Issue No. 16

1985 October

The GENSTAT Newsletter

NAG
NUMERICAL ALGORITHMS GROUP

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

Printed and produced by the Numerical Algorithms Group

ISSN 0269-0764

R w Payne

GENSTAT NEWSLETTER
Issue No. 16

NP1079

1985 October

Contents

Published Twice Yearly by
Rothamsted Experimental Station Statistics Department
and the Numerical Algorithms Group Limited

Printed 1986 February

Editorial

Many of the papers in this issue are based on talks presented at the Fourth Genstat Conference at York. We should like to offer our thanks to all the speakers, for the high standard of their presentations, to the local organisers, John Byrne and Tony Weekes, for the excellent arrangements and social programme, and to all who attended the conference, for helping to ensure its success.

There was much interest at the conference in the demonstrations of a prototype version of Genstat 5. Testing of a full version at Rothamsted and some other ARFC institutes is planned during 1986 and it is hoped that this can be released to implementors towards the end of the year, for conversion and subsequent release to user sites. Much work remains to be done, however, particularly in the development of completely new documentation.

Thanks are due to the many users (about a hundred and fifty to date) who have taken the trouble to complete and return the questionnaires distributed at the conference and with the last Genstat Newsletter. A full analysis of these will be a considerable task but it is already obvious that improved documentation has a very high priority for the majority of users. The many helpful comments made in this regard will be carefully considered in the development of the new documentation. It is hoped that at least a partial analysis of the questionnaires will appear in the next Newsletter.

Another demonstration which attracted much attention at the conference was that given by Dr J Coursol of an upgraded Genstat 4.03, known as 4.03E, for use on IBM personal computers and similar machines. A full description of this will appear in the next issue of the Newsletter. Appropriate documentation is in preparation and this version of Genstat will be released as soon as this is available.

A number of conversions of Genstat for workstations are also in progress. More details of these will appear in future issues.

A SIR/Genstat Interface

J Atkinson
Ministry of Agriculture and Fisheries
Private Bag
Wellington
New Zealand

Genstat is designed as a statistical data analysis package and has only limited database features. Because of the complexity or large volume of some sets of data it is often convenient to use a database package to organise, store and check the data and pass the relevant information to a statistical package when required.

At the New Zealand Ministry of Agriculture and Fisheries (MAF) we have available the Scientific Information Retrieval (SIR) database management system and we mainly use the Genstat and Minitab packages for our statistical analyses.

Don Wilson at MAF's Ruakura Agricultural Research Centre has written some SIR procedures which create Genstat input files from a SIR database. These files contain the data and Genstat commands including, if required, the level names for factors. (There is also an interface procedure available which creates a Minitab worksheet for SIR database information.)

The Genstat input file produced from the SIR procedure can then be combined with other Genstat commands for the analysis of the data.

Some advantages of using a SIR/Genstat interface program are:

(1) The user can concentrate on the analyses required and not worry about the mechanics involved in the transfer of data from SIR to Genstat.

(2) Variables keep the same names in both SIR and Genstat, so naming conventions are maintained throughout the project and advantage can be taken of the good documentation available for SIR.

(3) SIR retrievals are not cluttered up with programming associated with writing out the data, so it is easier to check if the correct data have been obtained or if the retrieval contains any other types of errors.

(4) Some Genstat declarations (including names, factors and factor level names) are written by the SIR procedure program, so there is less chance of making certain programming and typing mistakes in the Genstat program.

Some disadvantages of using a SIR/Genstat interface program are:

(1) Being a general program it is slower to run (in CPU time) than a program specially written for a particular application. However, less time need be spent in programming each retrieval and the user does not have to be an expert in programming in SIR.

(2) Occasionally the Genstat code produced by the interface is not exactly as required. However, this code can be used as a basis and only small areas of it may need changing.

The Interface

SIR procedures are available for creating a Genstat input file from a SIR database. To create the Genstat input file the procedures GENSTAT and either GENLAB or NOLAB should be called from the appropriate points in a SIR retrieval as detailed below. Before the procedures can be used in a database, they must first be read into the database using the SIR Editor command:

```
PREAD 'NN1.SOFT>SIR>GSTATS.PROC'
```

Procedure GENSTAT

Procedure GENSTAT creates a Genstat input file from a SIR database, carrying across data, variable names, factor names and labels. The file created is read as a secondary input file from a Genstat job.

Call

CALL GENSTAT(*filename*, *var list*)

where *filename* = Name of Genstat secondary input file to be created

var list = List of variables or names to be written to 12 parameters where each parameter may be a single variable name or a TO list (e.g. DT TO AHT – see SIR user manual section 2-9B). The total number of variables and factors retrieved should not exceed 99.

Example

```
CALL GENSTAT(NEG.DATA,FLIGHT TO NPROF,DATEMD,PROF TO AHT,YRBIRTH)
```

This would create the Genstat secondary input file NEG.DATA and would include the variables implied by the 4 parameters FLIGHT TO NPROF, DATEMD, PROF TO AHT and YRBIRTH.

On the Prime, the beginning of a Genstat job using the created data file and another data file may look like:

```
GENSTAT TTY NEG.OUT  -I2 NEG.DATA  -U1 NEGDUMP
'REFE' A_GENSTAT_JOB
'OUTPUT/RECL=132'
'INPUT' 2
'RUN'
'PUT/FILE=1, COMP=DUMP' AIRDUMP
'RUN'
....
```

Notes

(1) The call to GENSTAT must follow the main body of the user's retrieval program (i.e. only other retrieval procedures and the END RETRIEVAL statement may follow the GENSTAT call).

(2) A prior call must have been made to either:
   (i) the procedure GENLAB which sets up value labels for Genstat factors
   or
   (ii) the procedure NOLAB if no value labels are required.
   The calls to GENLAB or NOLAB should be inserted in the retrieval before any PROCESS CASES statements.

(3) All variables to be written to the Genstat file must be summary variables (they must, for instance, have been moved or computed, in the retrieval). Variables specified in the GENLAB call should not be included in the GENSTAT call but must be summary variables. (See note 7 in the next section.)

(4) Variable names beginning ZZ should not be used. Such names are used internally in these procedures.

(5) The SIR command AUTOSET must not be used in any retrieval which includes a call of GENLAB.

Procedure GENLAB

The procedure GENSTAT to create a Genstat input file from a SIR database is described above. If it is desired to pass labels of factors across into the Genstat file a prior call should be made to the procedure GENLAB. If such labels are not required the procedure NOLAB must be called.

Call

CALL GENLAB(*No vars*, *Var details*)

where *No vars* = Number (up to 12) variables (or factors) for which labels are required.

Var details = Details of variables for which labels are required. For each such variable three parameters are needed.

(i) the variables name

(ii) the type number of the record containing the variable (for common variables any valid type number can be given)

(iii) the number of values of the variable for which labels are required, i.e. the number of labels.

Example

```
CALL GENLAB(4,AIRPORT,1,2,DECTYPE,2,4,NORRAN,2,2,RVISIT,3,9)
```

This call would set up value labels for the 4 variables AIRPORT, DECTYPE, NORRAN and RVISIT, which are in different record types (1, 2 or 3) and have 2, 4, 2, 9 labels respectively.

Notes

(1) The call to GENLAB should be located in the retrieval prior to any PROCESS CASES statements.

(2) Up to 12 variables may be specified in GENLAB. The total number of labels for all variables specified should not exceed 99.

(3) GENLAB assumes that the variables called have integer values which run consecutively from 1 up to the specified maximum. If you want variables that do not comply with this restriction to be Genstat factors, you may either edit the resulting Genstat input file or pass such variables as data variables or names and convert them into factors in Genstat by using a 'GROUP' statement.

(4) If there is no SIR value label for a value in the range 1 to *max*, the Genstat label is set to the value itself. Thus unlabelled variables can be called in GENLAB and the result will be Genstat factors with labels equal to the values themselves. In Genstat terms the actual labels of the factor are the same as the formal levels.

(5) In the Genstat file produced, the name vector identifiers are generated by prefixing an N onto the corresponding variable name.

(6) If the labels used in SIR are not legal in Genstat, the resulting Genstat input file will have to be edited.

(7) GENLAB sets up the value labels but does not write the details into the Genstat input file. This is done by the procedure GENSTAT which the user calls later in the retrieval. The variables specified in GENLAB should not be included in the var list in the GENSTAT call. However, they should all be summary variables (i.e. moved in the retrieval) otherwise the values will not be included in the data.

(8) Do not use variable names which begin with ZZ. Such names are used internally within the procedure.

(9) The SIR command AUTOSET must not be used in any retrieval which includes a call of GENLAB.

Procedure NOLAB

The procedure GENSTAT creates a Genstat input file from a SIR database, carrying across data, variable names, factor names and labels. If labels are required for factors a call should also be made to the procedure GENLAB. If labels are not required a call must be made to NOLAB. This initialises variables otherwise set up in GENLAB.

Call

CALL NOLAB (i.e. no parameters are required)

Notes

(1) The call to NOLAB should be located in the retrieval before any PROCESS CASES statements. There should also be a GENSTAT call in the retrieval at the end of the user's retrieval (immediately before the END RETRIEVAL statement).

(2) If factor labels are required, use procedure GENLAB rather than NOLAB.

Examples of Use

Passing Variates and Factors across to Genstat

SIR Retrieval Program

```
RETRIEVAL
CALL GENLAB(8,AIRPORT,1,2,DECTYPE,2,4,NORRAN,2,2,SREQUEST,2,3,
            SCARRY,2,3,STYPE,2,3,SEX,3,2,RVISIT,3,9)
PROCESS CASES SAMPLE=0.10
.  MOVE VARS AIRPORT,FLIGHT,NPAX,POS,NEG,ORIGIN,NPROF
.  COMPUTE DATEMD=NUMBR(DATEC(DATE,'MMDD'))
.  PROCESS REC 2
COMMENT  SELECT ONLY THOSE PEOPLE WITH A NEGATIVE DECLARATION BUT
         WHERE SOMETHING WAS FOUND WHEN THEY WERE SEARCHED.
.  IF THEN ((DECTYPE EQ 1) AND ((IPCT+ISCT+ISHT) GT 0))
.    MOVE VARS DECTYPE,NORRAN,SREQUEST,SCARRY,STYPE,PROF, DT TO AHT
.    RECORD IS 3,(PROF)
.    MOVE VARS YRBIRTH,SEX,RVISIT
.    PERFORM PROCS
.    END RECORD IS
.  ENDIF
.  END PROCESS REC
END PROCESS CASES
CALL GENSTAT(NEG.DATA,FLIGHT TO NPROF,DATEMD,PROF TO AHT,YRBIRTH)
END RETRIEVAL
```

Resulting Genstat Input File

The resulting Genstat input file NEG.DATA is:

```
'NAME' FLIGHT,ORIGIN
'NAME' NAIRPORT =  Auckland, Wellington
'FACT' AIRPORT $NAIRPORT
'NAME' NDECTYPE =  Negative, Positive, Reverse, Incomplete
'FACT' DECTYPE $NDECTYPE
'NAME' NNORRAN =  Normal, Random
'FACT' NORRAN $NNORRAN
```

```
'NAME' NSREQUEST = Agric, Customs, Both
'FACT' SREQUEST $NSREQUEST
'NAME' NSCARRY = Agric, Customs, Neither
'FACT' SCARRY $NSCARRY
'NAME' NSTYPE = Full, Item, 3
'FACT' STYPE $NSTYPE
'NAME' NSEX = Male, Female
'FACT' SEX $NSEX
'NAME' NRVISIT = Returning_Resident, Immigrant, Visiting_Friends_&_,
                 Business_or_Work, Working_Holiday, Holiday,
                 Education, Stopover, Other
'FACT' RVISIT $NRVISIT
'READ/FLEV=F,NUN=V' AIRPORT,DECTYPE,NORRAN,SREQUEST,SCARRY,STYPE,
SEX,RVISIT,NPAX,POS,NEG,NPROF,DATEMD,PROF,DT,IPCT,ISCT,ISHT,ACT,AHT,
YRBIRTH,FLIGHT,ORIGIN
'RUN'

1 1 2 1 1 1 1 4 49 9 23 10 614 5 0 0 1 0 0 0 1949 C0001 HNL/LAX
1 1 1 1 1 2 1 1 216 24 71 116 212 8 0 0 0 1 0 0 1935 JL775 NAN/NRT
1 1 1 1 1 1 2 6 216 24 71 116 212 20 0 0 0 1 0 0 1961 JL775 NAN/NRT
1 1 1 1 1 1 2 6 216 24 71 116 212 45 0 0 1 0 0 0 1958 JL775 NAN/NRT
1 1 1 1 1 1 2 6 216 24 71 116 212 56 0 0 3 0 2 0 1962 JL775 NAN/NRT
1 1 1 1 1 1 2 6 216 24 71 116 212 58 0 0 0 1 0 0 1958 JL775 NAN/NRT
1 1 1 1 1 1 1 4 216 24 71 116 212 69 0 0 0 1 0 0 1957 JL775 NAN/NRT
1 1 1 1 1 1 2 6 216 24 71 116 212 88 0 0 0 1 0 0 1958 JL775 NAN/NRT
1 1 1 1 1 1 2 6 216 24 71 116 212 89 0 0 1 0 0 0 1959 JL775 NAN/NRT
1 1 1 1 1 1 2 6 216 24 71 116 212 92 0 0 1 0 0 0 1962 JL775 NAN/NRT
1 1 1 1 1 1 2 6 216 24 71 116 212 100 0 0 1 0 1 0 1957 JL775 NAN/NRT
1 1 1 1 1 1 1 6 216 24 71 116 212 109 0 0 0 1 0 0 1953 JL775 NAN/NRT
1 1 1 2 2 1 1 4 216 24 71 116 212 114 0 0 0 1 0 0 1949 JL775 NAN/NRT
1 1 1 2 2 1 1 1 216 24 71 116 212 115 0 0 1 0 0 0 1961 JL775 NAN/NRT
1 1 1 1 1 2 2 6 347 46 202 84 419 20 0 1 0 0 0 1 1920 QF043 SYD
1 1 1 1 1 2 2 6 347 46 202 84 419 22 0 0 0 1 0 1 1928 QF043 SYD
1 1 1 1 1 2 1 6 347 46 202 84 419 23 0 0 0 1 0 1 1928 QF043 SYD
2 1 1 1 1 1 1 3 192 23 129 28 611 8 0 0 0 2 0 2 1954 QF047 SYD
2 1 2 1 1 1 2 3 192 23 129 28 611 12 0 0 9 0 9 0 1925 QF047 SYD
2 1 1 1 1 1 1 4 192 23 129 28 611 23 0 0 0 6 0 0 1952 QF047 SYD
2 1 * 2 2 * 2 1 192 23 129 28 611 27 0 0 1 0 0 0 1956 QF047 SYD
2 1 * 2 2 * 1 6 192 23 129 28 611 28 0 0 0 1 0 1 1962 QF047 SYD
1 1 1 1 1 1 2 6 251 18 106 102 315 13 0 0 2 6 0 0 1923 SQ025 SIN
1 1 1 1 1 1 2 1 251 18 106 102 315 18 0 0 0 1 0 0 1932 SQ025 SIN
1 1 1 1 1 1 1 3 251 18 106 102 315 50 0 0 0 2 0 1 1962 SQ025 SIN
1 1 1 * 1 2 1 1 251 18 106 102 315 66 0 0 0 1 0 1 1928 SQ025 SIN
1 1 1 1 1 1 2 6 251 18 106 102 315 83 0 0 1 0 0 0 1916 SQ025 SIN
1 1 1 1 1 2 2 6 251 18 106 102 315 98 0 0 0 8 0 1 1929 SQ025 SIN
1 1 1 1 1 2 2 6 241 35 140 38 303 2 0 0 1 0 1 0 1938 TE003 NAN/LAX
1 1 1 1 1 1 1 4 241 35 140 38 303 27 0 0 0 1 0 0 1955 TE003 NAN/LAX
1 1 1 1 1 1 1 1 288 60 190 38 409 4 0 0 2 0 1 0 1939 TE006 SYD
1 1 1 2 1 1 1 6 288 60 190 38 409 8 0 0 0 1 0 0 1942 TE006 SYD
```

```
 1 1 1 1 1 * 2 6 288 60 190 38 409 9 0 0 0 1 0 0 1944 TE006 SYD
 1 1 1 1 1 1 2 6 288 60 190 38 409 17 0 0 0 1 0 0 1921 TE006 SYD
 1 1 * 2 2 * 1 1 412 58 300 42 526 17 0 0 0 4 0 0 1964 TE124 MEL
 1 1 2 1 1 1 1 1 412 58 300 42 526 21 0 0 0 1 0 1 1960 TE124 MEL
 1 1 2 1 1 1 1 4 412 58 300 42 526 27 0 0 0 1 0 1 1953 TE124 MEL
 1 1 1 1 1 2 2 1 412 58 300 42 526 40 0 0 1 0 0 0 1967 TE124 MEL
'EOD'
''No. records output = 38 ''
'UNIT' $38
'INPUT' 1
'RUN'
```

(Note: STYPE does not have a label in the database for the value 3. Thus the label is set to the value itself.)

