NAG Library Routine Document

G13AMF

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

G13AMF performs exponential smoothing using either single exponential, double exponential or a Holt–Winters method.

2 Specification

```
SUBROUTINE G13AMF (MODE, ITYPE, P, PARAM, N, Y, K, INIT, NF, FV, FSE,      &
                   YHAT, RES, DV, AD, R, IFAIL)

INTEGER           MODE, ITYPE, P, N, K, NF, IFAIL
REAL (KIND=nag_wp) PARAM(*), Y(N), INIT(*), FV(NF), FSE(NF), YHAT(N),       &
                   RES(N), DV, AD, R(*)
```

3 Description

Exponential smoothing is a relatively simple method of short term forecasting for a time series. G13AMF provides five types of exponential smoothing; single exponential, Brown's double exponential, linear Holt (also called double exponential smoothing in some references), additive Holt–Winters and multiplicative Holt–Winters. The choice of smoothing method used depends on the characteristics of the time series. If the mean of the series is only slowly changing then single exponential smoothing may be suitable. If there is a trend in the time series, which itself may be slowly changing, then double exponential smoothing may be suitable. If there is a seasonal component to the time series, e.g., daily or monthly data, then one of the two Holt–Winters methods may be suitable.

For a time series y_t, for $t = 1, 2, \ldots, n$, the five smoothing functions are defined by the following:

Single Exponential Smoothing

$$
\begin{aligned}
m_t &= \alpha y_t + (1 - \alpha) m_{t-1} \\
\hat{y}_{t+f} &= m_t \\
\mathrm{var}\left(\hat{y}_{t+f}\right) &= \mathrm{var}\left(\epsilon_t\right)\left(1 + (f - 1)\alpha^2\right)
\end{aligned}
$$

Brown Double Exponential Smoothing

$$
\begin{aligned}
m_t &= \alpha y_t + (1 - \alpha) m_{t-1} \\
r_t &= \alpha(m_t - m_{t-1}) + (1 - \alpha) r_{t-1} \\
\hat{y}_{t+f} &= m_t + ((f - 1) + 1/\alpha) r_t \\
\mathrm{var}\left(\hat{y}_{t+f}\right) &= \mathrm{var}\left(\epsilon_t\right)\left(1 + \sum_{i=1}^{f-1}\left(2\alpha + (i - 1)\alpha^2\right)^2\right)
\end{aligned}
$$

Linear Holt Smoothing

$$
\begin{aligned}
m_t &= \alpha y_t + (1 - \alpha)(m_{t-1} + \phi r_{t-1}) \\
r_t &= \gamma(m_t - m_{t-1}) + (1 - \gamma)\phi r_{t-1} \\
\hat{y}_{t+f} &= m_t + \sum_{i=1}^{f} \phi^i r_t \\
\mathrm{var}\left(\hat{y}_{t+f}\right) &= \mathrm{var}\left(\epsilon_t\right)\left(1 + \sum_{i=1}^{f-1}\left(\alpha + \frac{\alpha\gamma\phi(\phi^i - 1)}{(\phi - 1)}\right)^2\right)
\end{aligned}
$$

Additive Holt−Winters Smoothing

$$m_t = \alpha\big(y_t - s_{t-p}\big) + (1 - \alpha)(m_{t-1} + \phi r_{t-1})$$
$$r_t = \gamma(m_t - m_{t-1}) + (1 - \gamma)\phi r_{t-1}$$
$$s_t = \beta(y_t - m_t) + (1 - \beta)s_{t-p}$$
$$\hat{y}_{t+f} = m_t + \left(\sum_{i=1}^{f}\phi^i r_t\right) + s_{t-p}$$
$$\text{var}\left(\hat{y}_{t+f}\right) = \text{var}\left(\epsilon_t\right)\left(1 + \sum_{i=1}^{f-1}\psi_i^2\right)$$
$$\psi_i = \begin{cases} 0 & \text{if } i \geq f \\ \alpha + \frac{\alpha\gamma\phi(\phi^i-1)}{(\phi-1)} & \text{if } i \bmod p \neq 0 \\ \alpha + \frac{\alpha\gamma\phi(\phi^i-1)}{(\phi-1)} + \beta(1-\alpha) & \text{otherwise} \end{cases}$$

