R.W. Payne

Issue No.12

1983 October

The GENSTAT Newsletter

EDITORS: *R.W. PAYNE*
ROTHAMSTED EXPERIMENTAL STATION
HARPENDEN
HERTFORDSHIRE
AL5 2JQ

M.G. RICHARDSON
NAG CENTRAL OFFICE
MAYFIELD HOUSE
256 BANBURY ROAD
OXFORD
OX2 7DE

NAG

Contents

Published twice yearly by
Rothamsted Experimental Station Statistics Department
and the Numerical Algorithms Group Limited

Printed August 1984

Editorial

Many of the articles in this issue and the next one are based on talks presented at the Third Genstat Conference, which took place at Orsay from 10–12 October 1983. (I should like to thank the contributors for the high standards of presentation shown at the conference and for the trouble taken to make their talks available to the wider audience of Newsletter readers. RWP) The next conference will be at University of York, from 23–26 September 1985.

The typesetting of Newsletter 11 seems to have been well received by most readers. We have made further changes of format in this issue and should again welcome your views and comments. Finally, we repeat our invitation to readers to submit articles, letters or announcements for future issues.

Letters to the Editor

F.R. House
Department of Pharmacology
Guy's Hospital Medical School
London SE1 9RT
United Kingdom

Dear Sir

In reply to the Editorial of Newsletter No. 11, I approve of the typeset layout, but regret the illegibility of the code in the contribution by G. V. Stevens on the Box–Cox transformation.

Dr. Stevens has kindly sent me a legible copy of his program and I should like to make the following comments, perhaps, if you will permit, to encourage further discussion:

1. The program seems rather long, Dr. Stevens has explained (private communication) that this partly arises from the nature of its intended use, by unsophisticated users, and the need for it to 'defend' itself against wrong data, but says that such defence is not comprehensive. I am not entirely happy about this, as examination of the program shows that some quite common errors would produce unhelpful GENSTAT-type diagnostics. Experienced users will sort them out with the help of their well-thumbed Chapter 11 (could separate copies be made available to replace worn out ones?); but I suggest that GENSTAT, although excellent for analysis or correctly prepared data, is unkind to novices who get things wrong, and a program for unsophisticated users is an ideal case for use of a pre-processor.
2. The program as published deals only with designs that have equal replication in all cells, GENSTAT is more flexible than this, and a MACRO could be persuaded to cope with any design that the package can analyse. I hope that Dr. Stevens can find time to produce a compact MACRO which readers can compare with the CSIRO MACRO which does much the same job.
3. I should like to raise a general hare about graphical conventions, using this matter as an example. The program produces two graphs containing residuals from the fitted model: one of residuals vs. fitted values, and one of expected normal quantiles vs. residuals. It seems illogical, and potentially confusing to naive users, to place residuals as the ordinate in one case and the abscissa in the other. For the latter graph, ample support for both ways round can be found in the literature, but I suggest that the widely followed convention of putting fixed, accurately known, or otherwise given values on the abscissa, and measured or derived values, varying from one occasion to another on the ordinate, provides the sensible solution, and I should like to

advocate this most strongly. (See, for example, the previous contribution in the same Newsletter.)

Those of us who teach or advise in Statistics spend a lot of time trying to convince people that they ought to draw their graphs the right way round and use some discretion about which regression to calculate; should we not set a good example?

Finally, I should like to say that I learned a lot from studying the program and I am grateful to the Author for submitting it.

G. V. Stevens
Department of Mathematics
Liverpool Polytechnic
Byrom Street
Liverpool L3 3AF
United Kingdom

Dear Sir

In reply to Dr. House's letter I would like to make just a few points:

1. The time taken and extra length of the macro to make the macro 'idiot proof' is not worthwhile at the moment. Users of such a macro would have experience of statistics packages, including Genstat, with the result that the need for carefully prepared data files would be known.

   The Computing Services Department here at the Polytecnic has compiled the list of Genstat error diagnostics in the mini-introduction to the package. Such an included list does help users.

2. I have converted my program into a macro with a driver program, hopefully for inclusion in the macro library. In the use of such a macro to handle unbalanced factorial designs, interpretation of an ANOVA tables must be treated with more care.

3. The so called 'ANOVA' tables in my macro are ANOVA tables associated with fitting particular linear models.

   The point of such a macro is to find an alternative scale of measurement which, as far as possible, satisfies all of the assumptions made when fitting a linear model.

   Two assumptions are:

   (i) Normality of data

   (ii) Errors independent of any factor effects, usually translated to independence of errors from cell means.

The cell means, in the factorial case, are the fitted values and the residuals are estimates of the errors.

We are looking for any dependence of residuals upon fitted values. Hence the residuals reside on the vertical axis for a dependent variable.

For reference see Draper and Smith, chapter 3 and Davies and Goldsmith, chapter 8.

Having stated the foregoing I can understand the confusion when, suddenly, in the Expected Normal Quantile plot, the residuals are now on the horizontal axis usually preserved for independent and tightly controlled variables. This graph is needed to check assumption (i) above and it is for this **different** reason that the residuals are now on the horizontal axis.

If the data do come from a Normal distribution with possibly different means then the expected

normal quantiles will depend upon the residuals and this dependence should be linear. These graphs are customary and found in statistics journals. (See, for instance John and Draper.)

I hope these comments help answer Dr. House's queries.

References

Davies, O.L. and Goldsmith, R.L. (1976) Statistical Methods in Research and Production (Longman)

Draper, N.R. and Smith, H. (1981) Applied Regression Analysis (Wiley)

John, J.A. and Draper, N.R. (1980) An Alternative Family of Transformations Appl. Statist. **29**, No. 2, 190–197.

Editor's Note

We should like to take this opportunity to apologise for the poor quality of reproduction of the program mentioned in Dr. House's letter.

Dr. Stevens has kindly offered to supply listings of the program to interested readers.

Savage's Log-Rank Test

P. N. Appleby
Metal Box p.l.c.
R & D Division
Denchworth Road
WANTAGE
Oxon OX12 9BP
United Kingdom

The macro LOGRANK performs Savage's log-rank test of the equality of several survivor functions, based on samples of failure times, some of which may be right-censored. as presented in Kalbfleisch and Prentice (1980). The macro computes and prints values of the observed, O, and 'expected', E, number of failures in each sample from which the log-rank statistic v is calculated as v = O-E, the variance-covariance matrix V, and the asymptotic chi-square test statistic $v' V^{-1} v$ based on $r-1$ degrees of freedom, where r denotes the number of samples.

The user supplies the macro with:
scalar R, the number of samples

variate T, containing the failure times and censoring times for all the samples combined

factor C, containing the censoring codes (*1* = failure *2* = censored)

factor S, containing the sample codes (*1*, *2*,..., *R*

each of T, C and S being of length *N*, the size of the combined sample.

As an example, the macro was applied to the rats' vaginal cancer mortality data quoted in Kalbfleisch and Prentice (op.cit), comparison of the two samples giving a X_1^2 value of *3.123*, providing weak evidence (significance at the 10% level) of a difference between the survivor functions.

The Macro

```
'MACRO' LOGRANK $ ''LOG-RANK TEST OF EQUALITY OF SURVIVOR FUNCTIONS''
'LOCAL' F,FLEV,K,TO,DJ,DIJ(1...R),NJ,NIJ(1...R),OF,EF,V,VMELT,VM,POP,
        CHISQR,DF,TF,SV,DN,DONE,COVAROE
'SCAL' K,CHISQR,DF
'START' 'REST' T$C=1  'CALC' TF=T : SV=FLOAT(S)
  'GROU' F=RANK(TF;FLEV) 'CALC' K=NVAL(FLEV) 'RUN'
'DEVA' FLEV 'VARI' TO,DJ,DIJ(1...R),NJ,NIJ(1...R),DN$K
'FOR' FT=1...K 'REST' TF$F=FT 'CALC' ELEM(TO;FT)=MEAN(TF)¿
'FOR' I=1...R ; NIJ=NIJ(1...R) ; DIJ=DIJ(1...R)
'CALC' ELEM(NIJ;FT)=SUM(T.GE.ELEM(TO;FT).AND.SV.EQ.I)
'CALC' ELEM(DIJ;FT)=SUM(TF.EQ.ELEM(TO;FT).AND.SV.EQ.I)
'REPE' 'REPE' 'DEVA' TF,SV
'CALC' NJ=VSUM(NIJ(1...R)) : DJ=VSUM(DIJ(1...R))
: DN=DJ*(NJ-DJ) (NJ*NJ*(NJ-1))
'START' 'CALC' DF=R-1 : K=DF*R/2 'RUN'
'VARI' OF,EF$R : V$DF : VMELT$K 'SYMM' VM$DF
'FOR' I=1...DF ; NIJ=NIJ(1...DF) ; DIJ=DIJ(1...DF)
'CALC' ELEM(OF;I)=SUM(DIJ) : ELEM(EF;I)=SUM(NIJ*DJ/NJ)
'FOR' L=1...DF ; NLJ(1...DF) 'JUMP'DONE*(L.GT.I)
'CALC' K=I*(I-1)/2+L 'JUMP' COVARCE*(L.LT.I)
'CALC' ELEM(VMELT;K)=SUM(NIJ*(NJ-NIJ)*DN) 'JUMP' DONE
'LABEL' COVARCE 'CALC' ELEM(VMELT;K)=SUM(-NIJ*NLJ*DN)
'LABEL' DONE 'REPE' 'REPE'
'CALC' ELEM(OF,EF;R)=SUM(DJ)-SUM(OF,EF)
'EQUA' VM=VMELT 'CALC' CHISQR=RSYMRI(V;INV(VM))
'INTE' POP=1...R
'CAPT' ''LOG-RANK TEST OF EQUALITY OFSURVIVOR FUNCTIONS
OBSERVED AND 'EXPECTED' FAILURES''
'PRIN/P' POP,OF,EF$6.0,8.0,8.3
'CAPT' ''VARIANCE-COVARIANCE MATRIX'' 'PRIN' VM
'CAPT' ''TEST STATISTIC'' 'PRIN/P' CHISQR,DF$8.3,4.0
'ENDMAC'
```

Reference

Kalbfleisch, J.D. and Prentice, R.L. (1980) The Statistical Analysis of Failure Time Data. Wiley

Second Genstat Conference

The papers which follow were first presented at the Second Genstat Conference: some have been slightly modified since and many are somewhat shorter than the original presentations.

The papers have been arbitrarily divided between this issue of the Newsletter and the next, which will follow shortly.

Canonical Analysis of a Response Surface

J. Barnard
NYS Agricultural Experimental Station
Cornell University
Geneva
New York 14456
United States of America

Introduction

The macro RSCA performs second order response surface estimation. If there exists a stationary value for the response variate, the stationary value and x-set coordinates yielding that value are calculated. The canonical representation of the surface is produced as an aid to elucidating the nature of response at the stationary value.

Usage of RSCA

Two SET–lists must be established by the user: YSET must be set to the response variate, XSET to the list of treatment variates. The macro creates a series of global identifiers Q(1... *nf(nf+1)/2*), where *nf* is the number of treatment variates. The Q contain squares and cross products involving the treatment variates and are generated in triangular order (x_{11}, x_{21}, x_{22}, x_{31} ,etc). Regression coefficients are generated in a global identifier B.

Output

Standard regression statistics are printed. The stationary value of the response and coordinates of the stationary point are printed, together with the latent roots of the matrix comprising second order regression coefficients. Signs of the latent roots may be used to determine whether the stationary point is a maximum, a minimum or a saddle point. The transformations relating design variates to canonical variates are given.

The Macro

```
'MACRO' RSCA $
''ARGUMENTS  IN: YSET,XSET; OUT:B,Q''
'LOCA' XP,NF,NQ,K,C,D,E,SC,ROOTS,L1,L2,L3,L4,L5
'SCAL' NF,NQ,K,L1,L2,L3,L4,L5
'START'
  'POIN' XP=XSET 'CALC' NF=NVAL(XP) : NQ(NF*NF+NF)/2
'RUN'
'SET' NNF=1...NF : NNQ=1...NQ
'VARI' Q(NNQ) 'CALC' K=0
'FOR' I=NNF ; II=XSET : J=NNF; JJ=XSET
  'JUMP' L1*(I.LT.J)
    'CALC' K=K+1 'ASSIG' KK=Q(NNQ) $ K 'CALC' KK=II*JJ
'LABE' L1 'REPEAT' :
'TERM' YSET,XSET,Q(NNQ)  'Y' YSET
'FIT/Z,ANDEV=IN' XSET 'ADD/C,ANDEV=T' Q(NNQ) ; COEF=B
'VARI' C $ NF 'EQUA' C=B $ 1X,NF
```

```
'SYMM' D $ NF 'EQUA' D=B $ 1X,NF!X,NQ
'CALC' K=1
'FOR' I=NNQ
  'JUMP' L2*(I.EQ.(K*K+K)/2)
    'CALC' ELEM(D;I)=ELEM(D;I)/2 'JUMP' L3
  'LABE' L2 'CALC' K=K+1
'LABE' L3 'REPEAT'
'MATR' E $ XP,XP 'DIAG' ROOTS $ NF
'CALC' E=D : K=DET(E)  'JUMP' L4*(K.EQ.0)
'VARI' SC $ NF
'CALC' SC=-PDT(INV(E);C)/2
'CALC' K=ELEM(B;1)+TPDT(SC;C)+PDT(TPDT(SC;D);SC)
'LINE' 2 'CAPT' ''Stationary value is''
'PRIN/LABR=1' K
'LINE' 2 'CAPT' ''Coordinates of stationary point''
'PRIN/P,LABC=1' XP,SC $ 12.4
'LRV' MATRIX=D ; RESULTS=E,ROOTS,K
'LINE' 2 'CAPT' ''Latent roots''
'PRIN/LABC=1' ROOTS $ 12.4
'LINE' 2 'CAPT' ''Canonical variates''
'PRIN/LABC=1' E $ 10.4
'JUMP' L5
'LABE' L4 'LINE' 2 'CAPT' ''** Stationary point not determined''
'LABE' L5
'ENDM'
```

Example

The example is taken from Gill (1978). Viscosity of a stored milk product was determined to be minimised by warming for 30 minutes at some temperature between 80°C and 90°C, with the presence of a sodium polyphosphate additive in amounts ranging from 0.04% to 0.12%. An experiment was run with temperatures of 80°C, 85°C and 90°C and with levels of additive at 0.04%, 0.08% and 0.12%. The analysis produced positive roots, indicating that the stationary point is a minimum for the fitted surface. Canonical variates are given by the equations

$$w_1 = 0.5175(x_1 + 0.0026) + 0.8557(x_2 - 0.2650)$$
$$w_2 = 0.8557(x_1 + 0.0026) - 0.5175(x_2 - 0.2650)$$

```
'UNIT' $ 9
'VARI' TEMP=(-1,0,+1)3
     : ADDITIVE=3(-1,0,+1)
     : VISCOSITY=645,255,285,65,45,70,160,15,315
'SET' YSET=VISCOSITY : XSET=TEMP,ADDITIVE
'USE/R' RSCA $
'RUN'
```

```
***** REGRESSION ANALYSIS *****

*** REGRESSION COEFFICIENTS ***

Y-VARIATE: VISCOSIT
                  ESTIMATE        S.E.          T

CONSTANT           -41.111      61.746       -0.67
TEMP               -33.333      33.820       -0.99
ADDITIVE          -115.833      33.820       -3.43
Q(1)               151.667      58.578        2.59
Q(2)               128.750      41.421        3.11
Q(3)               219.167      58.578        3.74


*** SUMMARY OF ANALYSIS OF VARIANCE ***

Y-VARIATE: VISCOSIT

  TERMS                 RESIDUAL       CHANGE            MEAN
VARIANCE
                       DF      SS     DF      SS        CHANGE   RATIO
INITIAL MODEL
 CONSTANT               8     316139   *         *

*MODIFICATIONS TO MODEL
+TEMP
+ADDITIVE               6     228968   2      87171      43585    6.35
+Q(1)
+Q(2)
+Q(3)                   3      20588   3     208380      69460   10.12
**DENOMINATOR OF RATIO IS RES. SS /RES.DF FROM LINE ABOVE,=   6863

Stationary value is

 5.6417E 1

Coordinates of stationary point

        TEMP     -0.0026
    ADDITIVE      0.2650

Latent roots

    258.1023    112.7310

Canonical variates

           XP
        TEMP   0.5175   0.8557
    ADDITIVE   0.8557  -0.5175
```

References

Box, G.E.P. and Wilson, K.B. (1951) On the experimental attainment of optimum conditions. J. R. Statist. Soc. B, **13**, 1–45.

