Symbolic Description of Factorial Models for Analysis of Variance

By G. N. WILKINSON and C. E. ROGERS

Rothamsted Experimental Station

SUMMARY

The paper describes the symbolic notation and syntax for specifying factorial models for analysis of variance in the control language of the GENSTAT statistical program system at Rothamsted. The notation generalizes that of Nelder (1965). Algorithm AS 65 (Rogers, 1973) converts factorial model formulae in this notation to a list of model terms represented as binary integers.

A further extension of the syntax is discussed for specifying models generally (including non-linear forms).

1. INTRODUCTION

GENERAL computer programs for analysing experiments need a concise, flexible notation for specifying the appropriate factorial models. The notation in this paper, and various others due to Zyskind (1962), Hemmerle (1964), Nelder (1965), Fowlkes (1969) and Claringbold (1969), were discussed at an international workshop meeting on the computational aspects of analysis of variance at the University of Wisconsin in 1970 (Muller and Wilkinson, 1970).

The present notation for model formulae includes the *addition*, *crossing* and *nesting* operators common to most of the notations mentioned, a *dot* operator for defining multi-factor model terms and *deletion* operators for eliminating unwanted terms from otherwise simple formulae. *Submodel functions* may be substituted for factors in a formula, to specify regression sub-models for partitioning factorial effects.

The notation is implemented in the GENSTAT language (Nelder *et al.*, 1973) (which also includes a special pseudo-factor operator not described here). The GENSTAT system is currently in operation at Rothamsted, the Edinburgh Regional Computing Centre, Cambridge and Bristol Universities and other centres. Algorithm AS 65 (Rogers, 1973) converts symbolic factorial model formulae to a list of model terms represented as binary integers.

Further extensions of the notation are readily envisaged, e.g. a *diallel* function of parental genotype factors and a *similarity-link* operator for combining random terms with a common variance, such as rows and columns in a lattice square design. A general extension of notation to include linear or non-linear regression models is described in Section 4.

2. OUTLINE OF THE NOTATION

2.1. *Simple Factorial Models*

Factorial models can be expressed symbolically as a sum of model terms, using the operator +, and a *dot* operator to link factor names in multifactor terms. Thus the following two alternative models for a two-way table of observations indexed by factors A and B,

$$\left.\begin{aligned}&\text{(i)}\quad Y_{ij} = m+a_i+b_j+(ab)_{ij},\\&\text{(ii)}\quad Y_{ij} = m+a_i+(ab)_{ij},\end{aligned}\right\} \tag{1}$$

where $i = 1, 2, \ldots, p$ and $j = 1, 2, \ldots, q$, can be written symbolically with a grand mean term taken as implicit,

$$\left.\begin{aligned}&\text{(i)}\quad A+B+A\cdot B,\\&\text{(ii)}\quad A+A\cdot B.\end{aligned}\right\} \tag{2}$$

Formulae of this form are termed *simple factorial model formulae.*

Readers unfamiliar with computing languages should note that there is an essential modularization of information here. Formulae such as (2) specify only the factorial structure of the model. Other information needed for analysis, such as the numbers of levels of the factors and the positions of the observations in the $A \times B$ table, would be specified by other statements in the programming language.

Note also that the meaning of the term $A\cdot B$ in the models (2) is affected by the marginal terms that precede it, and a modern stepwise fitting algorithm (cf. Wilkinson, 1970) will *automatically* generate constraints in the parameter estimates (effects) corresponding to marginal aliasing, such as that of $\overline{(ab)}_{i\cdot}$ with a_i, etc. Thus $A\cdot B$ automatically represents, in formula (2) (i), interaction effects with zero row and column sums, and in (2) (ii), within-class deviations with respect to A.

2.2. *Crossing and Nesting Operators*

Simple factorial models can be usually expressed more concisely by using the *crossing* and *nesting* operators (Nelder, 1965), represented here by * and / respectively, to indicate a multi-way table with or without certain margins. The following expansions to simple form show the meaning of these operators:

$$\left.\begin{aligned}&\text{(i)}\quad A*B = A+B+A\cdot B,\\&\text{(ii)}\quad A/B\; = A+A\cdot B.\end{aligned}\right\} \tag{3}$$

The nesting operator suppresses the marginal term B which is irrelevant in the hierarchical model.

