

RESTRICTED ESTIMATION IN UNBALANCED FACTORIAL MODELS: AN APL PROGRAMS PACKAGE

JOSÉ M. PRIETO and JOSÉ M. CARIDAD

University of Córdoba

This paper describes a set of programs that provide researchers with restricted effect estimations in unbalanced factorial models when several weighing systems are imposed upon those models. The main program performing such an analysis is known as REUFM, and is written in APL®PLUS, for IBM/PC microcomputers. An example is given in order to illustrate the programs.

Keywords: Factorial designs, unbalanced models, surveys and experiments, APL.

1. INTRODUCTION

In many statistical studies, data can be displayed according to several factors in a complete factorial classification of treatments. With unbalanced data it is necessary to distinguish between the complete case and that in which some cells have no data, since algorithms valid for the former do not apply universally to the latter (/1/, p. 40). The case of missing cells should be made the subject of a separate study.

Let us consider initially a two-way design; the associated fixed-effect linear model in usual notation is written

$$(1) \quad E(y_{ijr}) = \mu + \alpha_i + \beta_j + \alpha\beta_{ij} \quad (i = 1, \dots, a; j = 1, \dots, b; r = 1, \dots, n_{ij})$$

—José M. Prieto - Universitat de Córdoba - E.T.S.I.A. - Dep. Matemàtica Aplicada - Aptat. 3048-14004 Córdoba.

—José M. Caridad - Universitat de Córdoba - E.T.S.I.A. - Dep. Estadística - Aptat. 3048-14080 Córdoba.

—Article rebut al febrer del 1988.

assuming uncorrelated and independently-distributed normal errors with common variance σ^2 . We are concerned with least-square estimations of the different factorial effects involved and their standard deviations. Although it is common to deal with hypothesis-testing as a related subject, in many anova situations it is the estimation of the effects *per se* that is desired (/2/, p. 67). Our study is based on this approach.

Since the above model is overparameterised, identifiability “constraints” are usually imposed on main and interaction effects, the form of the constraints depending on the “weights” assigned to the cells of the classification. Three well-known weighing systems will be considered:

$$(2) \quad \begin{aligned} \sum_i n_i \alpha_i = 0, \sum_j n_j \beta_j = 0, \sum_j n_{ij} \alpha \beta_{ij} = 0 (i = 1, \dots, a), \\ \sum_i n_{ij} \alpha \beta_{ij} = 0 (j = 1, \dots, b) \end{aligned}$$

$$(3) \quad \begin{aligned} \sum_i n_i \alpha_i = 0, \sum_j n_j \beta_j = 0, \sum_j n_j \alpha \beta_{ij} = 0 (i = 1, \dots, a), \\ \sum_i n_i \alpha \beta_{ij} = 0 (j = 1, \dots, b) \end{aligned}$$

$$(4) \quad \begin{aligned} \sum_i \alpha_i = 0, \sum_j \beta_j = 0, \sum_j \alpha \beta_{ij} = 0 (i = 1, \dots, a), \\ \sum_i \alpha \beta_{ij} = 0 (j = 1, \dots, b) \end{aligned}$$

The choice of system to be imposed on model parameters has been a matter of some controversy /3/.

In investigations in which sample sizes are known to be proportional to subpopulation sizes, we use the first weighing system (2), or “frequency” weights. This choice of weights is ideal in sample surveys. When the cells arise from “experimental” subpopulation through a given survey technique, as is the case of those studies which make a sample survey of the result of an “extensive” experiment, the choice of that weighing system may still be meaningful /4/.

It is common practice in experimentation to call initially for a balanced design, the normal thing being to use the system of weights given in (4), called “usual” weights. However, unequal subclass numbers inevitably appear in the completed experiment; if, as is to be expected, the lack of balance is unimportant, the analysis of such an unbalanced design with usual weights produces appropriate effect estimates /5/.

Other experiments are initially planned with a “proportional” design /6/, the weights normally associated being /7/ those given in (3), which we have termed “marginal” weights. In actual practice, however, a slight disproportionality is obtained; in this case, an appropriate analysis can be carried out using marginal weights /8/. On the other hand, in terms of the additional notation:

$$(5) \quad n_{i.j}^{\approx} = 1/n(n_{i.}n_{.j}) \quad , \quad n_{i.}^{\approx} = n_{i.} \quad , \quad n_{.j}^{\approx} = n_{.j}$$

the following weighing system may be established:

$$(6) \quad \sum_i n_{i.}^{\approx} \alpha_i = 0 \quad , \quad \sum_j n_{.j}^{\approx} \beta_j = 0 \quad , \quad \sum_j n_{i.j}^{\approx} \alpha \beta_{ij} = 0 (i = 1, \dots, a),$$

$$\sum_i n_{i.j}^{\approx} \alpha \beta_{ij} = 0 (j = 1, \dots, b)$$

If we introduce (5) in (6), it turns out that (6) is reduced to (3), in other words, (3) and (6) are similar expressions of the marginal weighing system; we shall use the latter.