Passing across Variates only to Genstat

SIR Retrieval Program

```
RETRIEVAL
CALL NOLAB
PROCESS CASES SAMPLE=0.10
.  MOVE VARS AIRPORT,FLIGHT,NPAX,POS,NEG,ORIGIN,NPROF
.  COMPUTE DATEMD=NUMBR(DATEC(DATE,'MMDD'))
.  PROCESS REC 2
COMMENT  SELECT RANDOM SEARCHES WHERE SOMETHING WAS FOUND.
.  IF THEN ((NORRAN EQ 2) AND ((IPCT+ISCT+ISHT) GT 0))
.    MOVE VARS DECTYPE,NORRAN,SREQUEST,SCARRY,STYPE,PROF,DT TO AHT
.    RECORD IS 3,(PROF)
.    MOVE VARS YRBIRTH,SEX,RVISIT
.    PERFORM PROCS
.    END RECORD IS
.  ENDIF
.  END PROCESS REC
END PROCESS CASES
CALL GENSTAT(RAND.DATA,FLIGHT TO NPROF,DATEMD,DECTYPE TO AHT,YRBIRTH TO
             RVISIT)
END RETRIEVAL
```

Resulting Genstat Input File

The resulting Genstat input file RAND.DATA is:

```
 'NAME' FLIGHT,ORIGIN
 'READ/FLEV=F,NUN=V' NPAX,POS,NEG,NPROF,DATEMD,DECTYPE,NORRAN,SREQUEST,
 SCARRY,STYPE,PROF,DT,IPCT,ISCT,ISHT,ACT,AHT,YRBIRTH,SEX,RVISIT,
 FLIGHT,ORIGIN
'RUN'
 49 9 23 10 614 1 2 1 1 1 5 0 0 1 0 0 0 1949 1 4 C0001 HNL/LAX
 49 9 23 10 614 2 2 1 1 1 8 1 0 1 0 1 0 1928 1 1 C0001 HNL/LAX
 192 23 129 28 611 4 2 1 1 1 11 1 0 1 0 1 0 1963 1 4 QF047 SYD
 192 23 129 28 611 1 2 1 1 1 12 0 0 9 0 9 0 1925 2 3 QF047 SYD
 251 18 106 102 315 2 2 1 1 1 10 1 0 0 1 0 0 1959 1 6 SQ025 SIN
 251 18 106 102 315 2 2 1 1 1 65 1 0 0 1 0 0 1926 1 1 SQ025 SIN
```

```
 412 58 300 42 526 2 2 1 1 1 9 1 0 0 1 0 0 1930 2 1 TE124 MEL
 412 58 300 42 526 2 2 1 1 1 10 1 0 0 1 0 0 1929 1 1 TE124 MEL
 412 58 300 42 526 1 2 1 1 1 21 0 0 0 1 0 1 1960 1 1 TE124 MEL
 412 58 300 42 526 1 2 1 1 1 27 0 0 0 1 0 1 1953 1 4 TE124 MEL
 250 29 132 17 518 2 2 1 1 1 16 1 0 0 1 0 0 1915 1 2 TE138 BNE
'EOD'
''No. records output = 11 ''
'UNIT' $11
'INPUT' 1
'RUN'
```

SIR Procedure Programs

GENSTAT

```
PROCEDURE      GENSTAT:T
PRINT BACK SAVE OFF
REPORT FILENAME='<1>'/
       PAGESIZE=9999999/
BREAK LEVEL 1
. INITIAL BLOCK
.   STRING*79 ZZINE
.   STRING*12 ZZFLD
.   STRING*20 ZZBL
.   STRING*8 ZZNAM
.   STRING*9 ZZFNM
.   INTEGER ZZLEV,ZZICT,ZZKK,ZZJJ,ZZVAL
.   STRING*8 ZZNT1 TO ZZNT30 ZZV1 TO ZZV99
.   INTEGER ZZNNV ZZNCT
.   IFTHEN (ZZNNN > 0)
.   FOR ZZKK=1,ZZNNN
.     COMPUTE ZZV1 TO ZZV99 (ZZKK) =ZZM1 TO ZZM12 (ZZKK)
.   END FOR
.   ENDIF
.   COMPUTE ZZNNV=ZZNNN; ZZNCT=0
.   DO REPEAT INDX=<2> <3> <4>
                   <5> <6> <7>
                   <8> <9> <10>
                   <11> <12> <13> /
.     IFTHEN (VARTYPE('INDX') EQ 1)
.           COMPUTE ZZNNV=ZZNNV+1;
                    ZZV1 TO ZZV99 (ZZNNV)=TRIM('INDX')
.     ELSE
.           COMPUTE ZZNCT=ZZNCT+1;
                    ZZNT1 TO ZZNT30 (ZZNCT)=TRIM('INDX')
.     ENDIF
.   END REPEAT
.   IFTHEN (ZZNCT > 0)
.     COMPUTE ZZINE="'NAME' "
.     FOR ZZKK=1,ZZNCT
.       COMPUTE ZZNAM=ZZNT1 TO ZZNT30 (ZZKK)
```

```
.       IF (LEN(ZZINE+ZZNAM) > 78) WRITE ZZINE; ZZINE=''
.       COMPUTE ZZINE=ZZINE+ZZNAM
.       IF (ZZKK < ZZNCT) ZZINE=ZZINE+','
.     END FOR
.     WRITE ZZINE
.   ENDIF
.   COMPUTE ZZICT=0
.   IFTHEN (ZZNNN > 0)
.     FOR ZZKK=1,ZZNNN
.       COMPUTE ZZLEV=ZZN1 TO ZZN12 (ZZKK);
                ZZNAM=ZZV1 TO ZZV99 (ZZKK);
                ZZFNM='N'+ZZNAM;
                ZZINE="'NAME' "+ZZFNM+" = "
.       FOR ZZJJ=1,ZZLEV
.         COMPUTE ZZICT=ZZICT+1;
                  ZZBL=' '+ZZB1 TO ZZB99 (ZZICT)
.         IF (LEN(ZZINE+ZZBL) > 78) WRITE ZZINE;
                                ZZINE='
.         COMPUTE ZZINE=ZZINE+ZZBL
.         IF(ZZJJ < ZZLEV) ZZINE=ZZINE+','
.       END FOR
.       WRITE ZZINE
.       WRITE "'FACT' " ZZNAM "$" ZZFNM
.     END FOR
.   ENDIF
.   COMPUTE ZZINE="'READ/FLEV=F,NUN=V' "
.   IFTHEN (ZZNNV > 0)
.     FOR ZZKK=1,ZZNNV
.       COMPUTE ZZNAM = ZZV1 TO ZZV99 (ZZKK)
.       IF (LEN(ZZINE+ZZNAM) > 78) WRITE ZZINE; ZZINE=''
.       COMPUTE ZZINE=ZZINE+ZZNAM
.       IF (ZZKK < ZZNNV OR ZZNCT > 0) ZZINE=ZZINE+','
.     END FOR
.   ENDIF
.   IFTHEN (ZZNCT > 0)
.     FOR ZZKK=1,ZZNCT
.       COMPUTE ZZNAM = ZZNT1 TO ZZNT30 (ZZKK)
.       IF (LEN(ZZINE+ZZNAM) > 78) WRITE ZZINE; ZZINE=''
.       COMPUTE ZZINE=ZZINE+ZZNAM
.       IF (ZZKK < ZZNCT) ZZINE=ZZINE+','
.     END FOR
.   ENDIF
.   WRITE ZZINE
.   WRITE "'RUN'"
.   SET ZZICT(0)
. DETAIL BLOCK
.   COMPUTE ZZINE=''
.   COMPUTE ZZICT=ZZICT+1
.   IFTHEN (ZZNNV > 0)
```

```
.   FOR ZZKK=1,ZZNNV
.     COMPUTE ZZNAM=ZZV1 TO ZZV99 (ZZKK);
              ZZVAL=NGET(ZZNAM)
.     IFTHEN (EXISTS(ZZVAL) EQ 1)
.       COMPUTE ZZFLD=' '+TRIM(FORMAT(ZZVAL))
.     ELSE
.       COMPUTE ZZFLD=' *'
.     ENDIF
.     IF (LEN(ZZINE+ZZFLD) > 78) WRITE ZZINE; ZZINE=''
.     COMPUTE ZZINE=ZZINE+ZZFLD
.    END FOR
.   ENDIF
.   IFTHEN (ZZNCT > 0)
.    FOR ZZKK=1,ZZNCT
.     COMPUTE ZZNAM=ZZNT1 TO ZZNT30 (ZZKK);
              ZZFLD=SGET(ZZNAM)
.     IFTHEN (EXISTS(ZZFLD) EQ 1)
.       COMPUTE ZZFLD=' '+FILL(TRIM(ZZFLD),'_')
.     ELSE
.       COMPUTE ZZFLD=' *'
.     ENDIF
.     IF (LEN(ZZINE+ZZFLD) > 78) WRITE ZZINE; ZZINE=''
.     COMPUTE ZZINE=ZZINE+ZZFLD
.    END FOR
.   ENDIF
.   WRITE ZZINE
. AT END BLOCK
.   WRITE "'EOD'"/
          "''No. records output = ",ZZICT,"''"/
          "'UNIT' $",ZZICT/
          "'INPUT' 1"/
          "'RUN'"
END BREAK LEVEL
PRINT BACK RESTORE
END PROCEDURE
```

GENLAB

```
PROCEDURE      GENLAB:T
PRINT BACK SAVE OFF
EXCLUDE ZZLEN ZZCT ZZII ZZAB
STRING*8 ZZM1 TO ZZM12
STRING*20 ZZAB ZZB1 TO ZZB99
INTEGER  ZZLEN ZZII ZZCT ZZBCT ZZN1 TO ZZN12 ZZNNN
SET ZZCT ZZBCT(0)
COMPUTE ZZNNN=<1>
IFTHEN (ZZNNN GT 0)
. PROCESS CASES COUNT=1
.   DO REPEAT ZZVAR=<2>  <5>  <8>
                   <11> <14> <17>
```

```
                    <20> <23> <26>
                    <29> <32> <35> /
          ZZREC=<3>  <6>  <9>
                    <12> <15> <18>
                    <21> <24> <27>
                    <30> <33> <36> /
          ZZLEV=<4>  <7> <10>
                    <13> <16> <19>
                    <22> <25> <28>
                    <31> <34> <37> /
          ZZCT=1 TO <1> /
.    PROCESS REC ZZREC
.     COMPUTE ZZM!ZZCT=TRIM('ZZVAR')
.     COMPUTE ZZN!ZZCT=ZZLEV
.     MOVE VARS ZZW!ZZCT=ZZVAR
.     FOR ZZII=1,ZZLEV
.       COMPUTE ZZBCT=ZZBCT+1
.       IF(ZZBCT GT 99) WRITE 'Total number labels exceeds max '
                                '- Retrieval abandoned';
                          EXIT RETRIEVAL
.       COMPUTE ZZW!ZZCT=ZZII
.       COMPUTE ZZAB=FILL(TRIM(VALLAB(ZZW!ZZCT)),'_')
.       COMPUTE ZZLEN=LEN(ZZAB)
.       IF(ZZLEN LE 0) ZZAB=TRIM(FORMAT(ZZII))
.       COMPUTE ZZB1 TO ZZB99 (ZZBCT)=ZZAB
.     END FOR
.     EXIT RECORD
.    END PROCESS REC
.   END REPEAT
. END PROCESS CASES
ENDIF
PRINT BACK RESTORE
END PROCEDURE
```

NOLAB

```
PROCEDURE      NOLAB:T
PRINT BACK SAVE OFF
STRING*8 ZZM1 TO ZZM12
STRING*20  ZZB1 TO ZZB99
INTEGER     ZZBCT ZZN1 TO ZZN12 ZZNNN
SET  ZZBCT ZZNNN (0)
PRINT BACK RESTORE
END PROCEDURE
```

Reference

[1] Robinson, B W *et al*
SIR User's Manual, Version 2, 1980.

Fourth Genstat Conference

The papers which follow were first presented at the Fourth Genstat Conference, although some have since been slightly revised.

The present selection is comprised of those papers which were received in time for this issue. It is hoped that the remaining papers will appear in the next issue, which will follow shortly.

Genstat in New Zealand

C J Thompson
Applied Mathematics Division
Department of Scientific and Industrial Research
Wellington
New Zealand

Introduction

We are now into our ninth year of Genstat in New Zealand. Genstat was originally purchased for use by two government departments: the Department of Agriculture (now Ministry of Agriculture and Fisheries, MAF) and the Department of Scientific and Industrial Research (DSIR). Today, the majority of users are in these two departments but Genstat has also been used by research organisations, universities, other government departments and private industry.

There is little doubt that, when it first arrived in New Zealand, Genstat represented a revolution in statistical computing. At the time, statistical computing was done almost exclusively by statisticians. They welcomed Genstat with open arms, delighting in the ease with which the whole range from simple to very complex analyses could be performed.

So much for history. How is Genstat seen in New Zealand today? To try to find out, I conducted a survey of Genstat users in New Zealand.

Survey of Genstat Users in New Zealand

In fact what I attempted to do was to conduct a census of Genstat users in New Zealand. I shall never know how close I came to achieving this, but I think I have come close enough for the results to reasonably represent the views of New Zealand users and the facts of their Genstat usage.

In all I received responses from 156 sometime users of Genstat. These people come from the following employment areas:

Ministry of Agriculture and Fisheries	40%	(63)
Department of Scientific & Industrial Research	30%	(47)
Universities	15%	(24)
Research Institutes	9%	(14)
Other Government Departments	3%	(4)
Private Industry	3%	(4)

The job descriptions of the people in the survey are:

Statistical Consultants	38%	(59)
Biologists	22%	(35)
Technicians	22%	(34)
Other Scientists	9%	(14)
Lecturers	5%	(8)
Administrators & Directors	4%	(6)

The No-longer-use-Genstat Group

Twenty of the 156 survey participants had used Genstat in the past but were not using it currently. I asked them to give the reason why they were no longer using Genstat. By far the major reasons were either that their job no longer needed such analyses or that they did not have Genstat available, rather than that they had 'defected' to SAS or some other package. (Although four out of the twenty had done so.) Seven of the twenty do indeed use SAS, ten use Minitab but no-one in this group uses P-Stat, GLIM or IMSL.

Usage of computing languages amongst this group: 30% use Fortran, 20% use Pascal, 45% use Basic and 15% use other languages.

In all, the people who have given up Genstat appear to be less involved in all computing than those who are still using it. When they had used it, 35% of them had limited their usage to modifying other people's programs and only one had used the interactive facilities.

Their main areas of usage within Genstat were I/O facilities, operations on variates and factors, ANOVA and linear regression.

Current Genstat Users

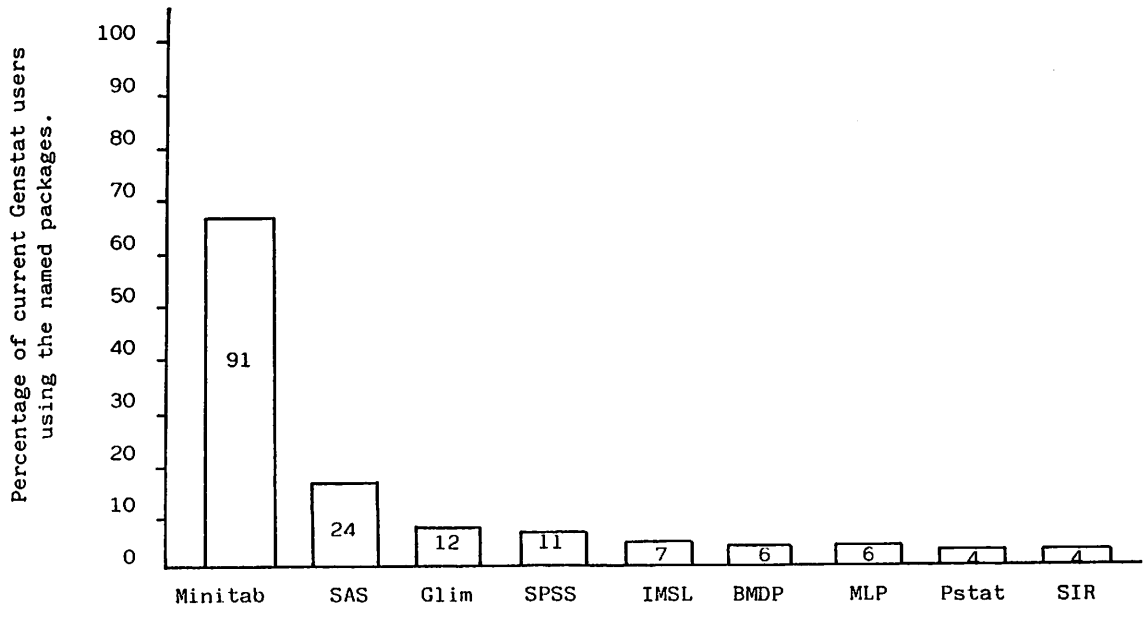
The 136 current users came from approximately the same employment areas as the total group except that no other government departments (outside of MAF and DSIR) have current Genstat users. They also had approximately the same distribution of job descriptions as the total group.

