NAG Library Routine Document

G13BEF

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

G13BEF fits a multi-input model relating one output series to the input series with a choice of three different estimation criteria: nonlinear least squares, exact likelihood and marginal likelihood. When no input series are present, G13BEF fits a univariate ARIMA model.

2 Specification

```
SUBROUTINE G13BEF (MR, NSER, MT, PARA, NPARA, KFC, NXXY, XXY, LDXXY, KEF,       &
                   NIT, KZSP, ZSP, ITC, SD, CM, LDCM, S, D, NDF, KZEF, RES,     &
                   STTF, ISTTF, NSTTF, WA, IWA, MWA, IMWA, KPRIV, IFAIL)

INTEGER            MR(7), NSER, MT(4,NSER), NPARA, KFC, NXXY, LDXXY, KEF,       &
                   NIT, KZSP, ITC, LDCM, NDF, KZEF, ISTTF, NSTTF, IWA,          &
                   MWA(IMWA), IMWA, KPRIV, IFAIL
REAL (KIND=nag_wp) PARA(NPARA), XXY(LDXXY,NSER), ZSP(4), SD(NPARA),             &
                   CM(LDCM,NPARA), S, D, RES(NXXY), STTF(ISTTF), WA(IWA)
```

3 Description

3.1 The Multi-input Model

The output series y_t, for $t = 1, 2, \ldots, n$, is assumed to be the sum of (unobserved) components $z_{i,t}$ which are due respectively to the inputs $x_{i,t}$, for $i = 1, 2, \ldots, m$.

Thus $y_t = z_{1,t} + \cdots + z_{m,t} + n_t$ where n_t is the error, or output noise component.

A typical component z_t may be either

(a) a simple regression component, $z_t = \omega x_t$ (here x_t is called a simple input), or

(b) a transfer function model component which allows for the effect of lagged values of the variable, related to x_t by

$$z_t = \delta_1 z_{t-1} + \delta_2 z_{t-2} + \cdots + \delta_p z_{t-p} + \omega_0 x_{t-b} - \omega_1 x_{t-b-1} - \cdots - \omega_q x_{t-b-q}.$$

The noise n_t is assumed to follow a (possibly seasonal) ARIMA model, i.e., may be represented in terms of an uncorrelated series, a_t, by the hierarchy of equations

(i) $\nabla^d \nabla_s^D n_t = c + w_t$

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

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

as outlined in Section 3 in G13AEF.

Note: the orders p, q appearing in each of the transfer function models and the ARIMA model are not necessarily the same; $\nabla^d \nabla_s^D n_t$ is the result of applying non-seasonal differencing of order d and seasonal differencing of seasonality s and order D to the series n_t: the differenced series is then of length $N = n - d - s \times D$; the constant term parameter c may optionally be held fixed at its initial value (usually, but not necessarily zero) rather than being estimated.

For the purpose of defining an estimation criterion it is assumed that the series a_t is a sequence of independent Normal variates having mean 0 and variance σ_a^2. An allowance has to be made for the effects of unobserved data prior to the observation period. For the noise component an allowance is always made using a form of backforecasting.

For each transfer function input, you have to decide what values are to be assumed for the pre-period terms $z_0, z_{-1}, \ldots, z_{1-p}$ and $x_0, x_{-1}, \ldots, x_{1-b-q}$ which are in theory necessary to re-create the component series $z_1, z_2, \ldots, z_n$, during the estimation procedure.

The first choice is to assume that all these values are zero. In this case, in order to avoid undesirable transient distortion of the early values $z_1, z_2, \ldots$, you are advised first to correct the input series x_t by subtracting from all the terms a suitable constant to make the early values $x_1, x_2, \ldots$, close to zero. The series mean $\bar{x}$ is one possibility, but for a series with strong trend the constant might be simply x_1.

The second choice is to treat the unknown pre-period terms as nuisance parameters and estimate them along with the other parameters. This choice should be used with caution. For example, if $p = 1$ and $b = q = 0$, it is equivalent to fitting to the data a decaying geometric curve of the form $A\delta^t$, for $t = 1, 2, \ldots$, along with the other inputs, this being the form of the transient. If the output y_t contains a strong trend of this form, which is not otherwise represented in the model, it will have a tendency to influence the estimate of δ away from the value appropriate to the transfer function model.

