From **Brunelli, Lina & Cicchitelli, Giuseppe (editors). Proceedings of the First Scientific Meeting (of the IASE).** Università di Perugia (Italy), 1994. Pages 173-190.

COGNITIVE TECHNOLOGIES FOR STATISTICS EDUCATION: RELATING THE PERSPECTIVE OF TOOLS FOR LEARNING AND OF TOOLS FOR DOING STATISTICS

Rolf Biehler
Institut für Didaktik der Mathematik, Universität Bielefeld
Postfach 100131, D - 33501 Bielefeld, Germany

1. The software dilemma in statistics education

For introductory statistics education, several types of software are relevant and in use: custom designed educational programs for a specific educational goal, statistical systems for data analysis (in full professional version, in student version or as a specifically designed tool for students), statistical programming environments, spreadsheets and general purpose programming languages. On both a practical and a theoretical level, we can perceive a double dilemma, which is the worse the lower the educational level we have in mind. On the one hand, we have professional statistical systems that are very complex and call for high cognitive entry costs, although they flexibly assist experts. On the other hand, custom designed educational software is of necessity *constrained* to enable students to concentrate on essential aspects of a learning situation and to make likely certain intended cognitive processes. Nevertheless, as these *micro-worlds*, as we will call them here for short, are often not adaptable to teachers' needs, they are often criticized as being too constrained. Their support for flexible data analysis is limited, and to satisfy the variety of demands one would need a collection of them. However, coping with uncoordinated interfaces, notations and ideas in one course would overtax the average teacher and student. This practical dilemma is reflected on a theoretical level. It is not yet clear enough what kind of software is required and helpful for statistics education. We need a critical evaluation and analysis of the design and use of existing educational and professional programs. The identification of key elements of software that are likely to survive the next quantum leap of technological development and that are fundamental for introductory statistics is an important research topic. Results should guide new "home grown" developments of educational programs or, facing the difficulty of such developments, should influence the adaptation and elaboration of existing statistical systems toward systems that are also more adequate for purposes of introducing and learning statistics.

We might imagine ideal software support for introductory statistics education as follows:

- a student tool for data analysis and for modelling that can grow and expand into a professional version on several paths – instead of mere technically reduced student versions;
- a system of co-ordinated computer experiments, learning environments and major visualisations – that can be adapted to students' and teachers' needs;
- a larger host system that contains the student tool and the learning environments as integral parts. By that it also guarantees some notational and conceptual coherence and cognitive economy between learning environments and tools for doing statistics. The host system can be used as a tool-maker or medium builder for statistics education, at least in the hands of average experts.

These goals and research issues provide a framework that is to be filled with content. We will give some ideas and directions that are partly based on the results of two projects. In the "Software tools for statistics"–project, we explored the potential of commercially available software tools for statistics from an educational point of view (see Biehler and Rach, 1990a, Biehler and Rach, 1990b). In the ensuing MEDASS project, we have used the results to develop a detailed paper-and-pencil prototype of a software tool for teaching and learning data analysis. The detailed specification is described in Biehler and Rach (1992), the didactical background in Biehler (1992).

2. Cognitive technologies and the meaning of the mean

The notion of cognitive technology has induced us to discuss computer software from a broader perspective. Written language, symbol systems (mathematical and others), films, pictorial media, and computer languages can be interpreted as cognitive technologies. Pea (1987, p. 91) describes cognitive technologies as any "medium that helps transcend the limitation of the mind ... in thinking, learning, and problem solving activities", and he underlines that cognitive technologies "have had remarkable consequences on the varieties of intelligence, the functions of human thinking, and past intellectual achievements". Computer software can be regarded as a new kind of cognitive technology that also has assimilated and transformed some traditional technologies like graphs and formal notations. We may regard education as a system that does not just transmit knowledge but also "tools of the intellect" provided by the culture to enhance students' thinking. The question of what software is adequate for statistics education can be reformulated to ask which cognitive tools for statistics should be provided.

How can cognitive technologies contribute to understanding statistical concepts? In the following, we shall take the concept of mean as an example. The meaning of concepts can be characterised by the relations in an epistemological triangle in Fig. 1, which we adapt for our purposes from Steinbring (1991). For example, the concept of mean acquires its meaning from its relation to other concepts such as distribution, median etc., from representations and tools to work with it, such as formulas, algorithms, visualisations and from its applications to solving statistical problems, from its interpretations in contexts of application.

