BU-699-M

THE GENSTAT LANGUAGE AND ITS USE

R.I. Baxter, CSIRO Division of Mathematics and Statistics, Sydney.
E.W. Jones, Biometrics Unit, Cornell University.

Interest in GENSTAT has recently been stimulated by favourable comparisons with other statistical packages. GENSTAT is a language which contains the usual arithmetic and logical operators as well as directives which carry out statistical procedures such as analyses of designed experiments, regression and generalised linear models, multivariate and cluster analyses. Operations are performed on data structures which include scalars, vectors, tables and matrices. Tables are multi-dimensional arrays that may optionally be augmented by marginal rows and columns for totals or means, and there is a calculus defined for manipulation of tables. The operations of matrix arithmetic are provided, and some special operations such as calculation of eigenvalues and eigenvectors are also available. A GENSTAT job can consist of several directives in a block which is compiled then executed, or may contain several program blocks which are compiled then executed sequentially or in a nested manner, but this is still only one job-step on the computer. A GENSTAT job may, at any point, save the data structures and essential directories in a user-file. The user-file can later be retrieved and the program status restored. Macros can be written in the GENSTAT language to define new statistical operations and a collection of these macros may be stored in a user-file as a macro library.

1. INTRODUCTION

GENSTAT is a general purpose statistical package that provides a powerful programming language, extensive facilities for managing data files, arithmetic operations and functions for deriving data variates, as well as a wide range of statistical procedures. It is documented in Alvey et al. [1].

GENSTAT is now widely distributed, predominantly in the United Kingdom, Western Europe, and Australasia. At some sites the system is heavily used, for example in CSIRO where GENSTAT has been running on a CYBER 76 since late 1974, there are now nearly 200 jobs run each day.

Recently there have been papers and notes comparing statistical packages from different points of view (Bernard [3]; Federer and Henderson [5]; Payne and Nelder [9]; Searle, Henderson and Federer [11]), and in all of these GENSTAT has compared favourably. These have stimulated renewed interest in GENSTAT and this paper aims to give an outline of the package with emphasis on those features that may justify the installation of GENSTAT at computer centres where other major statistical packages are already in use.

2. THE GENSTAT LANGUAGE

GENSTAT, like some other statistical packages and languages such as APL, has operations defined to accept complex data structures as operands, rather than single elements of these structures as is the case with FORTRAN.

The earliest version of GENSTAT only accepted a rectangular data matrix where the rows corresponded to experimental units or cases, and columns were data variates or classifying factors. However, this was found to be too restrictive, and GENSTAT now offers a range of data structures that can be defined with any dimensions, although many simple programs still only define variates and/or factors of the same length.

The major data structures are:

scalar	- a single real value
variate	- a vector of reals
factor	- a vector of integers in the range 1 to N (where N is the number of categories). The integer values 1 to N may be associated with names or reals.
names	- a vector of 8 character names
matrix	- a rectangular matrix of reals
symmat	- the lower triangle (by rows) of a symmetric matrix
diagmat	- a diagonal matrix of reals
table	- a multi-dimensional array of reals with or without margins
heading	- text string

All these structures may be defined in the program in explicit declarations, for example

```
'VARIATE' HT,WT,AGE(3)$25
```

defines 3 variates each of length 25. The identifiers may have up to 8 characters with an optional integer suffix appended in parentheses. The suffices permit shorthand notation for variate lists since the list X(1),X(2),X(3),X(4) may be written as X(1,2,3,4) or X(1...4).

A GENSTAT program consists of a sequence of directives in free format, which generally have the form

```
'directivename/options'sequence of lists
```

where the options and lists have names, so may either be written unnamed in correct order, or may be named in arbitrary order. Omitted options take default values, while omitted lists usually mean that particular output structures are not to be saved for further operations. The following directives are all equivalent

```
'ANOVA/LIMA=4,PR=414' VAR=YLD(1),YLD(3);RES=R1,R2
'ANOV/414,,,4' YLD(1,3);R1,R2
'ANOV/414,LIMA=4' YLD(1,3);RES=R1,R2
```

and specify an analysis of variance on variates YLD(1) and YLD(3) with residuals to be saved in variates R1 and R2. The options request that 4 factor interaction terms be computed and printed.

