

# Trellis Displays vs. Interactive Graphics

Martin Theus

Lehrstuhl für Rechnerorientierte Statistik und Datenanalyse  
Institut für Mathematik der Universität Augsburg  
86135 Augsburg, Germany  
email: theus@uni-augsburg.de

## Summary

The number of variables that can be displayed simultaneously in statistical graphics is very limited. Even a rotating plot cannot visualize more than three continuous variables at once.

With Trellis Displays Becker, Cleveland and Shyu offer the possibility of combining up to eight variables (continuous or categorical) in one plot.

This kind of visualizing data raises the question of how far we can obtain equivalent results with the techniques of interactive statistical graphics. The method of linking plots together seems to be a more flexible way of analysing high dimensional data.

In the end Trellis Displays are a good means of presenting results of interactive statistical graphics in a static way.

**Keywords:** Trellis Displays, highlighting, linking

## 1 Trellis Displays

### Definition

Historically Trellis Displays are based upon the so-called *Co-Plots*. They were first mentioned in (Chambers 1992). “Co” stands for “conditioning”, what means that a specific plot is drawn for different subsets of a conditioning variable.

The generalization to Trellis Displays is presented by William S. Cleveland in (Cleveland 1993), even though he does not yet use the name Trellis Display at all.

In (Becker 1994a) and (Becker 1994c) Becker, Cleveland and Shyu describe the setup of Trellis Displays by means of an example. Trellis Displays

are now available as an S-Plus version 3.2 library (see (Mathsoft 1994)).

The core of such a display is the one-, two- or three-dimensional graphics of the so-called *axis variables*. Those plots are designated as *panel functions*. The kind of plot is not limited any further; they can be dot-plots, scatter-plots, box-plots, surface-plots etc. .

Besides the *axis variables*, there can be up to three *conditioning variables* chosen, to build the trellis. These are either categorical, or they have to be subdivided into several overlapping intervals. Those variables, called *shingles*, are than interpreted as categorical. For each category (or interval) or combination of categories the *panel function* is called. The plots are arranged in a vector (one conditioning variable) a matrix (two conditioning variables) or a set of parallel matrices (three conditioning variables).

Finally two more categorical or *shingle* variables, called *adjunct variables*, can be displayed by using different colors and markers.

It is easy to see, that the number of categories of the *conditioning variables* is limited to about eight to ten. The limitation of categories with the *adjunct variables* is much tighter. Using more than three or four colors or markers would confuse the viewer of the plots too much.

A critical point of the design of Trellis Displays is the scaling of each *panel function*. Trellis Displays guarantee that all plots have the same scale, thus enabling the comparison of the plots.

## An Example

Figure 1 shows a Trellis Display of the dataset *Barley Yield* — see (Cleveland 1993) and (Becker 1994c). Measured was the yield of ten different barley species at six sites in the years 1931 and 1932. The following variables were measured:

1. yield: **continuous**
2. species: **10 categories**
3. site: **6 categories**
4. year: **2 categories**

The assignment of the specific elements of the Trellis Display was made as follows:

1. *Panel function*: Dot-Chart, see (Cleveland 1985)
2. *Axis variable*: yield by species
3. *Conditioning variables*: rows: site, columns: year

