NAG Toolbox

nag_correg_pls_fit (g02lc)

1 Purpose

nag_correg_pls_fit (g02lc) calculates parameter estimates for a given number of factors given the output from an orthogonal scores PLS regression (nag_correg_pls_svd (g02la) or nag_correg_pls_wold (g02lb)).

2 Syntax

```
[b, ob, vip, ifail] = nag_correg_pls_fit(nfact, p, c, w, rcond, orig, xbar,
ybar, iscale, xstd, ystd, vipopt, ycv, 'ip', ip, 'my', my, 'maxfac', maxfac)
[b, ob, vip, ifail] = g02lc(nfact, p, c, w, rcond, orig, xbar, ybar, iscale,
xstd, ystd, vipopt, ycv, 'ip', ip, 'my', my, 'maxfac', maxfac)
```

3 Description

The parameter estimates B for a l-factor orthogonal scores PLS model with m predictor variables and r response variables are given by,

$$B = W\left(P^{\mathrm{T}}W\right)^{-1}C^{\mathrm{T}}, \quad B \in \mathbb{R}^{m\times r},$$

where W is the m by k ($\geq l$) matrix of x-weights; P is the m by k matrix of x-loadings; and C is the r by k matrix of y-loadings for a fitted PLS model.

The parameter estimates B are for centred, and possibly scaled, predictor data X_1 and response data Y_1. Parameter estimates may also be given for the predictor data X and response data Y.

Optionally, nag_correg_pls_fit (g02lc) will calculate variable influence on projection (VIP) statistics, see Wold (1994).

4 References

Wold S (1994) PLS for multivariate linear modelling QSAR: chemometric methods in molecular design *Methods and Principles in Medicinal Chemistry* (ed van de Waterbeemd H) Verlag-Chemie

5 Parameters

5.1 Compulsory Input Parameters

1: **nfact** – INTEGER

l, the number of factors to include in the calculation of parameter estimates.

Constraint: $1 \leq$ **nfact** $\leq$ **maxfac**.

2: **p**(ldp, **maxfac**) – REAL (KIND=nag_wp) array

ldp, the first dimension of the array, must satisfy the constraint $ldp \geq$ **ip**.

x-loadings as returned from nag_correg_pls_svd (g02la) and nag_correg_pls_wold (g02lb).

3: **c**(ldc, **maxfac**) – REAL (KIND=nag_wp) array

ldc, the first dimension of the array, must satisfy the constraint $ldc \geq$ **my**.

y-loadings as returned from nag_correg_pls_svd (g02la) and nag_correg_pls_wold (g02lb).

4: **w**$(ldw, \mathbf{maxfac})$ – REAL (KIND=nag_wp) array

ldw, the first dimension of the array, must satisfy the constraint $ldw \geq$ **ip**.

x-weights as returned from nag_correg_pls_svd (g02la) and nag_correg_pls_wold (g02lb).

5: **rcond** – REAL (KIND=nag_wp)

Singular values of $P^{\mathrm{T}}W$ less than **rcond** times the maximum singular value are treated as zero when calculating parameter estimates. If **rcond** is negative, a value of 0.005 is used.

6: **orig** – INTEGER

Indicates how parameter estimates are calculated.

orig $= -1$
Parameter estimates for the centered, and possibly, scaled data.

orig $= 1$
Parameter estimates for the original data.

Constraint: **orig** $= -1$ or 1.

7: **xbar**$(\mathbf{ip})$ – REAL (KIND=nag_wp) array

If **orig** $= 1$, mean values of predictor variables in the model; otherwise **xbar** is not referenced.

8: **ybar**$(\mathbf{my})$ – REAL (KIND=nag_wp) array

If **orig** $= 1$, mean value of each response variable in the model; otherwise **ybar** is not referenced.

9: **iscale** – INTEGER

If **orig** $= 1$, **iscale** must take the value supplied to either nag_correg_pls_svd (g02la) or nag_correg_pls_wold (g02lb); otherwise **iscale** is not referenced.

Constraint: if **orig** $= 1$, **iscale** $= -1$, 1 or 2.

10: **xstd**$(\mathbf{ip})$ – REAL (KIND=nag_wp) array

If **orig** $= 1$ and **iscale** $\neq -1$, the scalings of predictor variables in the model as returned from either nag_correg_pls_svd (g02la) or nag_correg_pls_wold (g02lb); otherwise **xstd** is not referenced.