Multiplicative Holt−Winters Smoothing

$$m_t = \alpha y_t / s_{t-p} + (1 - \alpha)(m_{t-1} + \phi r_{t-1})$$
$$r_t = \gamma(m_t - m_{t-1}) + (1 - \gamma)\phi r_{t-1}$$
$$s_t = \beta y_t / m_t + (1 - \beta)s_{t-p}$$
$$\hat{y}_{t+f} = \left(m_t + \sum_{i=1}^{f}\phi^i r_t\right) \times s_{t-p}$$
$$\text{var}\left(\hat{y}_{t+f}\right) = \text{var}\left(\epsilon_t\right)\left(\sum_{i=0}^{\infty}\sum_{j=0}^{p-1}\left(\psi_{j+ip}\frac{s_{t+f}}{s_{t+f-j}}\right)^2\right)$$

and ψ is defined as in the additive Holt−Winters smoothing,

where m_t is the mean, r_t is the trend and s_t is the seasonal component at time t with p being the seasonal order. The f-step ahead forecasts are given by $\hat{y}_{t+f}$ and their variances by $\text{var}\left(\hat{y}_{t+f}\right)$. The term $\text{var}\left(\epsilon_t\right)$ is estimated as the mean deviation.

The parameters, α, β and γ control the amount of smoothing. The nearer these parameters are to one, the greater the emphasis on the current data point. Generally these parameters take values in the range 0.1 to 0.3. The linear Holt and two Holt−Winters smoothers include an additional parameter, ϕ, which acts as a trend dampener. For $0.0 < \phi < 1.0$ the trend is dampened and for $\phi > 1.0$ the forecast function has an exponential trend, $\phi = 0.0$ removes the trend term from the forecast function and $\phi = 1.0$ does not dampen the trend.

For all methods, values for α, β, γ and ψ can be chosen by trying different values and then visually comparing the results by plotting the fitted values along side the original data. Alternatively, for single exponential smoothing a suitable value for α can be obtained by fitting an $\text{ARIMA}(0, 1, 1)$ model (see G13BEF). For Brown's double exponential smoothing and linear Holt smoothing with no dampening, (i. e., $\phi = 1.0$), suitable values for α and γ can be obtained by fitting an $\text{ARIMA}(0, 2, 2)$ model. Similarly, the linear Holt method, with $\phi \neq 1.0$, can be expressed as an $\text{ARIMA}(1, 2, 2)$ model and the additive Holt−Winters, with no dampening, ($\phi = 1.0$), can be expressed as a seasonal ARIMA model with order p of the form $\text{ARIMA}(0, 1, p + 1)(0, 1, 0)$. There is no similar procedure for obtaining parameter values for the multiplicative Holt−Winters method, or the additive Holt−Winters method with $\phi \neq 1.0$. In these cases parameters could be selected by minimizing a measure of fit using one of the nonlinear optimization routines in Chapter E04.

In addition to values for α, β, γ and ψ, initial values, m_0, r_0 and s_{-j}, for $j = 0, 1, \ldots, p - 1$, are required to start the smoothing process. You can either supply these or they can be calculated by G13AMF from the first k observations. For single exponential smoothing the mean of the observations is used to estimate m_0. For Brown double exponential smoothing and linear Holt smoothing, a simple linear regression is carried out with the series as the dependent variable and the sequence $1, 2, \ldots, k$ as the independent variable. The intercept is then used to estimate m_0 and the slope to estimate r_0. In the case of the additive Holt−Winters method, the same regression is carried out, but a separate intercept is used for each of the p seasonal groupings. The slope gives an estimate for r_0 and the mean of the p intercepts is used as the estimate of m_0. The seasonal parameters s_{-j}, for $j = 0, 1, \ldots, p - 1$, are estimated as the p intercepts $- m_0$. A similar approach is adopted for the multiplicative Holt−Winter's method.

One step ahead forecasts, $\hat{y}_{t+1}$ are supplied along with the residuals computed as $(y_{t+1} - \hat{y}_{t+1})$. In addition, two measures of fit are provided. The mean absolute deviation,

$$\frac{1}{n}\sum_{t=1}^{n}|y_t - \hat{y}_t|$$

and the square root of the mean deviation

$$\sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}.$$

4   References

Chatfield C (1980) *The Analysis of Time Series* Chapman and Hall

5   Arguments

1:    MODE – INTEGER *Input*

On entry: indicates if G13AMF is continuing from a previous call or, if not, how the initial values are computed.

