Report on Data Mining and Data Visualisation

Namedropper visualisation software
Contents

Section A: Data mining

1. Overview
   1.1 Functions of data mining

2. Definitions of data mining
2.1 Historical data
2.2 Accessibility for end users
2.3 Data warehouses
2.4 Pattern recognition
2.5 Data visualisation
2.6 Automated queries and intelligent software tools

3. Data mining technologies
   3.1 Decision-tree approach
   3.2 Rule induction
   3.3 Neural networking
   3.4 Genetic programming
   3.5 Fuzzy logic
   3.6 Nearest neighbour approach

4. Knowledge Discovery in Databases (KDD)

5. Java and data mining applications
   5.1 Advantages of using Java
   5.2 Disadvantages of using Java

5. Applications (examples)

Section B: Data visualisation

1. Overview
   1.1 Advantages of visualising data
   1.2 Problems with data visualisation

2. Purposes of data visualisation
   2.1 Table look-up
   2.2 Comparison of values
   2.3 Pattern perception
      2.3.1 Pattern perception and Gestalt psychology
      2.3.2 Operations of pattern perception

3. Categories

4. Methods of visualisation
   4.1 Graphic formats
   4.2 Colour
      4.2.1 Advantages of the use of colour
      4.2.2 Limitations of the use of colour
Section A: Data Mining

1. Overview

The exploitation of historical data information sets for business purposes is well established. Companies such as Datastream/ICV and Reuters sell archived information stored in large purpose built databases. SQL enquiry tools and OLAP can be viewed as data mining tools, and so can internet search engines and web crawlers. For example Alta Vista Discovery software is marketed as a web-mining product.

Companies also use the term ‘data mining’ for marketing purposes, and all definitions need to be treated cautiously. Data mining can refer generally to the exploration, analysis and presentation of historical data, or can refer to the specific act of retrieving information, usually from a data warehouse.

1.1 Functions of data mining

Data mining procedures serve three main purposes:

- Knowledge Discovery
- Predictive modelling
- Forensic analysis
**Knowledge Discovery**

Data mining software can explore large sets of data without a predetermined hypothesis to find interesting patterns and then present this information to the user.

**Predictive modelling**

This function uses patterns discovered to predict future behaviour. For example, information gathered from credit card transactions can be used to identify customer priorities or instances of fraud.

**Forensic analysis**

This is the process of applying the extracted patterns to find anomalies and unusual patterns in the data. For example, a retail analyst could explore the reasons why a particular population group makes certain types of purchase in a particular store.

![Data mining functions diagram](image)

**2. Definitions of data mining**

It can be inferred that data mining software tools:

- Exploit historical data from a variety of sources.
- Are easily accessible to end users (i.e., business managers).
- Explore information stored in data warehouses.
- Recognise hidden patterns in large sets of disparate data.
- Interpret data for the human manager using visualisation techniques.
- Automate enquiry processes using up-to-date technologies such as Artificial Intelligence and machine learning processes.
2.1 Historical data
Historical data is the total collection of data collected by a company over a period of time. The information may be decades old, or be seconds old, and continually being produced (e.g. the call logging system of a telecommunications company). It may be numerical in format or non numerical (e.g. the book catalogue of a library).

2.2 Accessibility for end users
Data mining tools are designed for business managers rather than specialist computing managers. Software houses emphasise the portability of tools, the use of plain English queries, and the visualisation of data rather than specialist knowledge of programming languages. This implies a re-alignment of roles in companies, with IT professionals managing open access knowledge bases and data warehouses, rather than controlling secure databases and providing reports on an on-demand basis.

2.3 Data warehouses
Data mining tools can be used on single databases, or can explore many different environments, such as on the Web. However there may be inconsistencies in the way data is stored on multiple databases (e.g. dates may be written in different formats on different databases, or gender may be described as m and f, or 0 and 1). Therefore information is often stored in data warehouses. These are collections of data which are cohesive enough to allow data mining software agents to work efficiently, but which may come from different sources. Unlike operational databases, data warehouses are structured to hold raw data, are static, and may contain data tables from a wide variety of sources.

