1979

Clustar and Clustid: Programs for Hierarchical Cluster Analysis

K. A. Marshall

H. C. Romesburg

FINAL REPORT

CLUSTAR AND CLUSTID: PROGRAMS FOR HIERARCHICAL CLUSTER ANALYSIS

K. A. Marshall and H. C. Romesburg
Utah State University

US/IBP DESERT BIOME
RESEARCH MEMORANDUM 77-28

in

FINAL PROGRESS REPORTS
Data Processing, pp. 1-32

1976 Proposal No. 2.1.2

Citation format: Author(s). 1979. Title.
US/IBP Desert Biome Res. Memo. 77-28.
Utah State Univ., Logan. 32 pp.

Ecology Center, Utah State University, Logan, Utah 84322

Preface—The classic approach to hierarchical cluster analysis starts with a matrix of attribute data on the objects to be clustered. These data are used to calculate coefficients of similarity or dissimilarity between all pairs of objects. The method concludes with the combining of these coefficients to produce a tree (dendrogram) which hopefully summarizes the structure inherent in the original data. Cluster analysis is useful in applied research because its foundation assumptions are less stringent than other techniques such as principal component analysis or factor analysis which can sometimes be used for the same research purposes. But none of these methods, including cluster analysis, is suitable when the attribute data are a mixture of metric- and binary-scaled numbers. Yet most real problems do not come with this numerical invariance.

CLUSTAR was written to handle mixed- and single-scaled data. What's more:

1) CLUSTAR provides faster computing than most extant cluster analysis programs. Compared with MINT, a popular cluster analysis program used in numerical taxonomy, CLUSTAR solved the same problems on a Burroughs B6700 in less than one-half the time.

2) Output is labeled clearly. The output is neat and self-explanatory.

3) A wide variety of similarity and dissimilarity coefficients, data standardization formulas and hierarchical clustering methods are given.

4) Indices for comparing the agreement of trees produced by different clustering methods are provided.

5) The FORTRAN code is written so that adaptation to other computer hardware will require no program modifications. Realistically, some changes may be required, but these should be minimal.

From the tree produced by CLUSTAR the user can define clusters of objects. A common problem at this point is the need to establish cluster membership for a new set of objects. Where do the new objects belong in the clusters produced by CLUSTAR? CLUSTID performs the placement.

We use the terms "metric scale" data and "non-metric scale" data. By metric scale data we mean data on internal or ratio scales, and by non-metric scale data we mean data on a nominal scale of measurement.

Charles Romesburg designed the methods and wrote this manual. Kim Marshall designed the algorithms to execute the methods and wrote the FORTRAN code. We and Tim Mauk have independently checked the computed results with hand calculations. We believe CLUSTAR and CLUSTID to be error-free. But we are not idealists; should errors be found, please tell us. Corrections will be made and those receiving this report will be notified.

The FORTRAN code for CLUSTAR and CLUSTID comprises 1700 and 1000 card images, respectively. Rather than give program listings in this report, we will send them on magnetic tape.

Write:

H. Charles Romesburg
Dept. Forestry and Outdoor Recreation
Utah State University—UMC52
Logan, Utah 84322

Please enclose $30.00 payable to Ecology Center, Utah State University to cover costs.

The US/IBP Desert Biome and U.S. Forest Service ECOSYSM project (Grant No. YNE-079-1) in the Utah State University Dept. of Forestry and Outdoor Recreation provided funding for the creation of these programs.

INTRODUCTION

ASSUMING THAT WE CAN . . . GET RID OF ALL UNIVERSALS EXCEPT SIMILARITY, IT REMAINS TO BE CONSIDERED WHETHER SIMILARITY ITSELF COULD BE EXPLAINED AWAY.

Bertrand Russell, *An Inquiry into Meaning and Truth*

This manual presupposes a knowledge of cluster analysis at the level of the book by Everitt (1974). CLUSTAR and CLUSTID are separate programs. CLUSTAR performs hierarchical cluster analysis. That is, CLUSTAR is used to define classes (clusters) of objects. CLUSTID is used to assign "new" objects of unknown class membership into one of the classes produced by CLUSTAR. So, if CLUSTID is run, it is run after CLUSTAR. To further explain the differences between these programs, consider the analogy to taxonomy. The taxonomist sorts specimens of a taxon into sets of similar individuals, and if the distinction between these sets is judged great enough, the taxon is subdivided accordingly. This corresponds to the job done by CLUSTAR. Now suppose someone brings in one or more specimens of this taxon and asks to which of the classes the specimens belong. The problem is to identify the class membership for these individuals. This corresponds to the task done by CLUSTID.

A run of either CLUSTAR or CLUSTID consists of a stack of subroutine commands. Each program has a library of subroutines. The user selects from these subroutines and stacks them in the order in which he wants the job executed. Any order is permitted so long as the input data necessary for the execution of a given subroutine has either been read in or computed previously by another subroutine.

First, this manual gives a concise example of hierarchical cluster analysis worked by hand. These hand-calculated results should be compared to "Example 1 CLUSTAR Run" which is the computer-calculated version at the end of this report. "Example 2 CLUSTAR RUN" is a more complex CLUSTAR problem illustrating mixed-scale data. "Example 2 CLUSTID Run" is tailored to the CLUSTAR run to show how CLUSTID is to be used in conjunction with CLUSTAR. "Example 3 CLUSTAR Run" is the most complex, illustrating most everything CLUSTAR will do.

The middle section of this manual gives descriptions and formats for the CLUSTAR and CLUSTID subroutines.

EXAMPLE OF HIERARCHICAL CLUSTER ANALYSIS

Suppose this hypothetical problem: a wilderness manager wants to group t = 5 campsites (objects) into a hierarchical cluster arrangement in which each campsite is described by measurements made on n = 2 metric-scaled attributes. These values are entered into the Data Matrix (Fig. 1), a matrix *R* (called the Resemblance Matrix) of resemblance coefficients between all pairs of campsites is calculated and, in turn, a clustering method is used with the data in *R* to produce a tree showing the hierarchy. In general, the Data Matrix can contain nominal, ordinal and metric data. CLUSTAR allows data on different scales to be intermixed within a given Data Matrix.

For each measurement scale, there are many resemblance coefficients from which to choose; this example uses the average Euclidean distance (d_{jk}) as a measure of dissimilarity between objects j and k:

$$d_{jk} = \sqrt{\sum_{i=1}^{n} (X_{ij} - X_{ik})^2 / n} \qquad (1)$$

where

$$0.0 < d_{jk} < \infty$$

and

X_{ij}, X_{ik} = Data Matrix value of attribute i for objects j and k, respectively.

Both similarity and dissimilarity coefficients fall under the name "resemblance coefficient." Similarity coefficients between two objects take on their largest values when the two objects are most alike and their smallest values when least alike. Dissimilarity coefficients take on their smallest values when the two objects are most alike and their largest values when least alike.

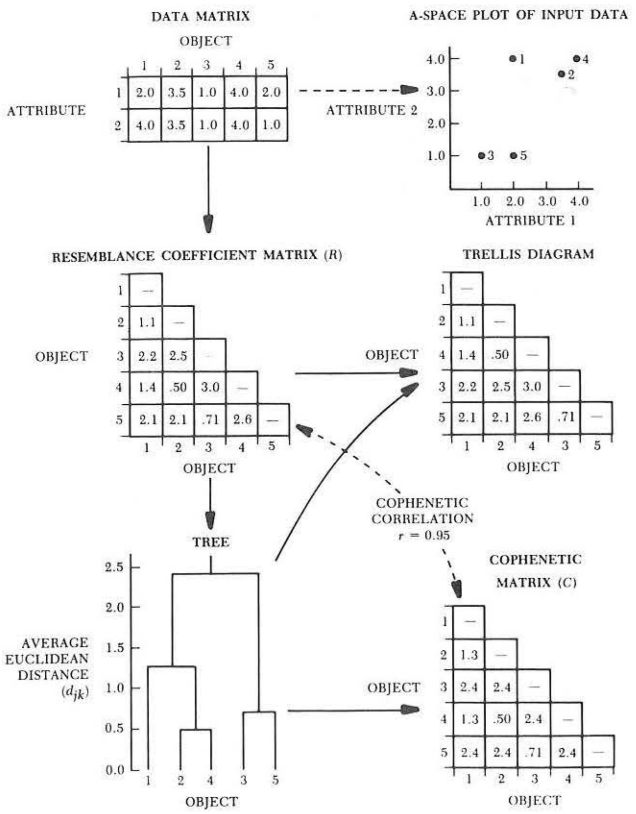


Figure 1. Diagram of steps in hierarchical cluster analysis.

Intuitively, each object is viewed as a point in n-dimensional space, or A-space. A pair of close points corresponds to a small d_{jk} value and high similarity, while distant points have a large d_{jk} value and low similarity. Using Equation 1 and the Data Matrix (Fig. 1), the 10 d_{jk} entries in R are calculated.

The tree is produced by operating on R with a clustering method. At the start, each campsite is a distinct cluster ($p = 5$) represented by the bottom of the tree in Figure 1. A clustering step consists of merging the two most similar clusters into a new cluster, reducing p by 1. The process must stop when $p = 1$, i.e., all campsites are in one cluster, as symbolized by the top of the tree.

Figure 1 illustrates "unweighted pair-group method using arithmetic averages" (UPGMA) clustering (Sneath and Sokal 1973; 230-234). The method begins by finding the smallest value of d_{jk}, i.e., $d_{24} = 0.50$. Campsites 2 and 4 are joined on the tree at this value and the new cluster is labeled with the next free integer (6). The number of p clusters has been reduced by one, leaving clusters labeled 1, 3, 5 and 6. Next, a new matrix of similarity coefficients R' (not shown in Fig. 1) is constructed for the four remaining clusters. Of the $4(4-1)/2 = 6$ entries in R', d_{13}, d_{15} and d_{35} are transcribed from R because the distances between clusters 1, 3 and 5 were unaffected by the creation of cluster 6. Values d_{16}, d_{36} and d_{56} needed to complete R' are found by averaging the distances between objects in one cluster and objects in the other. To illustrate, d_{16} is the average of the distances from every object in cluster 1 to every object in cluster 6. Since cluster 1 contains only object 1 while cluster 6 contains the two objects 2 and 4, $d_{16} = 1/2(d_{12} + d_{14}) = 1/2(1.1 + 1.4) = 1.25$. Having obtained in R' in this way, R' is searched for its smallest entry, and this turns out to be $d_{35} = 0.71$. Therefore, campsites 3 and 5 are joined at this level in the tree. At the next step ($p = 3$) the new cluster containing 3 and 5 is labeled 7, a new matrix R'' containing d_{16}, d_{17} and d_{67} is calculated [note: $d_{67} = 1/4(d_{23} + d_{25} + d_{43} + d_{45})$], and the smallest value in R'' is used to form a new cluster. Iterating in this manner, the process eventually reaches $p = 1$ and stops.

A comparison of the tree and A-space plot of the Data Matrix (Fig. 1) shows that the analytic method agrees with intuition. Campsites near one another in A-space are closely connected in the tree. The cophenetic correlation r, where $-1.0 < r < 1.0$, is a quantitative index of agreement between R and the tree. It is calculated by entering the d_{jk} value at which objects j and k join in the tree into the j,k cell of the Cophenetic Matrix, C. For example, campsites 2 and 3 join on the last clustering step at $d_{23} = 2.4$, and this value is recorded in C cell (2,3) (Fig. 1). Cell values in R and C are put into lists with corresponding cells adjacent, and r is calculated using the product moment correlation formula. A value near 1.0 indicates that the tree is a good surrogate for the similarity information contained in R.

The ordering of objects at the bottom of the tree can be used to rearrange R into a trellis diagram; for problems with a large number of objects, visual inspection of pairwise similarity becomes an easier task.

The example illustrates the steps in a cluster analysis. Sometimes these are augmented with other calculations (e.g., standardizing the data in the Data Matrix, performing matrix correlations to compare trees). These features are illustrated in the other examples at the end of this manual.

HOW TO PREPARE CLUSTAR INPUT CARDS

The CLUSTAR subroutines are named *INPT, *STAN, *SIMI, *SIMQ, *CLST, *RRNG, *COMB, *CMPR, *PNCH, *DUMP, *INST and *END. All runs will minimally have *INPT and *END. In the following section, the purpose and technicalities of each subroutine are given. The section ends with the subroutine formats.

SUBROUTINE *INPT

Purpose

Used to input a Data Matrix or Resemblance Matrix.

Technicalities

Four kinds of cards are required for this subroutine:

Card 1	*INPT card
Card 2	options used in *INPT
Card 3	(FORMAT) card
Card(s) 4	data card(s)

Either a Data Matrix or Resemblance Matrix can be read. If a Resemblance Matrix, it can be either a Similarity Matrix or a Dissimilarity Matrix.