A hypothetical classification having the numbers $n_{i.j}^{\approx}$ as subclass numbers would be proportional, and would have the same first-order marginal frequencies as the original one. This is why we have termed such numbers “expected” proportional frequencies.

2. COMPUTING MATRICES

2.1. EFFECT MATRIX

The matrix expression of (1) for a k -factorial model is $E(Y) = LX\beta$. Y reflects the vector of observations in the various subclasses, if these have been lexicographically ordered previously. β denotes the vector of factorial effects. The matrix product LX is the “design” matrix, where L is the “replications” matrix that accounts for the different number of observations in the cells, and where X is the “effect” matrix for β ; both L and X are matrices of incidence. A logical partitioning of β is assumed to exist such that each group corresponds to a meaningful subset of factorial effects due to the same treatment combination. Each effect group might be designated by a tuple whose elements were the factors indexes, the general mean being designated by the tuple (0), the i -th main effect ($i = 1, \dots, k$) by the tuple (i), and so on. There are, in all, 2^k tuples, which are arranged as follows: (0), (1), ..., (k), (1, 2), ..., ($k - 1, k$)... Thus, we have established a “complete” lexicographical order of all the tuples.

The system of “normal” equations in matrix terms of the above model may be written as $X'DX\hat{\beta} = X'D\bar{Y}$, where $D(= L'L)$ is a diagonal matrix with cell sizes along the main diagonal, $\hat{\beta}$ is the vector of the least squares estimators, and \bar{Y} is the vector containing the cell means. The coefficient matrix of the above system $X'DX$ is called the “information” matrix. The homogenous system of weights with regard to the effect estimators in matrix terms is $R\hat{\beta} = 0$ where the coefficient matrix R will be designated “restriction” matrix. From the above two systems, it is known that there is a unique solution for $\hat{\beta}$:

$$(7) \quad \hat{\beta} = [X'DX + R'R]^{-1}X'D\bar{Y}$$

The covariance matrix of the estimators is:

$$(8) \quad \text{Cov}(\hat{\beta}) = \hat{\sigma}^2[X'DX + R'R]^{-1}X'DX[X'DX + R'R]^{-1}$$

where an estimator of the common variance in the cells may be written:

$$(9) \quad \hat{\sigma}^2 = \frac{1}{(n-s)} [Y'Y - [X'D\bar{Y}]'\hat{\beta}]$$

and where n and s are the total number of observations and cells, respectively.

The effect matrix is structured in 2^k lexicographically ordered column-blocks, each one associated to an effects group of β . It is known that the structure of each column-block of X can be expressed in terms of the Kronecker matrix product of identity matrices and column vectors whose elements are ones /9/. Consequently, the information matrix may likewise be structured in $2^k \times 2^k$ submatrices.

2.2. RESTRICTION MATRIX

To facilitate restriction matrix structuration we have considered in (2) all the restrictions, including the “redundant” ones. An essential feature of the structure of R is that it is formed by “certain” submatrices of the information matrix $X'DX$. To see this, it is necessary to define an auxiliary matrix in such a way that if a submatrix of $X'DX$ is an integrant part of R such a submatrix is substituted by one, or otherwise by zero. The $2^k \times 2^k$ incidence matrix so formed, which refer to as the “restriction incidence” matrix, denoted by E , has the following structure /8/: A column of E , e.g., that associated to a group of effects denoted by a certain m -tuple (or tuple of m elements) contains exactly

m numbers equal to 1, which occurs in those rows of E designated by all the possible $(m - 1)$ -tuples whose elements belong to that m -tuple.