11: **ystd**$(\mathbf{my})$ – REAL (KIND=nag_wp) array

If **orig** $= 1$ and **iscale** $\neq -1$, the scalings of response variables as returned from either nag_correg_pls_svd (g02la) or nag_correg_pls_wold (g02lb); otherwise **ystd** is not referenced.

12: **vipopt** – INTEGER

A flag that determines variable influence on projections (VIP) options.

vipopt $= 0$
VIP are not calculated.

vipopt $= 1$
VIP are calculated for predictor variables using the mean explained variance in responses.

vipopt $=$ **my**
VIP are calculated for predictor variables for each response variable in the model.

Note that setting **vipopt** $=$ **my** when **my** $= 1$ gives the same result as setting **vipopt** $= 1$ directly.

Constraint: **vipopt** $= 0$, 1 or **my**.

13: **ycv**($ldycv$, **my**) – REAL (KIND=nag_wp) array

$ldycv$, the first dimension of the array, must satisfy the constraint if **vipopt** $\neq 0$, $ldycv \geq$ **nfact**.

If **vipopt** $\neq 0$, **ycv**(i,j) is the cumulative percentage of variance of the jth response variable explained by the first i factors, for $i = 1, 2, \ldots,$ **nfact** and $j = 1, 2, \ldots,$ **my**; otherwise **ycv** is not referenced.

5.2 Optional Input Parameters

1: **ip** – INTEGER

Default: the dimension of the arrays **xbar**, **xstd** and the first dimension of the arrays **p**, **w**. (An error is raised if these dimensions are not equal.)

m, the number of predictor variables in the fitted model.

Constraint: **ip** > 1.

2: **my** – INTEGER

Default: the dimension of the arrays **ybar**, **ystd** and the first dimension of the array **c** and the second dimension of the array **ycv**. (An error is raised if these dimensions are not equal.)

r, the number of response variables.

Constraint: **my** ≥ 1.

3: **maxfac** – INTEGER

Default: the second dimension of the arrays **p**, **c**, **w**. (An error is raised if these dimensions are not equal.)

k, the number of factors available in the PLS model.

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

5.3 Output Parameters

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

b(i,j) contains the parameter estimate for the ith predictor variable in the model for the jth response variable, for $i = 1, 2, \ldots,$ **ip** and $j = 1, 2, \ldots,$ **my**.

2: **ob**($ldob$, **my**) – REAL (KIND=nag_wp) array

If **orig** $= 1$, **ob**$(1,j)$ contains the intercept value for the jth response variable, and **ob**$(i+1,j)$ contains the parameter estimate on the original scale for the ith predictor variable in the model, for $i = 1, 2, \ldots,$ **ip** and $j = 1, 2, \ldots,$ **my**. Otherwise **ob** is not referenced.

3: **vip**($ldvip$, **vipopt**) – REAL (KIND=nag_wp) array

If **vipopt** $= 1$, **vip**$(i,1)$ contains the VIP statistic for the ith predictor variable in the model for all response variables, for $i = 1, 2, \ldots,$ **ip**.

If **vipopt** $=$ **my**, **vip**(i,j) contains the VIP statistic for the ith predictor variable in the model for the jth response variable, for $i = 1, 2, \ldots,$ **ip** and $j = 1, 2, \ldots,$ **my**.

Otherwise **vip** is not referenced.

4: **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 $= 1$

Constraint: if **orig** $= 1$, **iscale** $= -1$ or 1.

Constraint: **ip** > 1.

Constraint: **my** ≥ 1.

Constraint: **orig** $= -1$ or 1.

Constraint: **vipopt** $= 0$, 1 or **my**.

ifail $= 2$

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

Constraint: $1 \leq$ **nfact** $\leq$ **maxfac**.

Constraint: if **orig** $= 1$, $ldob \geq$ **ip** $+ 1$.

Constraint: if **vipopt** $\neq 0$, $ldvip \geq$ **ip**.

Constraint: if **vipopt** $\neq 0$, $ldycv \geq$ **nfact**.

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

Constraint: $ldc \geq$ **my**.

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

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

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

The calculations are based on the singular value decomposition of $P^{\mathrm{T}}W$.

8 Further Comments

nag_correg_pls_fit (g02lc) allocates internally $l(l + r + 4) + \max(2l, r)$ elements of double storage.

9 Example

This example reads in details of a PLS model, and a set of parameter estimates are calculated along with their VIP statistics.