Many GENSTAT directives have an extensive list of options that provide a considerable degree of control of the computations and printed output. The default values of these options have been carefully chosen so that in many cases the options need not be specified, and the source code remains relatively simple.

3. A SIMPLE EXAMPLE

Draper and Smith [4] present sample data with 25 observations on 10 variates, and they use these data to demonstrate simple linear regressions and a bivariate regression. The following GENSTAT program reads the data set from a separate file, and does the sequence of regressions, producing printout and graphs similar to those shown in Draper and Smith.

```
'REFERENCE' STEAM
'UNITS'$25
'INPUT'2
'READ'X(1...10)
'INPUT'1
'TERMS/C'X(1...10)
'Y'X(1,6)
'FIT/PRIN=CAU'X(8);RES=R(1,6)
'VARIATE'OBS_NO=1...25
'GRAPH' R(1);OBS_NO : R(6); OBS_NO
'TERMS'R(1,6) 'Y'R(1)
'FIT'R(6);RES=R 'GRAPH' R;OBS_NO
'TERMS'X(1,6,8)
'Y' X(1)
'FIT'X(6,8);RES=R
'GRAPH' R;OBS_NO
'RUN'
'CLOSE'
```

4. COMPILATION AND EXECUTION OF GENSTAT PROGRAMS

GENSTAT programs are "compiled" to an internal code which is interpreted during the execution phase. The compiler expands shorthand constructs in the source code, checks details of identifiers by looking up the directory of data structures, and makes new entries in this directory. Thus it needs to know the values of some structures at compile time, for example

```
          'VARIATE' X(1...NV)$100
```

can only be compiled if NV is a scalar with a known value so that the required number of entries can be made in the directory.

At first glance this requirement for values to be known at compile-time may seem to be a severe restriction of the language, and indeed, this would be the case if only one compilation phase and one execution phase was permitted in a job. However, GENSTAT permits a series of blocks of code where the first block is compiled and executed, then the second, and so on. Structures set up and given values during execution of the first block may be referenced by the compiler when compiling the second block. To illustrate this the following program reads the dimensions of a matrix X in the first block, then reads X and computes X'X in the second.

```
'REFERENCE' XPXJOB
'SCALAR' NROW,NCOL
'INPUT'2 'READ'NROW,NCOL 'INPUT'1
'RUN'
'MATRIX' X$NROW,NCOL
'SYMMAT' XPX$NROW
'INPUT'2 'READ'X 'INPUT'1
'CALC'XPX=RSYMRI(X;1)
'PRINT'X,XPX
'RUN'
'CLOSE'
'STOP'
```

In this example, if the first 'RUN' was omitted, then the whole program would be compiled as one block, and the 'MATRIX' declaration would produce a compilation error since NROW and NCOL would not have values defined.

As well as the sequential processing of program blocks described above, program blocks may also be nested. In this mode, execution of the upper-level block is suspended while the lower-level block is compiled then executed, and then the execution of the upper-level block is resumed. The lower level block may be enclosed between 'START' and 'RUN' directives, or may be contained in a macro that is activated by a 'USE/R' directive as described below.

5. MORE ABOUT THE GENSTAT LANGUAGE

One way of writing loops in GENSTAT is to use labels and conditional or unconditional jumps. The following program segment continues to read and analyse data sets of 2*2*2 factorial experiments (possibly with differing numbers of replicates) until an empty data set is encountered.

```
'SCALAR'NEXT,FINISH
'FACTOR'A,B,C$2
'LABEL'NEXT
'INPUT'2
'READ/NUN=V'A,B,C,X
'INPUT'1
'JUMP'FINISH*(NVAL(X).LE.0)
'TREAT'A*B*C 'ANOVA'X
'JUMP'NEXT
'LABEL'FINISH
```

The NUN=V option of READ means that the number of observations is not known in advance so the length of the variate and factors is set to the number of observations encountered in the data set. This will be zero if there is a null data set.