Definitions—

Similarity Matrix: a large-valued entry corresponds to high resemblance between two objects
Dissimilarity Matrix: a small-valued entry corresponds to high resemblance between two objects

Resemblance Matrix is read in lower triangular form by row without the main diagonal. For instance, if the Resemblance Matrix for the four objects O_1, O_2, O_3, O_4 is:

	O_1	O_2	O_3	O_4
O_1	—			
O_2	6.2	—		
O_3	3.0	2.4	—	
O_4	1.6	2.1	8.8	—

The three data cards under F5.1 format are:

	Card Column														
	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	1 0	1 1	1 2	1 3	1 4	1 5
Card 1				6	2										
Card 2				3	0				2	4					
Card 3				1	6				2	1				8	8

If data go past column 80, continue on next card.

The format statement on card type 3 is the standard FORTRAN floating point format enclosed between a left parenthesis in Column 1 and a right parenthesis at the end of the format.

A Data Matrix can be read by column or row. Example:

Data Matrix

		Objects			
		O_1	O_2	O_3	O_4
Attributes	A_1	1.9	3.1	7.0	5.6
	A_2	6.2	1.0	3.7	7.8
	A_3	1.8	4.1	1.6	4.2

When NDIR = 0 is specified, the data go onto cards this way:

Card 1	1.9	6.2	1.8
Card 2	3.1	1.0	4.1
Card 3	7.0	3.7	1.6
Card 4	5.6	7.8	4.2

When NDIR = 1 is specified, the data go onto cards this way:

Card 1	1.9	3.1	7.0	5.6
Card 2	6.2	1.0	3.7	7.8
Card 3	1.8	4.1	1.6	4.2

Missing values are coded as decimal numbers; the code used should not appear in the data as a real value. Even if there are no missing values, it is a good idea to enter a number that is not part of the data (e.g., —9999) since some computers read blanks as zero values and this will turn any valid zeros in the data into missing values.

Missing values are permissible only for Data Matrices, not Resemblance Matrices.

Either the Data Matrix or its transpose can be printed.

Subroutine *STAN

Purpose

Used to standardize a Data Matrix.

Technicalities

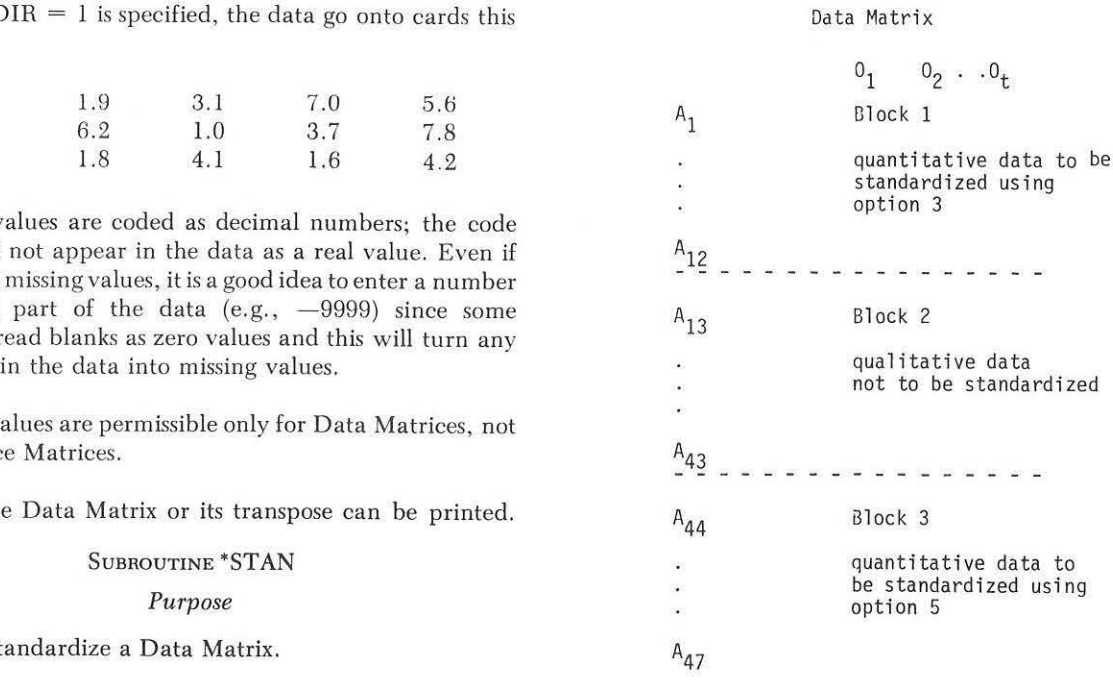
Five different methods of standardization exist. Let X_{ij} be the Dissimilarity Matrix (DM) value of object O_j on attribute A_i. Let $\overline{X}_i$ be the mean of the t values in the i'th attribute row. Let sd_i be the standard deviation of the t values in the i'th attribute row. Let min (X_{ij}) be the smallest value in the i'th row. Let max (X_{ij}) be the largest value in the i'th row.

The five standardization methods are:

1) $X_{ij}-\overline{X}_i$: adjusts for differences in the mean of each attribute.
2) X_{ij}/sd_i: adjusts for differences in the dispersion of each attribute.
3) $(X_{ij}-\overline{X}_i)/sd_i$: adjusts for mean and dispersion. See Sneath and Sokal (1973;154).
4) $X_{ij}/\max(X_{ij})$: makes X_{ij} a proportion of the largest value in the i'th row. Only makes sense when $X_{ij} \geq 0$. See Sneath and Sokal (1973; 153).
5) $[X_{ij} - \min(X_{ij})]/[\max(X_{ij}) - \min(X_{ij})]$: equalizes mean and variability of attributes. Scales the standardized variate to be between 0 and 1. See Sneath and Sokal (1973; 153).

Blocks of attributes (block = a series of adjacent attributes) in the Data Matrix can be standardized using different options. This feature is useful when one or more block(s) are quantitative data and it is desired to standardize these, while one or more block(s) are qualitative data for which standardization is not required. *STAN allows the user to standardize a given block of data according to any of the above methods, or to avoid standardizing certain blocks of data.

Example:



In this case, *STAN would be used twice. The first use standardizes the block of data A_1-A_{12} using option 3. The second standardizes block A_{44}-A_{47} using option 5.

Subroutine *SIMI

Purpose

Used to create a Resemblance Matrix from metric data.

Technicalities

A given block of data from either a Data Matrix or Standardized Data Matrix is transformed into a Resemblance Matrix. The following similarity and dissimilarity coefficients can be used:

Option	Similarity Coefficient	Reference	Range of values
1	correlation coefficient	Sneath and Sokal 1973;137-140	-1 ↔ 1
3	cos θ_{jk}	Boyce 1969;1-7	-1 ↔ 1
7	Bray-Curtis	Motyka et al. 1950;367-447	0 ↔ 1

Option	Dissimilarity Coefficient	Reference	Range of values
2	average Euclidean distance	Sneath and Sokal 1973;124	0 ↔ ∞
4	coefficient of shape difference	Boyce 1969;1-7	0 ↔ ∞
5	Clifford-Stephenson coefficient*	Clifford and Stephenson 1975;58	0 ↔ ∞
6	Canberra metric coefficient	Clifford and Stephenson 1975;58	0 ↔ 1

*Clifford and Stephenson (1975;58) call this the "Bray-Curtis coefficient." A different coefficient that is popularly called "Bray-Curtis" or "coefficient of community" is due to Motyka et al. (1950). To avoid confusion, we call Clifford and Stephenson's version by their names, and let option 7 be the commonly accepted meaning of "Bray-Curtis."

Formulas

Subscripts j and k are objects; subscript i is an attribute.

1) correlation coefficient r_{jk}:

$$r_{jk} = \frac{\left[\sum_{i=1}^{n} (X_{ij} - \bar{X}_j)(X_{ik} - \bar{X}_k)\right]}{\sqrt{\sum_{i=1}^{n} (X_{ij} - \bar{X}_j)^2 \sum_{i=1}^{n} (X_{ik} - \bar{X}_k)^2}}$$

2) average Euclidean distance d_{jk}:

$$d_{jk} = \sqrt{\sum_{i=1}^{n} (X_{ij} - X_{ik})^2 / n}$$

3) $\cos \Theta_{jk}$:

$$\cos \theta_{jk} = \frac{\sum_{i=1}^{n} X_{ij} X_{ik}}{\left(\sqrt{\sum_{i=1}^{n} X^2_{ij}} \sqrt{\sum_{i=1}^{n} X^2_{ik}}\right)}$$

4) coefficient of shape difference z_{jk}:

where

d_{jk} = average Euclidean distance,

and

$$Q^2_{jk} = 1/n^2 \left(\sum_{i=1}^{n} X_{ij} - \sum_{i=1}^{n} X_{ik}\right)^2,$$

$$Z_{jk} = [n/(n-1)] (d^2_{jk} - Q^2_{jk})$$

5) Clifford-Stephenson coefficient s_{jk}:

$$s_{jk} = \frac{\sum_{i=1}^{n} |X_{ij} - X_{ik}|}{\sum_{i=1}^{n} (X_{ij} + X_{ik})}$$

6) Canberra metric coefficient c_{jk}:

$$c_{jk} = (1/n) \sum_{i=1}^{n} [|X_{ij} - X_{ik}| / (X_{ij} + X_{ik})]$$

7) Bray-Curtis coefficient b_{jk}:

$$b_{jk} = \frac{2\sum_{i=1}^{n} \min (X_{ij}, X_{ik})}{\sum_{i=1}^{n} (X_{ij} + X_{ik})}$$

When one element of any comparison in the Canberra metric is zero, it is customary to replace zero values with a small arbitrary value (Clifford and Stephenson 1975;58). This arbitrary value is an input option.

SUBROUTINE *SIMQ

Purpose

Used to create a Resemblance Matrix from binary data.

Technicalities

A given block of data from a Data Matrix is transformed into a Resemblance Matrix. The following similarity and dissimilarity coefficients can be used:

Option	Similarity Coefficient	Reference	Range of values
1	Jaccard	Sneath and Sokal 1973;131	0 ↔ 1
2	Sorensen (also called Dice)	Sneath and Sokal 1973;131	0 ↔ 1
3	simple matching	Sneath and Sokal 1973;132	0 ↔ 1
4	Rogers and Tanimoto	Sneath and Sokal 1973;132	0 ↔ 1
5	Yule	Sneath and Sokal 1973;133	-1 ↔ 1
6	Hamann	Sneath and Sokal 1973;133	-1 ↔ 1

Option	Dissimilarity coefficient	Reference	Range of values
7	Russell and Rao	Anderberg 1973;89	0 ↔ 1
8	Sokal and Sneath	Clifford and Stephenson 1975;54	0 ↔ 1
9	Ochiai	Clifford and Stephenson 1975;55	0 ↔ 1
10	Pearson	Clifford and Stephenson 1975;62	-1 ↔ 1
13	Phi	Hohn 1976	-1 ↔ 1
11	Sokal binary distance	Hohn 1976	0 ↔ 1
12	proportion features different	Hohn 1976	0 ↔ 1

Resemblance Coefficient Formulas

The variables in the formulas are a, b, c and d. For objects j and k, a is the number of 1-1 matches, b is the number of 1-0 matches, c is the number of 0-1 matches and d is the number of 0-0 matches.

Object j \ Object k	1	0
1	a	b
0	c	d

Jaccard:
$a/(a + b + c)$

Sorenson:
$2a/(2a + b + c)$

Simple matching:
$(a + d)/(a + b + c + d)$

Rogers and Tanimoto:
$(a + d)/(a + 2b + 2c + d)$

Yule:
$(ad - bc)/(ad + bc)$

Hamann:
$(a + d - b - c)/(a + b + c + d)$

Russell and Rao:
$a/(a + b + c + d)$

Sokal and Sneath:
$2(a + d)/[2(a + d) + b + c]$

Ochiai:

$$\frac{a}{\sqrt{(a + b)(a + c)}}$$

Pearson:
$(ad - bc)/[(a + b)(c + d)(a + c)\ (b + d)]$

Sokal binary distance:

$$\sqrt{(b + c)/(a + b + c + d)}$$

Proportion features different:
$(b + c)/(a + b + c + d)$

Phi:

$$\frac{(ad - bc)}{\sqrt{(a + b)(a + c)(b + d)(c + d)}}$$

Matrix scale entries in a Data Matrix or a Standardized Data Matrix can be transformed to binary values of 0 or 1. All data values less than or equal to a specified value are assigned a value of 0; those greater are given a 1.

Subroutine *CLST

Purpose

Used to produce a tree from a Resemblance Matrix via a clustering method. Computes a Node Count Matrix, Cophenetic Matrix and cophenetic correlation.

