Exploiting Visualization in Knowledge Discovery

Hing-Yan Lee, Hwee-Leng Ong and Lee-Hian Quek

Information Technology Institute, National Computer Board
71 Science Park Drive, Singapore 0511
Republic of Singapore
email: {hingyan, hweeleng, leehian}@iti.gov.sg

Abstract

Todate visualization has not been extensively harnessed in knowledge discovery in databases (KDD). In this paper, we show that a multi-dimensional visualization (MDV) technique can be used synergistically with a machine learning program like C4.5 to uncover new knowledge. Used together, the two approaches span the KDD spectrum between complete automation on one hand and fully manual on the other. We introduce MDV, its implementation in a tool named WinViz, and show how WinViz supports the various tasks in KDD.

Keywords: data and knowledge visualization, multi-dimensional visualization, interactive data exploration

1. Introduction

The literature reveals several approaches to knowledge discovery in databases (KDD). These can be summarized by the KDD spectrum as follows: At one extreme, complete automation using machine learning to discover knowledge is advocated by one school of thought (e.g., data dredging), while at the other extreme, another school favors a dialectic and interactive orientation (e.g., data archaeology (Brachman et al. 1992, Brachman et al. 1993, Brachman et al. 1994)). In between are machine-assisted (e.g., (Bhandari 1994)) and human-assisted knowledge discovery methods. This spectrum can be depicted in Figure 1.

Data Dredging	Machine-Assisted Discovery Method	Human-Assisted Discovery Method	Data Archaeology

Figure 1: The KDD Spectrum

In practice, the truth lies somewhere along the KDD spectrum and seldom exclusively at the ends. In this paper, we describe an approach that spans between machine-assisted and human-assisted discovery methods. Our work exploits data visualization and machine learning to harness their respective strengths for KDD. Except for (Grinstein et al. 1992), the use of visualization technology has largely been ignored in KDD work. We describe a multi-dimensional visualization (MDV) technique that allows a user to visually examine a tabular database and to formulate query interactively and visually. MDV has been realized in a tool named WinViz, which a user can use to interactively explore a database and identify patterns and trends hidden in the data. The user can uncover new properties of the data and detect any deviations. We get a picture of 'what the data is trying to tell us' - the observations can then be confirmed using conventional statistical analysis. We show how WinViz, when integrated with a machine learning component can support KDD.

1.1 Motivation

The use of visualization in WinViz for KDD has its motivation from the adage that "a picture is worth a thousand words." Instead of using if-then production rules as a knowledge representation formalism, we seek to present the same knowledge visually and graphically. Visualization has been used in a limited extent in several KDD tools. The hope is that patterns and trends can be detected more easily than reading production rules.

1.2 Related Work

Projects that have incorporated some form of visualization to aid in KDD include IMACS, MMV, and Netmap. IMACS ((Brachman et al. 1992), (Brachman et al. 1993), (Terveen 1993), (Brachman et al. 1994)) uses conventional graphs and plots as an interface for the analyst to segment data with mouse clicks, appearing as breaks in a graph to indicate segment boundaries. MVV (Mihalisin et al. 1991) uses bar charts (histogram within histogram within histogram) and slider bars (with horizontal scales) to locate clusters in multidimensional space that allows the display of multiple views of a given data set. The nested histograms resemble the group bars in WinViz. Netmap ((Davidson 1993), (Wright 1994)) is a line-based visualization tool that uses a circle as the basic graphical device. Its circumference is divided into several groups, one for each attribute of interest. Individuals are represented by nodes within the group. Lines drawn across the circles indicate relationships between subgroups or individuals by linking their nodes. Netmap uses clustering

algorithms to group individuals and fuzzy matching techniques to identify near matches.

2. The WinViz Architecture

In this section, we introduce the MDV technique used in WinViz.

2.1 Parallel Coordinates

The concept of the parallel coordinates, as originally conceived by Alfred Inselberg (Inselberg 1987), can be described as follows:

> "In parallel coordinates, the principle coordinate axis are parallel and equidistant to each other. That is, for an *N*-dimensional data set, *N* vertical axis are placed on a plane, so that every two successive axis are one unit part from each other. An *N*-dimensional data entry is represented by a broken (polygonal) line whose vertices lie on the parallel axis and whose height (*y*-position) is determined by the entry's attributes, i.e., the value of the first attribute determines the height of the vertex placed on the first vertical axis, etc. A data set, then, is represented by a collection of these polygonal lines" (Chomut 1987).