Differences between database management and data warehouses

Figure 2 Data warehouses: inputs and output
Data warehousing and mining | Database Management systems
---|---
Small scale software tools | Query tools part of large software suite
Diverse information | Homogenous data focused on operational business needs.
Exploitation of diverse data to Generate knowledge | Elimination of data redundancy
Historical data sets | Operational data sets
Flexible | Dependence on systems development life cycle as the means of designing systems

2.4 Pattern recognition

The main purpose of data mining tools is to identify underlying patterns in historical data that would not otherwise be apparent. Business Managers or other professionals can then make appropriate decisions based on this data. Because the data that is mined is large scale and complex, data mining tools automatically identify patterns and store these patterns for end user access in data warehouses.

Data mining tools can be used to test hypotheses about trends and patterns, or can generate patterns automatically using machine learning technologies. The data is then visualised using graphical techniques.

Recent developments in data mining emphasise the automatic interpretation of data, with information stored as patterns, using a variety of mathematical approaches. The conversion of raw data into patterns also has savings in terms of memory usage, over older approaches to data warehousing which involved large-scale data retention.

2.5 Data visualisation

It is important to present processed information in a coherent and easily accessible way. Data visualisation is therefore an essential component of the data-mining concept. Information can be presented in a variety of ways, using traditional devices such as pie charts, scattergraphs, line charts etc. Computer graphics also allow more sophisticated approaches, such as multiple three-dimensional graphs, animated graphs, and real time monitoring.

Data visualisation techniques can also be used for manipulation of data mining queries, as well being the final output of a data exploration and mining process. Data visualisation is described in more detail below (section B).

2.6 Automated queries and intelligent software tools

As software tools have become more sophisticated, business analysis processes that were previously made by humans are now automated within software algorithms that utilise Machine learning and Artificial Intelligence techniques such as neural networks.
3. Data Mining Technologies

Data mining software uses a variety of different approaches to sift and sort data, identify patterns and process information. Methods adopted include:

- Decision-tree approach
- Rule discovery approach
- Neural network approach
- Genetic programming
- Fuzzy logic
- Nearest Neighbour approach

These methods can be combined in different ways to sift and sort complex data. Commercial software packages often use a combination of two or more of these methods.

3.1 Decision-tree approach

A decision tree system partitions data sets into smaller subsets, based on simple conditions, with a single starting point. In the example below, the decision tree is used to make decisions about expenses, from the starting point of ‘Grade’.
A disadvantage of this approach is that there will always be information loss, because a decision tree selects one specific attribute for partitioning at each stage with a single starting point. The decision tree can present one set of outcomes, but not more than one set, as there is a single starting point. Therefore decision trees are suited for data sets where there is one clear attribute to start with.

### 3.2 Rule induction

Data mining systems typically uses rule induction methods to formulate classification rules. Rules can express conditional relationships (IF..., THEN), or association relationships (WHEN..., ALSO), and also attributes (e.g. EXPORT = DATE) unlike decision trees.

A typical classification rule might state that married people are more likely to invest in insurance policies and might be formulated in the following way:

```
Typical rule formulated -
if STATUS = married and INCOME > 10000
```
and HOUSE_OWNER = yes
then INVESTMENT_TYPE = good

A probability score might also be added (this is an 'inexact' rule, and the rule can be applied at all points of the data. Data mining software can generate these rules.

### 3.3 Neural networking

Neural networking classifies large sets of data and assigns weights or scores to the data. This information is then retained by the software and adjusted as it undergoes further iterations.

A neural net consists of a number of interconnected elements (called neurones) which learn by modifying the connections between them. Each neurone has a set of weights that determine how it evaluates the combined strength of the input signals. Once the neural network has calculated the relative effect each of these characteristics has on the data, it can apply the knowledge it has learned to new sets of data. Neural networks can ‘learn’ from examples.

![Neural network structure](image)

However the disadvantage of neural networks is that input has to be numeric, which may lead to complications when dealing with non-scalar fields, such as Country or Product where numeric labels have to be given to fields of equal value. A neural network in the process of iteration may come to assign relationships or values based on these arbitrary numbers, which would corrupt the output.