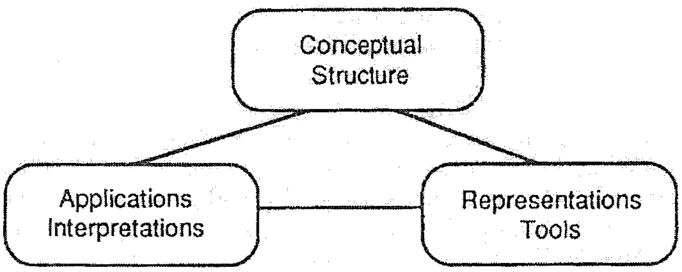

Figure 1. *Epistemological triangle*

Some empirical research, e.g. Pollatsek *et al.* (1981), confirms the impression that students often have a mere computational (procedural) understanding of the mean. In other words, the relational structure of Fig. 1 is an ideal seldom achieved. Important aspects of the mean are a geometric interpretation, i.e. its interpretation as a "center of gravity", and its varying significance as a summary statistics with regard to different types of data distributions. This is particularly important in comparison with other summary statistics for the center of a distribution. Furthermore, a functional interpretation is relevant, for instance, for understanding non-resistance to outliers: the mean is a function of the data and will change in a certain way when the data change.

New cognitive technologies obviously offer new representations providing didactical opportunities. In the case of summary statistics, we find the suggestion to use spreadsheets: When the data have been put into a range of cells, some further cells can be defined as the mean, the median or other functions of the data. Changing the data "by hand" is easy in a spreadsheet and will automatically lead to an update of the summary values: the "function machine" behind a spreadsheet is a new representation of the functional dependence. In particular, it supports an experimental study of the functional relationship by students. Whereas this example uses a general purpose tool, we also find more constrained and closed learning environments with new representations. The option *Stretchy Histogram* from the software *Statistics Workshop* provides opportunities for an interactive "geometrical experiment" in statistics education (see Fig. 2). The user is able to manipulate the shape of the histogram by directly pulling a bar in the graph. The mean, the median and the two quartiles are functionally dependent on the distribution with automatic updating. The relationship between the shape of a distribution and the four summaries can be explored empirically. The qualitative new feature of *direct manipulation* is used in this program. We could also transpose numerico-algebraic experiments from a spreadsheet environment into geometrical experiments. Imagine that you can move data points in a dot plot and observe how a graphical display of summary statistics, say a boxplot, will change.

One could also think of transforming other relevant visualisations of the summary statistics, as e.g. are discussed by Bentz and Borovcnik (1984), into interactive, dynamical geometrical computer experiments. For an empirical analysis of the pedagogical payoff of such environments it would also be helpful if a variety of visualisations were available. However, *Stretchy Histograms* is closed and too constrained and would not support these variations.

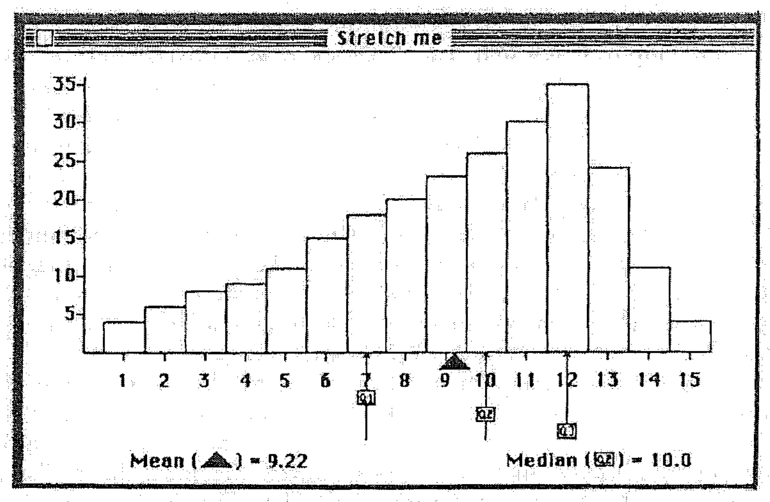

Figure 2. *Screen display from Statistics Workshop*

Due to new technological developments, the status of certain cognitive technologies may change. Algebra has always been a powerful cognitive technology for specifying how things have to be calculated and for depicting relations: when computers do the calculating, the latter becomes even more important. For instance, "experts" can "see", in the formal definition of the mean, that the mean is a linear function of each of the data values. Being able to interpret algebraic notation is still very relevant. Visual and experimental opportunities should be used to enhance this algebraic proficiency instead of aiming at its substitution.

Cognitive technologies affect the other elements of the epistemological triangle as well. From the perspective of teaching applied statistics, applications and interpretations of the mean are important constituents for understanding. Statistical software tools together with resources of statistical data make accessible qualitatively new kinds of applications and experiences while using the mean and other summary statistics in data analyses. For instance, students may re-discover the value of summarizing data to recognize lawlike relationships, whereas chaos and randomness prevail without summarization. This was a basic experience of the 19*th* century social statisticians when more data became available to them. The new technologies could turn this into an experience for students. Technology may support the sharing of rich data analytic experience that has been a basis for conceptual and methodological developments in history. For instance, different interpretations of the mean from substantialist interpretations as a "true value" to its more relational interpretations as a method for comparison or even as no more than a formal model parameter could be experienced in different applications.