2.2 MDV's Parallel Coordinates

The MDV version of the parallel coordinates differs from Inselberg's in several aspects:

- While the polygonal lines exist, they no longer play a significant role. WinViz supports a polygonal line display toggle whereby the user can select between the choice of having a polygonal line to represent a tuple (or database record) or several tuples satisfying the attribute values specified by the user.
- Group bars appear in the place of attribute values on each vertical axis. The group bars help to reduce the complexity of lines when the dataset gets too big.
- The concept of class is introduced where a class is a subset of the data. The user can group subjects of interest into classes and see how these classes are represented in the dataset and how they relate to other attributes.
- Horizontal histograms are provided on the right hand portion of group bars when the data is divided into classes. This allows the user to compare against other attributes.

2.3 The WinViz Interface

An MDV display in WinViz is divided into three main regions: the workspace region, the total population region, and the status region (see Figure 2). The labels correspond to the attribute names. The initial order in which the attributes appear is dependent only on the order of the attributes in the dataset, i.e., the leftmost attribute in the dataset is the leftmost attribute in the display. The display order can easily be changed, if desired, using a click-and-drag operation. The workspace region displays the data graphically. The values of each database attribute are represented by rectangles, known as *group bars*, on a vertical axis in the workspace region. The width of each group bar corresponds to the number of records with the value of the group bar. The height of each group bar is immaterial and has no significance. The total population region displays information with respect to the entire dataset. Initially, it is empty. It changes in values, according to the query or classes created. The status region provides statistical information of a group bar that the mouse cursor is pointing at.

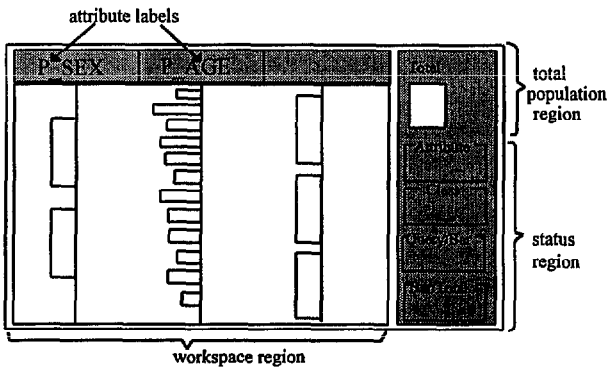

Figure 2: MDV Display Regions

3. The KDD Process

The process of KDD can be viewed as an iteration of stages, as depicted in Figure 3. After the data has been prepared in some appropriate file format and rendered consistent (e.g., removal of noisy data, filling in of missing values, and discarding hopelessly corrupted records), trends and patterns can then be generated, from which hypotheses are elicited for interpretation and analysis.

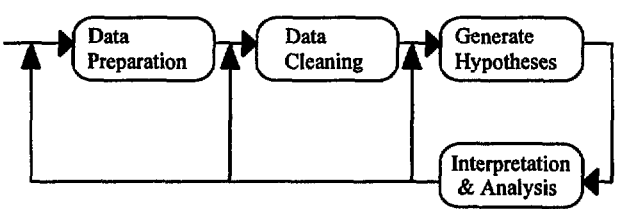

Figure 3: Knowledge Discovery in Databases Process

3.1 Data Preparation

For purpose of illustration, a dataset from the Machine Learning Repository maintained by the University of

California at Irvine is used. This credit screening dataset contains 125 records of past credit applications.

The MDV technique accepts conventional tabular data as input. Each column is treated as an attribute, which can be either discrete or continuous. When the credit screening dataset is loaded into WinViz, the MDV display appears as in Figure 4.

We can see that there are more successful applicants (indicated by the width of the group bar GRANTED=*Yes*) than rejected ones (indicated by GRANTED=*No*). The display allows us to deduce that most of the credit applicants are in the younger and middle age group (as indicated by the width of group bars for AGE where the age value increases up the vertical axis). In addition, we observe that there are more applicants who are employed versus those jobless; general distribution of items for which credit is being sought; and proportion of male and female applicants. Immediately upon initial data loading, we already have an overview of the record distribution visually. The same information is difficult to obtain using other techniques or tools.

