NAG Fortran Library Routine Document

G13DKF

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

G13DKF accepts a sequence of new observations in a multivariate time series and updates both the forecasts and the standard deviations of the forecast errors. A call to G13DJF must be made prior to calling this routine in order to calculate the elements of a reference vector together with a set of forecasts and their standard errors. On a successful exit from G13DKF the reference vector is updated so that should future series values become available these forecasts may be updated by recalling G13DKF.

2 Specification

```
 SUBROUTINE G13DKF(K, LMAX, M, MLAST, Z, IK, REF, LREF, V, PREDZ, SEFZ,
1                  WORK, IFAIL)
 INTEGER           K, LMAX, M, MLAST, IK, LREF, IFAIL
 real              Z(IK,M), REF(LREF), V(IK,M), PREDZ(IK,LMAX),
1                  SEFZ(IK,LMAX), WORK(K*M)
```

3 Description

Let $Z_t = (z_{1t}, z_{2t}, \ldots, z_{kt})^T$, for $t = 1, 2, \ldots, n$, denote a k-dimensional time series for which forecasts of $\hat{Z}_{n+1}, \hat{Z}_{n+2}, \ldots, \hat{Z}_{n+l_{\max}}$ have been computed using G13DJF. Given m further observations $Z_{n+1}, Z_{n+2}, \ldots, Z_{n+m}$, where $m < l_{\max}$, the routine updates the forecasts of $Z_{n+m+1}, Z_{n+m+2}, \ldots, Z_{n+l_{\max}}$ and their corresponding standard errors.

The routine uses a multivariate version of the procedure described in Box and Jenkins (1976). The forecasts are updated using the ψ weights, computed in G13DJF. If Z_t^* denotes the transformed value of Z_t and $\hat{Z}_t^*(l)$ denotes the forecast of Z_{t+l}^* from time t with a lead of l (that is the forecast of Z_{t+l}^* given observations $Z_t^*, Z_{t-1}^*, \ldots$), then

$$\hat{Z}_{t+1}^*(l) = \tau + \psi_l \epsilon_{t+1} + \psi_{l+1}\epsilon_t + \psi_{l+2}\epsilon_{t-1} + \cdots$$

and

$$\hat{Z}_t^*(l+1) = \tau + \psi_{l+1}\epsilon_t + \psi_{l+2}\epsilon_{t-1} + \cdots$$

where τ is a constant vector of length k involving the differencing parameters and the mean vector μ. By subtraction we obtain

$$\hat{Z}_{t+1}^*(l) = \hat{Z}_t^*(l+1) + \psi_l \epsilon_{t+1}.$$

Estimates of the residuals corresponding to the new observations are also computed as $\epsilon_{n+l} = Z_{n+l}^* - \hat{Z}_n^*(l)$, for $l = 1, 2, \ldots, m$. These may be of use in checking that the new observations conform to the previously fitted model.

On a successful exit, the reference array is updated so that G13DKF may be called again should future series values become available, see Section 8.

When a transformation has been used the forecasts and their standard errors are suitably modified to give results in terms of the original series Z_t; see Granger and Newbold (1976).

4 References

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

Granger C W J and Newbold P (1976) Forecasting transformed series *J. Roy. Statist. Soc. Ser. B* **38** 189–203

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

5 Parameters

The quantities K, LMAX, IK, REF and LREF output from G13DJF are suitable for input to G13DKF.

1: K – INTEGER *Input*

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

Constraint: K ≥ 1.

2: LMAX – INTEGER *Input*

On entry: the number, $l_{\max}$, of forecasts requested in the call to G13DJF.

Constraint: LMAX ≥ 2.

3: M – INTEGER *Input*

On entry: the number, m, of new observations available since the last call to either G13DJF or G13DKF. The number of new observations since the last call to G13DJF is then M + MLAST.

Constraint: $0 <$ M $<$ LMAX − MLAST.

4: MLAST – INTEGER *Input/Output*

On entry: on the first call to G13DKF, since calling G13DJF, MLAST must be set to 0 to indicate that no new observations have yet been used to update the forecasts; on subsequent calls MLAST must contain the value of MLAST as output on the previous call to G13DKF.

On exit: MLAST is incremented by m to indicate that MLAST + M observations have now been used to update the forecasts since the last call to G13DJF.

MLAST must not be changed between calls to G13DKF, unless a call to G13DJF has been made between the calls in which case MLAST should be reset to 0.

Constraint: $0 \leq$ MLAST $<$ LMAX − M.

5: Z(IK,M) – ***real*** array *Input*

On entry: Z(i,j) must contain the value of $z_{i,n+\text{MLAST}+j}$, for $i = 1,2,\ldots,k$; $j = 1,2,\ldots,m$ where n is the number of observations in the time series in the last call made to G13DJF.