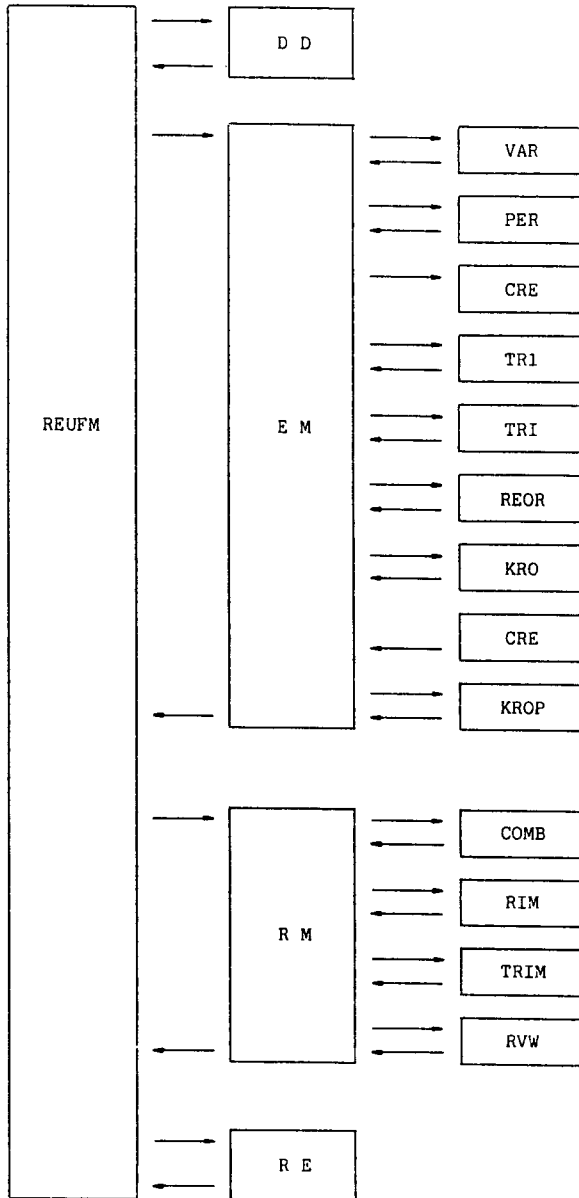
Once E is constructed and in order to obtain R , a matrix of diagonal-blocks, each number of E occupying a given position is replaced by the submatrix of $X'DX$ situated in the same "position", although multiplied by that number; this transformed E matrix is denoted by T . Finally, if in each column-block of T the null submatrices are omitted, the remainder forms the corresponding diagonal-block of R .

In the above process, if the matrix $X'DX$ is used, we obtain the restriction matrix for frequency weights. If we use the matrix $X'HX$, where H is a diagonal matrix whose entries are the generalized n_{ij}^{\sim} (5), the result is the restriction matrix form marginal weights. The restriction matrix for usual weights is easily obtained replacing each non-null number of the restriction matrix for frequency weights by an "1".

3. COMPUTER PROGRAMS

3.1. SYNTAX DIAGRAM

This paper presents a programs package consisting of a main program called REUFM (Restricted Estimation in Unbalanced Factorial Models), the only one that must be called up by the user, together with a set of auxiliary programs and subprograms. The following diagram illustrates the dependency hierarchy between them.



These programs are written in APL \otimes PLUS for IBM/PC and compatible computers. Their respective codes* are given in the Appendix.

*Availability: A copy of source code is available for IBM/PC format disk, by sending a floppy disk to the authors.

3.2. PROGRAM STRUCTURE

The following is a brief description of the functions, to help the user to understand what they do.

The input program is Design Data, DD. It accepts the number of factors k , the number of levels each factor, the subclass frequencies, and the observation values. DD produces as output SY, a vector whose elements are the sum of the observation values for each subclass.

The next program, EM, calculates the effect matrix, X , and other related results. To begin, the subprogram VAR generates a $2^k \times k$ incidence matrix A , for a given row of which, a "0" in a certain position denotes that the term occupying that position in the Kronecker product (referred to at the end of 2.1) is an identity matrix, whereas a "1" will denote a vector of ones. It is necessary to re-order the rows of A in such a way that the result matrix F has rows which naturally correspond to the lexicographically-ordered column-blocks of X . Such a matrix F is produced by PER.

The column-blocks of the effect matrix of the same position will correspond to F matrix z -th row ($z = 1, \dots, 2^k$), in the following way: first, the subprogram CRE generates a set of empty variates $I1, \dots, Ik, U1, \dots, Uk$. Then, if the t -th position in the row ($t = 1, \dots, k$) is occupied by a "1" the subprogram TR1 assigns to the variate I_t a column vector of a_t "1" s (where a_t is the level of the a_t -th factor); and if the t' -th position ($t' = 1, \dots, k; t' \neq t$) is occupied by a zero, the subprogram TRI assigns to $U_{t'}$ the identity matrix of order $a_{t'}$. The vectors of "1" s and identity matrices thus formed are assigned to the ordered variates TOT1, ..., TOT k by means of the subprogram REOR. The Kronecker matrix product of these k variates is carried out by KRO, the output being BXz , the z -th column-block of X . The different column-blocks are linked along the [2]-th dimension in line [23] of EM. The X matrix is calculated the last time the loop is run.

The diagonal matrix D is constructed in line [27] of EM, and the vector \bar{Y} in line [28]. Once X and D are constructed, the information matrix $X'DX$ is also constructed, a given submatrix of which, e.g., that situated in the z_1 -th row-block and in the z_2 -th column-block, is given by $(BXz_1)D(BXz_2)$. To such a submatrix we assign the (z_1, z_2) -th position.

The diagonal matrix H is formed in line [30] of EM although previously we construct an auxiliary matrix QQ equal to the ordered Kronecker product of the submatrices of $X'DX$ situated in the positions $(2, 2)$ -th to $(k+1, k+1)$ -th. The program producing QQ is KROP.

