1 \ldots, a_k$ are unknown parameters. This model is known as the *linear logistic model* and $\log(p/(1-p))$ is called the *logistic* or *logit* transformation of a probability. Unlike the multiple linear regression where the unknown parameters are estimated using the least squares method, the logistic regression procedure determines unknown parameters by the likelihood maximization method. With respect to the credit rating example, this means maximizing the likelihood of a good credit rating.

## Linear Discriminant Analysis

Linear discriminant analysis (LDA) is concerned with problems that are characterized as classification problems. In such problems, the dependent variable is categorical (nominal or ordinal) and the independent variables are metric. The objective of discriminant analysis is to construct a discriminant function that yields different scores when computed with data from different classes. A linear discriminant function has the following form:

$$z = w_1x_1 + w_2x_2 + \cdots w_kx_k$$

where $x_1, x_2 \ldots, x_k$ are the independent variables. The quantity $z$ is called the *discriminant score*, and $w_1, w_2 \ldots, w_k$ are called *weights*. A geometric interpretation of

the discriminant score is shown in Figure 5. As this figure shows, the discriminant score for a data record represents its projection onto a line defined by the set of weight values.

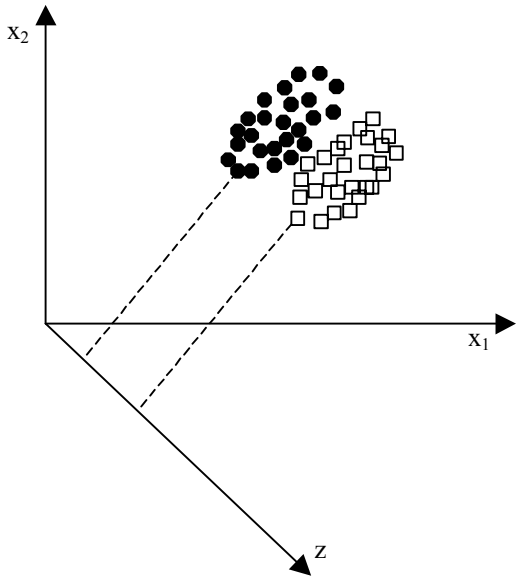



**Figure 5. Geometric interpretation of the discriminant score**

The construction of a discriminant function involves finding a set of weight values that maximizes the ratio of the between-group to the within-group variance of the discriminant score for pre-classified records (training examples) from the database. Once constructed, the discriminant function is used to predict the class of a given data record, i.e. the state of the dependent variable from the independent variables. The classification is performed by the following classification rule. Assign the *i-th* data record to class A (e.g. good credit rating) if its discriminant score $z_i$ is greater than or equal to the cutting score; otherwise assign it to class B (i.e. bad credit rating). The cutting score thus serves as a criterion against which each individual record's discriminant score is judged. The choice of cutting score depends upon whether both classes of records are present in equal proportions or not, as well as the underlying distributions. It is common to assume the underlying distributions to be normal. Letting $\tilde{z}_A$ and $\tilde{z}_B$ be the mean discriminant scores of pre-classified data records from class A and B, respectively, the optimal choice for the cutting score, $z_{cut}$, is given as

$$z_{cut} = \frac{\tilde{z}_A + \tilde{z}_B}{2}$$

when the two groups of records are of equal size and are normally distributed with uniform variance in all directions. A weighted average of mean discriminant scores, calculated as follows, is used as an optimal cutting score when the groups are not of equal size:

$$z_{cut} = \frac{n_A \tilde{z}_A + n_B \tilde{z}_B}{n_A + n_B}$$

The quantities $n_A$ and $n_B$ in above, respectively, represent the number of records in each group.

While a single discriminant function is constructed for two-way classification, multiple discriminant functions are required when dealing with more than two classes. The term *multiple discriminant analysis* is used in such situations. For an *M*-way classification problem, i.e. a dependent variable with *M* possible outcomes, *M* discriminant functions are constructed. The classification rule in such situations takes the following form: "Decide in favor of the class whose discriminant score is highest." This is illustrated in Figure 6.

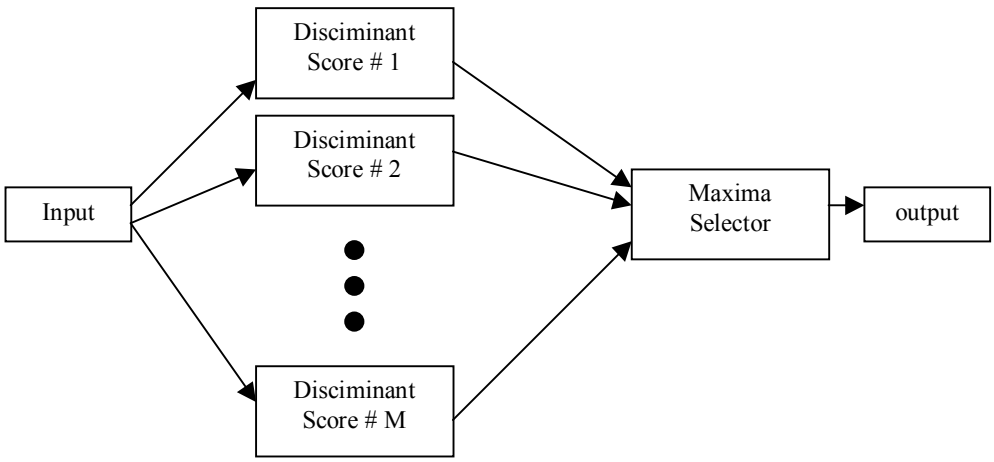


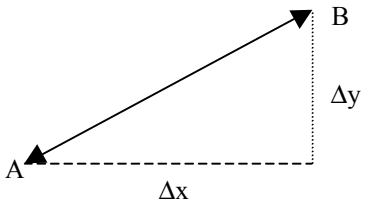

**Figure 6. Illustration of classification in multiple discriminant analysis**

### *Interdependence Methods*

There are many applications where there is no dependent variable. In such applications, interdependence methods such as clustering and multidimensional scaling are used.

## Clustering

Clustering or cluster analysis refers to methods for grouping objects – individuals, products, and services, in such a way that each object is more similar to objects in its own group than to objects in other groups. Since clustering methods are used in a wide range of disciplines, there exists a variety of names for clustering such as *unsupervised classification*, *Q analysis*, *typology*, and *numerical taxonomy*. A clustering method is characterized by how it measures similarity and what kind of method is employed to perform grouping. There are several ways of measuring similarity, with Euclidean distance being the most commonly used measure. Given two objects, A and B, the *Euclidean distance* between them is defined as the length of the line joining them. Other distance measures are the *absolute* and *maximum* distance functions. The differences between these different measures are shown in Figure 7. Irrespective of the distance measure being used, a small distance value between two objects implies high similarity and vice-versa a large distance value implies low similarity. Since objects in any application will have many attributes, each measured with a different scale, it is very common to use a normalized distance function in clustering. A normalized distance function incorporates a raw data normalization step so that each raw value is converted into a standard variate with a zero mean and a unit variance. The most commonly used normalized distance measure is the *Mahalanobis distance*, which not only takes care of different scales for different attributes but also accounts for inter-correlations among the attributes.




$$\text{Euclidean Distance} = (\Delta x^2 + \Delta y^2)^{1/2}$$

$$\text{Absolute Distance} = (|\Delta x| + |\Delta y|)$$

$$\text{Maximum Distance} = \max\{|\Delta x|, |\Delta y|\}$$

**Figure 7. Illustration of distance measures**

Most of the common clustering methods can be classified into two general categories: (1) *hierarchical* and (2) *partitional*. A hierarchical clustering procedure achieves its clustering through a nested sequence of partitions, which can be represented in a treelike structure. On the other hand, a partitional clustering

method performs clustering in one shot. Figure 8 shows these differences in two general types of clustering procedures.

### Hierarchical Clustering

Hierarchical clustering procedures can be further divided into two basic types – *agglomerative* and *divisive*. In agglomerative clustering, each object starts out as its own cluster. The subsequent stages involve pair-wise merging of two most similar clusters or objects. This process continues until the number of clusters is reduced to the desired number, or eventually all objects are grouped into a single cluster as shown in Figure 8. The treelike structure of Figure 8 is often referred to as a *dendrogram*. Divisive approach to hierarchical clustering is exactly opposite to the agglomerative approach. Here, we begin with one large cluster of all objects. In subsequent steps, objects most dissimilar are split off to yield smaller clusters. The process is continued until each object becomes a cluster by itself. Agglomerative procedures are much more popular and are provided by most statistical software packages. Some of the popular agglomerative clustering procedures are *single linkage*, *complete linkage*, and *average linkage*. These methods differ in how the similarity is computed between clusters. Figure 9 illustrates these differences.

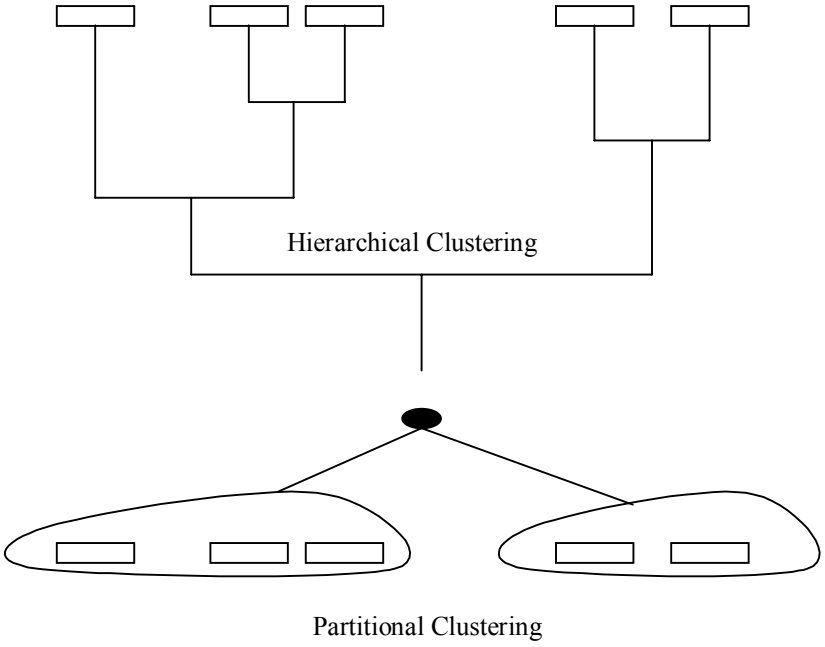



**Figure 8. Differences in hierarchical and partitional clustering**

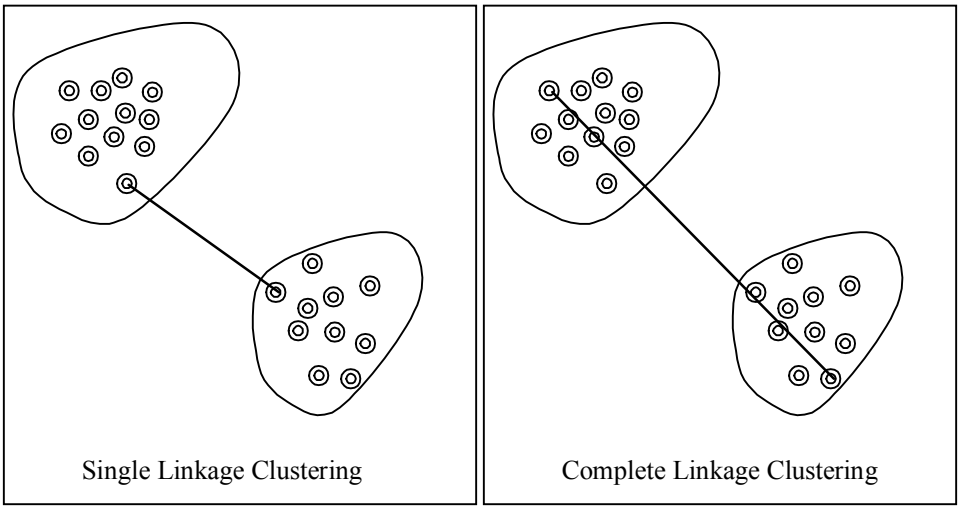



**Figure 9. Similarity computation in single linkage and complete linkage clustering**

### Partitional Clustering

Partitional clustering procedures can be classified as *sequential* or *simultaneous* procedures. In a sequential partitional clustering procedure, objects to be clustered are handled one by one and the ordering of presentation of objects usually has an influence on the final clusters. Simultaneous methods, in contrast, look at all objects at the same time and thus generally produce better results. Many partitional clustering methods achieve clustering via optimization. Such procedures are known as *indirect methods* in contrast with direct partitional clustering methods, which do not use any optimization method. The most well known partitional clustering method is the *K-means* method, which iteratively refines the clustering solution once the user specifies an initial partition or from a random initial partition.

### Comparison of Hierarchical and Partitional Clustering

Historically, the hierarchical clustering techniques have been more popular in biological, social, and behavioral sciences whereas partitional methods are more frequent in engineering applications. The main advantage of hierarchical procedures is their speed. The other advantage is that no knowledge of the number of clusters is required. The disadvantage of hierarchical methods is that they sometime lead to artificial groupings because grouping mistakes cannot be

reversed due to the hierarchical nature of the grouping process. In contrast, the partitional methods are relatively slow but tend to produce better results. However, these methods rely on the user to provide good initial seed points for clusters and thus demand a better domain understanding on the part of the user. Irrespective of the selected clustering procedure, it is generally advisable to compute several solutions before settling on one as the final solution.

