NAG Library Routine Document

G13AFF

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

G13AFF is an easy-to-use version of G13AEF. It fits a seasonal autoregressive integrated moving average (ARIMA) model to an observed time series, using a nonlinear least squares procedure incorporating backforecasting. Parameter estimates are obtained, together with appropriate standard errors. The residual series is returned, and information for use in forecasting the time series is produced for use in G13AGF and G13AHF.

The estimation procedure is iterative, starting with initial parameter values such as may be obtained using G13ADF. It continues until a specified convergence criterion is satisfied or until a specified number of iterations have been carried out. The progress of the iteration can be monitored by means of an optional printing facility.

2 Specification

```
SUBROUTINE G13AFF (MR, PAR, NPAR, C, KFC, X, NX, S, NDF, SD, NPPC, CM,      &
                   LDCM, ST, NST, KPIV, NIT, ITC, ISF, RES, IRES, NRES,     &
                   IFAIL)
INTEGER            MR(7), NPAR, KFC, NX, NDF, NPPC, LDCM, NST, KPIV,        &
                   NIT, ITC, ISF(4), IRES, NRES, IFAIL
REAL (KIND=nag_wp) PAR(NPAR), C, X(NX), S, SD(NPPC), CM(LDCM,NPPC),        &
                   ST(NX), RES(IRES)
```

3 Description

The time series $x_1, x_2, \ldots, x_n$ supplied to the routine is assumed to follow a seasonal autoregressive integrated moving average (ARIMA) model defined as follows:

$$\nabla^d \nabla_s^D x_t - c = w_t,$$

where $\nabla^d \nabla_s^D x_t$ is the result of applying non-seasonal differencing of order d and seasonal differencing of seasonality s and order D to the series x_t, as outlined in the description of G13AAF. The differenced series is then of length $N = n - d'$, where $d' = d + (D \times s)$ is the generalized order of differencing. The scalar c is the expected value of the differenced series, and the series $w_1, w_2, \ldots, w_N$ follows a zero-mean stationary autoregressive moving average (ARMA) model defined by a pair of recurrence equations. These express w_t in terms of an uncorrelated series a_t, via an intermediate series e_t. The first equation describes the seasonal structure:

$$w_t = \Phi_1 w_{t-s} + \Phi_2 w_{t-2\times s} + \cdots + \Phi_P w_{t-P\times s} + e_t - \Theta_1 e_{t-s} - \Theta_2 e_{t-2\times s} - \cdots - \Theta_Q e_{t-Q\times s}.$$

The second equation describes the non-seasonal structure. If the model is purely non-seasonal the first equation is redundant and e_t above is equated with w_t:

$$e_t = \phi_1 e_{t-1} + \phi_2 e_{t-2} + \cdots + \phi_p e_{t-p} + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \cdots - \theta_q a_{t-q}.$$

Estimates of the model parameters defined by

$$\begin{array}{l}\phi_1, \phi_2, \ldots, \phi_p, \theta_1, \theta_2, \ldots, \theta_q, \\ \Phi_1, \Phi_2, \ldots, \Phi_P, \Theta_1, \Theta_2, \ldots, \Theta_Q\end{array}$$

and (optionally) c are obtained by minimizing a quadratic form in the vector $w = (w_1, w_2, \ldots, w_N)'$.

The minimization process is iterative, iterations being performed until convergence is achieved (see Section 3 in G13AEF for full details), or until the user-specified maximum number of iterations are completed.

The final values of the residual sum of squares and the parameter estimates are used to obtain asymptotic approximations to the standard deviations of the parameters, and the correlation matrix for the parameters. The 'state set' array of information required by forecasting is also returned.

Note: if the maximum number of iterations are performed without convergence, these quantities may not be reliable. In this case, the sequence of iterates should be checked, using the optional monitoring routine, to verify that convergence is adequate for practical purposes.

4 References

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

Marquardt D W (1963) An algorithm for least squares estimation of nonlinear parameters *J. Soc. Indust. Appl. Math.* **11** 431

5 Arguments

1: MR(7) – INTEGER array *Input*

On entry: the orders vector (p, d, q, P, D, Q, s) of the ARIMA model whose parameters are to be estimated. p, q, P and Q refer respectively to the number of autoregressive (ϕ), moving average (θ), seasonal autoregressive (Φ) and seasonal moving average (Θ) parameters. d, D and s refer respectively to the order of non-seasonal differencing, the order of seasonal differencing and the seasonal period.

Constraints:

p, d, q, P, D, Q, $s \geq 0$;
$p + q + P + Q > 0$;
$s \neq 1$;
if $s = 0$, $P + D + Q = 0$;
if $s > 1$, $P + D + Q > 0$;
$d + s \times (P + D) \leq n$;
$p + d - q + s \times (P + D - Q) \leq n$.

