CHAPTER 4
The knowledge discovery process

Introduction

In this chapter we analyze the knowledge discovery process, discuss the different stages of this process in depth, and illustrate potential problem areas with examples. This chapter together with Chapter 5 provides a good basis for the structuring of a KDD pilot project. In principle, the knowledge discovery process consists of six stages:

- Data selection
- Cleaning
- Enrichment
- Coding
- Data mining
- Reporting

The fifth stage, data mining, is the phase of real discovery. Although the methodology as presented here gives the impression that there is a linear trajectory through the process, where you enter at the left, travel to the right, and exit (see Figure 4.1), this is not the case. At every stage, the data miner can step back one or more phases; for instance, when in the coding or the data mining phases, the data miner might realize that

Figure 4.1 The KDD process.

the cleaning phase is incomplete, or might discover new data and use it to enrich other existing data sets. It is impossible to describe in advance all the pollution that can be expected in a database, as most will be discovered only during the mining stage.

Data mining methodology states that in the optimal situation, data mining is an ongoing process. Organizations should continually work on their data, constantly identifying new information needs and trying to improve the data to make it match the goals better. In this way any organization will become a learning system. Since most of the phases need the input of a great deal of creativity, such a process enables and encourages this creativity by refusing to impose any limit on possible activities.

The knowledge discovery process in detail

In this chapter we will use one consistent example, which deals with the database of a magazine publisher. The publisher sells five types of magazine – on cars, houses, sports, music, and comics. The aim of the data mining process is to find new, interesting clusters of clients in order to set up a marketing exercise. Therefore, we are interested in questions such as 'What is the typical profile of a reader of a car magazine?' and 'Is there any correlation between an interest in cars and an interest in comics?' In this book, we have used a number of small sample databases of about 1000 records, which although much too small to be called a genuine application of KDD, works well for the sake of illustration. Almost all of the techniques we are discussing here scale up very well in principle to databases that contain millions of records.

Data selection

In our example we start with a rough database containing records of subscription data for the magazines. It is a selection of operational data from the publisher's invoicing system and contains information about people who have subscribed to a magazine. The records consist of: client number, name, address, date of subscription, and type of magazine. In order to facilitate the KDD process, a copy of this operational data is drawn and stored in a separate database. An illustration of the contents of this database is given in Figure 4.2.

Cleaning

There are several types of cleaning process, some of which can be executed in advance while others are invoked only after pollution is detected at the coding or the discovery stage.

Client number	Name	Address	Date purchase made	Magazine purchased
23003	Johnson	1 Downing Street	04–15–94	car
23003	Johnson	1 Downing Street	06–21–93	music
23003	Johnson	1 Downing Street	05–30–92	comic
23009	Clinton	2 Boulevard	01–01–01	comic
23013	King	3 High Road	02–30–95	sports
23019	***Jonson***	1 Downing Street	01–01–01	house

Figure 4.2 Original data.

A very important element in a cleaning operation is the de-duplication of records (see Figure 4.3). In a normal client database some clients will be represented by several records, although in many cases this will be the result of negligence, such as people making typing errors, or of clients moving from one place to another without notifying change of address. There are also cases in which people deliberately spell their names incorrectly or give incorrect information about themselves, especially in situations where individuals have been refused some type of insurance. By slightly mis-spelling their name or by giving a false address, they try to avoid a negative decision. Of course it is important for any company to be aware of such abnormalities in the database. Although data mining and data cleaning are two different disciplines, they have a lot in common and pattern recognition algorithms can be applied in cleaning data.

Client number	Name	Address	Date purchase made	Magazine purchased
23003	Johnson	1 Downing Street	04–15–94	car
23003	Johnson	1 Downing Street	06–21–93	music
23003	Johnson	1 Downing Street	05–30–92	comic
23009	Clinton	2 Boulevard	***01–01–01***	comic
23013	King	3 High Road	02–30–95	sports
23003	Johnson	1 Downing Street	***01–01–01***	house

Figure 4.3 De-duplication.

In the present example we have both a Mr Johnson and a Mr Jonson in the database. They have different client numbers but the same address, which is a strong indication that they are the same person but that one of the spellings is incorrect. Of course, we can never be sure of this, but a de-duplication algorithm using pattern analysis techniques could identify the situation and present it to a user to make a decision. This type of pollution occurs frequently: because of an error in the original database, there appear to be two clients when in reality there is only one, giving a company the impression that it has more clients than in fact is the case. As this is a situation that often occurs in real life, many large banks and insurance companies have no reliable idea about how many customers they really have. This presents a serious problem in setting up marketing activities, but after de-duplication the two subscriptions of Mr Johnson/Jonson can be definitely recognized as those of one individual.

The second type of pollution that frequently occurs is lack of domain consistency (Figure 4.4). Note that in our original table we have two records dated 1 January 1901, although the company probably did not even exist at that time. This type of pollution is particularly damaging, because it is hard to trace, but it will greatly influence the type of patterns you find when you apply data mining to this table. In some databases, analysis shows an unexpectedly high number of people born on 11 November. When people are forced to fill in a birth date on a screen and they either do not know or do not want to divulge it, they are inclined to type in '11-11-11.' Needless to say, this is disastrous in a data mining context, since if information is unknown it should be represented as such in the database. In our example, we have replaced part of the data with NULL values and corrected other domain inconsistencies.

Client number	Name	Address	Date purchase made	Magazine purchased
23003	Johnson	1 Downing Street	04–15–94	car
23003	Johnson	1 Downing Street	06–21–93	music
23003	Johnson	1 Downing Street	05–30–92	comic
23009	Clinton	2 Boulevard	NULL	comic
23013	King	3 High Road	02–30–95	sports
23003	Johnson	1 Downing Street	12–20–94	house

Figure 4.4 Domain consistency.

