DSC 2001 Proceedings of the 2nd International Workshop on Distributed Statistical Computing
March 15-17, Vienna, Austria
http://www.ci.tuwien.ac.at/Conferences/DSC-2001
K. Hornik & F. Leisch (eds.) *ISSN 1609-395X*

What Dataminers Want

Martin Theus*

Abstract

Dataming is obviously a "buzzword", which does not really describe an academic discipline, but more a field of application of the established disciplines statistics and computer science. Both disciplines do not really cover what dataminers need and lack a certain part.

This paper trys to illustrate what the traditional disciplines miss and why new pseudo disciplines like Knowledge Discovery in Databases (KDD) arise.

Several aspects of statistical computing like data management and graphical interrogations are investigated in the light of the daily business of a dataminer. Finally the question comes up, whether research in statistical computing focuses on these topics, which might influence commercial software development into the right direction.

1 The Curse of Flat Files

Statistics, especially mathematical statistics, grew up about 100 years ago. In those days datasets obviously were very small, because any processing of these data was done manually. The era of mathematical statistics just ended, when statistical computing arose. Still electronic storage was very expensive and dataset sizes were very limited. Since statistics is traditionally a topic "owned" by math and not by computer since, the primary focus of statistics is on the underlying mathematical theory and less on the computing skills. This is certainly the right balance.

Nevertheless when designing and writing statistical software, a profound knowledge of computer science is necessary. So far, statistical software is usually good a reading, respectively import flat ASCII files. Computational statistics unfortunately ignored the presence of data sitting in databases for too long a time. In the

*VIAG Interkom, P.O. Box: 80260 München, martin@theusRus.de

early 90s, the number of huge databases grew more and more. This data was mostly collected electronically or entered at distributed locations. Governmental agencies, big retailers or online traders now face huge databases, which were hard to analyze beyond simple summaries. At this point computer scientists took advantage of this lack of database knowledge of statisticians and "invented" a discipline called KDD (Knowledge Discovery in Databases). But is KDD a concurrent discipline to statistics? Since KDD is about gathering data, sampling data, experimental design, analyzing data, modeling, discovery, visualization, forecasting and classification it "lives" in the statistics domain. Mainly a lack of apropriate tools brought computer scientists into play. As Daryl Pregibon [4] put it:"KDD = Statistics at Scale and Speed".

The Datamining Process

Although there are several "all-in-one" datamining tools on the market, the typical "Datamining Process" (of a statistician) often is performed in four steps as illustrated in figure 1.

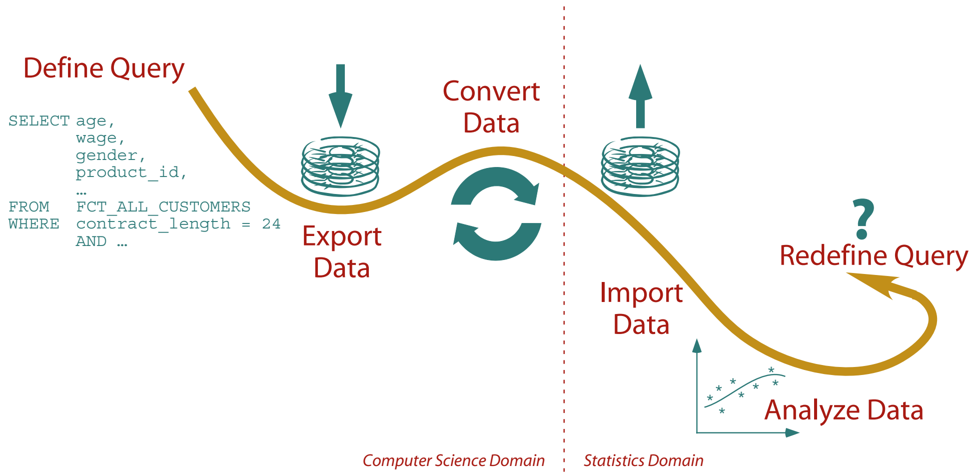


Figure 1: The usual datamining process

1. Define the SQL query, which gets all(?) the data out of the database we want to analyze.
2. Export the data to a flat file
3. (Optionally convert the flat file to a format the data analysis tool is inclined to import)
4. Import the data into the data analysis tool
5. Analyze the data in your preferred tool.