Gill, J.L. (1978) Design and analysis of experiments in the animal and medical sciences. Ames, Iowa: Iowa State University Press.

Indirect Calorimetry analysed using Genstat

T.J. Cole
M.R.C. Dunn Nutrition Unit
Downhams Lane
Milton Road
Cambridge CB4 1XJ
United Kingdom

All living animals produce heat. This heat is derived from the chemical combustion of stored food, a process which uses up oxygen and produces carbon dioxide and water. There is an exact analogy here with a coal fire, which also burns fuel with oxygen and gives out heat. In humans, the rate of heat production is often called the metabolic rate and is usually measured in basal or resting conditions, the BMR or RMR for short. As well as differences between individuals, there are many aspects of lifestyle that influence MR, which are of interest to a Nutrition Research Unit. By the Law of energy conservation, the amount of heat energy an individual generates has to be matched by the amount of fuel he or she consumes. The amount of fuel (i.e. food) can be obtained by measuring the rate at which oxygen is consumed, or conversely by the rate at which CO_2 is produced. Thus the equivalence of energy in and heat out means that the metabolic rate can be measured either directly – direct calorimetry – by measuring the amount of heat and sweat produced by the subject, or indirectly – indirect calorimetry – by measuring oxygen uptake and CO_2 production.

At the Dunn there are three whole-body calorimeters, rooms large enough for a human subject to live in for several days at a time. All three have facilities for indirect calorimetry, i.e. they measure small changes in oxygen and CO_2 concentration in the air passing through them, whilst the third and biggest calorimeter also has facilities for direct calorimetry, which are not of relevance here.

The concentration of oxygen in ambient air is about 20.94%, and for CO_2 0.04%. After air has passed through the calorimeter the oxygen concentration falls, and CO_2 rises, by typically 0.1%. This small change obviously requires very sensitive gas analysis equipment but the algebra of the situation also needs to be well understood to give appropriate answers (Brown, Cole, Dauncey, Marrs and Murgatroyd, 1983).

The air flow rate into the calorimeter is finely controlled but the flow rate out is in general unknown. This is because a subject's net consumption of gas is non-zero – more oxygen is consumed than the amount of CO_2 and water vapour produced.

For any gas G in the calorimeter, the following equation holds.

Rate of increase of Gas in chamber = (rate of Gas flow into chamber)
− (rate of Gas flow out of chamber)
+ (rate of Gas production by subject)

Gas volumes here have to be comparable and so are corrected to STP. Also, the gas concentrations are measured in dry air and need correcting to damp air. The two corrections are combined as $K = \left(1-\frac{P_w}{P}\right)\frac{T_s}{P_s} \times \frac{P}{T}$, where P and T are pressure and temperature, suffix w means water vapour and s means standard.

Hence:

$$\frac{d}{dt}(VK_o f_{G_o}) = F_i K_i f_{G_i} - F_o K_o f_{G_o} + R_G,$$

where the suffices $_i$ and $_o$ refer to ingoing and outgoing air, V is the constant calorimeter volume, F_i is the known flow rate but needs a barometric pressure adjustment, and f_G is the concentration of gas G in dry air. The nitrogen flow rate, f_N, is obtained by subtraction, given that dry air consists only of oxygen, CO_2 and nitrogen. In the steady state, the differential term vanishes. If the gas G is taken to be nitrogen, which is neither consumed nor produced, this equation allows the unknown flow rate out, F_o, to be calculated. This leads to the general equation for R_G,

$$R_G = F_i K_k f_{N_i}\left(\frac{f_{G_o}}{f_{N_o}} - \frac{f_{G_i}}{f_{N_i}}\right) + VK_o f_{N_o}\frac{d}{dt}\left(\frac{f_{G_o}}{f_{N_o}}\right).$$

The calorimeter works by taking measurements of pressure, temperature and gas concentration every five minutes, usually in the outgoing air, but every sixth measurement is taken on the ingoing air instead. In addition, every 3 hours two samples are used to re-calibrate the gas analysers.

The results are saved as variates in a file, together with a factor indicating whether each sample is calibration, outgoing air or ingoing air.

The computation of oxygen uptake and CO_2 production from the data is largely trivial when done in Genstat, with just one exception. The gas concentrations f_G and f_N are required for both ingoing and outgoing air at every sample, whereas they are measured for only one or the other. This requires that the ingoing air concentrations, measured for only one sample in six, need interpolating for the other 5 samples. A macro was needed to do this interpolation, making no assumptions about the structure of the missing and non-missing values, and optionally smoothing the known values prior to interpolation.

Interpolation is an interesting problem – for each unknown element four quantities are required: the previous known value and how many elements back it is, plus the next known value and how many elements forward it is. As a corollary, this means that a minimum of 4 variates are needed for workspace. Looking forwards and backwards are obviously analogous, so the problem resolves into: what is the previous known value and how far back is it?

One solution, suggested to me by Nick Maclaren, would be to take a copy of the variate and, by repeatedly using EQUATE with a scalar introduce a phase-shift between the two variates. As soon as a missing value in the original variate is matched with a non-missing value in the shifted variate this must be the previous known value in the original variate. Its distance back is given by the number of phase-shifts needed. This process continues until all missing values have been satisfactorily matched. The code for this macro, SELECT1, is shown on the next page.

As a method, this is ingenious and codes compactly but it does have a serious disadvantage: it can take a lot of time, depending on how many passes are required, which in turn depends on how far apart the known values are. In one example the known values are 36 elements apart, thus requiring 36 passes.

For this reason I tried to devise a method of doing the interpolation in a single pass. This method runs more than twice as fast and at the same time uses two less variates in workspace. As noted above, the minimum workspace for linear interpolation is 4 standard length variates, of which one can be the returned variate. This macro uses, in addition, one standard length factor, four structures equal in length to the number of indexed elements and two scalars.

The code for the macro, SELECT2, given on the next page, needs some explanation. Firstly, the positions of the indexed elements are saved in integer ISTATUS and the corresponding values are saved in variate V. At this stage, variate VIN is redundant and can be overwritten by VOUT. The moving average section of the macro can be ignored and the next step is to create two complementary variates VOUT and N1, the former consisting of zeros throughout, except for ones at the indexed elements, and the latter its complement. I used COPY rather than CALC here to save time.

The trick now is to accumulate VOUT and N1, using CUM. VOUT then consists of a string of zeros, *1s*, *2s* etc., each integer defining the position in V of the previous indexed element. It is then easy to define the corresponding factor FAC, indexed to V, using GROUP and VOUT can be replaced by the VARFAC of FAC.

A similar system works for N1, which is needed to indicate how far back the nearest indexed value is. Thus, for an element that is itself indexed, the value in N1 should be zero. This can be arranged by COPYing the indexed elements from N1 into V (which in turn points to FAC) and subtracting the VARFAC of FAC from N1.

Exactly the same procedure yields V2 and N2 and the interpolation can then be carried out, at the penultimate line.

The calorimeters have been used for a number of experiments. One compared the effect of eating 3 big meals a day (gorging) with the same amount of energy eaten as lots of small meals (nibbling). The results showed the metabolic rate on the nibbling diet was raised during the day and lowered during the night, relative to the gorging diet.

Another experiment looked to see how the body reacted to substantial overfeeding and found that it caused an increase in metabolic rate. However, the increase was only a small fraction of the excess food energy consumed, showing that the body has only an inefficient mechanism for handling overfeeding.

Reference

Brown, D., Cole, T.J.,and Dauncey, M.J. (1983) The analysis of gaseous exchange in open-circuit, indirect calorimetry. Med. and Biol. Eng. and Comput. **21** (in press).

The Macros

```
'MACRO' SELECT1 $
'' Given the elements of VIN
   indexed by level LEVEL of factor STATUS,
   macro SELECT calculates values for the
   remaining elements by linear interpolation
   and returns the result in variate VOUT.
''
'LOCAL' ISTATUS,DUMMY,MV,ZERO,ONE,TWO,COUNT,FILL,
        A,B,C,V,NV,VA,VB,VC,V2,N1,N2,L1,L2,I
'START'
'REST' VIN $ STATUS=LEVEL ; ISTATUS :VIN
'RUN'
```

```
'SCAL' DUMMY :MV=* :ZERO=0 :ONE=1 :TWO=2
:COUNT=0 :FILL=1E-9
'VARI' A,B,C,V,NV $ISTATUS
'VARI' VA,VB,VC,V2,N1,N2 $VIN
'COPY' V=VIN $ ISTATUS
'JUMP' L1*(MAVE.LT.ONE)
'' Moving Average Calculation.
   Each element of V is averaged with
   MAVE elements on either side of it.
   ''
   'CALC' NV=V.NE.MV
   'EQUA' A,B=V
   'FOR' I=MAVE!(ONE)
     'EQUA' DUMMY,A,C,B=A,MV,B,MV,C
     'CALC' V=V+VSUM(A,B)
     :NV=NV+TWO-VNMV(A,B)
   'REPEAT'
   'CALC' V=V/NV
'LABEL' L1
'COPY' VOUT $ISTATUS = V :N1,N2=ONE
'CALC' ELEM(VOUT;ONE)=ELEM(V;ONE)
:ELEM(VOUT;NVAL(VOUT))=ELEM(V;NVAL(V))
:VOUT=REPMV(FILL)
'EQUA' VA,VB,V2=VOUT
'LABEL' L2
  'EQUA' DUMMY,VA,VC,VB=VA,FILL,VB,FILL,VC
  'CALC' COUNT=COUNT+ONE
  :N1,VOUT,N2,V2=N1,VOUT,N2,V2*(2(VOUT,V2).NE.FILL)
            +COUNT,VA,COUNT,VB*(2(VOUT,V2 .EQ.FILL)
'JUMP' L2*(SUM(VOUT.EQ.FILL)+SUM(V2.EQ.FILL).GT.ZERO)
'CALC' VOUT=(VOUT*N2+N1*V2)/(N1+N2)
'DEVA' ISTATUS,A,B,C,V,NV,VA,VB,VC,V2,N1,N2
'ENDMACRO'


'MACRO' SELECT2 $
'' Given the elements of variate VIN
   indexed by level LEVEL of factor STATUS,
   macro SELECT calculates values for the
   remaining elements by linear inperpolation
   and returns the result in variate VOUT.
''
'LOCAL' ISTATUS,DUMMY,MV,ZERO,ONE,TWO,NUMER,V2,
        DENOM,N1,N2,V,NV,A,B,C,FAC,VAC,NAC,LABEL
'START'
'REST' VIN $ STATUS=LEVEL ;ISTATUS  :VIN
'RUN'
```

```
'SCAL' DUMMY :MV=* :ZERO=0 :ONE=1 :TWO=2
'SET' NUMER=VOUT,V2  :DENOM=N1,N2
'VARI' NUMER,DENOM $VIN  :V,NV,A,B,C $ISTATUS
'FACT' FAC $C,VIN
'COPY' V=VIN $ ISTATUS
'JUMP' LABEL*(MAVE.LT.ONE)
'' Moving Average Calculation.
   Each element of V is averaged with
   MAVE elements on either side of it.
   ''
   'CALC' NV=V.NE.MV
   'EQUA' A,B=V
   'FOR' NAC=MAVE!(ONE)
     'EQUA' DUMMY,A,C,B=A,MV,B,MV,C
     'CALC' V=V+VSUM(A,B)
     :NV=NV+TWO-VNMV(A,B)
   'REPEAT'
   'CALC' V=V/NV
'LABEL' LABEL
'COPY' VOUT=ZERO  :VOUT$ISTATUS=ONE
:N1=ONE  :N1$ISTATUS=ZERO  :N2=N1
'CALC' VOUT,N1=CUM(VOUT,N1)
:N2=N2-N1+MAX(N1)
'EQUA' V2=ZERO,VOUT
'CALC' V2=V2+ONE
:DUMMY=MAX(VOUT)+ONE
:NUMER=NUMER+ZERO/(NUMER.NE.ZERO,DUMMY)
'FOR' VAC=NUMER ;NAC=DENOM
   'GROUP' FAC=RANK(VAC;C)
   'COPY' C=V
   'CALC' VAC=VARFAC(FAC)
   'COPY' C=NAC$ISTATUS
   'CALC' NAC=NAC-VARFAC(FAC)
'REPEAT'
'COPY' DENOM$ISTATUS=ONE
'CALC' DENOM=ONE/DENOM
:NUMER=NUMER*DENOM
:VOUT=VSUM(NUMER)/VSUM(DENOM)
'DEVA' ISTATUS,V2,DENOM,V,NV,A,B,C,FAC
'ENDMACRO'
```

Rationalising the macros for Multivariate Analysis

P.G.N. Digby
S.A. Harding
Statistics Department
Rothamsted Experimental Station
Harpenden
Hertfordshire AL5 2JQ
United Kingdom

This article describes attempts to improve the use of certain multivariate analyses in Genstat. It is based on a talk given by the first-named author at the Genstat conference in October 1983. At this stage we have concentrated on the more commonly used analyses; thus many of the macros, e.g. CANCOR and GENPROC, remain unchanged. In accord with the talk the material below is given under three headings: Why?, How? and So What?

Why?

There are two main reasons why we have chosen to rationalise some of the macros:

1) the black-box approach is too restrictive, especially for interactive use;

2) many methods have common requirements, in particular the graphical display of results.

To illustrate the former, let us consider a typical problem: we have a data set of moderate size, say 100 units × 15 variates; we expect that most of the structure in these data can be explained within a solution of low dimensionality, we hope that two dimensions will suffice but accept that more dimensions may be necessary. We will use principal components analysis; having determined an appropriate number of dimensions, by inspecting the latent roots, we want the scores and residuals from the required solution to be printed. Also, we want to plot the scores in pairs of dimensions, e.g. 2v1, 3v1, 3v2, up to the required dimensionality. In the biplot style we may wish to add points for the variates to these graphs.

