NAG Fortran Library Routine Document

G02DDF

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

G02DDF calculates the regression parameters for a general linear regression model. It is intended to be called after G02DCF, G02DEF or G02DFF.

2 Specification

```
      SUBROUTINE G02DDF(N, IP, Q, LDQ, RSS, IDF, B, SE, COV, SVD, IRANK, P,
     1                  TOL, WK, IFAIL)
      INTEGER           N, IP, LDQ, IDF, IRANK, IFAIL
      real              Q(LDQ,IP+1), RSS, B(IP), SE(IP), COV(IP*(IP+1)/2),
     1                  P(IP*IP+2*IP), TOL, WK(IP*IP+(IP-1)*5)
      LOGICAL           SVD
```

3 Description

A general linear regression model fitted by G02DAF may be adjusted by adding or deleting an observation using G02DCF, adding a new independent variable using G02DEF or deleting an existing independent variable using G02DFF. Alternatively a model may be constructed by a forward selection procedure using G02EEF. These routines compute the vector c and the upper triangular matrix R. G02DDF takes these basic results and computes the regression coefficients, $\hat{\beta}$, their standard errors and their variance-covariance matrix.

If R is of full rank, then $\hat{\beta}$ is the solution to

$$R\hat{\beta} = c_1,$$

where c_1 is the first p elements of c.

If R is not of full rank a solution is obtained by means of a singular value decomposition (SVD) of R,

$$R = Q_* \begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix} P^T,$$

where D is a k by k diagonal matrix with non-zero diagonal elements, k being the rank of R, and Q_* and P are p by p orthogonal matrices. This gives the solution

$$\hat{\beta} = P_1 D^{-1} Q_{*_1}^T c_1.$$

P_1 being the first k columns of P, i.e., $P = (P_1 P_0)$, and Q_{*_1} being the first k columns of Q_*.

Details of the SVD are made available in the form of the matrix P^*:

$$P^* = \begin{pmatrix} D^{-1} P_1^T \\ P_0^T \end{pmatrix}.$$

This will be only one of the possible solutions. Other estimates may be obtained by applying constraints to the parameters. These solutions can be obtained by calling G02DKF after calling G02DDF. Only certain linear combinations of the parameters will have unique estimates; these are known as estimable functions. These can be estimated using G02DNF.

The residual sum of squares required to calculate the standard errors and the variance-covariance matrix can either be input or can be calculated if additional information on c for the whole sample is provided.

4 References

Golub G H and van Loan C F (1996) *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

Hammarling S (1985) The singular value decomposition in multivariate statistics *SIGNUM Newsl.* **20 (3)** 2–25

Searle S R (1971) *Linear Models* Wiley

5 Parameters

1: N – INTEGER *Input*

On entry: the number of observations.

Constraint: $N \geq 1$.

2: IP – INTEGER *Input*

On entry: the number of terms in the regression model, p.

Constraint: $IP \geq 1$.

3: Q(LDQ,IP+1) – ***real*** array *Input*

On entry: Q must be the array Q as output by G02DCF, G02DEF, G02DFF or G02EEF. If on entry $RSS \leq 0.0$ then all N elements of c are needed. This is provided by routines G02DEF, G02DFF or G02EEF.

4: LDQ – INTEGER *Input*

On entry: the first dimension of the array Q as declared in the (sub)program from which G02DDF is called.

Constraint: $LDQ \geq N$.

5: RSS – ***real*** *Input/Output*

On entry: either the residual sum of squares or a value less than or equal to 0.0 to indicate that the residual sum of squares is to be calculated by the routine.

On exit: if $RSS \leq 0.0$ on entry, then on exit RSS will contain the residual sum of squares as calculated by G02DDF. If RSS was positive on entry, then it will be unchanged.

6: IDF – INTEGER *Output*

On exit: the degrees of freedom associated with the residual sum of squares.

7: B(IP) – ***real*** array *Output*

On exit: the estimates of the p parameters, $\hat{\beta}$.

8: SE(IP) – ***real*** array *Output*

On exit: the standard errors of the p parameters given in B.

9: COV(IP*(IP+1)/2) – ***real*** array *Output*

On exit: the upper triangular part of the variance-covariance matrix of the p parameter estimates given in B. They are stored packed by column, i.e., the covariance between the parameter estimate given in $B(i)$ and the parameter estimate given in $B(j)$, $j \geq i$, is stored in $COV(j \times (j-1)/2 + i)$.

10: SVD – LOGICAL *Output*

On exit: if a singular value decomposition has been performed, then SVD = .TRUE., otherwise SVD = .FALSE..

11: IRANK – INTEGER *Output*

On exit: the rank of the independent variables.

If SVD = .FALSE., then IRANK = IP.

If SVD = .TRUE., then IRANK is an estimate of the rank of the independent variables.

IRANK is calculated as the number of singular values greater than TOL×(largest singular value). It is possible for the SVD to be carried out but IRANK to be returned as IP.

12: P(IP*IP+2*IP) – ***real*** array *Output*

On exit: P contains details of the singular value decomposition if used.

If SVD = .FALSE., P is not referenced.

If SVD = .TRUE., the first IP elements of P will not be referenced, the next IP values contain the singular values. The following IP × IP values contain the matrix P^* stored by columns.