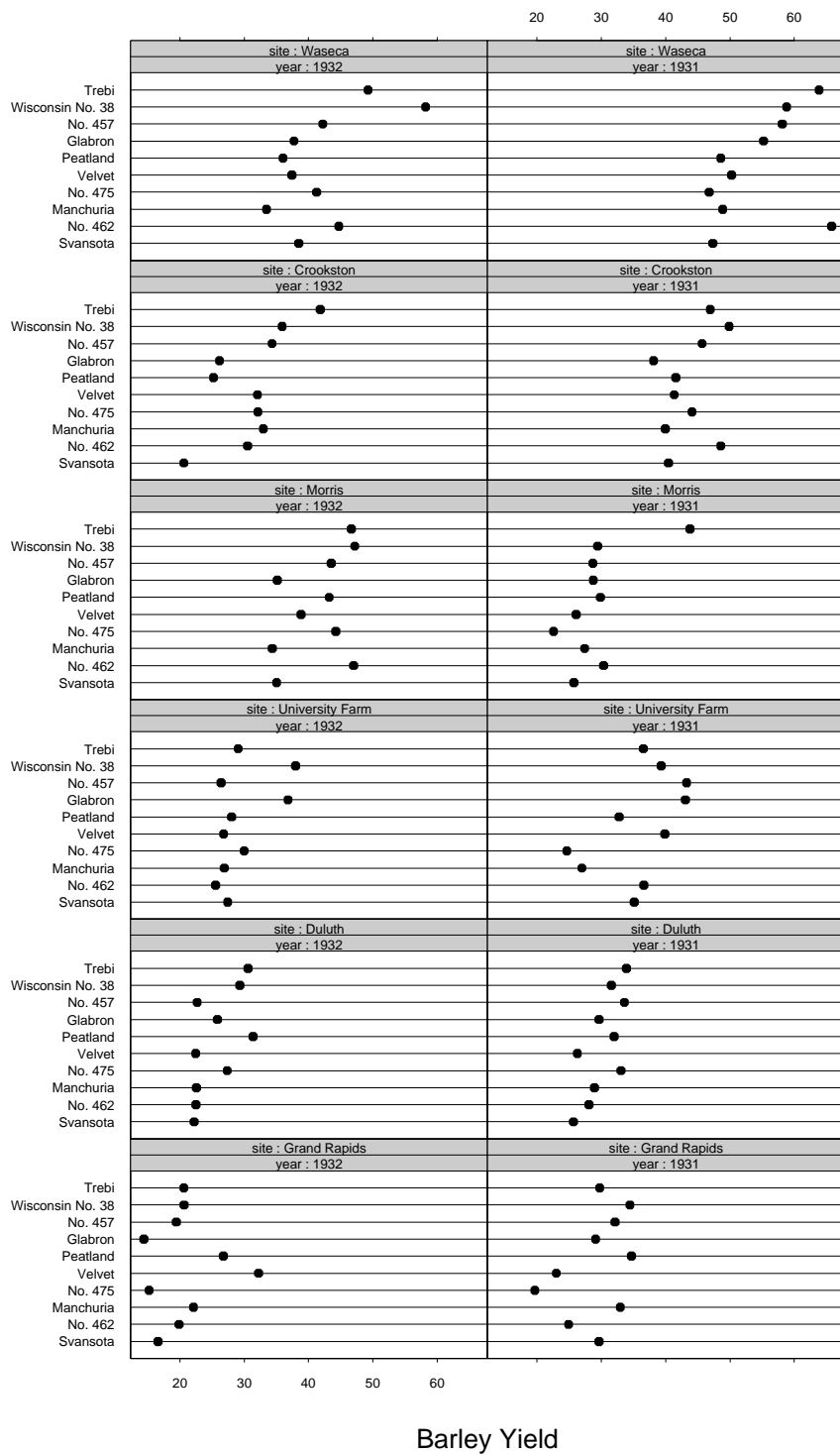


Figure 1: A Trellis Display for the barley data

4. *Adjunct variables*: none.

Later on we will see the meaning of the assignment.

### “Trellising”

*Trellising* means the selection of the different kinds of variables and the layout of a Trellis Display.

In a first step the variables of the dataset have to be assigned to the three groups of variables, namely

1. *Axis variables*
2. *Conditioning variables*
3. *Adjunct variables*.

This process is hard to motivate without a pre-knowledge of the data. Generally the following proceeding is advisable:

- As *axis variables* those variables should be selected, where we expect the most dependence on other variables; respectively those where we expect the greatest variation.
- Choose as *conditioning variables* those variables that have a great influence on the other variables.
- *Adjunct variables* should always be “real” categorical variables, because *shingles* could not be interpreted here.

In general it is often necessary to explore the data with some one- or two-dimensional plots, in order to find the right assignment of the variables.

Figure 2 and 3 show us two permutations of the Trellis Display shown in figure 1. It is very hard to compare the results with those of figure 1.

While fixing the layout of a display the number of categories of the *conditioning variables* is very meaningful. This number, and the order of the variables, determine the dimensions of the Trellis. This results in triples of the form  $(m, n, l)$ , where  $m, n$  und  $l$  each denote the number of categories of one of the three *conditioning variables*. In figure 1 the triple (site, year, –) was chosen, which yields us a  $(6, 2, 0)$ -matrix.

It is also possible to modify the layout of a Trellis Display manually, i.e. to vary the specific position of the plots inside the Trellis. This can always be useful, if the automatic order of the plots does not correspond to the subject specific background.