### Multidimensional Scaling

Multidimensional scaling (MDS) relies on a projection from a high dimensional space to a low dimensional space (two or three dimensions) to uncover similarities among objects. For the mapping from a high-dimensional space to a low-dimensional one to be useful in MDS, it is required that the mapping preserve inter-point distances as far as possible. MDS is also known as *perceptual mapping* and the resulting projection as the *perceived relative image*. The main application of MDS lies in evaluating customer preferences for products and services. There are two classes of MDS techniques – *decompositional* and *compositional*. The decompositional approach, also known as the *attribute-free* approach, is used in situations where only overall similarity data for different objects is available. In contrast, the compositional methods are used when detailed data across numerous attributes for each object is available. Most statistical software packages provide both kinds of MDS methods.

## Pattern Recognition

Pattern recognition theory and practice is concerned with the design, analysis, and development of methods for classification or description of patterns – objects, signals, and processes. The classification is performed using such physical properties of patterns as height, width, thickness, and color. These properties are called *features* or *attributes* and the process of obtaining feature measurements for patterns is called *feature extraction*. Pattern recognition systems are used in two kinds of applications. The first kind of applications are those where a pattern recognition system provides cost and speed benefits. Examples of such applications are part location and identification in manufacturing, handwriting recognition in banking and offices, and speech recognition. The second kind of applications are those where a pattern recognition system is used to perform a complex identification task either to assist or replace an expert. Examples of this kind of application include fingerprint classification, sonar signal classification, and flaw and crack detection in structures.

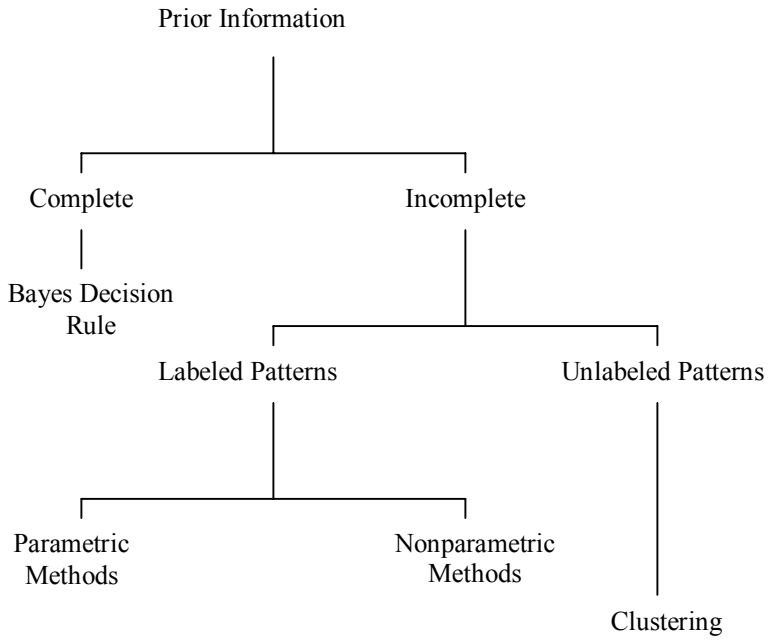



**Figure 10. Taxonomy of statistical pattern recognition techniques**

### *Pattern Recognition Approaches*

There are three basic approaches to pattern recognition (PR) – *statistical*, *structural*, and *neural*. In the context of data mining, statistical and neural approaches are useful and will be discussed. The neural approach is discussed under neural networks. We shall limit ourselves here to the statistical pattern recognition (SPR) approach. The statistical approach is rooted in statistical decision theory. This approach to pattern recognition treats each pattern (object or data record) as a point in an appropriate feature space. Similar patterns tend to lie close to each other, whereas dissimilar patterns, those from different classes, lie far apart in the feature space. The taxonomy of statistical pattern recognition techniques is shown in Figure 10. When complete information, i.e. a-priori probabilities and distribution parameters, about a pattern recognition task is available, the preferred PR approach is the *Bayes decision rule*, which provides optimal recognition performance. However, availability of complete information is rare and invariably a PR system is designed using a set of training or example patterns. This is analogous to data mining. When the example patterns are already classified, we say that we have *labeled patterns*. In such situations, *parametric* or *nonparametric* classification approaches are used. When example patterns do not have class labels, classification is achieved via clustering. The

clustering methods in SPR are the same as those discussed earlier under statistical data analysis and, hence, will not be discussed any further.

### Parametric Methods

The parametric methods are used when the form of class conditional densities is known. In practice, these densities are commonly assumed to be multivariate Gaussian. The Gaussian assumption leads to linear or quadratic classifiers. To implement these classifiers, the parameters of the class-conditional density functions are estimated using the available set of pre-classified training patterns.

### Nonparametric Methods

The nonparametric methods are used in those situations where the form of the underlying class conditional densities is unknown. There are two basic nonparametric approaches – *density estimation approach* and *posteriori probability estimation approach*. The most well-known example of the density estimation approach is the *Parzen window* technique where a moving window is used for interpolation to estimate the density function. The most well known example of the posteriori-probability estimation approach is the *k-nearest neighbor* (k-NN) method, which leads to the following rule for classification. Classify an unknown pattern to that class which is in majority among its k-nearest neighbors taken from the set of labeled training patterns. When k=1, this method is simply called the *nearest neighbor classifier*. It is easy to see that this classification rule is like a table look up procedure. The k-NN rule is a very popular classification method in data mining because it is purely a data-driven method, and does not imply any assumptions about the data. Furthermore, the k-NN rule is capable of producing complex decision boundaries. Figure 11 shows an example of this complexity for k=1. The main disadvantage of the k-NN rule is its computational burden, as an unknown pattern must be compared against every pattern in the training set, which can be exceedingly large. Many efficient implementation schemes are available in the literature to offset this drawback of the k-NN classifier.

All of the classification approaches – parametric and nonparametric, discussed so far are single shot approaches, i.e. a classification decision is made in one step. Decision tree classifiers in contrast offer a multistage decision methodology where stepwise exclusion of alternatives takes place to reach a final decision. Furthermore, the decision tree methodology does not require any assumption about class conditional densities. Consequently, tree classifiers are very popular in statistics, pattern recognition, machine learning, and neural networks. We shall discuss them later under machine learning.

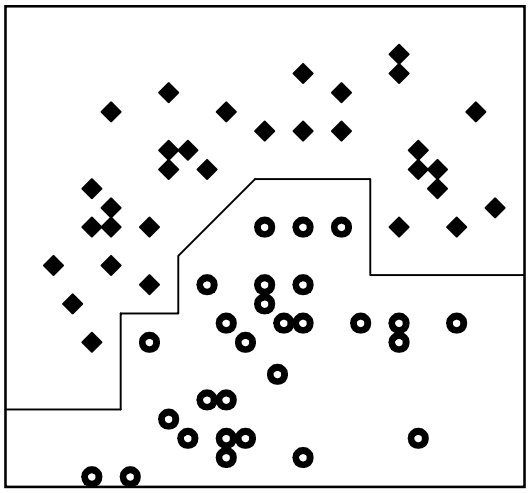


**Figure 11. An example of 1-NN rule decision boundary**

## Neural Networks

Artificial neural networks (ANNs) are currently enjoying tremendous popularity with successful applications in many disciplines. The interest in artificial neural networks is not new; it dates back to the work of McCulloch and Pitts, who about fifty years ago proposed an abstract model of living nerve cells or neurons. Since then, a very diverse set of researchers has been interested in ANNs because of a variety of different reasons. One of the main reasons that has led many to look at ANNs is their non-algorithmic learning capability to solve complex classification, regression, and clustering problems.

### *Neuron Model*

A neural network consists of a number of elementary processing units, called *neurons*. Figure 12 shows a typical neuron model. A neuron receives a number of inputs $x_1, x_2 \ldots, x_k$. Each input line has a connection strength, known as *weight*. The connection strength of a line can be *excitatory* (positive weight) or *inhibitory* (negative weight). In addition, a neuron is given a constant bias input of unity through the bias weight $w_0$. Two operations are performed by a neuron – summation and output computation. The summation operation generates a weighted linear sum of inputs and the bias according to the following equation:

$$net = \sum_{i=0}^{k} w_i x_i$$

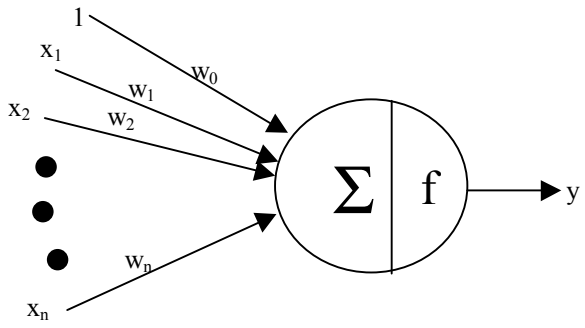



**Figure 12. A typical neuron model**

The output computation is done by mapping the weighted linear sum through an activation function:

$$y = f(net)$$

There are two basic types of activation functions – hard and soft. Hard activation function implies that the output of a neuron can exist in only one of the two possible states as shown below.

$$y = \mathrm{sgn}(net - w_0) = \begin{cases} 1, & net > 0 \\ 0, & net < 0 \end{cases}$$

Such neurons are generally called *discrete* neurons or *perceptrons*. Sometimes the two allowed states are 1 and -1. Such neurons are called *bipolar discrete* neurons. Neurons with soft activation functions are called *soft* neurons or *continuous* neurons. Two types of soft activation functions are used – *sigmoidal* and *hyperbolic tangent*. The sigmoidal activation function is given by

$$y = \frac{1}{(1 + \exp(-\alpha(net - w_0))}$$

which produces a continuously varying output in the range [0 1]. The hyperbolic tangent function for activation yields a continuous output in the range [–1 1]. This function is given by

$$y = \frac{(1 - \exp(-\alpha(net - w_0))}{(1 + \exp(-\alpha(net - w_0))}$$

The quantity $\alpha$ in the above equations determines the slope of the activation function.

### *Neural Network Models*

A neural network is a collection of interconnected neurons. Such interconnections could form a single layer or multiple layers. Furthermore, the interconnections could be unidirectional or bi-directional. The arrangement of neurons and their interconnections is called the *architecture* of the network. Different neural network models correspond to different architectures. Different neural network architectures use different learning procedures for finding the strengths (weights) of interconnections. Learning is performed using a set of training examples. When a training example specifies what output(s) should be produced for a given set of input values, the learning procedure is said to be a *supervised* learning procedure. This is the same as using a set of pre-classified examples in statistical data analysis and pattern recognition. In contrast, a network is said to be using an *unsupervised* learning procedure when a training example does not specify the output that should be produced by the network. While most neural network models rely on either a supervised or an unsupervised learning procedure, a few models use a combination of supervised and unsupervised learning.

There are a large number of neural network models, as shown in Figure 13, which have been studied in the literature. Each model has its own strengths and weaknesses as well as a class of problems for which it is most suitable. We will briefly discuss only three models here that are common in data mining applications. These are (1) *single-layer perceptron* (SLP) network; (2) *multiple-layer feed-forward network*; and (3) *self-organizing feature map*. For information on other models, the reader should refer to books on neural networks.

#### Single-Layer Perceptron Network Model

An SLP network consists of one or more neurons and several inputs. Each neuron may receive all or only some of the inputs. SLP networks are trained using supervised learning. The two well-known learning procedures for SLP networks are the *perceptron learning algorithm* and the *delta rule*. The perceptron training procedure is meant for classification learning and is one of the earliest neural training methods. It is an iterative error-correction procedure. In this method, a neuron is told to adjust its weights every time it makes a classification error on training examples. This process of weight adjustment continues until each neuron is able to produce the expected output. It has been shown that the

perceptron training procedure converges only when the underlying classification rules are linear. This is a major limitation of the perceptron training procedure, as many interesting problems have complex non-linear classification rules. There have been a few modifications suggested in recent years to the perceptron training procedure in order to use it in complex classification situations.

The delta rule in contrast is an error minimization procedure, which tries to determine the weights for each neuron using the gradient search procedure. Consequently, the delta learning rule has no convergence problem, but it has the drawback of occasionally producing locally minimum solutions instead of globally minimum solutions. The delta learning procedure is applicable to regression as well as classification tasks. However, the classification rules or the regression functions generated by delta learning for SLP networks are linear rules and thus do not yield acceptable performance for complex problems.