There are three ways to achieve this. We could guess at the required number of dimensions, write and run the Genstat program for this guess, and hope that we got it right. A better approach, at least in terms of obtaining the solution which we want, is to run the analysis and print all latent roots (and vectors), determine the required number of dimensions and then rewrite the program for that number of dimensions and rerun it. Of course, this means that the analysis is done twice. The best way is to run the analysis, printing the roots and vectors only, and saving all the results; now we copy the required set of scores into variates and calculate residuals from the remaining scores; finally the variates of scores and residuals can be printed (in parallel, to save paper) and graphs can be produced. Obviously this method is best, since we get what we want and we only have to run the analysis once; however, it is at best tedious and certainly liable to errors.

The second reason for reorganising some of the macros is obvious to anyone who has written the Genstat statements required to plot the results, in pairs of dimensions up to a specified number, from principal components analysis. The following methods all need similar, or identical, graphical output: principal components analysis, biplots, correspondence analysis. With a little modification results from other methods can be covered by any such section of Genstat code: these methods include principal coordinates analysis and the analysis of skew symmetric matrices. Similarly, the requirements for printed output from such analyses can be dealt with by a single section of code, i.e. a macro.

Many methods produce results which include latent roots or some similar set of values indicating the relative importance of the dimensions of the solution. Clearly, any method of displaying and examining these will be common to all such methods.

How?

This section can be split into two parts: firstly, the design of the group of macros; secondly, the implementation of the macros, in particular the problems encountered. An example is given after the first subsection.

How? – the design

It is apparent from the principal components analysis example discussed above that we require separate macros for different parts of the complete analysis, e.g. we may have one macro for the analysis itself, another to print the results and a third to do the plotting. In practise, we have found that some of these individual tasks are fairly complicated, when implemented in a general way, so that *secondary macros* are sometimes needed in addition to the *main macro* for a task. To avoid confusion we think in terms of *modules*, each of which may consist of one or more macros.

In general, we consider there to be six phases to an analysis:

(1) preliminary transformations;
(2) the computational part of the analysis;
(3) some determination of dimensionality;
(4) printing the results;
(5) plotting the results;
(6) subsequent (summary) analysis.

Subject to the comments below, each of these will have a separate module. In some situations (1) will not exist or will be sufficiently trivial, and standard, to be subsumed into (2). For some analyses (6) will not exist. We separate (4) from (5), and present them in that order, because it is expedient to use (4) to generate the variates needed by both (4) and (5). This is not restrictive because the printing module can be invoked, but with the actual printing of results suppressed by an option, to generate the variates to be plotted.

We have already remarked that the module for (3), a macro called `NLR`, will be common to many analyses. Also modules for (4), (5) and to some extent (6) will be common within groups of analyses. Thus, we can identify groups of analyses and supply a separate set of modules for each group; we call such a set of modules a *tool kit*. Of course each tool kit will have its own copy of `NLR`; however, apart from that module, there will be little, if anything, in common between two tool kits. At present we have only two tool kits: one for the group of analyses identified in Section 1, e.g. biplots; the second is for canonical variate analysis and related methods which can be used when the units are grouped. Subsequent tool kits, e.g. for Procrustes-type analyses, will be prepared as the need arises and opportunity permits.

For 'standard' analyses, where one merely wishes to invoke the relevant modules in order, with no user intervention, each tool kit contains a number of *conglomerates*. These are macros which merely call the modules as required. For example, the conglomerate `CORRESP` does a complete correspondence analysis; this should be useful to people with existing programs using the current macro `CORRESP`, since they should need minimal change to use the new scheme: in this example the disadvantage of having to make changes should be far outweighed by the superior output from the tool kit approach.

Although the use of conglomerates should help, we appreciate that there will be initial difficulties in using the tool kits. For this reason we provide a number of aids or *services*. Options for modules which print and plot results are specified by scalars which are decoded to determine what output is required. To assist with the encoding of these scalars, each tool kit contains a module that will set the scalars appropriately from a list of names given by the user.

Each tool kit contains its own help information, at a number of levels. We continue to use the approach of having pointers to all the macros within a module so that retrieving the pointer will automatically retrieve all the macros required. All modules should be retrieved via their pointers. Each conglomerate has two associated pointers: one for the conglomerate itself, the other for the conglomerate and all the required modules. A pointer is also available to retrieve the entire contents of the tool kit.

Example

Figure 1 shows the modules for the tool kit for canonical variate analysis and for the analyses of between-and-within-group distances (Digby and Gower, 1981). The module BGDIST is a preliminary module, type (1) above; CVASCOR and BWGDANAL provide the computational parts of the two analyses, respectively. Note that a third module may be added for an additional type of analysis, currently under consideration: because of the modular design described above this will simply fit in as a third route from the user's input data structures to the subsequent modules, NLR etc. The modules NLR, CVAPRINT and CVAPLOT perform obvious tasks; CVAOPT is the service for option setting; TABDIST produces the analysis-of-distance table (Digby and Gower, 1981), analagous to an analysis-of-variance table.

It can be seen from figure 1 that the user's input to the tool kit is minimal: the data structures required are a grouping factor, G, and either a set of variates, VSET, or a symmetric matrix of squared-distances D; if CVAOPT is to be used some names will be needed, also the user may wish to change the numbers of dimensions to print and to plot from those determined automatically by NLR. Most of the useful data structures formed by the modules are not destroyed, so they are available to the user subsequent to the analysis.

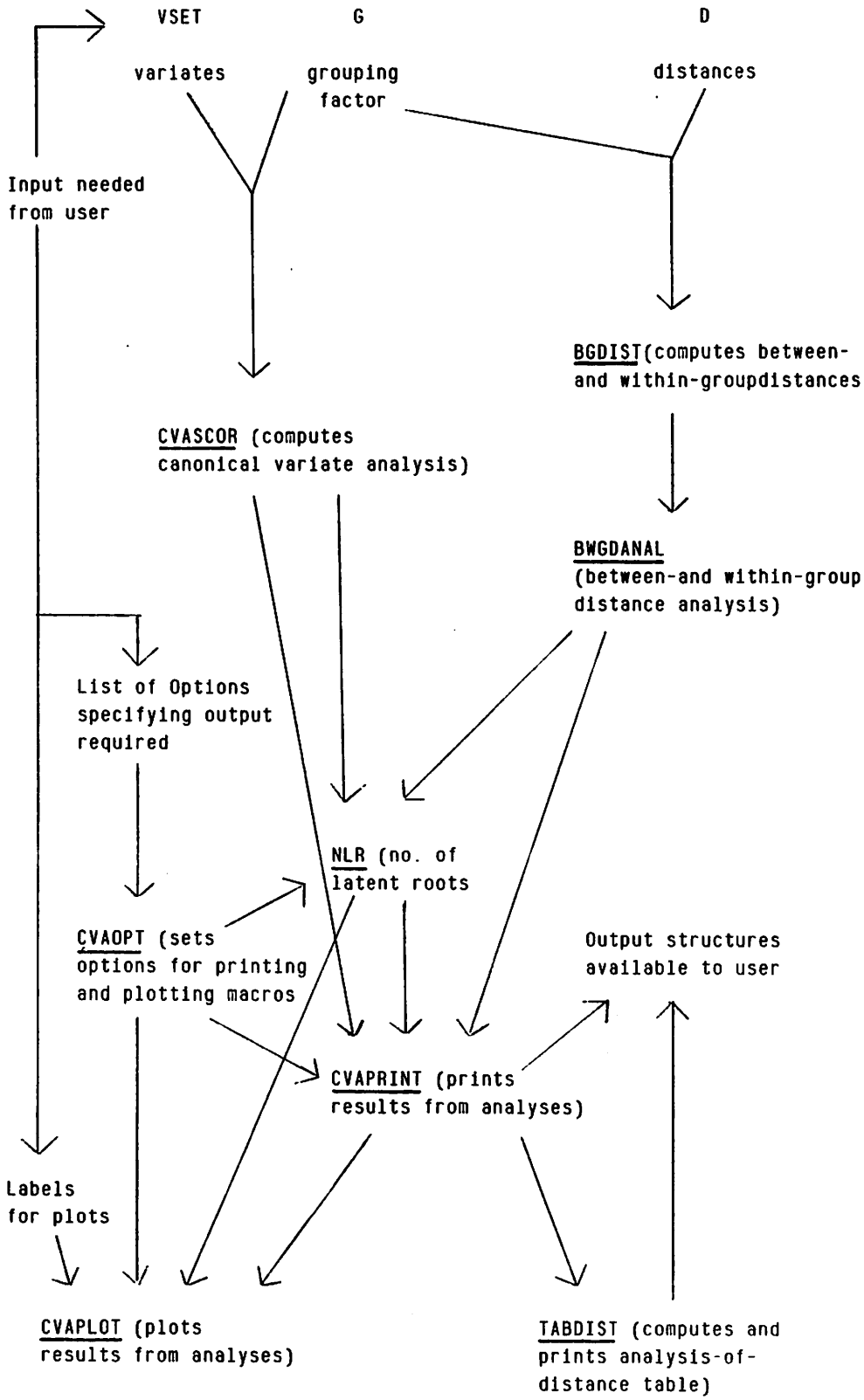
How? – the problems

Apart from the general problems encountered when preparing macros, we met two particular types of problem because of the approach adopted. The first involves communications, the second is concerned with the examination of latent roots.

For the modular approach above it is critical that the modules are able to communicate with each other. Basically, this communication is in terms of the data structures which are passed from one module to another: because there is no parameter passing to macros in Genstat we have had to adopt a very strict policy for identifying these data structures. This is unfortunate as far as the user is concerned, for example the grouping factor for canonical variate analysis (see figure 1) must have identifier G, even though the identifier SPECIES would be more appropriate in some situations. All the arrows in figure 1 show lines of communication; associated with each of these is one or more data structures; the sheer number of arrows indicates the complexity of the problem.

The second problem is very specific: given a set of latent roots, how many dimensions should be kept? There is no clear-cut answer to this, although various ad hoc rules have been suggested. We have assumed that one may wish to print results for a certain number of dimensions (NDIMP) but only produce graphs for a smaller number of dimensions (NDIMG). At present, we use rules based on finite differences to determine these values within the module NLR, although we may find a better way of doing this; in any event the statements which calculate these values are clearly marked within the macro so that users can change them if they so wish. Of course, this problem disappears when the tool kits are used interactively; NLR optionally produces a scree graph and a table showing the roots, their cumulative values and finite differences, so that the user merely sets the scalars as required before going on to the subsequent modules.

Figure 1 – Tool kit for CVA and related analyses



So What?

The immediate consequence of this work is the way in which the user can use multivariate methods. We believe that existing programs can use the new scheme, via the conglomerates, with minimal change (if indeed any is required). What is new is the opportunity for users to interact easily with the data analysis.

The longer-term consequence relates to any new version of Genstat. If the modular approach described here is the right way to use multivariate methods (and we believe that it is) then should the multivariate directives in Genstat be redesigned along these lines? As far as we are concerned, the answer to this is an unequivocal 'yes'. Now the problem is how to redesign the directives to achieve this approach; a problem to which we do not, at present, know the answer.

References

Digby, P.G.N. and Gower, J.C. (1981) Ordination between-and within-groups applied to soil classification. Down to Earth Statistics: Solutions looking for Geological Problems. Syracuse University Geology Contributions (D.F. Merriam ed), 63–75.

New facilities for GRAPH in version 4.04

K.J. Martin
and R. Gough
East Malling Research Station
East Malling
Maidstone
Kent ME19 6BJ
United Kingdom

A number of new facilities in the GRAPH directive have been introduced in version 4.04. This paper describes the changes from previous versions.

Graph annotation

Graph title

A new option, TITLE = *heading identifier*, allows a one-line heading immediately above the graph frame. As with the options ATY and ATX, care must be taken that the heading is not too long, otherwise truncation or even overwriting of the heading may occur.

X-axis labelling

ATX may now be equated to a heading containing more than one line. Each line is centred. We have found it useful to construct headings using JOIN, e.g.

```
'HEAD'          H=''OCTOBER 1983''
'SCAL'          TEMP=15:RAIN=2
'JOIN'          HX=H,TEMP,RAIN$1,(/,10)2
'GRAPH/ATX=HX'  etc
```

will label the X-axis with the current values of the scalars, beneath a heading.

If an X-axis label of more than one line is used, the graph and its labels will not all be contained on a single page of line printer output.

Default labelling

When none of the options ATY, ATX or TITLE is used, by default a title is printed below the graph, indicating which Y and X structures have been plotted.

Annotation within the frame

Text can now be written within the graph frame. To do this, the appropriate item in the Y and X lists is a structure containing one value, usually a scalar, which indicates the starting point for printing. The corresponding item in the *ilist3* of plotting symbols is the heading to be printed, which unlike other headings in this list need not contain only a single character. The corresponding character in the heading specifying the type of plot is a new symbol, T. Thus:

```
'HEAD'   HT=''<--Y=A+B*X''
  :      H=''LPT'':HDOT=''.''
  :      HAST=''*''
'SCAL'   Y=3:X=16
'GRAPH'  Y(1,2),Y;X(1,2),X$H;HDOT,HAST,HT
```

will plot Y(1) against X(1) as a line, using . as the plotting symbol, Y(2) against X(2) as points, using *, and the text contained in heading HT in the frame starting at X=16, Y=3.

The restriction to 10 graphs per frame refers only to line and point plots. Any number of text annotations may be added.

Side-by-side graphs

The options YCNT and XCNT now work slightly differently. Instead of producing four small graphs to a page in a 2 × 2 arrangement, the directive will now produce graphs of a size specified by NRF and NCF and print as many side-by-side as possible. On a line printer, this allows a number of larger graphs to be produced one beneath the other using a single directive.

Plotting on graphics devices

The DEVI Option

This new option is by default set to produce graphs on a line printer. At sites where graph plotters or graphics VDUs are available, setting this option will enable the correct device to be specified. The appropriate output channel needs also to be selected using OUTPUT. The association of particular devices with settings of DEVI is site-dependent.

Differences between graphs on the line printer and graphics devices

If a line-printer graph is converted to one for a graphics device, by changing the setting of DEVI, the results will look very similar. The main differences are that the graph frame is drawn as a continuous line and line plots are drawn as continuous or broken lines. However, the use of graphics devices enables a much better-looking result to be produced, with scope for producing very large graphs or those with different colours or symbol sizes.

The internal buffer

The setting BUFF=N should normally be used with output to graphics devices, as it saves space and leads to greater plotting accuracy. On plotters, each of a number of coincident points will then be drawn rather than numbers or lists of coincidences being produced. The default BUFF=Y, specifying the use of an internal page buffer, must be used for line printer graphs.

Symbol size

The new option SYMB = *scalar* specifies the size in centimetres used for plotting symbols on graphics devices. The same size of symbols is also used for characters in headings and axis scales, so that too large a size produces overlapping characters.

Colour

A generalisation of the heading which specifies the type of plot allows the choice of different pens on a plotter, enabling different colours or thicknesses to be used on one graph. E.g.

```
'HEAD'                  H=''L1P2L3S1''
'GRAPH/DEVI=1,BUFF=N'   Y(1...4);X(1...4)$H
```

will produce a line with pen 1, points with pen 2, a second line with pen 3 and a smoothed line with pen 1.

