NAG Library Routine Document

G07DCF

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

G07DCF computes an M-estimate of location with (optional) simultaneous estimation of scale, where you provide the weight functions.

2 Specification

```
SUBROUTINE G07DCF (CHI, PSI, ISIGMA, N, X, BETA, THETA, SIGMA, MAXIT, TOL,      &
                   RS, NIT, WRK, IFAIL)

INTEGER          ISIGMA, N, MAXIT, NIT, IFAIL
REAL (KIND=nag_wp) CHI, PSI, X(N), BETA, THETA, SIGMA, TOL, RS(N), WRK(N)
EXTERNAL         CHI, PSI
```

3 Description

The data consists of a sample of size n, denoted by $x_1, x_2, \ldots, x_n$, drawn from a random variable X.

The x_i are assumed to be independent with an unknown distribution function of the form,

$$F((x_i - \theta)/\sigma)$$

where θ is a location parameter, and σ is a scale parameter. M-estimators of θ and σ are given by the solution to the following system of equations;

$$\sum_{i=1}^{n} \psi\left(\left(x_i - \hat{\theta}\right)/\hat{\sigma}\right) = 0$$

$$\sum_{i=1}^{n} \chi\left(\left(x_i - \hat{\theta}\right)/\hat{\sigma}\right) = (n-1)\beta$$

where ψ and χ are user-supplied weight functions, and β is a constant. Optionally the second equation can be omitted and the first equation is solved for $\hat{\theta}$ using an assigned value of $\sigma = \sigma_c$.

The constant β should be chosen so that $\hat{\sigma}$ is an unbiased estimator when x_i, for $i = 1, 2, \ldots, n$ has a Normal distribution. To achieve this the value of β is calculated as:

$$\beta = E(\chi) = \int_{-\infty}^{\infty} \chi(z)\frac{1}{\sqrt{2\pi}}\exp\left\{\frac{-z^2}{2}\right\} dz$$

The values of $\psi\left(\dfrac{x_i - \hat{\theta}}{\hat{\sigma}}\right)\hat{\sigma}$ are known as the Winsorized residuals.

The equations are solved by a simple iterative procedure, suggested by Huber:

$$\hat{\sigma}_k = \sqrt{\frac{1}{\beta(n-1)}\left(\sum_{i=1}^{n} \chi\left(\frac{x_i - \hat{\theta}_{k-1}}{\hat{\sigma}_{k-1}}\right)\right)\hat{\sigma}_{k-1}^2}$$

and

$$\hat{\theta}_k = \hat{\theta}_{k-1} + \frac{1}{n}\sum_{i=1}^{n}\psi\left(\frac{x_i - \hat{\theta}_{k-1}}{\hat{\sigma}_k}\right)\hat{\sigma}_k$$

or

$$\hat{\sigma}_k = \sigma_c$$

if σ is fixed.

The initial values for $\hat{\theta}$ and $\hat{\sigma}$ may be user-supplied or calculated within G07DBF as the sample median and an estimate of σ based on the median absolute deviation respectively.

G07DCF is based upon subroutine LYHALG within the ROBETH library, see Marazzi (1987).

4   References

Hampel F R, Ronchetti E M, Rousseeuw P J and Stahel W A (1986) *Robust Statistics. The Approach Based on Influence Functions* Wiley

Huber P J (1981) *Robust Statistics* Wiley

Marazzi A (1987) Subroutines for robust estimation of location and scale in ROBETH *Cah. Rech. Doc. IUMSP, No. 3 ROB 1* Institut Universitaire de Médecine Sociale et Préventive, Lausanne

5   Parameters

1:    CHI – REAL (KIND=nag_wp) FUNCTION, supplied by the user.   *External Procedure*

CHI must return the value of the weight function χ for a given value of its argument. The value of χ must be non-negative.

> The specification of CHI is:
>
> ```
> FUNCTION CHI (T)
> ```
> ```
> REAL (KIND=nag_wp) CHI
> REAL (KIND=nag_wp) T
> ```
>
> 1:   T – REAL (KIND=nag_wp)   *Input*
>
>   *On entry*: the argument for which CHI must be evaluated.

