NAG Library Routine Document

G02EAF

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

G02EAF calculates the residual sums of squares for all possible linear regressions for a given set of independent variables.

2 Specification

```
SUBROUTINE G02EAF (MEAN, WEIGHT, N, M, X, LDX, VNAME, ISX, Y, WT, NMOD,        &
                   MODL, LDMODL, RSS, NTERMS, MRANK, WK, IFAIL)

INTEGER            N, M, LDX, ISX(M), NMOD, LDMODL, NTERMS(LDMODL),            &
                   MRANK(LDMODL), IFAIL
REAL (KIND=nag_wp) X(LDX,M), Y(N), WT(*), RSS(LDMODL), WK(N*(M+1))
CHARACTER(*)       VNAME(M), MODL(LDMODL,M)
CHARACTER(1)       MEAN, WEIGHT
```

3 Description

For a set of k possible independent variables there are 2^k linear regression models with from zero to k independent variables in each model. For example if $k = 3$ and the variables are A, B and C then the possible models are:

(i) null model

(ii) A

(iii) B

(iv) C

(v) A and B

(vi) A and C

(vii) B and C

(viii) A, B and C.

G02EAF calculates the residual sums of squares from each of the 2^k possible models. The method used involves a QR decomposition of the matrix of possible independent variables. Independent variables are then moved into and out of the model by a series of Givens rotations and the residual sums of squares computed for each model; see Clark (1981) and Smith and Bremner (1989).

The computed residual sums of squares are then ordered first by increasing number of terms in the model, then by decreasing size of residual sums of squares. So the first model will always have the largest residual sum of squares and the 2^kth will always have the smallest. This aids you in selecting the best possible model from the given set of independent variables.

G02EAF allows you to specify some independent variables that must be in the model, the forced variables. The other independent variables from which the possible models are to be formed are the free variables.

4 References

Clark M R B (1981) A Givens algorithm for moving from one linear model to another without going back to the data *Appl. Statist.* **30** 198–203

Smith D M and Bremner J M (1989) All possible subset regressions using the QR decomposition *Comput. Statist. Data Anal.* **7** 217–236

Weisberg S (1985) *Applied Linear Regression* Wiley

5 Parameters

1: MEAN – CHARACTER(1) *Input*

On entry: indicates if a mean term is to be included.

MEAN = 'M'
A mean term, intercept, will be included in the model.

MEAN = 'Z'
The model will pass through the origin, zero-point.

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

2: WEIGHT – CHARACTER(1) *Input*

On entry: indicates if weights are to be used.

WEIGHT = 'U'
Least squares estimation is used.

WEIGHT = 'W'
Weighted least squares is used and weights must be supplied in array WT.

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

3: N – INTEGER *Input*

On entry: n, the number of observations.

Constraints:

N ≥ 2;
N $\geq m$, is the number of independent variables to be considered (forced plus free plus mean if included), as specified by MEAN and ISX.

4: M – INTEGER *Input*

On entry: the number of variables contained in X.

Constraint: M ≥ 2.

5: X(LDX,M) – REAL (KIND=nag_wp) array *Input*

On entry: X(i,j) must contain the ith observation for the jth independent variable, for $i = 1,2,\ldots,$N and $j = 1,2,\ldots,$M.

6: LDX – INTEGER *Input*

On entry: the first dimension of the array X as declared in the (sub)program from which G02EAF is called.

Constraint: LDX $\geq$ N.

7: VNAME(M) – CHARACTER(*) array *Input*

On entry: VNAME(j) must contain the name of the variable in column j of X, for $j = 1,2,\ldots,$M.

8: ISX(M) – INTEGER array *Input*

On entry: indicates which independent variables are to be considered in the model.

ISX(j) ≥ 2
The variable contained in the jth column of X is included in all regression models, i.e., is a forced variable.

ISX(j) $= 1$
The variable contained in the jth column of X is included in the set from which the regression models are chosen, i.e., is a free variable.

ISX(j) $= 0$
The variable contained in the jth column of X is not included in the models.

Constraints:

ISX(j) ≥ 0, for $j = 1, 2, \ldots, \mathrm{M}$;
at least one value of ISX $= 1$.

9: Y(N) – REAL (KIND=nag_wp) array *Input*

On entry: Y(i) must contain the ith observation on the dependent variable, y_i, for $i = 1, 2, \ldots, n$.

10: WT($*$) – REAL (KIND=nag_wp) array *Input*

Note: the dimension of the array WT must be at least N if WEIGHT = 'W'.

On entry: if WEIGHT = 'W', WT must contain the weights to be used in the weighted regression.

