NAG Library Routine Document

E04YCF

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

E04YCF returns estimates of elements of the variance-covariance matrix of the estimated regression coefficients for a nonlinear least squares problem. The estimates are derived from the Jacobian of the function $f(x)$ at the solution.

This routine may be used following any one of the nonlinear least squares routines E04FCF, E04FYF, E04GBF, E04GDF, E04GYF, E04GZF, E04HEF or E04HYF.

2 Specification

```
SUBROUTINE E04YCF (JOB, M, N, FSUMSQ, S, V, LDV, CJ, WORK, IFAIL)

INTEGER              JOB, M, N, LDV, IFAIL
REAL (KIND=nag_wp) FSUMSQ, S(N), V(LDV,N), CJ(N), WORK(N)
```

3 Description

E04YCF is intended for use when the nonlinear least squares function, $F(x) = f^{\mathrm{T}}(x)f(x)$, represents the goodness-of-fit of a nonlinear model to observed data. The routine assumes that the Hessian of $F(x)$, at the solution, can be adequately approximated by $2J^{\mathrm{T}}J$, where J is the Jacobian of $f(x)$ at the solution. The estimated variance-covariance matrix C is then given by

$$C = \sigma^2 \left(J^{\mathrm{T}}J\right)^{-1}, \qquad J^{\mathrm{T}}J \qquad \text{nonsingular},$$

where σ^2 is the estimated variance of the residual at the solution, $\bar{x}$, given by

$$\sigma^2 = \frac{F(\bar{x})}{m-n},$$

m being the number of observations and n the number of variables.

The diagonal elements of C are estimates of the variances of the estimated regression coefficients. See the E04 Chapter Introduction, Bard (1974) and Wolberg (1967) for further information on the use of C.

When $J^{\mathrm{T}}J$ is singular then C is taken to be

$$C = \sigma^2 \left(J^{\mathrm{T}}J\right)^{\dagger},$$

where $\left(J^{\mathrm{T}}J\right)^{\dagger}$ is the pseudo-inverse of $J^{\mathrm{T}}J$, and

$$\sigma^2 = \frac{F(\bar{x})}{m-k}, \qquad k = \text{rank }(J)$$

but in this case the parameter IFAIL is returned as nonzero as a warning to you that J has linear dependencies in its columns. The assumed rank of J can be obtained from IFAIL.

The routine can be used to find either the diagonal elements of C, or the elements of the jth column of C, or the whole of C.

E04YCF must be preceded by one of the nonlinear least squares routines mentioned in Section 1, and requires the parameters FSUMSQ, S and V to be supplied by those routines (e.g., see E04FCF). FSUMSQ is the residual sum of squares $F(\bar{x})$ and S and V contain the singular values and right singular vectors respectively in the singular value decomposition of J. S and V are returned directly by the comprehensive routines E04FCF, E04GBF, E04GDF and E04HEF, but are returned as part of the workspace parameter W (from one of the easy-to-use routines). In the case of E04FYF, S starts at W(NS), where

$$NS = 6 \times \mathrm{N} + 2 \times \mathrm{M} + \mathrm{M} \times \mathrm{N} + 1 + \max(1, \mathrm{N} \times (\mathrm{N} - 1)/2)$$

and in the cases of the remaining easy-to-use routines, S starts at W(NS), where

$$NS = 7 \times \mathrm{N} + 2 \times \mathrm{M} + \mathrm{M} \times \mathrm{N} + \mathrm{N} \times (\mathrm{N} + 1)/2 + 1 + \max(1, \mathrm{N} \times (\mathrm{N} - 1)/2).$$

The parameter V starts immediately following the elements of S, so that V starts at W(NV), where

$$NV = NS + \mathrm{N}.$$

For all the easy-to-use routines the parameter LDV must be supplied as N. Thus a call to E04YCF following E04FYF can be illustrated as

```
.
.
.
CALL E04FYF (M, N, LFUN1, X, FSUMSQ, W, LW, IUSER, RUSER, IFAIL)
.
.
.
NS = 6*N _ 2*M + M*N + 1 MAX((1,(N*(N-1))/2)
NV = NS + N;
CALL E04YCF (JOB, M, N, FSUMSQ, W(NS), W(NV), N, CJ, WORK, IFAIL)
```

where the parameters M, N, FSUMSQ and the $(n + n^2)$ elements W(NS), W($NS + 1$), ..., W($NV + \mathrm{N}^2 - 1$) must not be altered between the calls to E04FYF and E04YCF. The above illustration also holds for a call to E04YCF following a call to one of E04GYF, E04GZF or E04HYF, except that NS must be computed as

$$NS = 7 \times \mathrm{N} + 2 \times \mathrm{M} + \mathrm{M} \times \mathrm{N} + (\mathrm{N} \times (\mathrm{N} + 1))/2 + 1 + \max((1, \mathrm{N} \times (\mathrm{N} - 1))/2).$$

4 References

Bard Y (1974) *Nonlinear Parameter Estimation* Academic Press

Wolberg J R (1967) *Prediction Analysis* Van Nostrand

5 Parameters

1: JOB – INTEGER *Input*

On entry: which elements of C are returned as follows:

JOB $= -1$
The n by n symmetric matrix C is returned.

