NAG Library Routine Document

G02HAF

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

G02HAF performs bounded influence regression (M-estimates). Several standard methods are available.

2 Specification

```
SUBROUTINE G02HAF (INDW, IPSI, ISIGMA, INDC, N, M, X, LDX, Y, CPSI, H1, H2,      &
                   H3, CUCV, DCHI, THETA, SIGMA, C, LDC, RS, WGT, TOL,            &
                   MAXIT, NITMON, WORK, IFAIL)
INTEGER            INDW, IPSI, ISIGMA, INDC, N, M, LDX, LDC, MAXIT,               &
                   NITMON, IFAIL
REAL (KIND=nag_wp) X(LDX,M), Y(N), CPSI, H1, H2, H3, CUCV, DCHI, THETA(M),        &
                   SIGMA, C(LDC,M), RS(N), WGT(N), TOL, WORK(4*N+M*(N+M))
```

3 Description

For the linear regression model

$$y = X\theta + \epsilon,$$

where y is a vector of length n of the dependent variable,

X is a n by m matrix of independent variables of column rank k,

θ is a vector of length m of unknown parameters,

and ϵ is a vector of length n of unknown errors with $\mathrm{var}\,(\epsilon_i) = \sigma^2$,

G02HAF calculates the M-estimates given by the solution, $\hat{\theta}$, to the equation

$$\sum_{i=1}^{n} \psi(r_i/(\sigma w_i)) w_i x_{ij} = 0, \qquad j = 1, 2, \ldots, m, \tag{1}$$

where r_i is the ith residual, i.e., the ith element of $r = y - X\hat{\theta}$,

ψ is a suitable weight function,

w_i are suitable weights,

and σ may be estimated at each iteration by the median absolute deviation of the residuals

$$\hat{\sigma} = \operatorname*{med}_i \, [|r_i|]/\beta_1$$

or as the solution to

$$\sum_{i=1}^{n} \chi(r_i/(\hat{\sigma} w_i)) w_i^2 = (n-k)\beta_2$$

for suitable weight function χ, where β_1 and β_2 are constants, chosen so that the estimator of σ is asymptotically unbiased if the errors, ϵ_i, have a Normal distribution. Alternatively σ may be held at a constant value.

The above describes the Schweppe type regression. If the w_i are assumed to equal 1 for all i then Huber type regression is obtained. A third type, due to Mallows, replaces (1) by

$$\sum_{i=1}^{n} \psi(r_i/\sigma) w_i x_{ij} = 0, \qquad j = 1, 2, \ldots, m.$$

This may be obtained by use of the transformations

$$\begin{aligned} w_i^* &\leftarrow \sqrt{w_i} \\ y_i^* &\leftarrow y_i \sqrt{w_i} \\ x_{ij}^* &\leftarrow x_{ij} \sqrt{w_i}, \quad j = 1, 2, \ldots, m \end{aligned}$$

(see Section 3 of Marazzi (1987a)).

For Huber and Schweppe type regressions, β_1 is the 75th percentile of the standard Normal distribution. For Mallows type regression β_1 is the solution to

$$\frac{1}{n} \sum_{i=1}^{n} \Phi\left(\beta_1 / \sqrt{w_i}\right) = 0.75,$$

where Φ is the standard Normal cumulative distribution function (see S15ABF).

β_2 is given by

$$\beta_2 = \int_{-\infty}^{\infty} \chi(z) \phi(z)\, dz \qquad \text{in the Huber case;}$$

$$\beta_2 = \frac{1}{n} \sum_{i=1}^{n} w_i \int_{-\infty}^{\infty} \chi(z) \phi(z)\, dz \qquad \text{in the Mallows case;}$$

$$\beta_2 = \frac{1}{n} \sum_{i=1}^{n} w_i^2 \int_{-\infty}^{\infty} \chi(z/w_i) \phi(z)\, dz \quad \text{in the Schweppe case;}$$

where ϕ is the standard Normal density, i.e., $\frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2} x^2\right)$.

