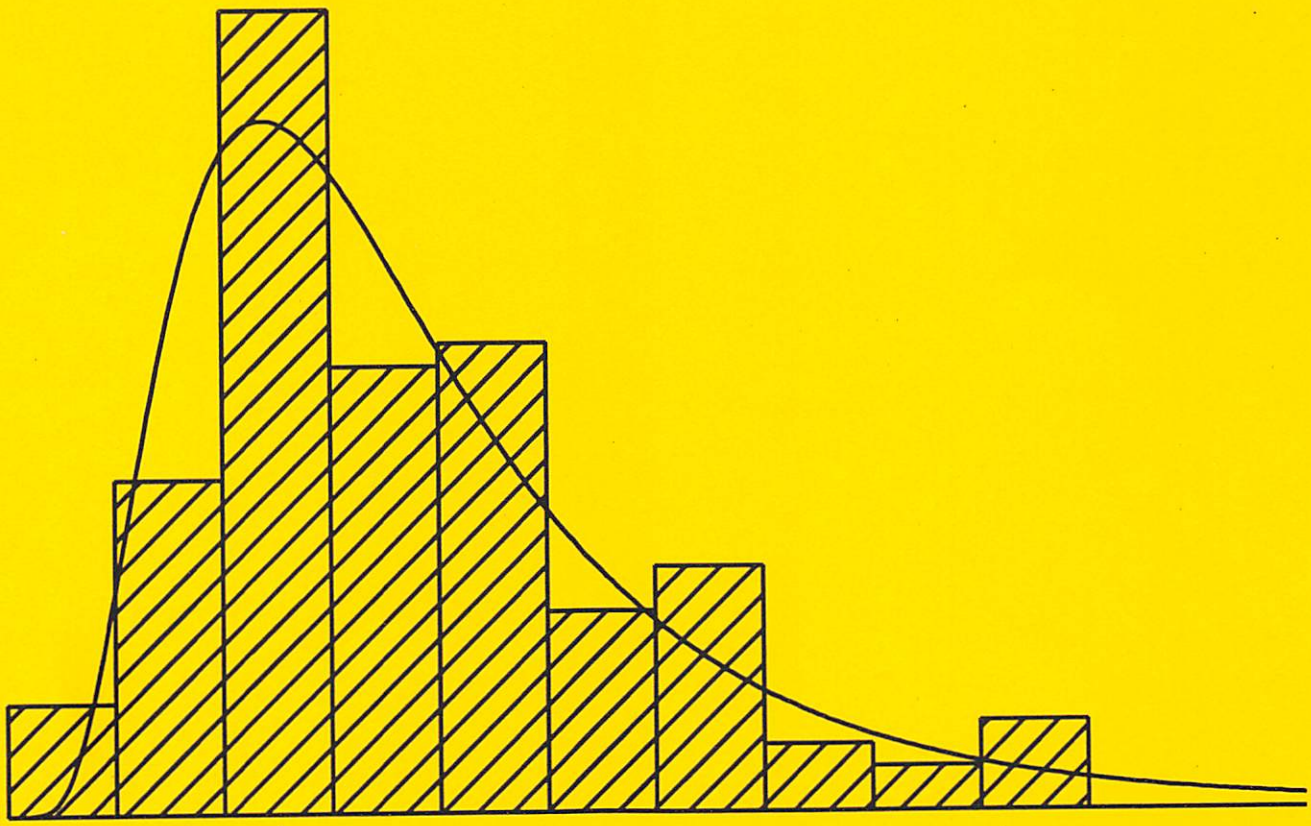
NAG®

GENSTAT

Newsletter

Issue 32



Editors

Sue Welham
AFRC Institute of Arable Crops Research
Rothamsted Experimental Station
HARPENDEN
Hertfordshire
United Kingdom AL5 2JQ

Anna Kane
NAG Ltd
Wilkinson House
Jordan Hill Road
OXFORD
United Kingdom OX2 8DR

Printed and Produced by NAG®
NAG is a registered trademark of:
The Numerical Algorithms Group Ltd
The Numerical Algorithms Group Inc
The Numerical Algorithms Group (Deutschland) GmbH
Genstat is a trademark of the Lawes Agricultural Trust

ISSN 0269-0764

NAG Bulletin Board:
Gopher: Name= NAG Gopher Server, Type=1, Port=70, Path=1/, Host=www.nag.co.uk
Mosaic: http//www.nag.co.uk:70/

Genstat Newsletter

Issue 32

NP2908

July 1995

Contents

Published by
The Rothamsted Experimental Station Statistics Department
and The Numerical Algorithms Group Ltd

Editorial

The editors would like to take the opportunity to remind Genstat users that this is a conference year; the Ninth International Conference of Genstat Users having taken place at University College Dublin in July. The conference offered an ideal opportunity for Genstat users to discuss ideas and influence the future development of Genstat, and to view a test version of the eagerly-awaited Genstat for Windows system. Papers arising from this conference will appear in future issues of the Newsletter. To give users some idea of exactly what happens at a Genstat Conference, over the page we have reprinted Jane Speijers report of the Third Australasian Genstat Conference which was hosted by Agriculture New South Wales in Wagga Wagga at the end of last year, and several articles in this issue are based on the presentations given by Chris Glasbey and David Smith at the conference.

This issue begins as usual with another helping of Genstat Talk - a selection of the diverse set of topics aired on the Genstat discussion list. Users should note that the address of the list has changed, due to action by the list administrators. Everyone currently on the list should have been changed over automatically, and users wishing to join the list should now send the message

```
SUBSCRIBE Genstat first-name last-name
```

to the new address

```
LISTSERV@LISTSERV.RL.AC.UK.
```

Genstat Talk is followed by more detailed articles discussing a wide variety of Genstat facilities and applications. The first of these articles is a short paper, illustrating how Genstat's generalized linear modelling facilities can be used to describe the colour preference of insects. Multivariate techniques are represented in the Newsletter once again, when the second article describes a method for simplifying single and complete linkage dendrograms for cluster analysis.

Next comes a detailed paper explaining the use of the new **DISTRIBUTION** directive when fitting the negative binomial distribution, with examples and including a helpful interpretation of the Poisson and Negative Binomial Indexes. Then continuing the Newsletter theme of introducing external programs which may be used in conjunction with Genstat, the next article introduces TWOD, a standalone program which now has an interface to Genstat via procedures, and which is used to produce efficient analyses of field experiments using two-dimensional spatial models.

The experimental design capabilities of Genstat are discussed next in detail, and some of the new design procedures in the 3[2] procedure library, which is about to be distributed, are introduced. This issue is then rounded off by two articles discussing the computation of generalized estimating equations (GEEs) for general repeated measurements, and repeated ordinal measurements respectively.

As usual, the code for any procedure listed in any Genstat Newsletter will be made available via the NAG bulletin board, which is run under the Gopher server. Connection details may be obtained on the inside of the front cover of the Genstat Newsletter.

A software repository has now been set up as part of the Statlib system at Carnegie Mellon University. Anyone can submit material, and anyone can access it by email, gopher or WWW:

WWW location:	http://lib.stat.cmu.edu/genstat
gopher:	connect to port 70 on lib.stat.cmu.edu (This gets you to the top level of Statlib)
email address:	statlib@lib.stat.cmu.edu

Statistical conference for Genstat users, November 1994

J Speijers
Biometrics Section
Department of Agriculture
Baron-Hay Court
South Perth 6150
Australia

Charles Sturt University in beautiful Wagga Wagga, the largest inland town in NSW, was the venue for this conference which was attended by eighty eight statistical types from Australia, New Zealand and the United Kingdom. Twenty-two participants were from the NSW Department of Agriculture who sponsored the conference along with the Statistical Consulting unit at ANU, the CSIRO Biometrics Unit and CEANET, distributors of Genstat in Australia. New Zealand (9) and Great Britain (8) were also well represented but there were only three of us from the West, that is if you include South Australia. But then, as we discovered, Perth is further than New Zealand from Wagga.

Congratulations to Brian Cullis and other members of the program committee and the local organising committee for an interesting and well organised conference. Fiona Thompson, the conference secretary, was looking very harassed on the Friday before everyone arrived, but relaxed as events proceeded successfully. In contrast, those that attended the pre-conference weekend of trout fishing or walking in the Brindabellas, organised by Ross Cunningham, looked relaxed right from the start. Talks included very clear expositions of several new modules in Genstat 5.3 given by Peter Lane, Roger Payne and Sue Welham, more theoretical talks by John Nelder and Robin Thompson and a presentation by John Deaker from CEANET regarding their approach to Genstat support in Australia. Bob Murison from NSW Ag gave an entertaining talk on the analysis of correlated ordered categorical responses in the isolation of Tamworth and Phil McCloud amused us all with his self-indulgent (his words) description of applying the EM algorithm to incomplete categorical repeated measurements. I have a lasting memory of John Nelder with a recent model laptop perched in front of him, baring his teeth in frustration as his newest Genstat macro failed to work.

The need for a good implementation of Genstat under Windows was stressed by all those using Genstat on a PC. It was comforting to hear that this project will be tackled very soon at Rothamsted with the help of David Baird from Ag Research, New Zealand, who impressed us all with his proforma windows interface to Genstat.

Interspersed with the serious business of statistics we managed a few social and sporting activities. The Charles Sturt University swimming pool, adjacent to the convention centre and our accommodation, was very popular, particularly at lunchtimes when the days were very hot. On the first evening of the conference we were invited to a barbecue where we were introduced to the very drinkable wines produced by the University's winery which recently won an award as the best boutique winery in Australia. The wine flowed freely, tongues were loosened and, as I remember, we all had a very pleasant evening. The organising committee must have presumed that we had really come to Wagga to taste the wines, because the conference dinner was held at the Wagga Wagga winery on the following evening. Another night of drinking and eating with a few interlocking metal link puzzles thrown in for those who had not had enough mental activity. At our table past and present residents of Perth were particularly successful at solving these. I'm sure that like myself most of those who attended the conference returned to work full of enthusiasm for using the new ideas to which we had been exposed. In Perth we have begun to make use of Genstat's high resolution graphics but as yet have not come to grips with John Nelder's double generalised linear models. On the question of suitable attire for Genstat users, our Department has ruled against canary yellow jackets as worn by Robin Thompson on the grounds that they would not match our corporate uniform.

Acknowledgement

This article first appeared in the March 1995 issue of the Statistical Society of Australia Newsletter.

GENSTAT TALK

Extracts from the Genstat electronic discussion list, May to December 1994, summarized and edited by Peter Lane, Rothamsted. To join the discussion, send the message:

`SUBSCRIBE Genstat first-name last-name`

to the address: `LISTSERV@LISTSERV.RL.AC.UK`

The opinions expressed here are not necessarily endorsed by either NAG or Rothamsted, and statements may not have been checked for accuracy. However, members of the Genstat development team and of NAG's Statistics Section are contributors to the discussion.

Re-ordering text

Query: A simple task that does not seem to admit a simple solution: take a text `T` and construct a second text `S` with the same elements but in a (given) arbitrarily permuted order. The use of `$` as in

```
CALC S = T$[!(3,2,1)]
```

does not work with text. Ugly solutions can be found using `EDIT` and `EQUATE`; but there must be a better way.

Triggs@mat.aukuni.ac.nz

Reply: Use `SORT`, as in

```
SORT [INDEX=!(3,2,1)] T; S
```

rod@maths.marc.cri.nz

Inconsistent structures

Query: I get some more or less incomprehensible messages when I use `DELETE`. For example,

```
DELETE [REDEFINE=yes; LIST=all]
```

after using procedure `GLMM` gives a warning VA19 about inconsistent structures, and lists 23 identifiers including `GLMM` and several beginning with underline. Is there a way of suppressing these messages? Are they telling me something I ought to take account of?

jeff@canopy.biom.csiro.au

Reply: The warning message is generated because there are some structures remaining after the `DELETE` operation that are "inconsistent". This means that they have been set up to refer to other structures that have now disappeared or changed. Such messages can be generated by, for example, deleting a text structure that has been used to define labels for a factor. However, in the reported case, the inconsistency is caused by undeleted structures in the `GLMM` procedure that point to deleted structures. The message can safely be ignored, and it can be suppressed by switching off warnings temporarily using the `DIAGNOSTIC` option of the `SET` directive.

peter.lane@bbsrc.ac.uk

Analysis of covariance

Query: I'm trying to do an analysis of covariance, but Genstat doesn't seem to be doing anything with my covariate. I have only one covariate value for each factor combination in a fractional factorial. The Manual says that the adjusted analysis of variance is the extra sum of squares removed by the covariates **after** eliminating all that can be ascribed by the treatments – but in my case, if I fit all the factors I can, there isn't any variation left for the covariate to account for. What I'd like to do is fit the covariate **first**.

duncan.hedderley@bbsrc.ac.uk

Reply: You could try using the regression directives, since there is only a single stratum. The covariate effect is completely confounded with treatments, so you will have to be very careful with the interpretation. But the `FIT` directive will fit the effects in the order you include them in the parameter. You can use `PREDICT` to produce tables of means adjusted for the covariate (setting the `ALIAS` option to deal with aliasing).

peter.lane@bbsrc.ac.uk

Fitting distributions

Query: I've never used `FITNONLINEAR` before, but I want to use it to fit Normal, LogNormal, Weibull and Gamma distributions. The Release 1 Manual (Page 384) shows how to deal with the Normal distribution; is there a quick and easy way to deal with the others? I don't want anyone to expend a lot of energy on this.

sparks@vaxa.nerc-monkswood.ac.uk

Reply: You can use the directive `DISTRIBUTION` in Release 3.1.

van.biezen@ibn.agro.surf.nl

Macros in procedures

Query: Can anybody help me with using macros inside a procedure? Maybe there is an alternative which doesn't require using them. I have about 25 lines of Genstat code that I want to repeat over and over inside a procedure. If I use the macro in a normal program there are no problems. Once I try to use it inside a procedure I get an error that the text is an improper structure.

p.baker@prospect.anprod.csiro.au

Reply: The problem is that macro substitution takes place when the procedure is being defined, not when it is being executed. The contents of the procedure are not interpreted until it is being executed, so no definitions of texts inside the procedure can be used. You can, however, define a text, say `T`, before defining the procedure, and then use the macro substitution `##T` in the definition of the procedure. In Release 3 there is no problem: you can use the new `EXECUTE` directive which provides execute-time substitution of the contents of a text.

peter.lane@bbsrc.ac.uk

S.e.s for variance components

Query: We are currently using Genstat to analyse results from measurement capability studies. The `VCOMPONENTS` and `REML` directives produce estimates of variance components and their s.e.s. How reliable are the `REML` s.e.s?

icox@theraj.enet.dec.com

Reply: The s.e.s come from the inverse information matrix of the estimated variance components and so cannot generally be used for reliable tests unless based on very large samples. One approach you could take is to perform likelihood-ratio tests on the components: since `REML` calculates (residual) maximum-likelihood estimates of the variance parameters, you can drop these in turn from the model (analogous to testing fixed effects using normal maximum likelihood) allowing one d.f. for each variance parameter estimated. You then look at the difference in deviance (-2×log-likelihood) between nested variance models compared to the appropriate chi-squared distribution. Note that you **must** keep the same fixed model throughout.

sue.welham@bbsrc.ac.uk

Limits on text

Query: Does anyone know what is the maximum number of elements in a text? I could be looking at typing in 800+ adjectives for a Procrustes analysis: am I likely to run into problems with storage or the maximum length of a `VALUES` option?

duncan.hedderley@bbsrc.ac.uk

Reply 1: I don't think you are going to hit problems with even 2000 adjectives. I tried a dummy run trying to read in a text of length 2000, with entry 'browny-green' 2000 times, and found the dataspace before and after with

```
HELP env,space
```

and there was plenty of space left on our system.

ian@sass.sari.ac.uk

Reply 2: The size of character space is fixed in each implementation of Genstat. In Release 3.1 there is space for 102,400 characters for Vax/VMS and 204,800 for PC and Sun/SunOs. This space is not used for identifiers (another block allows up to 8000 or 16000 respectively) but is just for storing texts. However, procedures are stored as texts, so you can soak up space if you use many procedures in a job.

peter.lane@bbsrc.ac.uk

Comparing PCAs

Query: Suppose I have a set of intercorrelated variables which I measure on a number of units. The units are classified into groups, and I want to know if the interrelationships between the variables are the same for each group. If I do a principal components analysis on each group separately, can anyone think how I might compare the loadings from each group?

duncan.hedderley@bbsrc.ac.uk

Reply 1: You could carry out PCA on all the units together and follow with analysis of variance using the component scores.

ccsphc@sunserver1.bath.ac.uk

Reply 2: The interrelationships are summarized in a covariance matrix for each group. I would then think of Bartlett's test for equality of covariance matrices.

h.van.der.voet@glw.agro.nl

Pentium PC

Query: One of my colleagues who runs Genstat on a 486 is planning to get a Pentium. What sort of improvement in performance can he expect?
jeff@canopy.biom.csiro.au
Reply: I compared a standard Genstat job on my 486 DX33 having 8Mb of memory with a Pentium machine with a large memory. It ran in about a quarter of the time. We estimated that the improvement was split about 50-50 in number-crunching and data access, but this is only a rough guess. Don't forget, however, that there are different speeds both of DXs and Pentia, as well as differences between SX and DX.
j.nelder@ma.ic.ac.uk
Postscript: A footnote to my previous posting on behalf of a scientist who was thinking of getting a Pentium machine. He now has a Pentium and is delighted with the performance of Genstat, even running under Windows. Thanks for the helpful comments.
jeff@canopy.biom.csiro.au
Rejoinder: Hopefully he has a bug-free version of the Pentium chip - if not, ring Intel for a replacement. Check to see whether your Pentium chip wears a fan or a heat-sink - they need to know when you order the replacement.
bairdd@agresearch.cri.nz

Continuing after a fault

Query: I want to perform an operation repeatedly within a loop. Each operation takes a long time, so I run the job overnight. I know that occasionally there will be a fault in the operation. I would like simply a report, and continue with the next pass of the loop. However, it crashes at the first fault. This means that to get any further I have to remove that case and run the job again the next night (when it crashes at the next fault, and so on.) The statement

```
SET [ERROR=0]
```

doesn't help. Any ideas?
graham@sass.sari.ac.uk
Reply: The only way to continue execution after a fault in batch mode is to set the **DIAGNOSTIC** option of the **JOB** or **SET** directive. If you give

```
SET [DIAGNOSTIC=*]
```

then Genstat will not report messages, warnings or faults, and, as a side effect, will continue execution after a fault. This facility is intended primarily for writers of procedures, but works outside as well. The **RUN** option of **SET** does not affect continuation after faults. The **ERRORS** option of **SET** controls the number of faults that are reported, and does not affect subsequent execution.
peter.lane@bbsrc.ac.uk

Rounding in calculations

Query: I'm looking at an experiment where each subject tastes a number of juices from a number of packs. I want to construct a variable indicating the juice/pack combinations for each subject. I tried to code the juice number for each pack in successive digits of a single number as follows

```
CALC code = juice*(10**pack)
```

but this doesn't seem to work. What am I doing wrong?
duncan.hedderley@bbsrc.ac.uk
Reply 1: I cannot see anything wrong with the code, but I suggest you try using **TABULATE** as an alternative.
mcnulty@hri.sari.ac.uk
Reply 2: The problem is caused by rounding errors. The calculation produces some very large and some very small values; consequently, adding them together depends on the number of significant places Genstat can cope with.
ian@sass.sari.ac.uk
Reply 3: The calculation is trying to do integer arithmetic in real numbers, and on most computers real number arithmetic is accurate only to seven digits.
rodger.white@bbsrc.ac.uk
Reply 4: I am not sure why you want the new variate, since if you wanted a compact way to see who got what you should be able to tabulate and get more informative output. On the other hand, if you wanted to see if any subjects had got the same sequence, maybe you could equate the table to eight variates and create a subject code, then sort on the eight variates and subject code, using all eight variates in the **INDEX** option of **SORT**.
p.baker@prospect.anprod.csiro.au

Generalized Procrustes

Query: I'm trying to interpret a GPA and wondering what the "Analysis of variation for the configuration" bit means. The first column (labelled "Scaling") is presumably the isotropic scaling factor, but what do the "Residual" and "Total" columns represent? Can I use them to judge if one or two assessors are not well represented by the consensus?
duncan.hedderley@bbsrc.ac.uk

Reply: The "Total" column is the total sum of squares for each of the configurations. If no external preliminary scaling nor isotropic scaling has been carried out then this may well represent large differences in overall "size". The "Residual" column is the residual sum of squares for each configuration after the Procrustes transformation. If you do have large size differences this is likely to be apparent also in the residual, but if these have been corrected for, then a large residual indicates a configuration that is not in agreement with the rest after the Procrustes transformations.
gillian.arnold@bbsrc.ac.uk

Labels for factors

Query: I have a list of about 50 names which have been given to clones in a breeding trial. I want to set up a factor to have levels corresponding to these names by reading the names from a file and then proceed to read the data file from my experiment where the actual names appear. Subsequently I will use this information in the ANOVA.
clarkep@cc.unp.uninet.za

Reply 1: One thing you might try (with Release 3.1) is just declaring a factor (no **LEVELS** or **LABELS** defined) and then reading the data file. Every new bit of text will be entered as representing a new level of the factor. So long as you enclose the names in single quotes, you can even have level names with odd characters (such as spaces) in them.
duncan.hedderley@bbsrc.ac.uk

Reply 2: In Release 3.1, you can define the labels of the factor from what occurred in the data:

```
OPEN 'data.dat'; CHANNEL=2
FACTOR clone
READ [CHANNEL=2] clone; FREP=labels
```

If you want to check the data against a master set of names, then read the master set first into a text:

```
OPEN 'master.dat','data.dat'; \
  CHANNEL=2,3
TEXT nclone
READ [CHANNEL=2] nclone
FACTOR [LABELS=nclone] clone
READ [CHANNEL=3] clone; FREP=labels
```

The clone names are allowed to be any quoted string, or any unquoted string that does not include a space, single or double quote, tab, colon, asterisk or backslash (Manual, Page 79). You may be able to avoid quoting complicated strings in the data file by using **READ** with a fixed format.
peter.lane@bbsrc.ac.uk

Premultipliers

Query: Can anyone explain why this works:

```
SCALAR n; VALUE=6
FACTOR [LEVELS=6; VALUES=4(1...n)] f
```

but this does not:

```
SCALAR n; VALUE=4
FACTOR [LEVELS=6; VALUES=n(1...6)] f
```

ccsphc@sunserver1.bath.ac.uk

Reply 1: The solution is that **#n** will work in both cases. Why **n** works in the first case but not the second, I cannot explain!
ian@sass.sari.ac.uk

Reply 2: There are two uses of **n** in the given example: as a list element in **1...n** and as a list-multiplier in **#n(1...6)**. The following simple example illustrates why the **#** is necessary. Suppose you have a list in an expression

```
CALC v[1...12] = 6(a,b)
```

then it would be nice to generalize it to

```
SCALAR n; VALUE=6
CALC v[1...12] = n(a,b)
```

But supposing the scalar was called **max**, not **n**:

```
CALC v[1...12] = max(a,b)
```

How does Genstat know whether you are using the scalar **max** or the function called **max**?
simon.harding@bbsrc.ac.uk

Formal levels of factors

Query: We used to find formal levels useful in earlier versions of Genstat. Can anyone suggest an elegant way of forming them from actual levels; for example, converting 1,3,78 to 1,2,3?
Fred.potter@bbsrc.ac.uk

Reply: Use the **NEWLEVELS** function:

```
FACTOR [LEVELS=!(1,3,78); VALUES=...]
f1
  FACTOR [LEVELS=3] f2
  CALC f2 = NEWLEV(f1; !(1...3))
```

ian@sass.sari.ac.uk and *fillmore@nsrske.agr.ca*

Negative binomial in GLM

Query: I am analysing some count data using a generalized linear model with a log link function and Poisson distribution. The residual deviance is much greater than 1. I now wish to try the negative binomial distribution. Has anyone experience of fitting such models using Genstat?
tony@sass.sari.ac.uk

Reply 1: A good (sometimes very good) approximation to the negative binomial within Genstat is to remember the relationship with the gamma distribution. If you specify that, together with a log link (not the canonical link for gamma), this seems to be very satisfactory in terms of a residual/fitted-values plot. You can call it a quasi-likelihood model, if you like. You will need to consider zero values quite carefully.
acadvm1.uottawa.ca

Reply 2: John Nelder has programmed model-fitting using the negative binomial. It is available in his K-system – a highly interactive environment within Genstat for fitting and checking GLMs with minimal typing, using procedures. It is available on the NAG gopher: www.nag.co.uk.
peter.lane@bbsrc.ac.uk

STOP PRESS: The negative binomial will be available as a standard option in Release 3.2. *Editor*

Diallel and NC designs

Query: Does anyone have any Genstat code to analyse the subject of Mather and Jinks, Chapter 8 – diallel crosses, North Carolina designs 1 and 2 – beyond that of getting the variance components?
ag144stat@ncccot.agr.ca

Reply: I have a Genstat procedure for analysing full and half diallels according to the methods of Hayan, Jinks and Jones. I haven't done NC designs yet.
Fred.potter@bbsrc.ac.uk

Rejoinder: Just to let you know that I have had some communication with Trevor Hohls in Natal, who has software for NC designs. I have placed your diallel code in my local library.
ag144stat@ncccot.agr.ca

Bootstrap

Query: Has anyone attempted to do any resampling statistics (bootstrap etc.) using Genstat? Are there any procedures or example programs around, please?
awm@pcmail.nerc-bas.ac.uk

Reply 1: I have written two procedures using Mantel's test of the significance of the correlation between distance matrices. They use randomization rather than his original test statistic. One is for the simple correlation between two matrices and the other for partialling out the effects of a third matrix. I also have two procedures for computing F-statistics by Weit and Cockerham's method, which are tested by bootstrap and jackknife. All procedures are rather slow!
lschmitt@anhb.uwa.edu.au