Enrichment

In the present example, we will suppose that we have purchased extra information about our clients consisting of date of birth, income, amount of credit, and whether or not an individual owns a car or a house (Figure 4.5). This is more realistic than it may initially seem, since it is quite possible to buy demographic data on average incomes for a certain neighborhood, and also car and house ownership can be traced fairly easily. Alternatively, we can interview small sub-samples of the client database which will give us very detailed information on the customers' behavior. For this example however, it is not particularly important how the information was gathered, but it is necessary to appreciate that the new information can easily be joined to the existing client records.

Client name	Date of birth	Income	Credit	Car owner	House owner
Johnson	04-13-76	$18,500	$17,800	no	no
Clinton	10-20-71	$36,000	$26,600	yes	no

Figure 4.5 Enrichment.

Coding

The data in our example can undergo a number of transformations. First the extra information that was purchased to enrich the database is added to the records describing the individuals.

In the next stage, we select only those records that have enough information to be of value (see Figure 4.6). Although it is difficult to give detailed rules for this kind of operation, this is a situation that occurs frequently in practice. In most tables that are collected from operational data, a lot of desirable data is missing, and most is impossible to retrieve. You therefore have to make a deliberate decision either to overlook it or to delete it. A general rule states that any deletion of data must be a conscious decision, after a thorough analysis of the possible consequences. In some cases, especially fraud detection, lack of information can be a valuable indication of interesting patterns.

In the present example, we lack vital data concerning Mr King, so we choose to exclude this record from the final sample (Figure 4.7). Of course, this decision is questionable, because there may be a causal con-

Client number	Name	Date of birth	Income	Credit	Car owner	House owner	Address	Date purchase made	Magazine purchased
23003	Johnson	04-13-76	$18,500	$17,800	no	no	1 Downing Street	04-15-94	car
23003	Johnson	04-13-76	$18,500	$17,800	no	no	1 Downing Street	06-21-93	music
23003	Johnson	04-13-76	$18,500	$17,800	no	no	1 Downing Street	05-30-92	comic
23009	Clinton	10-20-71	$36,000	$26,600	yes	no	2 Boulevard	NULL	comic
23013	King	NULL	NULL	NULL	NULL	NULL	3 High Road	02-30-95	sports
23003	Johnson	04-13-76	$18,500	$17,800	no	no	1 Downing Street	12-20-94	house

Figure 4.6 Enriched table.

Client number	Date of birth	Income	Credit	Car owner	House owner	Address	Date purchase made	Magazine purchased
23003	04-13-76	$18,500	$17,800	no	no	1 Downing Street	04-15-94	car
23003	04-13-76	$18,500	$17,800	no	no	1 Downing Street	06-21-93	music
23003	04-13-76	$18,500	$17,800	no	no	1 Downing Street	05-30-92	comic
23009	10-20-71	$36,000	$26,600	yes	no	2 Boulevard	NULL	comic
23003	04-13-76	$18,500	$17,800	no	no	1 Downing Street	12-20-94	house

Figure 4.7 Table with column and row removed.

nection between the lack of information and certain purchasing behavior by Mr King. For the moment we will suppose that we can omit this data without consequences for our final results. Next we carry out a projection of the records. In this example we are not interested in the clients' name, since we just want to identify certain types of client, so their names are removed from the sample database.

Up to this point, the coding phase has consisted of nothing more than simple SQL operations but now we are entering the stage where we will be able to perform more creative transformations on the data. By this time, the information in our database is much too detailed to be used as input for pattern recognition algorithms. Take for example the notion of a date of birth: an algorithm that puts people with the same date of birth into a certain customer class is obviously much too detailed for our purposes, whereas a similar algorithm that operates on age classes with an interval of, for instance, 10 years would be very applicable. The same holds true for addresses. Address information is much too detailed for pattern recognition and, in this case, we need to recode addresses into regional codes. The way in which we code the information will, to a great extent, determine the type of patterns we find. Coding, therefore, is a creative activity that has to be performed repeatedly in order to get the best results. Take, for example, the subscription date; again, this is much too detailed to be of any value as such, but there are various ways to recode such dates in a way that yields valuable patterns. One solution might be to transform purchase dates into month numbers, starting from 1990. In this way, we might be able to find patterns in time series of our customers' transactions. We could find dependencies similar to the following rule:

> *A customer with credit >13,000 and aged between 22 and 31 who has subscribed to a comics at time T will very likely subscribe to a car magazine five years later*

Or we might identify trends such as:

> *The number of house magazines sold to customers with credit between 12,000 and 31,000 living in region 4 is increasing*

We may also identify migration of client types, such as:

> *A customer with credit between 5,000 and 10,000 who reads a comics magazine will very likely become a customer with credit between 12,000 and 31,000 who reads a sports and a house magazine after 12 years*

Sometimes, however, we are not interested in time series but in information such as seasonal influence on customer behavior. In such cases we can recode the subscription dates to seasonal codes and try to find patterns in this data. Coding is a creative process – there can be an infinite number of different codes that are related to any number of different potential patterns we would like to find.

In our example we can apply the following coding steps:

1 *Address to region*: This is merely a simplification of the address information. In the area we are investigating there could be millions of different addresses, which is much too detailed for our purposes. We therefore compress the address information into four different area codes. Note however that this is not an arbitrary decision; we could decide to use 20 or 1000 different area codes, or to change the definition of the area. All these decisions might possibly affect the outcome of the data mining algorithms and must therefore be made in a deliberate and considered manner.

2 *Birth date to age*: This implies a division of birthday information into discrete values of about 100 age classes (people on average do not live to become much older than 100 years). Here also we could have chosen a smaller or bigger number of classes, for instance, ten classes of 10 years.

