NAG Fortran Library Routine Document

G13DLF

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

G13DLF differences and/or transforms a multivariate time series. It is intended to be used prior to G13DCF to fit a vector autoregressive moving average (VARMA) model to the differenced/transformed series.

2 Specification

```
SUBROUTINE G13DLF(K, N, Z, IK, TR, ID, DELTA, W, ND, WORK, IFAIL)
INTEGER           K, N, IK, ID(K), ND, IFAIL
real              Z(IK,N), DELTA(IK,*), W(IK,*), WORK(K*N)
CHARACTER*1       TR(K)
```

3 Description

For certain time series it may first be necessary to difference the original data to obtain a stationary series before fitting a VARMA model. This routine also allows the user to apply either a square root or a log transformation to the original time series to stabilize the variance if required.

If the order of differencing required for the ith series is d_i, then the differencing operator is defined by $\delta_i(B) = 1 - \delta_{i1}B - \delta_{i2}B^2 - \cdots - \delta_{id_i}B^{d_i}$, where B is the backward shift operator; that is, $BZ_t = Z_{t-1}$. Let d denote the maximum of the orders of differencing, d_i, over the k series. The routine computes values of the differenced/transformed series $W_t = (w_{1t}, w_{2t}, \ldots, w_{kt})^T$ for $t = d+1, d+2, \ldots, n$ as follows:

$$w_{it} = \delta_i(B)z^*_{it}, \quad i = 1, 2, \ldots, k$$

where z^*_{it} are the transformed values of the original k-dimensional time series $Z_t = (z_{1t}, z_{2t}, \ldots, z_{kt})^T$.

The differencing parameters δ_{ij}, for $i = 1, 2, \ldots, k$; $j = 1, 2, \ldots, d_i$ must be supplied by the user. If the ith series does not require differencing, then $d_i = 0$.

4 References

Box G E P and Jenkins G M (1976) *Time Series Analysis: Forecasting and Control* (Revised Edition) Holden-Day

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

5 Parameters

1: K – INTEGER *Input*

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

Constraint: K ≥ 1.

2: N – INTEGER *Input*

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

Constraint: N ≥ 1.

3: Z(IK,N) – ***real*** array *Input*

On entry: Z(i,t) must contain, z_{it}, the ith component of Z_t, for $i = 1,2,\ldots,k$; $t = 1,2,\ldots,n$.

Constraints:

if TR(i) = 'L', then Z$(i,t) > 0.0$ and
if TR(i) = 'S', then Z$(i,t) \geq 0.0$, for $i = 1,2,\ldots,k$; $t = 1,2,\ldots,n$.

4: IK – INTEGER *Input*

On entry: the first dimension of the arrays Z, DELTA and W as declared in the (sub)program from which G13DLF is called.

Constraint: IK $\geq$ K.

5: TR(K) – CHARACTER*1 array *Input*

On entry: TR(i) indicates whether the ith time series is to be transformed, for $i = 1,2,\ldots,k$.

If TR(i) = 'N', then no transformation is used;

If TR(i) = 'L', then a log transformation is used;

If TR(i) = 'S', then a square root transformation is used.

Constraint: TR(i) must equal either 'N', 'L' or 'S', for $i = 1,2,\ldots,k$.

6: ID(K) – INTEGER array *Input*

On entry: the order of differencing for each series, $d_1, d_2, \ldots, d_k$.

Constraint: $0 \leq$ ID$(i) <$ N, for $i = 1,2,\ldots,k$.

7: DELTA(IK,*) – ***real*** array *Input*

Note: the second dimension of the array DELTA must be at least $\max(1,d)$ where $d = \max(\text{ID}(i))$.

On entry: if ID$(i) > 0$, then DELTA(i,j) must be set equal to δ_{ij}, for $j = 1,2,\ldots,d_i$; $i = 1,2,\ldots,k$. If $d = 0$, then DELTA is not referenced.

8: W(IK,*) – ***real*** array *Output*

Note: the second dimension of the array W must be at least N $- d$.

On exit: W(i,t) contains the value of $w_{i,t+d}$, for $i = 1,2,\ldots,k$; $t = 1,2,\ldots,n-d$.

9: ND – INTEGER *Output*

On exit: the number of differenced values, $n - d$, in the series.

10: WORK(K$*$N) – ***real*** array *Workspace*

11: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details.

On exit: IFAIL $= 0$ unless the routine detects an error (see Section 6).

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, for users 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.**

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 N < 1,
or IK $<$ K.

IFAIL $= 2$

On entry, ID$(i) < 0$, for some $i = 1, 2, \ldots, k$,
or ID$(i) \geq$ N, for some $i = 1, 2, \ldots, k$.

IFAIL $= 3$

On entry, at least one of the first k elements of TR is not equal to 'N', 'L' or 'S'.

IFAIL $= 4$

On entry, one or more of the elements of Z is invalid, for the transformation requested; that is, the user may be trying to log or square root a series, some of whose values are negative.

7 Accuracy

The computations are believed to be stable.

