Chapter 8 *Cluster analysis*

8.0 A search for discontinuities

Humans have always tried to classify the animate and inanimate objects that surround them. Classifying objects into collective categories is a prerequisite to naming them. It requires the recognition of discontinuous subsets in an environment that is sometimes discrete, but most often continuous.

To cluster is to recognize that objects are sufficiently similar to be put in the same group and to also identify distinctions or separations between groups of objects. Measures of resemblance between objects (Q mode) or descriptors (R mode) have been discussed in Chapter 7. The present chapter considers the different criteria that may be used to decide whether objects are similar enough to be allocated to the same group when several groups have been defined, and shows that different clustering strategies correspond to different definitions of a what a cluster is. The chapter also examines special clustering approaches that are used to identify species associations.

Few ecological theories predict the existence of discontinuities in nature. Evolutionary theory tells taxonomists that discontinuities exist between species, which are the basic units of evolution, as a result of reproductive barriers; taxonomists use classification methods to reveal these discontinuities. For the opposite reason, taxonomists are not surprised to find continuous differentiation at the sub-species level. In contrast, the world that ecologists try to understand is most often a continuum. In numerical ecology, methods used to identify clusters must therefore be more contrasting than in numerical taxonomy.

Given a sufficiently large group of objects, ecological clustering methods should be able to recognize clusters of similar objects while ignoring the few intermediates that often persist between clusters. Indeed, one cannot expect to find discontinuities when clustering sampling sites unless the physical environment is itself discontinuous, or unless sampling occurred at opposite ends of a gradient, instead of within the gradient (Whittaker, 1962: 88). Similarly, when looking for associations of species, small groups of densely associated species are usually found, with the other species gravitating around one or more of the association nuclei.

Typology

The result of clustering ecological objects sampled from a continuum is often called a *typology* (i.e. a system of types). In such a case, the purpose of clustering is to identify various *object types*, which may be used to describe the structure of the continuum; it is thus immaterial to wonder whether these clusters are "natural" or unique.

For readers with no practical experience in clustering, Section 8.2 provides a detailed account of single linkage clustering, which is simple to understand and is used to introduce the principles of clustering. The review of other methods includes a survey of the main dichotomies among existing methods (Section 8.4), followed by a discussion of the most widely available methods of interest to ecologists (Sections 8.5, 8.7 and 8.8). Theoretical aspects are examined in Sections 8.3 and 8.6. Section 8.9 discusses clustering algorithms useful in identifying biological associations and indicator species analysis, whereas Section 8.10 gives an overview of seriation, a method useful in particular to cluster non-symmetric resemblance matrices. Section 8.11 describes multivariate regression tree analysis (MRT), a method that involves two data sets, i.e. response and explanatory, whose output is a tree. A review of clustering statistics, methods of cluster validation, and graphical representations, completes the chapter (Sections 8.12 to 8.14). The relationships between clustering and other steps of data analysis are depicted in Fig. 10.3.

Despite the wide applicability of clustering methods, one should remember that no single family of methods can answer all questions raised in numerical ecology. Before engaging in clustering, one should be able to justify why one believes that discontinuities exist in the data or explain why one has a practical need to divide a continuous set of objects into groups.

8.1 Definitions

Clustering
Partition

Clustering is an operation of multidimensional analysis that consists in partitioning a collection of objects or descriptors. Most of the methods described in this chapter can be used to cluster descriptors instead of objects. The presentation in the chapter focuses on objects for simplicity, except in Section 8.9 where methods especially designed to cluster species into associations are described. Explanatory variables can also be clustered to identify groups of collinear variables. A *partition* is a division of a set (collection) into subsets, such that each object belongs to one and only one subset for that partition (Legendre & Rogers, 1972). The classification of objects that results may include a single partition, or several hierarchically nested partitions of the objects (or descriptors), depending on the clustering model that has been selected (Table 8.1).

The clustering methods described in this chapter belong to the class of *hard* or *crisp* clustering, where the groups are mutually exclusive and each object belongs to a single group of a partition. In *fuzzy clustering* on the contrary (Bezdek, 1987), an object may simultaneously belong, to different degrees, to two or more groups of a

Table 8.1 Example of hierarchically nested partitions of a set of objects (e.g. sampling sites). The first partition divides the objects according to the environment where they were found. The second partition, hierarchically nested within the first, describes clusters of sites in each environment.

Partition 1	Partition 2	Sampling sites
Observations in environment A	Cluster 1	7, 12
	Cluster 2	3, 5, 11
	Cluster 3	1, 2, 6
Observations in environment B	Cluster 4	4, 9
	Cluster 5	8, 10, 13, 14

partition. In the study of species associations for example (Section 8.9), this approach is interesting because a species may be partly related to two or more associations. Methods of fuzzy clustering are not described in detail in this chapter but R software to compute them is mentioned in Section 8.15.

A partition resulting from a hard clustering method has the same definition as a descriptor (Section 1.4). Each object is characterized by a state (its cluster) of the classification and it belongs to only one of the clusters. This property is useful for the interpretation of classifications (Chapter 10) since any partition may be considered as a qualitative descriptor and compared as such to any other descriptor. A clustering of objects defined in this way imposes a discontinuous structure onto the data set, even if the objects have originally been sampled from a continuum. This structure results from the grouping into subsets of objects that are recognized as sufficiently similar given the variables considered. One can then look for characteristics that differentiate the clusters from one another.

Clustering has been part of ecological tradition for a long time. It goes back to the Polish ecologist Kulczynski (1928) who needed to cluster ecological observations; he developed a method quite remote from the modern clustering algorithms. The technique, called seriation, consisted in permuting the rows and columns of an association matrix in such a way as to get the largest values near the diagonal. The method is still used in phytosociology, anthropology, the social sciences, and other fields. Analytical solutions to the seriation problem are mentioned in Section 8.10.

Most methods of clustering (this chapter) and ordination (Chapter 9) proceed from association matrices (Chapter 7). Distinguishing between clustering and ordination

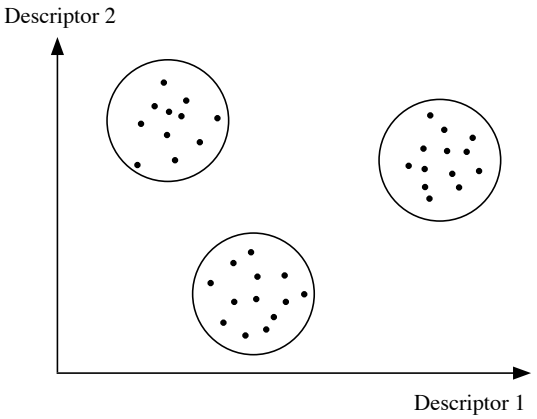


Figure 8.1 Empirically delineating clusters of objects in a scatter diagram is easy when there are no intermediate objects between the groups.

methods is somewhat recent. While ordination in reduced space goes back to Spearman (factor analysis: 1904), most modern clustering methods have only been developed since the era of second-generation computers. The first programmed method was developed by Sokal & Michener (1958) for biological purposes*. Before that, one simply plotted the objects in a scatter diagram with respect to a few variables or principal axes; clusters were then delineated manually (Fig. 8.1) following a method that, today, would be called centroid (Section 8.5) and based upon the Euclidean distances among points. This empirical clustering method still remains a valid approach when the number of variables is small and the structure to be delineated is not obscured by the presence of intermediate objects between the clusters.

Clustering is a family of methods undergoing rapid development. In their report on the literature they reviewed, Blashfield & Aldenderfer (1978) mentioned that they found 25 papers in 1964 that contained references to the basic texts on clustering; then they found 136 papers in 1970, 294 in 1973, and 501 in 1976. The number has been growing ever since. Nowadays, hundreds of mathematicians and researchers from various application fields are collaborating within national or multinational *Classification Societies* throughout the world, under the umbrella of the *International Federation of Classification Societies* founded in 1985.

* Historical note provided by Prof. F. James Rohlf: "Actually, Sokal & Michener (1958) did not use a computer for their very large study. They used an electromechanical accounting machine to compute the raw sums and sums of products. The coefficients of correlation and the cluster analysis itself were computed by hand with the use of mechanical desk calculators. Sneath did use a computer in his first study."

The commonly-used clustering methods are based on easy-to-understand mathematical constructs: arithmetic, geometric, graph-theoretic, or simple statistical models (minimizing within-group variance), leading to rather simple calculations on the dissimilarity or similarity values. It must be understood that most clustering methods are heuristic; they create groups by reference to some concept of what a group embedded in some space should be like, without reference, in most case, to the processes occurring in the application field — ecology in the present book. They have been developed by numerical taxonomists and numerical ecologists, later joined by other researchers in the physical sciences, economics and humanities. In several methods, clusters are delineated on the basis of statements such as: "$\mathbf{x}_1$ is closer to $\mathbf{x}_2$ than it is to $\mathbf{x}_3$", whereas other methods rest on probabilistic models of the type: "Chances are higher that $\mathbf{x}_1$ and $\mathbf{x}_2$ pertain to the same group than $\mathbf{x}_1$ and $\mathbf{x}_3$". In all cases, clustering models make it possible to link the points without requiring prior positioning in a graph (i.e. a metric space), which would be impractical in more than three dimensions. These models allow a graphical representation of other interesting relationships among the objects of the data set than their positions in a reference space of variables, for example the dendrogram of their hierarchical relationships. Chapter 10 will show how it is possible to combine clustering and ordination, computed with different methods, to obtain a more complete picture of the data structure.

The choice of a clustering method is as critical as the choice of an association measure. It is important to fully understand the properties of clustering methods in order to correctly interpret the ecological structure they bring out. Most of all, the methods to be used depend upon the type of clustering sought. Williams *et al*. (1971) recognized two major categories of methods. In a *descriptive clustering*, misclassifying objects is to be avoided, even at the expense of creating single-object clusters. In a *synoptic clustering*, all objects are forced into one of the main clusters; the objective is to construct a general conceptual model that encompasses a reality wider than the data under study. Both approaches have their usefulness.

Descriptive, synoptic clustering

When two or more clustering models seem appropriate to a problem, one should apply them all to the data and compare the results. Clusters that repeatedly come out of analyses that use appropriate methods are the robust solutions to the clustering problem. Differences among results must be interpreted in the light of the known properties of the clustering models, which are explained in the following sections.

