

E9 Visualization

E9.1 Use of visualization methods for KDD

E9.1.1 Visualization approaches

The term *Visualization Methods* refers to a collection of tools and techniques from several disciplines. These methods take a collection of data and produce a visual display that gives insight into the structure of the data. Although the term can be used for static or paper displays, it is generally used to refer to visual displays of data shown on a computer screen. Typically the user can interact with a visualization, zooming in and out to different parts of the representation, rotating 3-dimensional views, indicating parts of the views and displaying detailed information on views and view elements, changing view parameters and modifying the display.

One branch of visualization methods is derived from statistical graphics, and is described in detail in [cross ref. C4.1]. These tools concentrate on showing summary properties of the data and exploring relationships between variables. Exploring census data is one example of the sort of task these techniques are most suitable for. Although most early work in the field concentrated on continuous variables, categorical data is now receiving an increased share of attention. Within this discipline, visualization methods are often termed *dynamic graphics*, *interactive graphics*, or a combination of both terms. The former term is often used for views that continuously modify their appearance in a semi-automatic way. Rotating plots, displays of maps that evolve over time and grand tour plots are examples of these type of views. The second term is used more for views that allow a high degree of interaction with a user. They respond rapidly to a user indicating a feature of interest, requesting details on view elements and marking sets of data as potentially interesting. Scatterplot brushing is an early example of these types of plot, as are many views in *DataDesk* (Velleman, 1988).

A second branch of visualization methods is motivated by computer scientists searching for ways of investigating large databases and displaying the results of queries. Typically these methods are highly interactive and focus more on conveying lots of information rather than on exploring relationships. Another goal of these methods is to help the database user formulate a query. Eick, Sumner and Wills (1994) give an example of this kind of visualization. Database queries can be complex, and often the results of a query give either too few results or too many. Visualization techniques can ameliorate this problem by allowing users to display many returned items at once and explore them in order to see how to refine their queries. For these types of problems the phrases *dynamic queries* or *information visualization* are often used.

Scientific Visualization forms a third branch, which is often referred to in its own literature simply as *visualization*. Here, the task is to analyze a body of data that has some geographical or other spatial location. Classic examples of such problems are displaying results of medical scans and analyzing solutions to fluid flow problems. Usually there is a large data set to be analyzed, and the primary goal is to present this large data set in a display that uses the physical location as the most important characteristic, with as much additional data as possible superimposed on this basis. There is some overlap between this goal and the production of interactive geographical maps, but the world of *geographical information systems (GISs)*, in which research is being done on interactive mapping (see Muller 1993 for an overview, and MacEachren 1992 for an example of related work), does not in practice have much contact with the scientific visualization community, whose focus is more on displaying three-dimensional data than flat maps. So far, there has only been limited work within the GIS community on interactive visual techniques, of which Dykes *cdv* (Dykes, 1994) is the best recent example.

A fourth branch concerns itself with visualizations that show the *structure* of data elements, rather than characteristics of those elements. The graph drawing literature (see Di Battista et al., 1994, for a good introduction) is a branch of visualization with a long history, but techniques such as cognitive maps and meta-visualizations – visualizations that show the results of analyses or sequences of data mining algorithms – are attracting recent attention.

E9.1.2 Relationship between KDD and visualization

Both KDD and visualization have the same goal, namely, to take a body of data and extract information or knowledge from it. Both disciplines use computation to manipulate data and analyze it; discovering patterns, pointing out unusual cases, finding relationships and groups, and giving insight into the real-world processes that gave rise to the data. They face many similar challenges, including the following:

- *Large databases.* The need to adapt and develop methods for larger and larger data sets continues to be a key challenge. Methods that are quadratic or worse in the number of cases become a liability when applied to gigabyte-sized databases. Within visualization, methods that attempt to display cases individually on the screen run into the obvious problem with even moderate sized data sets; there are too few pixels available on the screen. Both disciplines are seeking new methods that can process large data sets.
- *Highly multivariate data.* Another size issue is that of exploring databases with many variables. Techniques that work well for five variables can fail dramatically when faced

with a hundred variables. A problem common to both disciplines is deciding which variables influence each other and which can be regarded as extraneous to an analysis.