Technicalities

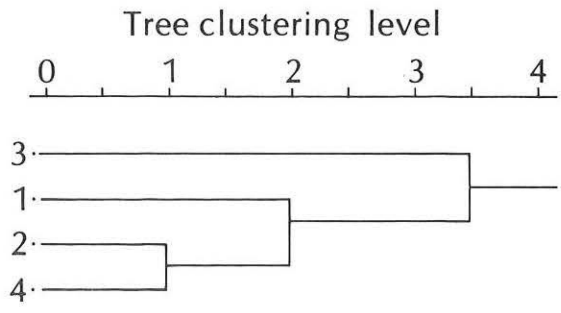
Three clustering methods are given:

1) single linkage
2) complete linkage
3) unweighted pair-group method using arithmetic averages (UPGMA)

For explanations, see Sneath and Sokal (1973;216-240).

The Node Count Matrix contains the number of nodes separating any two objects in the tree. The Cophenetic Matrix contains the level at which any two objects join in the tree.

Following is an example showing the construction of the Node Count Matrix (NCM) and Cophenetic Matrix (CM) from the tree:



Node count matrix

	1	2	3	4
1	—	—	—	—
2	2	—	—	—
3	2	3	—	—
4	2	1	3	—

Cophenetic matrix

	1	2	3	4
1	—	—	—	—
2	2.0	—	—	—
3	3.5	3.5	—	—
4	2.0	1.0	3.5	—

The NCM and CM are alternative representations of the tree. The CM gives a more accurate value of the clustering level than can be read off the scale provided with the tree. The tree can be scaled between upper and lower limits supplied by the user or can, by default, be self-scaling.

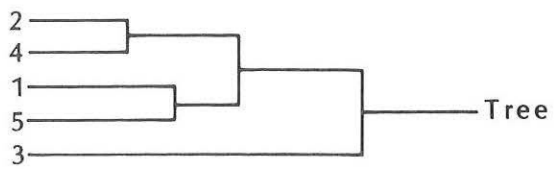
Subroutine *RRNG

Purpose

Used to rearrange a Data Matrix, Standardized Data Matrix or Resemblance Matrix according to the order in which objects are arranged down the left edge of the tree.

Technicalities

An example shows best:



Reading the order of object arrangement from the tree, top to bottom: 2, 4, 1, 5, 3.

Data Matrix:

	1	2	3	4	5
A_1	6	1	7	2	4
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
A_n	1	4	7	3	2

Rearranged Data Matrix:

	2	4	1	5	3
A_1	1	2	6	4	7
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
A_n	4	3	1	2	7

Resemblance Matrix:

	1	2	3	4	5
1	—	—	—	—	—
2	2.1	—	—	—	—
3	2.4	3.6	—	—	—
4	3.0	1.0	3.6	—	—
5	1.7	2.9	3.1	2.9	—

Rearranged Resemblance Matrix:

	2	4	1	5	3
2	—	—	—	—	—
4	1.0	—	—	—	—
1	2.1	3.0	—	—	—
5	2.9	2.9	1.7	—	—
3	3.6	3.6	2.4	3.1	—

Subroutine *COMB

Purpose

Used to combine two Resemblance Matrices into a new Resemblance Matrix.

Technicalities

Suppose you have Resemblance Matrices RM1, RM2 and RM3 and want to combine them to get RM4. Let RM(i,j) be the i,j entry of RM, where i = 2, . . . t and j = 1, 2, . . . t-1, and j<i. Let w_k be the weight given the k'th Resemblance Matrix, such that $\Sigma w_k = 1.0$.

For example, *COMB computes $RM4(i,j) = w_1 \cdot RM1(i,j) + w_2 \cdot RM2(i,j) + w_3 \cdot RM3(i,j)$ for all applicable i,j. Since only two Resemblance Matrices can be combined with *COMB, it is necessary in this example to make two passes through the subroutine. The first pass combines RM1 and RM2 into RM5:

$$RM5 = w_1 \cdot RM1 + w_2 \cdot RM2$$

On the second pass:

$$RM4 = 1.0 \cdot RM5 + w_3 \cdot RM3$$

The cards, using the appropriate format and weights $w_1 = 0.2$, $w_2 = 0.3$, $w_3 = 0.5$, are:

```
Card 1:  *COMB
Card 2:  RM1  RM2  RM5  0.2  0.3  1
Card 1:  *COMB
Card 2:  RM5  RM3  RM4  1.0  0.5  1
```

When all Resemblance Matrices to be combined are similarity (dissimilarity) matrices, the combined RM will be a similarity (dissimilarity) matrix. When the Resemblance Matrices to be combined are a mixture of similarity and dissimilarity matrices, *COMB changes the similarity matrices to dissimilarity matrices by multiplying by —1.0, and these are combined with the remaining dissimilarity matrices to give a combined RM that is a dissimilarity matrix.

*COMB allows standardization of Resemblance matrices. It performs the kind of transformation given in option 3 of *STAN. Let $\overline{X}$ be the mean of the RM (i,j) values and sd be the the standard deviation. If RM(i,j) is the entry, the standardized value is:

$$[RM(i,j) - \overline{X}]/sd$$

The subroutine is told to standardize when the Resemblance Matrix is assigned a weight. The rule is: a negative weight means "don't standardize"; a positive weight means "standardize." Suppose in the example we want to standardize the three matrices before combining. The cards appear as:

```
Card 1:  *COMB
Card 2:  RM1  RM2  RM5  0.2  0.3  1
Card 1:  *COMB
Card 2:  RM5  RM3  RM4  —1.0  0.5  1
```

The reason for the —1.0 is that RM1 and RM2 have already been standardized and therefore should not be standardized again.

Subroutine *CMPR

Purpose

Used to compute three indices of comparison between two trees.

Technicalities

Index 1: 2D/[K(K—1)]. See Williams and Clifford (1971; 521) for discussion of this measure. For a given pair of objects j and k, we compute the number of nodes separating them in each of the two trees, e.g. m_1 and m_2. We form the difference $d = m_1 - m_2$. We have K objects, and this is done for all K(K—1)/2 pairs of objects. We form the sums Σd_+ and Σd_-, that is, the sum of those d's that are positive and the sum of those that are negative. From this compute $D = |\Sigma d_+ + \Sigma d_-|$, where $|\cdot|$ indicates absolute value. Our index is 2D/[K(K—1)]. The closer this is to zero, the greater the conformity of the two trees.

Index 2: The subroutine finds the Node Count Matrix for each of the two trees. These matrices are "strung" into two lists (á la the computing of the cophenetic correlation) and correlated using the produce moment correlation formula r. This is called the "node correlation" on the output.

Index 3: The Cophenetic Matrix is obtained for each of the two trees and these are correlated using r. This is called the "tree correlation" on the output.

Subroutine *PNCH

Purpose

Used to punch a Data Matrix, Standardized Data Matrix or Resemblance Matrix.

Technicalities

A Data Matrix or a Standardized Data Matrix is punched in transposed form, i.e., the objects are the row, the attributes and the columns.

A Resemblance Matrix is punched in lower triangular form.

The format is [I5, 8F9.4/(5X, 8F9.4)] where I5 is for the object number and F9.4 is the datum.

Subroutine *DUMP

Purpose

Used to dump a matrix. This subroutine was used as a debug aid when CLUSTAR was developed. Users who modify the program may find it of value.

Subroutine *INST

Purpose

Gives a complete but textually abridged listing of the subroutine formats.

Subroutine *END

Purpose

Plays the role of "That's all folks."

CLUSTAR SUBROUTINE FORMATS

*INPT

Format

Card 1: *INPT in columns 1-5.

Card 2:

Columns	Variable Name	Description
7-10	ANAME	Input matrix name
14-15	NTYP	Type of matrix 0 Data Matrix —1 Similarity Matrix 1 Dissimilarity Matrix
18-20	NOBJ	
21-25	NATR	Number of attributes (Use only if NTYP = 0)
30	NDIR	Direction of input (USE only if NTYP = 0) 0 read Data Matrix by columns 1 read Data Matrix by rows

34-35	NOUT	Output control 0 don't print matrix 1 print Data Matrix on Resemblance Matrix 2 print transposed Data Matrix (Use only if NTYP = 0)
36-40	VM	Missing value code. Use format F5.0

Card 3: Left parenthesis in column 1 followed by right parenthesis

Card(s) 4: Data card(s)

*STAN

Format

Card 1: *STAN in columns 1-5.

Card 2:

Columns	Variable name	Description
7-10	ANAME	Data Matrix name to be standardized
12-15	BNAME	Standardized Data Matrix name
20	NSTAN	Standardization method 1 $X_{ij} - \overline{X}_i$ 2 X_{ij}/sd_i 3 $(X_{ij} - \overline{X}_i)/sd_i$ 4 $X_{ij}/\max(X_{ij})$ 5 $[X_{ij} - \min(X_{ij})]/[\max(X_{ij}) - \min(X_{ij})]$
21-25	NOUT	Output control 0 don't print Standardized Data Matrix (SDM) 1 print SDM 2 print transposed SDM
26-30	NA1	Attribute number defining the start of a block of data (blank value defaults to 1)
31-35	NA2	Attribute number defining the end of a block of data (blank value defaults to n, the total number of attributes)

*SIMI

Format

Card 1: *SIMI in columns 1-5.

Card 2:

Columns	Variable Name	Description
7-10	ANAME	Name of Data Matrix or Standardized Data Matrix
12-15	BNAME	Resemblance Matrix name
20	NTYP	Resemblance Matrix coefficient 1 correlation r_{jk} 2 average Euclidean distance d_{jk} 3 cost Θ_{jk} 4 coefficient of shape difference z_{jk} 5 Clifford-Stephenson coefficient s_{jk} 6 Canberra metric coefficient c_{jk} 7 Bray-Curtis coefficient b_{jk}

21-25	NOUT	Output control 0 don't print RM 1 print RM
26-30	SMALL	Small value used in Canberra metric coefficient. Use format F5.0
31-35	NA1	Attribute number defining the start of a block of data for which RM is to be computed (blank value defaults to 1). Not needed when computing RM from SDM
36-40	NA2	Attribute number defining the end of a block of data for which RM is to be computed (blank value defaults to 1). Not needed when computing RM from SDM

*SIMQ

Format

Card 1: *SIMQ in columns 1-5.

Card 2:

Columns	Variable Name	Description
7-10	ANAME	Data Matrix or Standardized Data Matrix name
12-15	BNAME	Resemblance Matrix name
19-20	NTYP	Binary resemblance coefficient 1 Jaccard 2 Sorensen 3 simple matching 4 Rogers and Tanimoto 5 Yule 6 Hamann 7 Russell and Rao 8 Sokal and Sneath 9 Ochiai 10 Pearson 11 Sokal binary distance 12 proportion features different 13 phi
25	NOUT	Output control 0 don't print RM 1 print RM
26-30	ZERO	Value for transformation to binary. Attribute value transformed to one if greater than this value (format F5.0)
31-35	NA1	Attribute number defining the start of a block of data for which RM is to be computed (blank value defaults to 1). Not needed when computing RM from SDM
36-40	NA2	Attribute number defining the end of a block of data for which RM is to be computed (blank value defaults to n, the total number of attributes). Not needed when RM is computed from SDM

*CLST

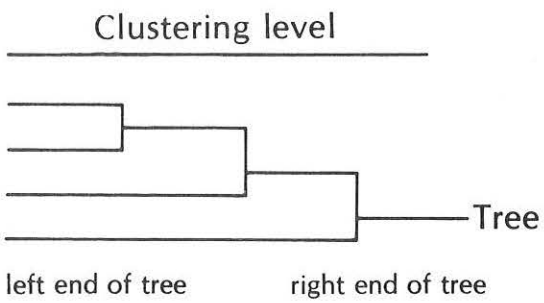
Format

Card 1: *CLST in columns 1-5.

Card 2:

Columns	Variable Name	Description
7-10	ANAME	Resemblance Matrix name
12-15	BNAME	Tree name
20	ITYP	Clustering method 1 single linkage 2 complete linkage 3 UPGMA
25	INODE	Node Count Matrix 0 don't print 1 print
29-30	ICOPH	Cophenetic Matrix —1 cophenetic correlation only 0 don't print CM or cophenetic correlation 1 print CM and cophenetic correlation
31-35	PMIN	Enter value of left end of tree
36-40	PMAX	Enter value right end of tree

where:



If RM is a similarity matrix, PMIN <PMAX. If RM is a dissimilarity matrix, PMAX< PMIN. PMIN or PMAX or both can be left blank, in which case the computer will supply the values.

*RRNG

Format

Card 1: *RRNG in columns 1-5.