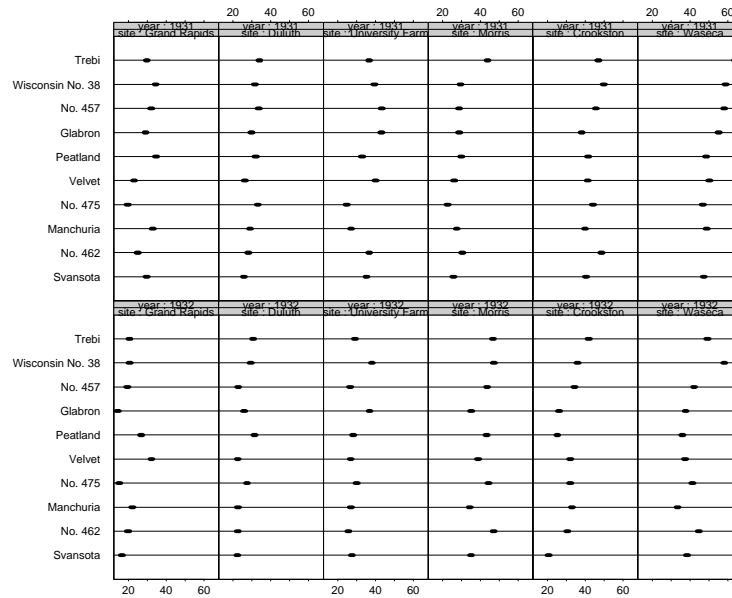


Figure 2: Variation of the variable assignment

## User Interface

It is very important for efficient data analysis that it can be carried out quickly, easily and intuitively. Trellis Displays are available within S-PLUS, but that is a command-line analysis system without the ease of use of a mouse driven system.

The abilities of S-PLUS are the means of programming own functions, but not the interaction with the user.

The Trellis Displays library offers a userfriendly syntax to specify a display. The call for figure 1 is just:

```
> dotplot( yield ~ variety | year * site , data=barley)
```

This input is to a great extent intuitive: *Yield* is displayed by *species* in a dotchart. *Year* and *site* form the columns and rows of the Trellis. The data are taken from the dataset *barley*. The inputs for the other figures are analogous — there the independent variables are merely permuted. The above syntax is taken from the statistical modelling language in S-PLUS. A more precise description can be obtained from (Chambers 1992).

If you want to change or extend the look of Trellis Displays, the *panel functions* have to be modified in a more complicated manner. Calling up a figure analogous to figure 1, where the two years are plotted in one plot-

