Patricia M.E.Altham
Statistical Laboratory, Cambridge University
Seminar for University of Piraeus, May 2006

Being lucky enough to be in the right place at the right time? My statistical education, and some developments in statistical computing.

`http://www.statslab.cam.ac.uk/~pat`

$\kappa\alpha\lambda\eta\sigma\pi\epsilon\rho\alpha$

I have been incredibly lucky to have had my career at Cambridge University, teaching many generations of excellent students. They probably taught me as much as I taught them, and this was particularly true as far as statistical computing went, where learning the new language was more often than not a joint effort by me and the graduate students.

Here is a rapid chronology.

1964-5 My Diploma (=MSc) year. We used firstly a mechanical calculator called a Brunsviga, and then upgraded to a Facit. These were large and clumsy electric calculators.

Putting all the data in **correctly** was the first hard task.

Doing a 2-way anova, on (y_{ijk}) by computing $\Sigma(y_{i..})^2$ and so on, was really hard work.

Since the principal aim was to find the residual sum of squares, the first cause of heart-sink was when you realised that that this 'sum of squares' had come out negative! Nowadays,

```
aov(y~ A+ B)
```

for example with factors A,B, will of course give you the results, all the F-Tests etc, the instant you have typed the formula.
Partly because of the constraints of calculation, it was really important to understand concepts such as

orthogonality

and this in turn meant that our lecture course 'Design of Experiments' was probably seen as much more important than we would find in a modern graduate course on statistical science. At Cambridge, our 'Design of Experiments' course disappeared some years ago, and new courses such as Financial Mathematics, Actuarial Statistics, have been introduced: such is the evolution of a course, partly reflecting the interests of our staff and of our lecturers.

Needless to say, the 4hr Diploma practical exam was a real test of endurance.

Nowadays, our MPhil practical exam is a three and a half day, 'open book' examination.

From 1965-70, I was doing a PhD, ‘Bayesian inference for multinomial data’ while at the same time working as Scientific Officer for the Medical Research Council, in effect as the resident statistican for the Applied Psychology Unit in Cambridge.

This proved to be very valuable experience in statistical consulting, with interesting (and demanding!) colleagues in Experimental Psychology. My expertise in computing made slow but steady progress, using Cambridge Autocode (did we then use Cambridge Phoenix as the control language? Perhaps Phoenix did not come in until the 1970’s).
This involved preparing the programs on punched paper tapes and transporting them to be processed on TITAN at the Computer Laboratory, about 10 minutes’ walk away.

Then the next day I would call back to collect the printout of the results, or (more usually) to collect the error message, perhaps telling me I had mispunched 2 characters. We used paper tape, with 5 binary digits + 1 'parity check' digit, and I got quite adept at repairing such tapes, with Sellotape and a hole-punch.

The fact that the computing was so **laborious** did have an influence on the way I did research in those years.
It was highly desirable to derive **closed-form analytic solutions**, if possible, preferably ones involving summations rather than integrals.

For example, and let ν be a positive integer. consider a Poisson process of rate λ. Let T_ν be the time from 0 to the νth event in this process.

We know that T_ν has a gamma distribution, parameters λ, ν, and we may seek a simple summation as a way to evaluate the tail area $P(T_\nu \leq t)$, for given $t > 0$. A simple way to do this is to integrate by parts, but a better way to arrive at the same expression is to use a probability argument, based on what we know about a Poisson process.

Thus, let N_t be the total number of Poisson events in the interval$[0, t]$.
You can easily check the following identity between the Poisson tail probability and the tail area of a gamma distribution

$$P(T_\nu \leq t) = P(N_t \geq \nu),$$

where of course $N_t \sim Po(\lambda t)$.

The Bayesian application of this result is that if we have a prior density $\pi(\lambda)$ for the unknown parameter λ, for example

$$\pi(\lambda) \propto \exp(-a_0\lambda)\lambda^{b_0-1}$$

for $\lambda > 0$ (a gamma prior density), and data $x_1, \ldots, x_n$ from the $Po(\lambda)$ distribution, then the posterior density for λ may be written

$$\pi(\lambda|x_1, \ldots, x_n) \propto \exp(-a_1\lambda)\lambda^{b_1-1}$$

for suitably defined a_1, b_1. The posterior density of λ is therefore another gamma density, and we can represent

$$P(\lambda \leq \ell | x_1, \ldots, x_n),$$

say, as the upper tail of a Poisson variable. This was dependent on choosing a prior density $\pi(\lambda)$ of a form conjugate to the Poisson. If, in addition, we choose the prior $\pi(\lambda)$ to represent 'no prior information', eg

$$\pi(\lambda) \propto 1/\lambda$$

for $\lambda > 0$ then we may be able to see a very close relationship between classical and Bayesian inference for λ.

This kind of simple mathematical relationship seemed important at the time, for 2 reasons:

i) while practical Bayesian statistics was still in its infancy, we who might have been trying to be good Bayesians needed all the help we could get in manipulating the posterior density: if we could write its distribution function as a summation, the computation was going to be that bit simpler.

ii) more importantly, much energy was expended in worrying/arguing/discussing the differences between the Bayesian and the classical approaches to statistical inference. If a few lines of mathematics could show you that you would be getting almost the same p-values, or confidence intervals, for example, so much the better. I for one was glad to be able to (almost) reconcile the 2 approaches, in a particular special case.

