NAG Toolbox for MATLAB

g02hd

1 Purpose

g02hd performs bounded influence regression (M-estimates) using an iterative weighted least-squares algorithm.

2 Syntax

```
[x, y, wgt, theta, k, sigma, rs, nit, ifail] = g02hd(chi, psi, psip0,
beta, indw, isigma, x, y, wgt, theta, sigma, tol, eps, 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 var $(\epsilon_i) = \sigma^2$,

g02hd 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 the vector $r = y - X\hat{\theta}$,

ψ is a suitable weight function,

w_i are suitable weights such as those that can be calculated by using output from g02hb,

and σ may be estimated at each iteration by the median absolute deviation of the residuals $\hat{\sigma} = \mathrm{med}_i[|r_i|]/\beta_1$

or as the solution to

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

for a 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

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

(see Marazzi 1987b).

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

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

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

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

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.

Observations with zero or negative weights are not included in the solution.

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.

g02hd is based on routines in ROBETH, see Marazzi 1987b.

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 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: **chi – string containing name of m-file**

If **isigma** > 0, **chi** must return the value of the weight function χ for a given value of its argument. The value of χ must be nonnegative.

Its specification is:

```
[result] = chi(t)
```

Input Parameters

1: **t – double scalar**

The argument for which **chi** must be evaluated.

Output Parameters

1: **result – double scalar**

The result of the function.

2: **psi – string containing name of m-file**

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

Its specification is:

```
[result] = psi(t)
```

Input Parameters

1: **t – double scalar**

The argument for which **psi** must be evaluated.

Output Parameters

1: **result – double scalar**

The result of the function.

3: **psip0 – double scalar**

The value of $\psi\prime(0)$.

4: **beta – double scalar**

If **isigma** < 0, **beta** must specify the value of β_1.

For Huber and Schweppe type regressions, β_1 is the 75th percentile of the standard Normal distribution (see g01fa). 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).

If **isigma** > 0, **beta** must specify the value of β_2.

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

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

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

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

If **isigma** $= 0$, **beta** is not referenced.

Constraint: if **isigma** $\neq 0$, **beta** > 0.0.

5: **indw – int32 scalar**

Determines the type of regression to be performed.

indw $= 0$

Huber type regression.

indw < 0

Mallows type regression.

indw > 0

Schweppe type regression.

6: **isigma – int32 scalar**

Determines how σ is to be estimated.

isigma $= 0$

σ is held constant at its initial value.

isigma < 0

σ is estimated by median absolute deviation of residuals.

isigma > 0

σ is estimated using the χ function.

Constraint: **isigma** $= 0$, **isigma** < 0 or **isigma** > 0.

7: **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. **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, 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**.

8: **y(n) – double array**

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, 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**.

9: **wgt(n) – double array**

The weight for the ith observation, for $i = 1, 2, \ldots, n$.

If **indw** < 0, during calculations elements of **wgt** 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 **wgt** and the output **wgt**.

If **wgt**$(i) \leq 0$, the ith observation is not included in the analysis.

If **indw** $= 0$, **wgt** is not referenced.

10: **theta(m) – double array**

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.

11: **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: **sigma** > 0.0.

12: **tol – double scalar**

The relative precision for the final estimates. Convergence is assumed when both the relative change in the value of **sigma** and the relative change in the value of each element of **theta** are less than **tol**.

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

Constraint: **tol** > 0.0.

13: **eps – double scalar**

A relative tolerance to be used to determine the rank of X. See f04jg for further details.

If **eps** $<$ ***machine precision*** or **eps** > 1.0 then ***machine precision*** will be used in place of **tol**.

A reasonable value for **eps** is 5.0×10^{-6} where this value is possible.

14: **maxit – int32 scalar**

The maximum number of iterations that should be used during the estimation.

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

Constraint: **maxit** > 0.

15: **nitmon – int32 scalar**

Determines the amount of information that is printed on each iteration.

nitmon ≤ 0

No information is printed.

nitmon > 0

On the first and every **nitmon** iterations the values of **sigma**, **theta** and the change in **theta** during the iteration are printed.

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 arrays **x**, **theta**. (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, wk

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: **wgt(n) – double array**

Unchanged, except as described above.

4: **theta(m) – double array**

The M-estimate of θ_i, for $i = 1, 2, \ldots, m$.

5: **k – int32 scalar**

The column rank of the matrix X.

6: **sigma – double scalar**

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

7: **rs(n) – double array**

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

8: **nit – int32 scalar**

The number of iterations that were used during the estimation.

9: **ifail – int32 scalar**

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

6 Error Indicators and Warnings

Note: g02hd 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**.

ifail $= 2$

On entry, **beta** ≤ 0.0, and **isigma** $\neq 0$,
or **sigma** ≤ 0.0.

ifail $= 3$

On entry, **tol** ≤ 0.0,
or **maxit** ≤ 0.

ifail $= 4$

A value returned by the user-supplied real function **chi** function is negative.

ifail = 5

During iterations a value of **sigma** ≤ 0 was encountered.

ifail = 6

A failure occurred in f04jg . This is an extremely unlikely error. If it occurs, please consult NAG.

ifail = 7

The weighted least-squares equations are not of full rank. This may be due to the X matrix not being of full rank, in which case the results will be valid. It may also occur if some of the G_{ii} values become very small or zero, see Section 8. The rank of the equations is given by **k**. If the matrix just fails the test for nonsingularity then the result **ifail** = 7 and **k** = **m** is possible (see f04jg).

ifail = 8

The function has failed to converge in **maxit** iterations.

ifail = 9

Having removed cases with zero weight, the value of **n** − **k** ≤ 0, i.e., no degree of freedom for error. This error will only occur if **isigma** > 0.

7 Accuracy

The accuracy of the results is controlled by **tol**. For the accuracy of the weighted least-squares see f04jg.

8 Further Comments

In cases when **isigma** $\neq 0$ 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, which will lead to convergence problems and may trigger the **ifail** = 7 error.

By suitable choice of the functions user-supplied real function **chi** and user-supplied real function **psi** this function may be used for other applications of iterative weighted least-squares.

For the variance-covariance matrix of θ see g02hf.

9 Example

g02hd_chi.m