Reply 2: I have an *ad hoc* translation of the S function **BCANON** obtainable from the FTP site mentioned in the book by Hastie and Tibshirani. The procedure implements the bias-corrected accelerated bootstrap which is the one the authors seem to favour in many applications. I have not elaborated on niceties.
h.van.der.voet@glw.agro.nl

Reply 3: Roger Payne and I have completed two procedures for the 3[2] Library, one for jack-knifing and one for bootstrapping. To use them, you need to write a short procedure (copied from a template) to calculate whatever statistic(s) you want to estimate from a set of data vectors; then call **BOOTSTRAP** or **JACKKNIFE** to carry out the resampling. The Library will be distributed to sites with 3.1 of Genstat (and the support service!) by NAG in due course.
peter.lane@bbsrc.ac.uk

The K System

Notice: The K System for Release 3.1 is now available on the NAG Bulletin Board and can be downloaded by anyone who can reach the gopher:
www.nag.co.uk 70
I would appreciate comments on the process of getting the code, the comprehensibility of the documentation, and the usefulness of the system. The K system provides a highly interactive environment for fitting and checking GLMs, with minimal typing. Please try it and let me know what you think.
j.nelder@ma.ic.ac.uk

Positioning of graphs

Query: I am using Genstat to generate PostScript graphics. Is there anyone else who finds it irritating that Genstat will write only to a square area at the left of a landscape page or bottom of a portrait page? Although the facility to write to the whole page would be nice, all I really want is to be able to write to the middle of the page. Have I overlooked a command that will let me centre my output? Is it possible to alter the PostScript code to tell the printer where on the page I want the graph?
ian.wakeling@bbsrc.ac.uk

Reply 1: I also find this irritating. A few years ago the Cambridge Computing Service patched the source code of Genstat 2.2 to make plots fill the whole A4 page, but unfortunately it has not been ported to later versions.
tim.cole@mrc-dunn.cam.ac.uk

Reply 2: PostScript writes on a nominal x-y plane with 0,0 at the bottom left corner. EPS is an even better option because it assumes the picture will be embedded, so makes it self-contained. However, for both PS and EPS you will find a "comment" near the start of the graphics file

```
%%BoundingBox x1 y1 x2 y2
```

where x1-x2 and y1-y2 define the plotting area, in units of about 1/72 inch. Unfortunately, many packages that generate PS files get the bounding-box values wrong! This may not matter if you just print the file as it is, but it's a real bummer if you want to merge the graph inside some text. You can edit the BoundingBox line, aided by a file called **bbfig.ps** (available from most network archives) and the GhostScript program.
r.a.reese@ucc.hull.ac.uk

Reply 3: You are obviously not using Genstat 3.1 on Sun/SunOs which lets you set the **YUPPER** parameter of **FRAME** up to 1.4 when using Device 4 or 5. This was not possible with Release 2.2, but when I took it up with Rothamsted they told me how to alter the code. They have obviously incorporated the change into 3.1.
vdjal1@hermes.cambridge.ac.uk

Rejoinder: Thanks to everyone who replied. I had trusted the Manual when I read the bit about **FRAME** only being able to define a square area (page 303). In fact, my version of 3.1 on Vax/VMS can write PS or EPS graphs that cover the whole page by setting **YUPPER** (portrait) or **XUPPER** (landscape) to 1.41. Unfortunately, this facility does not extend to the other devices that are available, so it is not possible to preview the image on a screen using Genstat. However, the GhostScript program could to do this.
ian.wakeling@bbsrc.ac.uk

Need for coprocessor

Query: I would be grateful to anyone who could advise me on the following. I wish to load Genstat 3.1 on a Viglen 486SX:

(1) Am I right that Genstat does not run on this machine?
(2) Will it run if I add a 487SX coprocessor?
(3) Will it run without a coprocessor if I replace the 486SX CPU with a 486DX?
(4) Are there any other reasons for choosing one rather than the other?

djal1@central-unix-service.cambridge.ac.uk

Reply 1: (1) Yes and no. Genstat was compiled to run only on machines with a coprocessor, which a 386SX does not have. There is Public Domain or Shareware software that emulates a coprocessor, but at best you can expect a marked reduction in speed. (2) Yes. (3) Yes. The 486DX has a coprocessor built in. (4) No. The 487SX coprocessor should work out cheaper.
stephen@nag.co.uk

Reply 2: We have recently discovered a 387 emulator called Q387. It is much cheaper ($25) than a hardware upgrade and will prove satisfactory in many cases. It is distributed as shareware so you can try it before paying. We have tested it with Genstat and it appears to function correctly, at reasonable speed. A copy of Q387 has been placed on the NAG bulletin board (URL http://www.nag.co.uk:70) and Genstat users are encouraged to take copies. Of course, Q387 will be of use for any software requiring a coprocessor.
simon.harding@bbsrc.ac.uk

Rejoinder: Have you checked the price of coprocessors recently? My impression is that a 387 intended for use with a 386/40 processor will also work well with any 486SX. I'm told that the new price for a 387 in NZ is $25. I could not get a sale for my discarded 387 when I advertised it recently for $15!
john@maths.marc.cri.nz

Formulation of a model to describe the colour preference of insects

K Phelps, G Edmonds, S Finch
Horticulture Research International
Wellesbourne
Warwick CV35 9EF, UK

V Kostal
Institute of Entomology
Czech Academy of Science
Branisovska 31
370 05 Ceske Budejovice
Czech Republic

1. Introduction

In a sequence of behavioural studies, the question of interest was: do insects exhibit a consistent colour preference and, if so, can it be quantified? Colour preferences were ascertained using plant models and water traps coloured with paint of known reflectance properties. Colours were tested in pairs; each experiment involved presenting insects with pairs of differently coloured traps and counting the number of insects landing in each trap. Exact details of designs are beyond the scope of this article.

A simple example illustrates the model fitted, based on that proposed by Bradley and Terry (1952). Say we have three colours A,B,C which we test in pairs and 12 flies are trapped each time. If 3 and 9 flies respectively are caught when traps coloured A and B are used and similarly 4,8 flies from A v C we can infer the results from B v C. To do this it is natural to think in terms of the ratio of flies choosing one colour compared with another. Since the ratio of flies on A to flies on B was 1/3 and the ratio of flies on A to flies on C was 1/2 , in a test of B v C we would expect the ratio of flies on B to flies on C to be 3/2.

2. Formulating the Model as a GLM

If Θ_{ij} is the ratio of the number of flies landing on colour i to the number landing on colour j, then the corresponding ratio for colour k against colour l is $\Theta_{kl}=\Theta_{kj}/\Theta_{lj}$. In a test where traps 1 and 2 are coloured k and l respectively, if n flies are trapped r of which are in trap 1, Θ_{kl} is estimated by $r/(n-r)$. Hence, ignoring the error terms

$$\log(r/(n-r)) = \log(\Theta_{kj}) - \log(\Theta_{lj})$$

and the problem becomes a GLM with binomial errors and a logit link function. In the example there are two parameters to be estimated, Θ_{kj} and Θ_{lj}, but in general the parameters will be Θ_h, h=1 to $(c-1)$, where c is the number of colours. Θ_h will be the ratio of landings on colour h to landings on colour c, where the choice of c is arbitrary but conveniently represents a control colour, yellow, which is included in all experiments.

3. Fitting the model in Genstat

For each test we arbitrarily label each trap 1 or 2. This allows us to assign each test to a unit and form variates r and n as defined above. We generate variates corresponding to each colour parameter, Θ_h, which take the value 1 if trap 1 was colour h, -1 if trap 2 was colour h and 0 otherwise .

The fit of the model can be tested by the residual deviance and the parameters estimated are $\log(\Theta_h)$, h=1 to $(c-1)$. These parameters can be interpreted as the log of the ratio of landings on a trap of a given colour to landings on the control.

4. Example

The following program fits the model to a small subset of the data from an experiment where pairs of coloured objects were exposed for sixty minutes to cabbage root flies. Note that is not necessary to test all possible pairs.

```
FACT [NVAL=5 ; LAB=!T(Blue,Green,Orange,Yellow)] COLOUR[1,2]
VARI [NVAL=5] NUM[1,2]
READ [PRIN=D] COLOUR[1], NUM[1], COLOUR[2], NUM[2]
Blue          11    Green          42
Green         47    Yellow         65
Yellow        64    Orange         72
Blue          14    Yellow         71
Green         33    Orange         74
:
CALC  N = NUM[1] + NUM[2]
  &     H[1...3] = (COLOUR[1].EQ.(1...3)) - (COLOUR[2].EQ.(1...3))
MODEL [DIST=BIN] NUM[1] ; NBIN=N
TERMS H[1...3]
FIT [CONST=OMIT] H[1...3]
```

The parameter estimates were –1.688 for Blue, –0.43 for Green, +0.224 for Orange. Thus the estimated ratios of landing on Blue, Green, Orange to landings on Yellow were 0.18, 0.65,and 1.25 respectively. The residual deviance was 1.34 on 2df. The results from such a small sample of colours can be inferred directly from the data but the modelling procedure is extremely useful when there are many colours. It was interesting to note the underdispersion that occurred where flies could land on a coloured object and then take off again. In experiments where the flies were drowned in coloured water-traps, the residual mean deviances were very close to 1.

The model can be easily extended to include other factors such as sex of flies or background colour:

```
FIT [CONST=OMIT] SEX*BACKGROUND*H[]
```

References

Bradley R A and Terry M A (1952) Rank analysis of incomplete block designs I. *Biometrika* **39** 324-45.

Acknowledgements

We thank the Ministry of Agriculture, Fisheries and Food for supporting this work as part of Project HH1801JFV and the British Ecological Society for supplying the Research Travel Grant for V K.

A method for simplifying single and complete linkage dendrograms

C A Glasbey
Scottish Agricultural Statistics Service
JCMB, King's Buildings
Edinburgh EH9 3JZ , UK

1. Introduction

In many forms of clustering 'two basic ideas are involved: internal cohesion and external isolation' (Cormack, 1971). Most clustering techniques seek partitions which satisfy both conditions. However, two commonly used methods, single and complete linkage, simply concentrate on one. They both take as their starting point a symmetric matrix of similarities between all pairs of objects in a set, and they operate agglomeratively, to produce a hierarchy of partitions in a dendrogram. Initially, every object is placed in a separate cluster. At each subsequent step, two clusters are selected and pooled to form a new cluster, with this process continuing until all the objects are in a single cluster. In single linkage the two clusters are chosen so that the maximum similarity between any object in one cluster and any object in another cluster is made as small as possible in the new partition (i.e. minimax); in complete linkage the clusters are chosen so that the minimum similarity between any two objects in the same cluster is as large as possible (i.e. maximin). Therefore isolated clusters and compact clusters are sought respectively by the two methods. In contrast, Ward's (1963) method seeks both compact and isolated clusters. The sum of squares of similarities within clusters, a measure of cohesion, is maximized at each agglomeration. At the same time the sum of squares of similarities between clusters, a measure of isolation, is minimized because the two terms sum to a constant.

2. Review of single and complete linkage

The single linkage method has many attractive features. It can be computed very quickly and can therefore be used with large data sets (Sibson, 1973). Gower and Ross (1969) pointed out its connection with the spanning tree of minimum length (i.e. maximum similarity). Also, single linkage solutions are optimal in the sense that, for any specified number of clusters, no partition exists which has a smaller maximum similarity between objects in different clusters. Therefore no other method for optimizing the particular criterion of minimum separation need be considered. Further, single linkage clustering is a method known to satisfy a set of axioms specified by Jardine and Sibson (1968) and it meets all but one of the conditions given by Fisher and Van Ness (1971). However, it frequently produces diffuse clusters, a phenomenon known as 'chaining' (Lance and Williams, 1967). This is not surprising as the method takes no account of the size of similarities within clusters.

Complete linkage clustering is, in a sense, the dual of single linkage. An efficient algorithm exists, similar to Sibson's, due to Defays (1977). However the agglomerative procedure is not necessarily optimal: other partitions may exist which contain the same number of clusters but have a larger minimum similarity within a cluster. The criterion of maximizing the minimum similarity within a cluster may be used divisively, to produce another hierarchy of partitions. Initially, all objects are placed in the same cluster. At each subsequent step, one cluster is selected to split into two, with this process continuing until all the objects are in separate clusters. The cluster chosen for splitting is the one containing the minimum similarity between two objects in the same cluster. The spanning tree of maximum length is constructed for the objects in this cluster, using a method analogous to the one for constructing the minimum spanning tree. Then, one arbitrarily chosen object in the cluster is placed in one of the two new clusters. All objects adjacent to this one in the spanning tree are placed in the other new cluster. All objects adjacent to these ones are placed in the first cluster, and so on until all objects have been allocated to one or other of the new clusters. By this method, no two objects in the same new cluster are adjacent in the spanning tree. This algorithm was proposed by Rao (1971). Again, the partitions may not be optimal.

The complete linkage partitions which, for each different number of clusters, have the largest minimum similarity between two objects in the same cluster are not necessarily nested; that is, they cannot be formed from one another by either a sequence of agglomerations or divisions. Baker and Hubert (1976) showed the connection between complete linkage clustering and the graph colouring problem, where linked points in a graph have to

be coloured differently using the minimum number of colours. In essence, if a graph is constructed by joining all objects which are less similar than a certain value and this graph is coloured, then a partition formed from the colouring will have a minimum internal similarity which is greater than this specified value. Hansen and Delattre (1978) gave an algorithm which produces optimal partitions for fixed numbers of clusters. Although the clusters which are formed are compact they are sometimes close together and find a dissection of compact groups of points.

3. Synthesized criterion

In order to retain the benefits of single and complete linkage, whilst obtaining clusters which are both compact and well separated, I proposed a synthesized criterion (Glasbey, 1980). Partitions are found which minimize $f(b,w)$, where f is a function of the maximum similiarity (b) between objects in different clusters and the minimum similarity (w) within clusters. Any function can be chosen, subject to the restriction that f decreases both as w increases for fixed b and as b decreases for fixed w. I showed that the criterion could be used in hierarchical clustering although, as with complete linkage, the results are not necessarily optimal. I also gave an algorithm for relocating points between clusters in order to find a locally-optimal partition for a fixed number of clusters. In a subsequent paper (Glasbey, 1987) I showed that all partitions which optimize the synthesized criterion are partitions in the single linkage dendrogram. In effect, complete linkage is used to provide a multiple stopping rule for single linkage clustering.

In this article I show how measures b and w can be combined graphically in Genstat, and the results used to simplify both the single linkage and complete linkage dendrograms.

4. Illustrative data

To illustrate the approach, I have used amino acid sequences for the protein 'cytochrome c' for twenty four animal species (Dayhoff, 1972). Sequences are between 100 and 120 in length, with missing values inserted where necessary to ensure correct alignment. McNicol *et al* (1993) derived similarities between species as the proportion of sequence positions with matching amino acids. They also produced displays like those shown in Figures 1-3.

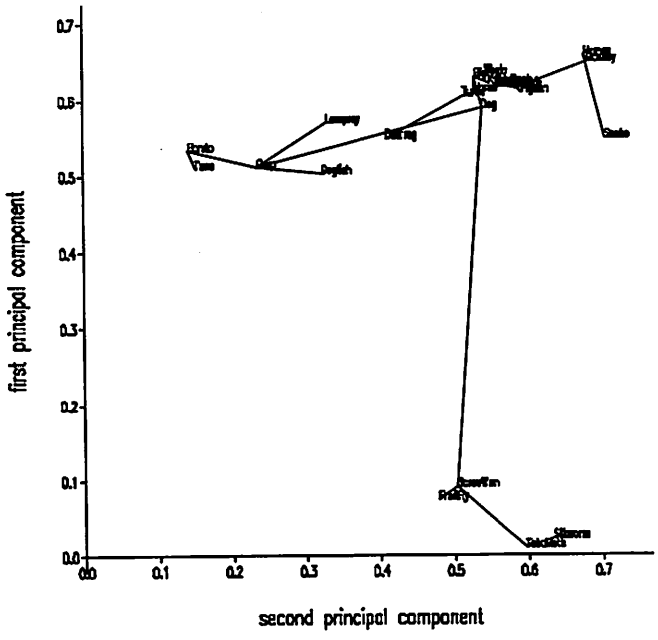


Figure 1

Figure 1 shows a plot of the first two principal coordinates obtained from the similarity matrix, together with the minimum spanning tree, produced by the procedure **DMST**.

Figure 2 shows the single linkage dendrogram output by the procedure **DDENDROGRAM** based on the tree produced by the Genstat command **HDISPLAY**. Figure 3 shows the complete linkage dendrogram from the Genstat command **HCLUSTER** and the procedure **DDENDROGRAM**.

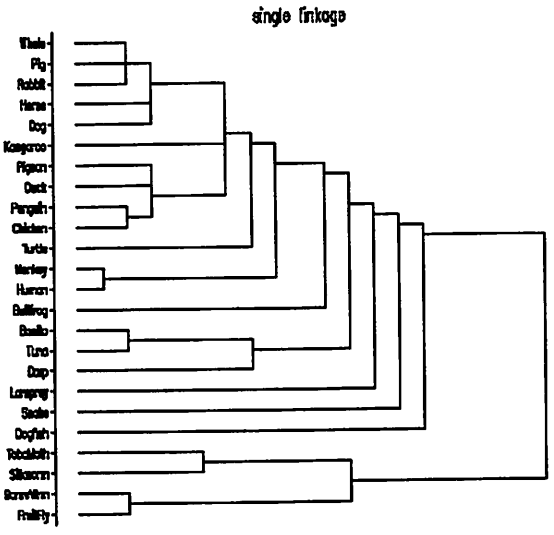


Figure 2

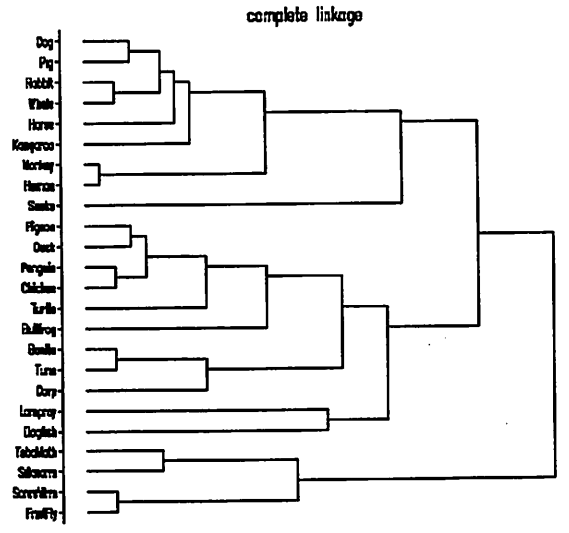


Figure 3

There are similarities between the dendrograms, such as the four insect species forming a separate cluster which is only amalgamated with the rest at the final level of aggregation. However, complete linkage groups all the mammals whereas single linkage includes birds and turtles as well.

5. Simplified single linkage dendrogram

Figure 4 shows minimum within-group similarity plotted against maximum between-group similarity for the partitions in the single linkage dendrogram.

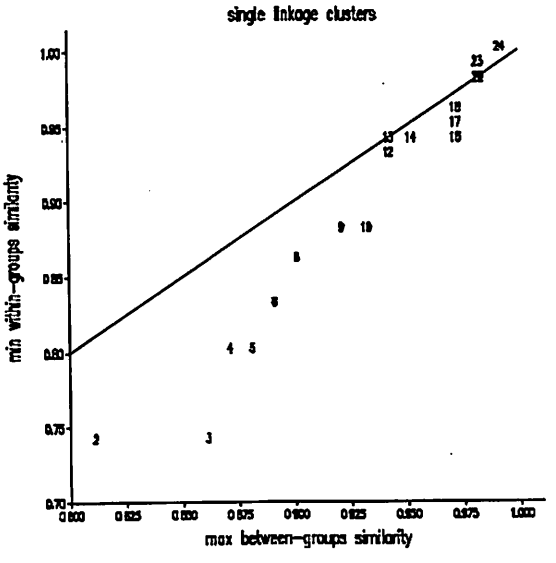


Figure 4

The Genstat code for obtaining a lineprinter version of this graph, given an **Ndata** by **Ndata** similarity matrix **Simmat** is as follows:

```
HDISPLAY [PRINT=*] Simmat; TREE=Mstree
CALCULATE Mstt = TRANSPOSE(Mstree)
EQUATE Mstt; !P(Clus2,Between)
FACTOR [LEVELS=Ndata; VALUES=Ndata...1] Nclus
VARIATE [VALUES=1...Ndata] Clus1,Within,Labels,Dummy
SORT [DIRECTION=descending] Between,Clus1,Clus2
CALCULATE Between = CIRCULATE(Between;-1)

CALCULATE Minsim=1
CALCULATE Index=1
CALCULATE Ndataz=Ndata-1
FOR [NTIMES=Ndataz]
  CALCULATE Index=Index+1
  CALCULATE Lab1 = ELEMENTS(Labels;ELEMENTS(Clus1;Index))
  CALCULATE Lab2 = ELEMENTS(Labels;ELEMENTS(Clus2;Index))
  CALCULATE Veclab1 = MIN(Lab1)*(Dummy>0)
  CALCULATE Veclab2 = MIN(Lab2)*(Dummy>0)
  RESTRICT Dummy; CONDITION=(Labels==Veclab1); SAVESET=Locs1
  RESTRICT Dummy; CONDITION=(Labels==Veclab2); SAVESET=Locs2
  RESTRICT Dummy
  CALCULATE Minnew = MIN(ELEMENTS(Simmat; Locs1; Locs2))
  IF Minsim>Minnew
    CALCULATE Minsim=Minnew
  ENDIF
  CALCULATE ELEMENTS(Within;Index) = Minsim
  CALCULATE Veclab2a = MIN(Lab2)*(Locs1>0)
  CALCULATE ELEMENTS(Labels;Locs1) = Veclab2a
  DELETE [REDEFINE=yes] Locs1,Locs2,Veclab2a
ENDFOR

GRAPH [YTITLE='minimum within-groups similarity';  \
  XTITLE='maximum between-groups similarity'] Y=Within; X=Between; SYMBOLS=Nclus
```

The first block of commands reformats the matrix **Mstree** output by **HDISPLAY** so that amalgamations are in order. The maximum similarities between clusters is stored in the variate **Between**. In the second block of commands, the minimum similarity within each newly formed partition of clusters is obtained by extracting submatrices from the similarity matrix, calculating their minimum value (**Minnew**) and comparing this with the current minimum, **Minsim**. This is stored as an element in the variate **Within**. In Figure 4 we look for partitions which have high within-group similarity and low between-group similarity, i.e. towards the top left corner of the figure. Ideally, partitions should lie above the 1:1 line which is included in Figure 4, because then all similarities within clusters exceed all similarities between objects in different clusters.

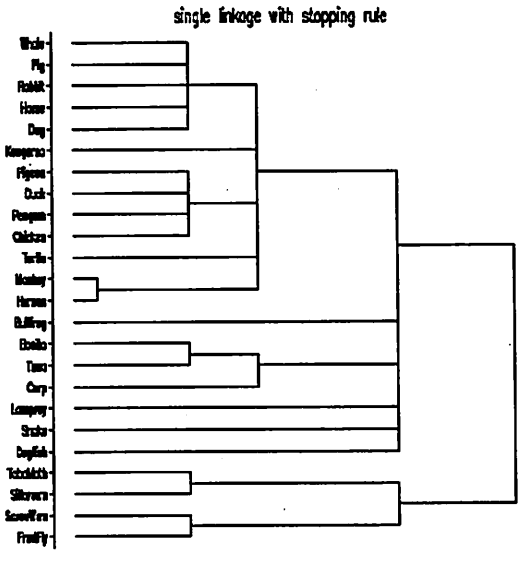


Figure 5

Partitions of size 2, 8, 13 and 23 can be identified in this figure as having better combined measures of compactness and separation than other nearby partitions. Figure 5 gives the simplified single linkage dendrogram, consisting only of these partitions. This is achieved by modifying the matrix **Mstree** as follows:

```
VARIATE [VALUES=23,13,8,2,1] Select
CALCULATE Index=0
CALCULATE Nselect = NVALUES(Select)
FOR [NTIMES=Nselect]
   CALCULATE Index=Index+1
   CALCULATE  Threshold = ELEMENTS(Between; Ndata - ELEMENTS(Select; Index))
   CALCULATE Index2=1
   FOR [NTIMES=Ndataz]
     CALCULATE Index2=Index2+1
     IF ELEMENTS(Mstree; Index2; 2) > Threshold - 0.0001
       CALCULATE ELEMENTS(Mstree; Index2; 2) = -Threshold
```

```
      ENDIF
    ENDFOR
ENDFOR
CALCULATE Index=1
FOR [NTIMES=Ndataz]
  CALCULATE Index=Index+1
  CALCULATE ELEMENTS(Mstree; Index; 2) = -ELEMENTS(Mstree; Index; 2)
ENDFOR

DDENDROGRAM [STYLE=centroid; REVERSE=yes; GRAPHICS=lineprinter] Mstree; \
  TITLE='single linkage with stopping rule'; LABELS=Specnam
```

The grouping into eight clusters is:

- Whale, Pig, Rabbit, Horse, Dog, Kangaroo, Pigeon, Duck, Penguin, Chicken, Turtle, Monkey, Human
- Bonito, Tuna, Carp
- Tobacco horn-worm Moth, Silk worm
- Screw worm, Fruit fly

and the other species (Bullfrog, Lamprey, Snake, Dogfish) form single clusters.

6. Simplified complete linkage dendrogram

It is also possible to reverse the procedure, and plot minimum within-group similarity plotted against maximum between-group similarity for the partitions in the complete linkage dendrogram, although this has less theoretical justification. Figures 6 and 7 show results analogous to the above.