The tension we have established between using statistical software tools and using educational programs is less extreme than it looks. One reason for developing *Stretchy Histograms* is certainly that currently available statistical systems will not support the above experiment. However, systems with properties that allow such more constrained environments to be defined could be also of use for the working statistician. For instance, being able to enrich the display of a data distribution by marking summary statistics would be also useful in data analysis, although many current systems do not support this as a general feature. Furthermore, the possibility of directly manipulating distribution shape could be a useful feature for a research statistician in times where the relative importance of algebraically defined curves has decreased in favour of *non-parametric* densities and functional relations. We shall come back to the possibility of using statistical systems as host systems for custom designed educational programs later.

3. Cognitive technologies as reorganizers

Cognitive technologies are frequently regarded as "cultural amplifiers" of the intellect. That means a quantitative intensification without a qualitative change of structure. However, Pea (1987) emphasizes the potential of cognitive technologies for leading towards a qualitative reorganization of thinking. There is great promise in the analysis of the educational possibilities of cognitive technologies from the reorganizational point of view. What may be called reorganization of thinking on an individual cognitive level may be called a scientific revolution on the level of science. Obviously, statistics provides good examples for this thesis. Let us take the development of Exploratory Data Analysis (EDA) as an outstanding example. Computer technology does not only support doing more analyses and graphs. The possibility of multiple analyses and interactive graphics has also given rise to qualitatively new types of data analyses and of theoretical research in statistics.

Examples of qualitatively new aspects of statistical thinking include:

- employing graphical displays as analytical tools;
- using concepts and methods for data analysis without a probabilistic justification;
- overcoming oversimplified model assumptions: diagnostic checking, data guided model choice, robust methods, effective handling of models with weaker (broader) assumptions;
- iterative, interactive ways of analysing data;
- exploration and insight as goals of data analysis, in addition to inference;
- resampling techniques as an alternative to classical inference methods.

Just as we can observe that certain statistical software systems have been an important (necessary but not sufficient) factor for making these ways of thinking available to a broader group of people, we have to think of tools for introducing statistics also in terms of their potential for supporting these new qualitative dimensions.

There is another important aspect in the development of statistics. The above changes of the objects of research have also affected the methods of research in statistics. It is said that statistics has become more like an experimental science: statisticians construct new methods and analyse them not only by mathematical (analytical) methods but also by sophisticated simulation studies and by trials on sets with real data. These research methods may also be more accessible to students than highly theory-laden analytical methods. Students might participate in a qualitatively new way of thinking (from their perspective), namely at the statistical workbench of constructing and evaluating new methods, instead of being only confronted with ready-made methods. For instance, students may suggest their own test statistics, which can be studied easily by simulation under different hypotheses. Or, students may design new graphs and test them on real data with real people.

How can we understand the cognitive role of concrete tools in more detail? There are three relevant features of software tools: "as user-directed agents that perform actions for the user"; as "built-in constraints or supports", as Kaput (1992, p. 525) expresses it, and as new kinds of "representations and objects to think with". The latter two are properties of a system as a whole, and not of individual agents and procedures.

A problem is the large variety of tools whose design has interacted with statistical practice. There is no agreement on standards for software tools with regard to certain domains of application and styles of thinking. Depending on their constraint-support structure, existing statistical systems are not equal in supporting the above mentioned qualitative ways of thinking. It is very likely that differences will show up between regular *SPSS*, *S-Plus* and *Data Desk* users. Furthermore, the same concepts can have specific meanings within the context of every particular software. For instance, a "table" is different in software such as *StatView*, *MINITAB* or in an ordinary spreadsheet, because there are different internal representations and different possibilities of manipulating a table; these also make electronic data tables different from tables on a sheet of paper. Another example are statistical graphs which are manipulable objects dynamically linked to the data in *Data Desk*, which can be defined and edited by scripts in *S-Plus*, and are mere bitmap or vector graphics in many educational programs. Although this shift of meaning in relation to the new objects and operations in a software tool, designated by Balacheff (1993) as the *computational transposition*, can be a source of misunderstanding, their potential for new types of activities has to be underlined, Dörfler (1993).

In the following paragraphs, we will critically examine some typical features of tools for learning and doing statistics in educational contexts.

