NAG Library Routine Document

G03BDF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

G03BDF calculates a ProMax rotation, given information following an orthogonal rotation.

2 Specification

```
SUBROUTINE G03BDF (STAND, N, M, X, LDX, RO, LDRO, POWER, FP, LDFP, R,        &
                   LDR, PHI, LDPHI, FS, LDFS, IFAIL)
INTEGER            N, M, LDX, LDRO, LDFP, LDR, LDPHI, LDFS, IFAIL
REAL (KIND=nag_wp) X(LDX,M), RO(LDRO,M), POWER, FP(LDFP,M), R(LDR,M),        &
                   PHI(LDPHI,M), FS(LDFS,M)
CHARACTER(1)       STAND
```

3 Description

Let X and Y denote n by m matrices each representing a set of n points in an m-dimensional space. The X matrix is a matrix of loadings as returned by G03BAF, that is following an orthogonal rotation of a loadings matrix Z. The target matrix Y is calculated as a power transformation of X that preserves the sign of the loadings. Let X_{ij} and Y_{ij} denote the (i,j)th element of matrices X and Y. Given a value greater than one for the exponent p:

$$Y_{ij} = \delta_{ij} \left\| X_{ij} \right\|^p,$$

for

$i = 1, 2, \ldots, n$;

$j = 1, 2, \ldots, m$;

$$\delta_{ij} = \begin{cases} -1, & \text{if } X_{ij} < 0; \\ 1, & \text{otherwise.} \end{cases}$$

The above power transformation tends to increase the difference between high and low values of loadings and is intended to increase the interpretability of a solution.

In the second step a solution of:

$$XW = Y, \quad X, Y \in \mathbb{R}^{n \times m}, W \in \mathbb{R}^{m \times m},$$

is found for W in the least squares sense by use of singular value decomposition of the orthogonal loadings X. The ProMax rotation matrix R is then given by

$$R = OW\tilde{W}, \quad O, \tilde{W} \in \mathbb{R}^{m \times m},$$

where O is the rotation matrix from an orthogonal transformation, and $\tilde{W}$ is a matrix with the square root of diagonal elements of $\left(W^{\mathrm{T}}W\right)^{-1}$ on its diagonal and zeros elsewhere.

The ProMax factor pattern matrix P is given by

$$P = XW\tilde{W}, \quad P \in \mathbb{R}^{n \times m};$$

the inter-factor correlations Φ are given by

$$\Phi = \left(Q^{\mathrm{T}}Q\right)^{-1}, \quad \Phi \in \mathbb{R}^{m \times m};$$

where $Q = W\tilde{W}$; and the factor structure S is given by

$$S = P\Phi, \quad S \in \mathbb{R}^{n \times m}.$$

Optionally, the rows of target matrix Y can be scaled by the communalities of loadings.

4 References

None.

5 Arguments

1: STAND – CHARACTER(1) *Input*

On entry: indicates how loadings are normalized.

STAND = 'S'
Rows of Y are (Kaiser) normalized by the communalities of the loadings.

STAND = 'U'
Rows are not normalized.

Constraint: STAND = 'U' or 'S'.

2: N – INTEGER *Input*

On entry: n, the number of points.

Constraint: N $\geq$ M.

3: M – INTEGER *Input*

On entry: m, the number of dimensions.

Constraint: M $\geq$ 1.

4: X(LDX, M) – REAL (KIND=nag_wp) array *Input*

On entry: the loadings matrix following an orthogonal rotation, X.

5: LDX – INTEGER *Input*

On entry: the first dimension of the array X as declared in the (sub)program from which G03BDF is called.

Constraint: LDX $\geq$ N.

6: RO(LDRO, M) – REAL (KIND=nag_wp) array *Input*

On entry: the orthogonal rotation matrix, O.

7: LDRO – INTEGER *Input*

On entry: the first dimension of the array RO as declared in the (sub)program from which G03BDF is called.

Constraint: LDRO $\geq$ M.

8: POWER – REAL (KIND=nag_wp) *Input*

On entry: p, the value of exponent.

Constraint: POWER $>$ 1.0.

9: FP(LDFP, M) – REAL (KIND=nag_wp) array *Output*

On exit: the factor pattern matrix, P.

10: LDFP – INTEGER *Input*

On entry: the first dimension of the array FP as declared in the (sub)program from which G03BDF is called.

Constraint: LDFP $\geq$ N.

11: R(LDR, M) – REAL (KIND=nag_wp) array *Output*

On exit: the ProMax rotation matrix, R.

12: LDR – INTEGER *Input*

On entry: the first dimension of the array R as declared in the (sub)program from which G03BDF is called.

Constraint: LDR $\geq$ M.

13: PHI(LDPHI, M) – REAL (KIND=nag_wp) array *Output*

On exit: the matrix of inter-factor correlations, Φ.

14: LDPHI – INTEGER *Input*

On entry: the first dimension of the array PHI as declared in the (sub)program from which G03BDF is called.

Constraint: LDPHI $\geq$ M.

15: FS(LDFS, M) – REAL (KIND=nag_wp) array *Output*

On exit: the factor structure matrix, S.

16: LDFS – INTEGER *Input*

On entry: the first dimension of the array FS as declared in the (sub)program from which G03BDF is called.

Constraint: LDFS $\geq$ N.

17: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this argument you should refer to Section 3.4 in How to Use the NAG Library and its Documentation for details.

For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this argument, the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

On exit: IFAIL $= 0$ unless the routine detects an error or a warning has been flagged (see Section 6).