The calculation of the estimates of θ can be formulated as an iterative weighted least squares problem with a diagonal weight matrix G given by

$$G_{ii} = \begin{cases} \dfrac{\psi(r_i/(\sigma w_i))}{(r_i/(\sigma w_i))}, & r_i \neq 0 \\ \psi'(0), & r_i = 0 \end{cases},$$

where $\psi'(t)$ is the derivative of ψ at the point t.

The value of θ at each iteration is given by the weighted least squares regression of y on X. This is carried out by first transforming the y and X by

$$\begin{aligned} \tilde{y}_i &= y_i \sqrt{G_{ii}} \\ \tilde{x}_{ij} &= x_{ij} \sqrt{G_{ii}}, \quad j = 1, 2, \ldots, m \end{aligned}$$

and then using F04JGF. If X is of full column rank then an orthogonal-triangular (QR) decomposition is used; if not, a singular value decomposition is used.

The following functions are available for ψ and χ in G02HAF.

(a) **Unit Weights**

$$\psi(t) = t, \qquad \chi(t) = \frac{t^2}{2}.$$

This gives least squares regression.

(b) **Huber's Function**

$$\psi(t) = \max(-c, \min(c, t)), \qquad \chi(t) = \begin{cases} \dfrac{t^2}{2}, & |t| \leq d \\ \dfrac{d^2}{2}, & |t| > d \end{cases}$$

(c) **Hampel's Piecewise Linear Function**

$$\psi_{h_1,h_2,h_3}(t) = -\psi_{h_1,h_2,h_3}(-t) = \begin{cases} t, & 0 \leq t \leq h_1 \\ h_1, & h_1 \leq t \leq h_2 \\ h_1(h_3 - t)/(h_3 - h_2), & h_2 \leq t \leq h_3 \\ 0, & h_3 < t \end{cases}$$

$$\chi(t) = \begin{cases} \dfrac{t^2}{2}, & |t| \leq d \\ \dfrac{d^2}{2}, & |t| > d \end{cases}$$

(d) **Andrew's Sine Wave Function**

$$\psi(t) = \begin{cases} \sin t, & -\pi \leq t \leq \pi \\ 0, & |t| > \pi \end{cases} \qquad \chi(t) = \begin{cases} \dfrac{t^2}{2}, & |t| \leq d \\ \dfrac{d^2}{2}, & |t| > d \end{cases}$$

(e) **Tukey's Bi-weight**

$$\psi(t) = \begin{cases} t\left(1 - t^2\right)^2, & |t| \leq 1 \\ 0, & |t| > 1 \end{cases} \qquad \chi(t) = \begin{cases} \dfrac{t^2}{2}, & |t| \leq d \\ \dfrac{d^2}{2}, & |t| > d \end{cases}$$

where c, h_1, h_2, h_3, and d are given constants.

Several schemes for calculating weights have been proposed, see Hampel *et al.* (1986) and Marazzi (1987a). As the different independent variables may be measured on different scales, one group of proposed weights aims to bound a standardized measure of influence. To obtain such weights the matrix A has to be found such that:

$$\frac{1}{n}\sum_{i=1}^{n} u\left(\|z_i\|_2\right) z_i z_i^{\mathrm{T}} = I$$

and

$$z_i = Ax_i,$$

where x_i is a vector of length m containing the ith row of X,

A is an m by m lower triangular matrix,

and u is a suitable function.

The weights are then calculated as

$$w_i = f\left(\|z_i\|_2\right)$$

for a suitable function f.

G02HAF finds A using the iterative procedure

$$A_k = (S_k + I)A_{k-1},$$

where $S_k = (s_{jl})$,

$$s_{jl} = \begin{cases} -\min\left[\max\left(h_{jl}/n, -BL\right), BL\right], & j > 1 \\ -\min\left[\max\left(\frac{1}{2}\left(h_{jj}/n - 1\right), -BD\right), BD\right], & j = 1 \end{cases}$$

and

$$h_{jl} = \sum_{i=1}^{n} u\left(\|z_i\|_2\right) z_{ij} z_{il}$$

and BL and BD are bounds set at 0.9.

Two weights are available in G02HAF:

(i) **Krasker–Welsch Weights**

$$u(t) = g_1\left(\frac{c}{t}\right),$$

where $g_1(t) = t^2 + \left(1 - t^2\right)(2\Phi(t) - 1) - 2t\phi(t)$,

$\Phi(t)$ is the standard Normal cumulative distribution function,

$\phi(t)$ is the standard Normal probability density function,

and $f(t) = \dfrac{1}{t}$.