CHI must either be a module subprogram USEd by, or declared as EXTERNAL in, the (sub)program from which G07DCF is called. Parameters denoted as *Input* must **not** be changed by this procedure.

2:    PSI – REAL (KIND=nag_wp) FUNCTION, supplied by the user.   *External Procedure*

PSI must return the value of the weight function ψ for a given value of its argument.

> The specification of PSI is:
>
> ```
> FUNCTION PSI (T)
> ```
> ```
> REAL (KIND=nag_wp) PSI
> REAL (KIND=nag_wp) T
> ```
>
> 1:   T – REAL (KIND=nag_wp)   *Input*
>
>   *On entry*: the argument for which PSI must be evaluated.

PSI must either be a module subprogram USEd by, or declared as EXTERNAL in, the (sub)program from which G07DCF is called. Parameters denoted as *Input* must **not** be changed by this procedure.

3:    ISIGMA – INTEGER   *Input*

On entry: the value assigned to ISIGMA determines whether $\hat{\sigma}$ is to be simultaneously estimated.

ISIGMA $= 0$
> The estimation of $\hat{\sigma}$ is bypassed and SIGMA is set equal to σ_c.

ISIGMA $= 1$
> $\hat{\sigma}$ is estimated simultaneously.

4:    N – INTEGER   *Input*

On entry: n, the number of observations.

Constraint: N > 1.

5:    X(N) – REAL (KIND=nag_wp) array   *Input*

On entry: the vector of observations, $x_1, x_2, \ldots, x_n$.

6:    BETA – REAL (KIND=nag_wp)   *Input*

On entry: the value of the constant β of the chosen CHI function.

Constraint: BETA > 0.0.

7:    THETA – REAL (KIND=nag_wp)   *Input/Output*

On entry: if SIGMA > 0, then THETA must be set to the required starting value of the estimate of the location parameter $\hat{\theta}$. A reasonable initial value for $\hat{\theta}$ will often be the sample mean or median.

On exit: the M-estimate of the location parameter $\hat{\theta}$.

8:    SIGMA – REAL (KIND=nag_wp)   *Input/Output*

On entry: the role of SIGMA depends on the value assigned to ISIGMA as follows.

If ISIGMA $= 1$, SIGMA must be assigned a value which determines the values of the starting points for the calculation of $\hat{\theta}$ and $\hat{\sigma}$. If SIGMA ≤ 0.0, then G07DCF will determine the starting points of $\hat{\theta}$ and $\hat{\sigma}$. Otherwise, the value assigned to SIGMA will be taken as the starting point for $\hat{\sigma}$, and THETA must be assigned a relevant value before entry, see above.

If ISIGMA $= 0$, SIGMA must be assigned a value which determines the values of σ_c, which is held fixed during the iterations, and the starting value for the calculation of $\hat{\theta}$. If SIGMA ≤ 0, then G07DCF will determine the value of σ_c as the median absolute deviation adjusted to reduce bias (see G07DAF) and the starting point for θ. Otherwise, the value assigned to SIGMA will be taken as the value of σ_c and THETA must be assigned a relevant value before entry, see above.

On exit: the M-estimate of the scale parameter $\hat{\sigma}$, if ISIGMA was assigned the value 1 on entry, otherwise SIGMA will contain the initial fixed value σ_c.

9:    MAXIT – INTEGER   *Input*

On entry: the maximum number of iterations that should be used during the estimation.

Suggested value: MAXIT $= 50$.

Constraint: MAXIT > 0.

10:    TOL – REAL (KIND=nag_wp)   *Input*

On entry: the relative precision for the final estimates. Convergence is assumed when the increments for THETA, and SIGMA are less than $\text{TOL} \times \max(1.0, \sigma_{k-1})$.

Constraint: $\text{TOL} > 0.0$.

11:    RS(N) – REAL (KIND=nag_wp) array   *Output*

On exit: the Winsorized residuals.

12:    NIT – INTEGER   *Output*

On exit: the number of iterations that were used during the estimation.

13:    WRK(N) – REAL (KIND=nag_wp) array   *Output*

On exit: if $\text{SIGMA} \le 0.0$ on entry, WRK will contain the n observations in ascending order.