In most applications the first choice should be adequate, with the option possibly being used as a refinement at the end of the modelling process. The number of nuisance parameters is then $\max(p, b + q)$, with a corresponding loss of degrees of freedom in the residuals. If you align the input x_t with the output by using in its place the shifted series x_{t-b}, then setting $b = 0$ in the transfer function model, there is some improvement in efficiency. On some occasions when the model contains two or more inputs, each with estimation of pre-period nuisance parameters, these parameters may be co-linear and lead to failure of the routine. The option must then be 'switched off' for one or more inputs.

3.2 The Estimation Criterion

This is a measure of how well a proposed set of parameters in the transfer function and noise ARIMA models matches the data. The estimation routine searches for parameter values which minimize this criterion. For a proposed set of parameter values it is derived by calculating

(i) the components $z_{1,t}, z_{2,t}, \ldots, z_{m,t}$ as the responses to the input series $x_{1,t}, x_{2,t} \ldots, x_{m,t}$ using the equations (a) or (b) above,

(ii) the discrepancy between the output and the sum of these components, as the noise

$$n_t = y_t - \left(z_{1,t} + z_{2,t} + \cdots + z_{m,t}\right),$$

(iii) the residual series a_t from n_t by reversing the recursive equations (i), (ii) and (iii) above.

This last step again requires treatment of the effect of unknown pre-period values of n_t and other terms in the equations regenerating a_t. This is identical to the treatment given in Section 3 in G13AEF, and leads to a criterion which is a sum of squares function S, of the residuals a_t. It may be shown that the finite algorithm presented there is equivalent to taking the infinite set of past values $n_0, n_{-1}, n_{-2}, \ldots$, as (linear) nuisance parameters. The pre-period nuisance parameters for the input series are included in the reduction of *df*, as is the constant if it is estimated.

The covariance matrix of the vector of model parameter estimates is given by

$$erv \times H^{-1}$$

where H is the linearized least squares matrix taken from the final iteration of the algorithm of Marquardt. From this expression are derived the vector of standard deviations, and the correlation matrix of parameter estimates. These are approximations which are only valid asymptotically, and must be treated with great caution when the parameter estimates are close to their constraint boundaries.

The residual series a_t is available upon completion of the iterations over the range $t = 1 + d + s \times D, \ldots, n$ corresponding to the differenced noise series w_t.

Because of the algorithm used for backforecasting, these are only true residuals for $t \geq 1 + q + s \times Q - p - s \times P - d - s \times D$, provided this is positive. Estimation of pre-period terms for the inputs will also tend to reduce the magnitude of the early residuals, sometimes severely.

The model component series $z_{1,t}, \ldots, z_{m,t}$ and n_t may optionally be returned in place of the supplied series values, in order to assess the effects of the various inputs on the output.

3.3 Forecasting Information

For the purpose of constructing forecasts of the output series at future time points $t = n+1, n+2, \ldots$ using G13BHF, it is not necessary to use the whole set of observations y_t and $x_{1,t}, x_{2,t}, \ldots, x_{m,t}$, for $t = 1, 2, \ldots, m$. It is sufficient to retain a limited set of quantities constituting the 'state set' as follows: for each series which appears with lagged subscripts in equations (a), (b), (i), (ii) and (iii) above, include the values at times $n+1-k$ for $k = 1$ up to the maximum lag associated with that series in the equations. Note that (i) implicitly includes past values of n_t and intermediate differences of n_t such as $\nabla^{d-1}\nabla_s^D$.

If later observations of the series become available, it is possible to update the state set (without re-estimating the model) using G13BGF. If time series data is supplied with a previously estimated model, it is possible to construct the state set (and forecasts) using G13BJF.

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 Parameters

1: MR(7) – INTEGER array *Input*

On entry: the orders vector (p, d, q, P, D, Q, s) of the ARIMA model for the output noise component.

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: NSER – INTEGER *Input*

On entry: the total number of input and output series. There may be any number of input series (including none), but always one output series.

