NAG Library Routine Document

G13DXF

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

G13DXF calculates the zeros of a vector autoregressive (or moving average) operator. This routine is likely to be used in conjunction with G05PJF, G13ASF, G13DDF or G13DSF.

2 Specification

```
SUBROUTINE G13DXF (K, IP, PAR, RR, RI, RMOD, WORK, IWORK, IFAIL)
INTEGER             K, IP, IWORK(K*IP), IFAIL
REAL (KIND=nag_wp) PAR(IP*K*K), RR(K*IP), RI(K*IP), RMOD(K*IP),          &
                    WORK(K*K*IP*IP)
```

3 Description

Consider the vector autoregressive moving average (VARMA) model

$$W_t - \mu = \phi_1(W_{t-1} - \mu) + \phi_2(W_{t-2} - \mu) + \cdots + \phi_p\left(W_{t-p} - \mu\right) + \epsilon_t - \theta_1\epsilon_{t-1} - \theta_2\epsilon_{t-2} - \cdots - \theta_q\epsilon_{t-q}, \quad (1)$$

where W_t denotes a vector of k time series and ϵ_t is a vector of k residual series having zero mean and a constant variance-covariance matrix. The components of ϵ_t are also assumed to be uncorrelated at non-simultaneous lags. $\phi_1, \phi_2, \ldots, \phi_p$ denotes a sequence of k by k matrices of autoregressive (AR) parameters and $\theta_1, \theta_2, \ldots, \theta_q$ denotes a sequence of k by k matrices of moving average (MA) parameters. μ is a vector of length k containing the series means. Let

$$A(\phi) = \begin{bmatrix} \phi_1 & I & 0 & . & . & . & 0 \\ \phi_2 & 0 & I & 0 & . & . & 0 \\ . & & & . & & & \\ . & & & & . & & \\ . & & & & & . & \\ \phi_{p-1} & 0 & . & . & . & 0 & I \\ \phi_p & 0 & . & . & . & 0 & 0 \end{bmatrix}_{pk\times pk}$$

where I denotes the k by k identity matrix.

The model (1) is said to be stationary if the eigenvalues of $A(\phi)$ lie inside the unit circle. Similarly let

$$B(\theta) = \begin{bmatrix} \theta_1 & I & 0 & . & . & . & 0 \\ \theta_2 & 0 & I & 0 & . & . & 0 \\ . & & & . & & & \\ . & & & & . & & \\ . & & & & & . & \\ \theta_{q-1} & 0 & . & . & . & 0 & I \\ \theta_q & 0 & . & . & . & 0 & 0 \end{bmatrix}_{qk\times qk} .$$

Then the model is said to be invertible if the eigenvalues of $B(\theta)$ lie inside the unit circle.

G13DXF returns the pk eigenvalues of $A(\phi)$ (or the qk eigenvalues of $B(\theta)$) along with their moduli, in descending order of magnitude. Thus to check for stationarity or invertibility you should check whether the modulus of the largest eigenvalue is less than one.

4 References

Wei W W S (1990) *Time Series Analysis: Univariate and Multivariate Methods* Addison–Wesley

5 Arguments

1: K – INTEGER *Input*

On entry: k, the dimension of the multivariate time series.

Constraint: $K \geq 1$.

2: IP – INTEGER *Input*

On entry: the number of AR (or MA) parameter matrices, p (or q).

Constraint: $IP \geq 1$.

3: PAR($IP \times K \times K$) – REAL (KIND=nag_wp) array *Input*

On entry: the AR (or MA) parameter matrices read in row by row in the order $\phi_1, \phi_2, \ldots, \phi_p$ (or $\theta_1, \theta_2, \ldots, \theta_q$). That is, PAR$((l-1) \times k \times k + (i-1) \times k + j)$ must be set equal to the (i,j)th element of ϕ_l, for $l = 1, 2, \ldots, p$ (or the (i,j)th element of θ_l, for $l = 1, 2, \ldots, q$).

4: RR($K \times IP$) – REAL (KIND=nag_wp) array *Output*

On exit: the real parts of the eigenvalues.

5: RI($K \times IP$) – REAL (KIND=nag_wp) array *Output*

On exit: the imaginary parts of the eigenvalues.

6: RMOD($K \times IP$) – REAL (KIND=nag_wp) array *Output*

On exit: the moduli of the eigenvalues.

7: WORK($K \times K \times IP \times IP$) – REAL (KIND=nag_wp) array *Workspace*
8: IWORK($K \times IP$) – INTEGER array *Workspace*

9: 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, $K < 1$,
or $IP < 1$.

IFAIL = 2

An excessive number of iterations are needed to evaluate the eigenvalues of $A(\phi)$ (or $B(\theta)$). This is an unlikely exit. All output arguments are undefined.

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 accuracy of the results depends on the original matrix and the multiplicity of the roots.

8 Parallelism and Performance

G13DXF is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

G13DXF 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

The time taken is approximately proportional to kp^3 (or kq^3).

10 Example

This example finds the eigenvalues of $A(\phi)$ where $k = 2$ and $p = 1$ and $\phi_1 = \begin{bmatrix} 0.802 & 0.065 \\ 0.000 & 0.575 \end{bmatrix}$.

10.1 Program Text

```
    Program g13dxfe

!     G13DXF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: g13dxf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
```

```
      Integer                          :: i, ifail, ip, k, kip, npar
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: par(:), ri(:), rmod(:), rr(:),       &
                                          work(:)
      Integer, Allocatable             :: iwork(:)
!     .. Executable Statements ..
      Write (nout,*) 'G13DXF Example Program Results'
      Write (nout,*)

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

!     Read in problem size
      Read (nin,*) k, ip

      kip = k*ip
      npar = k*kip
      Allocate (par(npar),rr(kip),ri(kip),rmod(kip),work(ip*npar),iwork(kip))

!     Read the AR (or MA) parameters
      Read (nin,*) par(1:npar)

!     Calculate zeros
      ifail = 0
      Call g13dxf(k,ip,par,rr,ri,rmod,work,iwork,ifail)

!     Display results
      Write (nout,*) '        Eigenvalues       Moduli'
      Write (nout,*) '        -----------       ------'
      Do i = 1, k*ip
        If (ri(i)>=0.0E0_nag_wp) Then
          Write (nout,99999) rr(i), ri(i), rmod(i)
        Else
          Write (nout,99998) rr(i), -ri(i), rmod(i)
        End If
      End Do

99999 Format (' ',F10.3,'  + ',F6.3,' i  ',F8.3)
99998 Format (' ',F10.3,'  - ',F6.3,' i  ',F8.3)
    End Program g13dxfe
```

10.2 Program Data

```
G13DXF Example Program Data
 2 1                     :: K,IP
 0.802 0.065 0.000 0.575 :: PAR
```

10.3 Program Results

```
 G13DXF Example Program Results

         Eigenvalues       Moduli
         -----------       ------
      0.802  +  0.000 i     0.802
      0.575  +  0.000 i     0.575
```