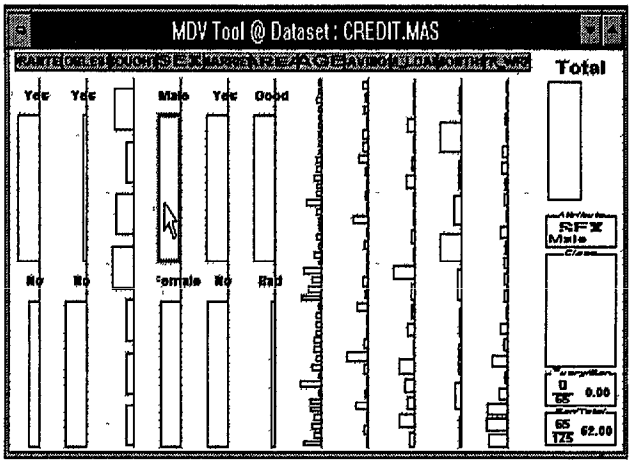

Figure 4: WinViz Display of Credit Screening Dataset

3.2 Data Cleaning

At a glance, several anomalies can be spotted very quickly. If a vertical axis has only a single group bar, we essentially have a one-value attribute. This is uninteresting for purpose of analysis and can be eliminated by interactively dropping the attribute; the selected attribute will disappear from the MDV display. If there is more than the expected number of attribute values, it is likely that we have invalid values. For example, instead of the normal two group bars appearing for the SEX attribute with values *MALE* and *FEMALE*, we encounter additional group bars labeled *M*, *F*, *m*, and *f*, this indicates that the noise exists for this attribute. Missing values for an attribute are indicated on the MDV display by an extra group bar is unlabeled (if it is a blank field in the actual data) or labeled as '?' (if it is indicated as such in the actual data).

3.3 Hypothesis Generation

In any analysis, no less in knowledge discovery, the most difficult task is to formulate a hypothesis and see if the dataset substantiates it. Often, this is an iterative process whereby it is necessary to refine and re-formulate the hypothesis, eventually either discarding it or having found one that is satisfactory. Where the domain expert is familiar with the dataset, the task of formulating a hypothesis is somewhat easier than when confronted with a totally new dataset. In hypothesis generation, WinViz supports only in the interactive data exploration and display aspects (as will be elaborated in the next subsection). The user still has to manually formulate a hypothesis.

To alleviate this shortcoming in WinViz, we have integrated the C4.5 machine learning program (Quinlan 1993) so that if-then production rules generated by the latter can be visualized on the former. This integration has several advantages. It harnesses the interactivity and visual representation of WinViz on one hand. On the other hand, it exploits the generalization capability of C4.5. Thus, induced knowledge is provided as shortlisted hypotheses, which can then be stepped through and visualized on MDV. Rules can be visualized one at a time. Alternatively, all the rules for a decision class can be viewed together. In the second case, the conditions of each rule of the selected decision class are "transferred" onto MDV one by one, with *outliers* of that class becoming apparent. An outlier is:

- one that satisfies the conditions of the rules of the selected decision class, but does not belong to that class; or
- one that belongs to the decision class, but does not satisfy the conditions of the rules of that class.

In Figure 5, the first rule (generated by selecting all the attributes except GRANTED which is the decision class) has been selected for display on MDV. The hatched boxes represent the conditions of the rules. We observe that there are 13 records that satisfy the rule (as shown in the third box in the status region). Polygonal lines shaded according to the classes are displayed. Of these only 12 records belong to the class GRANTED=*Yes* (this is obtained by moving the mouse cursor to this group bar and getting the statistic from the status region) while the remaining case belong to class GRANTED=*No* (as evident by the colored polygonal line intersecting GRANTED=*No*). We observe that for this exception case, the applicant is unemployed and female (as evident by the colored polygonal line intersecting JOBLESS=*Yes* and GRANTED=*No* and SEX=*Female*).