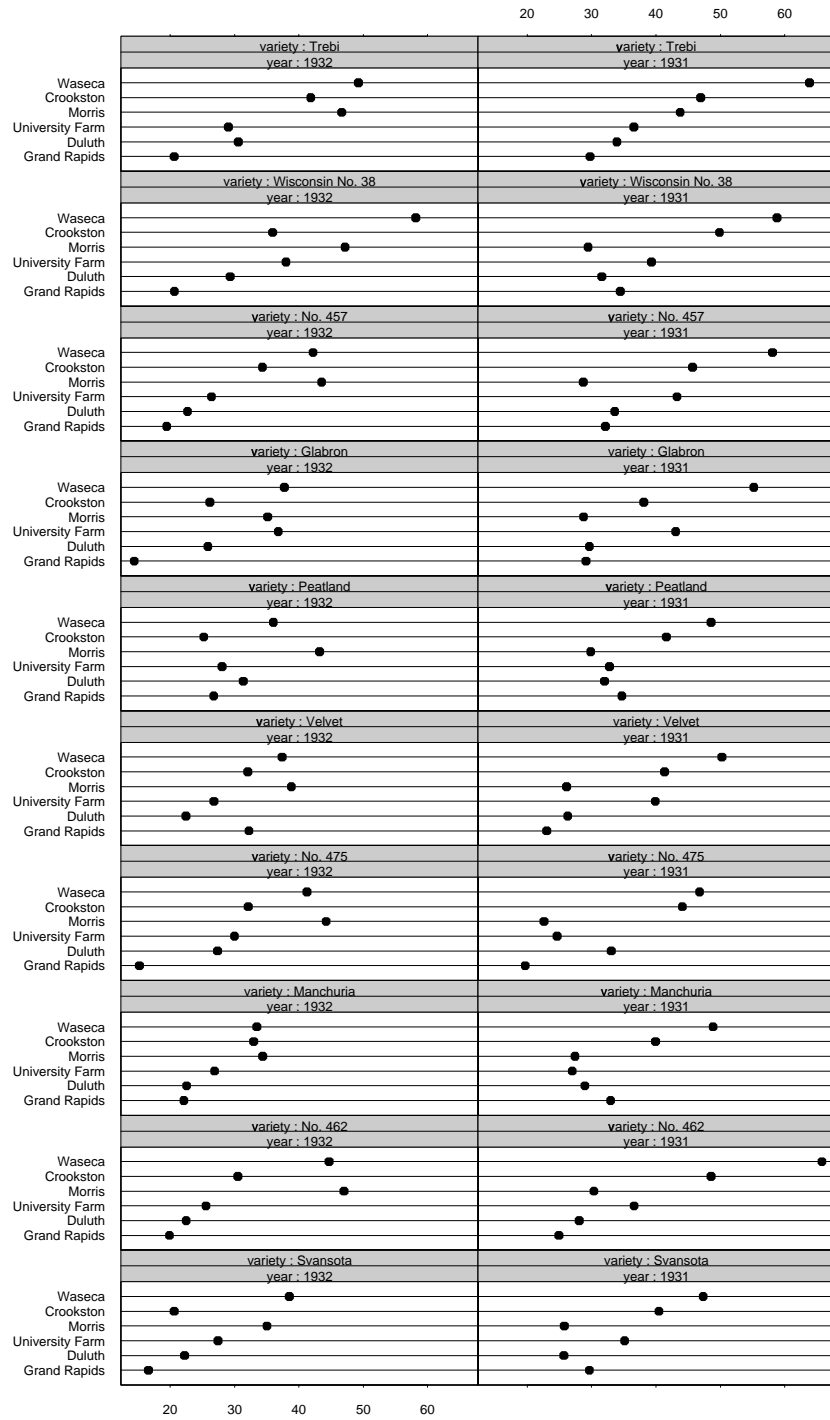


Figure 3: Variation of the variable assignment

chart with two different markers, i.e. the variable year is added as an *adjunct variable*, looks as follows:

```
> attach(barley)
> superpose.symbol <- trellis.par.get("superpose.symbol")
> n <- length(levels(year))
> dotplot(yield ~ (variety | site),
          panel = function(x, y, subscripts)
            {
              dot.line <- trellis.par.get("dot.line")
              abline(h = y, lwd = dot.line$lwd,
                    lty = dot.line$lty,
                    col = dot.line$col)
              panel.superpose(x, y, subscripts,
                              factor.groups = barley$year)
            }
          subscripts = TRUE,
          layout = c(1, 6),
          aspect = 0.5,
          xlab = "Barley Yield (bushels/acre)",
          key = list(points = Rows( superpose.symbol, 1:n),
                    text = list(levels(year)),
                    side = "Right",
                    columns = n))
```

Such inputs are more than most users can handle, even if they offer a great amount of extendibility.

The reason for this complicated input is that the *adjunct variable* cannot be specified directly, but has to be included with the function `panel.superpose()` to modify the *panel function*.

Most of the interactive, menu-based programs allow an assignment of different markers in an easy way. Compare the remarks in section 2.

## 2 Interactive Statistical Graphics

### Introduction

A central point of interactive statistical graphics is the method of *Linking*. Interactive mouse selections inside one graphic are also performed in all other graphs. Thus the plots are linked. This method is often implemented inside so-called “scatterplot-brushing” (DATADESK, JMP, SAS-INSIGHT, S-PLUS, SYSTAT etc. ), but usually is limited to scatterplots.

All graphs shown in this paper were done with DATADESK (see (Velleman 1992)).

An overview on the development of interactive statistical graphics is given in (Unwin 1994), for a more precise description of linking see (Wills 1992).

## A Comparison

What is the connection between Trellis Displays and interactive statistical graphics? Let us look at the first row of the Trellis Display in figure 1. Here we find a comparison of the yield for the field *Waseca* for all species in the year 1931 and 1932.

Such a comparison could be done in `DATADESK` in a dotplot of *yield* by *species* and a barchart for the sites. If one selects the bar for *Waseca*, and the dotplot is set to hotset selection mode, i.e. only the currently selected data are plotted, one obtains figure 4. This figure is equivalent to the above mentioned part of the Trellis Display.

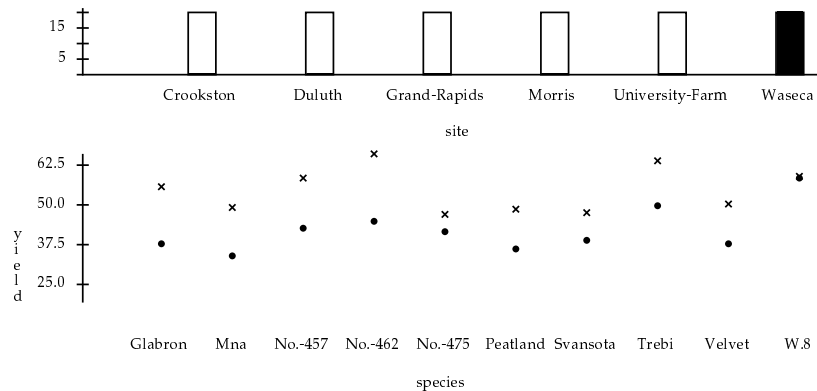


Figure 4: Yield on field *Waseca*:  $\times \hat{=}$  1931,  $\bullet \hat{=}$  1932

All the other rows of the Trellis Display can be obtained by just clicking the other fields in the barchart. This can be done as quickly as by looking at the plots inside the trellis sequentially.