4. Tools for data analysis as host systems for learning environments

The idea to co-use systems for statistical data analysis for building educational microworlds is not new. *MINITAB* has often been used in this sense by selecting a limited subset of commands for simulation and experimentation. The more coherent way of extending the language and the possibility of constructing new graphs makes *S-Plus* a better candidate for defining statistical microworlds, however at the price of much higher cognitive entry cost. A didactical proposal to use the programming language *APL*, extended by appropriate statistical functions for such purposes, is described in Naeve *et al.* (1991). Spreadsheets have also been considered as host systems. A recent reference is Arganbright (1992). As tools for flexible data analysis current spreadsheets have important weaknesses. Because their concepts and in-built functions are not well adapted to statistical concepts, data organisation and management may become awkward.

For secondary education, but also for introductory statistics in higher education, starting with a command language system is problematic. We are convinced that a host system with a graphical user interface and dynamically linked windows offers a more adequate basis. This would be even more the case if the limits of menu-based systems could be mitigated by allowing *formal language input* at those places where there is a real need, for instance, for defining data transformations or new procedures. An essential point is, how easy it is to define new micro-worlds, whether programming expertise is needed or whether we can do "programming by example". Two trends in software maturation meet these requirements: the general effort of providing easier ways of customization and the trend of statistical tools to support more open explorations.

The software *Data Desk* offers quite interesting possibilities for defining microworlds and we will discuss possibilities and limitations. *Data Desk* supports the definition of nested "derived variables" by formula, the linking of representations (data, graphs, equations of derived variables) and the possibility of saving environments with empty data or pseudo-variables (conserving all the defined relations).

In the screendump in Fig. 3, we have a set of car data where we have made a regression of *MPG* (miles per gallon) vs. *weight* of the car (window 4). As the graph has curvature, we would like to try experimentally a transformation of *MPG* from the family of transformations x^m. *Data Desk* offers *sliders* for an easy direct manipulation of parameters. We define a "slider" with name *m* (window 1), that is a parameter that can be directly maniputated by moving the vertical line on the screen. We define a "derived variable" *f(MPG)* = MPG^m (window 2). Window 3 shows a regression line of *f(MPG)* vs. *weight*, window 5 and 6 show two displays of residuals of that regression. If we change *m* in window 1, all the dependent windows will update. The dependence structure can be summarized as in the right part of Fig. 3.

This microworld can be used for studying the effect of different data transformations. One could also use artificial data or data from a certain known model to study these relations. In statistical practice, this microworld could be used for an interactive choice of a linearizing transformation – instead of using ready-made automatic methods for this purpose. As it is possible to save this environment and to replace the variables chosen by different ones, we have really defined a small new microworld for exploring the effects of data transformation. It would have also been possible to directly edit the formula in window 2. We note that formal-mathematical representations have re-entered at a reasonable location in a limited way.

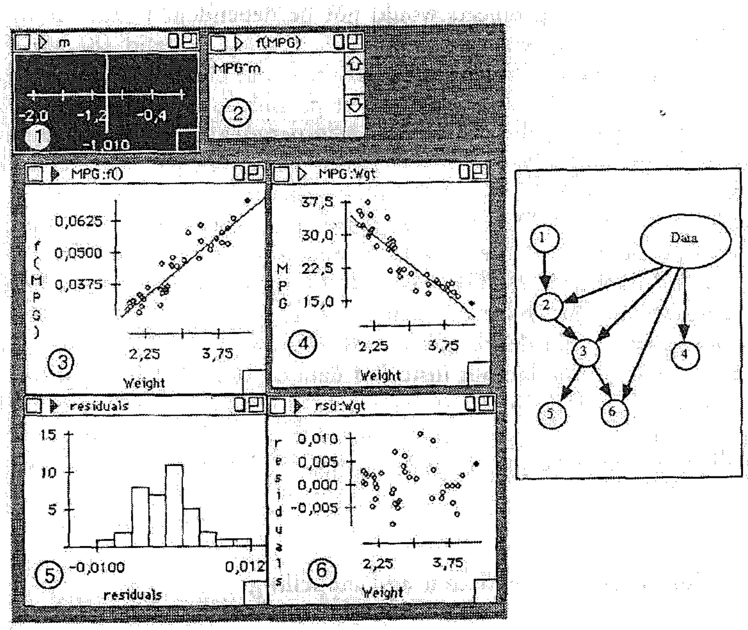

Figure 3. *An experimental environment in Data Desk 3.0; the linkage structure*

This example shows how new requirements from practice meet requirements from learning environments. In *Data Desk*, however, it would be impossible to define the standard visualisation of confidence intervals, namely simulating a sequence of samples and displaying the (random) sequence of pertinent confidence intervalls in one display.