The next program RM constructs the restriction matrix R for the different weights used. The subprogram COMB forms a vector of order 2^k whose elements are the tuples denoting the positions of the column-blocks of X or R ;

that vector VF is produced in line [29] of COMB. The subprogram RIM produces as output the restriction incidence matrix using as input VF . A $2^k \times 2^k$ matrix of zeros is initially formed and then "1"s are assigned to the positions determined by the criterion mentioned at (2.2). This is done in line [4] of RIM for the columns $2 - th$ to $(k + 1) - th$ of E , and in line [20] for the remaining columns.

The transformed restriction incidence matrix T , also referred to at (2.2), is produced by the subprogram TRIM. For a determined row of E , each number is substituted by the corresponding submatrix in line [10], the set of which is linked by the second dimension in line [14]. The resulting row-blocks, TET, are linked by the first dimension in line [17], thus forming the T matrix. The output of TRIM is the matrix T for frequency weights, TFW, if those weights are used (as in line [4]), whereas the output is the matrix T for marginal weights, TMW, if these are the weights considered (as in line [25]).

The subprogram RVW forms the restriction matrix for the various weighing systems. The R diagonal-blocks DBR are formed with the non-null rows of the corresponding column-blocks of T in line [23]. The various DBR are diagonally linked in line [44], and the resulting matrix is augmented with a first null column in line [51], the output being R . If the matrix TFW is initially used, we obtain the R matrix for frequency weights, RFW. If the matrix TMW is used (in line [53]) the output is the R matrix for marginal weights, RMW. Finally, the R matrix for usual weights, RUW, is easily deduced from RFW in line [57].

The last program RE, using the outputs of the above, gives the restricted estimators EFW, EMW, and EUW, corresponding to the three weighing systems in lines [8-10]. An estimation of error variance is given in line [11], and the standard deviations of the above estimations in lines [15-17] of RE.

4. REFERENCES

- [1] Searle, S.R. "Quirks in Linear Model Calculations with Unbalanced Data". Proceeding of the SAS Users Group International Conference, 1981, 6, pp. 38-45.
- [2] Fisher, LL. and McDonald, J. "Fixed Effects Analysis of Variance". Academic Press, New York, 1978.
- [3] Arnold, S.F. "The Theory of Linear Models and Multivariate Analysis". Wiley, New York, 1981.
- [4] Kendall, M.G. and Stuart, A. "The Advanced Theory of Statistics". Vol 3: Design and Analysis, and Time Series, Griffin, London, 1976.
- [5] Bargmann, R.E. "Computational Aids for the Analysis of Balanced and Unbalanced Multiple Factor-Factorial Designs". Proceeding of the

Statistical Computing Section, American Statistical Association, 1975, pp. 128-133.

- [6] **Bock, R.D.** "Multivariate Statistical Methods in Behavioral Research". McGraw-Hill Book Co., New York, 1975.
- [7] **Scheffe, H.** "The Analysis of Variance". Willey, New York, 1959.
- [8] **Prieto, J.M. and Caridad, J.M.** "Métodos Algebraicos en el Análisis de Modelos Factoriales Desequilibrados". Ph. D. Tesis, Tx: Authors, Córdoba, 1986.
- [9] **Mikos, H.** "Explicit Forms of Block Matrices in Unbalanced Cross-Classification". Zastosowania Matematyki, V. 16, n, 3, 1979, pp 459-463.

APPENDIX A. PROGRAM LISTING.

```

      ▽REUFM[0]▽
[0] REUFM
[1] A The Program Package REUFM computes the Restricted Estimations for Unbal-
      anced Factorial Models.
[2] DD
[3] EM
[4] RM
[5] RE

      ▽DD[0]▽
[0] DD;SS;I
[1] A Function DD accepts as inputs the Design Data.
[2] 'Introduce the numbers of factors'
[3] K←0
[4] 'Introduce the numbers of levels of the factors'
[5] FL←0
[6] 'Introduce the numbers of subclasses'
[7] SN←0
[8] 'Introduce the observation values'
[9] Y←((ρY),1)ρi←0
[10] SY←((ρSN),1)ρ0
[11] SS←0
[12] I←1
[13] L1:=S:[I;]←+;[SS+υSN[I];]
[14] SS←SS+SN[I]
[15] ←((I+1)ρSN)/L2
[16] →L1
[17] L2:

      ▽EM[0]▽
[0] EM;T;Z;A;F;C1;C1;SB;Q;QQ
[1] A This function computes the Effect Matrix,X.EM also computes the diagonal
      matrix D whose entries are the subclass numbers,the diagonal matrix H
      whose entries are the expected proportional frequencies,and the vector Y//
      whose elements are the subclass means.
[2] X←((1+ρSN),0)ρ0
[3] VGR
[4] PER
[5] A The following function initiates a loop;each time it is run a column-
      block of X is calculated.
[6] CRE
[7] SB←FL;A;F;C1;C1;SB;Q;QQ
[8] C1←C1+υ0
[9] Z←0
[10] L1:←((Z+1)(1+ρF))/L6
[11] A←F[Z;]
[12] T←1
[13] L2:←(ACT)=0)/L4

