**Tue Tjur**

# Statistics in the computer age - personal reflections

*Summary.*

It is a trivial observation that the computers have changed the way statistics is practiced. But has it also changed the theory of statistics and the way we teach it? I think yes — even if the changes appear to be surprisingly small in some contexts. This is an attempt to give a more detailed answer, based on experiences from my own corner of the world from 1964 till now.

## 1. Introduction.

When I started studying mathematics in 1964 (which gradually turned into statistics in the following years), the only tools we had for computations were some very large and heavy Frieden desk calculators, which were able to perform additions, subtractions, multiplications and (believe it or not) divisions. They were purely mechanical (electricity–driven), extremely slow and a lot more expensive than the PC's most of us have today.

One thing that we learned the hard way these years was not to divide by zero. The resulting "NAN" or "infinity" was very clearly demonstrated. The thing would simply start subtracting zeroes from the nominators register, and since this was quite frictionless (as it did not push any digits) the denominator's wheel would rotate faster and faster, the only thing you could do was to turn off the power as soon as possible, otherwise it would — well, I actually do not know what it would, we never tried.

Another thing that could happen was that a very large denominator forced the carriage so far to the left that it fell off, in which case it was advisable to draw your feet away very quickly.

Later we had smaller and more advanced electronic desk calculators. But around 1970, we were still on the level where, I remember, a "square root button" on a certain type of calculator would increase the price by 2000 DKK (400$).

It may have been this speed of development that led us to the belief that, even if the very large automatic computers could be made slightly faster, bigger and cheaper, we were probably close to the limits of what was technically possible here. A transistor requires a certain amount of space and produces a certain amount of heat, and so does a ferrite ring for storage of a bit. Whereas it was much more likely that the

development in the car industry would result in a complete revolution of our transport system within the next few years.

When I look back today I find it hard to admit how completely wrong our predictions were. I knew only one computer at that time, the wonderful GIER 1 (Geodætisk Instituts Elektroniske Regnemaskine) in the basement of the mathematics department. The capacity of this computer was, in all respects, much, much smaller than the capacity of the eightie's popular game computer Commodore 64. Physically, this computer together with its power supply and external units occupied a tightly packed big office, and the price of it had been around ten times the yearly salary of an assistent professor. I loved programming it in the new and elegant language Algol — a forerunner for Pascal — whenever I had the chance (which was not often, perhaps a quarter of an hour around 3.30 a.m. once a week for an undergraduate student like me). Being almost a boy, I had of course a boy's dream, which was to own (or just to have more or less unlimited access to) a computer like this. Sadly, I realized that this dream was to remain a dream.

And what happened? The computer I have on my desk today (and, for that sake, the computer that almost any household has for basic administration, communication and entertainment) is about a million times bigger in RAM capacity and a hundred million times faster than the GIER 1. It is a lot cheaper, a lot smaller and a lot smarter. For example, the slow electric typewriter and the punch tape reader connecting GIER 1 with the outer world have been replaced with a high resolution color screen, a cheap and noiseless printer, a CD reader etc. Not to mention the supply of software for all purposes you can imagine, including statistics. Whereas my car has a disappointing similarity with the cars we were driving in the sixties. If there are any fundamental differences at all, they probably have to do with computers ...

In this paper I will try to describe how this enormous change of the computational environment over less than two generations has influenced the way statistics is apprehended and taught. But since I am pretty sure that this change has been very much the same in the statistical societies all over the western world (though things happened, perhaps, ten years earlier in the US and five years earlier in England than in Denmark), I will not try to make a long and boring exhaustive description out of it. I will simply describe the differences between statistics 45 years ago and today as they happened to appear in the near neighbourhood of a certain Danish university teacher in statistics. Names of persons, however important they may be, will not be mentioned, because it is endless once you start it. This paper should be considered a written version of an informal talk, it is certainly not a research paper.

