PSYCHOMETRIKA-VOL. 42, NO. 3
SEPTEMBER 1977

ADDITIVE SIMILARITY TREES

SHMUEL SATTATH AND AMOS TVERSKY

HEBREW UNIVERSITY OF JERUSALEM

Similarity data can be represented by additive trees. In this model, objects are represented by the external nodes of a tree, and the dissimilarity between objects is the length of the path joining them. The additive tree is less restrictive than the ultrametric tree, commonly known as the hierarchical clustering scheme. The two representations are characterized and compared. A computer program, ADDTREE, for the construction of additive trees is described and applied to several sets of data. A comparison of these results to the results of multidimensional scaling illustrates some empirical and theoretical advantages of tree representations over spatial representations of proximity data.

Key words: proximity, clustering, multidimensional scaling.

The two goals of research on the representation of proximity data are the development of theories for explaining similarity relations and the construction of scaling procedures for describing and displaying similarities between objects. Indeed, most representations of proximity data can be regarded either as similarity theories or as scaling procedures. These representations can be divided into two classes: spatial models and network models. The spatial models—called multidimensional scaling—represent each object as a point in a coordinate space so that the metric distances between the points reflect the observed proximities between the objects. Network models represent each object as a node in a connected graph, typically a tree, so that the relations between the nodes in the graph reflect the observed proximity relations among the objects.

This paper investigates tree representations of similarity data. We begin with a critical discussion of the familiar hierarchical clustering scheme [Johnson, 1967], and present a more general representation, called the additive tree. A computer program (ADDTREE) for the construction of additive trees from proximity data is described and illustrated using several sets of data. Finally, the additive tree is compared with multidimensional scaling from both empirical and theoretical standpoints.

We thank Nancy Henley and Vered Kraus for providing us with data, and Jan deLeeuw for calling our attention to relevant literature. The work of the first author was supported in part by the Psychology Unit of the Israel Defense Forces.

Requests for reprints should be sent to Prof. Amos Tversky, Dept. of Psychology, The Hebrew University, Jerusalem, Israel.

TABLE 1

Dissimilarities between animals

	Donkey	Cow	Pig
Camel	5.0	5.6	7.2
Donkey		4.6	5.7
Cow			4.9

Consider the proximity matrix presented in Table 1, taken from a study by Henley [1969].

The entries of the table are average ratings of dissimilarity between the respective animals on a scale from 0 (maximal similarity) to 10 (maximal dissimilarity). Such data have commonly been analyzed using the hierarchical clustering scheme (HCS) that yields a hierarchy of nested clusters. The application of this scaling procedure to Table 1 is displayed in Figure 1.

The construction of the tree proceeds as follows. The two objects which are closest to each other (e.g., donkey and cow) are combined first, and are now treated as a single element, or cluster. The distance between this new element, z, and any other element, y, is defined as the minimum (or the average) of the distances between y and the members of z. This operation is repeated until a single cluster that includes all objects is obtained. In such a representation the objects appear as the external nodes of the tree, and the distance between objects is the height of their meeting point, or equivalently, the length of the path joining them.

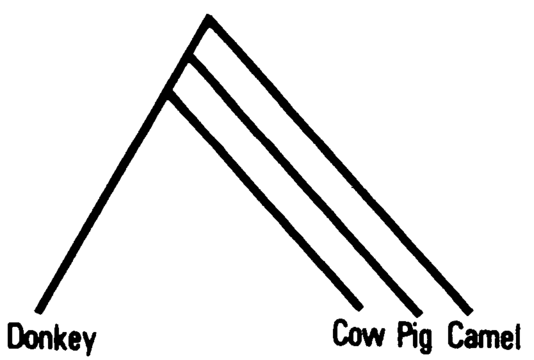

FIGURE 1
The representation of Table 1 as an HCS.

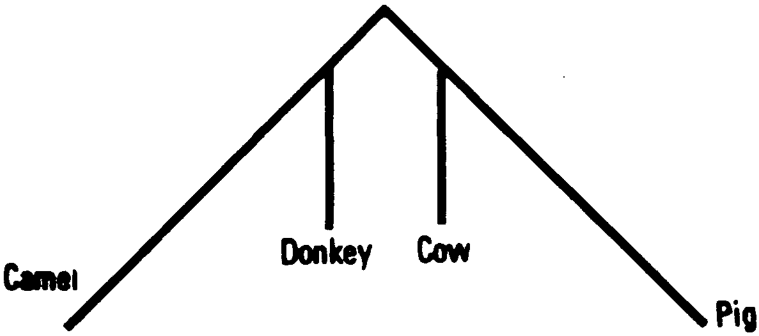

FIGURE 2
The representation of Table 1 as an additive tree, in rooted form.

This model imposes severe constraints on the data. It implies that given two disjoint clusters, all intra-cluster distances are smaller than all inter-cluster distances, and that all the inter-cluster distances are equal. This property is called the ultrametric inequality, and the representation is denoted an ultrametric tree. The ultrametric inequality, however, is often violated by data, see, e.g., Holman [Note 1]. To illustrate, note that according to Figure 1, camel should be equally similar to donkey, cow and pig, contrary to the data of Table 1.

The limitations of the ultrametric tree have led several psychologists, e.g., Carroll and Chang [1973], Carroll [1976], Cunningham [Note 2, Note 3], to explore a more general structure, called an additive tree. This structure appears under different names including: weighted tree, free tree, path-length tree, and unrooted tree, and its formal properties were studied extensively, see, e.g., Buneman [1971, pp. 387–395; 1974], Dobson [1974], Hakimi and Yau [1964], Patrinos and Hakimi [1972], Turner and Kautz [1970, Sections III-4 and III-6]. The representation of Table 1 as an additive tree is given in Figure 2. As in the ultrametric tree, the external nodes correspond to objects and the distance between objects is the length of the path joining them. A formal definition of an additive tree is presented in the next section.

It is instructive to compare the two representations of Table 1 displayed in Figures 1 and 2. First, note that the clustering is different in the two figures. In the ultrametric tree (Figure 1), cow and donkey form a single cluster that is subsequently joined by pig and camel. In the additive tree (Figure 2), camel with donkey form one cluster, and cow with pig form another cluster. Second, in the additive tree, unlike the ultrametric tree, intra-cluster distances may exceed inter-cluster distances. For example, in Figure 2 cow and donkey belong to different clusters although they are the two closest animals. Third, in an additive tree, an object outside a cluster is no longer equidistant from all objects inside the cluster. For example, both cow and pig are closer to donkey than to camel.

The differences between the two models stem from the fact than in the ultrametric tree (but not in an additive tree) the external nodes are all equally distant from the root. The greater flexibility of the additive tree permits a more faithful representation of data. Spearman's rank correlation, for example, between the dissimilarities of Table 1 and the tree distances is 1.00 for the additive tree and 0.64 for the ultrametric tree.

Note that the distances in an additive tree do not depend on the choice of root. For example, the tree of Figure 2 can be displayed in unrooted form, as shown in Figure 3. Nevertheless, it is generally more convenient to display similarity trees in a rooted form.

Analysis of Trees

In this section we define ultrametric and additive trees, characterize the conditions under which proximity data can be represented by these models, and describe the structure of the clusters associated with them.

Representation of Dissimilarity

A *tree* is a (finite) connected graph without cycles. Hence, any two nodes in a tree are connected by exactly one path. An *additive tree* is a tree with a metric in which the distance between nodes is the length of the path (i.e., the sum of the arc-lengths) that joins them. An additive tree with a distinguished node (named the root) which is equidistant from all external nodes is called an *ultrametric tree*. Such trees are normally represented with the root on top, (as

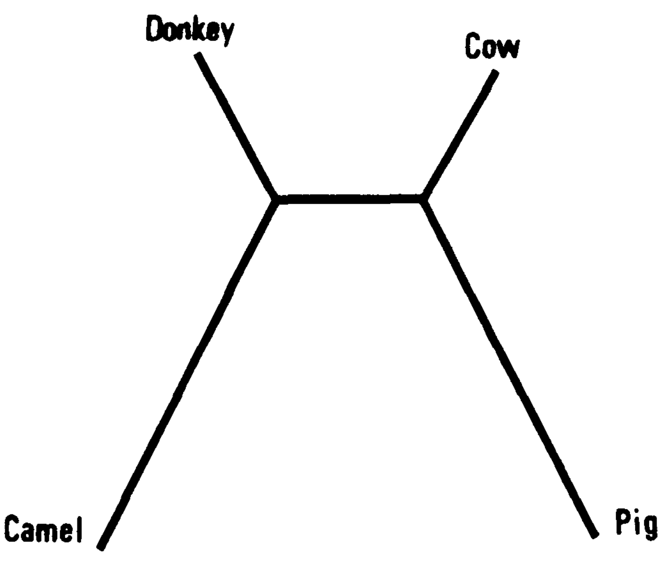

FIGURE 3
The representation of Table 1 as an additive tree, in unrooted form.