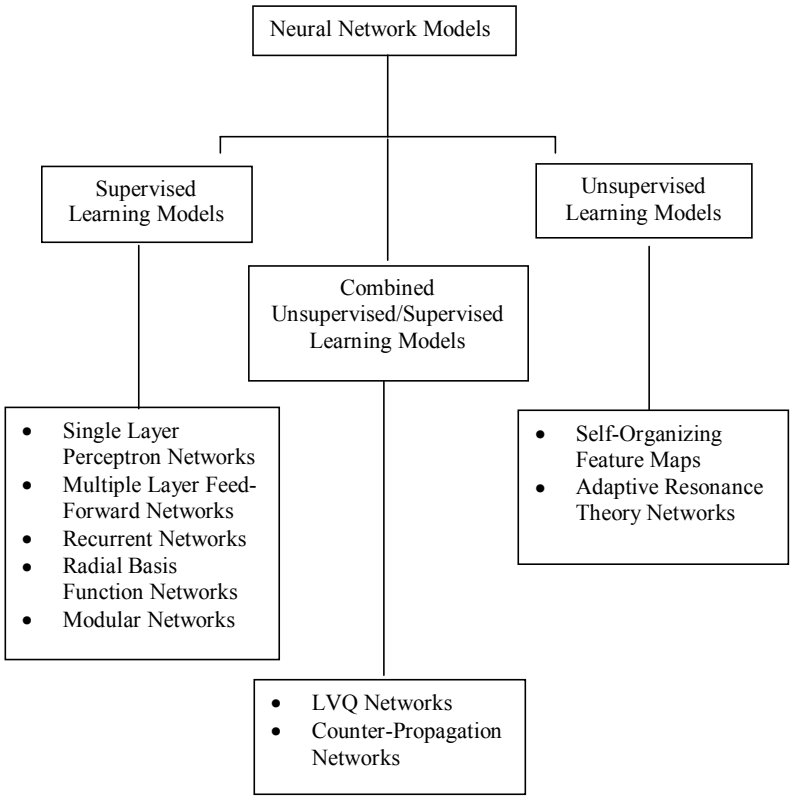



**Figure 13. A classification of neural network models**

## Multiple-Layer Feedforward Network Model

The multiple-layer feedforward neural network model is perhaps the most widely used neural network model. This model consists of two or more layers of interconnected neurons, as shown in Figure 14. Generally, all neurons in a layer are connected to all neurons in the adjacent layers through unidirectional links. The leftmost layer is called the *input* layer, the rightmost the *output* layer. The rest of the layers are known as *intermediate* or *hidden* layers. It is known that a three-layer feed-forward network is capable of producing an arbitrarily complex relationship between inputs and outputs. To force a feedforward network to produce a desired input-output relationship requires training the network in an incremental manner by presenting pairs of input-output mapping. This training is done following an error minimization process, which is a generalization of the delta learning rule. Hence, the training procedure is known as the *generalized delta rule*. However, the term *backpropagation* is more widely used to denote the error-minimization training procedure of multiple layer feedforward neural networks, which are often termed as *backpropagation neural networks* (BPN). One critical aspect of using a feedforward neural network is that its structure must match well with the complexity of the input-output mapping being learned. Failure to do so may result in the trained network not having good performance on future inputs. That is, the network may not generalize well. To obtain a good match, it is common to try several network structures before settling on one. Despite certain training difficulties, the multiple layer feedforward neural networks have been employed for an extremely diverse range of practical predictive applications with great success. These networks have also been used for sequence data mining. In such applications, data within a moving window of a certain pre-determined size is used at each recorded time instant as an input to capture the temporal relationships present in the data.

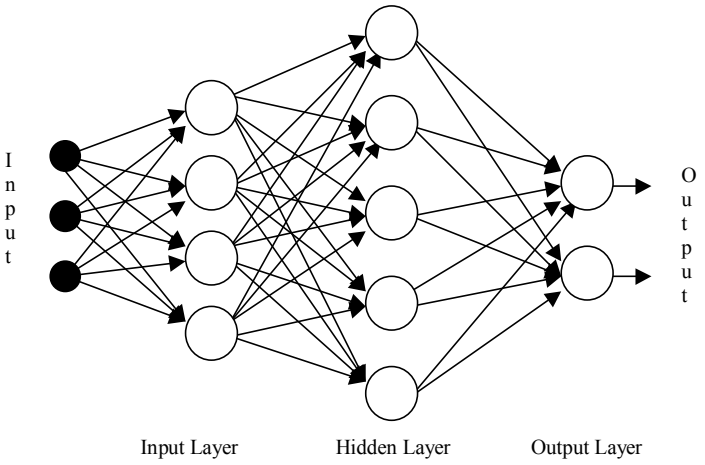



**Figure 14. An example of a multiple-layer feedforward neural network**

## Self-Organizing Feature Map Model

The backpropagation training procedure is applicable only when each example in the training set has an associated output response. As mentioned before, however, many important applications do not involve any dependent variable. In order to perform data analysis in such instances, a neural network must be able to self-organize, i.e. it should be able to analyze and organize data using its intrinsic features without any external guidance. Kohonen's self-organizing feature map (SOFM) is one such neural network model that has received large attention because of its simplicity and the neuro-physiological evidence of the similar self-organization of sensory pathways in the brain. The basic structure for a SOFM neural network is shown in Figure 15. It consists of a single layer of neurons with limited lateral interconnections. Each neuron receives an identical input. The network training is done following the *winner-takes-all* paradigm. Under this paradigm, each neuron competes with others to claim the input pattern. The neuron producing the highest (or smallest) output is declared the winner. The winning neuron and its neighboring neurons then adjust their weights to respond more strongly, when the same input is presented again. This training procedure is similar to k-means clustering of statistical data analysis, and suffers from the same merits and de-merits. Next to the feedforward neural networks, the SOFM networks are the most widely used neural networks. In addition to performing clustering, these networks have been used for feature extraction from raw data such as images and audio signals. A variation of SOFM is the *learning vector quantization* (LVQ) model that seeks to combine supervised and unsupervised modes of learning. In training LVQ, rough classification rules are first learned without making use of the known classification information for training examples. These rough rules are refined next using the known classification information to obtain finely tuned classification rules.

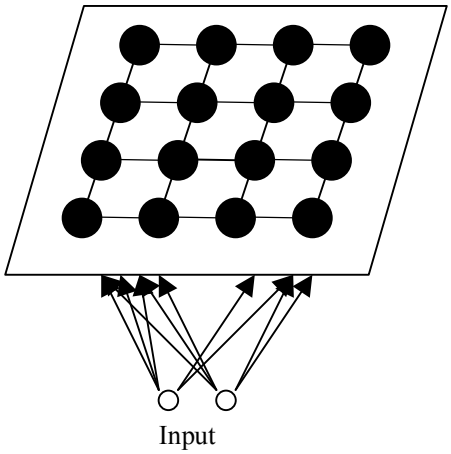



**Figure 15. Kohonen's SOFM model**

In addition to the above three models, other important models gaining widespread popularity include the *radial-basis function* (RBF) network model, and the *neural tree* model. The RBF network model consists of three layers and exhibits performance similar to multiple layer feedforward neural networks. However, the training time for RBF networks is much shorter. The neural tree is another class of feedforward neural networks. Such networks have limited interconnectivity, similar to a decision tree structure, but are more powerful than tree classifiers in terms of predictive capability. We will discuss neural trees under tree-based methods.

### Applying Neural Networks

A typical neural network application requires consideration of the following issues: model selection; input-output encoding; and learning rate. The choice of the model depends upon the nature of the predictive problem, its expected complexity, and the nature of the training examples. Since inputs and outputs in neural networks are limited to either [0-1] or [-1-1], the encoding of inputs and outputs requires careful considerations. Often, the encoding scheme for input-output has a large influence on the resulting predictive accuracy and training time. Another important factor in neural networks is the choice of learning rate. The learning rate determines the magnitude of weight changes at each training step. An improper learning rate (too small or too large) can cause an inordinately long training time, or can lead to sub-optimal predictive performance. Consequently, the use of neural networks is often called as an art.

## Machine Learning

Machine learning is a sub-discipline of artificial intelligence. The goal of machine learning is to impart computers with capabilities to autonomously learn from data or the environment. The machine learning community has been a major contributor of several new approaches to data mining. These include tree-based methods for learning, genetic algorithms, intelligent agents, and fuzzy and rough set-based approaches.

### *Tree-Based Methods*

The tree-based methods for classification and regression are popular across several disciplines – statistical data analysis, pattern recognition, neural networks, and machine learning. Many similar tree-based algorithms have been developed independently in each of these disciplines. There are several reasons for the popularity of tree-based methods. First, a tree-based method allows a complex problem to be handled as a series of simpler problems. Second, the tree

structure that results from successive decompositions of the problem usually provides a better understanding of the complex problem at hand. Third, the tree-based methods generally require a minimal number of assumptions about the problem at hand; and finally, there is usually some cost advantage in using a tree-based methodology in certain application domains.

## Decision Tree Classifiers

The term *decision tree* is commonly used for the tree-based classification approach. As shown in Figure 16, a decision tree classifier uses a series of tests or decision functions to assign a classification label to a data record. The evaluation of these tests is organized in such a way that the outcome of each test reduces uncertainty about the record being classified. In addition to their capability to generate complex decision boundaries, it is the intuitive nature of decision tree classifiers, as evident from Figure 16, that is responsible for their popularity and numerous applications. Like any other data mining methodology for classification, the use of decision tree classification requires automatic extraction of a tree classifier from training data. Several automatic algorithms exist for this purpose in the pattern recognition and machine learning literature. Most of these decision tree induction algorithms follow the top-down, divide-and-conquer strategy wherein the collection of pre-classified examples is recursively split to create example subsets of increasing homogeneity in terms of classification labels until some terminating conditions are satisfied.

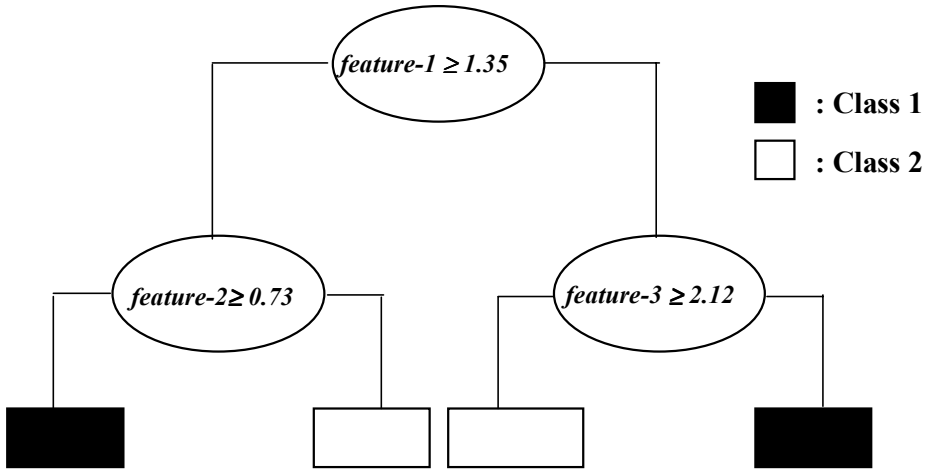



**Figure 16. An example of a decision tree classifier. The left branches correspond to positive outcomes and right branches to negative outcomes of the tests at internal tree nodes**

The top-down, divide-and-conquer decision-tree-induction methodology consists of four components. First, it needs a splitting criterion to determine the effectiveness of a given split on training examples. Second, it requires a method to generate candidate splits. Third, a stopping rule is needed to decide when to stop growing the tree. Finally, it needs a method to set up a decision rule at each terminal node. The last component is the easiest part of the tree induction process. The majority rule is often used for this purpose. Different decision tree induction methods differ essentially in terms of the remaining three components. In fact, the differences are generally found only in the splitting criterion and the stopping rule.

The three most well known decision-tree-induction methodologies in pattern recognition, statistical data analysis, and machine learning literature are AMIG, CART, and ID3. AMIG and ID3 both follow an information theory based measure, the *average-mutual information gain*, to select the desired partitioning or split of training examples. Given training examples from $c$ classes, and a partitioning $P$ that divides them into $r$ mutually exclusive partitions, the average mutual information gain measure of partitioning, $I(P)$, is given as

$$I(P) = \sum_{i=1}^{r} \sum_{j=1}^{c} p(r_i, c_j) \log_2 \frac{p(c_j / r_i)}{p(c_j)}$$

where $p(r_i, c_j)$ and $p(c_j / r_i)$, respectively, are the joint and conditional probabilities and $p(c_j)$ is the class probability. Using the maximum likelihood estimates for probabilities, the above measure can be written as

$$I(P) = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{n_{ij}}{N} \log_2 \frac{n_{ij} N}{N_i n_j}$$

where $n_j$ is the number of training examples from class $c_j$ and $n_{ij}$ is the number of examples of class $c_j$ that lie in partition $r_i$. The quantity $N$ is the total of all training examples of which $N_i$ lie in partition $r_i$. The split of training examples providing the highest value of $I(P)$ is selected. The CART procedure uses the *Gini index of diversity* to measure the impurity of a collection of examples. It is given as

$$G = 1 - \sum_{j=1}^{c} p^2(c_j)$$

The split providing maximum reduction in the impurity measure is then selected. The advantage of this criterion is its simpler arithmetic.

Determining when to stop top-down splitting of successive example subsets is the other important part of a decision-tree-induction procedure. The AMIG procedure relies for stopping on the following inequality that specifies the lower limit on the mutual information to be provided by the induced tree. The tree growing stops as soon as the accumulated mutual information due to successive splits exceeds the specified limit. CART and ID3 instead follow a more complex but a better approach of growing and pruning to determine the final induced decision tree. In this approach, the recursive splitting of training examples continues until 100% classification accuracy on them is achieved. At that point, the tree is selectively pruned upwards to find a best sub-tree according to some specified cost measure.

The generation of candidate splits at any stage of the decision-tree-induction procedure is done by searching for splits due to a single feature. For example in AMIG, CART*, and ID3, each top-down data split takes either the form of "Is $x_i \geq t$?" when the attributes are ordered variables or the form of "Is $x_i$ true?" when the attributes are binary in nature. The reason for using single feature splits is to reduce the size of the space of legal splits. For example with *n* binary features, a single feature split procedure has to evaluate only *n* different splits to determine the best split. On the other hand, a multi-feature-split procedure must search through a very large number of Boolean combinations, $2^{2^n}$ logical functions if searching for all possible Boolean functions, to find the best split. In recent years, many neural network-based methods, called *neural trees*, for finding decision tree splits have been developed. These methods are able to generate efficiently multi-feature splits. Most of these neural tree methods make use of modified versions of the perceptron training procedure. Although neural trees do not provide predictive capabilities identical to those exhibited by multi-layer feedforward networks, their popularity rests on the intuitive appeal of step-wise decision making.