Card 2:

Columns	Variable Name	Description
7-10	ANAME	Name of Data Matrix, Standardized Data Matrix or Resemblance Matrix to be rearranged.
12-15	BNAME	Tree name
20	NOUT	Direction DM or SDM is to be printed. 0 not transposed (objects as columns) 1 transposed (objects as rows)

*COMB

Format

Card 1: *COMB in columns 1-5

Card 2:

Columns	Variable Name	Description
7-10	ANAME	First Resemblance Matrix name
12-15	BNAME	Second Resemblance Matrix name
17-20	CNAME	Combined Resemblance Matrix name (a dissimilarity matrix)
21-25	WTA	Weight for first RM (if weight < 0, RM is not to be standardized)
26-30	WTB	Weight for second RM (if weight<0, RM is not to be standardized)
35	NOUT	Output control 0 don't print combined RM 1 print combined RM

*CMPR

Format

Card 1: *CMPR in columns 1-5

Card 2:

Columns	Variable Name	Description
7-10	ANAME	First tree name
12-15	BNAME	Second tree name

*PNCH

Format

Card 1: *PNCH in columns 1-5

Card 2:

Columns	Variable Name	Description
7-10	ANAME	Matrix name

*DUMP

Format

Card 1: *DUMP in columns 1-5

Card 2:

Columns	Variable Name	Description
7-10	ANAME	Matrix name

Columns	Variable name	Description
15		Direction of print (Data Matrix only) 0 transposed 1 not transposed

*INST

Format

Card 1: *INST in columns 1-5

*END

Format

Card 1: *END in columns 1-4

HOW TO PREPARE CLUSTID INPUT CARDS

The CLUSTID subroutines are named: *INPT, *STAN, *SIMI, *SIMQ, *IDEN, *COMB, *DUMP, *INST and *END. Some of these subroutines have the same names as CLUSTAR subroutines, but their logic is not the same. Their names are the same because at a more general level their purposes are the same (e.g., *INPT enters the input for each program, *STAN standardizes the input for each). The purpose and technicalities of the subroutines are given in the following section. The section ends with the subroutine formats.

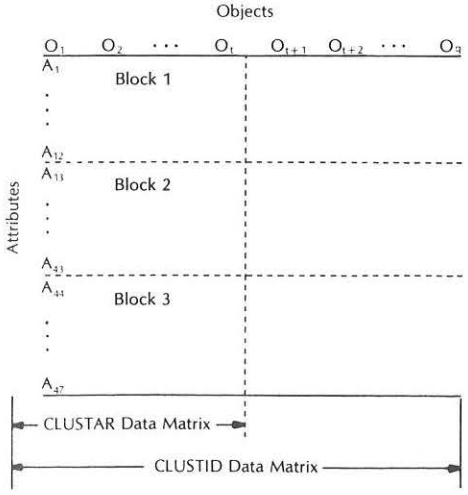
SUBROUTINE *INPT

Purpose

Used to input an augmented Data Matrix.

Technicalities

The n by t Data Matrix input to CLUSTAR is augmented by adding columns t + 1, t + 2, . . . q. The objects t + 1, t + 2, . . . q are objects to be identified and are called "unknowns," meaning the cluster to which each belongs is not known. In CLUSTID the augmented Data Matrix is referred to as the "Data Matrix." The layout is:



Note that the attributes of the unknown objects must be partitioned exactly as the partitioning of the known objects' attributes. Note that whereas CLUSTAR allows a previously computed Resemblance Matrix to be used, CLUSTID does not.

SUBROUTINE *STAN

Purpose

Used to standardize the augmented Data Matrix.

Technicalities

If the original CLUSTAR run used to define the clusters included *STAN, then *STAN with the same option settings for standardization must be in the CLUSTID run. The parameters for the standardization are computed using the t objects in the CLUSTAR run, and these are applied to the t objects as well as the unknown objects t + 1, t + 2, . . . q. Thus, the unknown objects are standardized using parameters produced by the original CLUSTAR run. As with *STAN in CLUSTAR, the user can specify which blocks of data are to be standardized and, of course, this must follow the steps used with *STAN in CLUSTAR.

SUBROUTINE *SIMI

Purpose

Used to create a Likeness Matrix from metric data.

Technicalities

A given block of metric data from either a Data Matrix or Standardized Data Matrix is transformed into a Likeness Matrix. The Likeness Matrix contains the similarity (or dissimilarity) coefficients between each of the original t objects and the q—t unknown objects. Whatever similarity coefficient or dissimilarity coefficient was used in *SIMI in CLUSTAR should be selected here also.

SUBROUTINE *SIMQ

Purpose

Used to create a Likeness Matrix from binary data.

Technicalities

A given block of binary data from a Data Matrix is transformed into a Likeness Matrix. The Likeness Matrix contains the similarity (or dissimilarity) coefficients between each of the original t objects and the q-t unknown objects. Whatever similarity coefficient or dissimilarity coefficient was used in *SIMQ in CLUSTAR should be selected here.

SUBROUTINE *COMB

Purpose

Used to combine two Likeness Matrices.

Technicalities

If *COMB was used in the original CLUSTAR run to combine two Resemblance Matrices, then it must be used in the CLUSTID run to combine the two corresponding Likeness Matrices.

This is described by building on the example in the CLUSTAR subroutine *COMB section of this manual. Recall that on the first pass through *COMB, RM1 and RM2 are combined into RM5 using weights w_1 and w_2:

$$RM5 = w_1 \cdot RM1 + w_2 \cdot RM2$$

On the second pass,

$$RM4 = 1.0 \cdot RM5 + w_3 \cdot RM3$$

Corresponding to each input Resemblance Matrix RM1, RM2 and RM3 is a Likeness Matrix LM1, LM2 and LM3. If the above example were used in *COMB of CLUSTAR, we would want *COMB in CLUSTID to calculate:

$$LM5 = w_1 \cdot LM1 + w_2 \cdot LM2$$

On the second pass,

$$LM4 = 1.0 \cdot LM5 + w_3 \cdot LM3$$

The weights w_i used in *COMB in CLUSTID must be the same values used in *COMB in CLUSTAR. If a Resemblance Matrix was standardized in the combining process in CLUSTAR, then the corresponding Likeness Matrix must be standardized here. The mean ($\overline{X}$) and standard deviation (sd) computed from the given Resemblance Matrix is used to standardize the Likeness Matrix. Thus, the user must supply $\overline{X}$ and sd for each Likeness Matrix that is standardized by transcribing them from the printout for the original CLUSTAR run. The cards for the above example using the appropriate format and weights $w_1 = 0.2$, $w_2 = 0.3$, $w_3 = 0.5$ and $\overline{X}_1 = 2.1$, $sd_1 = 1.6$, $\overline{X}_2 = 1.1$, $sd_2 = 0.80$, $\overline{X}_3 = 1.9$, $sd_3 = 3.7$, are:

```
Card Column →      5     10    15    20    25    30    35
                   ↓     ↓     ↓     ↓     ↓     ↓     ↓
Card 1:       *COMB
Card 2:             LM1   LM2   LM5   0.2   0.3     1
Card(s)3:                 2.1                     1.6
                          1.1                    0.80
Card 1:       *COMB
Card 2:             LM5   LM3   LM4   -1.0  0.5     1
Card 3:                   1.9                     3.7
```

Card(s) 3 contains $\overline{X}$ and sd for each Likeness Matrix that needs standardization. In this example, LM5 does not get standardized (note weight of −1.0 in second pass) and thus Card 3 contains only one card giving $\overline{X}$ and sd for LM3.

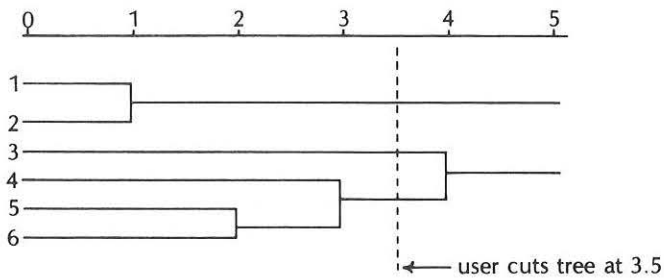
SUBROUTINE *IDEN

Purpose

Used to compute the Resemblance Coefficient between each unknown object and each cluster defined by the user.

Technicalities

From the CLUSTAR run, the user defines the clusters (m in number) by indicating which of the t objects they obtain. Suppose the tree from CLUSTAR appears as:



The user cuts the tree at a clustering level of 3.5. The resulting clusters are:

Cluster 1: {1,4}
Cluster 2: {6}
Cluster 3: {2, 3, 5}

Suppose further we have an unknown object with the identification number 7 and that average Euclidean distance d_{jk} was used to calculate the Resemblance Matrix and, hence, the Likeness Matrix.

Likeness Matrix

1	d_{17}
2	d_{27}
3	d_{37}
4	d_{47}
5	d_{57}
6	d_{67}

To which cluster is object 7 most similar? Under the single linkage criterion (option 1), we compute:

Cluster 1: min $\{d_{17}, d_{47}\}$

Cluster 2: min $\{d_{67}\}$
Cluster 3: min $\{d_{27}, d_{37}, d_{57}\}$

and assign object 7 to the cluster with the smallest of the three values.

Under complete linkage criterion (option 3), we compute:

Cluster 1: max $\{d_{17}, d_{47}\}$
Cluster 2: max $\{d_{67}\}$
Cluster 3: max $\{d_{27}, d_{37}, d_{57}\}$

and assign object 7 to the cluster with the smallest of the three values.

Under UPGMA (option 3), we compute:

Cluster 1: $(d_{17} + d_{47})/2$
Cluster 2: d_{67}
Cluster 3: $(d_{27} + d_{37} + d_{57})/3$

and assign object 7 to the cluster with the smallest of the three values.

The clustering option used in CLUSTAR (single linkage, complete linkage or UPGMA) for the original problem should be the same criterion used in *IDEN in CLUSTID.

The output of *IDEN is a matrix of resemblance coefficients between clusters (rows) and unknown objects (columns). For each unknown object, its most similar cluster is identified by finding the smallest (largest when similarity coefficients are used) value in the given column; the corresponding row identifies the cluster. *IDEN does this automatically, printing out this information below the heading "Cluster Unknown Identified With."

Subroutine *DUMP

Purpose

Used to dump a matrix. This subroutine was used as a debug aid when CLUSTID was developed. Users who modify the program may find it of value.

Subroutine *INST

Purpose

Gives a complete but textually abridged listing of the subroutine formats.

Subroutine *END

Purpose

Plays the role of, "That's all folks."

CLUSTID SUBROUTINE FORMATS

*INPT

Format

Card 1: *INPT in columns 1-5

Card 2:

Columns	Variable Name	Description
7-10	ANAME	Input matrix name.
14-15	NTYP	Type of matrix 0 Data Matrix
18-20	NOBJ	Number of objects
21-25	NATR	Number of attributes
30	NDIR	Direction of input 0 read Data Matrix by columns 1 read Data Matrix by rows
34-35	NOUT	Output control 0 don't print matrix 1 print Data Matrix or Resemblance Matrix 2 print transposed Data Matrix (use only if NTYP = 0)
36-40	VM	Missing value code. Use format F5.0
43-45	NUNK	Number of unknown objects

Card 3: Left parenthesis in column 1 followed by FORTRAN format followed by right parenthesis

Card(s) 4: Data card(s)

*STAN

Format

Card 1: *STAN in columns 1-5

Card 2:

Columns	Variable Name	Description
7-10	ANAME	Data Matrix name to be standardized
12-15	BNAME	Standardized Data Matrix name
20	NSTAN	Standardization method 1 $X_{ij} - \bar{X}_i$ 2 X_{ij}/sd_i 3 $(X_{ij} - \bar{X}_i)/sd_i$ 4 $X_{ij}/\max(X_{ij})$ 5 $[X_{ij} - \min(X_{ij})]/[\max(X_{ij}) - \min(X_{ij})]$
21-25	NOUT	Output control 0 don't print Standardized Data Matrix 1 print SDM 2 print transposed SDM
26-30	NA1	Attribute number defining the start of a block of data (blank value defaults to 1)
31-35	NA2	Attribute number defining the end of a block of data (blank value defaults to n, the total number of attributes)

*SIMI

Format

Card 1: *SIMI in columns 1-5

Card 2:

Columns	Variable Name	Description
7-10	ANAME	Name of Data Matrix or Standardized Data Matrix
12-15	BNAME	Likeness Matrix name
20	NTYPE	Likeness Matrix coefficient 1 correlation r_{jk} 2 average Euclidean distance d_{jk} 3 cos Θ_{jk} 4 coefficient of shape difference z_{jk} 5 Clifford-Stephenson coefficient s_{jk} 6 Canberra metric coefficient c_{jk} 7 Bray-Curtis coefficient b_{jk}
21-25	NOUT	Output control 0 don't print LM 1 print LM

26-30	SMALL	Small value used in Canberra metric coefficient (format F5.0)
31-35	NA1	Attribute number defining the start of a block of data for which LM is to be computed (blank value defaults to 1). Not needed when computing LM from SDM

*SIMQ

Format

Card 1: *SIMQ in columns 1-5

Card 2:

Columns	Variable Name	Description
7-10	ANAME	Data Matrix or Standardized Data Matrix name
12-15	BNAME	Likeness Matrix name
19-20	NTYPE	Binary resemblance coefficient 1 Jaccard 2 Sorensen 3 simple matching 4 Rogers and Tanimoto 5 Yule 6 Hamann 7 Russell and Rao 8 Sokal and Sneath 9 Ochiai 10 Pearson 11 Sokal binary distance 12 proportion features different 13 phi
25	NOUT	Output control 0 don't print LM 1 print LM
26-30	ZERO	Value for transformation to binary (attribute value transformed to 1 if greater than this value—format F5.0)
31-35	NA1	Attribute number defining the start of a block of data for which LM is to be computed (blank value defaults to one). Not needed when computing LM from SDM
36-40	NA2	Attribute number defining the end of a block of data for which LM is to be computed (blank value defaults to n, the total number of attributes). Not needed when LM is computed from SDM

*IDEN

Format

Card 1: *IDEN in columns 1-5

Card 2:

Columns	Variable Name	Description
7-10	ANAME	Likeness Matrix name

13-15	NCLUS	Number of clusters (m) determined from tree in CLUSTAR
20	ITYP	Identification method 1 single linkage 2 complete linkage 3 UPGMA

Card(s) 3:

1-4	IA (array)	Number of objects in cluster 1
5-8	IA (array)	Object number of first object in cluster 1
9-12	IA (array)	Object number of second object in cluster 1
13-16	IA (array)	Object number of third object in cluster 1
.		
.		
.		
etc.		