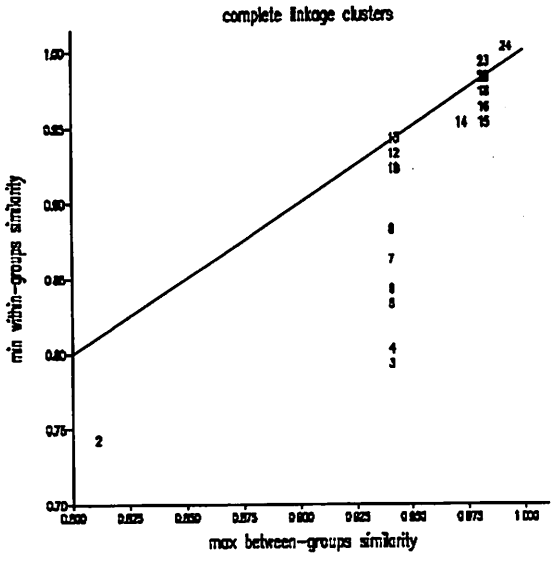


Figure 6

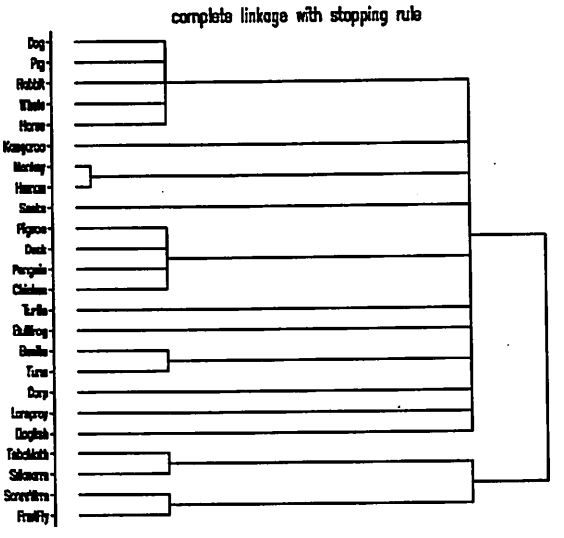


Figure 7

The Genstat code is similar, but now has to use the matrix array `Clink` output by `HCLUSTER`.

```
HCLUSTER [METHOD=complete] Simmat; AMALGAMATIONS=Clink
VARIATE [Ndataz] Clus1,Clus2,Between,Within
CALCULATE Clinkt = TRANSPOSE(Clink)
EQUATE Clinkt; !P(Clus1,Clus2,Within)
FACTOR [LEVELS=Ndata; VALUES=Ndata...2] Nclus
VARIATE [VALUES=1...Ndata] Labels,Dummy
CALCULATE Within = CIRCULATE(Within;1)
CALCULATE ELEMENTS(Within;1) = 1

CALCULATE Index=0
FOR [NTIMES=Ndata]
```

```
  CALCULATE Index=Index+1
  CALCULATE  ELEMENTS(Simmat; Index; Index) = -ELEMENTS(Simmat; Index; Index)
ENDFOR
CALCULATE Index=0
FOR [NTIMES=Ndataz]
  CALCULATE Index=Index+1
  CALCULATE ELEMENTS(Between;Index) = MAX(Simmat)
  CALCULATE Lab1 = ELEMENTS(Labels;ELEMENTS(Clus1;Index))
  CALCULATE Lab2 = ELEMENTS(Labels;ELEMENTS(Clus2;Index))
  CALCULATE Veclab1 = MIN(Lab1)*(Dummy>0)
  CALCULATE Veclab2 = MIN(Lab2)*(Dummy>0)
  RESTRICT Dummy; CONDITION=(Labels==Veclab1); SAVESET=Locs1
  RESTRICT Dummy; CONDITION=(Labels==Veclab2); SAVESET=Locs2
  RESTRICT Dummy
  CALCULATE ELEMENTS(Simmat; Locs1; Locs2) = -ELEMENTS(Simmat; Locs1; Locs2)
  CALCULATE Veclab2a = MIN(Lab2)*(Locs1>0)
  CALCULATE ELEMENTS(Labels;Locs1) = Veclab2a
  DELETE [REDEFINE=yes] Locs1,Locs2,Veclab2a
ENDFOR

GRAPH [YTITLE='minimum within-groups similarity';  \
  XTITLE='maximum between-groups similarity'] Y=Within; X=Between; SYMBOLS=Nclus

VARIATE [VALUES=23,13,2] Select
CALCULATE Index=0
CALCULATE Nselect=NVALUES(Select)
FOR [NTIMES=Nselect]
  CALCULATE Index=Index+1
  CALCULATE  Threshold = ELEMENTS(Within; Ndata +*1 - ELEMENTS(Select; Index))
  CALCULATE Index2=0
  FOR [NTIMES=Ndataz]
    CALCULATE Index2=Index2+1
    IF ELEMENTS(Clink; Index2; 3) > Threshold - 0.0001
      CALCULATE ELEMENTS(Clink; Index2; 3) = -Threshold
    ENDIF
  ENDFOR
ENDFOR
CALCULATE Index=0
FOR [NTIMES=Ndataz]
  CALCULATE Index=Index+1
  CALCULATE ELEMENTS(Clink; Index; 3) = -ELEMENTS(Clink; Index; 3)
ENDFOR
CALCULATE ELEMENTS(Clink; Ndataz; 3) = -ELEMENTS(Clink; Ndataz; 3)

DDENDROGRAM [STYLE=centroid; REVERSE=yes; GRAPHICS=lineprinter] Clink; \
  TITLE='complete linkage with stopping rule'; LABELS=Specnam
```

The same partition into thirteen clusters is identified. This partition is the largest grouping for which all species within a cluster are at least as similar to each other as they are to any species in other clusters. The grouping is:

- Dog, Pig, Rabbit, Whale, Horse
- Pigeon, Duck, Penguin, Chicken
- Monkey, Human
- Bonito, Tuna
- Tobacco horn-worm moth, Silk worm
- Screw worm, Fruit fly

and the other species (Kangaroo, Snake, Turtle, Bullfrog, Carp, Lamprey, Dogfish) form single clusters. This partition shows good agreement with the spatial distribution of species in the principal coordinates plot in Figure 1.

Acknowledgements

The work was supported by funds from the Scottish Office Agriculture and Fisheries Department.

References

Baker F B and Hubert L J (1976) A graph-theoretic approach to goodness-of-fit in complete-link hierarchical clustering. *Journal of the American Statistical Association* **71** 870-878.

Cormack R M (1971) A review of classification. *Journal of the Royal Statistical Society Series A* **134** 321-367.

Dayhoff M O (ed) (1972) Atlas of Protein Sequence and Structure. National Biomedical Research Foundation.

Defays D (1977) An efficient algorithm for a complete link method. *The Computer Journal* **20** 364-366.

Fisher L and Van Ness J W (1971) Admissible clustering procedures. *Biometrika* **58** 91-104.

Glasbey C A (1980) A synthesis of single linkage and complete linkage clustering criteria. In COMPSTAT 1980, 339-404, (Eds. M M Barritt and D Wishart) Vienna: Physica-Verlag.

Glasbey C A (1987) Complete linkage as a multiple stopping rule for single linkage clustering. *Journal of Classification* **4** 103-109.

Gower J C and Ross G J S (1969) Minimum spanning trees and single linkage cluster analysis. *Applied Statistics* **18** 54-64.

Hansen P and Delattre M (1978) Complete-link cluster analysis by graph colouring. *Journal of the American Statistical Association* **73** 397-403.

Jardine N and Sibson R (1968) The construction of hierarchic and non-hierarchic classifications. *Computer Journal* **11** 177-184.

Lance G N and Williams W T (1967) A general theory of classificatory sorting strategies. I. Hierarchical systems. *Computer Journal* **9** 373-380.

McNicol J W, Hirst D J and Kempton R A (1993) Graphical methods for multivariate data. Version 2. SASS Teaching Course Notes , Edinburgh.

Rao M R (1971) Cluster analysis and mathematical programming. *Journal of the American Statistical Association* **66** 622-626.

Sibson R (1973) SLINK: an optimally efficient algorithm for the single-link cluster method. *Computer Journal* **16** 30-34.

Ward J H (1963) Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association* **58** 236-244.

Fitting the negative binomial distribution

D A Preece
Institute of Mathematics and Statistics
Cornwallis Building, The University
Canterbury, Kent CT2 7NF, UK

G J S Ross
Statistics Department
IACR–Rothamsted
Harpenden, Herts AL5 2JQ, UK

1. Introduction

In a detailed paper on the negative binomial distribution, Ross and Preece (1985) discussed use of the Maximum Likelihood Program MLP (Ross, 1980) for fitting the distribution to data. As the relevant section of MLP has subsequently been incorporated in Genstat 5 Release 3, parts of Ross and Preece's paper are now offered in a rewritten form for Genstat users, particularly for the many students who find textbook accounts of the negative binomial distribution to be bewildering.

The illustrative examples in the present note are (a) the two biological examples discussed by Ross and Preece (1985) and taken from, respectively, Bliss (1953) and Fisher (1941), and (b) the actuarial example given by Currie (1993, pp. 31-33). Indeed, one of the aims of this note is to indicate that very simple use of Genstat could enhance teaching of the Actuarial Statistics component of the examination syllabuses of the Institute of Actuaries. For the Genstat analyses described in this note, only six Genstat directives are essential, namely **FACTOR, TABLE, READ, PRINT, DISTRIBUTION** and **STOP**.

2. Notation

Genstat has the negative binomial distribution set up with

$$\binom{r+k-1}{k-1}\left(\frac{m}{m+k}\right)^r\left(1+\frac{m}{k}\right)^{-k}$$

as the probability of the random variable taking the value r ($r = 0,1,2,...$). This is the formulation of, for example, Anscombe (1949). Here, the parameters m (the mean) and k are both positive, and neither is necessarily an integer. When k is not an integer, the above formula is taken to represent

$$\frac{(r+k-1)(r+k-2)\ \ldots\ (k+1)k}{r!}\left(\frac{m}{m+k}\right)^r\left(1+\frac{m}{k}\right)^{-k}$$

If we use x instead of r for a value of the random variable, and write

$$p = k/(m+k)\ ,\quad q = 1-p\ ,$$

then the formula becomes

$$\frac{(x+k-1)(x+k-2)\ \ldots\ (k+1)k}{x!}\,p^k q^x\ .$$

This is the notation used by the Institute of Actuaries (1980) for "subject 5". With k rewritten as α, we have the notation of Currie (1993, pp. 30-31). As pointed out by Ross and Preece (1985, p. 325), various authors have used p as just defined, whereas others, including Fisher (1941), have used $p = m/k$. As with other distributions in the examination syllabuses of the Institute of Actuaries, work on a negative binomial distribution must always include careful checking of how the distribution's parameters have been defined.

3. Sample statistics

In Genstat 5 Release 3, standard discrete and continuous distributions are fitted by use of the directive **DISTRIBUTION** (Genstat 5 Committee, 1993, pp. 330-343). For discrete distributions, this directive first causes the printing of various Sample Statistics for the data values x_i ($i = 1, 2, \ldots, n$), namely the mean m_s, variance s^2, Skewness m_3 / s^3, and the two "scale-free" indices:

Poisson Index	$(s^2 - m_s) / m_s^2$
Negative Binomial Index	$m_s(m_3 - 3s^2 + 2m_s) / (s^2 - m_s)^2$

where m_3 is the sample's third moment about the sample mean.

The Poisson Index and the Negative Binomial Index were devised by G J S Ross and are not to be found in standard textbooks. The Poisson Index was defined because of the difficulty of using the sample mean and sample variance alone to judge intuitively whether the Poisson distribution is likely to provide an adequate fit to an observed discrete distribution. The sample distribution of the Poisson Index is very similar for various common two-parameter discrete distributions; it behaves better than the sample distributions of the standard defining parameters for the two-parameter distributions; it is approximately Normal for moderate sample sizes.

The Negative Binomial Index was devised as a discriminator of long- and short-tailed distributions. It allows interpretation of the third moment in terms of the first two moments, and so is a sort of "discrete coefficient of skewness". As the numerator of the Poisson Index occurs in the denominator of the Negative Binomial Index, the latter index is very unreliable for data with a small Poisson Index; roughly speaking, the Negative Binomial Index is of value only if the Poisson Index is greater than, say, 0.5. Nor should the Negative Binomial Index be used if the sample mean or sample size is small; the sample size has to be at least 200 for the probability of encountering seemingly discrepant values of the Negative Binomial Index to be small.

For the negative binomial distribution, the theoretical values of the Sample Statistics printed by Genstat are, in the notations of Genstat and the Institute of Actuaries,

		Genstat		Institute of Actuaries
Mean	=	m	=	kq / p,
Variance	=	$m\,(1 + m/k)$	=	kq / p^2,
Skewness	=	$(1 + 2m/k) / \sqrt{(m(1+m/k))}$	=	$(2/p - 1) / \sqrt{\{kq / p^2\}}$,

Poisson Index = $1 / k$, Negative Binomial Index = 2.

For sample data, the value of the Negative Binomial Index can (as implied above) differ greatly from 2 even when the data are well fitted by a negative binomial distribution.

If the sample data have been grouped into classes, e.g. with all values 5, 6 and 7 of the variate put into a single class, and all values 8, 9, 10 and 11 put into the next class, then the Sample Statistics are calculated using mid-points of classes. For the final class covering the right-hand tail of the distribution, the notional mid-point is taken to be 1.25 times the smallest value in the class; thus if the tail is for the values 12, 13, 14, ... , the contribution of the tail is assumed to be equivalent to concentrating the tail-observations at the value 15; this may of course lead to gross underestimation of the variance and third moment, especially for long-tailed distributions. No corrections are made to the calculated Sample Statistics to compensate for the grouping.

4. How Genstat fits the negative binomial distribution

Genstat uses the maximum likelihood method to fit the negative binomial distribution. The algorithm first estimates the two 'working parameters' for the distribution, these being the mean m and variance $m(1 + m/k)$, which are chosen as they are stable functions of the 'defining parameters' m and k. Estimates of the defining parameters are then obtained from the estimates of the working parameters.

Standard errors (s.e.s) are printed for the estimates of the parameters and are the usual asymptotic approximations obtainable from maximum likelihood theory. Although the estimates of m and k are independent, the estimated s.e. for k is not reliable, as a confidence interval for k is skew (i.e. the point estimate of k does not lie at the centre of the interval).

5. Example 1 (Bliss, 1953)

The data of Bliss (1953) were obtained from apple trees. Twenty five leaves were selected at random from each of six similar trees in an orchard, and the adult female red mites on each of the leaves were counted:

No. of mites per leaf	0	1	2	3	4	5	6	7	>7	Total
No. of leaves (frequency)	70	38	17	10	9	3	2	1	0	150

For a Genstat run, we **READ** the frequencies

70 38 17 10 9 3 2 1 0

into a one-way **TABLE** that is classified by a nine-level **FACTOR** whose first eight **LEVELS** are 0, 1 , ... , 7 for the non-zero frequencies. The ninth level is for the zero frequency in the tail of the distribution. As the numerical value of this level is not used in any of Genstat's calculations, any convenient value can be chosen, e.g. 8, or 99, or 1000, according to taste; we here choose 8, for ease of coding the input. The corresponding output (incorporating the input, except for the data for the **READ** directive) is as follows.

```
Genstat 5  Release 3.1  (Vax/VMS5)              13-OCT-1994 15:34:47.02
Copyright 1994, Lawes Agricultural Trust (IACR-Rothamsted)

   1  JOB 'NEGATIVE BINOMIAL'
   2  OUTPUT [WIDTH=76] 1
   3
   4  "Analyses of data of Bliss (1953)"
  -5
   6
   7  "Analysis without any grouping, and with zero frequency for tail."
   8  FACTOR [LEVELS=!(0...8)] Mites; DECIMALS=0
   9  TABLE [CLASSIFICATION=Mites] Leaves; DECIMALS=0
  10  READ [PRINT=*] Leaves
  12  PRINT [ACROSS=Mites] Leaves; FIELDWIDTH=6

            Leaves
Mites        0      1      2      3      4      5      6      7      8
            70     38     17     10      9      3      2      1      0

  13  DISTRIBUTION [DISTRIBUTION=negativebinomial] Leaves

13.....................................................

***** Fit discrete distribution *****

*** Sample Statistics ***

     Sample Size          150
     Mean                1.15    Variance                    2.26
     Skewness            1.53
     Poisson Index       0.85    Negative Binomial Index     0.66

*** Summary of analysis ***

Observations:    Leaves
                 Parameter estimates from tabulated data values
Distribution:    Negative Binomial
                 Pr(X=r) = (r+k-1)C(k-1).(m/(m+k))**r.(1+m/k)**(-k)
     Deviance:   4.22 on 6 d.f.
```

```
*** Estimates of working parameters ***

            estimate        s.e.        Correlations
mean          1.1467        0.1273      1.0000
variance      2.4301        0.5379      0.7663  1.0000

*** Estimates of defining parameters ***

            estimate        s.e.        Correlations
m             1.1467        0.1273      1.0000
k             1.0246        0.2758      0.0001  1.0000
1/k           0.9760        0.2628      Poisson Index

*** Fitted values (expected frequencies) and residuals ***

        r           Number          Number          Weighted
                    Observed        Expected        Residual
        0             70              69.49           0.06
        1             38              37.60           0.07
        2             17              20.10          -0.71
        3             10              10.70          -0.22
        4              9               5.69           1.28
        5              3               3.02          -0.01
        6              2               1.60           0.30
        7              1               0.85           0.16
        8+             0               0.95          -1.38
```

Here the sample variance, 2.26, is roughly twice the sample mean, 1.15, indicating that a Poisson distribution would not have fitted very well; this fact is expressed in the Poisson Index, whose value of 0.85 indicates that the moment estimator of k (i.e. the estimator obtained from the sample moments) is 1 / 0.85 = 1.18. The value 0.66 of the Negative Binomial Index is smallish for a negative binomial distribution, indicating that a better fit might have been obtained from some other two-parameter discrete distribution. However, the value 4.22 of the residual deviance indicates a good fit, the number of degrees of freedom (d.f.) for this deviance being

```
   9    the number of classes
 - 1    for the constraint that the fitted class probabilities must sum to 1
 - 2    for the 2 fitted parameters
 = 6
```

For an approximate test of goodness-of-fit, the residual deviance can be compared with tabulated critical chi-squared values for 6 d.f. The correlation 0.0001 between the estimates of m and k is very close to its theoretical value of zero.

The above Genstat output shows that the expected frequency is less than 1 for each of the top two classes, and is less than 5 for each of the top four classes. This raises the question of whether there should have been some grouping of classes, as recommended in many textbooks. This matter was discussed by Ross and Preece (1985), who concluded:

> There are no hard and fast rules about when to group. It is preferable to have some values in the tail, and not too many cells with small frequencies.

Ross and Preece made no attempt to define 'too many' here, but they illustrated the differences of outcome that can arise from different groupings by producing two further analyses of the Bliss data. Firstly they merely assigned all leaves with 5 or more mites to a class for the tail of the distribution. Genstat output for this variant of the analysis is as follows; as the value 5 would now be misleading for the sixth level of the **FACTOR** for the **TABLE** of frequencies, the arbitrary value 100 has been chosen instead.

```
14
15  "Analysis with a tail frequency of 6."
16  FACTOR [LEVELS=!(0...4,100)] Mites1; DECIMALS=0
17  TABLE [CLASSIFICATION=Mites1] Leaves1; DECIMALS=0
18  READ [PRINT=*] Leaves1
20  PRINT [ACROSS=Mites1] Leaves1; FIELDWIDTH=7
 Leaves1
```

```
Mites1        0         1         2         3         4       100
             70        38        17        10         9         6

  21  DISTRIBUTION [DISTRIBUTION=negativebinomial] Leaves1

21.....................................................................

***** Fit discrete distribution *****

*** Sample Statistics ***

  Sample Size      150
  Mean             1.17   Variance                    2.46
  Skewness         1.64
  Poisson Index    0.94   Negative Binomial Index     0.90

*** Summary of analysis ***

Observations:  Leaves1
               Parameter estimates from tabulated data values
Distribution:  Negative Binomial
               Pr(X=r) = (r+k-1)C(k-1).(m/(m+k))**r.(1+m/k)**(-k)
Deviance:      2.13 on 3 d.f.

*** Estimates of working parameters ***

             estimate   s.e.       Correlations
mean         1.1686     0.1350     1.0000
variance     2.6073     0.6328     0.7851  1.0000

*** Estimates of defining parameters ***

             estimate   s.e.       Correlations
m            1.1686     0.1350     1.0000
k            0.9492     0.2593    -0.0745    1.0000
1/k          1.0535     0.2878     Poisson Index

*** Fitted values (expected frequencies) and residuals ***

  r          Number     Number     Weighted
             Observed   Expected   Residual
  0             70       70.03       0.00
  1             38       36.68       0.22
  2             17       19.72      -0.63
  3             10       10.70      -0.22
  4              9        5.83       1.22
  5+             6        7.04      -0.40
```

In this output (unlike in the corresponding output given by Ross and Preece for MLP), the sample statistics differ from those obtained previously, as they have now been calculated as described above for grouped data. The umber of d.f. for the residual deviance is, of course, now reduced to 3. Also, because of the grouping, the correlation between the estimates of m and k is no longer theoretically zero, and is in fact –0.0745.

In their final analysis of the Bliss data, Ross and Preece (1985) assigned the classes for 5, 6 and 7 mites per leaf to a group of their own, leaving a tail class with a zero frequency. For Genstat, the factor level for the class for 5, 6 and 7 mites per leaf must be the **largest** incorporated number of mites per leaf, namely 7. Genstat output for this variant of the analysis is as follows.

```
  22
  23    "Analysis with a group before the tail, and with zero
        frequency for tail."
  24  FACTOR [LEVELS=!(0...4,7,100)] Mites2; DECIMALS=0
  25  TABLE [CLASSIFICATION=Mites2] Leaves2; DECIMALS=0
  26  READ [PRINT=*] Leaves2
  28  PRINT [ACROSS=Mites2] Leaves2; FIELDWIDTH=7

      Leaves2
Mites2        0       1       2       3       4       7       100
             70      38      17      10       9       6         0
```

```
  29  DISTRIBUTION [DISTRIBUTION=negativebinomial] Leaves2

29.....

***** Fit discrete distribution *****

*** Sample Statistics ***

  Sample Size      150
  Mean             1.16   Variance                    2.36
  Skewness         1.56
  Poisson Index    0.89   Negative Binomial Index     0.72


*** Summary of analysis ***

Observations:  Leaves2
               Parameter estimates from tabulated data values
Distribution:  Negative Binomial
               Pr(X=r) = (r+k-1)C(k-1).(m/(m+k))**r.(1+m/k)**(-k)
  Deviance:    4.16 on 4 d.f.


*** Estimates of working parameters ***

             estimate   s.e.       Correlations
mean         1.1441     0.1275     1.0000
variance     2.4103     0.5415     0.7670  1.0000

*** Estimates of defining parameters ***

             estimate   s.e.       Correlations
m            1.1441     0.1273     1.0000
k            1.0336     0.2833     -0.0148    1.0000
1/k          0.9675     0.2651     Poisson Index


*** Fitted values (expected frequencies) and residuals ***

  r          Number     Number     Weighted
             Observed   Expected   Residual
  0            70         69.43      0.07
  1            38         37.70      0.05
  2            17         20.14     -0.72
  3            10         10.70     -0.22
  4             9          5.67      1.29
  5-7           6          5.42      0.25
  8+            0          0.93     -1.37
```

This last analysis almost meets the 'working rule' of Cochran (1954) that grouping for a unimodal distribution should be such that the expectation for a tail is at least 1.