MODE = 0
>   Required values for m_0, r_0 and s_{-j}, for $j = 0, 1, \ldots, p - 1$, are supplied in INIT.

MODE = 1
>   G13AMF continues from a previous call using values that are supplied in R.

MODE = 2
>   Required values for m_0, r_0 and s_{-j}, for $j = 0, 1, \ldots, p - 1$, are estimated using the first k observations.

Constraint: MODE = 0, 1 or 2.

2:    ITYPE – INTEGER *Input*

On entry: the smoothing function.

ITYPE = 1
>   Single exponential.

ITYPE = 2
>   Brown double exponential.

ITYPE = 3
>   Linear Holt.

ITYPE = 4
>   Additive Holt–Winters.

ITYPE = 5
>   Multiplicative Holt–Winters.

Constraint: ITYPE = 1, 2, 3, 4 or 5.

3:    P – INTEGER *Input*

On entry: if ITYPE = 4 or 5, the seasonal order, p, otherwise P is not referenced.

Constraint: if ITYPE = 4 or 5, P > 1.

4: PARAM(∗) – REAL (KIND=nag_wp) array *Input*

Note: the dimension of the array PARAM must be at least 1 if ITYPE = 1 or 2, 3 if ITYPE = 3 and at least 4 if ITYPE = 4 or 5.

On entry: the smoothing parameters.

If ITYPE = 1 or 2, PARAM(1) = α and any remaining elements of PARAM are not referenced.

If ITYPE = 3, PARAM(1) = α, PARAM(2) = γ, PARAM(3) = ϕ and any remaining elements of PARAM are not referenced.

If ITYPE = 4 or 5, PARAM(1) = α, PARAM(2) = γ, PARAM(3) = β and PARAM(4) = ϕ.

Constraints:

if ITYPE = 1, $0.0 \le \alpha \le 1.0$;
if ITYPE = 2, $0.0 < \alpha \le 1.0$;
if ITYPE = 3, $0.0 \le \alpha \le 1.0$ and $0.0 \le \gamma \le 1.0$ and $\phi \ge 0.0$;
if ITYPE = 4 or 5, $0.0 \le \alpha \le 1.0$ and $0.0 \le \gamma \le 1.0$ and $0.0 \le \beta \le 1.0$ and $\phi \ge 0.0$.

5: N – INTEGER *Input*

On entry: the number of observations in the series.

Constraint: $N \ge 0$.

6: Y(N) – REAL (KIND=nag_wp) array *Input*

On entry: the time series.

7: K – INTEGER *Input*

On entry: if MODE = 2, the number of observations used to initialize the smoothing.

If MODE $\ne$ 2, K is not referenced.

Constraints:

if MODE = 2 and ITYPE = 4 or 5, $2 \times P \le K \le N$;
if MODE = 2 and ITYPE = 1, 2 or 3, $1 \le K \le N$.

8: INIT(∗) – REAL (KIND=nag_wp) array *Input/Output*

Note: the dimension of the array INIT must be at least 1 if ITYPE = 1, 2 if ITYPE = 2 or 3 and at least $2 + P$ if ITYPE = 4 or 5.

On entry: if MODE = 0, the initial values for m_0, r_0 and s_{-j}, for $j = 0, 1, \ldots, p - 1$, used to initialize the smoothing.

If ITYPE = 1, INIT(1) = m_0 and the remaining elements of INIT are not referenced.

If ITYPE = 2 or 3, INIT(1) = m_0 and INIT(2) = r_0 and the remaining elements of INIT are not referenced.

If ITYPE = 4 or 5, INIT(1) = m_0, INIT(2) = r_0 and INIT(3) to INIT($p + 2$) hold the values for s_{-j}, for $j = 0, 1, \ldots, p - 1$. The remaining elements of INIT are not referenced.

On exit: if MODE $\ne$ 1, the values used to initialize the smoothing. These are in the same order as described above.

9: NF – INTEGER *Input*

On entry: the number of forecasts required beyond the end of the series. Note, the one step ahead forecast is always produced.