Another method of forming loops uses the 'FOR' and 'REPEAT' directives and is illustrated in the following program segment which prints a heading and a data structure on each page.

```
'HEADING'H1 =''PAGE 1 - STRUCTURE IS A VARIATE''
: H2 =''PAGE 2 - STRUCTURE IS A MATRIX''
: H3 =''PAGE 3 - STRUCTURE IS A SYMMAT''
'VARIATE'X=1...20
'MATRIX'A$5,6=101...130
'SYMMAT'S$6=201...221
'FOR'DUM1=H1,H2,H3 ; DUM2=X,A,S
'PAGE'
'PRINT' DUM1,DUM2
'REPEAT'
```

The directives within the loop are like a subprogram with 2 arguments (or dummies) DUM1 and DUM2. The 'FOR' directive assigns data structures to the dummies for the current pass through the loop. It is worth noting that most GENSTAT directives allow operations to be performed sequentially on a list of structures in one directive. Thus 'FOR' loops are only required where a more complex sequence of operations is to be repeated.

A sequence of GENSTAT directives may be formed into a macro and stored for recall and use as required. The following example shows a macro which takes as input a variate of residuals R and plots a histogram of standarised residuals.

```
'MACRO' RESHIS$
'LOCAL' BOUNDS,STDR,RHIST,XTITL,YTITL
'VARIATE' BOUNDS=-2.8,-2.6...2.8
: STDR$R
'CALC'STDR=(R-MEAN(R))/SQRT(VAR(R))
'GROUPS'RHIST=LIMITS(STDR;BOUNDS)
'HEAD' XTITL =''STANDARDISED RESIDUALS''
: YTITL =''FREQUENCY COUNT''
'GRAPH/ATX=XTITL,ATY=YTITL'RHIST;BOUNDS
'ENDMACRO'
```

This macro may be used as follows

```
'TERMS'Y,X(1...3)
'Y' Y 'FIT'X(1...3);RESID=R
'USE' RESHIS$
```

The 'USE' RESHIS$ directive in the above example is recognized by the compiler and the macro is substituted and compiled into the program block. If instead, 'USER/R' RESHIS$ is encountered by the compiler, it is not acted on at that time. When it is encountered during the execution phase, execution of the current block is suspended, the macro is compiled and executed, and then execution of the calling block is resumed. A use of this is illustrated in the following segment where the macro is used within a loop on variates of different lengths.

```
'VARIATE' VA$50 : VB$100 : VC$200
'CALCULATE'VA,VB,VC = RANDU(29,111,141)
'FOR'R=VA,VB,VC
'USE/R' RESHIS$
'REPEAT'
```

If the 'USE/R' was replaced by a 'USE', the compilation would fail on the declaration of STDR since the length of R would not be defined.

6. FACILITIES FOR DATA MANIPULATION

Values for data structures may be specified in the source code using various shorthand notations, or formats. Many options may be set to control processing of blank fields, missing value symbols, etc. Similarly, there is considerable control over the format for printing structures, and data can be displayed graphically on the printer.

The 'CALCULATE' directive allows element-by-element arithmetic and logical operations on all real valued structures and also permits operations on specific elements within structures. As well as the usual arithmetic operations, functions provide for means, medians, variances and other statistical values. The operations of matrix arithmetic are provided and permit easy programming of multivariate and regression procedures that are not included as standard in GENSTAT. Tables can have values read in or generated from data variates by the 'TABULATE' directive (means, totals, or counts) and then 'CALCULATE' may be used to manipulate these tables. A full table calculus is defined so that tables with different classifying factors can be included in the same expression.

Other directives allow for movement of values between different structures, definition of subsets of vector structures for use in subsequent analyses, and for forming factors from real variates.

A file storage system allows a collection of structures to be formed into a subfile, and a collection of sub-files to be stored as a user-file. Structures can be renamed on storage or retrieval to avoid conflicts. Using this system, jobs can be saved at any point together with directories and control information, and later restarted from that point. It also allows libraries of macros to be maintained.

