NAG Fortran Library Routine Document

G03ECF

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

G03ECF performs hierarchical cluster analysis.

2 Specification

```
      SUBROUTINE G03ECF(METHOD, N, D, ILC, IUC, CD, IORD, DORD, IWK, IFAIL)
      INTEGER           METHOD, N, ILC(N-1), IUC(N-1), IORD(N), IWK(2*N),
     1                  IFAIL
      real              D(N*(N-1)/2), CD(N-1), DORD(N)
```

3 Description

Given a distance or dissimilarity matrix for n objects (see G03EAF), cluster analysis aims to group the n objects into a number of more or less homogeneous groups or clusters. With agglomerative clustering methods, a hierarchical tree is produced by starting with n clusters, each with a single object and then at each of $n-1$ stages, merging two clusters to form a larger cluster, until all objects are in a single cluster. This process may be represented by a dendrogram (see G03EHF).

At each stage the clusters that are nearest are merged, methods differ as to how the distances between the new cluster and other clusters are computed. For three clusters i, j and k let n_i, n_j and n_k be the number of objects in each cluster and let d_{ij}, d_{ik} and d_{jk} be the distances between the clusters. Let clusters j and k be merged to give cluster jk, then the distance from cluster i to cluster jk, $d_{i.jk}$ can be computed in the following ways.

1. Single Link or nearest neighbour : $d_{i.jk} = \min(d_{ij}, d_{ik})$.
2. Complete Link or furthest neighbour : $d_{i.jk} = \max(d_{ij}, d_{ik})$.
3. Group average : $d_{i.jk} = \dfrac{n_j}{n_j + n_k} d_{ij} + \dfrac{n_k}{n_j + n_k} d_{ik}$.
4. Centroid : $d_{i.jk} = \dfrac{n_j}{n_j + n_k} d_{ij} + \dfrac{n_k}{n_j + n_k} d_{ik} - \dfrac{n_j n_k}{(n_j + n_k)^2} d_{jk}$.
5. Median : $d_{i.jk} = \frac{1}{2} d_{ij} + \frac{1}{2} d_{ik} - \frac{1}{4} d_{jk}$.
6. Minimum variance : $d_{i.jk} = \{(n_i + n_j) d_{ij} + (n_i + n_k) d_{ik} - n_i d_{jk}\} / (n_i + n_j + n_k)$.

For further details see Everitt (1974) or Krzanowski (1990).

If the clusters are numbered $1, 2, \ldots, n$ then for convenience if clusters j and k, $j < k$, merge then the new cluster will be referred to as cluster j. Information on the clustering history is given by the values of j, k and d_{jk} for each of the $n-1$ clustering steps. In order to produce a dendrogram, the ordering of the objects such that the clusters that merge are adjacent is required. This ordering is computed so that the first element is 1. The associated distances with this ordering are also computed.

4 References

Everitt B S (1974) *Cluster Analysis* Heinemann

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

5 Parameters

1: METHOD – INTEGER *Input*

On entry: indicates which clustering method is used.

METHOD = 1

Single link.

METHOD = 2

Complete link.

METHOD = 3

Group average.

METHOD = 4

Centroid.

METHOD = 5

Median.

METHOD = 6

Minimum variance.

Constraint: METHOD = 1, 2, 3, 4, 5 or 6.

2: N – INTEGER *Input*

On entry: the number of objects, n.

Constraint: N $\geq$ 2.

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

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

On exit: D is destroyed.

Constraint: D$(i)\geq 0.0$, for $i=1,2,\ldots,n(n-1)/2$.

4: ILC(N−1) – INTEGER array *Output*

On exit: ILC(l) contains the number, j, of the cluster merged with cluster k (see IUC), $j<k$, at step l for $l=1,2,\ldots,n-1$.

5: IUC(N−1) – INTEGER array *Output*

On exit: IUC(l) contains the number, k, of the cluster merged with cluster j, $j<k$, at step l for $l=1,2,\ldots,n-1$.

6: CD(N−1) – ***real*** array *Output*

On exit: CD(l) contains the distance d_{jk}, between clusters j and k, $j<k$, merged at step l for $l=1,2,\ldots,n-1$.

7: IORD(N) – INTEGER array *Output*

On exit: the objects in dendrogram order.

8: DORD(N) – ***real*** array *Output*

On exit: the clustering distances corresponding to the order in IORD. DORD(l) contains the distance at which cluster IORD(l) and IORD$(l+1)$ merge, for $l=1,2,\ldots,n-1$. DORD(n) contains the maximum distance.

9: IWK(2*N) – INTEGER array *Workspace*

10: 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, METHOD $\neq$ 1, 2, 3, 4, 5 or 6,
or N $<$ 2.

IFAIL $= 2$

On entry, D(i) $<$ 0.0 for some $i = 1, 2, \ldots, n(n-1)/2$.