### 3.4 Genetic Programming

Genetic programs generate rules from data sets by using the metaphor of mutation to make changes in patterns until a suitable form of pattern emerges via selective breeding. Genes represent real problems. The analyst/user sets how many genes are to be created and the number of generations needed. Patterns and accuracy becomes better, the greater the number of genes and generations.
These resulting patterns can be used for rule generation, or can be used to discover patterns, such as in text. For example genetic algorithms may be used to look for patterns in library catalogues, or to search for plagiarism in essays etc.

### 3.5 Fuzzy logic

Rules incorporate probability. So good might mean a 70% success rate, or a 90% success rate. This is called an inexact rule. A "fuzzy" rule can vary in terms of the numeric values in the body of the rule. For example the confidence might vary according to the value of the of one of the variables (e.g. as the age increases)

Fuzzy logic assesses data in terms of possibility and uncertainty. An example of fuzzy logic is:

\[
\text{IF income is low AND person is young}
\]

\[
\text{THEN credit limit is low}
\]

This rule is fuzzy because of the imprecise definitions of "income", "young" and "credit limit". The credit limit will change as the age or income changes.

### 3.6 Nearest Neighbour Approach

The nearest neighbour method matches patterns between different sets of data. This approach is based on data retention. When a new record is presented for prediction, the ‘distance’ between it and similar records in the data set is found (and the most similar neighbours are identified. For example, a bank may compare a new customer with all existing bank customers, by examining age, income etc. and so set an appropriate credit rating.
This process can then be repeated until the analyst is satisfied with the result. Because this method uses data retention, and so uses a lot of memory, a cross section of "typical cases" is sometimes retained, rather than the complete data set. However this short cut can lead to inaccuracies in the final result. The ‘nearest neighbour approach’ is also more suited to numeric values, such as income, rather than non-numeric data (such as product lists).

4. Knowledge Discovery in Databases (KDD)

KDD is the implementation of the above approaches to discover unknown patterns within data sets. Knowledge discovery is seen as the main tenet of data mining. For example the data mining web site at Birmingham University describes the two terms as synonymous.

Knowledge can be defined as distinct from data, in that it can be seen as the meaningful product of the relationship between two or more objects of data. For example, a company might have 20,000 customers in the current year, whilst in the previous year they had 10,000 customers. This is data. Knowledge is the fact that the company has doubled its customer base. It is on the basis of this knowledge that business decisions are made. Data mining software programs use technologies as described above to produce refined knowledge.

Technical development of Knowledge Discovery has been matched by new approaches in management theory such
as the concept of ‘Knowledge Management’ whereby more and more business is driven by the activities of non-process orientated knowledge analysis. Data mining techniques encourage this approach by automating the analysis of data.

5. Java and Data Mining applications

An example of the use of Java programming for a data mining application can be found at Infoleuth Java site. This site describes the advantages and disadvantages of using Java for data mining applications.

5.1 Advantages of using Java

- Ability to safely download platform independent applets. Data can be viewed through web browsers.
- Interpreted platform independent nature of Java
- Java’s strengths as a general-purpose language, including multithreading support, garbage collecting and compile time type checking.

5.2 Disadvantages of using Java

- Software agent has to run off the same host as the HTTP proxy server. This is due to the inability to make socket connections.
- Awkward security issues, due to the inability to replace the 'Class Loader' with a custom class loader in the browser
- Basic set of drawing functions in the AWT

6.0 Applications

Examples of Data Mining applications:

Customer Prospecting and Segmentation

In industries with a large amount of historical information about its customers, data mining can help to break the market into meaningful clusters. For example, a bank may use data mining techniques to identify customers with higher response rates to offers, and so adjust its marketing techniques according

Customer Relationship Management

Data Mining can be used to discover how customers use products and services, allowing business managers to develop and improve their products and services

Product Development
Retail outlets can help identify how products sell together, to enable optimal marketing opportunities, a profit space per square meter. Manufacturers can explore how their products sell in different geographical areas, or via different resellers.