2.3. *Block and Treatment Formulae*

It is statistically necessary, of course, to distinguish between the random (or error) terms in the model, arising from the physical (block) structure of the experimental units, and the systematic (treatment) terms. This is done in the GENSTAT language with separate declarations of what are termed *block* and *treatment* model formulae, for instance

'*blocks*' *blocks/plots*, (4)

'*treatments*' *nitrate**density.* (5)

2.4. *Crossed and Nested Formulae*

With the introduction of bracketing where necessary, crossed and nested formulae suffice to specify the usual models for most experimental designs. For instance, Latin squares, Youden squares, lattice squares and plaid designs have a block structure

$$\mathit{rows}*\mathit{cols} \tag{6}$$

$$\text{or}\quad \mathit{reps}/(\mathit{rows}*\mathit{cols}) \tag{7}$$

while split-plot or split-row and split-column designs have block structures such as

$$\mathit{blocks}/\mathit{mainplots}/\mathit{subplots} \tag{8}$$

$$\text{or}\quad \mathit{reps}/(\mathit{rows}*\mathit{cols})/\mathit{subplots} \tag{9}$$

$$\text{or}\quad (\mathit{rows}/\mathit{subrows})*(\mathit{cols}/\mathit{subcols}). \tag{10}$$

Treatment model structures are usually of the crossed form

$$\text{e.g.}\quad \mathit{nitrate}*\mathit{phos}*\mathit{potash}$$

but nested structures (hierarchical models) are sometimes needed,

$$\text{e.g.}\quad \mathit{group}/\mathit{variety} \tag{11}$$

$$\text{or}\quad \mathit{spray}/(\mathit{type}*\mathit{dose}), \tag{12}$$

where *spray* is a factor indicating whether experimental plots were sprayed or not (with insecticide, say). Note that in this example the factors *type* and *dose* would include *null* levels associated with the unsprayed plots.

The general rules for determining simple factorial formulae from formulae such as (7)–(12) are given in Section 3.

2.5. *Deletion Operators*

Corresponding to the operators

$$+ \quad * \quad /$$

are three deletion operators with meanings as follows:

Operator

(i) $-$ Delete the specified term(s) from the preceding model.
(ii) $-*$ As for (i), and also any corresponding higher-order terms.
(iii) $-/$ Delete *only* the corresponding higher-order terms.

These are useful for deleting unwanted terms from crossed and nested formulae when the corresponding simple factorial sum of terms would be otherwise too lengthy. The following equivalent model expressions illustrate their meaning (see rules in Section 3).

$$A*B*C-A\cdot B\cdot C = A+B+C+A\cdot B+A\cdot C+B\cdot C, \tag{13}$$

$$A*B*C-*B\cdot C = A*(B+C), \tag{14}$$

$$A*B*C-/A = A+B*C. \tag{15}$$

The ANOVA directive of the GENSTAT language also provides a model-order contro parameter for suppressing, from the analysis, all treatment terms above the order specified.

2.6. *Submodels*

An important requirement for the analysis of variance of factorial models is the ability to specify submodels for partitioning factorial effects into regression components; *linear*, *quadratic* and *cubic* trends of main effects for instance, and interactions of these such as *linear* $(A)\times$ *linear* (B), *linear* $(A)\times$ *quadratic* (B), etc., where A and B are different factors.

It is usually sufficient in practice to specify submodels only for the main effects of each factor, since these then define by implication the corresponding compound submodels for higher-order factorial terms. Submodels are specified in the GENSTAT language by substituting for factors in the treatment model formula the appropriate submodel functions of them, which are of the form

$$\textit{submodel-function}\,(\textit{factor},\ \textit{order}\ [,X]), \tag{16}$$

the square brackets indicating an optional third parameter.

The functions currently available are *POL* (polynomial regression), *REG* (multiple regression), *POLND* and *REGND*, where adding the affix *ND* (no deviations) indicates that a deviations term is not to be considered a part of the submodel for that factor when generating compound submodels for higher-order terms, and leads to suppression of terms like *deviations* $(A)\times$ *linear* (B) in the compound submodel for $A\times B$ interactions, say.

The *order* parameter indicates the order of polynomial required or the number of x-variates of a multiple regression.

The X parameter, if present, specifies an x-variate for polynomial regression or a matrix whose rows are the x-variates of a multiple regression. If omitted (in the case of *POL*, *POLND*) a polynomial regression on the quantitative levels of the factor is implied.

Example. The treatment formula

$$\textit{GENOTYPE*POL(SITE,}1\textit{,SITEMEAN)*POLND(DENSITY,}2) \tag{17}$$

would produce the type of *genotype* × *site* analysis described by Finlay and Wilkinson (1963), in which the sensitivity of each genotype with respect to site is characterized by a linear regression of yield values for that genotype on the site means (over all genotypes); together with an extension of the analysis for linear and quadratic trends of yield on *density* (sowing rate) and their interactions with *genotype* and *site*. If a linear submodel were also specified for *genotype*, as for *site*, single degrees of freedom for non-additivity would be produced.