8.2 The basic model: single linkage clustering

For natural scientists, a simple-to-understand clustering method (or *model*) is *single linkage* (or *nearest neighbour*) clustering (Sneath, 1957). Its logic seems natural, so that it is used to introduce readers to the principles of clustering. Its name, *single linkage*, distinguishes it from other clustering models, called complete or intermediate

linkage, detailed in Section 8.5. The algorithm for single linkage clustering is sequential, agglomerative and hierarchical, following the nomenclature of Section 8.4. Its starting point is any association matrix (distance or similarity) among the objects to be clustered. One assumes that the association measure has been carefully chosen, following the recommendations of Section 7.6. In the examples that follow, a distance matrix **D** will be used as the starting point for clustering because this is the standard in the clustering functions of the R language.

The method proceeds in two steps:

- First, the association matrix is rewritten in order of increasing distances or decreasing similarities, heading the list with the two closest objects (smallest distance) of the association matrix, followed by the second most similar pair, and proceeding until all the measures comprised in the association matrix have been listed.

- Second, the clusters are formed hierarchically, starting with the two closest objects, and then letting the objects combine into groups, and the groups aggregate to one another, as the distance increases. The following example illustrates this method.

Ecological application 8.2

Five ponds characterized by 38 zooplankton species were studied by Legendre & Chodorowski (1977). The data were counts, recorded on a relative abundance scale from 0 = absent to 5 = very abundant. These ponds have been used as example for the computation of Goodall's coefficient (S_{23}, Chapter 7; only eight zooplankton species were used in that example). These five ponds, with others (see Ecological application 10.1), were subjected to single linkage clustering after computing similarity coefficient S_{20} with parameter $k = 2$. The symmetric similarity matrix, transformed into distances using the equation $D = 1 - S$, is represented by its lower triangle. The diagonal is trivial because it contains distances of 0 by construct.

Ponds	Ponds				
	212	214	233	431	432
212	—				
214	0.400	—			
233	1.000	0.929	—		
431	1.000	0.937	0.700	—	
432	1.000	0.786	0.800	0.500	—

The first clustering step consists in rewriting the distance values in increasing order:

$D = 1 - S_{20}$	Pairs formed
0.400	212-214
0.500	431-432
0.700	233-431
0.786	214-432
0.800	233-432
0.929	214-233
0.937	214-431
1.000	212-233
1.000	212-431
1.000	212-432

Link

As the distance levels increases, pairs of objects are formed. These pairs are called "links"; they serve to link the objects or groups into a chain, as discussed below.

Connected subgraphs are one of the many possible graphical representations of cluster formation (Fig. 8.2a). As the distance increases, clusters are formed, following the list of links in the table of ordered distances above. Only the distance levels at which clusters are modified by addition of objects are represented in the figure. The first link is formed between ponds 212 and 214 at $D = 0.4$, then between 431 and 432 at $D = 0.5$. Pond 233 joins this second cluster nucleus at $D = 0.7$. Finally these two clusters merge at $D = 0.786$ due to a link formed between ponds 214 and 432. The clustering may stop at this point since, according to the single linkage rule (below), all ponds now belong to the same cluster. If the distance criterion is allowed to relax down to $D = 1$ (Fig. 8.2a), links form between members of the cluster up to a point where all ponds are linked to one another. That part of the clustering is of no interest in single linkage clustering, but these links will be of interest in the other forms of linkage clustering below.

Dendrogram

Edge

Node

A dendrogram (Fig. 8.2b) is a commonly-used representation of hierarchical clustering results. Dendrograms only display the clustering topology and object labels, not the links between objects. Dendrograms are made of branches ("edges") that meet at "nodes" which are drawn at the distance values where fusions of branches takes place. For graphical convenience, vertical lines are used in Fig. 8.2b to connect branches at the distance levels of the nodes; the lengths of these lines are of no consequence. Branches could be connected directly to nodes. The branches furcating from a node may be switched ("swivelled") without affecting the information contained in a dendrogram.

The clustering results were interpreted by Legendre & Chodorowski (1977) with respect to the conditions prevailing in the ponds. In their larger study summarized in Ecological application 10.1, all non-permanent ponds (including 212 and 214) formed a cluster while the permanent ponds (including 233, 431 and 432) formed a distinct group (Fig. 10.2).

Single linkage rule

From this example, it should be clear that the rule for assigning an object to a cluster, in single linkage clustering, requires that the object be no more distant than the considered D level from *at least one object already member of the cluster.* In complete linkage hierarchical clustering (Subsection 8.5.2), the assignment rule differs and requires the object to be no more distant than the given level from *all* the objects

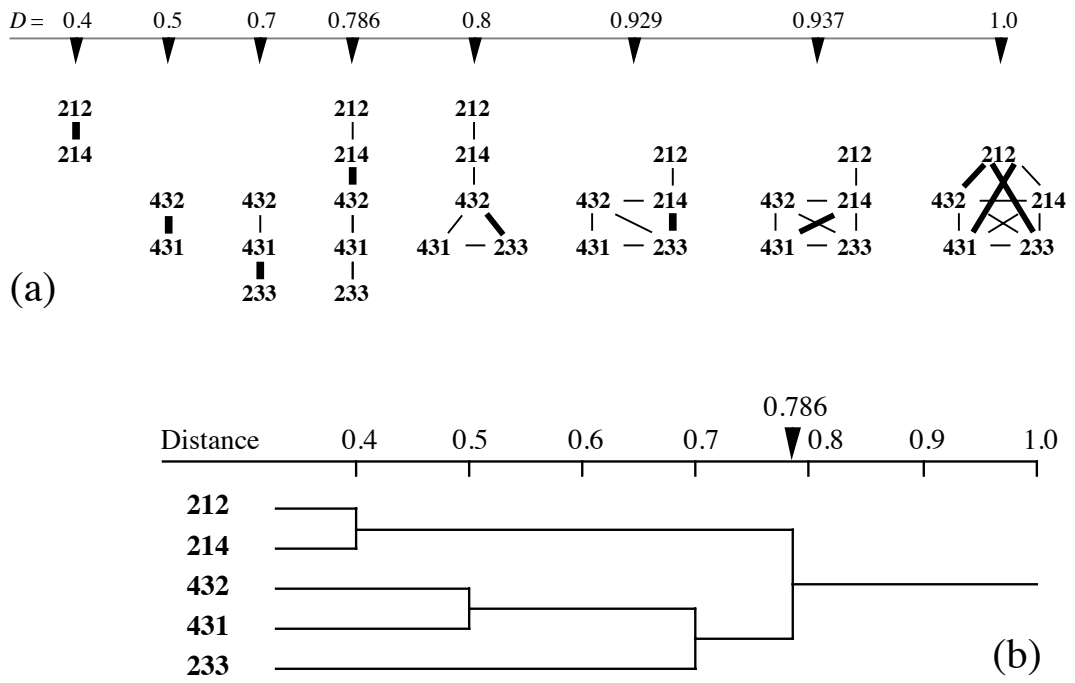


Figure 8.2 Illustrations of single linkage agglomerative clustering for the ponds in the example. (a) Connected subgraphs: groups of objects are formed as the distance level is relaxed from left to right. The levels where clusters are modified by addition of objects are represented; they are ordered along the distance scale (D). New links between ponds are represented by heavy lines; thin lines are used for links formed at previous (lower) distance levels. (b) Dendrogram representing the result of single linkage clustering.

already members of the cluster. The chaining rule used in single linkage clustering may be stated as follows: at each partition level, two objects must be allocated to the same subset if their dissimilarity (distance) is less than or equal to that of the partitioning level considered. The same rule can be formulated in terms of similarities: two objects must be allocated to the same subset if their similarity is equal to or higher than that of the partitioning level considered.

Estabrook (1966) discussed single linkage clustering using the language of graph theory. The exercise has didactic value. A cluster is defined through the following steps:

Link

a) For any pair of objects $\mathbf{x}_1$ and $\mathbf{x}_2$, a *link* is defined between them by a relation G_c:

$$\mathbf{x}_1 \, G_c \, \mathbf{x}_2 \text{ if and only if } D(\mathbf{x}_1, \mathbf{x}_2) \leq c$$

$$\text{or equally, if } S(\mathbf{x}_1, \mathbf{x}_2) \geq (1 - c)$$

assuming distances between 0 and 1. Index c in the clustering relation G_c is the distance level considered. At a distance level of 0.45, for instance, ponds 212 and 214 of the example are in relation $G_{0.45}$ since $D(212, 214) \leq 0.45$. This definition of a link has the properties of symmetry ($\mathbf{x}_1\ G_c\ \mathbf{x}_2$ if and only if $\mathbf{x}_2\ G_c\ \mathbf{x}_1$) and reflexivity ($\mathbf{x}_i\ G_c\ \mathbf{x}_i$ is always true since $D(\mathbf{x}_i, \mathbf{x}_i) = 0$). A group of links for a set of objects, such as defined by relation G_c, is called an *undirected graph*.

Undirected graph

Chain
Chaining

b) The chaining that characterizes single linkage clustering may be described by a G_c-chain. A G_c-chain is said to extend from $\mathbf{x}_1$ to $\mathbf{x}_2$ if there exist other points $\mathbf{x}_3, \mathbf{x}_4, \ldots, \mathbf{x}_i$ in the collection of objects under study, such that $\mathbf{x}_1\ G_c\ \mathbf{x}_3$ and $\mathbf{x}_3\ G_c\ \mathbf{x}_4$ and … and $\mathbf{x}_i\ G_c\ \mathbf{x}_2$. For instance, at similarity level $c = 0.786$ of the example, there exists a $G_{0.786}$-chain from pond 212 to pond 233 since there are intermediate ponds such that $212\ G_{0.786}\ 214$ and $214\ G_{0.786}\ 432$ and $432\ G_{0.786}\ 431$ and $431\ G_{0.786}\ 233$. The number of links in a G_c-chain defines the *connectedness* of a cluster (Subsection 8.12.1).

c) There only remains to delineate the clusters resulting from single linkage chaining. For that purpose, an equivalence relation R_c ("member of the same cluster") is defined as follows:

$\mathbf{x}_1\ R_c\ \mathbf{x}_2$ if and only if there exists a G_c-chain from $\mathbf{x}_1$ to $\mathbf{x}_2$ at distance level c.