The different plotmarkers (here: ' $\times$ ' and ' $\bullet$ ') can be assigned by choosing the desired symbol, while the target subset is highlighted.

The comparison of a single species from site to site is very hard, so it is sensible to summarize the data in a box-plot. See figure 5.

Because of the small amount of data (10 cases per plot) it would also be possible to use dotplots, but then, nearly identical data could not be distinguished any longer. To avoid overlaying of datapoints, it is sensible to switch to so-called *Textured Dotplots* (see (Tukey 1990)).

If one aims more at a comparison depending on fields, the part of *field* and *species* should be interchanged. This holds for Trellis Displays as well as for interactive graphics.

At this point we want to figure out, starting from the above example, how many "sensible" plots we need to analyse a dataset:

- Using Trellis Displays we obtain  $6 = 3!$  plots, if we fix yield as the

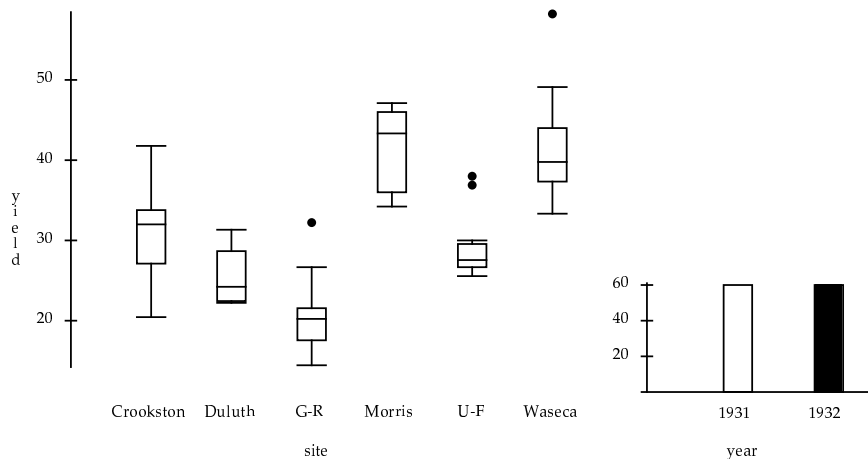


Figure 5: Box-plots for all fileds — 1932 selected.

dependent variable in the *panel function*. (i.e. all permutations of the variables: *species*, *field* and *year*). Half of those plots are redundant, because the plots are only interchanged in rows and columns. Figure 1 and 2 form such a “transposed pair”.

Therefore the number of Trellis Displays grows exponentially with the number of *conditioning* and *adjunct variables*. For  $n = 6$  we already need 310 displays, and that is far to many to examine.

- Interactive statistical graphics require 6 graphs. All other views can be derived from this set of linked graphs. We need three barcharts for the *conditioning* and *adjunct variables*, and three dot- or boxplots for the dependent variable yield by one of the three *conditioning* or *adjunct variables*. Here the number of graphs grows linearly with  $2 * n$  in contrast to  $n!/2$ .

Figure 6 shows these six plots. In these plots the names of the species and types are abbreviated from the program, because the space inside the plot is too small for the full name. This will appear in all the following plots.

An important limitation of Trellis Displays is the fact that, continuous *conditioning* or *adjunct variables* have to be made categorical. This is not necessary for interactive graphics, because in the latter case we can always choose suitable plots that fit the form of the individual variable. For these plots it only has to be guaranteed, that the necessary highlighting functionalities are available. (Although this is true for almost all the plots in DATADESK, box plots do not have full highlighting functionality.)

Another criticism of Trellis Displays is the fact, that the number of ob-

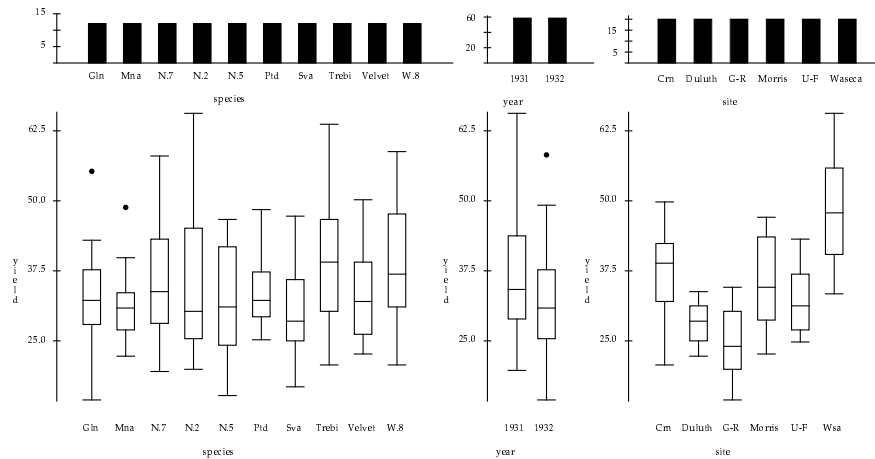


Figure 6: The six plots from which all Trellis Displays can be derived.

