NAG Fortran Library Routine Document

G03EAF

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

G03EAF computes a distance (dissimilarity) matrix.

2 Specification

```
SUBROUTINE G03EAF(UPDATE, DIST, SCALE, N, M, X, LDX, ISX, S, D, IFAIL)
INTEGER           N, M, LDX, ISX(M), IFAIL
real              X(LDX,M), S(M), D(N*(N-1)/2)
CHARACTER*1       UPDATE, DIST, SCALE
```

3 Description

Given n objects, a distance or dissimilarity matrix is a symmetric matrix with zero diagonal elements such that the ijth element represents how far apart or how dissimilar the ith and jth objects are.

Let X be an n by p data matrix of observations of p variables on n objects then the distance between object j and object k, d_{jk}, can be defined as:

$$d_{jk} = \left\{ \sum_{i=1}^{p} D(x_{ji}/s_i, x_{ki}/s_i) \right\}^{\alpha},$$

where x_{ji} and x_{ki} are the (ji)th and (ki)th elements of X, s_i is a standardization for the ith variable and $D(u, v)$ is a suitable function. Three functions are provided in G03EAF.

(a) Euclidean distance: $D(u, v) = (u - v)^2$ and $\alpha = \frac{1}{2}$.

(b) Euclidean squared distance: $D(u, v) = (u - v)^2$ and $\alpha = 1$.

(c) Absolute distance (city block metric): $D(u, v) = |u - v|$ and $\alpha = 1$.

Three standardizations are available.

(a) Standard deviation: $s_i = \sqrt{\sum_{j=1}^{n} (x_{ji} - \bar{x})^2/(n-1)}$

(b) Range: $s_i = \max(x_{1i}, x_{2i}, \ldots, x_{ni}) - \min(x_{1i}, x_{2i}, \ldots, x_{ni})$

(c) User supplied values of s_i.

In addition to the above distances there are a large number of other dissimilarity measures, particularly for dichotomous variables (see Krzanowski (1990) and Everitt (1974)). For the dichotomous case these measures are simple to compute and can, if suitable scaling is used, be combined with the distances computed by G03EAF using the updating option.

Dissimilarity measures for variables can be based on the correlation coefficient for continuous variables and contingency table statistics for dichotomous data, see chapters G02 and G11 respectively.

G03EAF returns the strictly lower triangle of the distance matrix.

4 References

Everitt B S (1974) *Cluster Analysis* Heinemann

Krzanowski W J (1990) *Principles of Multivariate Analysis* Oxford University Press

5 Parameters

1: UPDATE – CHARACTER*1 *Input*

On entry: indicates whether or not an existing matrix is to be updated.

If UPDATE = 'U', the matrix D is updated and distances are added to D.

If UPDATE = 'I', the matrix D is initialised to zero before the distances are added to D.

Constraint: UPDATE = 'U' or 'I'.

2: DIST – CHARACTER*1 *Input*

On entry: indicates which type of distances are computed.

If DIST = 'A', absolute distances.

If DIST = 'E', Euclidean distances.

If DIST = 'S', Euclidean squared distances.

Constraint: DIST = 'A', 'E' or 'S'.

3: SCALE – CHARACTER*1 *Input*

On entry: indicates the standardization of the variables to be used.

If SCALE = 'S', standard deviation.

If SCALE = 'R', range.

If SCALE = 'G', standardizations given in array S.

If SCALE = 'U', unscaled.

Constraint: SCALE = 'S', 'R', 'G' or 'U'.

4: N – INTEGER *Input*

On entry: the number of observations, n.

Constraint: N ≥ 2.

5: M – INTEGER *Input*

On entry: the total number of variables in array X.

Constraint: M > 0.

6: X(LDX,M) – ***real*** array *Input*

On entry: X(i,j) must contain the value of the jth variable for the ith object, for $i = 1, 2, \ldots, n$; $j = 1, 2, \ldots, \mathrm{M}$.

7: LDX – INTEGER *Input*

On entry: the first dimension of the array X as declared in the (sub)program from which G03EAF is called.

Constraint: LDX $\geq$ N.

8: ISX(M) – INTEGER array *Input*

On entry: ISX(j) indicates whether or not the jth variable in X is to be included in the distance computations.

If ISX$(j) > 0$ the jth variable is included, for $j = 1, 2, \ldots, \mathrm{M}$; otherwise it is not referenced.

Constraint: ISX$(j) > 0$ for at least one j, $j = 1, 2, \ldots, \mathrm{M}$.

9: S(M) – ***real*** array *Input/Output*

On entry: if SCALE = 'G' and ISX(j) > 0 then S(j) must contain the scaling for variable j, for $j = 1, 2, \ldots, \mathrm{M}$.