JOB $= 0$
The diagonal elements of C are returned.

JOB > 0
The elements of column JOB of C are returned.

Constraint: $-1 \leq \mathrm{JOB} \leq \mathrm{N}$.

2: M – INTEGER *Input*

On entry: the number m of observations (residuals $f_i(x)$).

Constraint: $\mathrm{M} \geq \mathrm{N}$.

3: N – INTEGER *Input*

On entry: the number n of variables (x_j).

Constraint: $1 \leq \mathrm{N} \leq \mathrm{M}$.

4: FSUMSQ – REAL (KIND=nag_wp) *Input*

On entry: the sum of squares of the residuals, $F(\bar{x})$, at the solution $\bar{x}$, as returned by the nonlinear least squares routine.

Constraint: FSUMSQ ≥ 0.0.

5: S(N) – REAL (KIND=nag_wp) array *Input*

On entry: the n singular values of the Jacobian as returned by the nonlinear least squares routine. See Section 3 for information on supplying S following one of the easy-to-use routines.

6: V(LDV,N) – REAL (KIND=nag_wp) array *Input/Output*

On entry: the n by n right-hand orthogonal matrix (the right singular vectors) of J as returned by the nonlinear least squares routine. See Section 3 for information on supplying V following one of the easy-to-use routines.

On exit: if JOB ≥ 0, V is unchanged.

If JOB $= -1$, the leading n by n part of V is overwritten by the n by n matrix C. When E04YCF is called with JOB $= -1$ following an easy-to-use routine this means that C is returned, column by column, in the n^2 elements of W given by $\mathrm{W}(NV), \mathrm{W}(NV+1), \ldots, \mathrm{W}(NV + \mathrm{N}^2 - 1)$. (See Section 3 for the definition of NV.)

7: LDV – INTEGER *Input*

On entry: the first dimension of the array V as declared in the (sub)program from which E04YCF is called. When V is passed in the workspace parameter W (following one of the easy-to-use least square routines), LDV must be the value N.

Constraint: if JOB $= -1$, LDV $\geq$ N.

8: CJ(N) – REAL (KIND=nag_wp) array *Output*

On exit: if JOB $= 0$, CJ returns the n diagonal elements of C.

If JOB $= j > 0$, CJ returns the n elements of the jth column of C.

If JOB $= -1$, CJ is not referenced.

9: WORK(N) – REAL (KIND=nag_wp) array *Workspace*

If JOB $= -1$ or 0, WORK is used as internal workspace.

If JOB > 0, WORK is not referenced.

10: 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: E04YCF 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, JOB < -1,
or JOB $>$ N,
or N < 1,
or M $<$ N,
or FSUMSQ < 0.0,
or LDV $<$ N.

IFAIL $= 2$

The singular values are all zero, so that at the solution the Jacobian matrix J has rank 0.

IFAIL > 2

At the solution the Jacobian matrix contains linear, or near linear, dependencies amongst its columns. In this case the required elements of C have still been computed based upon J having an assumed rank given by IFAIL $- 2$. The rank is computed by regarding singular values $SV(j)$ that are not larger than $10\epsilon \times SV(1)$ as zero, where ϵ is the ***machine precision*** (see X02AJF). If you expect near linear dependencies at the solution and are happy with this tolerance in determining rank you should call E04YCF with IFAIL $= 1$ in order to prevent termination (see the description of IFAIL). It is then essential to test the value of IFAIL on exit from E04YCF.

Overflow

If overflow occurs then either an element of C is very large, or the singular values or singular vectors have been incorrectly supplied.

7 Accuracy

The computed elements of C will be the exact covariances corresponding to a closely neighbouring Jacobian matrix J.

8 Further Comments

When JOB $= -1$ the time taken by E04YCF is approximately proportional to n^3. When JOB ≥ 0 the time taken by the routine is approximately proportional to n^2.

9 Example

This example estimates the variance-covariance matrix C for the least squares estimates of x_1, x_2 and x_3 in the model

$$y = x_1 + \frac{t_1}{x_2 t_2 + x_3 t_3}$$

using the 15 sets of data given in the following table:

y	t_1	t_2	t_3
0.14	1.0	15.0	1.0
0.18	2.0	14.0	2.0
0.22	3.0	13.0	3.0
0.25	4.0	12.0	4.0
0.29	5.0	11.0	5.0
0.32	6.0	10.0	6.0
0.35	7.0	9.0	7.0
0.39	8.0	8.0	8.0
0.37	9.0	7.0	7.0
0.58	10.0	6.0	6.0
0.73	11.0	5.0	5.0
0.96	12.0	4.0	4.0
1.34	13.0	3.0	3.0
2.10	14.0	2.0	2.0
4.39	15.0	1.0	1.0

The program uses $(0.5, 1.0, 1.5)$ as the initial guess at the position of the minimum and computes the least squares solution using E04FYF. See the routine document E04FYF for further information.