After this is done for cluster 1 begin the procedure again for cluster 2 starting in columns 1-4. Continue until data for all n clusters are entered.

*COMB

Format

Card 1: *COMB in columns 1-5

Card 2:

Columns	Variable Name	Description
7-10	ANAME	First Likeness Matrix name
12-15	BNAME	Second Likeness Matrix name
17-20	CNAME	Combined Likeness Matrix name (a dissimilarity matrix)
21-25	WTA	Weight for first LM (if weight <0, LM is not to be standardized)
26-30	WTB	Weight for second LM (if weight <0, LM is not to be standardized)
35	NOUT	Output control 0 don't print combined LM 1 print combined LM

Card(s) 3:

1-15	SA, SB	Mean of values in corresponding Resemblance Matrix from CLUSTAR run (format E15.6)
16-30	AS, BS	Standard deviation of values in corresponding Resemblance Matrix from CLUSTAR run (format E15.6)

The order of card type 3 corresponds to WTA first, WTB second. If either WTA or WTB is negative, then the corresponding card type 3 should not be placed in the deck. Thus, if WTA is negative but WTB positive, then only the card type 3 for WTB is entered into the deck.

*DUMP

Format

Card 1: *DUMP in columns 1-5

*INST

Format

Card 1: *INST in columns 1-5

*END

Format

Card 1: *END in columns 1-4

EXAMPLES

EXAMPLE 1: CLUSTAR RUN

This is the problem illustrated in Figure 1. *SIMI is used to compute the Resemblance Matrix (average Euclidean distance).

*CLST executes UPGMA clustering.

*RRNG produces the trellis diagram.

Input Deck Listing

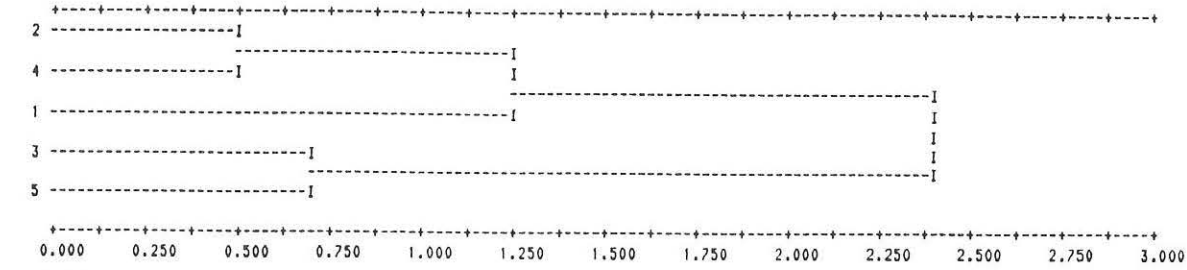
```
*INPT
      DAT          5    2    1    1
(5F5.1)
  2.0  3.5  1.0  4.0  2.0
  4.0  3.5  1.0  4.0  1.0
*SIMI
      DAT    RM    2    1
*CLST
       RM NICE    3    0    1  0.0  3.0
*RRNG
       RM NICE    0
*END
```

Output Listing

```
- - - - - - DATA MATRIX - - - - - -
INPUT FORMAT : (5F5.1)
MATRIX NAME : DAT
TYPE OF MATRIX : DATA
NUMBER OF OBJECTS :    5
NUMBER OF ATTRIBUTES :    2
MISSING VALUE CODE :    0.
OUTPUT OPTION  :    1
- - - - - - DATA MATRIX - - - - - -

          1        2        3        4        5
1    2.0000   3.5000   1.0000   4.0000   2.0000
2    4.0000   3.5000   1.0000   4.0000   1.0000



- - - - RESEMBLANCE MATRIX FOR METRIC DATA - - - -
DATA MATRIX NAME : DAT
RESEMBLANCE MATRIX NAME :   RM
USE ATTRIBUTES  1  THRU  2
OUTPUT OPTION :  1
RESEMBLANCE COFFICIENT : AVERAGE EUCLIDEAN DISTANCE
- - - - RESEMBLANCE MATRIX FOR METRIC DATA - - - -

          1        2        3        4
2    1.1180
3    2.2361   2.5000
4    1.4142   0.5000   3.0000
5    2.1213   2.0616   0.7071   2.5495
```

Output listing, continued

```
- - - - - - - TREE - - - - - - -
RESEMBLANCE MATRIX NAME :   RM
TREE NAME : NICE
NODE COUNT OPTION :  0
COPHENETIC OPTION :  1
CLUSTERING METHOD : UPGMA
MINIMUM VALUE ON TREE :      0.0000
MAXIMUM VALUE ON TREE :      3.0000
- - - - - - - TREE - - - - - - -
```




```
COPHENETIC CORRELATION MATRIX
          1         2         3         4
2    1.2661
3    2.4114    2.4114
4    1.2661    0.5000    2.4114
5    2.4114    2.4114    0.7071    2.4114
COPHENETIC CORRELATION =      0.9470


- - - - - - REARRANGED DATA - - - - - -
NAME OF MATRIX TO BE REARRANGED :   RM
TREE NAME : NICE
OUTPUT OPTION :  0
- - - - - - REARRANGED DATA - - - - - -

          2         4         1         3
4    0.5000
1    1.1180    1.4142
3    2.5000    3.0000    2.2361
5    2.0616    2.5495    2.1213    0.7071
```

Example 2: CLUSTAR Run

The Data Matrix contains t = 7 objects and n = 10 attributes. Attributes 1-4 are scaled metrically and attributes 5-10 are binary.

*STAN is used to standardize the partition of the Data Matrix containing attributes 1-4.

*SIMI is used to compute a Resemblance Matrix (average Euclidean distance) from the Standardized Data.

*SIMQ is used to compute a Resemblance Matrix (Jaccard Coefficient) from the binary data in the Data Matrix.

*CLST is used with Resemblance Matrix RM1.

*CLST is used with Resemblance Matrix RM2.

*COMB is used to combine RM1 and RM2, producing RM3.

*CLST is used with Resemblance Matrix RM3.

Input Deck Listing

```
*INPT
      DATA    0    7   10    1    1 999.
(7F5.0)
  1.2  1.6  9.2  1.1  0.5  8.6  1.4
  3.7  3.4  1.2  1.2  6.2  1.4  5.8
  1.7  6.8  3.7  0.5  1.5 999.  1.2
  4.3  0.5  3.9  6.7  3.1 999.  3.1
    1    1    0    1    1    1    1
    1    1    0    1    0    0    0
    1    1    0 999.    1    0    1
    0    0    1    0    0    1    1
    1    1    1    1    0    1    0
    0    0    0    1    1    0    1
*STAN
      DATA DATS    3    1    1    4
*SIMI
      DATS  RM1    2    1
*CLST
       RM1 TRE1    3    1    1
*SIMQ
      DATA  RM2    1    1         5   10
*CLST
       RM2 TRE2    3    1    1  1.0  0.0
*COMB
       RM1  RM2  RM3  0.5  0.2    1
*CLST
       RM3 TRE3    3    1    1  1.0  0.0
*END
```

Output Listing

```
- - - - - - DATA MATRIX - - - - - -
INPUT FORMAT : (7F5.0)
MATRIX NAME : DATA
TYPE OF MATRIX : DATA
NUMBER OF OBJECTS :    7
NUMBER OF ATTRIBUTES :   10
MISSING VALUE CODE :  999.
OUTPUT OPTION  :    1
- - - - - - DATA MATRIX - - - - - -
```

	1	2	3	4	5	6	7
1	1.2000	1.6000	9.2000	1.1000	0.5000	8.6000	1.4000
2	3.7000	3.4000	1.2000	1.2000	6.2000	1.4000	5.8000
3	1.7000	6.8000	3.7000	0.5000	1.5000	999.0000	1.2000
4	4.3000	0.5000	3.9000	6.7000	3.1000	999.0000	3.1000
5	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000	1.0000
6	1.0000	1.0000	0.0000	1.0000	0.0000	0.0000	0.0000
7	1.0000	1.0000	0.0000	999.0000	1.0000	0.0000	1.0000
8	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	1.0000
9	1.0000	1.0000	1.0000	1.0000	0.0000	1.0000	0.0000
10	0.0000	0.0000	0.0000	1.0000	1.0000	0.0000	1.0000

```
- - - - STANDARDIZED DATA MATRIX - - - -
DATA MATRIX NAME : DATA
STANDARDIZED DATA MATRIX NAME : DAT6
STANDARDIZATION OPTION :    3
USE ATTRIBUTES  1  THRU  4
OUTPUT OPTION :  1
- - - - STANDARDIZED DATA MATRIX - - - -
```

	1	2	3	4
MIN	0.5000	1.2000	0.5000	0.5000
MAX	9.2000	6.2000	6.8000	6.7000
NOB	7.0000	7.0000	6.0000	6.0000
MEA	3.3714	3.2714	2.5667	3.6000
S.D	3.7959	2.1313	2.3338	2.0149

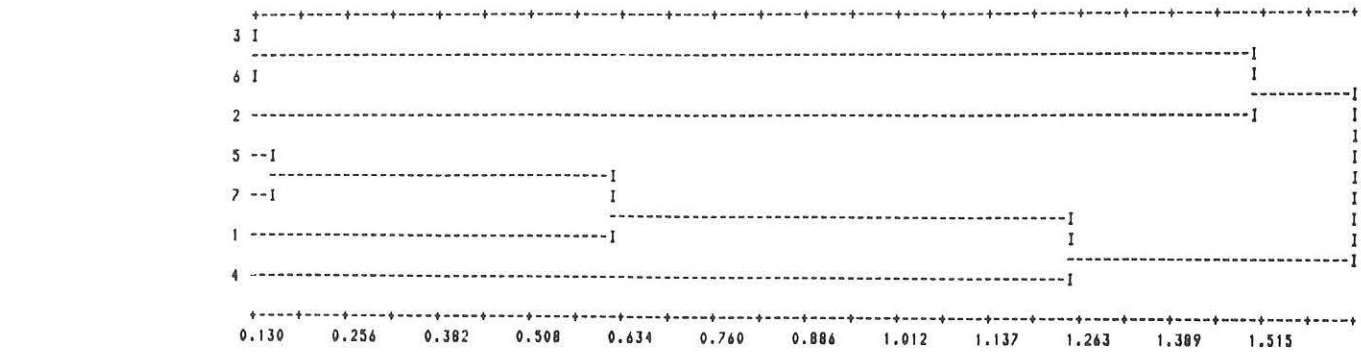
	1	2	3	4	5	6	7
1	-0.5720	-0.4667	1.5355	-0.5984	-0.7565	1.3774	-0.5194
2	0.2011	0.0603	-0.9719	-0.9719	1.3741	-0.8781	1.1864
3	-0.3714	1.8139	0.4856	-0.8855	-0.4570	999.0000	-0.5856
4	0.3474	-1.5385	0.1489	1.5385	-0.2481	999.0000	-0.2481

```
- - - - RESEMBLANCE MATRIX FOR METRIC DATA - - - -
DATA MATRIX NAME : DAT5
RESEMBLANCE MATRIX NAME :  RM1
USE ATTRIBUTES  1  THRU  4
OUTPUT OPTION :  1
RESEMBLANCE COFFICIENT : AVERAGE EUCLIDEAN DISTANCE
- - - - RESEMBLANCE MATRIX FOR METRIC DATA - - - -
```

