G13DMF – NAG Fortran Library Routine Document

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

G13DNF calculates the sample partial lag correlation matrices of a multivariate time series. A set of χ^2 statistics and their sigificance levels are also returned. A call to G13DMF is usually made prior to calling this routine in order to calculate the sample cross-correlation matrices.

2 Specification

```
SUBROUTINE G13DMF(MATRIX, K, N, M, W, IK, WMEAN, R0, R, IFAIL)
INTEGER           K, N, M, IK, IFAIL
real              W(IK,N), WMEAN(K), R0(IK,K), R(IK,IK,M)
CHARACTER*1       MATRIX
```

3 Description

Let $W_t = (w_{1t}, w_{2t}, \ldots, w_{kt})^T$, for $t = 1, 2, \ldots, n$ denote n observations of a vector of k time series. The sample cross-covariance matrix at lag l is defined to be the k by k matrix $\hat{C}(l)$, whose (i, j)th element is given by

$$\hat{C}_{ij}(l) = \frac{1}{n} \sum_{t=l+1}^{n} (w_{i(t-l)} - \bar{w}_i)(w_{jt} - \bar{w}_j), \quad \text{for } l = 0, 1, 2, \ldots, m; \ \ i = 1, 2, \ldots, k \ ; \ j = 1, 2, \ldots, k,$$

where $\bar{w}_i$ and $\bar{w}_j$ denote the sample means for the ith and jth series respectively. The sample cross-correlation matrix at lag l is defined to be the k by k matrix $\hat{R}(l)$, whose (i, j)th element is given by

$$\hat{R}_{ij}(l) = \frac{\hat{C}_{ij}(l)}{\sqrt{\hat{C}_{ii}(0)\hat{C}_{jj}(0)}}, \quad \text{for } l = 0, 1, 2, \ldots, m; \ \ i = 1, 2, \ldots, k \ ; \ j = 1, 2, \ldots, k.$$

The number of lags, m, is usually taken to be at most $n/4$.

If W_t follows a vector moving average model of order q, then it can be shown that the theoretical cross-correlation matrices $(R(l))$ are zero beyond lag q. In order to help spot a possible cut-off point, the elements of $\hat{R}(l)$ are usually compared to their approximate standard error of $1/\sqrt{n}$. For further details see, for example, Wei [1].

The routine uses a single pass through the data to compute the means and the cross-covariance matrix at lag zero. The cross-covariance matrices at further lags are then computed on a second pass through the data.

4 References

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

[2] West D H D (1979) Updating mean and variance estimates: An improved method *Comm. ACM* **22** 532–535

5 Parameters

1: MATRIX — CHARACTER*1 *Input*

On entry: indicates whether the cross-covariance or cross-correlation matrices are to be computed;

If MATRIX = 'V', then the cross-covariance matrices are computed, If MATRIX = 'R', then the cross-correlation matrices are computed.

Constraint: MATRIX = 'V' or 'R'.

2: K — INTEGER *Input*

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

Constraint: K ≥ 1.

3: N — INTEGER *Input*

On entry: the number of observations in the series, n.

Constraint: N ≥ 2.

4: M — INTEGER *Input*

On entry: the number, m, of cross-correlation (or cross-covariance) matrices to be computed. If in doubt set M = 10. However it should be noted that M is usually taken to be at most N/4.

Constraint: $1 \leq$ M $<$ N.

5: W(IK,N) — ***real*** array *Input*

On entry: W(i,t) must contain the observation w_{it}, for $i = 1,2,\ldots,k$; $t = 1,2,\ldots,n$.

6: IK — INTEGER *Input*

On entry: the first dimension of the arrays W and R0 and the first and second dimensions of the array R as declared in the (sub)program from which G13DMF is called.

Constraint: IK $\geq$ K.

7: WMEAN(K) — ***real*** array *Output*

On exit: the means, $\bar{w}_i$, for $i = 1,2,\ldots,k$.

8: R0(IK,K) — ***real*** array *Output*

On exit: if $i \neq j$, then R0(i,j) contains an estimate of the (i,j)th element of the cross-correlation (or cross-covariance) matrix at lag zero, $\hat{R}_{ij}(0)$; if $i = j$, then if MATRIX = 'V', R0(i,i) contains the variance of the ith series, $\hat{C}_{ii}(0)$, and if MATRIX = 'R', R0(i,i) contains the standard deviation of the ith series, $\sqrt{\hat{C}_{ii}(0)}$.

If IFAIL = 2 and MATRIX = 'R', then on exit all the elements in R0 whose computation involves the zero variance are set to zero.

9: R(IK,IK,M) — ***real*** array *Output*

On exit: R(i,j,l) contains an estimate of the (i,j)th element of the cross-correlation (or cross-covariance) at lag l, $\hat{R}_{ij}(l)$, for $l = 1,2,\ldots,m$; $i = 1,2,\ldots,k$; $j = 1,2,\ldots,k$.

If IFAIL = 2 and MATRIX = 'R', then on exit all the elements in R whose computation involves the zero variance are set to zero.

10: IFAIL — INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.

