Prolegomena to the Theory and Practice of Data Analysis
or:
What we should have learned from the past 50 years.

Huber, Peter J.
Pardennweg 13
7250 Klosters, Switzerland
E-mail: peterj.huber@bluewin.ch

Introduction

The past 50 years have seen fundamental changes in the field of statistics. The most obvious development has been the advent of computing and with it the impact of ever larger data sets. It seems that most people initially thought (and many still think) that this was just more of the same, which it is not. There are deeper issues.

At the beginning of the same 50 years John Tukey, in his revolutionary paper on the Future of Data Analysis (1962), prepared the conceptual ground for the subsequent developments. His paper went against the grain of most mathematical statisticians and therefore initially was largely ignored. He emphasized that data analysis, and the parts of statistics which adhere to it, must take on the characteristics of a science rather than those of mathematics. He shifted the primacy of statistical thought from mathematical rigor and optimality proofs to judgment. He emphasized: "Large parts of data analysis are inferential in the sample-to-population sense, but these are only parts, not the whole." As of today, too many statisticians still seem to cling to the traditional view that statistics *is* inference from samples to populations (or: virtual populations). Such a view may serve to separate mathematical statistics from probability theory, but is much too exclusive otherwise. For mathematical statisticians Tukey's new emphasis was hard to digest – it looked like a regress to previous naive ways of doing descriptive statistics – and possibly it was listened to only because of Tukey's stature and his background in pure mathematics. Incidentally, in 1962 Tukey still had underestimated the future impact of the computer.

In short, Tukey was initiating a change away from the then prevalent Fisherian paradigm, which had been concerned with small, homogeneous data sets, with precisely specified distributional models, and with devising procedures optimized for them. The time was ripe to concern ourselves with large, heterogeneous sets. Larger data sets tend to be structurally different from smaller ones. They are not just more of the same, they are larger because they have to be larger. In particular, they are, as a rule, much more heterogeneous.

We might say: if Fisher was concerned with the tactics of statistics, we now had to concern ourselves with its strategy. We learned (or should have learned) that a practicing data analyst had to come to grips with the following interconnected areas, all belonging to non-mathematical statistics: (1) strategic thinking, (2) dealing with massive, inhomogeneous data sets, (3) providing proper support to statistical computing, and (4) the handling of complex, approximate models.

But how do you teach data analysis to students, and how do you make sure that they pay attention to such issues? After some negative experiences with the course work approach, I have come to the conviction that data analysis has to be learned through apprenticeship and anecdotes rather than through systematic exposition. This may very well be the best way to teach it; George Box (1990) once pointedly remarked that you do not learn to swim from books and lectures on the theory of buoyancy. The higher aspects of the art you have to learn on the job.

For a more detailed discussion of these issues see Huber (2011).

Strategy

With large data sets, data analysis begins to acquire the characteristics of a Big Science project. Things have to be planned well in advance, and one has to keep in mind that unforeseen events can throw you off track. Here is a check list of the most important stages. All of them are strategically important; the precise way how their purpose is achieved is irrelevant and a matter of tactics. These stages are listed roughly in the order in which they are encountered in typical analyses, and I shall briefly comment on their roles. Though, strictly speaking, ordering the pieces is impossible, one naturally and repeatedly cycles between different actions.

- Planning and conducting the data collection
- Inspection
- Error checking
- Modification
- Comparison
- Modeling and model fitting
- Simulation
- What-if analyses
- Interpretation
- Presentation of conclusions

Careful **planning** will facilitate the subsequent analysis. One should keep in mind that there are hushed-up cases where million dollar data collections had to be junked because in the planning stage one had forgotten to randomize the experiment. If one is planning to collect massive data, one should never forget to reserve a certain percentage of the total budget for data analysis and for presentation of conclusions.

The principal purpose of **inspection**, whether graphical or non-graphical, is to see things one was not looking for. The problem with inspection is that only a very limited amount of information can be meaningfully presented to, and digested by, a human being at any one time. With larger data sets one must rely on subset selection (random or targeted) and summarization (such as averages and density estimates); with huge, highly structured sets, all approaches are guaranteed to leave large and unexplored holes in the data landscape. It is not easy, but necessary, to identify and to occupy the strategically important positions.

Errors can have many causes and can surface at any stage of the analysis. Some careful detective work may be needed. Automated error checking programs typically are based on legality checks and the like. By throwing out erroneous data they sometimes may hide systematic blunders affecting large sub-batches of the data. Some of the ugliest kinds of errors we have met were programming errors in the data recording software – such things may be beyond the event horizon of a statistician not involved with the actual process of data collection.

Data **modification** ranges from simple transformations, like taking logarithms in order to equalize variances in plots of spectrum estimates, to complex aggregation or grouping operations. On the strategy level, modification serves primarily for the preparation of a streamlined base set, obviating the need to access a huge raw data set, and facilitating the subsequent analysis.