	1	2	3	4	5	6
2	1.4459					
3	1.2837	1.5561				
4	0.8746	2.1117	1.4461			
5	0.6656	1.4690	1.7178	1.4920		
6	1.5756	1.4631	0.1300	1.3987	2.1938	
7	0.5861	1.4742	1.5958	1.4095	0.1643	1.9824

Output listing, continued

```
- - - - - - - TREE - - - - - - -
RESEMBLANCE MATRIX NAME :  RM1
TREE NAME : TRE1
NODE COUNT OPTION :  1
COPHENETIC OPTION :  1
CLUSTERING METHOD : UPGMA
MINIMUM VALUE ON TREE :      0.1300
MAXIMUM VALUE ON TREE :      1.6412
- - - - - - - TREE - - - - - - -
```



NODE COUNT MATRIX

	1	2	3	4	5	6
2	4.0					
3	5.0	2.0				
4	2.0	3.0	4.0			
5	2.0	5.0	6.0	3.0		
6	5.0	2.0	1.0	4.0	6.0	
7	2.0	5.0	6.0	3.0	1.0	6.0

COPHENETIC CORRELATION MATRIX

	1	2	3	4	5	6
2	1.6412					
3	1.6412	1.5096				
4	1.2587	1.6412	1.6412			
5	0.6259	1.6412	1.6412	1.2587		
6	1.6412	1.5096	0.1300	1.6412	1.6412	
7	0.6259	1.6412	1.6412	1.2587	0.1643	1.6412

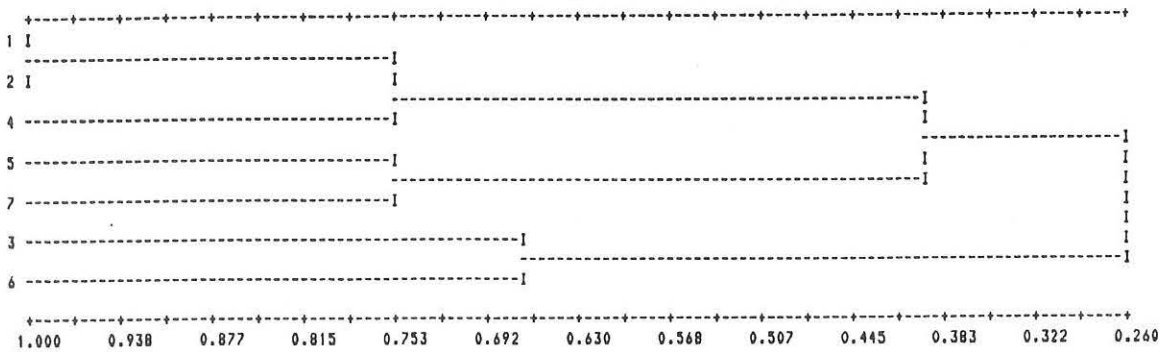
COPHENETIC CORRELATION = 0.8980

```
- - - - RESEMBLANCE MATRIX FOR BINARY DATA - - - -
DATA MATRIX NAME : DATA
RESEMBLANCE MATRIX NAME :  RM2
RESEMBLANCE COFFICIENT : JACCARD
VALUE FOR CONVERTING
METRIC TO BINARY DATA :    0.
USE ATTRIBUTES  5  THRU 10
OUTPUT OPTION :  1
- - - - RESEMBLANCE MATRIX FOR BINARY DATA - - - -
```

	1	2	3	4	5	6
2	1.0000					
3	0.2000	0.2000				
4	0.7500	0.7500	0.2000			
5	0.4000	0.4000	0.0000	0.5000		
6	0.4000	0.4000	0.6667	0.4000	0.2000	
7	0.3333	0.3333	0.2000	0.4000	0.7500	0.4000

```
- - - - - - - TREE - - - - - - -
RESEMBLANCE MATRIX NAME :  RM2
TREE NAME : TRE2
NODE COUNT OPTION :  1
COPHENETIC OPTION :  1
CLUSTERING METHOD : UPGMA
MINIMUM VALUE ON TREE :      0.2600
MAXIMUM VALUE ON TREE :      1.0000
- - - - - - - TREE - - - - - - -
```

Output listing, continued



NODE COUNT MATRIX

	1	2	3	4	5	6
2	1.0					
3	5.0	5.0				
4	2.0	2.0	4.0			
5	4.0	4.0	4.0	3.0		
6	5.0	5.0	1.0	4.0	4.0	
7	4.0	4.0	4.0	3.0	1.0	4.0

COPHENETIC CORRELATION MATRIX

	1	2	3	4	5	6
2	1.0000					
3	0.2600	0.2600				
4	0.7500	0.7500	0.2600			
5	0.3944	0.3944	0.2600	0.3944		
6	0.2600	0.2600	0.6667	0.2600	0.2600	
7	0.3944	0.3944	0.2600	0.3944	0.7500	0.2600

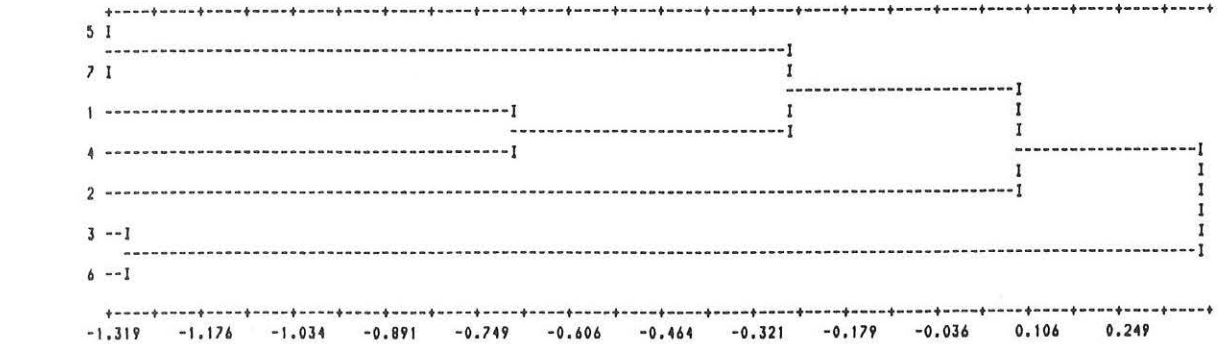
COPHENETIC CORRELATION = 0.9183

- - - - - COMBINE RESEMBLANCE MATRICIES - - - - -
INPUT RESEMBLANCE MATRICIES : RM1 RM2
OUTPUT RESEMBLANCE MATRIX : RM3
WEIGHTING FACTORS : 0.5000 0.2000
OUTPUT OPTION : 1
- - - - COMBINE RESEMBLANCE MATRICIES - - - -

MEAN	S.D.
.133505E+01	.558575E+00
.423016E+00	.241493E+00

	1	2	3	4	5	6
2	-0.3706					
3	0.1387	0.3826				
4	-0.6830	0.4244	0.2841			
5	-0.5802	0.1390	0.6929	0.0767		
6	0.2344	0.1337	-1.2805	0.0760	0.9534	
7	-0.5961	0.1989	0.4181	0.0857	-1.3188	0.5985

- - - - - - - TREE - - - - - - -
RESEMBLANCE MATRIX NAME : RM3
TREE NAME : TRE3
NODE COUNT OPTION : 1
COPHENETIC OPTION : 1
CLUSTERING METHOD : UPGMA
MINIMUM VALUE ON TREE : -1.3188
MAXIMUM VALUE ON TREE : 0.3912
- - - - - - - TREE - - - - - - -



Output listing, continued

```
NODE COUNT MATRIX
          1         2         3         4         5         6
2       3.0
3       5.0       3.0
4       1.0       3.0       5.0
5       3.0       3.0       5.0       3.0
6       5.0       3.0       1.0       5.0       5.0
7       3.0       3.0       5.0       3.0       1.0       5.0

COPHENETIC CORRELATION MATRIX
          1         2         3         4         5         6
2    0.0959
3    0.3912    0.3912
4   -0.6830    0.0959    0.3912
5   -0.2535    0.0959    0.3912   -0.2535
6    0.3912    0.3912   -1.2805    0.3912    0.3912
7   -0.2535    0.0959    0.3912   -0.2535   -1.3188    0.3912
COPHENETIC CORRELATION =    0.8886
```

EXAMPLE 2: CLUSTID RUN

The Data Matrix contains t = 7 objects, q — t = 3 unknown objects, and n = 10 attributes. Attributes 1-4 are metric and attributes 5-10 are binary.

*STAN is used to standardize the partition of the Data Matrix containing attributes 1-4. $\overline{X}_i$ and sd_i are computed using data for objects 1-7.

*SIMI is used to compute the likeness coefficients (average Euclidean distance) from the Standardized Data Matrix. Likeness Matrix is named LM1.

*SIMQ is used to compute the likeness coefficients (Jaccard coefficient) from the binary data in the Data Matrix. The Likeness Matrix is named LM2.

*COMB is used to combine Likeness Matrices LM1 and LM2 to form LM3. LM1 and LM2 are standardized using the means and standard deviations produced by *COMB in CLUSTAR. These values must be entered by hand into *COMB in CLUSTID.

*IDEN is used to define the clusters and to compute the likeness coefficient between clusters and the unknown objects.

Input Deck Listing

```
*INPT
      DATA    0    7   10    1    1 999.    3
(7F5.0/(16F5.0))
  1.2  1.6  9.2  1.1  0.5  8.6  1.4
  0.7  1.0  5.5
  3.7  3.4  1.2  1.2  6.2  1.4  5.8
  3.5  1.4  2.7
  1.7  6.8  3.7  0.5  1.5 999.  1.2
  5.0  1.4 999.
  4.3  0.5  3.9  6.7  3.1 999.  3.1
  4.0  2.2  6.0
    1    1    0    1    1    1    1
    1    1    1
    1    1    0    1    0    0    0
    0    1    0
    1    1    0 999.    1    0    1
    1    0    1
    0    0    1    0    0    1    1
    0    0    1
    1    1    1    1    0    1    0
    1    1    0
    0    0    0    1    1    0    1
    1    0    0
*STAN
      DATA DATS    3    1    1    4
*SIMI
      DATS  LM1    2    1
*SIMQ
      DATA  LM2    1    1              5   10
*COMB
      LM1   LM2   LM3  0.5  0.2    1
 1.33505          0.558575
 0.423016         0.241493
*IDEN
      LM3    4    3
   2   5   7
   2   1   4
   1   2
   2   3   6
*IDEN
      LM3    3    1
   3   1   2   4
   2   5   7
   2   3   6
*IDEN
      LM3    4    2
   2   3   6
   1   2
   3   5   7   1
   1   4
*END
```

Output Listing

```
- - - - - - DATA MATRIX - - - - - -
INPUT FORMAT :  (7F5.0/(16F5.0))
MATRIX NAME :  DATA
TYPE OF MATRIX :  DATA
NUMBER OF OBJECTS :     7
NUMBER OF ATTRIBUTES :    10
MISSING VALUE CODE :  999.
NUMBER OF UNKNOWN OBJECTS :     3
OUTPUT OPTION :     1
- - - - - - DATA MATRIX - - - - - -
```

	1	2	3	4	5	6	7	8	9	10
1	1.2000	1.6000	9.2000	1.1000	0.5000	8.6000	1.4000	0.7000	1.0000	5.5000
2	3.7000	3.4000	1.2000	1.2000	6.2000	1.4000	5.8000	3.5000	1.4000	2.7000
3	1.7000	6.8000	3.7000	0.5000	1.5000	999.0000	1.2000	5.0000	1.4000	999.0000
4	4.3000	0.5000	3.9000	6.7000	3.1000	999.0000	3.1000	4.0000	2.2000	6.0000
5	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
6	1.0000	1.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000
7	1.0000	1.0000	0.0000	999.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000
8	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	1.0000	0.0000	0.0000	1.0000
9	1.0000	1.0000	1.0000	1.0000	0.0000	1.0000	0.0000	1.0000	1.0000	0.0000
10	0.0000	0.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	0.0000

