NAG Fortran Library Routine Document

G02JBF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

G02JBF fits a linear mixed effects regression model using maximum likelihood (ML).

2 Specification

```
 SUBROUTINE G02JBF (N, NCOL, LDDAT, DAT, LEVELS, YVID, CWID, NFV, FVID,
1                   FINT, NRV, RVID, NVPR, VPR, RINT, SVID, GAMMA, NFF,
2                   NRF, DF, ML, LB, B, SE, MAXIT, TOL, WARN, IFAIL)
 INTEGER            N, NCOL, LDDAT, LEVELS(NCOL), YVID, CWID, NFV,
1                   FVID(NFV), FINT, NRV, RVID(NRV), NVPR, VPR(NRV),
2                   RINT, SVID, LB, MAXIT, WARN, IFAIL
 double precision   DAT(LDDAT,NCOL), GAMMA(NVPR+2), NFF, NRF, DF, ML,
1                   B(LB), SE(LB), TOL
```

3 Description

G02JBF fits a model of the form:

$$y = X\beta + Z\nu + \epsilon$$

where y is a vector of n observations on the dependent variable,

X is a known n by p design matrix for the fixed independent variables,

β is a vector of length p of unknown *fixed effects*,

Z is a known n by q design matrix for the random independent variables,

ν is a vector of length q of unknown *random effects*;

and ϵ is a vector of length n of unknown random errors.

Both ν and ϵ are assumed to have a Gaussian distribution with expectation zero and

$$\mathrm{Var}\begin{bmatrix}\nu\\ \epsilon\end{bmatrix} = \begin{bmatrix}G & 0\\ 0 & R\end{bmatrix}$$

where $R = \sigma_R^2 I$, I is the $n \times n$ identity matrix and G is a diagonal matrix. It is assumed that the random variables, Z, can be subdivided into $g \leq q$ groups with each group being identically distributed with expectations zero and variance σ_i^2. The diagonal elements of matrix G therefore take one of the values $\left\{\sigma_i^2 : i = 1, \ldots, g\right\}$, depending on which group the associated random variable belongs to.

The model therefore contains three sets of unknowns, the fixed effects, β, the random effects μ and a vector of $g + 1$ variance components, γ, where $\gamma = \left\{\sigma_1^2, \sigma_2^2, \ldots, \sigma_{g-1}^2, \sigma_g^2, \sigma_R^2\right\}$. Rather than working directly with γ, G02JBF uses an iterative process to estimate $\gamma^* = \left\{\sigma_1^2/\sigma_R^2, \sigma_2^2/\sigma_R^2, \ldots, \sigma_{g-1}^2/\sigma_R^2, \sigma_g^2/\sigma_R^2, 1\right\}$. Due to the iterative nature of the estimation a set of initial values, γ_0, for γ^* is required. G02JBF allows these initial values either to be supplied by the user or calculated from the data using the minimum variance quadratic unbiased estimators (MIVQUE0) suggested by Rao (1972).

G02JBF fits the model using a quasi-Newton algorithm to maximize the log-likelihood function:

$$-2l_R = \log(|V|) + (n)\log\left(r'V^{-1}r\right) + \log(2\pi/n)$$

where

$$V = ZGZ' + R, \quad r = y - Xb$$

and

$$b = \left(X'V^{-1}X\right)^{-1} X'V^{-1}y.$$

Once the final estimates for γ^* have been obtained, the value of σ_R^2 is given by:

$$\sigma_R^2 = \left(r'V^{-1}r\right)/(n-p).$$

Case weights, W_c, can be incorporated into the model by replacing $X'X$ and $Z'Z$ with $X'W_cX$ and $Z'W_cZ$ respectively, for a diagonal weight matrix W_c.

The log likelihood, l_R, is calculated using the sweep algorithm detailed in Wolfinger *et al.* (1994). Additional information on the quasi-Newton algorithm used can be found in the documentation for E04UFA.

