NAG Toolbox

nag_correg_lars_param (g02mc)

1 Purpose

nag_correg_lars_param (g02mc) calculates additional parameter estimates following Least Angle Regression (LARS), forward stagewise linear regression or Least Absolute Shrinkage and Selection Operator (LASSO) as performed by nag_correg_lars (g02ma) and nag_correg_lars_xtx (g02mb).

2 Syntax

```
[nb, ifail] = nag_correg_lars_param(b, fitsum, ktype, nk, 'nstep', nstep, 'ip', ip, 'lnk', lnk)
[nb, ifail] = g02mc(b, fitsum, ktype, nk, 'nstep', nstep, 'ip', ip, 'lnk', lnk)
```

3 Description

nag_correg_lars (g02ma) and nag_correg_lars_xtx (g02mb) fit either a LARS, forward stagewise linear regression, LASSO or positive LASSO model to a vector of n observed values, $y = \{y_i : i = 1, 2, \ldots, n\}$ and an $n \times p$ design matrix X, where the jth column of X is given by the jth independent variable x_j. The models are fit using the LARS algorithm of Efron *et al.* (2004).

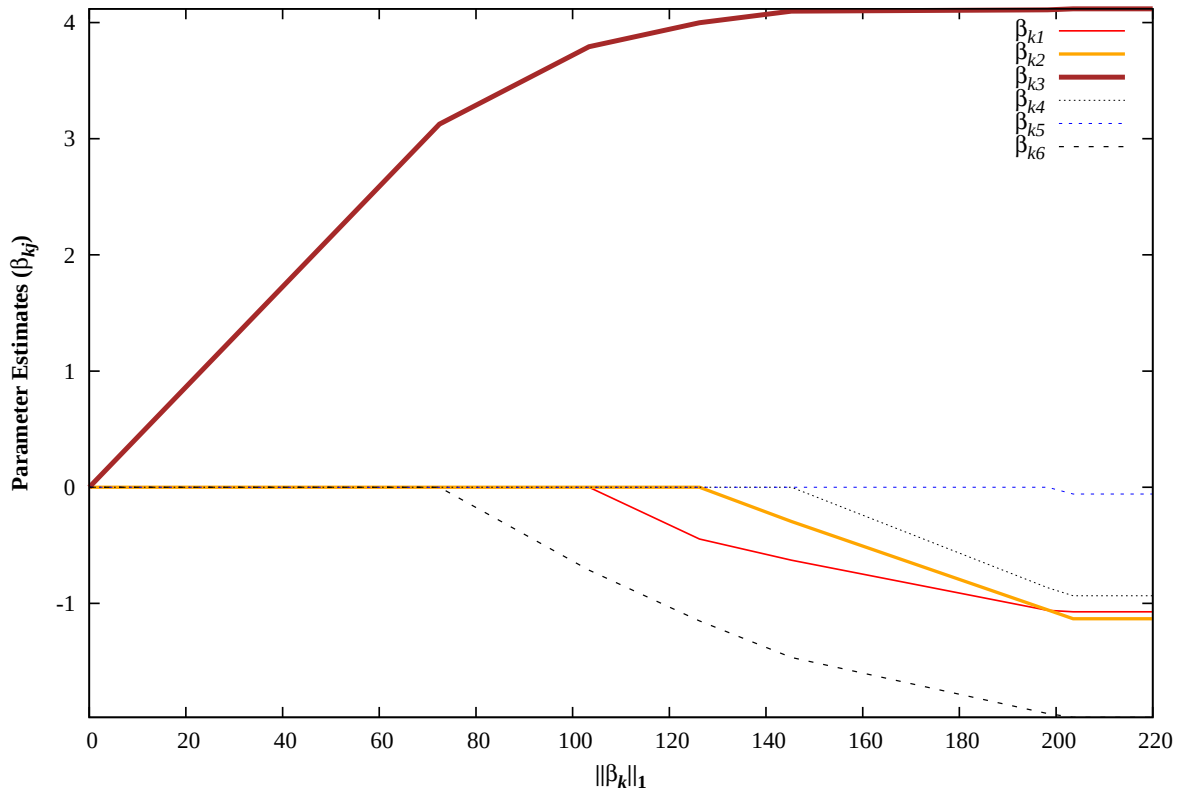

Figure 1

The full solution path for all four of these models follow a similar pattern where the parameter estimate for a given variable is piecewise linear. One such path, for a LARS model with six variables $(p = 6)$ can be seen in Figure 1. Both nag_correg_lars (g02ma) and nag_correg_lars_xtx (g02mb) return the vector of p parameter estimates, β_k, at K points along this path (so $k = 1, 2, \ldots, K$). Each point

