Algorithm AS 102

Ultrametric Distances for a Single Linkage Dendrogram

By C. F. Banfield

Rothamsted Experimental Station, Harpenden, Herts, Britain

Keywords: ULTRAMETRIC DISTANCES; MINIMUM SPANNING TREE; SINGLE LINKAGE DENDROGRAM

Language

ISO Fortran

Description and Purpose

This algorithm computes ultrametric distances for a single linkage dendrogram using the minimum spanning tree derived from a given distance matrix. The ultrametric distances are the maximum links on the chain joining each pair of points of the tree. They are also the levels in a dendrogram at which the pairs of points join. Such distances derived from a dendrogram may be compared with the given distances in a variety of goodness-of-fit statistics. Gower and Banfield (1975) give a brief review and an account of a simulation study in which the algorithm given below was extensively used.

Method

Prim (1957) gives an algorithm for constructing sequentially a tree joining n points with $n-1$ links so that the sum of the lengths of these links is a minimum (termed the minimum spanning tree). *ULTRAM* uses a version of Prim's algorithm to store the points in array P in the order they are joined to the tree, and the $n-1$ distances (link lengths) similarly ordered in array W. These distances are the initial $n-1$ entries of the array U, which ultimately holds all the ultrametric distances stored by rows in lower triangular matrix form. An iterative process overwrites W, so that at step i the first $(n-i-1)$ elements of W are replaced by

$$W(j) = \max(W(j), W(j+1)), \quad j = 1, \ldots, n-i-1.$$

These are then the $(n-i-1)$ ultrametric distances between the points held in $P(j)$ and $P(j+i+1)$, $j = 1, \ldots, n-i-1$, and are transferred to U before the next iteration. After $n-2$ iterations U holds all the ultrametric distances.

Structure

SUBROUTINE ULTRAM (*N, NN, D, U, W, P, IND, IFAULT*)

Formal parameters

N	Integer	input:	n, the number of points
NN	Integer	input:	$n \times (n-1)/2$, the number of distances
D	Real array (*NN*)	input:	the lower triangular distance matrix stored by rows
U	Real array (*NN*)	output:	the lower triangular matrix of ultrametric distances stored by rows
W	Real array (*N*)	work array:	the ultrametric distances obtained at each iteration
P	Integer array (*N*)	work array:	the points in the order they are joined to the tree
IND	Integer array (*N*)	work array:	indicates which points have been joined to the tree
IFAULT	Integer	output:	set to 1 if $n<2$, otherwise 0

Restrictions

The minimum permissible value of n is 2. Missing values are not allowed in D. If any are present they can be coded as negative distances and tests added to the algorithm so that they are ignored.

Time

The time required depends on an^2+bn+c. On the ICL 4.70 the time in milliseconds is given by $0{\cdot}19\,n^2+0{\cdot}04\,n+6{\cdot}78$.

Accuracy

The ultrametric distances will be unique for a given distance matrix, even when ties induce a non-unique minimum spanning tree.

Related Algorithms

Ross (1969) gives an Algol procedure for Prim's algorithm.

References

Gower, J. C. & Banfield, C. F. (1975). Goodness of fit criteria for hierarchical classification and their empirical distributions. Proceedings of the 8th International Biometrics Conference, Rumania, 1974.
Prim, R. C. (1957). *Bell System Tech. J.*, 36, 1389–1401.
Ross, G. J. S. (1969). Algorithm AS 13: minimum spanning tree. *Appl. Statist.*, **18**, 103–104.

```
      SUBROUTINE ULTRAM(N, NN, D, U, W, P, IND, IFAULT)
C
C        ALGORITHM AS 102  APPL. STATIST. (1976) VOL.25, NO.3
C
C        COMPUTES ULTRAMETRIC DISTANCES FOR A SINGLE LINKAGE DENDROGRAM.
C
C        THE NN DISTANCES BETWEEN N POINTS ARE GIVEN IN D. THE NN
C        ULTRAMETRIC DISTANCES ARE HELD IN U. W, P AND IND ARE WORKING
C        ARRAYS AND IFAULT INDICATES IF N IS LESS THAN 2.
C
      REAL D(NN), U(NN), W(N)
      INTEGER P(N), IND(N)
C
C        DLARGE IS A CONSTANT LARGER THAN ANY VALUE OF D.
C
      DATA DLARGE /1.0E50/
C
C        INITIALISE IND( ) AND THE FIRST N ELEMENTS OF U( ).
C
      IFAULT = 1
      IF (N .LT. 2) RETURN
      IFAULT = 0
      DO 1 I = 1, N
      IND(I) = 0
      U(I) = DLARGE
    1 CONTINUE
C
C        COMPUTE THE LINKS, U( ) AND THE ORDER THE LINKS ARE JOINED
C        TO THE MINIMUM SPANNING TREE, P( ). INITIALISE W( ) WITH
C        THE ORDERED LINKS.
C
      J = 1
      P(1) = 1
      DO 3 I = 2, N
      A = DLARGE
      KK = 0
      JJ = ((J - 1) * (J - 2)) / 2
      DO 2 K = 2, N
      KK = KK + K - 2
      IF (IND(K) .NE. 0) GOTO 2
```

```
      ID = KK + J
      IF (J .GT. K) ID = JJ + K
      DIST = D(ID)
      IF (DIST .LT. U(K)) U(K) = DIST
      IF (A .LE. U(K)) GOTO 2
      A = U(K)
      NEXT = K
    2 CONTINUE
      J = NEXT
      P(I) = NEXT
      W(I - 1) = U(NEXT)
      IND(J) = 1
    3 CONTINUE
C
C        COMPUTE THE ULTRAMETRIC DISTANCES BY FINDING THE MAXIMUM OF
C        EACH ADJACENT PAIR OF PREVIOUSLY COMPUTED DISTANCES IN W( ).
C
      N1 = N - 1
      DO 5 I = 1, N1
      N2 = N - I
      DO 4 J = 1, N2
      K = J + I
      L = MAX0(P(J), P(K))
      M = MIN0(P(J), P(K))
      IU = ((L - 1) * (L - 2)) / 2 + M
      U(IU) = W(J)
      IF (J .EQ. N2) GOTO 4
      IF (W(J + 1) .GT. W(J)) W(J) = W(J + 1)
    4 CONTINUE
    5 CONTINUE
      RETURN
      END
```

Algorithm AS 103

Psi (Digamma) Function

By J. M. Bernardo

University College London, Britain

Keywords: PSI; DIGAMMA; BETA DENSITIES; GAMMA DENSITIES

Language

ISO Fortran

Description and Purpose

A routine is presented to compute

$$\psi(x) = d\{\log \Gamma(x)\}/dx = \Gamma'(x)/\Gamma(x)$$

the psi or digamma function, for real positive values of x.

While the functions $\Gamma(x)$ and $\log \Gamma(x)$ are provided in most systems, this is not so with $\psi(x)$ which nevertheless often occurs in statistical practice, particularly when Beta or Gamma densities are involved.