NAG Library Routine Document

G02DCF

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

G02DCF adds or deletes an observation from a general regression model fitted by G02DAF.

2 Specification

```
SUBROUTINE G02DCF (UPDATE, MEAN, WEIGHT, M, ISX, Q, LDQ, IP, X, IX, Y, WT,      &
                   RSS, WK, IFAIL)

INTEGER             M, ISX(M), LDQ, IP, IX, IFAIL
REAL (KIND=nag_wp) Q(LDQ,IP+1), X(*), Y, WT, RSS, WK(3*IP)
CHARACTER(1)        UPDATE, MEAN, WEIGHT
```

3 Description

G02DAF fits a general linear regression model to a dataset. You may wish to change the model by either adding or deleting an observation from the dataset. G02DCF takes the results from G02DAF and makes the required changes to the vector c and the upper triangular matrix R produced by G02DAF. The regression coefficients, standard errors and the variance-covariance matrix of the regression coefficients can be obtained from G02DDF after all required changes to the dataset have been made.

G02DAF performs a QR decomposition on the (weighted) X matrix of independent variables. To add a new observation to a model with p parameters, the upper triangular matrix R and vector c_1 (the first p elements of c) are augmented by the new observation on independent variables in x^{T} and dependent variable y_{new}. Givens rotations are then used to restore the upper triangular form.

$$\begin{pmatrix} R : c_1 \\ x : y_{\mathrm{new}} \end{pmatrix} \rightarrow \begin{pmatrix} R^* : c_1^* \\ 0 : y_{\mathrm{new}}^* \end{pmatrix}.$$

Note: only R and the upper part of c are updated the remainder of the Q matrix is unchanged.

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

5 Parameters

1: UPDATE – CHARACTER(1) *Input*

On entry: indicates if an observation is to be added or deleted.

UPDATE = 'A'
The observation is added.

UPDATE = 'D'
The observation is deleted.

Constraint: UPDATE = 'A' or 'D'.

2: MEAN – CHARACTER(1) *Input*

On entry: indicates if a mean has been used in the model.

MEAN = 'M'
A mean term or intercept will have been included in the model by G02DAF.

MEAN = 'Z'
A model with no mean term or intercept will have been fitted by G02DAF.

Constraint: MEAN = 'M' or 'Z'.

3: WEIGHT – CHARACTER(1) *Input*

On entry: indicates if a weight is to be used.

WEIGHT = 'U'
The new observation is unweighted.

WEIGHT = 'W'
The new observation is to be weighted and the weight must be supplied in WT.

Constraint: WEIGHT = 'U' or 'W'.

4: M – INTEGER *Input*

On entry: m, the total number of independent variables in the dataset.

Constraint: M ≥ 1.

5: ISX(M) – INTEGER array *Input*

On entry: if ISX(j) is greater than 0, the value contained in X($(j-1) \times \mathrm{IX}+1$) is to be included as a value of x^{T}, for $j = 1, 2, \ldots, \mathrm{M}$.

Constraint: if MEAN = 'M', exactly IP $-$ 1 elements of ISX must be > 0 and if MEAN = 'Z', exactly IP elements of ISX must be > 0.

6: Q(LDQ,IP + 1) – REAL (KIND=nag_wp) array *Input/Output*

On entry: must be array Q as output by G02DAF, G02DEF, G02DFF or G02EEF, or a previous call to G02DCF.

On exit: the first IP elements of the first column of Q will contain c_1^* the upper triangular part of columns 2 to IP + 1 will contain R^* the remainder is unchanged.

7: LDQ – INTEGER *Input*

On entry: the first dimension of the array Q as declared in the (sub)program from which G02DCF is called.

Constraint: LDQ $\geq$ IP.

8: IP – INTEGER *Input*

On entry: the number of linear terms in general linear regression model (including mean if there is one).

Constraint: IP ≥ 1.

9: X($*$) – REAL (KIND=nag_wp) array *Input*

Note: the dimension of the array X must be at least $(\mathrm{M}-1) \times \mathrm{IX}+1$.

On entry: the IP values for the dependent variables of the new observation, x^{T}. The positions will depend on the value of IX.

10: IX – INTEGER *Input*

On entry: the increment for elements of X. Two situations are common:

IX = 1
The values of x are to be chosen from consecutive locations in X, i.e., $\mathrm{X}(1), \mathrm{X}(2), \ldots, \mathrm{X}(\mathrm{M})$.