* CART provides for limited Boolean and linear combination of features.

### Regression Trees

While the majority of tree-based methods are concerned with the classification task, i.e. the data mining situations where the dependent variable is a discrete variable, the tree-based approach is equally suitable for regression. In such situations, the tree is called a *regression tree*. The best known examples of the regression tree approach are CART (Classification and Regression Trees) and CHAID (Chi-Square Automatic Interaction Detection). CHAID is an extension of AID (Automatic Interaction Detection). The difference between the two is that AID is limited to applications where the dependent variable has a metric scale (interval or ratio). In contrast, CHAID is much broader in scope and can be applied even when the dependent variable happens to be a categorical variable; for example in market segmentation applications using brand preference data. The steps involved in developing a regression tree from training examples are similar to the decision-tree-induction process discussed earlier. The major difference is in the splitting criterion. Unlike using the average-mutual information gain type measures suitable for classification tasks, measures such as the least square regression error are used in building a regression tree. There are two main differences between CART and CHAID. First, the trees produced by CART are binary trees, i.e. each node divides data into two groups only. In CHAID, the number of splits can be higher. The second difference is that CHAID is a pure top-down procedure, while CART methodology also uses bottom-up pruning to determine the proper tree size.

## *Genetic Algorithms*

Genetic algorithms (GAs) belong to the broad area of evolutionary computing, which in recent years has gained widespread popularity in machine learning. *Evolutionary computing* is concerned with problem solving by applying ideas of natural selection and evolution. Essentially, genetic algorithms are derivative-free search or optimization methods that use a metaphor based on evolution. In this approach, each possible solution is encoded into a binary bit string, called a chromosome. Also associated with each possible solution is a fitness function, which determines the quality of the solution. The search process in genetic algorithms begins with a random collection of possible solutions, called the *gene pool* or *population*. At each search step, the GA constructs a new population, i.e. a new generation, using genetic operators such as crossover and mutation. Just like the natural evolution process, members of successive generations exhibit better fitness values. After several generations of genetic changes, the best solution among the surviving solutions is picked as the desired solution.

The application of the GA to a problem requires consideration of several factors, including the encoding scheme, choice of the fitness function, parent selection,

and genetic operators. The selection of an encoding scheme often depends upon how well it captures the problem-specific knowledge. The selected encoding scheme also influences the design of the genetic operators. The choice of fitness function is tied with the nature of the problem being solved. For classification problems, the fitness function may be the classification accuracy on the training data. On the other hand, the fitness function may be related to the mean square error for regression problems. The parent selection component of the GA specifies how the members of the present population will be paired to create the next generation of possible solutions. The usual approach is to make the probability of mating dependent upon a member's value of the fitness function. This ensures that members with better fitness values reproduce, leading to survival of the fittest behavior. Crossover and mutation operators are the two genetic operators that determine the traits of the next generation solutions. The *crossover operator* consists of interchanging a part of the parent's genetic code. This ensures that good features of the present generation are retained for future generations. In *one-point crossover operation*, a point on the genetic code of a parent is randomly selected and two parent chromosomes are interchanged at that point. In *two-point crossover operation*, two crossover points are selected and the genetic code lying between those points is interchanged. Figure 17 illustrates these two crossover operations. While crossover operators with many more points can be defined, one and two-point crossover operators are typically used in GAs.

The *mutation operator* consists of randomly selecting a bit in the chromosome string and flipping its value with a probability equal to a specified mutation rate. An example of mutation is shown in Figure 18. The mutation operator is needed because crossover operation alone cannot necessarily provide a satisfactory solution, as it only involves exploiting the current genes. Without mutation, there is a danger of obtaining locally optimum solutions. It is common to use a very low mutation rate to preserve good genes. Furthermore, a high mutation rate produces behavior similar to random search, and is, therefore, not used. In addition to selection, crossover, and mutation, GAs often use additional rules to determine the members of the next generation. One popular rule in this regard is the *principle of elitism*, which requires a certain number of best members of the present population to be cloned.

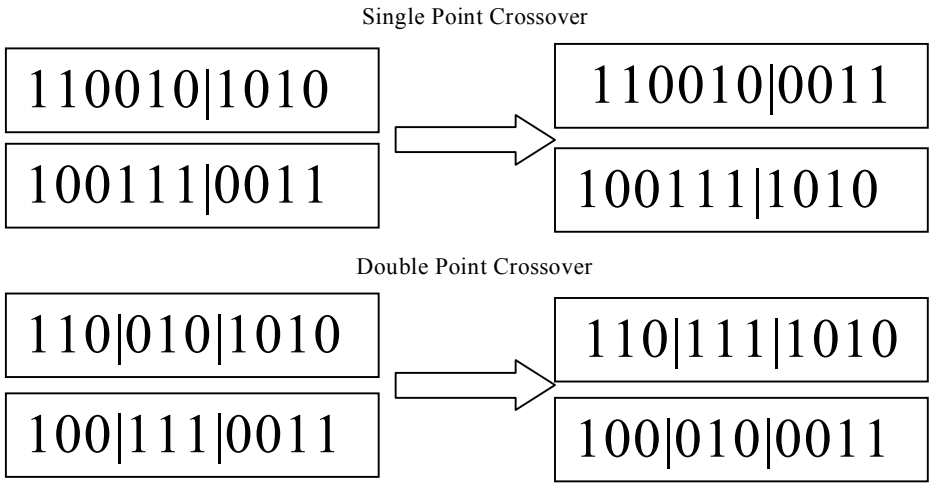



**Figure 17. Examples of crossover operations**

While GAs are often employed on their own to solve problems, it is not uncommon to see GAs being used in conjunction with other data mining methods. For example, GAs have been used in the decision tree methodology to determine data partitions. In neural networks, these have been used for network pruning or for finding the optimal network configuration. In addition to the attractiveness of the evolution paradigm, there are several other reasons that have contributed to the overall popularity of GAs. First, GAs are inherently parallel, and thus can be implemented on parallel processing machines for massive speedups. Second, GAs are applicable to discrete as well as continuous optimization problems. Finally, GAs are less likely to get trapped in a local minimum. Despite these advantages, these algorithms have not yet been applied to very large-scale problems. One possible reason is that GAs require a significant computational effort with respect to other methods, when parallel processing is not employed.

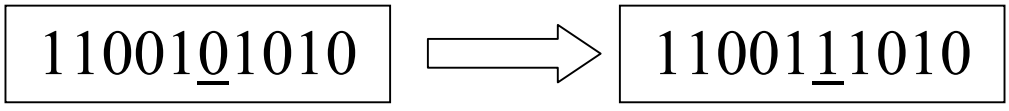



**Figure 18. An example of a mutation operation**

### *Intelligent Agents*

Intelligent-agents (IA) is another machine learning approach that is rapidly becoming very popular in several applications, including data mining. Like GAs, the IA approach is also a population-based approach. However, unlike GAs where a member of the population interacts with another member to generate a new solution, the intelligent agents approach consists of finding solutions through social interaction among population members, each known as an *intelligent agent*. Each intelligent agent behaves like an "individual" with different intelligent agents having different habits and behavior patterns. The habits and behavior patterns of agents are assigned based on existing data. As these simulated agents interact with each other, they adjust and adapt to other agents. Monitoring of behavior of a large collection of agents yields valuable information hidden in the data. For example, letting each agent simulate the buying pattern behavior of a consumer and running IA simulation for a sufficiently long time, we can count how many units of the new products are sold and at what rate. The information thus coming out of simulation can tell us whether the new product will be successful or not.

### *Fuzzy and Rough Sets*

All of the methods discussed thus far have no provision for incorporating the vagueness and imprecision that is common in everyday life. The concepts of fuzzy sets and rough sets provide this provision and are useful in many applications. The fuzzy set concept was introduced by Lotfi Zadeh in 1965 and since then has been applied to numerous applications. The last few years have especially seen a rapid increase in interest in fuzzy sets. Unlike conventional sets where an object either belongs to a set or not, objects in fuzzy sets are permitted varying degrees of memberships. As an example, consider the set of "tall" persons. In the conventional approach, we would define a cutoff height, say 5'10'', such that every person with height equal to or greater than the cutoff height would be considered tall. In contrast, the fuzzy set of "tall" people includes everyone in it. However, each person belongs to the "tall" fuzzy set with a different grade of membership in the interval of 0-1. The main advantage of fuzzy sets is that it allows inference rules, e.g. classification rules, to be expressed in a more natural fashion, providing a way to deal with data containing imprecise information. Most of the fuzzy set-based methods for data mining are extensions of statistical pattern recognition methods. The application of fuzzy set-based methods, e.g. the fuzzy k-means clustering procedure, to clustering is especially appealing. The statistical clustering techniques assume that an item can belong to one and only one cluster. Often in practice, no clear boundaries exist between different clusters, as shown in the several examples of

Figure 19. In such situations, the notion of fuzzy clustering offers many advantages by letting each object have a membership in each cluster.

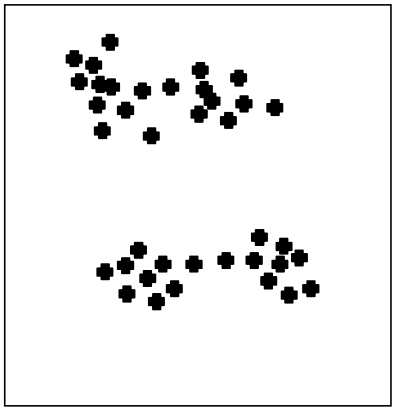


**Figure 19. Examples of fuzzy clusters**

The concept of rough sets is relatively new; it was introduced by Pawlak in 1982. The basic philosophy of rough set methodology is that lowering the degree of data precision makes data regularities easier to find and characterize in terms of rules. However, this lowering of precision is not without risk, as it can lead to loss of information and differentiation between different objects. The theory of rough sets provides tools to deal with this problem by letting roughness vary in a controlled manner to find a data precision level that ensures sufficient pattern discrimination. Often, rough sets are confused with fuzzy sets as both deal with imperfect knowledge. However, both deal with different aspects of imperfection. The imperfection dealt with in fuzzy sets is with respect to objects within the same class. In contrast, rough sets deal with imperfections between groups of objects in different classes. The main application of rough sets so far has been in classification tasks. Furthermore, most of the reported applications have been for relatively small size problems. However, this is expected to change as rough sets gain more popularity and commercial software starts becoming available.

## Conclusion

In this section we provided a brief introduction to important data mining methods from the different disciplines of statistical data analysis, pattern recognition, neural networks, and machine learning. While discussing different methods, we have tried to highlight important aspects of each method. There are two related questions which are yet to be answered. The first question concerns how to go about evaluating the performance of a data mining method. The second question is which data mining method is better, or how to choose a

method for a given data mining task. Answers to these two questions are discussed in the next sections.

# 6

# HOW GOOD IS THE MINED MODEL

---

Irrespective of the data mining methodology selected for a particular task, it is important to assess the quality of the discovered knowledge. There are two components to this assessment – *predictive accuracy* and *domain consistency*. Predictive accuracy implies how well the discovered classification or regression model will perform on future records. Domain consistency means whether the discovered knowledge is consistent with other domain knowledge that the end user might have. Since the model assessment based on domain consistency is problem specific, we will discuss only the predictive assessment component in the following.

The goal of assessing predictive accuracy is to find the true error rate - an error rate that would be obtained on an asymptotically large number of future cases. The most common approach for assessing predictive accuracy is the train-and-test error rate estimation approach. In this approach, the entire data set is randomly divided into two groups. One group is called the training set and the other the testing set. The selected data mining methodology is applied to the test set to build the predictive model. Once the model is built, its performance is evaluated on the test set to determine the test-sample error rate, which provides a very good indication of the predictive performance on future records. When the test set consists of 5000 test cases, selected independent of the training set, then the test-sample error rate is considered virtually the same as the true error rate. For test sets of smaller size, the test-sample error rate is considered a slightly optimistic estimate of the true error rate. It is therefore a good practice to have as large a test set as possible. When this is not possible due to the small size of the total data set, a situation not likely to occur in data mining, various re-sampling techniques such as cross-validation and boot-strapping are used to obtain the predictive accuracy.

# 7

# WHICH DATA MINING METHODOLOGY IS BEST?

Having described a number of methodologies, a natural question to ask is "Which data mining methodology is best?" This is a difficult question to answer. All that can be said is that there is no universally best methodology. Each specific application has its own unique characteristics that must be carefully considered and matched with different methodologies to determine the best method for that specific application. However, there are a few general remarks that can be made with respect to different methodologies. One can also look at the performance of different methods on some typical data sets that have been well studied by different researchers and are often used as benchmarks. Finally, one can rate different methods with respect to several useful factors to select the most appropriate method for a specific data mining task.

## Statistical Methods

The strength of statistical methods is that these methods come from a mature field. The methods have been thoroughly studied and have a highly developed theoretical foundation. In consequence, the knowledge discovered using these methods is more likely to be accepted by others. However, the weakness of the statistical methods is that they require many assumptions about the data. Furthermore, these methods require a good statistical knowledge on the part of an end-user to properly interpret the results. Another weakness of the statistical methods is that they do not scale well for large amounts of non-numeric data.

## Pattern Recognition Methods

Many of the statistical pattern recognition methods share the same strengths and weaknesses as those of the statistical methods. However, the case of the k-NN rule is entirely different. It is a true data-driven method, which does not require any assumption about the data. Its decision making model is also intuitively understandable. The weakness of this method is its predictive capability; it can not provide accuracy at par with other methods. This weakness of the k-NN rule, however, is offset by its zero learning time.