In other words, $\mathbf{x}_1$ and $\mathbf{x}_2$ are assigned to the same cluster at distance level c if there exists a chain of links joining $\mathbf{x}_1$ to $\mathbf{x}_2$. Thus, at level $D = 0.786$ in the example, ponds 212 and 233 are assigned to the same cluster ($212\ R_{0.786}\ 233$) because there exists a $G_{0.786}$-chain from 212 to 233. The relationship "member of the same cluster" has the following properties: (1) it is reflexive ($\mathbf{x}_i\ R_c\ \mathbf{x}_i$) because G_c is reflexive; (2) the G_c-chains may be reversed because G_c is symmetric; as a consequence, $\mathbf{x}_1\ R_c\ \mathbf{x}_2$ implies that $\mathbf{x}_2\ R_c\ \mathbf{x}_1$; and (3) it is transitive because, by G_c-chaining, $\mathbf{x}_1\ R_c\ \mathbf{x}_2$ and $\mathbf{x}_2\ R_c\ \mathbf{x}_3$ implies that $\mathbf{x}_1\ R_c\ \mathbf{x}_3$. Each cluster thus defined is a connected subgraph, which means that the objects of a cluster are all connected in their subgraph; in the graph of all the objects, distinct clusters (subgraphs) have no links attaching them to one another.

Connected subgraph

Single linkage clustering provides an accurate picture of the relationships between pairs of objects, but its propensity to chaining is often not desirable in ecological analysis. This is because the presence of an object midway between two compact clusters, or a few intermediate objects connecting two clusters, are enough to turn them into a single cluster. Of course, clusters do not chain unless intermediates are present; so, the occurrence of chaining provides information about the data. To describe this phenomenon, Lance & Williams (1967c) wrote that this method "contracts the reference space". Picture the objects as laying in descriptor space (A-space, Fig. 7.2): the presence of a cluster increases the probability of inclusion, by chaining, of neighbouring objects into the cluster. This is as if the distances between objects were smaller in that region of the space; see also Fig. 8.24a.

Section 10.1 will show how to take advantage of the interesting properties of single linkage clustering by combining it with ordination results, while avoiding the undue influence of chaining on the clustering structure.

Minimum spanning tree

The set of edges that first connect objects to clusters or small graphs into larger graphs, in single linkage clustering, form a graph called *minimum spanning tree* (MST, Gower & Ross, 1969). For Ecological application 8.2, the first four edges represented by heavy links in the left-hand part of Fig. 8.2a, down to $D = 0.786$, form the MST.

That tree has been described a number of times in the literature and has received several names: *dendrites* (Lukaszewicz, 1951), *network* (Prim, 1957), *Prim network* (Cavalli-Sforza & Edwards, 1967), *shortest spanning tree* or *minimum-length tree* (Sneath & Sokal, 1973). MSTs are very useful when analysing clusters drawn in an ordination space (Section 10.1). If the MST is drawn on a scatter diagram of the objects, one can obtain a non-hierarchical clustering of the objects by removing the single largest or the few largest distance links. Such graphs are illustrated in Figs. 10.1 and 10.2. A MST is also used to calculate the truncation distance in the computation of spatial eigenfunctions in Chapter 14. Section 8.15 shows how to compute a MST in R.

Chain of primary connections

A related concept is the *chain of primary connections* (Legendre, 1976): this is the set of links that first connect objects to groups, or groups to one another, in any type of hierarchical clustering. For single linkage clustering, that chain is identical to the MST, but it may differ for other methods if the clustering topology they produce is different. How to compute it is described at the end of Subsection 8.5.4 for the UPGMA case.

8.3 Cophenetic matrix and ultrametric property

Any classification or partition can be fully described by a cophenetic matrix. This matrix is used for comparing different classifications of the same objects.

1 — Cophenetic matrix

Cophenetic distance

The *cophenetic distance* (or *similarity*) of two objects $\mathbf{x}_1$ and $\mathbf{x}_2$ is defined as the distance (or similarity) level at which objects $\mathbf{x}_1$ and $\mathbf{x}_2$ become members of the same cluster during the course of clustering (Jain & Dubes, 1988), as depicted by connected subgraphs or by a dendrogram (e.g. Fig. 8.2a, b). Any dendrogram can be uniquely represented by a matrix in which the distance (or similarity) for a pair of objects is their cophenetic distance (or similarity). Consider the single linkage clustering dendrogram of Fig. 8.2b. The clustering levels, read directly on the dendrogram, lead to the following distance (**D**) and similarity (**S**, where $S = 1 - D$) matrices:

D	212	214	233	431	432
212	—			(upper triangle	
214	0.400	—		symmetric to lower)	
233	0.786	0.786	—		
431	0.786	0.786	0.700	—	
432	0.786	0.786	0.700	0.500	—

S	212	214	233	431	432
212	—			(upper triangle	
214	0.600	—		symmetric to lower)	
233	0.214	0.214	—		
431	0.214	0.214	0.300	—	
432	0.214	0.214	0.300	0.500	—

Cophenetic matrix

These matrices are called *cophenetic matrices* (Sokal & Rohlf, 1962; Jain & Dubes, 1988). The ordering of objects in the cophenetic matrix is irrelevant; any order that suits the researcher is acceptable. The same applies to dendrograms; the order of the

objects may be changed at will, provided that the dendrogram is redrawn to accommodate the new ordering.

For a *partition* of the data set (as in the *K*-means method, below), the resulting groups of objects are not related through a dendrogram. A cophenetic matrix may nevertheless be computed. Consider the groups (212, 214) and (233, 431, 432) obtained by cutting the dendrogram of Fig. 8.2b at distance level $D = 0.75$, ignoring the hierarchical structure of the two clusters. The cophenetic matrices would be:

D	212	214	233	431	432
212	—		(upper triangle		
214	0	—	symmetric to lower)		
233	1	1	—		
431	1	1	0	—	
432	1	1	0	0	—

S	212	214	233	431	432
212	—		(upper triangle		
214	1	—	symmetric to lower)		
233	0	0	—		
431	0	0	1	—	
432	0	0	1	1	—

2 — Ultrametric property

If there are no *reversals* in the clustering (Fig. 8.16), a classification has the following *ultrametric property*:

$$D(\mathbf{x}_1, \mathbf{x}_2) \leq \max[D(\mathbf{x}_1, \mathbf{x}_3), D(\mathbf{x}_2, \mathbf{x}_3)] \qquad \textbf{(8.1)}$$

for every triplet of objects $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$ in the study. The cophenetic matrix is then called ultrametric. Cophenetic distances also possess the four *metric properties* of Section 7.4. The ultrametric property may be expressed in terms of similarities:

$$S(\mathbf{x}_1, \mathbf{x}_2) \geq \min[S(\mathbf{x}_1, \mathbf{x}_3), S(\mathbf{x}_2, \mathbf{x}_3)] \qquad \textbf{(8.2)}$$

As an exercise, readers can verify that the five properties apply to all doublets and triplets of distances in the cophenetic **D** matrix shown above.

8.4 The panoply of methods

Clustering algorithms have been developed using a wide range of conceptual models and for studying a variety of problems. Sneath & Sokal (1973) proposed a classification of clustering procedures. Its main dichotomies are briefly described.

1 — Sequential versus simultaneous algorithms

Most clustering algorithms are sequential in the sense that they proceed by applying a recurrent sequence of operations to the objects. The agglomerative single linkage

clustering of Section 8.2 is an example of a sequential method: the search for the equivalence relation R_c is repeated at all distance levels in the association matrix, up to the point where all objects are in the same cluster. In *simultaneous* algorithms, which are less frequent, the solution is obtained in a single step. Ordination techniques (Chapter 9), which may be used for delineating clusters, are of the latter type. This is also the case of the direct complete linkage clustering method presented in Subsection 8.9.1. The *K*-means (Section 8.8) and other non-hierarchical partitioning methods may be computed using sequential algorithms, although these methods are neither agglomerative nor divisive (next paragraph).

2 — Agglomeration versus division

Among the sequential algorithms, *agglomerative* procedures begin with the discontinuous partition of all objects, i.e. the objects are considered as being separate from one another. They are successively grouped into larger and larger clusters until a single, all-encompassing cluster is obtained. If the continuous partition of all objects is used instead as the starting point of the procedure (i.e. a single group containing all objects), *divisive* algorithms subdivide the group into sub-clusters, and so on until the discontinuous partition is reached. In either case, it is left to users to decide which of the intermediate partitions is to be retained, given the problem under study. Agglomerative algorithms are the most developed for two reasons. First, they are easier to program. Second, in clustering by division, the erroneous allocation of an object to a cluster at the beginning of the procedure cannot be corrected afterwards (Gower, 1967) unless a complex procedure is embedded in the algorithm to do so.

3 — Monothetic versus polythetic methods

Divisive clustering methods may be monothetic or polythetic. *Monothetic* models use a single descriptor at each step as the basis for partitioning, whereas *polythetic* models use several descriptors which, in most cases, are combined into an association matrix (Chapter 7) prior to clustering. Divisive monothetic methods proceed by choosing, for each partitioning level, the descriptor considered to be the best for that level; objects are then partitioned following the state to which they belong with respect to that descriptor. For example, the most appropriate descriptor at each partitioning level could be the one that best represents the information contained in all other descriptors, after measuring the reciprocal information between descriptors (Subsection 8.7.1). When a single partition of the objects is sought, monothetic methods produce the clustering in a single step.

4 — Hierarchical versus non-hierarchical methods

In *hierarchical* methods, the members of inferior-ranking clusters become members of larger, higher-ranking clusters. Most of the time, hierarchical methods produce non-overlapping clusters, but this is not a necessity according to the definition of "hierarchy" in the dictionary or the usage recognized by Sneath & Sokal (1973).

Single linkage clustering of Section 8.2 and the methods of Sections 8.5 and 8.7 are hierarchical.

Non-hierarchical methods are very useful in ecology. They produce a single partition that optimizes within-group homogeneity, instead of a hierarchical series of partitions optimizing the hierarchical attribution of objects to clusters. Lance & Williams (1967d) restrict the term "clustering" to the non-hierarchical methods and call the hierarchical methods "classification". Non-hierarchical methods include *K*-means partitioning, the ordination techniques (Chapter 9) used as clustering methods, the creation of clusters by removing edges from a graph (which may be a minimum spanning tree), the methods of matrix seriation of Section 8.10, and the algorithm described in Subsection 8.9.1 for the clustering of species into biological associations. These methods should be used in cases where the aim is to obtain a direct representation of the relationships among objects instead of a summary of their hierarchy. Hierarchical methods are easier to compute and more often available in statistical packages than non-hierarchical procedures.

Most hierarchical methods use a resemblance matrix as their starting point. This prevents their use with very large data sets because the resemblance matrix, with its $n(n-1)/2$ values, may become extremely large. Algorithms have been developed for hierarchical agglomeration of very large numbers of objects after computing only a small fraction of the distances (e.g. Jambu & Lebeaux, 1983; Rohlf, 1978, 1982a).