servations covered in one plot-panel is not obvious. Panels summarising the data cannot easily be compared from plot to plot, because the numbers in subgroups can vary so much. In the above example the number of datapoints is equal for all groups.

### Further Examination of the Example

In (Cleveland 1993) Cleveland points out a transcription error in the barley dataset. The data for the field *Morris* is exchanged between the years. This can be found easily in two ways:

- In the barchart for *year* one selects alternately the years 1931 and 1932. In the boxplot *yield by site* the inverse behavior of the field *Morris* can be seen at once.
- One selects sequentially all the bars in the barchart for *site*. The inverse behavior of the field *Morris* can be found in the barchart *yield by year*, too.

Compare figures 7 and 8. These static plots show only a weak impact on the visualizing effect. The switching between different views cannot be presented in a snapshot. Therefore another view is integrated in the plot. This is not a feature of `DATADESK`, but will be described later on.

One can derive some more results from the three linked barcharts and boxplots. We will see what enormous possibilities interactive statistical graphics offer.

For this we use the corrected dataset.

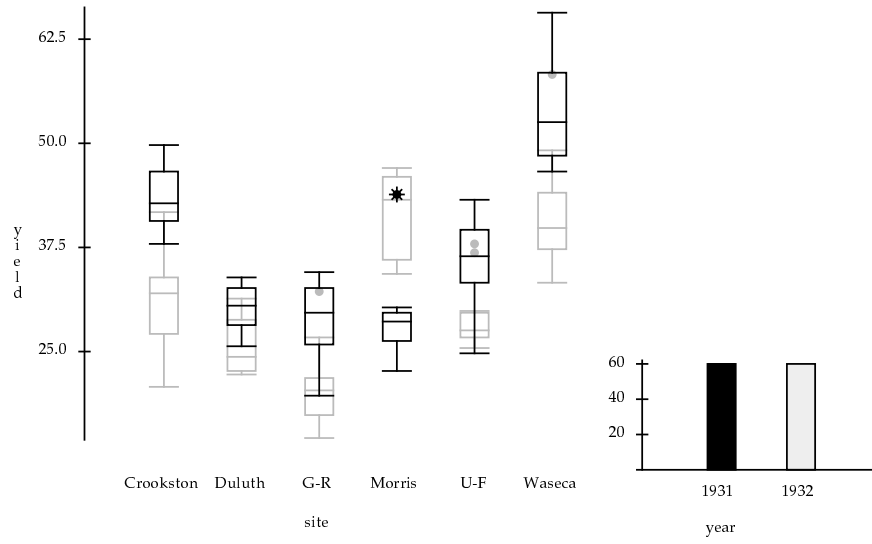


Figure 7: Identification of the transcription error via the barchart for *year*.

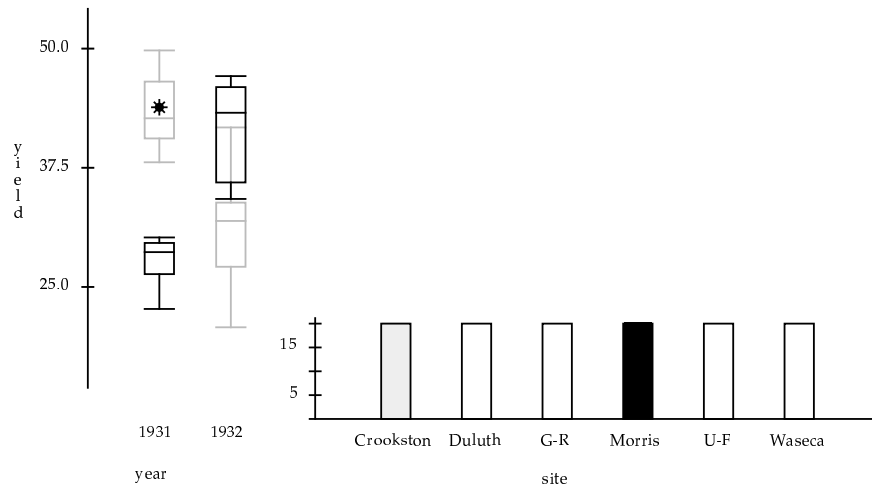


Figure 8: Identification of the transcription error via the barchart for *field*.