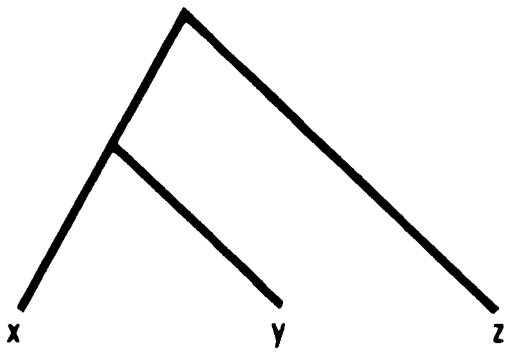

FIGURE 4
The relations among three objects in an ultrametric tree.

in Figure 1) so that the distance between external nodes is expressible as the height of the lowest (internal) node that lies above them.

A *dissimilarity measure* d on a finite set of objects $S = \{x, y, z, \cdots\}$ is a non-negative function on $S \times S$ such that $d(x, y) = d(y, x)$, and $d(x, y) = 0$ iff $x = y$. A tree (ultrametric or additive) *represents* a dissimilarity measure on S iff the external nodes of the tree can be associated with the objects of S so that the tree distances between external nodes coincide with the dissimilarities between the respective objects.

If a dissimilarity measure d on S is represented by an ultrametric tree, then the relation among any three objects in S has the form depicted in Figure 4. It follows, therefore, that for all x, y, z in S

$$d(x, y) \leq \max \{d(x, z), d(y, z)\}.$$

This property, called the *ultrametric inequality,* is both necessary and sufficient for the representation of a dissimilarity measure by an ultrametric tree [Johnson, 1967; Jardine & Sibson, 1971]. As noted in the previous section, however, the ultrametric inequality is very restrictive. It implies that for any three objects in S, two of the dissimilarities are equal and the third does not exceed them. Thus the dissimilarities among any three objects must form either an equilateral triangle or an isosceles triangle with a narrow base.

An analogous analysis can be applied to additive trees. If a dissimilarity measure d on S is represented by an additive tree, then the relations among any four objects in S has the form depicted in Figure 5, with non-negative α, β, γ, δ, ϵ. It follows, therefore, that in this case

$$\begin{aligned} d(x, y) + d(u, v) &= \alpha + \beta + \gamma + \delta \\ &\leq \alpha + \beta + \gamma + \delta + 2\epsilon \\ &= d(x, u) + d(y, v) \\ &= d(x, v) + d(y, u). \end{aligned}$$

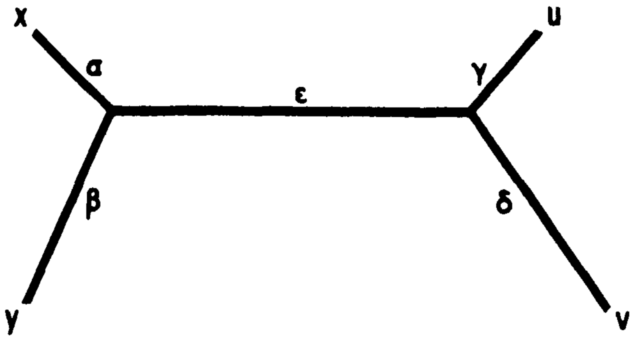

FIGURE 5
The relations among four objects in an additive tree.

Hence, any four objects can be labeled so as to satisfy the above inequality.

Consequently, in an additive tree,

$$d(x, y) + d(u, v) \leq \max \{d(x, u) + d(y, v), d(x, v) + d(y, u)\}$$

for all x, y, u, v in S (not necessarily distinct).

It is easy to verify that this condition, called the *additive inequality* (or the four-points condition), follows from the ultrametric inequality and implies the triangle inequality. It turns out that the additive inequality is both necessary and sufficient for the representation of a dissimilarity measure by an additive tree. For a proof of this assertion, see, e.g., Buneman [1971, pp. 387–395; 1974], Dobson [1974]. To illustrate the fact that the additive inequality is less restrictive than the ultrametric inequality, note that the distances between any four points on a line satisfy the former but not the latter.

The ultrametric and the additive trees differ in the number of parameters employed in the representation. In an ultrametric tree all $\binom{n}{2}$ inter-point distances are determined by at most $n - 1$ parameters where n is the number of elements in the object set S. In an additive tree, the distances are determined by at most $2n - 3$ parameters.

Trees and Clusters

A dissimilarity measure, d, can be used to define different notions of clustering, see, e.g., Sokal and Sneath [1973]. Two types of clusters—tight and loose—are now introduced and their relations to ultrametric and additive trees are discussed.

A nonempty subset A of S is a *tight cluster* if

$$\max_{x,y \in A} d(x, y) < \min_{\substack{x \in A \\ z \in S-A}} d(x, z) .$$

That is, A is a tight cluster whenever the dissimilarity between any two objects in A is smaller than the dissimilarity between any object in A and any object outside A, i.e., in $S - A$. It follows readily that a subset A of an ultrametric tree is a tight cluster iff there is an arc such that A is the set of all objects that lie below that arc. In Figure 1, for example, {donkey, cow} and {donkey, cow, pig} are tight clusters whereas {cow, pig} and {cow, pig, camel} are not.

A subset A of S is a *loose cluster* if for any x, y in A and u, v in $S - A$

$$d(x, y) + d(u, v) < \min \{d(x, u) + d(y, v), d(x, v) + d(y, u)\}.$$

In Figure 5, for example, the binary loose clusters are $\{x, y\}$ and $\{u, v\}$. Let A, B denote disjoint nonempty loose clusters; let $D(A)$, $D(B)$ denote the average intra-cluster dissimilarities of A and B, respectively; and let $D(A, B)$ denote the average inter-cluster dissimilarity between A and B. It can be shown that $1/2(D(A) + D(B)) < D(A, B)$. That is, the mean of the average dissimilarity within loose clusters is smaller than the average dissimilarity between loose clusters.

The deletion of an arc divides a tree into two subtrees, thereby partitioning S into two nonempty subsets. It follows readily that, in an additive tree, both subsets are loose clusters, and all loose clusters can be obtained in this fashion. Thus, an additive tree induces a family of loose clusters whereas an ultrametric tree defines a family of tight clusters. In Table 1, for example, the cluster {Donkey, Cow} is tight but not loose, whereas the clusters {Donkey, Camel} and {Cow, Pig} are loose but not tight, see Figures 1 and 2. Scaling methods for the construction of similarity trees are generally based on clustering: HCS is based on tight clusters, whereas the following procedure for the construction of additive trees is based on loose clusters.

Computational Procedure

This section describes a computer algorithm, ADDTREE, for the construction of additive similarity trees. Its input is a symmetric matrix of similarities or dissimilarities, and its output is an additive tree.

If the additive inequality is satisfied without error, then the unique additive tree that represents the data can be constructed without difficulty. In fact, any proof of the sufficiency of the additive inequality provides an algorithm for the errorless case. The problem, therefore, is the development of an efficient algorithm that constructs an additive tree from fallible data.

This problem has two components: (i) construction, which consists of finding the most appropriate tree-structure, (ii) estimation, which consists of finding the best estimates of arc-lengths. In the present algorithm the construction of the tree proceeds in stages by clustering objects so as to maximize the number of sets satisfying the additive inequality. The estimation of arc lengths is based on the least square criterion. The two components of the program are now described in turn.

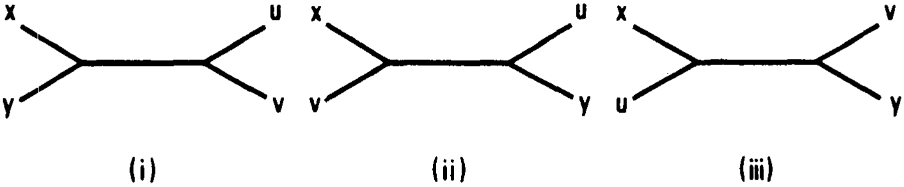

FIGURE 6
The three possible configurations of four objects in an additive tree.

Construction

In an additive tree, any four distinct objects, x, y, u, v, appear in one of the configurations of Figure 6. The patterns of distances which correspond to the configurations of Figure 6 are:

$$\begin{array}{rl}
\text{(i)} & d(x, y) + d(u, v) < d(x, u) + d(y, v) = d(x, v) + d(y, u) \\
\text{(ii)} & d(x, v) + d(y, u) < d(x, u) + d(y, v) = d(x, y) + d(u, v) \\
\text{(iii)} & d(x, u) + d(y, v) < d(x, y) + d(u, v) = d(x, v) + d(y, u).
\end{array}$$

Our task is to select the most appropriate configuration on the basis of an observed dissimilarity measure δ. It is easy to see that any four objects can be relabeled so that

$$\delta(x, y) + \delta(u, v) \leq \delta(x, u) + \delta(y, v) \leq \delta(x, v) + \delta(y, u).$$

It is evident, in this case, that Configuration (i) represents these dissimilarities better than (ii) or (iii). Hence, we obtain the following rule for choosing the best configuration for any set of four elements: label the objects so as to satisfy the above inequality, and select Configuration (i). The objects x and y (as well as u and v) are then called *neighbors*. The construction of the tree proceeds by grouping elements on the basis of the neighbors relation. The major steps of the construction are sketched below.