```

```

[14] L3:TR1
[15] +((T+T+1)K)/L5
[16] +L2
[17] L4:TR1
[18] +((T+T+1)K)/L5
[19] +L2
[20] L5:REGR
[21] KRO
[22] @'BX',(SZ),'+Q'
[23] X←X,[2]Q
[24] CRE
[25] +L1
[26] L6:
[27] De((ρSN),(ρSN))ρ(((ρSN)*2)ρ1,((ρSN)ρ0))\SN)
[28] YMe((BD)+.xSY)
[29] KROP
[30] H←:1÷((1+ρY)*(K-1)))×QQ

```

▽VAR[0]▽

```

[0] VAR:I
[1] # The function VAR calculates an incidence matrix,A,required to compute X.
[2] A← 2 1 ρ 1 0
[3] # The initial A matrix is supplemented first with a final column of 1's;
    subsequently it is supplemented with a final column of 0's.The two re-
    sulting matrices are linked by the first dimension,and so on.
[4] I←1
[5] L1:A+(A,1),[1](A,0)
[6] +((I+I)=K)/L2
[7] +L1
[8] L2:

```

▽PER[0]▽

```

[0] PER:I
[1] # This function re-orders the rows of A to produce the matrix F.
[2] I←K+1
[3] F←(O,K)ρ0
[4] # All the rows of A with the same numbers of 1's form a row-block of F.
    Within each block,rows are ordered naturally from lower to higher.
[5] L1:+((I+I-1)(O)/L2
[6] F←F,[1]:(ΦA[(I=(+/A=0))/v1+ρA;])=0)
[7] +L1
[8] L2:

```

▽CRE[0]▽

```

[0] CRE:M
[1] # This function creates a set of empty variates(with no assigned values),
    U and I.
[2] M←1
[3] @'I',(SM),'+v0'
[4] +((M+M+1)K)/L1
[5] +3

```

```

[6] L1:Me1
[7]  a'U', (aH), 'e0'
[8]  +((MeH+1)K)/L2
[9]  +7
[10] L2:

```

▽TRIC00▽

```

[0] TR1
[1]  a For each row of F, if an element is '1', this function assigns to the cor-
    responding empty variate U a specified vector of unities.
[2]  a'U', (aT), 'e((SBCT1,1)ρ1)'
[3]  C1←C1,T

```

▽TRIC00▽

```

[0] TR1
[1]  a For each row of F, if an element is '0', this function assigns to the cor-
    responding empty variate I a specified identity matrix.
[2]  a'I', (aT), 'e((SBCT1,SBCT1)ρ(1,(SBCT1ρ0)))'
[3]  CI←CI,T

```

▽REORC00▽

```

[0] REOR;P;XX
[1]  a This function assigns the above vectors and matrices to the ordered vari-
    ates TOT.
[2]  P←0
[3]  L1:→((P+P+1)K)/L3
[4]  XX←ρa'U', (aP)
[5]  +((+/XX)=0)/L2
[6]  a'TOT', (aP), 'e', 'U', (aP)
[7]  +L1
[8]  L2:a'TOT', (aP), 'e', 'I', (aP)
[9]  +L1
[10] L3:

```

▽KROC00▽

```

[0] KRO;P;I;II
[1]  a The Kronecker matrix product of the variates TOT is carried out by this
    function, the output being a column-block of X, EX.
[2]  P←K
[3]  I1←a'TOT', (aP)
[4]  I←a'TOT', (aP+P-1)
[5]  L1:Q←(((1↑ρI)×(1↑ρII)),((~1↑ρI)×(~1↑ρII)))ρ 1 3 2 4 ρQ←I1.xII
[6]  +((P+P-1)=0)/L2
[7]  I1←Q
[8]  I←a'TOT', (aP)
[9]  +L1
[10] L2:

```

```

      ◊RMC0]◊
[0] RM
[1] A This function calculates the Restriccion Matrices for the several weigh-
    ing systems.
[2] COMB
[3] RIM
[4] TRIM
[5] RW