4 References

Goodnight J H (1979) A Tutorial on the SWEEP operator *The American Statistician* **33 (3)** 149–158

Harville D A (1977) Maximum likelihood approaches to variance component estimation and to related problems *JASA* **72** 320–340

Rao C R (1972) Estimation of variance and covariance components in a linear model *J. Am. Stat. Assoc.* **67** 112–115

Stroup W W (1989) Predictable functions and prediction space in the mixed model procedure *Applications of Mixed Models in Agriculture and Related Disciplines* **Southern Cooperative Series Bulletin No. 343** 39–48

Wolfinger R, Tobias R and Sall J (1994) Computing Gaussian Likelihoods and Their Derivatives for General Linear Mixed Models *SIAM Sci. Statist. Comput.* **15** 1294–1310

5 Parameters

1: N – INTEGER *Input*

On entry: n, the number of observations.

Constraint: $\text{N} \geq 1$.

2: NCOL – INTEGER *Input*

On entry: the number of columns in the data matrix, DAT.

Constraint: $\text{NCOL} \geq 2$.

3: LDDAT – INTEGER *Input*

On entry: the first dimension of the array DAT as declared in the (sub)program from which G02JBF is called.

Constraint: $\text{LDDAT} \geq \text{N}$.

4: DAT(LDDAT,NCOL) – ***double precision*** array *Input*

On entry: array containing all of the data. For the ith observation: $\text{DAT}(i, \text{YVID})$ holds the dependent, y variable; if $\text{CWID} \neq 0$ then $\text{DAT}(i, \text{CWID})$ holds the case weights; if $\text{SVID} \neq 0$ then $\text{DAT}(i, \text{SVID})$ holds the subject variable. The remaining columns hold the values of the independent variables.

Constraints:

if $\mathrm{CWID} \neq 0$, $\mathrm{DAT}(i, \mathrm{CWID}) \geq 0$;
if $\mathrm{LEVELS}(j) \neq 1$, $\mathrm{DAT}(i,j) > 0$, $\mathrm{DAT}(i,j) \leq \mathrm{LEVELS}(j)$.

5: LEVELS(NCOL) – INTEGER array *Input*

On entry: $\mathrm{LEVELS}(i)$ contains the number of levels associated with the variable in column i of the data matrix DAT. If the variable associated with column i of DAT is continuous or binary (i.e., only takes the values zero or one) then $\mathrm{LEVELS}(i)$ should be 1; if the variable is discrete then $\mathrm{LEVELS}(i)$ is the number of levels associated with that variable and column i is assumed to take the values 1 to $\mathrm{LEVELS}(i)$.

Constraint: $\mathrm{LEVELS}(i) \geq 1$, for $i = 1, 2, \ldots, \mathrm{NCOL}$.

6: YVID – INTEGER *Input*

On entry: the column of DAT holding the dependent, y, variable.

Constraint: $1 \leq \mathrm{YVID} \leq \mathrm{NCOL}$.

7: CWID – INTEGER *Input*

On entry: the column of DAT holding the case weights. If $\mathrm{CWID} = 0$ then no weights are used.

Constraint: $0 \leq \mathrm{CWID} \leq \mathrm{NCOL}$.

8: NFV – INTEGER *Input*

On entry: the number of independent variables in the model which are to be treated as being fixed.

Constraint: $0 \leq \mathrm{NFV} < \mathrm{NCOL}$.

9: FVID(NFV) – INTEGER array *Input*

On entry: the columns of the data matrix DAT holding the fixed independent variables with $\mathrm{FVID}(i)$ holding the column number corresponding to the ith fixed variable.

Constraint: $1 \leq \mathrm{FVID}(i) \leq \mathrm{NCOL}$, for $i = 1, 2, \ldots, \mathrm{NFV}$.

10: FINT – INTEGER *Input*

On entry: flag indicating whether a fixed intercept is included ($\mathrm{FINT} = 1$).

Constraint: $\mathrm{FINT} = 0$ or 1.

11: NRV – INTEGER *Input*

On entry: the number of independent variables in the model which are to be treated as being random.

Constraint: $0 \leq \mathrm{NRV} < \mathrm{NCOL}$.

12: RVID(NRV) – INTEGER array *Input*

On entry: the columns of the data matrix DAT holding the random independent variables with $\mathrm{RVID}(i)$ holding the column number corresponding to the ith random variable.

