NAG Fortran Library Routine Document

G02HKF

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

G02HKF computes a robust estimate of the covariance matrix for an expected fraction of gross errors.

2 Specification

```
 SUBROUTINE G02HKF(N, M, X, LDX, EPS, COV, THETA, MAXIT, NITMON, TOL,
1                  NIT, WK, IFAIL)
 INTEGER           N, M, LDX, MAXIT, NITMON, NIT, IFAIL
 real              X(LDX,M), EPS, COV(M*(M+1)/2), THETA(M), TOL,
1                  WK(N+M*(M+5)/2)
```

3 Description

For a set of n observations on m variables in a matrix X, a robust estimate of the covariance matrix, C, and a robust estimate of location, θ, are given by

$$C = \tau^2 (A^T A)^{-1},$$

where τ^2 is a correction factor and A is a lower triangular matrix found as the solution to the following equations:

$$z_i = A(x_i - \theta),$$

$$\frac{1}{n}\sum_{i=1}^{n} w(\|z_i\|_2) z_i = 0,$$

and

$$\frac{1}{n}\sum_{i=1}^{n} u(\|z_i\|_2) z_i z_i^T - I = 0,$$

where x_i is a vector of length m containing the elements of the ith row of X,
z_i is a vector of length m,
I is the identity matrix and 0 is the zero matrix,
and w and u are suitable functions.

G02HKF uses weight functions:

$$u(t) = \frac{a_u}{t^2}, \quad \text{if } t < a_u^2$$

$$u(t) = 1, \quad \text{if } a_u^2 \leq t \leq b_u^2$$

$$u(t) = \frac{b_u}{t^2}, \quad \text{if } t > b_u^2$$

and

$$w(t) = 1, \quad \text{if } t \leq c_w$$

$$w(t) = \frac{c_w}{t}, \quad \text{if } t > c_w$$

for constants a_u, b_u and c_w.

These functions solve a minimax problem considered by Huber (see Huber (1981)). The values of a_u, b_u and c_w are calculated from the expected fraction of gross errors, ϵ (see Huber (1981) and Marazzi (1987a)). The expected fraction of gross errors is the estimated proportion of outliers in the sample.

In order to make the estimate asymptotically unbiased under a Normal model a correction factor, τ^2, is calculated, (see Huber (1981) and Marazzi (1987a)).

The matrix C is calculated using G02HLF. Initial estimates of θ_j, for $j = 1, 2, \ldots, m$, are given by the median of the jth column of X and the initial value of A is based on the median absolute deviation (see Marazzi (1987a)). G02HKF is based on routines in ROBETH; see Marazzi (1987a).

4 References

Huber P J (1981) *Robust Statistics* Wiley

Marazzi A (1987a) Weights for bounded influence regression in ROBETH *Cah. Rech. Doc. IUMSP, No. 3 ROB 3* Institut Universitaire de Médecine Sociale et Préventive, Lausanne

5 Parameters

1: N – INTEGER *Input*

On entry: the number of observations, n.

Constraint: N > 1.

2: M – INTEGER *Input*

On entry: the number of columns of the matrix X, i.e., number of independent variables, m.

Constraint: $1 \leq$ M $\leq$ N.

3: X(LDX,M) – ***real*** array *Input*

On entry: X(i, j) must contain the ith observation for the jth variable, for $i = 1, 2, \ldots, n$; $j = 1, 2, \ldots, m$.

4: LDX – INTEGER *Input*

On entry: the first dimension of the array X as declared in the (sub)program from which G02HKF is called.

Constraint: LDX $\geq$ N.

5: EPS – ***real*** *Input*

On entry: the expected fraction of gross errors expected in the sample, ϵ.

Constraint: $0.0 \leq$ EPS < 1.0.

6: COV(M*(M+1)/2) – ***real*** array *Output*

On exit: a robust estimate of the covariance matrix, C. The upper triangular part of the matrix C is stored packed by columns. C_{ij} is returned in COV$(j \times (j-1)/2 + i)$, $i \leq j$.

7: THETA(M) – ***real*** array *Output*

On exit: the robust estimate of the location parameters θ_j, for $j = 1, 2, \ldots, m$.

8: MAXIT – INTEGER *Input*

On entry: the maximum number of iterations that will be used during the calculation of the covariance matrix.

Suggested value: 150.

Constraint: MAXIT > 0.

9: NITMON – INTEGER *Input*

On entry: indicates the amount of information on the iteration that is printed.

If NITMON > 0, then the value of A, θ and δ (see Section 7) will be printed at the first and every NITMON iterations.

If NITMON ≤ 0, then no iteration monitoring is printed.

When printing occurs the output is directed to the current advisory message unit (see X04ABF).