Constraints:

NSER ≥ 1;
if there are no parameters in the model (that is, $p = q = P = Q = 0$ and KFC $= 0$), NSER > 1.

3: MT(4,NSER) – INTEGER array *Input*

On entry: the transfer function model orders b, p and q of each of the input series. The order parameters for input series i are held in column i. Row 1 holds the value b_i, row 2 holds the value q_i and row 3 holds the value p_i. For a simple input, $b_i = q_i = p_i = 0$.

Row 4 holds the value r_i, where $r_i = 1$ for a simple input, $r_i = 2$ for a transfer function input for which no allowance is to be made for pre-observation period effects, and $r_i = 3$ for a transfer function input for which pre-observation period effects will be treated by estimation of appropriate nuisance parameters.

When $r_i = 1$, any nonzero contents of rows 1, 2, and 3 of column i are ignored.

Constraint: $\mathrm{MT}(4, i) = 1$, 2 or 3, for $i = 1, 2, \ldots, \mathrm{NSER} - 1$.

4: PARA(NPARA) – REAL (KIND=nag_wp) array *Input/Output*

On entry: initial values of the multi-input model parameters. These are in order, firstly the ARIMA model parameters: p values of ϕ parameters, q values of θ parameters, P values of Φ parameters and Q values of Θ parameters. These are followed by initial values of the transfer function model parameters $\omega_0, \omega_1, \ldots, \omega_{q_1}$, $\delta_1, \delta_2, \ldots, \delta_{p_1}$ for the first of any input series and similarly for each subsequent input series. The final component of PARA is the initial value of the constant c, whether it is fixed or is to be estimated.

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

5: NPARA – INTEGER *Input*

On entry: the exact number of $\phi, \theta, \Phi, \Theta$, ω, δ and c parameters.

Constraint: $\mathrm{NPARA} = p + q + P + Q + \mathrm{NSER} + \sum(p_i + q_i)$, the summation being over all the $p_i \neq q_i$ supplied in MT. c must be included, whether fixed or estimated.

6: KFC – INTEGER *Input*

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

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

7: NXXY – INTEGER *Input*

On entry: the (common) length of the original, undifferenced input and output time series.

8: XXY(LDXXY,NSER) – REAL (KIND=nag_wp) array *Input/Output*

On entry: the columns of XXY must contain the NXXY original, undifferenced values of each of the input series and the output series x_t in that order.

On exit: if $\mathrm{KZEF} = 0$, XXY remains unchanged on exit.

If $\mathrm{KZEF} \neq 0$, the columns of XXY hold the corresponding values of the input component series z_t in place of x_t and the output noise component n_t in place of y_t, in that order.

9: LDXXY – INTEGER *Input*

On entry: the first dimension of the array XXY as declared in the (sub)program from which G13BEF is called.

Constraint: $\mathrm{LDXXY} \geq \mathrm{NXXY}$.

10: KEF – INTEGER *Input*

On entry: indicates the likelihood option.

$\mathrm{KEF} = 1$
: Gives least squares.

$\mathrm{KEF} = 2$
: Gives exact likelihood.

$\mathrm{KEF} = 3$
: Gives marginal likelihood.

Constraint: $\mathrm{KEF} = 1$, 2 or 3.

11: NIT – INTEGER *Input*

On entry: the maximum required number of iterations.

NIT = 0
: No change is made to any of the model parameters in array PARA except that the constant c (if KFC = 1) and any ω relating to simple input series are estimated. (Apart from these, estimates are always derived for the nuisance parameters relating to any backforecasts and any pre-observation period effects for transfer function inputs.)

Constraint: NIT ≥ 0.

12: KZSP – INTEGER *Input*

On entry: must be set to 1 if the routine is to use the input values of ZSP in the minimization procedure, and to any other value if the default values of ZSP are to be used.

13: ZSP(4) – REAL (KIND=nag_wp) array *Input/Output*

On entry: if KZSP = 1, then ZSP must contain the four values used to control the strategy of the search procedure.

ZSP(1)
: Contains α, the value used to constrain the magnitude of the search procedure steps.