2: PAR(NPAR) – REAL (KIND=nag_wp) array *Input/Output*

On entry: the initial estimates of the p values of the ϕ parameters, the q values of the θ parameters, the P values of the Φ parameters and the Q values of the Θ parameters, in that order.

On exit: contains the latest values of the estimates of these parameters.

3: NPAR – INTEGER *Input*

On entry: the total number of ϕ, θ, Φ and Θ parameters to be estimated.

Constraint: NPAR $= p + q + P + Q$.

4: C – REAL (KIND=nag_wp) *Input/Output*

On entry: if KFC $= 0$, C must contain the expected value, c, of the differenced series.

If KFC $= 1$, C must contain an initial estimate of c.

Therefore, if C and KFC are both zero on entry, there is no constant correction.

On exit: if KFC $= 0$, C is unchanged.

If KFC $= 1$, C contains the latest estimate of c.

5: KFC – INTEGER *Input*

On entry: must be set to 1 if the constant, c, is to be estimated and 0 if it is to be held fixed at its initial value.

Constraint: $\mathrm{KFC} = 0$ or 1.

6: X(NX) – REAL (KIND=nag_wp) array *Input*

On entry: the n values of the original undifferenced time series.

7: NX – INTEGER *Input*

On entry: n, the length of the original undifferenced time series.

8: S – REAL (KIND=nag_wp) *Output*

On exit: the residual sum of squares after the latest series of parameter estimates has been incorporated into the model. If the routine exits with a faulty input argument, S contains zero.

9: NDF – INTEGER *Output*

On exit: the number of degrees of freedom associated with S,
$\mathrm{NDF} = n - d - D \times s - p - q - P - Q - \mathrm{KFC}$.

10: SD(NPPC) – REAL (KIND=nag_wp) array *Output*

On exit: the standard deviations corresponding to the parameters in the model (p autoregressive, q moving average, P seasonal autoregressive, Q seasonal moving average and c, if estimated, in that order). If the routine exits with IFAIL containing a value other than 0 or 9, or if the required number of iterations is zero, the contents of SD will be indeterminate.

11: NPPC – INTEGER *Input*

On entry: the number of ϕ, θ, Φ, Θ and c parameters to be estimated. $\mathrm{NPPC} = p + q + P + Q + 1$ if the constant is being estimated and $\mathrm{NPPC} = p + q + P + Q$ if not.

Constraint: $\mathrm{NPPC} = \mathrm{NPAR} + \mathrm{KFC}$.

12: CM(LDCM, NPPC) – REAL (KIND=nag_wp) array *Output*

On exit: the correlation coefficients associated with each pair of the NPPC parameters. These are held in the first NPPC rows and the first NPPC columns of CM. These correlation coefficients are indeterminate if IFAIL contains on exit a value other than 0 or 9, or if the required number of iterations is zero.

13: LDCM – INTEGER *Input*

On entry: the first dimension of the array CM as declared in the (sub)program from which G13AFF is called.

Constraint: $\mathrm{LDCM} \geq \mathrm{NPPC}$.

14: ST(NX) – REAL (KIND=nag_wp) array *Output*

On exit: the value of the state set in its first NST elements. If the routine exits with IFAIL containing a value other than 0 or 9, the contents of ST will be indeterminate.

15: NST – INTEGER *Output*

On exit: the size of the state set. $\mathrm{NST} = P \times s + D \times s + d + q + \max(p, Q \times s)$.

NST should be used subsequently in G13AGF and G13AHF as the dimension of ST.

16: KPIV – INTEGER *Input*

On entry: must be nonzero if the progress of the optimization is to be monitored using the built-in printing facility. Otherwise KPIV must contain zero. If selected, monitoring output will be sent to the current advisory message unit defined by X04ABF. For each iteration, the heading

```
G13AFZ MONITORING OUTPUT - ITERATION n
```

followed by the argument values, and residual sum of squares, are printed.

17: NIT – INTEGER *Input*

On entry: the maximum number of iterations to be performed.

Constraint: $\mathrm{NIT} \geq 0$.

18: ITC – INTEGER *Output*

On exit: the number of iterations performed.

19: ISF(4) – INTEGER array *Output*

On exit: the first four elements of ISF contain success/failure indicators, one for each of the four types of parameter in the model (autoregressive, moving average, seasonal autoregressive, seasonal moving average), in that order.

Each indicator has the interpretation:

-2 On entry parameters of this type have initial estimates which do not satisfy the stationarity or invertibility test conditions.