For each pair x, y, ADDTREE examines all objects u, v and counts the number of quadruples in which x and y are neighbors. The pair x, y with the highest score is selected, and its members are combined to form a new element z which replaces x and y in the subsequent analysis. The dissimilarity between z and any other element u is set equal to $(\delta(u, x) + \delta(u, y))/2$. The pair with the next highest score is selected next. If its elements have not been previously selected, they are combined as above, and the scanning of pairs is continued until all elements have been selected. Ties are treated here in a natural manner.

This grouping process is first applied to the object set S yielding a collection of elements which consists of the newly formed elements together with the original elements that were not combined in this process. The grouping process is then applied repeatedly to the outcome of the previous phase until the number of remaining elements is three. Finally, these elements are combined to form the last element, which is treated as the root of the tree.

It is possible to show that if only one pair of elements are combined in each phase, then perfect subtrees in the data appear as subtrees in the representation. In particular, any additive tree is reproduced by the above procedure.

The construction procedure described above uses sums of dissimilarities to define neighbors and to compute distances to the new (constructed) elements. Strictly speaking, this procedure is applicable to cardinal data, i.e., data measured on interval or ratio scales. For ordinal data, a modified version of the algorithm has been developed. In this version, the neighbors relation is introduced as follows. Suppose δ is an ordinal dissimilarity scale, and

$$\begin{aligned} &\delta(x, y) < \delta(x, u), \qquad \delta(x, y) < \delta(x, v), \\ &\delta(u, v) < \delta(y, v), \qquad \delta(u, v) < \delta(y, w). \end{aligned}$$

Then we conclude that x and y (as well as u and v) are *neighbors*. (If the inequalities on the left [right] alone hold, then x and y [as well as u and v] are called *semi-neighbors*, and are counted as half neighbors.)

If x and y are neighbors in the ordinal sense, they are also neighbors in the cardinal sense, but the converse is not true. In the cardinal case, every four objects can be partitioned into two pairs of neighbors; in the ordinal case, this property does not always hold since the defining inequality may fail for all permutations of the objects. To define the distances to the new elements in the ordinal version of the algorithm, some ordinal index of average dissimilarity, e.g., mean rank or median, can be used.

Estimation

Although the construction of the tree is independent of the estimation of arc lengths, the two processes are performed in parallel. The parameters of the tree are estimated, employing a least-square criterion. That is, the program minimizes

$$\sum_{x,y \in S} (d(x, y) - \delta(x, y))^2,$$

where d is the distance function of the tree. Since an additive tree with n objects has $m \leq 2n - 3$ parameters (arcs), one obtains the equation $CX = \delta$ where δ is the vector of dissimilarities, X is the vector of (unknown) arc lengths, and C is an $\binom{n}{2} \times m$ matrix where

$$c_{ij} = \begin{cases} 1 \text{ if the } i\text{-th tree-distance includes the } j\text{-th arc} \\ 0 \text{ otherwise} \end{cases}$$

The least-square solution of $CX = \delta$ is $X = (C^T C)^{-1} C^T \delta$, provided $C^T C$

is positive definite. In general, this requires inverting an $m \times m$ matrix which is costly for moderate m and prohibitive for large m. However, an exact solution that requires no matrix inversion and greatly simplifies the estimation process can be obtained by exploiting the following property of additive trees. Consider an arc and remove its endpoints; this divides the tree into a set of disjoint subtrees. The least-square estimate of the length of that arc is a function of (i) the average distances between the subtrees and (ii) the number of objects in each subtree. The proof of this proposition, and the description of that function are long and tedious and are therefore omitted. It can also be shown that all negative estimates (which reflect error) should be set equal to zero.

The present program constructs a rooted additive tree. The graphical representation of a rooted tree is unique up to permutations of its subtrees. To select an informative graphical representation, the program permutes the objects so as to maximize the correspondence of the similarity between objects and the ordering of their positions in the display—subject to the constraint imposed by the structure of the tree. Under the same constraint, the program can also permute the objects so as to maximize the ordinal correlation (γ) with any prespecified ordering.

Comparison of Algorithms

Several related methods have recently been proposed. Carroll [1976] discussed two extensions of HCS. One concerns an ultrametric tree in which internal as well as external nodes represent objects [Carroll & Chang, 1973]. Another concerns the representation of a dissimilarity matrix as the sum of two or more ultrametric trees [Carroll & Pruzansky, Note 4]. The first effective procedure for constructing an additive tree for fallible similarity data was presented by Cunningham [Note 2, Note 3]. His program, like ADDTREE, first determines the tree structure, and then obtains least-square estimates of arc-lengths. However, there are two problems with Cunningham's program. First, in the presence of noise, it tends to produce degenerate trees with few internal nodes. This problem becomes particularly severe when the number of objects is moderate or large. To illustrate, consider the additive tree presented in Figure 8, and suppose that, for some reason or another (e.g., errors of measurement), monkey was rated as extremely similar to squirrel. In Cunningham's program, this single datum produces a drastic change in the structure of the tree: It eliminates the arcs labeled 'rodents' and 'apes', and combines all rodents and apes into a single cluster. In ADDTREE, on the other hand, this datum produces only a minor change. Second, Cunningham's estimation procedure requires the inversion of a $\binom{n}{4} \times \binom{n}{4}$ matrix, which restricts the applicability of the program to relatively small data sets, say under 15 objects.

ADDTREE overcomes the first problem by using a "majority" rule rather than a "veto" rule to determine the tree structure, and it overcomes the second problem by using a more efficient method of estimation. The core memory required for ADDTREE is of the order of n^2, hence it can be applied to sets of 100 objects, say, without any difficulty. Furthermore, ADDTREE is only slightly more costly than HCS, and less costly than a multidimensional scaling program in two dimensions.

Applications

This section presents applications of ADDTREE to several sets of similarity data and compares them with the results of multidimensional scaling and HCS.

Three sets of proximity data are analyzed. To each data set we apply the cardinal version of ADDTREE, the average method of HCS [Johnson, 1967], and smallest space analysis [Guttman, 1968; Lingoes, 1970] in 2 and 3 dimensions-denoted SSA/2D and SSA/3D, respectively. (The use of the ordinal version of ADDTREE, and the min method of HCS did not change the results substantially.) For each representation we report two measures of correspondence between the solution and the original data: the product-moment correlation r, and Kruskal's ordinal measure of stress defined as

$$\left\{ \frac{\sum_x \sum_y (d(x,y) - \hat{d}(x,y))^2}{\sum_x \sum_y d(x,y)^2} \right\}^{1/2}$$

where d is the distance in the respective representation, and $\hat{d}$ is an appropriate order-preserving transformation of the original dissimilarities [Kruskal, 1964].

Since ADDTREE and HCS yielded similar tree structures in all three data sets, only the results of the former are presented along with the two-dimensional (Euclidean) configurations obtained by SSA/2D. The two-dimensional solution was chosen for comparison because (i) it is the most common and most interpretable spatial representation, and (ii) the number of parameters of a two-dimensional solution is the same as the number of parameters in an additive tree.

Similarity of Animals

Henley [1969] obtained average dissimilarity ratings between animals from a homogeneous group of 18 subjects. Each subject rated the dissimilarity between all pairs of 30 animals on a scale from 0 to 10.

The result of SSA/2D is presented in Figure 7. The horizontal dimension is readily interpreted as size, with elephant and mouse at the two extremes, and the vertical dimension may be thought of as ferocity [Henley, 1969], although the correspondence is far from perfect.

The result of ADDTREE is presented in Figure 8 in parallel form. In this

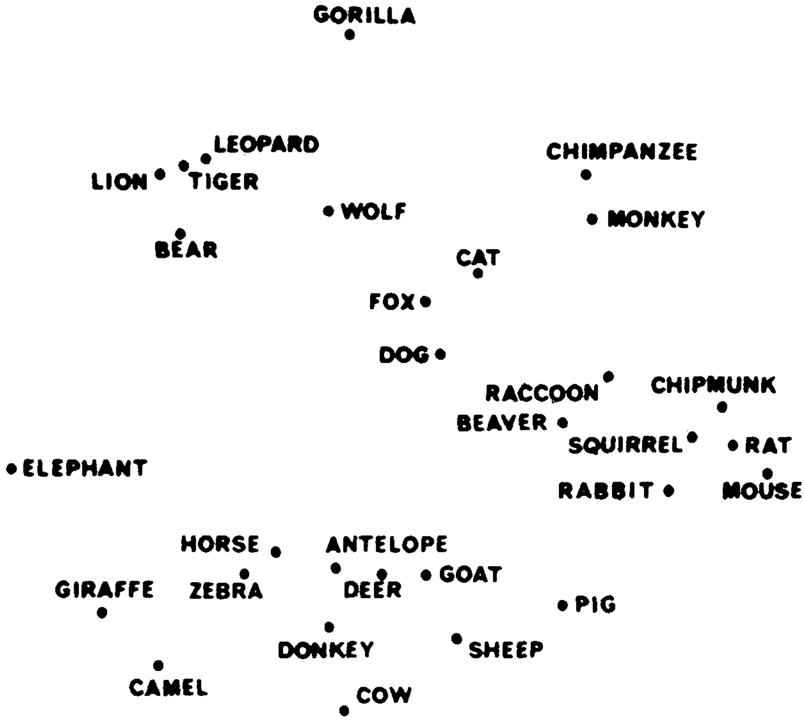

FIGURE 7
Representation of animal similarity (Henley, 1969) by SSA/2D.

form all branches are parallel, and the distance between two nodes is the sum of the *horizontal* arcs on the path joining them. Clearly, every (rooted) tree can be displayed in parallel form which we use because of its convenience.