3 *Divide income by 1000*: This not only simplifies the income information but also creates income classes with the same order of magnitude as the age classes. After this operation, most people will have an income class somewhere between 10 and 100, so it is much easier to compare this information with the age classes we have created, since the numbers are close to each other.

4 *Divide credit by 1000*: The reasoning for this is the same as for the income classes.

5 *Convert cars yes–no to 1–0*: In data mining applications it is sometimes useful to code binary attributes (that is, attributes that can take only two values) into one bit, as this facilitates an efficient execution of pattern recognition algorithms.

6 *Convert purchase date to month numbers starting from 1990*: A purchase in January 1990 is allocated month number 1; a purchase in December 1991 month number 24. This last operation helps us to perform time series analyses on the data. Again this is a creative decision – coding in days is probably much too detailed to uncover general time dependencies. On the other hand, one would need a coding in days to identify untypical customer behavior on special days such as Christmas, Easter and other public holidays.

Instead of doing a time analysis, our publisher is more interested in relationships between readers of different magazines. This means that we will not be investigating the connections between a product and subscription dates, but between classes of products, so subscription dates are of less importance at the moment. In Chapter 6, on real-life applications, we consider an example of time series analysis using genetic algorithms.

Client number	Age	Income	Credit	Car owner	House owner	Region	Month of purchase	Magazine purchased
23003	20	18.5	17.8	0	0	1	52	car
23003	20	18.5	17.8	0	0	1	42	music
23003	20	18.5	17.8	0	0	1	29	comic
23009	25	36.0	26.6	1	0	1	NULL	comic
23003	20	18.5	17.8	0	0	1	48	house

Figure 4.8 An intermediate coding stage.

Figure 4.8 represents the new table that results from the coding process. A table in this format, however, is not very helpful if one wants to find relationships between different magazines. Each subscription is represented by one record, although it would be more efficient to have an overview of all the magazines subscribed to by each reader. So we perform a final transformation on the table and create just one record for each reader. Instead of having one attribute 'magazines' with five different possible values, we create five binary attributes, one for every magazine. If the value of the attribute is '1' this means that the reader is a subscriber, otherwise the value is '0'. Such an operation is called 'flattening' – an attribute with cardinality n is replaced by n binary attributes. This is a coding operation that occurs frequently in a KDD context.

Now we have finally coded our data set in such a way that we have: client number, age, income, credit, information concerning car and house ownership, area code, and five bits indicating to which magazines the customer has subscribed (Figure 4.9). This is a good basis from which to start the real data mining process.

							Magazines purchased				
Client number	Age	Income	Credit	Car owner	House owner	Region	car magazine	house magazine	sports magazine	music magazine	comic magazine
23003	20	18.5	17.8	0	0	1	1	1	0	1	1
23009	25	36.0	26.6	1	0	1	0	0	0	0	1

Figure 4.9 The final table.

Data mining

The discovery stage of the KDD process is fascinating. Here we shall discuss some of the most important machine-learning and pattern recognition algorithms, and in this way get an idea of the opportunities that are available as well as some of the problems that occur during the discovery stage. We shall see that some learning algorithms do well on one part of the data set where others fail, and this clearly indicates the need for hybrid learning. We shall also show that there is a relationship between discovery and data cleaning. However, not all forms of pollution are detected during the data mining stage.

Data mining is not so much a single technique as the idea that there is more knowledge hidden in the data than shows itself on the surface. From this point of view, data mining is really an 'anything goes' affair. Any technique that helps extract more out of your data is useful, so data mining techniques form quite a heterogeneous group. Although various different techniques are used for different purposes, those that are of interest in the present context are:

- Query tools
- Statistical techniques
- Visualization
- Online analytical processing (OLAP)
- Case-based learning (*k*-nearest neighbor)
- Decision trees
- Association rules
- Neural networks
- Genetic algorithms

Throughout the rest of this chapter we will use the magazines data set that was described previously. All sample bases that we have used contain about 1000 records and all the results that are presented here are created by applying real-life learning algorithms to these data sets.

Preliminary analysis of the data set using traditional query tools

The first step in a data mining project should always be a rough analysis of the data set using traditional query tools. Just by applying simple structured query language (SQL) to a data set, you can obtain a wealth of information. However, before we can apply more advanced pattern analysis algorithms, we need to know some basic aspects and structures of the

	Average
Age	46.9
Income	20.8
Credit	34.9
Car owner	0.59
House owner	0.59
car magazine	0.329
house magazine	0.702
sports magazine	0.447
music magazine	0.146
comic magazine	0.081

Figure 4.10 Averages.

data set. With SQL we can uncover only shallow data, which is information that is easily accessible from the data set; yet although we cannot find hidden data, for the most part 80% of the interesting information can be abstracted from a database using SQL. The remaining 20% of hidden information requires more advanced techniques, and for large marketing-driven organizations, this 20% can prove of vital importance. A good way to start is to extract some simple, statistical information from the data set, and averages are an important example in this respect.

In our data set (see Figure 4.10) we see that the average age is 46 years old, the average income 20, the average credit 34, and so on. It is interesting to look at the averages of the output fields: we see that 329 clients out of every 1000 subscribe to a car magazine, whereas only 81 out of 1000 subscribe to a comic. These numbers are very important, because they give us a norm by which to judge the performance of pattern recognition and learning algorithms. Suppose that you want to predict how many clients will buy a car magazine. Now an algorithm that always predicts 'no car magazine' would be correct in 671 out of 1000 cases, which is about 70%. Any learning algorithm that claims to give some insight into the data set and do some real predicting has to improve on this. A trivial result that is obtained by an extremely simple method is called a naïve prediction, and an algorithm that claims to learn anything must always do better than the naïve prediction (Figure 4.11). Here we can see also that it is more difficult to make predictions for the small group in our sample set. Since only 81 out of 1000 clients subscribe to a comics, a learning algorithm that claims to predict which clients will subscribe to comics has to give a

Magazine	*a priori* probability that client buys magazine	Naïve prediction accuracy
car	32.9 %	67.1 %
house	70.2 %	70.2 %
sports	44.7 %	55.3 %
music	14.6 %	85.4 %
comic	8.1 %	91.9 %

Figure 4.11 Naïve predictions.