- *Heterogeneous data.* Examining a single table of data is a relatively straightforward affair. When we have multiple tables of related data, or when we have data in unusual formats, such as text documents, network connections or spatially referenced data, then current approaches often prove inadequate.
- *Accessibility for Domain Experts.* When used in real-world situations, it is vital that any analytic technique be made accessible to people who have little formal training in visualization, data mining, computer science, or statistics, yet who have a great deal of knowledge about the data set being studied. Techniques that domain experts can understand and control are very important.
- *Integration.* As young disciplines, both KDD and visualization started by producing stand-alone tools that experts could use for that purpose only. As these disciplines mature, one challenge is to create tools and methods that can be used with other necessary features in an integrated environment. Database access, web delivery systems, report generation, systems for combining multiple tools and methods -- these are a sample of the challenges facing both disciplines today.

Although similar in goals and challenges, there is clearly a major difference in the approach the two disciplines take. To make a broad generalization, KDD methods attempt to analyze the data as automatically as possible, minimizing human interaction, whereas visualization methods attempt to maximize human interaction so that people's innate analytic ability can be brought to bear on problems. An ideal KDD computer program would, after reading the data, ask a few simple questions of the user and then present the user with a complete answer to their query. An ideal visualization computer program would, after reading the data, present a set of views that the user could interact with and in which they could immediately see valuable information.

The difference is illustrative of how the two disciplines can be used to aid each other. Visualization is a powerful technique for aiding users in understanding the data and suggesting relationships. It is weak at predictive and quantitative tasks. It does not build formal models of the data, but instead suggests models and aids the analyst in deciding what to model. Knowledge discovery techniques tend towards the opposite approach; they are powerful for generating quantitative models and for predictive purposes, but provide little guidance in deciding on the modeling method itself.

The example in section C4.1.4 illustrated how KDD and visualization methods can be used to complement each other. The analyst had a goal in mind at the start; modeling the dependence of

baseball players' salaries on their performances. After building a neural net model, visualization was used to explore the errors in the predictions, and one extremely poor prediction was found to be inexplicable given the available data. This validated the results of the formal analysis and suggested that more data were necessary to explain the wide discrepancy between this one player's predicted and actual salary. The example suggests the following as a basis for inter-disciplinary ties between KDD and visualization.

E9.1.3 Visualization as a pre- and post- modeling tool for KDD

With this motivation, we suggest that visualization should be used both before and after building a formal model. In this paradigm, analysts would first look at data using visual tools to explore the data. They can look for data errors and possible data transformations as well as informally checking that domain knowledge about the data is correct; they would be surprised to see salary decreasing as batting prowess increased in a display, for example. Then, analysts would build a qualitative model in which they hypothesize about plausible relationships. This qualitative model would suggest which formal KDD techniques might be most applicable and the analysis would move to the next stage, fitting a formal model.

After a suitable model has been chosen, analysts can use visualization tools to explore the fit of the model. It might be that this examination of the model would reveal deficiencies in the model that require a new model to be built, or it may indicate that certain subsets of the data are unsuitable for this model and should be treated separately. Several iterations between modeling and exploration of the results of the model may be required. At the end of this phase, visual tools can then be used to present the results and to explain what they mean in terms that domain experts can understand. Again, a domain expert might be able to offer suggestions that would improve the model.

