NAG Toolbox

nag_mv_factor (g03ca)

1 Purpose

nag_mv_factor (g03ca) computes the maximum likelihood estimates of the arguments of a factor analysis model. Either the data matrix or a correlation/covariance matrix may be input. Factor loadings, communalities and residual correlations are returned.

2 Syntax

```
[e, stat, com, psi, res, fl, ifail] = nag_mv_factor(matrix, n, x, nvar, isx, nfac, iop, 'm', m, 'wt', wt)
[e, stat, com, psi, res, fl, ifail] = g03ca(matrix, n, x, nvar, isx, nfac, iop, 'm', m, 'wt', wt)
```

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

At Mark 24: *weight* was removed from the interface; **wt** was made optional.

3 Description

Let p variables, $x_1, x_2, \ldots, x_p$, with variance-covariance matrix Σ be observed. The aim of factor analysis is to account for the covariances in these p variables in terms of a smaller number, k, of hypothetical variables, or factors, $f_1, f_2, \ldots, f_k$. These are assumed to be independent and to have unit variance. The relationship between the observed variables and the factors is given by the model:

$$x_i = \sum_{j=1}^{k} \lambda_{ij} f_j + e_i, \quad i = 1, 2, \ldots, p$$

where λ_{ij}, for $i = 1, 2, \ldots, p$ and $j = 1, 2, \ldots, k$, are the factor loadings and e_i, for $i = 1, 2, \ldots, p$, are independent random variables with variances ψ_i, for $i = 1, 2, \ldots, p$. The ψ_i represent the unique component of the variation of each observed variable. The proportion of variation for each variable accounted for by the factors is known as the communality. For this function it is assumed that both the k factors and the e_i's follow independent Normal distributions.

The model for the variance-covariance matrix, Σ, can be written as:

$$\Sigma = \Lambda\Lambda^{\mathrm{T}} + \Psi \tag{1}$$

where Λ is the matrix of the factor loadings, λ_{ij}, and Ψ is a diagonal matrix of unique variances, ψ_i, for $i = 1, 2, \ldots, p$.

The estimation of the arguments of the model, Λ and Ψ, by maximum likelihood is described by Lawley and Maxwell (1971). The log-likelihood is:

$$-\tfrac{1}{2}(n-1)\log\left(|\Sigma|\right) - \tfrac{1}{2}(n-1)\mathrm{trace}\left(S, \Sigma^{-1}\right) + \text{constant},$$

where n is the number of observations, S is the sample variance-covariance matrix or, if weights are used, S is the weighted sample variance-covariance matrix and n is the effective number of observations, that is, the sum of the weights. The constant is independent of the arguments of the model. A two stage maximization is employed. It makes use of the function $F(\Psi)$, which is, up to a constant, $-2/(n-1)$ times the log-likelihood maximized over Λ. This is then minimized with respect to Ψ to give the estimates, $\hat{\Psi}$, of Ψ. The function $F(\Psi)$ can be written as:

$$F(\Psi) = \sum_{j=k+1}^{p} \left(\theta_j - \log \theta_j\right) - (p-k)$$

where values θ_j, for $j = 1, 2, \ldots, p$ are the eigenvalues of the matrix:

$$S^* = \Psi^{-1/2} S \Psi^{-1/2}.$$

The estimates $\hat{\Lambda}$, of Λ, are then given by scaling the eigenvectors of S^*, which are denoted by V:

$$\hat{\Lambda} = \Psi^{1/2} V (\Theta - I)^{1/2}.$$

where Θ is the diagonal matrix with elements θ_i, and I is the identity matrix.

The minimization of $F(\Psi)$ is performed using nag_opt_bounds_mod_deriv2_comp (e04lb) which uses a modified Newton algorithm. The computation of the Hessian matrix is described by Clark (1970). However, instead of using the eigenvalue decomposition of the matrix S^* as described above, the singular value decomposition of the matrix $R\Psi^{-1/2}$ is used, where R is obtained either from the QR decomposition of the (scaled) mean centred data matrix or from the Cholesky decomposition of the correlation/covariance matrix. The function nag_opt_bounds_mod_deriv2_comp (e04lb) ensures that the values of ψ_i are greater than a given small positive quantity, δ, so that the communality is always less than one. This avoids the so called Heywood cases.