Constraint: $1 \leq \mathrm{RVID}(i) \leq \mathrm{NCOL}$, for $i = 1, 2, \ldots, \mathrm{NRV}$.

13: NVPR – INTEGER *Input*

On entry: if $\mathrm{RINT} = 1$ and $\mathrm{SVID} \neq 0$, NVPR is the number of variance components being estimated $- 2$, $(g - 1)$, else $\mathrm{NVPR} = g$. If $\mathrm{NRV} = 0$, NVPR is not referenced.

Constraint: if $\mathrm{NRV} \neq 0$, $1 \leq \mathrm{NVPR} \leq \mathrm{NRV}$.

14: VPR(NRV) – INTEGER array *Input*

On entry: VPR(i) holds a flag indicating the variance of the ith random variable. The variance of the ith random variable is σ_j^2, where $j = \text{VPR}(i) + 1$ if RINT $= 0$ and SVID $\neq 0$ and $j = \text{VPR}(i)$ otherwise. Random variables with the same value of j are assumed to be taken from the same distribution.

Constraint: $1 \leq \text{VPR}(i) \leq \text{NVPR}$, for $i = 1, 2, \ldots, \text{NRV}$.

15: RINT – INTEGER *Input*

On entry: flag indicating whether a random intercept is included (RINT $= 1$). If SVID $= 0$ RINT is not referenced.

Constraint: RINT $= 0$ or 1.

16: SVID – INTEGER *Input*

On entry: the column of DAT holding the subject variable. If SVID $= 0$ then no subject variable is used.

Specifying a subject variable is equivalent to specifying the interaction between that variable and all of the random-effects. Letting the notation $Z_1 \times Z_S$ denote the interaction between variables Z_1 and Z_S, fitting a model with RINT $= 0$, random-effects $Z_1 + Z_2$ and subject variable Z_S is equivalent to fitting a model with random-effects $Z_1 \times Z_S + Z_2 \times Z_S$ and no subject variable. If RINT $= 1$ the model is equivalent to fitting $Z_S + Z_1 \times Z_S + Z_2 \times Z_S$ and no subject variable.

Constraint: $0 \leq \text{SVID} \leq \text{NCOL}$.

17: GAMMA(NVPR + 2) – ***double precision*** array *Input/Output*

On entry: holds the initial values of the variance components, γ_0, with GAMMA(i) the initial value for σ_i^2/σ_R^2, $i = 1, 2, \ldots, g$. If RINT $= 1$ and SVID $\neq 0$ then $g = \text{NVPR} + 1$, else $g = \text{NVPR}$.

If GAMMA(1) $= -1$ then the remaining elements of GAMMA are ignored and the initial values for the variance components are estimated from the data using MIVQUE0.

On exit: GAMMA(i), for $i = 1, 2, \ldots, g$ holds the final estimate of σ_i^2 and GAMMA($g + 1$) holds the final estimate for σ_R^2.

Constraint: GAMMA(1) $= -1$ or GAMMA(i) ≥ 0, for $i = 1, 2, \ldots, g$.

18: NFF – ***double precision*** *Output*

On exit: the number of fixed effects estimated (i.e., the number of columns, p, in the design matrix X).

19: NRF – ***double precision*** *Output*

On exit: the number of random effects estimated (i.e., the number of columns, q, in the design matrix Z).

20: DF – ***double precision*** *Output*

On exit: the degrees of freedom.

21: ML – ***double precision*** *Output*

On exit: $-2l_R(\hat{\gamma})$ where l_R is the log of the maximum likelihood calculated at $\hat{\gamma}$, the estimated variance components returned in GAMMA.

22: LB – INTEGER *Input*

On entry: the size of the array B.

Constraint:

$$\mathrm{LB} \geq \mathrm{FINT} + \sum_{i=1}^{\mathrm{NFV}} \max(\mathrm{LEVELS}(\mathrm{FVID}(i)) - 1, 1) + L_S \times \left(\mathrm{RINT} + \sum_{i=1}^{\mathrm{NRV}} \mathrm{LEVELS}(\mathrm{RVID}(i)) \right)$$

where $L_S = \mathrm{LEVELS}(\mathrm{SVID})$ if $\mathrm{SVID} \neq 0$ and 1 otherwise.