Output listing, continued

```
- - - - STANDARDIZED DATA MATRIX - - - -
DATA MATRIX NAME : DATA
STANDARDIZED DATA MATRIX NAME : DATS
STANDARDIZATION OPTION :    3
NUMBER OF UNKNOWN OBJECTS    3
USE ATTRIBUTES  1  THRU  4
OUTPUT OPTION :    1
- - - - STANDARDIZED DATA MATRIX - - - -

          1        2        3        4
MIN    0.5000   1.2000   0.5000   0.5000
MAX    9.2000   6.2000   6.8000   6.7000
NOB    7.0000   7.0000   6.0000   6.0000
MEA    3.3714   3.2714   2.5667   3.6000
S.D    3.7959   2.1313   2.3338   2.0149

          1        2        3        4        5        6        7        8        9       10
  1   -0.5720  -0.4667   1.5355  -0.5984  -0.7565   1.3774  -0.5194  -0.7038  -0.6247   0.5608
  2    0.2011   0.0603  -0.9719  -0.9719   1.3741  -0.8781   1.1864   0.1072  -0.8781  -0.2681
  3   -0.3714   1.8139   0.4856  -0.8855  -0.4570 999.0000  -0.5856   1.0426  -0.4999 999.0000
  4    0.3474  -1.5385   0.1489   1.5385  -0.2481 999.0000  -0.2481   0.1985  -0.6948   1.1911



- - - - LIKENESS MATRIX FOR METRIC DATA - - - -
DATA MATRIX NAME : DATS
LIKENESS MATRIX NAME :  LM1
USE ATTRIBUTES  1  THRU  4
OUTPUT OPTION :    1
LIKENESS COFFICIENT : AVERAGE EUCLIDEAN DISTANCE
- - - - LIKENESS MATRIX FOR METRIC DATA - - - -

          8        9       10
  1   0.7155   0.7533   0.8593
  2   0.9579   1.3201   1.6945
  3   1.2739   1.2608   0.9186
  4   1.2932   1.1342   0.8082
  5   1.0070   1.1501   1.4723
  6   1.6282   1.4157   0.7208
  7   1.0061   1.0583   1.3359


- - - - LIKENESS MATRIX FOR BINARY DATA - - - -
DATA MATRIX NAME : DATA
LIKENESS MATRIX NAME :  LM2
LIKENESS COFFICIENT : JACCARD
VALUE FOR CONVERTING
METRIC TO BINARY DATA :    0.
USE ATTRIBUTES  5  THRU 10
OUTPUT OPTION :    1
- - - - LIKENESS MATRIX FOR BINARY DATA - - - -

          8        9       10
  1   0.6000   0.7500   0.4000
  2   0.6000   0.7500   0.4000
  3   0.2000   0.2500   0.2500
  4   0.7500   0.7500   0.2000
  5   0.7500   0.2000   0.5000
  6   0.4000   0.5000   0.5000
  7   0.6000   0.1667   0.7500



- - - - - COMBINE LIKENESS MATRICIES - - - - -
INPUT LIKENESS MATRICIES : LM1 LM2
OUTPUT LIKENESS MATRIX : LM3
WEIGHTING FACTORS :    0.5000    0.2000
OUTPUT OPTION :    1
- - - - - COMBINE LIKENESS MATRICIES - - - - -

          8        9       10
  1  -0.7012  -0.7915  -0.4068
  2  -0.4841  -0.2841   0.3408
  3   0.1300   0.0768  -0.2295
  4  -0.3083  -0.4506  -0.2869
  5  -0.5644   0.0191   0.0591
  6   0.2815   0.0085  -0.6136
  7  -0.4410  -0.0354  -0.2701
```

Output listing, continued

```
- - - - - IDEN - - - - -
LIKENESS MATRIX NAME :  LM3
NUMBER OF CLUSTERS :    4
  1   5   7
  2   1   4
  3   2
  4   3   6
CLUSTERING METHOD : UPGMA
- - - - - IDEN - - - - -

          8         9        10
  1   -0.5027   -0.0081   -0.1055
  2   -0.5047   -0.6210   -0.3468
  3   -0.4841   -0.2841    0.3408
  4    0.2057    0.0426   -0.4215

CLUSTERS UNKNOWNS IDENTIFIED WITH
          2         2         4


- - - - - IDEN - - - - -
LIKENESS MATRIX NAME :  LM3
NUMBER OF CLUSTERS :    3
  1   1   2   4
  2   5   7
  3   3   6
CLUSTERING METHOD : SINGLE LINKAGE
- - - - - IDEN - - - - -

          8         9        10
  1   -0.7012   -0.7915   -0.4068
  2   -0.5644   -0.0354   -0.2701
  3    0.1300    0.0085   -0.6136

CLUSTERS UNKNOWNS IDENTIFIED WITH
          1         1         3


- - - - - IDEN - - - - -
LIKENESS MATRIX NAME :  LM3
NUMBER OF CLUSTERS :    4
  1   3   6
  2   2
  3   5   7   1
  4   4
CLUSTERING METHOD : COMPLETE LINKAGE
- - - - - IDEN - - - - -

          8         9        10
  1    0.2815    0.0768   -0.2295
  2   -0.4841   -0.2841    0.3408
  3   -0.4410    0.0191    0.0591
  4   -0.3083   -0.4506   -0.2869

CLUSTERS UNKNOWNS IDENTIFIED WITH
          2         4         4
```

EXAMPLE 3: CLUSTAR RUN

This example uses all of the subroutines in CLUSTAR.

*INPT reads in a Data Matrix for t = 7 objects, n = 10 attributes. The Data Matrix is partitioned: partition 1 contains metric data (attributes 1-4); partition 2 contains binary data (attributes 5-10).

*INPT reads in a Resemblance Matrix containing pairwise similarity coefficients for the seven objects. This matrix is named RM3.

*STAN standardizes partition 1 of the Data Matrix.

*SIMI computes Resemblance Matrix named RM1 using the standardized data for partition 1.

*SIMQ computes Resemblance Matrix named RM2 using the binary data in the Data Matrix.

*CLST is used to produce a tree from RM1.

*CLST is used to produce a tree from RM2.

*COMB is used to combine RM1 and RM2, forming RM4.

*COMB is used to combine RM3 and RM4, forming the combined Resemblance Matrix RM5.

*CLST is used to produce a tree from RM5.

*CMPR compares the tree produced from RM1 (TRE1) to the tree produced from RM5 (TRE5).

*CMPR compares the tree produced from RM2 (TRE2) to the tree produced from RM5 (TRE5).

*CMPR compares the tree produced from RM3 (TRE3) to the tree produced from RM5 (TRE5).

Input Deck Listing

```
*INPT
      DTA    0    7   10    1    1 999.
(7F5.0)
  1.2  1.6  9.2  1.1  0.5  8.6  1.4
  3.7  3.4  1.2  1.2  6.2  1.4  5.8
  1.7  6.8  3.7  0.5  1.5 999.  1.2
  4.3  0.5  3.9  6.7  3.1 999.  3.1
    1    1    0    1    1    1    1
    1    1    0    1    0    0    0
    1    1    0 999.    1    0    1
    0    0    1    0    0    1    1
    1    1    1    1    0    1    0
    0    0    0    1    1    0    1
*INPT
      RS3   -1    7              1
(6F10.2)
       85
        5        61
       76        42        29
       69        71        33        77
       91        67        32        31        22
       42        66        12        24        71        50
*STAN
      DTA  DTS    3    1    1    4
*SIMI
      DTS  RS1    2    1
*SIMQ
      DTA  RS2    1    1         5   10
*CLST
      RS1  TR1    3    1    1
*CLST
      RS2  TR2    3    1    1  1.0  0.0
*CLST
      RS3  TR3    3    1    1  1.0  0.0
*COMB
      RS1  RS2  RS4  0.5  0.2    1
*COMB
      RS4  RS3  RS5 -1.  0.3    1
*CLST
      RS5  TR5    3    1    1
*CMPR
      TR1  TR5
*CMPR
      TR2  TR5
*CMPR
      TR3  TR5
*END
```

Output Listing

```
- - - - - - DATA MATRIX - - - - - -
INPUT FORMAT : (7F5.0)
MATRIX NAME :  DTA
TYPE OF MATRIX : DATA
NUMBER OF OBJECTS :    7
NUMBER OF ATTRIBUTES :   10
MISSING VALUE CODE :  999.
OUTPUT OPTION  :    1
- - - - - - DATA MATRIX - - - - - -
```

	1	2	3	4	5	6	7
1	1.2000	1.6000	9.2000	1.1000	0.5000	8.6000	1.4000
2	3.7000	3.4000	1.2000	1.2000	6.2000	1.4000	5.8000
3	1.7000	6.8000	3.7000	0.5000	1.5000	999.0000	1.2000
4	4.3000	0.5000	3.9000	6.7000	3.1000	999.0000	3.1000
5	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000	1.0000
6	1.0000	1.0000	0.0000	1.0000	0.0000	0.0000	0.0000
7	1.0000	1.0000	0.0000	999.0000	1.0000	0.0000	1.0000
8	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	1.0000
9	1.0000	1.0000	1.0000	1.0000	0.0000	1.0000	0.0000
10	0.0000	0.0000	0.0000	1.0000	1.0000	0.0000	1.0000

```
- - - - - - DATA MATRIX - - - - - -
INPUT FORMAT : (6F10.2)
MATRIX NAME :  RS3
TYPE OF MATRIX : RESEMBLANCE(SIMILARITY)
NUMBER OF OBJECTS :    7
OUTPUT OPTION :    1
- - - - - - DATA MATRIX - - - - - -
```

	1	2	3	4	5	6
2	0.8500					
3	0.0500	0.6100				
4	0.7600	0.4200	0.2900			
5	0.6900	0.7100	0.3300	0.7700		
6	0.9100	0.6700	0.3200	0.3100	0.2200	
7	0.4200	0.6600	0.1200	0.2400	0.7100	0.5000

```
- - - - STANDARDIZED DATA MATRIX - - - -
DATA MATRIX NAME :  DTA
STANDARDIZED DATA MATRIX NAME :  DTS
STANDARDIZATION OPTION :    3
USE ATTRIBUTES  1  THRU  4
OUTPUT OPTION :  1
- - - - STANDARDIZED DATA MATRIX - - - -
```

	1	2	3	4
MIN	0.5000	1.2000	0.5000	0.5000
MAX	9.2000	6.2000	6.8000	6.7000
NOB	7.0000	7.0000	6.0000	6.0000
MEA	3.3714	3.2714	2.5667	3.6000
S.D	3.7959	2.1313	2.3338	2.0149

	1	2	3	4	5	6	7
1	-0.5720	-0.4667	1.5355	-0.5984	-0.7565	1.3774	-0.5194
2	0.2011	0.0603	-0.9719	-0.9719	1.3741	-0.8781	1.1864
3	-0.3714	1.8139	0.4856	-0.8855	-0.4570	999.0000	-0.5856
4	0.3474	-1.5385	0.1489	1.5385	-0.2481	999.0000	-0.2481

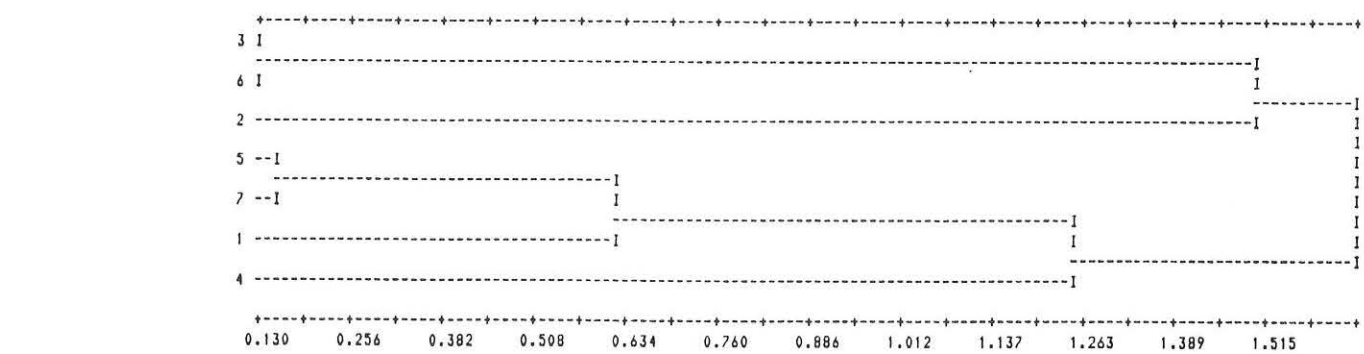
```
- - - - RESEMBLANCE MATRIX FOR METRIC DATA - - - -
DATA MATRIX NAME :  DTS
RESEMBLANCE MATRIX NAME :  RS1
USE ATTRIBUTES  1  THRU  4
OUTPUT OPTION :  1
RESEMBLANCE COFFICIENT : AVERAGE EUCLIDEAN DISTANCE
- - - - RESEMBLANCE MATRIX FOR METRIC DATA - - - -
```

	1	2	3	4	5	6
2	1.4459					
3	1.2837	1.5561				
4	0.8746	2.1117	1.4461			
5	0.6656	1.4690	1.7178	1.4920		
6	1.5756	1.4631	0.1300	1.3987	2.1938	
7	0.5861	1.4742	1.5958	1.4095	0.1643	1.9824