Computing done by Genstat Users

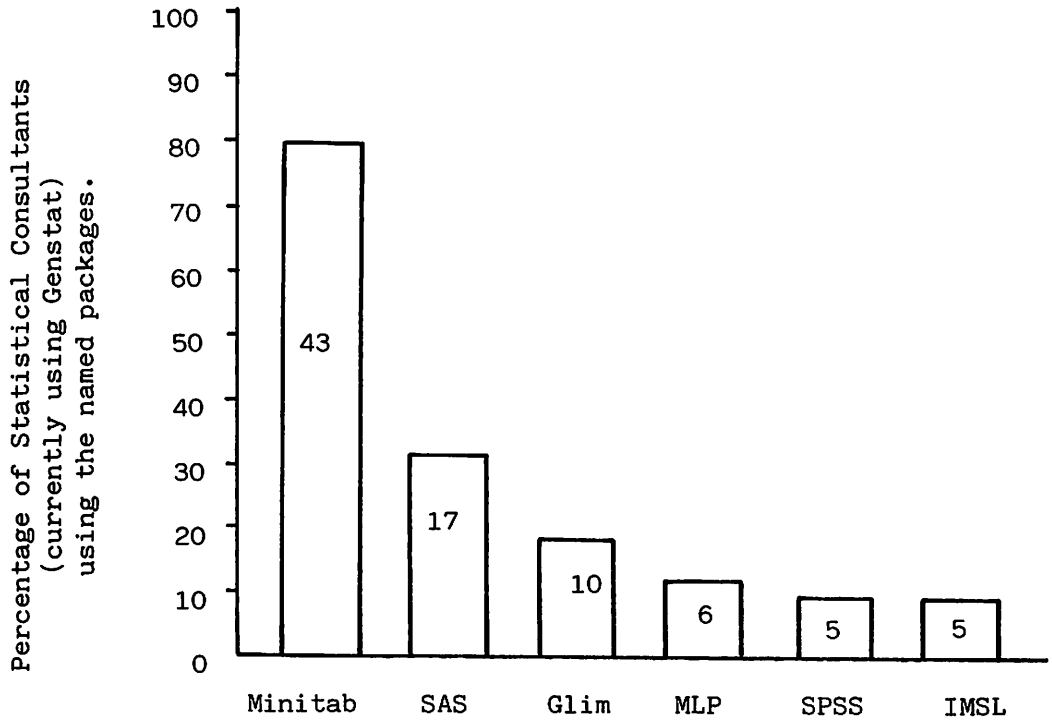
Usage of other packages amongst this group is common; it is mainly use of Minitab.

Note: (1) In these 3 bar charts, one user may fall into more than one category.

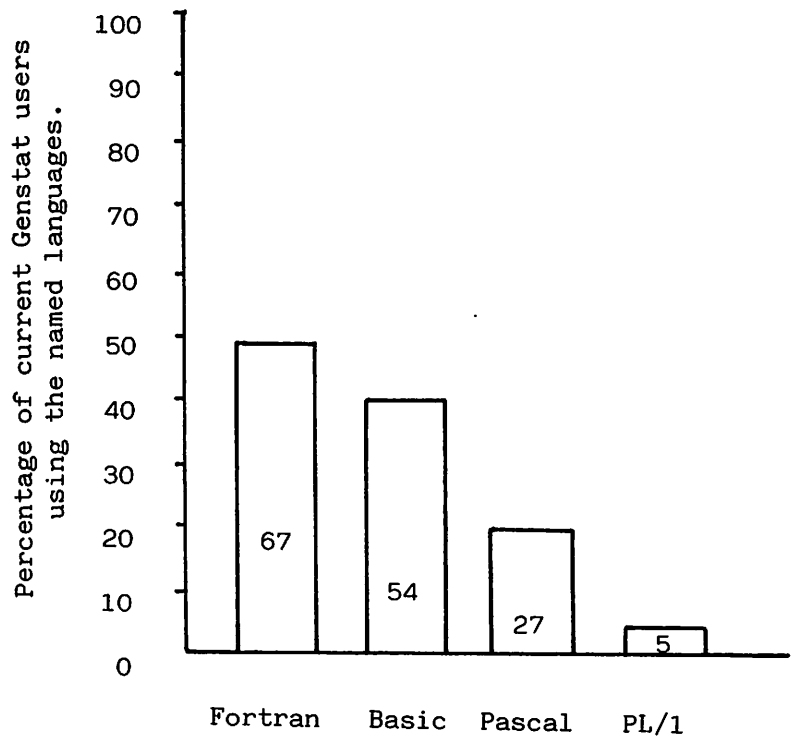
(2) The number in each block is the number of users of the named package or language.



Within this group of 136 Genstat users, 54 are statistical consultants. Their usage of other packages is slightly different, a higher percentage using Minitab, SAS, GLIM and MLP.



The people who currently use Genstat also currently use the following computing languages:



Amongst statistical consultants, the language usage is much the same except that there are more Fortran users (not surprisingly).

The two main computer types on which Genstat is installed in New Zealand are Vaxes and Primes.

The following table gives the numbers of current users of Genstat on various types of computers:

Vax 11/780	73
Prime (various models)	54
Burroughs (various models)	14
IBM 4341	14
ICL 2980	8
Vax 11/750	8

Genstat installations in New Zealand and numbers of users
(Note: some users have access to more than one machine.)
Table 1

This then gives a background of the computing done by Genstat users.

Time spent on Computing

Time spent on average per month on computing (including the tasks of writing, debugging and interpreting output) for current Genstat users has the following summary characteristics:

LO = 2 hours, LQ = 20 hours, Median = 45 hours, UQ = 80 hours,
HI = 200 hours (!)

In broad terms then, Genstat users spend on average approximately one third of their work time computing. One third of this time is spent using Genstat.

Computing time is greater amongst statistical consultants.

LO = 13 hours, LQ = 40 hours, Median = 60 hours, UQ = 80 hours,
HI = 200 hours (!)

Half of this time is spent using Genstat.

Within the major users (MAF and the Applied Mathematics Division of DSIR (AMD)), AMD users are the biggest users with an average of 38 hours per month or broadly one quarter of their total work time spent using Genstat. MAF users spend on average 26 hours per month. The other heavy user group is the research organisations, whose employees spend on average 20 hours per month using Genstat.

Genstat is thus a major tool in statistical analysis in New Zealand.

Areas of Application

In the last two years, agriculture, horticulture and biology were the main fields where Genstat was used. 68% of current users apply Genstat to problems in agriculture, 40% to problems in horticulture and 35% to problems in biology. A few users apply Genstat to problems in medicine and the physical sciences. Other areas of application include fisheries research, forestry, soil science, oceanography, social sciences and quality assurance in various industries.

Twenty-three percent of the current users limit their usage to minor modifications of other people's programs. This group comprises mostly biologists and technicians.

Twenty-six percent of the current users use the interactive mode, the greatest use being made by MAF employees (more than half of the interactive mode users work for MAF).

General areas of Genstat being used

I asked a series of questions concerning usage of the general areas within Genstat. To describe the areas I used essentially the chapter headings from the manual. I asked for areas that had been used in the last two years and main areas of expertise within Genstat.

Tables 2 and 3 below give the results with areas ranked into descending order and grouped into similar levels of usage.

General areas of Genstat in order of use during the last two years.

(Current users only are considered here.)

Area within Genstat	Approx % of current users using the named areas in the last two years
Input and output	100
Analysis of designed experiments Operations on variates and factors Linear regression Output of graphs, histograms, etc	75-85
Operations on tables Generalized linear models Operations on matrices Using macros	40-55
Multivariate analysis Optimization	30-35
Structure storage and retrieval Cluster analysis Writing macros	20-25
Time series analysis	10

Usage within Genstat

Table 2

Perceived main areas of expertise within Genstat in descending order.

(Current users only are considered here.)

Area within Genstat	Approx % of current users who feel these are their main areas of expertise
Analysis of designed experiments	70
Linear regression Input and output Operations on variates and factors	40-50
Output of graphs, histograms, etc Generalized linear models Operations on tables	20-30
Optimization Operations on matrices Multivariate analysis Using macros	7-10
Writing macros Structure storage and retrieval Cluster analysis	3-5
Time series analysis	1.5 (only 2 people!)

Usage within Genstat
Table 3

Observations:

(1) Whilst everybody obviously uses the Input and Output section, fewer than half the users consider they are expert in its use.

(2) Confidence in ANOVA is not far behind its use.

(3) Although many users use output of graphs and histograms, fewer than half of these feel expert in doing so.

(4) There is very little confidence in the areas Optimization, Operations on Matrices, Multivariate analysis, Using Macros, Writing Macros, Structure storage and retrieval, Cluster analysis and Time series.

An explanation for the lack of confidence in the areas noted in (4) above could be lack of practice. Although around one third of the current users have tried using these areas, it could be that their usage has been small in actual numbers of problems. ANOVA, on the other hand, is clearly Genstat's big success story and probably what most people first get into Genstat for. Problems with input/output however cannot be so simply explained away, as users certainly get plenty of practice.

Areas for Development in the future

In this section of the survey I asked for opinions in three areas:

(1) Existing things in Genstat seen as possibly needing change.

(2) New areas of statistical computing not at present covered by Genstat.

(3) New technological developments and facilities users would like to see being used with Genstat.

Needing change?

I asked for opinions on documentation, I/O, data handling, syntax and information on Macros.

Below is a summary of the responses to each question.

The results are for current users only.

The class called 'no response' comprises those respondents who either had no opinion or did not feel their familiarity was sufficient to justify an opinion.

(i) Do you find the documentation
good (21)
****
fair (69)
****
poor (41)
****
no response: 5

(ii) Do you find Input/Output facilities
good (67)
****
fair (55)
****
poor (6)
******
no response: 8

(iii) Do you find the data handling facilities
good (70)
****
fair (51)
****
poor (4)
****
no response: 11

(iv) Do you find the syntax for directives and options
good (41)
****
fair (60)
****
poor (22)
****
no response: 13

(v) Do you find the information given on library Macros
good (11)
****
fair (40)
****
poor (15)
****
no response: 70

Documentation

Most respondents had an opinion about the documentation and fewer than one sixth thought it was good.

Comments were also sought from users.

One comment which perhaps sums up the vast majority of the others was 'I suppose I have used Genstat for so long I know where to go now, but for the beginner ...'.

The two-part system for the manual was frequently criticised.

A suggestion was made that the 'example programs' currently available on file should be printed in the manual.

There was a widespread call for an introductory manual to get beginners started painlessly.

One notable New Zealand statistician and very big Genstat user commented: 'The documentation is the biggest hurdle to making Genstat generally and widely accessible. Not only is it not clear to novices, but gems, tricks and bugs are hidden even from the eyes of regular users. It is worth stressing problems in documentation'.

Input/Output

There is general satisfaction with the I/O facilities although there were still quite a number of critical comments, amongst them dislike of 'EOD', dislike of the non-regular E format (making reading output files into other programs impossible without editing), frustration with the non-acceptance of tabs, and lack of consistency between equivalent input and output commands (e.g. `'input/recl = 132' n` and `'output' n $132`).

Data Handling

There is also general satisfaction with the data handling facilities. Problems which do arise seem to be more concerned with documentation than with the facilities themselves. Quite a few users would like easier access to elements of structures. Many users would like to see subsetting of the data made simpler and better provision for the handling of variable-length records.

Syntax

The syntax comments can probably best be summarised by 'Generally OK when you're used to it'.

Information on Macros

Fewer than half the current users have ever tried to use Macros, but those who do seem to have encountered some difficulties. There is a call for more comments to be included in the macros, so users can see easily what is going on. A frequent comment was that the information given was too sketchy.

This summarises opinions on these 5 aspects of Genstat. Finally I asked for 'other comments (both positive and negative)'. Many of these were encouraging but with some reservations – for example 'I appreciate Genstat's great flexibility, especially its elegant handling of designed experiments. Genstat certainly isn't the easiest package to use for many standard analyses, but for some reason I frequently encounter non-standard problems. Error messages aren't too helpful for the unsuspecting. The manuals have all the answers, but it's not always easy to find them. In short, Genstat is wonderful but quirky. Long may it live and soon may its documentation improve so that mere biologists like myself can more easily penetrate its crust'.

And finally another comment from probably New Zealand's most experienced Genstat user 'Genstat was a marvellous innovation when it was written – it still is miles ahead of most (all?) other statistical packages. However, it also shows its age, particularly in graphics, EDA operations, language structure, macro structure and ease of input'.

To focus the users' attention on specific things they didn't like in Genstat I asked the question 'What are your 3 biggest gripes (if you have any!)'.

Summary of Gripes

Of all the respondents,

one third listed no gripes,
one half produced a 'second gripe'
and only one quarter had a full set of 3 gripes.

Gripe No. 1:

35% of all first gripes were to do with documentation
20% of all first gripes were to do with error messages
15% of all first gripes were to do with unfriendly syntax

Some other gripes:

no 'if then else'
no DBMS interface
$a1 = 0$ as constraint in regression
insufficient worked examples
poor graphics

and two complimentary 'gripes':

slow conversion to other machines
not having Genstat

from frustrated would-be Genstat users.

Gripe No. 2:

More of the same: often when documentation was given as the first gripe, error messages were given as the second and vice versa.

Overall:

one half of the people declaring gripes didn't like the manual
and one quarter to one third of the people declaring gripes listed interpretation of error messages as a problem.
The third runner was syntax.

The message appears to be pretty clear. The documentation of Genstat, while frequently considered better than much other computing documentation, is considered to fall somewhat short of ideal and is an area where some revision would be welcome. The problems with interpretation of error messages really also come under the broad umbrella of documentation. The proposed syntax changes in Version 5 should help those having problems with syntax.

New Areas

The most commonly requested new areas of statistical computing not at present covered by Genstat, in order of preference, are:

Exploratory Data Analysis (EDA)
Variance Component Analysis
Easy access to Regression Diagnostics
Non-parameteric tests and Bootstrapping
Multiple comparison tests after ANOVA.

New Facilities

The most commonly requested new facilities, in order of preference, are:

Better graphics (of the 45 respondents who expressed an opinion on new facilities they would like to see, 31 wanted better graphics facilities)
Improved interactive use
High-resolution dot-matrix printers
Windows
Mouse
Rothamsted-compiled introductory manual

However, two respondents answered:

'No – let Genstat concentrate on statistics and not try to outsas SAS'

and

'No – Genstat stands alone – it doesn't need gimmicks'.

Conclusions

Genstat is used in New Zealand by a wide range of people coming from many different jobs and with widely different backgrounds in statistical training. Only around 40% of the users are statisticians. The actual amount of computing done by Genstat users, both with and without Genstat, also varies tremendously. Genstat is thus expected to be suitable for 'dipping in and out of' as well as for full-time use. This is, in my opinion, a pretty tall order, but one that can, I think, be achieved with the proposed syntax rationalisation and simplification and with new documentation. Genstat is widely viewed as the only 'true statistical package', although some competition is creeping in from SAS. If SAS were as widely available as Genstat the picture might look a little different. SAS is considered to be ahead on graphics capabilities and documentation – two areas in which it is widely considered there is room for improvement in Genstat. The inclusion of Exploratory Data Analysis routines would also greatly enhance Genstat.

In this paper I have attempted to paint a picture of Genstat in New Zealand rather than to simply criticise its shortcomings. I will end with another quotation from another leading New Zealand statistician just to reassure the authors of Genstat of the general feelings of New Zealand users: 'Be sure to tell them we really wouldn't have thought up all these comments and criticisms if we didn't love Genstat and want to keep on using it'.

The Use of Genstat for the Analysis of Designed Experiments at the International Institute of Tropical Agriculture

Nguyen Ky Nam
International Institute of Tropical Agriculture
P M B 5320
Ibadan
Nigeria

Introduction

The International Institute of Tropical Agriculture (IITA) is one of the 13 non-profit international agricultural research and training centres supported by the Consultative Group on International Agricultural Research (CGIAR). Each year, IITA conducts about 1200 to 1500 agricultural experiments with cassava, cowpeas, maize, rice, soybeans, sweet potatoes and yams. The value of these experiments depends not only on the soundness of their designs but also on how they are run and whether or not the collected data are properly analysed.