**Quality Control Management**

Data Mining tools can identify unusual patterns in industrial processes, allowing managers to identify problems. Data mining applications can automatically identify patterns in disparate sets of data.

**Health statistics**

Health Authorities can use data mining tools to explore medical records in order to target members of the population who might be at greater risk for a particular condition.

**Identification of computer use**

I.T managers could use data mining tools to explore and analyse patterns of computer use within companies. For example certain clusters of employees might access the web or the company database much more often than other users. Resources could then be allocated as appropriate. The Namedropper program could be adapted for this use, by measuring clusters of activity on web sites.

**Fraud detection**

Fraud cases can be identified, by looking at patterns produced by data mining tools. For example managers could use historical data to identify ways in which purchases differ for each customer after the date of each theft. Counter measures could then be taken.

**Engineering**

Data mining techniques may be applied in diagnostic expert systems, fault detection etc. For example stress loads on prototype components can be applied virtually, before actual manufacture of object. Data mining tools can be used to create iterative tests and visualisations.

**Section B: Data Visualisation**

1. Overview
The visualisation of data encodes digital information into analogue form for easy access by the user. Analogue visualisations are well suited to identifying spatial distances, unlike digital information. For example the physical difference between 79994 and 80000 is substantial, but the difference in values is small. An analogue visualisation, such as a dial demonstrates the true difference more intuitively. Graphical visualisations can be contrasted with table lists, or text description.

1.1 Advantages of visualising data
- The user can absorb large sets of data, so data can be easily accessed and patterns perceived. Pointing devices can also more easily access it.
- Visual pattern perception is a ‘natural’ function of the human brain (see Pattern perception and Gestalt psychology below).
- Navigation through complex and disparate sets of data is easier
- Communication with other people is made more straightforward

1.2 Problems with data visualisation
- Visualisations need to be put into context, to compliment rather than replace numeric values and textual meaning
- Visualisation is a translation of values, so there is the possibility of error due to the design of the graph. For example, Cleveland (1992) describes the misreading of a badly designed graph which led to the destruction of the Space Shuttle Challenger
- Visualisation is resource hungry. It needs large amounts of processing power and memory space
- Visualisations can be over-elaborate and lose their point
The principles underlying this approach by Lucent Technologies are to reduce representation to a single unifying view. Interactive filters enable the analyst to focus on specific points in the data. The above example is used to track program code development: colours and glyphs represent code lines, coders, the age of the code and its function.

2. Purposes of data visualisation

Visualisations can be used for one or more of the following purposes

- To read one precise value (table look up)
- To compare two or more values
- To infer a more complex relationship (perceive patterns)

2.1 Table look up

Computer visualisations enable flexibility in identifying specific units of data. For example the Namedropper uses a text frame below the graph to identify specific texts on a particular co-ordinate.

2.2 Comparison of values
Decisions are often made on the comparison of values. Graph designs such as histograms can demonstrate relative differences in values.

2.3 Pattern perception

Pattern perception is an important function for data mining applications, as it enables the easy recognition of unexpected, unusual or significant patterns hidden within data. The format of the visualisation affects the identification of these patterns. For example scattergraphs are useful tools in identifying data clusters (as with Namedropper).

![Lucent Technologies visualisation of local exchange telephone calls](image)

*Figure 9 Lucent Technologies visualisation of local exchange telephone calls*

*Colour encodes the direction and length of individual conversation whilst node placement displays calling pattern, node shape defines amount of time spent talking.*

2.3.1 Pattern perception and Gestalt psychology

Gestalt psychological theory holds that we perceive the visual field as a perceptual whole: recognition of patterns is called a Gestalt - a pattern that is:

- Regular, (predictable)
- Stable (i.e. unambiguous)
Simple (having the smallest practical number of parts)
■ Complete (accounting for as much information as possible)

This pattern is called a *figure*; all else in the field is named the *ground*. It is important for any visualisation, to make a clear distinction between figure and ground, by using clear headings and colours, and not over cluttering the visual space with highlights and headings. The aim of data visualisations should be to create this Gestalt - a visual whole that appears as a single object on a graph.