We would need to define an experiment in the following steps:

(1) choose a probability *model* with expected value μ;

(2) draw m samples of size n;

(3) calculate the m means;

(4) chose the level $(1-\alpha)$;

(5) calculate the interval lengths according to a method *K*;
(6) plot the *m* means vs. 1, ... , *m*;
(7) add symmetric bars to the scatterplot, corresponding to the confidence intervals;
(8) add a dotted line at $y = \mu$.

These steps should be executed by using commands from the menu system. They should result in new objects (data or plots) that are dependent of the previous objects and that can be represented in separate windows. In Data Desk, steps 1-6 would be possible for some models; however, the resulting objects would not be dependent in the required sense. The necessary enrichments of the display (7) and (8) are not possible. Now, the user might like to control and change the parameters *model*, μ, *m*, *n*, (1-α), *K*. Also, it should be possible to use another random seed, that is repeating the whole random experiment in a simple way, for instance by pressing a "recalculation button". The final plot with the series of confidence intervals must dynamically *depend* on these parameters. This is not the case in Data Desk. An adequate host system should provide better dependence structures. We have made some suggestions for that in Biehler and Rach (1992). The repeated sampling principle which should be visualized by the above experiment could be similarly demonstrated by applying the confidence method to a series of real data sets, as in Biehler (1990) to the famous historical data on boys and girls christened in London from 1629 to 1710. This may contribute to meaning in the sense of applications and interpretations in Fig. 1. The similarity of both visualisations supports also the need for integrating data analysis tools and learning environments.

5. Student tools for simulation and modelling

The software dilemma is especially grave with regard to elementary tools for simulation and modelling, as was described in more detail in Biehler (1991). Apart from statistical programming environments, statistical systems for data analysis are weak in this respect. *Models* are not separate objects of the systems. It is difficult to generate the requested random data, especially with simple multistage experiments, and data analysis is not well adapted to thinking in terms of *events* and *random variables* whose frequency or distribution has to be recorded during simulation. Educational programs with ready-made simulations are valuable but will probably not contribute to involving students in the process of modelling. If we are interested in this, it will be essential that the students themselves define the model as well as transform the verbally described events and random variables into a more abstract notation.

The educational software *Probability Simulator* is an important step towards closing this gap by means of a menu-driven multi-window environment with many relevant applications in elementary probability education (see Konold, 1991). The students can:

(1) choose and change a box model;
(2) fix sample size and number of repetitions;
(3) run and re-run the simulation;
(4) define (several) events that are to be counted.

The results of counting the events are displayed in a separate window which is dynamically linked to the data window containing the raw data. The language for defining events is a delicate problem, *ProbSim* uses wildcards (*) and *variables* (v_1, ...) for that purpose and makes a distinction between *ordered* and *unordered* events. For instance, the event "at least one pair of equal outcomes in an experiment of sample size 6" is defined by the *unordered* type of event (v_1,v_1,*,*,*,*), the *ordered* type would mean that at least one pair occurs in the first two places. As compared to formal programming languages, students can define models in terms of urns and define events in a simpler albeit abstract notational system. The constraint-support structure of *ProbSim* corresponds to a certain idea of what should be essential steps in model building for secondary probability education. Every student tool should have at least similar properties.

However, it is not (yet) possible to define random variables and record their distribution. Also, there is an abundance of simple multi-stage experiments in elementary probability with dependent trials that cannot be simulated because *ProbSim*, as nearly all similar programs, is confined to independent repetitions. After some experience, one would also appreciate the availability of a more general language for defining events by the user. These are some of the possibilities for expanding a student tool towards wider educational applications and professional requirements. Disparate development projects and constrained resources for educational software development have not yet led to overall satisfying solutions.

6. Student tools for data analysis

An obvious criterion for tool design is making "simple things simple". The interpretation of this depends on those intended applications of the tool that should be simple to work on. There are some recent student tools for data analysis that have such a profile dependent on a certain curriculum or a didactical intention, namely *DataScope*, *Statistics Workshop* and *TableTop*.

In Biehler and Steinbring (1991), we have described experience with the GRAPHDAS pilot project in introducing ideas of data analysis in the classroom. We used *StatView* and *Data Desk* in our exploratory work with teachers and students, and found the moderate openness of these tools helpful in two respects. Firstly, we needed this openness as we did not want to make strong assumptions beforehand concerning what kind and content of data analysis would be adequate for the secondary level. Using teaching experiments for exploring this problem could not have worked with less open, curriculum dependent, tools. Secondly, we assumed that some small project activity with open problems referring to complex data should be part of a course. In other words, we included problems that could lead to new methodology, at least from the students' perspective – sometimes from that of teachers and from our own. For instance, although teachers introduced data on traffic accidents as a topic where boxplots should be used to analyse seasonal variation, students asked for predictions of future developments and the teacher found himself in a situation of time series forecasting. If data analysis is to be stimulated and controlled by students' subject matter interests and knowledge, a tool's closedness must not crush such involvement. From this perspective, the above curriculum dependent tools were not adequate for our purpose. The professional tools, however, would have been much more useful, especially as tools for student work, if they had provided greater adaptability. For instance, an easy redesignability of the menu system would have been very valuable. Nevertheless, there was not one tool that fulfilled all the elementary functions we found useful in the project work. This claim can be illustrated with regard to statistical graphs. No tool offered the direct interaction with graphs of *Data Desk*, the composite graphs of *StatView*, the variety of elementary graphs of *Systat*, and the possibility of enhancing graphs by further statistical information of *S-Plus* – not to speak of *S-Plus'* capability to define new graphs.