3. General Syntax and Interpretation

3.1. *Syntax*

A factorial model formula in the GENSTAT language is an expression, interpreted from left to right, with *factors* as the basic operands, bracketing where needed and the following *dyadic operators* with precedence values (1 = highest) as indicated:

$$\left.\begin{array}{lccccccc} \textit{Operator} & \cdot & / & * & + & - & -/ & -* \\ \textit{Precedence} & 1 & 2 & 3 & 4 & 4 & 4 & 4 \end{array}\right\} \tag{18}$$

(We omit from consideration here the substitution of submodels for factors, which does not affect the primary, factorial model.)

The assignment of operator-precedence reduces the need for bracketing to define the left and right operands of each operator in model expressions. Compare the familiar assignment of precedence to the operators × and / over + and − in arithmetic expressions. The brackets in the expression $(a+b)\times c$ are needed to define the left-hand operand of ×, whereas in the expression $a+b\times c$ the left-hand operand of × is unambiguously defined as b by the operator precedence, so that bracketing as in $a+(b\times c)$ is unnecessary.

The *left-to-right* rule also reduces the amount of bracketing when successive operators have the same precedence. Thus $a/b/c$ is unambiguously interpreted as $(a/b)/c$.

3.2. *Evaluation Rules*

The interpretation of a model formula follows from its expansion as a simple factorial model, i.e. its evaluation as a sum of factorial terms, a term being either a factor or a *dot*-product of factors. In the rules given below, A and B denote model terms, S and T sums of model terms and L and M model formulae:

Simplification.

Delete repetitions of operands in a *dot*-product, (19)

e.g. $A\cdot B\cdot A = A\cdot B$.

Delete repetitions of model terms in a sum, (20)

e.g. $A+B+A = A+B$.

Ordering of model terms. Some of the evaluation rules below may not produce a statistically appropriate ordering of model terms, so that re-ordering may be required. An essential order requirement is that any term in a simple factorial model should precede all terms to which it is marginal, i.e. A before $A\cdot B$, etc. A stronger requirement (implemented in GENSTAT) is that terms be arranged in increasing order with respect to the number of factors in a term, with terms of the same order arranged in a natural sequence with respect to the factors defining them.

Distributive rule for dot-product

$$S\cdot T = \sum A\cdot B \quad \text{for all terms } A \text{ in } S,\ B \text{ in } T. \tag{21}$$

For example,

$$(A+B)\cdot C = A\cdot C+B\cdot C.$$

General definitions of crossing and nesting operations. These may be defined in terms of + and *dot* operations as follows:

$$\text{(i)}\quad L*M = L+M+L\cdot M, \tag{22}$$

$$\text{(ii)}\quad L/M = L+FAC(L)\cdot M, \tag{23}$$

where $FAC(L)$ is the *dot*-product of all factors in L. It will usually be a term in the expansion of L. For example,

$$\begin{aligned}(A+B)*C &= A+B+C+(A+B)\cdot C\\ &= A+B+C+A\cdot C+B\cdot C,\\ (A*B)/C &= A+B+A\cdot B+A\cdot B\cdot C,\\ (A+B)/C &= A+B+A\cdot B\cdot C.\end{aligned}$$

Deletion operations

$S-T$: Delete from S any terms in T, (24)

$S-/T$: Delete from S any terms to which a term in T is marginal, (25)

$$S-{*}T = S-T-/T. \tag{26}$$

4. Extension to General Models

A set of n observations may be regarded as a point in a n-dimensional vector space, the sample space. A model for the observations specifies a subspace, not necessarily linear, of the sample space, and a parameterization of this subspace.

4.1. *Linear Models*

For the linear factorial models considered so far the model subspace is determined by the incidence variates associated with the factorial parameters. The symbolic notation so far described characterizes the relevant subspace without explicit naming of the parameters, which are assumed to be in 1–1 correspondence with the incidence vectors.

The notation can be extended to describe any linear subspace by admitting as operands not only factors but also x-variates, or more generally matrices whose columns define a set of x-variates, and by extending the definition of *dot*-product.

Definition of dot-product. If *X1*, *X2* denote $n \times p$, $n \times q$ matrices (incidence matrices if *X1* and/or *X2* are factor names), the symbolic *dot*-product $X1 \cdot X2$ represents an $n \times pq$ matrix, each row of which is the direct product of the corresponding rows of *X1*, *X2* respectively, i.e. comprises all products $x_1 x_2$ of elements x_1, x_2 from the respective rows.

Note that the simplification rule (19) for repeated operands in a *dot*-product applies only to factors (or matrices X with the property that X and $X \cdot X$ determine the same subspace).

Examples. (i) With *X1* and *X2* defined as above, the symbolic model

$$X1{*}X2 = X1 + X2 + X1 \cdot X2 \tag{27}$$

indicates the linear model subspace determined by the $p+q+pq$ variates associated with *X1*, *X2* and $X1 \cdot X2$.