At the end of this process we might find out that we did not get all relevant data out of the database, so we are forced to redo the whole process. This is also true if the underlying data changes in the database and we need to update the results.

Is ODBC/JDBC all that bad?

Obviously interfaces like ODBC/JDBC are not really fast. Nevertheless, these interfaces are universal and enable to connect to a variety of data sources. Although industrial strength software must use native database interface to transfer huge amounts very quickly, research software can facilitate the more universal interfaces (ODBC/JDBC) in order to start taking advantage of direct database connections. Then ther would no longer be the need to use flat files as a temporary storage/exchange method.

Interactivity vs. Databases

Exploratory Data Analysis (EDA) is an interactive process by its nature. Thus software supporting EDA must help enabling this interactive process. When working with data inside databases the usual interaction is ”to wait”. To be able to achieve fast response times from databases, one must take a lot of care of setting up indexes and optimize queries carefully. But this is an operation which must take place inside the database and is hard to generalize to arbitrary datasets. Nevertheless, if the size of the datasets gets too big, one cannot handle it outside a database.

Many graphical representations of data like e.g. Boxplots, Barcharts and Mosaicplots or even Scatterplots can be drawn with only a summary of the underlying data. These summaries usually can be collected easily by database queries. Another crucial point when directly working on databases is to store additional information like group attributes (selected, color, etc.) inside the database. The back propagation of this information (e.g. color all male customers under 30 with high wages in red) from a graphical front-end into the database is usually a bottleneck.

2 Interactive Statistical Graphics

Interactive Statistical Graphics is no new technique. It comes to a renaissance with datamining, since most of the classical mathematical statistical techniques deteriorate when working with too big datasets. Although even graphical displays might be no longer sensible with bigger datasets (the number of outliers to plot in a boxplot grows linearly with the sample size, which obviouly makes no sense) graphical representations of data are often the only means to get an understanding of the data.

Interactive Statistical graphics can be characterized as follows:

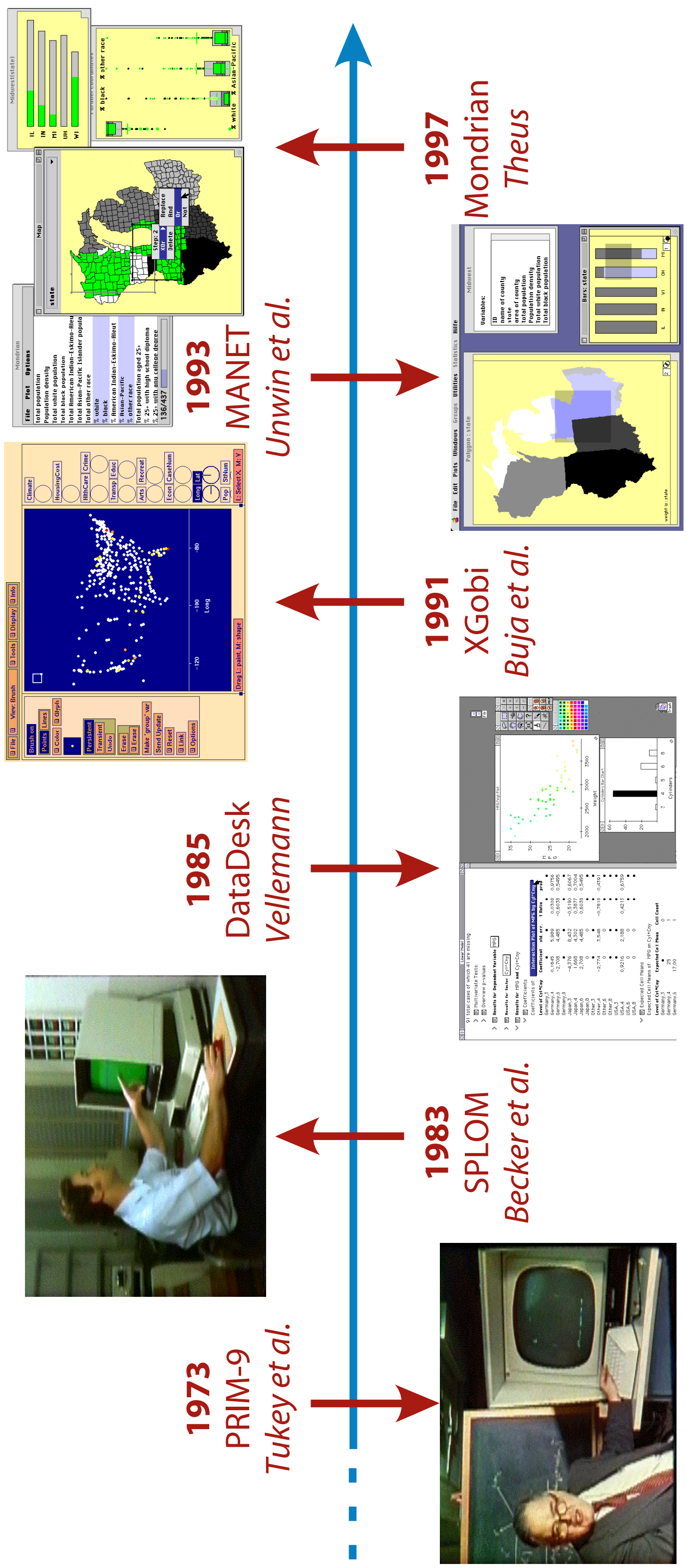


Figure 2: A brief history of interactive statistical graphics systems

what it is	what it **not** is
• about graphical user interfaces	• about interactive computing
• about direct manipulation of statistical/graphical objects	• about manipulation of code
• about linking analyses & graphs	• about static
• about exploring data	• about presenting results
• pro **E**xploratory **D**ata **A**nalysis	• contra mathematical statistics

Figure 2 shows a brief history of several interactive statistical graphics systems. Most of them are research software, only Data Desk is commercial software and has industrial strength.

All recently build systems proof, that datasets with up to a million observations still can be handled with interactive graphical methods. Unfortunately commercial software like SAS-Insight or SPSS-Clementine (which directly connects to a database) deteriorate above 50.000 data records and thus are not really usable for datamining purposes.

3 Selections for Drill Down

Determine and selecting subsets of data is a basic datamining task. Obviously graphical methods support this task more easily than standard SQL-queries.

Evolution of Selection Techniques

The way how data are selected in interactive statistical graphics software is maybe the best way to show the evolving research results.

1. The *standard* way of selecting data is to select data and by doing so replacing any other selection that might have been present. There is no way of refining a selection or selecting over different plots and/or variables. This standard selection technique is implemented e.g. in XGobi [1].

2. A more *advanced* way to handle selections is to allow to combine the current selection with a new selection with boolean functions like *and, or, Xor, not.* This allows the analyst to refine a selection step by step to drill down to a very specific subset of the data. DataDesk [6] implements this selection technique.

3. When dealing with a whole *sequence* of selections, it is often desirable to change a selection at an earlier stage, without having to redefine all preceding and successive selection steps. By storing the sequence of selections it is possible to make changes to any step in the sequence. Selection Sequences have been first implemented in MANET [5].

4. Although a selection is always performed on the computer screen in the first place, i.e. in terms of screen coordinates, the data selection must be stored in terms of data coordinates. The approach used by Mondrian [3] keeps a list of any selection associated with a dataset. For each entry in the list the

- selection area in screen coordinates and data coordinates,
- selection step,
- the corresponding plot window and
- the selection mode (i.e. and, or, not etc.)

is stored. The currently selected subset of the data can then be determined by processing all elements of the list, no matter which kind of modification to the list the reason for an update of the selection subset was.

Selection Rectangles

Allowing multiple selections in a single window as well as across different windows makes a visual guide to the performed selections indispensable.