### 2.3.2 Operations of pattern perception

Three operations of pattern perception are:

- Detection - the visual recognition of a geometric aspect that encodes a physical value
- Assembly - the visual grouping of detected graphical elements (for exempt the use of the same colour)
- Estimation - the visual assessment of the relative magnitudes of two or more quantitative physical values via discrimination (a does or does not = b), ranking (a is greater or less than b) and ratio-ing (a/b)

### 3. Categories

Categorical data can be distinguished from quantitative values in that it:

- Takes on discrete values - (that is, can be divided into smaller sections. For example the categorical value ‘university’ can be divided the discrete values of individual departments)
- Has no natural order

Some data sets are entirely categorical. For example the IBM data visualisation site gives an example of the classification of mushrooms into discrete data sets.
The data file (left) is a comma-separated file. The program assigns numerical values to the distinct values and sizes a sphere proportional to that count. Colour is used to distinguish edible from poisonous varieties. The graph uses a colour spectrum to measure categories, with blue and red at either extreme.

4. Methods of visualisation

4.1 Graphic Formats

There are many data visualisation formats (examples include bar graphs, pie charts, scattergraphs, and linear plots), usually held within a two-dimensional data rectangle. In addition to ‘traditional’ formats, computer visualisations allow more complex and dynamic approaches to presenting information.

Different formats are suitable for different purposes. For example a scattergraph allows clustering of large sets of data. A line graph on the other hand can allow comparison of two or more sets of data as they change in value.

4.2 Colour

4.2.1 Advantages of the use of colour

- Colour stands out from a monochrome background... the speed of search for a target that is uniquely colour coded in a cluttered field is independent of the size of the field, therefore colour coding of targets or critical elements in a display is quite effective for rapid localisation.

- Colour doing can act like a preventive organising structure to tie together multiple elements that may be spatially separated. For example data which shares the same value may be pooled in different areas of a scattergraph, as on the Namedropper identifies areas of equal density with the same colour.
Colour is coded automatically by the brain and this enhances its value as a redundant coding device (e.g. traffic lights and threat classification).

4.2.2 Limitations of the use of colour

- Colour is subject to the limits of absolute judgement. Wickens (1992) recommends a limit of five or six colours in a single display.
- Colour hues do not naturally define an ordered continuum.
- Colour stereotypes (e.g. red = danger, blue = cold), may create contradictory values. For example, in a refrigeration system, a decrease in temperature might be dangerous.
- Irrelevant colour coding can be distracting.

![Figure 11 NASA visualisation of aerosol distribution over the Pacific.](image)

The colour spectrum range is used to distinguish quantitative values resulting in a colourful but potentially ambiguous display with contradicting orders of brightness and hue. The human brain responds to order of brightness, but not naturally to the spectrum colour range.

4.2.3 Colour and categorical and quantitative values

Colour can be used to delineate categories, or to denote changes in value. Changing colour can provide efficient visual assembly when we encode a categorical variable. If the hue is fixed, but if brightness or saturation changes, then this provides efficient ranking when we encode a quantitative variable.
A general rule for the use of colour is:

Category value - Change hue

Quantitative value - Same hue - Vary brightness/saturation

However sometimes two or more hues are used in a data visualisation to achieve clearly perceived boundaries between adjacent levels of colour coding. For example the contours of a landmass below and above sea level may be reflected in the use of blue and green. The Namedropper uses several hues to delineate boundaries.

It is useful to note that there is note that changes in hue do not represent an ordered continuum in psychological terms. Even the rainbow spectrum is not always seen as ordered in terms of value: "It is erroneous to assume that we have some hard wired intuitions for a spectral sequence (i.e. red, orange, yellow, green blue, indigo, violet)" (DT Travis 1991)

4.3 Multiple Graphs

Multiple graphs are an established approach to producing complex sets of data, where patterns can be inferred from simultaneously viewed graphs. Three issues arise from using multiple graphs.