Broken lines

If the plotting symbols in *ilist3* of the GRAPH directive are digits and correspond to line plots and a graphics device is being used, broken lines will be drawn, the digit representing the ratio of unbroken to broken length. E.g.

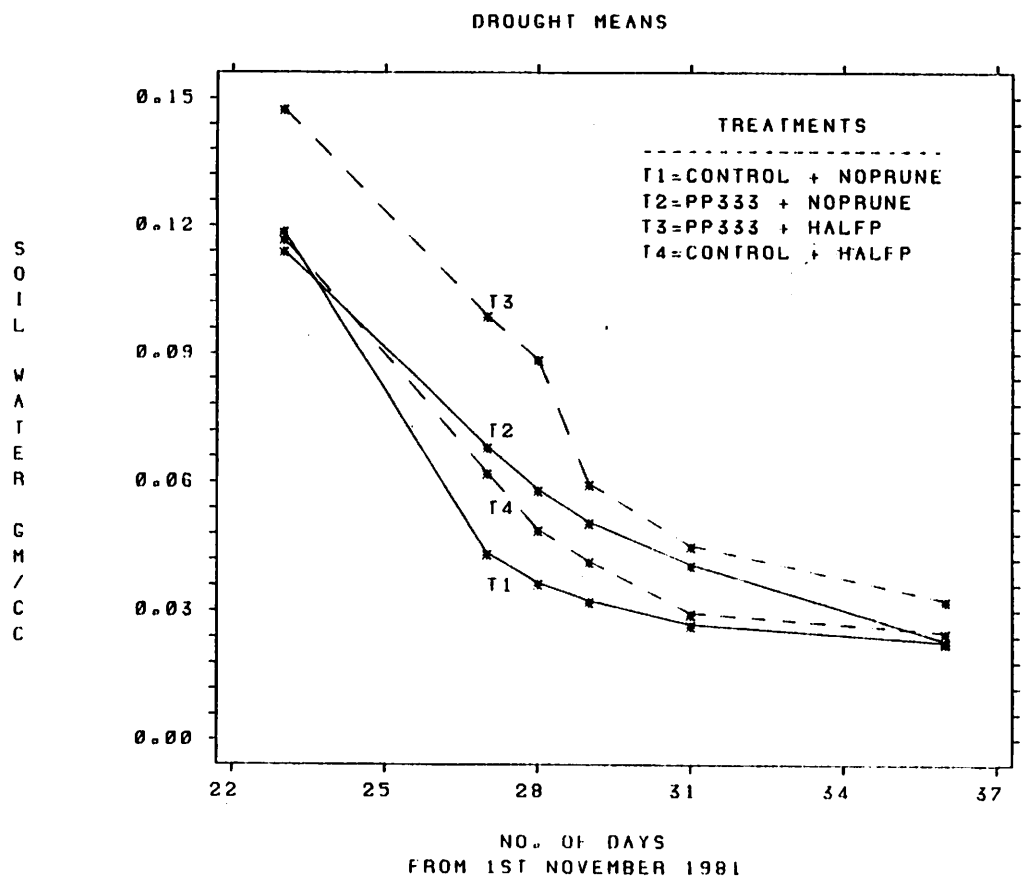
```
'HEAD'                  H=''L'':H1=''G013''
'GRAPH/DEVI=1,BUFF=N'   Y(1...4);X(1...4)$H;H1
```

The G and 0 will result in the drawing of two continuous lines and the 1 and 3 will produce two lines with different degrees of brokenness. The broken line facility is, however, not completely satisfactory at present because the degree of brokenness appears to be calculated in the X–direction rather than along the line plotted and the points joined by broken lines cause irregularities in the length of the dashes.

Finally, to illustrate a number of the new facilities together, the following directives produced the graph below, drawn by a Benson 1202 plotter.

```
'HEAD'     HY=''SOIL WATER GM/CC''
          :HX=''
            NO. OF DAYS
           FROM 1ST NOVEMBER 1981''
          :MHEAD2=''       DROUGHT MEANS''
          :KEY(1)=''T1''
          :KEY(2)=''T4''
          :KEY(3)=''T2''
          :KEY(4)=''T3''
          :HK=''      TREATMENTS
--------------------
```

```
T1=CONTROL + NOPRUNE
T2=PP333 + NOPRUNE
T3=PP333 + HALFP
T4=CONTROL + HALFP''
          :HL=''PPPPLLLLTTTTT''
          :HSY(1)=''0''
          :HSY(2)=''1''
'VARI'     TD(1...4)$6
          :DATE1=23,27,28,29,31,36
          :BV1=0,0.15,22,37
'SCAL'     PD(1)=0.036 : PD(2)=0.054
          :PD(3)=0.072 : PD(4)=0.102
          :XC=27    :Y=0.144    :X=30
'OUTPUT'   2
'GRAPH/HY,HX,BV=BV1,NRF=26,NCF=51,TITLE=MHEAD2,DEVI=1,BUFF=N,SYMB=0.2'
           (TD(1...4))2,PD(1...4),Y ; (DATE1)8,(XC)4,X$HL
          : (*)4,(HSY(1,2))2,KEY(1...4),HK
'OUTPUT'   1
```



Plans for Genstat Mark 5

R.W. Payne
Statistics Department
Rothamsted Experimental Station
Harpenden
Hertfordshire AL5 2JQ
United Kingdom

Introduction

The last major change to Genstat was in 1973, when Mark 3 replaced Mark 2. In the ten-year lifetime of Genstat releases Mark 3 and Mark 4, the computing environment has changed considerably. In 1973, interactive computing was unavailable (or else strongly discouraged) at most computer centres, input was generally via cards or paper tape and it was difficult to produce graphics, except on line printers. Genstat 4.04 will run interactively but it is not as convenient as programs specially designed for interactive work, such as GLIM. It can also produce high-quality plots of points and lines; however there are other types of graphical output (e.g. pie charts, histograms, contour plots) which should be made available on graph plotters, as well as interactive facilities such as the use of cursors or light pens to identify interesting points. It is to the credit of the original design of Genstat Mark 3 that it has been possible to incorporate many improvements in an evolutionary way, without major changes to the existing syntax. However, there are now various inconsistencies in the syntax, directives with overlapping facilities, redundant directives and data types etc., all of which could usefully be rationalised.

Consequently, the Genstat Programs Committee has suggested the following strategy. Development of the current Genstat will be frozen at Release 4.04. This release will continue to be supported and errors will be corrected but future improvements will be made only in the context of a much revised Genstat Mark 5, some details of which are given below.

As final decisions have not yet been made, we would welcome any comments, alternative suggestions – even encouragement, if appropriate!

Areas for Improvement

Ease of Use

As John Fenlon described at the Genstat Conference (see next Newsletter), Genstat consistently scores highly in Package Comparisons for the facilities which it offers, but less well for its ease of use. Genstat is not an easy tool for those without an aptitude for statistics and for computing. This is not something that we should view with complacency. With the growing availability of computers to non-statisticians, within the Agricultural Research Council and elsewhere, there will be more non-statisticians doing their own analyses. If they regard Genstat as too difficult, they may be led to use some other package which appears easier but fails to provide the correct method of analysis.

Genstat Mark 5 should improve this situation in the following ways. The simplified and more consistent syntax (Section 3.1.) should be easier to learn and remember. It will also require us to rewrite the Manual(!) The new PROCEDURES, which replace the MACROS of Mark 3 and 4, will look very much like directives and will enable 'customised' versions of Genstat to be produced for non-expert users in various fields of application (Section 3.3.). The conversational interface, described by Peter Lane elsewhere in this issue, will provide an easy method for novices to start using Genstat.

Graphics

The need for high-quality interactive graphics has already been mentioned. Further suggestions and ideas from users would be very welcome.

Text Handling

Names and headings will be replaced by a more general text vector. There will no longer be any limit on the number of characters per unit of the vector, so that each unit may be regarded as representing a line of a document. Text will be able to be PRINTed (with the number of characters printed per unit controlled by the format) and GROUPed into factors as now. There will also be a text editor allowing the usual substitution, deletion and insertion of strings, deletion and insertion of lines of text and insertion of other text vectors, as well as allowing numerical values to be inserted from variates, tables etc, according to user-defined formats. It will be possible to save a copy of output in a secondary file and then read it back into a text vector, also to save the most recent lines of input instructions. Text can be substituted into a program by preceding the identifier of the text vector by the characters '##'. Thus, if an error occurs during an interactive run, the offending instruction can be edited and resubmitted.

Statistical Techniques

New facilities under discussion include a module for fitting standard curves (polynomials, ratios of polynomials, growth curves, exponentials, splines etc) without requiring the specialised knowledge needed to use OPTIMISE, the ability to fit distributions (Normal, Double Normal, Poisson, Geometric etc) to counts formed by HISTOGRAM, directives or procedures for EDA and a module for Probit Analysis which would differ from the GLM facilities by providing specially designed output, the fitting of natural mortality and immunity, and the analysis of Wadley's Problem.

Specification

Syntax

The Mark 3-4 syntax requires special characters (') at the start and end of directive names. One of these is redundant – it is only necessary to distinguish the directive name from identifiers of structures etc. This can be done either be marking the start of a directive (as in GLIM) or the end of the previous one (as in MLP). Of these two alternatives, the latter seems more sensible. For interactive work it is more important to know that the end of a directive has been reached (so that it can be executed) than to know that a new one is about to begin. Users of GLIM-3 have always found the form $FIT X $ rather unnatural.

The-end-of-directive symbol will be colon (:). Users will be able to specify whether newline is to be synonymous with space (as in Mark 4) or with colon, the latter being appropriate for the many Genstat users who specify just one instruction per line. Single quotes (') can thus be used to delimit text; comments will be delimited by double quotes ("). For options etc which expect text, the quotes can be omitted. There is no need for any special continuation character when newline has been made synonymous with colon, one can write e.g.

```
ANOVA    X, Y  "
" , Z
```

As the newline is within a comment, it is ignored.

Directives will have options and keywords with lists (just like many of the Mark 3–4 directives). Lists specified by keywords are in parallel, whereas options modify globally the way in which the directive works (i.e. for all items in the lists). Different options with their settings or keywords with lists will be separated by semi-colons (;) and the lists of option settings will be enclosed in square brackets. Formally the syntax can be defined as follows:

<directive>::=
<directive name> {[<option-keyword>=<value> {;<option-keyword>=<value> }]}[1]
{ <keyword> = <item(s)> {;<keyword> = <item(s)> }}[1]:

Keywords and option keywords will be able to be abbreviated to the minimum unambiguous number of letters (within the directive concerned). The symbol for repeating the previous directive (colon in Marks 3-4) will be ampersand (&). Perhaps surprisingly, all Mark 4 directives (except CALCULATE which has *identifiers=expression* instead of keywords and lists) can be specified within this framework – even those that previously required $, like declarations, RESTRICT and EXTRACT. The syntax has rules that are very much simpler than those of Marks 3-4; it is also more succinct and thus more convenient, particularly for interactive work, as shown in the examples in Section 4.

Data Structures

Scalars, variates, factors, matrices, symmats, diagmats and tables will be available with few changes from Mark 4.

Integers have always had a somewhat anomalous role in Genstat. It is only recently that they have been usable in CALCULATE, although the actual arithmetic is done using real variables. Variates can be used wherever integers are required and so, to simplify Genstat both for the implementers and for users (who should not be required to distinguish between different types of numbers), integers will be abolished.

Sets will also be abolished. SET/E and SET/M become *expressions* and *formulae* respectively and pointers will provide the substitution of lists of identifiers previously done by SETS of structures. However, this substitution will occur at RUN time not at compile time.

Elements of pointers can be referred to by using suffices; P[1] is the first element (i.e. structure) of the pointer P, while P[] refers to all the structures in P. Pointers can also be declared with labels associated with their elements; one can then refer to an element by its label instead of its element number e.g. P['A','B'] for the elements labelled A and B. As explained in the next Section, substitution of the scalar S in P[S] will take place at RUN time.

Pointers can, of course, have other pointers as their elements, so the complete range of hierarchical data structures, described by Lamacraft and Payne, can be represented. P[1][2] refers to the second element of (pointer) P[1]; P[1,2][1,2] is the list of elements P[1][1], P[1][2], P[2][1], P[2][2]; and so on.

Suffixed identifiers will be replaced by pointers and their elements. Thus, if the user declares a variate YIELD[1], the compiler will automatically set up a pointer called YIELD and give it one element, defined to be a variate. Use of square brackets for elements of pointers and other changes in bracketing rules, not described here, mean that it is no longer necessary to make functions in CALCULATE system reserved words.

Structures like SSPs will also be stored as pointers – the SSP has 3 elements, 'SS', 'MEANS' and 'REP'.

Control Structures etc

A major source of difficulty for novice and expert Genstat users alike has been the dichotomy between compilation and execution. In Mark 5, it is intended that, by default, each Genstat statement will be compiled and then immediately executed. Users will be able to request that FOR loops be compiled either in full (and then executed) or else incrementally, i.e. on the first pass through the loop, the statements are compiled, executed (and remembered) one at a time; then on subsequent passes they are executed from the stored, compiled form. To avoid any

incompatibility between the first and subsequent passes, compilation will merely interpret and check a statement (e.g. to see that all the keywords are valid). Thus, for example, scalar pre- or post-multipliers will be substituted afresh each time a statement is executed, declarations will be done at execution time not at compile time, suffices of pointers (see the previous section) will not be substituted until the statement containing them is actually executed, and so on. It will thus no longer make any difference to the user when a statement is compiled.

For loops etc in interactive work, there will be a directive to request Genstat to pause after a specified number (*n*, say) of lines of output; the user can then read and digest them, before requesting the next *n* lines.

JUMPS and LABELS seem to be only rarely used in Genstat programs. To accommodate them in Mark 3-4, Genstat instructions have to be submitted in blocks (terminated by the directive RUN). This is incompatible with compilation and execution of single statements so, for simplicity, it is proposed that they be omitted. There will, however, be other program-control structures like IF and CASE.

Macros will be replaced by procedures:

PROC *procname*
Option and Parameter definitions
Statements
ENDPROC

Unlike the identifiers in macros (which are, by default, global), identifiers in a procedure will all be local. Input/output from procedures will be via the option and parameter setting mechanisms. Each of these will define the identifier of a (local) pointer within the procedure which, on execution, will point to the external structure which is to provide data or receive output from the procedure. To facilitate checking, procedures will be compiled when they are defined.

The invocation of a procedure will look exactly like a Genstat directive:

procname [list of *options*] list of *parameter-settings*

Procedures will be accessed automatically from libraries when required – when Genstat Mark 5 meets an instruction it will first check whether it is a valid directive name, then see whether it is one of the procedures already (compiled) in store, otherwise it will check the user's procedure library or else the Genstat procedure library, if attached to the job. We hope that users will write procedures to 'customise' Genstat for a wide range of applications.

Examples

Mark 4.04

(1) Part of an interactive ANOVA

```
...
'TREAT' A*B
'BLOCKS' BLOCK/PLOT
'ANOVA/PR=10' Y ; OUT=OY 'R'
'ANOVA/PR=2' OUT=OY 'R'
'ANOVA/PR=2000' OUT=OY 'R'
...
```

Mark 5

```
...
TREAT A*B
BLOCKS BLOCK/PLOT
ANOVA [PRINT=A] Y
& [PRINT=M]
& [PR=E]
...
```

(Note that, in Mark 5 the ANOVA output structure will, by default be saved, for future use. Also notice how the PRINT option has been spelt out in full in the first ANOVA in the Mark 5 column but subsequently abbreviated.)