5 — Constrained clustering methods

In constrained clustering, external information about the sampling design is used by the clustering algorithm, in addition to the distance relationships among objects. Two forms of constrained clustering are described in this book: time-constrained (Section 12.6) and space-constrained clustering (Subsection 13.3.2).

6 — Probabilistic versus non-probabilistic methods

Probabilistic methods include a clustering model by Clifford & Goodall (1967), designed to be used in conjunction with Goodall's probabilistic index (S_{23}, Chapter 7), in which clusters are formed in such a way that the within-group association matrices have a given probability of being homogeneous. That method is described in the previous edition of this book (Legendre & Legendre, 1998, Subsection 8.9.2). This category also includes the parametric and nonparametric methods for estimating density functions in multivariate space.

Sneath & Sokal (1973) describe other dichotomies for clustering methods, which are of lesser interest to ecologists. These are: global or local criteria, direct or iterative solutions, equal or unequal weights, and adaptive or non-adaptive clustering.

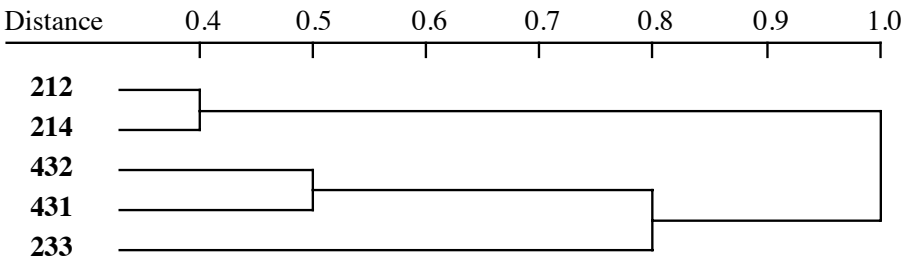


Figure 8.3 Complete linkage clustering of the ponds of Ecological application 8.2.

8.5 Hierarchical agglomerative clustering

Most methods of hierarchical agglomeration can be computed as special cases of a general model which is discussed in Subsection 8.5.9.

1 — Single linkage agglomerative clustering

In single linkage agglomeration (Section 8.2), two clusters fuse when the two objects closest to each other (one in each cluster) reach the distance level of the considered partition (Fig. 8.2). As a consequence of chaining, results of single linkage clustering are sensitive to noise in the data (Milligan, 1996), because noise changes the distance values and may thus modify the order in which objects cluster. The origin of single linkage clustering is found in a collective work by mathematicians Florek, Lukaszewicz, Perkal, Steinhaus, and Zubrzycki, published by Lukaszewicz in 1951.

2 — Complete linkage agglomerative clustering

Opposite to the single linkage approach is *complete linkage agglomeration*, also called *furthest neighbour sorting*. In this method, first proposed by Sørensen (1948), the fusion of two clusters depends on the most distant pair of objects instead of the closest. Thus, an object joins a cluster only when it is linked (relationship G_c, Section 8.2) to all the objects already members of that cluster. In the same way, two clusters can fuse only when all objects of the first are linked to all objects of the second, and vice versa.

Complete linkage rule

Coming back to the ponds of Ecological application 8.2, the steps of complete linkage clustering (Fig. 8.3) can be followed on the subgraphs shown in Fig. 8.2a. Examine the connected subgraphs and locate the D levels where completely connected groups of 2, 3, 4, and 5 objects are found. The pair (212, 214) is formed at $D = 0.4$ and the pair (431, 432) at $D = 0.5$. The next clustering step must wait until $D = 0.8$ since it is only then that pond 233 is finally linked (relationship G_c) to both ponds 431 and

432. Although there is a group of four completely linked ponds at $D = 0.937$, these ponds do not form a cluster in the agglomerative framework because pond 214 is already linked to pond 212, hence the two clusters (212, 214) and (233, 431, 432) cannot fuse at that level. It is only at $D = 1$ that ponds 212 and 214 are linked to all the ponds of cluster (233, 431, 432) and the five ponds form a single cluster.

In the complete linkage strategy, as a cluster grows, it becomes more and more difficult for new objects to join to it because the new objects should bear links with all the objects already in the cluster before being incorporated. For that reason, the growth of a cluster seems to move it away from the other objects or clusters in the analysis. According to Lance & Williams (1967c), this is equivalent to dilating the reference space in the neighbourhood of that cluster; see also Fig. 8.24c and related text. This effect is opposite to what was found in single linkage clustering, which contracted the reference space. In reference space A (Fig. 7.2), complete linkage produces maximally linked and rather spherical clusters, whereas single linkage may produce elongated clusters with loose chaining. Complete linkage clustering is often desirable in ecology, when one wishes to delineate clusters with clear discontinuities.

The intermediate (next subsection) and complete linkage clustering models have one drawback when compared to single linkage. In all cases where two incompatible candidates present themselves at the same time to be included in a cluster, algorithms use a preestablished and often arbitrary rule, called a "right-hand rule", to choose one and exclude the other. This problem does not exist in single linkage. An example is when two objects or two clusters could be included in a third cluster, while these two objects or clusters have not completed the linkage with each other. For this problem, Sørensen (1948) recommended the following: (1) choose the fusion leading to the largest cluster; (2) if equality persists, choose the fusion that most reduces the number of clusters; (3) as a last criterion, choose the fusion that minimizes the average distance within the cluster.

3 — Intermediate linkage clustering

Between the chaining of single linkage and the extreme space dilation of complete linkage, the most interesting solution in ecology may be a type of linkage clustering that approximately conserves the metric properties of reference space A; see also Fig. 8.24b. If the interest only lies in the clusters shown in the dendrogram, and not in the actual links between clusters shown by the subgraphs, the average clustering methods of Subsections 8.5.4 to 8.5.7 below could be useful since they also conserve the metric properties of the reference space.

Connectedness

In intermediate linkage clustering, the fusion criterion of an object or a cluster with another cluster is considered satisfied when a given proportion of the total possible number of links is reached. For example, if the criterion of connectedness (*Co*) is 0.5, two clusters are only required to share 50% of the possible links in order to fuse; in other words, the fusion is authorized when $£/n_1n_2 \geq Co$ where £ is the actual number of *between-group* links at sorting level *L,* while n_1 and n_2 are the numbers of objects in

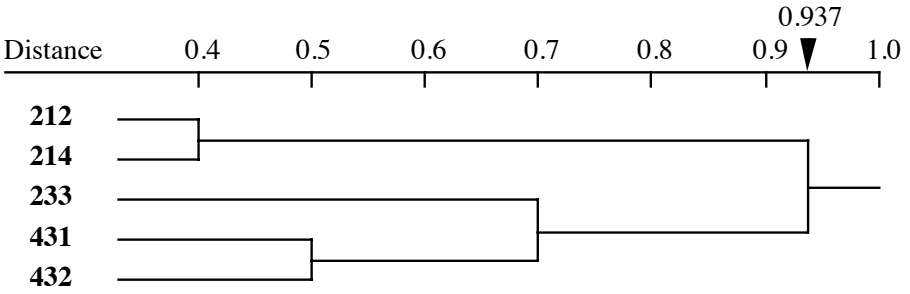


Figure 8.4 Intermediate linkage clustering, using the proportional link linkage criterion (Co = 50%), for the ponds of Ecological application 8.2.

Proportional link linkage

the two clusters, respectively. This criterion has been called *proportional link linkage* by Sneath (1966). Figure 8.4 gives the results of proportional link linkage clustering with Co = 50% for the pond example.

Sneath (1966) described three other ways of defining intermediate linkage clustering criteria: (1) by *integer link linkage*, which specifies the number of links required for the fusion of two groups (fusion when £ is larger than or equal to a fixed integer, *or else* when £ = $n_1 n_2$); (2) by their *absolute resemblance*, based on the sum of similarity links between the members of two clusters (the sum of the realized between-group similarities, $\sum S_{12}$, must reach a given threshold before fusion occurs); or (3) by their *relative resemblance*, where the sum of similarity links between the two clusters, $\sum S_{12}$, is divided by the number of between-group similarities, $n_1 n_2$ (fusion occurs at level L when the ratio $\sum S_{12}/n_1 n_2$ is greater than cL, where c is an arbitrary constant). When c equals 1, the method is called *average linkage clustering*. Similarities, not distances, must be used for criteria 2 and 3. These strategies are not combinatorial in the sense of Subsection 8.5.9.

4 — Unweighted arithmetic average clustering (UPGMA)

Average clustering

There are four methods of *average clustering* that conserve the metric properties of reference space A. These four methods were called "average linkage clustering" by Sneath & Sokal (1973), although they do not tally the links between clusters. As a consequence they are not object-linkage methods in the sense of the previous three subsections. They rely instead on the calculation of average distances among objects or the centroids of clusters. The four methods have nothing to do with Sneath's (1966) "average linkage clustering" described in the previous paragraph, so that we prefer calling them "average clustering". These methods (Table 8.2) result from the combinations of two dichotomies: (1) arithmetic average *versus* centroid clustering and (2) weighting *versus* non-weighting.

Table 8.2 Average clustering methods discussed in Subsections 8.5.4 to 8.5.7.

	Arithmetic average	Centroid clustering
Equal weights	4. Unweighted arithmetic average clustering (UPGMA)	6. Unweighted centroid clustering (UPGMC)
Unequal weights	5. Weighted arithmetic average clustering (WPGMA)	7. Weighted centroid clustering (WPGMC)

The first method in Table 8.2 is the *unweighted arithmetic average clustering* (Rohlf, 1963), also called "UPGMA" ("Unweighted Pair-Group Method using Arithmetic averages") by Sneath & Sokal (1973) or "group-average sorting" by Lance & Williams (1966a and 1967c). It is also called "average linkage" by SAS, SYSTAT and some other statistical packages, thus adding to the confusion pointed out in the previous paragraph. This is method = "average" in function ***hclust()*** of R. The lowest distance (or highest similarity) identifies the next cluster to be formed. Following this event, the method computes the arithmetic average of the distances between a candidate object and each of the cluster members or, in the case of a previously formed cluster, between all members of the two clusters. All objects receive equal weights in the computation. The distance matrix is updated and reduced in size at each clustering step. Clustering proceeds by agglomeration as the distance criterion increases, just as it does in single linkage clustering.