In an additive tree the root is not determined by the distances, and any point on the tree can serve as a root. Nevertheless, different roots induce different hierarchies of partitions or clusters. ADDTREE provides a root that tends to minimize the variance of the distances to the external nodes. Other criteria for the selection of a root could readily be incorporated. The choice of a root for an additive tree is analogous to the choice of a coordinate system in (euclidean) multidimensional scaling. Both choices do not alter the distances, yet they usually affect the interpretation of the configuration.

In Figure 8 the 30 animals are first partitioned into four major clusters: herbivores, carnivores, apes, and rodents. The major clusters in the figure are labeled to facilitate the interpretation. Each of these clusters is further partitioned into finer clusters. For example, the carnivores are partitioned into three clusters: felines (including cat, leopard, tiger, and lion), canines (including dog, fox, and wolf), and bear.

Recall that in a rooted tree, each arc defines a cluster which consists of all the objects that follow from it. Thus, each arc can be interpreted as the features

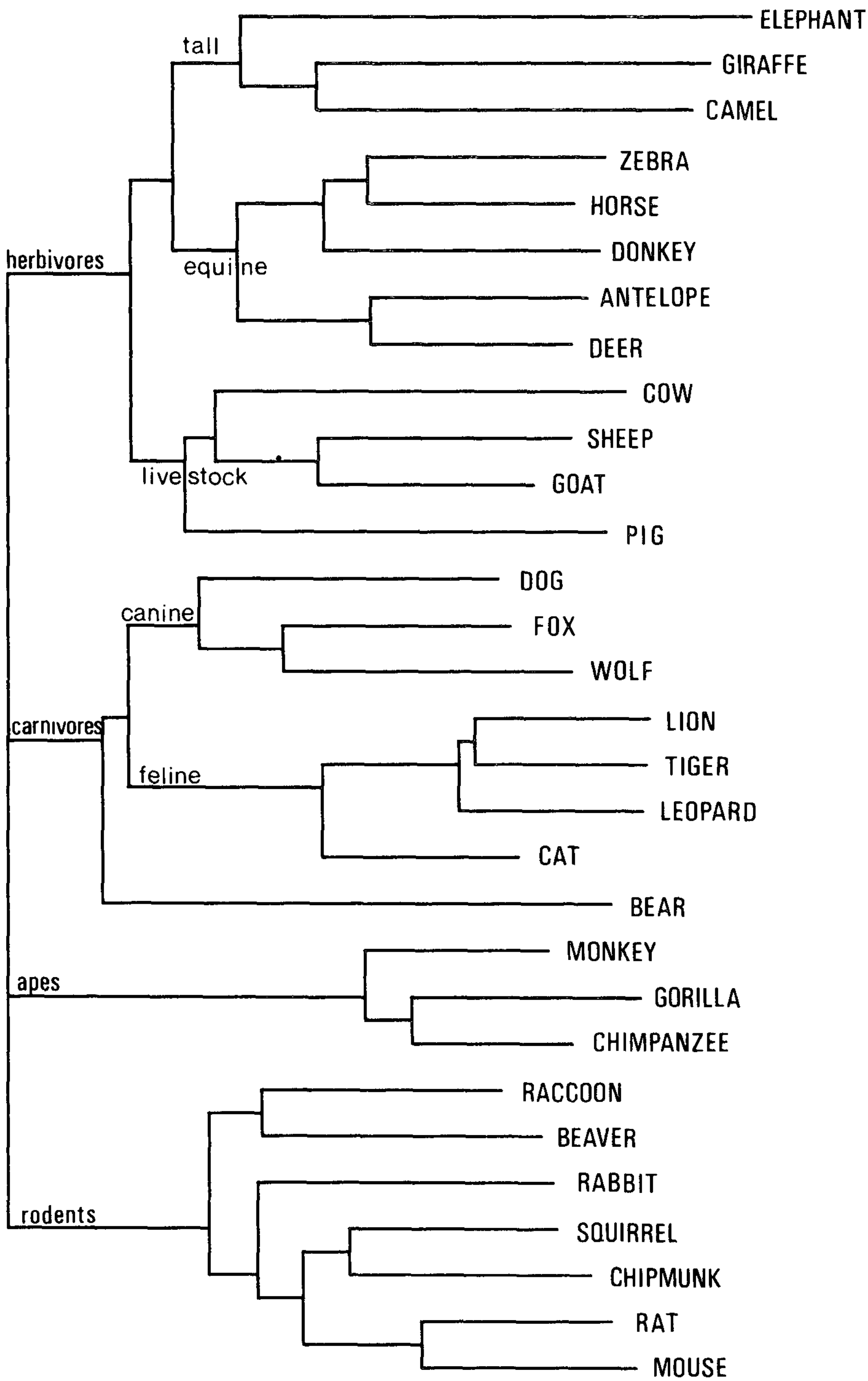

FIGURE 8
Representation of animal similarity (Henley, 1969) by ADDTREE.

shared by all objects in that cluster and by them alone. The length of the arc can thus be viewed as the weight of the respective features, or as a measure of the distinctiveness of the respective cluster. For example, the apes in Figure 8 form a highly distinctive cluster because the arc labeled 'apes' is very long. The interpretation of additive trees as feature trees is discussed in the last section.

The obtained (vertical) order of the animals in Figure 8 from top to bottom roughly corresponds to the dimension of size, with elephant and mouse at the two endpoints. The (horizontal) distance of an animal from the root reflects its average distance from other animals. For example, cat is closer to the root than tiger, and indeed cat is more similar, on the average, to other animals than tiger. Note that this property of the data cannot be represented in an ultrametric tree in which all objects are equidistant from the root.

The correspondence indices for animal similarity are given in Table 2.

Similarity of Letters

The second data set consists of similarity judgments between all lower-case Swedish letters obtained by Kuennapas and Janson [1969]. They reported average similarity ratings for 57 subjects using a 0–100 scale. The modified letters *å*, *ö*, *ä* are omitted from the present analysis. The result of SSA/2D is displayed in Figure 9. The type-set in the figure is essentially identical to that used in the experiment. The vertical dimension in Figure 9 might be interpreted as round-vs.-straight. No interpretable second dimension, however, emerges from the configuration.

The result of ADDTREE is presented in Figure 10 which reveals a distinct set of interpretable clusters. The obtained clusters exhibit excellent correspondence with the factors derived by Kuennapas and Janson [1969] via a principle-component analysis. These investigators obtained six major factors which essentially coincide with the clustering induced by the additive tree. The factors together with their high-loading letters are as follows:

Factor I: roundness (*o*, *c*, *e*)
Factor II: roundness attached to veritical linearity (*p*, *q*, *b*, *g*, *d*)
Factor III: parallel vertical linearity (*n*, *m*, *h*, *u*)
Factor IV: zigzaggedness (*s*, *z*)

TABLE 2

Correspondence Indices (Animals)

	ADDTREE	HCS	SSA/2D	SSA/3D
Stress	.07	.10	.17	.11
r	.91	.84	.86	.93

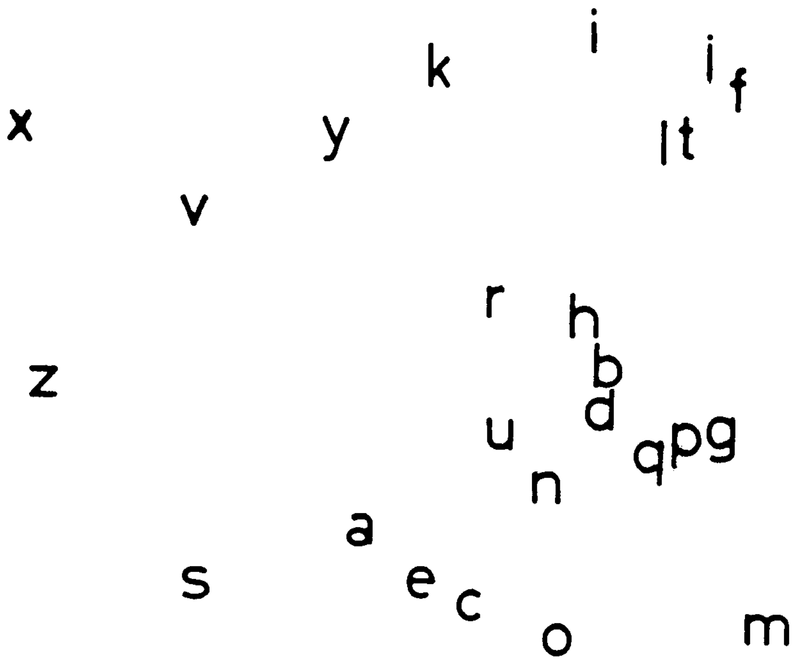

FIGURE 9
Representation of letter similarity (Kuennapas and Janson, 1969) by SSA/2D.

Factor V: angularity open upward (v, y, x)
Factor VI: vertical linearity (t, f, l, r, j, i)

The vertical ordering of the letters in Figure 10 is interpretable as roundness vs. angularity. It was obtained by the standard permutation procedure with the additional constraint that o and x are the end-points.