These are for use with Schweppe type regression.

(ii) **Maronna's Proposed Weights**

$$u(t) = \begin{cases} \dfrac{c}{t^2} & |t| > c \\ 1 & |t| \leq c \end{cases}$$
$$f(t) = \sqrt{u(t)}.$$

These are for use with Mallows type regression.

Finally the asymptotic variance-covariance matrix, C, of the estimates θ is calculated.

For Huber type regression

$$C = f_H\left(X^{\mathrm{T}}X\right)^{-1}\hat{\sigma}^2,$$

where

$$f_H = \frac{1}{n-m} \frac{\sum_{i=1}^{n} \psi^2(r_i/\hat{\sigma})}{\left(\frac{1}{n}\sum_{i=1}^{n} \psi'\left(\frac{r_i}{\hat{\sigma}}\right)\right)^2} \kappa^2$$

$$\kappa^2 = 1 + \frac{m}{n} \frac{\frac{1}{n}\sum_{i=1}^{n}\left(\psi'(r_i/\hat{\sigma}) - \frac{1}{n}\sum_{i=1}^{n}\psi'(r_i/\hat{\sigma})\right)^2}{\left(\frac{1}{n}\sum_{i=1}^{n}\psi'\left(\frac{r_i}{\hat{\sigma}}\right)\right)^2}.$$

See Huber (1981) and Marazzi (1987b).

For Mallows and Schweppe type regressions C is of the form

$$\frac{\hat{\sigma}^2}{n} S_1^{-1} S_2 S_1^{-1},$$

where $S_1 = \frac{1}{n} X^{\mathrm{T}} D X$ and $S_2 = \frac{1}{n} X^{\mathrm{T}} P X$.

D is a diagonal matrix such that the ith element approximates $E\left(\psi'(r_i/(\sigma w_i))\right)$ in the Schweppe case and $E\left(\psi'(r_i/\sigma) w_i\right)$ in the Mallows case.

P is a diagonal matrix such that the ith element approximates $E\left(\psi^2(r_i/(\sigma w_i)) w_i^2\right)$ in the Schweppe case and $E\left(\psi^2(r_i/\sigma) w_i^2\right)$ in the Mallows case.

Two approximations are available in G02HAF:

1. Average over the r_i

Schweppe Mallows

$$D_i = \left(\frac{1}{n}\sum_{j=1}^{n} \psi'\left(\frac{r_j}{\hat{\sigma} w_i}\right)\right) w_i \qquad D_i = \left(\frac{1}{n}\sum_{j=1}^{n} \psi'\left(\frac{r_j}{\hat{\sigma}}\right)\right) w_i$$

$$P_i = \left(\frac{1}{n}\sum_{j=1}^{n} \psi^2\left(\frac{r_j}{\hat{\sigma} w_i}\right)\right) w_i^2 \qquad P_i = \left(\frac{1}{n}\sum_{j=1}^{n} \psi^2\left(\frac{r_j}{\hat{\sigma}}\right)\right) w_i^2$$

2. Replace expected value by observed

Schweppe Mallows

$$D_i = \psi'\left(\frac{r_i}{\hat{\sigma} w_i}\right) w_i \qquad D_i = \psi'\left(\frac{r_i}{\hat{\sigma}}\right) w_i\,.$$

$$P_i = \psi^2\left(\frac{r_i}{\hat{\sigma} w_i}\right) w_i^2 \qquad P_i = \psi^2\left(\frac{r_i}{\hat{\sigma}}\right) w_i^2$$

See Hampel *et al.* (1986) and Marazzi (1987b).

Note: there is no explicit provision in the routine for a constant term in the regression model. However, the addition of a dummy variable whose value is 1.0 for all observations will produce a value of $\hat{\theta}$ corresponding to the usual constant term.

G02HAF is based on routines in ROBETH; see Marazzi (1987a).