IX = LDX
The values of x are to be chosen from a row of a two-dimensional array with first dimension LDX, i.e., $\mathrm{X}(1), \mathrm{X}(\mathrm{LDX}+1), \ldots, \mathrm{X}((\mathrm{M}-1)\mathrm{LDX}+1)$.

Constraint: IX ≥ 1.

11: Y – REAL (KIND=nag_wp) *Input*

On entry: the value of the dependent variable for the new observation, y_{new}.

12: WT – REAL (KIND=nag_wp) *Input*

On entry: if WEIGHT = 'W' , WT must contain the weight to be used with the new observation.

If WT = 0.0, the observation is not included in the model.

If WEIGHT = 'U', WT is not referenced.

Constraint: if WT ≥ 0.0, WEIGHT = 'W'.

13: RSS – REAL (KIND=nag_wp) *Input/Output*

On entry: the value of the residual sums of squares for the original set of observations.

Constraint: RSS ≥ 0.0.

On exit: the updated values of the residual sums of squares.

Note: this will only be valid if the model is of full rank.

14: WK($3 \times$ IP) – REAL (KIND=nag_wp) array *Workspace*

15: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction for details.

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, if you are 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.**

On exit: IFAIL = 0 unless the routine detects an error or a warning has been flagged (see Section 6).

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, IP < 1,
or LDQ $<$ IP,
or M < 1,
or IX < 1,
or RSS < 0.0,

or UPDATE $\neq$ 'A' or 'D',
or MEAN $\neq$ 'M' or 'Z',
or WEIGHT $\neq$ 'U' or 'W',
or MEAN = 'M' and there are not exactly IP $-$ 1 nonzero values of ISX,
or MEAN = 'Z' and there are not exactly IP nonzero values of ISX,

IFAIL = 2

On entry, WEIGHT = 'W' and WT $<$ 0.0.

IFAIL = 3

The R matrix could not be updated. This may occur if an attempt is made to delete an observation which was not in the original dataset or to add an observation to a R matrix with a zero diagonal element. This error is also possible when removing an observation which reduces the rank of design matrix. In such cases the model should be recomputed using G02DAF.

IFAIL = 4

The residual sums of squares cannot be updated. This will occur if the input residual sum of squares is less than the calculated decrease in residual sum of squares when the new observation is deleted.

7 Accuracy

Higher accuracy is achieved by updating the R matrix rather than the traditional methods of updating $X'X$.

8 Further Comments

Care should be taken with the use of G02DCF.

(a) It is possible to delete observations which were not included in the original model.

(b) If several additions/deletions have been performed you are advised to recompute the regression using G02DAF.

(c) Adding or deleting observations can alter the rank of the model. Such changes will only be detected when a call to G02DDF has been made. G02DDF should also be used to compute the new residual sum of squares when the model is not of full rank.

G02DCF may also be used after G02DEF, G02DFF and G02EEF.

9 Example

A dataset consisting of 12 observations with four independent variables is read in and a general linear regression model fitted by G02DAF and parameter estimates printed. The last observation is then dropped and the parameter estimates recalculated, using G02DDF, and printed. Finally a new observation is added and new parameter estimates computed and printed.