23: B(LB) – ***double precision*** array *Output*

On exit: the parameter estimates, (β, ν), with the first NFF elements of B containing the fixed effect parameter estimates, β and the next NRF elements of B containing the random effect parameter estimates, ν.

Fixed effects

If FINT $= 1$, B(1) contains the estimate of the fixed intercept. Let L_i denote the number of levels associated with the ith fixed variable, that is $L_i = \mathrm{LEVELS}(\mathrm{FVID}(i))$. Define

if FINT $= 1$, $F_1 = 2$ else if FINT $= 0$, $F_1 = 1$;

$F_{i+1} = F_i + \max(L_i - 1, 1)$, $i \geq 1$.

Then for $i = 1, 2, \ldots, \mathrm{NFV}$:

if $L_i > 1$, $\mathrm{B}(F_i + j - 2)$ contains the parameter estimate for the jth level of the ith fixed variable, $j = 2, 3, \ldots, L_i$;
if $L_i \leq 1$, $\mathrm{B}(F_i)$ contains the parameter estimate for the ith fixed variable.

Random effects

Redefining L_i to denote the number of levels associated with the ith random variable, that is $L_i = \mathrm{LEVELS}(\mathrm{RVID}(i))$. Define

if RINT $= 1$, $R_1 = 2$ else if RINT $= 0$, $R_1 = 1$;

$R_{i+1} = R_i + L_i$, $i \geq 1$.

Then for $i = 1, 2, \ldots, \mathrm{NRV}$:

if SVID $= 0$,

if $L_i > 1$, $\mathrm{B}(\mathrm{NFF} + R_i + j - 1)$ contains the parameter estimate for the jth level of the ith random variable, $j = 1, 2, \ldots, L_i$;
if $L_i \leq 1$, $\mathrm{B}(\mathrm{NFF} + R_i)$ contains the parameter estimate for the ith random variable;

if SVID $\neq 0$,

let L_S denote the number of levels associated with the subject variable, that is $L_S = \mathrm{LEVELS}(\mathrm{SVID})$;
if $L_i > 1$, $\mathrm{B}(\mathrm{NFF} + (s-1)L_S + R_i + j - 1)$ contains the parameter estimate for the interaction between the sth level of the subject variable and the jth level of the ith random variable, $s = 1, 2, \ldots, L_S$ and $j = 1, 2, \ldots, L_i$;
if $L_i \leq 1$, $\mathrm{B}(\mathrm{NFF} + (s-1)L_S + R_i)$ contains the parameter estimate for the interaction between the sth level of the subject variable and the ith random variable, $s = 1, 2, \ldots, L_S$;
if RINT $= 1$, $\mathrm{B}(\mathrm{NFF} + 1)$ contains the estimate of the random intercept.

24: SE(LB) – ***double precision*** array *Output*

On exit: the standard errors of the parameter estimates given in B.

25: MAXIT – INTEGER *Input*

On entry: the maximum number of iterations. If $\mathrm{MAXIT} < 0$ then the default value of 100 is used. If MAXIT $= 0$, the parameter estimates (β, μ) and corresponding standard errors are calculated based on the value of γ_0 supplied in GAMMA.

26: TOL – ***double precision*** *Input*

On entry: the tolerance used to assess convergence. If TOL $= 0$ then the default value of $\epsilon^{0.7}$ is used, where ϵ is the ***machine precision***.

27: WARN – INTEGER *Output*

On exit: WARN is set to 1 if a variance component was estimated to be a negative value during the fitting process. Otherwise WARN is set to 0.

If WARN $=$ 1, the negative estimate is set to zero and the estimation process allowed to continue.

28: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details.

On exit: IFAIL $= 0$ unless the routine detects an error (see Section 6).