Before 1984, CRISP (Crop Research Integrated Statistical Package) was the only statistical package used for the analysis of designed experiments at IITA. Until recently, this 'user-friendly' package was also used at two sister CGIAR institutes: ICRISAT and ICARDA. Genstat became available at IITA with the installation of two new Vax 11/750's in August 1983. SAS has been available since mid-1984 and is used mainly for the analysis of our socio-economic survey data.

Why we prefer Genstat to other Statistical Packages

At IITA we consider Genstat to be more suitable for the analysis of designed experiments than other statistical packages, for the following reasons:

(i) Genstat ANOVA can analyse any designs belonging to the class of generally balanced designs, covered in Cochran and Cox [2]. This class of designs includes complex designs like confounded block designs and lattice designs. These designs, with their advantage of small block sizes, are increasingly used by IITA owing to the problem of soil-heterogeneity in Africa.

(ii) Unbalanced designs are rarely used at IITA. Genstat is the only statistical package having the algorithm capable of detecting that a design becomes unbalanced due to wrong entry of factor subscripts in the data file.

(iii) Missing plots in our field experiments are not uncommon. Genstat is the first statistical package we have found which can handle missing data with ease. We prefer Genstat's method of handling missing values to SAS's, as the 'balancedness' of the design is maintained, easing interpretation of the results.

(iv) Genstat output is neater and more informative than the output of other statistical packages, particularly when the designs have multiple error structure, like split-plots and strip-plots. The plots of residuals against fitted values are very useful for detecting outliners and checking the validity of the assumptions underlining the analysis of variance.

There are more detailed comparisons between Genstat and other packages for the analysis of designed experiments elsewhere, for instance, Federer and Henderson [3], Heiberger [4], Preece [6].

How we make Genstat User-friendly at IITA

Unlike CRISP, Genstat is not user-friendly and, for this reason, it was rarely used by IITA scientists when it was first introduced. Like scientists elsewhere, they consider Genstat a

statisticians' package rather than a statistical package (Bryan-Jones, [1]). To make Genstat popular, we wrote a conversational interface for it in the manner of CRISP. The program, called IITAG, was completed in July 1984: it has 20 options, indicated below.

CONVERT:	convert a CRISP data file to an IITAG data file.
COR:	print the correlation matrix of variates in the data file.
CRD:	analyse a completely randomised design.
EXIT:	exit from IITAG and return to DCL $ prompt.
HELP:	display the list of available options.
HISTO:	draw a histogram of each variate in the data file.
LATBAL:	perform an intra-block analysis of a balanced lattice design.
LATSIM:	perform an intra-block analysis of a simple lattice design.
LATSQ:	perform the analysis of a Latin square or Youden square design.
PLOT:	draw simple point plots and line plots.
POLREG:	perform a polynomial regression analysis.
PRINT:	list all data values in the data file.
RBD:	perform the analysis of randomised block, balanced incomplete block and balanced confounded block designs.
REGRES:	perform simple or multiple regression analysis.
RUN:	instruct IITAG to process the data.
SETUP:	create a new data file.
SPLIT:	analyse a split plot design.
SPLSPL:	analyse a split-split plot design.
STRIP:	analyse a strip-plot design.
TRAN:	transform data, e.g. COUNT=LOG(COUNT+1).

All IITAG's ANOVA options have provisions for checking whether the design becomes unbalanced due to wrong entry of factor subscripts, for handling missing data, for covariate adjustments and for plotting of residuals against fitted values. CRD, RBD and LATSQ can deal with multi-factorial situations. SPLIT can deal with cases with more than one main treatment factor and more than one sub-treatment factor. IITAG exploits the specific order of blocking and treatment factors in the IITAG data file to generate the correct BLOCK and TREATMENT directives for the specified ANOVA option. For each ANOVA option, the user has to specify whether a variate in the data file is to be analysed (V), to be used as a covariate (C) or to be skipped (S).

For the REGRES option, the user has to specify whether a variate in the data file is to be used as dependent variate (Y), to be used as independent variate (X), or to be skipped (S).

For almost a year, IITAG has been used satisfactorily by IITA scientists, research scholars and technicians. However, IITAG also has a number of drawbacks related to the structure of its data file, as follows:

(i) It is very time consuming to verify an IITAG data file by computer (using the Vax DIFFERENCE command) because the data have to be entered twice, together with information about the data, such as the description of the experiment, the number of experimental units, the number of factors, factor names and associated factor level labels, and the number of variates and variate names.

(ii) IITAG does not generate a Genstat program suitable for the analysis of data sets from similar experiments at different locations. It is also difficult to merge data from different locations for a combined analysis.

(iii) Although IITAG does not expect users to derandomise field records before data input, some users still like to do so (arranging the data in neat tables like those in Cochran and Cox [2]), use CRISP's SETUP for data input and then convert CRISP's data file to IITAG's. Errors are introduced during the process of derandomising field records.

The IITA Genstat code generator became available in June 1985 as our second attempt to make Genstat user-friendly. The program, called GENI, accepts data files which consist only of numeric values created by a separate data entry program or Vax CREATE command. The data file resembles a matrix, with rows representing experimental units and columns representing factors and variates. When invoked, GENI asks the user for information about his experiment and about the type of analysis required, at the same time generating Genstat codes. GENI has 16 options: BASIC, COR, CRD, EXIT, HISTO, LATSQ, LATTICE, PLOT, POLREG, PRINT, REGRESS, RBD, SPLIT, STEPWISE, STRIP and TRAN. Two of these make use of the CSIRO macros BASIC and LATTICE. GENI has no CONVERT, HELP, RUN and SETUP options. The functions of some GENI options are defined below.

BASIC:	print basic statistics of each variate in the data file.
LATTICE:	perform an analysis of a square or rectangular lattice design, with recovery of inter-block information.
SPLIT:	perform an analysis of a split-plot or split-split-plot design.
STEPWISE:	perform a step-wise regression analysis.
STRIP:	perform an analysis of a strip-plot or strip-split-plot design.

The functions of other GENI options are similar to IITAG's. GENI's ANOVA options are designed to accept only data files without derandomisation. GENI is still at the experimental stage; preliminary observations show that users who have no experience with CRISP seem to learn GENI faster.

Neither IITAG nor GENI has options like CRISP's for generating randomisation plans. We have, however, DSIGNX, a package which can generate randomisation plans and field maps for all designs commonly used at IITA.

The following examples illustrate the use of GENI to analyse a simple 3 × 3 × 2 factorial experiment with cowpeas in 4 randomised blocks. The three treatment factors are planting date, variety and seed treatment (seed dressing versus no seed dressing). The recorded variates are counts of numbers of infected plants and yields.

```
@BIOMET:GENI

Enter name of your data file (without extension): SH52
Enter title? INTEGRATED CONTROL OF COWPEA SAVANA DISEASES ZARIA 84
Enter number of experimental units? 48
Enter number of factors including blocking factors? 4
Enter name of factor 1 (max 8 characters)? BLOCK
Enter number of levels for BLOCK? 4
Enter label for level 1 of BLOCK? <RET>
Enter name of factor 2 (max 8 characters)? DATE
Enter number of levels for DATE? 3
Enter label for level 1 of DATE? JULY 31
```

```
INVALID label.  Enter label for level 1 of DATE? JULY31
Enter label for level 2 of DATE? AUGUST8
Enter label for level 3 of DATE? AUGUST16
Enter name of factor 3 (max 8 characters)? VARIETY
Enter number of levels for VARIETY? 3
Enter label for level 1 of VARIETY? TVX3236
Enter label for level 2 of VARIETY? ITA6Ø
Enter label for level 3 of VARIETY? IFEBROWN
Enter name of factor 3 (max 8 characters)? SEEDTRMT
Enter number of levels for SEEDTRMT? 2
Enter label for level 1 of SEEDTRMT? WITH
Enter label for level 2 of SEEDTRMT? WITHOUT
Enter number of variates in your data file? 2
Enter name of variate 1 (max 8 characters)? COUNT
Enter name of variate 2 (max 8 characters)? YIELD
Enter one option: [BASIC..TRAN]? TRAN
Enter transformation equation? LOGCOUNT=LOG(COUNT)
Enter one option: [BASIC..TRAN]? HISTO
Enter one option: [BASIC..TRAN]? RBD
Enter one letter (V variate, C covariate, S skip) for COUNT? V
Enter one letter (V variate, C covariate, S skip) for LOGCOUNT? V
Enter one letter (V variate, C covariate, S skip) for YIELD? V
Enter one option: [BASIC..TRAN]? EX

Enter YES to continue the data analysis with GENSTAT : YES
Please wait for GENSTAT to analyse your data.
Enter YES to display the GENSTAT output on the screen: YES
```

{ *GENSTAT output is displayed* }

```
Enter YES to queue the GENSTAT output for printing   : YES
Job 999 entered on queue SYS$PRINT.
```

The following Genstat command file was generated by GENI.

```
'REFE' ''INTEGRATED CONTROL OF COWPEA SAVANA DISEASES ZARIA 84''
'UNIT $ 48
'FACTOR' BLOCK$ 4
'NAME' N2=JULY31,AUGUST8,AUGUST16 'FACTOR' DATE$ N2
'NAME' N3=TVX3236,ITA6Ø,IFEBROWN 'FACTOR' VARIETY$ N3
'NAME' N4=WITH,WITHOUT 'FACTOR' SEEDTRMT$ N4
'SET' VARSET=COUNT,YIELD
'INPUT' 2
'READ/NUN=Q,FLEV=F' BLOCK,DATE,VARIETY,SEEDTRMT,VARSET
'INPUT' 1
'CALC/M' LOGCOUNT=LOG(COUNT) 'SET' VARSET=VARSET,LOGCOUNT
'HISTOGRAM' VARSET
'FACTOR' PLOT$ 12  'GENE'  4 ,PLOT
'BLOCK' BLOCK/PLOT 'TREAT' DATE*VARIETY*SEEDTRMT ''RBD''
'FOR' YSET=COUNT,YIELD,LOGCOUNT
```

```
'ANOVA/PROB=Y' YSET;RES=RESIDS;FVAL=FITTED
'GRAPH' RESIDS;FITTED
'REPEAT'
'RUN'
'CLOSE'
'STOP'
```

In this example, everything has gone smoothly. If something goes wrong, Genstat will fail to analyse the data and the user will receive a message

```
Please ask for assistance from the biometrician!
```

Conclusion

IITAG and GENI together serve most of the routine data analysis needs of IITA. For complex analyses, the experimenters can always obtain help from the biometrician. A number of IITAG and GENI users are now able to write their own Genstat programs without the help of these interfaces. This is the ultimate objective of IITAG and GENI, like any conversational interface designed as an attractive way to start using a program (Lane, [5]).

If the new version of Genstat can accomodate factor level labels with 16-20 characters, there will be increasing use of Genstat by our breeders. Our variety names are rather lengthy.

Both IITAG and GENI are written in BASIC. The program listing for GENI (about 200 lines of BASIC), the listing of the associated Vax command file and the user's guide can be obtained from the author.

References

[1] Bryan-Jones, J
A conversational approach to using Genstat.
Genstat Newsletter **9**, 23-27, 1982.

[2] Cochran, W G and Cox, G M
Experimental Designs.
New York: John Wiley, 1957.

[3] Federer, W T and Henderson, H V
Covariance analysis of designed experiments using statistical packages.
Statistical Computing Section, Proceedings of the American Statistical Association, 332-337, 1978.

[4] Heiberger, R M
The specification of experimental designs to ANOVA programs.
The American Statistician **35**, 98-104, 1981.

[5] Lane, P W
A conversational interface for Genstat Mark 5.
Genstat Newsletter **12**, 28-33, 1983.

[6] Preece, D A
The design and analysis of experiments: What has gone wrong?
Utilitas Mathematica **21A**, 201-244, 1982.

A Genstat Analysis for Intercropping Stability

J Riley
Overseas Development Administration Biometrician
Statistics Department
Rothamsted Experimental Station
Harpenden
Hertfordshire AL5 2JQ
United Kingdom

Most agricultural research aims to produce recommendations for growing sole crops which will ultimately be harvested by machine. This is true of most tropical agricultural research, as sole cropping has now been introduced to a large number of developing countries. However, ninety-five percent of the farmers in the tropics and semi-arid tropics farm less than two hectares of land, an area that must provide a whole range of subsistence or cash crops. Fertilisers, pesticides and herbicides are expensive and not always available. Rainfall is unpredictable and the tropical soils suffer from sudden storms and long periods of drought. Farming systems have developed over the centuries to enable the farmer to achieve the maximum return from his land even when such difficult conditions prevail. The most common farming system is intercropping, where different species are grown on one area of land for some time, or all, of their life-cycles, the species intermingled or in alternate rows. By mixing the different crops in this way, greater yields can be achieved, greater ground cover is possible for a longer part of the growing season, better control of weeds, pest and disease is gained and better use of light and nutrients can be achieved. Typical mixtures of crops are maize and beans, millet and sorghum or sugarcane and maize, although the small-scale farmer mixes many more than two species at any one time – up to thirteen different crops in one mixture is not an uncommon sight in Northern Nigeria!

The persistence of tropical farmers in practising intercropping and the indication that better yields can be achieved by mixing crops than by growing them sole, have resulted in research to improve the choice of genotypes for mixtures, to determine optimum spacings for different combinations of crops, to determine which additional nutrients, if any, are needed to improve yields from intercrops and to estimate the most beneficial planting and harvesting dates for mixtures. Designs for intercropping experiments have been investigated and suitable statistical analyses have been proposed. A review of such methods is given in Mead and Riley [3]; a summary of them is given here together with the Genstat instructions for programming the statistical analyses.

Design of Experiments

The main questions under investigation when intercropping experiments are performed are:

(1) which combinations of crops are suitable for intercropping?

(2) which genotypes of each species are suitable for intercropping?

(3) are intercrop returns greater than those achieved from sole crops?

(4) does an intercrop yield more with one treatment than with another?

(5) are yields from intercrops more reliable over time than monocrops would be when resources are limited and the climate is unreliable?