(2) Save the means from (1), de-transform (from log(x)) and print

```
'EXTRACT' OY ; A.B $ MEAN=TAB
'CALC' TAB=EXP(TAB)
'PRINT' TAB $ 10.4
```

```
EXTRACT TERM=A.B ; MEAN=TAB
CALC TAB=EXP(TAB)
PRINT TAB
```

(In Mark 5, PRINT will deduce a *sensible* format by default.)

(3) Plot the fitted regression line with the data

```
...
'FIT' X ; FV=F
'HEADING' HX=''LOGARITHM OF DOSE''
: HY=''WEIGHT GAIN''
: HLP=''LP''
'GRAPH/ATX=HX,ATY=HY' F,Y ; X $ HLP
```

```
...
FIT X
ACCESS FITTED=F
GRAPH [YHEAD=WEIGHT GAIN; XHEAD= ''
''LOGARITHM OF DOSE] F,Y ; X ; L,P
```

(Note that, in Mark 5, a continuation can be achieved by "commenting out" a new line with a pair or double quotes.)

Conclusion

This is just a brief sketch of the proposed changes, but I hope that it conveys something of what we hope to achieve. It will not be ready overnight of course but if we go ahead we hope to have a prototype by the next Genstat Conference. We look forward to demonstrating it to you there!

References

Fenlon, J. (1984) Some Considerations in choosing a Package for a Multi-Functional Organisation. Genstat Newsletter, **14**, (to appear)

Lamacraft, R.R. and Payne, R.W. (1980) A new look at data structures for statistical languages. In COMPSTAT 1980 Proceedings in Computational Statistics (Barritt, M.M and Wishart, D.J., ed), Physica–Verlag, Wien, 463–469

A Conversational Interface for Genstat Mark 5

Peter Lane
Rothamsted Experimental Station
Harpenden
Hertfordshire AL5 2JQ
United Kingdom

A common criticism of computer programs now is that they are not friendly. Many people who use computers have come across some program which has been designed for conversational use, and found it a pleasant change from conventional programming. Further acquaintance with such a program may bring problems to light: conversation can get tedious when you are repeating a standard analysis for the tenth time; you may quickly exhaust the range of subjects the program can talk about and find there is no way to do what you want. However, there is no doubt that a conversational style of working is often an attractive way to start using a program and can be convenient for people who only do standard tasks or use the program infrequently.

We propose to make Genstat friendly in interactive use by adding a conversational interface. When you want it, it will stand between you and the command language. From the answers to a series of questions it will effectively construct the necessary Genstat commands, get them executed and deal with problems arising from the results. The style of conversation will be designed for people who use Genstat occasionally or for standard tasks only. Though everyone is a beginner once, no-one stays that way for long, so the beginner will have to ask (by typing ? in response to a prompt) for extra information when a question is not clear.

The method of implementation is not yet decided. Bryan-Jones has described a conversational system using macros in the Genstat language. However, she found problems, particularly in dealing adequately with errors. The alternative is to write the interface in Fortran: it would take more effort but should ensure faster response and closer control.

Only a limited range of facilities will be made available. The interface will lead the user through several standard stages, allowing a choice from the major statistical sections in Genstat. Stages can be skipped, repeated or carried out in any order. In particular, it will be possible to abandon the current stage if something goes wrong.

There is no intention to provide any type of expert system. Genstat will not advise you to transform your data but will ask you if you want to; it will not tell you that your regression relationship is meaningless but will ask you if you want to do diagnostic checks. At all stages, the system will avoid asking questions when information can be deduced from previous answers. All questions will be concise and most will be answerable by a single letter code from a menu, by a number, by an identifier or by a series of these.

We would welcome any advice or suggestions about the design of this conversational interface.

The following hypothetical example is designed to illustrate the features of the planned conversational interface described above. Material shown in italics is typed by a person using Genstat interactively and the rest is produced by Genstat, displayed on a screen or printed at a terminal.

Introduction to the system

genstat

```
Genstat 5 Interactive Statistical Program, Release 1

Please choose how you want to use Genstat.  Type one letter:
  Q  Genstat prompts with ? for commands in the Genstat language.
  S  Genstat gives short prompts for information about what you want to do.
  E  Genstat gives explanation with each prompt for information.
Mode? (Q, S or E)e

You may type the following special symbols in response to a question:
  ?  Asks for more information to help answering the question.
  !  Quits the current stage of analysis and allows choice of new stage.
  :  Chooses default action (this is the first choice when a menu is listed).

Please choose which stage you want next:
  I  Input data              S  Summarize data in tables or graphs
  C  Calculate new data      A  Analyze by standard statistical methods
  G  Form groups             Q  Quit using Genstat
Stage?  (I, C, G, S, A or Q)i
```

Data Input

```
Where are your data?  Type colon (:) if you will type the data directly.
Source of data?  (: or 'name of computer file') 'contam.dat'

The first line of 'contam.dat' is:
  5.54  S1b  32.70   3.730
Please say how each item in this line is to be handled, and then,
if relevant, give an identifier to refer to this and corresponding items later:
  V  Store item as a number (the first value of a variate)
  F  Store item as a group name (a level of a factor).
  Q  Quit: ignore this item.
5.54? (V, F or Q)v
Identifier for this variate? Dryweight

S1b? (V, F or Q)f
Identifier for this factor? Contam
Please list all possible group names, separated by commas, and end with :
? (group names)S1b,S1c,S2a,S2b:

32.70? (V, F or Q)q

3.730? (V, F or Q)v
Identifier for this variate? Cuplant
```

```
How many values are there for each variate and factor?
Type colon (:) if you do not know (the data must then be terminated by :)
Number of units? 40

  Identifier   Minimum    Mean  Maximum   Values   Missing

   Dryweigh     1.730   3.979    5.860       40         0
    Cuplant     2.913   7.439   11.707       40         0

Please choose which stage you want next:
  I  Input data              S  Summarize data in table or graphs
  C  Calculate new data      A  Analyze by standard statistical methods
  G  Form groups             Q  Quit using Genstat
Stage? (I, C, G, S, A or Q) c
```

Transformation

```
Do you want to apply a standard transformation to one or more variates
? (Yes or No) y

Which variates (separate identifiers with commas and end with :)
? (identifiers) Plantcu:
```

Dealing with errors

```
Variate Plantcu does not exist.  Type ?I for list of identifiers.
? (identifiers) ?I

  Identifier      Type   Length    Values   Missing   Ref.No.

    Dryweigh  Variate       40   present         0        -1
      Contam   Factor       40   present         0        -3
     Cuplant  Variate       40   present         0        -4

? (identifiers) Cuplant
? (identifiers) ?

Please give a list of identifiers of variates which you wish to
transform.  The variates should all have values already input
or calculated.  Separate the identifiers with commas and end with colon (:).
? (identifiers):
```

```
Please choose a transformation:
  A Add a constant        H Angular               P Power by constant
  B Absolute value        I Inverse               R Square root
  C Cosine (of radians)   L Log base 10           S Sine (of radians)
  E Exponential           M Multiply by constant  W Integer part
  G Logistic              N Log base e
Transformation? (A, B, C, E, G, H, I, L, M, N, P, R, S, or W) l

  Identifier   Minimum     Mean  Maximum  Values   Missing

  Cuplant       0.4643   0.8402   1.0684      40         0

Please choose which stage you want next:
  I  Input data                 S  Summarize data in tables or graphs
  C  Calculate new data         A  Analyze by standard statistical methods
  G  Form groups                Q  Quit using Genstat
Stage? (I, C, G, S, A or Q) a
```

Parallel regression analysis

```
Please choose which type of standard analysis you want:
  A  Analysis of balanced experiments     M  Other multivariate analysis
  R  Regression analysis                  T  Time-series analysis
  C  Cluster analysis
Type of analysis? (A, R, C, M or T) r

Which is to be your dependent variate? (identifier) Drywt

Please choose the distribution to be assumed for your dependent variable:
  L  Normal (linear regression)         B  Binomial
  N  Normal (non-linear regression)     G  Gamma
  P  Poisson                            I  Inverse Normal
Distribution? (L, N, P, B, G or I):

Please choose the type of effects you want in your model:
  M  Multiple regression, with effects of variates only
  P  Parallel regression, with effects of 1 variate and 1 factor only
  G  Grouped regression, with effects of variates and main effects of factors
  I  Grouped regression, including interactions between factors
Type of regression? (M, P, G or I) p
```

```
***** REGRESSION ANALYSIS *****

*** Regression coefficients ***
    Y-variate: Dryweigh

             Estimate        s.e.          t

Constant         4.91        1.53       3.21
Cuplant       -0.1724      0.0448      -3.85

***** REGRESSION ANALYSIS *****

*** Regression coefficients ***
    Y-variate: Dryweigh

             Estimate        s.e.          t

Cuplant       -0.1456      0.0391      -3.72
Contam S1b       4.62        1.94       2.38
Contam S1c       4.91        1.94       2.53
Contam S2a       4.13        1.94       2.13
Contam S2b       5.10        1.94       2.63

*** Summary of Analysis of Variance ***
    Y-variate: Dryweigh

Terms                  Residual          Change          Mean    Variance
                     DF        SS      DF       SS      Change       Ratio
Initial Model
Constant             39   1.264E3       *        *

Modifications to model
+Cuplant             38   8.017E2       1   4.623E2    4.623E2       35.24
+Contam
-Constant            35   4.818E2       3   3.199E2    1.066E2        8.13
+Contam.Cuplant      32   4.197E2       3   6.214E1    2.071E1        1.56
* Denominator or ratio is Res SS/Res DF from line above = 1.312E1

Do you want more information about the model, or will you modify it?
  C  Check the current model      M  Modify the current model
  I  Print more information       Q  Quit regression
Further action? (C, I, M or Q)c
```

Checking the regression model

```
You may check the fit of your model in the following ways:
  F  Graph residuals against fitted values     I  List influential observations
  N  Normal plot of residuals                  O  List possible outliers
Type of check? (F, N, I or O)i

  Unit   Observed   Fitted   Residual   Leverage

     5       2.56     2.49       0.07      0.471
    32       2.14     2.38       0.24      0.413
    40       1.73     1.75       0.02      0.691
  *Average leverage is:                     0.221

Do you want more information about the model, or will you modify it?
  C  Check the current model     M  Modify the current model
  I  Print more information      Q  Quit regression
Further action? (C, I, M or Q)q

Please choose which stage you want next:
  I  Input data              S  Summarize data in tables or graphs
  C  Calculate new data      A  Analyze by standard statistical methods
  G  Form groups             Q  Quit using Genstat
Stage? (I, C, G, S, A or Q)q

******** End of job.  Maximum of 1473 data units used.
```

Reference

Bryan–Jones, J (1982) A Conversational Approach to Using Genstat, Genstat Newsletter, **9**, 23–27

Essai de Modélisation des Relations Rendement–Peuplement Epis de Blés d'Hiver

O. Philipe
Biométrie
INRA
Station Montfavet
84140 Montfavet
France

La donnée du peuplement épi peut fournir des informations utiles à l'amélioration des prévisions de récoltes de blés d'hiver. La mesure objective d'un tel peuplement, interprété comme potentiel intermédiaire de rendement, est réalisable dès l'épiaison. Ce calendrier correspond à des prévisions précoces susceptibles d'importantes améliorations. Nous avons proposé un modèle agrotechnique décrivant le rendement à partir du nombre d'épis au m^2. Cette étude doit se poursuivre par l'introduction de paramètres climatiques.

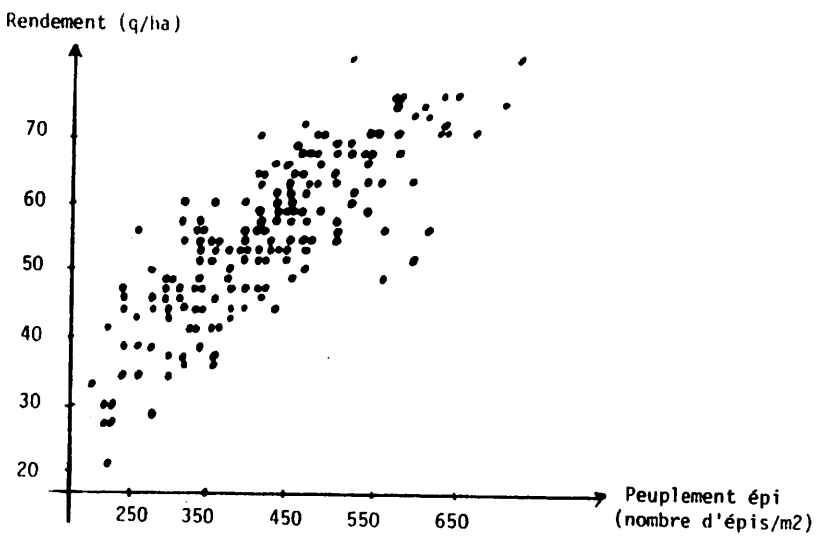
Les données de base sont issues d'enquêtes effectuées par le Service Central d'Enquêtes et d'Etudes Statistiques (SCEES) du Ministère de l'Agriculture, de 1971 à 1978, sur onze départements céréaliers, soit près de 4000 observations pour certaines variétés.

L'existence de deux prélèvements intra-champ permet d'évaluer une variance résiduelle correspondant à la variabilité de la résponse de la plante à un environnement agroclimatique. Cette étude des variabilités spatiales du rendement, à travers les facteurs de localisation (Départment, Région Agricole) et d'identification (champ, prélèvement) donne une estimation des écart-types intra-champ de l'ordre de 5 à 6 q/ha. Concernant notre objectif de modélisation, cette analyse permet de calculer la part maximale de la variance totale qu'il est possible d'expliquer par un modèle agroclimatique fondé sur des données détaillées au niveau des champ: de 92,4% en 1977 à 85,8% en 1978.

L'ensemble de ces travaux préliminaires, tant pour les analyses de variance, que pour la sélection et la combinaison de niveaux de facteurs correspond à des directiv es standart Genstat (figure 1).

Figure 1

Relation Rendement – Peuplement Epi. ANNEE 1972. Variété CAPITOLE.



L'étude de la réponse Rendement–Peuplement épi (figure 1) nous a amené à considérer des approximations paraboliques $y = aX + bX^2$ indexées par un facteur géographique départemental (11 niveaux), ou agrotechnique (association de techniques culturales les plus usitées, 15 modalités), pour traduire la diversité des situations culturales.

Le modèle général proposé est de la forme:

$$(*) \qquad Y_{ij} = a_i X_{ij} + b_i X_{ij^2} + e_{ij},$$

ou $i = 1,...,11$ décrit par exemple les niveaux du facteur départemental.

Les ajustements sont effectués variété par variété. Il est orthogonal pour la partition:

$$\begin{pmatrix} a_1 \\ b_1 \\ \cdot \\ \cdot \\ \cdot \\ a_{11} \\ b_{11} \end{pmatrix}$$

ce qui se vérifie par un calcul direct de la matrice de variance-covariance estimée.