6. Example 2 (Fisher, 1941)

The frequency table of Fisher (1941) was obtained from counting ticks on each of 82 sheep. For this example, Fisher's own estimate of k can be obtained from ungrouped data, with a zero frequency for the tail of the distribution. Genstat output for this analysis is as follows.

```
  30
  31  "Analyses of data of Fisher (1941)
 -32   ---------------------------------"
  33
  34  "Analysis without any grouping, and with zero frequency for tail."
  35  FACTOR [LEVELS=!(0...26)] Ticks; DECIMALS=0
  36  TABLE [CLASSIFICATION=Ticks] Sheep; DECIMALS=0
  37  READ [PRINT=*] Sheep
  41  PRINT [ACROSS=Ticks] Sheep; FIELDWIDTH=6
```

```
      Sheep
Ticks     0     1     2     3     4     5     6     7     8     9    10
          4     5    11    10     9    11     3     5     3     2     2

Ticks    11    12    13    14    15    16    17    18    19    20    21
          5     0     2     2     1     1     0     0     1     0     1

Ticks    22    23    24    25    26
          1     1     0     2     0
```

```
  42  DISTRIBUTION [DISTRIBUTION=negativebinomial] Sheep

42.....................................................................

***** Fit discrete distribution *****

*** Sample Statistics ***

  Sample Size          82
  Mean               6.56      Variance                        34.34
  Skewness           1.53
  Poisson Index      0.65      Negative Binomial Index          1.86

*** Summary of analysis ***

Observations:       Sheep
                    Parameter estimates from tabulated data values
Distribution:       Negative Binomial
                    Pr(X=r) = (r+k-1)C(k-1).(m/(m+k))**r.(1+m/k)**(-k)
Deviance:           30.75 on 24 d.f.

*** Estimates of working parameters ***

                    estimate      s.e.      Correlations
mean                  6.5611    0.6132    1.0000
variance             30.7804    7.0377    0.7317      1.0000

*** Estimates of defining parameters ***

                    estimate      s.e.      Correlations
m                     6.5611    0.6126    1.0000
k                     1.7774    0.3515    0.0002      1.0000
1/k                   0.5626    0.1113    Poisson Index

*** Fitted values (expected frequencies) and residuals ***

 r          Number          Number          Weighted
            Observed        Expected        Residual
 0             4              5.26            -0.57
 1             5              7.35            -0.92
 2            11              8.03             0.99
 3            10              7.96             0.70
 4             9              7.48             0.54
 5            11              6.80             1.48
 6             3              6.04            -1.37
 7             5              5.28            -0.12
 8             3              4.56            -0.78
 9             2              3.90            -1.06
10             2              3.31            -0.78
11             5              2.79             1.19
12             0              2.33            -2.16
13             2              1.95             0.04
14             2              1.62             0.29
15             1              1.34            -0.31
16             1              1.10            -0.10
17             0              0.91            -1.35
18             0              0.75            -1.22
19             1              0.61             0.46
20             0              0.50            -1.00
21             1              0.41             0.78
22             1              0.33             0.93
23             1              0.27             1.08
24             0              0.22            -0.66
25             2              0.18             2.46
26+            0              0.75            -1.22
```

Fisher himself, having obtained his fitted frequencies, used them as the basis of a grouping for his chi-squared test, with the total expected frequency being at least 5 for each group (including the tail group). A Genstat analysis for this grouping is as follows.

```
 43
 44  "Analysis with Fisher's grouping."
 45  FACTOR [LEVELS=!(0...6,8,11,15,100)] Ticks1; DECIMALS=0
 46  TABLE [CLASSIFICATION=Ticks1] Sheep1; DECIMALS=0
 47  READ [PRINT=*] Sheep1
 49  PRINT [ACROSS=Ticks1] Sheep1; FIELDWIDTH=7

            Sheep1
Ticks1  0      1      2      3      4      5      6      8     11     15    100
        4      5     11     10      9     11      3      8      9      5      7

 50  DISTRIBUTION [DISTRIBUTION=negativebinomial] Sheep1

50.....................................................................

***** Fit discrete distribution *****

*** Sample Statistics ***

  Sample Size              82
  Mean               6.38        Variance                          29.09
  Skewness           1.29
  Poisson Index      0.56        Negative Binomial Index            1.59


*** Summary of analysis ***

Observations:     Sheep1
                  Parameter estimates from tabulated data values
Distribution:     Negative Binomial
                  Pr(X=r) = (r+k-1)C(k-1).(m/(m+k))**r.(1+m/k)**(-k)
Deviance:         8.17 on 8 d.f.


*** Estimates of working parameters ***

                  estimate      s.e.          Correlations
mean               6.3468       0.5992        1.0000
variance          27.2680       6.7936        0.7379        1.0000


*** Estimates of defining parameters ***

                  estimate      s.e.          Correlations
m                  6.3468       0.5991        1.0000
k                  1.9254       0.4241        -0.1004       1.0000
1/k                0.5194       0.1144        Poisson Index

*** Fitted values (expected frequencies) and residuals ***

  r                Number       Number        Weighted
                   Observed     Expected      Residual
  0                   4            4.95         -0.44
  1                   5            7.32         -0.91
  2                  11            8.21          0.92
  3                  10            8.24          0.59
  4                   9            7.79          0.42
  5                  11            7.08          1.36
  6                   3            6.27         -1.46
  7-8                 8           10.11         -0.69
  9-11                9           10.00         -0.32
  12-15               5            6.91         -0.77
  16+                 7            5.11          0.79
```

The goodness-of-fit test on 8 d.f. here is more reliable than one on 24 d.f., particularly because, with 24 d.f., there are so many small expected frequencies.

For a first run using the **DISTRIBUTION** directive, the user cannot know in advance what the fitted frequencies will be, and so must make somewhat arbitrary grouping decisions based on the observed frequencies. For the Fisher data, Ross and Preece (1985) suggested the grouping adopted in the following Genstat output.

```
  51
  52  "Analysis with Ross & Preece's grouping."
  53  FACTOR [LEVELS=!(0...8,10,12,15,19,100)] Ticks2; DECIMALS=0
  54  TABLE [CLASSIFICATION=Ticks2] Sheep2; DECIMALS=0
  55  READ [PRINT=*] Sheep2
  57  PRINT [ACROSS=Ticks2] Sheep2; FIELDWIDTH=7

        Sheep2
Ticks2       0    1     2    3    4    5   6    7    8   10
             4    5    11   10    9   11   3    5    3    4

Ticks2      12   15   19   100
             5    5    2     5

  58  DISTRIBUTION [DISTRIBUTION=negativebinomial] Sheep2

58.....................................................

***** Fit discrete distribution *****

*** Sample Statistics ***

  Sample Size          82
  Mean                6.71        Variance                    38.38
  Skewness            1.62
  Poisson Index       0.70        Negative Binomial Index      1.90


*** Summary of analysis ***

Observations:    Sheep2
                 Parameter estimates from tabulated data values
Distribution:    Negative Binomial
                 Pr(X=r) = (r+k-1)C(k-1).(m/(m+k))**r.(1+m/k)**(-k)
Deviance:        11.56 on 11 d.f.

*** Estimates of working parameters ***

             estimate      s.e.        Correlations
mean          6.6616      0.6493      1.0000
variance     32.7155      8.1238      0.7538   1.0000

*** Estimates of defining parameters ***

             estimate      s.e.        Correlations
m             6.6616      0.6484      1.0000
k             1.7033      0.3492     -0.0712   1.0000
1/k           0.5871      0.1204      Poisson Index

*** Fitted values (expected frequencies) and residuals ***

  r          Number       Number      Weighted
             Observed     Expected    Residual
  0             4          5.45        -0.65
  1             5          7.40        -0.94
  2            11          7.96         1.02
  3            10          7.83         0.74
  4             9          7.33         0.60
  5            11          6.66         1.54
  6             3          5.92        -1.33
  7             5          5.19        -0.08
  8             3          4.50        -0.75
 9-10           4          7.15        -1.29
11-12           5          5.14        -0.06
13-15           5          5.00         0.00
16-19           2          3.53        -0.89
20+             5          2.95         1.08
```

The grouping in this last variant of the analysis of the Fisher data is satisfactory.

7. Example 3 (Currie, 1993)

The frequency table of Currie (1993, p.31) is for the number of claims made on each of 100,000 car insurance policies in one year, the policies all belonging to the same portfolio:

Number of claims	0	1	2	3	4	>4	Total
Number of policies (frequency)	81056	16174	2435	295	36	4	100000

Here, the tail is already grouped; we are not told exactly how many policies produced, respectively, 5, 6, 7 , ..., claims.

Currie (1993, p.32) fitted the negative binomial distribution by the method of moments. Genstat output for maximum likelihood fitting of the distribution is as follows; as a reminder of the arbitrariness of the value chosen for the last level of the **FACTOR** for the **TABLE** of frequencies, the value 99 has now been chosen.

```
 59
 60  "Analysis of actuarial data of Currie (1993)
-61   ----------------------------------------"
 62
 63  FACTOR [LEVELS=!(0...4,99)] Claims; DECIMALS=0
 64  TABLE [CLASSIFICATION=Claims] Policies; DECIMALS=0
 65  READ [PRINT=*] Policies
 67  PRINT [ACROSS=Claims] Policies; FIELDWIDTH=7

     Policies
Claims          0          1          2          3          4         99
           81056      16174       2435        295         36          4

 68  DISTRIBUTION [DISTRIBUTION=negativebinomial] Policies

68.....................................................................

***** Fit discrete distribution *****

*** Sample Statistics ***

     Sample Size          100000
     Mean                 0.22        Variance                      0.24
     Skewness             2.46
     Poisson Index        0.48        Negative Binomial Index       2.36

*** Summary of analysis ***

Observations:       Policies
                    Parameter estimates from tabulated data values
Distribution:       Negative Binomial
                    Pr(X=r) = (r+k-1)C(k-1).(m/(m+k))**r.(1+m/k)**(-k)
Deviance:           6.45 on 3 d.f.


*** Estimates of working parameters ***

                    estimate          s.e.          Correlations
mean                0.2210            0.0016        1.0000
variance            0.2441            0.0023        0.8197    1.0000

*** Estimates of defining parameters ***

                    estimate          s.e.          Correlations
m                   0.2210            0.0016        1.0000
k                   2.1103            0.1220        -0.0150   1.0000
1/k                 0.4739            0.0274        Poisson Index
```

```
*** Fitted values (expected frequencies) and residuals ***

 r        Number      Number          Weighted
          Observed    Expected        Residual
 0        81056       81040.44          0.05
 1        16174       16211.66         -0.30
 2         2435        2389.92          0.92
 3          295         310.40         -0.88
 4           36          37.59         -0.26
 5+           4          10.00         -2.16
```

The following table permits a comparison of expected frequencies from the two methods of fitting:

Number of claims	0	1	2	3	4	>4
Observed number of policies	81056	16174	2435	295	36	4
Expected number (moments)	81034	16234	2383	307	37	5
Expected number (max. like'd)	81040	16212	2390	310	38	10

In this example, neither method of fitting produces a poor fit. In assessing the maximum likelihood fit, we must recall that Genstat took all 4 tail observations to have been for (1.25 × 5) = 6.25 claims.

Acknowledgement

The authors are grateful for the Actuarial Education Service's willingness for the frequency table of Currie (1993, p.31) to be used in this paper.

References

Anscombe F J (1949) The statistical analysis of insect counts based on the negative binomial distribution. *Biometrics* **5** 165-173.

Bliss C I (1953) Fitting the negative binomial distribution to biological data (with 'Note on the efficient fitting of the negative binomial' by R A Fisher). *Biometrics* **9** 176-200.

Cochran W G (1954) Some methods for strengthening the common chi-squared tests. *Biometrics* **10** 417-451.

Currie I D (1993) *Loss Distributions*. Actuarial Education Service, London.

Fisher R A (1941) The negative binomial distribution. *Annals of Eugenics* **11** 182-187.

Institute of Actuaries and Faculty of Actuaries (1980). Formulae and Tables for Actuarial Examinations. London.

Ross G J S (1980) *MLP: Maximum Likelihood Program* Statistics Department, IACR-Rothamsted, Harpenden, UK.

Ross G J S and Preece D A (1985) The negative binomial distribution. *The Statistician* **34** 323-336.

Efficient analysis of field experiments using two-dimensional spatial models

Arthur Gilmour
NSW Agriculture
Agricultural Research and Veterinary Centre
Forest Road, Orange, NSW, 2800, Australia

Sue Welham, Simon Harding
IACR-Rothamsted
Harpenden AL5 2JQ, UK

1. Introduction

For several years, NSW Agriculture has used variance models based on the spatial arrangement of plots for the analysis of cereal trials. These include models in which the plot covariance is modelled in two dimensions by the class of separable ARIMA processes (Cullis and Gleeson, 1991; Martin, 1990). ARIMA models are often more efficient then incomplete-block models (Patterson and Hunter, 1983; Cullis and Gleeson, 1989; Gilmour and Cullis, 1995).

The program TwoD (Gilmour, 1992) is widely used in Australia to fit spatial models. It is distributed with a procedure, TWOD, to run TwoD from within Genstat 5. This article provides a basic description of spatial models, comments on when and why they are appropriate and discusses some practical issues in variance modelling through an example using the TWOD procedure.

2. What are spatial models?

Crop variables such as yield and protein levels usually contain variation which is associated with the actual location of the plots. Spatial models seek to remove this variation to obtain a more efficient analysis. Incomplete-block designs are popular spatial models but neighbour models are often more efficient (Gilmour and Cullis, 1995). In neighbour models, the covariance between plots depends directly on the distance between plots. For incomplete-block designs, the block boundaries are often artificial so, in one dimension, some contiguous plots are assumed independent, being in different blocks, but most are not.

The neighbour model of Gleeson and Cullis (1987) assumes that the plot errors are a realisation of a random ARIMA process. Cullis and Gleeson (1991) extended these models to two dimensions by assuming separability of the random row and column processes. The size of field experiments typically permits only low-order models. We find that autoregressive models normally suffice but ARIMA (0,1,1), (1,1,1) and/or row/column effects are sometimes required. Lill *et al* (1988) showed improved accuracy of treatment estimates, low levels of bias in the treatment F ratio and approximate validity of SEDs from spatial analyses.

Cullis and Gleeson (1989) reported an average efficiency of 1.73 over 1019 trials using ARIMA (0,1,1) in one dimension relative to the complete block analysis. They reported an efficiency of 1.50 over 239 trials for incomplete-block analysis. Patterson and Hunter (1983) reported similar values of 1.79 and 1.43 respectively.

Differencing was advocated in the neighbour models of Wilkinson *et al* (1983), Besag and Kempton (1986) and Gleeson and Cullis (1987). The ARIMA (0,2,2) model is similar to that proposed by Wilkinson *et al* (1983); the ARIMA (0,1,1) model is equivalent to the first difference model of Besag and Kempton (1986) and programmed in Genstat by Baird (1987).

Kempton *et al* (1994) found no gain in average efficiency, relative to incomplete-block analysis, from fitting ARIMA (0,1,1) in both directions: but here the differencing discarded treatment information when there was little trend since treatments were not orthogonal to differencing. Gilmour and Cullis (1995), using the same data, reported an average efficiency of 1.71 from fitting ARIMA (1,0,0) in both directions compared with 1.43 for the incomplete-block analyses. Differencing was advocated to remove non-stationarity in spatial variation.

However, we currently have no reliable test for non-stationarity, and differencing is counter-productive when it discards treatment information. Also, it is difficult to objectively compare the efficiency using differencing with efficiency for other spatial models. Therefore, while differencing may sometimes be necessary, we usually avoid it. If differencing is used, a moving average term should always be included in the spatial model.

3. The linear model

We consider the general linear mixed model

$$y = X\beta + Zu + e$$

where X [Z] is a design matrix for fixed [random] effects β [u], u is N(0, $\sigma^2\Gamma$), e is N(0, $\sigma^2\Sigma$). Assume the observations y are in plot order, rows nested within columns, so that $\Sigma = \Sigma_C \otimes \Sigma_R$ where Σ_C and Σ_R are proportional to variance matrices for column and row processes respectively, and $\otimes$ indicates the matrix cross product operator.

In the program TwoD, the user nominates the form of Σ_R and Σ_C (Identity, Autoregressive, etc.). TwoD estimates the variance parameters using restricted maximum likelihood (REML, Cullis and Gleeson, 1991) and forms the GLS (generalised least squares) solution $\hat{\beta}$ for β, the BLUP (best linear unbiased predictor) $\tilde{u}$ for u and the residuals

$$\tilde{e} = y - X\hat{\beta} - Z\tilde{u}.$$

To check that the spatial model is adequate, we examine the residuals for outliers and patterns. One tool for this is the spatial correlation matrix advocated by Cullis and Gleeson (1991) (see Martin, 1990). The elements $r_{i,j}$ of this matrix are the correlations between pairs of residuals i rows and j columns apart. Note that $r_{0,0}$ is always 1 and $r_{0,i} = r_{0,-i}$. Under the separability assumption, $r_{1,1}$ should be close to $r_{1,0} \times r_{0,1}$. If based on *whitened* residuals,

$$\tilde{e}^* = \hat{\Sigma}^{-0.5}\tilde{e},$$

which are adjusted for trend, all the correlations should be low. If we are not satisfied, we repeat the process with an alternate variance model. These tools are not definitive diagnostics for choosing a spatial model, and likelihood ratio tests can also be used to help choose between certain spatial models (see Section 5). However, the estimates of treatment effects appear robust to mild misspecification of the variance model (Gilmour and Cullis, 1995).

4. The Genstat TWOD interface to the TwoD program

The **TWOD** procedure has arguments that specify the field layout, fixed and random effects in the linear model and the variance structure to be applied to rows and columns. It sorts the observations into field order, constructs the design matrix, X, writes the information TwoD needs to the file **g52d.g2d** and runs TwoD using **SUSPEND**. TwoD reads **g52d.g2d**, fits the model and stores the results in **g52d.o2d**. Genstat retrieves the results into structures that the user can access using **TWODISPLAY** and **TWOKEEP**. For circumstances where Genstat and TwoD cannot be run concurrently, there is an option to tell Genstat to stop while TwoD runs; **TWORESUME** is then used to retrieve all relevant information when Genstat is restarted.

4.1 Procedure TWOD

The field layout is specified with options **ROWS** and **COLUMNS**. If the data is already sorted rows within columns, these may be integers specifying the number of rows and number of columns. Otherwise they must be factors and are used to sort the data.

The **FIXED** option supplies the fixed effects model formula. **TWOD** uses the regression directives **MODEL** and **FIT** to form the design matrix, X, for up to about 150 effects. Up to six random factors may be declared using the **RANDOM** option; starting values for their variance ratios, relative to the residual variance, are given with the **GAMMAS** option. Random factors often represent blocks, rows, columns or plots in an analysis. A random factor

may contain missing values as in the analysis of unreplicated early generation trials proposed by Cullis *et al* (1989). In their analysis, the factor denoting the test lines is random and has missing values for plots where check varieties were grown. A fixed factor codes for the check varieties and includes an extra level to represent the mean of the test lines.

The variance structures Σ_R and Σ_C are specified by options **RMODEL** and **CMODEL** respectively. The following settings are available:

identity	no spatial correlation (default)
ar1	first order autoregressive
ar2	second order autoregressive
arma11	first order autoregressive + moving average
ma1	first order moving average
ma2	second order moving average
uniform	uniform variance pattern across field, $\mathbf{I}+\phi\mathbf{J}$
linearvariance	linear variance model
iar	first order autoregressive + independent random error
diggle1	autoregressive on irregular grid (based on distance)
diggle2	autoregressive on irregular grid (based on squared distance)

The **diggle** models provide for autoregression when plots are not on a regular grid (Diggle, 1988), in which case options **RPOSITION** and **CPOSITION** are used to give the plot coordinates for rows and columns respectively. Initial values for the parameters can be supplied using the **INITIAL** option.

Differencing the data or fitting fixed row/column effects or trends is controlled by the **RDIFFERENCE** and **CDIFFERENCE** options: values 0, 1 or 2 specify the degree of differencing; –1, –2 or –3 specify a linear, quadratic or cubic trend and –4 requests that row/column effects be fitted as fixed effects.

Other useful options include: **MODEL** names a pointer to hold the model specification; **MAXCYCLE** specifies the number of REML iterations required; **BATCH** controls whether alternate models can be fitted interactively in TwoD before returning to Genstat; **SIZE** is used to request more memory in TwoD if needed for large problems; and **STOP=yes** means that Genstat is stopped (rather than suspended) so that TwoD is run independently.

The **Y** parameter specifies the dependent variable and the **SAVE** parameter names a pointer to hold the results. Missing data values are estimated from the fixed and random models by covariance.

4.2 Other procedures used with TWOD

Procedure **TWORESUME** has one option, **MODEL**, corresponding to the **TWOD MODEL** option and one parameter, **SAVE**, corresponding to the **TWOD SAVE** parameter. When **STOP=yes** is used in **TWOD**, then Genstat is stopped, TwoD is run and then returns to the operating system (rather than back to Genstat with **STOP=no**). The user then restarts Genstat and uses procedure **TWORESUME** to get back all the information on the TwoD fit. Full instructions are given on screen or in the output file, as appropriate, before Genstat stops.

Procedure **TWODISPLAY** is used to display results from a TwoD fit. The **PRINT** option has settings: **model** prints a description of the model fitted, **summary** gives summary statistics from the fit, **fixed** prints estimates of fixed effects plus standard errors, Wald tests for each fixed model term and predicted means for each fixed model term, **random** prints BLUP estimates of random effects, **residuals** gives raw ($\tilde{e}$) and whitened ($\tilde{e}^*$) residuals and **correlations** prints the raw and whitened spatial correlation matrix. The **SAVE** and **MODEL** options name the corresponding structures declared in the **TWOD** procedure.

Procedure **TWOKEEP** extracts components of the analysis into vectors and scalars from the pointer structure identified with the **SAVE** option (the **TWOD SAVE** parameter). The parameters are: **ESTIMATES** of spatial parameters; **LIKELIHOOD** value at final iteration; residual **DF**; **NITERATION** number of iterations; **VCOVARIANCE** for spatial parameters; **FIXED** estimates of full set of fixed effects; **FVCOVARIANCE** covariance matrix for full set of fixed effects; **RANDOM** pointer to effects for each random term; **SEDRANDOM** pointer to sed for each random term; **NROWS**; **NCOLUMNS**; **RESIDUALS** as nrows × ncols matrix; **WRESIDUALS** whitened residual matrix; spatial **CORRELATIONS**; and **SPSE**, the standard errors for the spatial correlations. **NROWS** and **NCOLUMNS** are the

dimensions of the **RESIDUALS** matrix; they will be less than the field dimensions if differencing has occurred.

5. Example

Below is a Genstat job fitting three 3 spatial models to wheat yields obtained in 1976 at Slate Hall Farm. The data and treatment codes are in Gilmour *et al* (1995). The output produced from the TwoD program and the **TWOD** procedure is considered for each model separately below:

```
"
  Example 3: Slate Hall Farm 1976   Balanced Lattice Design
  reps laid out as  1 1 1 1 1 2 2 2 2 2 3 3 3 3 3
                    1 1 1 1 1 2 2 2 2 2 3 3 3 3 3
                    1 1 1 1 1 2 2 2 2 2 3 3 3 3 3
                    1 1 1 1 1 2 2 2 2 2 3 3 3 3 3
                    1 1 1 1 1 2 2 2 2 2 3 3 3 3 3
                    4 4 4 4 4 5 5 5 5 5 6 6 6 6 6
                    4 4 4 4 4 5 5 5 5 5 6 6 6 6 6
                    4 4 4 4 4 5 5 5 5 5 6 6 6 6 6
                    4 4 4 4 4 5 5 5 5 5 6 6 6 6 6
                    4 4 4 4 4 5 5 5 5 5 6 6 6 6 6

  Data fields: Replicate rowblock columnblock variety yield
                   1:6      1:30       1:30       1:25     "

factor rep,rowblk,colblk,variety; dec=0
open 'shfbld.dat'; inchan
read [ch=inchan] rep,rowblk,colblk,variety,yield
close inchan

" Model 1: *** Balanced Lattice Analysis ****
twod [rows=15; columns=10; fixed=variety; \
  random=rep,rowblk,colblk; gammas=0.2,0.5,0.5] yield
twodisplay [print=model,sum,fixed,corr]

" Model 2: *** AR1 by AR1 Analysis *** "
twod [rows=15; columns=10; fixed=variety ;\
     rmod=ar1; cmod=ar1] yield
twodisplay [print=model,sum,fixed,corr]

" Model 3: *** (AR1 by AR1) + independent error ****
factor [level=150;values=1...150] plot
twod [rows=15; columns=10; fixed=variety; \
     random=plot; gammas=0.1; rmod=ar1; cmod=ar1] yield
twodisplay [print=model,sum,fixed,corr]
```

In this program, the option settings **rows=15; columns=10** indicates the data is ordered row-wise with respect to the layout displayed; the 15 rows are nested within the 10 columns. Spatial analysis requires correct specification of the field plan. If the data is not already in this order, then factors specifying the layout can be given in the **rows** and **columns** options, and the procedure will get the correct ordering from these factors.

Below is the TwoD output from model 1, a balanced lattice analysis with reps, row and column random factors, no spatial model:

```
   32-bit Power for Lahey Computer Systems
  Phar Lap's 386|DOS-Extender(tm) Version 5.1
 Copyright (C) 1986-93 Phar Lap Software, Inc.
         Available Memory = 17344 Kb

Two Dimensional Spatial Analysis      (C) NSW Agriculture, 2800, Australia
Min Mean Max of yield          917.000 1470.440 2119.000
Work space: Using        94 of      7969 K bytes
Iter- LogLike-  Error    DF  Rows  model    Columns model      Random
ation lihood  Variance        Identity       Identity          Vu/Ve
 1  -654.2  0.1452E+05 125  0.0000 0.0000 0.0000 0.0000 0.2000 0.5000 0.5000
 2  -645.9      9164.   125  0.0000 0.0000 0.0000 0.0000 0.3846 1.3816 1.3601
 3  -645.3      8176.   125  0.0000 0.0000 0.0000 0.0000 0.4995 1.8549 1.7828
 4  -645.3      8063.   125  0.0000 0.0000 0.0000 0.0000 0.5275 1.9333 1.8374
 5  -645.3      8062.   125  0.0000 0.0000 0.0000 0.0000 0.5287 1.9344 1.8373
```

```
Analysis of residuals (trend component included)
 Residual Plot   and    Autocorrelations
  <LOo- +xXH>                [se   0.089]
 |-oxO+  X+-x---  |    1 -0.128 -0.169  1.000 -0.169 -0.128
 |++oxo - -++- +  |    2 -0.036  0.106 -0.067  0.158 -0.136
 |oo-++ +- --++ox |    3  0.118  0.004 -0.204 -0.010  0.064
 |x-Lxx+  +- ++ o |    4  0.075 -0.006 -0.128 -0.043  0.128
 |o>H-o++-o+o -x+ |    5 -0.031  0.004  0.053 -0.009 -0.028
 |xX- o++-  +-o+  |
 |+ --xx+-o--++o- |
 |Oo+++-x +o-o  - |
 |- -x --X- O++ + |
  -X-     ox -
```

After displaying the banner and range of the data values, TwoD displays the likelihoods and parameter values for the iterations. In this first model, there are no parameters associated with Σ. The last three columns refer to random factors **rep**, **rowblk** and **colblk** respectively. There is then a symbolic plot of the residuals and a matrix of spatial autocorrelations among residuals. The symbols < and > indicate the corresponding residual is less (more) than –2.25 (2.25) standard deviations. The symbols 'LOo– +xXH' represent intervals of 0.5 standard deviations between –2.25 and 2.25 in order. The description "trend component included" indicates that these are the raw residuals $\tilde{e}$.