(ii) Introducing a symbolic exponentiation operator $**$, a complete second-degree model with respect to *X1* and *X2* is concisely described as

$$\begin{aligned}(X1+X2){*}{*}2 &= (X1+X2){*}(X1+X2)\\ &= X1+X2+X1\cdot X1+X1\cdot X2+X2\cdot X2,\end{aligned} \tag{28}$$

deleting repeated terms.

4.2. *Non-linear Models*

In the context of a programming language an important feature of the symbolic notation for linear models is that explicit naming of the parameters associated with the model vectors is avoided.

A non-linear model, however, requires an algebraic specification which necessarily involves explicit names for parameters, as in the asymptotic regression formula

$$MAXVAL*\{1-exp(-RATE*X)\} \tag{29}$$

in which *MAXVAL* and *RATE* are the parameters, *X* is a column vector of known *x*-values and * here denotes item-by-item multiplication. This raises two points:

(1) *Declaration of parameters.* A distinction must be made between symbols in an algebraic expression that represent parameters and those that represent variables with known values. This can be done for instance with declarations such as

$$'PARAMETERS' \quad MAXVAL,RATE. \tag{30}$$

(2) *Modes of expression.* Since the algebraic and symbolic modes of expression involve the same operator symbols +, −, *, / but with different meanings, an unambiguous indication of which mode of expression is being used in particular contexts is required. Thus, a directive '*MODEL*' might carry a modifier to indicate mode:

$$\left.\begin{array}{ll} & 'MODEL/A' \quad \text{algebraic expression} \\ \text{or} & \\ & 'MODEL/S' \quad \text{symbolic expression (linear model only).} \end{array}\right\} \tag{31}$$

Mixed mode expressions may sometimes be necessary. For instance, the parameters *MAXVAL* and *RATE* in (29) might depend on certain factorial treatments with symbolic model *A*B*. This can be effected by introducing a special function

$$LM(\text{symbolic expression, parameter name}), \tag{32}$$

where *LM* stands for linear model. The symbolic argument defines a set of *x*-variates and the second argument is a name for identifying the corresponding array of parameters. The functional notation enables symbolic expressions to be introduced into otherwise algebraic expressions. Thus, (29) could be modified to

$$LM(A*B,MAXVAL)*\{1-exp(-LM(A*B,RATE)*X)\}. \tag{33}$$

Mixing of modes as in (33) can be avoided by allowing models to be named and used in specifying other models, or by extending the definition of parameters. For instance if *MAXVAL* and *RATE* are defined to be linear models with the declaration

$$'MODEL/S' \quad MAXVAL=A*B : RATE=A*B \tag{34}$$

or if, alternatively, the parameter definition (30) is extended to include the appropriate symbolic models, e.g.

$$'PARAMETERS' \quad MAXVAL,RATE \ \$ \ A*B, \tag{35}$$

the simpler formula (29) can be used in place of (33).

5. Acknowledgement

The authors wish to thank a referee for valuable suggestions which substantially improved the presentation.

References

Claringbold, P. (1969). An approach to conversational statistics. In *Statistical Computation* (R. C. Milton and J. A. Nelder, eds.) pp. 267–283. New York: Academic Press.

Finlay, K. W. and Wilkinson, G. N. (1963). The analysis of adaptation in a plant-breeding programme. *Aust. J. Agric. Res.*, 14, 742–754.

FOWLKES, E. B. (1969). Some operators for ANOVA calculations. *Technometrics*, **11**, 511–526.

HEMMERLE, W. J. (1964). Algebraic specifications of statistical models for analysis of variance computations. *J. Ass. Comput. Mach.*, **11**, 234–239.

MULLER, M. E. and WILKINSON, G. N. (1970). Statistical algorithms and computational aspects of the analysis of variance. *Report on* ANOVA *Workshop*, 1970, University of Wisconsin.

NELDER, J. A. (1965). The analysis of randomized experiments. I. Block structure and the null analysis of variance. II. Treatment structure and the general analysis of variance. *Proc. Roy. Soc.* A, **283**, 147–178.

NELDER, J. A. *et al.* (1973). GENSTAT *Reference Manual*, Rothamsted Experimental Station.

ROGERS, C. E. (1973). Algorithm AS 65. Interpreting structure formulae. *Appl. Statist.*, **22**, 414–424.

WILKINSON, G. N. (1969). Facilities in a statistical program system for analysis of multiple-indexed data. In *Statistical Computation* (R. C. Milton and J. A. Nelder, eds.), pp. 201–228. New York: Academic Press.

WILKINSON, G. N. (1970). A general recursive procedure for analysis of variance. *Biometrika*, **57**, 19–46.

ZYSKIND, G. (1962). On structure, relation, sigma and expectation of mean squares. *Sankhyā*, A, **24**, 115–148.