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

Marazzi A (1987b) Subroutines for robust and bounded influence regression in ROBETH *Cah. Rech. Doc. IUMSP, No. 3 ROB 2* Institut Universitaire de Médecine Sociale et Préventive, Lausanne

5 Parameters

1: INDW – INTEGER *Input*

On entry: specifies the type of regression to be performed.

INDW < 0
Mallows type regression with Maronna's proposed weights.

INDW = 0
  Huber type regression.

INDW > 0
  Schweppe type regression with Krasker–Welsch weights.

2: IPSI – INTEGER *Input*

On entry: specifies which ψ function is to be used.

IPSI = 0
  $\psi(t) = t$, i.e., least squares.

IPSI = 1
  Huber's function.

IPSI = 2
  Hampel's piecewise linear function.

IPSI = 3
  Andrew's sine wave.

IPSI = 4
  Tukey's bi-weight.

Constraint: $0 \leq \text{IPSI} \leq 4$.

3: ISIGMA – INTEGER *Input*

On entry: specifies how σ is to be estimated.

ISIGMA < 0
  σ is estimated by median absolute deviation of residuals.

ISIGMA = 0
  σ is held constant at its initial value.

ISIGMA > 0
  σ is estimated using the χ function.

4: INDC – INTEGER *Input*

On entry: if INDW $\neq 0$, INDC specifies the approximations used in estimating the covariance matrix of $\hat{\theta}$.

INDC = 1
  Averaging over residuals.

INDC $\neq$ 1
  Replacing expected by observed.

INDW = 0
  INDC is not referenced.

5: N – INTEGER *Input*

On entry: n, the number of observations.

Constraint: N > 1.

6: M – INTEGER *Input*

On entry: m, the number of independent variables.

Constraint: $1 \leq \text{M} < \text{N}$.

7: X(LDX,M) – REAL (KIND=nag_wp) array *Input/Output*

On entry: the values of the X matrix, i.e., the independent variables. X(i,j) must contain the ijth element of X, for $i = 1,2,\ldots,n$ and $j = 1,2,\ldots,m$.

If INDW < 0, then during calculations the elements of X will be transformed as described in Section 3. Before exit the inverse transformation will be applied. As a result there may be slight differences between the input X and the output X.

On exit: unchanged, except as described above.

8: LDX – INTEGER *Input*

On entry: the first dimension of the array X as declared in the (sub)program from which G02HAF is called.

Constraint: LDX $\geq$ N.

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

On entry: the data values of the dependent variable.

Y(i) must contain the value of y for the ith observation, for $i = 1,2,\ldots,n$.

If INDW < 0, then during calculations the elements of Y will be transformed as described in Section 3. Before exit the inverse transformation will be applied. As a result there may be slight differences between the input Y and the output Y.

On exit: unchanged, except as described above.

10: CPSI – REAL (KIND=nag_wp) *Input*

On entry: if IPSI $= 1$, CPSI must specify the parameter, c, of Huber's ψ function.

If IPSI $\neq 1$ on entry, CPSI is not referenced.

Constraint: if CPSI > 0.0, IPSI $= 1$.

11: H1 – REAL (KIND=nag_wp) *Input*
12: H2 – REAL (KIND=nag_wp) *Input*
13: H3 – REAL (KIND=nag_wp) *Input*

On entry: if IPSI $= 2$, H1, H2, and H3 must specify the parameters h_1, h_2, and h_3, of Hampel's piecewise linear ψ function. H1, H2, and H3 are not referenced if IPSI $\neq 2$.

Constraint: if IPSI $= 2$, $0.0 \leq$ H1 $\leq$ H2 $\leq$ H3 and H3 > 0.0.

14: CUCV – REAL (KIND=nag_wp) *Input*

On entry: if INDW < 0, must specify the value of the constant, c, of the function u for Maronna's proposed weights.

If INDW > 0, must specify the value of the function u for the Krasker–Welsch weights.

If INDW $= 0$, is not referenced.

Constraints:

if INDW < 0, CUCV $\geq$ M;
if INDW > 0, CUCV $\geq \sqrt{\text{M}}$.

15: DCHI – REAL (KIND=nag_wp) *Input*

On entry: d, the constant of the χ function. DCHI is not referenced if IPSI $= 0$, or if ISIGMA ≤ 0.