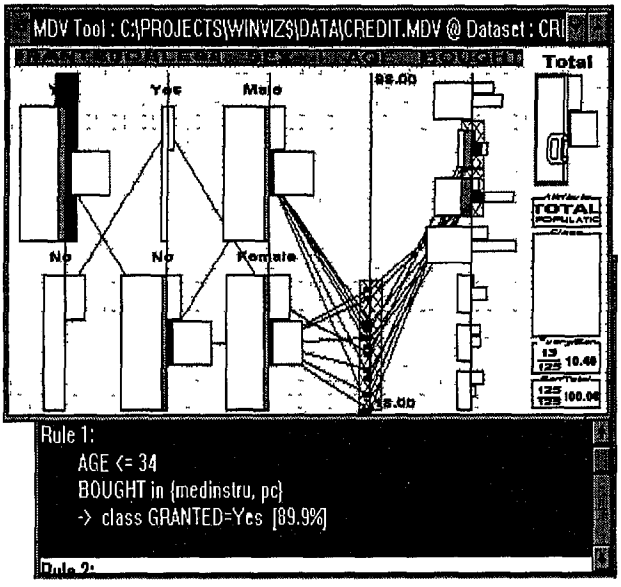

Figure 5: Visualizing a Rule

3.4 Interpretation & Analysis

This stage comprises examining the hypothesis, viewing to see whether the data supports it, refining the hypothesis by querying the dataset, segmenting it where necessary, and generally making sense of what the data says.

3.4.1 Viewing Data. An individual record in the dataset can be represented by a polygonal line that intersect all the vertical axes at the coordinate points (or group bar values). This display mechanism allows the user to identify critical regions of interest indicated by the cluttering of polygonal lines. If the display of the entire dataset shows a large number of lines intersecting a certain region of the AGE attribute, it may be a potential region of interest which the user can zoom in to 'drill down' into the data further. However, as lines can get quite cluttered as the database gets bigger, the group bar display becomes useful to determine the critical areas of interest.

3.4.2 Querying Data The MDV display interface of multi-dimensional data is also the query interface. WinViz is in *query mode* when the mouse cursor changes from being an arrow to being a question-mark as the cursor is moved near any attribute axis. A user can formulate a query interactively using a point-and-click metaphor. There is no need to learn any arcane command-based query language. In SQL, the result of a query is a relation. In WinViz, query results can be visualized by observing the shaded group bars or the polygonal lines.

A visual query in WinViz is specified by clicking the group bars corresponding to the conditions used in the query. For example, clicking the mouse cursor over the group bar where JOBLESS=*No* selects all records relating to non-jobless applicants (Figure 6). A hatched patch appears around the group bar indicating the query condition chosen.

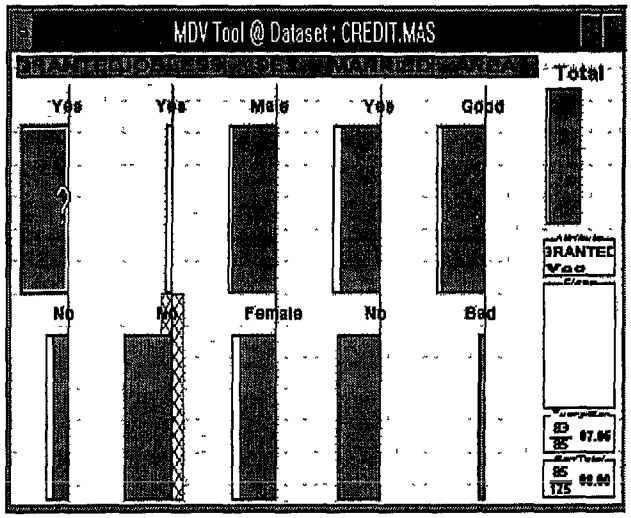

Figure 6: Query on Applicants who are Employed normalized mode

Such a query causes the group bars for the all other attributes to become partially shaded indicating the number of records satisfying the current query. In the normalized mode, the sizes of all the group bars are the same and the shading in each group bar corresponds to the percentage of records satisfying the query within the group. In this case, we observe that 97.65% (as shown in the third box of the status region of Figure 6) of successful applicants have jobs, indicating that employment is an important requirement for credit approval.

WinViz allows one to formulate simple AND and OR type queries. AND conditions are permitted across different attributes while OR conditions are allowed across the values of an attribute. For example, Figure 7 shows a query (GRANTED=*Yes*) AND (BOUGHT=*medinstr* OR BOUGHT=*jewel*). The hatched patches indicate the conditions that have been specified. A more comprehensive discourse of the WinViz query mechanism is described in (Lee et al. 1995).