Magazine	Averages				
	Age	Income	Credit	Car	House
car	29.3	17.1	27.3	0.48	0.53
house	48.1	21.1	35.5	0.58	0.76
sports	42.2	24.3	31.4	0.70	0.60
music	24.6	12.8	24.6	0.30	0.45
comic	21.4	25.5	26.3	0.62	0.60

Figure 4.12 Results of applying a naïve prediction.

prediction accuracy better than the 92% achieved by using the naïve prediction. This will be difficult in most cases. Figure 4.12 illustrates the averages per magazine.

It is interesting to see how these averages change when we focus on different magazines. For example, we see that the average age of a reader of a car magazine is 29, which is considerably lower than the average age of the clients – about 47. As was to be expected, the average age of a comics reader is the lowest. Another interesting piece of information is the number of multiple buyers in the sample, and this is illustrated in Figure 4.13.

Here we see that almost 40% of clients subscribe to only one magazine. However, it is interesting to note that 31% subscribe to two magazines, which indicates that there might be interesting patterns to discover between groups of multiple and single buyers. Later in this chapter we will look at techniques for quantifying what is interesting and what is not. Quite alarm-

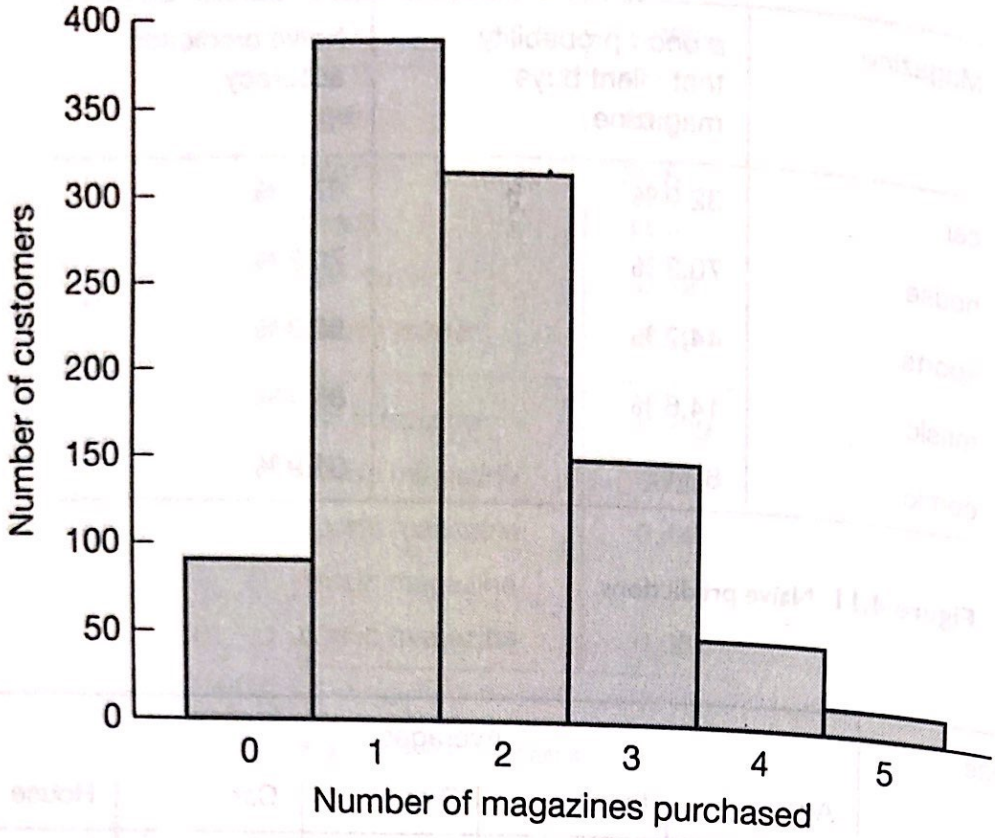

Figure 4.13 Overview of multiple subscription.

ing, however, is the fact that almost 9% of clients in the sample subscribe to no magazine at all, which can only be the result of pollution in the database, and it is essential to investigate how this pollution has occurred and what can be done to prevent it in the future. This illustrates the developmental nature of data mining: an ongoing process by which knowledge and understanding of the data improves and deepens all the time.

We have seen some interesting patterns in the age attributes and we would like to concentrate on this to extract more information. In order to demonstrate this process, we will investigate the general age structure of our sample. We see that the ages are, apart from very young and very old people, almost equally spread over the sample (Figure 4.14).

Interesting differences occur when we analyze certain sub-groups. Readers of the car magazine cluster around the age class of 30 (Figure 4.15) while readers of the sports magazine are spread much more evenly over the population (Figure 4.16). SQL can yield detailed information on the structure of a data set and this information can be very useful for marketing or other purposes. We have to go through this phase before we can turn our attention to more advanced learning algorithms. Remember, however, that we can never judge the performance of an advanced learning algorithm properly if we have no information concerning the naïve probabilities of what it is supposed to predict.