In addition to the values of Λ, Ψ and the communalities, nag_mv_factor (g03ca) returns the residual correlations, i.e., the off-diagonal elements of $C - \left(\Lambda\Lambda^{\mathrm{T}} + \Psi\right)$ where C is the sample correlation matrix. nag_mv_factor (g03ca) also returns the test statistic:

$$\chi^2 = [n - 1 - (2p+5)/6 - 2k/3] F\left(\hat{\Psi}\right)$$

which can be used to test the goodness-of-fit of the model (1), see Lawley and Maxwell (1971) and Morrison (1967).

4 References

Clark M R B (1970) A rapidly convergent method for maximum likelihood factor analysis *British J. Math. Statist. Psych.*

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

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

Morrison D F (1967) *Multivariate Statistical Methods* McGraw–Hill

5 Parameters

5.1 Compulsory Input Parameters

1: **matrix** – CHARACTER(1)

Selects the type of matrix on which factor analysis is to be performed.

matrix = 'D'
The data matrix will be input in **x** and factor analysis will be computed for the correlation matrix.

matrix = 'S'
The data matrix will be input in **x** and factor analysis will be computed for the covariance matrix, i.e., the results are scaled as described in Section 9.

matrix = 'C'
The correlation/variance-covariance matrix will be input in **x** and factor analysis computed for this matrix.

See Section 9.

Constraint: **matrix** = 'D', 'S' or 'C'.

2: **n** – INTEGER

If **matrix** = 'D' or 'S' the number of observations in the data array **x**.

If **matrix** = 'C' the (effective) number of observations used in computing the (possibly weighted) correlation/variance-covariance matrix input in **x**.

Constraint: **n** > **nvar**.

3: **x**(ldx, **m**) – REAL (KIND=nag_wp) array

ldx, the first dimension of the array, must satisfy the constraint

if **matrix** = 'D' or 'S', $ldx \geq$ **n**;
if **matrix** = 'C', $ldx \geq$ **m**.

.

The input matrix.

If **matrix** = 'D' or 'S', **x** must contain the data matrix, i.e., **x**(i, j) must contain the ith observation for the jth variable, for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots,$ **m**.

If **matrix** = 'C', **x** must contain the correlation or variance-covariance matrix. Only the upper triangular part is required.

4: **nvar** – INTEGER

p, the number of variables in the factor analysis.

Constraint: **nvar** ≥ 2.

5: **isx**(**m**) – INTEGER array

isx(j) indicates whether or not the jth variable is included in the factor analysis. If **isx**(j) ≥ 1, the variable represented by the jth column of **x** is included in the analysis; otherwise it is excluded, for $j = 1, 2, \ldots,$ **m**.

Constraint: **isx**(j) > 0 for **nvar** values of j.

6: **nfac** – INTEGER

k, the number of factors.

Constraint: $1 \leq$ **nfac** $\leq$ **nvar**.

7: **iop**(**5**) – INTEGER array

Options for the optimization. There are four options to be set:

$iprint$	controls iteration monitoring; if $iprint \leq 0$, then there is no printing of information else if $iprint > 0$, then information is printed at every *iprint* iterations. The information printed consists of the value of $F(\Psi)$ at that iteration, the number of evaluations of $F(\Psi)$, the current estimates of the communalities and an indication of whether or not they are at the boundary.
$maxfun$	the maximum number of function evaluations.
acc	the required accuracy for the estimates of ψ_i.
eps	a lower bound for the values of ψ, see Section 3.

Let $\epsilon =$ ***machine precision*** then if **iop**(1) = 0, then the following default values are used:

$iprint = -1$

$maxfun = 100p$

$acc = 10\sqrt{\epsilon}$

$eps = \epsilon$

If **iop**$(1) \neq 0$, then

$iprint =$ **iop**(2)

$maxfun =$ **iop**(3)

$acc = 10^{-l}$ where $l =$ **iop**(4)

$eps = 10^{-l}$ where $l =$ **iop**(5)

Constraint: if **iop**$(1) \neq 0$, **iop**(i) must be such that $maxfun \geq 1$, $\epsilon \leq acc < 1$ and $\epsilon \leq eps < 1$, for $i = 3, 4, 5$.

5.2 Optional Input Parameters

1: **m** – INTEGER

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

The number of variables in the data/correlation/variance-covariance matrix.