-1 The search procedure has failed to converge because the latest set of parameter estimates of this type is invalid.

0 No parameter of this type is in the model.

1 Valid final estimates for parameters of this type have been obtained.

20: RES(IRES) – REAL (KIND=nag_wp) array *Output*

On exit: the first NRES elements of RES contain the model residuals derived from the differenced series. If the routine exits with IFAIL holding a value other than 0 or 9, these elements of RES will be indeterminate. The rest of the array RES is used as workspace.

21: IRES – INTEGER *Input*

On entry: the dimension of the array RES as declared in the (sub)program from which G13AFF is called.

Constraint: $\mathrm{IRES} \geq 15 \times Q' + 11n + 13 \times \mathrm{NPPC} + 8 \times P' + 12 + 2 \times (Q' + \mathrm{NPPC})^2$, where $P' = p + (P \times s)$ and $Q' = q + (Q \times s)$.

22: NRES – INTEGER *Output*

On exit: the number of model residuals returned in RES.

23: 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, because for this routine the values of the output arguments may be useful even if $\mathrm{IFAIL} \neq 0$ on exit, the recommended value is -1. **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).

Note: G13AFF may return useful information for one or more of the following detected errors or warnings.

Errors or warnings detected by the routine:

IFAIL $= 1$

On entry, NPAR $\neq p + q + P + Q$,
or the orders vector MR is invalid (check the constraints in Section 5),
or KFC $\neq 0$ or 1,
or NPPC $\neq$ NPAR $+$ KFC.

IFAIL $= 2$

On entry, NX $- d - D \times s \leq$ NPAR $+$ KFC, i.e., the number of terms in the differenced series is not greater than the number of parameters in the model. The model is over-parameterised.

IFAIL $= 3$

On entry, NIT < 0.

IFAIL $= 4$

On entry, the required size of the state set array ST is greater than NX. This occurs only for very unusual models with long seasonal periods or large numbers of parameters. First check that the orders vector MR has been set up as intended. If it has, change to G13AEF with ST dimensioned at least (NST), where NST is the value returned by G13AFF, or computed using the formula in Section 5 of this document.

IFAIL $= 5$

On entry, the workspace array RES is too small. Check the value of IRES against the constraints in Section 5.

IFAIL $= 6$

On entry, LDCM $<$ NPPC.

IFAIL $= 7$

The search procedure in the algorithm has failed. This may be due to a badly conditioned sum of squares function, or the default convergence criterion may be too strict. Use G13AEF with a less strict convergence criterion.

Some output arguments may contain meaningful values; see Section 5 for details.

IFAIL $= 8$

The inversion of the Hessian matrix in the calculation of the covariance matrix of the parameter estimates has failed.

Some output arguments may contain meaningful values; see Section 5 for details.

IFAIL $= 9$

This indicates a failure when solving the equations giving the latest estimates of the backforecasts.

Some output arguments may contain meaningful values; see Section 5 for details.

IFAIL $= 10$

Satisfactory parameter estimates could not be obtained for all parameter types in the model. Inspect array ISF for further information on the parameter type(s) in error.

IFAIL $= 11$

An internal error has arisen in partitioning RES for use by G13AEF. This error should not occur; report it to NAG via your site representative.

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

G13AFF is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

G13AFF 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 by G13AFF is approximately proportional to $\mathrm{NX} \times \mathrm{ITC} \times (q + Q \times s + \mathrm{NPPC})^2$.

10 Example

This example reads 30 observations from a time series relating to the rate of the earth's rotation about its polar axis. Differencing of order 1 is applied, and the number of non-seasonal parameters is 3, one autoregressive (ϕ) and two moving average (θ). No seasonal effects are taken into account.

The constant is estimated. Up to 50 iterations are allowed.

The initial estimates of ϕ_1, θ_1, θ_2 and c are zero.

Some intermediate monitoring output from G13AFZ has been omitted.