I will, however, focus on the things that are particularly important for technical or applied mathematical areas like statistics. The revolution of

the society in general (and the entire scientific society in particular) due to the upcoming of e–mail, large data bases and the internet is not the topic of this article. The first generations of computers were intended for computing, as the name indicates. The idea of using them for trivialites like text handling and communication had hardly come up to the surface in the early sixties. Computations in astronomy, physics, chemistry, geodetics, meteorology, statistics, insurance and economy were the tasks considered. As to statistics (and all the other disciplines as well, as far as I know) the use of "electronic calculation machines" was practically non–existent in Denmark up to around 1960. In my treatment of the main topic, the impact of computing power on the theory and teaching of statistics (section 4), the focus will be on the way computing power has changed our way of thinking. The aim is not to dwell on the triviality that the increased computer power has allowed us to handle more computer–intensive models.

## 2. The mainframe era.

Around 1964, a GIER 1 computer, a Danish computer designed and produced by the Geodetics Institute at University of Copenhagen and "Regnecentralen" (a sort of independent research institute, originally funded by public money) was installed in the ground floor of the math department at University of Copenhagen. GIER 1 was a modern computer, based on transistor technology. The size of it was no more than that of a big closet, containing the network of ferrite rings constituting its 5 K of RAM and (later) a 64 K "drum" for background storage (replacing, in particular, the very long punch tape containing the Algol compiler to be read in sequentially when a program was too big to leave sufficient RAM for the compiler). The computer was run by the Mathematics Department's Numerical Analysis Group (later to become the Computer Science Dept.), but it could be used by other researchers and students at the science faculty.

Of course, no statistical software was available for the GIER 1. Some more or less homemade programs had to be used. I remember writing programs for one–way analysis and for the computation of tail probabilities in the $\chi^2$ and $F$ distributions. Simultaneously, I am sure, with thousands of statistics students and researchers all over the world. The problem at that time was that there was no easy way of communicating such programs. The barriers set up by differences in operative system, software, physical formats etc. were so overwhelming that it would usually be a lot easier to write a program for your own computer than to translate a program written by somebody else.

These communication barriers remained a great problem throughout the mainframe era. I can hardly believe how much time I have spent trying to connect plotters and computers that disagreed about their communi-

cation protocol; connecting printers and computers of different types via boxes of a third type with cables that did not fit any of them; breaking lines of output files in pieces before sending them through some stupid network server that was programmed to truncate lines of more than 80 characters, and then connecting the pieces again before sending them to a printer; or — our national sport — fighting to make keyboards, screens and printers handle the Danish–Norwegian special characters ÆØÅ(æøå) correctly. The university administration and our dear leaders did not always recognize how much work it was. The common attitude these days was that teachers who wanted to use computers in their courses would obviously have to take care of the computers themselves. If computers created more work than they saved, why have them at all? One had to admit, in between, that they had a point there. On the other hand, if we look at how things are today, it must also be admitted that the idea of introducing computers in statistics was not a complete flop. I am pretty sure that most participants at the DSC2009 conference here tend to agree. And even if the kind of problems mentioned above have not disappeared entirely, they are, fortunately, a lot smaller today.

Gradually it became clear also to the more theoretical fractions of the statistical society that the computers were here to stay, because they could do things that were otherwise impossible. Just to mention a local example, in Copenhagen a group of people working with Rasch's item response model came out quite early with a program that could compute the conditional maximum–likelihood estimates in that model. Lots of similar things happened all over the world. A large number of "kitchen table" programs for all sorts of statistical computations circulated. Commercial packages, gluing such programs together by more or less elegant data handling modules, gradually started to appear (SAS, GLIM, GENSTAT, BMDP, SPSS).

However, in the beginning of the seventies, the intensive use of computers in statistics was still an exclusive privilege for the few. The reason for this was, first of all, that computer time was so very expensive. Computer screens were also expensive and not very good, the standard way of communicating with a computer was still via punch tape or cards for input, line printers (sometimes quite far away) for output. Interactive access to a computer was also expensive, most jobs had to be carried out as batch jobs, sometimes with a delay time of up to 24 hours, because the more heavy computations would have to be done during the night hours.