The spatial correlations from this block analysis are reasonably good given their approximate standard error of 0.089; the next model is better. The value of –0.169 indicates moderate negative correlation between residuals in the smaller dimension based on 135 pairs. In the long dimension (ie. rows here), the $r_{2,0}$ and $r_{3,0}$ values of –0.204 and –0.128 show a similar pattern based on 130 and 120 pairs respectively. The correlations $r_{1,-1}$ and $r_{1,1}$ of 0.106 and 0.158 are based on 126 diagonally associated pairs in the negative (↙) or positive (↘) directions.

The Genstat output from procedure **TWODISPLAY** follows:

```
 30  twodisplay [print=model,est,sum,fixed,corr]

***** TwoD Analysis *****

***** Fitted Model *****

       Y Variate:  yield
     Fixed Terms:  variety
    Random Terms:  rep + rowblk + colblk
Estimated gammas:  0.529  1.934  1.837

      Row Factor:  15 rows
       Row Model:  Identity
   Column Factor:  10 columns
    Column Model:  Identity

***** Summary of TwoD Fit *****

      sigma**2  8062
log-likelihood  -645.3
            df  125
    iterations  5

***** Fixed Effects with Standard Errors *****

   Constant      1283.6       60.20
  variety 2       265.4       62.02
  variety 3       137.3       62.02
  variety 4       168.3       62.02
  variety 5       249.7       62.02
          .
          .       intermediate values omitted to save space
          .
 variety 21       209.9       62.02
 variety 22       360.8       62.02
 variety 23        45.5       62.02
 variety 24       262.9       62.02
 variety 25       347.0       62.02

*** Wald statistics for fixed model terms ***
```

```
Term            Wald test    d.f.    p-value

variety          212.26       24      0.000

Note: These tests are relevant to dropping terms out of the fixed model
      They are not valid for any term which is marginal to other terms
      in the fixed model.
      Eg. FIXED=A*B => tests for A and B main effects are not valid

*** Predicted means for variety ***

Average SED of effects = 62.02

     variety
           1        1284        60.20
           2        1549        60.20
           3        1421        60.20
           .
           .        intermediate values omitted to save space
           .
          23        1329        60.20
          24        1546        60.20
          25        1631        60.20

***** Spatial Correlations *****

                      1           2           3           4           5

           1     -0.1276     -0.1690      1.0000     -0.1690     -0.1276
           2     -0.0362      0.1060     -0.0675      0.1576     -0.1360
           3      0.1184      0.0035     -0.2039     -0.0098      0.0645
           4      0.0753     -0.0056     -0.1277     -0.0425      0.1282
           5     -0.0308      0.0035      0.0527     -0.0093     -0.0280

      s.e.      0.08944
```

TWODISPLAY is used to present the information from the TwoD analysis. The only additional information in this output is for the fixed effects: the estimated effects (parameterised as for regression, with first levels of factors constrained to zero) are printed together with a Wald test for the fixed term **variety** and predictions for the **variety** means, obtained through regression directives using the correct covariance matrix for the parameters.

The second model fits an autoregressive term of order 1 (AR1) across both rows and columns:

```
Two Dimensional Spatial Analysis      (C) NSW Agriculture, 2800, Australia
Min Mean Max of yield          917.000 1470.440 2119.000
Work space: Using        24 of      7969 K bytes
Iter- LogLike-  Error    DF     Rows     model      Columns model      Random
ation lihood  Variance           AutoRegressive     AutoRegressive      Vu/Ve
 1-641.3     0.1789E+05 125    0.5000    0.0000    0.5000    0.0000
 2-637.8     0.1631E+05 125    0.6786    0.0000    0.4625    0.0000
 3-637.8     0.1630E+05 125    0.6835    0.0000    0.4593    0.0000
 4-637.8     0.1630E+05 125    0.6837    0.0000    0.4587    0.0000
Analysis of residuals (trend component removed)
 Residual Plot   and    Autocorrelations
  <LOo- +xXH>              [se   0.089]
 |O +o+H+Ho  ---x|    1  0.055  0.006  1.000  0.006  0.055
 |+XOHO     +++++|    2 -0.004  0.129  0.008  0.108 -0.063
 |L -++   -+o-x ox|    3  0.071  0.058  0.042  0.021  0.006
 |+ oxx+-++--+  o|    4  0.039 -0.012 -0.040 -0.026 -0.017
 |x>X o++-o+- ox+|    5 -0.032  0.027  0.058  0.000 -0.011
 |x o++x+Ooo+-o+-|
 |     X+ -ooo- -o|
 |L-X+- XooL-O -o|
 |  +-x x xO- X ++|
 |+ Ho - o-L> -+ |
Analysis of residuals (trend component included)
 Residual Plot   and    Autocorrelations
  <LOo- +xXH>              [se   0.089]
 |O- o XX>x++   -+|    1  0.250  0.540  1.000  0.540  0.250
 | x x ++x++x+xxx|    2  0.209  0.444  0.702  0.401  0.168
 |LoO  +    --++ +|    3  0.178  0.329  0.461  0.254  0.101
 |- o+xx+++ -++ -|    4  0.085  0.167  0.226  0.081 -0.017
 |x>>Hxxx+  - -  |    5 -0.004  0.056  0.078 -0.002 -0.065
 |xHxxxXX oO--o--|
```

```
|++++XXx oLOOOOO|
|OO  +++X o<L<OOL|
|-   x+XXx-OO-   |
|  Xx+++ o<      |
```

Here, the log likelihood of –637.8 (after fitting 2 variance parameters) is significantly higher than –645.3 (with three parameters) from the previous analysis. Note that log-likelihoods are not comparable between models with different fixed effects or different levels of differencing.

The first residual display (trend removed or whitened residuals, $\tilde{e}^*$) should be compared with that in the second residual display (trend removed). The correlations are all much lower except $r_{1,-1}$ and $r_{1,1}$ (values 0.129 and 0.108). A pattern, not related to the block effects, is evident in the second display (trend included or raw residuals $\tilde{e}$). Under the separability assumption, $r_{1,-1}$ and $r_{1,1}$ should be approximately equal to $0.36 = 0.702 \times 0.540$. The agreement is reasonable.

The effect of fitting the spatial model on estimates of fixed effects can be seen in the output from **TWODISPLAY**:

```
***** TwoD Analysis *****

***** Fixed Effects with Standard Errors *****

   Constant      1258.0        64.62
  variety 2       243.5        61.50
  variety 3       147.0        61.16
  variety 4       154.6        61.70
  variety 5       256.5        62.35
          .
          .       intermediate rows omitted to save space
          .
 variety 21       259.6        62.94
 variety 22       347.1        59.27
 variety 23        53.5        61.04
 variety 24       328.8        57.50
 variety 25       334.0        57.95


*** Wald statistics for fixed model terms ***

Term           Wald test    d.f.    p-value

variety           313.07      24      0.000


*** Predicted means for variety ***

Average SED of effects = 59.05

     variety
           1        1258        64.62
           2        1501        64.99
           3        1405        64.63
           4        1413        64.91
           5        1514        65.60
           .
           .
          21        1518        64.71
          22        1605        64.39
          23        1311        64.08
          24        1587        64.71
          25        1592        63.60
```

Here, the estimates of variety effects show some small changes, and the average sed of variety effects is reduced, so that comparisons between varieties are more accurate on average than in the balanced lattice analysis, although standard errors of predictions are slightly increased.

The third model fits an AR1 spatial model to rows and columns plus an independent error term:

```
Two Dimensional Spatial Analysis      (C) NSW Agriculture, 2800, Australia
Min Mean Max of yield          917.000 1470.440 2119.000
Work space: Using       275 of      7969 K bytes
Iter- LogLike-  Error    DF     Rows     model      Columns model      Random
```

```
ation lihood  Variance        AutoRegressive    AutoRegressive     Vu/Ve
 1-641.7     0.1601E+05 125   0.5000   0.0000   0.5000   0.0000 0.1000
*WARNING* Parameter values changed from      0.9924    0.0000 to    0.9800    0.0000
 2-641.3      5865.      125   0.9800   0.0000   0.6874   0.0000 0.8571
 3-634.4      6034.      125   0.8729   0.0000   0.7120   0.0000 0.9043
 4-634.3      7044.      125   0.8478   0.0000   0.6887   0.0000 0.6797
 5-634.3      7011.      125   0.8447   0.0000   0.6839   0.0000 0.6960
Analysis of residuals (trend component removed)
 Residual Plot   and    Autocorrelations
  <LOo- +xXH>              [se   0.089]
|o++ xH+xo-  --+|    1  0.043  0.195  1.000  0.195  0.043
|xxoxo--  +x++++|    2  0.012  0.059  0.298  0.051 -0.059
|L -x+  - - x -+|    3  0.064  0.026  0.051  0.002 -0.027
|++-x++   --+- o|    4  0.053 -0.044 -0.089 -0.070 -0.031
|HH+--  oo   -x |    5 -0.043  0.031 -0.065  0.033 -0.007
|  --+++ Ooo+-- -|
|--++x++oOO-- --|
|O-x+  x-OO -  -|
|++++ ++++-xx+++|
|+ x-------H -+ |
Analysis of residuals (trend component included)
 Residual Plot   and    Autocorrelations
  <LOo- +xXH>              [se   0.089]
|o--- xxXx+++    |    1  0.330  0.637  1.000  0.637  0.330
|  + ++x+x+++x++x|    2  0.278  0.529  0.803  0.488  0.239
|Ooo  +   - ++ +|    3  0.199  0.378  0.551  0.307  0.126
|- -++x+++  +    |    4  0.093  0.201  0.289  0.110 -0.012
|xHHHXXx+        |    5  0.001  0.068  0.102 -0.010 -0.086
|xXxxxXx -oo-o--|
|+++xxXx oOOoOoO|
|oo ++xx oLLLOOO|
|--+xxxXx-Oo-   |
| +xx+++ oO-    |
```

In model 3, we added an independent error term to the model. While the improvement in the log likelihood is significant ($2(-634.3 - -637.8)=7.0 > \chi^2_{1,0.05}= 3.84$), the changes in the solutions for the fixed effects are small, as can be seen in the **TWODISPLAY** output:

```
***** Fixed Effects with Standard Errors *****

   Constant       1245.5        98.28
  variety 2        270.7        62.16
  variety 3        158.4        61.83
  variety 4        159.3        62.46
  variety 5        225.9        62.67
          .
          .
 variety 21        269.2        62.60
 variety 22        363.4        60.77
 variety 23         71.4        62.02
 variety 24        311.9        58.59
 variety 25        328.3        60.33

*** Wald statistics for fixed model terms ***

Term           Wald test    d.f.   p-value

variety          245.24      24     0.000

*** Predicted means for variety ***

Average SED of effects = 60.51
     variety
           1        1245       98.28
           2        1516       98.27
           3        1404       98.66
           4        1405       98.41
           .
           .
          22        1609       98.64
          23        1317       98.46
          24        1557       98.55
          25        1574       98.41
```

The major consequence of including the independent error is to slightly increase the SED of treatment contrasts (Gilmour and Cullis, 1995) compared to model 2, also the SE of treatment effects increases. Note that the first update of the parameters was too large but TwoD adjusted them and they subsequently converged.

The *trend removed* residuals in this analysis, ie. $\tilde{e}^*$, are the lack of fit between the autoregressive trend and the independent residuals fitted by the random factor. The second plot shows the fitted trend (with independent error removed), $\tilde{e} = y - X\beta - Z\tilde{u}$.

6. Conclusion

We have described the TWOD procedure and shown the screen output from the TwoD program when run from within Genstat. The fitted effects and residuals are returned to Genstat where they can be displayed or manipulated as desired. Ultimately, these methods will be formally included in Genstat.

The use of TWOD should result in more efficient analysis of field experiments and greater awareness of the residuals among experimenters. There is however a learning curve as experimenters discover how spatial models address features of the data which were previously ignored.

TwoD is available from NSW Agriculture for A$200 to licensed Genstat users. It may be obtained by anonymous ftp from directory /pub/genstat/twod at ftp.res.bbsrc.ac.uk for evaluation and for non-commercial use.

References

Baird D B (1987) A Genstat 5 procedure for a First Difference Analysis. *The Genstat Newsletter* **19** 40-47.

Besag J and Kempton R (1986) Statistical analysis of field experiments using neighbouring plots. *Biometrics* **42** 231-251.

Cullis B R and Gleeson A C (1989) Efficiency of neighbour analysis for replicated variety trials in Australia. *Journal of Agricultural Science, Cambridge* **113** 233-239.

Cullis B R, Gleeson A C, Lill W J, Fisher J A and Read B J (1989) A new procedure for the analysis of early generation trials. *Applied Statistics* **38** 361-375.

Cullis B R and Gleeson A C (1991) Spatial analysis of field experiments - an extension to two dimensions. *Biometrics* **47** 1449-1460.

Cullis B R, Gleeson A C and Thomson F M (1992) The response to selection of different procedures for the analysis of early generation variety trials. *Journal of Agricultural Science, Cambridge* **118** 141-148.

Diggle P J (1988) An approach to the analysis of repeated measures. *Biometrics* **44** 959-971.

Gilmour A R (1992) *TwoD, a program to fit a mixed linear model with two dimensional spatial adjustment for local trend.* NSW Agriculture, Orange, NSW, 2800, Australia. 73pp.

Gilmour A R and Cullis B R (in preparation) An assessment of autoregressive spatial models for accommodating two dimensional trend in field trials.

Gilmour A R, Thompson R and Cullis B R (1995) AIREML an efficient algorithm for variance parameter estimation in linear mixed models. *Biometrics* **51** (in press)

Gleeson A C and Cullis B R (1987) Residual maximum likelihood (REML) estimation of a neighbour model for field experiments. *Biometrics* **43** 277-288.

Kempton R A, Seraphin J C and Sword A M (1994) Statistical analysis of two dimensional variation in variety yield trials. *Journal of Agricultural Science, Cambridge* **122** 335-342.

Lill W J, Gleeson A C and Cullis B R (1988) Relative accuracy of a neighbour method for field trials. *Journal of Agricultural Science, Cambridge* **111** 339-346.

Martin R J (1990) The use of time-series models and methods in the analysis of agricultural field trials. *Communications in Statistics Theory Methods* **19** 55-81.

Patterson H D and Hunter E A (1983) The efficiency of incomplete block designs in National List and Recommended cereal variety trials. *Journal of Agricultural Science, Cambridge* **101** 427-433.

Wilkinson G N, Eckert S R, Hancock T W and Mayo O (1983) Nearest Neighbour (NN) analysis of field experiments (with Discussion). *Journal of the Royal Statistical Society B* **45** 152-212.

Design of Experiments in Genstat

Roger Payne
IACR-Rothamsted
Harpenden
Herts AL5 2JQ, UK

This article is based on a talk given at the *Statistical Conference of Genstat Users* that took place at Wagga Wagga during 28-30 November 1994.

1. Introduction

Recent releases of Genstat and of the Genstat Procedure Library have contained much enhanced facilities for the design of experiments. In Release 2[3] of the Library a **design** module was instigated which contained procedures for the construction of particular designs given suitable generators: for example design keys, initial blocks for cyclic designs and generating arrays for alpha designs. There was also a procedure for printing designs, and manipulation procedures for producing a factor equivalent to the dot-product of a set of factors and for forming the ***units*** factor required to index the final stratum of a design.

In Release 3.1 of Genstat the **GENERATE** directive was extended to generate factors using design keys while, in Release 3[1] of the Procedure Library, procedures were added with both conversational and command-based interfaces to allow suitable generators to be selected from a stored repertoire. A standard repertoire was provided, based on that used by the program **DSIGNX** (Franklin and Mann 1986), but facilities were also provided to allow this to be extended or customised (see Payne and Franklin, 1994).

Release 3[2] extended the standard repertoire, and added further facilities for the manipulation of designs with procedures to form the product of two designs, to merge two designs, to append several factors together and to plot an experimental plan.

These facilities are known collectively as the *Genstat Design System*. Below, we review the philosophy and structure of the system, and explain some of the underlying methodology.

2. The Genstat Design System

The Genstat design system is intended to provide a coordinated set of facilities for the selection and construction of effective experimental designs. It aims to give

- a non-technical interface for the non-Genstat user, and a faster (command-based) method for the cognoscenti,
- a good range of standard designs, but without constraining the users just to this pre-defined set,
- display facilities for designs, plans and data forms, and
- manipulation procedures to construct more complicated designs.

Most of the facilities have been implemented as Genstat procedures, using the programming facilities in the Genstat command language. For example, **GENERATE** is used to generate factor values in standard order, or from a design key, **RANDOMIZE** to randomize the allocation of treatment factors, **CALCULATE** to generate more complicated relationships between factors, **RESTRICT** to use different generators for different replicates, **QUESTION** to implement the menu-driven interface, **GET** to check whether interactive or to use **BLOCK** or **TREAT** directives for defaults, **ASSIGN** to define the factors in the user's programme, outside the procedure, **OPEN** and **RETRIEVE** to recover information including standard generators, **FCLASSIFICATION** and **FORMULA** to process the model formulae, **BLOCKSTRUCTURE**, **TREATMENTSTRUCTURE** and **ANOVA** to produce dummy analyses, and **IF**, **ELSIF**, **ELSE**, **ENDIF**, **FOR**, **ENDFOR**, **CASE**, **OR**, **ENDCASE** and **EXIT** for the programming.

The **design** module of Release 3[2] of the Procedure Library contains the following procedures:

AFALPHA	generates alpha designs
AFCYCLIC	generates block and treatment factors for cyclic designs
AFORMS	prints data forms for an experimental design

AFUNITS	forms a factor to index the units of the final stratum of a design
AGALPHA	forms alpha designs by standard generators for up to 100 treatments
AGCYCLIC	generates cyclic designs from standard generators
AGDESIGN	generates generally balanced designs
AGFRACTION	generates fractional factorial designs
AGHIERARCHICAL	generates orthogonal hierarchical designs
AKEY	generates values for treatment factors using the design key method
AMERGE	merges extra units into an experimental design
APRODUCT	forms a new experimental design from the product of two designs
ARANDOMIZE	randomizes and prints an experimental design
DDESIGN	plots the plan of an experimental design
DESIGN	helps to select and generate effective experimental designs
FACPRODUCT	forms a factor with a level for every combination of other factors
FDESIGNFILE	forms a backing-store file of information for **AGDESIGN**
PDESIGN	prints or stores treatment combinations tabulated by the block factors

These can be classified by functionality and type of design as shown in Table 1.

	Generation	Selection	Display	Manipulation
Generally balanced design	**AKEY** **AGHIERARCH**	**AGDESIGN** **AGFRACTION**		**FDESIGNFILE**
Alpha design	**AFALPHA**	**AGALPHA**		
Cyclic design	**AFCYCLIC**	**AGCYCLIC**		
Any of these types of design		**DESIGN**	**PDESIGN** **DDESIGN** **AFORMS**	**AMERGE** **APRODUCT** **ARANDOMIZE** **AFUNITS** **FACPRODUCT**

Table 1.

The non-technical interface is provided by the **DESIGN** procedure, which can be used interactively to form experimental designs of several different types. The process involves answering questions, posed by Genstat, first to select the particular type of design, then to give details such as names of factors, numbers of treatments, and so on. Subsidiary procedures are called which depend on the type of design selected. In Release 3[2] of the Procedure Library the following types are available:

• Orthogonal hierarchical designs	**AGHIERARCHICAL**
• Factorial designs (with blocking)	**AGDESIGN**
• Fractional factorial designs (with blocking)	**AGFRACTION**
• Lattice designs	**AGDESIGN**
• Lattice squares	**AGDESIGN**
• Latin squares	**AGDESIGN**
• Cyclic designs	**AGCYCLIC**

If you wish to avoid some of the question-and-answer process, the subsidiary procedures can be called directly. They all have options and parameters to supply the information otherwise obtained by the various questions and, provided you supply *all* the required information, they can also be used in batch.

Four of the design types are handled as instances of *generally balanced designs*, using procedure **AGDESIGN**, and we now discuss in more detail the facilities for designing these *and* providing all the necessary information so that they can later be analysed.

3. Generally Balanced Designs

Generally balanced designs are very widely applicable and have the particular advantage that they can contain more than one *block* (or *error*) term. The total sum of squares can be partitioned up into components known as *strata*, one for each block term. Each stratum contains the sum of squares for the treatment terms estimated between the units of that stratum, and a residual representing the random variability of those units.

One simple example is the split-plot design, in which there are three strata:

```
***** Analysis of variance *****

Variate: Yield of oats in cwt. per acre

Source of variation      d.f.       s.s.       m.s.      v.r.  F pr.

Blocks stratum              5    506.227    101.245      5.28

Blocks.Wplots stratum
Variety                     2     56.963     28.482      1.49  0.272
Residual                   10    191.751     19.175      3.40

Blocks.Wplots.Subplots stratum
Nitrogen                    3    638.409    212.803     37.69  <.001
Variety.Nitrogen            6     10.260      1.710      0.30  0.932
Residual                   45    254.106      5.647

Total                      71   1657.715
```

In Genstat the structure of the design is specified separately from the treatment terms to be estimated. Genstat uses this to determine the strata in the design, and thus the error terms for the analysis. Here we have

```
BLOCKSTRUCTURE  Blocks / Wplots / Subplots
```

where the operator / indicates that a factor is nested within another factor. The model formula expands to give the model terms for the (three) strata

```
Blocks  +  Blocks.Wplots  +  Blocks.Wplots.Subplots
```

The treatment formula here uses the *factorial* operator *

```
TREATMENTSTRUCTURE  Variety * Nitrogen
```

This expands to define the main effects of `Variety` and `Nitrogen`, and their interaction:

```
Variety  +  Nitrogen  +  Variety.Nitrogen
```

Further details of this design can be found in the Genstat 5 Release 3 Reference Manual, pages 484-7.

The properties of a generally balanced design are that

(i) the block (or error) terms are mutually orthogonal,
(ii) the treatment terms are also mutually orthogonal, and
(iii) the contrasts of each treatment term all have equal efficiency factors in each of the strata where they are estimated.

This is very closely related to the *first-order* balance required to avoid `ANOVA` diagnostic `AN 1`, which requires just (i) and (iii); see Payne and Tobias (1992).

Thus, general balance includes all orthogonal designs (completely randomized orthogonal designs, randomized block designs, split plots, Latin squares, Graeco-Latin squares, and so on), and all designs in which there is balanced confounding between treatment and block terms (for example balanced incomplete blocks, square lattices, lattice squares etc).

The plan and analysis-of-variance table below illustrate a more complicated design, a 3^3 factorial in 4 replicates of 3 blocks of 9 plots. Suppose that the treatment factors are `A`, `B` and `C`: in replicate 1 AB^2C^2 is confounded with blocks, in replicate 2 AB^2C is confounded with blocks, in replicate 3 ABC^2 is confounded with blocks, and in replicate 4 `ABC` is confounded with blocks. Every contrast of the interaction `A.B.C` is confounded in one out of the four replicates, and so the design is balanced.

	Block 1	Block 2	Block 3
Replicate 1	A1 B1 C1 A2 B1 C2 A3 B1 C3 A2 B2 C1 A3 B2 C2 A1 B2 C3 A3 B3 C1 A1 B3 C2 A2 B3 C3	A2 B1 C1 A3 B1 C2 A1 B1 C3 A3 B2 C1 A1 B2 C2 A2 B2 C3 A1 B3 C1 A2 B3 C2 A3 B3 C3	A3 B1 C1 A1 B1 C2 A2 B1 C3 A1 B2 C1 A2 B2 C2 A3 B2 C3 A2 B3 C1 A3 B3 C2 A1 B3 C3
Replicate 2	A1 B1 C1 A3 B1 C2 A2 B1 C3 A2 B2 C1 A1 B2 C2 A3 B2 C3 A3 B3 C1 A2 B3 C2 A1 B3 C3	A2 B1 C1 A1 B1 C2 A3 B1 C3 A3 B2 C1 A2 B2 C2 A1 B2 C3 A1 B3 C1 A3 B3 C2 A2 B3 C3	A3 B1 C1 A2 B1 C2 A1 B1 C3 A1 B2 C1 A3 B2 C2 A2 B2 C3 A2 B3 C1 A1 B3 C2 A3 B3 C3
Replicate 3	A1 B1 C1 A2 B1 C2 A3 B1 C3 A3 B2 C1 A1 B2 C2 A2 B2 C3 A2 B3 C1 A3 B3 C2 A1 B3 C3	A2 B1 C1 A3 B1 C2 A1 B1 C3 A1 B2 C1 A2 B2 C2 A3 B2 C3 A3 B3 C1 A1 B3 C2 A2 B3 C3	A3 B1 C1 A1 B1 C2 A2 B1 C3 A2 B2 C1 A3 B2 C2 A1 B2 C3 A1 B3 C1 A2 B3 C2 A3 B3 C3
Replicate 4	A1 B1 C1 A3 B1 C2 A2 B1 C3 A3 B2 C1 A2 B2 C2 A1 B2 C3 A2 B3 C1 A1 B3 C2 A3 B3 C3	A2 B1 C1 A1 B1 C2 A3 B1 C3 A1 B2 C1 A3 B2 C2 A2 B2 C3 A3 B3 C1 A2 B3 C2 A1 B3 C3	A3 B1 C1 A2 B1 C2 A1 B1 C3 A2 B2 C1 A1 B2 C2 A3 B2 C3 A1 B3 C1 A3 B3 C2 A2 B3 C3

```
***** Analysis of variance *****

Source of variation       d.f.      efficiency factor
rep stratum                  3

rep.block stratum
A.B.C                        8            0.25

rep.block.plot stratum
A                            2            1.00
B                            2            1.00
C                            2            1.00
A.B                          4            1.00
A.C                          4            1.00
B.C                          4            1.00
A.B.C                        8            0.75
Residual                    70

Total                      107
```

Designs can also occur in which some treatment terms contain several sets of contrasts, each with their own efficiency factor. These too can be accommodated, by allowing such terms to be specified by several *pseudo* terms, one for each set of contrasts (Payne and Wilkinson, 1977). The design is then balanced with respect to the pseudo terms, and the sums of squares, effects and means for the original terms can be obtained by adding together the information from the appropriate pseudo terms. This is particularly useful in partially confounded designs, where different sets of treatment contrasts may be confounded with the blocks in each replicate. For example, if we included only the first 2 replicates above, we would have

```
***** Analysis of variance *****

Source of variation      d.f.      efficiency factor      pseudo-terms

rep stratum                 1

rep.block stratum
A.B.C                       4                0.50      A B² C²,  A B² C

rep.block.plot stratum
A                           2                1.00
B                           2                1.00
C                           2                1.00
A.B                         4                1.00
A.C                         4                1.00
B.C                         4                1.00
A.B.C                       8   {  4         0.50      A B² C²,  A B² C
                                {  4         1.00      A B C²,   A B C

Residual                   22

Total                      53
```

3.1 Definition of a Generally Balanced Design

Defining a generally balanced design requires
- the block structure formula,
- the block factors (and their values), and
- a means of constructing the values of the treatment factors from the block factors, in such a way as to ensure that the design will exhibit all the required confounding and aliasing properties.