Constraint: if the transformation defined in TR in G13DJF for the ith series is the log transformation, then Z$(i,j) > 0.0$, and if it is the square-root transformation, then Z$(i,j) \geq 0.0$, for $j = 1,2,\ldots,m$; $i = 1,2,\ldots,k$.

6: IK – INTEGER *Input*

On entry: the first dimension of the arrays Z, PREDZ and SEFZ as declared in the (sub)program from which G13DKF is called.

Constraint: IK $\geq$ K.

7: REF(LREF) – ***real*** array *Input/Output*

On entry: REF must contain the first (LMAX − 1) × K × K + 2 × K × LMAX + K elements of the reference vector as returned on a successful exit from G13DJF (or a previous call to G13DKF).

On exit: the elements of REF are updated. The first (LMAX − 1) × K × K elements store the ψ weights $\psi_1, \psi_2, \ldots, \psi_{l_{\max}-1}$, stored column-wise. The next K × LMAX elements contain the forecasts of the transformed series and the next K × LMAX elements contain the variances of the forecasts of the transformed variables; see G13DJF. The last K elements are not updated.

8: LREF – INTEGER *Input*

On entry: the dimension of the array REF as declared in the (sub)program from which G13DKF is called.

Constraint: LREF $\geq$ (LMAX $-$ 1) $\times$ K $\times$ K $+$ 2 $\times$ K $\times$ LMAX $+$ K.

9: V(IK,M) – ***real*** array *Output*

On exit: V(i,j) contains an estimate of the ith component of $\epsilon_{n+\text{MLAST}+j}$, for $i = 1,2,\ldots,k$; $j = 1,2,\ldots,m$.

10: PREDZ(IK,LMAX) – ***real*** array *Output*

On exit: PREDZ(i,j) contains the updated forecast of $z_{i,n+j}$, for $i = 1,2,\ldots,k$; $j = \text{MLAST} + \text{M} + 1, \text{MLAST} + \text{M} + 2, \ldots, l_{\max}$.

The columns of PREDZ corresponding to the new observations since the last call to either G13DJF or G13DKF are set equal to the corresponding columns of Z.

11: SEFZ(IK,LMAX) – ***real*** array *Output*

On exit: SEFZ(i,j) contains an estimate of the standard error of the corresponding element of PREDZ, for $i = 1,2,\ldots,k$; $j = \text{MLAST} + \text{M} + 1, \text{MLAST} + \text{M} + 2, \ldots, l_{\max}$.

The columns of SEFZ corresponding to the new observations since the last call to either G13DJF or G13DKF are set equal to zero.

12: WORK(K*M) – ***real*** array *Workspace*

13: 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 LMAX $<$ 2,
or M $\leq$ 0,
or MLAST $+$ M $\geq$ LMAX,
or MLAST $<$ 0,
or IK $<$ K,
or LREF $<$ (LMAX $-$ 1) $\times$ K $\times$ K $+$ 2 $\times$ K $\times$ LMAX $+$ K.

IFAIL $= 2$

On entry, some of the elements of the reference vector, REF, have been corrupted since the most recent call to G13DJF (or G13DKF).

IFAIL = 3

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

IFAIL = 4

This is an unlikely exit. For one of the series, overflow will occur if the forecasts are updated. The user should check whether the elements of REF have been corrupted.

7 Accuracy

The matrix computations are believed to be stable.

8 Further Comments

If a further m^* obervations, $Z_{n+\text{MLAST}+1}, Z_{n+\text{MLAST}+2}, \ldots, Z_{n+\text{MLAST}+m^*}$, become available, then forecasts of $Z_{n+\text{MLAST}+m^*+1}, Z_{n+\text{MLAST}+m^*+2}, \ldots, Z_{n+l_{\max}}$ may be updated by recalling G13DKF with $\text{M} = m^*$. Note that M and the contents of the array Z are the only quantities which need updating; MLAST is updated on exit from the previous call. On a successful exit, V contains estimates of $\epsilon_{n+\text{MLAST}+1}, \epsilon_{n+\text{MLAST}+2}, \ldots, \epsilon_{n+\text{MLAST}+m^*}$; columns $\text{MLAST}+1, \text{MLAST}+2, \ldots, \text{MLAST}+m^*$ of PREDZ contain the new observed values $Z_{n+\text{MLAST}+1}, Z_{n+\text{MLAST}+2}, \ldots, Z_{n+\text{MLAST}+m^*}$ and columns $\text{MLAST}+1, \text{MLAST}+2, \ldots, \text{MLAST}+m^*$ of SEFZ are set to zero.

9 Example

A program to update the forecasts of two series each of length 48. No transformation has been used and no differencing applied to either of the series. G13DCF is first called to fit an AR(1) model to the series. μ is to be estimated and $\phi_1(2,1)$ constrained to be zero. A call to G13DJF is then made in order to compute forecasts of the next five series values. After one new observation becomes available the four forecasts are updated. A further observation becomes available and the three forecasts are updated.