For the ponds of Section 8.2, UPGMA clustering proceeds as shown in Table 8.3 and Fig. 8.5. At step 1, the lowest distance value in the matrix is $D(212, 214) = 0.400$; hence the two objects fuse at level 0.400. As a consequence of this fusion, the distance values of these two objects with each of the remaining objects in the study must be averaged (values in the inner boxes in the table, step 1); this results in a reduction of the size of the distance matrix. Considering the reduced matrix (step 2), the smallest distance value is $D = 0.500$; it indicates that objects 431 and 432 fuse at level 0.500. The fused distance values are obtained by averaging the boxed values in the step 2 panel; this produces a new reduced distance matrix for the next step. In step 3, the lowest distance is 0.750; it leads to the fusion of the already-formed group (431, 432) with object 233 at level 0.750. In the example, this last fusion is the difficult point to understand. Before averaging the values, each one must be multiplied by the number of objects in the corresponding group. There is one object in group (233) and two in group (431, 432), so that the fused distance value is calculated as $[(0.9645 \times 1) + (0.93075 \times 2)]/3 = 0.942$. This is equivalent to averaging the six boxed distances in the top panel (larger box) with equal weights; the result would also be 0.942. So, this method is "unweighted" in the sense that it gives equal weights to the original

Table 8.3 Unweighted arithmetic average clustering (UPGMA) of the pond data. At each step, the lowest distance value is identified (italicized boldface value) and the two corresponding objects or groups are fused by averaging their distances as described in the text (boxes).

Objects	212	214	233	431	432	
212	—					**Step 1**
214	***0.400***	—				
233	1.000	0.929	—			
431	1.000	0.937	0.700	—		
432	1.000	0.786	0.800	0.500	—	
212-214		—				**Step 2**
233		0.9645	—			
431		0.9685	0.700	—		
432		0.8930	0.800	***0.500***	—	
212-214		—				**Step 3**
233		0.9645	—			
431-432		0.93075	***0.750***	—		
212-214		—				**Step 4**
233-431-432		***0.942***	—			

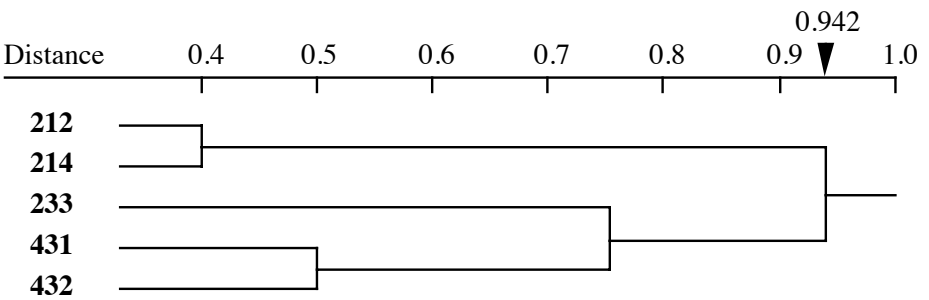


Figure 8.5 Unweighted arithmetic average clustering (UPGMA) of the ponds from Ecological application 8.2. This type of clustering only produces a dendrogram. It cannot be represented by connected subgraphs since it is not a linkage clustering as found in Figs. 8.2 to 8.4.

distances. To achieve this at step 3, one has to use weights that are equal to the number of objects in the groups. At step 4, there is a single remaining distance value; it is used to perform the last fusion at level 0.942. In the dendrogram, fusions are drawn at the identified distance levels.

Because it gives equal weights to the original distances, the UPGMA method assumes that the objects in each group form a representative sample of the corresponding larger groups of objects in the reference population under study. For that reason, UPGMA clustering should only be used in connection with simple random or systematic sampling designs if the results are to be extrapolated to a larger reference population.

Unlike the linkage clustering methods, information about the relationships between pairs of objects is lost in methods based on progressive reduction of the distance matrix, since only the relationships among groups are considered. This information can be extracted from the original distance matrix by making a list containing, for each fusion level, the lowest distance found between objects of the two groups. For the pond example, the *chain of primary connections* corresponding to the dendrogram would be made of the following links: (212, 214) for the first fusion level, (431, 432) for the second level, (233, 431) for the third level, and (214, 432) for the last level (Table 8.3). The topology obtained by UPGMA clustering may differ from that of single linkage. If this had been the case here, the chain of primary connections would have been different from that of single linkage clustering.

5 — Weighted arithmetic average clustering (WPGMA)

It often occurs in ecology that groups of objects, representing different regions of a territory, are of unequal sizes. Eliminating objects to equalize the clusters would mean discarding valuable information. However, the presence of a large group of objects, which are more similar *a priori* because of their common origin, may distort the UPGMA results when a fusion occurs with a smaller group of objects. Sokal & Michener (1958) proposed a solution to this problem, called *weighted arithmetic average clustering* ("WPGMA" in Sneath & Sokal, 1973: "Weighted Pair-Group Method using Arithmetic averages"; method = "mcquitty" in function ***hclust()*** of R). This solution consists in giving equal weights, when computing fusion distances, to the two *branches* of the dendrogram that are about to fuse. This is equivalent, when computing a fusion distance, to giving different weights to the original distances, i.e. down-weighting the distances of the largest group. Hence the name of the method.

Table 8.4 and Fig. 8.6 describe the WPGMA clustering sequence for the pond data. In this example, the only difference with UPGMA is the last fusion value. It is computed here by averaging the two distances from the previous step: (0.9645 + 0.93075)/2 = 0.947625. Weighted arithmetic average clustering increases the separation of the two main clusters, compared to UPGMA. This gives sharper contrast to the classification.

Table 8.4 Weighted arithmetic average clustering (WPGMA) of the pond data. At each step, the lowest distance value is identified (italicized boldface value) and the two corresponding objects or groups are fused by averaging their distances (boxes).

Objects	212	214	233	431	432	
212	—					**Step 1**
214	***0.400***	—				
233	1.000	0.929	—			
431	1.000	0.937	0.700	—		
432	1.000	0.786	0.800	0.500	—	
212-214		—				**Step 2**
233		0.9645	—			
431		0.9685	0.700	—		
432		0.8930	0.800	***0.500***	—	
212-214		—				**Step 3**
233		0.9645	—			
431-432		0.93075	***0.750***	—		
212-214		—				**Step 4**
233-431-432		***0.9476***	—			

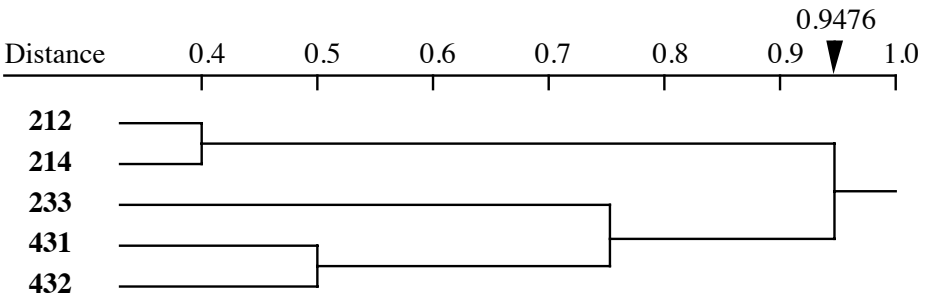


Figure 8.6 Weighted arithmetic average clustering (WPGMA) of the ponds from Ecological application 8.2. This type of clustering only produces a dendrogram. It cannot be represented by connected subgraphs since it is not a linkage clustering as found in Figs. 8.2 to 8.4.

6 — Unweighted centroid clustering (UPGMC)

Centroid

The *centroid* of a cluster of objects can be imagined as the type-object of the cluster, whether that object actually exists or is only a mathematical construct. In A-space (Fig. 7.2), the coordinates of the centroid of a cluster are computed by averaging the coordinates of the objects in the cluster.

Unweighted centroid clustering (Lance & Williams, 1967c; "UPGMC" in Sneath & Sokal, 1973: "Unweighted Pair-Group Centroid Method") is based on a simple geometric approach. This is method = "centroid" in function ***hclust()*** of R. Along a decreasing scale of distances, UPGMC proceeds to the fusion of objects or clusters presenting the lowest distance, as in the previous methods. At each step, the members of a cluster are replaced by their common centroid (i.e. "mean point"). The centroid is considered to represent a new object for the remainder of the clustering procedure; in the next step, one looks again for the pair of objects with the lowest distances, on which the fusion procedure is repeated.

Gower (1967) proposed the following formula for centroid clustering, where the distance of the centroid (**hi**) of objects or clusters **h** and **i** to a third object or cluster **g** is computed from the distances $D(\mathbf{h}, \mathbf{g})$, $D(\mathbf{i}, \mathbf{g})$, and $D(\mathbf{h}, \mathbf{i})$:

$$D(\mathbf{hi}, \mathbf{g}) = \frac{w_h}{w_h + w_i} D(\mathbf{h}, \mathbf{g}) + \frac{w_i}{w_h + w_i} D(\mathbf{i}, \mathbf{g}) - \frac{w_h w_i}{(w_h + w_i)^2} D(\mathbf{h}, \mathbf{i}) \qquad \textbf{(8.3)}$$

were the w's are weights given to the clusters. To simplify the symbolism, letters **g**, **h**, and **i** designate three objects considered in the course of clustering; they may also represent centroids of clusters obtained during previous clustering steps.

Gower's formula insures that the centroid **hi** of objects (or clusters) **h** and **i** is geometrically located on the line between **h** and **i**. In classical centroid clustering, the numbers of objects n_h and n_i in clusters **h** and **i** are taken as values for the weights w_h and w_i; these weights are 1 at the start of the clustering because there is then a single object per cluster. If initial weights are attached to individual objects, they may be used instead of 1's in eq. 8.3.

Centroid clustering may lead to reversals (Section 8.6). Some authors feel uncomfortable about reversals since they violate the ultrametric property (eq. 8.1); such violations make dendrograms more difficult to draw. A reversal is found with the pond example (Table 8.5, Fig. 8.7): the fusion distance found at step 4 is lower than that of step 3. The last fusion distance (step 4) is calculated as follows:

$$D[(233, 431\text{-}432), (212\text{-}214)] = \frac{1}{3} \times 0.8645 + \frac{2}{3} \times 0.70575 - \frac{2}{3^2} \times 0.625 = 0.61978$$

As indicated above, UPGMC clustering is geometrically interpreted as the fusion of objects into cluster centroids. Figure 8.8 presents the four clustering steps depicted

Table 8.5 Unweighted centroid clustering (UPGMC) of the pond data. At each step, the lowest distance value is identified (italicized boldface value) and the two corresponding objects or groups are fused using eq. 8.3.