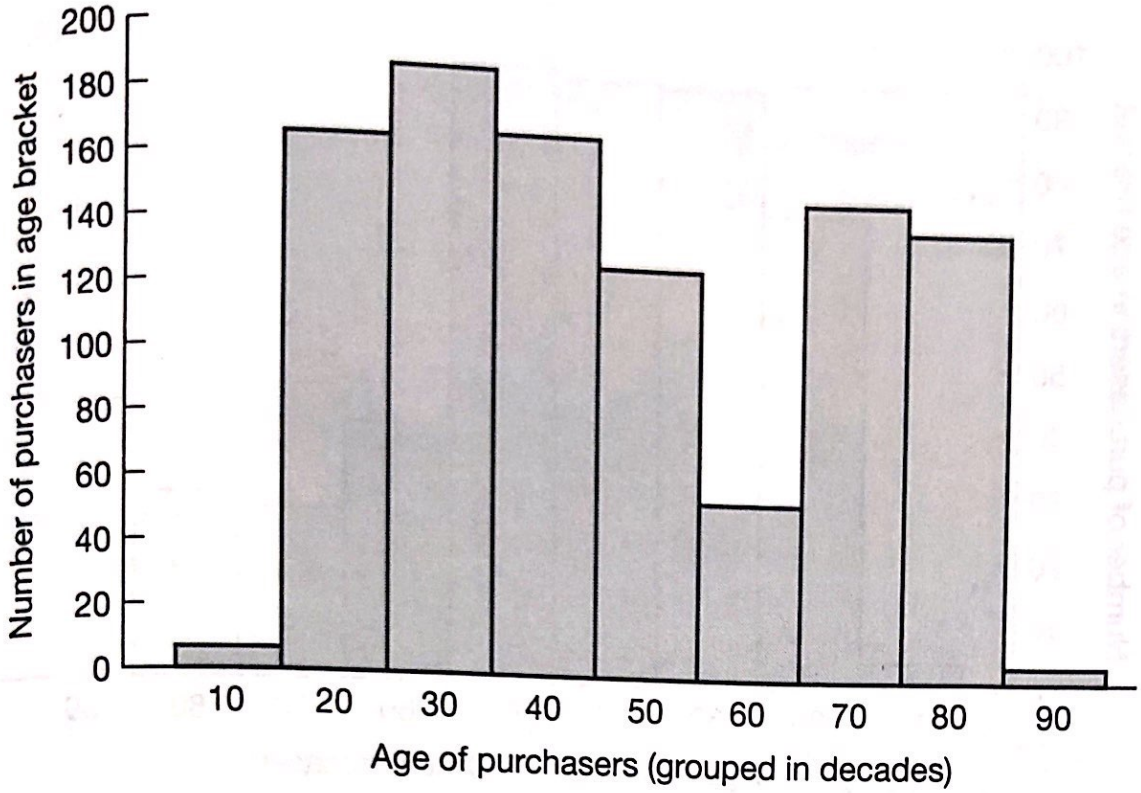

Figure 4.14 Age distribution of readers.

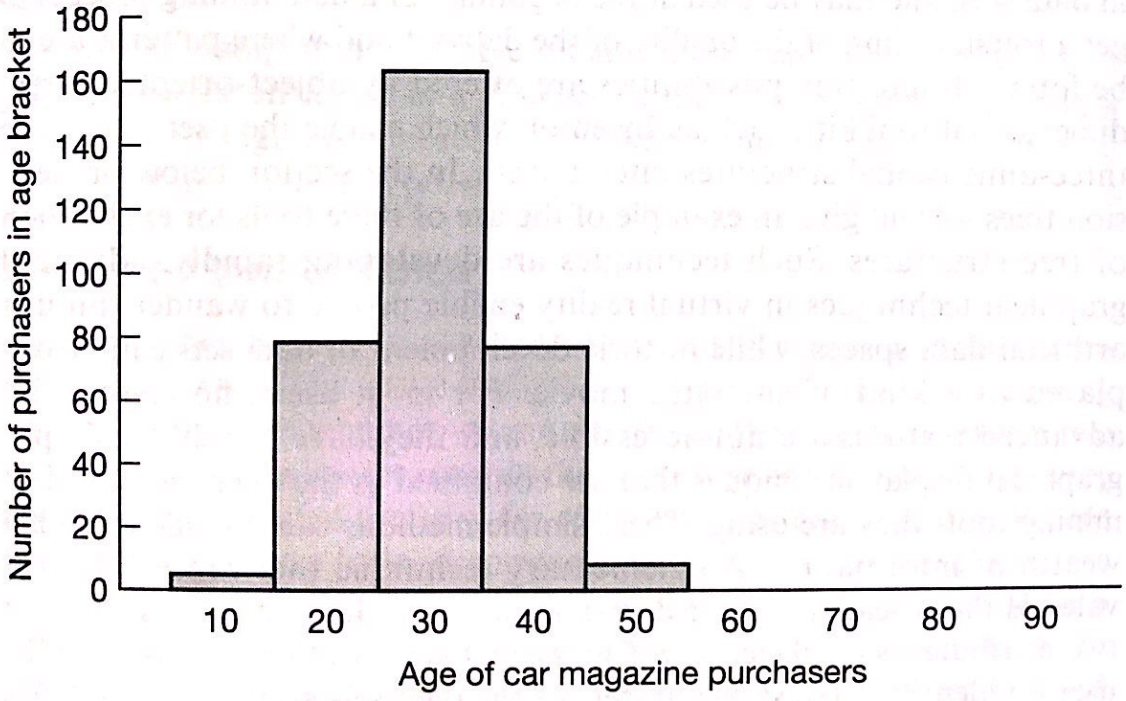

Figure 4.15 Age distribution of readers of the car magazine.