Some of the facilities described, are demonstrated in the following program which is taken from Chapter 3 of Alvey et al. [1]. It shows the calculation of the Kruskal-Wallis test statistic for data from an abrasion test where the weight loss of three types of paint was recorded. The three paints were applied to 8 panels.

```
'REFERENCE' CALC
'UNITS'PANEL$8
'SCALAR'MNRANK(1...3) : KRUSKWAL=0 : EXPMNRNK=12.5
'READ/S'PAINT(1...3)
'VARIATE'PAINT$24 ; TYPINDIC=8(1...3)
'EQUATE'PAINT=PAINT(1...3)
'GROUP'FPAINT=RANK(PAINT;FACLEV)
'CALCULATE'PAINT=VARFAC(FPAINT)
: TYPINDIC,PAINT=ORDER(TYPINDIC,PAINT;PAINT)
'GROUPS'FACPAINT=INTPT(TYPINDIC)
'FOR'I=1...3;MEANRANK=MNRANK(1...3)
'RESTRICT'PAINT$FACPAINT=I
'CALCULATE'MEANRANK=MEAN(PAINT)
: KRUSWAL=KRUSWAL+(MEANRANK-EXPMNRNK)**2
'REPEAT'
'CALCULATE'KRUSWAL=(96*KRUSWAL)/(24*24-1)
'PRINT'KRUSWAL$10.2
'RUN'
17 10 19 17 33 20 22 8 'EOD'
13 6 15 9 12 7 5 17 'EOD'
19 17 16 2 35 25 23 21 'EOD'
'CLOSE'
```

The program reads 3 sets of 8 values then merges them by using 'EQUATE', gets the ranks of the merged data in factor FPAINT, and then transfers these values to variate PAINT. This is used to and PAINT itself. FACPAINT is a factor with the same values as TYPINDIC and is used in the 'RESTRICT' within the loop so that the mean rank for each paint-type can be calculated. The Kruskal-Wallis statistic is then easily obtained.

7. REGRESSION AND GENERALIZED LINEAR MODELS

A simple example showing the 'TERMS', 'Y', and 'FIT' directives has already been shown. Other directives are provided for making explicit or conditional changes to the set of x-variates. As well as quantitative variates, factors may be added to the set of x-variates. A factor of N levels implies the addition of N (or N-1) dummy variates. Further dummy variates may be implied for the interactions between factors by using structure formulae for the symbolic description of factorial models (Wilkinson and Rogers [12]). In the following program segment a bivariate regression is fitted to grouped data and then the model is extended to test the difference among intercepts for the groups, and finally to test the differences among regression coefficients for the groups.

```
'REFERENCE' GRPREG
'UNITS'$60
'FACTOR' GRP$5
'READ' GRP,Y,X1,X2
'TERMS' Y+(X1+X2)*GRP
'Y' Y
'FIT/ANDEV=I'X1+X2
'ADD' GRP
'ADD/ANDEV=T'(X1+X2).GRP
```

The ANDEV option initiates and terminates the collection of residual sums of squares of the successive models, and a summary ANOVA is printed from these values.

As well as the usual regression model with normal errors, GENSTAT can fit the class of generalized linear models defined by Nelder and Wedderburn [8]. This includes the log-linear model for contingency table analysis, analysis of binomial data using logits, probits, or the complementary log-log transformation, and analysis of data with gamma errors.

Optimization procedures are provided for other non-linear regression models such as growth curves. Sums of squares (or log likelihood or a user-defined function), are computed for trial values of parameters either as part of a search for a minimum or on a systematic grid which can be displayed.

8. ANALYSIS OF VARIANCE FOR DESIGNED EXPERIMENTS

While the regression directives can provide the analysis of variance for models with fixed effects, the ANOVA directive uses a different algorithm (Payne and Wilkinson [9]) to produce the analysis of variance, tables of means or effects (adjusted where appropriate), and standard errors. It can be used on a wide range of generally balanced designs, latin squares, multi-way classifications with proportional replication, and multi-level split-plot or repeated measures designs. The designs are described by specifying the structure of the error components of the model, and separately specifying the structure of the treatment effects (Nelder [7]). As an illustration, consider a split-plot design. ith apple-trees as main-plots and single fruit as ib-plots. Let treatment factor A be applied to iole trees, while treatment factor B is applied to ndividual fruit. Finally, let the experiment be repeated in several orchards (i.e. blocks). The following GENSTAT directives would describe the design, and analyse the data.

```
'FACTOR'ORCHARD$4 : TREE$10 : APPLE$3
: A$5 : B$3
'READ'ORCHARD,TREE,APPLE,A,B,X
'BLOCK' ORCHARD/TREE/APPLE
'TREAT' A*B
'ANOVA'X
```

The 'BLOCK' directive specifies the factors associated with the random error terms of the model and the relationships betweem them (i.e. nested), while the 'TREAT' directive describes the fixed effects. This description does not need to identify A as the treatment applied to whole trees since the algorithm automatically identifies such aliasing of factors.

