NAG 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), IFAIL
REAL (KIND=nag_wp) 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 : $d_{i.jk} = \left\{(n_i + n_j) d_{ij} + (n_i + n_k) d_{ik} - n_i d_{jk}\right\} / (n_i + n_j + n_k)$.

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: n, the number of objects.

Constraint: $\mathrm{N} \geq 2$.

3: $\mathrm{D}(\mathrm{N} \times (\mathrm{N} - 1)/2)$ – REAL (KIND=nag_wp) array *Input/Output*

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

On exit: is overwritten.

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

4: $\mathrm{ILC}(\mathrm{N} - 1)$ – INTEGER array *Output*

On exit: $\mathrm{ILC}(l)$ 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: $\mathrm{IUC}(\mathrm{N} - 1)$ – INTEGER array *Output*

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

6: $\mathrm{CD}(\mathrm{N} - 1)$ – REAL (KIND=nag_wp) array *Output*

On exit: $\mathrm{CD}(l)$ 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 (KIND=nag_wp) array *Output*

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

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

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, if you are 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.**

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, 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

```
    Program g03ecfe

!     G03ECF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: g03eaf, g03ecf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter              :: nin = 5, nout = 6, rnlen = 3
!     .. Local Scalars ..
      Integer                         :: i, ifail, ld, ldx, liwk, m, method,  &
                                         n, n1
      Character (1)                   :: dist, scal, update
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable :: cd(:), d(:), dord(:), s(:), x(:,:)
      Integer, Allocatable            :: ilc(:), iord(:), isx(:), iuc(:),     &
                                         iwk(:)
      Character (rnlen), Allocatable  :: row_name(:)
!     .. Executable Statements ..
      Write (nout,*) 'G03ECF Example Program Results'
      Write (nout,*)

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

!     Read in the problem size
      Read (nin,*) n, m

!     Read in information on the type of distance matrix to use
      Read (nin,*) update, dist, scal

      ldx = n
      ld = n*(n-1)/2
      n1 = n - 1
      liwk = 2*n
      Allocate (x(ldx,m),isx(m),s(m),d(ld),ilc(n1),iuc(n1),cd(n1),iord(n), &
        dord(n),iwk(liwk),row_name(n))

!     Read in the data used to construct distance matrix
      Read (nin,*)(x(i,1:m),i=1,n)

!     Read in variable inclusion flags
      Read (nin,*) isx(1:m)

!     Read in scaling
      If (scal=='G' .Or. scal=='g') Then
        Read (nin,*) s(1:m)
      End If

!     Compute the distance matrix
      ifail = 0
      Call g03eaf(update,dist,scal,n,m,x,ldx,isx,s,d,ifail)

!     Read in information on the clustering method to use
      Read (nin,*) method

!     Read in first RNLEN characters of row names. Used to make example
!     output easier to read
      Read (nin,*) row_name(1:n)

!     Perform clustering
      ifail = 0
      Call g03ecf(method,n,d,ilc,iuc,cd,iord,dord,iwk,ifail)

!     Display results
      Write (nout,*) '  Distance   Clusters Joined'
      Write (nout,*)
      Write (nout,99999)(cd(i),row_name(ilc(i)),row_name(iuc(i)),i=1,n1)

99999 Format (F10.3,5X,2A)
    End Program g03ecfe
```

9.2 Program Data

```
G03ECF Example Program Data
5 3                    : N,M (G03EAF)
'I' 'S' 'U'            : UPDATE,DIST,SCAL (G03EAF)
 1  5.0 2.0
 2  1.0 1.0
 3  4.0 3.0
 4  1.0 2.0
 5  5.0 0.0            : End of X (G03EAF)
 0   1   1             : ISX (G03EAF)
5                      : METHOD (G03ECF)
'A' 'B' 'C' 'D' 'E'   : Row names (ROW_NAME)
```

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
```