In the Copenhagen area, an important step was taken when SAS was implemented on NEUCC's IBM computer (that must have been around 1977). SAS was founded in 1976 in partnership with IBM. The first PC (DOS) version is from 1985, before that SAS was closely linked to IBM's mainframe computers. Which — typically for the time — was why SAS

was used earlier at the technical university in Lyngby, 20 kilometers north of Copenhagen, than at the science departments of University of Copenhagen. Lyngby, where the new institution NEUCC was located together with the new technical university, was simply too far away. At University of Copenhagen, statistical packages were essentially not in use before 1979, and the package that was introduced first was not SAS, but the Rothamstead package GLIM, soon followed by GENSTAT, installed on the Univac computer at the new Regional Computing Center at University of Copenhagen, geographically close to the mathematical institute.

NEUCC, the Northern Europe University Computing Center founded in 1965, deserves a few words here because the construction was so typical for the time and demonstrates so very well how the proportion between manpower and computerpower has changed. This center was geographically located at the new campus of the Technical University in Lyngby north of Copenhagen. The original idea was that it should supply the necessary computer power to the scientific communities of the Nordic countries and Holland. To this end, it was equipped with a staff of (to begin with) 22 persons and an IBM7090 with a RAM of 127 K and a speed of 400 Hz, donated by IBM for who knows what reasons. Nevertheless, NEUCC was a very important player in scientific and technical computing in Denmark for many years (with a somewhat increasing computer power, of course).

### 3. The personal computer era.

During the eighties, the "IBM–compatible PC's" started taking over. In the beginning, these small personal computers were mostly used for text handling and administration. Hard core computational statisticians would turn up their noses at these small computers. But very quickly, they grew bigger and bigger, and lots of software for them — including PC versions of statistical packages — came on the market.

One of these software packages deserves to be mentioned here, Turbo Pascal (ver. 1 1983, ver. 3 1986). This pascal compiler with its elegant "integrated environment" was a scoop, which really changed the working conditions for programmers on all levels. The programmer's full control over the entire computer (in particular the screen image) made it a pleasure to do programming. Many statisticians have made a lot of statistical programming connected with graphics and computations, which could not have been done any easier by other systems. For example, the reading and merging of complicated data files, the estimation of non–standard models, large simulation studies etc. was (and is) quite often a lot easier to do by a Pascal (or C++) program than by a statistical package.

The history from then on is, perhaps, less interesting because it is more

or less well known to most of us. SAS expanded to an immense, menu–driven giant which, on the statistical side, is not much more powerful than the earlier mainframe versions. GENSTAT seems to be living its own life in a rather narrow niche. These packages have extended somewhat on the model side, in particular as concerns regression models with random effects. But in the same period, S–plus and R have developed to extremely powerful programming languages. From an amateur's point of view, the availability of point–and–click interfaces to SAS, JMP, SPSS and many smaller packages for special purposes may be an advantage. From a more professional point of view it is a disaster, because everybody tend to use them when they are there, and this inspires to a use of these packages which is difficult to document and communicate to others, and sets up an artificial barrier between the ordinary user and the more advanced programmer. The purely command–driven language R with its open source and free license policy seems to be the closest we can come to an intermediate winner of this ever–lasting contest.

## 4. The impact of increased computer power on the theory of statistics.

My first course in statistics in 1965–66 included a rather modern introduction to the the unified theory of linear models, based on linear algebra and matrix calculus. The characterization of the least square estimates and the analysis of variance tables in terms of orthogonal projections on linear subspaces of the observation space was the main result, from which Cochran's theorem and various distributional results about test statistics etc. were derived. Thus, we were essentially presented to the theory exactly as it is today.

However, a fundamental difference from the way we would do it today was that the computational aspects were not discussed as a general matter. Nobody told us about the model matrix $X$ and the expression for the least squares estimates involving the inverse of $X'X$. These subjects would be rather useless to us anyway, because the matrix inversion would, in most cases, be impossible with the tools we had. Instead, the explicit formulas for the estimators in simple regression and balanced, orthogonal analysis of variance models, were derived directly from the principle of least squares.