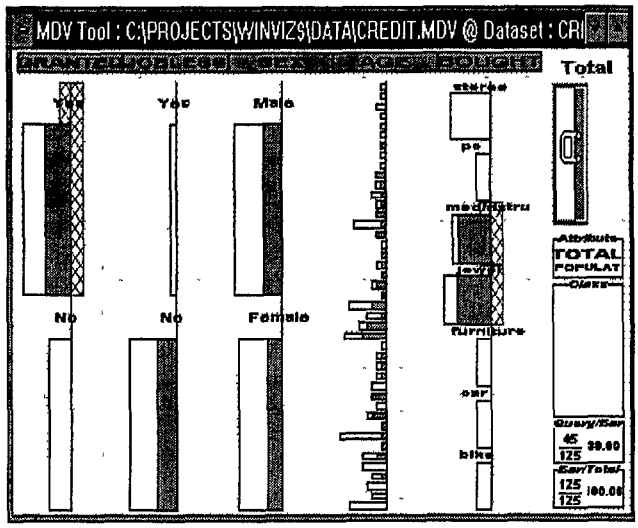

Figure 7: Making Queries on WinViz

3.4.3 Segmenting Data A dataset can be segmented in two ways to facilitate analysis: vertically and/or horizontally. A vertical subset can be created interactively by selecting attributes to be included or excluded from the analysis. Attributes that are added appear on the MDV display while those that are dropped disappear on the fly.

Horizontal segments can be created by discretization of numeric attribute and by defining classes. For the convenience of analysis, it is often useful to refer to a range of numeric values as a group. In the credit screening dataset, AGE is a numeric attribute that, for the purpose of the analysis, can be splitted into several groups such as young adults, adults, and senior citizens, with each associated with a specific age range. Very often such ranges are domain-sensitive. In WinViz, this discretization function can be achieved easily and interactively by a feature called partitioning. For example, attribute AGE can be split into five subgroups each represented by a group bar. The AGE attribute axis now has only five group bars instead of the previous forty-three in Figure 7.

In addition to merely grouping numeric attributes, WinViz also supports the creation of groups, more precisely called classes, based on several attributes. Exploring potential correlations among attributes is supported by allowing the user to interactively classify subjects of interest and see how these classes are represented in the database. For example, we may try to correlate credit approval with other attributes by color coding the applicants according to credit approvals and rejections.

In Figure 8, two classes are formed for GRANTED=*Yes* (represented by darker boxes on the right of the axis) and GRANTED=*No* (represented by lighted boxes on the right of the axis). For all types of items bought except bikes, the chances of getting a credit approval is higher, evident by the longer darker boxes than the lighter boxes along the attribute, BOUGHT. We also observe that although the number of males and females are about the same (as indicated by the widths of the group bars on the left of the axis for SEX), males are more successful at getting credit approvals (as evident by the longer darker box on the right of the axis at SEX=*Male*).

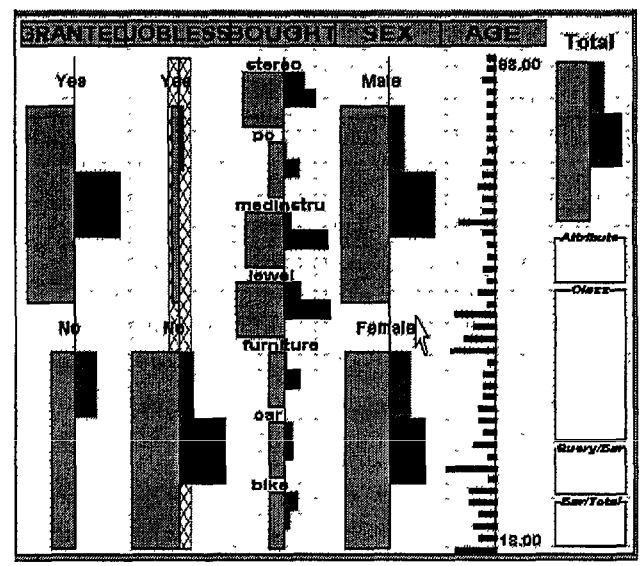

Figure 8: Classification by Credit Approvals

3.4.4 Other KDD Support Operators WinViz additionally provides several operations in support of knowledge discovery; they include:

a. It is unnecessary to load all the attributes for a dataset into MDV at the same time. After all, the human mind is limited in its ability to handle too many attributes and their relationships at any instant. Studies (e.g., (Miller 1956)) have shown that the magic number is 7, plus or minus 2. We can therefore reasonably expect a user to select between 5 and 9 attributes for analysis. Conceptually, there is no limit on the number of attributes that can be displayed on MDV; rather the physical screen size will force the display to be squeezed and hence may appear very small.

b. The order of attribute display may be changed by using a click-and-drag on the attribute label to its intended position on the MDV display. This is useful when the placement of related attributes close together facilitates analysis.

c. Upon data loading, the MDV display is initially in the *unnormalized* mode. The relationship among the group bars can also be viewed through *normalization*, in which all the group bars are forced to the same width but the shadings are adjusted accordingly to reflect percentage. In this way, a particular trend such as, increasing percentage as an attribute value increase, could be easily seen.