Research into intercropping is at a very early stage and there is much ignorance about suitable levels of the factors involved – ignorance which is compounded by the addition of a second crop. Designs for intercropping experiments are not dissimilar from those for monocropping experiments, but the factors of interest are many more because of the complications introduced by the relative spatial arrangement of the two species. A typical intercropping experiment consists of

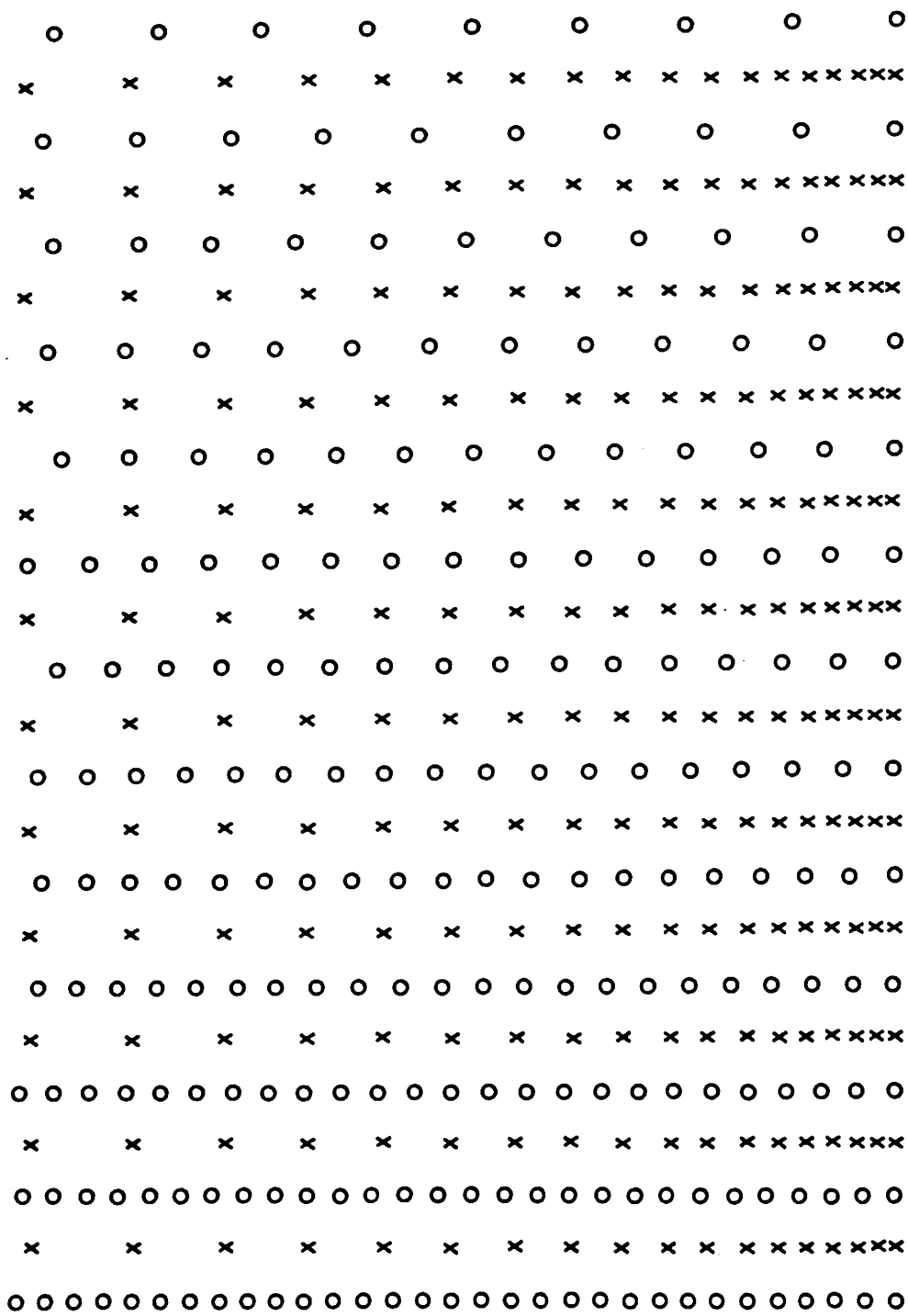
replicates of a number of plots on each of which an intercrop of the same two species is grown with a different treatment. This may be, say, an application of fertiliser, a spacing treatment or a combination of two particular genotypes whose performance is being tested when they are intercropped. The pairs of yields from each plot can then be used to compare the effects of the different treatments. In addition to the intercropped plots, there may be a number of plots sown with sole crops of the two species, each with one of the treatments that are being tested. These sole-cropped plots may be randomised amongst the intercropped plots or they may be positioned around the edge of the experimental area, rather like control plots in a monocropping experiment. Comparisons can then be made between the intercrop yields and the sole-crop yields for those treatments for which a comparison is to be made. Figure 1 shows one replicate block of a 3 × 2 factorial experiment with two sole-crop treatments. S_1, S_2 are two sorghum population densities; P_1, P_2, P_3 are three pigeonpea genotypes; S and P are sole-crop sorghum and sole-crop pigeonpea, these two plots being randomised among the intercrop plots.

S_1P_2	S_2P_3	P	S_2P_2
S_2P_1	S	S_1P_1	S_1P_3

Figure 1

The experimental designs that are used for intercropping are usually randomised complete block designs. The use of split-plot designs should be discouraged unless their use is required because of practical necessity. Comparisons between levels of several factors are likely to be of equal importance in an intercropping trial; a split-plot design involves different precision for different comparisons and it is unlikely that those comparisons of interest will be estimated with the required precision if such a design is used.

Systematic designs have been used to investigate spacing effects in intercropping experiments where the crop densities or the relative plant arrangement, or both, change monotonically in one or more directions across each large experimental plot. (Wahua and Miller, [5]; Huxley and Maingu, [1]). Figure 2 shows a two-way systematic design for two crops, x and o, where the densities vary in perpendicular directions (Mead, [2]). The main advantage of a systematic design over a randomised block design is that a greater proportion of the total area can be harvested, this being a major gain when a large number of crop densities is being investigated. Care should always be taken, however, that a systematic plot is not laid along a trend, whose effect will influence the shape of the response curve for the spacing treatments that are being examined. Ideally, the large systematic plots should be laid out so that the changes in density run across trends, not along them.



Two-way systematic spacing design for two crops (x and o) with the densities varying in perpendicular directions
Figure 2

Stability of Intercrop Yields

The value of a farming system is reflected not only in the yields it provides but also in its reliability over time in giving sufficient yields for the farmer. To assess whether an intercropping system is risky or not, its performance over time, or over a number of different sites, can be compared with the performance of a sole-crop system grown at the same places and at the same times. This approach results in a set of pairs of data values which can then be analysed by bivariate analysis. The following approach to the assessment of the stability of a sorghum/pigeonpea intercrop compared with a sole sorghum crop is in Mead *et al* [4]. The intercrop and the sole crop were grown at eleven different sites for up to seven years, the yields obtained totalling 51 pairs of values. The sole sorghum returns (SORG) ranged from 6.67 to 74.5 monetary units and the intercrop returns (XCROP) ranged from 6.70 to 80.94 monetary units. Simple use of the 'GRAPH' directive with appropriate headings displays the bivariate scatter within the data as in Figure 3. An empirical relative risk curve can be constructed by specifying a series of levels of return d_i and then calculating the number of observations that fall below d_i for each farming system. Thus for $d_1 = 6.8$, there is no value for each system returning less than this, so the estimated risk probabilities for each system are $\frac{1}{51}$. Pairs of risk probabilities can be built up in this way for all possible values of d_i and the resultant two variates of risk probabilities can then be plotted against each other as in Figure 4, again using simple 'GRAPH' instructions. The figure shows that for risks up to 50% the risk for the intercrop is half that for the sole crop. When the risk is larger than 50%, the intercrop is still more reliable than the sole crop, although the chance of failure is more likely than is indicated in the earlier part of the graph.

If a suitable bivariate distribution can be fitted to the joint data set then the theoretical risks can be calculated and applied to other data sets. To fit a distribution to the whole data set we must examine the temporal and spatial variation to see whether the relationship between the two sets of returns is the same for these two components of the total variation. Since the pattern of year × site combinations is irregular, the bivariate analysis of variance can be produced using the 'REGRESS' directive and by fitting the year and site effects in either order:

```
'CALC' TOTAL = SORG + XCROP
     : DIFF = XCROP - SORG

'FOR' VL = XCROP, SORG, TOTAL, DIF
'REGRESS' VL, YEAR, SITE
'Y' VL
'FIT/A, ANDEV = IT' YEAR, SITE

'FIT/A, ANDEV = IT' SITE, YEAR

'REPE'
```

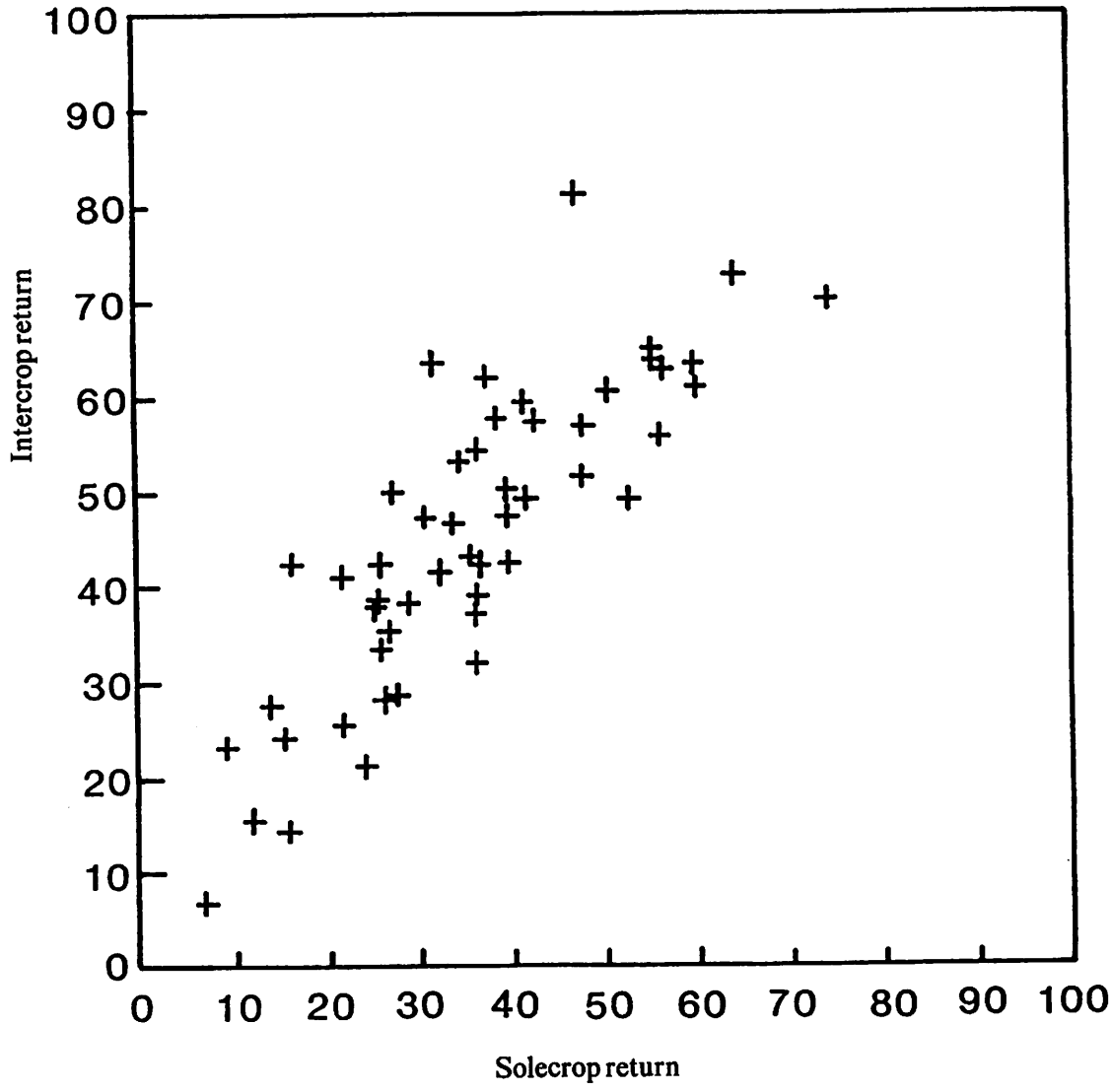

Intercrop return
Solecrop return
0
10
20
30
40
50
60
70
80
90
100

Figure 3

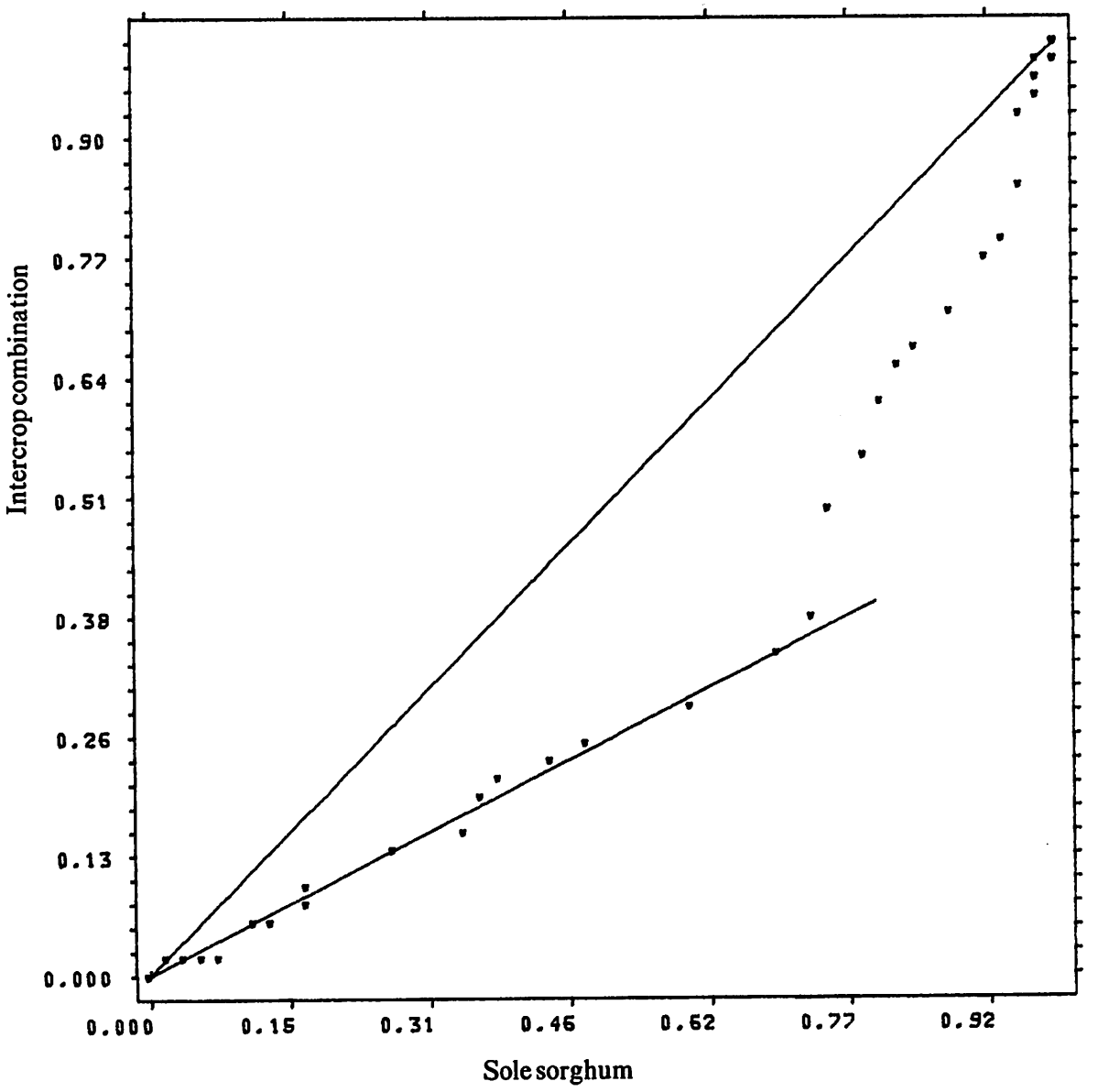

0.90
0.77
0.64
0.51
0.38
0.26
0.13
0.000
Intercrop combination
0.000
0.15
0.31
0.46
0.62
0.77
0.92
Sole sorghum

Figure 4

The sums of products can be calculated from the sums of squares for TOTAL and DIF and the resultant bivariate analysis takes the form:

Source	df	XCROP	SP	SORG
Sites (ignoring years)	10	4685	4249	4613
Years (eliminating sites)	6	792	469	736
Years (ignoring sites)	6	1133	686	1219
Sites (eliminating years)	10	4344	4032	4130
Years + sites	16	5477	4718	5349
Residual	34	5392	5145	7353
Total	50	10869	9863	12702

Examining the correlation between the two returns, we see that the sums of products are all positive and similar in size to the corresponding sums of squares. Further analysis shows that regressions of intercrop return on sole sorghum return, accounting individually for each of the above sources of variation, differ very little. For this data set it can therefore be assumed that the relationship between the two variables is consistent over years and sites and the data set can be used as a whole.