ZSP(2)
: Contains β, the multiplier which regulates the value of α.

ZSP(3)
: Contains δ, the value of the stationarity and invertibility test tolerance factor.

ZSP(4)
: Contains γ, the value of the convergence criterion.

If KZSP $\neq 1$ before entry, default values of ZSP are supplied by the routine. These are 0.01, 10.0, 1000.0 and $\max(100 \times$ ***machine precision***$, 0.0000001)$, respectively.

On exit: contains the values, default or otherwise, used by the routine.

Constraint: if KZSP = 1, ZSP(1) > 0.0, ZSP(2) > 1.0, ZSP(3) ≥ 1.0, $0 \leq$ ZSP(4) < 1.0.

14: ITC – INTEGER *Output*

On exit: the number of iterations carried out.

ITC $= -1$
: Indicates that the only estimates obtained up to this point have been for the nuisance parameters relating to backforecasts, unless the marginal likelihood option is used, in which case estimates have also been obtained for simple input coefficients ω and for the constant c (if KFC = 1). This value of ITC usually indicates a failure in a consequent step of estimating transfer function input pre-observation period nuisance parameters.

ITC = 0
: Indicates that estimates have been obtained up to this point for the constant c (if KFC = 1), for simple input coefficients ω and for the nuisance parameters relating to the backforecasts and to transfer function input pre-observation period effects.

15: SD(NPARA) – REAL (KIND=nag_wp) array *Output*

On exit: the NPARA values of the standard deviations corresponding to each of the parameters in PARA. When the constant is fixed its standard deviation is returned as zero. When the values of PARA are valid, the values of SD are usually also valid. However, if an exit value of IFAIL = 3, 8 or 10, then the contents of SD will be indeterminate.

16: CM(LDCM,NPARA) – REAL (KIND=nag_wp) array *Output*

On exit: the first NPARA rows and columns of CM contain the correlation coefficients relating to each pair of parameters in PARA. All coefficients relating to the constant will be zero if the constant is fixed. The contents of CM will be indeterminate under the same conditions as SD.

17: LDCM – INTEGER *Input*

On entry: the first dimension of the array CM as declared in the (sub)program from which G13BEF is called.

Constraint: LDCM $\geq$ NPARA.

18: S – REAL (KIND=nag_wp) *Output*

On exit: the residual sum of squares, S, at the latest set of valid parameter estimates.

19: D – REAL (KIND=nag_wp) *Output*

On exit: the objective function, D, at the latest set of valid parameter estimates.

20: NDF – INTEGER *Output*

On exit: the number of degrees of freedom associated with S.

21: KZEF – INTEGER *Input*

On entry: must not be set to 0, if the values of the input component series z_t and the values of the output noise component n_t are to overwrite the contents of XXY on exit, and must be set to 0 if XXY is to remain unchanged.

22: RES(NXXY) – REAL (KIND=nag_wp) array *Output*

On exit: the values of the residuals relating to the differenced values of the output series. The remainder of the first NXXY terms in the array will be zero.

23: STTF(ISTTF) – REAL (KIND=nag_wp) array *Output*

On exit: the NSTTF values of the state set array.

24: ISTTF – INTEGER *Input*

On entry: the dimension of the array STTF as declared in the (sub)program from which G13BEF is called.

Constraint: ISTTF $\geq (P \times s) + d + (D \times s) + q + \max(p, Q \times s) + ncg$, where $ncg = \sum(b_i + q_i + p_i)$ over all input series for which $r_i > 1$.

25: NSTTF – INTEGER *Output*

On exit: the number of values in the state set array STTF.

26: WA(IWA) – REAL (KIND=nag_wp) array *Workspace*
27: IWA – INTEGER *Input*

On entry: the dimension of the array WA as declared in the (sub)program from which G13BEF is called.

It is not practical to outline a method for deriving the exact minimum permissible value of IWA, but the following gives a reasonably good conservative approximation. (It should be noted that if IWA is too small (but not grossly so) then the exact minimum is returned in MWA(i) and is also printed if KPRIV $\neq$ 0.)

Let $q' = q + (Q \times s)$ and $d' = d + (D \times s)$ where the orders of the output noise model are p, d, q, P, D, Q, s.