A design or evaluation of software cannot be performed neutrally, but only with regard to a class of intended applications for which the software should provide easy solutions. An explicit discussion of systems of intended key applications may contribute to a more objective judgment. We shall discuss some intended applications in the following, where we found that existing software was not very strong, but which are relevant from our view of elementary data analysis. A more comprehensive account can be found in Biehler and Rach (1992) and Biehler (1992).

Categorical and nominal variables. Let us take as an example a data set with *height* and *weight* measurements for a group of male and female students coming from five age groups. We represent the *weight* for each student in a bar graph labeled by the students' names. Usually, an alphabetical order of the names or the underlying index numbers is taken as the basis for the plot. Re-sorting the names in the plot is an elementary method of analysis. Now, we would like an easy access to the following options: switch from barchart to dotchart in the sense of Cleveland (1985); project the points of the dotchart to an axis to explicitly show the marginal distribution as a dotplot; re-sorting the names according to the variable *weight*, re-sorting according to the variable *height* as a means to approach association of these two variables, regrouping the names according to a categorical variable like sex and then sorting within the categories to produce a grouped dotchart or bar chart. We can conceive of these operations as manipulating a categorical or nominal axis and not as transforming the data into auxiliary data first and then replotting them by re-calling a graphical procedure. We might generalize this idea to plotting two categorical variables against each other. This results in a "scatterplot with categorical or nominal axis". The latter idea is realized in the software *TableTop* (see Hancock and Kaput, 1990). If we now had the opportunity of re-sorting and permuting the order of the 2 categories on each axis "by hand", we might discover new relationships and possible classifications. In fact this would be a way of implementing the graphical method of Bertin (1977).

Raw and aggregated data. The computer has greatly extended the possibilities of data analysis with raw data. However, there are still many important societal (very large) data sets which are only available in aggregated form. It should be easy to work with the same statistical methods on either data type (with slightly different interpretations of the commands). For instance, although it is often possible to vary the length of a class interval, when calling a histogram to display a distribution for ungrouped data, this possibility should be also available for grouped data. In the GRAPHDAS project where we worked with children and teachers of grade 8, we had data on the number of traffic accidents per age group of length 1. For a comparison of the fairly irregular curves of males and females, a flexible regrouping to smooth the two histograms would be helpful.

Analysis by groups should be a simple possibility. For some statistical methods, this is realized in educational programs like *Statistics Workshop* and *DataScope. DataScope* makes it also easy to select two grouping variables to base analyses on a two-way classification. *Data Desk* shows a fairly general realisation, where we can have graphical displays and numerical summaries by group, although no easy two-way classification. Also, for a small number of category levels one would like to have not only many windows for each group but integrated graphs. For instance, selecting a stem-and-leaf display by group (of two categories) should result in a back-to-back stem-and-leaf.

Composition of graphs. The juxtaposition of graphs should be an easy possibility. One might wish for some "intelligence" in juxtaposing graphs, for instance with regard to identifying or adapting axes of the same variable. The superposition of graphs or the composition and enrichment of statistical graphs should become as easy as in modern drawing programs, but with the essential difference that the graphical elements can be mathematically or statistically defined and processed.

Graphs as an interface to data: geometrization of data analysis. Velleman (1989) employs the metaphor of using graphs to talk to the data. We may think of options for a geometrical classification into groups "by hand", selecting subsets and displaying information of a data case represented by a point in a graph (identification). This has been realised in some pieces of recent software. However, having elementary educational applications in mind, we may also think of possibilities of graphical input of data (to generate a data set with specific geometric properties, realized in *Stat-Lab*), and of graphical change of data (graphical dragging of points to study how analyses depend on a data point, similar to the dragging mode in the geometry software *Cabri Géomètre*). Movable lines that can be drawn by hand into a data set, should not only be didactical means to prepare the method of least squares as in *Statistics Workshop*, but can be an implementation of the method of "eye-fitting" lines to data. However, it should be as easy as with least squares lines to get residuals from such lines. It should be equally easy to draw and manipulate a stepwise linear curve "by hand" into a set of points, and then work with this line as with other statistical summaries. A slicing of a scatterplot into a number of stripes (graphical discretization), then calculating summary points to get a rough smooth is an elementary smoothing method that has the advantage of being transparent to the user with regard to its construction – as compared to more advanced methods.