The correspondence indices for letter similarity are presented in Table 3.

Similarity of Occupations

Kraus [Note 5] instructed 154 Israeli subjects to classify 90 occupations into disjoint classes. The proximity between occupations was defined as the number of subjects who placed them in the same class. A representative subset of 35 occupations was selected for analysis.

The result of SSA/2D is displayed in Figure 11. The configuration could be interpreted in terms of two dimensions: white collar vs. blue collar, and autonomy vs. subordination. The result of ADDTREE is presented in Figure 12 which yields a coherent classification of occupations. Note that while some of the obtained clusters (e.g., blue collar, academicians) also emerge from Figure 11, others (e.g., security, business) do not. The vertical ordering of occupations produced by the program corresponds to collar color, with academic white collar at one end and manual blue collar at the other.

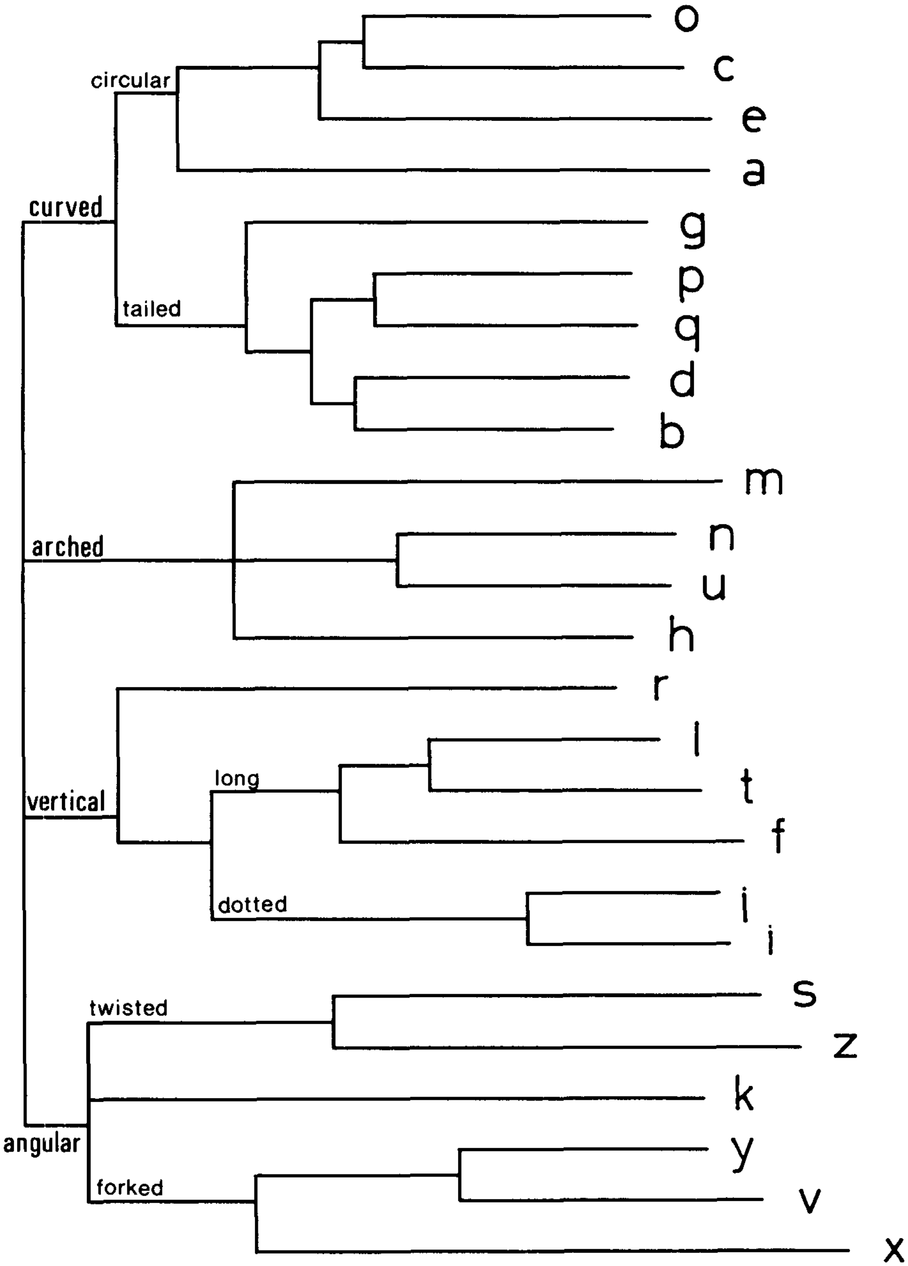

FIGURE 10
Representation of letter similarity (Kuennapas and Janson, 1969) by ADDTREE.

TABLE 3

Correspondence Indices (Letters)

	ADDTREE	HCS	SSA/2D	SSA/3D
Stress	.08	.11	.24	.16
r	.87	.82	.76	.84

The correspondence indices for occupations are presented in Table 4.

In the remainder of this section we comment on the robustness of tree structures and discuss the appropriateness of tree vs. spatial representations.

Robustness

The stability of the representations obtained by ADDTREE was examined using artificial data. Several additive trees (consisting of 16, 24, and 32 objects) were selected. Random error was added to the resulting distances according to the following rule: to each distance d we added a random number selected from a uniform distribution over $[-d/3, +d/3]$. Thus, the expected error of measurement for each distance is 1/6 of its length. Several sets of such data were analyzed by ADDTREE. The correlations between the solutions and the data were around .80. Nevertheless, the original tree-structures were recovered with very few errors indicating that tree structures are fairly robust. A

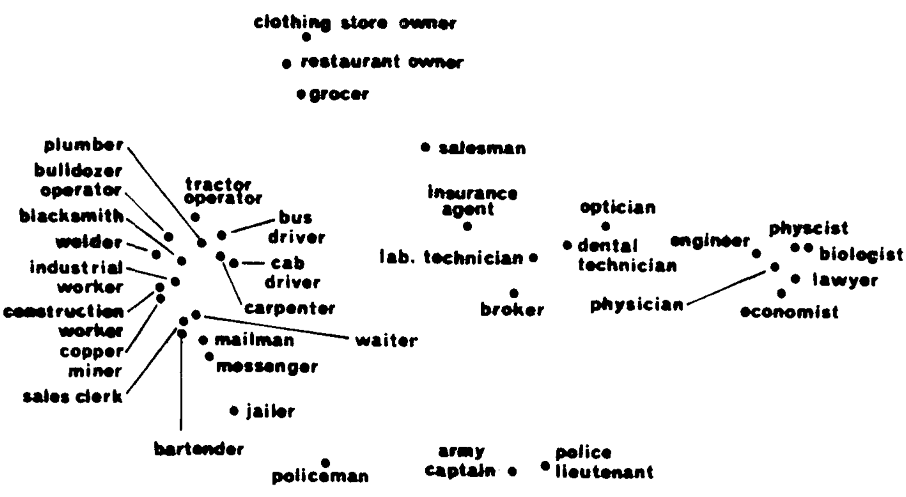

FIGURE 11
Representation of similarity between occupations (Kraus, 1976) by SSA/2D.