For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, for users not familiar with this parameter the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

6 Error Indicators and Warnings

If on entry IFAIL $=$ 0 or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL $= 1$

On entry, N $<$ 2,
or NCOL $<$ 1,
or LDDAT $<$ N,
or YVID $<$ 1 or YVID $>$ NCOL,
or CWID $<$ 0 or CWID $>$ NCOL,
or NFV $<$ 0 or NFV $\geq$ NCOL,
or FINT $\neq$ 0 and FINT $\neq$ 1,
or NRV $<$ 0 or NRV $\geq$ NCOL,
or NVPR $<$ 0 or NVPR $>$ NRV or (NRV $>$ 0 and NVPR $<$ 1),
or RINT $\neq$ 0 and RINT $\neq$ 1,
or SVID $<$ 0 or SVID $>$ NCOL,
or LB is too small.

IFAIL $= 2$

On entry, LEVELS$(i) < 1$, for at least one i,
or FVID$(i) < 1$, or FVID$(i) >$ NCOL, for at least one i,
or RVID$(i) < 1$, or RVID$(i) >$ NCOL, for at least one i,
or VPR$(i) < 1$ or VPR$(i) >$ NVPR, for at least one i,
or at least one discrete variable in array DAT has a value greater than that specified in LEVELS,
or GAMMA$(i) < 0$, for at least one i, and GAMMA$(1) \neq -1$.

IFAIL $= 3$

Degrees of freedom < 1. The number of parameters exceed the effective number of observations.

IFAIL = 4

The routine failed to converge to the specified tolerance in MAXIT iterations. See Section 8 for advice.

7 Accuracy

The accuracy of the results can be adjusted through the use of the TOL parameter.

8 Further Comments

Wherever possible any block structure present in the design matrix Z should be modelled through a subject variable, specified via SVID, rather than being explicitly entered into DAT.

G02JBF uses an iterative process to fit the specified model and for some problems this process may fail to converge (see IFAIL $= 4$). If the routine fails to converge then the maximum number of iterations (see MAXIT) or tolerance (see TOL) may require increasing. Alternatively, the model can be fit using restricted maximum likelihood (see G02JAF) or using the non-iterative MIVQUE0.

To fit the model just using MIVQUE0, the first element of GAMMA should be set to -1 and MAXIT should be set to zero.

Although the quasi-Newton algorithm used in G02JBF tends to require more iterations before converging compared to the Newton–Raphson algorithm recommended by Wolfinger *et al.* (1994), it does not require the second derivatives of the likelihood function to be calculated and consequentially takes significantly less time per iteration.

9 Example

The following dataset is taken from Stroup (1989) and arises from a balanced split-plot design with the whole plots arranged in a randomised complete block-design.

In this example the full design matrix for the random independent variable, Z, is given by:

$$Z = \begin{pmatrix}
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1
\end{pmatrix}$$

$$= \begin{pmatrix} A & 0 & 0 & 0 \\ 0 & A & 0 & 0 \\ 0 & 0 & A & 0 \\ 0 & 0 & 0 & A \\ A & 0 & 0 & 0 \\ 0 & A & 0 & 0 \\ 0 & 0 & A & 0 \\ 0 & 0 & 0 & A \end{pmatrix}, \tag{1}$$

where

$$A = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

The block structure evident in (1) is modelled by specifying a four-level subject variable, taking the values $\{1,1,1,2,2,2,3,3,3,4,4,4,1,1,1,2,2,2,3,3,3,4,4,4\}$. The first column of 1s is added to A by setting RINT $=$ 1. The remaining columns of A are specified by a three level factor, taking the values, $\{1,2,3,1,2,3,1,\ldots\}$.