For the general linear model we did not even discuss parameterizations of the mean vector, it was merely assumed that it belonged to some linear subspace of the observation space. Consequently, we did not learn either about conventions for how to select a one–to–one parametrization of the mean vector and the many traps connected with this. A topic which, still today, is slightly underrated in textbooks, in my opinion. In order to interprete the parameter estimates correctly one must know the rules for how a model matrix is generated from a model specification, and how

the thinning of its columns to a set of linearly independent columns is performed. One of the most common error sources in applied statistics on the amateurs' level today is probably the lack of understanding of the lists of parameter estimates that are produced by statistical packages.

A small thing that confused many of us a lot was that the simple regression model was often written on the form $\mathrm{E}y_i = \alpha' + \beta(x_i - \bar{x})$ rather than just $\mathrm{E}y_i = \alpha + \beta x_i$. This was, of course, to simplify the estimation by introduction of a model matrix with orthogonal columns. But today, we would probably not care to change the parameterization in this way, we would leave the computations to a computer and write the regression line on the standard form with a slope and an intercept. In those days, the centered independent variable $x_i - \bar{x}$ was much more concrete to us, because we would have to work with it more or less directly when we did the calculations.

Quite generally, it is a characteristic feature of the computer age that it has allowed us to let the computations play a less dominationg role, thus leaving more time and attention to the statistical model and its properties. Earlier, it was more common to think of statistical models in terms of the computations. Most extremely, this can be seen in classical presentations of analysis of variance, where almost everything is interpreted and explained in terms of computed averages and variances, with a minimum of reference to the underlying probabilistic model.

This development is strongly supported by the upcoming of powerful facilities for simulation. In most statistics packages — and also in many other programs — it is easy to take a given statistical model with a given set of parameters and simulate a dataset from that model, or for that sake a million datasets from that model. This is extremely useful in situations where the asymptotic distributions of estimates and test statistics are not quite reliable. But at the same time, it gives us a very concrete interpretation of what we are actually doing when we analyze a dataset by a given model. Before (say) 1970, we would say that "the assumption behind the model is, that data are (so and so) distributed", for example "independent normal with the same variance and expectations of the form ... ". Today, we would perhaps tend to say (or at least to think, the words we say have a tendencey to change very slowly) that we analyze the data set by *comparing it* with (real or imagined) data sets created by simulation. Comparison with simulated data sets is, for example, exactly what is going on when we make permutation tests or bootstrapping. We do not, quite as much as in the old days, need to imagine some random mechanism that generated the data.

At the same time, we have quite generally developed a more relaxed attitude to distributional assumptions, like variance homogeneity and normality. Lots and lots of computer simulations have shown us, that even a model that clearly does not hold, due to a significant deviation

from normality or a significant heteroscedasticity, can produce quite reliable inference when the deviations from the model assumptions are moderate. The hunting for small deviations from normality by q–q–plots etc. has become less hysterical. At the same time, we have realized that overdispersion is a more serious matter, which can produce very misleading results if it is ignored (usually in models for Poisson or binomial data). I think that these changes of the weights we put on our assumptions have to do with the fact that it is so very easy to perform simple experiments with real or simulated data. Who has not tried, for example, to analyze the same data set with both a weighted and an unweighted model to see what difference it makes, or to produce a number of q–q–plots for simulated data, just to see what they look like when data are actually normal. The sum of all these experiences is a part of our knowledge about what is important and what is not important when we practice statistics.