In this spirit, I was very happy to be able to find an exact result linking the classical and Bayesian approaches in the context of a 2×2 contingency table (n_{ij}). Here, as you probably know, for an exact test of independence of the rows and columns of the table, we can use Fisher's test

```
fisher.test()
```

This is based on the hypergeometric distribution, with frequency function

$$\frac{\binom{n_{1+}}{n_{11}}\binom{n_{2+}}{n_{21}}}{\binom{n}{n_{+1}}}.$$

The Bayesian version of the same test would be to evaluate

$$P(\theta_1 < \theta_2 | (n_{ij}))$$

where, given the data (n_{ij}), the parameters θ_1, θ_2 are independent random variables $\beta(n_{11}, n_{12})$,

$\beta(n_{21}, n_{22})$ respectively.
I was able to show (published 1969) that this posterior probability is almost identical to the hypergeometric tail probability computed in the classical approach to this problem.

Nowadays, we can tackle very complex multi-parameter problems with enormous computing power (eg using MCMC methods) but perhaps there is little incentive to try old-fashioned analytic methods to help give us more insight into our inference procedures.

Galloping rapidly from the 1960's to the present day:
1970's FORTRAN (I was fairly inexpert at computing in this decade. Was there specialist statistical software available in the 1970's? perhaps SPSS?)
early 1980's BBC Basic, on BBC micro-computers, then at last, software purpose-built for statisticians:
mid-1980's Glim, firstly on the IBM 'mainframe', and then via a disc on my own BBC micro-computer.

It wasn't until about 1985 that I could confidently sit down with a scientist, (a 'client'), look at his/her data via GLIM on the BBC computer, discuss the results and their interpretation, and print them out, probably all in a single 1 hour period.

From 1986 (via the mainframe) I could also use Genstat
(two excellent textbooks here, one being the Glim book by Aitkin, Anderson, Francis and Hinde (1989), the other being the Genstat manual).

All of these languages were used by our graduate students, eg for the practical classes, but not yet available to undergraduates, though I do remember getting students to help me to carry into the lecture-room a BBC microcomputer to show undergraduates the use of GLIM via a disc.

Graphics facilities were still very limited, mostly line-printer plots, on the screen (but in fact perfectly adequate for getting the 'general picture', eg with the standard diagnostic plots in linear regression.)

Significant dates (ie I remember them!)
We started using Unix as the control language in 1987. Later we moved to Linux. (I avoid using Windows if possible!)

- I started teaching undergraduates about glm, with practical classes using GLIM, in 1992. Practical classes for large groups, eg 20 or more, required me to learn new sorts of teaching skills. I grew up exclusively with 'talk and chalk' teaching.

Generally I acquired such skills the hard way, ie by making mistakes, and realising what does work and what does not.

At first I made 2 particular (related) errors
i) worrying too much about when I would do when students asked me questions about the software (or hardware) that I couldn't answer at once
ii) trying to do too much on my own: things went much more smoothly when I enlisted the help of a colleague, Dr Brian Tom, for the first 2 practical sessions of the undergraduate course.

- I started learning S-Plus in 1992, and using it with graduate students. The MASS text-book by Venables and Ripley (of course I have bought all 4 editions), with its online data-library, was a tremendous help here.
- I started Latex in 1994, followed in about 1997 by learning how to put documents on my webpage: thus I have a basic knowledge of HTML.
- I started learning R in 1998, and using it for the undergraduate course. Of course, the free access to R was (and still is) a big plus for the students.

Important things I have not yet tried to learn include

i) other packages, eg STATA, SAS.

ii) more snazzy presentation techniques, eg Powerpoint.

I'm still rather unambitious in my use of graphics, eg with R.

There are still important features of R, such as 'scoping', that I don't fully understand, and so I fail to make use of them.

Conclusions

The early days: how on earth did we **stick with it?**
Looking back on the 1960's, it seems that it was **so** laborious to do what we now regard as trivial calculations.
No doubt we had a big incentive to stick with it since we had good lecturers, and we trusted them. I'ld like to think that we were also aware that being in the UK, and in Cambridge, we were privileged to be even a small part of the great leaps forward in scientific computing.

I have to say that sometimes I met new technology with sheer ingratitude, for example

grumbling

at having to update from using Glim on a disc to using it via Unix on our new network (1987) of Hewlett Packard workstations.
(would I still be using an abacus if left to my own devices?)

What lessons can we learn for the future?
A personal lesson that I learned was that in applied statistics, and particularly in computing, one often learns by one's mistakes.
(Of course, this feature is at first a bit upsetting to those of us whose initial university education was traditional formal mathematics.)

Many of the computing developments of say the 1980's and 1990's were, in the 1960's say, completely unimaginable.
What will be the developments in applied statistics in the 2000's?
Examples:
the analysis of **very large datasets**
eg from genomes, as in modern bioinformatics, or in data-mining (as for financial databases).
We will need to develop both appropriate theory and computing methodology.

$\epsilon\upsilon\chi\alpha\rho\iota\sigma\tau\omega$