Constraint: $NF \ge 0$.

10:     FV(NF) – REAL (KIND=nag_wp) array   *Output*

On exit: $\hat{y}_{t+f}$, for $f = 1, 2, \ldots, \mathrm{NF}$, the next NF step forecasts. Where $t = \mathrm{N}$, if MODE $\neq 1$, else t is the total number of smoothed and forecast values already produced.

11:     FSE(NF) – REAL (KIND=nag_wp) array   *Output*

On exit: the forecast standard errors for the values given in FV.

12:     YHAT(N) – REAL (KIND=nag_wp) array   *Output*

On exit: $\hat{y}_{t+1}$, for $t = 1, 2, \ldots, \mathrm{N}$, the one step ahead forecast values, with YHAT(i) being the one step ahead forecast of Y($i - 1$).

13:     RES(N) – REAL (KIND=nag_wp) array   *Output*

On exit: the residuals, $(y_{t+1} - \hat{y}_{t+1})$, for $t = 1, 2, \ldots, \mathrm{N}$.

14:     DV – REAL (KIND=nag_wp)   *Output*

On exit: the square root of the mean deviation.

15:     AD – REAL (KIND=nag_wp)   *Output*

On exit: the mean absolute deviation.

16:     R($*$) – REAL (KIND=nag_wp) array   *Input/Output*

Note: the dimension of the array R must be at least 13 if ITYPE $= 1$, 2 or 3 and at least $13 + \mathrm{P}$ if ITYPE $= 4$ or 5.

On entry: if MODE $= 1$, R must contain the values as returned by a previous call to G05PMF or G13AMF, R need not be set otherwise.

If ITYPE $= 1$, 2 or 3, only the first 13 elements of R are referenced, otherwise the first $13 + p$ elements are referenced.

On exit: the information on the current state of the smoothing.

Constraint: if MODE $= 1$, R must have been initialized by at least one previous call to G05PMF or G13AMF with MODE $\neq 1$, and R should not have been changed since the last call to G05PMF or G13AMF.

17:     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, MODE = $\langle value \rangle$.
Constraint: MODE = 0, 1 or 2.

IFAIL = 2

On entry, ITYPE = $\langle value \rangle$.
Constraint: ITYPE = 1, 2, 3, 4 or 5.

IFAIL = 3

On entry, P = $\langle value \rangle$.
Constraint: if ITYPE = 4 or 5, P > 1.

IFAIL = 4

On entry, PARAM($\langle value \rangle$) = $\langle value \rangle$.
Constraint: $0.0 \leq$ PARAM($\langle value \rangle$) ≤ 1.0.

On entry, PARAM($\langle value \rangle$) = $\langle value \rangle$.
Constraint: if ITYPE = 2, $0.0 <$ PARAM($\langle value \rangle$) ≤ 1.0.

On entry, PARAM($\langle value \rangle$) = $\langle value \rangle$.
Constraint: PARAM($\langle value \rangle$) ≥ 0.0.

IFAIL = 5

On entry, N = $\langle value \rangle$.
Constraint: N $\geq$ 0.

IFAIL = 6

A multiplicative Holt–Winters model cannot be used with the supplied data.

IFAIL = 7

On entry, K = $\langle value \rangle$, $2 \times$ P = $\langle value \rangle$.
Constraint: if MODE = 2 and ITYPE = 4 or 5, $2 \times$ P $\leq$ K.

On entry, K = $\langle value \rangle$ and N = $\langle value \rangle$.
Constraint: if MODE = 2 and ITYPE = 4 or 5, $1 \leq$ K $\leq$ N.

IFAIL = 9

On entry, NF = $\langle value \rangle$.
Constraint: NF $\geq$ 0.

IFAIL = 16

On entry, the array R has not been initialized correctly.

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

Not applicable.

8   Parallelism and Performance

G13AMF is not threaded in any implementation.

9   Further Comments

Single exponential, Brown's double exponential and linear Holt smoothing methods are stable, whereas the two Holt–Winters methods can be affected by poor initial values for the seasonal components.

See also the routine document for G05PMF.

10   Example

This example smooths a time series relating to the rate of the earth's rotation about its polar axis.