Output listing, continued

```
- - - - RESEMBLANCE MATRIX FOR BINARY DATA - - - -
DATA MATRIX NAME :  DTA
RESEMBLANCE MATRIX NAME :  RS2
RESEMBLANCE COFFICIENT : JACCARD
VALUE FOR CONVERTING
METRIC TO BINARY DATA :    0.
USE ATTRIBUTES  5  THRU 10
OUTPUT OPTION :  1
- - - - RESEMBLANCE MATRIX FOR BINARY DATA - - - -

          1        2        3        4        5        6
   2  1.0000
   3  0.2000   0.2000
   4  0.7500   0.7500   0.2000
   5  0.4000   0.4000   0.0000   0.5000
   6  0.4000   0.4000   0.6667   0.4000   0.2000
   7  0.3333   0.3333   0.2000   0.4000   0.7500   0.4000
```

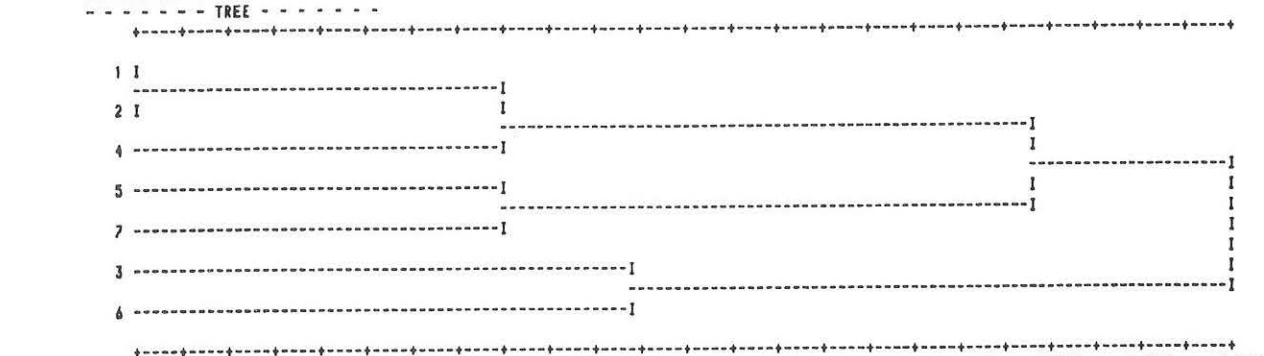
```
- - - - - - - TREE - - - - - - -
RESEMBLANCE MATRIX NAME :  RS1
TREE NAME :  TR1
NODE COUNT OPTION :  1
COPHENETIC OPTION :  1
CLUSTERING METHOD : UPGMA
MINIMUM VALUE ON TREE :     0.1300
MAXIMUM VALUE ON TREE :     1.6412
- - - - - - - TREE - - - - - - -
```




```
NODE COUNT MATRIX
          1        2        3        4        5        6
   2   4.0
   3   5.0      2.0
   4   2.0      3.0      4.0
   5   2.0      5.0      6.0      3.0
   6   5.0      2.0      1.0      4.0      6.0
   7   2.0      5.0      6.0      3.0      1.0      6.0

COPHENETIC CORRELATION MATRIX
          1        2        3        4        5        6
   2  1.6412
   3  1.6412   1.5096
   4  1.2587   1.6412   1.6412
   5  0.6259   1.6412   1.6412   1.2587
   6  1.6412   1.5096   0.1300   1.6412   1.6412
   7  0.6259   1.6412   1.6412   1.2587   0.1643   1.6412
COPHENETIC CORRELATION =     0.8980
```

Output listing, continued

```
- - - - - - - TREE - - - - - - -
RESEMBLANCE MATRIX NAME :  RS2
TREE NAME :  TR2
NODE COUNT OPTION :  1
COPHENETIC OPTION :  1
CLUSTERING METHOD : UPGMA
MINIMUM VALUE ON TREE :      0.2600
MAXIMUM VALUE ON TREE :      1.0000
- - - - - - - TREE - - - - - - -
```



NODE COUNT MATRIX

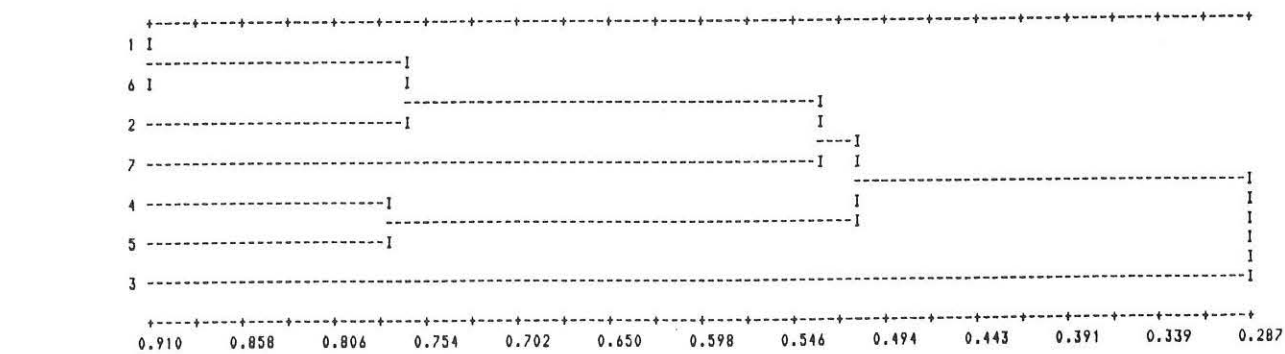
	1	2	3	4	5	6
2	1.0					
3	5.0	5.0				
4	2.0	2.0	4.0			
5	4.0	4.0	4.0	3.0		
6	5.0	5.0	1.0	4.0	4.0	
7	4.0	4.0	4.0	3.0	1.0	4.0

COPHENETIC CORRELATION MATRIX

	1	2	3	4	5	6
2	1.0000					
3	0.2600	0.2600				
4	0.7500	0.7500	0.2600			
5	0.3944	0.3944	0.2600	0.3944		
6	0.2600	0.2600	0.6667	0.2600	0.2600	
7	0.3944	0.3944	0.2600	0.3944	0.7500	0.2600

COPHENETIC CORRELATION = 0.9183

```
- - - - - - - TREE - - - - - - -
RESEMBLANCE MATRIX NAME :  RS3
TREE NAME :  TR3
NODE COUNT OPTION :  1
COPHENETIC OPTION :  1
CLUSTERING METHOD : UPGMA
MINIMUM VALUE ON TREE :      0.2867
MAXIMUM VALUE ON TREE :      0.9100
- - - - - - - TREE - - - - - - -
```



NODE COUNT MATRIX

	1	2	3	4	5	6
2	2.0					
3	5.0	4.0				
4	5.0	4.0	3.0			
5	5.0	4.0	3.0	1.0		
6	1.0	2.0	5.0	5.0	5.0	
7	3.0	2.0	3.0	3.0	3.0	3.0

Output listing, continued

```
      COPHENETIC CORRELATION MATRIX
              1         2         3         4         5         6
      2    0.7600
      3    0.2867    0.2867
      4    0.5075    0.5075    0.2867
      5    0.5075    0.5075    0.2867    0.7700
      6    0.9100    0.7600    0.2867    0.5075    0.5075
      7    0.5267    0.5267    0.2867    0.5075    0.5075    0.5267
    COPHENETIC CORRELATION =     0.7166


 - - - - - COMBINE RESEMBLANCE MATRICIES - - - - -
 INPUT RESEMBLANCE MATRICIES :  RS1   RS2
 OUTPUT RESEMBLANCE MATRIX : RS4
 WEIGHTING FACTORS :    0.5000    0.2000
 OUTPUT OPTION :  1
 - - - - - COMBINE RESEMBLANCE MATRICIES - - - - -

      MEAN           S.D.
     .133505E+01    .558575E+00
     .423016E+00    .241493E+00

            1         2         3         4         5         6
    2   -0.3786
    3    0.1387    0.3826
    4   -0.6830    0.4244    0.2841
    5   -0.5802    0.1390    0.6929    0.0767
    6    0.2344    0.1337   -1.2805    0.0760    0.9534
    7   -0.5961    0.1989    0.4181    0.0857   -1.3188    0.5985

 - - - - - COMBINE RESEMBLANCE MATRICIES - - - - -
 INPUT RESEMBLANCE MATRICIES :  RS4   RS3
 OUTPUT RESEMBLANCE MATRIX : RS5
 WEIGHTING FACTORS :   -1.0000    0.3000
 OUTPUT OPTION :  1
 - - - - - COMBINE RESEMBLANCE MATRICIES - - - - -

      MEAN           S.D.
    0.              .100000E+01
     .502857E+00    .251896E+00

            1         2         3         4         5         6
    2   -0.7920
    3    0.6781    0.2550
    4   -0.9892    0.5231    0.5376
    5   -0.8031   -0.1077    0.8988   -0.2414
    6   -0.2505   -0.0654   -1.0627    0.3057    1.2903
    7   -0.4974    0.0117    0.8740    0.3988   -1.5655    0.6019

 - - - - - - - TREE - - - - - - -
 RESEMBLANCE MATRIX NAME :  RS5
 TREE NAME :  TR5
 NODE COUNT OPTION :  1
 COPHENETIC OPTION :  1
 CLUSTERING METHOD : UPGMA
 MINIMUM VALUE ON TREE :    -1.5655
 MAXIMUM VALUE ON TREE :     0.5125
 - - - - - - - TREE - - - - - - -

   +----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
 5 I
   ------------------------------------------------------------------------I
 7 I                                                                       I
                                                                           ------------I
 1 ----------------------------------I                                     I           I
                                     ----------------------------------------I         I
 4 ----------------------------------I                                               ----------------------------------I
                                                                                     I                                 I
 2 ------------------------------------------------------------------------------------I                                 I
                                                                                                                       I
 3 ----------------------------I                                                                                      I
                               ------------------------------------------------------------------------------------------I
 6 ----------------------------I

   +----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
-1.565    -1.392    -1.219    -1.046    -0.873    -0.700    -0.526    -0.353    -0.180    -0.007     0.166     0.339     0.513

 NODE COUNT MATRIX
        1         2         3         4         5         6
 2      3.0
 3      5.0       3.0
 4      1.0       3.0       5.0
 5      3.0       3.0       5.0       3.0
 6      5.0       3.0       1.0       5.0       5.0
 7      3.0       3.0       5.0       3.0       1.0       5.0
```

Output listing, continued

```
   COPHENETIC CORRELATION MATRIX
          1         2         3         4         5         6
   2   -0.0912
   3    0.5125    0.5125
   4   -0.9892   -0.0912    0.5125
   5   -0.2858   -0.0912    0.5125   -0.2858
   6    0.5125    0.5125   -1.0627    0.5125    0.5125
   7   -0.2858   -0.0912    0.5125   -0.2858   -1.5655    0.5125
COPHENETIC CORRELATION =     0.8225

- - - - - COMPARE TREES - - - - -
FIRST TREE NAME : TR1
SECOND TREE NAME : TR5
- - - - - COMPARE TREES - - - - -

 2D/(N*(N-1))          0.7619
 NODE CORRELATION      0.8198
 TREE CORRELATION      0.8050



- - - - - COMPARE TREES - - - - -
FIRST TREE NAME : TR2
SECOND TREE NAME : TR5
- - - - - COMPARE TREES - - - - -

 2D/(N*(N-1))          0.8571
 NODE CORRELATION      0.6787
 TREE CORRELATION     -0.7247



- - - - - COMPARE TREES - - - - -
FIRST TREE NAME : TR3
SECOND TREE NAME : TR5
- - - - - COMPARE TREES - - - - -

 2D/(N*(N-1))          1.5238
 NODE CORRELATION     -0.1029
 TREE CORRELATION     -0.0410
```

LITERATURE CITED

ANDERBERG, M. R. 1973. Cluster analysis for application. Academic Press, New York. 359 pp.

BOYCE, A. J. 1969. Mapping diversity: a comparative study of some numerical methods. Pages 1-31 *in* A. J. Cole, ed. Numerical taxonomy. Academic Press, New York.

CLIFFORD, H. T., and W. STEPHENSON. 1975. An introduction to numerical classification. Academic Press, New York. 229 pp.

EVERITT, B. 1974. Cluster analysis. John Wiley and Sons, New York. 122 pp.

HOHN, M. E. 1976. Binary coefficients: a theoretical and empirical study. Pages 137-150 *in* J. of Internatl. Assn. for Math. Geol., Vol. 8, No. 2.

MOTYKA, J., B. DOBRZANSKI, and S. ZAWAZKI. 1950. Preliminary studies on meadows in the southeast of the province Lublin. Pages 367-447 *in* Univ. Mariae Curie-Sklodowska Ann., sect. E, 5.

SNEATH, P. H. A., and R. R. SOKAL. 1973. Numerical taxonomy. W. H. Freeman, San Francisco. 573 pp.

WILLIAMS, W. T., and H. T. CLIFFORD. 1971. On the comparison of two classifications of the same set of elements. Pages 519-522 *in* Taxonomy, Vol. 20.