It can be found soon, that fields as well as species are homogeneous in the decrease of yield from 1931 to 1932. Only if one looks closer at the data, one sees that there are two more anomalies in the overall structure:

The species *Glabron* had an above-average yield on the field *University Farm* — independent of the year.

To recognize this, you have to order the species and the fields by yield. It is sensible to use both years for the ordering. Now you can select all barley species in succession. It is noticeable, that the species *Glabron* moves forward from rank four to rank two. This is shown in figure 9.

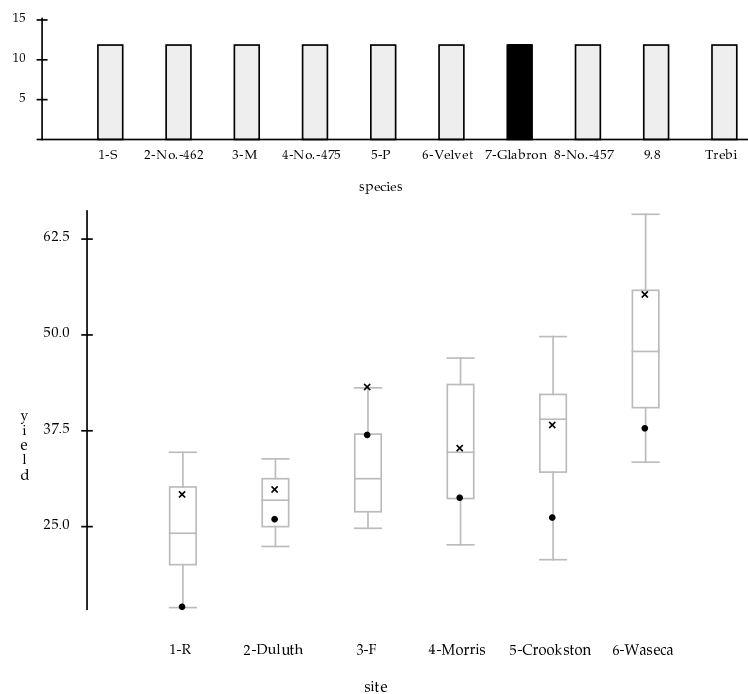


Figure 9: The yield of *Glabron* on *University Farm* falls out of line.

If you interchange the part of species and site, you discover that the yield on the field *Grand Rapids* was under-average for the species *No. 475* — however the overall yield was not good on this field at all.

You could criticise this approach on the grounds that the results were obtained by only two observations (1931, 1932). This reproach holds for Trellis Displays too. But the main difference was the process for achieving these results. It was highly interactive and explorative, so that the data could be analysed quickly and flexibly to find out interesting phenomena in the data.

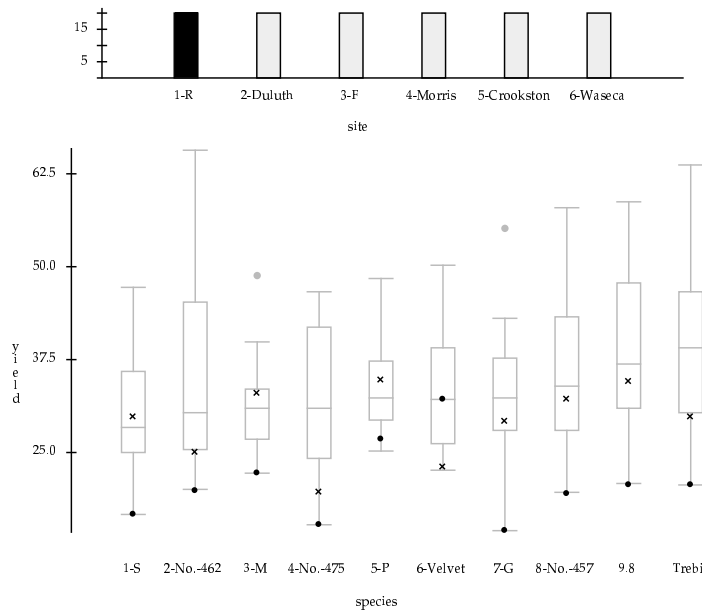


Figure 10: The yield on *Grand Rapids* also falls out of line for *No. 475*.

## User Interface

We will have a look at the user interface here too. The quality and the use of interactive statistical data analysis programs depends strongly on the quality of the user interface. This can be divided into the user interface and the functionality of the program itself and the user interface of the underlying operating system. `DATADESK` running on `APPLE` seems to be one out of only a few programs that unite both components in an attractive manner.