If WT(i) = 0.0, the ith observation is not included in the model, in which case the effective number of observations is the number of observations with nonzero weights.

If WEIGHT = 'U', WT is not referenced and the effective number of observations is N.

Constraint: if WEIGHT = 'W', WT(i) ≥ 0.0, for $i = 1, 2, \ldots, n$.

11: NMOD – INTEGER *Output*

On exit: the total number of models for which residual sums of squares have been calculated.

12: MODL(LDMODL,M) – CHARACTER(*) array *Output*

On exit: the first NTERMS(i) elements of the ith row of MODL contain the names of the independent variables, as given in VNAME, that are included in the ith model.

Constraint: the length of MODL should be greater or equal to the length of VNAME.

13: LDMODL – INTEGER *Input*

On entry: the first dimension of the array MODL and the dimension of the arrays RSS, NTERMS and MRANK as declared in the (sub)program from which G02EAF is called.

Constraints:

LDMODL $\geq$ M;
LDMODL $\geq 2^k$, k is the number of free variables in the model as specified in ISX, and hence 2^k is the total number of models to be generated.

14: RSS(LDMODL) – REAL (KIND=nag_wp) array *Output*

On exit: RSS(i) contains the residual sum of squares for the ith model, for $i = 1, 2, \ldots,$ NMOD.

15: NTERMS(LDMODL) – INTEGER array *Output*

On exit: NTERMS(i) contains the number of independent variables in the ith model, not including the mean if one is fitted, for $i = 1, 2, \ldots,$ NMOD.

16: MRANK(LDMODL) – INTEGER array *Output*

On exit: MRANK(i) contains the rank of the residual sum of squares for the ith model.

17: WK(N × (M + 1)) – REAL (KIND=nag_wp) array *Workspace*

18: 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, N < 2,
or M < 2,
or LDX < N,
or LDMODL < M,
or MEAN ≠ 'M' or 'Z',
or WEIGHT ≠ 'U' or 'W'.

IFAIL = 2

On entry, WEIGHT = 'W' and a value of WT < 0.0.

IFAIL = 3

On entry, a value of ISX < 0,
or there are no free variables, i.e., no element of ISX = 1.

IFAIL = 4

On entry, LDMODL < the number of possible models $= 2^k$, where k is the number of free independent variables from ISX.

IFAIL = 5

On entry, the number of independent variables to be considered (forced plus free plus mean if included) is greater or equal to the effective number of observations.

IFAIL = 6

The full model is not of full rank, i.e., some of the independent variables may be linear combinations of other independent variables. Variables must be excluded from the model in order to give full rank.

7 Accuracy

For a discussion of the improved accuracy obtained by using a method based on the QR decomposition see Smith and Bremner (1989).

8 Further Comments

G02ECF may be used to compute R^2 and C_p-values from the results of G02EAF.

If a mean has been included in the model and no variables are forced in then RSS(1) contains the total sum of squares and in many situations a reasonable estimate of the variance of the errors is given by RSS(NMOD)/(N − 1 − NTERMS(NMOD)).

9 Example

The data for this example is given in Weisberg (1985). The independent variables and the dependent variable are read, as are the names of the variables. These names are as given in Weisberg (1985). The residual sums of squares computed and printed with the names of the variables in the model.

9.1 Program Text

```
    Program g02eafe

!     G02EAF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: g02eaf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6, vnlen = 3
!     .. Local Scalars ..
      Integer                          :: i, ifail, k, ldmodl, ldx, lwt, m, n, &
                                          nmod
      Character (1)                    :: mean, weight
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: rss(:), wk(:), wt(:), x(:,:), y(:)
      Integer, Allocatable             :: isx(:), mrank(:), nterms(:)
      Character (vnlen), Allocatable   :: modl(:,:), vname(:)
!     .. Intrinsic Procedures ..
      Intrinsic                        :: count, max
!     .. Executable Statements ..
      Write (nout,*) 'G02EAF Example Program Results'
      Write (nout,*)

!     Skip heading in data file
      Read (nin,*)

!     Read in the problem size
      Read (nin,*) n, m, mean, weight

      If (weight=='W' .Or. weight=='w') Then
        lwt = n
      Else
        lwt = 0
      End If
      ldx = n
      Allocate (x(ldx,m),vname(m),isx(m),y(n),wt(lwt))

!     Read in data
      If (lwt>0) Then
        Read (nin,*)(x(i,1:m),y(i),wt(i),i=1,n)
      Else
        Read (nin,*)(x(i,1:m),y(i),i=1,n)
```

```
      End If

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

!     Read in first VNLEN characters of the variable names
      Read (nin,*) vname(1:m)

!     Calculate the number of free variables
      k = count(isx(1:m)==1)

      ldmodl = max(m,2**k)
      Allocate (modl(ldmodl,m),rss(ldmodl),nterms(ldmodl),mrank(ldmodl),wk(n*( &
        m+1)))

!     Calculate residual sums of squares for all possible models
      ifail = 0
      Call g02eaf(mean,weight,n,m,x,ldx,vname,isx,y,wt,nmod,modl,ldmodl,rss, &
        nterms,mrank,wk,ifail)

!     Display results
      Write (nout,*) 'Number of     RSS    RANK  MODL'
      Write (nout,*) 'parameters'
      Do i = 1, nmod
        Write (nout,99999) nterms(i), rss(i), mrank(i), modl(i,1:nterms(i))
      End Do

99999 Format (1X,I8,F11.4,I4,3X,5(1X,A))
    End Program g02eafe
```