Objects	212	214	233	431	432
212	—				**Step 1**
214	***0.400***	—			
233	1.000	0.929	—		
431	1.000	0.937	0.700	—	
432	1.000	0.786	0.800	0.500	—
212-214		—			**Step 2**
233		0.8645	—		
431		0.8685	0.700	—	
432		0.7930	0.800	***0.500***	—
212-214		—			**Step 3**
233		0.8645	—		
431-432		0.70575	***0.625***	—	
212-214		—			**Step 4**
233-431-432		***0.6198***	—		

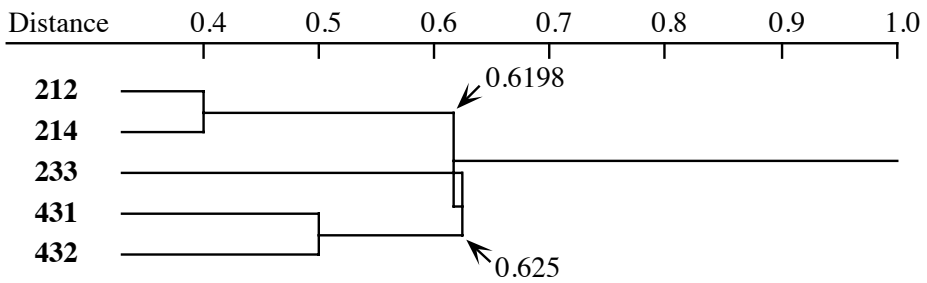


Figure 8.7 Unweighted centroid clustering (UPGMC) of the ponds from Ecological application 8.2. This type of clustering only produces a dendrogram. The reversal in the structure of the dendrogram is explained in Section 8.6.

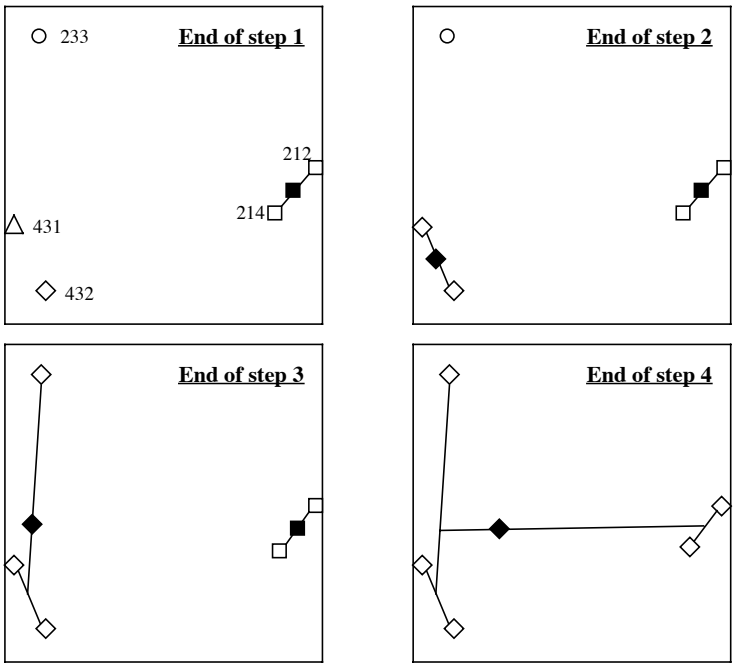


Figure 8.8 The four UPGMC clustering steps of Fig. 8.7 are drawn in A-space. Objects are represented by open symbols and centroids by dark symbols; object identifiers are shown in the first panel only. Separate clusters are represented by different symbols. The first two principal coordinates, represented here, account for 87% of the variation of the objects in the full A-space.

by the dendrogram, drawn in an A-space (Fig. 7.2) reduced to two dimensions through principal coordinate analysis (Section 9.3) to facilitate representation. At the end of each step, a new cluster is formed and its centroid is represented at the *centre of mass* of the cluster members; examine especially steps 3 and 4.

Unweighted centroid clustering may be used with any measure of distance, but Gower's formula (eq. 8.3) only retains its geometric properties for distances that are Euclidean (Table 7.2). Note also that in this clustering procedure, the links between clusters do not depend upon identifiable pairs of objects; this was also the case with clustering methods 4 and 5 above. Thus, if the chain of primary connections is needed, its links be identified by the method described at the end of Subsection 8.5.4.

The assumptions of this model with respect to representativeness of the observations are the same as in UPGMA since equal weights are given to all objects during clustering. So, UPGMC should only be used in connection with simple random or systematic sampling designs if the results are to be extrapolated to a larger reference population. When the branching pattern of the dendrogram displays asymmetry (many

more objects in one branch than in the other), this can be attributed to the structure of the reference population if the sampling design was random.

7 — Weighted centroid clustering (WPGMC)

Weighted centroid clustering was proposed by Gower (1967). This is method = "median" in function ***hclust()*** of R. It plays the same role with respect to UPGMC as WPGMA (method 5) plays with respect to UPGMA (method 4). When many observations of a given type have been included in the set to be clustered, next to other types that were not as well-sampled (sampling design other than simple random or systematic), the positions of the centroids may be biased towards the over-represented types, which in turn could distort the clustering. In *weighted centroid clustering*, which Sneath & Sokal (1973) called "WPGMC" ("Weighted Pair-Group Centroid Method"), this problem is corrected by giving equal weights to two clusters on the verge of fusing, independently of the number of objects in each cluster. To achieve this, eq. 8.3 is replaced by the following formula (Gower, 1967):

$$D(\mathbf{hi}, \mathbf{g}) = \frac{1}{2}[D(\mathbf{h}, \mathbf{g}) + D(\mathbf{i}, \mathbf{g})] - \frac{1}{4}D(\mathbf{h}, \mathbf{i}) \qquad \textbf{(8.4)}$$

The five ponds of Ecological application 8.2 are clustered as described in Table 8.6 and Fig. 8.9. The last fusion distance (step 4), for example, is calculated as follows:

$$D[(233, 431\text{-}432), (212\text{-}214)] = \frac{1}{2}[0.8645 + 0.70575] - \frac{1}{4} \times 0.625 = 0.62888$$

This value is the level at which the last fusion takes place. Note that no reversal appears in this result, although WPGMC can produce reversals like UPGMC clustering.

As indicated above, WPGMC clustering is geometrically interpreted as the fusion of objects into cluster centroids. Figure 8.10 presents the four clustering steps depicted by the dendrogram, in A-space (Fig. 7.2) reduced to two dimensions through principal coordinate analysis (Section 9.3) to facilitate representation. At the end of each step, a new cluster is formed and its centroid is represented at the *geometric centre* of the last line drawn; examine especially steps 3 and 4 and compare to Fig. 8.8.

8 — Ward's minimum variance method

Ward's (1963) minimum variance method is related to the centroid methods (Subsections 8.5.6 and 8.5.7 above) in that it also leads to a geometric representation in which cluster centroids play a key role. To form clusters, the method minimizes an *objective function* which is, in this case, the same "squared error" criterion as that used in multivariate analysis of variance.

Objective function

Table 8.6 Weighted centroid clustering (WPGMC) of the pond data. At each step, the lowest distance value is identified (italicized boldface value) and the two corresponding objects or groups are fused using eq. 8.4.

Objects	212	214	233	431	432
212	—				**Step 1**
214	***0.400***	—			
233	1.000	0.929	—		
431	1.000	0.937	0.700	—	
432	1.000	0.786	0.800	0.500	—
212-214		—			**Step 2**
233		0.8645	—		
431		0.8685	0.700	—	
432		0.7930	0.800	***0.500***	—
212-214		—			**Step 3**
233		0.8645	—		
431-432		0.70575	***0.625***	—	
212-214		—			**Step 4**
233-431-432		***0.6289***	—		

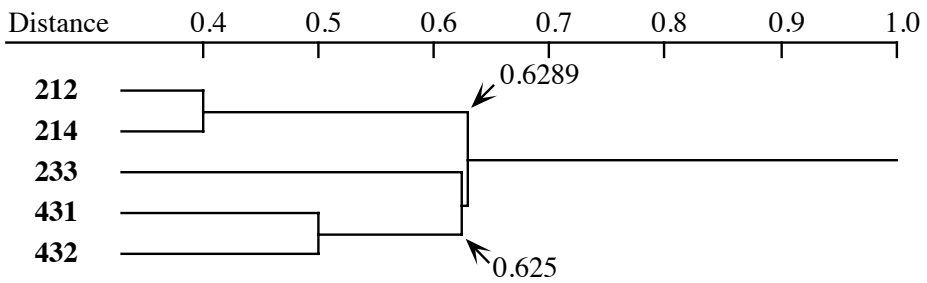


Figure 8.9 Weighted centroid clustering (WPGMC) of the ponds from Ecological application 8.2. This type of clustering only produces a dendrogram.

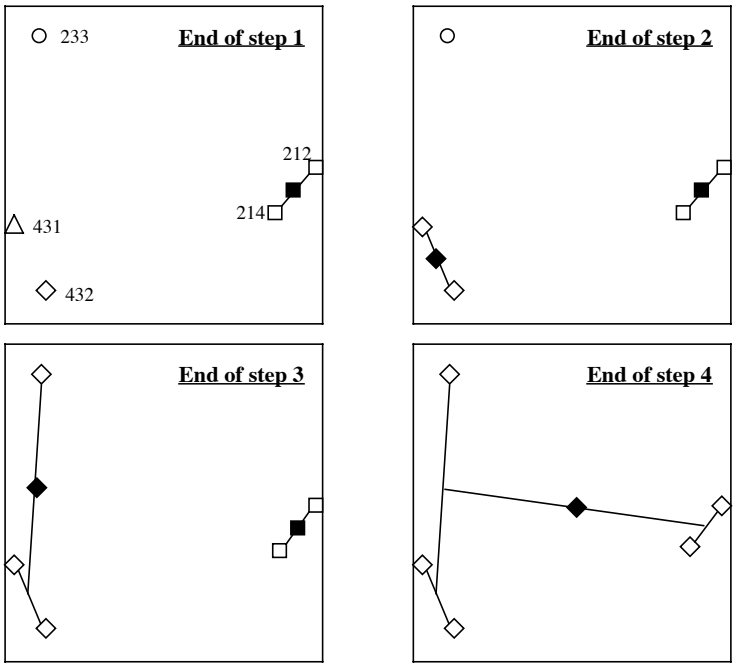


Figure 8.10 The four WPGMC clustering steps of Fig. 8.9 are drawn in A-space. Objects are represented by open symbols and centroids by dark symbols; object identifiers are shown in the first panel only. Distinct clusters are represented by different symbols. The first two principal coordinates, represented here, account for 87% of the variation of the objects in the full A-space.

