NAG Toolbox for MATLAB

g02ha

1 Purpose

g02ha performs bounded influence regression (M-estimates). Several standard methods are available.

2 Syntax

```
[x, y, theta, sigma, c, rs, wgt, work, ifail] = g02ha(indw, ipsi,
isigma, indc, x, y, cpsi, h1, h2, h3, cucv, dchi, theta, sigma, tol,
maxit, nitmon, 'n', n, 'm', 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 $\operatorname{var}(\epsilon_i) = \sigma^2$,

g02ha 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 s15ab).

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

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

and then using f04jg. 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 g02ha.

(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**

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

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

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

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

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

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

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.

g02ha finds A using the iterative procedure

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

where $S_k = \left(s_{jl}\right)$,

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

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 g02ha:

(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 $f(t) = \frac{1}{t}$.

These are for use with Schweppe type regression.

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

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

$$f(t) = \sqrt{u(t)}.$$

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 $S_1 = \frac{1}{n}X^{\mathrm{T}}DX$ and $S_2 = \frac{1}{n}X^{\mathrm{T}}PX$.

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 g02ha:

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 function 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.

g02ha 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

5.1 Compulsory Input Parameters

1: **indw – int32 scalar**

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 – int32 scalar**

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: $0 \leq$ **ipsi** ≤ 4.

3: **isigma – int32 scalar**

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 – int32 scalar**

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: **x(ldx,m) – double array**

ldx, the first dimension of the array, must be at least **n**.

The values of the X matrix, i.e., the independent variables. $\mathbf{x}(i,j)$ must contain the ijth element of X, for $i = 1,2,\ldots,n$ and $j = 1,2,\ldots,m$.

If $\mathbf{indw} < 0$, then during calculations the elements of $\mathbf{x}$ 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 $\mathbf{x}$ and the output $\mathbf{x}$.

6: **y(n) – double array**

The data values of the dependent variable.

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

If $\mathbf{indw} < 0$, then during calculations the elements of $\mathbf{y}$ 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 $\mathbf{y}$ and the output $\mathbf{y}$.

7: **cpsi – double scalar**

If $\mathbf{ipsi} = 1$, **cpsi** must specify the parameter, c, of Huber's ψ function.

If $\mathbf{ipsi} \neq 1$ on entry, **cpsi** is not referenced.

Constraint: if $\mathbf{cpsi} > 0.0$, $\mathbf{ipsi} = 1$.

8: **h1 – double scalar**

9: **h2 – double scalar**

10: **h3 – double scalar**

If $\mathbf{ipsi} = 2$, **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 $\mathbf{ipsi} \neq 2$.

Constraint: if $\mathbf{ipsi} = 2$, $0.0 \leq \mathbf{h1} \leq \mathbf{h2} \leq \mathbf{h3}$ and $\mathbf{h3} > 0.0$.

11: **cucv – double scalar**

If $\mathbf{indw} < 0$, must specify the value of the constant, c, of the function u for Maronna's proposed weights.

If $\mathbf{indw} > 0$, must specify the value of the function u for the Krasker–Welsch weights.

If $\mathbf{indw} = 0$, is not referenced.

Constraints:

if $\mathbf{indw} < 0$, $\mathbf{cucv} \geq \mathbf{m}$;
if $\mathbf{indw} > 0$, $\mathbf{cucv} \geq \sqrt{\mathbf{m}}$.

12: **dchi – double scalar**

d, the constant of the χ function. **dchi** is not referenced if $\mathbf{ipsi} = 0$, or if $\mathbf{isigma} \leq 0$.

Constraint: if $\mathbf{ipsi} \neq 0$ and $\mathbf{isigma} > 0$, $\mathbf{dchi} > 0.0$.

13: **theta(m) – double array**

Starting values of the parameter vector θ. These may be obtained from least-squares regression. Alternatively if $\mathbf{isigma} < 0$ and $\mathbf{sigma} = 1$ or if $\mathbf{isigma} > 0$ and **sigma** approximately equals the standard deviation of the dependent variable, y, then $\mathbf{theta}(i) = 0.0$, for $i = 1,2,\ldots,m$ may provide reasonable starting values.

14: **sigma – double scalar**

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: $\mathbf{sigma} > 0.0$.

15: **tol – double scalar**

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.

16: **maxit – int32 scalar**

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.

17: **nitmon – int32 scalar**

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 x04ab).

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The dimension of the arrays **y**, **wgt**, **rs**. (An error is raised if these dimensions are not equal.)

n, the number of observations.

Constraint: **n** > 1.

2: **m – int32 scalar**

Default: The dimension of the array **x** and the first dimension of the array **x**. (An error is raised if these dimensions are not equal.)

m, the number of independent variables.

Constraint: $1 \leq$ **m** $<$ **n**.

5.3 Input Parameters Omitted from the MATLAB Interface

ldx, ldc

5.4 Output Parameters

1: **x(ldx,m) – double array**

Unchanged, except as described above.

2: **y(n) – double array**

Unchanged, except as described above.

3: **theta(m) – double array**

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

4: **sigma – double scalar**

Contains the final estimate of σ if **isigma** $\neq 0$ or the value assigned on entry if **isigma** $= 0$.

5: **c(ldc,m) – double array**

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).

6: **rs(n) – double array**

The residuals from the model evaluated at final value of **theta**, i.e., **rs** contains the vector $\left(y - X\hat{\theta}\right)$.

7: **wgt(n) – double array**

The vector of weights. **wgt**(i) contains the weight for the ith observation, for $i = 1, 2, \ldots, n$.

8: **work(4 × n + m × (n + m)) – double array**

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.

9: **ifail – int32 scalar**

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Note: g02ha may return useful information for one or more of the following detected errors or warnings.

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** $< \sqrt{\mathbf{m}}$,
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 $\mathrm{WK}(3) =$ **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

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

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** $= 11$.

g02hb, g02hd and g02hf together carry out the same calculations as g02ha but for user-supplied functions for ψ, χ, ψ' and u.

9 Example