```

```

      ◊COMB0]◊
[0] COMB:I;HH;U;N;XT;HH1;G
[1] A Function COMB computes a vector, VF, whose elements are the tuples de-
    noting the positions of the column-blocks of X(or R) in a complete lexi-
    cographical order.
[2] HH←0
[3] U←0, vK
[4] A A initial VF vector is generated with the tuples of one element.
[5] VF←0, vK
[6] A From line [7] to [16], the tuples of two elements are added two elements
    are added to VF.
[7] I←0
[8] L1:U←1vU
[9] →((I+I+1)=K)/L3
[10] N←U[C1]
[11] XT←0
[12] L2:→((XT+XT+1)=pU)/L1
[13] HH←HH, s(≠N), sU[XT+1]
[14] →L2
[15] L3:HH←1+HH
[16] VF←VF, HH
[17] A From line [18] to [30], the tuples of three elements are added to VF, and
    so on.
[18] L4:I←0
[19] U←vK
[20] HH1←0
[21] L5:→((I+I+1)≠pHH)/L7
[22] N←HH[C1]
[23] G←s-1≠N
[24] XT←0
[25] L6:→((XT+XT+1)≠pU)/L5
[26] →(G≠U[XT])/L6
[27] HH1←HH1, H←s(≠HH[C1]), sU[XT]
[28] →L6
[29] L7:VF←VF, HH1←1+HH1
[30] HH←HH1
[31] →((p≠HH)=K)/L3
[32] →((p≠HH)=0)/L3
[33] L3:

```

```

      ◊RIM0]◊
[0] RIM:I;AX;Z:BT;P;HO;M;NU;UE
[1] A This function gives as output the Restriccion Incidence Matrix, E, using as
    input the vector VF.
[2] A A null E matrix is initially generated.

```

```

[3] E+(2+k),(2+k)ρ0
[4] EI;(2+k+1)←1
[5] a From line [6] to [18], for each tuple of VF, this function generates all
    possible 'linked' tuples (eliminating one element in the initial tuple).
[6] I←k+1
[7] L1:=(I+1)ρVF/L4
[8] AX←VF[I]
[9] Z←ρAX
[10] BT←P+0
[11] M←(H0,H0←ρAX)ρ1
[12] L2:=(ET+BT+1)H0/L3
[13] MC←BT;Z]←0
[14] Z←Z-1
[15] →L2
[16] L3:=(P+P+1)H0/L1
[17] VE←MCP;]
[18] NU←(VE/ρAX)
[19] a If the position of the initial tuple (in the natural ordering) denotes a
    column of E, and the positions of the linked tuples denote rows of E, i's
    are assigned to the resulting positions in E.
[20] EI((VF,NU);(VF,AX))←1
[21] →L3
[22] L4:
[23] ∇

```

∇TRIMEDJ∇

```

[0] TRIM:I;DDD;YY;NN:FFF;TET;T
[1] a This function computes the Transformed Restriccion Incidence matrix, T.
[2] I←0
[3] a If frequency weights are used (by means of the D matrix), the output is
    the T matrix for Frequency Weights, TFW.
[4] DDD←E
[5] a From line [6] to [17], a row of E is selected, its elements being replaced
    by submatrices, the result being a row-block of X, TET, which is linked to
    the previous one by the first dimension.
[6] L1:T←(0,(1ρX))ρ0
[7] NN←Y+1
[8] FFF←0
[9] L2:=(0+(E[NN;I]≠0))/L6
[10] TET←E[NN;YY]x(ρE'BX',(ρNN))+.xDDD+.xρE'BX',(ρYY)
[11] YY←Y+1
[12] L3:=(E[NN;YY]=0)/L4
[13] FFF←FFF,(ρE[NN;YY]x(ρE'BX',(ρNN))+.xDDD+.xρE'BX',(ρYY))
[14] L4:TET←TET,[2]E[NN;YY]x(ρE'BX',(ρNN))+.xDDD+.xρE'BX',(ρYY)
[15] →((Y+Y+1)(1ρE))/L5
[16] →L3
[17] L5:T←T,[1]TET
[18] a A new row of E is taken.
[19] →((NN+NN+1)(1ρE))/L6
[20] YY←1
[21] →L2
[22] L6:FFF+((ρFFF)÷2),2)ρFFF
[23] →((I+I+1)=2)/L7
[24] a If marginal weights are used (by means of H matrix), the function TRIM is
    run again, the result being the T matrix for Marginal Weights.
[25] DDD←H