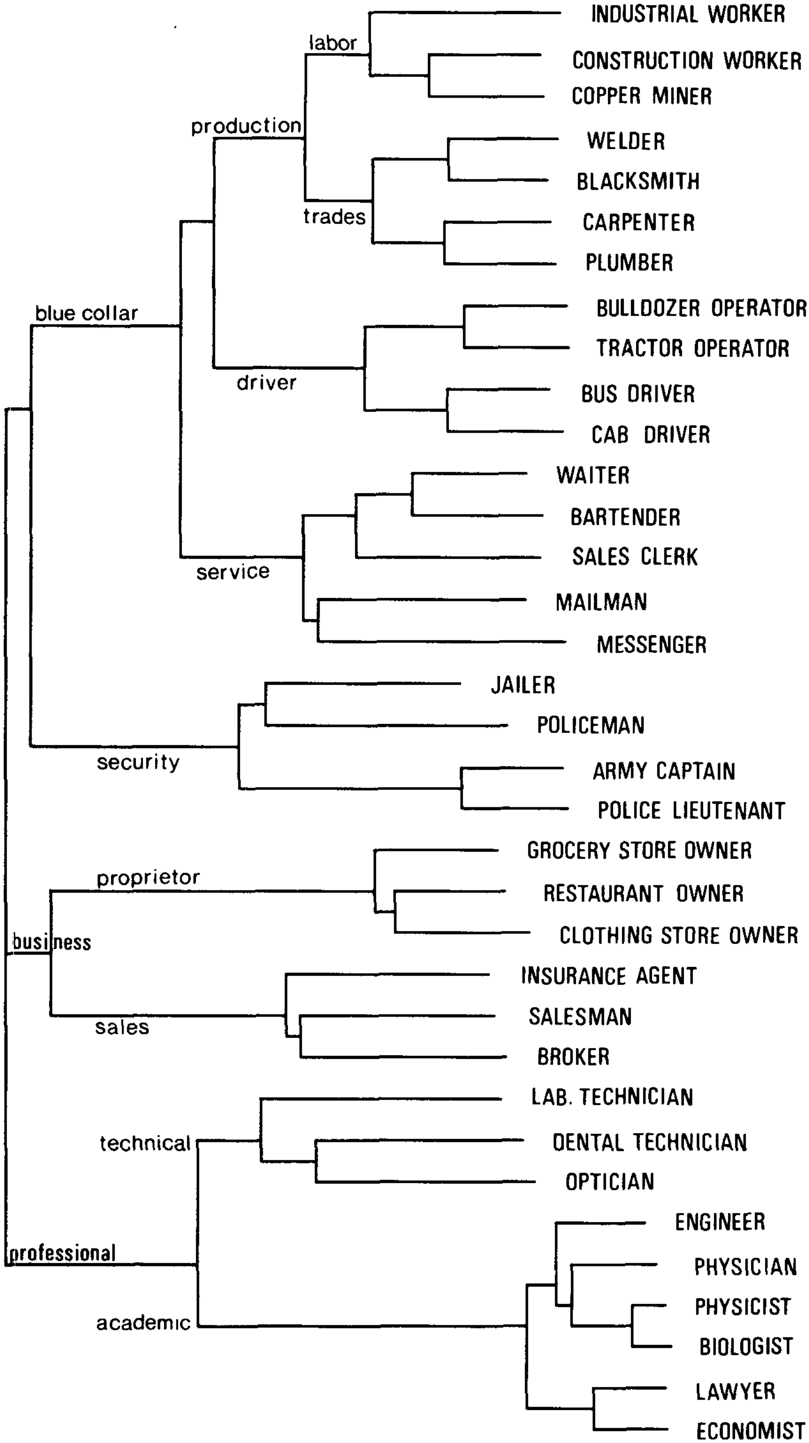

FIGURE 12
Representation of similarity between occupations (Kraus, 1976) by ADDTREE.

TABLE 4

Correspondence Indices (Occupations)

	ADDTREE	HCS	SSA/2D	SSA/3D
Stress	.06	.06	.15	.09
r	.96	.94	.86	.91

noteworthy feature of ADDTREE is that as the noise level increases, the internal arcs become shorter. Thus, when the signal-to-noise ratio is low, the major clusters are likely to be less distinctive.

In all three data sets analyzed above, the ordinal and the cardinal versions of ADDTREE produce practically the same tree-structures. This observation suggests that the tree-structure is essentially determined by the ordinal properties of the data. To investigate this question, we have performed order-preserving transformations on several sets of real and artificial data, and applied ADDTREE to them. The selected transformations were the following: ranking, and $d \to d^{\theta}$, $\theta = 1/4, 1/3, 1/2, 1, 2, 3, 4$. The obtained tree-structures for the different transformations were highly similar. There was a tendency, however, for the high-power transformations to produce non-centered subtrees such as Figure 1.

Tree vs. Spatial Representations

The applications of ADDTREE described above yielded interpretable tree structures. Furthermore, the tree distances reproduced the observed measures of similarity, or dissimilarity, to a reasonably high degree of approximation. The application of HCS to the same data yielded similar tree structures, but the reproduction of the observed proximities was, naturally, less satisfactory in all three data sets.

The comparison of ADDTREE with SSA indicates that the former provided a better account of the data than the latter, as measured by the product-moment correlation and by the stress coefficient. The fact that ADDTREE achieved lower stress in all data sets is particularly significant because SSA/3D has more free parameters, and it is designed to minimize stress while ADDTREE is not. Furthermore, while the clusters induced by the trees were readily interpretable, the dimensions that emerged from the spatial representations were not always readily interpretable. Moreover, the major dimension of the spatial solutions (e.g., size of animals, and prestige of occupations) also emerged as the vertical ordering in the corresponding trees.

These results indicate that some similarity data are better described by a tree than by a spatial configuration. Naturally, there are other data for which dimensional models are more suitable, see, e.g., Fillenbaum and Rapoport [1971], and Shepard [1974]. The appropriateness of tree vs. spatial representation depends on the nature of the task and the structure of the stimuli. Some object sets have a natural product structure, e.g., emotions may be described in terms of intensity and pleasantness; sound may be characterized in terms of intensity and frequency. Such object sets are natural candidates for dimensional representations. Other objects sets have a hierarchical structure that may result, for instance, from an evolutionary process in which all objects have an initial common structure and later develop additional distinctive features. Alternatively, a hierarchal structure may result from people's tendency to classify objects into mutually exclusive categories. The prevalence of hierarchical classifications can be attributed to the added complexity involved in the introduction of cross classifications with overlapping clusters. Structures generated by an evolutionary process or classification scheme are likely candidates for tree representations.

It is interesting to note that tree and spatial models are opposing in the sense that very simple configurations of one model are incompatible with the other model. For example, a square grid in the plane cannot be adequately described by an additive tree. On the other hand, an additive tree with a single internal node cannot be adequately represented by a non-trivial spatial model [Holman, 1972]. These observations suggest that the two models may be appropriate for different data and may capture different aspects of the same data.

Discussion

Feature Trees

As was noted earlier, a rooted additive tree can be interpreted as a feature tree. In this interpretation, each object is viewed as a set of features. Furthermore, each arc represents the set of features shared by all the objects that follow from that arc, and the arc length corresponds to the measure of that set. Hence, the features of an object are the features of all arcs which lead to that object, and its measure is its distance from the root. The tree-distance d between any two objects, therefore, corresponds to their set-distance, i.e., the measure of the symmetric difference between the respective feature sets:

$$d(x, y) = f(X - Y) + f(Y - X)$$

where X, Y are the feature sets associated with the objects x, y, respectively, and f is the measure of the feature space.

A more general model of similarity, based on feature matching, was developed in Tversky [1977]. In this theory, the dissimilarity between x and y is monotonically related to

$$d(x, y) = \alpha f(X - Y) + \beta f(Y - X) - \theta f(X \cap Y) \qquad \alpha, \beta, \theta \geq 0,$$

where X, Y, and f are defined as above. According to this form (called the contrast model) the dissimilarity between objects is expressed as a linear combination of the measures of their common and distinctive features. Thus, an additive tree is a special case of the contrast model in which symmetry and the triangle inequality hold, and the feature space has a tree structure.

Decomposition of Trees

There are three types of additive trees that have a particularly simple structure: ultrametric, singular, and linear. In an ultrametric tree all objects are equidistant from the root. A *singular* tree is an additive tree with a single internal node. A *linear* tree, or a line, is an additive tree in which all objects lie on a line (see Figure 13). Recall that an additive tree is ultrametric iff it satisfies the ultrametric inequality. An additive tree is singular iff for each object x in S there exists a length $\bar{x}$ such that $d(x, y) = \bar{x} + \bar{y}$. An additive tree is a line iff the triangle equality $d(x, y) + d(y, z) = d(x, z)$ holds for any three elements in S. Note that all three types of trees have no more than n parameters.

Throughout this section let T, T_1, T_2, etc. be additive trees defined on the same set of objects. T_1 is said to be *simpler* than T_2 iff the graph of T_1 (i.e., the structure without the metric) is obtained from the graph of T_2 by cancelling one or more internal arcs and joining their endpoints. Hence, a singular tree is simpler than any other tree defined on the same object set. If T_1 and T_2 are both simpler than some T_3, then T_1 and T_2 are said to be *compatible*. (Note that compatibility is not transitive.) Let d_1 and d_2 denote, respectively, the distance functions of T_1 and T_2. It is not difficult to prove that the distance function $d = d_1 + d_2$ can be represented by an additive tree iff T_1 and T_2 are compatible. (Sufficiency follows from the fact that the sum of two trees with the same graph is a tree with the same graph. The proof of necessity relies on the fact that for any two incompatible trees there exists a quadruple on which they are incompatible.)

This result indicates that data which are not representable by a single additive tree may nevertheless be represented as the sum of incompatible additive trees. Such representations are discussed by Carroll and Pruzansky [Note 3].

Another implication of the above result is that tree-structures are preserved by the addition of singular trees. In particular, the sum of an ultrametric tree T_U and a singular tree T_S is an additive tree T with the same graph as T_U (see Figure 13). This leads to the converse question: can an additive tree T be expressed as $T_U + T_S$? An interesting observation (attributed to J. S. Farris) is that the distance function d of an additive tree T can be expressed as $d(x, y) = d_U(x, y) + \bar{x} + \bar{y}$, where d_U is the distance function of an ultrametric tree, and $\bar{x}$, $\bar{y}$ are real numbers (not necessarily positive). If all these numbers are nonnegative then d is decomposable into an ultrametric and a singular tree, i.e., $d = d_U + d_S$. It is readily verified that T is expressible as $T_U + T_S$ iff there is a point on T whose distance to any internal node does not exceed its distance to any external node. Another structure of interest is obtained by the addition of a singular tree T_S and a line T_L (see Figure 13). It can be shown that an additive tree T is expressible as $T_S + T_L$ iff no more than two internal arcs meet at any node.