*Generalized linear models.* This topic is a genuine child of the computer age. When these models are explained mathematically, in terms of link functions and one–parameter exponential families, it is difficult to convince anyone about the relevance of the topic. It is obvious that the class includes a lot of important models, but it is not obvious at all that there is anything to gain by putting them into this common frame of reference. The real justification of the idea lies in its computational aspects. Historically, the important observation was that a single iteration step in the Newton–Raphson maximization of the log–likelihood is computationally equivalent to the solution of the normal equations for a weighted multiple regression model with the same model matrix. This meant that the core of the program was already written, the remaining work was roughly a matter of putting the linear model program into a loop and add a few details before and after. The tedious part of it, which is the translation of a model formula to a model matrix (with all its complex rules for handling of factors and interaction terms), the thinning of the columns of that matrix to a set of linearly independent columns, the listing of the estimates and their approximate standard deviations etc. etc., all these problems had already been solved. The interactive Rothamstead package GLIM was a great success because it extended the class of "computable" models from the multiple regression models to a class that contained the log–linear Poisson models, the logit–linear binomial models and a lot of non–linear regression models for normal (and even $\Gamma$–distributed) variables. As an important by–product, a reasonable way of correcting for overdispersion was more or less automatically included.

In the context of hypothesis testing, it is my impression that there has also been some change of attitude which can be ascribed to the computers' influence, even though this has not yet had much impact on the textbook literature. The classical way of explaining the significance test

is to say that *first* we choose a level $\alpha$ for the test, typically $\alpha = 0.05$ or 0.01; *then* we compute the test statistic, and finally we compute the P–value (or use a table) to check whether the test statistic is more or less extreme in its distribution than this level indicates. If it is more extreme, we "reject" the hypothesis, if it is less extreme we "accept" it.

It is difficult to imagine that any sensible person has ever followed this absurd scheme. Except, perhaps, in very special situations where a proper binary decision should actually be taken (like "send application to FDA or do not"). Nevertheless, this way of explaining the concept of significance testing has persisted for around 70 years, and is still the standard in many textbooks. It is, however, also my impression that this awkward attitude is somewhat declining in the more informal teaching, in particular when it comes to situations where a test should actually be performed. Personally, I have a long time ago decided to explain it in a way which is more in accordance with the way statisticians actually behave. Namely roughly as follows.

The basic idea, which many students have a lot of difficulties with the first time they see it, is that if an extremely large value of the test statistic comes out, we are forced to the conclusion that the hypothesis must be wrong, because the alternative is to accept that an extremely rare event has happened. If the test statistic is not extreme, we cannot say anything (so we "accept" the hypothesis). Extremeness of a (one–sided) test statistic $t$ is measured by its P–value $1 - F(t) = P(T \geq t)$ on a (reversed) scale, where 0.05 and 0.01 are traditionally taken as important benchmarks. And — to some extend in conflict with the classical textbook explanation — the P–value should usually be reported as a measure of the conclusions validity, in particular when it is small. It is *very misleading and therefore strictly forbidden* to restrict the reporting to phrases like "significance on the 5% level" in situations where the actual P–value is, say, smaller than $10^{-6}$.

This, I think, is the important part of the message. All the stuff about type 1 and 2 errors, power functions etc., does not belong in an introductory statistics course, in my opinion. And I am pretty sure that I am not the only one who thinks like that. But, now to the point; I think that the real reason for this change in attitude is not only the absurdity of the textbook explanation, but also the fact that statistical tables with their standard thresholds at 95% and 99% are no longer used, except in artificial situations like written examinations. The textbook explanation made a little more sense when we used tables. Or, perhaps we should put it this way, the textbook explanation appears even more absurd when we have a six–digit P–value printed out for any test that we perform.

## 5. Some computers I happen to have known.

| | | Comparisons with modern PC | |
|---|---|---|---|
| 1960 | GIER 1 | | |
| RAM | 5 K | 800,000 | |
| Clock | 50 MicroSec | | 120,000,000 |
| 1959 | IBM7090 | | |
| RAM | 127 K | 31,500 | |
| Clock | 2.5 MicroSec | | 6,000,000 |
| 1982 | Commodore 64 | | |
| RAM | 64 K | 62,500 | |
| Clock | 1 MicroSec | | 2,400,000 |
| 1983 | Olivetti M24 | | |
| RAM | 512 K | 7,800 | |
| Clock | 5 MHz | | 480 |
| 2007 | My computer today | | |
| RAM | 4 GB | 1 | |
| Clock | 2.40 GHz | | 1 |