7. Conclusion: Requirements from an ideal softwaretool for supporting learning and doing statistics

What features of a software for supporting learning and doing statistics are important? Let us summarize the discussion in some key points:

(1) The software should possess a *graphical user interface* with multiple, dynamically linked interactive representations (graphics, tables, formulas, analysis results).

(2) The limits of menu-based systems should be mitigated by allowing *formal language input* at those places where there is a real need, for instance for defining data transformations or "derived variables".

(3) The software should have the character of a *meta-tool* or *meta-medium*, which allows the definition of more constrained exploratory learning environments or more constrained tools that are better suited for certain users and certain domains of application.

(4) The software should not only provide ready-made methods, but support the *definition or customisation of new and personal, non-standard methods*, for instance graphs, tables, statistical tests. This is optimally realized by "*programming by example*" to avoid programming in formal languages.

(5) The software should support an easy access and a flexible use of *elementary* graphical and numerical methods of *exploratory and inferential data analysis. Bootstrapping* should be "at hand" for analysing the variability of summaries.

(6) The software should support the *exploration of statistical methods* on real, simulated and artificially constructed data sets. Exploring the changes in results when changing data or method parameters should be easy.

(7) The software should support the *definition of computer experiments* to explore the consequences of parameter changes as well as of chance variations.

(8) The *construction of visualisations* should be supported by something like a graph construction set.

(9) The *definition of statistical and probabilistic models* should be supported. It should be possible to define events and random variables of interest, and to analyze the simulated data like real data.

Of necessity, these points define a framework that has to be elaborated further. In any case, we should be motivated as statistics educators to develop our own perspective and needs, hoping that we can get intellectual and material support to improve the state of the art.

SOFTWARE

Cabri Géomètre. Laborde J. M., *et al.*, LSDD, B.P. 53 X, F-38041 Grenoble-Cedex (Macintosh, MS-DOS).

Data Desk 3.0 (1989), Velleman P., Data Description Inc., P.O. Box 4555, Ithaca, NY 14852 (Macintosh).

DataScope (1992) version 1.16. Konold C. and Miller C.D., Scientific Reasoning Research Institute, Hasbrouck Lab, University of Massachussetts, Amherst, MA 01003 (Macintosh).

MINITAB. MINITAB Inc., 3081 Enterprise Drive, State College, PA, 16801 USA (MS-DOS, Macintosh and others).

ProbSim - A Probability Simulator. (1992) version 1.3.2, Konold C. and Miller C.D., Scientific Reasoning Research Institute, Hasbrouck Lab, University of Massachussetts, Amherst, MA 01003 (Macintosh).

S. and *S-Plus* (1988), Becker R.A., Chambers J.M. and Wilks, A. R., see: Becker, Richard A., Chambers, John M., Wilks, Allan R., The New S Language, Wadsworth & Brooks, Pacific Grove. (Unix, MS-DOS/386 and others).

SPSS. SPSS Inc., 444 North Michigan Avenue, Chicago, IL 60611 (Macintosh, MS-DOS and various other systems).

Statistics Workshop (1991), Rubin A. and Bruce B. (BBN Inc., Cambridge, MA, USA). Wings for Learning/Sunburst, 1600 Green Hills Road P.O. Box 660002, Scotts Valley, CA 95067-0002 (Macintosh).

StatLab (1987). W. Douglas Stirling. The New Zealand Statistical Association, P.O. Box 1731, Wellington, New Zealand (Macintosh)

StatView SE+Graphics (1988), Feldman D. S. jr and Gagnon J., Abacus Concepts, Inc., 1984 Bonita Ave., Berkeley, CA 94704, USA. (Macintosh).

StatView 4.0 (1992), Feldman D.S. and Gagnon J., Abacus Concepts, Inc., 1984 Bonita Ave., Berkeley, CA 94704, USA (Macintosh).

Systat 5.0 (1989), SYSTAT Inc., 1800 Sherman Avenue, Evanston, IL 60201-3793 (Macintosh, MS-DOS).

TableTop, software in development, Hancock C. and Kaput J.J., TERC, 2067 Massachusetts Avenue, Cambridge MA 02140.

Bibliography

Arganbright D. E. (1992), Using spreadsheets in teaching statistics, in F. Gordon and S. Gordon (eds.), *Statistics for the Twenty-First Century. MAA Notes n. 26*, pp. 226-242, Washington, DC, Mathematical Association of America.

