seriation R Documentation

Seriation of a (dis)similarity matrix

Description

This function maximizes the affinity values near the diagonal of a resemblance matrix (or, more generally, an affinity matrix), as per Beum and Brundage (1950). This method represents an interesting alternative to clustering in the case of non-symmetric resemblance matrices. The method is also described in section 8.10 of Legendre and Legendre (2012).

Usage

```
seriation(mat, verbose=FALSE)
```

Arguments

`mat` A square, symmetric or non-symmetric similarity or distance matrix. If it is a symmetric dissimilarity matrix of class ‘dist’, produced for example by dist {stats}, vegdist {vegan}, or dist.binary {ade4}, it will be converted to square similarity matrix form by the function.

verbose =TRUE : print results of intermediate calculations to the R window. Default: verbose=FALSE.

Details

The input matrix may be a symmetric or non-symmetric matrix of resemblance containing non-negative values. If it is square and contains distances (0’s on the main diagonal), the function transforms the distances to similarities (1’s on the diagonal) before the calculation. The diagonal elements are excluded from the seriation calculations.

If the input matrix belongs to class 'dist', it is assumed to contain dissimilarities, which are converted by the function to similarities.

Note: the Beum & Brundage algorithm does not always move the highest similarity values near the diagonal. Check the results before using them. For an unknown reason, it seems to succeed more often with asymmetric than with symmetric **S** matrices.

Value

Function ***seriation()*** returns a list with five elements:

starting.S Similarity matrix used at the beginning of the function. A distance matrix **D** is converted by the function to a similarity matrix **S** in the range [0,1].

reordered.S The re-ordered similarity matrix (**S**), with the original names for the rows and columns.

ranking The vector used to reorder the objects (rows and columns).

dat The three statistics computed on the matrix:
sum.col = column sums;
sum.pr = column sums multiplied by 'mult';
pr.ave = sum.pr divided by sum.col.

sdev standard deviation of the pr.ave values. 'mat' is the reordered matrix with the largest value of 'sdev'.

References

Beum, C. O. Jr. and E. G. Brundage. 1950. A method for analyzing the sociomatrix. Sociometry 13: 141-145.

Legendre, P. and Legendre, L. 2012. *Numerical Ecolog*y, 3rd English ed. Elsevier, Amsterdam.

Author

Translated to R by Pierre Legendre and Sébastien Durand, Département de sciences biologiques, Université de Montréal, from a FORTRAN program written by P. Legendre.

Examples

```
# Non-symmetric matrix from Beum and Brundage (1950)
# The affinity values are 2 (alrgest affinity), 1, or 0
mat.Beum = matrix(c(1,1,0,0,1,0,2,1,0,0,2,0,0,0,1,2,0,2,0,0,2,1,0,1,1,2,0,0,1,0,0,0,1,1,0,1), 6, 6)
ser.out = seriation(mat.Beum)
ser.out

# Symmetric dissimilarity matrix computed from a table of random numbers.
mat = matrix(rnorm(140),7,20)
mat.D = dist(mat)
ser.out = seriation(mat.D)
ser.out
# Compare with hierarchical clustering results
mat.clust = hclust(mat.D, method="average")
plot(mat.clust)
```