14:    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: $\text{IFAIL} = 0$ unless the routine detects an error or a warning has been flagged (see Section 6).

6   Error Indicators and Warnings

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

Errors or warnings detected by the routine:

$\text{IFAIL} = 1$

On entry, $N \le 1$,
or $\text{MAXIT} \le 0$,
or $\text{TOL} \le 0.0$,
or $\text{ISIGMA} \ne 0$ or 1.

$\text{IFAIL} = 2$

On entry, $\text{BETA} \le 0.0$.

$\text{IFAIL} = 3$

On entry, all elements of the input array X are equal.

$\text{IFAIL} = 4$

SIGMA, the current estimate of σ, is zero or negative. This error exit is very unlikely, although it may be caused by too large an initial value of SIGMA.

$\text{IFAIL} = 5$

The number of iterations required exceeds MAXIT.

IFAIL = 6

> On completion of the iterations, the Winsorized residuals were all zero. This may occur when using the ISIGMA = 0 option with a redescending ψ function, i.e., $\psi = 0$ if $|t| > \tau$, for some positive constant τ.

> If the given value of σ is too small, then the standardized residuals $\dfrac{x_i - \hat{\theta}_k}{\sigma_c}$, will be large and all the residuals may fall into the region for which $\psi(t) = 0$. This may incorrectly terminate the iterations thus making THETA and SIGMA invalid.

> Re-enter the routine with a larger value of σ_c or with ISIGMA = 1.

IFAIL = 7

> The value returned by the CHI function is negative.

7 Accuracy

On successful exit the accuracy of the results is related to the value of TOL, see Section 5.

8 Further Comments

Standard forms of the functions ψ and χ are given in Hampel *et al.* (1986), Huber (1981) and Marazzi (1987). G07DBF calculates M-estimates using some standard forms for ψ and χ.

When you supply the initial values, care has to be taken over the choice of the initial value of σ. If too small a value is chosen then initial values of the standardized residuals $\dfrac{x_i - \hat{\theta}_k}{\sigma}$ will be large. If the redescending ψ functions are used, i.e., $\psi = 0$ if $|t| > \tau$, for some positive constant τ, then these large values are Winsorized as zero. If a sufficient number of the residuals fall into this category then a false solution may be returned, see page 152 of Hampel *et al.* (1986).

9 Example

The following program reads in a set of data consisting of eleven observations of a variable X.

The PSI and CHI functions used are Hampel's Piecewise Linear Function and Hubers CHI function respectively.

Using the following starting values various estimates of θ and σ are calculated and printed along with the number of iterations used:

(a) G07DCF determined the starting values, σ is estimated simultaneously.

(b) You must supply the starting values, σ is estimated simultaneously.

(c) G07DCF determined the starting values, σ is fixed.

(d) You must supply the starting values, σ is fixed.