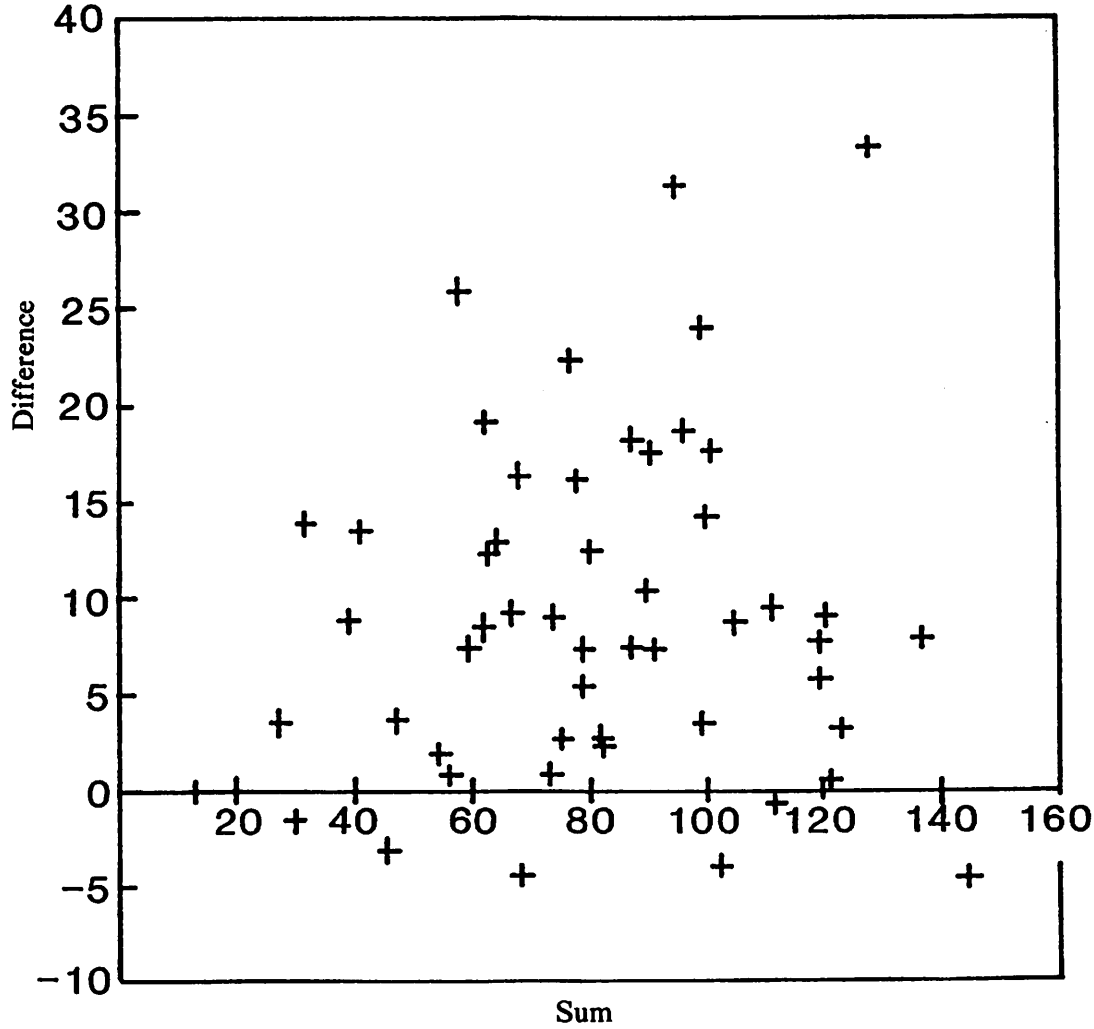


Figure 5

It is often easier to study the relationship between the original two return variables by examining their sum (TOTAL) and their difference (DIF). Fitting a model to these two variables can be done in two stages: by fitting a distribution to one variable ignoring the other and then by fitting a distribution to the second variable allowing for its dependence on the first variable. Using 'GRAPH' and appropriate headings again, a display of the sums and differences as in Figure 5 suggests that the distribution of sums might be symmetric, such as the Normal. A simple probability plot confirms this and so the estimated mean and variance for the Normal distribution of TOTAL are

$$mean = 82.8, \quad variance = 852.7$$

For the conditional distribution of differences there is a suggestion in Figure 5 that the mean difference is a quadratic function of the sum and that the variance of the difference may also be a symmetric function of the sum. Using 'REGRESS', a quadratic relationship between DIF and TOTAL can be found:

```
'CALC' S1 = TOTAL - 82.8
     : S2 = S1**2
'REGRESS' DIF, S1, S2

'Y' DIF

'FIT/AC, ANDEV = IT' S1, S2; RES = R1
```

The resultant equation is

$$E(\mathtt{DIF}) = -6.97 + 0.443\, sum - 0.00274\, sum^2$$

Although the assumption of non-constant variance of the differences suggests weighting by the reciprocal of the estimate of the variance at each point, using such weights makes very little difference to the fitted curves.

The variance of the difference is then modelled as the exponential of a quadratic function of the sum, to ensure that the fitted function will everywhere be positive. The data for this are the squared residuals (R1**2) from the estimated mean difference and the fitting is done using a log link and gamma errors:

```
'CALC' R2 = R1**2

'REGRESS' R2, S1, S2

'Y/ERROR = GAMMA, LINK = LOG' R2

'FIT/AC, ANDEV = IT' S1, S2
```

The resulting equation for the log variance of the difference is

$$log\,(Var(\mathtt{DIF})) = 4.32 - (\mathtt{TOTAL} - 79.77)^2/2302$$

Study of the residuals from the fitted equation shows that it describes the data sufficiently well.

Using these estimated distributions for TOTAL and DIF, the distribution functions can be found, by integration, for the original variables XCROP and SORG. From these, the probabilities of each system falling below a particular disaster level can then be estimated and used to compare the relative reliabilities of the two farming systems.

Acknowledgement

The author is funded by the U.K. Overseas Development Administration.

References

[1] Huxley, P A and Maingu, Z
Use of a systematic spacing design as an aid to the study of intercropping: some general considerations.
Expl. Agric., **14**, 49-56, 1978.

[2] Mead, R
Competition Experiments
Biometrics, **35**, 41-54, 1979.

[3] Mead, R and Riley, J
A review of statistical ideas relevant to intercropping research (with discussion).
J.R.S.S. (A), **144**, 462-509, 1981.

[4] Mead, R, Riley, J, Dear, K and Singh, S P
Stability comparison of intercropping and monocropping systems.
Proceedings of the XIIth International Biometrics Conference, Tokyo, Japan, September 1-8, 1984, 86-99, 1984.

[5] Wahua, T A T and Miller, D A
Relative yield totals and yield components of intercropped sorghum and soybeans.
Agron. J., **70**, 287-291, 1978.

A Genstat Program for General Block Designs

R E Kempson
Wye College
University of London
Wye
Ashford
Kent TN25 5AH
United Kingdom

Abstract

The general method of design and analysis of blocked experiments provides a useful tool for the selection of the best design for each particular problem as well as the exact analysis for a non-orthogonal experiment. The computer program `CLIFFORD` was written in Genstat, to provide the calculations which result from the general method, and was originally used as a teaching aid.

Introduction

In a designed experiment, all the blocks may be composed in the same way with respect to treatments. In this case the design is orthogonal and the analysis is straightforward. However, it is sometimes preferable or necessary to apply a non-orthogonal design to a particular problem for a variety of reasons:

(a) an orthogonal design may have missing data,

(b) there may be physical or practical restrictions on the design,

(c) certain treatment contrasts may be of major importance.

The general method for block designs not only provides an exact method of data analysis but, in addition, enables the experimenter to allocate the treatments in a way which makes effective use of resources in order to improve the chances of detecting real treatment differences, should these exist.

The model for the general block design is

$$\mathbf{Y} = \mathbf{D}'\beta + \Delta'\gamma + \varepsilon$$

where the explanatory terms are the block means and the treatment effects adjusted for blocks.

Notation

It will be helpful at this stage to introduce some notation

β	=	vector of block means	
γ	=	vector of treatment effects	
$\mathbf{D}$	=	design matrix for blocks	(blocks × cases)
Δ	=	design matrix for treatments	(treatments × cases)
$\mathbf{N}$	=	incidence matrix	(treatments × blocks)
$\mathbf{B}$	=	vector of block totals	
$\mathbf{T}$	=	vector of treatment totals	
$\mathbf{r}$	=	vector of replications	
$\mathbf{k}$	=	vector of block sizes	
$\mathbf{r}^{\delta}$	=	diagonal matrix formed from $\mathbf{r}$	
$\mathbf{k}^{\delta}$	=	diagonal matrix formed from $\mathbf{k}$	
$\mathbf{x}^{-\delta}$	=	inverse of $\mathbf{x}^{\delta}$	

These definitions lead to some important identities which simplify the algebra considerably.

$$\mathbf{DD}' = \mathbf{k}^{\delta}$$
$$\Delta\Delta' = \mathbf{r}^{\delta}$$
$$\Delta\mathbf{D}' = \mathbf{N}$$
$$\mathbf{DY} = \mathbf{B}$$
$$\Delta\mathbf{Y} = \mathbf{T}$$

Minimization of the error sum of squares leads to the equations

$$\mathbf{k}^{\delta}\hat{\beta} + \mathbf{N}'\hat{\gamma} = \mathbf{B}$$
$$\mathbf{N}\hat{\beta} + \mathbf{r}^{\delta}\hat{\gamma} = \mathbf{T}$$

If $\hat{\beta}$ is eliminated from these simultaneous equations the vector of adjusted treatment effects $\hat{\gamma}$ is given by

$$(\mathbf{r}^{\delta} - \mathbf{N}\mathbf{k}^{-\delta}\mathbf{N}')\hat{\gamma}) = \mathbf{T} - \mathbf{N}\mathbf{k}^{-\delta}\mathbf{B}$$

Now define

$$\mathbf{C} = \mathbf{r}^{\delta} - \mathbf{N}\mathbf{k}^{-\delta}\mathbf{N}'$$
$$\mathbf{Q} = \mathbf{T} - \mathbf{N}\mathbf{k}^{-\delta}\mathbf{N}'$$

and the equation becomes

$$\mathbf{C}\hat{\gamma} = \mathbf{Q}$$

The solution of this equation is difficult because C is singular. One method of solving the problem is to reduce the number of parameters, to eliminate their linear dependence, and the other method is to employ generalized inverses. There are infinitely many solutions $\mathbf{C}^{-}$ for the equation, but the most famous and useful solutions are the following:

Ω	Tocher
Ξ	Pearce
$\mathbf{C}^{+}$	Moore and Penrose
T	Kuiper and Corsten

Ω was introduced by Tocher [6] and is given by the equation

$$\Omega^{-1} = \mathbf{C} + \frac{1}{m}\mathbf{r}\mathbf{r}'$$

where m is the total number of units. After a matrix inversion Ω may be used to give the following results, but any of the other generalized inverses would do equally well.

$$\hat{\gamma} = \Omega\mathbf{Q}$$
$$V(\hat{\gamma}) = \Omega\sigma^2$$
$$\textit{Adjusted Treatments SS} = \mathbf{Q}'\Omega\mathbf{Q}$$
$$V(\mathbf{c}'\hat{\gamma}) = \mathbf{c}'\Omega\mathbf{c}\sigma^2$$

where $\mathbf{c}'\gamma$ is a treatment contrast. Proofs of these results are found in Pearce [5]. The partial analysis of variance table for b blocks, v treatments and m units is

Source	df	ss
Blocks	$b-1$	$\mathbf{B}'\mathbf{k}^{-\delta}\mathbf{B} - \mathbf{CT}$
Treatments (adj)	$v-1$	$\mathbf{Q}'\Omega\mathbf{Q}$
Error	$m-b-v+1$	*difference*
Total	$m-1$	$\mathbf{Y}'\mathbf{Y} - \mathbf{CT}$

Other Generalized Inverses

Let $\mathbf{P} = \mathbf{N}\mathbf{k}^{-\delta}\mathbf{N}'$ then put $\mathbf{W} = \mathbf{P} - diag(\mathbf{P})$ so the leading diagonal of **P** is replaced by zeros to form **W**. Now collapse the rows of **W** by postmultiplication by a vector of 1s to form $\mathbf{q} = \mathbf{W1}$. If $\mathbf{q}'\mathbf{1} = u$ where u is a scalar then another generalized inverse Ξ is found from the equation

$$\Xi^{-1} = \mathbf{C} + \frac{1}{u}\mathbf{q}\mathbf{q}'.$$

This solution was discovered by Pearce [4].

Another solution is known as the Moore-Penrose inverse $\mathbf{C}^{+}$ and is formed from the spectral decomposition of **C** in terms of its eigenvalues λ and their normalised eigenvectors **U**.

$$\mathbf{C} = \lambda_1 \mathbf{U}_1 \mathbf{U}'_1 + \lambda_2 \mathbf{U}_2 \mathbf{U}'_2 + \ldots + \lambda_v \mathbf{U}_v \mathbf{U}'_v$$

Since the eigenvectors are orthonormal the powers of **C** are easily found from this expression as

$$\mathbf{C}^j = \sum_{i=1}^{v} \lambda_i^j \mathbf{U}_i \mathbf{U}'_i$$

No eigenvalue is negative since **C** is symmetric but as **C** is singular it has a zero eigenvalue. The last term of the spectral decomposition vanishes and the inverse $\mathbf{C}^{+}$ is found when j takes the value -1.

$$\mathbf{C}^{+} = \sum_{i=1}^{v-1} \frac{1}{\lambda_i} \mathbf{U}_i \mathbf{U}'_i.$$

The other generalized inverse is found by the Kuiper-Corsten iteration method. This method consists of a series of projections between two vector spaces

$$\mathbf{u}_j = \mathbf{k}^{-\delta}\mathbf{N}'\mathbf{v}_j \qquad \mathbf{v}_{j+1} = \mathbf{r}^{-\delta}\mathbf{N}\mathbf{u}_j.$$

At the design stage the initialisation takes $\mathbf{v}_1$ as the first column of

$$\mathbf{r}^{-\delta} - \frac{1}{m}\mathbf{1}\mathbf{1}'$$

but if data are available take

$$\mathbf{v}_1 = \mathbf{r}^{-\delta}\mathbf{Q}.$$

The spectral decomposition of **F** gives $\mathbf{F}^{+}$, corresponding to $\mathbf{C}^{+}$, and **T** is given by

$$\mathbf{T} = \mathbf{r}^{-\delta/2}\mathbf{F}^{+}\mathbf{r}^{-\delta/2}$$

In addition, the parameter estimator is found as the infinite sum

$$\hat{\gamma} = \sum_{i=1}^{\infty} \mathbf{v}_i.$$

The general theory of linear models is described by Graybill [2]. A thorough but early account of the general theory applied to block designs was presented by Tocher [6]. More recently, Pearce [5] has given an extensive treatment of the subject, which is otherwise not generally available in textbook form. A thorough treatment of generalized inverses for block designs was presented by Catchpole [1]. The use of the general method for the selection of optimal designs was described by Jones [3], who wrote a useful computer program in this connection.

Example

A well-known example from Pearce [5, p 83] will serve as an illustration. Four types of herbicide A, B, C, D and a control O were applied to strawberry plants and their total spread in inches was measured.

Block 1	D 107,	A 166,	B 133,	C 166,	O 177,	A 163,	O 190
Block 2	A 136,	O 146,	D 104,	C 152,	B 159,	O 164,	B 132
Block 3	C 118,	A 117,	O 176,	B 132,	C 139,	O 186,	D 103
Block 4	O 173,	D 95,	D 109,	A 130,	B 103,	O 185,	C 147

A computer program CLIFFORD was written in Genstat to perform the calculations described above, and the example will serve to illustrate the output. A listing of the program appears below. For the example, a segment of the output may be presented as follows:

$$\mathbf{N}^{+} = \begin{matrix} 2&1&1&1 \\ 1&2&1&1 \\ 1&1&2&1 \\ 1&1&1&2 \\ 2&2&2&2 \end{matrix} \quad \mathbf{r} = \begin{matrix} 5 \\ 5 \\ 5 \\ 5 \\ 8 \end{matrix} \quad \mathbf{k} = \begin{matrix} 7 \\ 7 \\ 7 \\ 7 \end{matrix} \quad \mathbf{T} = \begin{matrix} 712 \\ 619 \\ 722 \\ 518 \\ 1397 \end{matrix} \quad \mathbf{B} = \begin{matrix} 1102 \\ 953 \\ 971 \\ 942 \end{matrix}$$

$$\mathbf{C} = \begin{matrix} 4.00000 & & & & \\ -0.85714 & 4.00000 & & & \\ -0.85714 & -0.85714 & 4.00000 & & \\ -0.85714 & -0.85714 & -0.85714 & 4.00000 & \\ -1.42857 & -1.42857 & -1.42857 & -1.42857 & 5.71429 \end{matrix}$$

$$\Omega = \begin{matrix} 0.20441 & & & & \\ -0.00147 & 0.20441 & & & \\ -0.00147 & -0.00147 & 0.20441 & & \\ -0.00147 & -0.00147 & -0.00147 & 0.20441 & \\ 0.00000 & 0.00000 & 0.00000 & 0.00000 & 0.12500 \end{matrix}$$

$$\hat{\gamma} = \begin{matrix} -2.14216 \\ -16.90689 \\ 3.76957 \\ -37.37752 \\ 32.91075 \end{matrix} \qquad \mathbf{Q} = \begin{matrix} -12.28540 \\ -83.99976 \\ 16.42871 \\ -183.42847 \\ 263.28589 \end{matrix}$$

Analysis of variance

Source	df	ss	ms	F
Blocks	3	2366.000	788.667	
Treatments (adj)	4	17029.457	4257.363	24.61 ***
Error	20	3460.293	173.015	
Total	27	22855.750		

$$\Xi = \begin{matrix} 0.21258 & & & & \\ 0.00670 & 0.21258 & & & \\ 0.00670 & 0.00670 & 0.21258 & & \\ 0.00670 & 0.00670 & 0.00670 & 0.21258 & \\ 0.01212 & 0.01212 & 0.01212 & 0.01212 & 0.14107 \end{matrix}$$

$$\mathbf{C}^{+} = \begin{matrix} 0.16141 & & & & \\ -0.04447 & 0.16141 & & & \\ -0.04447 & -0.04447 & 0.16141 & & \\ -0.04447 & -0.04447 & -0.04447 & 0.16141 & \\ -0.02800 & -0.02800 & -0.02800 & -0.02800 & 0.11200 \end{matrix}$$

$$\mathbf{F} = \begin{matrix} 0.80000 & & & & \\ -0.17143 & 0.80000 & & & \\ -0.17143 & -0.17143 & 0.80000 & & \\ -0.17143 & -0.17143 & -0.17143 & 0.80000 & \\ -0.22588 & -0.22588 & -0.22588 & -0.22588 & 0.71429 \end{matrix}$$

$$\mathbf{F}^{+} = \begin{matrix} 0.84349 & & & & \\ -0.18592 & 0.84349 & & & \\ -0.18592 & -0.18592 & 0.84349 & & \\ -0.18592 & -0.18592 & -0.18592 & 0.84349 & \\ -0.22588 & -0.22588 & -0.22588 & -0.22588 & 0.71429 \end{matrix}$$

$$\mathbf{T} = \begin{matrix} 0.16870 & & & & \\ -0.03718 & 0.16870 & & & \\ -0.03718 & -0.03718 & 0.16870 & & \\ -0.03718 & -0.03718 & -0.03718 & 0.16870 & \\ -0.03571 & -0.03571 & -0.03571 & -0.03571 & 0.08929 \end{matrix}$$

Eigenvectors of **C**

-0.22361	-0.28868	0.70711	-0.40825	-0.44721
-0.22361	-0.28868	-0.70711	-0.40825	-0.44721
-0.22361	-0.28868	-0.00000	0.81650	-0.44721
-0.22361	0.86603	0.00000	0.00000	-0.44721
0.89443	0.00000	0.00000	0.00000	-0.44721

Eigenvectors of **F**

-0.26726	0.70711	-0.28868	-0.40825	-0.42258
-0.26726	-0.70711	-0.28868	-0.40825	-0.42258
-0.26726	0.00000	-0.28868	0.81650	-0.42258
-0.26726	-0.00000	0.86603	-0.00000	-0.42258
0.84515	0.00000	0.00000	0.00000	-0.53452

The Program CLIFFORD

A Genstat program CLIFFORD was written to perform the calculations described in the first three chapters of Pearce [5]. The program contains a small driver program and six macros which are outlined below.