Let there be l input series, where $l = \text{NSER} - 1$.

Let

$$mx_i = \max(b_i + q_i, p_i), \quad \text{if } r_i = 3, \text{ for } i = 1, 2 \ldots, l$$
$$mx_i = 0, \quad \text{if } r_i \neq 3, \text{ for } i = 1, 2 \ldots, l$$

where the transfer function model orders for input i are given by b_i, q_i, p_i, r_i.

Let $qx = \max(q', mx_1, mx_2, \ldots, mx_l)$.

Let $ncd = \text{NPARA} + \text{KFC} + qx + \sum_{i=1}^{l} mx_i$ and $nce = \text{NXXY} + d' + 6 \times qx$.

Finally, let $ncf = \text{NSER}$, and then increment ncf by 1 every time any of the following conditions is satisfied. (The last six conditions should be applied separately to each input series, so that, for example, if we have two input series and if $p_1 > 0$ and $p_2 > 0$ then ncf is incremented by 2.)

The conditions are:

$$p > 0$$
$$q > 0$$
$$P > 0$$
$$Q > 0$$
$$qx > 0$$
$$\text{KFC} > 0$$

$$\left.\begin{array}{r} p > 0 \\ q > 0 \\ P > 0 \\ Q > 0 \end{array}\right\} \text{ and } q > 0 \text{ and KEF} > 1.$$

$$\left.\begin{array}{r} p > 0 \\ q > 0 \\ P > 0 \\ Q > 0 \end{array}\right\} \text{ and KFC} > 0 \text{ and KEF} = 3.$$

$$\left.\begin{array}{r} mx_i > 0 \\ p_i > 0 \\ p > 0 \\ q > 0 \\ P > 0 \\ Q > 0 \end{array}\right\} \text{ and } r_i = 1 \text{ and KEF} > 3 \text{ separately, for } i = 1, 2, \ldots, l.$$

Then $\text{IWA} \geq 2 \times (ncd)^2 + (nce) \times (ncf + 4)$.

28: MWA(IMWA) – INTEGER array *Workspace*

29: IMWA – INTEGER *Input*

On entry: the dimension of the array MWA as declared in the (sub)program from which G13BEF is called.

Constraint: $\text{IMWA} \geq (16 \times \text{NSER}) + (7 \times ncd) + (3 \times \text{NPARA}) + (3 \times \text{KFC}) + 27$, where the derivation of ncd is shown under IWA.

If IMWA is too small then the exact minimum needed is returned in IMWA and if $\text{KPRIV} \neq 0$ it is also printed.

30: KPRIV – INTEGER *Input*

On entry: must not be set to 0, if it is required to monitor the course of the optimization or to print out the requisite minimum values of IWA or IMWA in the event of an error of the type IFAIL = 6 or 7. The course of the optimization is monitored by printing out at each iteration the iteration count (ITC), the residual sum of squares (S), the objective function (D) and a description and value for each of the parameters in the PARA array. The descriptions are PHI for ϕ, THETA for θ, SPHI for Φ, STHETA for Θ, OMEGA/SI for ω in a simple input, OMEGA for ω in a transfer function input, DELTA for δ and CONSTANT for c. In addition SERIES 1, SERIES 2, etc. indicate the input series relevant to the OMEGA and DELTA parameters.

KPRIV must be set to 0 if the print-out of the above information is not required.

31: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction 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 parameters may be useful even if 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: G13BEF 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, KFC < 0,
or KFC > 1,
or LDXXY $<$ NXXY,
or LDCM $<$ NPARA,
or KEF < 1,
or KEF > 3,
or NIT < 0,
or NSER < 1,
or NSER $= 1$ and there are no parameters in the model ($p = q = P = Q = 0$ and KFC $= 0$).

IFAIL $= 2$

On entry, there is inconsistency between NPARA and KFC on the one hand and the orders in arrays MR and MT on the other, or one of the r_i, stored in MT$(4, i) \neq 1$, 2 or 3.

IFAIL $= 3$

On entry or during execution, one or more sets of δ parameters do not satisfy the stationarity or invertibility test conditions.