The block-factor values generally occur in an easily-constructed lexicographic order, and the block structure formula is necessary to define the randomization of the design, once the treatment factors have been generated (see Nelder, 1965).

The inter-relationship between the treatment and block factors can be represented very conveniently by a matrix known as the *design key* (Patterson, 1976; Patterson and Bailey, 1978). The construction method requires the factors to have prime numbers of levels, and so the definition may involve defining treatment *pseudo* factors in terms of block *pseudo* factors (usually known as the *plot* factors), and then constructing the original factors from the outer products of the pseudo factors. (So, our plot factor would be need to be represented by two 3-level pseudo factors.). However, these *design* pseudo factors are *not* usually capable of defining the pseudo terms required for the *analysis*. The *design key* indicates how the levels of each treatment factor are to be calculated from the plot factors. In Genstat, the matrix has a row for each treatment factor and a column for each plot factor. (This is the transpose of the form used by Patterson (1976), but in Genstat it seems more convenient to specify the treatments by rows.) There can also be a *base vector* to allow levels of some treatment factors to be permuted cyclically (this is sometimes useful with quantitative factors). We define

$(\beta_j)_u$	to be the value in unit u of *plot factor* j (this is assumed to be an integer in the range 0 upwards)	$j = 1, \ldots, c$
$(\alpha_i)_u$	to be the value generated for unit u of treatment factor i (again as an integer in range 0, 1, ...)	$i = 1, \ldots, m$
t_i	to be the number of levels of treatment factor i	
k_{ij}	to be the value at row i and column j of the *design key*, and	
b_i	to be the value in unit i of the *base vector* (by default b_i=0)	

The value in unit u of treatment factor i is given by

$$(\alpha_i)_u = b_i + k_{i1} \times (\beta_1)_u + k_{i2} \times (\beta_2)_u + \ldots + k_{ic} \times (\beta_c)_u \quad \text{modulo } t_i$$

Essentially, the key identifies each treatment factor i with the set of plot-factor effects

$$\beta_1^{k_{i1}} \quad \beta_2^{k_{i2}} \quad \ldots \quad \beta_c^{k_{ic}}$$

To start with a simple example, the treatments to be allocated (before randomization) to the plots of an $n \times n$ Latin square may be calculated as

Latin-factor-value = Row-factor-value + Column-factor-value modulo n

and values of the extra factor in a Graeco-Latin square can be formed as

Graeco-factor-value = Row-factor-value + 2 × Column-factor-value modulo n

The key is thus $\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$ and it can be used as follows to form a 5 × 5 Graeco-Latin square as follows:

```
FACTOR  [NVALUES=25; LEVELS=!(0...4)] Row,Column,A,B; DECIMALS=0
" specify key matrix (row and column labelling is unnecessary
  other than to indicate how the matrix is stored) "
MATRIX  [ROWS=!t(A,B); COLUMNS=!t(Row,Column); VALUES=1,1, 1,2] GLkey
AKEY    [PRINT=design; BLOCKFACTORS=Row,Column; KEY=GLkey] A,B

*** Treatment combinations on each unit of the design ***

Column   0      1      2      3      4
   Row
     0   0 0    1 2    2 4    3 1    4 3
     1   1 1    2 3    3 0    4 2    0 4
     2   2 2    3 4    4 1    0 3    1 0
     3   3 3    4 0    0 2    1 4    2 1
     4   4 4    0 1    1 3    2 0    3 2

Treatment factors are listed in the order: A B
```

Designs containing sets of factors with several different (prime) numbers of levels can be generated as direct products of designs for each particular prime (procedure **APRODUCT**) or, more conveniently, by forming a design key combining the rows and columns from the keys of all the individual designs.

The design key allows treatment factors to be generated that are completely confounded with block factors. Often, as in the 3^3 example above, there are several ways in which sets of contrasts can be selected to be confounded and, by using different keys for different parts of a design, partially confounded designs can be formed. Partially confounded designs, however, require treatment terms to be partitioned into *pseudo terms* for successful analysis. The factors to generate these terms can be formed by inverting the key matrix (using the field of arithmetic modulo p) to obtain a key for defining the block factors in each version in terms of the treatment factors. The block factors identify the treatment contrasts that are confounded in each such *version*. Thus these keys allow the necessary (*analysis*) pseudo factors to be generated from the values of the treatment factors.

For the 3^3 example above, the keys and inverse keys are as follows:

replicate 1	1 1 1	1 2 2
(**A B²** **C²** confounded with blocks)	0 1 0	0 1 0
	0 0 1	0 0 1
replicate 2	1 1 2	1 2 1
(**A B²** **C** confounded with blocks)	0 1 0	0 1 0
	0 0 1	0 0 1

replicate 3	1 2 1	1 1 2
(**A B C²** confounded with blocks)	0 1 0	0 1 0
	0 0 1	0 0 1
replicate 4	1 2 2	1 1 1
(**A B C** confounded with blocks)	0 1 0	0 1 0
	0 0 1	0 0 1

The necessary information to define the construction and analysis of a generally balanced design can now be illustrated using the 3^3 example as follows.

1)	'3 × 3 × 3 factorial in blocks of size 9'	(description of the design)
2)	2	(number of block factors)
3)	1 2	(number of pseudo factors for each block factor)
4)	3 3 3	(number of levels for each block pseudo factor)
5)	B_1 / B_2	(block-structure formula for the design)
6)	3	(number of treatment factors)
7)	1 1 1	(number of pseudo factors for each treatment factor)
8)	3 3 3	(number of levels for each treatment pseudo factor)
9)	4	(number of different "versions" of the design)
10)	'T_1+2T_2+2T_3 confounded with B_1'	(description of version 1)
	1 1 1	(design key for version 1)
	0 1 0	
	0 0 1	
	'T_1+2T_2+T_3 confounded with B_1'	(description of version 2)
	1 1 2	(design key for version 2)
	0 1 0	
	0 0 1	
	'T_1+T_2+2T_3 confounded with B_1'	(description of version 3)
	1 2 1	(design key for version 3)
	0 1 0	
	0 0 1	
	'T_1+T_2+T_3 confounded with B_1'	(description of version 4)
	1 2 2	(design key for version 4)
	0 1 0	
	0 0 1	
11)	1	(number of analysis pseudo factors per version in treatment formula)
12)	3	(number of levels of each analysis pseudo factor, if any)
13)	1 2 2	(pseudo factor for version 1)
	1 2 1	(pseudo factor for version 2)
	1 1 2	(pseudo factor for version 3)
	1 1 1	(pseudo factor for version 4)
14)	(T_1 * T_2 * T_3) // P_1	(one version only)
	(T_1 * T_2 * T_3) // (P_1 + P_2)	(two different versions)
	(T_1 * T_2 * T_3) // (P_1 + P_2 + P_3)	(three different versions)
	(T_1 * T_2 * T_3)	(all four versions: design balanced)

This specification is used to define the data base for procedure **AGDESIGN** which allows the user to work through a sequence of pop-up menus to select a design, name the various factors, randomize the design, print the design in a tabular representation, and produce a skeleton analysis of variance showing where each treatment term is estimated and the corresponding efficiency factors. Release 3[2] of the Procedure Library includes the following designs via **AGDESIGN**.

Factorial designs (with interactions confounded with blocks)

1 Single replicate of a 2^3 factorial in blocks of size 4
2 Single replicate of a 2^4 factorial in blocks of size 8
3 Single replicate of a 2^6 factorial in blocks of size 16
4 Single replicate of a 2^4 factorial in blocks of size 4
5 Single replicate of a 2^5 factorial in blocks of size 8
6 Single replicate of a 2^6 factorial in blocks of size 8
7 Single replicate of a 3^3 factorial in blocks of size 9
8 Single replicate of a 3^4 factorial in blocks of size 9
9 Three replicates of a $2^2 \times 3$ factorial in blocks of size 6
10 Three replicates of a $2^3 \times 3$ factorial in blocks of size 6
11 Single replicate of a 2×3^2 factorial in blocks of size 6
12 Single replicate of a 4^2 factorial in blocks of size 4
13 Single replicate of a 4×2^2 factorial in blocks of size 8
14 Three replicates of a $4 \times 2 \times 3$ factorial in blocks of size 12
15 Single replicate of a 4×2^3 factorial in blocks of size 8
16 Half replicate of a 4×2^4 factorial in blocks of size 8

Lattice designs: 3×3, 4×4, 5×5, 6×6, 7×7, 8×8, 9×9, 10×10, 11×11, 12×12.

Lattice squares: 3×3, 4×4, 5×5, 7×7, 8×8, 9×9, 11×11, 13×13.

Latin squares: 3×3, 4×4, 5×5, 6×6, 7×7, 8×8, 9×9, 10×10, 11×11, 12×12.

The information is contained in standard Genstat data structures and stored in a backing-store file. **AGDESIGN** forms the menus to list the choices by collating the contents of the file, and a procedure **FDESIGNFILE** is provided to construct new files or to modify the existing file. The system thus allows users to add new designs as required by their own working environments.

4. Alpha Designs

Alpha designs are a very flexible class of resolvable incomplete block designs. (A resolvable design is one in which each block contains only a selection of the treatments, but the blocks can be grouped together into subsets in which each treatment is replicated once.) The groupings of blocks thus form replicates, and the block structure of the design is

```
Replicates / Blocks / Units
```

Such designs are particularly useful when there are many treatments to examine and the variability of the units is such that the block size needs to be kept small. Alpha designs were thus devised originally for the analysis of plant breeding trials (Patterson and Williams, 1976), where many varieties may need to be evaluated in a single trial, and have the advantage that they can provide effective designs for any number of treatments.

The construction of an alpha design requires a $k \times r$ generating array α of integers between 0 and $s-1$, where r is the number of replicates, and s is the number of blocks per replicate. If the number of treatments, v, is a multiple of the number of blocks per replicate, k will be the number of units in each block, and v will be given by $s \times k$. Otherwise, the design will have some blocks of size k and some of size $k-1$, and v will lie between $s \times (k-1)$ and $s \times k$.

Given the array α, procedure **AFALPHA** can be used to generate the design. The treatment values for replicate q of the design are obtained from column c_q of the array α by the following operations.

1) form $s-1$ further columns: column j is given by $c_q + j - 1$ modulo s
2) add $(i-1) \times s$ to each row i
3) if $n < s \times k$ delete units that have been allocated treatments $n+1 \ldots s \times k$.

For example

```
MATRIX  [ROWS=5; COLUMNS=3; \
  VALUES=0,0,0, 0,1,2, 0,2,3, 0,3,1, 0,3,2] Array
```

```
AFALPHA [PRINT=design] Array; TREATMENTS=Treat; REPLICATES=Rep; \
  BLOCKS=Block; UNITS=Plot
AFALPHA [PRINT=design] Array; LEVELS=!(0...18) SEED=274903
```

would generate the treatments as follows

```
Array      column     1      2      3                                  (k=5  r=3  s=4)

  row      1          0      0      0
           2          0      1      2
           3          0      2      3
           4          0      3      1
           5          0      3      2


  -> add   0   1   2   3      0   1   2   3      0   1   2   3   mod(4)


  Rep      1                  2                  3
Block      1   2   3   4      1   2   3   4      1   2   3   4
Plot                                                      add
   1       0   1   2   3      0   1   2   3      0   1   2   3     0
   2       4   5   6   7      5   6   7   4      6   7   4   5     1 × 4
   3       8   9  10  11     10  11   8   9     11   8   9  10     2 × 4
   4      12  13  14  15     15  12  13  14     13  14  15  12     3 × 4
   5      16  17  18  19     19  16  17  18     18  19  16  17     4 × 4
```

Clearly, the properties of the design that is formed will be very dependent on the choice of array. Procedure **AGDESIGN** allows alpha designs to be formed using standard generators taken (via **DSIGNX**) from Patterson, Williams and Hunter (1978) and Williams (1975) which provide 2, 3 or 4 replicates for $k \leq s$, and 2 replicates for $k > s$.

5. Cyclic Designs

The cyclic method is a very powerful way of constructing incomplete block designs. In its simplest form, it starts with an initial block, containing some subset of the treatments. This subset is then represented by the ordinal number in the range 0, ... , m–1 where m is the number of treatment levels. The second and subsequent blocks are then generated by successively addition modulo m of one to the numbers in the subset.

Thus, for seven treatments (0, ... , 6) and an initial block (0,1,4), the design would be

0	1	2	3	4	5	6
1	2	3	4	5	6	0
4	5	6	0	1	2	3

As can be seen, if m is a prime number, m blocks are generated with each initial block. However, if m can be expressed as the product of other integers, shorter cycles can occur. For example, for m=8 and initial block (0,1,4,5), 4 blocks are generated altogether:

0	1	2	3
1	2	3	4
4	5	6	7
5	6	7	0

It is also possible to have more than one initial block, and the increment need not be one.

Procedure **AFCYCLIC** allows cyclic designs to be generated given the necessary generators (that is, the initial blocks), while procedure **AGCYCLIC** provides a range of standard generators for cyclic designs, cyclic change-over designs and cyclic superimposed designs. Cyclic change-over designs (Davis and Hall, 1969) are used for trials in which subjects are given different treatments in different time periods; these thus have a crossed block structure **subjects*periods**. Cyclic superimposed designs (Hall and Williams, 1973) have two treatment factors (each with the same number of levels); the design is intended to estimate their main effects but not their

interaction. **AGCYCLIC** again uses standard generators taken from **DSIGNX**, but this time the information is obtained from a backing-store file, instead of being programmed into the procedure. The backing-store file is formed using an as-yet-unreleased procedure **FCYCLICFILE**, details of which can be obtained from the author.

6. Fractional factorials

Fractional factorial designs can be generated using procedure **AGFRACTION**. This again uses standard design keys from **DSIGNX**, and the information is stored a backing-store file, formed using another as-yet-unreleased procedure **FFRACTIONFILE**. For factors with 2 levels, the fractions currently available are 1/2 (with 3-12 treatment factors), 1/4 (4-12 treatment factors), 1/8 (5-14 treatment factors), 1/16 (6-15 treatment factors), 1/32 (6-16 treatment factors), 1/64 (7-16 treatment factors). For factors with 3 levels the available fractions are 1/3 (2-7 treatment factors), 1/9 (3-8 treatment factors), 1/27 (4-9 treatment factors), 1/81 (5-10 treatment factors), 1/243 (6-11 treatment factors).

7. Orthogonal Hierarchical Designs

Procedure **AGHIERARCHICAL** generates orthogonal hierarchical designs. The details of the required strata and the treatments in each one are supplied by the parameters of the procedure, and there is no limitation on the size or the complexity of the design.

8. Conclusion

The Genstat design system provides an open and very flexible set of facilities for the selection and generation of experimental designs. There is also a wide range of facilities for the display and manipulation of designs, with procedures to allow designs to be formed as the outer product of other designs or by "adding" designs together, as well as procedures for plotting an experimental plan and generating data forms for the experimenter.

References

Davis A W and Hall W B (1969) Cyclic change-over designs. *Biometrika* **56** 283-293.

Franklin M F and Mann A D (1986) *DSIGNX a program for the construction of randomized experimental plans.* Scottish Agricultural Statistics Service, Edinburgh (revised edition).

Hall W B and Williams E R (1973) Cyclic superimposed designs. *Biometrika* **60** 47-53.

Nelder J A (1965) The analysis of randomised experiments with orthogonal block structure. I. Block structure and the null analysis of variance. II. Treatment structure and the general analysis of variance. *Proceedings of the Royal Society of London* **A283** 147-178.

Patterson H D (1976) Generation of factorial designs. *J. R. Statist. Soc. B* **38** 175-179.

Patterson H D and Bailey R A (1978) Design keys for factorial experiments. *Applied Statistics* **27** 335-343.

Patterson H D and Williams E R (1976) A new class of resolvable incomplete block designs. *Biometrika* **63** 83-92.

Patterson H D, Williams E R and Hunter E A (1978) Block designs for variety trials. *J. Agric. Sci* **90** 395-400.

Payne R W, Arnold G M and Morgan G W (ed) (1993) *Genstat 5 Procedure Library Manual Release 3[1].* Numerical Algorithms Group, Oxford.

Payne R W and Franklin M F (1994) Data structures and algorithms for an open system to design and analyse generally balanced designs. In: *COMPSTAT 94 Proceedings in Computational Statistics* (ed. R Dutter and W Grossmann) pp 429-434 Physica-Verlag, Hiedelberg

Payne R W, Lane P W, Digby P G N, Harding S A, Leech P K, Morgan G W, Todd A D, Thompson R, Tunnicliffe Wilson G, Welham S J and White R P (1993) *Genstat 5 Reference Manual, Release 3.* Oxford: Oxford University Press.

Payne R W and Wilkinson G N (1977) A general algorithm for analysis of variance. *Applied Statistics* **26** 251-260.

Williams E R (1975) A new class of resolvable block designs. PhD Thesis. University of Edinburgh.

Computing the generalized estimating equations with quadratic covariance estimation for repeated measurements

M G Kenward
IACR-Rothamsted
Harpenden
Herts AL5 2JQ, UK

D M Smith
Department of Mathematics
Statistics and Computing Science
The University of New England
Armidale NSW 2351, Australia

Summary

The application of general estimating equation methodology to the analysis of repeated measurement data is reviewed. The implementation of this methodology into Genstat is described. Estimation of the correlation matrix for various models is reviewed. Examples involving a variety of types of data and models are given.

1. Introduction

Much experimental and observational research involves the collection of sequences of observations from each unit, for example, person, animal or system. Such repeated measurements experiments are common in agriculture, medicine and psychology and the resulting data present special problems for statistical analysis. Typically the sequence of observations from a unit will be statistically dependent with a potentially awkward and possibly non-stationary covariance structure. The modelling of *non-Gaussian* repeated measurements data presents an added difficulty in that there is no natural generalization of the multivariate Gaussian distribution for such data. The main implication of this is that, in general, it is not possible to separate the modelling of the first and second moments. An important first step in the modelling of such data is therefore to decide how the model will be constructed in terms of the moments of the joint distribution of the data. For a discussion of the alternatives see Diggle *et al* (1994, chapter 7). Here we consider the so-called *marginal* model (Liang and Zeger, 1986). A model is constructed for the marginal expectation of the observations at each time point and the parameters of this model are assumed to represent quantities, such as treatment effects or regression relationships, that are of sustantive interest. The model in this form says nothing directly about the dependence among the repeated measurements but, because of the relationships among first- and higher-order moments in non-Gaussian distributions, it will typically have implications for these higher-order moments. These relationships also mean that it is only in special cases that a comparatively simple *joint* distribution of the observations can be constructed and so full likelihood analyses for marginal models are typically awkward.

A practically valuable method for fitting such marginal models to data that avoids the construction of a full likelihood is based on the use of so-called *generalized estimating equations* (GEE), see for example Liang and Zeger (1986), Zeger and Liang, (1986) and Liang *et al* (1992). This approach can be used with only a comparatively simple modification of the methodology of generalized linear models although, since its introduction to a biometric audience, the basic technique has undergone considerable refinement. In particular, in its original form, estimating equations were used only for the marginal model and simple empirical methods were used to obtain measures of precision. We employ this so-called GEE1 method here, with some modification to allow structure to be imposed on the correlation matrix of the data. The modification is essentially an extension of the 'Gaussian estimation' procedure of Crowder (1985), recalled in the present context by Crowder (1992). It has the additional advantage that when applied to the Gaussian setting it leads to full maximum likelihood. Further refinements that lead to the so-called GEE2 method are discussed briefly at the end of this paper.

The GEE methodology is ideally suited to implementing in statistical packages such as Genstat in which the

necessary basic routines already exist but which also have the facilties to manipulate data and so permit the modifications required for repeated measurements. This article describes a procedure that has been written to implement the GEE1 methodology with quadratic estimation for the correlation matrix.

While GEE methodology for marginal generalized linear models is now well established the same is not as true for categorical, and in particular ordinal, repeated measurements. The GEE methodology can be extended for such settings and we describe in a Genstat Newsletter article (Kenward and Smith, 1995) how the current procedure can be used for this through the representation of a categorical observation as a set of correlated binary observations, a suggestion originally of Clayton (1992) and developed in Kenward *et al* (1994).

2. Computing the GEE

Suppose that we have n independent experimental units all *potentially* observed at the same set of q times. In practice we may well have unbalanced sets of data, whether by design or chance, due to units being observed only at a subset of these times. The observation from time j from unit i, Y_{ij}, say, is assumed to have a *marginal* representation in terms of the mean and variance associated with a generalized linear model:

$$E[Y_{ij}] = \mu_{ij}$$
$$\eta_{ij} = X_{ij}'\beta = g(\mu_{ij})$$

where X_{ij} is the $(p \times 1)$ vector of explanatory variables, β the corresponding $(p \times 1)$ vector of unknown parameters and $g(\cdot)$ is the link function, and

$$V[Y_{ij}] = \phi_j v_{ij} ,$$

where v_{ij} is a function of the mean, and hence of the parameters β, sometimes called the variance function, and ϕ_j is the scale factor *for time point j*. The scale factor may be allowed to vary across time, but is assumed to be constant across units.

If it is assumed that the observations from each unit are independent we can express the conventional estimating equations for β as

$$\sum_{i=1}^{n} X_i' D_i \Phi_i^{-1} V_i^{-1} (Y_i - \mu_i) = 0 \qquad (1)$$

where Y_i, X_i and μ_i are assembled in an obvious way from Y_i, X_i and μ_{ij}, $V_i = \text{diag}\{v_{ij}\}$, $\Phi = \text{diag}\{\phi_{ij}\}$, and

$$D_i = \text{diag}\left(\frac{\partial \mu_{ij}}{\partial \eta_{ij}}\right).$$

The index i on Φ_i is present to account for possible lack of balance among the repeated measurements not to indicate differences in the ϕ_{ij} among units.

The equation (1) can be solved for β (to produce $\hat{\beta}$, say) in a straightforward way, assuming that the scale factor(s) are known, using iterative weighted least squares (see for example McCullagh and Nelder, 1989, section 2.5) and the solution is a consistent one even when the observations from a unit are not in fact dependent, as will typically be the case with repeated measurements. In an attempt to improve the efficiency of the method of estimation a *working correlation matrix R* can be introduced to account for the within-unit dependence. The resulting equations,

$$\sum_{i=1}^{n} X_i' D_i \Phi_i^{-1/2} V_i^{-1/2} R^{-1} V_i^{-1/2} \Phi_i^{-1/2} (Y_i - \mu_i) = 0 \qquad (2)$$

can be solved in much the same way. The working correlation matrix R can be chosen in a variety of ways. It can be a simple fixed matrix, possibly based on previous analyses. Alternatively it may be calculated from the data and updated at each cycle. In this latter case the consistency of the resulting estimating method is not as

clear-cut as in the Gaussian setting.

Although, in most settings, the equations can be assumed to produce consistent estimates of β, the usual generalized linear model estimates of error will not be appropriate. If the matrix R is either fixed at the true value, or estimated in a consistent way as part of the algorithm then a simple modification of the "independence" based estimates of error can be used, the so-called *naive* estimate of the variance-covariance matrix of $\hat{\beta}$:

$$\hat{V}_N[\hat{\beta}] = \hat{W}_N = \left[\sum_{i=1}^{n} X_i' D_i \Phi_i^{-1/2} V_i^{-1/2} R V_i^{-1/2} \Phi_i^{-1/2} D_i X_i \right]^{-1}$$

in which quantities are replaced by their estimated values where appropriate.