This paradigm has a long history in statistical literature. When performing a linear regression analysis, the student is invariably told to look at the data graphically before the analysis to ensure that the model can be applied, and also to examine the results of the analysis afterwards in case remedial action is required. Neter, Wasserman and Kutner (1985) write: "When a regression model ... is selected for an application, one can usually not be certain in advance that the model is appropriate for that application ... Hence it is important to examine the aptness of the model for the data before further analysis on that model is undertaken." (p. 109). The rest of their chapter is devoted to describing graphical methods for this task. They suggest drawing a scatterplot of residuals against independent variables to check for a variety of problems with the model. They also suggest that "residuals should be plotted against variables omitted from the model ... The purpose of this additional analysis is to determine whether there are any other key independent variables that

could provide important additional descriptive and predictive power to the model” (ibid., p. 120). Unfortunately, while they discuss post-analysis visualization to some length, they underplay pre-analysis visualization. Especially for large data sets and computationally intensive algorithms, modeling and then exploring residuals is an expensive technique for uncovering data problems. Even simple visual exploration will highlight common data quality problems by highlighting unusual values, showing rounding or truncation problems, indicating poor choices of coding missing data values and impossible combinations of variables that might cause an otherwise good model to behave badly.

What is true for statistical models holds equally true for data mining models, if not more so, due to the lesser number of diagnostic statistics that are available for the analyst to use. In sections E9.2 and E9.3 below, the interactions between the disciplines are explored with the thought in mind that visualization and modeling of a domain should proceed hand in hand.

E9.2 Research problems in visualization relevant to KDD

E9.2.1 Visualization of large data sets

As discussed in E9.1.2, exploring data sets with large numbers of cases or large numbers of variables is an important area of research. One approach that has proved fruitful is to pre-process a large database and create a smaller one (by aggregation) that captures a large proportion of the information inherent in the original. One such method is to discretize numerical variables into a few ranges and create a large multi-way table of counts. For example, if we wished examine a database of everyone in the United States, with information recorded on age (*A*), sex (*S*), education (*E*) and income (*I*), we could discretize age into decade ranges, and income into a few ranges. Sex and education are usually recorded as categorical variables, so we can store this information as a table of $A \times S \times E \times I$. With 5 recorded levels of education and 6 income ranges, we would have a table of size $10 \times 2 \times 5 \times 6 = 600$ -- a small sized table. Clearly this approach works well only for few variables as with even twenty variables taking on 3 values each we have over 3 billion cells in the table. It is likely that this table will be sparse and clever allocation methods can be used so that empty cells take up no space, but even so, this method is restricted in the number of variables that can be used.

Mihalisin (1991) has used a modification of this method in which statistical information is stored on pre-defined response variables for each cell in a multi-way table of independent variables. To visualize this table, the user defines a grid, where each grid cell corresponds to a combination of independent variables' values. Figure E9.2.1 is an example of such a layout for the sample data discussed above. To construct the grid, assign each independent variable to an axis. When more

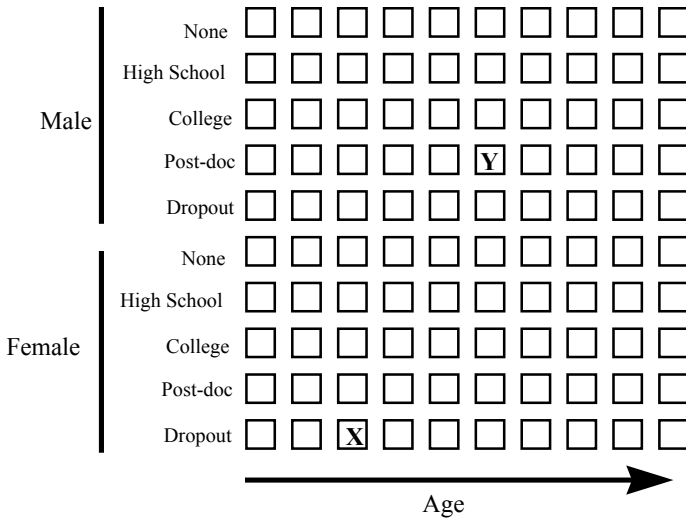


Figure E9.2.1 Multi-way table layout for sample data

than one variable maps to an axis, vary one of the variables within each category of the other. In figure E9.2.1, the education level is varied within each sex category. Within this table cell **X** should produce a representation of the response variable (income) when *age* is in the third level for *females* who *dropped out* of college. Similarly table cell **Y** should represent *males* who completed a *post-doc* who are in the sixth *age* range. The method of representation within a cell can be varied; a small histogram can be displayed, a color to indicate the average