Une première estimation des coefficients de régression donne les valeurs suivantes pour la variété Capitole, tableau 1:

Tableau 1

CAPITOLE DEP

	(1971)	(1972)	(1973)	(1974)	(1975)	(1976)	(1977)	(1978)
$\hat{a}_1$	1.3171	1.4978	1.2320	1.0401	0.3602	0.0006	0.7431	1.2720
.	1.3825	1.5000	1.5346	2.0926	1.3601	1.2241	1.0591	1.5190
.	1.5983	1.2005	1.3535	0.9760	3.0616	0.8423	1.4412	1.6287
.	1.7439	1.9214	1.6450	1.8005	1.4634	1.8854	1.6766	1.8932
.	1.3481	1.7048	1.1108	1.6770	0.9957	1.3854	1.2182	*
.	1.3570	1.5408	1.2769	1.0504	1.2447	0.8785	1.1787	1.5424
.	1.1036	*	*	1.1050	1.1911	1.0733	0.6868	1.3614
.	1.2617	1.4135	0.9064	*	1.2389	1.1277	0.7768	1.5016
.	1.2462	2.2484	1.7194	1.3556	1.1140	1.0518	1.0408	1.0531
.	0.9979	1.4455	1.1783	1.0411	1.0813	1.0315	1.0821	1.6701
$\hat{a}_{11}$	1.7503	2.0305	1.5555	1.4476	0.9444	1.0502	0.7549	1.4747
	(1971)	(1972)	(1973)	(1974)	(1975)	(1976)	(1977)	(1978)
$\hat{b}_1$	–0.0011	–0.0010	–0.0007	0.0007	0.0022	0.0020	0.0006	0.0004
.	–0.0010	–0.0006	–0.0009	–0.0018	–0.0008	–0.0003	–0.0000	–0.0007
.	–0.0014	–0.0002	–0.0006	0.0006	–0.0059	0.0005	–0.0009	–0.0012
.	–0.0014	–0.0017	–0.0010	–0.0011	–0.0010	–0.0022	–0.0013	–0.0014
.	–0.0005	–0.0009	–0.0001	–0.0013	–0.0003	–0.0009	–0.0002	*
.	–0.0007	–0.0007	–0.0004	0.0002	–0.0006	0.0001	–0.0003	–0.0007
.	–0.0001	*	*	0.0001	–0.0007	–0.0004	0.0005	–0.0003
.	–0.0006	–0.0003	0.0003	*	–0.0009	–0.0003	0.0006	–0.0004
.	–0.0010	–0.0025	–0.0015	–0.0004	–0.0005	–0.0000	–0.0004	0.0003
.	–0.0002	–0.0003	–0.0007	–0.0001	–0.0006	–0.0004	–0.0006	–0.0011
$\hat{b}_{11}$	–0.0020	–0.0018	–0.0011	–0.0010	–0.0002	–0.0005	–0.0001	–0.0007

Les variances-covariances estimées sont de l'ordre:

$$var\ \hat{a}_i \simeq 10^{-2},\ var\ \hat{b}_i \simeq 10^{-8}$$

et $cov(\hat{a}_i,\hat{a}_j) \simeq \pm 10^{-16}$ pour $i \neq j$,

$cov(\hat{b}_i,\hat{b}_j) \simeq \pm 10^{-21}$ pour $i \neq j$,

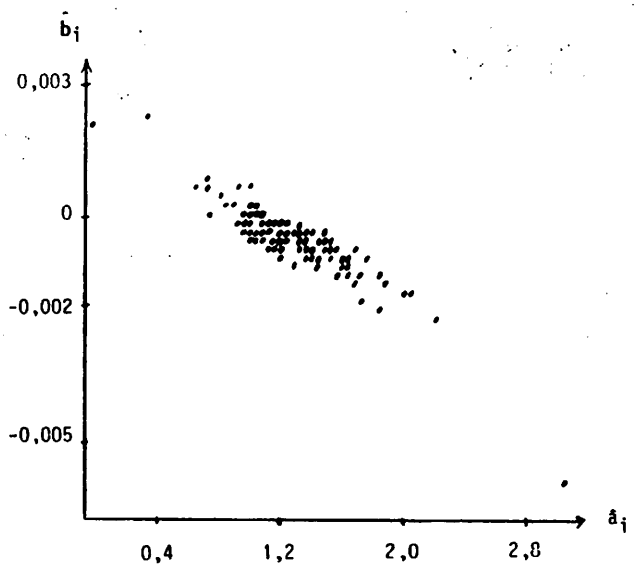
$$cov(\hat{x}_i,\hat{b}_j) \simeq \begin{cases} -10^{-5} & \text{si } i = j \\ \pm 10^{-19} & \text{si } i \neq j \end{cases}$$

Les couples $(\hat{a}_i,\hat{b}_i)$ étant non corrélés, la colinéarité des points $(\hat{a}_i,\hat{b}_i)$ représentés a la figure 2, est une caractéristique de notre modélisation du corps de données.

Figure 2

Représentation des couples $(\hat{a}_i,\hat{b}_i)$, $i=1,...,88$

Modèle départemental, variété Capitole



En première approximation, une régression linéaire des $\hat{b}_i$ en fonction des $\hat{a}_i$, donne une estimation des paramètres u et v tels que:

$$\hat{b}_i = u - v\hat{a}_i \quad : \quad \hat{u} = 2{,}22.10^{-3},\ \hat{e.t.} = 0{,}13.10^{-3}$$

$$\hat{v} = 2{,}13.10^{-3},\ \hat{e.t.} = 0{,}09.10^{03}$$

Cependant, en toute rigueur, il s'agit d'ajuster le modèle (*) muni de la contrainte $\hat{b}_i = u - v\hat{a}_i$, soit :

$$(**)\quad Y_{ij} = a_i X_{ij} + (u - va_i) X_{ij}^2 + e_{ij},$$

$i = 1,...,88$ pour les onze départements sur les huit années.

La directive 'OPTIMIZE' ne pouvant estimer autant de paramètres, le calcul des u, v et a_i a nécessité l'élaboration d'un programme spécifique.

Le critère d'ajustement retenu est la minimisation de la somme des carrés résiduels

$$SS_R(u,v) + \sum(Y_{ij} - a_i X_{ij} - (u - va_i) X_{ij}^2)^2,$$

et pour des valeurs données de u et v, a_i peut s'écrire :

$$a_i = \frac{\sum_j (Y_{ij} - u X_{ij}^2)(X_{ij} - v X_{ij}^2}{\sum_j (X_{ij} - v X_{ij}^2)^2}$$

Une procédure identique sert à estimer :

- pour un u donné, le v minimisant la SS_R ;
- le u optimum (et donc le v correspondant) pour ce même critère.

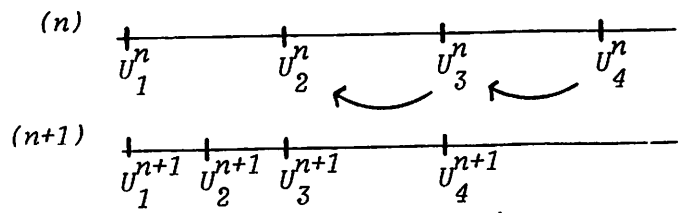
Partant d'un encadrement initial $(U_1^{\cdot}, U_4^{\cdot})$, on en déduit deux nouvelles valeurs $(U_2^{\cdot}, U_3^{\cdot})$ telles que :

$$U_2^{\cdot} = U_1^{\cdot} + R^2 (U_4^{\cdot} - U_1^{\cdot})$$

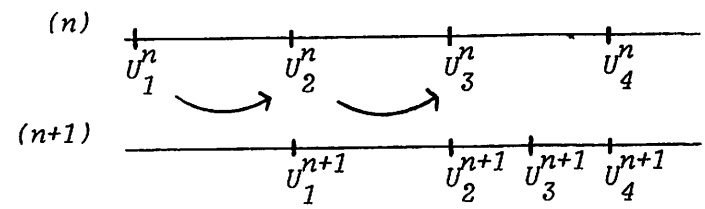
$$U_3^{\cdot} = U_1^{\cdot} + R (U_4^{\cdot} - U_1^{\cdot}) \quad \text{ou} \quad R = \frac{\sqrt{5} - 1}{2} \text{ (section d'or)}$$

Pour chacune d'elle, on calcule un v minimisant la SS_R (macro: `DOREV` et `CALCUL`), et suivant les SS_R calules, on affine l'encadrement à l'itération $(n+1)$ en proposant comme nouvelles bornes :

- si $SS_R(U_2^n) < SS_R(U_3^n)$, alors
$$\begin{cases} U_4^{n+1} = U_3^n, U_3^{n+1} = U_2^n \\ U_2^{n+1} = U_1^{n+1} + R^2 (U_4^{n+1} - U_1^{N+1}) \\ U_1^{n+1} = U_1^N \end{cases}$$



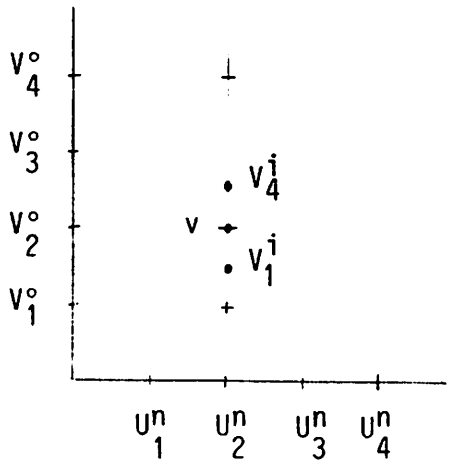
- si $SS_R(U_n^2) > SS_R(U_3^n)$, alors
$$\begin{cases} U_1^{n+1} = U_2^n, U_2^{n+1} = U_3^n \\ U_3^{n+1} = U_1^{n+1} + R(U_4^{n+1} - u_1^{n+1}) \\ U_4^{n+1} = U_4^n \end{cases}$$



Le critère d'arrêt est : a l'étape (i), $U^i - U^i > C_o$ fixé ; alors $u = \frac{U_4^i + U_1^i}{2}$

Pour une valeur U_k^n, k = 1,2 fixee, la même procédure d'itération est utilisée : partant d'un encadrement initial $(V_1^{\cdot}, V_4^{\cdot})$ on en déduit $(V_2^{\cdot}, V_3^{\cdot})$ (macro `DOREV`) et les deux SS_R

(macro CALCUL) correspondantes. Suivant leur valeur, on affine l'intervalle de variation à l'étape suivante pour aboutir à une estimation $v = \frac{V_1^i + V_4^i}{2}$, à l'étape *(i)* si $V_4^i - V_1^i > C_o$, étant donné U_k^n (macro DOREV).



On trouvera en annexe l'intégralité du programme et des deux macros.

Compte tenu des intervalles de départ suivant et du paramètre d'arrêt $c_0 = 10^{-4}$, on trouve pour:

$u \in (0{,}5.10^{-3} ; 2{,}5.10^{-3})$, $\quad u = 0{,}95.10^{-3}$
$v \in (10^{-3} ; 3.10^{-3})$, $\quad v = 1{,}15.1^{-3}$

La procédure converge en sept intérations sur u, chacune d'elles en occasionnant sept pour la recherche du v optimum correspondant. Les estimations des coefficients départementaux valent:

Tableau 2

	(1971)	(1972)	(1973)	(1974)	(1975)	(1976)	(1977)	(1978)
$\hat{a}_I$	0.979	1.329	1.051	1.517	1.112	0.707	1.165	1.308
.	1.133	1.696	1.445	1.844	1.141	1.325	1.321	1.603
.	1.193	1.378	1.352	1.597	0.894	1.196	1.281	1.386
.	1.510	1.634	1.575	1.815	1.354	1.379	1.309	1.758
.	1.432	1.830	1.378	1.360	0.898	1.162	1.453	*
.	1.297	1.651	1.417	1.539	1.147	1.012	1.345	1.519
.	1.249	*	*	1.386	0.874	0.982	0.974	1.605
.	1.182	1.696	1.198	*	0.981	1.151	1.140	1.721
.	0.916	1.716	1.351	1.460	1.034	1.227	0.917	1.409
.	1.019	1.664	0.927	1.148	0.859	0.940	0.864	1.600
$\hat{a}_{II}$	1.194	1.758	1.366	1.263	0.877	0.390	0.641	1.523

La famille de modèles paramétree en a:

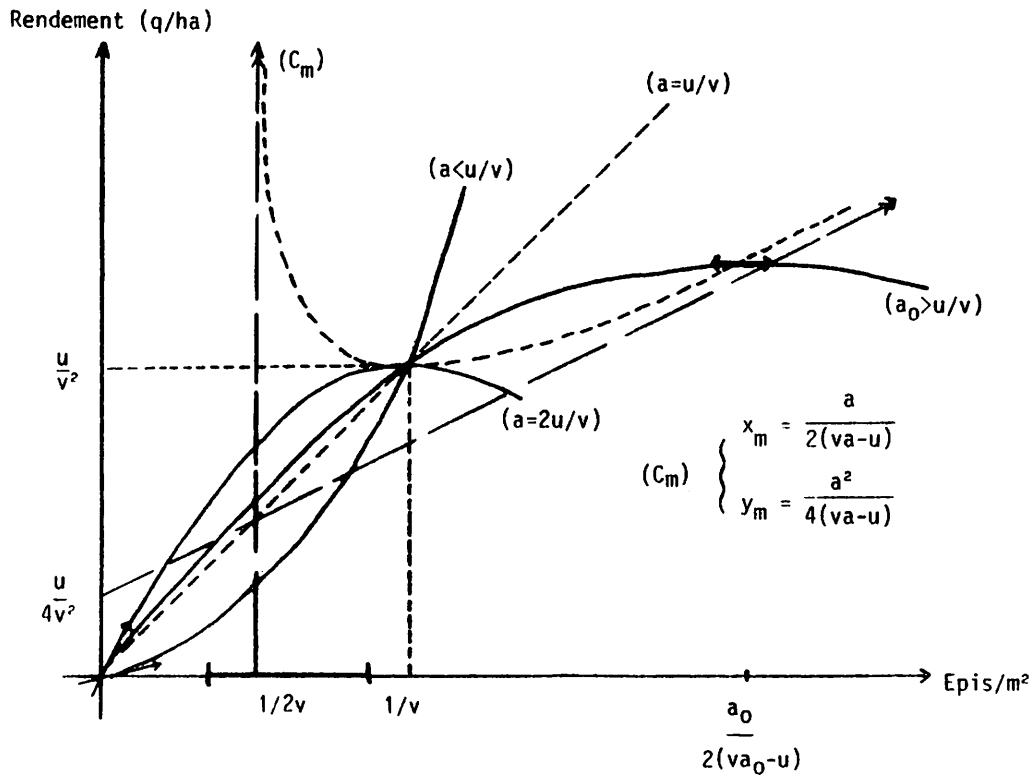
$Rendement = a\,Epi + (u\text{-}va)Epi^2$,

est un faisceau de paraboles passant par les points $(0,0)$ et $(\frac{1}{b}, \frac{a}{b^2})$, de sommet

$(Epi_{max}, R^{dt}_{max}) = (\frac{a}{2(va-u)}, \frac{a^2}{4(va-u)})$ situé sur l'hyperbole (Cm).