At the beginning of the procedure, each objects is in a cluster of its own, so that the distance of an object to its cluster's centroid is 0; hence, the sum of all these distances is also 0. As clusters form, the centroids move away from actual object coordinates and the sums of the squared distances from the objects to the centroids increase. At each clustering step, Ward's method finds the pair of objects or clusters whose fusion increases as little as possible the sum, over all groups formed so far, of the *squared distances* between objects and cluster centroids; that sum is the total within-group sum-of-squares. The distance of object $\mathbf{x}_i$ to centroid $\mathbf{m}$ of its cluster is computed using the squared Euclidean distance formula (eq. 7.33) over the various descriptors $\mathbf{y}_j$ ($j = 1 \ldots p$):

$$\sum_{j=1}^{p} [y_{ij} - m_j]^2$$

The centroid **m** of a cluster was defined at the beginning of Subsection 8.5.6. The sum of the squared distances of all objects in cluster k to their common centroid, which is called "error" in ANOVA (hence the symbol e_k^2), is the sum of the squared Euclidean distances between the members of the cluster and its centroid:

Squared error

Error in cluster k: $$e_k^2 = \sum_{i=1}^{n_k} \sum_{j=1}^{p} [y_{ij}^{(k)} - m_j^{(k)}]^2 \quad \textbf{(8.5)}$$

where $y_{ij}^{(k)}$ is the value of descriptor $\mathbf{y}_j$ for an object i member of group (k) and $m_j^{(k)}$ is the mean value of descriptor j over all members of group k. e_k^2 is used as a measure of the tightness of a cluster. If all data points in a cluster have the same coordinates in multidimensional space, or there is a single point in a cluster, the within-cluster variation is 0. Alternatively, the within-cluster sums of squared errors e_k^2 can be computed as the mean of the squared distances among cluster members:

Error in cluster k: $$e_k^2 = \left[\sum_{h,i=1}^{n_k} D_{hi}^2 \right] / n_k \quad \textbf{(8.6)}$$

where the D_{hi}^2 are the squared distances among objects in cluster k (Table 8.7) and n_k is the number of objects in that cluster. Equations 8.5 and 8.6 have already been shown in Box 6.1 (eqs. 6.55 and 6.56); they both allow the calculation of the squared error statistic. The equivalence of these two equations is stated in a theorem whose demonstration is found in Kendall & Stuart (1963, parag. 2.22) for the univariate case and in Legendre & Fortin (2010, Appendix 1) for the multivariate case. Numerical examples illustrating the calculation of eqs. 8.5 and 8.6 are given at the end of Section 8.8 on K-means partitioning.

The sum of squared errors E_K^2, over all K clusters corresponding to a given partition, is the criterion to be minimized at each clustering step:

Sum of squared errors

Total error, K clusters: $$E_K^2 = \sum_{k=1}^{K} e_k^2 \quad \textbf{(8.7)}$$

At each clustering step, two objects or clusters **h** and **i** are merged into a new cluster **hi**, as in previous sections. Since changes occurred only in groups **h**, **i**, and **hi**, the change in the overall sum of squared errors, $\Delta E_{\mathbf{hi}}^2$, can be computed from the changes that occurred in these groups only:

Change in total error: $$\Delta E_{\mathbf{hi}}^2 = e_{\mathbf{hi}}^2 - e_{\mathbf{h}}^2 - e_{\mathbf{i}}^2 \quad \textbf{(8.8)}$$

Table 8.7 Ward's minimum variance clustering of the pond data. Step 1 of the table contains *squared distances* computed as D^2 from the distance values in the upper panels of Tables 8.3 to 8.6. At each step, the lowest squared distance is identified (italicized boldface value) and the two corresponding objects or groups are fused using eq. 8.10.

Objects	212	214	233	431	432	
212	—					**Step 1**
214	***0.16000***	—				
233	1.00000	0.86304	—			
431	1.00000	0.87797	0.49000	—		
432	1.00000	0.61780	0.64000	0.25000	—	
212-214		—				**Step 2**
233		1.18869	—			
431		1.19865	0.49000	—		
432		1.02520	0.64000	***0.25000***	—	
212-214		—				**Step 3**
233		1.18869	—			
431-432		1.54288	***0.67000***	—		
212-214		—				**Step 4**
233-431-432		***1.6795***	—			

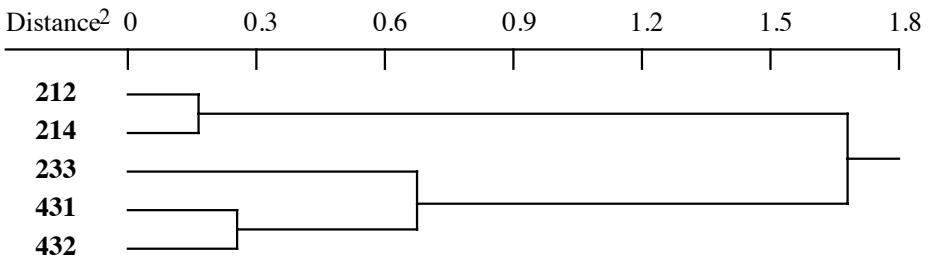


Figure 8.11 Ward's minimum variance clustering of the ponds from Ecological application 8.2. The scale of this dendrogram is here the squared distances computed in Table 8.7.

Table 8.8 Clustering steps in Ward's minimum variance clustering for the pond data. The objects are renamed 1 to 5 for shortness. K is the number of clusters, represented by underscored groups of objects. The total sum of squares (SS_{Total}) of the 5 objects is 1.37976 (eq. 8.6). SS_{Within} is also computed using eq. 8.6; $SS_{Among} = SS_{Total} - SS_{Within}$. Between clustering levels, $\Delta E^2_{\mathbf{hi}}$ is computed using eq. 8.8 or by difference between the successive values of SS_{Within} or SS_{Among}. D^2_{min} was computed in Table 8.7; D_{min} is the square root of D^2_{min}.

K	Objects	SS_{Within}	SS_{Among}	$\Delta E^2_{\mathbf{hi}}$	D^2_{min}	D_{min}
5	1 2 3 4 5	0	1.37976			
				0.08000	0.16000	0.40000
4	1 2 3 4 5	0.08000	1.29976			
				0.12500	0.25000	0.50000
3	1 2 3 4 5	0.20500	1.17476			
				0.33500	0.67000	0.81854
2	1 2 3 4 5	0.54000	0.83976			
				0.83976	1.67952	1.29596
1	1 2 3 4 5	1.37976	0			

It can be shown that this change depends only on the distance between the centroids of clusters **h** and **i** and on their numbers of objects n_h and n_i:

Change in total error: $$\Delta E^2_{\mathbf{hi}} = \frac{n_h n_i}{n_h + n_i} \sum_{j=1}^{p} \left[m_j^{(\mathbf{h})} - m_j^{(\mathbf{i})} \right]^2 \quad \textbf{(8.9)}$$

So, one way of identifying the next fusion would be to compute the $\Delta E^2_{\mathbf{hi}}$ statistic for all possible pairs and select the pair that generates the smallest value for the next fusion. An easier way is to use the following updating formula to compute the fusion distances between the new cluster **hi** and all other objects or clusters **g** in the agglomeration table (Table 8.7):

$$D^2(\mathbf{hi}, \mathbf{g}) = \frac{n_h + n_g}{n_h + n_i + n_g} D^2(\mathbf{h}, \mathbf{g}) + \frac{n_i + n_g}{n_h + n_i + n_g} D^2(\mathbf{i}, \mathbf{g}) - \frac{n_g}{n_h + n_i + n_g} D^2(\mathbf{h}, \mathbf{i}) \quad \textbf{(8.10)}$$

Wishart (1969) and Kaufman & Rousseeuw (1990) demonstrated mathematically that the smallest distance computed using this updating formula corresponds to the fusions that obeys Ward's (1963) criterion at each clustering step. Note that *squared distances* are used instead of distances in eq. 8.10 and in Table 8.7. This algorithm is called Ward.D2. Table 8.8 shows the clustering steps for the example data.

An alternative formula found in some manuals, e.g. Jain & Dubes (1988), uses distances D instead of D^2 in eq. 8.10. This formula is implemented in some programs and functions; it will be called the Ward.D algorithm*. One can show that the resulting updating formula produces cluster fusions that do not necessarily minimize the change in total error (eq. 8.8), so the clustering does not follow Ward's rule.

Ward clustering is a hierarchical agglomerative method; it proceeds sequentially by binary group fusions. Each fusion, going from (k+1) to k groups, satisfies Ward's criterion. This hierarchical method does not guarantee, however, that globally, all partitions into k = {(n–1), (n–2), ..., 4, 3, 2} groups satisfy that criterion. One should use K-means partitioning (Section 8.8) to obtain a partition into a specified number of groups (K) that minimizes the sum of residual sums-of-squares.

Dendrograms for Ward's clustering may be represented along a variety of scales although these dendrograms all represent the same clustering topology.

• In Fig. 8.11, the dendrogram is drawn using the scale of *squared distances* computed in Table 8.7.

• One can compute the *square roots of the fusion distances* of Table 8.7 and draw the dendrogram accordingly. This solution, illustrated in Fig. 8.12a, is often used in computer programs and functions, including ***agnes()*** of CLUSTER in R; it removes the distortions created by squaring the distances. It is especially suitable when one wants to compare the fusion distances of Ward's clustering to the original distances, either graphically (Shepard-like diagrams, Fig. 8.24) or numerically (cophenetic correlations, Subsection 8.12.2).

• The sum of squared errors E_K^2 (eq. 8.7) is used in some computer programs as the dendrogram scale. This statistic is also called the *total error sum of squares* (TESS) by Everitt (1980) and other authors. This solution is illustrated in Fig. 8.12b.

TESS

• The SAS package recommends two scales for Ward's clustering. The first one is the proportion of variance (R^2) accounted for by the clusters at any given partition level. It is computed as the total sum of squares (i.e. the sum of squared distances from the centroid of all objects) minus the within-cluster squared errors E_K^2 of eq. 8.7 for the given partition, divided by the total sum of squares. R^2 decreases as clusters grow. When all the objects are lumped in a single cluster, the resulting one-cluster partition does not explain any of the objects' variation so that $R^2 = 0$. The second scale recommended by SAS is called the *semipartial* R^2. It is computed as the between-

* In R, function ***hclust()*** of package STATS with method = "ward" implements the Ward.D algorithm (at least up to version 2.12.1), whereas function ***agnes()*** of package CLUSTER with method = "ward" implements the Ward.D2 algorithm. ***hclust()*** can be made to produce results corresponding to the Ward.D2 algorithm by using squared distances in the input matrix. To obtain the Ward.D2 dendrogram with correct scale, one has to modify the $height element of the output list to make it contain the square roots of the height values before calling ***plot()***.

cluster sum of squares divided by the (corrected) total sum of squares. This statistic increases as the clusters grow.