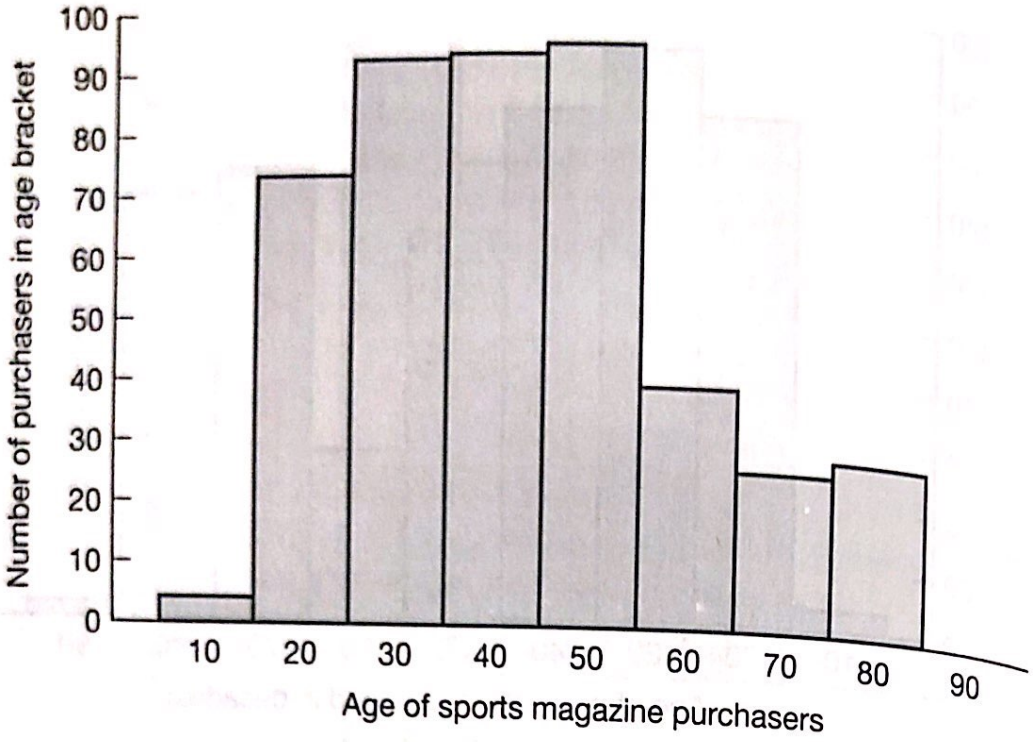

Figure 4.16 Age distribution of readers of the sports magazine.

Visualization techniques

Visualization techniques are a very useful method of discovering patterns in data sets, and may be used at the beginning of a data mining process to get a rough feeling of the quality of the data set and where patterns are to be found. Interesting possibilities are offered by object-oriented three-dimensional tool kits, such as Inventor, which enable the user to explore three-dimensional structures interactively. In the section below on decision trees we will give an example of the use of these tools for exploration of tree structures. Such techniques are developing rapidly: advanced graphical techniques in virtual reality enable people to wander through artificial data spaces, while historic development of data sets can be displayed as a kind of animated movie. For most users, however, these advanced features are not accessible, and they have to rely on simple, graphical display techniques that are contained in the query tool or data mining tools they are using. These simple methods can provide us with a wealth of information. An elementary technique that can be of great value is the so-called scatter diagram; in this technique, information on two attributes is displayed in a Cartesian space. Scatter diagrams can be used to identify interesting sub-sets of the data sets so that we can focus on the rest of the data mining process. There is a whole field of research dedicated to the search for interesting projections of data sets – this is called projection pursuit. In our example (Figure 4.17) we have made a projection along two dimensions: income and age. We see that on average young people with a low income tend to read the music magazine.