IFAIL $= 3$

A true dendrogram cannot be formed because the distances at which clusters have merged are not increasing for all steps, i.e., CD(l) $<$ CD($l-1$) for some $l = 2, 3, \ldots, n-1$. This can occur for the median and centroid methods.

7 Accuracy

For METHOD $\geq$ 3 slight rounding errors may occur in the calculations of the updated distances. These would not normally significantly affect the results, however there may be an effect if distances are (almost) equal.

If at a stage, two distances d_{ij} and d_{kl}, $(i < k)$ or $(i = k)$, and $j < l$, are equal then clusters k and l will be merged rather than clusters i and j. For single link clustering this choice will only affect the order of the objects in the dendrogram. However, for other methods the choice of kl rather than ij may affect the shape of the dendrogram. If either of the distances d_{ij} and d_{kl} is affected by rounding errors then their equality, and hence the dendrogram, may be affected.

8 Further Comments

The dendrogram may be formed using G03EHF. Groupings based on the clusters formed at a given distance can be computed using G03EJF.

9 Example

Data consisting of three variables on five objects are read in. Euclidean squared distances based on two variables are computed using G03EAF, the objects are clustered using G03ECF and the dendrogram computed using G03EHF. The dendrogram 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.

```
*     G03ECF Example Program Text
*     Mark 17 Revised.  NAG Copyright 1995.
*
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, MMAX, LENC
      PARAMETER        (NMAX=10,MMAX=10,LENC=20)
*     .. Local Scalars ..
      real             DMIN, DSTEP, YDIST
      INTEGER          I, IFAIL, J, LDX, M, METHOD, N, NSYM
      CHARACTER        DIST, SCALE, UPDATE
*     .. Local Arrays ..
      real             CD(NMAX-1), D(NMAX*(NMAX-1)/2), DORD(NMAX),
     +                 S(MMAX), X(NMAX,MMAX)
      INTEGER          ILC(NMAX-1), IORD(NMAX), ISX(MMAX), IUC(NMAX-1),
     +                 IWK(2*NMAX)
      CHARACTER*60     C(LENC)
      CHARACTER*3      NAME(NMAX)
*     .. External Subroutines ..
      EXTERNAL         G03EAF, G03ECF, G03EHF
*     .. Intrinsic Functions ..
      INTRINSIC        real, MOD
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G03ECF 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,*) METHOD
         READ (NIN,*) UPDATE, DIST, SCALE
         DO 20 J = 1, N
            READ (NIN,*) (X(J,I),I=1,M), NAME(J)
   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)
*
*     Perform clustering
*
         IFAIL = 0
*
         CALL G03ECF(METHOD,N,D,ILC,IUC,CD,IORD,DORD,IWK,IFAIL)
*
         WRITE (NOUT,*)
         WRITE (NOUT,*) '  Distance   Clusters Joined'
         WRITE (NOUT,*)
         DO 40 I = 1, N - 1
            WRITE (NOUT,99999) CD(I), NAME(ILC(I)), NAME(IUC(I))
   40    CONTINUE
*
*     Produce dendrogram
*
         IFAIL = 0
         NSYM = LENC
         DMIN = 0.0e0
         DSTEP = (CD(N-1))/real(NSYM)
*
         CALL G03EHF('S',N,DORD,DMIN,DSTEP,NSYM,C,LENC,IFAIL)
```

```
*
         WRITE (NOUT,*)
         WRITE (NOUT,*) 'Dendrogram'
         WRITE (NOUT,*)
         YDIST = CD(N-1)
         DO 60 I = 1, NSYM
            IF (MOD(I,3).EQ.1) THEN
               WRITE (NOUT,99999) YDIST, C(I)
            ELSE
               WRITE (NOUT,99998) C(I)
            END IF
            YDIST = YDIST - DSTEP
   60    CONTINUE
         WRITE (NOUT,*)
         WRITE (NOUT,99998) (NAME(IORD(I)),I=1,N)
      END IF
      STOP
*
99999 FORMAT (F10.3,5X,2A)
99998 FORMAT (15X,20A)
      END
```

9.2 Program Data

```
G03ECF Example Program Data
5 3
5
'I' 'S' 'U'
 1  5.0 2.0 'A  '
 2  1.0 1.0 'B  '
 3  4.0 3.0 'C  '
 4  1.0 2.0 'D  '
 5  5.0 0.0 'E  '
 0   1   1
1.0 1.0 1.0
```

9.3 Program Results

```
 G03ECF Example Program Results

   Distance   Clusters Joined

      1.000     B   D
      2.000     A   C
      6.500     A   E
     14.125     A   B

 Dendrogram

     14.125           -------
                      I     I
                      I     I
     12.006           I     I
                      I     I
                      I     I
      9.887           I     I
                      I     I
                      I     I
      7.769           I     I
                   ---*     I
                   I  I     I
      5.650        I  I     I
                   I  I     I
                   I  I     I
      3.531        I  I     I
                   I  I     I
                ---*  I     I
      1.412     I  I  I  ---*
                I  I  I  I  I

                A  C  E  B  D
```