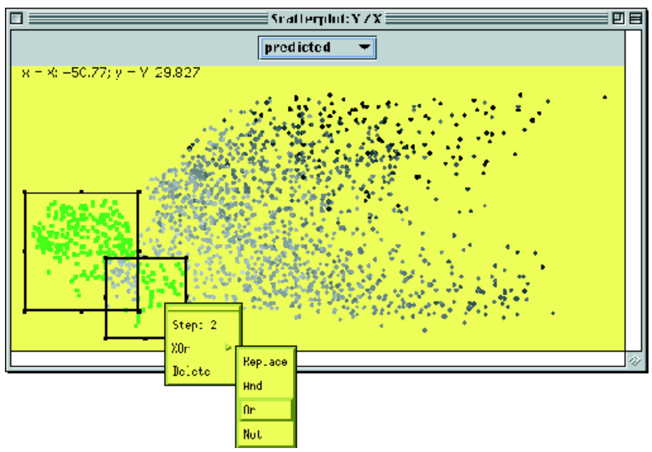


Figure 3: *Selection Rectangles in* Mondrian.

Mondrian introduces *Selection Rectangles*. Figure 3 gives an example of a scatterplot containing two selection rectangles. Selection rectangles indicate the area which was selected. An existing selection rectangle can be used as a brush by simply dragging the selection rectangle The eight handles at the rectangle allow for a flexible resizing of the rectangles or a slicing (i.e. the systematical enlargement or reduction of a selection along one dimension).

The selection mode can be maintained via a pop-up menu. The deletion of a selection can be performed via this pop-up, too. The last selection which was performed can be deleted by simply pressing the backspace key.

Only the last selection is plotted in black. Selections performed at an earlier stage are plotted in a lighter gray to make them less dominant in the plot.

Since selections are stored in terms of the data coordinates they are invariant to any alterations of a plot. Typical scenarios are things like interactive reordering of the axes in parallel coordinate plot, flipping the axes in a scatterplot or zooming a view. These operations automatically update the selection rectangles. The new

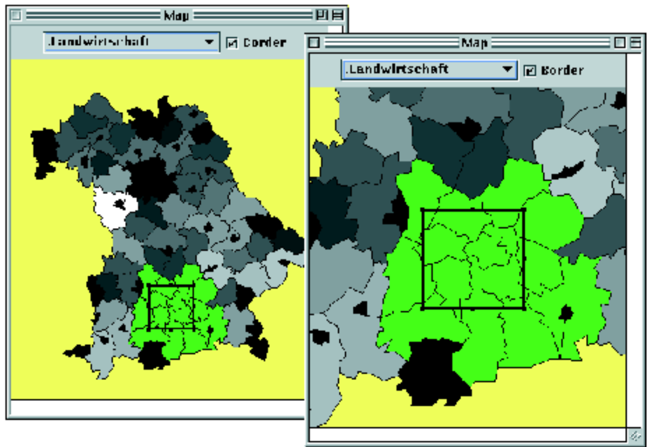


Figure 4: *The Zoom Operation in the window does not affect the selected data, but the Selection Rectangle.*

screen coordinates of the selection rectangles are calculated from the data coordinates. Figure 4 shows how a selection rectangle reacts on a zoom inside a map.

The ability of handling more than one selection in one window is indispensable when dealing with parallel coordinates.

3.1 Translating Graphical Selections into SQL

Using a graphical front-end to query databases makes it necessary that graphical queries can be translated into SQL-queries. This task is actually relatively easy, since every selection can directly be translated into restricting `where`-clauses. The

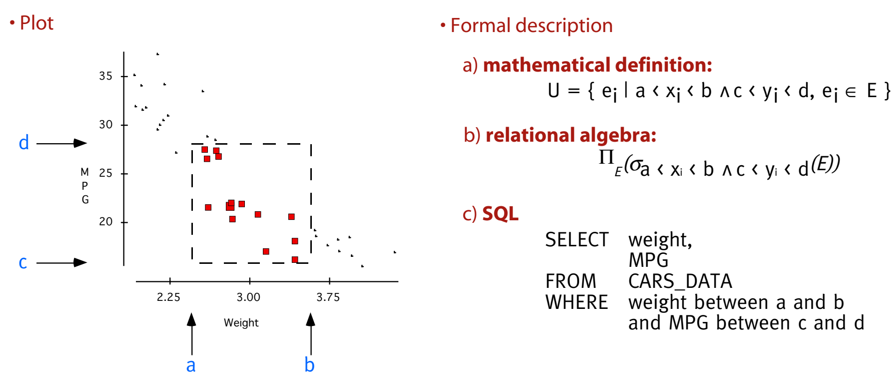


Figure 5: *Translating graphical selections into SQL.*

different selection modes like *and, or, xor* or *not* are used to combine these parts in the `where`-clause. Figure 5 shows the trivial example of a selection in a scatterplot.