income, or a more specialized representation. If this same display was used to present the results of a decision tree analysis by choosing a cell representation that shows how the model would classify that cell, we might additionally use the cell to display information on the accuracy and support of the corresponding leaf of the tree. Note that this recursive method of defining cell entries to create a grid layout of conditional plot is also used in mosaic plots (Friendly 1994; Unwin et al. 1996 presents an interactive version) and trellis displays (Becker et al. 1996).

The problem of many variables has been approached using parallel coordinates to display up to a hundred variables simultaneously (Inselberg 1990). An example of that method has been introduced briefly in section C4.1. Projection methods where highly multivariate data are projected into two or three dimensions under interactive control have been explored, notably in Cook et al. (1995). Section C4.2 has indicated other approaches to this problem.

Visualization of the results of database queries is another area where the number of variables makes the problem challenging from a research perspective. Shneiderman's dynamic queries (Shneiderman 1994) give the user intuitive one-dimensional controls to set parameters of a query where each control is linked to the rest so that changing a query on, e.g., *education* will alter the

appearance of the control for *age*, so that the relationship between these variables can be taken into account when constructing the query.

Techniques for visualization of large data sets are of fundamental importance to KDD. Models built on large data sets are expensive to calculate, so pre-modeling visualization is vital to ensure that time is not wasted discovering simple data errors. Visualization of the model performance as it iterates to a solution can be useful for a wide variety of algorithms. Again, the goal is to spend compute cycles working on useful models, so understanding how a model is behaving and in what areas it is having trouble allows the analyst to refine techniques more rapidly. In the post-modeling stage, large amounts of input data lead to large amounts of output data, requiring good visual techniques for exploration. Finally, domain experts are not necessarily computer scientists and visual tools that help domain experts understand large, complex models are valuable.

E9.2.2 Visualization of textual data

There are many applications where users want to analyze the relationships between textual documents or within documents. These include applications where users want to search for articles that have a certain set of phrases or are similar to a given document, applications that seek to cluster documents into meaningful groups and applications where insight is sought into the structure of documents. N-gram and similar analysis tools (Suen 1979 [cross-ref. C5.5]) can be used to construct similarities between documents, and Kohonen self-organizing maps have been used to organize documents into 'landscapes' that represent the results of document queries (Kohonen 1995). These landscapes are then typically visualized using a three-dimensional surface rendering which shows the density of documents at each point in the map as a height, with significant peaks being labeled. Such systems are an example of how visualization and data mining techniques complement and enhance each other, with the automatic procedure producing a visual display that can be investigated and explored by a domain expert.

In order to understand structure within text, some systems have been developed for specific areas where the text has certain well-known properties. An example of this is the *SeeSoft* application (Eick et al. 1992), which represents tens of thousands of lines of source code on one screen, coding it by variables associated with its modification history such as date of change, programmer name, purpose of change, etc. The same application has been used to discover relationships between story elements such as key characters in prose. Another challenging growth area in text visualization is the analysis of web pages, which combine well-defined elements with free-form text and images. Web pages also form a network, which is another active area of visualization research.

E9.2.3 Network Visualization

A large important analysis area consists of graphs with multivariate data on both the nodes and links. Telecommunication networks, computer networks, traffic flows and migration paths are examples of such information. Although there is a large body of knowledge on displaying small, static networks (Di Battista et al. 1994), relatively little has been done for large, time-evolving networks. Wills (1998) has created a tool for exploring networks with up to a few millions of nodes and links within a linked windows [cross ref. C4.1.3] environment, but this remains a challenging area for both formal modeling and visualization.

