NAG Library Routine Document

G02AAF

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

G02AAF computes the nearest correlation matrix, in the Frobenius norm, to a given square, input matrix.

2 Specification

```
SUBROUTINE G02AAF (G, LDG, N, ERRTOL, MAXITS, MAXIT, X, LDX, ITER,          &
                   FEVAL, NRMGRD, IFAIL)

INTEGER            LDG, N, MAXITS, MAXIT, LDX, ITER, FEVAL, IFAIL
REAL (KIND=nag_wp) G(LDG,N), ERRTOL, X(LDX,N), NRMGRD
```

3 Description

A correlation matrix may be characterised as a real square matrix that is symmetric, has a unit diagonal and is positive semidefinite.

G02AAF applies an inexact Newton method to a dual formulation of the problem, as described by Qi and Sun (2006). It applies the improvements suggested by Borsdorf and Higham (2010).

4 References

Borsdorf R and Higham N J (2010) A preconditioned (Newton) algorithm for the nearest correlation matrix *IMA Journal of Numerical Analysis* **30(1)** 94–107

Qi H and Sun D (2006) A quadratically convergent Newton method for computing the nearest correlation matrix *SIAM J. Matrix AnalAppl* **29(2)** 360–385

5 Parameters

1: G(LDG, N) – REAL (KIND=nag_wp) array *Input/Output*

On entry: G, the initial matrix.

On exit: a symmetric matrix $\frac{1}{2}(G + G^{\mathrm{T}})$ with the diagonal set to I.

2: LDG – INTEGER *Input*

On entry: the first dimension of the array G as declared in the (sub)program from which G02AAF is called.

Constraint: LDG $\geq$ N.

3: N – INTEGER *Input*

On entry: the size of the matrix G.

Constraint: N > 0.

4: ERRTOL – REAL (KIND=nag_wp) *Input*

On entry: the termination tolerance for the Newton iteration. If ERRTOL ≤ 0.0 then N $\times \sqrt{\textbf{\textit{machine precision}}}$ is used.

5: MAXITS – INTEGER *Input*

On entry: MAXITS specifies the maximum number of iterations used for the iterative scheme used to solve the linear algebraic equations at each Newton step.

If MAXITS ≤ 0, $2 \times \mathrm{N}$ is used.

6: MAXIT – INTEGER *Input*

On entry: specifies the maximum number of Newton iterations.

If MAXIT ≤ 0, 200 is used.

7: X(LDX, N) – REAL (KIND=nag_wp) array *Output*

On exit: contains the nearest correlation matrix.

8: LDX – INTEGER *Input*

On entry: the first dimension of the array X as declared in the (sub)program from which G02AAF is called.

Constraint: LDX $\geq$ N.

9: ITER – INTEGER *Output*

On exit: the number of Newton steps taken.

10: FEVAL – INTEGER *Output*

On exit: the number of function evaluations of the dual problem.

11: NRMGRD – REAL (KIND=nag_wp) *Output*

On exit: the norm of the gradient of the last Newton step.

12: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction 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 parameter, 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, LDG $= \langle value \rangle$ and N $= \langle value \rangle$.
Constraint: LDG $\geq$ N.

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

On entry, N = ⟨*value*⟩.
Constraint: N > 0.

IFAIL = 2

Newton iteration fails to converge in ⟨*value*⟩ iterations.

IFAIL = 3

Machine precision is limiting convergence.
The array returned in X may still be of interest.

IFAIL = 4

An intermediate eigenproblem could not be solved. This should not occur. Please contact NAG with details of your call.

IFAIL = −99

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

See Section 3.8 in the Essential Introduction for further information.

IFAIL = −399

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

See Section 3.7 in the Essential Introduction for further information.

IFAIL = −999

Dynamic memory allocation failed.

See Section 3.6 in the Essential Introduction for further information.

7 Accuracy

The returned accuracy is controlled by ERRTOL and limited by ***machine precision***.

8 Parallelism and Performance

G02AAF is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

G02AAF 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

Arrays are internally allocated by G02AAF. The total size of these arrays is $11 \times \mathrm{N} + 3 \times \mathrm{N} \times \mathrm{N} + \max(2 \times \mathrm{N} \times \mathrm{N} + 6 \times \mathrm{N} + 1, 120 + 9 \times \mathrm{N})$ real elements and $5 \times \mathrm{N} + 3$ integer elements.

10 Example

This example finds the nearest correlation matrix to:

$$G = \begin{pmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{pmatrix}$$

10.1 Program Text

```
    Program g02aafe

!     G02AAF Example Program Text

!     Mark 25 Release. NAG Copyright 2014.

!     .. Use Statements ..
      Use nag_library, Only: g02aaf, nag_wp, x04caf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: errtol, nrmgrd
      Integer                          :: feval, i, ifail, iter, ldg, ldx,     &
                                          maxit, maxits, n
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: g(:,:), x(:,:)
!     .. Executable Statements ..
      Write (nout,*) 'G02AAF Example Program Results'
      Write (nout,*)
      Flush (nout)

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

!     Read in the problem size
      Read (nin,*) n

      ldg = n
      ldx = n
      Allocate (g(ldg,n),x(ldx,n))

!     Read in the matrix G
      Read (nin,*)(g(i,1:n),i=1,n)

!     Use the defaults for ERRTOL, MAXITS and MAXIT
      errtol = 0.0E0_nag_wp
      maxits = 0
      maxit = 0

!     Calculate nearest correlation matrix
      ifail = 0
      Call g02aaf(g,ldg,n,errtol,maxits,maxit,x,ldx,iter,feval,nrmgrd,ifail)

!     Display results
      ifail = 0
      Call x04caf('General',' ',n,n,x,ldx,'Nearest Correlation Matrix',ifail)
      Write (nout,*)
      Write (nout,99999) ' Number of Newton steps taken:', iter
      Write (nout,99998) ' Number of function evaluations:', feval

99999 Format (1X,A,I11)
99998 Format (1X,A,I9)
    End Program g02aafe
```

10.2 Program Data

```
G02AAF Example Program Data
 4                          :: N
 2.0   -1.0    0.0    0.0
-1.0    2.0   -1.0    0.0
 0.0   -1.0    2.0   -1.0
 0.0    0.0   -1.0    2.0  :: End of G
```

10.3 Program Results

```
 G02AAF Example Program Results

 Nearest Correlation Matrix
          1       2       3       4
 1   1.0000 -0.8084  0.1916  0.1068
 2  -0.8084  1.0000 -0.6562  0.1916
 3   0.1916 -0.6562  1.0000 -0.8084
 4   0.1068  0.1916 -0.8084  1.0000

  Number of Newton steps taken:          3
  Number of function evaluations:        4
```