## Neural Network Methods

The neural network and machine learning methods are relatively new. In many cases, these techniques have not been theoretically well studied. However, the main strength of neural network and machine learning methods is that they require relatively fewer assumptions about data, and are thus data-driven. The neural network methods are criticized mainly on two counts. First, the difficulty of replicating the results, and second, the concern about interpretability of the behavior of a trained neural network. The concern about replicating the results arises due to the algorithmic nature of neural network training. Since several components, such as the random initial weights and the presentation order of training examples, of this algorithmic process are not necessarily duplicated every time a network is trained, it is generally difficult to replicate a network performance. The interpretability concern about neural network is associated with the *black box* image of neural networks. Since the explanations about a neural network behavior are stored in its weights, they are not easily comprehensible to a user. This has led to the black box view of neural networks. This view is further compounded by the fact that obtaining a proper neural network model for an application involves iteratively identifying proper network size, learning rate, and stopping criteria. A new user is often overwhelmed by this iterative process. In consequence, the acceptance of neural networks is occasionally hard to come by. However, the black box image of neural networks is slowly changing as researchers are finding ways to uncover the knowledge captured by a neural network.

## Tree-Based Methods

In comparison to other methods, the decision tree methodology appears to offer most advantages – a competitive predictive accuracy, minimal assumptions about data, better computational efficiency, and good scalability. Furthermore, the tree-based methods are able to deal with all types of variables. Consequently, this methodology is highly popular in data mining. The decision tree methodology also yields a mined model that is extremely easy to understand and use. On the negative side, the performance of decision trees is occasionally unsatisfactory because of single feature splits for data partitioning. Another criticism of decision tree methods is their greedy tree growing methodology, which often hides better models from discovery. An example of this deficiency of decision tree methods is shown in Figure 20 which shows two decision trees obtained for the data set of Table 1. The data consists of three binary independent variables $x_1$, $x_2$, $x_3$ and one dependent variable $f$. The tree on the left is obtained by one of the current decision tree methods; the tree on the right is produced by a global search, which is easy in this case because of the problem size. It is easy to see that the tree on the right captures the hidden relationships much better. It

should be noted that many new decision tree induction methods combining machine and neural learning are being developed to address the above drawbacks of the decision tree methodology. Their availability in commercial software packages is expected to lead to more usage of the tree methodology in data mining.

**Table 1**

**An Illustrative Data Set**

| $x_1$ | $x_2$ | $x_3$ | $f$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

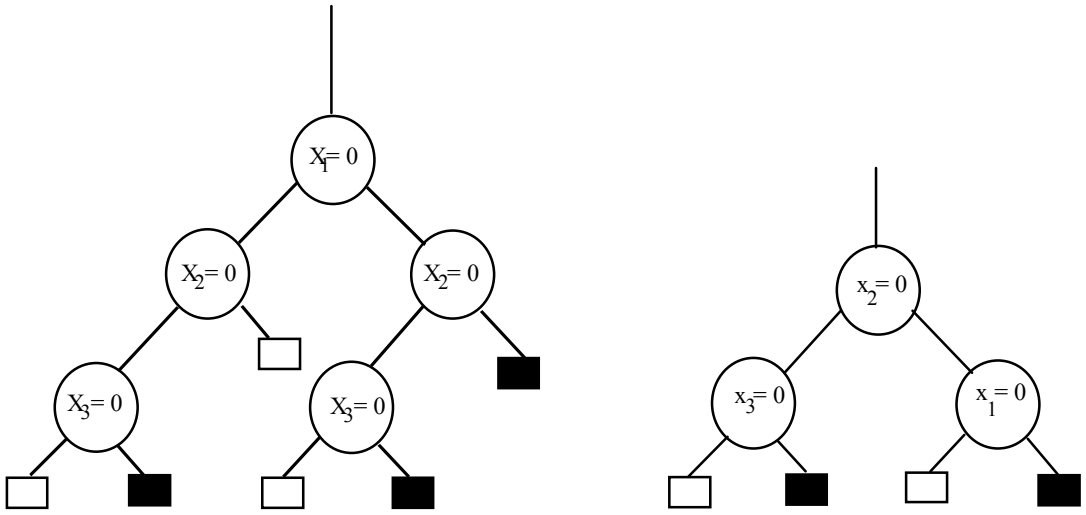



**Figure 20. Two decision trees for the data of Table 1**

## Performance Comparison Using Benchmark Data Sets

Continuing with the question "Which method is best?" let us take a look at the performance of some of the representative methods on some benchmark data sets. We will do this using seven data sets of different size and characteristics. These data sets are (1) Iris Data; (2) Appendicitis Data; (3) Thyroid Data; (4) Cancer Data; (5) Sonar Data; (6) Glass Data; and (7) Dot Matrix Data. All of these data sets are publicly available and have been modeled using different methods. The iris data set consists of 150 measurements on four continuous independent variables. The 150 measurements are made on three classes of objects, 50 per class. The appendicitis data consists of seven diagnostic tests performed on 106 patients suspected of having appendicitis. The thyroid data consists of over 6,000 cases with 22 measurements per case. It has three classes of patients. The cancer data has 286 records with nine tests per record and two classes. The sonar data has 208 sonar returns from two different types of objects. Each return is described in terms of 60 measurements. The glass data contains composition information about six different types of glasses. Each item is described in terms of nine measurements. The dot matrix data contains 2,000 records for 10 different types of objects. Each object is described by a set of 35 binary measurements. Thus, we see that these seven data sets together represent a variety of complex classification tasks.

As representatives of different methodologies, we use four of the most popular data mining methods. The first is linear discriminant analysis (LDA) representing the statistical data analysis methodology. The statistical pattern recognition methodology is represented by the k-NN rule with k equal to one. The third method is the backpropagation network (BPN) from neural networks. Finally, the machine learning methodology is represented by the decision tree classifier (DT). The performance of these different methods on the chosen data sets is shown in Table 2. These performance results are derived from the reported research literature. The performance in each case represents the test-sample-error rate.

As results of Table 2 show, no single method is consistently able to outperform other methods. The linear discriminant analysis works best for the iris and appendicitis data sets. The decision tree method gives best results for the thyroid and cancer data sets. The backpropagation network yields best performance for the sonar, glass and dot matrix data sets. Two other observations can be made from Table 2. First, the k-NN rule is never able to outperform other methods, and second, the decision tree classifier performs very poorly in some cases. This behavior of decision tree classifiers is in line with the earlier remark that the use of single feature splits can occasionally lead to poor predictive performance. The inability of the k-NN rule to outperform the other methods is also in line with the

theoretical properties of k-NN classification. Despite this, k-NN is a popular method, as its use requires a minimal design effort on the part of a user.

**Table 2**

**The Performance of Different Methods**

| | LDA | 1-NN | BPN | DT |
|---|---|---|---|---|
| Iris | 0.020 | 0.040 | 0.033 | 0.047 |
| Appendicitis | 0.132 | 0.179 | 0.142 | 0.151 |
| Thyroid | 0.0615 | 0.0473 | 0.0146 | 0.0064 |
| Cancer | 0.294 | 0.347 | 0.285 | 0.229 |
| Sonar | 0.185 | 0.174 | 0.107 | 0.248 |
| Glass | - | 0.191 | 0.079 | 0.173 |
| Dot Matrix | - | 0.030 | 0.024 | 0.214 |

## Other Comparison Factors

Another way to answer the question "Which method is best?" is to rate different methods with respect to several factors other than the predictive accuracy. One such rating is shown in Table 3. This rating is based on seven factors. The first two factors *scalability* and *dimensionality* refer to the nature of data mining input. Scalability implies how well a method is able to deal with very large number of records. Dimensionality implies the ability of a method to handle records of large size, i.e. large numbers of variables. Since the end-goal of data mining is to provide valuable information for decision support, it is important that the knowledge uncovered by data mining be comprehensible to the user. The *model comprehension factor* accounts for this aspect of a data mining method. The next two factors, *CPU load* and *memory load*, reflect the computational aspects of a method in its learning or discovery mode. The remaining two factors reflect the same thing, but during the predictive mode or the actual application mode, when the results of data mining are being applied.

As Table 3 shows, the decision tree methodology has the best ranking overall in terms of the different factors. Linear discriminant analysis comes next, followed by the backpropagation method from the neural network approach. The k-NN rule has extreme ratings, good or poor, on each of the seven factors.

**Table 3**

**Ratings of Different Methods**

| | LDA | k-NN | BPN | DT |
|---|---|---|---|---|
| Scalability | Average | Poor | Average | Good |
| Dimensionality | Average | Poor | Average | Good |
| Model Comprehension | Average | Good | Poor | Good |
| CPU Load (LM) | Average | Good | Poor | Average |
| Memory Load (LM) | Average | Good | Average | Average |
| CPU Load (PM) | Good | Poor | Average | Good |
| Memory Load (PM) | Good | Poor | Good | Good |

## Conclusion

In this section, we have provided a discussion on the relative strengths and weaknesses of the different data mining methods. The predictive accuracy of four representative methods on seven benchmark data sets was also provided to show that no single method is consistently better than the other methods. We also presented a set of seven factors, in addition to predictive accuracy, that can be used to rate different methods. It is hoped that the information presented in this section can help each user to choose the most appropriate method for the data mining task at hand.

# 8

# SUMMARY

This report has provided an introduction to data mining and the core technologies behind it. Data mining is not a single technique but rather a collection of techniques that allows us to reach out to valuable hidden information present in large business databases. Although computer-based learning technologies - neural networks, pattern recognition, decision trees - form the core of data mining, there is more to data mining than simply using a neural network or decision tree algorithm. It is an interactive and an iterative process of several stages driven with the goal of generating useful business intelligence. The success of a data mining effort is not determined by the accuracy of the mined predictive or classification model but by the value of the model to the business. Effective data mining not only requires a clear understanding of the business issues involved but also needs an inordinate amount of data preparation - identifying important variables, cleaning data, coding and analyzing data. Without proper data preparation, data mining is apt to generate useless information. The proverbial garbage-in, garbage-out is never more apt than in a data mining application without a proper understanding of the business aspects of the problem and careful data preparation.

Data mining has emerged as a strategic tool in the hands of decision-makers to gain valuable business intelligence from their corporate data. Such business intelligence is helping companies improve their operations in many critical areas including marketing, product development and customer services. In consequence, the applications of data mining continue to grow. Banking and telecommunication industries are at the forefront of using data mining technology with impressive performance results. If the experience of the deregulated telecommunication companies is an indicator of things to come, data mining is going to prove very valuable to utilities in staying competitive and generating new products in the deregulated environment.

# 9

# FURTHER READING

## Books and Edited Volumes


P. Adriaans and D. Zantinge, *Data Mining*, Addison-Wesley Longman, London, England, 1996.

R. Beale and T. Jackson, *Neural Computing: An Introduction*, Adam Hilger, Bristol, UK, 1990.

M.J. Berry and G. Linoff, *Data Mining Techniques*, John Wiley and Sons, Inc., New York, NY, 1997.

L. Breiman, J. Friedman, R. Olshen, and C.J. Stone, *Classification and Regression Trees*, Wadsworth Int'l Group, Belmont, CA, 1984.

U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth and R. Uthurusamy (Eds.), *Advances in Knowledge Discovery and Data Mining*, AAAI Press/MIT Press, Cambridge, MA, 1996.

D.E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, MA, 1989.

J.F. Hair, Jr., R.E. Anderson, R.L. Tatham and W.C. Black, *Multivariate Data Analysis*, Macmillan Publishing Company, New York, NY, 1987.

W. Inmon, *Building the Data Warehouse*, John Wiley & Sons, New York, NY, 1996.

C.G. Looney, *Pattern Recognition Using Neural Networks*, Oxford University Press, New York, NY, 1997.

R. Mattison, *Data Warehousing*, McGraw-Hill, New York, NY, 1996.

J.H. Myers, *Segmentation and Positioning for Strategic Marketing Decisions*, American Marketing Association, Chicago, Illinois, 1996.

K. Parsaye and M. Chignell, *Intelligent Database Tools and Applications*, John Wiley, New York, NY, 1993.

Z. Pawlak, *Rough Sets: Theoretical Aspects of Reasoning About Data*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1991.

G. Piatetsky-Shapiro and W. Frawley (Eds.), *Knowledge Discovery in Databases*, AAAI Press/MIT Press, Cambridge, MA, 1991.

S. Sestito and T.S. Dillon, *Automated Knowledge Acquisition*, Prentice-Hall, Sydney, Australia, 1994.

S.M. Weiss and C.A. Kulikowski, *Computer Systems That Learn*, Morgan Kaufmann Publishers, San Mateo, CA, 1991.

J.M. Zurada, *Artificial Neural Systems*, West Publishing, St. Paul, MN, 1992.

## Research Articles

R. Agarwal, T. Imielinski, and A. Swami, "Database Mining: A Performance Perspective," *IEEE Trans. Knowledge and Data Engineering*, Vol. 5, pp. 914-925, December 1993.

U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "The KDD Process for Extracting Useful Knowledge from Volumes of Data," Communications of the ACM, Vol. 39, pp. 27-34, November 1996.

W.H. Inmon, "The Data Warehouse and Data Mining," Communications of the ACM, Vol. 39, pp. 49-50, November 1996.

R.A. Jacobs, M.I. Jordan, S.J. Nowlan, and G.E. Hinton, "Adaptive Mixtures of Local Experts," *Neural Computation*, Vol. 3, pp. 79-87, 1991.