IFAIL $= 4$

On entry, when KZSP $= 1$, ZSP$(1) \leq 0.0$,
or ZSP$(2) \leq 1.0$,
or ZSP$(3) < 1.0$,
or ZSP$(4) < 0.0$,
or ZSP$(4) \geq 1.0$.

IFAIL = 5

On entry, IWA is too small by a considerable margin. No information is supplied about the requisite minimum size.

IFAIL = 6

On entry, IWA is too small, but the requisite minimum size is returned in MWA(1), which is printed if KPRIV $\neq 0$.

IFAIL = 7

On entry, IMWA is too small, but the requisite minimum size is returned in MWA(1), which is printed if KPRIV $\neq 0$.

IFAIL = 8

This indicates a failure in F04ASF which is used to solve the equations giving the latest estimates of the parameters.

IFAIL = 9

This indicates a failure in the inversion of the second derivative matrix. This is needed in the calculation of the correlation matrix and the standard deviations of the parameter estimates.

IFAIL = 10

On entry or during execution, one or more sets of the ARIMA (ϕ, θ, Φ or Θ) parameters do not satisfy the stationarity or invertibility test conditions.

IFAIL = 11

On entry, ISTTF is too small. The state set information will not be produced and if KZEF $\neq 0$ array XXY will remain unchanged. All other parameters will be produced correctly.

IFAIL = 12

The routine has failed to converge after NIT iterations. If steady decreases in the objective function, D, were monitored up to the point where this exit occurred, then the exit probably occurred because NIT was set too small, so the calculations should be restarted from the final point held in PARA.

IFAIL = 13

On entry, ISTTF is too small (see IFAIL = 11) and NIT iterations were carried out without the convergence conditions being satisfied (see IFAIL = 12).

7 Accuracy

The computation used is believed to be stable.

8 Further Comments

The time taken by G13BEF is approximately proportional to NXXY $\times$ ITC $\times$ NPARA^2.

9 Example

After the full 11 iterations, the following are computed and printed out: the final values of the PARA parameters and their standard errors, the correlation matrix, the residuals for the 36 differenced values, the values of z_t and n_t, the values of the state set and the number of degrees of freedom.