Constraint: if IPSI $\neq 0$ and ISIGMA > 0, DCHI > 0.0.

16: THETA(M) – REAL (KIND=nag_wp) array *Input/Output*

On entry: starting values of the parameter vector θ. These may be obtained from least squares regression. Alternatively if ISIGMA < 0 and SIGMA $= 1$ or if ISIGMA > 0 and SIGMA approximately equals the standard deviation of the dependent variable, y, then THETA$(i) = 0.0$, for $i = 1, 2, \ldots, m$ may provide reasonable starting values.

On exit: THETA(i) contains the M-estimate of θ_i, for $i = 1, 2, \ldots, m$.

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

On entry: a starting value for the estimation of σ. SIGMA should be approximately the standard deviation of the residuals from the model evaluated at the value of θ given by THETA on entry.

Constraint: SIGMA > 0.0.

On exit: contains the final estimate of σ if ISIGMA $\neq 0$ or the value assigned on entry if ISIGMA $= 0$.

18: C(LDC,M) – REAL (KIND=nag_wp) array *Output*

On exit: the diagonal elements of C contain the estimated asymptotic standard errors of the estimates of θ, i.e., C(i, i) contains the estimated asymptotic standard error of the estimate contained in THETA(i).

The elements above the diagonal contain the estimated asymptotic correlation between the estimates of θ, i.e., C(i, j), $1 \leq i < j \leq m$ contains the asymptotic correlation between the estimates contained in THETA(i) and THETA(j).

The elements below the diagonal contain the estimated asymptotic covariance between the estimates of θ, i.e., C(i, j), $1 \leq j < i \leq m$ contains the estimated asymptotic covariance between the estimates contained in THETA(i) and THETA(j).

19: LDC – INTEGER *Input*

On entry: the first dimension of the array C as declared in the (sub)program from which G02HAF is called.

Constraint: LDC $\geq$ M.

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

On exit: the residuals from the model evaluated at final value of THETA, i.e., RS contains the vector $\left(y - X\hat{\theta}\right)$.

21: WGT(N) – REAL (KIND=nag_wp) array *Output*

On exit: the vector of weights. WGT(i) contains the weight for the ith observation, for $i = 1, 2, \ldots, n$.

22: TOL – REAL (KIND=nag_wp) *Input*

On entry: the relative precision for the calculation of A (if INDW $\neq 0$), the estimates of θ and the estimate of σ (if ISIGMA $\neq 0$). Convergence is assumed when the relative change in all elements being considered is less than TOL.

If INDW < 0 and ISIGMA < 0, TOL is also used to determine the precision of β_1.

It is advisable for TOL to be greater than $100 \times$ ***machine precision***.

Constraint: TOL > 0.0.

23: MAXIT – INTEGER *Input*

On entry: the maximum number of iterations that should be used in the calculation of A (if INDW $\neq 0$), and of the estimates of θ and σ, and of β_1 (if INDW < 0 and ISIGMA < 0).

A value of MAXIT $= 50$ should be adequate for most uses.

Constraint: MAXIT > 0.

24: NITMON – INTEGER *Input*

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

NITMON $= 0$
: No information is printed.

NITMON $\neq 0$
: The current estimate of θ, the change in θ during the current iteration and the current value of σ are printed on the first and every abs(NITMON) iterations.

Also, if INDW $\neq 0$ and NITMON > 0 then information on the iterations to calculate A is printed. This is the current estimate of A and the maximum value of S_{ij} (see Section 3).

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

25: WORK($4 \times \mathrm{N} + \mathrm{M} \times (\mathrm{N} + \mathrm{M})$) – REAL (KIND=nag_wp) array *Output*

On exit: the following values are assigned to WORK:

WORK(1) $= \beta_1$ if ISIGMA < 0, or WORK(1) $= \beta_2$ if ISIGMA > 0.

WORK(2) $=$ number of iterations used to calculate A.

WORK(3) $=$ number of iterations used to calculate final estimates of θ and σ.

WORK(4) $= k$, the rank of the weighted least squares equations.

The rest of the array is used as workspace.

26: 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, because for this routine the values of the output parameters may be useful even if IFAIL $\neq 0$ on exit, the recommended value is -1. **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).