9.2 Program Data

```
G02EAF Example Program Data
  20 6 'M' 'U'                                        :: N,M,MEAN,WEIGHT
   0.0 1125.0 232.0 7160.0 85.9 8905.0  1.5563
   7.0  920.0 268.0 8804.0 86.5 7388.0  0.8976
  15.0  835.0 271.0 8108.0 85.2 5348.0  0.7482
  22.0 1000.0 237.0 6370.0 83.8 8056.0  0.7160
  29.0 1150.0 192.0 6441.0 82.1 6960.0  0.3010
  37.0  990.0 202.0 5154.0 79.2 5690.0  0.3617
  44.0  840.0 184.0 5896.0 81.2 6932.0  0.1139
  58.0  650.0 200.0 5336.0 80.6 5400.0  0.1139
  65.0  640.0 180.0 5041.0 78.4 3177.0 -0.2218
  72.0  583.0 165.0 5012.0 79.3 4461.0 -0.1549
  80.0  570.0 151.0 4825.0 78.7 3901.0  0.0000
  86.0  570.0 171.0 4391.0 78.0 5002.0  0.0000
  93.0  510.0 243.0 4320.0 72.3 4665.0 -0.0969
 100.0  555.0 147.0 3709.0 74.9 4642.0 -0.2218
 107.0  460.0 286.0 3969.0 74.4 4840.0 -0.3979
 122.0  275.0 198.0 3558.0 72.5 4479.0 -0.1549
 129.0  510.0 196.0 4361.0 57.7 4200.0 -0.2218
 151.0  165.0 210.0 3301.0 71.8 3410.0 -0.3979
 171.0  244.0 327.0 2964.0 72.5 3360.0 -0.5229
 220.0   79.0 334.0 2777.0 71.9 2599.0 -0.0458   :: End of X, Y
  0     1    1      1     1    1                 :: ISX
 'DAY' 'BOD' 'TKN' 'TS ' 'TVS' 'COD'              :: VNAME
```

9.3 Program Results

```
 G02EAF Example Program Results

 Number of     RSS    RANK  MODL
 parameters
        0     5.0634  32
        1     5.0219  31    TKN
        1     2.5044  30    TVS
        1     2.0338  28    BOD
        1     1.5563  25    COD
        1     1.5370  24    TS
        2     2.4381  29    TKN TVS
```

```
 2     1.7462  27    BOD TVS
 2     1.5921  26    BOD TKN
 2     1.4963  23    BOD COD
 2     1.4707  22    TKN TS
 2     1.4590  21    TS  TVS
 2     1.4397  20    BOD TS
 2     1.4388  19    TKN COD
 2     1.3287  15    TVS COD
 2     1.0850   8    TS  COD
 3     1.4257  18    BOD TKN TVS
 3     1.3900  17    TKN TS  TVS
 3     1.3894  16    BOD TS  TVS
 3     1.3204  14    BOD TVS COD
 3     1.2764  13    BOD TKN COD
 3     1.2582  12    BOD TKN TS
 3     1.2179  10    TKN TVS COD
 3     1.0644   7    BOD TS  COD
 3     1.0634   6    TS  TVS COD
 3     0.9871   4    TKN TS  COD
 4     1.2199  11    BOD TKN TS  TVS
 4     1.1565   9    BOD TKN TVS COD
 4     1.0388   5    BOD TS  TVS COD
 4     0.9871   3    BOD TKN TS  COD
 4     0.9653   2    TKN TS  TVS COD
 5     0.9652   1    BOD TKN TS  TVS COD
```