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.

```
*     G13DKF Example Program Text
*     Mark 15 Release. NAG Copyright 1991.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          KMAX, IK, IPMAX, IQMAX, NMAX, NPARMX, ICM, LWORK,
     +                 LIWORK, IDMAXL, LLMAX, LREF
      PARAMETER        (KMAX=3,IK=KMAX,IPMAX=3,IQMAX=3,NMAX=100,
     +                 NPARMX=(IPMAX+IQMAX)*KMAX*KMAX+KMAX,ICM=NPARMX,
     +                 LWORK=2000,LIWORK=100,IDMAXL=2,LLMAX=10,
     +                 LREF=(LLMAX-1)*KMAX*KMAX+2*KMAX*LLMAX+KMAX)
*     .. Local Scalars ..
      real             CGETOL, RLOGL
      INTEGER          I, IDMAX, IDMIN, IFAIL, IP, IPRINT, IQ, ISHOW, J,
     +                 K, LMAX, LPAR, M, MAXCAL, MLAST, N, ND, NITER
      LOGICAL          EXACT, MEANL
      CHARACTER        MEAN
*     .. Local Arrays ..
      real             CM(ICM,NPARMX), DELTA(IK,IDMAXL), G(NPARMX),
     +                 PAR(NPARMX), PREDZ(IK,LLMAX), QQ(IK,KMAX),
     +                 REF(LREF), SEFZ(IK,LLMAX), V(IK,NMAX),
     +                 W(IK,NMAX), WORK(LWORK), Z(IK,NMAX)
      INTEGER          ID(KMAX), IWORK(LIWORK)
      LOGICAL          PARHLD(NPARMX)
      CHARACTER        TR(KMAX)
*     .. External Subroutines ..
```

```
      EXTERNAL         FPRINT, G13DCF, G13DJF, G13DKF, G13DLF
*     .. Intrinsic Functions ..
      INTRINSIC        MAX, MIN
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G13DKF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) K, N, IP, IQ, MEAN, LMAX
      MEANL = .FALSE.
      LPAR = (IP+IQ)*K*K
      IF (MEAN.EQ.'M' .OR. MEAN.EQ.'m') THEN
         LPAR = LPAR + K
         MEANL = .TRUE.
      END IF
      IF (K.GT.0 .AND. K.LE.KMAX .AND. N.GE.1 .AND. N.LE.NMAX .AND.
     +    LPAR.GE.1 .AND. LPAR.LE.NPARMX .AND. LMAX.GE.1 .AND. LMAX.LE.
     +    LLMAX) THEN
         READ (NIN,*) (ID(I),I=1,K)
         IDMAX = 0
         IDMIN = 0
         DO 20 I = 1, K
            IDMIN = MIN(ID(I),IDMIN)
            IDMAX = MAX(ID(I),IDMAX)
   20    CONTINUE
         IF (IDMIN.GE.0 .AND. IDMAX.LE.IDMAXL) THEN
            DO 40 I = 1, K
               READ (NIN,*) (Z(I,J),J=1,N)
   40       CONTINUE
            READ (NIN,*) (TR(I),I=1,K)
            IF (IDMAX.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)
*
            DO 80 I = 1, LPAR
               PAR(I) = 0.0e0
               PARHLD(I) = .FALSE.
   80       CONTINUE
            DO 120 J = 1, K
               DO 100 I = J, K
                  QQ(I,J) = 0.0e0
  100          CONTINUE
  120       CONTINUE
            PARHLD(3) = .TRUE.
            EXACT = .TRUE.
*           ** Set IPRINT.gt.0 for no intermediate monitoring
            IPRINT = -1
            CGETOL = 0.0001e0
            MAXCAL = 40*LPAR*(LPAR+5)
*           ** Set ISHOW.eq.0 for no results from G13DCF
            ISHOW = 0
            IFAIL = 1
*
            CALL G13DCF(K,ND,IP,IQ,MEANL,PAR,LPAR,QQ,IK,W,PARHLD,EXACT,
     +                  IPRINT,CGETOL,MAXCAL,ISHOW,NITER,RLOGL,V,G,CM,
     +                  ICM,WORK,LWORK,IWORK,LIWORK,IFAIL)
*
            IF (IFAIL.EQ.0 .OR. IFAIL.GE.4) THEN
               IFAIL = 0
*
               CALL G13DJF(K,N,Z,IK,TR,ID,DELTA,IP,IQ,MEAN,PAR,LPAR,QQ,
     +                     V,LMAX,PREDZ,SEFZ,REF,LREF,WORK,LWORK,IWORK,
     +                     LIWORK,IFAIL)
*
               CALL FPRINT(K,N,LMAX,PREDZ,SEFZ,IK,NOUT)
               M = 1
               MLAST = 0
```

```
               Z(1,1) = 8.1e0
               Z(2,1) = 10.2e0
               IFAIL = 0
*
               CALL G13DKF(K,LMAX,M,MLAST,Z,IK,REF,LREF,V,PREDZ,SEFZ,
     +                     WORK,IFAIL)
*
               CALL FPRINT(K,N+MLAST,LMAX,PREDZ,SEFZ,IK,NOUT)
               M = 1
*              Leave MLAST unchanged from last call
               Z(1,1) = 8.5e0
               Z(2,1) = 10.0e0
               IFAIL = 0
*
               CALL G13DKF(K,LMAX,M,MLAST,Z,IK,REF,LREF,V,PREDZ,SEFZ,
     +                     WORK,IFAIL)
*
               CALL FPRINT(K,N+MLAST,LMAX,PREDZ,SEFZ,IK,NOUT)
            END IF
         END IF
      END IF
      STOP
      END
*
      SUBROUTINE FPRINT(K,NM,LMAX,PREDZ,SEFZ,IK,NOUT)
*     .. Scalar Arguments ..
      INTEGER           IK, K, LMAX, NM, NOUT
*     .. Array Arguments ..
      real              PREDZ(IK,LMAX), SEFZ(IK,LMAX)
*     .. Local Scalars ..
      INTEGER           I, I2, J, L, L2, LOOP
*     .. Intrinsic Functions ..
      INTRINSIC         MIN, MOD
*     .. Executable Statements ..
      WRITE (NOUT,*)
      WRITE (NOUT,*) ' FORECAST SUMMARY TABLE'
      WRITE (NOUT,*) ' ----------------------'
      WRITE (NOUT,*)
      WRITE (NOUT,99999) ' Forecast origin is set at t = ', NM
      WRITE (NOUT,*)
      LOOP = LMAX/5
      IF (MOD(LMAX,5).NE.0) LOOP = LOOP + 1
      DO 40 J = 1, LOOP
         I2 = (J-1)*5
         L2 = MIN(I2+5,LMAX)
         WRITE (NOUT,99998) 'Lead Time ', (I,I=I2+1,L2)
         WRITE (NOUT,*)
         I = 1
         WRITE (NOUT,99997) 'Series ', I, ' : Forecast    ',
     +     (PREDZ(1,L),L=I2+1,L2)
         WRITE (NOUT,99996) ' : Standard Error ', (SEFZ(1,L),L=I2+1,L2)
         DO 20 I = 2, K
            WRITE (NOUT,99997) 'Series ', I, ' : Forecast    ',
     +        (PREDZ(I,L),L=I2+1,L2)
            WRITE (NOUT,99996) ' : Standard Error ',
     +        (SEFZ(I,L),L=I2+1,L2)
   20    CONTINUE
         WRITE (NOUT,*)
   40 CONTINUE
      RETURN
*
99999 FORMAT (1X,A,I4)
99998 FORMAT (1X,A,12X,5I10)
99997 FORMAT (1X,A,I2,A,5F10.2)
99996 FORMAT (10X,A,4(F7.2,3X),F7.2)
      END
```