Note: G02HAF may return useful information for one or more of the following detected errors or warnings.

Errors or warnings detected by the routine:

IFAIL $= 1$

On entry, N ≤ 1,
or M < 1,
or N $\leq$ M,
or LDX $<$ N,
or LDC $<$ M.

IFAIL $= 2$

On entry, IPSI < 0,
or IPSI > 4.

IFAIL $= 3$

On entry, SIGMA ≤ 0.0,
or IPSI $= 1$ and CPSI ≤ 0.0,
or IPSI $= 2$ and H1 < 0.0,
or IPSI $= 2$ and H1 $>$ H2,
or IPSI $= 2$ and H2 $>$ H3,
or IPSI $= 2$ and H1 $=$ H2 $=$ H3 $= 0.0$,
or IPSI $\neq 0$ and ISIGMA > 0 and DCHI ≤ 0.0,
or INDW > 0 and CUCV $< \sqrt{\text{M}}$,
or INDW < 0 and CUCV $<$ M.

IFAIL $= 4$

On entry, TOL ≤ 0.0,
or MAXIT ≤ 0.

IFAIL $= 5$

The number of iterations required to calculate the weights exceeds MAXIT. (Only if INDW $\neq 0$.)

IFAIL $= 6$

The number of iterations required to calculate β_1 exceeds MAXIT. (Only if INDW < 0 and ISIGMA < 0.)

IFAIL $= 7$

Either the number of iterations required to calculate θ and σ exceeds MAXIT (note that, in this case WORK$(3) =$ MAXIT on exit), or the iterations to solve the weighted least squares equations failed to converge. The latter is an unlikely error exit.

IFAIL $= 8$

The weighted least squares equations are not of full rank.

IFAIL $= 9$

If INDW $= 0$ then $\left(X^{\mathrm{T}}X\right)$ is almost singular.

If INDW $\neq 0$ then S_1 is singular or almost singular. This may be due to too many diagonal elements of the matrix being zero, see Section 8.

IFAIL $= 10$

In calculating the correlation factor for the asymptotic variance-covariance matrix either the value of

$$\frac{1}{n}\sum_{i=1}^{n}\psi'(r_i/\hat{\sigma}) = 0, \qquad \text{or} \qquad \kappa = 0, \qquad \text{or} \qquad \sum_{i=1}^{n}\psi^2(r_i/\hat{\sigma}) = 0.$$

See Section 8. In this case C is returned as $X^{\mathrm{T}}X$.

(Only if INDW $= 0$.)

IFAIL $= 11$

The estimated variance for an element of $\theta \leq 0$.

In this case the diagonal element of C will contain the negative variance and the above diagonal elements in the row and column corresponding to the element will be returned as zero.

This error may be caused by rounding errors or too many of the diagonal elements of P being zero, where P is defined in Section 3. See Section 8.

IFAIL $= 12$

The degrees of freedom for error, $n - k \leq 0$ (this is an unlikely error exit), or the estimated value of σ was 0 during an iteration.

7 Accuracy

The precision of the estimates is determined by TOL. As a more stable method is used to calculate the estimates of θ than is used to calculate the covariance matrix, it is possible for the least squares equations to be of full rank but the $\left(X^{\mathrm{T}}X\right)$ matrix to be too nearly singular to be inverted.

8 Further Comments

In cases when ISIGMA ≥ 0 it is important for the value of SIGMA to be of a reasonable magnitude. Too small a value may cause too many of the winsorized residuals, i.e., $\psi(r_i/\sigma)$, to be zero or a value of $\psi'(r_i/\sigma)$, used to estimate the asymptotic covariance matrix, to be zero. This can lead to errors IFAIL $= 8$ or 9 (if INDW $\neq 0$), IFAIL $= 10$ (if INDW $= 0$) and IFAIL $= 11$.

G02HBF, G02HDF and G02HFF together carry out the same calculations as G02HAF but for user-supplied functions for ψ, χ, ψ' and u.

9 Example

The number of observations and the number of x variables are read in followed by the data. The option parameters are then read in (in this case giving Schweppe type regression with Hampel's ψ function and Huber's χ function and then using the 'replace expected by observed' option in calculating the covariances). Finally a set of values for the constants are read in.