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

2: **wt**(:) – REAL (KIND=nag_wp) array

The dimension of the array **wt** must be at least **n** if $weight =$ 'W' and **matrix** = 'D' or 'S', and at least 1 otherwise

If $weight =$ 'W' and **matrix** = 'D' or 'S', **wt** must contain the weights to be used in the factor analysis. The effective number of observations in the analysis will then be the sum of weights. If **wt**$(i) = 0.0$, the ith observation is not included in the analysis.

If $weight =$ 'U' or **matrix** = 'C', **wt** is not referenced and the effective number of observations is n.

Constraint: if $weight =$ 'W', the sum of weights $>$ **nvar**, **wt**$(i) \geq 0.0$, for $i = 1, 2, \ldots, n$.

5.3 Output Parameters

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

The eigenvalues θ_i, for $i = 1, 2, \ldots, p$.

2: **stat**(**4**) – REAL (KIND=nag_wp) array

The test statistics.

stat(1)
Contains the value $F\left(\hat{\Psi}\right)$.

stat(2)
Contains the test statistic, χ^2.

stat(3)
Contains the degrees of freedom associated with the test statistic.

stat(4)
Contains the significance level.

3: **com**(**nvar**) – REAL (KIND=nag_wp) array

The communalities.

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

The estimates of ψ_i, for $i = 1, 2, \ldots, p$.

5: **res**(**nvar** $\times$ (**nvar** $-$ **1**)/**2**) – REAL (KIND=nag_wp) array

The residual correlations. The residual correlation for the ith and jth variables is stored in **res**$((j-1)(j-2)/2+i)$, $i < j$.

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

The factor loadings. **fl**(i, j) contains λ_{ij}, for $i = 1, 2, \ldots, p$ and $j = 1, 2, \ldots, k$.

7: **ifail** – INTEGER

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

6 Error Indicators and Warnings

Note: nag_mv_factor (g03ca) may return useful information for one or more of the following detected errors or warnings.

Errors or warnings detected by the function:

ifail $= 1$

On entry, $ldfl <$ **nvar**,
or **nvar** < 2,
or **n** $\leq$ **nvar**,
or **nfac** < 1,
or **nvar** $<$ **nfac**,
or **m** $<$ **nvar**,
or **matrix** $=$ 'D' or 'S' and $ldx <$ **n**,
or **matrix** $=$ 'C' and $ldx <$ **m**,
or **matrix** $\neq$ 'D', 'S' or 'C',
or **matrix** $=$ 'D' or 'S' and $weight \neq$ 'U' or 'W',
or **iop**$(1) \neq 0$ and **iop**(3) is such that $maxfun < 1$,
or **iop**$(1) \neq 0$ and **iop**(4) is such that $acc \geq 1.0$,
or **iop**$(1) \neq 0$ and **iop**(4) is such that $acc <$ ***machine precision***,
or **iop**$(1) \neq 0$ and **iop**(5) is such that $eps \geq 1.0$,
or **iop**$(1) \neq 0$ and **iop**(5) is such that $eps <$ ***machine precision***,
or **matrix** $=$ 'C' and $lwk <$ (5 $\times$ **nvar** $\times$ **nvar** $+$ 33 $\times$ **nvar** $-$ 4)/2,
or **matrix** $=$ 'D' or 'S' and
$lwk <$ max((5 $\times$ **nvar** $\times$ **nvar** $+$ 33 $\times$ **nvar** $-$ 4)/2, **n** $\times$ **nvar** $+$ 7 $\times$ **nvar** $+$ **nvar** $\times$ (**nvar** $-$ 1)/2).

ifail $= 2$

On entry, $weight =$ 'W' and a value of **wt** < 0.0.

ifail $= 3$

On entry, there are not exactly **nvar** elements of **isx** > 0, or the effective number of observations $\leq$ **nvar**.

ifail $= 4$

On entry, **matrix** $=$ 'D' or 'S' and the data matrix is not of full column rank, or **matrix** $=$ 'C' and the input correlation/variance-covariance matrix is not positive definite.

This exit may also be caused by two of the eigenvalues of S^* being equal; this is rare (see Lawley and Maxwell (1971)), and may be due to the data/correlation matrix being almost singular.

ifail $= 5$

A singular value decomposition has failed to converge. This is a very unlikely error exit.

ifail $= 6$

The estimation procedure has failed to converge in the given number of iterations. Change **iop** to either increase number of iterations $maxfun$ or increase the value of acc.

ifail $= 7$ (*warning*)

The convergence is not certain but a lower point could not be found. See nag_opt_bounds_mod_deriv2_comp (e04lb) for further details. In this case all results are computed.