```
indw = int32(1);
ipsi = int32(2);
isigma = int32(1);
indc = int32(0);
x = [1, -1, -1;
     1, -1, 1;
     1, 1, -1;
     1, 1, 1;
     1, -2, 0;
     1, 0, -2;
     1, 2, 0;
     1, 0, 2];
y = [2.1;
     3.6;
     4.5;
     6.1;
     1.3;
     1.9;
     6.7;
     5.5];
cpsi = 0;
h1 = 1.5;
h2 = 3;
h3 = 4.5;
cucv = 3;
dchi = 1.5;
theta = [0;
     0;
     0];
sigma = 1;
tol = 5e-05;
nitmon = int32(0);
[xOut, yOut, thetaOut, sigmaOut, c, rs, wgt, work, ifail] = ...
      g02ha(indw, ipsi, isigma, indc, x, y, cpsi, h1, h2, h3, cucv, dchi,
theta, sigma, tol, nitmon)
```

```
xOut =
     1    -1    -1
     1    -1     1
     1     1    -1
     1     1     1
     1    -2     0
     1     0    -2
     1     2     0
     1     0     2
yOut =
    2.1000
    3.6000
    4.5000
    6.1000
    1.3000
    1.9000
    6.7000
    5.5000
thetaOut =
    4.0423
    1.3083
    0.7519
sigmaOut =
    0.2026
c =
    0.0384   -0.5299   -0.5929
   -0.0006    0.0272    0.0546
   -0.0007    0.0000    0.0311
rs =
    0.1179
    0.1141
   -0.0987
   -0.0026
   -0.1256
   -0.6385
    0.0410
   -0.0462
wgt =
    0.5783
    0.5783
    0.5783
    0.5783
    0.4603
    0.4603
    0.4603
    0.4603
work =
     array elided
ifail =
           0
```