- Consistency - comparative graphs should share the same scales
- Highlighting differences - the point of using multiple graphs is to highlight the changes in value from graph to graph either in the text or the symbols. For example a series of graphs presenting different y values as a function of the same x variable should highlight the y label
- Distinctive legends - legends of similar graphs should highlight the distinctive features. For example graph 1 might be labelled "1992 production", whereas graph 2 might be labelled "Increased 1993 production".
Figure 12 Visualisation of multiple graphs viewed simultaneously

4.4 Visual momentum

Data visualisations can often involve a series of graphs, or a mixture of graphs and data. This momentum can create problems in interpreting data, and can cause confusion. Wickens (1992) provides four basic guidelines to overcome these problems.

1. Use consistent representations.
2. Graceful transitions - For example on a map the transition from small scale to large scale will be less cognitively less disorientating if this change is made by a rapid but continuous blow-up.
3. Highlight anchors. An anchor may be described as a constant invariant feature of the displayed world, whose identity and location is always prominently highlighted on successive displays. For example the direction of north arrow on a geographical map.
4. Overlaps. When successive displays are introduced, each new frame should include overlapping areas or features with the previous frame, and these common landmarks should be prominently highlighted (colour is an effective highlight).
5. Cut outs. For example on an atlas displaying a map of a country, a cut out might show its location on the world map.

4.5 Animation
According to Gallagher (1995) animation has two main functions in result visualisation:

- The display of behaviour over time
- Navigating a model in 3-D space

Namedropper uses animation in another way, creating a pulse (at 16 frames per second) in order to highlight peaks of data. Slowed down to real time, the Namedropper data visualisation could be used in a variety of situations. For example an IT manager could monitor access to the internet in a large network, and measure rates of access, the location of users and the different sites visited. Sites would be given co-ordinates according to keywords, similar to the indexing of the library catalogue. Such a system could run at real time and be speeded up to analyse quickly activity over time.

4.6 Interactive visualisations

Interactive visualisations allow the user to operate dynamically with the data, whereby a user might zoom into a particular section of the graph or might change variables to explore the effect on other areas of data. A user might test a system by changing a value, by altering its visual representation. The software could then modify the visualisation, or even change the data set if appropriate. For example the IT manager using the Namedropper format as a monitor of computer activity might be able to shut down proscribed terminals by clicking on the appropriate area on the scattergraph.

4.7 Three dimensional designs

The use of 3-D design has several advantages

- People are used to perceiving space in three dimensions. Highly complex data relationships can be generated by using depth perception at little cost in terms of human cognition (i.e. you don’t have to think too hard!).
- More information can be put into the same space.
- Pattern perception is more immediate. For example a 3-D contour map of a mountain range is easier to perceive that 2-D contour lines
- Data clusters can be analysed from different angles (in conjunction with animation techniques).
- Visualisations can be simplified. For example fewer colours can be used, and contour lines can be removed, as appropriate.

However there are costs in terms of program complexity and processing power. There is also the temptation to visualise irrelevant information. Namedropper is designed in two dimensions. If it were designed using three dimensions it would require more memory.
Figure 13 3-D visualisation of United States with 3-D histograms

5. Conclusion: data mining and data visualisation

Visualisation is an integral part of the data mining process, as a primary function of data mining is to identify hidden patterns (and anomalies) in data. These can be best identified using analogue graphical processes. Data visualisations can be seen as a description of the final iteration of a data mining operation.

However visualisations are not only passive outputs but can also be used as dynamic interfaces between the user and the data set. They can be used as part of the interrogation process, enabling the user to frame further questions, or to explore clusters of data in more depth. Furthermore some data mining operations are inherently visual in their approach. For example the decision tree method is conceptually a visualisation of data.
The conceptual idea of space inherent in the decision tree model is easily expressed by visualisation. This graph predicts the outcome probabilities of groups of students at a university. The size of each sphere represents the number of students.