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 accuracy achieved is discussed in nag_opt_bounds_mod_deriv2_comp (e04lb) with the value of the argument **xtol** given by acc as described in parameter **iop**.

8 Further Comments

The factor loadings may be orthogonally rotated by using nag_mv_rot_orthomax (g03ba) and factor score coefficients can be computed using nag_mv_factor_score (g03cc). The maximum likelihood estimators are invariant to a change in scale. This means that the results obtained will be the same (up to a scaling factor) if either the correlation matrix or the variance-covariance matrix is used. As the correlation matrix ensures that all values of ψ_i are between 0 and 1 it will lead to a more efficient optimization. In the situation when the data matrix is input the results are always computed for the correlation matrix and then scaled if the results for the covariance matrix are required. When you input the covariance/correlation matrix the input matrix itself is used and you are advised to input the correlation matrix rather than the covariance matrix.

9 Example

This example is taken from Lawley and Maxwell (1971). The correlation matrix for nine variables is input and the arguments of a factor analysis model with three factors are estimated and printed.

9.1 Program Text

```
function g03ca_example

fprintf('g03ca example results\n\n');

matrix = 'C';
n = nag_int(211);
x = [1,     0.523, 0.395, 0.471, 0.346, 0.426, 0.576, 0.434, 0.639;
     0.523, 1,     0.479, 0.506, 0.418, 0.462, 0.547, 0.283, 0.645;
     0.395, 0.479, 1,     0.355, 0.270, 0.254, 0.452, 0.219, 0.504;
```

```
     0.471, 0.506, 0.355, 1,     0.691, 0.791, 0.443, 0.285, 0.505;
     0.346, 0.418, 0.270, 0.691, 1,     0.679, 0.383, 0.149, 0.409;
     0.426, 0.462, 0.254, 0.791, 0.679, 1,     0.372, 0.314, 0.472;
     0.576, 0.547, 0.452, 0.443, 0.383, 0.372, 1,     0.385, 0.68;
     0.434, 0.283, 0.219, 0.285, 0.149, 0.314, 0.385, 1,     0.47;
     0.639, 0.645, 0.504, 0.505, 0.409, 0.472, 0.680, 0.470, 1];
nvar = nag_int(size(x,1));
isx  = ones(nvar,1,nag_int_name);
nfac = nag_int(3);
iop = [nag_int(1); -1;  500;  2;  5];

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

disp(' Eigenvalues');
fprintf('%12.4e%12.4e%12.4e\n',e);
fprintf('\n     Test Statistic = %6.3f\n', stat(2));
fprintf('                 df = %6.3f\n', stat(3));
fprintf(' Significance level = %6.3f\n', stat(4));
fprintf('\n Residuals\n\n');
l = 1;
for i = 1:nvar-1
  fprintf('%8.3f', res(l:(l+i-1)));
  fprintf('\n');
  l = l + i;
end
fprintf('\n\n Loadings, Communalities and PSI\n\n');
for i = 1:nvar
  fprintf('%8.3f', fl(i,1:nfac), com(i), psi(i));
  fprintf('\n');
end
```

9.2 Program Results

```
     g03ca example results

 Eigenvalues
  1.5968e+01  4.3577e+00  1.8474e+00
  1.1560e+00  1.1190e+00  1.0271e+00
  9.2574e-01  8.9508e-01  8.7710e-01

     Test Statistic =  7.149
                 df = 12.000
 Significance level =  0.848

 Residuals

   0.000
  -0.013   0.022
   0.011  -0.005   0.023
  -0.010  -0.019  -0.016   0.003
  -0.005   0.011  -0.012  -0.001  -0.001
   0.015  -0.022  -0.011   0.002   0.029  -0.012
  -0.001  -0.011   0.013   0.005  -0.006  -0.001   0.003
  -0.006   0.010  -0.005  -0.011   0.002   0.007   0.003  -0.001

 Loadings, Communalities and PSI

   0.664  -0.321   0.074   0.550   0.450
   0.689  -0.247  -0.193   0.573   0.427
   0.493  -0.302  -0.222   0.383   0.617
   0.837   0.292  -0.035   0.788   0.212
   0.705   0.315  -0.153   0.619   0.381
   0.819   0.377   0.105   0.823   0.177
   0.661  -0.396  -0.078   0.600   0.400
   0.458  -0.296   0.491   0.538   0.462
   0.766  -0.427  -0.012   0.769   0.231
```