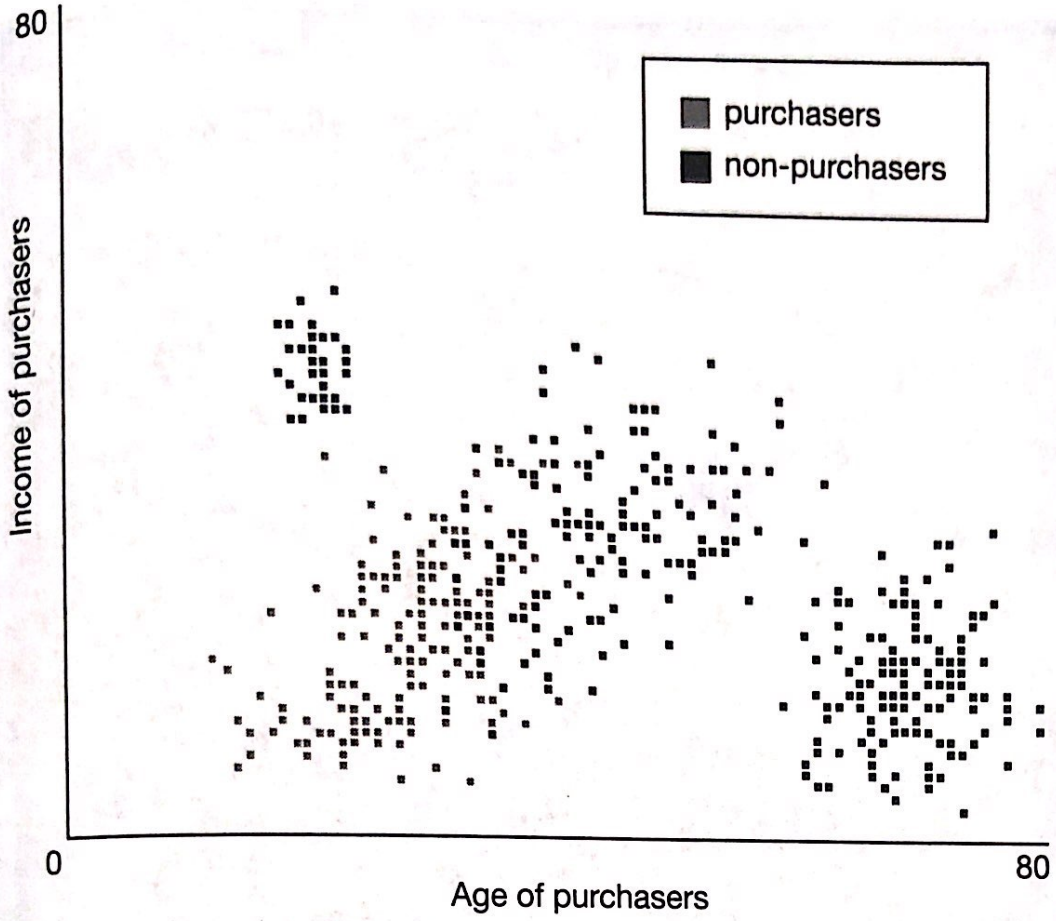

Figure 4.17 Age and income distribution of readers and non-readers of the music magazine.

Now we can compare how simple visualization techniques can help give a feeling for the structure of a data set. A much better way to explore a data set is through an interactive three-dimensional environment, and Figure 4.18 illustrates this possibility.

Likelihood and distance

There are other reasons to conceive records as points in a multi-dimensional data space. The space-metaphor is very useful in a data mining context. Using this metaphor we can determine the distance between two records in this data space: records that are close to each other are very alike, and records that are very far removed from each other represent individuals that have little in common. Our sample database contains attributes such as age, income, and credit; these three attributes form a three-dimensional data space and we can analyze the distances between records in this space (Figure 14.19). Here another advantage of good coding comes to light – in order to achieve a good comparison between values, we must normalize the attributes. Age, for example, ranges from 1 to about 100 years, while income has a range from 0 to approximately 100,000 dollars a month. Now if we use this data without correction, income will of course be a much more distinctive attribute than age, and

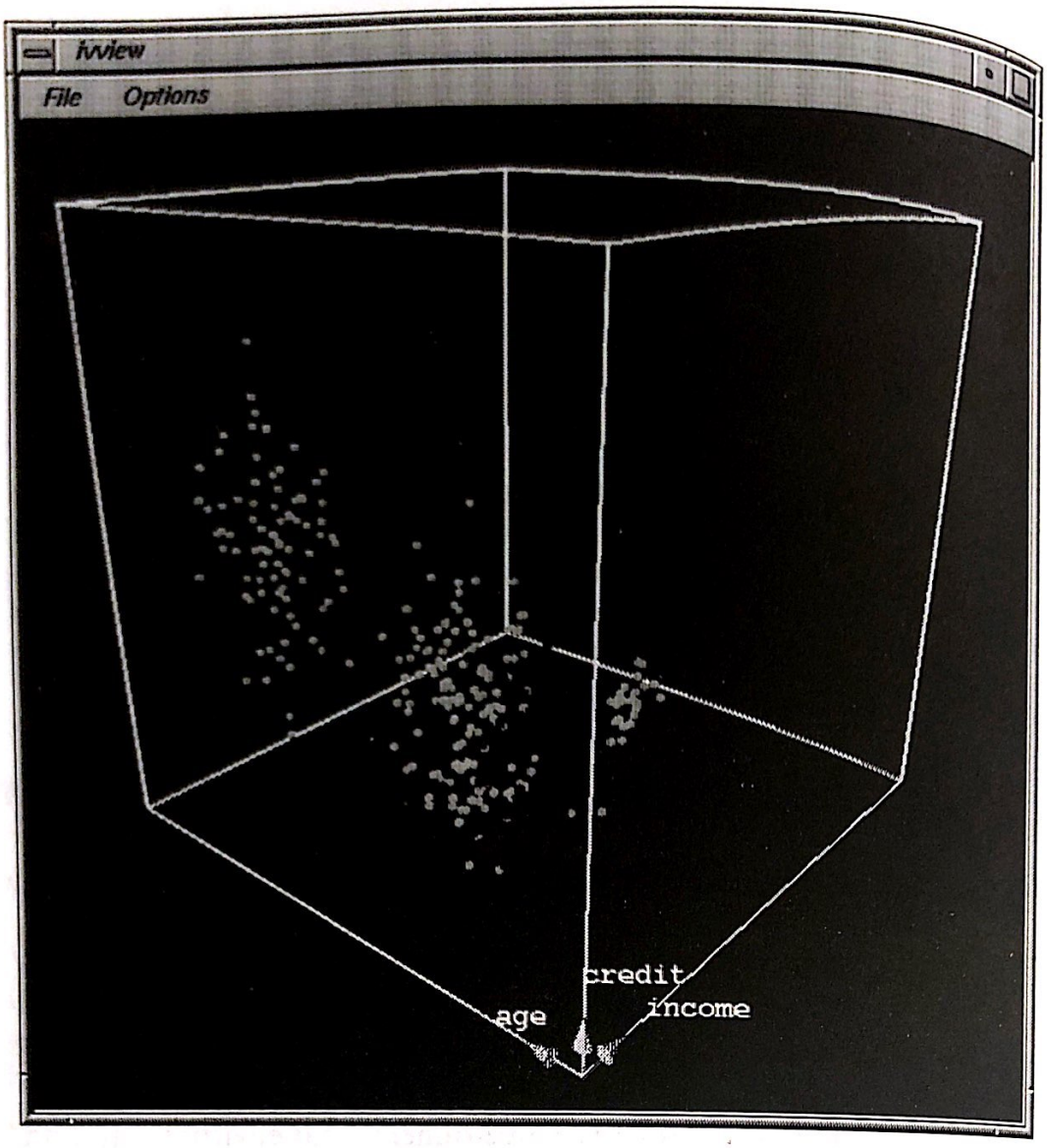

Figure 4.18 Interactive three-dimensional environment with data about age, income and credit of readers of the music magazine.