Section C: References

1. Bibliography

<table>
<thead>
<tr>
<th>Author</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cleveland, W.S</td>
<td>The Elements of Graphing Data, Hobart Press, New Jersey, 1994</td>
</tr>
<tr>
<td>Lemay, L &amp; Perkins, C.L</td>
<td>Teach Yourself Java in 21 days, Sams.net, Indianapolis, 1996</td>
</tr>
<tr>
<td>Mattison, Rob</td>
<td>Data Warehousing and Data Mining for Telecommunications, Artech House, 1997</td>
</tr>
</tbody>
</table>
2. Websites

Selection of websites used directly in compiling report

<table>
<thead>
<tr>
<th>Company</th>
<th>Products</th>
<th>Internet address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attar Software limited</td>
<td>XpertRule, Profiler, DM software</td>
<td><a href="http://www.attar.com">www.attar.com</a></td>
</tr>
<tr>
<td>Birmingham University</td>
<td>Data mining resource list</td>
<td><a href="http://www.cs.bham.ac.uk/~anp/TheDataMine.html">www.cs.bham.ac.uk/~anp/TheDataMine.html</a></td>
</tr>
<tr>
<td>IBM</td>
<td>Data explorer visualisation</td>
<td><a href="http://www.hurley.ibm.com/dx/">www.hurley.ibm.com/dx/</a></td>
</tr>
<tr>
<td>Information Discovery Inc</td>
<td>KA-suite</td>
<td><a href="http://www.datamining.com">www.datamining.com</a></td>
</tr>
<tr>
<td>ISL</td>
<td>Clementine DM software</td>
<td><a href="http://www.isl.co.uk">www.isl.co.uk</a></td>
</tr>
<tr>
<td>MCC Inc</td>
<td>Infosleuth Java DM software</td>
<td><a href="http://www.mcc.com/projects/infosleuth">www.mcc.com/projects/infosleuth</a></td>
</tr>
<tr>
<td>NASA</td>
<td>Data visualisation of aerosol spray distribution</td>
<td><a href="http://www.jpl.nasa.gov/lidar">www.jpl.nasa.gov/lidar</a></td>
</tr>
<tr>
<td>Queens University, Belfast</td>
<td>Data mining lecture notes (Ruth Dilly)</td>
<td><a href="http://www.qub.ac.uk/tec/courses/datamining/ohp">www.qub.ac.uk/tec/courses/datamining/ohp</a></td>
</tr>
<tr>
<td>Salford Inc.</td>
<td>Data mining software</td>
<td><a href="http://www.salford.com">www.salford.com</a></td>
</tr>
<tr>
<td>Visible Decisions</td>
<td>3-D visualisation software</td>
<td><a href="http://www.vdi.com">www.vdi.com</a></td>
</tr>
<tr>
<td>Visual Insights</td>
<td>Data visualisation</td>
<td><a href="http://www.bell-labs.com/project/visualinsights/">www.bell-labs.com/project/visualinsights/</a></td>
</tr>
<tr>
<td>York University, Canada</td>
<td>Gallery of data visualisation</td>
<td><a href="http://www.math.yorku.ca/scs/statresource.html">www.math.yorku.ca/scs/statresource.html</a></td>
</tr>
</tbody>
</table>

3. Location (source) of figures and diagrams used in report

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1</td>
<td>Data mining functions</td>
<td>Information Discovery inc.</td>
</tr>
<tr>
<td>Figure 2</td>
<td>Data warehouses: input &amp; output</td>
<td>Information Discovery inc.</td>
</tr>
<tr>
<td>Figure 3</td>
<td>Decision tree screenshot</td>
<td>Attar Xpert Rule</td>
</tr>
<tr>
<td>Figure 4</td>
<td>Neural network structure</td>
<td>Information Discovery inc.</td>
</tr>
</tbody>
</table>
Section D: Testing of Namedropper

1. Basic functionality

The first step in testing was to ensure that the program worked reliably on the machine on which it was written i.e. limited to one platform (Windows 95) and one browser (Netscape 4.06).

2. Other computers/platforms

Next the program was tested under different operating systems i.e. Windows NT. This revealed a problem that does not manifest itself under Windows 95.