H. Lu, R. Setiono and H. Liu, "NeuroRule: A Connectionist Approach to Data Mining," in *Proceedings of the 21st VLDB Conference*, pp. 478-489, Zurich, Switzerland, September 1995.

J.R. Quinlan, "Induction of Decision Trees," *Machine Learning*, Vol. 1, pp. 81-106, 1986.

I.K. Sethi and J.H. Yoo, “Design of Multicategory Multifeature Split Decision Trees Using Perceptron Learning,” *Pattern Recognition*, Vol. 27, No. 7, pp. 939-947, 1994.

I.K. Sethi and J.H. Yoo, "Symbolic Mapping of Neurons in Feedforward Networks," *Pattern Recognition Letters*, Vol. 17, No. 10, pp. 1035-1046, 1996.

I.K. Sethi, J.H. Yoo, and C. Brickman, “Extraction of Diagnostic Rules Using Neural Networks,” *Proceedings 6th Annual Symposium on Computer-Based Medical Systems*, pp. 217-222, Ann Arbor, June 1993.

R. Srikant and R. Agarwal, “Mining Generalized Association Rules,” in *Proceedings of the 21st VLDB Conference*, pp. 407-419, Zurich, Switzerland, September 1995.

L.A. Zadeh, "Fuzzy Sets," *Information and Control*, Vol. 8, pp. 338-353, 1965.

## Trade Magazine Articles

P. Coy, "He Who Mines Data May Strike Fool's Gold," *Business Week*, pp. 40, June 16, 1997.

H. Edelstein, " Mining For Gold," *Information Week*, April 21, 1997.

A.S. Kay, "Decisions, Decisions: How to Maintain Customer Loyalty -- Telcos Invest in Decision-Support Tools," *Information Week*, July 07, 1997.

R.P. Lippmann, "An Introduction to Computing with Neural Networks," *IEEE ASSP Magazine*, Vol. 4, pp. 4-22, April 1987.

M. Marshall, " IBM's Data-Miner Sifts Through Data to Find Fraud,” *Information Week*, April 29, 1996.

L. Nadile, "Data Mining You Can Afford," *Information Week*, March 24, 1997.

J. Teresko, “Data Warehouses: Build Them for Decision-Making Power,” *Industry Week*, Vol. 245, No. 6, pp. 43-52, March 18, 1996.

N. Wreden, "The Mother Lode -- Data Mining Digs Deep For Business Intelligence," *Information Week*, February 17, 1997.

# APPENDIX A

# COMMERCIAL DATA MINING TOOLS

---

This summary of currently available commercial, off-the-shelf data mining products is being provided to help readers better understand what products are available and some of their features. It is not intended to endorse or critique any specific product. Potential users will need to decide for themselves the suitability of each product for their specific application and data mining environment. It is primarily intended as a starting point from which users can obtain more information. There is a constant stream of new products appearing on the market and hence this list is by no means comprehensive.

Generally, there are two types of products - *single strategy tools* and *multiple strategy tools*. A single strategy tool provides only a single approach to data mining - be it classification, regression, or one of the other approaches. Such tools are usually based on a single methodology, for example tree classification for data mining. As expected, single strategy tools are relatively inexpensive. In contrast, multiple strategy tools utilize two or more data mining methodologies and support the entire range of data mining approaches. The multiple strategy tools range in price from less than few thousand dollars for smaller, stand-alone versions to several tens of thousand dollars for remote data mining. Many vendors offer an integrated line of products including data warehousing and OLAP. The cost of such products can run to several millions. Many of the products make use of sampling to minimize storage and computational effort. Sampling does not use all data records for mining; only a randomly selected subset is used.

## Listing of Data Mining Products

A listing of the data mining products is presented in Table-A1. Only the general-purpose data mining products are listed. Many vendors also sell specialized products targeted at specific applications. Many vendors offer their product bundled with technical services as a complete customized solution to data mining. For each product, we specify its operating architecture, platforms for

which it is available, the data mining methodologies it employs, and the data mining tasks it performs. A brief description for each product also follows. More details for each product can be obtained from its listed web site given at the end of the product description. Although most product descriptions imply a load-and-go scenario for data mining, it is good to remember that extracting useful hidden information from large data is a complex task. The success of a data mining endeavor depends largely upon how well a user is prepared for the task.

### *AgentBase/Marketeer*

AgentBase/Marketeer is the industry's first second-generation data mining product, according to its creators. It is based on emerging intelligent-agents technology. The system comes with a group of wizards to guide a user through different stages of data mining. This makes it easy to use. AgentBase/Marketeer is primarily aimed at marketing applications. It uses several data mining methodologies whose results are combined by intelligent-agents. It runs on Windows 95, Windows NT and Sun workstations running Solaris. It can access data from all major data sources. (www.dazsi.com)

### *BusinessMiner*

BusinessMiner is a single strategy, easy to use tool based on decision trees. It can access data from multiple sources including Oracle, Sybase, Informix, Microsoft SQL Server, CA-Ingres, Teradata and RedBrick. BusinessMiner runs on all Windows platforms. It can be used stand-alone or in conjunction with an OLAP tool from the same vendor. (www.busobj.com)

### *Clementine*

Clementine is a comprehensive toolkit. It uses neural networks and rule induction methodologies for data mining. It can access data from Oracle, Ingres, Sybase and Informix. The toolkit includes data manipulation and limited visualization capabilities. It runs on Windows 95, Windows NT and UNIX platforms. (www.isl.co.uk/clem.html)

### *Darwin*

Darwin is an integrated, multiple strategy tool that uses neural networks, classification and regression trees, nearest neighbor rule, and genetic algorithms to provide a variety of data mining approaches. These techniques are implemented in an open, client/server architecture with a scalable parallel computing implementation. Darwin can access data from a variety of networked

data sources including all major relational databases, and provides support for many data manipulation and transformation tasks needed to prepare data for mining. It also provides a tool to interpret results. Darwin runs on single/multiple CPU, UNIX servers. A user can use a desktop client (UNIX workstation or a Windows platform) to access Darwin, run models and view results. (www.think.com)

### *Data Mining Suite*

Data Mining Suite is a comprehensive and integrated set of data mining tools. The main tools are IDIS (Information Discovery System) for finding classification rules, IDIS-PM (Predictive Modeler) for prediction and forecasting, and IDIS-Map for finding geographical patterns. The Data Mining Suite has a three-tiered client/server architecture. It runs on all major platforms. It works on several databases such as Oracle, Sybase, Informix, etc. (www.datamining.com)

### *Data Surveyor*

Data Surveyor is a single strategy (classification) tool. It consists of two components - a front-end and a back-end. The front-end is responsible for data mining, which is performed via the tree methodology. The back-end consists of a fast, parallel database server. The data in this server is loaded from the user's databases. The back-end runs on parallel UNIX servers. The front-end can run on UNIX and Windows platforms. (www.cwi.nl/~marcel/ds.html)

### *DataBase Mining Marksman*

Marksman is a single methodology tool; it is based on neural networks. It provides a number of useful data manipulation features. Marksman runs on PCs in Windows environment. Its design is optimized for the database analysis needs of direct marketing professionals. It is available as a combination of hardware (a standard PC with an accelerator board containing 16 parallel processors) and software. (www.hnc.com)

### *DataMind*

DataMind's architecture consists of two components - DataMind DataCruncher for server-side data mining, and DataMind Professional for client-side specification and viewing of results. The data mining methodology used by DataMind is based on intelligent-agents. It can implement classification, clustering, and association approaches of data mining. DataMind Professional runs on Windows95, Windows 3.1, Intel NT 3.5.1 and Intel NT 4.0. DataMind

DataCruncher runs on Intel NT 3.5.1, Intel NT 4.0, HP-UX 10.x, Sun Solaris 2.5.x, SGI IRIX 6.2 and IBM AIX 3.2.5. It can be set up to mine data locally or on a remote server. The data can come from any major relational database. (www.datamind.com)

### *Datasage*

Datasage is a comprehensive data mining product whose architecture incorporates a data mart in its data mining server providing a fast, flexible and scalable system for data mining. The user accesses Datasage through an interface operating as a thin client, using either a Windows client or a Java enabled browser client. The data mining approach used in Datasage is proprietary. (www.cirrusrec.com)

### *Decision Series*

Decision Series is a multiple strategy tool that comes bundled with knowledge engineering services. It uses neural networks, clustering, and genetic algorithms to perform data mining. It can be operated on scalable, parallel platforms to provide speedy solutions. It runs on standard industry platforms from HP, SUN, and DEC, and supports Oracle, Informix, and Sybase. (www.neovista.com)

### *Discovery*

Discovery is a multiple strategy data mining tool. It runs on Windows NT with SQL server. It is part of an integrated system, Pilot Decision Support Suite, which provides OLAP environment. (www.pilotsw.com)

### *Intelligent Miner*

Intelligent Miner is an integrated and comprehensive set of data mining tools. It uses decision trees, neural networks and clustering. Most of its algorithms have been parallelized for scalability. A user can build models using either a GUI or an API. A weakness of Intelligent Miner is its tight coupling with DB2, which also must be installed. (www.software.ibm.com)

### *KATE Tools*

KATE is a single strategy tool consisting of four tools: KATE-Editor, KATE-CBR, KATE-Datamining and KATE-Runtime. KATE currently runs on Windows 95 and Windows NT; it is being ported to UNIX platform. KATE is open to several

databases and can be integrated with SAP R/3 service management. (www.acknosoft.com)

### *Knowledge Discovery Workbench*

Knowledge Discovery Workbench is based on the Clementine data mining tool from ISL of UK and a data server from NCR. It thus provides a comprehensive set of tools. Knowledge Discovery Workbench comes bundled with technical services from NCR. (www.ncr.com)

### *Knowledge Seeker*

Knowledge Seeker is a single strategy desktop or client/server tool relying on tree-based methodology for data mining. It provides a nice GUI for model building and letting the user explore data by splitting a selected tree node or even forcing a particular split that might be of interest. It allows users to export the discovered data model as text or as SQL or Prolog languages. It runs on Windows and UNIX platforms, and can accept data from a variety of sources. It is also available as an integrated component of a data mining solution from NCR. (www.angoss.com)

### *MineSet*

Mineset is a comprehensive tool for data mining. Its features include extensive data manipulation and transformation capabilities, various data mining approaches, and powerful visualization capability to do exploratory data analysis. MineSet supports client/server architecture and runs on Silicon Graphics platforms. (www.sgi.com)

### *NETMAP*

NETMAP is a general purpose, information visualization tool. According to its creators, NETMAP has proved most effective for large qualitative, text-based data sets. It runs on UNIX workstations. (www.alta-oh.com)

### *OMEGA*

OMEGA is a system for developing, evaluating and implementing predictive models using the genetic programming approach. It is suitable for classification and visualization approaches to data mining. It runs on Windows 95 and Windows NT platforms. (www.capgemini.com)

### *Partek*

Partek is a multiple strategy data mining product. It is based on several methodologies including statistical techniques, neural networks, fuzzy logic, genetic algorithms, and data visualization. It runs on UNIX platforms. (www.partek.com)

### *Pattern Recognition Workbench (PRW)*

PRW is a comprehensive multiple strategy tool. It uses neural networks, statistical pattern recognition, and machine learning methodologies. It runs on Windows 95 and Windows NT platforms. PRW uses a spreadsheet-style GUI. The data must be brought into one or more spreadsheets, after which the data is prepared for mining with the product's extensive set of functions. PRW automatically generates alternative models and searches for a best solution. It also provides a variety of visualization tools to monitor model building and interpret results. A model can be deployed as a spreadsheet function, as a dynamic link library, or as C code. (www.ultranet.com/~unica)

### *SAS*

SAS Institute offers one of the most comprehensive sets of integrated tools for data mining. The SAS Data Mining Solution is an integrated software product that incorporates tools for all stages of data mining. It offers a variety of data manipulation and transformation features. In addition to statistical methods, the SAS Data Mining Solution employs neural networks and decision trees. SAS runs on Windows 95, Windows NT, and UNIX platforms. (www.sas.com)

### *Scenario*

Scenario is a single strategy tool that uses tree-based approach to data mining. It runs on Windows 95. Scenario relies on wizards to guide a user through different tasks and is easy to use. Scenario can import data from a variety of sources via Impromptu, a data query tool. (www.cognos.com)

### *SPSS*

Like SAS, SPSS offers one of the most comprehensive sets of integrated tools for data mining. SPSS includes data management and data summarization capabilities as well as tools for both verification and discovery. The complete suite of SPSS techniques includes statistical methods, neural networks, and

visualization methods. SPSS is available on a variety of platforms including Windows, OS/2, IBM AIX RS/6000, and Sun workstations. (www.spss.com)

### *STATlab*

STATlab is a single strategy tool that relies on interactive visualization to help a user perform exploratory data analysis. It can import data from common relational databases. STATlab runs on Windows, Macs, and UNIX platforms. (www.slp.fr)

### *Syllogic*

The Syllogic Data Mining Tool is a toolbox that combines many data mining methodologies and offers a variety of approaches to uncover hidden information. The toolbox is built around a central worksheet, that allows access to different techniques and help organize data. The Syllogic Data Mining Tool includes several data preprocessing and transformation functions. It is currently available on X/Motif on Silicon Graphics, X/Motif on IBM AIX, and Windows NT. It supports access to Oracle 7, Sybase, DB2, and Tandem. (www.syllogic.nl)

## Product Evaluation Factors

Since there are so many different data mining products, it is necessary to develop a metric to compare them so that a user can make a judicious choice. The selected metric should consider all aspects of using a data mining product. Based upon the data mining-literature review, user's comments, and experience with other software products, a metric consisting of the following components is suggested.

