

NAG 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 G13DDF to fit a vector autoregressive moving average (VARMA) model to the differenced/transformed series.

2 Specification

```
SUBROUTINE G13DLF (K, N, Z, KMAX, TR, ID, DELTA, W, ND, WORK, IFAIL)
  INTEGER          K, N, KMAX, ID(K), ND, IFAIL
  REAL (KIND=nag_wp) Z(KMAX,N), DELTA(KMAX,*), W(KMAX,*), 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 calculating autocorrelations, etc. This routine also allows you 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 i th series is d_i , then the differencing operator is defined by $\delta_i(B) = 1 - \delta_{i1}B - \delta_{i2}B^2 - \dots - \delta_{id_i}B^{d_i}$, 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 $W_t = (w_{1t}, w_{2t}, \dots, w_{kt})^T$, for $t = d + 1, \dots, n$, as follows:

$$w_{it} = \delta_i(B)z_{it}^*, \quad i = 1, 2, \dots, k$$

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

The differencing parameters δ_{ij} , for $i = 1, 2, \dots, k$ and $j = 1, 2, \dots, d_i$, must be supplied by you. If the i th 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 Arguments

- 1: K – INTEGER *Input*
On entry: k , the dimension of the multivariate time series.
Constraint: $K \geq 1$.
- 2: N – INTEGER *Input*
On entry: n , the number of observations in the series, prior to differencing.
Constraint: $N \geq 1$.

- 3: Z(KMAX,N) – REAL (KIND=nag_wp) array Input
On entry: Z(*i*, *t*) must contain, z_{it} , the *i*th component of Z_t , for $i = 1, 2, \dots, k$ and $t = 1, 2, \dots, n$.
Constraints:
 if TR(*i*) = 'L', $Z(i, t) > 0.0$;
 if TR(*i*) = 'S', $Z(i, t) \geq 0.0$, for $i = 1, 2, \dots, k$ and $t = 1, 2, \dots, n$.
- 4: KMAX – 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: $KMAX \geq K$.
- 5: TR(K) – CHARACTER(1) array Input
On entry: TR(*i*) indicates whether the *i*th time series is to be transformed, for $i = 1, 2, \dots, k$.
 TR(*i*) = 'N'
 No transformation is used.
 TR(*i*) = 'L'
 A log transformation is used.
 TR(*i*) = 'S'
 A square root transformation is used.
Constraint: TR(*i*) = 'N', 'L' or 'S', for $i = 1, 2, \dots, k$.
- 6: ID(K) – INTEGER array Input
On entry: the order of differencing for each series, d_1, d_2, \dots, d_k .
Constraint: $0 \leq ID(i) < N$, for $i = 1, 2, \dots, K$.
- 7: DELTA(KMAX,*) – REAL (KIND=nag_wp) array Input
Note: the second dimension of the array DELTA must be at least $\max(1, d)$, where $d = \max(ID(i))$.
On entry: if $ID(i) > 0$, then DELTA(*i*, *j*) must be set equal to δ_{ij} , for $j = 1, 2, \dots, d_i$ and $i = 1, 2, \dots, k$.
 If $d = 0$, then DELTA is not referenced.
- 8: W(KMAX,*) – REAL (KIND=nag_wp) array Output
Note: the second dimension of the array W must be at least $N - d$, where $d = \max(ID(i))$.
On exit: W(*i*, *t*) contains the value of $w_{i,t+d}$, for $i = 1, 2, \dots, k$ and $t = 1, 2, \dots, n - d$.
- 9: ND – INTEGER Output
On exit: the number of differenced values, $n - d$, in the series, where $d = \max(ID(i))$.
- 10: WORK(K × N) – REAL (KIND=nag_wp) array Workspace
- 11: 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 $N < 1$,
or $KMAX < K$.

IFAIL = 2

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

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, you may be trying to log or square root a series, some of whose values are negative.

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 computations are believed to be stable.

8 Parallelism and Performance

G13DLF 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 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

$$\text{DELTA} = \begin{bmatrix} \delta_{11} & \delta_{12} \\ \delta_{21} & \delta_{22} \end{bmatrix} = \begin{bmatrix} 2 & -1 \\ 1 & 0 \end{bmatrix}.$$

10 Example

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

10.1 Program Text

```

Program g13dlfe

!      G13DLF Example Program Text

!      Mark 26 Release. NAG Copyright 2016.

!      .. Use Statements ..
Use nag_library, Only: g13dlf, nag_wp
!      .. Implicit None Statement ..
Implicit None
!      .. Parameters ..
Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..
Integer                    :: d, i, ifail, k, kmax, n, nd, tddelta
!      .. Local Arrays ..
Real (Kind=nag_wp), Allocatable :: delta(:,,:), w(:,,:), work(:,), z(:,,:)
Integer, Allocatable       :: id(:)
Character (1), Allocatable :: tr(:)
!      .. Intrinsic Procedures ..
Intrinsic                  :: max, maxval

!      .. Executable Statements ..
Write (nout,*) 'G13DLF Example Program Results'
Write (nout,*)

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

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

Allocate (id(k))

!      Read in differencing
Read (nin,*) id(1:k)

d = maxval(id(1:k))
tddelta = max(d,1)
nd = n - d
kmax = k
Allocate (z(kmax,n),tr(k),delta(kmax,tddelta),w(kmax,nd),work(k*n))

!      Read in series and the transformation flag
Read (nin,*)(z(i,1:n),i=1,k)
Read (nin,*) tr(1:k)

```

```

!      If required, read in delta
      If (d>0) Then
        Read (nin,*)(delta(i,1:id(i)),i=1,k)
      End If

!      Difference and / or transform series
      ifail = 0
      Call g13dlf(k,n,z,kmax,tr,id,delta,w,nd,work,ifail)

!      Display results
      Write (nout,*) ' Transformed/Differenced series'
      Write (nout,*) ' -----'
      Do 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,1:nd)
      End Do

99999 Format (1X,A,I2)
99998 Format (1X,A,I6)
99997 Format (1X,8F9.3)
      End Program g13dlfe

```

10.2 Program Data

G13DLF Example Program Data

```

2 48 1
1 1
-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
'N' 'N'
1.0
1.0

```

:: K,N
:: ID
:: End of Z
:: TR
:: End of DELTA

10.3 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	