Translations get more complex when selecting in highdimensional rotation plots. Since the performance of SQL-databases varies very much depending on the actual `where`-clause supplied, an optimizer might be needed to get a decent response-time from the system.

4 Working with Categorical Data

The major proportion of data in databases are categorical. A popular method to deal with categorical data is to condition a dependent, mostly continuous, variable upon the different categories of the influencing variables. But these plots, known as Trellis Displays, can not really deal with categorical data itself, and are very limited when the number of categories is greater than 10 — which is very likely in huge databases.

To deal with categorical data datamining tool must implement interactive barcharts and mosaic plots. Both plots are not very revealing in a static setting, but are very insightful in an interactive environment providing linked highlighting and interactive reordering of variables and categories. Especially when the number of categories gets high, efficient sorting mechanisms are very useful.

Housing Factors			Housing Type			
Satisfaction	Influence	Contact	App.	Atr.	Terr.	Tower
low	low	low	61	13	18	21
		high	78	20	57	14
	medium	low	43	8	15	34
		high	48	10	31	17
	high	low	26	6	7	10
		high	15	7	5	3
medium	low	low	23	9	6	21
		high	46	23	23	19
	medium	low	35	8	13	22
		high	45	22	21	23
	high	low	18	7	5	11
		high	25	10	6	5
high	low	low	17	10	7	28
		high	43	20	13	37
	medium	low	40	12	13	36
		high	86	24	13	40
	high	low	54	9	11	36
		high	62	21	13	23

Table 1: The Housing Factors Data: Cross-classification of 1681 tenants

The *Housing Factors* Example

The *Housing Factors* example shall underline why interactivity is a key-feature for a graphical exploration of categorical data. The data is taken from Cox & Snell's [2] investigation. The interaction presented are implemented in Manet and Mondrian.

Data on the housing situation of 1681 tenants has been classified according to:

- **Housing Type**
  Apartments, Atrium House, Terraced House, Tower Block
- **Influence on the housing situation**
  low, medium high
- **Contact to other tenants**
  low, high
- **Satisfaction with the housing situation**
  low, medium, high

The data is distributed over all 72 cells, i.e. there are no empty cells with no observations. Table 1 lists the complete data set.

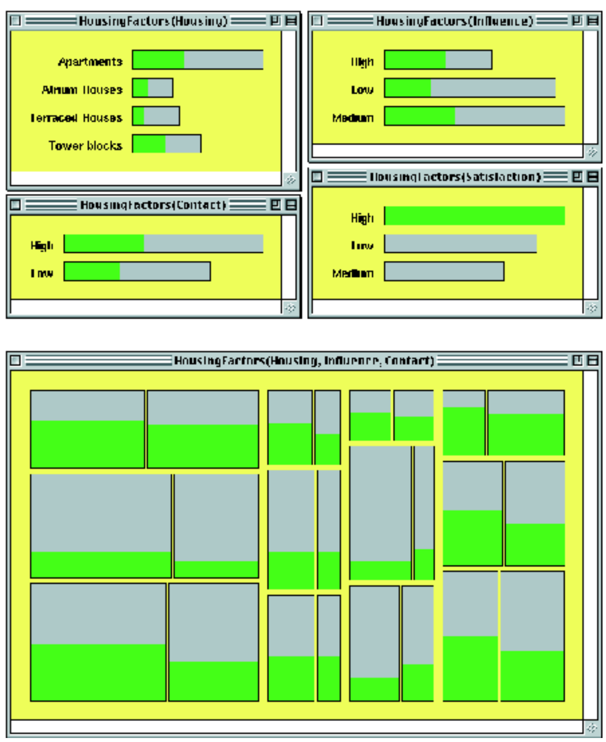
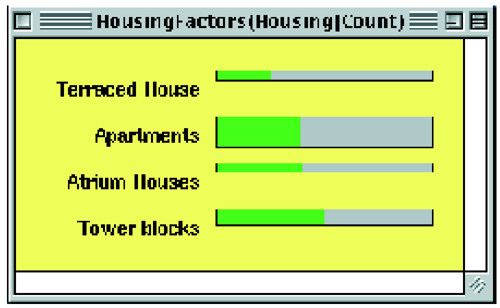