- Scope: This component measures the versatility of a data mining product. An ideal product should allow a user to implement any one of the data mining approaches - classification, clustering, regression, etc. through more than one model building methodology. However, only a handful of products provides the entire range of data mining approaches.

- Input: This factor reflects the ability of a product to support different data types and ability to access data from different data sources. It also considers data manipulation and transformation capabilities present in the product.

- Output: This reflects the overall quality of the discovered information including how well it is presented to the user. It also includes factors such as how easy it is for a user to execute the discovered model.

- Ease of Use: This incorporates factors such as: "How easy is the product to use?" "Does it require a user to learn a scripting language?" and "How good and intuitive is the user interface?"

- Vendor Support: This reflects the quality of vendor support through documentation, training, and post-sales help. This is an important factor as data mining is prone to garbage-in and garbage-out syndrome.

- Efficiency and Scalability: This component measures the "strength" of the product. Is it designed for mining gigabytes to terabytes of data? Is it meant only for small data subsets? Does the product include or require special hardware, for example parallel processing, for speed?

- Visualization: Since data mining is an interactive process, visualization plays an important role in the final quality of discovered information. This factor measures the power of visualization features in a product.

- Compatibility: Compatibility implies how well other commercial tools such as OLAP and visualization tools can gain access to data selected for mining and mining results. This is important not only for proper understanding of data before mining it, but also for model interpretation.

Since each user is unique in preferences and working style, it is strongly advised to obtain experience with a selected set of products either using demo versions or by hands-on experience to solve some minor data mining task. Only then should the selection for a product be finalized.

**Table A-1 Listing of Data Mining Products**

| Product | Vendor | Arch. | Platform | DM Strategy | DM Models |
|---|---|---|---|---|---|
| AgentBase/ Marketeer | DAZ System, Inc. | Desktop, C/S | Windows 95, NT Sun Solaris | Statistical Machine Learning | Cl/Ct/R |
| BusinessMiner | Business Objects, Inc. | Desktop | Windows 95, NT | Decision Trees | Classification |
| Clementine | Integral Solutions Ltd. (UK) | Desktop, C/S | Windows 95, NT Unix | Neural Nets Rule Induction | A/Cl/Ct/R |
| Darwin | Thinking Machines | C/S | HP-UX, IBM AIX Sun Solaris | Neural Nets CART, GA, MBR | Classification Regression |
| Data Mining Suite | Information Discovery, Inc. | 3-Tier C/S | Windows 95, NT Unix | Rule Induction Spatial Analysis | Cl/R Visualization |
| Data Surveyor | Data Distilleries (The Netherlands) | C/S | Windows 95, NT Unix | Rule Induction | Classification |
| DataBase Mining Marksman | HNC Software Inc. | Desktop | Windows NT Accelerator Board | Neural Nets | Cl/Ct/R |

**Table A-2 Listing of Data Mining Products (Continued)**

| PRODUCT | VENDOR | ARCH. | PLATFORM | DM STRATEGY | DM MODELS |
|---|---|---|---|---|---|
| DataMind | DataMind Corporation | Desktop, C/S | 95, NT, Intel NT<br>HP,SGI, Sun, IBM | Intelligent Agents | A/Cl/Ct |
| Datasage | Cirrus Recognition Systems, Inc. | 3-Tier C/S | Windows 95<br>Unix | Proprietary | Cl/R/V |
| Decision Series | NeoVista Corp. | C/S<br>Parallel | HP, Sun, DEC | Statistical<br>Neural Nets, GA | A/Cl/Ct/S |
| Discovery | Pilot Software, Inc. | C/S | Windows NT | Statistical<br>Neural Nets | A/Cl/Ct/S |
| Intelligent Miner | IBM | C/S<br>Parallel | RS6000, ES/9000<br>AS/400 | N Nets, Decision Trees, Statistical | A/Cl/Ct/R/S |

**Table A-3 Listing of Data Mining Products (Continued)**

| PRODUCT | **VENDOR** | **ARCH.** | **PLATFORM** | **DM STRATEGY** | **DM MODELS** |
|---|---|---|---|---|---|
| Knowledge Discovery Bench | NCR | C/S Parallel | Windows 95, NT Unix | Neural Nets Rule Induction | A/Cl/Ct/R |
| Knowledge Seeker | ANGOSS Software | Desktop C/S | Windows 95, NT Unix | Decision Trees | Classification |
| MineSet | Silicon Graphics | C/S | SGI Platforms | Decision Trees Visualization | A/Cl/V |
| NETMAP | ALTA Analytics, Inc. | Desktop | Unix | Visualization | A/Cl/Ct |
| OMEGA | Cap Gemini bv (The Netherlands) | Desktop | Windows 95, NT | Genetic Programming | Classification Visualization |

**Table A-4 Listing of Data Mining Products (Continued)**

| PRODUCT | VENDOR | ARCH. | PLATFORM | DM STRATEGY | DM MODELS |
|---|---|---|---|---|---|
| Partek | Partek Inc. | C/S | HP, SGI, Sun | Statistical, Fuzzy<br>Neural Nets, GA | A/Cl/Ct/V |
| PRW | Unica, Inc. | Desktop | Windows 95, NT | N Nets, Machine Learning, Statistical | Cl/Ct/R/S/V |
| SAS | SAS Institute, Inc. | Desktop<br>C/S, MF | PCs-to-main frames | N Nets, Statistical<br>Decision Trees | A/Cl/Ct/R/S/V |
| Scenario | Cognos, Inc. | Desktop | Windows 95 | Decision Trees | Classification<br>Clustering |
| SPSS | SPSS, Inc. | Desktop<br>C/S, MF | PCs-to-main frames | N Nets, Statistical<br>Decision Trees | A/Cl/Ct/R/S/V |

**Table A-5 Listing of Data Mining Products (Continued)**

| PRODUCT | VENDOR | ARCH. | PLATFORM | DM STRATEGY | DM MODELS |
|---|---|---|---|---|---|
| STATlab | InfoWare, Inc. | Desktop | Windows 95<br>Mac, Unix | Visualization | Visualization |
| Syllogic Data Mining Tool | Syllogic (The Netherlands) | Desktop | Windows NT<br>IBM AIX, SGI | Neural Nets<br>Statistical, GA | A/Cl/Ct/S/V |

A: Association; Cl: Classification; Ct: Clustering; R: Regression; S: Sequence Analysis; V: Visualization

# APPENDIX B

# DATA MINING EXAMPLES

---

Many businesses are currently employing data mining technology, and their number continues to grow as more and more data mining success stories become known. Here we present a small collection of real-life examples of data mining implementations from the business world. We also present various pitfalls of data mining to make readers aware that data mining needs to be applied with care to obtain useful results.

## Real-life Examples of Data Mining

### *Capital One Financial Corp.*

Financial companies are one of the biggest users of data mining. One such user is Capital One Financial Corp., one of the nation's largest credit card issuers. It offers 3,000 financial products, including secured, joint, co-branded and college-student cards. It is using data mining to help sell the most appropriate financial product to 150 million potential prospects residing in its over 2-terabyte Oracle7-based data warehouse. Even after a customer has signed up, Capital One continues to use data mining for tracking the ongoing profitability and other characteristics of each of its customers. This continuous monitoring enables the company to lure customers from other competing cards by offering them a temporarily low interest rate for balance-transfer. The use of data mining and other strategies has helped Capital One expand from $1 billion to $12.8 billion in managed loans over past eight years. According to David Buch, Capital One's IT director for data warehousing, " This is a great testimony as to how well data mining has worked." Another data mining application at Capital One is fraud detection. This is an extremely serious problem for credit card companies. For example, Visa and MasterCard lost over $700 million in 1995 from fraud. Based in part on its proprietary data mining tools and San Diego-based HNC Software Inc.'s neural network-based credit card fraud-detection system Falcon - Capital One has been able to cut its losses from fraud by more than 50 percent since 1996.

### *Cablevision System Inc.*

Cablevision Systems Inc., the Woodbury, New York, based cable TV provider was concerned about its competitiveness with deregulation letting telecom companies into the cable industry. In consequence, it decided that it needed a central data repository so that its marketing people could have faster and more accurate access to data. Using data mining, the marketing people at Cablevision were able to identify nine primary customer segments among the company's 2.8 million customers. This included customers in the "indispensable" segment to customers likely to "switch." This has allowed Cablevision to focus on those segments most likely to buy its offerings for new services. The company has also used data mining to compare the profiles of two sets of targeted customers - those who bought new services and those who did not. This has led the company to make some changes in its messages to customers, which, in turn, has led to a 30 percent increase in targeted customers signing up for new services.

### *Southern California Gas Company*

The Southern California Gas Company is using SAS software as a strategic marketing tool. The company maintains a data mart called the Consumer Marketing Information Database that contains internal billing and order data along with external demographic data. According to the company, it has saved hundreds of thousands of dollar by identifying and discarding ineffective marketing practices.

### *American Express*

Another example of data mining is at American Express, where data warehousing and data mining are being used to cut spending. American Express has created a single Microsoft SQL Server database by merging its worldwide purchasing system, corporate purchasing card and corporate card databases. This allows American Express to find exceptions and patterns to target for cost cutting. To detect these exceptions and patterns, American Express is using dual-processor Compaq servers and KnowledgeSeeker from Angoss Software, Toronto.

### *MCI*

MCI is another company which has found great value in data mining. By mining databases of its customer-service and telemarketing data, MCI has discovered new ways to sell voice and data services. For example, it has found that people who buy two or more services were likely to be relatively loyal customers. It also

found that people were willing to buy packages of products such as long-distance, cellular-phone, Internet, and other services. In consequence, MCI now offers such packages.

### *Safeway, UK*

Grocery Chains, like financial institutions, have been another big user of data mining technology. Safeway UK is one such grocery chain with more than $10 billion in sales. It uses Intelligent Miner from IBM to continually extract business knowledge from its product transaction data. According to Mike Winch, IT director at Safeway UK, Intelligent Miner has discovered "correlations that are beyond human conceptual capability." For example, Intelligent Miner found that a particular cheese product, ranked below 200 in sales, was often purchased by the top-spending 25 percent customers. Normally, the product would have been discontinued, disappointing the best customers. Thanks to data mining, Safeway UK is also able to generate customized mailings to its customers by applying the sequence-discovery function of Intelligent Miner. This has allowed Safeway UK to maintain its competitive edge.

## Pitfalls of Data Mining

Despite the above and many other success stories often presented by vendors and consultants to show the benefits that data mining might provide, data mining has several pitfalls. When used improperly, data mining can generate lots of garbage. The best example of the incorrect application of data mining in the popular literature is Michael Drosnin's recent book *The Bible Code* about hidden messages in the Bible. According to Drosnin, the Hebrew Bible contains forecasts about all events - major and minor. He demonstrates this by arranging the Hebrew Bible on a grid of letters and using a computer to look for words formed in four major directions - across, up, down, and diagonally. This methodology is a glaring example of one of the pitfalls of data mining. As Andrew Lo, a professor at MIT, points out: "Given enough time, enough attempts, and enough imagination, almost any pattern can be teased out of any data." David J. Leinweber, managing director of First Quadrant Corp. in Pasadena, California, gives another example of the pitfalls of data mining. Working with a United Nations data set, he found that historically, butter production in Bangladesh is the single best predictor of the Standard & Poor's 500-stock index. This example is similar to another absurd correlation that is heard yearly around Super Bowl time - a win by the NFC team implies a rise in stock prices.

Peter Coy, Business Week's associate economics editor, warns of four pitfalls in data mining. The first pitfall, according to Coy, is that it is tempting to develop a

theory to fit an oddity found in the data, although some oddities are pure chance. The second pitfall is that one can find evidence to support any preconception provided you let the computer churn long enough. Coy terms the third pitfall as "Story-Telling" and says "A finding makes more sense if there's a plausible theory for it. But a beguiling story can disguise weaknesses in the data." The fourth pitfall that Coy warns about is using too many variables. "The more factors the computer considers, the more likely the program will find relationships, valid or not," states Coy.

## Advice to a New User

It is crucial to realize that data mining can involve a great deal of planning and preparation. Just having a large amount of data alone is no guarantee of the success of a data mining project. The consensus among the experienced users and consultants is that the following rules are vital to the success of a data mining project:

- Always remember that business need is more important than the razzle-dazzle of a technical solution. Focus on a specific business problem and gather clear, unambiguous knowledge about the business problem; the end-users, who will leverage the results; and available data sources. Most importantly, the business needs must be clearly defined and acceptable return-on-investment must be earmarked.

- "It is the data preparation, stupid" is the rule-number two. Almost every one who has mined data even once agrees that preparing data is as much as 80 percent of the data mining process. Don't just throw in your data and expect a data mining tool to reward you; prepare it carefully if you want to be rewarded.

- Don't rely on a single methodology. Have a collection of methodologies. Try more than one methodology for the same problem before settling on one for final use. This is necessary, as there is no way to determine beforehand which methodology will work best for a given task.

- Keep the end-users informed and involved. They must be prepared to accept and act upon the results of data mining. This is facilitated when end-users are part of the data mining loop.

In addition to above rules, the final advice to deal with data mining is to remember that it is an iterative, interactive process. In the words of Jagadish Mirani, Oracle's senior product manager: "Be prepared to generate a lot of garbage until you hit something that's actionable and meaningful for your

business. Data mining has to be institutionalized within an analytical, systematic framework. Do that, and data mining will benefit your business in ways you never thought possible."