corresponds to a step of the LARS algorithm. The number of steps taken depends on the model being fitted. In the case of a LARS model, $K = p$ and each step corresponds to a new variable being included in the model. In the case of the LASSO models, each step corresponds to either a new variable being included in the model or an existing variable being removed from the model; the value of K is therefore no longer bound by the number of parameters. For forward stagewise linear regression, each step no longer corresponds to the addition or removal of a variable; therefore the number of possible steps is often markedly greater than for a corresponding LASSO model.

nag_correg_lars_param (g02mc) uses the piecewise linear nature of the solution path to predict the parameter estimates, $\tilde{\beta}$, at a different point on this path. The location of the solution can either be defined in terms of a (fractional) step number or a function of the L_1 norm of the parameter estimates.

4 References

Efron B, Hastie T, Johnstone I and Tibshirani R (2004) Least Angle Regression *The Annals of Statistics (Volume 32)* **2** 407–499

Hastie T, Tibshirani R and Friedman J (2001) *The Elements of Statistical Learning: Data Mining, Inference and Prediction* Springer (New York)

Tibshirani R (1996) Regression Shrinkage and Selection via the Lasso *Journal of the Royal Statistics Society, Series B (Methodological) (Volume 58)* **1** 267–288

Weisberg S (1985) *Applied Linear Regression* Wiley

5 Parameters

5.1 Compulsory Input Parameters

1: **b**$(ldb, :)$ – REAL (KIND=nag_wp) array

The first dimension of the array **b** must be at least **ip**.

The second dimension of the array **b** must be at least **nstep** $+ 1$.

β the parameter estimates, as returned by nag_correg_lars (g02ma) and nag_correg_lars_xtx (g02mb), with **b**$(j, k) = \beta_{kj}$, the parameter estimate for the jth variable, for $j = 1, 2, \ldots, p$, at the kth step of the model fitting process.

Constraint: **b** should be unchanged since the last call to nag_correg_lars (g02ma) or nag_correg_lars_xtx (g02mb).

2: **fitsum**$(\mathbf{6}, \mathbf{nstep} + \mathbf{1})$ – REAL (KIND=nag_wp) array

Summaries of the model fitting process, as returned by nag_correg_lars (g02ma) and nag_correg_lars_xtx (g02mb).

Constraint: **fitsum** should be unchanged since the last call to nag_correg_lars (g02ma) or nag_correg_lars_xtx (g02mb)..

3: **ktype** – INTEGER

Indicates what target values are held in **nk**.

ktype $= 1$
: **nk** holds (fractional) LARS step numbers.

ktype $= 2$
: **nk** holds values for L_1 norm of the (scaled) parameters.

ktype $= 3$
: **nk** holds ratios with respect to the largest (scaled) L_1 norm.

ktype $= 4$
: **nk** holds values for the L_1 norm of the (unscaled) parameters.

ktype $= 5$
nk holds ratios with respect to the largest (unscaled) L_1 norm.

If nag_correg_lars (g02ma) was called with **pred** $= 0$ or 1 or nag_correg_lars_xtx (g02mb) was called with **pred** $= 0$ then the model fitting routine did not rescale the independent variables, X, prior to fitting the model and therefore there is no difference between **ktype** $= 2$ or 3 and **ktype** $= 4$ or 5.

Constraint: **ktype** $= 1$, 2, 3, 4 or 5.

4: **nk**(**lnk**) – REAL (KIND=nag_wp) array

Target values used for predicting the new set of parameter estimates.