```

```

[26] FFW←T
[27] →L1
[28] L7:TMW←T

▽RVWCD]▽
[0] RVW;J;C;I;CO;DBR;TT;ADD;WW;C;F;N;ARFW;FI;XM;NF;NC;FI;T;NNN
[1] * This function computes the Restriccion matrix,R,for Various Weighing
    systems.
[2] * The matrix TFW is used initially as inputs.
[3] T←TFW
[4] * From line [5] to [24],all the non-null rows of a column-block of T are
    taken to construct a diagonal-block of R,DBR.
[5] J←0
[6] L1:CO←t0
[7] C←2*K
[8] I←2
[9] L2:CO←CO,(↑1↑ρ(±'BX',(±I)))
[10] →((I←I+1)C)/L3
[11] →L2
[12] L3:I←1
[13] L4:±'DBR',(±I),'+((CO,(COI)))ρ0'
[14] →((I←I+1)ρCO)/L5
[15] →L4
[16] L5:I←TT←1;CO;DBR;TT;ADD;WW;C;F;N;ARFW;FI;XM;NF;NC;FI;T;NNN
[17] ADD←1
[18] L6:→((+/(↑(↑TI;(ADD+tCOCTT)))≠0))21)/L8
[19] L7:→((I←I+1)(↑ρT))/L9
[20] →L6
[21] L8:NNN←TI;(ADD+tCOCTT)]
[22] WW←±'DBR',(±TT)
[23] ±'DBR',(±TT),'+WW','[1]NNN'
[24] →L7
[25] * A new column-block of T is taken.
[26] L9:I←1
[27] ADD←ADD+COCTT]
[28] →((TT←TT+1)ρCO)/L10
[29] →L6
[30] L10:
[31] * Once the diagonal-blocks of R are constructed,they are diagonally linked
    from line [32] to [44] of RVW,the result being the matrix ARFW.
[32] I←(2*K)-1
[33] C←F←0
[34] N←1
[35] L11:C←C+(↑1↑ρ±'DBR',(±N))
[36] F←F+(↑1↑ρ±'DBR',(±N))
[37] →((N←N+1)I)/L12
[38] →L11
[39] L12:ARFW←(F,R)ρ0
[40] FI←CO←0
[41] XM←1
[42] L13:N←N+(↑1↑ρ±'DBR',(±XM))
[43] NC←(↑1↑ρ±'DBR',(±XM))
[44] ARFW←(FI+tNF);(CO+tNC)I←(±'DBR',(±XM))
[45] →((XM←XM+1)I)/L14
[46] FI←FI+NF

```



```

[47] CC=CC+NC
[48] →L13
[49] L14:+( (J+J+1)=2)/L15
[50] * A first column of zeros is added to ARFW, and the R matrix for Frequency
Weights, RFW, is calculated.
[51] RFW=((F,1)ρ0), [2]ARFW
[52] * If the TNW matrix is used as input, function RUW is run again, the output
now being the R matrix for Marginal Weights, RMW.
[53] T=TNW
[54] →L1
[55] L15:RMW=((F,1)ρ0), [2]ARFW
[56] * Finally, the matrix R for Usual Weights, RUW, is deduced from RFW.
[57] RUW=RFW÷0

```

▽REC01▽

```

[0] RE; XD; XDX; XDY; IFW; IMW; IUW; EEV; CMFW; CMMW; CMUW
[1] * This function, using the outputs of the above, gives the Restricted Esti-
mations for frequency, marginal, and usual weights, EFW, EMW, EUW, respective-
ly. Similarly, the estimations both of the error variance, EEV, and of the
standard deviations, SDEFW, SDEMw, and SDEUW, are calculated.
[2] XD=(QX)+, XD
[3] XDX=XD+, XX
[4] XDY=XD+, XYM
[5] IFW=B: XDX+( (QRFW)+, XFFW)
[6] IMW=B: XDX+( (QRMW)+, XRMW)
[7] IUW=B: XDX+( (QRUW)+, XRUW)
[8] EFW=IFW+, XDX
[9] EMW=IMW+, XDX
[10] EUW=IUW+, XDX
[11] EEV=(1÷((1+ρY)-(ρSH)))×((QY)+, XY)-(QYM)+, XD+, XX+, XEFW)
[12] CMFW=EEV×IFW+, XDX+, XIFW
[13] CMMW=EEV×IMW+, XDX+, XIMW
[14] CMUW=EEV×IUW+, XDX+, XIUW
[15] SDEFW=(( (1+ρCMFW), 1)ρ(1 1 QCMFW))÷(1÷2)
[16] SDEMw=(( (1+ρCMMW), 1)ρ(1 1 QCMMW))÷(1÷2)
[17] SDEUW=(( (1+ρCMUW), 1)ρ(1 1 QCMUW))÷(1÷2)

```

APPENDIX B. PROGRAM EXAMPLE.

An example is included in order to illustrate the above-listed functions. The main program, REUFM, is the only one that must be called by the user. For reasons of space, a two-factorial model is estimated, whose equation has already been given (see eq. 1, Sec. 1). This example shows the input data, the main intermediate results, and the final outputs described in the text (see Sec. 3.3 and App. A).