# APPENDIX C

# CASE STUDIES

---

Here we present three case studies of data mining applications in industry. The first case study is from a telecommunication company, where data mining was used to keep customers from switching to competitors. The second case study is from a large bank. Operating under the commonly accepted industry models of home equity loan customers, the bank was concerned about poor customer response. By applying data mining, the bank soon discovered two hidden customer models and was able to make significant gains in its home equity loan portfolio. The third case study is from a utility. It illustrates how data mining was used to target an unregulated product to utility customers. These case studies are being presented to give readers an appreciation of how data mining is applied in practice and what benefits can be derived from it.

## How ABC Cellular Is Minimizing Churning*

The competition in wireless communications industry is fierce; there is a continuous customer turnover as different wireless-phone service providers try to steal customers from each other. It is common for service providers to loose 7 to 8 percent of their customers every month to competition. The wireless-phone service industry calls this customer turnover churning. Most companies try to make up for lost customers through signing new subscribers by offering all kinds of rebates and inducements.

ABC Cellular, a wireless-phone service provider in a large metropolitan market of over 7.5 million people, was resigned to live with churning, losing about 8 percent of its customers every month. Although ABC was easily able to attract new subscribers to make up for lost customers, the churning had started hurting

---

* This and the bank case study are drawn from *Data Mining Techniques*, M. Berry and G. Linoff, John Wiley & Sons, New York, 1997.

ABC; it was spending $500 to $600 to acquire each new customer. With several new wireless-phone service providers appearing on the horizon, ABC knew that it could not continue spending $500 to $600 to get each new customer; it needed to act to minimize churning. It was time to identify loyal customers and keep them satisfied. It was also the time to identify customers likely to leave and make proactive efforts to retain them.

After several rounds of discussions on how best to control churning, the management at ABC decided to apply data mining. Not being sure of how far data mining could help minimize churning, the management opted to go for a pilot project first. To do the pilot, ABC assembled a team consisting of its own experts on business practices and procedures, a data mining consulting company, and a telemarketing service bureau to generate a campaign to keep the potential defectors. The idea was to use data mining to identify customers likely to remain and those likely to leave. Furthermore, it was decided that the data mining results would be used to develop customized telemarketing scripts to approach customers identified as likely to leave. The company felt that this would reduce churning.

The first task in the pilot was to identify sources of data. After several interviews and discussion sessions, two sets of data sources were identified. The first data source was a customer profile database made available by a database marketing company. It contained summary information for each ABC subscriber. This included the billing plan, type of phone, local monthly usage time, roaming** monthly usage time, inbound and outbound number of calls to each known cellular market in the United States, and many other similar fields. The second data source was identified as the ABC's cellular switching office responsible for keeping call record data. Each call record contains information that includes the subscriber id, the call originating number, the number called, the originating cell, the cell duration, and several other similar items.

The next task was to combine the call record data with the customer profile data. This yielded for each subscriber a detailed calling pattern as well as summary information. This was done for about 50,000 customers selected randomly. For each selected customer, six months of data was gathered. However, several selected customers canceled their service during these six months, effectively yielding a training data set of customers who are likely to stay and those who are likely to leave.

---

** Roaming is the cellular industry term for leaving the service area of the primary wireless-phone service provider.

Having gathered the necessary data, the pilot team decided to use automatic cluster detection and decision tree approaches to data mining. The underlying idea was to detect clusters of subscribers that had similar behavior patterns and then, apply a decision tree method to each cluster to obtain prediction rules for service cancellation. The decision tree approach for this task was selected because it not only provided the predictive score for each subscriber but also yielded clues as to why particular groups of subscribers were likely to cancel their service. Several iterations of clustering and decision tree building were performed with ABC's own business experts assessing the validity of the data mining rules at the end of every iteration.

Once the stable set of clusters and the rules for subscribers behavior were obtained, the pilot team at ABC decided to score the entire subscriber population of the pilot study. Each subscriber was assigned a cluster membership tag and a cancellation risk flag. Next, the team decided to evaluate the validity of the data mining results. To this end, two groups of subscribers were created by equally dividing the 50,000 scored subscribers of the pilot study. Subscribers to these groups were assigned randomly. The first group was designated as the *control group*. The second group was designated as the *test group*. The telemarketing service bureau was asked to prepare scripts tailored to the concerns of high-cancellation-risk subscribers in the test group.

Next, the pilot team launched a proactive telemarketing campaign going after the high-cancellation risk subscribers in the test group. This campaign was run with two aims. The first aim was to obtain information from the high-risk subscribers to check whether their reasons for dissatisfaction and likely cancellation matched with those suggested by the data mining results. The second aim was to retain the high cancellation subscribers by offering them some kind of enticement.

Nothing was done to subscribers assigned to the control group. They were simply tracked. After three months of monitoring of the control and test groups, the difference in the cancellation rate between high-risk and low-risk subscribers in the control group was determined. The pilot team expected this difference to be close to the classification accuracy of the data mining model that it had built. The pilot team was pleasantly surprised to see that indeed it was so. The pilot team also examined the difference in the cancellation rate between high-risk subscribers in the test group and high-risk subscribers in the control group. The team expected this difference to provide a measure of how effective the telemarketing campaign has been and how effective the decision tree rules were to explain why a particular group of subscribers was likely to leave. Again, the team had a pleasant surprise.

Having thus found that data mining can indeed help control churning in a cost-effective manner, ABC Cellular is now extending the pilot to its entire subscriber base. It is also looking at new applications for data mining in its entire business process.

## Targeting of Home equity Loan Customers

Banking industry is one of the major users of data mining technology. This case study from a bank illustrates how data mining can yield hidden market segments.

XYZ Bank was interested in expanding its base of home equity loan customers. The bank was targeting through direct mail campaigns two groups of potential customers. The first group included people with college-age children who, the bank thought, would want to borrow against their home equity to pay tuition bills. The second group consisted of people with high but fluctuating income who might want to even out high and low income periods by tapping into their home equity. These two models of the home equity loan customers were based on the traditional wisdom of the banking industry. However, several direct mail campaigns had not produced the expected results and the executives were looking for reasons why the home equity loan campaign was not going anywhere. It was at this point that the management at XYZ Bank decided to seek help from data mining.

A team of home equity loan experts, information technology personnel, and outside data mining consultants was put together to apply data mining to identify the potential customers who should be targeted for home equity loans. XYZ Bank is one of a group of forward-looking banks with a strong tradition of being at the forefront of technology. Thus, data for data mining was not much of a problem. The bank had all kinds of information on its ten million retail customers in a relational database on a large parallel computer. The database contained information on customers collected directly by the bank as well as demographic information such as income, number of children, type of home, and other relevant items obtained from outside vendors.

To perform data mining, the team first compiled a training set of customer records. This was done using the records of customers who had been approached in earlier direct mail campaigns. The customers who had opted for home equity loans were put in the training set as examples of customers likely to ask for a home equity loan. The records of the customers who had not responded at all to previous direct mail campaigns were included in the training set as examples of customers unlikely to request a home equity loan. A decision tree-based data mining tool was next used to learn to classify customers as likely or unlikely

home equity loan candidates. Through several rounds of training and testing the decision tree model, eventually an accurate tree classifier was built.

In the next step, the team decided to determine whether or not there was any pattern in when customers applied for loans. It was done with the idea that it is not enough to identify potential loan seekers but also the appropriate time to approach customers should be determined. A sequential pattern analysis tool was used for this task.

Finally, the team decided to perform clustering on customers to segment them into homogeneous groups. Through several iterations, 14 different segments of clusters were identified. A detailed analysis of each segment was performed to seek out interesting relationships that might be present. While many of the clusters did not provide any illuminating relationships, one cluster did show two very interesting relationships. First, it contained a large number of customers who had both business and personal accounts with the bank. Second, the customers in this cluster accounted for over one-fourth of the 11 percent of the bank customers whom the decision tree classification model had identified as likely home equity loan candidates.

The above analysis suggested that many customers were using home equity loans to finance their businesses. This was verified by field investigations. Once the bank had a clear understanding of its home equity loan customers through data mining, a new customized campaign was developed to aggressively go after potential home equity loan seekers. Contrary to its previous campaigns, XYZ Bank saw this time more than 10 percent of its home equity loan offers being accepted. This acceptance rate is more than double the rate that past direct mail campaigns had yielded. Buoyed by this success, XYZ Bank is currently implementing data mining to market its other products and services. In fact, data mining is transforming the retail marketing at XYZ Bank. From a mass-marketing approach, the bank is evolving into an institution of targeted-marketing.

## How ABC Utility Revitalized Sales of Unregulated Products

This case study illustrates how a utility increased its sales of unregulated products through data mining, and in the process readied itself for deregulation.

ABC Utility Company is a multi-billion dollar electric and gas company, serving over five million residential customers. With deregulation on the horizon, the management at ABC had been anxious to boost sales of unregulated service products. The company had recently started offering a service contract for household appliances, and was in the process of launching several other

products and services. Based on $0.25/customer direct mailing cost, the company estimated that a mass mailing campaign would cost $1.25 million. Given the average success rate of 2.5 percent for mass mailing campaigns and an expected profit of $10 for every customer subscribing to the service contract, the company figured that it needed to do more targeted marketing. It needed to focus on a selected group of customers in order to have a higher success rate. Thus, it decided to go for data mining to identify customers who might opt for a household appliances service contract. The management also felt that data mining would help the company gain a better understanding of its customers at this stage of imminent deregulation and prepare the company for launching of other new products and services. Thus, ABC Utility was willing to invest in data mining for future results. Since the company lacked expertise in data mining, it was decided to work with an outside consulting company.

A consulting company with a good track record in data warehousing and data mining was hired by ABC Utility to work with its own people from the information technology department. After several discussion sessions, the project group developed a two-pronged approach for data mining. First, it decided to build a classification model to identify customers who would favorably respond to new service from ABC. Second, it decided to perform segmentation of the entire customer database to locate different groups of customers who had similar characteristics. This, the group felt would help in devising the marketing plan for the home appliance service contract as well as give information for future products.

After analyzing the data in the utility's database, the project group felt that the existing data at ABC needed to be supplemented through two additional means. First, the demographic and psycho-graphic (life style) data for all of ABC's customers needed to be purchased from database sellers, because the utility's database did not have enough information on each customer to perform a useful segmentation. Second, the group saw a need for acquiring customer response data through test marketing to build a decision tree-based classification model. It was deemed cost effective to collect response data from only a very small fraction of total customers. Thus, a random sample of 100,000 ABC customers, 2 percent of the total customers, was selected to collect test marketing data. These customers were mailed the promotional literature for the new service from ABC.

The responses from 100,000 test customers were pooled with the purchased demographic and psycho-graphic data, and with the existing data at ABC. The consultants and the ABC team on the project then obtained some simple summary statistics, using SQL, on customers who had opted for the service. This allowed the team to determine which of the customer attributes were important for building the classification model. The next step was to build the decision tree-

based classification model. For this, the project team at ABC decided to break the set of 100,000 strong customer data into training and test sets following the 80/20 rule. Using 80,000 records from the training set, a decision tree model was built through an interactive process. Testing with the remaining 20,000 responses of the test set, the classification accuracy of the decision tree model was checked and found acceptable.

Using the classification model thus built, all five million customers of ABC Utility were scored in terms of the likelihood of accepting the service contract. At the same time, an automatic cluster identification tool was applied to the entire customer data set to perform segmentation. This process yielded three large segments and three small segments. Analyzing each segment, the largest segment, consisting of 40 percent of ABC's residential customers, was found to be of heavy energy users, mostly suburban professionals. The customers in this segment were all homeowners and the majority of them were dual income families. The next largest segment, 35 percent of ABC's residential customers, turned out to consist mostly of renters. This group of people carried fewer credit cards and used less energy. The third largest segment consisted of 16 percent of ABC's residential customers. This segment was dominated by elderly customers. Each of the remaining three segments had about 3 percent of the customers. There were no dominant characteristics that the project team could associate with each of these segments.

Looking at the likelihood score for the service contract for customers in each segment, the group found 14 percent of the customers in the first segment and 23 percent of the customers in the third segment showed very high scores. Two separate promotional booklets describing the advantages of the service contract were prepared to suit the needs of each of the two segments. These were then mailed to high scoring customers, 280,000 in the first segment and 184,000 in the third segment. The importance of customer targeting was visible at ABC when about 50 percent of the targeted customers opted for the service contract. Thus, ABC Utility was able to sign about 230,000 customers, nearly the double the number that it would have signed through mass mailing.

Pleased with the above success, ABC Utility is currently looking at many data mining applications. For example, it is evaluating the use of data mining in identifying and predicting staff absenteeism. It has also found that data mining can help forecast transformer failures. It has also found some applications for data mining to environmental problems. Most importantly, ABC Utility is ready for deregulation.

# APPENDIX D

# WEB RESOURCES

---

The following web sites contain valuable information about data mining and other related technologies. It is advisable to visit these sites before embarking on a major data mining project.

### www.kdnuggets.com

A must visit site for a data miner. The site contains invaluable information about data mining activities and pointers to past and current research. It maintains a guide to commercial and public-domain tools for data mining. It also provides links to companies providing software, consulting, and data mining services.

### www.datamining.org

It is a site run by a group of users and international data mining tool providers. It is geared more for business users. It contains links to many data mining sites.

### www.cs.bham.ac.uk/~anp/TheDataMine.html

In addition to information about data mining, this site also contains load of information about OLAP.

### pwp.stametinc.com/larryg/index.html

A valuable site for business users of data mining. It contains pointers to all kinds of end user tool vendors, infrastructure technology vendors, and function and industry specific tool vendors.