8 Further Comments

The same differencing operator does not have to be applied to all the series. For example, suppose we have $k = 2$, and wish to apply the second-order differencing operator ∇^2 to the first series and the first-order differencing operator ∇ to the second series:

$$w_{1t} = \nabla^2 z_{1t} = (1 - B)^2 z_{1t} = (1 - 2B + B^2) z_{1t}, \quad \text{and}$$

$$w_{2t} = \nabla z_{2t} = (1 - B) z_{2t}.$$

Then $d_1 = 2, d_2 = 1$, $d = \max(d_1, d_2) = 2$, and

$$\text{DELTA} = \begin{bmatrix} \delta_{11} & \delta_{12} \\ \delta_{21} & \end{bmatrix} = \begin{bmatrix} 2 & -1 \\ 1 & \end{bmatrix}.$$

9 Example

A program to difference (nonseasonally) each of two time series of length 48. No transformation is to be applied to either of the series.

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.

```
*     G13DLF Example Program Text
*     Mark 15 Release. NAG Copyright 1991.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          KMAX, NMAX, IK, IDMAX
      PARAMETER        (KMAX=3,NMAX=100,IK=KMAX,IDMAX=2)
```

```
*     .. Local Scalars ..
      INTEGER          I, IFAIL, J, K, MAXD, MIND, N, ND
*     .. Local Arrays ..
      real             DELTA(IK,IDMAX), W(IK,NMAX), WORK(KMAX*NMAX),
     +                 Z(IK,NMAX)
      INTEGER          ID(KMAX)
      CHARACTER        TR(KMAX)
*     .. External Subroutines ..
      EXTERNAL         G13DLF
*     .. Intrinsic Functions ..
      INTRINSIC        MAX, MIN
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G13DLF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) K, N
      IF (K.GT.0 .AND. K.LE.KMAX .AND. N.GT.0 .AND. N.LE.NMAX) THEN
         READ (NIN,*) (ID(I),I=1,K)
         MIND = 0
         MAXD = 0
         DO 20 I = 1, K
            MIND = MIN(MIND,ID(I))
            MAXD = MAX(MAXD,ID(I))
   20    CONTINUE
         IF (MIND.GE.0 .AND. MAXD.LE.IDMAX) THEN
            DO 40 I = 1, K
               READ (NIN,*) (Z(I,J),J=1,N)
   40       CONTINUE
            READ (NIN,*) (TR(I),I=1,K)
            IF (MAXD.GT.0) THEN
               DO 60 I = 1, K
                  READ (NIN,*) (DELTA(I,J),J=1,ID(I))
   60          CONTINUE
            END IF
            IFAIL = 0
*
            CALL G13DLF(K,N,Z,IK,TR,ID,DELTA,W,ND,WORK,IFAIL)
*
            WRITE (NOUT,*)
            WRITE (NOUT,*) ' Transformed/Differenced series'
            WRITE (NOUT,*) ' -------------------------------'
            DO 80 I = 1, K
               WRITE (NOUT,*)
               WRITE (NOUT,99999) ' Series ', I
               WRITE (NOUT,*) ' -----------'
               WRITE (NOUT,*)
               WRITE (NOUT,99998) ' Number of differenced values = ', ND
               WRITE (NOUT,*)
               WRITE (NOUT,99997) (W(I,J),J=1,ND)
   80       CONTINUE
*
         END IF
      END IF
      STOP
*
99999 FORMAT (1X,A,I2)
99998 FORMAT (1X,A,I6)
99997 FORMAT (1X,8F9.3)
      END
```

9.2 Program Data

```
G13DLF Example Program Data
2 48 1  :  K, N
 1  1   :  ID(1), ID(2)
-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
 'N'  'N'     :  TR(1), TR(2)
 1.0   :  delta(1,1)
 1.0   :  delta(2,1)
```

9.3 Program Results

```
 G13DLF Example Program Results

  Transformed/Differenced series
  ------------------------------

  Series  1
  ----------

  Number of differenced values =     47

    -0.130    6.820    1.030   -0.020   -0.350   -1.770   -0.910   -0.560
    -1.130   -0.320   -0.320   -1.200    0.590    2.200    0.140   -0.540
    -1.640    1.860    1.080    1.750   -0.090   -0.220   -0.670   -0.660
    -1.800   -0.360    2.800    2.230    1.180   -3.610   -2.310    3.630
     0.040   -0.320    2.450   -1.510   -0.060    3.870   -0.610   -1.780
     0.550   -0.290   -0.230    0.650   -0.210   -0.760    1.350

  Series  2
  ----------

  Number of differenced values =     47

    -0.990    0.610    1.580   -1.920   -1.650   -0.420    0.260   -0.060
     0.010    6.120   -0.870    1.610   -1.260   -3.960   -0.160    1.690
    -3.890   -0.310    1.870   -0.250    1.460    1.460   -0.590   -1.030
    -0.570    4.170   -1.110    8.610   -4.450   -3.670   -2.830   -2.720
     0.980   -0.120    1.710    1.290    2.820    1.130   -6.040    3.160
    -1.510    2.520    0.360   -2.230    0.220    2.360    1.410
```