Balacheff N. (1993), Artificial intelligence and real teaching, in C. Keitel and K. Ruthven (eds.), *Learning from Computers: Mathematics Education and Technology*, Springer, Berlin, pp. 159-186.

Bentz H.J. and Borovcnik M. (1984), Mittelwert, Median und Streuung: Eine Zusammenschau, in H. Kautschitsch and W. Metzler (eds.), *Anschauung als Anregung zum mathematischen Tun: 3. Workshop zur 'Visualisierung in der Mathematik' vom 11. - 16. Juli 1983 in Klagenfurt*, pp. 208-220, Hölder-Pichler-Tempsky, Wien.

Bertin J. (1977), *La Graphique et le Traitement Graphique de l'Information. [Engl. Transl.: Graphics and Graphic Information-Processing, Berlin: de Gruyter 1981]*, Flammarion, Paris.

Biehler R. (1990), Daten analysieren mit dem Computer: Unterstützung von Begriffsbildung und Anwendungsorientierung in der Stochastik, *Der Mathematikunterricht*, 36(6), pp. 50-71.

Biehler R. (1991), Computers in probability education, in R. Kapadia and M. Borovcnik (eds.), *Chance Encounters - Probability in Education*, pp. 169-211, Kluwer, Dordrecht.

Biehler R. (1992), *Intendierte Anwendungen und didaktische Begründungen zu einem Softwarewerkzeug zur Explorativen Datenanalyse und stochastischen Simulation für Schule und Ausbildung*, Inst. f. Film und Bild in Wissenschaft und Unterricht (FWU), München.

Biehler R. and Rach W. (1990a), Softwaretools for statistical data analysis in education and teacher training? An analysis of software conceptions from an educational perspective, in F. Faulbaum *et al.* (eds.), *SOFTSTAT '89: Fortschritte der Statistiksoftware 2*, pp. 575-583, G. Fischer, Stuttgart.

Biehler R. and Rach W. (1990b), *Softwaretools zur Statistik und Datenanalyse: Beispiele, Anwendungen und Konzepte aus didaktischer Sicht*, Soester Verlagskontor, Soest.

Biehler R. and Rach W. (1992), *MEDASS: Explorative Datenanalyse und Stochastische Simulation - Anforderungsbeschreibung zu einem Softwarewerkzeug für Schule und Ausbildung*, Institut für Didaktik der Mathematik, Universität Bielefeld, Bielefeld,.

Biehler R. and Steinbring, H. (1991), Entdeckende Statistik, Stengel-und-Blätter, Boxplots: Konzepte, Begründungen und Erfahrungen eines Unterrichtsversuches, *Der Mathematikunterricht*, 37(6), pp. 5-32.

Cleveland W.S. (1985), *The Elements of Graphing Data*, Wadsworth, Monterey.

Dörfler W. (1993), Computer use and views of the mind, in C. Keitel and K. Ruthven (eds.), *Learning from Computers: Mathematics Education and Technology*, Springer, Berlin (to appear).

Hancock C. and Kaput J.J. (1990), Computerized tools and the process of data modeling, in G. Booker, P. Cobb and T.N. d. Mendicutti (eds.), *Proceedings of the 14th International Conference on the Psychology of Mathematics Education*, 3, pp. 165-172, The Program Committee of the 14*th* PME Conference, Mexico City.

Kaput J.J. (1992), Technology and mathematics education, in D. Grouws (ed.), *Handbook of Research on Mathematics Teaching and Learning*, pp. 515-555, Macmillan, New York.

Konold C. (1991), ChancePlus: A Computer Based Currriculum for Probability and Statistics. *Second Year Report*, Scientific Reasoning Research Institute, University of Massachusetts, Amherst.

Naeve P., Trenkler D. and Wolf H.P. (1991), How to make the teaching of statistics roar - Some thoughts on computer based experiments. *Computational Statistics Quarterly*, 6(4), pp. 325-353.

Pea R.D. (1987), Cognitive technologies for mathematics education, in A.H. Schoenfeld (ed.), *Cognitive Science and Mathematics Education*, pp. 89-122, L. Erlbaum, Hillsdale.

Pollatsek A., Lima S. and Well A.D. (1981), Concept or computation: students' understanding of the mean, *Educational Studies in Mathematics*, 12, pp. 191-204.

Steinbring H. (1991), The theoretical nature of probability in the classroom, in R. Kapadia and M. Borovcnik (eds.), *Chance Encounters - Probability in Education*, pp. 135-167, Kluwer, Dordrecht.

Velleman P.F. (1989), *Handbook of Data Desk 3.0*, IL, Odesta Corporation, Northbrook.