9.1 Program Text

```
!   G07DCF Example Program Text
!   Mark 24 Release. NAG Copyright 2012.

    Module g07dcfe_mod

!     G07DCF Example Program Module:
!            Parameters and User-defined Routines

!     .. Use Statements ..
      Use nag_library, Only: nag_wp
!     .. Implicit None Statement ..
      Implicit None
```

```
!     .. Parameters ..
      Real (Kind=nag_wp), Parameter          :: dchi = 1.5_nag_wp
      Real (Kind=nag_wp), Parameter          :: h1 = 1.5_nag_wp
      Real (Kind=nag_wp), Parameter          :: h2 = 3.0_nag_wp
      Real (Kind=nag_wp), Parameter          :: h3 = 4.5_nag_wp
      Real (Kind=nag_wp), Parameter          :: zero = 0.0_nag_wp
      Integer, Parameter                     :: nin = 5, nout = 6
    Contains
      Function hampels_psi(t)
!       Hampel's Piecewise Linear Function.

!       .. Function Return Value ..
        Real (Kind=nag_wp)                   :: hampels_psi
!       .. Scalar Arguments ..
        Real (Kind=nag_wp), Intent (In)      :: t
!       .. Local Scalars ..
        Real (Kind=nag_wp)                   :: abst
!       .. Intrinsic Procedures ..
        Intrinsic                            :: abs, min
!       .. Executable Statements ..
        abst = abs(t)
        If (abst<h3) Then
          If (abst<=h2) Then
            hampels_psi = min(h1,abst)
          Else
            hampels_psi = h1*(h3-abst)/(h3-h2)
          End If
          If (t<zero) hampels_psi = -hampels_psi
        Else
          hampels_psi = zero
        End If
        Return
      End Function hampels_psi
      Function hubers_chi(t)
!       Huber's CHI function.

!       .. Function Return Value ..
        Real (Kind=nag_wp)                   :: hubers_chi
!       .. Scalar Arguments ..
        Real (Kind=nag_wp), Intent (In)      :: t
!       .. Local Scalars ..
        Real (Kind=nag_wp)                   :: abst, ps
!       .. Intrinsic Procedures ..
        Intrinsic                            :: abs, min
!       .. Executable Statements ..
        abst = abs(t)
        ps = min(dchi,abst)
        hubers_chi = ps*ps/2.0E0_nag_wp
        Return
      End Function hubers_chi
    End Module g07dcfe_mod
    Program g07dcfe

!     G07DCF Example Main Program

!     .. Use Statements ..
      Use nag_library, Only: g07dcf, nag_wp
      Use g07dcfe_mod, Only: hampels_psi, hubers_chi, nin, nout
!     .. Implicit None Statement ..
      Implicit None
!     .. Local Scalars ..
      Real (Kind=nag_wp)                     :: beta, sigma, sigsav, thesav,   &
                                                theta, tol
      Integer                                :: ifail, isigma, maxit, n, nit
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable        :: rs(:), wrk(:), x(:)
!     .. Executable Statements ..
      Write (nout,*) 'G07DCF Example Program Results'
      Write (nout,*)

!     Skip heading in data file
```

```
        Read (nin,*)

!       Read in the problem size and control parameters
        Read (nin,*) n, beta, maxit

        Allocate (x(n),wrk(n),rs(n))

!       Read in data
        Read (nin,*) x(1:n)

!       Display titles
        Write (nout,*) '              Input parameters     Output parameters'
        Write (nout,*) 'ISIGMA    SIGMA    THETA    TOL     SIGMA    THETA'

d_lp: Do
        Read (nin,*,Iostat=ifail) isigma, sigma, theta, tol
        If (ifail/=0) Then
          Exit d_lp
        End If

!       Save input parameters
        sigsav = sigma
        thesav = theta

!       Compute M-estimates
        ifail = 0
        Call g07dcf(hubers_chi,hampels_psi,isigma,n,x,beta,theta,sigma,maxit, &
          tol,rs,nit,wrk,ifail)

!       Display results
        Write (nout,99999) isigma, sigsav, thesav, tol, sigma, theta
      End Do d_lp

99999 Format (1X,I3,3X,2F8.4,F7.4,1X,2F8.4)
    End Program g07dcfe
```

9.2 Program Data

```
G07DCF Example Program Data
11   0.3892326   50                 :: N,BETA,MAXIT
13.0 11.0 16.0  5.0 3.0 18.0
 9.0  8.0  6.0 27.0 7.0             :: End of X
   1     -1.0     0.0    0.0001     :: ISIGMA,SIGMA,THETA,TOL
   1      7.0     2.0    0.0001     :: ISIGMA,SIGMA,THETA,TOL
   0     -1.0     0.0    0.0001     :: ISIGMA,SIGMA,THETA,TOL
   0      7.0     2.0    0.0001     :: ISIGMA,SIGMA,THETA,TOL
```

9.3 Program Results

```
 G07DCF Example Program Results

          Input parameters     Output parameters
 ISIGMA    SIGMA    THETA    TOL     SIGMA    THETA
    1     -1.0000   0.0000  0.0001   6.3247  10.5487
    1      7.0000   2.0000  0.0001   6.3249  10.5487
    0     -1.0000   0.0000  0.0001   5.9304  10.4896
    0      7.0000   2.0000  0.0001   7.0000  10.6500
```

_____