The data, which are required by DD, are:

```

a) FL (factor numbers)
   = 2 3

b) SH (subclass numbers)
   = 2 5 6 4 7 3

c) BY (observation values)
   = 9 11 4 10 8 6 7 5 10 13 7 11 8 5 8 9 2 10 15 6 9 13 8 16 6 11 7
  
```

The main intermediate results are:

```

a) Function EIM:

      1      1 0      1 0 0      1 0 0 0 0 0
      1      1 0      0 1 0      0 1 0 0 0 0
BX1 = 1 ; BX2 = 1 0 ; BX3 = 0 0 1 ; BX4 = 0 0 1 0 0 0
      1      0 1      1 0 0      0 0 0 1 0 0
      1      0 1      0 1 0      0 0 0 0 1 0
      1      0 1      0 0 1      0 0 0 0 0 1

      (BX = Blocks of the effect matrix X)

      1 1 0 1 0 0 1 0 0 0 0 0
      1 1 0 0 1 0 0 1 0 0 0 0
      X = 1 1 0 0 0 1 0 0 1 0 0 0
      1 0 1 1 0 0 0 0 0 1 0 0
      1 0 1 0 1 0 0 0 0 0 1 0
      1 0 1 0 0 1 0 0 0 0 0 1

      2 0 0 0 0 0      2.89 0      0      0      0      0      10
      0 5 0 0 0 0      0      5.78 0      0      0      0      7
D = 0 0 6 0 0 0 ; H = 0      0      4.33 0      0      0      9
      0 0 0 4 0 0      0      0      0      3.11 0      0      6
      0 0 0 0 7 0      0      0      0      0      6.22 0      11
      0 0 0 0 0 3      0      0      0      0      0      4.67      8
  
```

```

b) Function COMB:

      VF (vector of tuples) = 0 1 2 12
  
```

```

c) Function RIM:

      0 1 1 0
      E (Restriction Incidence Matrix) = 0 0 0 1
      0 0 0 1
      0 0 0 0
  
```

```

d) Function TRIM:

      TFW (Transformed E matrix for Frequency Weights)
  
```

```

0 13 14 6 12 9 0 0 0 0 0 0
0 0 0 0 0 0 2 5 6 0 0 0
= 0 0 0 0 0 0 0 0 0 4 7 3
0 0 0 0 0 0 2 0 0 4 0 0
0 0 0 0 0 0 0 5 0 0 7 0
0 0 0 0 0 0 0 0 6 0 0 3

```

TMW (Transformed E matrix for Marginal Weights)

```

0 13 14 6 12 9 0 0 0 0 0 0
0 0 0 0 0 0 2.89 5.78 4.33 0 0 0
= 0 0 0 0 0 0 0 0 0 3.11 6.22 4.67
0 0 0 0 0 0 2.89 0 0 3.11 0 0
0 0 0 0 0 0 0 5.78 0 0 6.22 0
0 0 0 0 0 0 0 0 4.33 0 0 4.67

```

e) Function RWJ:

RFW (R matrix for Frequency Weights)

```

0 13 14 0 0 0 0 0 0 0 0 0
0 0 0 6 12 9 0 0 0 0 0 0
= 0 0 0 0 0 0 2 5 6 0 0 0
0 0 0 0 0 0 0 0 0 4 7 3
0 0 0 0 0 0 2 0 0 4 0 0
0 0 0 0 0 0 0 5 0 0 7 0
0 0 0 0 0 0 0 0 6 0 0 3

```

RMW (R matrix for Marginal Weights)

```

0 13 14 0 0 0 0 0 0 0 0 0
0 0 0 6 12 9 0 0 0 0 0 0
= 0 0 0 0 0 0 2.89 5.78 4.33 0 0 0
0 0 0 0 0 0 0 0 0 3.11 6.22 4.67
0 0 0 0 0 0 2.89 0 0 3.11 0 0
0 0 0 0 0 0 0 5.78 0 0 6.22 0
0 0 0 0 0 0 0 0 4.33 0 0 4.67

```

RUW (R matrix for Usual Weights)

```

0 1 1 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 0 0 0 0 0 0
= 0 0 0 0 0 0 1 1 1 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1
0 0 0 0 0 0 1 0 0 1 0 0
0 0 0 0 0 0 0 1 0 0 1 0
0 0 0 0 0 0 0 0 1 0 0 1

```

The final outputs, produced by function RE, are the Restricted Estimation of the effects and their Standard Deviations. These effects, in order of appearance, are: general mean, first-factor main effects (two levels), second-factor main effects, and interaction effects (two by three levels).

EFW,SDEFW (Estimations for Frequency Weights, and Standard Deviations)

8.67	0.582
^-0.36	0.627
0.334	0.582
^-1.44	1.1
0.622	0.655
= 0.128	0.853
3.13	1.55
^-1.93	0.754
0.564	0.588
^-1.56	0.774
1.38	0.539
^-1.13	1.18

EMW,SDEMW (Idem for Marginal Weights)

8.62	0.605
^-0.288	0.628
0.267	0.583
^-0.695	1.14
0.453	0.672
= ^-0.14	0.87
2.36	1.19
^-1.79	0.698
0.807	0.897
^-2.19	1.11
1.66	0.648
^-0.749	0.832

EUW,SDEUW (Idem for Usual Weights)

8.5	0.636
0.167	0.636
^-0.167	0.636
^-0.5	0.988
0.5	0.816
= 0.0	0.886
1.83	0.988
^-2.17	0.816
0.333	0.886
^-1.83	0.988
2.17	0.816
^-0.333	0.886