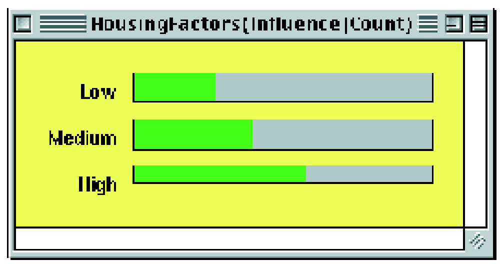
Figure 6: *The Housing Factors Data in the default view.*

Figure 6 shows the default barcharts and mosaic plots for the four variables. The cases with high satisfaction are selected, to mark the most interesting response. Obviously the order of at least two of the variables make no sense, and the mosaic plot does not reveal any systematic pattern, worthwhile to fit a model for. The necessary steps to make the plots more insightful comprise:

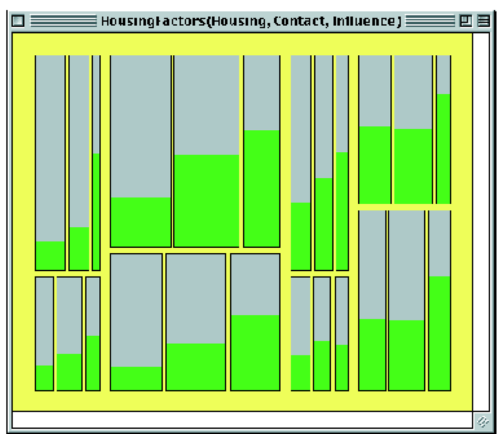
- Sort the categories of Housing Type according to the relative amount of high satisfaction cases (via the plot option pop-up). The plot has been switched to the Spineplot view, to make the sorting more obvious.



- Sort Influence and Satisfaction to: *low, medium, high* (via alt-click and drag):



- Reorder the variables in the mosaic plot such that the plot is conditioned upon the Housing Type and as a variable with many categories at the deepest stage. The order is then: Housing Type, Contact, Influence. The reordering is done with the four arrow keys.



Certainly it is much harder to read the plots without the interactive queries. But in contrast to the default views, the reordered plots now reveal a clear pattern along with some deviations, which can now be investigated more closely using statistical models as well as the suitable meta-information.

5 Summary

This paper tried to show what statistical computing still needs to face the challenges of datamining. The ability to handle data located in databases is a very important feature which many statistical software tools still lack. Working on data in databases with interactive tools seems to be much harder as on data in "private" memory of the application itself.

Only if statistical software works on databases as seamless as on flat files, the domain of Knowledge Discovery in Databases (KDD) can be gained back from computer scientists.

In the light of massive datasets interactive graphical methods seem to be a good choice to analyze the structure of data, especially of data which are mostly categorical. Relatively simple operations like sorting, joining or splitting categories and reordering of variables in Barcharts and Mosaic Plots are very helpful.

Selection techniques well known from interactive statistical graphics can easily be used as a graphical selection front-end for data which is located in databases.

In my subjective view *"What Dataminers Want"* is a seamless integration of interactive statistical graphics with advanced selection techniques directly working on data in databases.

References

[1] A. Buja, D. Cook, and D. F. Swayne. Interactive high-dimensional data visualization. *Journal of Computational and Graphical Statistics*, pages 78–99, 1996.

[2] D. R. Cox and E. J. Snell. *Applied Statistics — Principles and Examples*. Chapman & Hall, London, 1991.

[3] Mondrian. http:\\www.theusRus.de/Mondrian, 2001.

[4] Daryl Pregibon. 2001: A statistical odyssey. *KDD Conference '99*, 1999.

[5] M. Theus, H. Hofmann, and Wilhelm A. Selection sequences — interactive analysis of massive data sets. In *Proceedings of the 29th Symposium on the Interface: Computing Science and Statistics*, 1998.

[6] Paul F. Velleman. *DataDesk Version 6.0 — Statistics Guide*. Data Description Inc. Ithaka, NY, 1997.