Already in 1821 Playfair had stressed the importance of **comparison**: "It is surprising how little use is derived from a knowledge of facts, when no comparison is drawn between them." Playfair is considered the father of statistical graphics, and it certainly is no accident that the main tools of comparison are graphical. Comparison between results from two or more unrelated sources of information may provide a more convincing confirmation of the correctness of an interpretation than any statistical test. Remember that overly complex designs can make comparisons difficult.

Modeling is a scientific, not a statistical task. This creates difficulties on the interface both on a human level, between the scientist and the data analyst, and on the software level, between programs of different origins (whose bug is it?). Modeling notoriously involves a lot of non-trivial *ad hoc* programming, precisely because of interface problems. A data analysis package is useless if it does not facilitate the programming and interfacing of arbitrary models. Note that one may have to access external modeling programs from the inside of simulation loops and the like. Insight is gained by thinking in models, but reliance on models can prevent insight.

With more complex data and more complex analysis procedures, **simulation** gets ever more important. Resampling methods (bootstrap) have been overrated. Resampling works best with unstructured, homogeneous data (that is: with data that can be viewed as a sample from a larger population). It still works with well-designed stratified samples. It fails with highly structured data (which are becoming the rule rather than the exception), and for example also with most time series data. Stochastic simulation still works: prepare synthetic data sets similar to the real one and check how the analysis performs on them. For this, decent computing support, that is, the availability of a suitable programmable command language, is essential, otherwise the programming effort will create a bottleneck. Admittedly, the results depend on the somewhat arbitrary choice of the model behind the simulation, but a comparison between simulated synthetic data sets and the actual set (usually such comparisons are not trivial) will give an idea of the phenomenological quality of the model, and one can get at least some crude estimates of the variability of estimates.

In his 1962 paper Tukey had emphasized the increasing importance of empirical sampling (p. 61), that is of simulation. Curiously, he had shunned modeling. But it seems to me that by now the combination of modeling, model fitting and simulation has acquired a central position on the stage of statistics and data analysis. It has largely supplanted the role of classical mathematical statistics. Specifically, its three components are taking over and expanding positions that traditionally had been occupied by Fisher's (1922) three problems: Problems of Specification, Problems of Estimation, Problems of Distribution. The reason of course is that with heterogenous data and complex, composite models the analytical approaches of classical mathematical statistics are no longer able to give quantitative answers.

The more complex the data structure and the possible explanatory models, the more important it is to conduct alternative **what-if analyses**. In this category, we have anything from relatively simple sensitivity analyses to involved checks of alternative theories. What happens if we omit a certain subset of the data from the analysis? What if we pool some subsets? What if we use a simpler, or a more complicated, or just a different model?

Interpretation, just like modeling, belongs into the domain of the scientist. Statisticians tend to think of interpretation in terms of "inference", that is, a Bayesian will assign probabilities to statements, a frequentist will think in terms of tests and P-values, scientists expect from the statisticians assistance with the quantification of the conclusions. This is too narrow a framework, covering only that subset of interpretation that can be numerically quantified by probability values (P-values, significance levels, confidence intervals, and so on).

The larger the data sets are, the more difficult it is to **present the conclusions**. The presentation must be adapted to the language and customs of the customers, and one may have to educate them – and in particular the journal editors! – that sometimes a P-value is worse than useless. With

massive data sets, the sets of conclusions become massive too, and it is simply no longer possible to answer all potentially relevant questions. We found that some kind of decision support system (DSS), that is: a customized software system to generate answers to questions of the customers, almost always is a better solution than a thick volume of precomputed tables and graphs. It is straightforward to design a system duplicating the functions of such a volume, and it is easy to go a little beyond, for example by providing hypertext features or facilities for zooming in on graphs. But the appetite grows with the eating, trickier problems will arise, and the DSS then begins to develop into a full-fledged, sophisticated, customized data analysis system adapted to the particular data set(s).

Massive data

Massive data create technical problems through sheer size. But curiously, some of the trickiest problems are on the human interface side, that is, they have to do with mutual misunderstandings.

Data Mining sometimes is touted as a cure-all for the problems caused by data glut. By now, some of the original hype has abated. Most of the so-called data mining tools are nothing more than plain and simple, good old-fashioned methods of statistics, with a fancier terminology and in a glossier wrapping. What made those methods work in the first place, namely the common sense of a good old-fashioned statistician applying them, did not fit into supposedly fully automated, "all artificial, no natural ingredients" packages. Unfortunately, the resulting pattern – a traditional product in a different package, combined with hard sell and exaggerated claims – matches that of snake oil.

There are a few success stories of data mining. But as far as I can judge, all of them are based on *ad hoc* approaches and *ad hoc* programming, not on pre-packaged data mining tools. After perusing some of the literature on data mining, I have begun to wonder: too much emphasis is put on futile attempts to automate non-routine tasks, and not enough effort is spent on facilitating routine work.

Massive data necessitate a conscious effort at data base management (DBM). Unfortunately, DBM people as a rule do not understand that data analysts/statisticians have requirements rather different from those they themselves are used to. We have learned that their goals are at cross-purposes in a literal sense. If a data set is organized as a matrix, with cases as the rows and variables as the columns, then for data base people the extraction or addition of a few cases must be handled efficiently, while for the data analysts it is more important to handle the extraction or addition of a few variables efficiently. With more complex data organizations (e.g. hierarchical ones) the problems get even harder.