Constraints:

if **ktype** $= 1$, $0 \leq$ **nk**$(i) \leq$ **nstep**, for $i = 1, 2, \ldots,$ **lnk**;
if **ktype** $= 2$, $0 \leq$ **nk**$(i) \leq$ **fitsum**$(1,$ **nstep**$)$, for $i = 1, 2, \ldots,$ **lnk**;
if **ktype** $= 3$ or 5, $0 \leq$ **nk**$(i) \leq 1$, for $i = 1, 2, \ldots,$ **lnk**;
if **ktype** $= 4$, $0 \leq$ **nk**$(i) \leq \|\beta_K\|_1$, for $i = 1, 2, \ldots,$ **lnk**.

5.2 Optional Input Parameters

1: **nstep** – INTEGER

Default: the second dimension of **fitsum** $- 1$

K, the number of steps carried out in the model fitting process.

Constraint: **nstep** ≥ 0.

2: **ip** – INTEGER

Default: the first dimension of the array **b**.

p, number of parameter estimates.

Constraint: **ip** ≥ 1.

3: **lnk** – INTEGER

Default: the dimension of the array **nk**.

Number of values supplied in **nk**.

Constraint: **lnk** ≥ 1.

5.3 Output Parameters

1: **nb**$(ldnb, :)$ – REAL (KIND=nag_wp) array

The first dimension of the array **nb** will be **ip**.

The second dimension of the array **nb** will be **lnk**.

$\tilde{\beta}$ the predicted parameter estimates, with **b**$(j, i) = \tilde{\beta}_{ij}$, the parameter estimate for variable j, $j = 1, 2, \ldots, p$ at the point in the fitting process associated with **nk**(i), $i = 1, 2, \ldots,$ **lnk**.

2: **ifail** – INTEGER

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

6 Error Indicators and Warnings

Note: nag_correg_lars_param (g02mc) may return useful information for one or more of the following detected errors or warnings.

Errors or warnings detected by the function:

ifail $= 11$

Constraint: **nstep** ≥ 0.

ifail $= 21$

Constraint: **ip** ≥ 1.

ifail $= 31$

b has been corrupted since the last call to nag_correg_lars (g02ma) or nag_correg_lars_xtx (g02mb).

ifail $= 41$

Constraint: $ldb \geq$ **ip**.

ifail $= 51$

fitsum has been corrupted since the last call to nag_correg_lars (g02ma) or nag_correg_lars_xtx (g02mb).

ifail $= 61$

Constraint: **ktype** $= 1$, 2, 3, 4 or 5.

ifail $= 71$

Constraint: $0 \leq$ **nk**$(i) \leq$ **nstep** for all i.

ifail $= 72$

Constraint: $0 \leq$ **nk**$(i) \leq$ **fitsum**$(1,$ **nstep**$)$ for all i.

ifail $= 73$

Constraint: $0 \leq$ **nk**$(i) \leq 1$ for all i.

ifail $= 74$

Constraint: $0 \leq$ **nk**$(i) \leq \|\beta_K\|_1$ for all i.

ifail $= 81$

Constraint: **lnk** ≥ 1.

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

Not applicable.

8 Further Comments

None.

9 Example

This example performs a LARS on a set a simulated dataset with 20 observations and 6 independent variables.

Additional parameter estimates are obtained corresponding to a LARS step number of $0.2, 1.2, 3.2, 4.5$ and 5.2. Where, for example, 4.5 corresponds to the solution halfway between that obtained at step 4 and that obtained at step 5.