```
function [result] = chi(t)
  if (abs(t) < 1.5)
    ps=t;
  else
    ps=1.5;
  end
  result = ps*ps/2;
```

g02hd_psi.m

```
function [result] = psi(t)
  if t < -1.5
    result = -1.5;
  elseif abs(t) < 1.5
    result = t;
  else
    result = 1.5;
  end;
```

```
psip0 = 1;
beta = 0.1443849979905463;
indw = int32(1);
isigma = int32(1);
x = [1, -1, -1;
     1, -1, 1;
     1, 1, -1;
     1, 1, 1;
     1, 0, 3];
y = [10.5;
     11.3;
     12.6;
     13.4;
     17.1];
wgt = [0.4039;
     0.5012;
     0.4039;
     0.5012;
     0.3862];
theta = [0;
     0;
     0];
sigma = 1;
tol = 5e-05;
eps = 5e-06;
maxit = int32(50);
nitmon = int32(0);
[xOut, yOut, wgtOut, thetaOut, k, sigmaOut, rs, nit, ifail] = ...
       g02hd('g02hd_chi', 'g02hd_psi', psip0, beta, indw, isigma, x, y,
wgt, ...
     theta, sigma, tol, eps, maxit, nitmon)
```

```
xOut =
     1    -1    -1
     1    -1     1
     1     1    -1
     1     1     1
     1     0     3
yOut =
   10.5000
   11.3000
   12.6000
   13.4000
   17.1000
wgtOut =
    0.4039
    0.5012
    0.4039
    0.5012
    0.3862
thetaOut =
   12.2321
    1.0500
    1.2464
k =
           3
sigmaOut =
    2.7783
rs =
    0.5643
   -1.1286
    0.5643
   -1.1286
    1.1286
nit =
           5
ifail =
           0
```