On exit: IFAIL = 0 unless the routine detects an error (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 detected by the routine:

IFAIL = 1

On entry, MATRIX ≠ 'V' or 'R',
or K < 1,
or N < 2,
or M < 1,
or M ≥ N,
or IK < K.

IFAIL = 2

On entry, at least one of the k series is such that all its elements are practically equal giving zero (or near zero) variance. In this case if MATRIX = 'R' all the correlations in R0 and R involving this variance are set to zero.

7 Accuracy

For a discussion of the accuracy of the one-pass algorithm used to compute the sample cross-covariances at lag zero see West [2]. For the other lags a two-pass algorithm is used to compute the cross-covariances; the accuracy of this algorithm is also discussed in [2]. The accuracy of the cross-correlations will depend on the accuracy of the computed cross-covariances.

8 Further Comments

The time taken is roughly proportional to mnk^2.

9 Example

This program computes the sample cross-correlation matrices of two time series of length 48, up to lag 10. It also prints the cross-correlation matrices together with plots of symbols indicating which elements of the correlation matrices are significant. Three *'s represent significance at the 0.5% level, two *'s represent significance at the 1% level and a single * represents significance at the 5% level. The *'s are plotted above or below the line depending on whether the elements are significant in the positive or negative direction.

9.1 Program Text

Note. The listing of the example program presented below uses bold italicised terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     G13DMF Example Program Text
*     Mark 15 Release. NAG Copyright 1991.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          IK, NMAX, MMAX
      PARAMETER        (IK=3,NMAX=100,MMAX=20)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, J, K, M, N
*     .. Local Arrays ..
```

```
      real              R(IK,IK,MMAX), R0(IK,IK), W(IK,NMAX), WMEAN(IK)
*     .. External Subroutines ..
      EXTERNAL          CPRINT, G13DMF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G13DMF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) K, N, M
      IF (K.GT.0 .AND. K.LE.IK .AND. N.GE.1 .AND. N.LE.NMAX .AND. M.GE.
     +    1 .AND. M.LE.MMAX) THEN
         DO 20 I = 1, K
            READ (NIN,*) (W(I,J),J=1,N)
   20    CONTINUE
         IFAIL = 0
*
         CALL G13DMF('R',K,N,M,W,IK,WMEAN,R0,R,IFAIL)
*
         CALL CPRINT(K,N,IK,M,WMEAN,R0,R,NOUT)
      END IF
      STOP
*
      END
*
      SUBROUTINE CPRINT(K,N,IK,M,WMEAN,R0,R,NOUT)
*     .. Scalar Arguments ..
      INTEGER           IK, K, M, N, NOUT
*     .. Array Arguments ..
      real              R(IK,IK,M), R0(IK,K), WMEAN(K)
*     .. Local Scalars ..
      real              C1, C2, C3, C5, C6, C7, CONST, SUM
      INTEGER           I, I2, IFAIL2, J, L, LL
*     .. Local Arrays ..
      CHARACTER*1       CLABS(1), RLABS(1)
      CHARACTER*80      REC(7)
*     .. External Subroutines ..
      EXTERNAL          X04CBF
*     .. Intrinsic Functions ..
      INTRINSIC         real, SQRT
*     .. Executable Statements ..
*
*     Print the correlation matrices and indicator symbols.
*
      CONST = 1.0e0/SQRT(real(N))
      WRITE (NOUT,*)
      WRITE (NOUT,*) ' THE MEANS'
      WRITE (NOUT,*) ' ---------'
      WRITE (NOUT,99999) (WMEAN(I),I=1,K)
      WRITE (NOUT,*)
      WRITE (NOUT,*) ' CROSS-CORRELATION MATRICES'
      WRITE (NOUT,*) ' --------------------------'
      DO 20 L = 1, M
         WRITE (NOUT,99998) ' Lag = ', L
         IFAIL2 = 0
         CALL X04CBF('G','N',K,K,R(1,1,L),IK,'F9.3',' ','N',RLABS,'N',
     +               CLABS,80,5,IFAIL2)
   20 CONTINUE
*
*     Print indicator symbols to indicate significant elements.
```