4. Status & Future Directions

WinViz has been implemented in C++ for the Microsoft Windows platform on the PC. It currently supports and accepts data using dBase (dbf) file format, Lotus 1-2-3 spreadsheet file format, and IBI Hold file format. To handle very large databases, a client-server version of WinViz is also being developed to provide ODBC support and to download computation to the server.

5. Conclusion

We have presented a system that integrates multi-dimensional visualization, exemplified by MDV in WinViz, and machine learning, through using C4.5. Such a system harnesses the respective strengths of the two technologies to provide support for interactive data exploration and knowledge discovery.

Acknowledgments The development of the MDV technique and the WinViz software have been the work of many talented individuals whose contributions are gratefully noted; they include (in alphabetical order) Chan, S. K., Eickemeyer, J. S., Quek, C. Y., Sodhi, K. S. , Toh, E. W., and Yit, L. L.

References

Bhandari, I. 1994. Attribute Focusing: Machine-Assisted Knowledge Discovery Applied to Software Production Process Control. In Proceedings of AAAI-93 Knowledge Discovery in Databases Wkshp, Seattle, WA.

Brachman, R.J., Selfridge, P.G., Terveen, L.G., Altman, B., Borgida, A., Halper, F., Kirk, T., Lazar, A., McGuinness, D.L., and Resnick, L.A. Nov 1992. Knowledge Representation Support for Data Archaeology. In Proceedings of the Intl. Conf. on Information & Knowledge Management (CIKM '92), Baltimore, MD, pp 457-464.

Brachman, R.J., Selfridge, P.G., Terveen, L.G., Altman, B., Borgida, A., Halper, F., Kirk, T., Lazar, A., McGuinness, D.L., and Resnick, L.A. 1993. Integrated Support for Data Archaeology. *Intl. Journal of Intelligent & Cooperative Information Systems*.

Brachman, R.J., Selfridge, P.G., Terveen, L.G., Altman, B., Borgida, A., Halper, F., Kirk, T., Lazar, A., McGuinness, D.L., and Resnick, L.A. Aug 1994. Integrated Support for Data Archaeology. In Proceedings of the Proc. AAAI-93 Knowledge Discovery in Databases Wkshp, Seattle, WA.

Chomut, T. 1987. Exploratory Data Analysis Using Parallel Coordinates, MSc Thesis, UCLA Computer Science Dept., IBM LA Sc. Cen. Rep. No. 1987-2811,.

Davidson, C. 9 Jan 1993. What Your Database Hides Away. *New Scientist*, pp 28-31.

Grinstein, G., Sieg, J., Smith, S., and Williams, M. Sep 1992. Visualization for Knowledge Discovery. *Intl. Journal of Intelligent Systems*, Vol. No. 7, pp 637 - 648.

Inselberg, A., and Dimsdale, B. 1987. Parallel Coordinates for Visualizing Multi-Dimensional Geometry. In Proceedings of the Computer Graphics Intl. Conf.

Lee, H. Y., Ong, H. L., Toh, E. W., and Chan, S. K., 1995. A Multi-Dimensional Data Visualization Tool for Knowledge Discovery in Databases. Forthcoming.

Mihalisin, T., Timlim, J., and Schwegler, J. 1991. Visualization and Analysis of Multi-Variate Data: A Technique for All Fields. In Proceedings of Visualization '91.

Miller, G.A. Mar 1956. The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information, *Psychological Review*, Vol. 63, pp. 81-97.

Quinlan, R. J. 1993. *C4.5: Machine Learning Programs*. Morgan Kaufmann Publishers.

Terveen, L.G. 1993. Interface Support for Data Archaeology. In Proceedings of the Intl. Conf. on Information & Knowledge Management (CIKM '93).

Wright, M. 1994. Drilling for Data, *InterCity*, pp 38.