9.1 Program Text

```
    Program g13befe

!     G13BEF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: g13bef, nag_wp, x04abf, x04caf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter              :: iset = 1, nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)              :: d, s
      Integer                         :: dp, i, ifail, imwa, inc, isttf, itc, &
                                         iwa, kef, kfc, kpriv, kzef, kzsp,    &
                                         ldcm, ldxxy, mx, nadv, ncd, nce,     &
                                         ncf, ncg, ndf, ndv, nis, nit, npara, &
                                         nser, nsttf, nxxy, qp, qx, smx
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable :: cm(:,:), para(:), res(:), sd(:),     &
                                         sttf(:), wa(:), xxy(:,:)
      Real (Kind=nag_wp)              :: zsp(4)
      Integer                         :: mr(7)
      Integer, Allocatable            :: mt(:,:), mwa(:)
!     .. Intrinsic Procedures ..
      Intrinsic                       :: max, sum
!     .. Executable Statements ..
      Write (nout,*) 'G13BEF Example Program Results'
      Write (nout,*)

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

!     Read in problem size
      Read (nin,*) kzef, kfc, nxxy, nser, kef, nit, kzsp, kpriv
      If (kzsp/=0) Then
        Read (nin,*) zsp
      End If

!     Number of input series
      nis = nser - 1

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

      Allocate (mt(4,nser))

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

!     Read in transfer function
      Do i = 1, nis
        Read (nin,*) mt(1:4,i)
      End Do

!     Calculate NPARA and various other quantities required
!     for calculate array sizes
      npara = 0
      ncg = 0
      qx = 0
      smx = 0
      ncf = nser
      inc = 1
      Do i = 1, nis
        npara = npara + mt(2,i) + mt(3,i)
```

```
          If (mt(4,i)>1) Then
            ncg = ncg + sum(mt(1:3,i))
            If (mt(4,i)==3) Then
              mx = max(mt(1,i)+mt(2,i),mt(3,i))
              qx = max(qx,mx)
              smx = smx + mx
            End If
          Else If (mt(4,i)==1 .And. kef==3) Then
            If (mt(3,i)>0) Then
              ncf = ncf + 1
            End If
            inc = inc + 1
          End If
        End Do
        npara = npara + mr(1) + mr(3) + mr(4) + mr(6) + nser

!       Calculate size of arrays
        isttf = mr(4)*mr(7) + mr(2) + mr(5)*mr(7) + mr(3) + &
          max(mr(1),mr(6)*mr(7)) + ncg
        ldxxy = nxxy
        ldcm = npara
        qp = mr(3) + mr(6)*mr(7)
        dp = mr(2) + mr(5)*mr(7)
        If (mr(3)>0 .And. kef>1) Then
          inc = inc + 1
        End If
        If (kfc>0 .And. kef==3) Then
          inc = inc + 1
        End If
        qx = qp
        ncd = npara + kfc + smx
        If (mr(1)>0) Then
          ncf = ncf + inc
        End If
        If (mr(3)>0) Then
          ncf = ncf + inc
        End If
        If (mr(4)>0) Then
          ncf = ncf + inc
        End If
        If (mr(6)>0) Then
          ncf = ncf + inc
        End If
        If (qx>0) Then
          ncf = ncf + 1
        End If
        If (kfc>0) Then
          ncf = ncf + 1
        End If
        ncd = ncd + qx
        nce = nxxy + dp + 6*qx
        iwa = 2*ncd**2 + nce*(ncf+4)
        iwa = 2*iwa
        imwa = 16*nser + 7*ncd + 3*npara + 3*kfc + 27

        Allocate (xxy(ldxxy,nser),para(npara),sd(npara),cm(ldcm,npara), &
          res(nxxy),sttf(isttf),wa(iwa),mwa(imwa))

!       Read in rest of data
        Read (nin,*) para(1:npara)
        Read (nin,*)(xxy(i,1:nser),i=1,nxxy)

        ifail = -1
        Call g13bef(mr,nser,mt,para,npara,kfc,nxxy,xxy,ldxxy,kef,nit,kzsp,zsp, &
          itc,sd,cm,ldcm,s,d,ndf,kzef,res,sttf,isttf,nsttf,wa,iwa,mwa,imwa, &
          kpriv,ifail)
        If (ifail/=0) Then
          If (ifail/=8 .And. ifail/=9) Then
            Go To 100
          End If
        End If
```

```
!     Display results
      Write (nout,99999) 'The number of iterations carried out is', itc
      Write (nout,*)
      Write (nout,*) &
        'Final values of the parameters and their standard deviations'
      Write (nout,*)
      Write (nout,*) '   I            PARA(I)                 SD'
      Write (nout,*)
      Write (nout,99998)(i,para(i),sd(i),i=1,npara)
      Write (nout,*)
      Flush (nout)
      ifail = 0
      Call x04caf('General',' ',npara,npara,cm,ldcm, &
        'The correlation matrix is',ifail)
      Write (nout,*)
      Write (nout,*) 'The residuals and the z and n values are'
      Write (nout,*)
      Write (nout,*) '   I         RES(I)          z(t)           n(t)'
      Write (nout,*)
      ndv = nxxy - mr(2) - mr(5)*mr(7)
      Do i = 1, nxxy
        If (i<=ndv) Then
          Write (nout,99997) i, res(i), xxy(i,1:nser)
        Else
          Write (nout,99996) i, xxy(i,1:nser)
        End If
      End Do
      If (mr(2)/=0 .Or. mr(5)/=0) Then
        Write (nout,*)
        Write (nout,*) &
          '** Note that the residuals relate to differenced values **'
      End If
      Write (nout,*)
      Write (nout,99995) 'The state set consists of', nsttf, ' values'
      Write (nout,*)
      Write (nout,99994) sttf(1:nsttf)
      Write (nout,*)
      Write (nout,99999) 'The number of degrees of freedom is', ndf

100   Continue

99999 Format (1X,A,I4)
99998 Format (1X,I4,2F20.6)
99997 Format (1X,I4,3F15.3)
99996 Format (1X,I4,F30.3,F15.3)
99995 Format (1X,A,I4,A)
99994 Format (1X,6F10.4)
    End Program g13befe
```