The 'ANOVA' directive also provides for estimation of missing values, fitting of covariates, and partitioning of treatment effects into polynomial or specified regression components.

9. MULTIVARIATE ANALYSIS

Many multivariate procedures can be programmed in GENSTAT since 'CALCULATE' provides for the operations of matrix arithmetic (transportation, multiplication, inversion, etc.) and special directives are provided for finding eigenvalues and eigenvectors of a symmetrix matrix ('LRV'), and the singular value decomposition for rectangular matrices ('SVD'). Also the following multivariate procedures are available as single directives

'PCP'	- principal components analysis
'CVA'	- canonical variates analysis
'FACROT'	- rotation of factor loadings to some simple structure
'PCO'	- principal coordinates analysis
'ROTATE'	- procrustes rotation of one set of cooordinates to match another

Other procedures have been written as macros in the GENSTAT language and are available in the macro library (see below).

The multivariate procedures are very straightforward to use as is demonstrated by the program below which is one of the standard examples distributed with GENSTAT. It does a principal components analysis on five measurements on each of nine foods.

```
'REFERENCE' PCPANAL
'UNITS/N' FOODS $ 9=BEEF,CHICKEN,CLAMS,CRABMEAT,
 MACKEREL,SALMON,SARDINES,TUNA,SHRIMP
'SET' VARIATES=ENERGY,PROTEIN,FAT,CALCIUM,IRON
'READ/P'VARIATES
'PRIN/P'FOODS,VARIATES $ 9.1
'MATRIX'PCSCORES $ FOODS,2
'PCP/PRIN=LTSC'VARIATES;SCORES=PCSCORES
'FACTOR' LABELS $ 9,FOODS=1...9
'EQUATE' PCS1,PCS2=PCSCORES $ (1,1X)9,1X
'GRAPH/EQXY=Y,NRF=61' PCS2;PCS1 $ ; LABELS
'RUN'
<data>
```

The nine experimental units are named, and the five variates declared as a set. The data file is read and printed, a matrix is declared to receive the eigenvectors, and then the 'PCP' does the analysis printing the requested information and saving the vectors. A factor is declared so that the plotted points can be identified by number, and the 'EQUATE' picks out the columns of the matrix for plotting by 'GRAPH'.

10. CLUSTER ANALYSIS

Starting with a data matrix where the variates may be quantitative or qualitative a matrix of similarity coefficients may be obtained. These matrices may be extended, re-ordered, and printed in compact form. From this the minimum spanning tree, and dendograms can be obtained by several methods for hierarchical clustering (Gower [6]). Other directives display information about the groups discovered in order to help interpret the clustering. The 'CLASSIFY' directive provides for non-hierarchical classification using a transfer algorithm which optimizes any of four criteria (Banfield and Basil [2]).

11. MACRO LIBRARIES

GENSTAT has a file system for storing and retrieving data structures, and this enables a library of macros to be maintained and used by GENSTAT jobs. A standard macro library is distributed with GENSTAT and the following list shows the macros distributed with version 4.02

CANCOR	- canonical correlations
GENPROC	- generalised procrustes analysis
MULTMISS	- estimate multivariate missing values
GLMODEL	- generalized linear model macro
ORTHPOL	- orthogonal polynomials
ORTHPOLW	- orthogonal polynomials for specified weights
HIERANOV	- hierarchical analysis of variance
CLASSF	- classification using furthest nuclei
MANOVA	- multivariate analysis of variance
FIELLER	- estimation of log dose percentage (e.g. LD50)
CENSOR	- analysis of censored data
ALIAS	- details of aliased model terms from anova
DSQUARE	- discriminant analysis: Mahalanobis squared distances between group means
ALLOCATE	- discriminant analysis: allocating a unit to a group
MISALLOC	- discriminant analysis: assessing numbers of units misallocated using group covariance matrices
MISALLOP	- discriminant analysis: assessing numbers of units misallocated using pooled covariance matrix
JACKNIFE	- discriminant analysis: assessing number of units misallocated using jackknifing
ASYMANAL	- analysis of asymmetric matrices
CVAID	- canonical variate analysis with additional output
D3PLOT	- perspective plotting of three variates
CORRESP	- correspondence analysis (reciprocal averaging)

To indicate the generality of the macro facility, the macros written by GENSTAT users of the CSIRO computer for the macro library are also listed.