After a call to G02HAF, $\hat{\theta}$, its standard error and $\hat{\sigma}$ are printed. In addition the weight and residual for each observation is printed.

9.1 Program Text

```
    Program g02hafe

!     G02HAF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: g02haf, nag_wp, x04abf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter                :: iset = 1, nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)                :: cpsi, cucv, dchi, h1, h2, h3, sigma, &
                                           tol
      Integer                           :: i, ifail, indc, indw, ipsi, isigma,  &
                                           ldc, ldx, lwork, m, maxit, n, nadv,  &
                                           nitmon
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable   :: c(:,:), rs(:), theta(:), wgt(:),     &
                                           work(:), x(:,:), y(:)
!     .. Executable Statements ..
      Write (nout,*) 'G02HAF Example Program Results'
      Write (nout,*)

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

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

      ldx = n
      ldc = m
      lwork = 4*n + m*(n+m)
      Allocate (x(ldx,m),y(n),theta(m),c(ldc,m),work(lwork),rs(n),wgt(n))

!     Read in data
      Read (nin,*)(x(i,1:m),y(i),i=1,n)

!     Read in control parameters
      Read (nin,*) indw, ipsi, isigma, nitmon, maxit, tol

!     Read in appropriate weight function parameters
      If (indw/=0) Then
        Read (nin,*) cucv, indc
      End If
      If (ipsi>0) Then
        If (ipsi==1) Then
          Read (nin,*) cpsi
        Else If (ipsi==2) Then
          Read (nin,*) h1, h2, h3
        End If
        If (isigma>0) Then
          Read (nin,*) dchi
        End If
      End If

!     Set the advisory channel to NOUT for monitoring information
      If (nitmon/=0) Then
        nadv = nout
        Call x04abf(iset,nadv)
      End If

!     Read in initial values
      Read (nin,*) sigma
      Read (nin,*) theta(1:m)

!     Perform M-estimate regression
      ifail = -1
      Call g02haf(indw,ipsi,isigma,indc,n,m,x,ldx,y,cpsi,h1,h2,h3,cucv,dchi, &
        theta,sigma,c,ldc,rs,wgt,tol,maxit,nitmon,work,ifail)
      If (ifail/=0) Then
        If (ifail<7) Then
          Go To 100
        Else
          Write (nout,*) &
            '       Some of the following reslts may be unreliable'
        End If
      End If

!     Display results
      Write (nout,99999) 'Sigma = ', sigma
      Write (nout,*)
      Write (nout,*) '       THETA      Standard'
      Write (nout,*) '                   errors'
      Write (nout,99998)(theta(i),c(i,i),i=1,m)
      Write (nout,*)
      Write (nout,*) '     Weights     Residuals'
      Write (nout,99998)(wgt(i),rs(i),i=1,n)

100   Continue

99999 Format (1X,A,F10.4)
99998 Format (1X,F12.4,F13.4)
    End Program g02hafe
```

9.2 Program Data

```
G02HAF Example Program Data
 8  3                       : N,M
 1. -1. -1. 2.1
 1. -1.  1. 3.6
 1.  1. -1. 4.5
 1.  1.  1. 6.1
 1. -2.  0. 1.3
 1.  0. -2. 1.9
 1.  2.  0. 6.7
 1.  0.  2. 5.5             : End of X1 X2 X3 and Y values
 1  2  1 0 50 1.0E-5        : INDW,IPSI,ISIGMA,NITMON,MAXIT,TOL
 3.0  0                     : CUCV,INDC
 1.5 3.0 4.5                : H1,H2,H3
 1.5                        : DCHI
 1.0                        : Initial value for SIGMA
 0.0 0.0 0.0                : Initial values for THETA
```

9.3 Program Results

```
 G02HAF Example Program Results

 Sigma =     0.2026

        THETA      Standard
                    errors
       4.0423       0.0384
       1.3083       0.0272
       0.7519       0.0311

      Weights     Residuals
       0.5783       0.1179
       0.5783       0.1141
       0.5783      -0.0987
       0.5783      -0.0026
       0.4603      -0.1256
       0.4603      -0.6385
       0.4603       0.0410
       0.4603      -0.0462
```