9.1 Program Text

```
    Program g02dcfe

!     G02DCF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: g02daf, g02dcf, g02ddf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter                :: nin = 5, nout = 6
!     .. Local Scalars ..
```

```
      Real (Kind=nag_wp)              :: rss, tol, wt, y
      Integer                         :: i, idf, ifail, ip, irank, ix, ldq,   &
                                         ldxm, lwt, m, n
      Logical                         :: svd
      Character (1)                   :: mean, update, weight
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable :: b(:), cov(:), h(:), p(:), q(:,:),    &
                                         res(:), se(:), wk(:), wtm(:), x(:),  &
                                         xm(:,:), ym(:)
      Integer, Allocatable            :: isx(:)
!     .. Intrinsic Procedures ..
      Intrinsic                       :: count
!     .. Executable Statements ..
      Write (nout,*) 'G02DCF Example Program Results'
      Write (nout,*)

!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n, m, weight, mean

      If (weight=='W' .Or. weight=='w') Then
        lwt = n
      Else
        lwt = 0
      End If
      ldxm = n
      Allocate (xm(ldxm,m),ym(n),wtm(lwt),isx(m),x(m))

!     Read in data
      If (lwt>0) Then
        Read (nin,*)(xm(i,1:m),ym(i),wtm(i),i=1,n)
      Else
        Read (nin,*)(xm(i,1:m),ym(i),i=1,n)
      End If

!     Read in variable inclusion flags
      Read (nin,*) isx(1:m)

!     Calculate IP
      ip = count(isx>0)
      If (mean=='M' .Or. mean=='m') Then
        ip = ip + 1
      End If

      ldq = n
      Allocate (b(ip),cov((ip*ip+ip)/2),h(n),p(ip*(ip+ &
        2)),q(ldq,ip+1),res(n),se(ip),wk(ip*ip+5*(ip-1)))

!     Use suggested value for tolerance
      tol = 0.000001E0_nag_wp

!     Fit initial model using G02DAF
      ifail = 0
      Call g02daf(mean,weight,n,xm,ldxm,m,isx,ip,ym,wtm,rss,idf,b,se,cov,res, &
        h,q,ldq,svd,irank,p,tol,wk,ifail)

!     Display results from initial model fit
      Write (nout,*) 'Results from initial model fit using G02DAF'
      If (svd) Then
        Write (nout,*)
        Write (nout,*) 'Model not of full rank'
      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,*)
      Write (nout,99997)(i,b(i),se(i),i=1,ip)

!     Updating data is held in X consecutively
      ix = 1
```

```
!     Add or delete observations supplied in the data file
u_lp: Do
        Read (nin,*,Iostat=ifail) update
        If (ifail/=0) Then
          Exit u_lp
        End If

        If (lwt>0) Then
          Read (nin,*) x(1:m), y, wt
        Else
          Read (nin,*) x(1:m), y
        End If

!       Update the regression
        ifail = 0
        Call g02dcf(update,mean,weight,m,isx,q,ldq,ip,x,ix,y,wt,rss,wk,ifail)

!       Display titles and update observation count
        Write (nout,*)
        Select Case (update)
        Case ('a','A')
          Write (nout,*) 'Results from adding an observation using G02DCF'
          n = n + 1
        Case ('d','D')
          Write (nout,*) 'Results from dropping an observation using G02DCF'
          n = n - 1
        Case Default
          Write (nout,*) 'Unknown update flag read in from data file'
          Go To 100
        End Select

!       Recalculate the parameter estimates etc
        ifail = 0
        Call g02ddf(n,ip,q,ldq,rss,idf,b,se,cov,svd,irank,p,tol,wk,ifail)

!       Display updated results
        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,*)
        Write (nout,99997)(i,b(i),se(i),i=1,ip)
      End Do u_lp

100   Continue

99999 Format (1X,A,E12.4)
99998 Format (1X,A,I4)
99997 Format (1X,I6,2E20.4)
    End Program g02dcfe
```

9.2 Program Data

```
G02DCF Example Program Data
 12 4  'U' 'Z'                :: N, M, MEAN, WEIGHT
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
1.0 1.0 1.0 1.0 37.89        :: End of X, Y
```

```
 1   1   1   1              :: ISX
'D'                         :: UPDATE (delete observation)
1.0 1.0 1.0 1.0 37.89       :: X and Y for observation to be deleted
'A'                         :: UPDATE (add observation)
0.0 1.0 0.0 0.0 37.89       :: X and Y for observation to be added
```

9.3 Program Results

```
 G02DCF Example Program Results

 Results from initial model fit using G02DAF
 Residual sum of squares =   0.5275E+04
 Degrees of freedom =    8

 Variable   Parameter estimate   Standard error

      1          0.2072E+02          0.1380E+02
      2          0.1409E+02          0.1624E+02
      3          0.2632E+02          0.1380E+02
      4          0.2260E+02          0.1380E+02

 Results from dropping an observation using G02DCF
 Residual sum of squares =   0.2170E+02
 Degrees of freedom =    7

 Variable   Parameter estimate   Standard error

      1          0.3600E+02          0.1017E+01
      2          0.3701E+02          0.1245E+01
      3          0.4160E+02          0.1017E+01
      4          0.3788E+02          0.1017E+01

 Results from adding an observation using G02DCF
 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
```