NAG Toolbox

nag_mv_factor_score (g03cc)

1 Purpose

nag_mv_factor_score (g03cc) computes factor score coefficients from the result of fitting a factor analysis model by maximum likelihood as performed by nag_mv_factor (g03ca).

2 Syntax

```
[fs, ifail] = nag_mv_factor_score(method, rotate, fl, psi, e, r, 'nvar', nvar,
'nfac', nfac)
```

```
[fs, ifail] = g03cc(method, rotate, fl, psi, e, r, 'nvar', nvar, 'nfac', nfac)
```

3 Description

A factor analysis model aims to account for the covariances among p variables, observed on n individuals, in terms of a smaller number, k, of unobserved variables or factors. The values of the factors for an individual are known as factor scores. nag_mv_factor (g03ca) fits the factor analysis model by maximum likelihood and returns the estimated factor loading matrix, Λ, and the diagonal matrix of variances of the unique components, Ψ. To obtain estimates of the factors, a p by k matrix of factor score coefficients, Φ, is formed. The estimated vector of factor scores, $\hat{f}$, is then given by:

$$\hat{f} = x^{\mathrm{T}}\Phi,$$

where x is the vector of observed variables for an individual.

There are two commonly used methods of obtaining factor score coefficients.

The regression method:

$$\Phi = \Psi^{-1}\Lambda\left(I + \Lambda^{\mathrm{T}}\Psi^{-1}\Lambda\right)^{-1},$$

and Bartlett's method:

$$\Phi = \Psi^{-1}\Lambda\left(\Lambda^{\mathrm{T}}\Psi^{-1}\Lambda\right)^{-1}.$$

See Lawley and Maxwell (1971) for details of both methods. In the regression method as given above, it is assumed that the factors are not correlated and have unit variance; this is true for models fitted by nag_mv_factor (g03ca). Further, for models fitted by nag_mv_factor (g03ca),

$$\Lambda^{\mathrm{T}}\Psi^{-1}\Lambda = \Theta - I,$$

where Θ is the diagonal matrix of eigenvalues of the matrix S^*, as described in nag_mv_factor (g03ca).

The factors may be orthogonally rotated using an orthogonal rotation matrix, R, as computed by nag_mv_rot_orthomax (g03ba). The factor scores for the rotated matrix are then given by ΛR.

4 References

Lawley D N and Maxwell A E (1971) *Factor Analysis as a Statistical Method* (2nd Edition) Butterworths

5 Parameters

5.1 Compulsory Input Parameters

1: **method** – CHARACTER(1)

Indicates which method is to be used to compute the factor score coefficients.

method = 'R'
: The regression method is used.

method = 'B'
: Bartlett's method is used.

Constraint: **method** = 'B' or 'R'.

2: **rotate** – CHARACTER(1)

Indicates whether a rotation is to be applied.

rotate = 'R'
: A rotation will be applied to the coefficients and the rotation matrix, R, must be given in **r**.

rotate = 'U'
: No rotation is applied.

Constraint: **rotate** = 'R' or 'U'.

3: **fl**($ldfl$, **nfac**) – REAL (KIND=nag_wp) array

$ldfl$, the first dimension of the array, must satisfy the constraint $ldfl \geq$ **nvar**.

Λ, the matrix of unrotated factor loadings as returned by nag_mv_factor (g03ca).

4: **psi**(**nvar**) – REAL (KIND=nag_wp) array

The diagonal elements of Ψ, as returned by nag_mv_factor (g03ca).

Constraint: **psi**(i) > 0.0, for $i = 1, 2, \ldots, p$.

5: **e**(**nvar**) – REAL (KIND=nag_wp) array

The eigenvalues of the matrix S^*, as returned by nag_mv_factor (g03ca).

Constraint: **e**(i) > 1.0, for $i = 1, 2, \ldots, p$.

6: **r**(ldr, :) – REAL (KIND=nag_wp) array

The first dimension, ldr, of the array **r** must satisfy

if **rotate** = 'R', $ldr \geq$ **nfac**;
otherwise 1.