For data mining, rule discovery [cross ref.C5.2] often leads to large sets of rules. Using a network visualization tool to display these rules as connections between the items they relate allows the analyst to understand the interactions between the rules more easily than a list representation would allow.

Another interesting line of research is in exploring ways of visualizing networks that are produced as part of the modeling process itself. Artificial neural nets are a prime candidate as they are a common tool, yet it is often difficult to form intuition about their internal workings. Modifying existing network visualization tools for this purpose might prove valuable for model building, evaluation and sensitivity analysis.

E9.2.4 Visualization of time-based data

Time series data [cross ref. C5.7] are ubiquitous and important. It is surprising, therefore, that tools for visualizing time series have not progressed further than displaying series as simple traces, with little interactivity or ability to modify representations. Visual detection of autocorrelation within a series, or correlation at some lag between series can be performed easily by allowing the user to drag one series over another. Human beings have excellent ability to spot similarities in sections of series this way and the crude method of printing series onto transparencies and overlaying them on paper versions has been used for many years. It is then surprising that computer techniques for doing this and other similar tasks have yet to become more available.

The analysis of sequences of categorical data is another important area. Visual methods and data mining techniques for exploring sequences can be applied to many problem domains including genetic sequencing (Wu et al. 1993) and visualization of multimedia events (Hibino and Rundensteiner 1997).

E9.3 Added value of KDD for visualization

E9.3.1 Enhancement of subtle effects

Within an analysis, some interesting features will be readily apparent in a display. Others are more subtle and might be missed. A common example is a multivariate outlier; a case that does not appear unusual when examined on a variable by variable basis, but does look unusual when combinations of variables are taken. A frequent occurrence in census data is the three-year old with two children. Three-year old children are not unusual and people with two children are common, but the combination is indicative of a data error.

By building models that explain the main effects relating variables, KDD algorithms allow more subtle features to stand out clearly. In section C4.1.4, the neural net drew our attention to a multivariate outlier that had much lower salary than the performance indicated. The rule here is that once the main effects have been noted, a model should be built that accounts for these effects so that the next level of effects can be explored visually.

E9.3.2 Variable selection and guided visualization

With many variables, the user of a visualization system is often overwhelmed. It is hard to decide what to explore and where to start. Ways of helping the user decide which variables to consider initially are needed. One possibility would be to run some naïve, fast data mining algorithm for a large number of variables, and let the user look at the results to decide which relationships might need exploring in more detail. By looking at the rules in a decision tree, the user would be able to see which variables are most useful for predicting a given categorical variable. This gives an initial starting point from which a visual exploration can be launched. In neural net analyses, this is a common methodology already. The user builds a model using all of the variables and then discards variables until the model deteriorates. A similar process in statistics is termed *stepwise regression* (Neter et al., 1985, section 12.4).

A related problem is aiding the user when they have created a view and would like to explore it in more detail. Context sensitive help such as in *Data Desk* (Velleman 1988) is useful, but what is really needed is for some analytic engine to explore possible variations of the plot and point out useful ones to the user. A simple example would be to look at a matrix of scatterplots and suggest which additional variables have strong correlations with the selected ones and so might be added to the display.

E9.3.3 Confirmatory analysis

One of the strongest and most persistent criticisms of visualization methods is the lack of quantitative results, often phrased along the lines of "I can't make a million-dollar decision based on a pretty picture." Sometimes making decisions based on displays is justified -- especially when the display is unambiguous (for an excellent example, Tufte 1997, p. 45 gives a plot of O-ring damage against temperature that, had it been created, would have shown immediately the danger of the Challenger shuttle launch). Often that is not the case, and the user is justified in asking whether the pattern they see has any real meaning, or if it is an artifact or product of random effects.