Figure 3

Modélisation Rendement/Peuplement Epi



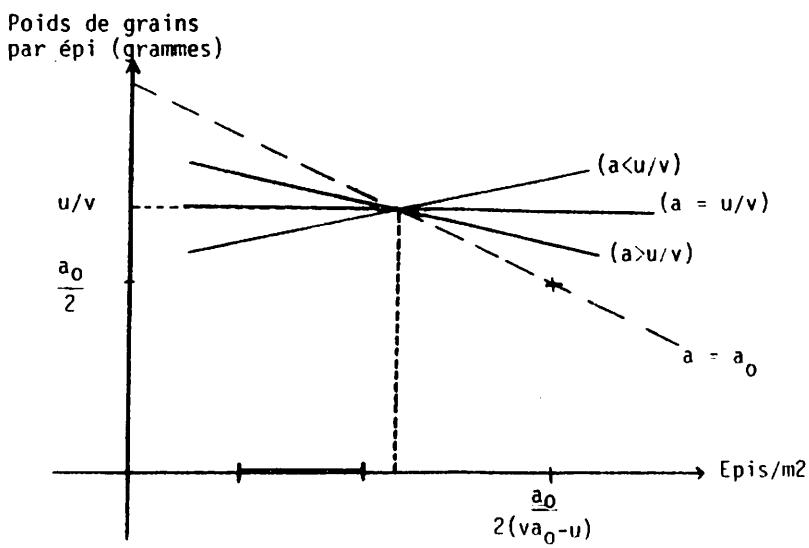
Si nous considérons le poids de grains par épi *(Pg/Epi)*, la relation s'écrit :

$$Pg/Epi = a+(u-va)\ Epi$$

qui décrit un faisceau de droites passant par le point $(\frac{1}{b}, \frac{a}{b})$

Figure 4

Modélisation Poids de grains par épi/Peuplement épi



L'intervalle de variation de notre peuplement épi se situe entre *300* et *600* épis/m^2. Les estimations faites sur la variété Capitole pour le modèle départemental donnent comme ordre de grandeur:

$1/v = 870$ epis/m^2, $u/v^2 = 72$ q/ha, $u/v = 0{,}82$ gr.

Mis à part quelques départements comptant peu de répétitions, on a généralement $a_i > u/v$.

Le poids de grains par épi décroît donc linéairement en fonction du nombre d'épis au m^2, on peut interpréter ce résultat par des phénomènes de compétition. Le rendement est maximum pour un peuplement épi de $\frac{a}{2(va-u)}$, le poids de grain par épi étant alors égal à $\frac{a}{2}$.

Pour un nombre d'épi/m^2 inférieur à Epi_{max}, il y a compensation au niveau du rendement: la baisse régulière du poids de grains par épi est alors compensée par l'augmentation du peuplement épi.

Les résultats obtenus sur une autre variété (Hardi) dont les caractéristiques génétiques (tallage, fertilité, épi, poids d'un grain) sont contrastées vis-à-vis de Capitole font apparaitre u et v comme des coefficients variétaux.

La modélisation des rendements d'une parcelle à partir de factuers agrotechniques (du champ) et du peuplement épi mesure est de la forme:

$$R^{dt} = f(Epi,a) + E$$

Elle permet d'expliquer, suivant les années, de 50% a 70% de la variabilité initiale, sans faire intervenir directement de paramètres climatiques. Sa partie déterministe s'écrit :

$$f(Epi,a) = aEpi + (u\text{-}v_a)\, Epi^2$$

où u et v sont des parametres varietaux.

Les valeurs des coefficients agrotechniques (a_i) doivent être discutées au vu des données climatiques, pour aboutir à l'attribution d'un a_i fixe (interannuel) à chaque modalité agrotechnique. De même, le terme d'erreur aléatoire E fera l'objet d'analyses incluant des variables climatiques post-épiaison.

La majeure partie de ces travaux s'est faite en utilisant des directives standard Genstat. Par contre, pour l'essai de modélisation qui vient d'être exposé, la directive `'OPTIMIZE'` s'est révélée inutilisable. Cette fâcheuse constatation m'a néanmoins permi d'apprécier les facilités de programmation offertes par Genstat.

Annexe

```
''PROGRAMME PRINCIPAL''
'CALC'ERREUR=.0001
'CALC'R=(SQRT(5)-1)/2
'SCAL'U1=.0005 :U4=.0025
'SCAL'ITERU=0
'CALC'U2=U1+R*R*(U4-U1)
'CALC'U3=U1+R*(U4-U1)
'CALC'U=U2 'USE'DOREV$
'CALC'SS2=QQ'RUN'
'CALC'U=U3'USE'DOREV$
'CALC'SS3=QQ'RUN'
'LABEL'LAU
'CALC'ERR=U4-U1
'JUMP'LAFU*(ERR.LT.ERREUR)
'CALC'ITERU=ITERU+1
'JUMP'LAU1*(SS2.LT.SS3)
'CALC'U1=U2 :U2=U3
'CALC'U3=U1+R*(U4-U1)
'CALC'SS2=SS3
'CALC'U=U3 'USE'DOREV$
'CALC'SS3=QQ
'JUMP'LAU
'LABEL'LAU1
'CALC'U4=U3 :U3=U2
'CALC'U2=U1+R*R*(U4-U1)
'CALC'SS3=SS2
'CALC'U=U2 'USE'DOREV$
'CALC'SS2=QQ
'JUMP'LAU
'LABEL'LAFU
'VARI'ALP(1971...1978)$11
'EQUA'ALP(1971...1978)=ALP
```

```
'MACRO' DOREV$
'CALC'V1=.001 :V4=.003
'SCAL'ITERV=0
'CALC'V2=V1+R*R*(V4-V1)
'CALC'V3=V1+R*(V4-V1)
'CALC'V=V2 'USE'CALCUL$
'CALC'SSQ2=QQ
'CALC'V=V3 'USE'CALCUL$
'CALC'SSQ3=QQ
'LABEL'LAB
'CALC'ERR=V4-V1
'JUMP'LABFIN*(ERR.LT.ERREUR)
'CALC'ITERV=ITERV+1
'JUMP'LAB1*(SSQ2.LT.SSQ3)
'CALC'V1=V2 :V2=V3
'CALC'V3=V1=R*(V4-V1)
'CALC'SSQ2=SSQ3
'CALC'V=V3 'USE'CALCUL$
'CALC'SSQ3=QQ
'JUMP'LAV
'LABEL'LAV1
'CALC'V4=V3 :V3=V2
'CALC'V2=V1+R*R*(V4-V1)
'CALC'SSQ3=SSQ2
'CALC'V=V2 'USE'CALCUL$
'CALC'SSQ2=QQ
'JUMP'LAV
'LABEL'LABFIN
'CALC'V=(V1+V4)/2
'USE'CALCUL$
'ENDM'
'MACRO' CALCUL$
'CALC/ZDZ=MV'ALP=XY-U*X3-V*X2Y+U*V*X*X4
:ALP=ALP/(X2-2*B*X3+V*V*X4)
'CALC'TT=Y2+2*(ALP*V-U)+X2Y-2*ALP*XY
:TT=TT+2*(U-ALP*V)*ALP*X3+ALP*ALP*X2
:TT=TTD+X4*(ALP*ALP*V*V-2*U*V*V-2*U*V*ALP+U*U)
'CALC'QQ=SUM(TT)
'ENDM'
```

How Steep is the Genstat Learning Curve?

Helen Talbot
Edinburgh Regional Computing Centre
University of Edinburgh
The King's Buildings
Mayfield Road
Edinburgh EH9 3JZ
United Kingdom

Edinburgh Regional Computing Centre was set up in 1966 to provide computing services to Edinburgh University and to Government Research Organisations in the Edinburgh area. In particular, ERCC was asked to develop a high quality multi-access service.

The scale of ERCC's activities has increased steadily to the present time. We provide mainframe facilities on an ICL 2988, a dual ICL 2972 and a VAX 11/750. We have almost 5,000 registered users in nearly 600 departments and organisations. The two ICL machines run under the locally developed Edinburgh Multi-Access System (EMAS). The systems together can handle up to 200 simultaneous users around the community through an extensive computing network. The VAX, which is also connected to the network, operates under VMS and supports about 12 users. In addition, around the community there is widespread use of mini and micro computers, many of which are also connected to the network so that they can access the mainframe facilities.

Edinburgh was one of the first sites outside Rothamsted to offer Genstat as a statistical tool. Over the years, the ARC Unit of Statistics has offered help to Genstat users working in agriculture. Many of these people have used the facilities at ERCC to run their jobs. More recently, the ERCC Advisory Service has become involved in supporting Genstat as more and more non-agricultural users have started to use it. As a pure computing organisation, we are not able to offer statistical advice but, instead, we run a joint scheme with the University's Department of Statistics to help all non-agricultural users from the University who seek joint computing and statistical help. Under this scheme, the Statistical Computing Clinic is run once a week and users are encouraged to bring all statistical computing problems. Those requiring statistical help are referred to the University's Department of Statistics. It is at this clinic that users are directed to a suitable package and, if necessary, arrangements are made to provide the necessary instruction.

The use made of Genstat today is considerable, with an average of 1800 jobs run per month, averaging 16 cpu seconds in length. It is the second most popular statistical package in use at Edinburgh. The majority of those using Genstat are either attached to research institutes or are writing postgraduate theses in the University. Many of these people use Genstat as a familiar tool, performing the same series of operations on varying sets of data. Such people, once they have mastered the techniques involved, rarely ask for help.

People who call on the Advisory Service are usually inexperienced with Genstat and perhaps half of them are also inexperienced with computers. In general, those who call on Advisory for Genstat help may be divided into three main categories:

1. The full time scientific research worker using statistical methods and packages as a tool to his research.
2. The science student or the academic who only does research during vacation.
3. The non-scientific user.

Individuals in the first category, having recognised the need to equip themselves with the ability to process their results by a good statistical package, find learning Genstat relatively easy. They are well motivated and are happy to read the literature before starting work and thereafter concentrate on particular parts of the language which they know they will require.

The science student or occasional research worker finds that with sufficient help directed at their particular problems they are able to use Genstat to produce the results they require. However, unless they have reason to make regular use of Genstat, their knowledge is soon lost and they have to start the learning process all over again. Such people who need the power of Genstat would benefit greatly from a user friendly interface to Genstat to help them with the syntax.

Users in the third category need a great deal of help, in understanding their problems, formulating them in terms of the Genstat language, running them on the computer and interpreting the results. Indeed, we cannot provide the level of support which these users require to run Genstat and, where possible, we encourage them to use another package or find themselves someone who is able to do the work for them.

Teaching Statistical Computing

With research facilities at a premium, it is essential to users that they learn the statistical tools as quickly as possible. Courses in SPSS are provided each term. We run occasional courses for all other packages depending on user requirements. Genstat is taught in small groups of not more than 10 people and generally only 3 or 4. Users with no experience of our system are given a specific introduction which teaches them how to use the local text editor to enter data and Genstat coding into files and then how to run a simple job and direct their results to an appropriate line printer. All other facilities are ignored at this early stage.

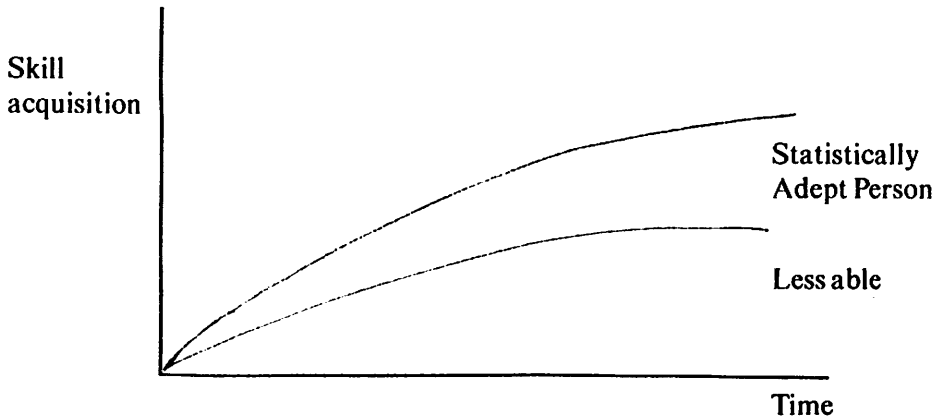
Having grasped the essential parts of the system, they may attend a half-day introductory course in Genstat. They are given a series of short talks and provided with the Introductory Guide to Genstat. This has been written specifically for our computing environment and concentrates on getting the user started by demonstrating simple plotting, regression and analysis of variance. At this early stage, it is important to avoid the learner's having to reference the full Genstat Manual, so some of the more common faults have been included in the Introductory Guide.

At the end of the course I set a group exercise in which the group tells me how to program a simple Genstat job. This has a dual purpose, in that it provides me with a feedback as to how much has been understood, as well as giving each member of the group the satisfaction of at least having taken part in the preparation of a piece of Genstat code.

This early stage is the most critical of all in the learning process. It is vital that advice given can be easily obtained and fully understood and the student sees himself as being able to write simple code. Failure to grasp the package at this stage is critical, with most people unwilling to give it a second try. For those users who require further help, we provide more advanced courses on specific topics within Genstat. These are spaced out in time and usually last for one half day.

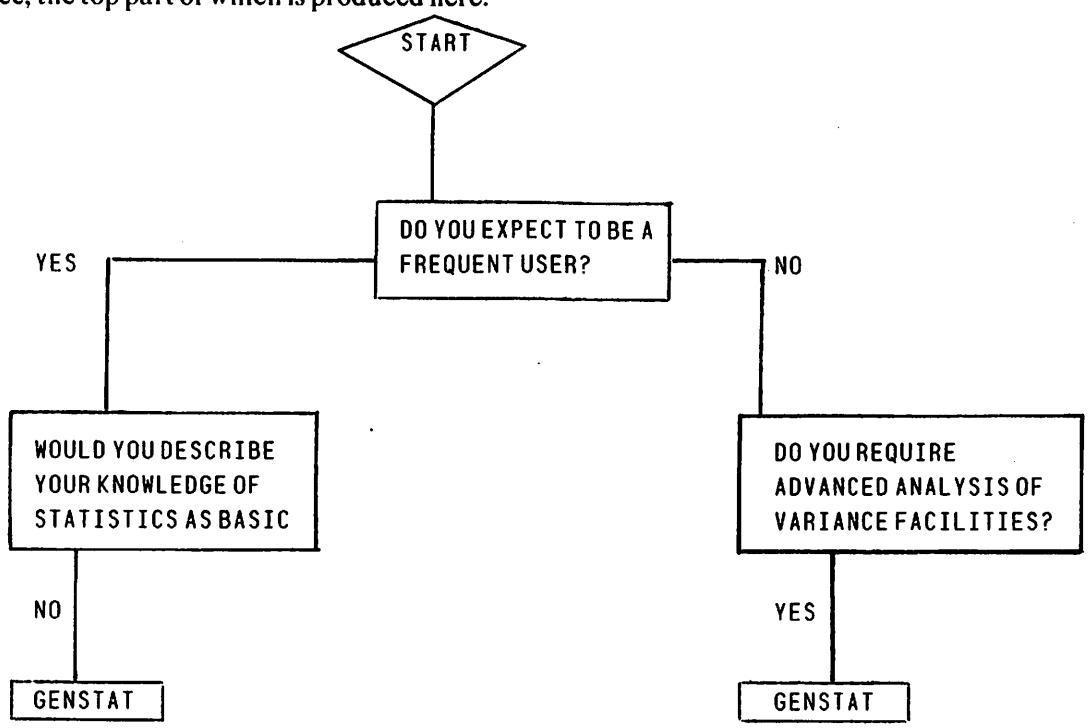
How Steep in the Genstat Learning Curve?

