NAG Toolbox

nag_correg_linregm_var_del (g02df)

1 Purpose

nag_correg_linregm_var_del (g02df) deletes an independent variable from a general linear regression model.

2 Syntax

```
[q, rss, ifail] = nag_correg_linregm_var_del(q, indx, rss, 'ip', ip)
[q, rss, ifail] = g02df(q, indx, rss, 'ip', ip)
```

Note: the interface to this routine has changed since earlier releases of the toolbox:

At Mark 22: **ip** was made optional.

3 Description

When selecting a linear regression model it is sometimes useful to drop independent variables from the model and to examine the resulting sub-model. nag_correg_linregm_var_del (g02df) updates the QR decomposition used in the computation of the linear regression model. The QR decomposition may come from nag_correg_linregm_fit (g02da) or nag_correg_linregm_var_add (g02de), or a previous call to nag_correg_linregm_var_del (g02df).

For the general linear regression model with p independent variables fitted nag_correg_linregm_fit (g02da) or nag_correg_linregm_var_add (g02de) compute a QR decomposition of the (weighted) independent variables and form an upper triangular matrix R and a vector c. To remove an independent variable R and c have to be updated. The column of R corresponding to the variable to be dropped is removed and the matrix is then restored to upper triangular form by applying a series of Givens rotations. The rotations are then applied to c. Note only the first p elements of c are affected.

The method used means that while the updated values of R and c are computed an updated value of Q from the QR decomposition is not available so a call to nag_correg_linregm_var_add (g02de) cannot be made after a call to nag_correg_linregm_var_del (g02df).

nag_correg_linregm_update (g02dd) can be used to calculate the parameter estimates, $\hat{\beta}$, from the information provided by nag_correg_linregm_var_del (g02df).

4 References

Golub G H and Van Loan C F (1996) *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

Hammarling S (1985) The singular value decomposition in multivariate statistics *SIGNUM Newsl.* **20(3)** 2–25

5 Parameters

5.1 Compulsory Input Parameters

1: **q**$(ldq, \mathbf{ip}+\mathbf{1})$ – REAL (KIND=nag_wp) array

ldq, the first dimension of the array, must satisfy the constraint $ldq \geq \mathbf{ip}$.

The results of the QR decomposition as returned by functions nag_correg_linregm_fit (g02da), nag_correg_linregm_obs_edit (g02dc), nag_correg_linregm_var_add (g02de) or nag_correg_linregm_fit_onestep (g02ee), or previous calls to nag_correg_linregm_var_del (g02df).

2: **indx** – INTEGER

Indicates which independent variable is to be deleted from the model.

Constraint: $1 \leq$ **indx** $\leq$ **ip**.

3: **rss** – REAL (KIND=nag_wp)

The residual sum of squares for the full regression.

Constraint: **rss** ≥ 0.0.

5.2 Optional Input Parameters

1: **ip** – INTEGER

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

p, the number of independent variables already in the model.

Constraint: **ip** ≥ 1.

5.3 Output Parameters

1: **q**$(ldq, \mathbf{ip}+\mathbf{1})$ – REAL (KIND=nag_wp) array

The updated QR decomposition.

2: **rss** – REAL (KIND=nag_wp)

The residual sum of squares with the (**indx**)th variable removed. Note that the residual sum of squares will only be valid if the regression is of full rank, otherwise the residual sum of squares should be obtained using nag_correg_linregm_update (g02dd).

3: **ifail** – INTEGER

ifail $= 0$ unless the function detects an error (see Section 5).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

On entry, **ip** < 1,
or $ldq <$ **ip**,
or **indx** < 1,
or **indx** $>$ **ip**,
or **rss** < 0.0.

ifail $= 2$

On entry, a diagonal element of R is zero.

ifail $= -99$

An unexpected error has been triggered by this routine. Please contact NAG.

ifail $= -399$

Your licence key may have expired or may not have been installed correctly.

ifail $= -999$

Dynamic memory allocation failed.

7 Accuracy

There will inevitably be some loss in accuracy in fitting a model by dropping terms from a more complex model rather than fitting it afresh using nag_correg_linregm_fit (g02da).

8 Further Comments

None.

9 Example

A dataset consisting of 12 observations on four independent variables and one dependent variable is read in. The full model, including a mean term, is fitted using nag_correg_linregm_fit (g02da). The value of **indx** is read in and that variable dropped from the regression. The parameter estimates are calculated by nag_correg_linregm_update (g02dd) and printed. This process is repeated until **indx** is 0.

9.1 Program Text

```
function g02df_example

fprintf('g02df example results\n\n');

x = [ 1.0 1.4 0.0 0.0;
      1.5 2.2 0.0 0.0;
      2.0 4.5 0.0 0.0;
      2.5 6.1 0.0 0.0;
      3.0 7.1 0.0 0.0;
      3.5 7.7 0.0 0.0;
      4.0 8.3 1.0 4.0;
      4.5 8.6 1.0 4.5;
      5.0 8.8 1.0 5.0;
      5.5 9.0 1.0 5.5;
      6.0 9.3 1.0 6.0;
      6.5 9.2 1.0 6.5];
y = [ 4.32  5.21  6.49  7.10  7.94  8.53 ...
      8.84  9.02  9.27  9.43  9.68  9.83];
[n,m]  = size(x);
isx    = ones(m,1,nag_int_name);
mean_p = 'M';
ip     = nag_int(m+1);

% Fit general linear regression model
[rss, idf, b, se, covar, res, h, q, svd, irank, p, wk, ifail] = ...
  g02da(mean_p, x, isx, ip, y);

% Display initial results
if svd
  fprintf('Model not of full rank\n\n', irank);
end
fprintf('Residual sum of squares = %12.4e\n', rss);
fprintf('Degrees of freedom      = %4d\n', idf);

% Variables to drop
indx = nag_int([2, 4]);
perm = double([1:ip]);

for j = 1:numel(indx)
  % drop j-th variable
```

```
  [q, rss, ifail] = g02df( ...
                           q, indx(j), rss, 'ip', ip);

  perm(indx(j):ip-1) = perm(indx(j)+1:ip);
  ip = ip - 1;

  fprintf('\nVariable %2d dropped\n', indx(j));

  % Calculate parameter estimates
  [rss, idf, b, se, covar, svd, irank, p, ifail] = ...
  g02dd(nag_int(n), ip, q, rss);

  % Display results having dropped indx(j)
  if svd
    fprintf('Model not of full rank, rank = %4d\n\n', irank);
  end
  fprintf('Residual sum of squares = %12.4e\n', rss);
  fprintf('Degrees of freedom      = %4d\n', idf);
  fprintf('\nVariable   Parameter estimate   Standard error\n\n');
  ivar = perm(1:ip)';
  fprintf('%6d%20.4e%20.4e\n',[ivar b se]');
end
```

9.2 Program Results

```
     g02df example results

Residual sum of squares =   8.4066e-02
Degrees of freedom      =    7

Variable  2 dropped
Residual sum of squares =   2.1239e-01
Degrees of freedom      =    8

Variable   Parameter estimate   Standard error

     1          3.6372e+00          1.5083e-01
     3          6.1264e-01          2.8007e-02
     4         -6.0154e-01          4.2335e-01
     5          1.6709e-01          7.8656e-02

Variable  4 dropped
Residual sum of squares =   3.3220e-01
Degrees of freedom      =    9

Variable   Parameter estimate   Standard error

     1          3.5974e+00          1.7647e-01
     3          6.2088e-01          3.2706e-02
     4          2.4247e-01          1.7235e-01
```