9.2 Program Data

```
G13DKF Example Program Data
2 48 1 0 'M' 5  :  K, N, IP, IQ, MEAN, LMAX
0  0            :  ID(I),I=1,K
-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)
```

9.3 Program Results

```
 G13DKF Example Program Results

  FORECAST SUMMARY TABLE
  ----------------------

  Forecast origin is set at t =   48

 Lead Time                      1         2         3         4         5

 Series  1 : Forecast          7.82      7.28      6.77      6.33      5.95
           : Standard Error    1.72      2.23      2.51      2.68      2.79
 Series  2 : Forecast         10.31      9.25      8.65      8.30      8.10
           : Standard Error    2.32      2.68      2.78      2.82      2.83


  FORECAST SUMMARY TABLE
  ----------------------

  Forecast origin is set at t =   49

 Lead Time                      1         2         3         4         5

 Series  1 : Forecast          8.10      7.49      6.94      6.46      6.06
           : Standard Error    0.00      1.72      2.23      2.51      2.68
 Series  2 : Forecast         10.20      9.19      8.61      8.28      8.08
           : Standard Error    0.00      2.32      2.68      2.78      2.82


  FORECAST SUMMARY TABLE
  ----------------------

  Forecast origin is set at t =   50

 Lead Time                      1         2         3         4         5

 Series  1 : Forecast          8.10      8.50      7.80      7.18      6.65
           : Standard Error    0.00      0.00      1.72      2.23      2.51
 Series  2 : Forecast         10.20     10.00      9.08      8.54      8.24
           : Standard Error    0.00      0.00      2.32      2.68      2.78
```