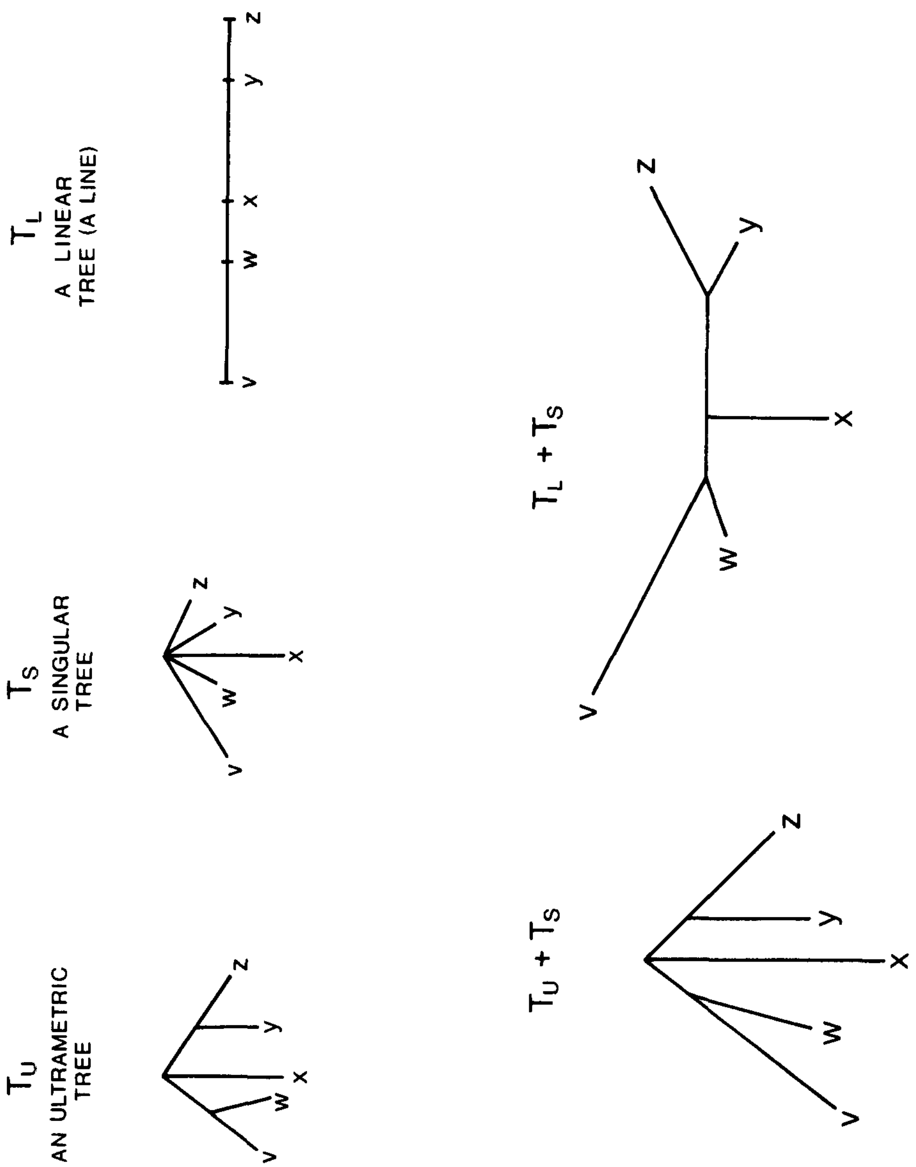

FIGURE 13
An illustration of different types of additive trees.

Distribution of Distances

Figure 14 presents the distribution of dissimilarities between letters [from Kuennapas & Janson, 1969] along with the corresponding distributions of distances derived via ADDTREE, and via SSA/2D. The distributions of

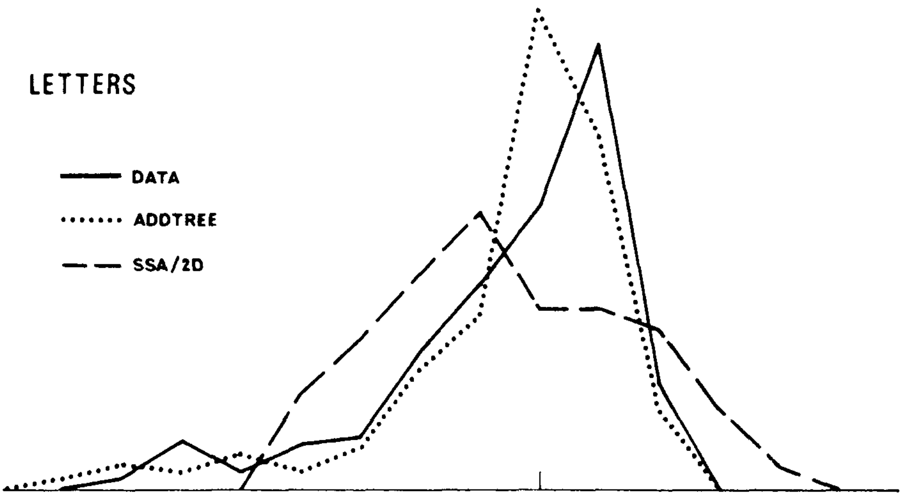

FIGURE 14
Distributions of dissimilarities and distances between letters.

derived distances were standardized so as to have the same mean and variance as the distribution of the observed dissimilarities.

Note that the distribution of dissimilarities and the distribution of distances in the additive tree are skewed to the left, whereas the distribution of distances from the two-dimensional representation is skewed to the right. This pattern occurs in all three data sets, and reflects a general phenomenon.

In an additive tree, there are generally many large distances and few small distances. This follows from the observation that in most rooted trees, there are fewer pairs of objects that belong to the same cluster than pairs of objects that belong to different clusters. In contrast, a convex Euclidean configuration yields many small distances and fewer large distances. Indeed, under fairly natural conditions, the two models can be sharply distinguished by the skewness of their distance distribution.

The skewness of a distribution can be defined in terms of different criteria, e.g., the relation between the mean and the median, or the third central moment of the distribution. We employ here another notion of skewness that is based on the relation between the mean and the midpoint. A distribution is skewed to the left, according to the mean-midpoint criterion, iff the mean μ exceeds the midpoint $\lambda = 1/2 \max_{x,y} d(x, y)$. The distribution is skewed to the right, according to the mean-midpoint criterion, iff $\mu < \lambda$. From a practical standpoint, the mean-midpoint criterion has two drawbacks. First, it requires ratio scale data. Second, it is sensitive to error since it depends on the maximal distance. As demonstrated below, however, this criterion is useful for the investigation of distributions of distances.

A rooted additive tree (with n objects) is *centered* iff no subtree contains more than $n/2 + n(\text{mod } 2)$ objects. (Note that this bound is $n/2$ when n is even, and $(n + 1)/2$ when n is odd.) In an additive tree, one can always select a root such that the resulting rooted tree is centered. For example, the tree in Figure 2 is centered around its root, whereas the tree in Figure 1 is not. We can now state the following.

Skewness Theorem. I. Consider an additive tree T that is expressible as a sum of an ultrametric tree T_U and a singular tree T_S such that (i) T_U is centered around its natural root, and (ii) in T_S the longest arc is no longer than twice the shortest arc. Then the distribution of distances satisfies $\mu > \lambda$.

II. In a bounded convex subset of the Euclidean plane with the uniform measure, the distribution of distances satisfies $\mu < \lambda$.

Part I of the theorem shows that in an additive tree the distribution of distances is skewed to the left (according to the mean-midpoint criterion) whenever the distances between the centered root and the external nodes do not vary "too much". This property is satisfied, for example, by the trees in Figures 8 and 10, and by T_U, T_S, and $T_U + T_S$ in Figure 13. Part II of the theorem shows that in the Euclidean plane the distribution of distances is skewed to the right, in the above sense, whenever the set of points "has no holes". The proof of the Skewness Theorem is given in the Appendix.

The theorem provides a sharp separation of these two families of representations in terms of the skewness of their distance distribution. This result does not hold for additive trees and Euclidean representations in general. In particular, it can be shown that the distribution of distances between all points on the circumference of a circle (which is a Euclidean representation, albeit non-convex) is skewed to the left. This fact may explain the presence of "holes" in some configurations obtained through multidimensional scaling [see Cunningham, Note 3, Figure 1.1]. It can also be shown that the distribution of distances between all points on a line (which is a limiting case of an additive tree which cannot be expressed as $T_U + T_S$) is skewed to the right. Nevertheless, the available computational and theoretical evidence indicates that the distribution of distances in an additive tree is generally skewed to the left, whereas in a Euclidean representation it is generally skewed to the right. This observation suggests the intriguing possibility of evaluating the appropriateness of these representations on the basis of distributional properties of observed dissimilarities.