10.1 Program Text

```
      Program g13amfe

!     G13AMF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: g13amf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter                 :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)                 :: ad, dv
      Integer                            :: i, ifail, itype, ival, k, mode, n,   &
                                            nf, p
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: fse(:), fv(:), init(:), param(:),      &
                                            r(:), res(:), y(:), yhat(:)
!     .. Executable Statements ..
      Write (nout,*) 'G13AMF Example Program Results'
      Write (nout,*)

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

!     Read in the initial arguments and check array sizes
      Read (nin,*) mode, itype, n, nf

      Allocate (y(n),fv(nf),fse(nf),yhat(n),res(n))

!     Read in data
      Read (nin,*) y(1:n)
```

```
!       Read in the ITYPE dependent arguments (skipping headings)
        Select Case (itype)
        Case (1)
!         Single exponential smoothing
          Allocate (param(1))
          Read (nin,*) param(1)
          p = 0
          ival = 1

        Case (2)
!         Brown double exponential smoothing
          Allocate (param(2))
          Read (nin,*) param(1), param(2)
          p = 0
          ival = 2

        Case (3)
!         Linear Holt smoothing
          Allocate (param(3))
          Read (nin,*) param(1), param(2), param(3)
          p = 0
          ival = 2

        Case Default
!         Additive or multiplicative Holt-Winter smoothing
          Allocate (param(4))
          Read (nin,*) param(1), param(2), param(3), param(4), p
          ival = p + 2
        End Select

        Allocate (init(ival),r(p+13))

!       Read in the MODE dependent arguments (skipping headings)
        Select Case (mode)
        Case (0)
!         User supplied initial values
          Read (nin,*) init(1:ival)
        Case (1)
!         Continuing from a previously saved R
          Read (nin,*) r(1:(p+13))
        Case (2)
!         Initial values calculated from first K observations
          Read (nin,*) k
        End Select

!       Call the library routine
        ifail = 0
        Call g13amf(mode,itype,p,param,n,y,k,init,nf,fv,fse,yhat,res,dv,ad,r,    &
          ifail)

!       Display output
        Write (nout,*) 'Initial values used:'
        Write (nout,99997)(i,init(i),i=1,ival)
        Write (nout,*)
        Write (nout,99999) 'Mean Deviation     = ', dv
        Write (nout,99999) 'Absolute Deviation = ', ad
        Write (nout,*)
        Write (nout,*) '           Observed      1-Step'
        Write (nout,*) ' Period   Values      Forecast      Residual'
        Write (nout,*)
        Write (nout,99998)(i,y(i),yhat(i),res(i),i=1,n)
        Write (nout,*)
        Write (nout,*) '            Forecast      Standard'
        Write (nout,*) ' Period    Values        Errors'
        Write (nout,*)
        Write (nout,99996)(n+i,fv(i),fse(i),i=1,nf)
```

```
99999 Format (A,E12.4)
99998 Format (I4,1X,F12.3,1X,F12.3,1X,F12.3)
99997 Format (I4,1X,F12.3)
99996 Format (I4,1X,F12.3,1X,F12.3)
    End Program g13amfe
```

10.2 Program Data

```
G13AMF Example Program Data
2 3 11 5                                        : MODE,ITYPE,N,NF
180 135 213 181 148 204 228 225 198 200 187 : Y
0.01 1.0 1.0                                    : PARAM(1:3) (since ITYPE=3)
11                                              : K (since MODE=2)
```

10.3 Program Results

```
 G13AMF Example Program Results

 Initial values used:
    1       168.018
    2         3.800

Mean Deviation     =   0.2547E+02
Absolute Deviation =   0.2123E+02

          Observed      1-Step
  Period   Values      Forecast      Residual

    1      180.000     171.818         8.182
    2      135.000     175.782       -40.782
    3      213.000     178.848        34.152
    4      181.000     183.005        -2.005
    5      148.000     186.780       -38.780
    6      204.000     189.800        14.200
    7      228.000     193.492        34.508
    8      225.000     197.732        27.268
    9      198.000     202.172        -4.172
   10      200.000     206.256        -6.256
   11      187.000     210.256       -23.256

          Forecast     Standard
  Period   Values       Errors

   12      213.854       25.473
   13      217.685       25.478
   14      221.516       25.490
   15      225.346       25.510
   16      229.177       25.542
```