A learning curve may be defined as being generated by plotting facility in statistical computing over time. For Genstat, I suggest the curves for a number of user types may look something like this:



The steepness of this learning curve reflects our experience that a statistically adept person can make maximum use of Genstat in a short time whereas the less experienced are unable to benefit fully. With less effort, some may learn more from a less sophisticated package. Thus, in my view, for people who are able to grasp Genstat reliably, its flexibility and wide ranging facilities are very valuable, but, for people who are likely to find that it is beyond their capabilities, they may be wasting their time even attempting to learn it. After all, many of these are not concerned with the quality of the statistics but simply require a plausible answer which will plug the gap in a paper and convince a referee or examiner that the results are reliable.

In an attempt to help our users decide which statistical package to use, we have provided a decision tree, the top part of which is produced here:



This shows how much emphasis we place on Genstat and how much it is valued. Is it possible to help these users to learn Genstat in any further direction? We saw at the last Genstat conference how an attempt had been made by Jane Bryan–Jones to find the answer from within Genstat. Her approach was to use Genstat to set up a Genstat job. I would prefer to see another language, like Fortran, used to provide pre-processing facilities. A computer dialogue of the following form would be relatively straightforward to provide and could be used as a teaching system.

```
Please provide answers to all questions and terminate each by pressing RETURN.
Enter an overall heading for your Genstat job.
PRE PROCESSOR TEST
Your first statement is
'REFE' PRE PROCESSOR TEST
How many cases or units do you have in your data?
12
The next statement is
'UNIT' $ 12
Is the data stored by variate?
YES
What are the names of the variates? Please separate them by commas.
VARA, VARB, VARC.
You have 3-variables and you intend to enter the whole of VARA followed
by VARB and then VARC. Is this correct?
YES
Is your data in a file?
YES
Please enter the file name
TEST1
Your read statements are then:
'INPUT' 2
'READ/S' VARA,VARB,VARC
'INPUT' 1
If you wish to run your program please note your Genstat code is now
stored in T#CODE so you should enter the command
GENSTAT(T#CODE,TEST1).
```

The error messages generated by the system provide users who are learning it with many problems. What they need is a fuller explanation with more examples. At the moment they may be confronted with a statement:

```
*** line 12 STATEMENT 0 FAULT IO9
```

This raises two questions in their minds:

1. Which is line 12?
2. What does 'IO9' mean?

By using the pre-processor already described it would be easy to pick up line 12, print it out and then give the user the opportunity of asking for either a brief or a full explanation of the error message. The brief explanation would be taken directly from the manual but the full version would provide, in addition, access to a database of annotated examples which could be easily updated. The database would require a system whereby all calls made on it could be logged so that frequently used areas could be refined, to provide improved error reporting facilities.

Finally, I would like to consider the future of statistical computing in the University of Edinburgh. As time progresses, we are likely to move away from mainframe facilities towards the micro computer. It is vital that we provide good statistical packages on these machines. At the moment, we do not have any packages which give the sort of facilities which Genstat provides on the mainframe. Indeed, the ideal situation would be to mount a modular form of Genstat which is easier to use and comprises a subset of the more commonly used facilities.

Minitab has been mounted on a Superbrain operating under CP/M. I would be interested to know if any attempts have been made with Genstat.

Multiple copies, the Genstat Analysis of Variance algorithm and Neighbour Analyses

Robin Thompson
ARC Unit of Statistics
University of Edinburgh
James Clerk Maxwell Building
The King's Buildings
Mayfield Road
Edinburgh EH9 3JZ
United Kingdom

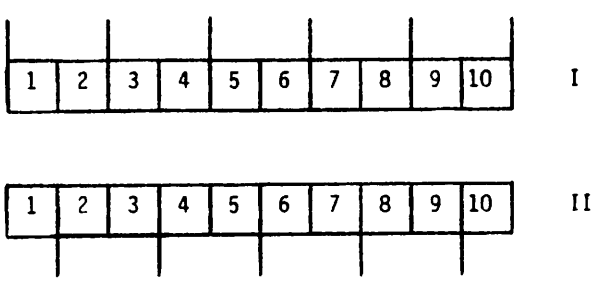
The Genstat analysis of variance algorithm is very flexible and can analyse a large number of designs (some would say all the designs in the book of Cochran and Cox). There are, however, some highly symmetric designs, in particular diallel crosses, partially balanced incomplete blocks with triangular association scheme and rectangular lattices, which Genstat cannot analyse directly using 'ANOVA'. I show that, by introducing multiple copies of the data and using appropriate block and treatment factors with each copy, these designs can be analysed using 'ANOVA'. This application is dealt with, at length, in Thompson (1983).

The idea of using information on residuals of neighbouring plots as a covariate to improve the analysis of field experiments was suggested by Papadakis (1937). Following the paper of Bartlett (1978) there has been much discussion of the method recently. Until the recent paper of Wilkinson et al.(1983) this neighbour method had been thought to be logically quite different from the classical analysis based on using blocks. Wilkinson et al. (1983) introduced the idea of using moving blocks. If the plots were arranged linearly they would use plots $i-1, i, i+1$ as a 'block' for plot i. This leads them to concentrate on terms like $y_i - \frac{1}{2}y_{i-1} - \frac{1}{2}y_{i+1}$, sometimes called second differences (SD). It is not obvious (to me, at least) how correlated the block information from adjacent blocks is assumed to be, as Wilkinson et al. use, implicitly or explicitly, at least five different variance models (it is perhaps appropriate that seven of the eleven initials of the authors' names can be rearranged to make the work MONSTER).

The idea of multiple copies gives a convenient conceptual framework for considering the neighbour analyses of Wilkinson et al. and the many discussants of their paper. There is only space to summarise the results. Some of the justification is given in Thompson (1984).

For convenience, consider an experiment laid out linearly and suppose two copies of the data are taken with different blocks of two, superimposed on the two copies as in Figure 1.

Figure 1



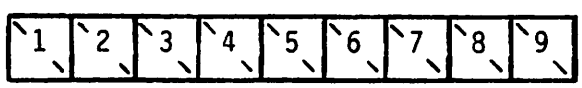
Then the Papadakis treatment estimation equations correspond to recovery of inter-block information from the two copies. There is a relationship between the ratio of between block variance to within block variance and the covariate regression coefficient used in the Papadakis analysis.

Further, Bartlett (1978) has shown that the Papadakis treatment estimation equations are efficient under a (first-order) autoregressive variance model, (AR). With this model, efficient estimation of the variance parameters follows from equating sums of squares of residuals within and between plots to their expectation.

In the symmetric designs of the first paragraph it was appropriate to introduce a blocking factor, say UNITS, to link the multiple copies of the plots together. If this extra stratum is imposed on the block structure of Figure 1, then recovery of information from the UNITS stratum and the blocks of size two in the two copies corresponds to efficient treatment estimation under a model where the residual variance is $(1 + \lambda)\sigma^2$ and the variance between plots i apart is $\rho^i \sigma^2$. This exponential variance (EV) model was found empirically by Patterson and Hunter (1983) to be appropriate for 166 cereal variety trials in Great Britain. Besag (1977) has suggested an error-in-variables motivation for such a model from consideration of uniformity data. (It is fortunate that EV is still an appropriate abbreviation.)

Several limiting cases of this analysis are of interest. The parameter λ measures how close the two copies of the data are. When $\lambda = 0$ then the EV model reduces to the AR model. Using no within-unit information is efficient when a moving average (MA) model is appropriate. Using unit and within-blocks-of-two information is efficient when a linear variance (LV) model is appropriate. This model was suggested by Williams and Patterson (1984) as an approximation to the EV model when ρ is large. The use of inter-block information from the blocks of two can scarcely ever be justified but it was essentially all that was available in a trial reported by Finney (1962). This trial was laid out in long narrow plots and a corner was lost, so that two triangular half plots were harvested together as in Figure 2.

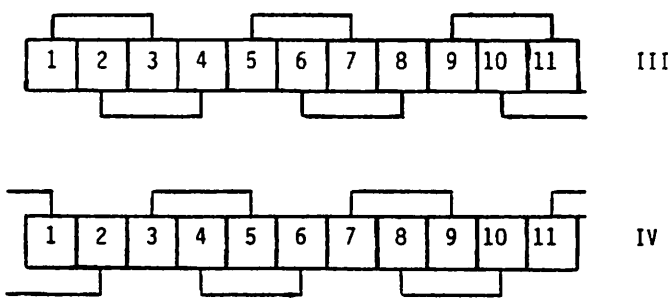
Figure 2



The expectations of block totals in Figure 1 have the same structure as the expectations of harvested plots in Figure 2.

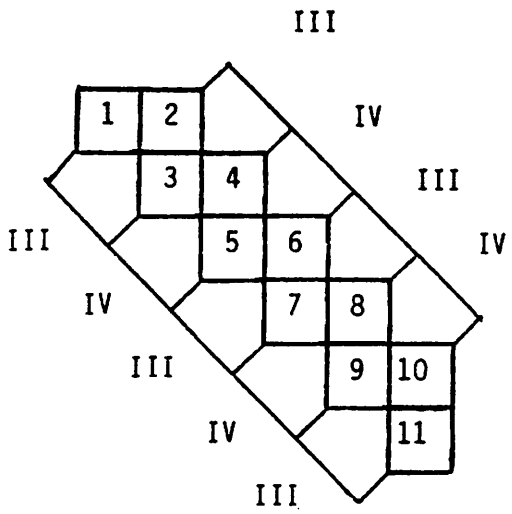
With a little further (perhaps misplaced) ingenuity, a second, different analysis gives that considered by Nelder and Green, Jennison and Seheult in the discussion following the Wilkinson et al. paper. For this we require the two copies of Figure 1 and also two more copies of the data with blocks of two representing second neighbours as in Figure 3.

Figure 3



Or alternatively we may think of the plots laid out as in Figure 4 with copy I (II) blocks representing rows (columns) and the blocks of III and IV linking plots of the two diagonals.

Figure 4



Nelder and Green, Jennison and Seheult essentially give negative weight (−1/3) to the information from copies III and IV, use information within blocks of two and also recover the information from the UNIT stratum. The negative weight can perhaps be interpreted as saying the EV analysis puts the second neighbour plots nearer than they should be.

This approach is suggested as a conceptual aid in linking neighbour methods with classical block methods. It can scarcely be computationally efficient, but it may give clues on constructing valid randomisation analyses and suggests that efficient designs with block size two might be merged to give efficient neighbour designs (at least for AR models).

References

Bartlett, M.S. (1978) Nearest neighbour models in the analysis of field experiments. JRSS(B), **40**, 147–174.

Besag, J. (1977) Error in variables estimation for Gaussean lattice schemes. JRSS(B), **39**, 73–78.

Cochran, W.G. and Cox, G.M. (1957) Experimental Designs. New York: John Wiley.

Finney, D.J. (1962) An Unusual Salvage Operation. Biometrics, **18** 247–250

Papadakis, J.S. (1937) Methode statistique pour des experiences sur champ. Bull. Inst. Amel. Plantes à salonique, **23**.

Patterson, H.D. and Hunter, E.A. (1983) The efficiency of incomplete block designs in national list and recommended list cereal variety trials. J. Agri. Sci., **101**, 427–433

Thompson, R.	(1983)	Diallel Cross, Incomplete Block designs and rectangular Lattices. Genstat Newsletter, **10**, 16–32.
Thompson, R.	(1984)	The Use of Multiplie Copies of Data in Forming and Interpreting Analysis of Variance. In 'Experimental Design, Statistical Methods and Genetic Statistics', (Hinkelmann, J. ed), Marcel Decker, New York.
Williams, E.R. and Patterson, H.D.	(1984)	A Neighbour Model for Field experiments (submitted to Biometrika).
Wilkinson, G.N., Eckert, S.R., Hancock, T.N. and Mayo, O.	(1983)	Nearest neighbour (NN) analysis of field experiments JRSS(B) **45**, 151–211.

NOTICES

NAG Users Association 1984 Meeting

NAG Users Association
c/o Miss Janet Bentley
Shore Lane Farm
Blackstone Edge Old Road
Littleborough
Lancashire OL15 OLQ
United Kingdom

The 1984 meeting of the NAG Users Association will be held at Unviersity College, London from Wednesday, 19th September to Friday 21st September. Accommodation and meals will be at Connaught Hall with lectures in the Chemistry Lecture Theatre. Booking forms for the meeting are available from NAG Central Office or from the address above.

International Time Series Meeting (ITSM) 1985

O.D. Anderson
9 Ingham Grove
Lenten Gardens
Nottingham NG7 2LQ
United Kingdom.

Prospective authors should write immediately to the above address for details on submitting abstracts and papers. Please enclose a self-addressed adhesive label.

New Address for Macro Library Editor

Jane Bryan–Jones
Department of Science
Cambridgeshire College of Arts and Technology
East Road
Cambridge CB1 1PT
United Kingdom

All correspondence to the Macro Library Editor should now be sent to the above address. Machine readable copy should be on unlabelled magnetic tape (ASCII, 1600 bpi, record length 80, any convenient blocksize)

A Course on 'Analysis of Counts Using Genstat'

Jane Bryan–Jones
Department of Science
Cambridgeshire College of Arts and Technology
East Road
Cambridge CB1 1PT

A course on 'Analysis of Counts Using Genstat' will be run at CCAT this summer: the dates are September 3 to 7.

For further particulars contact J. Bryan–Jones (Statistics) or B. Fingleton (Geography) at the above address.

GENSTAT NEWSLETTER ORDER FORM

To order future issues of the Genstat Newsletter, please complete the form below and return it to:

The Genstat Co-ordinator
NAG Central Office
Mayfield House
256 Banbury Road
OXFORD OX2 7DE
United Kingdom

(Each Genstat site representative *automatically* receives one copy of each issue, free of charge.)

Please note that each subscription to the Newsletter costs £ 5.00 per annum (2 issues). This price includes 2nd class/surface postage. Postage at other rates will be charged at cost.

Back issues of the Newsletter are available on microfiche (24×). The first contains issues 1 – 6 and each subsequent fiche contains two issues – 7/8, 9/10 etc. Each fiche costs the same as a year's subscription to the Newletter (£5.00).

To: NAG Ltd., Mayfield House, 256 Banbury Road, OXFORD OX2 7DE, U.K.

Please supply me with ..... copies of each future issue of the Genstat Newsletter

*for ..... years/until further notice beginning with issue number ..... .

(Minimum subscription period: 2 years)

Please supply me with ..... microfiche of each of the following issues ..... .

☐ Enclosed is my remittance for ..... .

☐ Please invoice me.

Signature _____

Name and address for posting _____

(please type or print) _____

_____

_____

_____

Special mailing instructions _____

Cheques to be made payable to the Numerical Algorithms Group Ltd.

** delete one alternative*

 # tick one box