BASIC	- calculates a set of basic statistics for each variate
CONTRAST	- tests specified linear combinations of regression lines
DISTRIB	- forms and plots the frequency histogram for a variate and identifies outliers
HOMVAR	- does various tests for homogeneity of variance
LATTICE	- analyses square or rectangular lattice designs with recovery of inter-block information
MAN	- performs various MANOVA significance tests for given SSP matrices
PARALLEL	- tests homogeneity of regression equations across groups of data
POWER	- uses Box-Cox method to find power transformation
QQNORM	- calculates predicted normal quantiles
RYL	- for residuals plots scatter, histogram and normal plot
SQLAT	- square lattice designs for s*s treatments and r replicates
TESTFIT	- for a regression, observations with identical x-values are located and the residual is partitioned into 'lack of fit' and 'within group error'

12. CONCLUSION

While the capabilities of GENSTAT overlap those of other statistical packages, there are areas where it has real advantages. Some of these are

1. The powerful programming language
2. The ability to extract values computed in one procedure as data structures for input to other procedures
3. The extension of the regression procedures to include the class of generalized linear models
4. The excellent facilities for analysing designed experiments
5. The ability to expand the available procedures by writing macros, which has been well demonstrated in the area of multivariate analysis
6. The operations defined for tables and matrices.

13. REFERENCES

[1] ALVEY, N.G., BANFIELD, C.F., BAXTER, R.I., GOWER, J.C., KRZANOWSKI, W.J., LANE, P.W., LEECH, P.K., NELDER, J.A., PAYNE, R.W., PHELPS, K.M., ROGERS, C.E., ROSS, G.J.S., SIMPSON, H.R., TODD, A.D., WEDDERBURN, R.M.W. and WILKINSON, G.N. (1977). GENSTAT a general statistical program. Rothamsted Experimental Station.

[2] BANFIELD, C.F. and BASIL, L.C. (1977). A transfer algorithm for non-heirarchical classification. Appl. Statist., 26, Algorithm AS113.

[3] BERNARD, G. (1979). A comparison of three statistical packages: GENSTAT, BMDP, and SPSS. COMPSTAT 1978, Physica Verlag, Vienna.

[4] DRAPER, N.R. and SMITH, H. (1966). Applied regression analysis. John Wiley & Son, New York.

[5] FEDERER. W.T. and HENDERSON, H.V. (1979). Covariance analysis of designed experiments X statistical packages: an update. Biometrics Unit, Cornell University.

[6] GOWER, J.C. (1967). A comparison of some methods of cluster analysis. Biometrics, 23, 626-637.

[7] NELDER, J.A. (1965). The analysis of randomized experiments with orthogonal block structure. (parts I and II.) Proc. Roy. Statist. Soc. A, 135, 370-384.

[8] NELDER, J.A. and WEDDERBURN, R.M.W. (1972). Generalized linear models. J. Roy. Statist. Soc. A, 175, 370-384.

[9] PAYNE, R.W. and NELDER, J.A. (1977). Data structures in statistical computing. Proc. of the Ninth Inter. Biometric Conf., II, 191-207.

[10] PAYNE, R.W. and WILKINSON, G.N. (1977). A general algorithm for analysis of variance. Appl. Statist., 26, 251-260.

[11] SEARLE, S.R., HENDERSON, H.V. and FEDERER, W.T. (1978). Annotated output from computer packages that calculate linear models analysis of unbalanced data. Biometrics Unit, Cornell University.

[12] WILKINSON, G.N. and ROGERS, C.E. (1973). Symbolic description of factorial models for analysis of variance. Appl. Statist., 22, 392-399.