```
*
      WRITE (NOUT,99997) ' Standard error = 1 / SQRT(N) = ', CONST
      WRITE (NOUT,*)
      WRITE (NOUT,*) ' TABLES OF INDICATOR SYMBOLS'
      WRITE (NOUT,*) ' ---------------------------'
      WRITE (NOUT,99998) ' For Lags 1 to ', M
*
*     Set up annotation for the plots.
*
      WRITE (REC(1),99996) '              0.005  :'
      WRITE (REC(2),99996) '        +     0.01   :'
      WRITE (REC(3),99996) '              0.05   :'
      WRITE (REC(4)(1:23),99996) '   Sig. Level        :'
      WRITE (REC(4)(24:),99996) '- - - - - - - - - -  Lags'
      WRITE (REC(5),99996) '              0.05   :'
      WRITE (REC(6),99996) '        -     0.01   :'
      WRITE (REC(7),99996) '              0.005  :'
*
*     Set up the critical values
*
      C1 = 3.29e0*CONST
      C2 = 2.58e0*CONST
      C3 = 1.96e0*CONST
      C5 = -C3
      C6 = -C2
      C7 = -C1
*
      DO 120 I = 1, K
         DO 100 J = 1, K
            WRITE (NOUT,*)
            IF (I.EQ.J) THEN
               WRITE (NOUT,99995) ' Auto-correlation function for',
     +           ' series ', I
            ELSE
               WRITE (NOUT,99994) ' Cross-correlation function for',
     +           ' series ', I, ' and series', J
            END IF
            DO 60 L = 1, M
               LL = 23 + 2*L
               SUM = R(I,J,L)
*
*              Clear the last plot with blanks
*
               DO 40 I2 = 1, 7
                  IF (I2.NE.4) REC(I2) (LL:LL) = ' '
   40          CONTINUE
*
*              Check for significance
*
               IF (SUM.GT.C1) REC(1) (LL:LL) = '*'
               IF (SUM.GT.C2) REC(2) (LL:LL) = '*'
               IF (SUM.GT.C3) REC(3) (LL:LL) = '*'
               IF (SUM.LT.C5) REC(5) (LL:LL) = '*'
               IF (SUM.LT.C6) REC(6) (LL:LL) = '*'
               IF (SUM.LT.C7) REC(7) (LL:LL) = '*'
   60       CONTINUE
*
*           Print
```

```
*
            DO 80 I2 = 1, 7
               WRITE (NOUT,99996) REC(I2)
   80       CONTINUE
  100    CONTINUE
  120 CONTINUE
      RETURN
*
99999 FORMAT (/1X,2(2X,F9.3))
99998 FORMAT (/1X,A,I2)
99997 FORMAT (/1X,A,F5.3,A)
99996 FORMAT (1X,A)
99995 FORMAT (//1X,A,A,I2,/)
99994 FORMAT (//1X,A,A,I2,A,I2,/)
      END
```

9.2 Program Data

```
G13DMF Example Program Data
2 48 10 :  K, no. of series,  N, no. of obs in each series,  M, no. of lags
-1.490 -1.620  5.200  6.230  6.210  5.860  4.090  3.180
 2.620  1.490  1.170  0.850 -0.350  0.240  2.440  2.580
 2.040  0.400  2.260  3.340  5.090  5.000  4.780  4.110
 3.450  1.650  1.290  4.090  6.320  7.500  3.890  1.580
 5.210  5.250  4.930  7.380  5.870  5.810  9.680  9.070
 7.290  7.840  7.550  7.320  7.970  7.760  7.000  8.350
 7.340  6.350  6.960  8.540  6.620  4.970  4.550  4.810
 4.750  4.760 10.880 10.010 11.620 10.360  6.400  6.240
 7.930  4.040  3.730  5.600  5.350  6.810  8.270  7.680
 6.650  6.080 10.250  9.140 17.750 13.300  9.630  6.800
 4.080  5.060  4.940  6.650  7.940 10.760 11.890  5.850
 9.010  7.500 10.020 10.380  8.150  8.370 10.730 12.140 : End of time series
```

9.3 Program Results

```
 G13DMF Example Program Results

  THE MEANS
  ---------

       4.370      7.868

  CROSS-CORRELATION MATRICES
  --------------------------

  Lag =  1
           0.736    0.174
           0.211    0.555

  Lag =  2
           0.456    0.076
           0.069    0.260

  Lag =  3
           0.379    0.014
           0.026   -0.038
```

```
Lag =  4
         0.322    0.110
         0.093   -0.236

Lag =  5
         0.341    0.269
         0.087   -0.250

Lag =  6
         0.363    0.344
         0.132   -0.227

Lag =  7
         0.280    0.425
         0.207   -0.128

Lag =  8
         0.248    0.522
         0.197   -0.085

Lag =  9
         0.240    0.266
         0.254    0.075

Lag = 10
         0.162   -0.020
         0.267    0.005

Standard error = 1 / SQRT(N) = 0.144

TABLES OF INDICATOR SYMBOLS
---------------------------

For Lags 1 to 10



Auto-correlation function for series  1

              0.005  : *
        +     0.01   : * * *
              0.05   : * * * * * *
   Sig. Level        : - - - - - - - - - -  Lags
              0.05   :
        -     0.01   :
              0.005  :



Cross-correlation function for series  1 and series 2

              0.005  :              *
        +     0.01   :            * *
              0.05   :          * * *
   Sig. Level        : - - - - - - - - - -  Lags
              0.05   :
        -     0.01   :
              0.005  :
```

```
Cross-correlation function for series  2 and series 1

              0.005  :
        +     0.01   :
              0.05   :
   Sig. Level        : - - - - - - - - - - -  Lags
              0.05   :
        -     0.01   :
              0.005  :



Auto-correlation function for series  2

              0.005  : *
        +     0.01   : *
              0.05   : *
   Sig. Level        : - - - - - - - - - - -  Lags
              0.05   :
        -     0.01   :
              0.005  :
```