The preferred version of the program has 2 threads running at the same time – the thread that creates the images and the thread that animates the panel. The effect is that the animation begins almost immediately and the image gradually fills the panel from left to right. This works perfectly under Windows 95 but results in a deterioration of the images under Windows NT. We were unable to establish exactly how the threads were interfering with each other. As a work-around, an alternative version of the program has been written that does not begin the animation until the images are completely created. However, this is not really a satisfactory solution.
Also some of the machines in the university display fewer colours than the machine on which the applet was developed. However, this did not seriously affect the overall effect on the user.

After several re-writes of the computational methods performance is just about acceptable on the machine on which the program was developed, but it was found to be very poor on some of the university’s machines. This is unavoidable unless:

- the whole notion of a ‘colour map’ is abandoned, or
- we accept that the applet can only be run on powerful PC’s, or
- the number crunching is done beforehand so that the program can simply read in pre-existing images.

3. Other browsers

The program has been tested on the browsers that were available in the university labs. The results were varied – however the applet appears to work fairly consistently with Netscape Navigator.

Section E: Structure of team

Two team members - William Morgan

Tom Chapple

William Morgan System Design and programming

Review of data mining and visualisation report

Tom Chapple Research into data mining and visualisation

Writing report
<table>
<thead>
<tr>
<th></th>
<th>William Morgan</th>
<th>Tom Chapple</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oct</td>
<td>Defined team roles</td>
<td>Defined team roles</td>
</tr>
<tr>
<td></td>
<td>Prototype developed to check animation concept – discarded</td>
<td>Library search. Notes taken</td>
</tr>
<tr>
<td>Nov</td>
<td>Analysed program into two groups of classes: animation and data capture</td>
<td>web search of resources. information collected onto disk</td>
</tr>
<tr>
<td></td>
<td>Created interface</td>
<td>Design of initial structure</td>
</tr>
<tr>
<td></td>
<td>Construction of applet</td>
<td>Writing of data visualisation section</td>
</tr>
<tr>
<td>Dec</td>
<td>Initial testing on different platforms</td>
<td>Installation and initial testing of Namedropper</td>
</tr>
<tr>
<td></td>
<td>Continued development – re-write of slow sorting routines</td>
<td>Redesign of document structure.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>First draft of report competed</td>
</tr>
<tr>
<td>Jan</td>
<td>Final design – re-write of some of the drawing methods</td>
<td>Further drafts of report</td>
</tr>
<tr>
<td></td>
<td>Final testing</td>
<td>Final testing</td>
</tr>
<tr>
<td></td>
<td>Proof reading of report</td>
<td>Report given to William</td>
</tr>
<tr>
<td></td>
<td>Hand in 13 January 1999</td>
<td></td>
</tr>
</tbody>
</table>
Section F: UML design

1. Overall effect

The applet should appear to a user in the following fashion:

- A black rectangular panel with a choice box and a button beneath it.
- The panel will gradually be filled by an animating image.
- Areas of the panel that contain a high density of points will appear to pulsate in addition to being brighter than other areas of the panel.
- When the user clicks on an area of interest in the panel, the ‘choice’ component will display the 5 nearest points to the mouse click.
- When the user clicks on an item in the ‘choice box’ the panel will stop animating for a second and display a flashing indicator on the panel to show exactly where the selected URL is located.
- When the user clicks on the button, the browser will display the currently selected URL (from the ‘choice’ component).

1. UML diagrams

The main functions of the program will be described using collaboration diagrams in UML notation. Before that, here are all the classes used and a brief description of what they are for:
There are 4 main object interactions to consider:

1. Start-up
2. User clicking on panel
3. User clicking on URL
4. User clicking on button to switch browser to new URL

1. Start-up
2. User clicking on panel

1: getCoords ()

中途：findURLs:
方法，该方法找到任何点在面板中的5个最近的URLs

: ClusterPanel

2: findURLs ()

: ClusterVector

3: setURLs ()

: URLChoice

displays URLs in 'choice' component
3. User clicking on URL

2: displayPoint ()

displayPoint: displays a flashing point where the URL is located on the panel

1: getCoords()

: ClusterPanel → : ClusterVector

2.4 User clicking on button to switch browser to new URL

2: displayURL ()

1: getSelected()

: GotoButton → : URLChoice