Appendix: Proof of the Skewness Theorem

Part I

Consider an additive tree $T = T_U + T_S$ with n external nodes. Hence,

$$\mu = \frac{\sum d(x, y)}{\binom{n}{2}} = \frac{\sum d_U(x, y)}{\binom{n}{2}} + \frac{\sum d_S(x, y)}{\binom{n}{2}} = \mu_U + \mu_S$$

and

$$\lambda = \tfrac{1}{2} \max d(x, y) \leq \lambda_U + l_S$$

where $\lambda_U = 1/2 \max d_U(x, y)$ is the distance between the root and the external nodes in the ultrametric tree, and l_S is the length of the longest arc in the singular tree. To show that T satisfies $\mu > \lambda$, it suffices to establish the inequalities: $\mu_U > \lambda_U$ and $\mu_S > l_S$ for its ultrametric and singular components. The inequality $\mu_S > l_S$ follows at once from the assumption that, in the singular tree, the shortest arc is not less than half the longest arc. To prove $\mu_U > \lambda_U$, suppose the ultrametric tree has k subtrees, with $n_1, n_2, \cdots n_k$ objects, that originate directly from the root. Since the tree is centered $n_i \leq n/2 + n(\text{mod } 2)$ where $n = \sum_i n_i$. Clearly $\mu_U = \sum_{x,y} d_U(x, y)/n(n - 1)$. We show that $\sum_{x,y} d_U(x, y) > n(n - 1)\lambda_U$.

Let P be the set of all pairs of objects that are connected through the root. Hence,

$$\sum_{(x,y)} d_U(x, y) \geq \sum_{(x,y)\epsilon P} d_U(x, y) = 2\lambda_U \sum_{i=1}^{k} n_i(n - n_i)$$

where the equality follows from the fact that $d_U(x, y) = 2\lambda_U$ for all (x, y) in P. Therefore, it suffices to show that $2 \sum_i n_i(n - n_i) > n(n - 1)$, or equivalently that $n^2 + n > 2 \sum_i n_i^2$. It can be shown that, subject to the constraint $n_i \leq n/2 + n(\text{mod } 2)$, the sum $\sum_i n_i^2$ is maximal when $k = 2$. In this case, it is easy to verify that $n^2 + n > 2(n_1^2 + n_2^2)$ since $n_1, n_2 = n/2 \pm n(\text{mod } 2)$.

Part II

Croften's Second Theorem on convex sets [see Kendall & Moran, 1963, pp. 64-66] is used to establish Part II of the Skewness Theorem.

Let S be a bounded convex set in the plane, hence

$$\mu = \frac{\iiiint_S ((x_1 - x_2)^2 + (y_1 - y_2)^2)^{1/2}\, dx_1\, dy_1\, dx_2\, dy_2}{\iiiint_S dx_1\, dy_1\, dx_2\, dy_2}$$

We replace the coordinates (x_1, y_1, x_2, y_2) by $(p, \theta, \rho_1, \rho_2)$ where p and θ are the polar coordinates of the line joining (x_1, y_1) and (x_2, y_2), and ρ_1, ρ_2 are the distances from the respective points to the projection of the origin on that line. Thus,

$$x_1 = \rho_1 \sin\theta + p\cos\theta, \qquad y_1 = -\rho_1 \cos\theta + p \sin\theta,$$
$$x_2 = \rho_2 \sin\theta + p\cos\theta, \qquad y_2 = -\rho_2 \cos\theta + p \sin\theta.$$

Since the Jacobian of this transformation is $\rho_2 - \rho_1$,

$$\mu = \frac{\iiiint |\rho_1 - \rho_2|^2 \, d\rho_1 \, d\rho_2 \, dp \, d\theta}{\iiiint |\rho_1 - \rho_2| \, d\rho_1 \, d\rho_2 \, dp \, d\theta} .$$

To prove that $\mu < \lambda$ we show that for every p and θ

$$\lambda \geq \frac{\iint |\rho_1 - \rho_2|^2 \, d\rho_1 d\rho_2}{\iint |\rho_1 - \rho_2|^2 \, d\rho_1 d\rho_2}.$$

Given some p and θ, let L be the length of the cord in S whose polar coordinates are p, θ. Hence,

$$\begin{aligned}
\int_a^b d\rho_1 \int |\rho_1 - \rho_2|^n \, d\rho_2 &= \int_a^b d\rho_1 \left(\int_a^{\rho_1} (\rho_1 - \rho_2)^n \, d\rho_2 + \int_{\rho_1}^b (\rho_2 - \rho_1)^n \, d\rho_2 \right) \\
&= \int_a^b d\rho_1 \left(- \left. \frac{(\rho_1 - \rho_2)^{n+1}}{n+1} \right|_a^{\rho_1} + \left. \frac{(\rho_2 - \rho_1)^{n+1}}{n+1} \right|_{\rho_1}^b \right) \\
&= \int_a^b \frac{(\rho_1 - a)^{n+1} + (b - \rho_1)^{n+1}}{n+1} \, d\rho_1 \\
&= \frac{1}{(n+1)(n+2)} \left. ((\rho_1 - a)^{n+2} - (b - \rho_1)^{n+2}) \right|_a^b \\
&= \frac{(b-a)^{n+2} + (b-a)^{n+2}}{(n+1)(n+2)} \\
&= \frac{2L^{n+2}}{(n+1)(n+2)}
\end{aligned}$$

where a and b are the distances from the endpoints of the chord to the projection of the origin on that chord, whence $L = b - a$. Consequently,

$$\frac{\iint |\rho_1 - \rho_2|^2 \, d\rho_1 \, d\rho_2}{\iint |\rho_1 - \rho_2| \, d\rho_1 \, d\rho_2} = \frac{L}{2} \leq \lambda$$

since λ is half the supremal chord-length. Moreover, $L/2 < \lambda$ for a set of chords with positive measure, hence $\mu < \lambda$.

REFERENCE NOTES

1. Holman, E. W. *A test of the hierarchical clustering model for dissimilarity data.* Unpublished manuscript, University of California at Los Angeles, 1975.
2. Cunningham, J. P. *Finding an optimal tree realization of a proximity matrix.* Paper presented at the mathematical psychology meeting, Ann Arbor, August, 1974.

3. Cunningham, J. P. *Discrete representations of psychological distance and their applications in visual memory*. Unpublished doctoral dissertation, University of California at San Diego, 1976.
4. Carroll, J. D. & Pruzansky, S. *Fitting of hierarchical tree structure (HTS) models, mixture of HTS models, and hybrid models, via mathematical programming and alternating least squares.* paper presented at the US-Japan Seminar on Theory, Methods, and Applications of Multidimensional Scaling and related techniques, San Diego, August, 1975.
5. Kraus, personal communication, 1976.

REFERENCES

Buneman, P. The recovery of trees from measures of dissimilarity. In F. R. Hodson, D. G. Kendall & P. Tautu (Eds.), *Mathematics in the Archaeological and Historical Sciences.* Edinburgh: Edinburgh University Press, 1971.

Buneman, P. A note on the metric properties of trees. *Journal of Combinatorial Theory,* 1974, *17*(B), 48–50.

Carroll, J. D. Spatial, non-spatial and hybrid models for scaling. *Psychometrika,* 1976, *41,* 439–463.

Carroll, J. D. & Chang, J. J. A method for fitting a class of hierarchical tree structure models to dissimilarities data and its application to some "body parts" data of Miller's. *Proceedings, 81st Annual Convention, American Psychological Association,* 1973, *8,* 1097–1098.

Dobson, J. Unrooted trees for numerical taxonomy. *Journal of Applied Probability,* 1974, *11,* 32–42.

Fillenbaum, S. & Rapoport, A. *Structures in the subjective lexicon.* New York: Academic Press, 1971.

Guttman, L. A general nonmetric technique for finding the smallest coordinate space for a configuration of points. *Psychometrika,* 1968, *33,* 469–506.

Hakimi, S. L. & Yau, S. S. Distance matrix of a graph and its realizability. *Quarterly of Applied Mathematics,* 1964, *22,* 305–317.

Henley, N. M. A psychological study of the semantics of animal terms. *Journal of Verbal Learning and Verbal Behavior,* 1969, *8,* 176–184.

Holman, E. W. The relation between hierarchical and Euclidean models for psychological distances. *Psychometrika,* 1972, *37,* 417–423.

Jardine, N., & Sibson, R. *Mathematical taxonomy.* New York: Wiley, 1971.

Johnson, S. C. Hierarchical clustering schemes. *Psychometrika,* 1967, *32,* 241–254.

Kendall, M. G., & Moran, M. A. *Geometrical Probability.* New York: Hafner Publishing Company, 1963.

Kruskal, J. B. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika,* 1964, *29,* 1–27.

Kuennapas, T., & Janson, A. J. Multidimensional similarity of letters. *Perceptual and Motor Skills,* 1969, *28,* 3–12.

Lingoes, J. C. An IBM 360/67 program for Guttman–Lingoes smallest space analysis-PI. *Behavioral Science,* 1970, *15,* 536–540.

Patrinos, A. N., & Hakimi, S. L. The distance matrix of a graph and its tree realization. *Quarterly of Applied Mathematics,* 1972, *30,* 255–269.

Shepard, R. N. Representation of structure in similarity data: Problems and prospects. *Psychometrika,* 1974, *39,* 373–421.

Sneath, P. H. A., & Sokal, R. R. *Numerical taxonomy: the principles and practice of numerical classification.* San Francisco: W. H. Freeman, 1973.

Turner, J., & Kautz, W. H. A survey of progress in graph theory in the Soviet Union. *Siam Review,* 1970, *12,* 1–68. (Supplement)

Tversky, A. Features of similarity. *Psychological Review,* 1977, *84,* 327–352.

Manuscript received 4/27/76
Final version received 10/13/76