9.1 Program Text

```
     function g02mc_example

fprintf('g02mc example results\n\n');

% Going to be fitting a LAR model via g02ma and getting g02ma
% to mean center y and normalise X around the mean
mtype = nag_int(1);
pred = nag_int(3);
prey = nag_int(1);

% Independent variables
d = [10.28  1.77  9.69 15.58  8.23 10.44;
      9.08  8.99 11.53  6.57 15.89 12.58;
     17.98 13.10  1.04 10.45 10.12 16.68;
     14.82 13.79 12.23  7.00  8.14  7.79;
     17.53  9.41  6.24  3.75 13.12 17.08;
      7.78 10.38  9.83  2.58 10.13  4.25;
     11.95 21.71  8.83 11.00 12.59 10.52;
     14.60 10.09 -2.70  9.89 14.67  6.49;
      3.63  9.07 12.59 14.09  9.06  8.19;
      6.35  9.79  9.40 12.79  8.38 16.79;
      4.66  3.55 16.82 13.83 21.39 13.88;
      8.32 14.04 17.17  7.93  7.39 -1.09;
     10.86 13.68  5.75 10.44 10.36 10.06;
      4.76  4.92 17.83  2.90  7.58 11.97;
      5.05 10.41  9.89  9.04  7.90 13.12;
      5.41  9.32  5.27 15.53  5.06 19.84;
      9.77  2.37  9.54 20.23  9.33  8.82;
     14.28  4.34 14.23 14.95 18.16 11.03;
     10.17  6.80  3.17  8.57 16.07 15.93;
      5.39  2.67  6.37 13.56 10.68  7.35];

% Dependent variable
y = [-46.47; -35.80; -129.22;  -42.44; -73.51;
     -26.61; -63.90;  -76.73;  -32.64; -83.29;
     -16.31;  -5.82;  -47.75;   18.38; -54.71;
     -55.62; -45.28;  -22.76; -104.32; -55.94];

% g02ma can issue warnings, but return sensible results,
% so save current warning state and turn warnings on
warn_state = nag_issue_warnings();
nag_issue_warnings(true);

% Call the model fitting routine
[b,fitsum,ifail] = g02ma(mtype,d,y);

% Reset the warning state to its initial value
nag_issue_warnings(warn_state);

% Set how the additional estimates will be specified

% Location of additional parameter estimates (as defined by the
% LARS step number)
```

```
ktype = nag_int(1);
nk = [0.2; 1.2; 3.2; 4.5; 5.2];

% Calculate the additional parameter estimates
[nb,ifail] = g02mc(b,fitsum,ktype,nk);

% Print the results
ip = size(b,1);
K = size(b,2) - 2;
lnk = size(nk,1);

fprintf(' Parameter Estimates from g02ma\n');
fprintf('  Step %s Parameter Estimate\n ',repmat(' ',1,max(ip-2,0)*5));
fprintf(repmat('-',1,5+ip*10));
fprintf('\n');
for k = 1:K
  fprintf('  %3d',k);
  for j = 1:ip
    fprintf(' %9.3f',b(j,k));
  end
  fprintf('\n');
end
fprintf('\n');

fprintf(' Additional Parameter Estimates from g02mc\n');
fprintf('   nk  %s Parameter Estimate\n ',repmat(' ',1,max(ip-2,0)*5));
fprintf(repmat('-',1,5+ip*10));
fprintf('\n');
for k = 1:lnk
  fprintf('  %4.1f',nk(k));
  for j = 1:ip
    fprintf(' %9.3f',nb(j,k));
  end
  fprintf('\n');
end
```

9.2 Program Results

```
     g02mc example results

 Parameter Estimates from g02ma
  Step                       Parameter Estimate
 ------------------------------------------------------------------
    1     0.000     0.000     3.125     0.000     0.000     0.000
    2     0.000     0.000     3.792     0.000     0.000    -0.713
    3    -0.446     0.000     3.998     0.000     0.000    -1.151
    4    -0.628    -0.295     4.098     0.000     0.000    -1.466
    5    -1.060    -1.056     4.110    -0.864     0.000    -1.948
    6    -1.073    -1.132     4.118    -0.935    -0.059    -1.981

 Additional Parameter Estimates from g02mc
   nk                        Parameter Estimate
 ------------------------------------------------------------------
   0.2     0.000     0.000     0.625     0.000     0.000     0.000
   1.2     0.000     0.000     3.258     0.000     0.000    -0.143
   3.2    -0.483    -0.059     4.018     0.000     0.000    -1.214
   4.5    -0.844    -0.676     4.104    -0.432     0.000    -1.707
   5.2    -1.062    -1.071     4.112    -0.878    -0.012    -1.955
```