We have had to improvise our own approaches to DBM for the simple reason that typically the data base software coming with the DBM systems in which massive data were submitted for analysis turned out to be intolerably slow for data analytic applications.

Computing support

We learned that we need an open ended software system that facilitates improvisation – from reformating data that arrive in awkward binary formats, to running simulations with arbitrarily modifiable models, and to creating a customized decision support system for the benefit of the end users. Such a system best is based on the middle ground, on a command language, but must offer facilities for branching out towards either side: batch programming and menu interfaces. It is important that as a student one becomes fluent in a suitable programmable command language (sometimes, not entirely accurately, called a script language), supporting these goals.

Remember that a statistics program cannot be used in a simulation loop (and therefore must

be rewritten) if it insists on interactive input or if it produces output that cannot easily be processed further by other statistics programs.

Approximate models

Over the past 50 years we have been confronted with ever more complex complex models in the life sciences and elsewhere, and on assessing their goodness-of-fit with the help of simulation. Typically, these models are not supposed to render every nook and cranny of the real-life situation, but merely its essential aspects. While many applied statisticians still seem to live exclusively within the classical framework of tests and confidence levels, it became increasingly obvious that one had to go beyond mere tests of goodness-of-fit. One must take seriously the admonition by McCullagh and Nelder (1983, p.6) "that *all models are wrong*; some, though, are better than others and we can search for the better ones." With large data sets, perfectly good approximate models may be rejected for irrelevant reasons, that is for reasons of no concern for the scientific issues of interest. Apart from searching for better models, we must also learn when to stop the search, that is: we must address questions of *model adequacy*.

Pitfalls

Two examples should suffice: Simpson's paradox and missing values. In 1940, Deming had admonished the statistical community that "Students are not usually admonished against grouping data from heterogeneous sources." What he warned against, later became known under the name of "Simpson's paradox." A typical manifestation of the paradox is that two subgroups, when considered separately, both show a positive correlation between a pair of variables, but when they are pooled, the correlation is negative. Even though the paradox now is being treated in a few introductory statistics texts, Deming's warning still holds true, and the problem has been aggravated by the advent of data mining and of blind, automated procedures, which are almost guaranteed to run into Simpson's paradox without their user ever knowing.

Some kinds of missing values are easily recognizable, in particular if their presence creates problems with the algorithms of linear algebra. One usually tries to patch up such holes in the data by imputation, often using rather sophisticated iterative approaches (such as the EM algorithm). However, unless the values are missing at random (MAR), this will not get rid of deeper problems posed by them. Even nastier problems are posed by unrecognized missingness. We met examples in a big data set on highway maintenance, where one had forgotten to record a number of interventions (repairs). In this case, the missing values left no recognizable trace in the data set itself. We have met other examples, where the "missing" values actually were available in the data set under a different categorization (positive and negative outcomes had been treated separatedly). The original investigator had been puzzled by curious, counter intuitive effects that merely were due to his failure to recognize that he had missed some values at all.

Create order in data

Large, inhomogeneous data sets can be confusing when one first encounters them. As an initial inspection step, one has to try to create some conceptual order to promote understanding. Typically, this involves graphics – quite simple types of graphics may be the most helpful ones.

In his same 1940 paper, Deming also had pointed out "The modern student, and too often his teacher, overlook the fact that such a simple thing as a scatter diagram is a more important tool of prediction than the correlation coefficient, especially if the points are labeled so as to distinguish the different sources of the data." This still holds true, but with high dimensional data, straightforward scatter plots and scatterplot matrices soon reach their limitations. Some thoughtful preparatory dimension reduction may be needed before one can resort to scatter plots. Approaches based on clustering or on projection pursuit are in trouble for various reasons, in part because with massive or high dimensional data their computational complexity explodes. In my experience the most helpful approaches are based on principal components, or more precisely, on variants of the singular value decomposition, in particular on a version that has become known under the name of Correspondence Analysis, developed by the French statistician Jean Paul Benzécri from the 1960s onward.

REFERENCES

Benzécri, J. P. (1992). *Correspondence Analysis Handbook.* New York, Marcel Dekker.

Box, G. E. P. (1990). Comment. *Statistical Science*, **5**, 390-391.

Deming, W. E. (1940). Discussion of Professor Hotelling's Paper. *Ann. Math. Statist.*, **11**, 470-471.

Fisher, R. A. (1922). On the Mathematical Foundations of Theoretical Statistics. *Phil. Trans. R. Soc. Lond.* **A 222**, 309-368.

Huber, P. J. (2011). *Data Analysis.* Wiley, NJ.

Playfair, W. (1821). *A letter on our agricultural distresses, their causes and remedies.* London.

Tukey, J. W. (1962). The Future of Data Analysis. *Ann. Math. Statist.*, **33**, 1-67.