However it is often the case that R is known to be incorrect, for example when the common choice of $R = I$ is used. In this situation the estimate of error must be adjusted to take account of the actual dependence among the repeated measurements. This can be done through the so-called *sandwich* estimator of the variance-covariance matrix of β, sometimes called a robust estimator, in the limited sense that it is robust to departures from the assumed correlation structure. This takes the form

$$\hat{V}_N[\hat{\beta}] = \hat{W}_N \left[\sum_{i=1}^{n} X_i' D_i \Phi_i^{-1/2} V_i^{-1/2} R V_i^{-1/2} \Phi_i^{1/2} (Y_i - \mu_i)(Y_i - \mu_i)' \Phi_i^{-1/2} V_i^{-1/2} R V_i^{-1/2} \Phi_i^{-1/2} D_i X_i \right] \hat{W}_N .$$

3. Quadratic estimation of the correlation structure and scale factor

We define the scaled residuals for the ith unit as

$$r_i = V_i^{-1/2}(Y_i - \hat{\mu}_i) .$$

If we assume that these are approximated by a $N(0, \Sigma_i)$ distribution, where $\Sigma_i = \Phi_i^{-1/2} R \Phi_i^{-1/2}$ then the appropriate estimating equations for Σ, a function of parameters $\sigma = (\sigma_1, \ldots \sigma_r)'$ say, can be expressed as

$$\sum_{i=1}^{n} \frac{\partial \Sigma_i^{-1}}{\partial \sigma_j} (\Sigma_i - r_i r_i') , \quad j = 1 \ldots r .$$

These equations can solved using a simple iterative scheme. Define D_i to be the $q \times q_i$ matrix obtained by deleting from the identity matrix the columns corresponding to times for which subject i is *not* observed. Suppose that we have an existing estimate $\Sigma_{(l)} = \Sigma(\sigma_{(l)})$. Updated estimates, $\sigma_{(l+1)}$ are obtained by solving the equations

$$\sum_{i=1}^{n} \frac{\partial \Sigma^{-1}}{\partial \sigma_j} (\Sigma - S_i) , \quad j = 1 \ldots r \tag{3}$$

where

$$S_i = \Sigma_{(l)} - F_i' (D_i' \Sigma_{(l)} D_i - r_i r_i') F_i \tag{4}$$

for

$$F_i = (D_i' \Sigma_{(l)} D_i)^{-1} D_i \Sigma_{(l)} .$$

If the r_i are exactly normally distributed, or if they arise from the estimating equations for normally distributed data, then this iterative procedure corresponds to the EM algorithm (Dempster *et al*, 1977) for the full likelihood solution. In the normal case we might also consider using the REML estimates instead (Patterson and Thompson, 1971) and the equations for this are obtained by substituting

$$r_i r_i' + X_i \left(\sum_{i=1}^{N} X_i' D_i' (D_i' \Sigma_{(l)} D_i)^{-1} D_i X_i \right)^{-1} X_i'$$

for $r_i r_i'$ in (4). For balanced data, that is, when measurements exist for all units on all occasions, (4) reduces to

$$\Sigma_{(l+1)} = \frac{1}{n}\sum_{i=1}^{n} r_i r_i'$$

as should be expected and no iteration is required. Denote the solution by $\hat{\Sigma}$.

Up to this point we have ignored the structure in Σ. We need to be able to separate out the estimation of R from that of Φ, at least in the non-normal case, because one or other of the two quantities may be fixed. This amounts to the problems of estimating a set of variances with known correlation structure; and estimating a correlation structure with known variances. These problems do not seem to have been widely explored, although Styan (1968) provides a detailed investigation of the former.

3.1 Estimating the scale factors with fixed correlation matrix

With some manipulation in can be shown that the solution of the equations (3) reduces in the case of time constant scale factor (ϕ say) to

$$\hat{\phi} = \frac{1}{q} tr(R^{-1}\hat{\Sigma})$$

and for time varying scale factors to the solution of the matrix equation

$$\text{diag}\ (\phi^{-1/2} R^{-1} \phi^{-1/2} \hat{\Sigma}) = j_q$$

for j_q a $(q \times 1)$ vector of 1s. Setting $z = \text{diag}(\Phi^{-1/2})$ this can be expressed as

$$zz' * Hj_q - j_q = 0_q$$

where $H = R^{-1} * \hat{\Sigma}$, 0_q is a $(q \times 1)$ vector of zeros and * denotes the Hadamard product. This matrix equation can be solved iteratively for z using Newton's method for a system of equations (Henrici, 1964).

3.2 Estimating the correlation matrix with fixed scale

It can be shown through straightforward manipulation of (3) that we can obtain the appropriate estimate of α, the parameters of R, by fitting the correlation structure to the matrix $\hat{R}$ that is the solution of the matrix equation

$$\text{lower triangle of } (\hat{R}^{-1} - \hat{R}^{-1}\phi^{-1/2}\hat{\Sigma}\phi^{-1/2}\hat{R}^{-1}) = \text{lower triangle}(0)$$

subject to the constraint $\text{diag}(\hat{R}) = I$.

This can be solved for $\hat{R}$ using an iterative procedure with starting value

$$\hat{R}_0 = \Phi^{-1/2}\hat{\Sigma}\Phi^{-1/2}$$

standardised as a correlation matrix,

$$\Lambda_k = \text{diag}\left[F_K^{-1}\left\{j_q - \text{vec.diag}(\Phi^{-1/2}\hat{\Sigma}\Phi^{-1/2})\right\}\right]$$

and

$$\hat{R}_{k+1} = (\Phi^{-1/2}\hat{\Sigma}\Phi^{-1/2} + \hat{R}_k\Lambda_k\hat{R}_k)$$

where

$$\{F_k\}_{ij} = (\{\hat{R}\}_{ij})^2 .$$

There follows the various correlation structures included in the Genstat procedure together with the corresponding method for estimating α from $\hat{R}$. Except for a matrix involved with the ante-dependence correlation structure all the correlation (covariance) matrices are symmetric, therefore only the lower triangle is displayed.

1. Independence structure

$$\hat{R}_{ind} = R_{ind} = I_q \ .$$

2. *Unconstrained structure*

$$\hat{R}_{uns} = \hat{R} \ .$$

3. Uniform (exchangeable) structure

For the uniform structure

$$R_{uni} = \begin{bmatrix} 1 & & & & \\ \alpha & 1 & & & \\ \alpha & \alpha & 1 & & \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \alpha & 1 \end{bmatrix}$$

the parameter α is estimated as the average of the off diagonal elements of $\hat{R}$.

4. Autoregressive structure.

The autoregressive structure is

$$R_{ar} = \begin{bmatrix} 1 & & & & \\ \alpha & 1 & & & \\ \alpha^2 & \alpha & 1 & & \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \alpha^{q-1} & \ldots & \ldots & \alpha & 1 \end{bmatrix}$$

i.e. the correlation between the responses at times t_j and t_k is $\alpha^{|tj-tk|}$. To accommodate unequally spaced repeated measurements the *ad hoc* method of estimation for α as described in Liang and Zeger (1986) is used. It is estimated from the regression of the logarithm of the off-diagonal elements of $\hat{R}$ on $|t_j - t_k|$. The relationship between these two items can be expressed

$$\ln(R_{j,k}) = |t_j - t_k| \ln(\alpha) \ .$$

This is the correlation structure of a first-order autoregressive process. Correlation structures of higher-order processes have not been incorporated into the procedure because it is not clear how the correlations should be defined when the time points are unequally spaced. When the time points are equally spaced however estimation of the autoregressive parameters is straightforward and these cases may be added at a later date.

5. Ante-dependence structure

The analysis of repeated measurements under an ante-dependence covariance structure is described in Kenward (1987).

The structure can be derived in several ways, for example by generalizing the stationary autoregressive structure. An intuitively appealing derivation is in terms of conditonal independence. An ante-dependence covariance structure of order r, an AD(r) structure say, implies that two repeated measurements at least $r + 1$ steps apart in time are conditionally independent given the intervening measurements. The structure does not imply

stationarity and can be shown to provide a good fit to a wide range of repeated measurements covariance structures, at the expense of depending on $O(q)$ parameters. It has been implemented only for normal data and this means that it has not been necessary to separate the estimation of the scale factors and R, that is, the entire covariance matrix Σ is estimated directly.

The estimate of the ante-dependence covariance structure can be shown to be

$$\hat{\Sigma}_{ad} = \hat{H}\hat{\Lambda}\hat{H}'$$

where $\hat{\Lambda}$ is a $(q \times q)$ diagonal matrix and (for example for an AD(3) structure)

$$\hat{H}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \dots & \dots \\ -\hat{\eta}_{21} & 1 & 0 & 0 & 0 & \dots & \dots \\ -\hat{\eta}_{31} & -\hat{\eta}_{32} & 1 & 0 & 0 & \dots & \dots \\ -\hat{\eta}_{41} & -\hat{\eta}_{42} & -\hat{\eta}_{43} & 1 & 0 & \dots & \dots \\ 0 & -\hat{\eta}_{52} & -\hat{\eta}_{53} & -\hat{\eta}_{54} & 1 & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & 0 & -\hat{\eta}_{t,t-3} & -\hat{\eta}_{t,t-2} & -\hat{\eta}_{t,t-1} & 1 \end{bmatrix}$$

Let c_i^T represent the $(i\text{-}1)$ elements to the left of the diagonal (i, i)th element of Σ_i and C_i represent the $(i\text{-}1) \times (i\text{-}1)$ matrix of the elements of $\hat{\Sigma}$ immediately above c_i. So $c_2 = -\{\hat{\Sigma}\}_{2,1}$, $C_2 = (1)$; $c_3 = (-\{\hat{\Sigma}\}_{3,1}$, $-\{\hat{\Sigma}\}_{3,2})$,

$$C_3 = \begin{bmatrix} 1 & \\ \{\hat{\Sigma}\}_{2,1} & 1 \end{bmatrix};$$

$$c_4 = (-\{\hat{\Sigma}\}_{4,1}\ ,\ -\{\hat{\Sigma}\}_{4,2}\ ,\ -\{\hat{\Sigma}\}_{4,3})\ ,$$

and

$$C_4 = \begin{bmatrix} 1 & & \\ \{\hat{\Sigma}\}_{2,1} & 1 & \\ \{\hat{\Sigma}\}_{3,1} & \{\hat{\Sigma}\}_{3,2} & 1 \end{bmatrix}.$$

It can be shown that for $i > 1$

$$\hat{\eta}_i^T = c_i^T C_i^{-1}$$

$$\text{and that } \hat{\lambda}_i = c_{ii} - \hat{\eta}_i^T c_i.$$

where $\hat{\eta}_i = (\hat{\eta}_{i,i-3}\ ,\ \dots\ ,\ \hat{\eta}_{i,i-1})$ and $\hat{\Lambda} = (\hat{\lambda}_1, \dots, \hat{\lambda}_t)$. The parameter $\hat{\lambda}_1$ is set equal to $\{\hat{\Sigma}\}_{1,1}$ (the variance at the first time point).

6. Dependence structure

For the dependence structure of order 2

$$R_{dep} = \begin{bmatrix} 1 & & & & \\ \alpha_1 & 1 & & & \\ \alpha_2 & \alpha_1 & 1 & & \\ 0 & \alpha_2 & \alpha_1 & 1 & \\ \dots & 0 & \alpha_2 & \alpha_1 & 1 \end{bmatrix}.$$

The parameters α_1, α_2 , ... , are estimated as in an obvious way as the averages of the appropriate diagonal elements of $\hat{R}$.

For this correlation structure there is an option in the procedure (**TIMEDEPENDENT**) that allows α_1, α_2 etc. to take different values at the different time points. These time-varying αs are estimated by the appropriate elements

of $\hat{R}$. This structure is defined simply by setting the appropriate corner elements of R to zero. The matrix with time-varying dependent structure of order r is precisely the inverse of the AD(r) matrix and it can be seen from this, and from experience, that the dependence structure has much less appeal in practice than the AD structure.

4. Implementation in Genstat

The procedure as implemented has thirteen options and eleven parameters. Most of these have obvious uses. Two that may not are the **OUTCOME** option and the **WEIGHT** parameter. The **OUTCOME** option was set up to enable data to be input and analysed by outcome rather than subject, there being a variate **COUNT** (i.e. the number of subjects) associated with outcome. It is primarily for use with binary and count (Poisson) data. Where the data comprise multiple subjects with the same outcome its use considerably reduces the time the procedure takes to run. Its use is illustrated in Example 2. The **WEIGHT** parameter allows a weight to be associated with each observation. This enables heterogeneity of variances across the data to be handled.

Not every possible combination of type of scalefactor (fixed, constant, time-varying) and correlation/covariance model is sensible. Details of permitted combinations and other constraints are presented in Table 1.

Correlation/covariance model	Type of scale factor	Model allowed	Notes
User defined	Fixed Constant Varying over time	Yes Yes Yes	Care is needed in using this model
Independence	Fixed Constant Varying over time	Yes Yes Yes	
Unstructured	Fixed Constant Varying over time	No No Yes	
Exchangeable	Fixed Constant Varying over time	Yes Yes Yes	
Autoregressive	Fixed Constant Varying over time	Yes Yes No	Only AR(1) has been implemented.
Dependence	Fixed Constant Varying over time	Yes Yes No	Order of dependence can be specified. Also time varying dependence is allowed.
Ante-dependence	Fixed Constant Varying over time	No No Yes	Order of dependence can be specified.

Table 1: Implementation of the covariance structures

The procedure also allows for different link functions (including user defined) and different error distributions (including user defined). Use of the **TIMEDEPENDENT=yes** option with **ORDER=ntimes-1** and **SCALEFACTOR=varytime** for a dependence structure is equivalent to unstructured. This provides a means of obtaining 'unstructured' correlation matrices with a fixed or constant scale factor.

5. Examples

5.1 Example 1

The hip replacement data of page 79 of Crowder and Hand (1990).

Input program:

```
PRINT [IPRINT=*; SERIAL=Y; SQUASH=Y] \
        'Data from Crowder and Hand, 1990, p79.'
UNITS [NVALUES=120]
READ Hema
47.10 31.05    *   32.80
44.10 31.50    *   37.00
39.70 33.70    *   24.50
43.30 18.35    *   36.60
37.40 32.25    *   29.05
45.70 35.50    *   39.80
44.90 34.10    *   32.05
42.90 32.05    *     *
46.05 28.80    *   37.80
42.10 34.40 34.00 36.05
38.25 29.40 32.85 30.50
43.00 33.70 34.10 36.65
37.80 26.60 26.70 30.60
37.25 26.50    *   38.45
  *   27.95    *   33.95
27.00 32.50    *   31.95
38.35 32.30    *   37.90
38.80 32.55    *   26.85
44.65 32.25    *   34.20
38.00 27.10    *   37.85
34.00 23.20    *   25.95
44.80 37.20    *   29.70
45.95 29.10    *   26.70
41.85 31.95 37.15 37.60
38.00 31.65 38.40 35.70
42.20 34.00 32.90 33.25
39.70 33.45 26.60 32.65
37.50 28.20 28.80 30.30
34.55 30.95 30.60 28.75
35.50 24.70 28.10 29.75
 :
VARIATE [NVALUES=4] Time; VALUES=!(1,2,3,4)
 &      [NVALUES=120] Age; VALUES=!(4(66,70,44,70,74,65,54,63,71,68,\
        69,64,70,60,52,52,75,72,54,71,58,77,66,53,74,78,74,79,71,68))
FACTOR [LEVELS=30] Patient; VALUES=!(4(1...30))
 &     [LEVELS=Time] Occasion; VALUES=!((#Time)30)
 &     [LEVELS=2] Sex; VALUES=!(52(1),68(2))
PRINT [IPRINT=*; SERIAL=Y; SQUASH=Y] \
      'Identity link : normal error'
GEE   [LINK=IDENTITY; DISTRIBUTION=NORMAL; CRTYPE=UNSTRUCTURED; \
       TERMS=Occasion + Sex + Age] SUBJECT=Patient; TIME=Time; Y=Hema
```

Output:

```
 Data from Crowder and Hand, 1990, p79.

 Identity link : normal error

***** Regression Analysis *****

 Response variate: workvar
   Weight variate: weight
     Fitted terms: Constant + Occasion + Sex + Age

*** Summary of analysis ***

              d.f.         s.s.         m.s.      v.r.
Regression       5       1674.       334.77     19.10
Residual        93       1630.        17.53
Total           98       3304.        33.71
```

```
Percentage variance accounted for 48.0

*** Estimates of regression coefficients ***

                          estimate         s.e.         t
Constant                     40.10         3.36     11.95
Occasion 2                   -9.76         1.09     -8.95
Occasion 3                   -8.44         1.49     -5.65
Occasion 4                   -7.37         1.10     -6.70
Sex 2                       -1.703        0.857     -1.99
Age                         0.0179       0.0493      0.36

*** Correlations ***

estimate                  ref    correlations

Constant                    1     1.000
Occasion 2                  2    -0.186  1.000
Occasion 3                  3    -0.015  0.370  1.000
Occasion 4                  4    -0.179  0.505  0.367  1.000
Sex 2                       5    -0.035 -0.014 -0.037 -0.029  1.000
Age                         6    -0.963  0.023 -0.103  0.020 -0.108  1.000
                                    1      2      3      4      5      6

 Unstructured covariance structure.

 Matrix of covariances

   1       18.11
   2        3.83       16.76
   3       -2.76        1.61       35.44
   4        4.58        0.94       20.79       18.18
               1           2           3           4

 *** Model estimates of s.e.***
    Estimate        s.e.
       39.67       3.907
       -9.75       0.952
       -8.45       1.403
       -7.36       0.951
       -1.82       1.032
        0.03       0.058

 *** Correlations ***
   1      1.0000
   2     -0.1279   1.0000
   3     -0.1269   0.4654   1.0000
   4     -0.1214   0.3917   0.9265   1.0000
   5     -0.0683   0.0000   0.0000   0.0000   1.0000
   6     -0.9686   0.0000   0.0000   0.0000  -0.0829   1.0000
               1        2        3        4        5        6

 *** Sandwich estimates of s.e.***
    Estimate        s.e.
       39.67       3.441
       -9.75       0.926
       -8.45       0.785
       -7.36       0.915
       -1.82       0.951
        0.03       0.051

 *** Correlations ***
   1      1.0000
   2     -0.4374   1.0000
   3     -0.3888   0.6364   1.0000
   4     -0.1559   0.3833   0.6557   1.0000
   5      0.2312  -0.1751  -0.1111  -0.1398   1.0000
   6     -0.9691   0.3353   0.2330   0.0462  -0.3660   1.0000
               1        2        3        4        5        6
```

5.2 Example 2

Data of Example 2 of Jones and Kenward (1987), also discussed on pages 154-158 of Diggle *et al* (1994). These data are binary and this example illustrates use of the parameters OUTCOME and COUNT.

Input program:

```
PRINT [IPRINT=*; SERIAL=Y; SQUASH=Y] \
  'Jones and Kenward (1987) and Diggle et al (pages 155 to 158, 1994).'
FACTOR [NVALUES=90;LEVELS=30] Outcome; VALUES=!(3(1...30))
 &     [NVALUES=90;LEVELS=6] Seq; VALUES=!(15(1),12(2),18(3),15(4),15(5),15(6))
 &     [NVALUES=90;LEVELS=3] Period; VALUES=!((1...3)30)
 &     [NVALUES=90;LABELS=!T(A,B,C)] Trt; VALUES=!T((A,B,C)5, \
       (A,C,B)4,(B,A,C)6,(B,C,A)5,(C,A,B)5,(C,B,A)5)
VARIATE [NVALUES=90] r,n,Count; VALUES= \
 !( 0,1,3(0),3(1),2(0),5(1),3(0),1,3(0),6(1),3(0),1,3(0),2(1),0,1,0,6(1),\
    3(0),1,3(0),3(1),0,3(1),3(0),2(1),(0,1)3,4(1),3(0),1,2(0),2(1),(0,1)3,1 ), \
   !( 90(1) ), \
   !( 3(2,2,1,9,1,2,1,9,4,3(1),8,3,4(1),8,1,3,1,7,2,2(1),5,4,3,1) )
 &      [NVALUES=3] Atime; VALUES=!(1...3)
FACTOR [NVALUES=90;LEVELS=!(0,1)] x1,x2,x3,x4,x5,x6; VALUES=4(*), \
  !( (0,0,1)5,(0,0,0)4,(0,1,0)6,(0,1,0)5,(0,0,0)5,(0,0,1)5 ), \
  !( (0,0,0)5,(0,0,1)4,(0,0,0)6,(0,0,1)5,(0,1,0)5,(0,1,0)5 )
CALCULATE x1 = (Period.EQ.2) & x2 = (Period.EQ.3)
 &        x3 = (Trt.EQ.2)    & x4 = (Trt.EQ.3)

PRINT [IPRINT=*; SERIAL=Y; SQUASH=Y] \
      'Logit link : binomal error'
GEE   [LINK=LOGIT; DISTRIBUTION=BINOMIAL; CRTYPE=EXCHANGEABLE; \
      TERMS=x1+x2+x3+x4+x5+x6] OUTCOME=Outcome; COUNT=Count; Y=r;  \
      TIME=Atime; NBINOMIAL=n
```

Output:

```
 Jones and Kenward (1987) and Diggle et al (pages 155 to 158, 1994).

 Logit link : binomal error

 The OUTCOME option has really been set up for use with count (Poisson) and binomial
 data where rather than inputting individual subject data, outcomes are input with the
 number of subjects with each outcome input as a count variate.
 Use of OUTCOME and COUNT is much faster.
 WHETHER THE DISTRIBUTION IS POISSON OR BINOMIAL IS NOT CHECKED.
 This enables, for example, overdispersion to be handled by use of the own
 DISTRIBUTION option and/or weights.

***** Regression Analysis *****

 Response variate: workvar
   Weight variate: weight
     Fitted terms: Constant + x1 + x2 + x3 + x4 + x5 + x6

*** Summary of analysis ***
    Dispersion parameter is 1

             d.f.        s.s.        m.s.      v.r.
Regression      6        54.6       9.097      2.91
Residual       83       259.2       3.123
Total          89       313.8       3.526

Percentage variance accounted for 11.4

*** Estimates of regression coefficients ***

                            estimate         s.e.         t
Constant                      -1.087        0.328     -3.31
x1 1                           0.414        0.461      0.90
x2 1                           0.589        0.475      1.24
x3 1                           1.949        0.389      5.01
x4 1                           2.222        0.395      5.63
x5 1                          -0.192        0.507     -0.38
x6 1                          -0.831        0.482     -1.72
* MESSAGE: s.e.s are based on dispersion parameter with value 1

*** Correlations ***

estimate          ref    correlations
```

```
Constant            1    1.000
x1 1                2   -0.232  1.000
x2 1                3   -0.215  0.699  1.000
x3 1                4   -0.578 -0.218 -0.225  1.000
x4 1                5   -0.563 -0.209 -0.215  0.473  1.000
x5 1                6   -0.266 -0.564 -0.584  0.432  0.250  1.000
x6 1                7   -0.170 -0.552 -0.569  0.100  0.348  0.520  1.000
                        1      2      3      4      5      6      7

 Exchangeable correlation structure.

 Scale factor fixed to         1.000  .

 Scale factor       1.000

 Matrix of correlations

   1      1.0000
   2     -0.0315      1.0000
   3     -0.0315     -0.0315      1.0000

              1           2           3

 *** Model estimates of s.e.***
    Estimate        s.e.
      -1.084       0.3299
       0.417       0.4667
       0.593       0.4810
       1.946       0.3939
       2.220       0.4000
      -0.202       0.5096
      -0.832       0.4848

 *** Correlations ***
   1      1.0000
   2     -0.2397   1.0000
   3     -0.2230   0.6971   1.0000
   4     -0.5810  -0.2156  -0.2219   1.0000
   5     -0.5675  -0.2063  -0.2123   0.4734   1.0000
   6     -0.2639  -0.5623  -0.5808   0.4274   0.2469   1.0000
   7     -0.1679  -0.5492  -0.5681   0.0988   0.3448   0.5210   1.0000
              1        2        3        4        5        6        7

 *** Sandwich estimates of s.e.***
    Estimate        s.e.
      -1.084       0.3169
       0.417       0.4203
       0.593       0.4582
       1.946       0.4138
       2.220       0.4196
      -0.202       0.5152
      -0.832       0.4194

 *** Correlations ***
   1      1.0000
   2     -0.1696   1.0000
   3     -0.0754   0.6610   1.0000
   4     -0.6075  -0.2721  -0.2917   1.0000
   5     -0.6099  -0.2952  -0.1850   0.6783   1.0000
   6     -0.4075  -0.4601  -0.5855   0.3746   0.2592   1.0000
   7     -0.2589  -0.2603  -0.5051   0.0096   0.2414   0.4252   1.0000
              1        2        3        4        5        6        7
```

6. Discussion

For normally distributed data the *naive* estimates of $V[\hat{\beta}]$ obtained under the different covariance models are all that are required. The statistics involved have known distributional properties and comparisions between different covariance models can be made using likelihood ratio tests, provided likelihood methods have been used to estimate the covariance structure. Otherwise, conventional Wald type procedures can be used.

For non-normal data both *naive* and *sandwich* estimates of *Vbh* are needed. Assumption of a correlation model is required to estimate β and this estimate is robust against incorrect specification of that model. However, *VNbh*

is sensitive to incorrect specification of the correlation model. The *naive* estimate of $Var_N(\beta)$ assumes the correlation model holds. The *sandwich* estimate is a data based estimate robust against incorrect specification of the correlation model. As several authors (e.g. Paik, 1992; Verbyla, 1993) comment, R is not required to be the true correlation structure for the *sandwich* estimate to be a reasonable estimate of $V[\hat{\beta}]$. The only condition is that the estimate of correlation used must be consistent.

The methodology implemented is based on that described in Liang and Zeger (1986). In the terminology of Liang, Zeger and Qaqish (1992) it is GEE1. GEE1 assumes that β is independent of α and that they can be estimated separately. As implemented, only the parameters β and their variances $V[\hat{\beta}]$ are estimated using GEE methodology. Quadratic extimation (Crowder, 1985; Crowder, 1992) is used for the parameters α and ϕ. As described in Liang and Zeger (1986) and implemented in the procedure, this is appropriate in situations where the emphasis is on making inferences about β with α being considered a nuisance parameter. Liang *et al* (1992) show that the estimates of α obtained by GEE1 can be seriously inefficient, and this is most likely to be a problem in small data sets. GEE2 (in the terminology of Liang *et al* (1992) recognises that the independence assumption of GEE1 is not correct and jointly estimates β and α by GEE methods. If α is of primary interest the GEE2 methods (proposed by Prentice, 1988 and Zhao and Prentice, 1990) should be used. Liang *et al* (1992) showed these GEE2 estimates to be reasonably efficient for α in the cases studied, but there was little increase in the efficiency of β. As commented e.g. in Carey *et al* (1993) the computational load of GEE2 can be heavy and is a major reason against its implementation, particularly when primary interest is in β. Some authors (e.g. Paik, 1992; Verbyla, 1993) have considered the situation where ϕ is a vector of standard length (number of observations) whose values vary according to a linear model based on a set of explanatory variables. Estimation of the parameters of this model again being by use of GEE. Such modelling of the scale factors requires third and fourth distributional moments. These are available for the standard distributions implemented as options in the procedures but may be difficult to obtain for user defined distributions. Modelling of the scale factors is difficult to implement, and with regard to modelling the fitted values μ allowing for heterogeneity of variance, other approaches i.e. weighting and use of another distribution (e.g. negative binomial for overdispersed binomial data) exist and can be used with the procedure. As shown by several authors (e.g. Paik, 1992), it is possible to use GEE methodology to estimate β, α, ϕ and their associated variances for particular types of data. Diggle *et al* (1994) contains further discussion of these points. Development of the procedure to provide GEE estimation of all these parameters for the range of distributions, etc., implemented in the current procedure is a possibility for the future.