9.1 Program Text

Note: the listing of the example program presented below uses ***bold italicised*** terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     G02JBF Example Program Text
*     Mark 21 Release. NAG Copyright 2004.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          LDDAT, VMAX, FMAX, RMAX, LMAX, SMAX, TEMAX
      PARAMETER        (LDDAT=24,VMAX=5,FMAX=3,RMAX=1,LMAX=3,SMAX=4,
     +                 TEMAX=2)
      INTEGER          NZZ
      PARAMETER        (NZZ=RMAX*LMAX*SMAX+SMAX)
      INTEGER          MLB
      PARAMETER        (MLB=FMAX*LMAX+NZZ)
*     .. Local Scalars ..
      DOUBLE PRECISION LIKE, TOL
      INTEGER          CWID, DF, FINT, I, IFAIL, J, K, L, LB, MAXIT, N,
     +                 NCOL, NFF, NFV, NRF, NRV, NV, NVPR, RINT, SVID,
     +                 WARNP, YVID
*     .. Local Arrays ..
      DOUBLE PRECISION B(MLB), DAT(LDDAT,VMAX), GAMMA(TEMAX+2), SE(MLB)
      INTEGER          FVID(FMAX), LEVELS(VMAX), RVID(RMAX), VPR(RMAX)
*     .. External Subroutines ..
      EXTERNAL         G02JBF
*     .. Executable Statements ..
      CONTINUE

      WRITE (NOUT,*) 'G02JBF Example Program Results'

      LB = MLB

*     Skip heading in data file
      READ (NIN,*)

*     Read in the problem size information
      READ (NIN,*) N, NCOL, NFV, NRV, NVPR
      NV = NFV + NRV

*     Check array sizes
      IF (NCOL.GT.VMAX .OR. NV.GT.(FMAX+RMAX)) THEN
         WRITE (NOUT,*) 'Problem too large, increase size of arrays'
         STOP
```

```
      END IF

*     Read in number of levels for each variable
      READ (NIN,*) (LEVELS(I),I=1,NCOL)

*     Read in model information
      READ (NIN,*) YVID, (FVID(I),I=1,NFV), (RVID(I),I=1,NRV), SVID,
     +  CWID, FINT, RINT

*     If no subject specified, then ignore RINT
      IF (SVID.EQ.0) RINT = 0

*     Check remaining array sizes
      IF (N.GT.LDDAT .OR. (NVPR+RINT).GT.TEMAX) THEN
         WRITE (NOUT,*) 'Problem too large, increase size of arrays'
         STOP
      END IF

*     Read in the variance component flag
      READ (NIN,*) (VPR(I),I=1,NRV)

*     Read in the Data matrix
      DO 20 I = 1, N
         READ (NIN,*) (DAT(I,J),J=1,NCOL)
   20 CONTINUE

*     Read in the initial values for GAMMA
      READ (NIN,*) (GAMMA(I),I=1,NVPR+RINT)

*     Read in the maximum number of iterations
      READ (NIN,*) MAXIT

*     Run the analysis
      IFAIL = 0
      TOL = 0.0D0
      WARNP = 0
      CALL G02JBF(N,NCOL,LDDAT,DAT,LEVELS,YVID,CWID,NFV,FVID,FINT,NRV,
     +            RVID,NVPR,VPR,RINT,SVID,GAMMA,NFF,NRF,DF,LIKE,LB,B,SE,
     +            MAXIT,TOL,WARNP,IFAIL)

*     Output results
      IF (WARNP.NE.0) THEN
         WRITE (NOUT,*) 'Warning: At least one variance component was ',
     +     'estimated to be negative and then reset to zero'
      END IF
      WRITE (NOUT,*) 'Fixed effects (Estimate and Standard Deviation)'
      WRITE (NOUT,*)
      K = 1
      IF (FINT.EQ.1) THEN
         WRITE (NOUT,99999) 'Intercept             ', B(K), SE(K)
         K = K + 1
      END IF
      DO 60 I = 1, NFV
         DO 40 J = 1, LEVELS(FVID(I))
            IF (LEVELS(FVID(I)).NE.1 .AND. J.EQ.1) GO TO 40
            WRITE (NOUT,99995) 'Variable', I, ' Level', J, B(K), SE(K)
            K = K + 1
   40    CONTINUE
   60 CONTINUE

      WRITE (NOUT,*)
      WRITE (NOUT,*) 'Random Effects (Estimate and Standard',
     +  ' Deviation)'
      IF (SVID.EQ.0) THEN
         DO 100 I = 1, NRV
            DO 80 J = 1, LEVELS(RVID(I))
               WRITE (NOUT,99995) 'Variable', I, ' Level', J, B(K),
     +           SE(K)
               K = K + 1
   80       CONTINUE
  100    CONTINUE
```