9.1 Program Text

```
!   E04YCF Example Program Text
!   Mark 24 Release. NAG Copyright 2012.

    Module e04ycfe_mod

!     E04YCF Example Program Module:
!            Parameters and User-defined Routines

!     .. Use Statements ..
      Use nag_library, Only: nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter                   :: mdec = 15, ndec = 3, nin = 5,    &
                                              nout = 6
      Integer, Parameter                   ::                                  &
             lwork = 7*ndec + ndec*ndec + 2*mdec*ndec + 3*mdec+ ndec*(ndec-1)/2
!     .. Local Arrays ..
      Real (Kind=nag_wp)                   :: t(mdec,ndec), y(mdec)
    Contains
      Subroutine lsfun1(m,n,xc,fvec,iuser,ruser)
!       Routine to evaluate the residuals

!       .. Scalar Arguments ..
        Integer, Intent (In)                 :: m, n
!       .. Array Arguments ..
        Real (Kind=nag_wp), Intent (Out)     :: fvec(m)
        Real (Kind=nag_wp), Intent (Inout)   :: ruser(*)
        Real (Kind=nag_wp), Intent (In)      :: xc(n)
        Integer, Intent (Inout)              :: iuser(*)
!       .. Executable Statements ..
        fvec(1:m) = xc(1) + t(1:m,1)/(xc(2)*t(1:m,2)+xc(3)*t(1:m,3)) - y(1:m)

        Return

      End Subroutine lsfun1
    End Module e04ycfe_mod
    Program e04ycfe

!     E04YCF Example Main Program

!     .. Use Statements ..
```

```
      Use nag_library, Only: e04fyf, e04ycf, nag_wp
      Use e04ycfe_mod, Only: lsfun1, lwork, mdec, ndec, nin, nout, t, y
!     .. Implicit None Statement ..
      Implicit None
!     .. Local Scalars ..
      Real (Kind=nag_wp)                   :: fsumsq
      Integer                              :: i, ifail, job, ldv, m, n, ns, nv
!     .. Local Arrays ..
      Real (Kind=nag_wp)                   :: cj(ndec), ruser(1), work(lwork), &
                                              x(ndec)
      Integer                              :: iuser(1)
!     .. Intrinsic Procedures ..
      Intrinsic                            :: max
!     .. Executable Statements ..
      Write (nout,*) 'E04YCF Example Program Results'

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

      m = mdec
      n = ndec

!     Observations of TJ (J = 1, 2, ..., n) are held in T(I, J)
!     (I = 1, 2, ..., m)

      Do i = 1, m
        Read (nin,*) y(i), t(i,1:n)
      End Do

      x(1:n) = (/0.5E0_nag_wp,1.0E0_nag_wp,1.5E0_nag_wp/)

      ifail = -1
      Call e04fyf(m,n,lsfun1,x,fsumsq,work,lwork,iuser,ruser,ifail)

      Select Case (ifail)
      Case (0,2:)
        Write (nout,*)
        Write (nout,99999) 'On exit, the sum of squares is', fsumsq
        Write (nout,*) 'at the point'
        Write (nout,99998) x(1:n)

!       Compute estimates of the variances of the sample regression
!       coefficients at the final point.
!       Since NS is greater than N we can use the first N elements
!       of the array WORK for the dummy argument WORK.

        ns = 6*n + 2*m + m*n + 1 + max(1,(n*(n-1))/2)
        nv = ns + n
        job = 0
        ldv = n

        ifail = -1
        Call e04ycf(job,m,n,fsumsq,work(ns),work(nv),ldv,cj,work,ifail)

        Select Case (ifail)
        Case (0,3:)
          Write (nout,*)
          Write (nout,*) 'and estimates of the variances of the sample'
          Write (nout,*) 'regression coefficients are'
          Write (nout,99998) cj(1:n)
        End Select

      End Select

99999 Format (1X,A,F12.4)
99998 Format (1X,3F12.4)
    End Program e04ycfe
```

9.2 Program Data

```
E04YCF Example Program Data
 0.14  1.0 15.0  1.0
 0.18  2.0 14.0  2.0
 0.22  3.0 13.0  3.0
 0.25  4.0 12.0  4.0
 0.29  5.0 11.0  5.0
 0.32  6.0 10.0  6.0
 0.35  7.0  9.0  7.0
 0.39  8.0  8.0  8.0
 0.37  9.0  7.0  7.0
 0.58 10.0  6.0  6.0
 0.73 11.0  5.0  5.0
 0.96 12.0  4.0  4.0
 1.34 13.0  3.0  3.0
 2.10 14.0  2.0  2.0
 4.39 15.0  1.0  1.0
```

9.3 Program Results

```
 E04YCF Example Program Results

 On exit, the sum of squares is      0.0082
 at the point
       0.0824      1.1330      2.3437

 and estimates of the variances of the sample
 regression coefficients are
       0.0002      0.0948      0.0878
```