DECLARE: declares the sizes of variates and matrices

DESIGN: forms the elementary vectors and matrices for design calculations
Output: $\mathbf{D}, \Delta, \mathbf{N}, \mathbf{r}, \mathbf{k}, \mathbf{P}, \mathbf{C}, \Omega^{-1}, \Omega$

PEARCE: calculates Pearce's generalized inverse Ξ
Output: $\mathbf{W}, \mathbf{q}, \Xi^{-1}, \Xi$

RESMAT: calculates the residual matrices Φ and Ψ and, if data are available, the residuals may be computed
Output: data and residuals; Φ, Ψ

ANOVAR: obtains the analysis of variance
Output: analysis of variance table

KUIPER: performs the Kuiper-Corsten iteration, the spectral decomposition of the matrices C and F, and obtains the Moore-Penrose $\mathbf{C}^{+}$ and Kuiper's T.
Output: projection vectors u and v from the iteration, sum of the vs, eigenvalues, eigenvectors and trace of C and $\mathbf{F}, \mathbf{C}^{+}, \mathbf{F}^{+}, \mathbf{T}$

The user stipulates the levels of detail and analysis he requires. The levels of detail are DT = 0,1,2,3 and for analysis AN = 0,1. The driver program calls the macros which satisfy these requirements.

	Detail level			
Analysis	0	1	2	3
0	DESIGN	PEARCE	KUIPER(a)	RESMAT(b)
1	ANOVAR	RESMAT(a)	KUIPER(b)	RESMAT(b)

The macro KUIPER uses different initial vectors for the iteration process, depending upon whether analysis is required. RESMAT obtains the residuals if AN = 1 but prints the residual matrices only if DT = 3.

Input

NB number of blocks
NT number of treatments
M number of units
DT detail level
AN analysis level

Block levels
Treatment levels
Data values (optional)

The block levels, treatment levels and data are in corresponding order. The data may be omitted for an experiment at the design stage.

Listing of the Program CLIFFORD

```
'REFE/NUNN=400,NID=250' CLIFFORD
'MACR' DECLARE $
''DECLARATIONS''
'VARI' DVAL,DELVAL $ M
'MATR' K,B $ NB,1
:      R,T,Q,GAMMA,Q1 $ NT,1
:      Y,RES $ M,1
:      N $ NT,NB
:      D $ NB,M = (0)A1
:      DELTA $ NT,M = (0)A2
:      ONEM $ M,1 = (1)M
:      ONET $ NT,1 = (1)NT
:      CPLUS,FPLUS $ NT,NT = (0)A3
:      UM,VM $ NT,NT
'SYMM' PHI,PSI $ M
:      OMEGA,OMINV,C,F,UPSILON,XI,XIINV,W,P,RD,RMD,FS $ NT
:      KD,KMD $ NB
'DIAG' Q1D,SV,PD $ NT
:      IM $ M = (1)M
'ENDM'

'MACR' DESIGN $
''DESIGN CALCULATIONS''
'UNIT' $ M
'VARI' V1 $ M = 1...M
'CALC' V2 = (DVAL-1)*M+V1
:      V3 = (DELVAL-1)*M+V1
'COPY' D $ V2 = 1
:      DELTA $ V3 = 1
'CALC' N = PDTT(DELTA;D)
:      K = PDT(D;ONEM)
:      KD = PDTT(D;D)
:      KMD = INV(KD)
:      R = PDT(DELTA;ONEM)
:      RD = PDTT(DELTA;DELTA)
:      RMD = INV(RD)
:      P = RSYMRI(N;KMD)
:      C = RD -  P
:      OMINV = C + A4*PDT(R;TRANS(R))
:      OMEGA = INV(OMINV)
'PRIN/S' D,DELTA,N,R,K $ 3
:      P,C,OMINV,OMEGA $ 10.5
'ENDM'

'MACR' PEARCE $
'CALC' PD = P
:      W = P - PD
```

```
:       Q1 = PDT(W;ONET)
'COPY' Q1D = Q1
'CALC' A6 = TPDT(Q1;ONET)
:       A5 = 1/A6
:       XIINV = C + A5*PDTT(Q1;Q1)
:       XI = INV(XIINV)
'PRIN/S' W,Q1,XIINV,XI $ 10.5
'ENDM'

'MACR' RESMAT $
''RESIDUALS AND RESIDUAL MATRICES''
'JUMP' LABR1*(DT.NE.3.AND.AN.NE.1)
'CALC' PHI = IM -  RSYMRI(TRANS(D);KMD)
:       PSI = PHI - RSYMRI(PDT(PHI;TRANS(DELTA));OMEGA)
'LABE' LABR1
'JUMP' LABR2*(AN.EQ.0.OR.DT.EQ.3)
'CALC' RES = PDT(PSI;Y)
'LINE' 2
'CAPT' ''DATA VALUES AND RESIDUALS''
'INTE' IND $ M=1...M
'PRIN/P,VAR=1,LABC=1' IND,Y,RES $ 10,(10.5)2
'LABE' LABR2
'JUMP' LABR3*(DT.LT.3)
'PRIN/S' PHI,PSI $ 10.5
'LABE' LABR3
'ENDM'

'MACR' ANOVAR $
''ANALYSIS CALCULATIONS''
'CALC' B = PDT(D;Y)
:       T = PDT(DELTA;Y)
:       Q = T -  PDT(N;PDT(KMD;B))
:       TSS = RSYMRI(TRANS(Q);OMEGA)
:       GAMMA = PDT(OMEGA;Q)
:       RTSS = TPDT(Y;Y)
:       G = SUM(Y)
:       CT = G*G/M
:       CTSS = RTSS-CT
:       BSS = RSYMRI(TRANS(B);KMD)-CT
:       RSS = CTSS-BSS-TSS
'PRIN/S' B,T,Q,GAMMA $ 10.5
'HEAD' H1 = ''BLOCKS''
:       H2 = ''TREATMENTS(ADJ)''
:       H3 = ''RESIDUAL''
:       H4 = ''TOTAL''
'CALC' BDF = NB-1
:       RDF = M-NT-NB+1
:       DF  = M-1
:       BMS = BSS/BDF
```

```
:       TMS = TSS/TDF
:       RMS = RSS/RDF
:       BVR = BMS/RMS
:       TVR = TMS/RMS
'LINE' 6
'CAPT' ''ANALYSIS OF VARIANCE *****''
'LINE' 1
'CAPT' ''SOURCE                DF              SS             MS          VR''
'LINE' 1
'PRIN/C,LABR=1' H1,BDF,BSS,BMS,BVR $ 6,14X,3,2X,(14.4)2,12.2
:               H2,TDF,TSS,TMS,TVR $ 15,5X,3,2X,(14.4)2,12.2
:               H3,RDF,RSS,RMS     $ 8,12X,3,2X,(14.4)2
:               H4,DF,CTSS         $ 5,15X,3,2X,14.4
'ENDM'

'MACR' KUIPER $
'LINE' 2
'CAPT' ''KUIPER-CORSTEN ITERATION''
'MATR' U(1...10) $ NB,1
:      V(1...10),SUMV $ NT,1
'MATR' IK $ NT,NT
'CALC' IK = RMD - A4*PDTT(ONET;ONET)
'EQUA' V(1) = IK $ 1,A7
'JUMP' LABK*(AN.EQ.0)
'CALC' V(1) = PDT(RMD;Q)
'LABE' LABK
'CALC' SUMV = V(1)
'FOR' UA=U(1...9);VA=V(1...9);VB=V(2...10)
'CALC' UA = PDT(KMD;TPDT(N;VA))
:      VB = PDT(RMD;PDT(N;UA))
:      SUMV = SUMV + VB
'REPE'
'PRIN/P' U(1...9) $ 10.5
:          V(1...10) $ 10.5
'PRIN' SUMV $ 10.5
'LINE' 2
'CAPT' '' EFFICIENCY MATRIX''
'VARI' RS $ NT
'CALC' RS =  1/SQRT(R)
'DIAG' RT $ NT
'MATR' RMHD $ NT,NT
'EQUA' RT = RS
'CALC' RMHD = RT
'SYMM' F $ NT
'CALC' F = RSYMRI(RMHD;C)
'PRIN' F $ 10.5
'LINE' 2
'CAPT' ''SPECTRAL DECOMPOSITION OF THE MATRICES C AND F''
'MATR' VM $ NT,NT
```

```
'DIAG' RTS $ NT
'SCAL' TR
'SCAL' LR(1...NT)
'MATR' LV(1...NT) $ NT,1
'FOR'  MAT=C,F;MATP=CPLUS,FPLUS
'LRV'  MAT;VM,RTS,TR
'PRIN/S' RTS,VM,TR $ 10.5
'EQUA' LV(1...NT) = VM $ (1,(X)TDF)NT,X
'EQUA' LR(1...NT) = RTS
'FOR' LAMBDA = LR(1...TDF);LV = LV(1...TDF)
'CALC' MATP = MATP + (1/LAMBDA)*PDTT(LV;LV)
'REPE'
'PRIN' MATP $ 10.5
'REPE'
'CALC' FS = FPLUS
'CALC' UPSILON = RSYMRI(RMHD;FS)
'LINE' 2
'PRIN' UPSILON $ 10.5
'ENDM'
'PRIN' DECLARE  : DESIGN  : PEARCE  : RESMAT  : ANOVAR  : KUIPER
''HERBICIDE TREATMENT OF STRAWBERRIES EXAMPLE FROM PEARCE P 83''
''STATE THE NUMBERS OF BLOCKS, TREATMENTS, UNITS AS NB, NT,M
  DETAIL LEVEL     DT = 0,1,2,3  (1 RESIDUALS, 2 F C+ UPSILON, 3 PHI+PSI)
  ANALYSIS LEVEL   AN = 0,1      (0 DESIGN, 1 ANALYSIS)      ''
'SCAL' NB,NT,M,DT,AN
'READ/P' NB,NT,M,DT,AN
'SCAL' BSS,TSS,RTSS,G,CTSS,RSS,CT,V,BDF,TDF,RDF,DF,BMS,TMS,RMS,BVR,TVR
:      A1,A2,A3,A4,A5,A6,A7
'CALC' A4=1/M : A1=NB*M : A2=NT*M : A3=NT*NT : A7=NT-1
:      TDF=NT-1
'RUN'
4 5 28 1 0
'USE/R' DECLARE $
'READ/S' DVAL,DELVAL
'JUMP' LAB1*(AN.EQ.0)
'READ' Y
'LABE' LAB1
'RUN'
 1 1 1 1 1 1 1 2 2 2 2 2 2 2 3 3 3 3 3 3 3 4 4 4 4 4 4 4 'EOD'
 4 1 2 3 5 1 5 1 5 4 3 2 5 2 3 1 5 2 3 5 4 5 4 4 1 2 5 3 'EOD'
107 166 133 166 177 163 190 136 146 104 152 119 164 132
118 117 176 132 139 186 103 173  95 109 130 103 185 147  'EOD'
'USE/R' DESIGN $
'JUMP' LAB2*(DT.EQ.0)
'USE/R' PEARCE $
'LABE' LAB2
'JUMP' LAB3*(AN.EQ.0)
'USE/R' ANOVAR $
'LABE' LAB3
```

```
'JUMP' LAB4*(DT.LT.1)
'USE/R' RESMAT $
'LABE' LAB4
'JUMP' LAB5*(DT.LT.2)
'USE/R' KUIPER $
'JUMP' LAB5*(DT.LT.3)
'USE/R' RESMAT $
'LABE' LAB5
'RUN'
'CLOS'
'STOP'
```

Note The program in its present form is intended only for designs which are not confounded.

Confounding may be detected when the last row of Ω consists of zeros, in which case **C** has more than one zero eigenvalue.

A more general version of the program is being developed and will be the subject of a future note in the Newsletter.

References

[1] Catchpole, E A
Generalized inverses in block designs.
J.S.P.I. **10** Number 2, 195-202, 1984.

[2] Graybill, F A
Theory and application of the linear model.
Duxbury, 1976.

[3] Jones, B J
An algorithm for deriving optimal block designs.
Technometrics **18**, 451-458, 1976.

[4] Pearce, S C
Concurrences and quasi-replications: an alternative approach to precision in designed experiments.
Biometrics J. **18**, 105-112, 1976.

[5] Pearce, S C
The agricultural field experiment: a statistical examination of theory and practice.
Wiley, 1983.

[6] Tocher, K D
The design and analysis of block experiments.
J.R.S.S. (B) **14**, 45-100, 1952.

The Testing of Anti-Dandruff Shampoos – An Application of Genstat

G Jagger, K A Whinney, J Fincham
Life Science Research Ltd
Elm Farm Laboratories
Occold
Suffolk IP23 7PX
United Kingdom

Introduction

The Human Studies unit of Life Science Research, CTC International, has been performing trials in human volunteers since 1972 and has become one of the leading laboratories in Europe for such work. CTC provides an independent evaluation of the safety, efficacy and consumer acceptability of a wide range of products. Investigations are designed to meet the requirements of regulatory authorities and to substantiate marketing claims worldwide. Materials investigated routinely include perfumes, cosmetics, toiletries, personal care, household and industrial products.

Trials are frequently conducted on behalf of a number of internationally active clients to assess the relative efficacies of different anti-dandruff shampoos. These shampoos may contain different active ingredients or they may have different concentrations of the same active ingredient. Various formulations may also be tested which are intended for different hair types, such as dry, normal or greasy. In addition an inactive shampoo, or the current market leader, may sometimes be included as a 'control'.

This paper describes the conduct of such a trial and the subsequent analysis of the measurements obtained by means of Genstat.

Conduct of Trial

The organisation of each shampoo trial follows the same general pattern. A large number of volunteers are recruited in a number of different locations. From these volunteers are chosen a number of groups, each group being treated with a different shampoo. The dandruff score before and after treatment is assessed, in the subjects' home areas, by trained examiners or graders. The precise number of subjects selected for any given trial depends upon the number of shampoos being tested and the required power of the test, and is calculated using power function tables. In the trial described here, 387 males and females were recruited and examined to assess the amount and severity of dandruff present. They were issued with a bland, commercially available, shampoo to use at home twice a week for two weeks.

Females who admitted to being pregnant or were nursing mothers, and volunteers who had seborrhoeic dermatitis or psoriasis with visible inflammation of the scalp, who were currently receiving medical treatment for a scalp condition, who were under 14 years old, or who intended to tint or perm their hair during the trial, were excluded from the trial.

On the day before the first treament (Day 0), 320 volunteers were assessed for dandruff and were selected to proceed with treatment.