```
      ELSE
         DO 160 L = 1, LEVELS(SVID)
            IF (RINT.EQ.1) THEN
               WRITE (NOUT,99998) 'Intercept for Subject Level', L,
     +           '          ', B(K), SE(K)
               K = K + 1
            END IF
            DO 140 I = 1, NRV
               DO 120 J = 1, LEVELS(RVID(I))
                  WRITE (NOUT,99997) 'Subject Level', L, ' Variable', I,
     +              ' Level', J, B(K), SE(K)
                  K = K + 1
  120          CONTINUE
  140       CONTINUE
  160    CONTINUE
      END IF

      WRITE (NOUT,*)
      WRITE (NOUT,*) ' Variance Components'
      DO 180 I = 1, (NVPR+RINT)
         WRITE (NOUT,99996) I, GAMMA(I)
  180 CONTINUE

      WRITE (NOUT,99994) 'SIGMA^2     = ', GAMMA(NVPR+RINT+1)
      WRITE (NOUT,99994) '-2LOG LIKE  = ', LIKE
      WRITE (NOUT,*) 'DF          = ', DF
      STOP

99999 FORMAT (A,2F10.4)
99998 FORMAT (A,I4,A,2F10.4)
99997 FORMAT (3(A,I4),2F10.4)
99996 FORMAT (I4,F10.4)
99995 FORMAT (2(A,I4),2F10.4)
99994 FORMAT (A,F10.4)
      END
```

9.2 Program Data

```
G02JBF Example Program Data
24 5 3 1 1
1 4 3 2 3
1 3 4 5 3 2 0 1 1
1
56  1  1  1  1
50  1  2  1  1
39  1  3  1  1
30  2  1  1  1
36  2  2  1  1
33  2  3  1  1
32  3  1  1  1
31  3  2  1  1
15  3  3  1  1
30  4  1  1  1
35  4  2  1  1
17  4  3  1  1
41  1  1  2  1
36  1  2  2  2
35  1  3  2  3
25  2  1  2  1
28  2  2  2  2
30  2  3  2  3
24  3  1  2  1
27  3  2  2  2
19  3  3  2  3
25  4  1  2  1
30  4  2  2  2
18  4  3  2  3
1.0 1.0
-1
```

9.3 Program Results

```
 G02JBF Example Program Results
 Fixed effects (Estimate and Standard Deviation)

Intercept                 37.0000    4.0421
Variable   1 Level   2     1.0000    3.0461
Variable   1 Level   3   -11.0000    3.0461
Variable   2 Level   2    -8.2500    1.8736
Variable   3 Level   2     0.5000    2.6497
Variable   3 Level   3     7.7500    2.6497

 Random Effects (Estimate and Standard Deviation)
Intercept for Subject Level   1             10.7631    3.8855
Subject Level   1 Variable   1 Level   1     3.7276    2.6268
Subject Level   1 Variable   1 Level   2    -1.4476    2.6268
Subject Level   1 Variable   1 Level   3     0.3733    2.6268
Intercept for Subject Level   2             -0.5269    3.8855
Subject Level   2 Variable   1 Level   1    -3.7171    2.6268
Subject Level   2 Variable   1 Level   2    -1.2253    2.6268
Subject Level   2 Variable   1 Level   3     4.8125    2.6268
Intercept for Subject Level   3             -5.6450    3.8855
Subject Level   3 Variable   1 Level   1     0.5903    2.6268
Subject Level   3 Variable   1 Level   2     0.3987    2.6268
Subject Level   3 Variable   1 Level   3    -2.3806    2.6268
Intercept for Subject Level   4             -4.5912    3.8855
Subject Level   4 Variable   1 Level   1    -0.6009    2.6268
Subject Level   4 Variable   1 Level   2     2.2742    2.6268
Subject Level   4 Variable   1 Level   3    -2.8052    2.6268

  Variance Components
   1   46.7969
   2   11.5365
SIGMA^2     =     7.0208
-2LOG LIKE  =   141.6877
 DF          =  16
```