13: TOL – ***real*** *Input*

On entry: the value of TOL is used to decide if the independent variables are of full rank and, if not, what is the rank of the independent variables. The smaller the value of TOL the stricter the criterion for selecting the singular value decomposition. If TOL = 0.0, then the singular value decomposition will never be used, this may cause run time errors or inaccuracies if the independent variables are not of full rank.

Suggested value: TOL = 0.000001.

Constraint: TOL $\geq$ 0.0.

14: WK(IP*IP+(IP−1)*5) – ***real*** array *Workspace*

15: 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 < 1,
or IP < 1,
or LDQ < IP,
or LDQ < N,
or TOL < 0.0.

IFAIL = 2

The degrees of freedom for error are less than or equal to 0. In this case the estimates of β are returned but not the standard errors or covariances.

IFAIL = 3

The singular value decomposition, if used, has failed to converge, see F02WEF. This is an unlikely error exit.

7 Accuracy

The accuracy of the results will depend on the accuracy of the input R matrix, which may lose accuracy if a large number of observations or variables have been dropped.

8 Further Comments

None.

9 Example

A data set consisting of 12 observations and four independent variables is input and a regression model fitted by calls to G02DEF. The parameters are then calculated by G02DDF and the results printed.

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.

```
*     G02DDF Example Program Text
*     Mark 14 Release.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          MMAX, NMAX
      PARAMETER        (MMAX=5,NMAX=12)
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*     .. Local Scalars ..
      real             RSS, TOL
      INTEGER          I, IDF, IFAIL, IP, IRANK, J, M, N
      LOGICAL          SVD
      CHARACTER        WEIGHT
*     .. Local Arrays ..
      real             B(MMAX), COV(MMAX*(MMAX+1)/2), P(MMAX*(MMAX+2)),
     +                 Q(NMAX,MMAX+1), SE(MMAX), WK(MMAX*MMAX+5*MMAX),
     +                 WT(NMAX), X(NMAX,MMAX)
*     .. External Subroutines ..
      EXTERNAL         G02DDF, G02DEF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G02DDF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N, M, WEIGHT
      IF (N.LE.NMAX .AND. M.LT.MMAX) THEN
         IF (WEIGHT.EQ.'W' .OR. WEIGHT.EQ.'w') THEN
            DO 20 I = 1, N
               READ (NIN,*) (X(I,J),J=1,M), Q(I,1), WT(I)
   20       CONTINUE
         ELSE
            DO 40 I = 1, N
               READ (NIN,*) (X(I,J),J=1,M), Q(I,1)
   40       CONTINUE
         END IF
*        Set tolerance
         TOL = 0.000001e0
         IP = 0
```

```
         DO 60 I = 1, M
            IFAIL = -1
*
*           Fit model using G02DEF
            CALL G02DEF(WEIGHT,N,IP,Q,NMAX,P,WT,X(1,I),RSS,TOL,IFAIL)
*
            IF (IFAIL.EQ.0) THEN
               IP = IP + 1
            ELSE IF (IFAIL.EQ.3) THEN
               WRITE (NOUT,*) ' * New variable not added *'
            ELSE
               GO TO 100
            END IF
   60    CONTINUE
         RSS = 0.0e0
         IFAIL = 0
*
         CALL G02DDF(N,IP,Q,NMAX,RSS,IDF,B,SE,COV,SVD,IRANK,P,TOL,WK,
     +               IFAIL)
*
         WRITE (NOUT,*)
         IF (SVD) THEN
            WRITE (NOUT,*) 'Model not of full rank'
            WRITE (NOUT,*)
         END IF
         WRITE (NOUT,99999) 'Residual sum of squares = ', RSS
         WRITE (NOUT,99998) 'Degrees of freedom = ', IDF
         WRITE (NOUT,*)
         WRITE (NOUT,*) 'Variable   Parameter estimate   Standard error'
         WRITE (NOUT,*)
         DO 80 J = 1, IP
            WRITE (NOUT,99997) J, B(J), SE(J)
   80    CONTINUE
      END IF
  100 CONTINUE
      STOP
*
99999 FORMAT (1X,A,e12.4)
99998 FORMAT (1X,A,I4)
99997 FORMAT (1X,I6,2e20.4)
      END
```

9.2 Program Data

```
G02DDF Example Program Data
 12 4 'U'
1.0 0.0 0.0 0.0 33.63
0.0 0.0 0.0 1.0 39.62
0.0 1.0 0.0 0.0 38.18
0.0 0.0 1.0 0.0 41.46
0.0 0.0 0.0 1.0 38.02
0.0 1.0 0.0 0.0 35.83
0.0 0.0 0.0 1.0 35.99
1.0 0.0 0.0 0.0 36.58
0.0 0.0 1.0 0.0 42.92
1.0 0.0 0.0 0.0 37.80
0.0 0.0 1.0 0.0 40.43
0.0 1.0 0.0 0.0 37.89
```

9.3 Program Results

```
G02DDF Example Program Results

Residual sum of squares =   0.2223E+02
Degrees of freedom =    8

Variable   Parameter estimate   Standard error

     1         0.3600E+02          0.9623E+00
     2         0.3730E+02          0.9623E+00
     3         0.4160E+02          0.9623E+00
     4         0.3788E+02          0.9623E+00
```