Subjects were allocated to four treatment groups by a block procedure such that there were 23 male and 57 female subjects in each group. The blocks were formed on the basis of location, sex, examiner and initial score. Subjects' ages were balanced between shampoos by a Latin square method of allocating subjects to treatment within blocks. This ensured that the group means for dandruff were approximately similar and that the groups were balanced by age and sex, as far as possible. The balancing procedure also provided that the panels at each location contained balanced proportions of subjects from each treatment group as far as possible. Members of the same household were allocated to the same treatment group.

The accepted subjects were issued with one of the four test shampoos to use twice weekly at home for six weeks. They were given instruction sheets incorporating a calendar showing the dates on which they were to use the shampoos allocated to them. They were asked to mark the days when they washed their hair and not to wash their hair for four days before attending for dandruff assessment. Neither the subjects nor the investigator were aware of the identity of the test products at any stage of the trial. The dandruff examiners were not aware of the treatment groups to which the subjects were allocated.

Allocation of test shampoo to individual subjects took place on Day 1, the day following the pre-treatment assessment. All subjects were instructed not to wash their hair between the time they attended for assessment and the time when they received the test shampoo. Subjects collected and signed for their coded test shampoo at the assessment centre. Supplementary supplies of coded test shampoo were distributed on Day 28 or sent by post to subjects, for which they signed and returned a receipt.

Fifteen days before treatment started (Day −14), one day before treatment started (Day 0), and on Days 14, 28 and 42, subjects attended for assessment. Observations were made on subjects while they were seated under a 40-watt white fluorescent strip light 122 cm in length, positioned 183 cm above floor level. The same examiner assessed the same subjects throughout the trial as far as possible. The examiners assessed dandruff at each attendance.

Examiners assessed the four quadrants of the scalp separately for dandruff in terms of proportion of the scalp affected and severity. The hair was parted and secured with clips, leaving the area to be examined free; the examiner then used the handle of a tailcomb to make successive partings in the quadrant concerned. The amount and type of dandruff on the scalp were assessed by observation, and by drawing the tip of the comb handle along the partings to ascertain whether the scales were loosely attached or adhering to the scalp.

Dandruff was scored according to the following criteria:

Area of quadrant affected	**Score**
Less than 10%	0
10% or more, but less than 30%	1
30% or more, but less than 50%	2
50% or more, but less than 70%	3
70% or more	4
Severity within quadrants	
Small flakes resembling a coarse greyish white powder (Grade A)	1
Intermediate (Grade AB)	2
Large flakes very loosely attached to the scalp and giving an irregular whitish surface (Grade B)	3
Intermediate (Grade BC)	4
Flakes adhered to the scalp as white or yellow plates (Grade C)	5
Small flakes as 'A' partially adhering to scalp as 'C' (Grade AC)	3

For each quadrant, the score for area was multiplied by the score for severity. The scores for the four quadrants were then added together. The maximum possible score was therefore $(4 \times 5) \times 4 = 80$.

Statistical Analysis

The dandruff is measured as objectively as possible, as described above, the score being derived from both the severity involved and the area of the scalp affected. Measurement is on an integer

scale from 0 to 80; this gradation makes it reasonable to consider the dandruff scores to be continuous for practical purposes.

Several potential systematic sources of variation exist, in addition to the inherent random variation in response between subjects. These include sex, location (possibly due to varying weather conditions), subjects' ages, normal frequency of usage, the grader performing the assessments. Most important of all, subjects with a high initial score can be expected, on average, to show an appreciably greater reduction than those subjects with a comparatively low initial score. It is quite common for a subject with an initial score of 75 or greater to finish the trial with a score less than 5. In contrast subjects starting with a score of say, 15, can display a much lower maximum reduction or improvement.

The possible sources of variation also mean that care must be taken to ensure that no undue bias exists in comparing treatment effects. In practice this is achieved by a randomised block procedure such that each treatment (or shampoo) occurs once in each block, each block containing subjects of the same sex and of similar initial score: they also come from the same location and are assessed by the same grader. Depending on the products being tested they may also be of the same hair type. Thus the subjects assigned to each shampoo will have similar distributions with respect to sex, location, grader and initial score. By extending the blocking to a Latin square design the distribution of ages within each shampoo is also equalised as far as possible. In addition to blocking, analysis of covariance can also be used, both to reduce the variability and through adjusted means to lessen any bias.

The trial chosen as an illustration contains 320 subjects, four shampoos, three locations, five graders and, of course, two sexes. The response variable is the change in dandruff score between the start of the trial and the score after six weeks of treatment. Analysed as a completely randomised design, without analysis of covariance, the F-ratio for differences between treatments is 1.889, the corresponding p-value is greater than 0.1 (=0.13). There is thus no evidence from this analysis of differences in efficacy between shampoos. The ANOVA table is shown in Table 1.

Source of variation	df	ss	ss%	ms	F
Shampoo	3	1911.0	1.76	637.0	1.889
Residual	316	106573.8	98.24	337.3	
Total	319	108484.8	100.00		

Completely randomised block design
Table 1

Taking blocks into account and remembering that one blocking factor is sex, which is a frequently required subset of the analysis, the variation can be partitioned as follows:
between blocks; sex, blocks within sexes:
within blocks; shampoo, shampoo × sex interaction, residual:
total.
The F-ratio for treatment differences is now 3.200, the corresponding p-value being less than 0.025 (=0.024). The shampoo × sex interaction is not signifcant (p=0.065). The block effect is very highly significant (p<0.001). The ANOVA table is shown in Table 2.

Source of variation	df	ss	ss%	ms	F
Between blocks	79	58532.3	53.95	740.9	3.722
Sex	1	278.9	0.26	278.9	0.373
Residual	78	58253.4	53.70	746.8	
Within blocks	240	49952.5	46.05	208.1	
Shampoo	3	1911.0	1.76	637.0	3.200
Shampoo × sex	3	1455.4	1.34	485.1	2.437
Residual	234	46586.1	42.94	199.1	
Total	319	108484.8	100.00		

Nested randomised block design
Table 2

Since the blocks are formed separately by sex and grader within each location, there necessarily exists some heterogeneity with respect to initial score within blocks. For this reason, analysis of covariance could be expected to provide a more sensitive analysis and the adjusted means more unbiased estimates of treatment effects. This technique also allows age and normal frequency of usage to be used as covariates.

The F-ratio for treatment differences has become 5.072, $p<0.01$ ($=0.002$). There is now strong evidence of a true difference between treatment effects. The shampoo × sex interaction is also significant ($p=0.03$). From the Genstat output, age and normal frequency of usage have little influence; in fact, they could profitably be dropped since the loss of two degrees of freedom had slightly inflated the error variance. The ANOVA table is shown in Table 3.

Source of variation	df	ss	ss%	ms	F
Between blocks	79	58288.7	53.73	737.8	6.384
Sex	1	35.2	0.03	35.2	0.255
Covariates	3	47907.5	44.16	15969.2	115.764
Residual	75	10346.0	9.54	137.9	
Within blocks	240	49383.1	45.52	205.8	
Shampoo	3	1758.7	1.62	586.2	5.072
Shampoo × sex	3	1038.3	0.96	346.1	2.994
Covariates	3	19886.2	18.33	6628.7	57.350
Residual	231	26699.9	24.61	115.6	
Total	319	107671.8	99.25		

Nested randomised block design with covariates
Table 3

Covariate	Regression coefficient	Standard error
Age	0.040	0.056
Normal usage	0.6	1.23
Initial score	0.779	0.0605

Nested randomised block design with covariates
Table 3 (Contd)

Using only initial score as covariate, the F-ratio is 5.103 (p=0.0019), the error mean square being 114.9.

The proportion of variation attributable to random factors has been reduced from 0.98 in the first analysis to 0.25 in the final analysis. The standard error of the difference between two means has been reduced from 2.90 to 1.70.

The mean responses are as shown in Table 4.

	Shampoo			
	1	2	3	4
Unadjusted	21.0	19.2	25.9	22.5
Adjusted	21.35	18.66	24.23	24.36

Mean responses
Table 4

The Genstat program which produces the above analyses is straightforward and is as follows:

```
'REFERENCE' SHAMPOO_TRIAL
'UNITS' $ 320
'NAMES' SE =M,F
   :    LOC=L1,L2,L3
   :    GRA=G1,G2,G3,G4,G5
'FACTORS' SEX     $ SE
   :      BLOCKS  $ 80
   :      SHAMPOO $ 4
   :      GRAD    $ GRA
   :      PLACE   $ LOC
'INPUT' 2
'READ/P' SEX,PLACE,GRAD,WASH,AGE,W0,W6
'INPUT' 1
'CALCULATE' SCORE=W0-W6
'GENERATE' BLOCKS,SHAMPOO
'TREATMENTS' SHAMPOO
'ANOVA' SCORE
'RUN'
'BLOCKS' BLOCKS+BLOCKS.SHAMPOO
'TREATMENTS' SHAMPOO+SEX+SHAMPOO.SEX
'ANOVA' SCORE
'RUN'
```

```
'COVARIATES' AGE,WASH,WØ
'ANOVA' SCORE
'RUN'
'COVARIATES' WØ
'ANOVA' SCORE
'RUN'
'CLOSE'
'STOP'
```

Conclusion

No two shampoo trials are ever the same and the fact that the experimental units are human beings makes any desire for perfection in the experimental design a lost cause. Because of the competitive nature of the trials market no testing organisation can afford to tailor-make computer software for each trial. Fortunately this is not necessary since Genstat provides a very flexible and ready means of analysing the data and one which lends itself to the Exploratory Data Analysis type of approach.

Not only have the authors found Genstat to be particularly suitable for the analysis of the kind of trial described in this paper, but client companies have been suitably impressed by the speed and detail of the results. Genstat has proven to be an invaluable tool in the analysis of a wide range of experiments in the life sciences.

An Enquiry into the Relation of Accident Numbers to Traffic Flow and Vehicle Speeds

U Engel, L K Thomsen
Raadet for Trafiksikkerhedsforskning
- Sekretariatet
Ermelundsvej 101
DK-2820 Gentofte
Denmark

Introduction

The study treated in this paper is being carried out at the secretariat of the Danish Council of Road Safety Research. The aim is twofold.

In the short run, the data collected on vehicle-speeds can be used to answer simple questions such as 'What is the average speed in Danish urban areas?', 'How many motorists exceed the 60 km/h speed limit?' and 'What is the trend of these speeds?'.

In the long run, the study will provide useful information on the influence of vehicle speeds, traffic flow etc on accidents. Different street types are compared, thus providing necessary information for accident prevention and evaluation.

This paper is on methodology and only a few results are mentioned for the sake of completeness.

The Measuring Technique

Speed measurements are carried out from a car parked at the side of the street, using an 'Electromatic type S5' radar. For each vehicle, 10 speed signals are stored on a cassette recorder. If the signals are steady, they are accepted as the vehicle speed. Shortly after the operator has collected the signals for a vehicle, he keys in information on the type of vehicle (taxi, van, lorry, coach, bus, emergency vehicle or motorcycle) and on the traffic situation (whether the vehicle is alone, in front of a queue, travelling in a queue or changing speed for some reason).

Whenever the speed of a vehicle is recorded, the following information is also registered: location, time, weather and road conditions, the identification number of the operator and the identification number of the equipment.

At any time, up to eight persons are counting the traffic at the same spot. Eleven categories of road users are counted: car, taxi, light goods vehicle, lorry, coach, bus, emergency vehicle, motorcycle, moped, bicycle and pedestrian.

Each measuring interval lasts four hours. After this, the radar operator goes to the nearest telephone box and makes a call to the secretariat of the Council. The audible radar signals are transmitted over the telephone and punched on papertape at the secretariat. This procedure ensures a fast check on the quality of the data.

This system is very convenient and enables us to finish even complicated statistical analysis within four hours from the end of the speed measuring session.

The Design of the Experiment

Seventeen towns and villages in different areas, with populations in the range 203 to 536,931, were selected for the speed measurements. Wherever possible, speeds on four types of street are collected. Speeds are measured for eight hours in a total of 65 streets.

In addition to these measurements, some speed measurements are made to establish particulars of hour to hour variations, day to day variations and, to some extent, the effect of the type of measuring vehicle.

This number of streets is needed for a satisfactory accident analysis, as discussed below.

Four main categories of street are considered. They are:

(1) Shopping street

(2) Residential area, tall buildings

(3) Residential or other area with low buildings

(4) A mixed group made up of
   (a) Industrial area
   (b) Other buildings (with openings on road)
   (c) No buildings, buildings with no openings on road.

Danish traffic accidents are recorded by the police using the above list and, for this reason, the same categorisation is used for the speed measuring sites.

The number of streets selected was decided on knowledge of accident rates in Copenhagen. It is estimated that the 65 streets will provide us with 400 accidents in a year. Accident data were first collected for the years 1976-1980, in order to get an idea of the numbers of accidents and their characteristics.

Danish traffic accidents are recorded by the police with numerous details. For use in this study, the main variables will be: road user (11 categories as mentioned above), type of road and the estimated speed of the road user before the accident.

A lot of (to some extent) secondary information is available as well. This concerns, for example, road surface conditions, weather conditions, location, level of blood alcohol and use of seat belt. Such variables might be checked to ensure stable conditions.

The Mathematical Model

The basic assumption behind this study is that the number of accidents and casualties can be described by

$$A = f(\mathbf{x}) \tag{1}$$

where A designates the number of accidents on a certain section of road during a fixed period of years. The number of accidents is assumed to be a function of the vector $\mathbf{x}$, which describes street geometry, speed characteristics and traffic flows.

If the categorisation of accidents is thorough, the related distribution is Poisson (Thomsen, [2]). The probability distribution is given by

$$g(A=a) = \frac{\lambda^a}{a!} e^{-\lambda} \tag{2}$$

where a is the observed number of accidents and λ the accident intensity.

Up to this point the assumptions made are theoretically and empirically well-founded. The choice of the function f is of a much more heuristic nature. Two possible simple models are

$$f_1(\mathbf{x}) = \beta_0 + \sum_{i=1}^{n} \beta_i x_i \tag{3}$$

and

$$f_2(\mathbf{x}) = \gamma_0 \prod_{i=1}^{n} x_i^{\mu_i} \tag{4}$$

Both (3) and (4) have been tried and, in both cases, it has proved possible to obtain a reasonable description of the accident figures for the 65 streets.

For the sake of completeness it should be stressed that the combination of formulae (1), (2) and, for example, (3) gives a multiple regression (covariance) model with Poisson error structure.

A few Results and Future Prospects

The covariance analysis mentioned above yields interesting results. The main ones are

(1) Mean speeds are positively correlated with accident numbers.

(2) The standard deviations of the speeds do not influence the number of accidents.

(3) The skewness of the speed distribution does not influence the number of accidents.

In addition to these results, several clues relating to flow and street geometry have been found but are not reported here.

The collection of data as described above is well suited for evaluation studies. The covariance analysis gives a thorough insight into accident fluctuations and an induced change in one of the independent variables will permit a before and after study with control group and thus form an efficient experimental design.

References

[1] Engel U and Thomsen, L K
Speed limits and traffic accidents in urban areas.
Danish Council of Road Safety Research, working paper No. 8, 1981.

[2] Thomsen, L K
An inquiry into the relation between traffic flow and accidents.
Danish Council of Road Safety Research, working paper No. 9, 1985.

Macro Library, Manual and Notice Board Amendment

H R Simpson
Statistics Department
Rothamsted Experimental Station
Harpenden
Hertfordshire AL5 2JQ
United Kingdom

Macro Library: Error

The macro BIPLOTV will not work if the number of variates NV is greater than eight. The source should be changed as follows:–

Add a new identifier DUMMY to the list of LOCALS in lines 1 and 2. After the SCALAR statement (line 3), insert

```
'START'
'ASSIGN'  DUMMY=VSET $ 1
'CALC'    N=NVAL(DUMMY)
'RUN'
'START'
```

and replace VSET by END in the next line. (The calculation of N at line 8 is now redundant, and can be removed if the : at the beginning of line 9 is replaced by 'CALC'.) Finally, delete line 6 ('STAR').

Manual Amendments

Amended pages for the Genstat 4.04 Manual are enclosed with this Newsletter. Any users holding Genstat 4.03 manuals should apply the same amendments, deleting sections marked '!4'.

Further copies of the amendments may be obtained from NAG Central Office.

Genstat 4.04B Notice Board

These error notices should be inserted in the appropriate (alphabetic) position in the Notice Board.

```
R77          ***** Error ****
'ANOVA' &    The function FPROB is unreliable for large numbers (over about 100)
'CALC'       of degrees of freedom, giving probabilities that are (much) too low.
             The consequences of this carry over to the Fprob column of ANOVA.

R78          ***** Error ****
'CALC' &     Whenever the PRINT option of 'CALCULATE' or 'READ' is set to M
'READ'       and the mean of a set of values is exactly zero, no decimal places
             are used when printing the minimum, mean and maximum.
```