The second dimension of the array **r** must be at least 1 if **rotate** = 'U' and at least **nfac** if **rotate** = 'R'.

If **rotate** = 'R', **r** must contain the orthogonal rotation matrix, R, as returned by nag_mv_rot_orthomax (g03ba).

If **rotate** = 'U', **r** need not be set.

5.2 Optional Input Parameters

1: **nvar** – INTEGER

Default: the dimension of the arrays **psi**, **e** and the first dimension of the array **fl**. (An error is raised if these dimensions are not equal.)

p, the number of observed variables in the factor analysis.

Constraint: **nvar** $\geq$ **nfac**.

2: **nfac** – INTEGER

Default: the second dimension of the array **fl**.

k, the number of factors in the factor analysis.

Constraint: **nfac** ≥ 1.

5.3 Output Parameters

1: **fs**($ldfs$, **nfac**) – REAL (KIND=nag_wp) array

The matrix of factor score coefficients, Φ. **fs**(i,j) contains the factor score coefficient for the jth factor and the ith observed variable, for $i = 1, 2, \ldots, p$ and $j = 1, 2, \ldots, k$.

2: **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, **nfac** < 1,
or **nvar** $<$ **nfac**,
or $ldfl <$ **nvar**,
or $ldfs <$ **nvar**,
or **rotate** = 'R' and $ldr <$ **nfac**,
or **method** $\neq$ 'R' or 'B',
or **rotate** $\neq$ 'R' or 'U'.

ifail $= 2$

On entry, a value of **psi** ≤ 0.0,
or a value of **e** ≤ 1.0.

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

Accuracy will depend on the accuracy requested when computing the estimated factor loadings using nag_mv_factor (g03ca).

8 Further Comments

If principal component analysis is required the function nag_mv_prin_comp (g03aa) computes the principal component scores directly. Hence, the factor score coefficients are not needed.

9 Example

This example is taken from Lawley and Maxwell (1971). The correlation matrix for 220 observations on six school subjects is input and a factor analysis model with two factors fitted using nag_mv_factor (g03ca). The factor score coefficients are computed using the regression method.

9.1 Program Text

```
function g03cc_example

fprintf('g03cc example results\n\n');

matrix = 'C';
n = nag_int(220);
x = [1.000 0.439 0.410 0.288 0.329 0.248;
     0.439 1.000 0.351 0.354 0.320 0.329;
     0.410 0.351 1.000 0.164 0.190 0.181;
     0.288 0.354 0.164 1.000 0.595 0.470;
     0.329 0.320 0.190 0.595 1.000 0.464;
     0.248 0.329 0.181 0.470 0.464 1.000];

nvar = nag_int(size(x,1));
isx  = ones(nvar,1,nag_int_name);
nfac = nag_int(2);
iop  = [nag_int(1); -1;  500;  3;  5];

% Fit factor analysis model
[e, stat, com, psi, res, fl, ifail] = ...
  g03ca( ...
  matrix, n, x, nvar, isx, nfac, iop);

fprintf(' Loadings, Communalities and Psi\n\n');
for i = 1:nvar
  fprintf('%8.3f', fl(i,1:nfac), com(i), psi(i));
  fprintf('\n');
end

% Compute factor scores
method = 'R';
rotate = 'U';
r      = [0];
[fs, ifail] = g03cc( ...
      method, rotate, fl, psi, e, r);

mtitle = 'Factor score coefficients';
matrix = 'General';
diag   = ' ';

fprintf('\n');
[ifail] = x04ca( ...
                 matrix, diag, fs, mtitle);
```

9.2 Program Results

```
     g03cc example results

 Loadings, Communalities and Psi

   0.553  -0.429   0.490   0.510
   0.568  -0.288   0.406   0.594
   0.392  -0.450   0.356   0.644
   0.740   0.273   0.623   0.377
   0.724   0.211   0.569   0.431
   0.595   0.132   0.372   0.628

 Factor score coefficients
          1       2
 1   0.1932 -0.3920
```

```
 2   0.1703 -0.2265
 3   0.1085 -0.3262
 4   0.3495  0.3374
 5   0.2989  0.2286
 6   0.1688  0.0978
```