KDD methods can be used to take a qualitative model and give it quantitative meaning. If a series of views indicates a certain model for how the data interact, the user should be able to create such a model and see if there is any justification for it. Sometimes this might involve a fair amount of work for the user if they wanted to quantify a complex model. But often the questions are more of the order of "I see what appear to be several clusters in this scatterplot. Are they really there?". In such cases simple algorithms can be run to help determine if that is the case. An ideal system for data mining will allow the user to confirm any visual pattern they detect, and to visually explore the results of every model they create.

References

- Becker, R.A., Cleveland, W. S. and Shyu, M. J. (1996) The Design and Control of Trellis Display
Journal of Computational and Statistical Graphics **5**: 123-155
- Cook, D., Buja, A., Cabrera, J. & Hurley, C. (1995) Grand Tour and Projection Pursuit *Journal of Computational and Graphical Statistics* **4** #3: 155-171
- Di Battista, G., Eades, P., Tamassia, R. and Tollis, I. (1994) Algorithms For Drawing Graphs: An Annotated Bibliography *Computational Geometry* **4**: 235-282
- Dykes, J.A. (1998) Cartographic Visualization: Exploratory Spatial Data Analysis with Local Indicators of Spatial Association using Tcl/Tk and cdv. *The Statistician* **47** #3: 485-497
- Eick, S., Steffen, J. and Sumner E. (1992) Seesoft - a Tool for Visualizing Line Oriented Software Statistics *IEEE Transactions on Software Engineering* **18/11**: 957-968

- Eick, S., Sumner, E. and Wills, G. (1994) Visualizing Bibliographic Databases. In *Database Issues for Data Visualization*, 186-193. Springer-Verlag (Lee and Grinstein, eds.)
- Friendly, M. (1994) Mosaic displays for n-way contingency tables *Journal of the American Statistical Association* **89**: 190-200
- Hibino, S. and Rundensteiner, E.A. (1997) Interactive Visualizations for Temporal Analysis: Application to CSCW Multimedia Data *Intelligent Multimedia Information Retrieval*: 313-335 ed. Maybury, (MIT Press Boston, MA)
- Inselberg, A. and Dimsdale, B. (1990) Parallel Coordinates: A Tool for Visualizing Multi-Dimensional Geometry *Proceedings of the First IEEE Conference on Visualization*: 361
- Kohonen, T. (1995) *Self-Organizing Maps*. Springer, Berlin.
- MacEachren, A. (1992) Visualizing Uncertain Information *Cartographic Perspective* **13**: 10-19
- Mihalisin, T., Timlim, J & Schwegler, J. (1991). Visualization and analysis of multi-variate data: A Technique for All Fields *Proceedings of 1991 IEEE Visualization Conference* eds. G. M. Nielsen & L. Rosenblum: 171-178. Los Alamitos, CA
- Muller, J.-C. (1993) Latest Developments in GIS/LIS *International Journal of Geographic Information Systems* **7** #4: 293-303
- Neter, J., Wasserman, W. and Kutner, M. (1985) *Applied Linear Statistical Models* (Irwin, Homewood IL)
- Shneiderman, B. (1994) Dynamic Queries for Visual Information Seeking *IEEE Software* **11** #6: 70-77
- Suen, C.Y. (1979) N-gram statistics for natural language understanding and text processing *IEEE Transactions on Pattern Analysis and Machine Intelligence* **1** #2: 164-172
- Tufte, E. (1997) *Visual Explanation* (Graphics Press, Cheshire, CO)

Unwin, A., Hawkins, G., Hofmann H. and Siegel, B. (1996) Interactive Graphics for Data Sets with Missing Values - MANET *Journal of Computational and Graphical Statistics* **5** #2: 113-122

Velleman, P.F. (1988) The Datadesk Handbook (Odesta Corporation)

Wills, G. (1998) Nicheworks – Interactive Visualization of Very Large Graphs *Journal of Computational and Graphical Statistics*: to appear

Wu D., Roberge J., Cork D., N. Bao and Grace T. (1993) Computer Visualization of Long Genomic Sequences *IEEE Visualization* **93**: 308-315