Setting up interactive graphics can be done very quickly using a few mouseclicks, although the user needs experience to do this in the most effective way. This experience cannot be formalised easily, but it can be practised by exploring known datasets with interactive methods.

Moreover methods for e.g. sorting groups of data by size are often not available in menu based programs. Usually this can only be done by complicated manual manipulations. The reason therefore is, that those programs are not very extendible. The sorting of the data in the above example inside `DATADESK` was done by adding the rank as a prefix to the name of each group. `DATADESK` sorts the bars inside a barchart by the name of the group, so this manipulation enables us to sort data.

The one and only program that unites interactivity and extendibility is the program `X-LISP-STAT` by Luke Tierney (see (Tierney 1990)). Unfortunately the standard set-up of `X-LISP-STAT` does not offer enough functionality to

use it as a data analysis package.

## Static Representation

In static presentation results of interactive statistical graphics lose a lot of their power. For this there is another selection added to figures 7 to 10. This secondary selection is plotted with a greyscale of 20%. It enables us to compare the different selection inside one plot, what we normally achieve by interactive operations on the data. This presentation technique can be extended to other kinds of interactive or dynamic graphics, but it will not be discussed here any further.

Figures 7 to 10 were made manually with the help of a graphics program and the easy to use copy- and paste functions of the MACINTOSH-environment.

## 3 Résumé

Trellis Displays offer a lot of useful extensions to standard static graphics. They enable us to visualize data in an uniform and comparable way. It is easy to arrange or sort the data in the plots to improve their comparability. Moreover, Trellis Displays offer the possibility of displaying a lot of static and systematic views of a dataset. This can be very helpful in the presentation of results obtained by interactive statistical graphics.

On the other hand, Trellis Displays also seem to have disadvantages. The assignment of the individual variables to the different elements of the Trellis Displays is hard to decide. Often it can only be done with the help of interactive statistical graphics. At this point the question arises, why the overall analysis should not be done only by interactive graphics, without the use of Trellis Displays.

In the end, interactive statistical graphics are a more flexible way of analysing datasets. Unfortunately there are only a few methodically good implementations of this technique, they should not be seen as a replacement for other statistical methods but as an addition to complement the tools for good data analysis.

The comparison shown in this paper only refers to the single dataset, which has been used most often to discuss Trellis Displays. A complete comparison would refer to more graphic types than only dot- and boxplots. Such an investigation will be part of a wider, further investigation.

## References

- Becker, Richard A., Chambers, John M., Wilks, Allan R. (1988), *The New S Language, A Programming Environment for Data Analysis and Graphics*, Wadsworth & Brooks/Cole, Pacific Grove CA.

- Becker, Richard A., Cleveland, William S., Shyu, Ming-Jen (1994a), *Trellis Display: A Framework for Visualizing 2D and 3D Data*, AT&T Bell Laboratories Statistics Research Report No. 8.
- Becker, Richard A., Cleveland, William S., Shyu, Ming-Jen (1994b), *Trellis Display: Questions and Answers*, AT&T Bell Laboratories Statistics Research Report No. 9.
- Becker, Richard A., Cleveland, William S., Shyu, Ming-Jen Kaluzny, Stephen P. (1994c), *Trellis Display: User's Guide*, AT&T Bell Laboratories Statistics Research Report No. 10.
- Chambers, John M., Hastie, Trevor J. eds. (1992), *Statistical Models in S*, Wadsworth & Brooks/Cole, Pacific Grove CA.
- Cleveland, William S. (1991), *The Elements of Graphing Data*, Wadsworth, Monterey CA.
- Cleveland, William S. (1993), *Visualizing Data*, Hobart Press, Summit NJ.
- MathSoft (1994) *S-Plus Trellis Displays User's Manual, Version 1.0*, MathSoft Inc., Seattle.
- Tierney Luke (1990) *Lisp-Stat: An Object-Oriented Environment for Statistical Computing and Dynamic Graphics* Wiley, New York.
- Tukey, J., Tukey, P. (1990) *Strips Displaying Empirical Distributions: I. Textured Dot Strips*, Bellcore Technical Memorandum.
- Wills, Graham J. (1992) *Spatial Data: Exploration and Modelling via Distance-Based and Interactive Graphics Methods*, Ph. D. Thesis, Trinity College Dublin, Dublin.
- Unwin, Antony R. (1994) *Interaktive Statistische Graphik — eine Übersicht?* to be published
- Velleman, Paul F. (1992) *Data Desk, Data Description* Ithaca, New York.