9.2 Program Data

```
G13BEF Example Program Data
1  1 40  2  3 20  0  0   :: KZEF,KFC,NXXY,NSER,KEFF,NIT,KZSP,KPRIV
1  0  0  0  0  1  4      :: MR
1 0 1 3                  :: Transfer fun. for series 1, MT(:,1)
0.0  0.0  2.0  0.5  0.0  :: PARA
 8.075     105.0
 7.819     119.0
 7.366     119.0
 8.113     109.0
 7.380     117.0
 7.134     135.0
 7.222     126.0
 7.768     112.0
 7.386     116.0
 6.965     122.0
 6.478     115.0
 8.105     115.0
 8.060     122.0
```

```
7.684     138.0
7.580     135.0
7.093     125.0
6.129     115.0
6.026     108.0
6.679     100.0
7.414      96.0
7.112     107.0
7.762     115.0
7.645     123.0
8.639     122.0
7.667     128.0
8.080     136.0
6.678     140.0
6.739     122.0
5.569     102.0
5.049     103.0
5.642      89.0
6.808      77.0
6.636      89.0
8.241      94.0
7.968     104.0
8.044     108.0
7.791     119.0
7.024     126.0
6.102     119.0
6.053     103.0        :: End of XXY
```

9.3 Program Results

```
G13BEF Example Program Results

The number of iterations carried out is  11

Final values of the parameters and their standard deviations

   I            PARA(I)                SD

   1           0.380924            0.166379
   2          -0.257786            0.178178
   3           8.956084            0.948061
   4           0.659641            0.060239
   5         -75.435521           33.505341

The correlation matrix is
          1       2       3       4       5
1   1.0000 -0.1839 -0.1775 -0.0340  0.1394
2  -0.1839  1.0000  0.0518  0.2547 -0.2860
3  -0.1775  0.0518  1.0000 -0.3070 -0.2926
4  -0.0340  0.2547 -0.3070  1.0000 -0.8185
5   0.1394 -0.2860 -0.2926 -0.8185  1.0000

The residuals and the z and n values are

   I         RES(I)          z(t)           n(t)

   1          0.397       180.567        -75.567
   2          3.086       191.430        -72.430
   3         -2.818       196.302        -77.302
   4         -9.941       195.460        -86.460
   5         -5.061       201.594        -84.594
   6         14.053       199.076        -64.076
   7          2.624       195.211        -69.211
   8         -5.823       193.450        -81.450
   9         -2.147       197.179        -81.179
  10         -0.216       196.217        -74.217
  11         -2.517       191.812        -76.812
  12          7.916       184.544        -69.544
  13          1.423       194.322        -72.322
  14         11.936       200.369        -62.369
```

```
  15         5.117        200.990        -65.990
  16        -5.672        200.468        -75.468
  17        -5.681        195.763        -80.763
  18        -1.637        184.025        -76.025
  19        -1.019        175.360        -75.360
  20        -2.623        175.492        -79.492
  21         3.283        182.162        -75.162
  22         6.896        183.857        -68.857
  23         5.395        190.797        -67.797
  24         0.875        194.327        -72.327
  25        -4.153        205.558        -77.558
  26         6.206        204.261        -68.261
  27         4.208        207.104        -67.104
  28        -2.387        196.423        -74.423
  29       -11.803        189.924        -87.924
  30         6.435        175.158        -72.158
  31         1.342        160.761        -71.761
  32        -4.924        156.575        -79.575
  33         4.799        164.256        -75.256
  34        -0.074        167.783        -73.783
  35        -6.023        184.483        -80.483
  36        -6.427        193.055        -85.055
  37        -2.527        199.390        -80.390
  38         2.039        201.302        -75.302
  39         0.243        195.695        -76.695
  40        -3.166        183.738        -80.738

 The state set consists of   6 values

     6.0530  183.7384   -5.7855   -0.1645    0.1800   -3.0977

 The number of degrees of freedom is  34
```