Records form points in a space determined by their attributes, and distance between them can be measured.

	Age	Income	Credit
Customer 1	32	40,000	10,000
Customer 2	24	30,000	2,000
	8	10 (rescaled)	8 (rescaled)

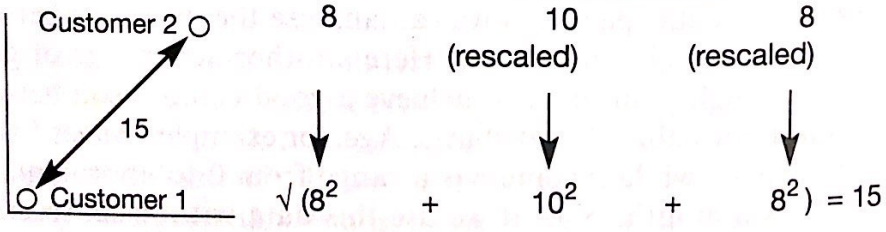

Figure 4.19 Distance between datapoints.

this is not what we want. Therefore we divide income by 1000, in order to obtain a measure that is the same order of magnitude as age. We do the same for the credit attribute. If we scale all the attributes to the same order of magnitude we obtain a reliable distance measure between the different records. In our example, using a Euclidean distance measure, the distance between customer 1 and customer 2 is 15.

In this way, records become points in a multi-dimensional data space. For data spaces with low dimensionality it is easy to visualize data clouds, and sometimes we can identify interesting clusters merely by visual inspection. In most cases, however, we need more advanced search programs to uncover such clusters, and interesting predictions can also be visualized in this way: sometimes it is possible to identify a visual cluster of potential customers that are very likely to buy a certain product. In our sample data set (Figure 4.20) age, income, and credit form an ideal three-dimensional space in which to do this kind of clustering analysis.

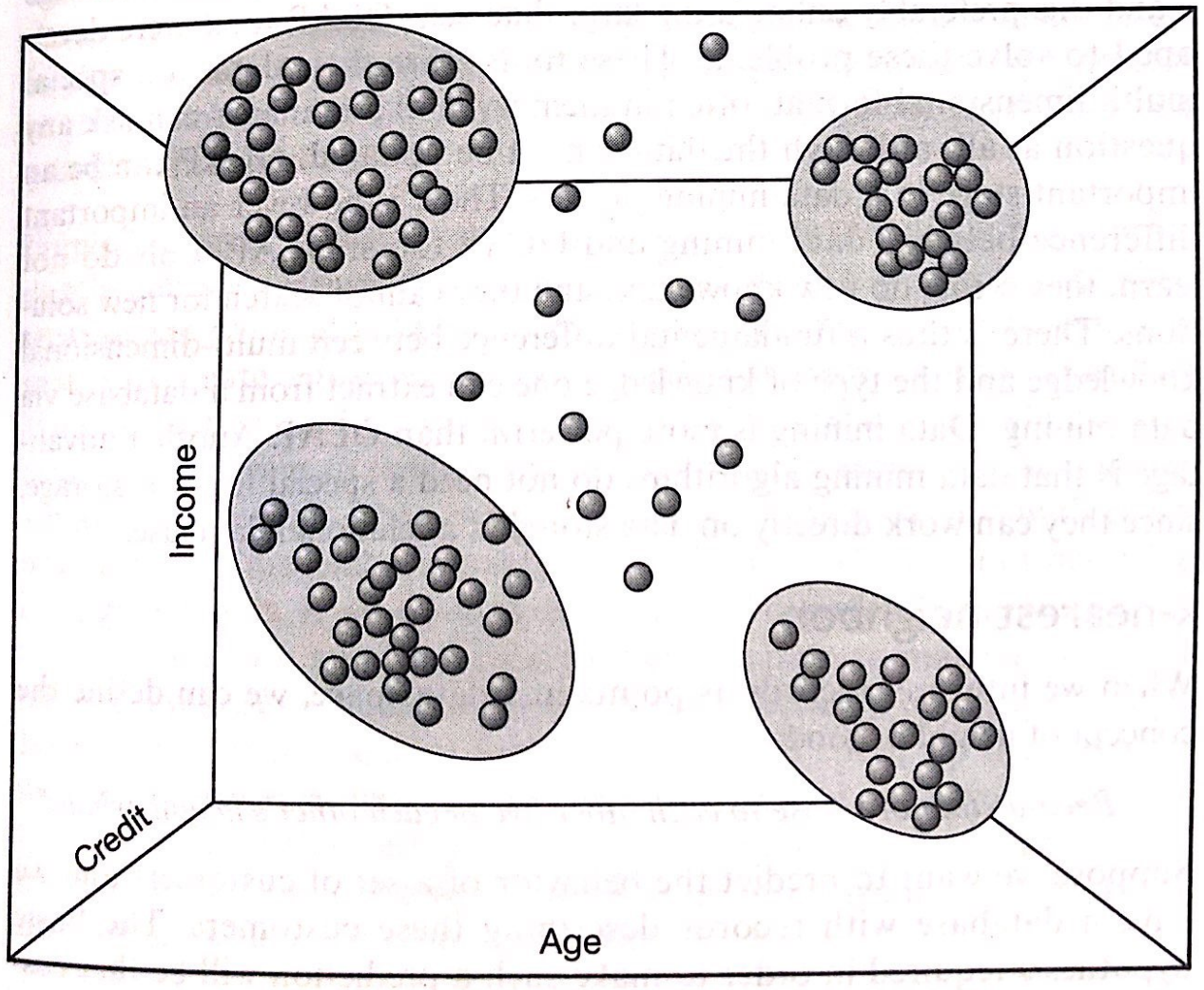

Figure 4.20 Finding interesting clusters.