9.1 Program Text

```
     function g02lc_example

fprintf('g02lc example results\n\n');

nfact = nag_int(2);
p = [-0.6708, -1.0047,  0.6505,  0.6169;
      0.4943,  0.1355, -0.9010, -0.2388;
     -0.4167, -1.9983, -0.5538,  0.8474;
      0.3930,  1.2441, -0.6967, -0.4336;
      0.3267,  0.5838, -1.4088, -0.6323;
      0.0145,  0.9607,  1.6594,  0.5361;
     -2.4471,  0.3532, -1.1321, -1.3554;
      3.5198,  0.6005,  0.2191,  0.0380;
      1.0973,  2.0635, -0.4074, -0.3522;
     -2.4466,  2.5640, -0.4806,  0.3819;
      2.2732, -1.3110, -0.7686, -1.8959;
     -1.7987,  2.4088, -0.9475, -0.4727;
      0.3629,  0.2241, -2.6332,  2.3739;
      0.3629,  0.2241, -2.6332,  2.3739;
     -0.3629, -0.2241,  2.6332, -2.3739];
c = [ 3.5425,  1.0475,  0.2548,  0.1866];
w = [-1.5764e-1  -1.5935e-1   1.7774e-1   5.4029e-2;
      8.5680e-2  -1.5240e-4  -1.2179e-1   1.0989e-1;
     -1.6931e-1  -3.7431e-1   9.4348e-2   3.1878e-1;
      1.2153e-1   2.0589e-1  -1.8144e-1  -4.4610e-2;
      7.1133e-2   5.5884e-2  -2.6916e-1   5.4912e-2;
      6.5188e-2   2.4170e-1   2.3365e-1  -1.8849e-1;
     -4.2481e-1  -1.8798e-3  -3.2413e-1  -1.1600e-1;
      6.5370e-1   1.6725e-1   2.1908e-1   2.5461e-1;
      2.8504e-1   3.6549e-1  -1.9244e-1  -1.5430e-1;
     -2.9341e-1   5.0464e-1  -1.0952e-2   1.3881e-1;
      2.9829e-1  -3.6979e-1  -4.9942e-1  -4.9355e-1;
     -2.0313e-1   4.1952e-1  -2.5684e-1  -7.5647e-2;
      5.6905e-2  -2.3197e-2  -3.0503e-1   3.9673e-1;
      5.6905e-2  -2.3197e-2  -3.0503e-1   3.9673e-1;
     -5.6905e-2   2.3197e-2   3.0503e-1  -3.9673e-1];
vipopt = nag_int(1);
ycv  = [89.638060; 97.476270; 97.939839; 98.188474];

% Means and scalings
orig = nag_int(1);
xbar = [-2.6137; -2.3614; -1.0449;  2.8614;  0.3156;
         -0.2641; -0.3146; -1.1221;  0.2401;  0.4694;
         -1.9619;  0.1691;  2.5664;  1.3741; -2.7821];
ybar = [0.452];
iscale = nag_int(1);
xstd = [1.4956;   1.3233;  0.5829;  0.7735;  0.6247;
         0.7966;   2.4113;  2.0421;  0.4678;  0.8197;
         0.9420;   0.1735;  1.0475;  0.1359;  1.3853];
ystd = [0.9062];

% Calculate predictions
rcond = -1;
[b, ob, vip, ifail] = ...
g02lc( ...
       nfact, p, c, w, rcond, orig, xbar, ybar, ...
       iscale, xstd, ystd, vipopt, ycv);

% Display results
disp('Parameter estimates');
disp(b);
disp('Intercept values');
disp(ob);
disp('VIP statistics');
disp(vip);
```

9.2 Program Results

```
     g02lc example results

Parameter estimates
   -0.1383
    0.0572
   -0.1906
    0.1238
    0.0591
    0.0936
   -0.2842
    0.4713
    0.2661
   -0.0914
    0.1226
   -0.0488
    0.0332
    0.0332
   -0.0332

Intercept values
   -0.4374
   -0.0838
    0.0392
   -0.2964
    0.1451
    0.0857
    0.1065
   -0.1068
    0.2091
    0.5155
   -0.1011
    0.1180
   -0.2548
    0.0287
    0.2214
   -0.0217

VIP statistics
    0.6111
    0.3182
    0.7513
    0.5048
    0.2712
    0.3593
    1.5777
    2.4348
    1.1322
    1.2226
    1.1799
    0.8840
    0.2129
    0.2129
    0.2129
```