References

Carey V, Zeger S L and Diggle P J (1993) Modelling multivariate binary data with alternating logistic regressions. *Biometrika* **80** 517-526.

Clayton D (1992) Repeated ordinal measurements: a generalized estimating equation approach. Technical Report, Medical Research Council Biostatistics Unit, Cambridge, U.K.

Crowder M J (1985) Gaussian estimation for correlated binary data. *Journal of the Royal Statistical Society, Series B* **47** 229-237.

Crowder M J (1992) Contribution to the discussion of Multivariate regression analyses for categorical data by Liang K-Y, Zeger S L and Qaqish B. *Journal of Royal Statistical Society, Series B* **53** 3-40.

Crowder M J and Hand D J (1990) *Analysis of Repeated Measurements*. Chapman and Hall, London.

Dempster A P, Laird N M and Rubin D B (1977) Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of Royal Statistical Society, Series B* **39** 1-38.

Diggle P J, Liang, K-Y and Zeger S L (1994) *Analysis of Longitudinal Data* Oxford University Press, Oxford.

Henrici P (1964) *Elements of Numerical Analysis*. Wiley, New York.

Jones B and Kenward M G (1987) Modelling binary data from a three-point cross-overtrial. *Statistics in Medicine* 555-564.

Kenward M G (1987) A method for comparing profiles of repeated measurements. *Applied Statistics* **36** 296-308.

Kenward M G, Lesaffre E and Molenberghs G (1994) An application of maximum likelihood and generalized estimating equations to the analysis of ordinal data from a longitudinal study with cases missing at random. *Biometrics* **50** 945-953.

Kenward M G and Smith D M (1994) Computing the generalized estimating equations for repeated ordinal, categorical measurements. *Genstat Newsletter* (submitted for publication).

McCullagh P and Nelder J A (1989) *Generalized Linear Models*. Chapman and Hall, London.

Liang K-Y and Zeger S L (1986) Longitudinal data analysis using generalized linear models. *Biometrika* **73** 13-22.

Liang K-Y, Zeger S L and Qaqish B (1992) Multivariate regression analyses for categorical data (with Discussion). *Journal of Royal Statistical Society, Series B* **53** 3-40.

Paik M C (1992) Parametric variance function estimation for nonnormal repeated measurements data. *Biometrics* **48** 19-30.

Patterson H D and Thompson R (1971) Recovery of inter-block information when block sizes are unequal. *Biometrika* **58** 545-554.

Prentice R L (1988) Correlated binary regression with covariates specific to each binary observation. *Biometrics* **44** 1033-1048.

Styan G P H (1968) Inference in multivariate normal populations with structure. Part I: inference of variances when correlations are known. University of Minnesota, Department of Statistics, Technical Report.

Verbyla A P (1993) Modelling variance heterogeneity: residual maximum likelihood and diagnostics. *Journal of the Royal Statistical Society, Series B* **55** 493-508.

Zhao L P and Prentice R L (1990) Correlated binary regression using a quadratic exponential model. *Biometrika* **77** 642-648.

Computing the generalized estimating equations for repeated ordinal measurements

M G Kenward
IACR-Rothamsted
Harpenden
Herts AL5 2JQ, UK

D M Smith
Department of Mathematics
Statistics and Computing Science
The University of New England
Armidale, NSW 2351, Australia

Summary

The application of general estimating equation methodology to the analysis of repeated ordinal data is reviewed. The use of an available Genstat procedure for generalized estimating equations to analyse this type of data is described. Estimation of the correlation matrix for various models is reviewed. An example involving two equivalent analyses of a set of data is given.

1. Introduction

While analyses for continuous repeated measurements data are well established the same is not so true for categorical, and in particular ordinal, repeated measurements. The GEE (generalized estimating equations) methodology described in Kenward and Smith (1995) and implemented as a Genstat procedure can however be extended into the analysis of repeated ordinal measurements. Modifications of GEE methodology using binary representations (i.e. generalized linear models) for fitting proportional odds models have only recently appeared (Clayton, 1992; Kenward *et al* 1994) and a further Genstat procedure has been written that allows the user to fit such models, with an appropriate data transformation, using the procedure of Kenward and Smith (1995) which should be read in conjunction with this paper.

2. Methodology

The method used is the development of Clayton (1992) as described in Kenward *et al* (1994). A K category ordinal response Y_{it} subject i, times point t) can be expressed as $K-1$ binary responses Z_{itk} where

$$Z_{itk} = \begin{bmatrix} 0 & Y_{it} \leq k \\ 1 & Y_{it} > k \end{bmatrix}$$

A proportional odds model for Y_{it}

$$\text{logit}\{P(Y_{it} > k)\} = \alpha_k + x_{it}\beta \qquad k = 1, \ldots, K-1$$

implies a logistic regression model for each binary response

$$\text{logit}\{P(Z_{itk} = 1)\} = \alpha_k + x_{it}\beta \qquad k = 1, \ldots, K-1. \tag{1}$$

The GEE method can be applied to these derived binary responses, taking account of their dependence through a suitable choice of correlation matrix for the the binary variables (R_i say for the ith subject). Clayton (1992) obtains an empirical estimate of this matrix for each group of subjects with common covariates based on the observed proportions of observations in different categories at different times and substitutes these into the

estimating equations. However, by exploiting the relationship between the cutpoint parameters (α_k) and the correlations between the binary responses, the correlations in the estimating equations can be expressed in terms of the model parameters and reliance upon the empirical estimate of R_i avoided. Define

$$P(Y_{it} > k) = P(Z_{itk} = 1) = \Phi_{itk}, \quad k = 1, \ldots, K-1.$$

then, for ρ_{jk} = correlation(Z_{itj}, Z_{itk}), $j < k$,

$$\rho_{jk}^2 = \frac{\phi_{itj}(1 - \phi_{itk})}{(1 - \phi_{itj})\phi_{itk}} = exp(\alpha_j - \alpha_k) . \qquad (2)$$

Note that a single set of correlations applies to all subjects at all time points. This is a consequence of the proportionality in the original model.

In the correlation matrix R_i used by the fitted model and used to produce the *naive* estimate of $V[\hat{\beta}]$, the correlations between observations from different time points are set to zero (the independence assumption for the repeated measurements) and the correlations between the binary responses from a particular time are calculated from equation (2) using the current estimates of the cutpoint parameters. That is, for a complete sequence $Z_i = (Z_{i11}, Z_{i12}, \ldots, Z_{iT(K-1)})$,

$$R_i = \begin{bmatrix} R & 0 & 0 & \ldots \\ 0 & R & 0 & \ldots \\ \vdots & \vdots & \vdots & \vdots \\ \ldots & 0 & 0 & R \end{bmatrix}.$$

for $r_{jk} = r_{kj} = \rho_{jk}$. The elements of R_i are updated each GEE iteration using the current estimates of the cutpoints α_j, α_k. To produce the *sandwich* estimate of $V[\hat{\beta}]$, the zero off-diagonal elements of R_i are replaced by the appropriate residual cross-products, modified where necessary if the data are unbalanced. For details see Kenward and Smith (1995).

After the GEE iterative procedure has converged a *sandwich* estimate of the parameter covariance matrix is obtained. The correlations between the observations from the same time point are obtained as for the *naive* estimate. The remaining across times correlations are obtained empirically from the residuals.

3. Construction of R_i

To exploit the procedure for binary data the ordinal measurements must first be expanded into the set of binary responses as defined above, and a user-defined correlation procedure is then used (**GEECORREL**). The version of **GEECORREL** given in the Appendix (for the expanded data set) can be used for any set of ordinal categorical data provided the scalar **ncut** is set to the appropriate value, which is the number of categories minus one. Also, the variate **TIME** needs to be set to a dummy factor with **(ncut-one)*(ntime)** levels, where **ntime** is the number of time points. For the data of the example (Kenward and Jones, 1992) **ncut** is set to two as there are three categories. In the procedure **GEECORREL** the **OPTION [SANDWICH=no,yes]** refers to whether the R_i matrix for the *naive* estimate of $V[\hat{\beta}]$ is being constructed or the R_i matrix for the *sandwich* estimate.

4. Example

The data set of **Example 1** of Kenward and Jones (1992) is used to examplify how to perform the analysis. Two equivalent analyses will be given. In the first, data are input for each subject, with the subjects specified using the **SUBJECT** parameter. In the second, the data are input as number per outcome group and the **OUTCOME** and **COUNT** parameters must be set. The second analysis is considerably faster than the first. Analysis of repeated ordinal categorical data as number per outcome group is standard practise in analysing log linear models. For both analyses the Genstat code prior to calling **GEE** is for expanding the data into the cumulative logit form of

equation (1).

4.1 First analysis

Only the first and last six lines of data are given.

Input program:

```
PRINT [IPRINT=*; SERIAL=Y; SQUASH=Y] \
        'Data from Kenward and Jones, (1992), data of Example 1.'
FACTOR [LEVELS=2] Period
 &     [LEVELS=3] Trt
 &     [LEVELS=!(0,1)] Resp[1...3]
READ [SETNVALUES=YES] Trt,Period,Resp[1...3]
1 1 1 0 0
1 2 1 0 0
1 1 1 0 0
1 2 1 0 0
1 1 1 0 0
1 2 1 0 0
. . . . .
3 1 0 0 1
3 2 0 0 1
3 1 0 0 1
3 2 0 0 1
3 1 0 0 1
3 2 0 0 1
:
FACTOR [LEVELS=256] Sub; VALUES=!(2(1...256))

VARIATE Y[1...2]
CALCULATE Y[1] = Resp[2] + Resp[3]
 &        Y[2] = Resp[3]
VARIATE [NVALUES=1024] Mresp
FACTOR [NVALUES=1024;LEVELS=256] Msub
 &     [LEVELS=3;NVALUES=1024] MTrt
 &     [LEVELS=2;NVALUES=1024] MPeriod
 &     [LEVELS=2;NVALUES=1024] Mcut; VALUES=!((1...2)512)
VARIATE [NVALUES=1024] Mn; VALUES=!(1024(1))
 &      [NVALUES=4] Time; VALUES=!(1...4)
"
Period is a dummy time set as 1 to 4.
"
EQUATE [OLDFORMAT=!((1,-511)2,-1)] !P(Y[1...2]); Mresp
 &     [OLDFORMAT=!((1,-511)2,-1)] !P(Sub,Sub); Msub
 &     [OLDFORMAT=!((1,-511)2,-1)] !P(Trt,Trt); MTrt
 &     [OLDFORMAT=!((1,-511)2,-1)] !P(Period,Period); MPeriod

PRINT [IPRINT=*; SERIAL=Y; SQUASH=Y] \
      'Logit link : binomal error'

GEE [LINK=LOGIT; DISTRIBUTION=BINOMIAL; TERMS=Mcut+MTrt+MPeriod] \
       SUBJECT=Msub; Y=Mresp; TIME=Time; NBINOMIAL=Mn
```

Output:

```
 Data from Kenward and Jones, (1992), data of Example 1.

 Logit link : binomal error

***** Regression Analysis *****

 Response variate: workvar
   Weight variate: weight
     Fitted terms: Constant + Mcut + MTrt + MPeriod

*** Summary of analysis ***
    Dispersion parameter is 1

              d.f.         s.s.         m.s.      v.r.
Regression       4         221.       55.300     54.27
Residual      1019         1038.       1.019
Total         1023         1260.       1.231
```

```
Percentage variance accounted for 17.2

*** Estimates of regression coefficients ***

                              estimate          s.e.          t
Constant                         0.859         0.174       4.94
Mcut 2                          -2.754         0.188     -14.65
MTrt 2                          -0.625         0.202      -3.09
MTrt 3                          -0.222         0.191      -1.16
MPeriod 2                       -0.598         0.160      -3.75
* MESSAGE: s.e.s are based on dispersion parameter with value 1

*** Correlations ***

estimate                    ref     correlations

Constant                      1     1.000
Mcut 2                        2    -0.367  1.000
MTrt 2                        3    -0.615  0.100  1.000
MTrt 3                        4    -0.620  0.043  0.523  1.000
MPeriod 2                     5    -0.493  0.123  0.030  0.010  1.000
                                       1      2      3      4      5

 User supplied correlation structure.

 Scale factor fixed to          1.000   .

 Scale factor        1.000

 Matrix of correlations

   1      1.0000
   2      0.2519       1.0000
   3      0.0000       0.0000       1.0000
   4      0.0000       0.0000       0.2519       1.0000

                1            2            3            4

*** Model estimates of s.e.***
    Estimate         s.e.
       0.864       0.1849
      -2.757       0.1668
      -0.622       0.2201
      -0.204       0.2099
      -0.628       0.1741

*** Correlations ***
   1      1.0000
   2     -0.2592       1.0000
   3     -0.6293       0.0914       1.0000
   4     -0.6375       0.0381       0.5291       1.0000
   5     -0.4996       0.1197       0.0299       0.0090       1.0000
                1            2            3            4            5

*** Sandwich estimates of s.e.***
    Estimate         s.e.
       0.864       0.2015
      -2.757       0.1866
      -0.622       0.2584
      -0.204       0.2496
      -0.628       0.1341

*** Correlations ***
   1      1.0000
   2     -0.2633       1.0000
   3     -0.6666       0.0784       1.0000
   4     -0.6693       0.0239       0.5177       1.0000
   5     -0.3732       0.1701       0.0336       0.0087       1.0000
                1            2            3            4            5
```

4.2 Second Analysis

Input program:

```
PRINT [IPRINT=*; SERIAL=Y; SQUASH=Y] \
```

```
          'Data from Kenward and Jones, (1992), Example 1.'

FACTOR [NVALUES=54;LEVELS=27] Outcome; VALUES=!(2(1...27))
 &     [NVALUES=54;LEVELS=3] Trt; VALUES=!(18(1...3))
 &     [NVALUES=54;LEVELS=2] Time; VALUES=!((1,2)27)
VARIATE [NVALUES=54] Resp[1...3],Count

READ Resp[1...3],Count
1 0 0 17
1 0 0 17
1 0 0  6
0 1 0  6
1 0 0  2
0 0 1  2
. . .  .
0 0 1  1
1 0 0  1
0 0 1  3
0 1 0  3
0 0 1  3
0 0 1  3
:
VARIATE Y[1...2]
CALCULATE Y[1] = Resp[2] + Resp[3]
 &        Y[2] = Resp[3]
VARIATE [NVALUES=108] Mresp
EQUATE [OLDFORMAT=!((1,-53)2,-1)] !P(Y[1...2]); Mresp
FACTOR [NVALUES=108;LEVELS=27] MOutcome
 &     [LEVELS=3;NVALUES=108] MTrt
 &     [LEVELS=2;NVALUES=108] Mper; VALUES=!(2(1...2)27)
 &     [LEVELS=2;NVALUES=108] Mcut; VALUES=!((1...2)54)
VARIATE [NVALUES=108] Mn; VALUES=!(108(1))
 &      [NVALUES=108] MCount
 &      [NVALUES=4] Period; VALUES=!(1...4)
"
Period is a dummy time set as 1 to 4.
"
EQUATE [OLDFORMAT=!((1,-53)2,-1)] !P(Outcome,Outcome); MOutcome
 &     [OLDFORMAT=!((1,-53)2,-1)] !P(Trt,Trt); MTrt
 &     [OLDFORMAT=!((1,-53)2,-1)] !P(Count,Count); MCount

PRINT [IPRINT=*; SERIAL=Y; SQUASH=Y] \
      'Logit link : binomal error'

GEE [LINK=LOGIT; DISTRIBUTION=BINOMIAL; TERMS=Mcut+MTrt+Mper] \
    OUTCOME=MOutcome;  COUNT=MCount; Y=Mresp; TIME=Period; NBINOMIAL=Mn
```

Output:

```
 Data from Kenward and Jones, (1992), Example 1.

 Logit link : binomal error

The OUTCOME option has really been set up for use with count (Poisson) and
binomial data where rather than inputting individual subject data, outcomes are
input with the number of subjects with each outcome input as a count variate.
Use of OUTCOME and COUNT is much faster.
WHETHER THE DISTRIBUTION IS POISSON OR BINOMIAL IS NOT CHECKED.
This enables, for example, overdispersion to be handled by use of the own
DISTRIBUTION option and/or weights.

***** Regression Analysis *****

 Response variate: workvar
   Weight variate: weight
     Fitted terms: Constant + Mcut + MTrt + Mper

*** Summary of analysis ***
    Dispersion parameter is 1

              d.f.         s.s.         m.s.      v.r.
Regression       4         221.        55.30      5.49
Residual       103         1038.       10.08
Total          107         1260.       11.77
```

```
Percentage variance accounted for 14.4

*** Estimates of regression coefficients ***

                              estimate         s.e.          t
Constant                         0.859        0.174       4.94
Mcut 2                          -2.754        0.188     -14.65
MTrt 2                          -0.625        0.202      -3.09
MTrt 3                          -0.222        0.191      -1.16
Mper 2                          -0.598        0.160      -3.75
* MESSAGE: s.e.s are based on dispersion parameter with value 1

*** Correlations ***

estimate                  ref    correlations

Constant                    1    1.000
Mcut 2                      2   -0.367  1.000
MTrt 2                      3   -0.615  0.100  1.000
MTrt 3                      4   -0.620  0.043  0.523  1.000
Mper 2                      5   -0.493  0.123  0.030  0.010  1.000
                                    1      2      3      4      5

 User supplied correlation structure.

 Scale factor fixed to          1.000  .

 Scale factor       1.000

 Matrix of correlations

   1      1.0000
   2      0.2519      1.0000
   3      0.0000      0.0000      1.0000
   4      0.0000      0.0000      0.2519      1.0000

               1           2           3           4

*** Model estimates of s.e.***
   Estimate         s.e.
      0.864       0.1849
     -2.757       0.1668
     -0.622       0.2201
     -0.204       0.2099
     -0.628       0.1741

*** Correlations ***
   1      1.0000
   2     -0.2592      1.0000
   3     -0.6293      0.0914      1.0000
   4     -0.6375      0.0381      0.5291      1.0000
   5     -0.4996      0.1197      0.0299      0.0090      1.0000
                1           2           3           4           5

*** Sandwich estimates of s.e.***
   Estimate         s.e.
      0.864       0.2015
     -2.757       0.1866
     -0.622       0.2584
     -0.204       0.2496
     -0.628       0.1341

*** Correlations ***
   1      1.0000
   2     -0.2633      1.0000
   3     -0.6666      0.0784      1.0000
   4     -0.6694      0.0239      0.5177      1.0000
   5     -0.3733      0.1701      0.0337      0.0087      1.0000
                1           2           3           4           5
```

5. Discussion

For the analyses given the scale factor has been fixed at 1.0. For these data using **SCALEFACTOR=constant** gives a scale factor of 1.024 suggesting that fixing the scale factor to 1.0 is reasonable. If use of this option had produced a scale factor very much greater than 1.0 (representing overdispersion) then having obtained an estimate of the heterogeneity factor the parameter **WEIGHT** or the user defined setting of the **DISTRIBUTION** option could be used to handle the overdispersion. The heterogeneity factor can be estimated from the residuals as in Williams (1982), then either weights could be calculated following the quasi-likelihood approach of Williams (1982), or a beta binomial distribution fitted by means of the user defined procedure **GEEDISTRIBUTION**. For overdispersed Poisson data, Breslow (1984) is the equivalent reference and the negative binomial is the equivalent distribution.

Availability of the own correlation procedure **GEECORREL** enables many models for the correlation between time points to be tried. In the context of repeated ordered categorical data, correlations between the time points can be introduced as well as between the binary responses at the same time point.

References

Breslow N E (1984) Extra-Poisson variation in log-linear models. *Applied Statistics* **33** 38-44.

Clayton D (1992) Repeated ordinal measurements: a generalized estimating equation approach. Technical Report, Medical Research Council Biostatistics Unit, Cambridge, U.K.

Kenward M G and Jones B (1992) Alternative approaches to the analysis of binary and categorical repeated measurements. *Journal of Biopharmaceutical Statistics* 137-170.

Kenward M G, Lesaffre E and Molenberghs G (1994) An application of maximum likelihood and generalized estimating equations to the analysis of ordinal data from a longitudinal study with cases missing at random. *Biometrics* **50** 945-953.

Kenward M G and Smith D M (1995) Computing the generalized estimating equations with quadratic estimation for repeated measurements. *Genstat Newsletter* **32** 49-61.

Williams D A (1982) Extra-binomial variation in logistic linear models. *Applied Statistics* **31** 144-148.

Appendix - Procedure GEECORREL

```
PROCEDURE   'GEECORRELATION'
"
  Calculation of correlation matrix

  For SANDWICH = NO
      input is the R matrix as for UNSPECIFIED
      output is the desired R matrix.
  For SANDWICH = YES
      input is the (Y-MU)*T(Y-MU) matrix
      output is the desired modified (Y-MU)*T(Y-MU) matrix.

  N.B. For the normal distribution both the input and output R's
       should be variance/covariance matrices not correlation matrices.
"
OPTION      NAME = 'CONSTANT', "I: text; how to treat constant (estimate,
                                  omit); default e"\
                   'SANDWICH'; "I; text; whether the sandwich central matrix
                                  product or not) (no,yes); default no"\
            MODE=2(T);NVALUES=2(1); \
            VALUES=!T(ESTIMATE,OMIT),!T(NO,YES); \
            DEFAULT=!T(ESTIMATE),!T(NO);

PARAMETER NAME = 'CORRELATIONS',"I/O: matrix; the correlation matrix"\
                 'ESTIMATES',   "I: variate; estimates of parameters in model"\
                 'Y',           "I: variate; response variate"\
                 'RESIDUALS',   "I: variate; residuals"\
                 'FITTEDVALUES',"I: variate; fitted values"\
                 'TIME',        "I: variate; times of repeated measures"\
                 'MARKER',      "I: factor; identifier of subject or outcome"\
                 'DISTRIBUTION',"I: text; identifier of distribution"\
                 'SCALEFACTOR', "I: text; scalefactor option in use"\
                 'SFVALUE';     "I: scalar; value of scalefactor if FIXED"\
```

```
          SET=10(yes);DECLARED=10(yes); \
          TYPE='symmetric',5('variate'),'factor',2('text'),'scalar'; \
          PRESENT=9(yes),no

  GETATTRIBUTE [ATTRIBUTE=NVALUES] ESTIMATES; SAVE=!P(ncol)
   &           [ATTRIBUTE=NROWS] CORRELATIONS; SAVE=!P(ntime)

  DIAGONALMATRIX [ROWS=ntime;MODIFY=yes] done; VALUES=!(#ntime(1))

  CALC const = 'ESTIMATE' .IN. CONSTANT
   &   sandw = 'NO' .IN. SANDWICH

  IF sandw
    CALC CORRELATIONS = done
  ELSE
    CALC   ntime1 = ntime - 1
     &     n1time = -ntime1
     &     n2time = n1time + 1
     &   ntime1t2 = 2*ntime1
    VARIATE [NVALUES=ntime1t2;MODIFY=YES] cii
     &      [NVALUES=ntime1;MODIFY=YES] ci[1...2]; \
             VALUES=!(#ntime1(1)),!(1...ntime1)
    CALC ci[2] = -ci[2]
    EQUATE [OLDFORMAT=!(1,#n2time,1,#n1time)] ci; cii
     &     [OLDFORMAT=!(#ntime);NEWFORMAT=cii] done; CORRELATIONS
  ENDIF
"
  Set ncut equal to the number of cutpoints i.e. number of
  categories minus one.
"
  CALC    ncut = 2
   &     ncut1 = ncut - 1
   &    ncut12 = ncut*ncut1/2
   &     ncii1 = 2*ncut1
   &      ncii = ncii1 + 1
   &     n1cut = -ncut1
   &     n2cut = n1cut + 1
   &        nr = ntime/ncut

  VARIATE [NVALUES=ncut] cutpt
   &      [NVALUES=ncut12] dr
   &      [NVALUES=ncii;MODIFY=YES] cii; VALUES=!(#ncii(*))
   &      [NVALUES=ncut1;MODIFY=YES] ci[1...2]; \
           VALUES=!(#ncut1(-1)),!(1...ncut1)

  EQUATE ESTIMATES; cutpt

  IF const
    CALC          cutpt = cutpt + ELEM(cutpt;1)
     &   ELEM(cutpt;1) = ELEM(cutpt;1)/2
  ENDIF

  CALC im1 = 0
  FOR ind2 = 2...ncut
    CALC im2 = ind2 - 1
    FOR ind1 = 1...im2
      CALC          im1 = im1 + 1
       &   ELEM(dr;im1) = SQRT(EXP(ELEM(cutpt;ind2)-ELEM(cutpt;ind1)))
    ENDFOR
  ENDFOR

  FOR ind1 = 1...nr
    EQUATE [OLDFORMAT=!(1,#n2cut,1,#n1cut);NEWFORMAT=!(#ncii1,*)] ci; cii
     &     [OLDFORMAT=!(#ncut12);NEWFORMAT=cii] dr; CORRELATIONS
    CALC          ci[1] = ci[1] - ncut
     &   ELEM(ci[1];1) = -((ncut*ind1)**2+5*ncut*ind1+2)/2
  ENDFOR

ENDPROCEDURE
```