6 Error Indicators and Warnings

If on entry IFAIL $= 0$ or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL $= 1$

On entry, M $= \langle value \rangle$.
Constraint: M ≥ 1.

On entry, POWER $= \langle value \rangle$.
Constraint: POWER > 1.0.

On entry, STAND $= \langle value \rangle$.
Constraint: STAND = 'U' or 'S'.

IFAIL $= 2$

On entry, LDFP $= \langle value \rangle$ and N $= \langle value \rangle$.
Constraint: LDFP $\geq$ N.

On entry, LDFS $= \langle value \rangle$ and N $= \langle value \rangle$.
Constraint: LDFS $\geq$ N.

On entry, LDPHI $= \langle value \rangle$ and M $= \langle value \rangle$.
Constraint: LDPHI $\geq$ M.

On entry, LDR $= \langle value \rangle$ and M $= \langle value \rangle$.
Constraint: LDR $\geq$ M.

On entry, LDRO $= \langle value \rangle$ and M $= \langle value \rangle$.
Constraint: LDRO $\geq$ M.

On entry, LDX $= \langle value \rangle$ and N $= \langle value \rangle$.
Constraint: LDX $\geq$ N.

On entry, N $= \langle value \rangle$ and M $= \langle value \rangle$.
Constraint: N $\geq$ M.

IFAIL $= 20$

SVD failed to converge.

IFAIL $= 100$

An internal error has occurred in this routine. Check the routine call and any array sizes. If the call is correct then please contact NAG for assistance.

IFAIL $= -99$

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

See Section 3.9 in How to Use the NAG Library and its Documentation for further information.

IFAIL $= -399$

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

See Section 3.8 in How to Use the NAG Library and its Documentation for further information.

IFAIL $= -999$

Dynamic memory allocation failed.

See Section 3.7 in How to Use the NAG Library and its Documentation for further information.

7 Accuracy

The calculations are believed to be stable.

8 Parallelism and Performance

G03BDF is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

G03BDF makes calls to BLAS and/or LAPACK routines, which may be threaded within the vendor library used by this implementation. Consult the documentation for the vendor library for further information.

Please consult the X06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this routine. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

9 Further Comments

None.

10 Example

This example reads a loadings matrix and calculates a varimax transformation before calculating P, R and σ for a ProMax rotation.

10.1 Program Text

```
    Program g03bdfe

!     G03BDF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: g03baf, g03bdf, nag_wp, x04caf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: acc, g, power
      Integer                          :: i, ifail, iter, ldfp, ldfs, ldphi,   &
                                          ldr, ldro, ldx, lwk, m, maxit, n
      Character (1)                    :: stand
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: fp(:,:), fs(:,:), phi(:,:), r(:,:),  &
                                          ro(:,:), wk(:), x(:,:)
!     .. Executable Statements ..
      Write (nout,*) 'G03BDF Example Program Results'
      Write (nout,*)
      Flush (nout)

!     Skip heading in data file
      Read (nin,*)

!     Read in problem size
      Read (nin,*) n, m, g, stand, acc, maxit, power

      ldx = n
      ldro = m
      ldfp = n
      ldfs = n
      ldr = m
      ldphi = m
      lwk = 2*n + m*m + 5*(m-1)
      Allocate (fp(ldx,m),x(ldx,m),ro(ldro,m),wk(lwk),phi(ldphi,m),fs(ldfs,m), &
        r(ldr,m))

!     Read loadings matrix
      Read (nin,*)(fp(i,1:m),i=1,n)

!     Calculate orthogonal rotation
      ifail = 0
      Call g03baf(stand,g,n,m,fp,ldx,x,ro,ldro,acc,maxit,iter,wk,ifail)
```

```
!     Calculate ProMax rotation
      ifail = 0
      Call g03bdf(stand,n,m,x,ldx,ro,ldro,power,fp,ldfp,r,ldr,phi,ldphi,fs,    &
        ldfs,ifail)

!     Display results
      ifail = 0
      Call x04caf('General',' ',n,m,fp,ldfp,'Factor pattern',ifail)
      Write (nout,*)
      Flush (nout)
      ifail = 0
      Call x04caf('General',' ',m,m,r,ldr,'ProMax rotation',ifail)
      Write (nout,*)
      Flush (nout)
      ifail = 0
      Call x04caf('General',' ',m,m,phi,ldphi,'Inter-factor correlations',     &
        ifail)
      Write (nout,*)
      Flush (nout)
      ifail = 0
      Call x04caf('General',' ',n,m,fs,ldfs,'Factor structure',ifail)

    End Program g03bdfe
```

10.2 Program Data

```
G03BDF Example Program Data
5 2 1 'S' 1.0E-5 200 3.0 : N,M,G,STAND,ACC,MAXIT,POWER
0.74215 -0.57806
0.71370 -0.55515
0.87899 -0.15847
0.62533  0.76621
0.71447  0.67936         :: End of FP
```

10.3 Program Results

```
 G03BDF Example Program Results

 Factor pattern
             1          2
 1      0.9556    -0.0979
 2      0.9184    -0.0935
 3      0.7605     0.3393
 4     -0.0791     1.0019
 5      0.0480     0.9751

 ProMax rotation
          1       2
 1   0.7380  0.5420
 2  -0.7055  0.8653

 Inter-factor correlations
          1       2
 1   1.0000  0.2019
 2   0.2019  1.0000

 Factor structure
          1       2
 1   0.9358  0.0950
 2   0.8995  0.0919
 3   0.8290  0.4928
 4   0.1232  0.9860
 5   0.2448  0.9848
```