10.1 Program Text

```
    Program g13affe

!     G13AFF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: g13aff, nag_wp, x04abf, x04caf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: c, s
      Integer                          :: ifail, ipd, iqd, ires, itc, kfc,      &
                                          kpiv, ldcm, nadv, ndf, nit, npar,     &
                                          nppc, nres, nst, nx
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: cm(:,:), par(:), res(:), sd(:),       &
                                          st(:), x(:)
      Integer                          :: isf(4), mr(7)
!     .. Executable Statements ..
      Write (nout,*) 'G13AFF Example Program Results'
      Write (nout,*)

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

!     Read in the problem size etc
      Read (nin,*) nx, kfc, c

!     Read in the orders
      Read (nin,*) mr(1:7)

!     Calculate NPAR and various array lengths
      npar = mr(1) + mr(3) + mr(4) + mr(6)
      nppc = npar + kfc
      iqd = mr(6)*mr(7) + mr(3)
      ipd = mr(4)*mr(7) + mr(1)
      ires = 15*iqd + 11*nx + 13*nppc + 8*ipd + 12 + 2*(iqd+nppc)**2

      ldcm = nppc
      Allocate (x(nx),par(npar),sd(nppc),cm(ldcm,nppc),st(nx),res(ires))

!     Read in data
      Read (nin,*) x(1:nx)

!     Read in initial values
      Read (nin,*) par(1:npar)

!     Read in control parameters
      Read (nin,*) kpiv, nit

!     Set the advisory channel to NOUT for monitoring information
      If (kpiv/=0) Then
        nadv = nout
        Call x04abf(1,nadv)
      End If

!     Fit ARIMA model
      ifail = -1
      Call g13aff(mr,par,npar,c,kfc,x,nx,s,ndf,sd,nppc,cm,ldcm,st,nst,kpiv,     &
        nit,itc,isf,res,ires,nres,ifail)
      If (ifail/=0) Then
        If (ifail<7) Then
          Go To 100
        End If
      End If
```

```
!     Display results
      If (ifail==0) Then
        Write (nout,99999) 'Convergence was achieved after', itc, ' cycles'
      Else
        Write (nout,99999) 'Iterative process ran for', itc, ' cycles'
      End If
      Write (nout,*)
      Write (nout,*)                                                           &
        'Final values of the PAR parameters and the constant are as follows'
      Write (nout,99996) par(1:npar), c
      Write (nout,*)
      Write (nout,99995) 'Residual sum of squares is', s, '  with', ndf,       &
        ' degrees of freedom'
      If ((ifail==0 .Or. ifail==9) .And. itc>0) Then
        Write (nout,*)
        Write (nout,*) 'The corresponding SD array holds'
        Write (nout,99994) sd(1:nppc)
        Write (nout,*)
        Flush (nout)
        ifail = 0
        Call x04caf('General',' ',nppc,nppc,cm,ldcm,                           &
          'The correlation matrix is as follows',ifail)
      End If
      If (ifail==0 .Or. ifail==9) Then
        Write (nout,*)
        Write (nout,99999) 'The residuals consist of', nres, ' values'
        Write (nout,99998) res(1:nres)
        Write (nout,*)
        Write (nout,99997) 'The state set consists of', nst, ' values'
        Write (nout,99993) st(1:nst)
      End If

100   Continue

99999 Format (1X,A,I4,A)
99998 Format (1X,5F10.4)
99997 Format (1X,A,I3,A)
99996 Format (1X,4F10.4)
99995 Format (1X,A,F10.3,A,I4,A)
99994 Format (1X,10F9.4)
99993 Format (1X,6F11.3)
    End Program g13affe
```

10.2 Program Data

```
G13AFF Example Program Data
 30  1 0.0                                          :: NX,KFC,C
1  1  2  0  0  0  0                                 :: MR
 -217 -177 -166 -136 -110  -95  -64  -37  -14  -25
  -51  -62  -73  -88 -113 -120  -83  -33  -19   21
   17   44   44   78   88  122  126  114   85   64  :: End of X
0.0 0.0 0.0                                         :: PAR (initial values)
0  50                                               :: KPIV,NIT
```

10.3 Program Results

```
 G13AFF Example Program Results

 Convergence was achieved after  25 cycles

 Final values of the PAR parameters and the constant are as follows
   -0.0543   -0.5548   -0.6734    9.9848

 Residual sum of squares is  9397.220  with  25 degrees of freedom

 The corresponding SD array holds
   0.3457   0.2636   0.1665   7.4170

 The correlation matrix is as follows
          1       2       3       4
 1   1.0000  0.8072  0.3548 -0.0404
 2   0.8072  1.0000  0.4681 -0.0491
 3   0.3548  0.4681  1.0000 -0.0376
 4  -0.0404 -0.0491 -0.0376  1.0000

 The residuals consist of  29 values
    19.6275   -5.3093    9.7983   15.2412   -9.1693
    16.1107   15.3929   -5.4500  -27.6205  -18.1306
     5.7202  -13.0881  -22.7151  -14.9256    4.6930
    33.5406   19.7138  -27.3360   32.1231  -11.7681
     1.1524   -1.7756   23.6821  -10.6238   13.9619
    -5.2727  -28.7868  -20.6573   -2.2555

 The state set consists of  4 values
      64.000    -30.985    -20.657     -2.256
```