Constraint: if SCALE = 'G' and ISX(j) > 0 then S(j) > 0.0, for $j = 1, 2, \ldots, \mathrm{M}$.

On exit: if SCALE = 'S' and ISX(j) > 0 then S(j) contains the standard deviation of the variable in the jth column of X. If SCALE = 'R' and ISX(j) > 0 then S(j) contains the range of the variable in the jth column of X. If SCALE = 'U' and ISX(j) > 0 then S(j) = 1.0 and if SCALE = 'G' then S is unchanged.

10: D(N*(N−1)/2) – ***real*** array *Input/Output*

On entry: if UPDATE = 'U' then D must contain the strictly lower triangle of the distance matrix D to be updated. D must be stored packed by rows, i.e., D($(i-1)(i-2)/2+j$), $i > j$ must contain d_{ij}.

Constraint: if UPDATE = 'U' then D(j) ≥ 0.0, for $j = 1, 2, \ldots, n(n-1)/2$.

On exit: the strictly lower triangle of the distance matrix D stored packed by rows, i.e., d_{ij} is contained in D($(i-1)(i-2)/2+j$), $i > j$.

11: 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, N < 2,
or LDX < N,
or M ≤ 0,
or UPDATE ≠ 'I' or 'U',
or DIST ≠ 'A', 'E' or 'S',
or SCALE ≠ 'S', 'R', 'G' or 'U'.

IFAIL = 2

On entry, ISX(j) ≤ 0 for $j = 1, 2, \ldots, \mathrm{M}$,
or UPDATE = 'U' and D(j) < 0.0, for some $j = 1, 2, \ldots, n(n-1)/2$,
or SCALE = 'S' or 'R' and X(i, j) = X($i+1, j$) for $i = 1, 2, \ldots, n-1$, for some j with ISX(i) > 0.
or S(j) ≤ 0.0 for some j when SCALE = 'G' and ISX(j) > 0.

7 Accuracy

The computations are believed to be stable.

8 Further Comments

G03ECF can be used to perform cluster analysis on the computed distance matrix.

9 Example

A data matrix of five observations and three variables is read in and a distance matrix is calculated from variables 2 and 3 using squared Euclidean distance with no scaling. This matrix is then printed.

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.

```
*     G03EAF Example Program Text
*     Mark 16 Release. NAG Copyright 1992.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, MMAX
      PARAMETER        (NMAX=10,MMAX=10)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, J, LDX, M, N
      CHARACTER        DIST, SCALE, UPDATE
*     .. Local Arrays ..
      real             D(NMAX*(NMAX-1)/2), S(MMAX), X(NMAX,MMAX)
      INTEGER          ISX(MMAX)
*     .. External Subroutines ..
      EXTERNAL         G03EAF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G03EAF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N, M
      IF (N.LE.NMAX .AND. M.LE.MMAX) THEN
         READ (NIN,*) UPDATE, DIST, SCALE
         DO 20 J = 1, N
            READ (NIN,*) (X(J,I),I=1,M)
   20    CONTINUE
         READ (NIN,*) (ISX(I),I=1,M)
         READ (NIN,*) (S(I),I=1,M)
*
*        Compute the distance matrix
*
         IFAIL = 0
         LDX = NMAX
*
         CALL G03EAF(UPDATE,DIST,SCALE,N,M,X,LDX,ISX,S,D,IFAIL)
*
*        Print the distance matrix
*
         IFAIL = 0
         WRITE (NOUT,*)
         WRITE (NOUT,*) ' Distance Matrix'
         WRITE (NOUT,*)
         WRITE (NOUT,99999) '    1       2       3       4'
         WRITE (NOUT,*)
         DO 40 I = 2, N
            WRITE (NOUT,99998) I, (D(J),J=(I-1)*(I-2)/2+1,I*(I-1)/2)
   40    CONTINUE
      END IF
      STOP
*
99999 FORMAT (5X,A)
99998 FORMAT (1X,I2,2X,4(3X,F5.2))
      END
```

9.2 Program Data

```
G03EAF Example Program Data
5 3
'I' 'S' 'U'
1.0 1.0 1.0
2.0 1.0 2.0
3.0 6.0 3.0
4.0 8.0 2.0
5.0 8.0 0.0
 0   1   1
1.0 1.0 1.0
```

9.3 Program Results

```
 G03EAF Example Program Results

  Distance Matrix

         1       2       3       4

  2      1.00
  3     29.00   26.00
  4     50.00   49.00    5.00
  5     50.00   53.00   13.00    4.00
```