10: TOL – ***real*** *Input*

On entry: the relative precision for the final estimates of the covariance matrix.

Constraint: TOL > 0.0.

11: NIT – INTEGER *Output*

On exit: the number of iterations performed.

12: WK(N+M*(M+5)/2) – ***real*** array *Workspace*

13: 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 M < 1,
or N $<$ M,
or LDX $<$ N,
or EPS < 0.0,
or EPS ≥ 1.0,
or TOL ≤ 0.0,
or MAXIT ≤ 0.

IFAIL $= 2$

On entry, a variable has a constant value, i.e., all elements in a column of X are identical.

IFAIL = 3

The iterative procedure to find C has failed to converge in MAXIT iterations.

IFAIL = 4

The iterative procedure to find C has become unstable. This may happen if the value of EPS is too large for the sample.

7 Accuracy

On successful exit the accuracy of the results is related to the value of TOL; see Section 5. At an iteration let

(i) $d1 =$ the maximum value of the absolute relative change in A

(ii) $d2 =$ the maximum absolute change in $u(\|z_i\|_2)$

(iii) $d3 =$ the maximum absolute relative change in θ_j

and let $\delta = \max(d1, d2, d3)$. Then the iterative procedure is assumed to have converged when $\delta <$ TOL.

8 Further Comments

The existence of A, and hence c, will depend upon the function u (see Marazzi (1987a)); also if X is not of full rank a value of A will not be found. If the columns of X are almost linearly related, then convergence will be slow.

9 Example

A sample of 10 observations on three variables is read in and the robust estimate of the covariance matrix is computed assuming 10% gross errors are to be expected. The robust covariance is then 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.

```
*     G02HKF Example Program Text
*     Mark 14 Release.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          NMAX, MMAX, LDX
      PARAMETER        (NMAX=20,MMAX=5,LDX=NMAX)
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*     .. Local Scalars ..
      real             EPS, TOL
      INTEGER          I, IFAIL, J, K, L1, L2, M, MAXIT, N, NIT, NITMON
*     .. Local Arrays ..
      real             COV(MMAX*(MMAX+1)/2), THETA(MMAX),
     +                 WK(2*MMAX+NMAX+MMAX*(MMAX+1)/2), X(LDX,MMAX)
*     .. External Subroutines ..
      EXTERNAL         G02HKF, X04ABF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G02HKF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      CALL X04ABF(1,NOUT)
*     Read in the dimensions of X
      READ (NIN,*) N, M
      IF ((N.LE.NMAX) .AND. (M.LE.MMAX)) THEN
*        Read in the X matrix
         DO 20 I = 1, N
            READ (NIN,*) (X(I,J),J=1,M)
   20    CONTINUE
*        Read in value of eps
```

```
         READ (NIN,*) EPS
*        Set up remaining parameters
         MAXIT = 100
         TOL = 0.5e-4
*        Set NITMON to positive value for iteration monitoring
         NITMON = 0
         IFAIL = 0
*
         CALL G02HKF(N,M,X,LDX,EPS,COV,THETA,MAXIT,NITMON,TOL,NIT,WK,
     +               IFAIL)
*
         WRITE (NOUT,*)
         WRITE (NOUT,99999) 'G02HKF required ', NIT,
     +     ' iterations to converge'
         WRITE (NOUT,*)
         WRITE (NOUT,*) 'Covariance matrix'
         L2 = 0
         DO 40 J = 1, M
            L1 = L2 + 1
            L2 = L2 + J
            WRITE (NOUT,99998) (COV(K),K=L1,L2)
   40    CONTINUE
         WRITE (NOUT,*)
         WRITE (NOUT,*) 'THETA'
         DO 60 J = 1, M
            WRITE (NOUT,99997) THETA(J)
   60    CONTINUE
      END IF
      STOP
*
99999 FORMAT (1X,A,I4,A)
99998 FORMAT (1X,6F10.3)
99997 FORMAT (1X,F10.3)
      END
```

9.2 Program Data

```
G02HKF Example Program Data
    10    3                    : N  M
  3.4  6.9  12.2               : X1  X2  X3
  6.4  2.5  15.1
  4.9  5.5  14.2
  7.3  1.9  18.2
  8.8  3.6  11.7
  8.4  1.3  17.9
  5.3  3.1  15.0
  2.7  8.1   7.7
  6.1  3.0  21.9
  5.3  2.2  13.9               : End of X1 X2 and X3 values
    0.1                        : EPS
```

9.3 Program Results

```
 G02HKF Example Program Results

 G02HKF required   23 iterations to converge

 Covariance matrix
      3.461
     -3.681     5.348
      4.682    -6.645    14.439

 THETA
      5.818
      3.681
     15.037
```