Because the Ward method minimizes the sum of within-group sums of squares (squared error criterion), the clusters tend to be hyperspherical, i.e. spherical in multidimensional A-space, and to contain roughly equal numbers of objects if the observations are evenly distributed through A-space. The same applies to the centroid methods of the previous subsections. This may be seen as either an advantage or a problem, depending on the researcher's conceptual model of a cluster.

9 — General agglomerative clustering model

Lance & Williams (1966a, 1967c) proposed a general model that encompasses all the agglomerative clustering methods presented up to now, except intermediate linkage (Subsection 8.5.3). The general model offers the advantage of being translatable into a single, simple computer program, so that it is used in most statistical packages that offer agglomerative clustering, including R. The general model allows one to select an agglomerative clustering model by choosing the values of four parameters called α_h, α_i, β, and γ that determine the clustering strategy. This model only outputs the

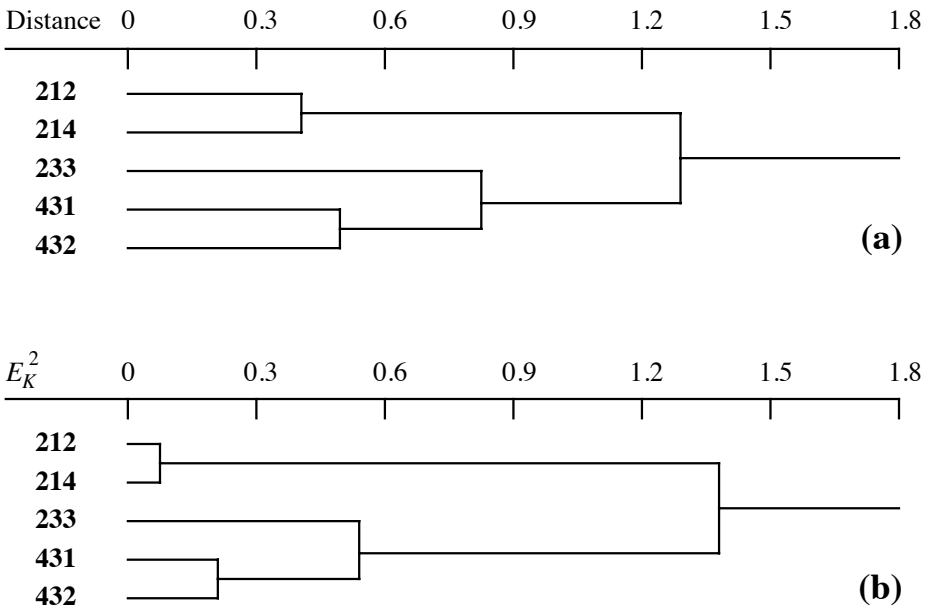


Figure 8.12 Ward's minimum variance clustering of the ponds from Ecological application 8.2. The scale of dendrogram (a) is the square root of the squared distances computed in Table 8.7; that scale can be compared to the original distances. In dendrogram (b), it is the E_K^2 (or TESS) statistic.

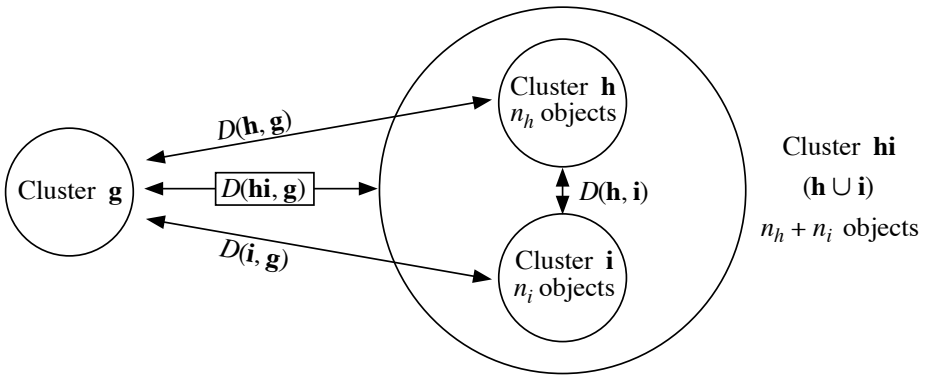


Figure 8.13 In combinatorial clustering methods, the distance between a cluster **hi**, resulting from the fusion of two previously formed clusters **h** and **i**, and an external cluster **g** is a function of the three distances between (**h** and **i**), (**h** and **g**), and (**i** and **g**), and of the number of objects in **h**, **i**, and **g**.

branching pattern of the clustering tree (the dendrogram), as it was the case for the methods described in Subsections 8.5.4 to 8.5.8. For the linkage clustering strategies (Subsections 8.5.1 to 8.5.3), the list of primary links responsible for cluster formation can be obtained afterwards by comparing the dendrogram to the distance matrix.

Combinatorial method

The model of Lance & Williams is limited to *combinatorial* clustering methods, i.e. those for which the distance $D(\mathbf{hi}, \mathbf{g})$ between an external cluster **g** and a cluster **hi**, resulting from the prior fusion of clusters **h** and **i**, is a function of the three distances $D(\mathbf{h}, \mathbf{g})$, $D(\mathbf{i}, \mathbf{g})$, and $D(\mathbf{h}, \mathbf{i})$ and also, eventually, of the numbers n_h, n_i, and n_g of objects in clusters **h**, **i**, and **g**, respectively (Fig. 8.13). Individual objects are considered to be single-member clusters. Since the distance of cluster **hi** to an external cluster **g** can be computed from the above six values, **h** and **i** can be condensed into a single row and a single column in the updated distance matrix; following that, the clustering proceeds as in the tables of the previous subsections. Since the new distances at each step can be computed by *combining* those from the previous step, it is not necessary for a computer program to retain the original distance matrix or data set. Non-combinatorial methods do not have this property. For distances, the general model for combinatorial methods is the following:

$$D(\mathbf{hi}, \mathbf{g}) = \alpha_h D(\mathbf{h}, \mathbf{g}) + \alpha_i D(\mathbf{i}, \mathbf{g}) + \beta D(\mathbf{h}, \mathbf{i}) + \gamma |D(\mathbf{h}, \mathbf{g}) - D(\mathbf{i}, \mathbf{g})| \qquad \textbf{(8.11)}$$

When using similarities, the combinatorial equation is:

$$S(\mathbf{hi}, \mathbf{g}) = (1 - \alpha_h - \alpha_i - \beta) + \alpha_h S(\mathbf{h}, \mathbf{g}) + \alpha_i S(\mathbf{i}, \mathbf{g}) + \beta S(\mathbf{h}, \mathbf{i}) - \gamma |S(\mathbf{h}, \mathbf{g}) - S(\mathbf{i}, \mathbf{g})| \qquad \textbf{(8.12)}$$

Table 8.9 Values of parameters α_h, α_i, β, and γ in Lance and Williams' general model for combinatorial agglomerative clustering. Modified from Sneath & Sokal (1973) and Jain & Dubes (1988).

Clustering method	α_h	α_i	β	γ	Effect on space A
Single linkage	1/2	1/2	0	–1/2	Contracting*
Complete linkage	1/2	1/2	0	1/2	Dilating*
UPGMA	$\frac{n_h}{n_h+n_i}$	$\frac{n_i}{n_h+n_i}$	0	0	Conserving*
WPGMA	1/2	1/2	0	0	Conserving
UPGMC	$\frac{n_h}{n_h+n_i}$	$\frac{n_i}{n_h+n_i}$	$\frac{-n_h n_i}{(n_h+n_i)^2}$	0	Conserving
WPGMC	1/2	1/2	–1/4	0	Conserving
Ward's	$\frac{n_h+n_g}{n_h+n_i+n_g}$	$\frac{n_i+n_g}{n_h+n_i+n_g}$	$\frac{-n_g}{n_h+n_i+n_g}$	0	Conserving
Flexible	$\frac{1-\beta}{2}$	$\frac{1-\beta}{2}$	$-1 \leq \beta < 1$	0	Contracting if $\beta \approx 1$ Conserving if $\beta \approx -.25$ Dilating if $\beta \approx -1$

* Terms used by Sneath & Sokal (1973).

Clustering proceeds in the same way for all combinatorial agglomerative methods. As the distances increases, a new cluster is obtained by the fusion of the two closest objects or groups, after which the algorithm proceeds to the fusion of the two corresponding rows and columns in the distance (or similarity) matrix using eq. 8.11 or 8.12. The matrix is thus reduced by one row and one column at each step. Table 8.9 gives the values of the four parameters for the most commonly used combinatorial agglomerative clustering strategies. Values of the parameters for some other clustering strategies are given by Gordon (1996a).

In the case of equality between two mutually exclusive pairs, the decision may be made on an arbitrary basis (the so-called "right-hand rule" used in most computer programs) or based upon ecological criteria as, for example, Sørensen's criteria reported at the end of Subsection 8.5.2, or those described in Subsection 8.9.1.

In several strategies, $\alpha_h + \alpha_i + \beta = 1$, so that the term $(1 - \alpha_h - \alpha_i - \beta)$ becomes zero and disappears from eq. 8.12. One can show how the values chosen for the four

parameters make the general equation correspond to each specific clustering method. For single linkage clustering, for instance, the general equation becomes:

$$D(\mathbf{hi}, \mathbf{g}) = \frac{1}{2}[D(\mathbf{h}, \mathbf{g}) + D(\mathbf{i}, \mathbf{g}) - |D(\mathbf{h}, \mathbf{g}) - D(\mathbf{i}, \mathbf{g})|]$$

The last term (absolute value) corrects the largest of the two distances $D(\mathbf{h}, \mathbf{g})$ and $D(\mathbf{i}, \mathbf{g})$, making it equal to the smallest one. Hence, $D(\mathbf{hi}, \mathbf{g}) = \min[D(\mathbf{h}, \mathbf{g}), D(\mathbf{i}, \mathbf{g})]$. In other words, the distance between a newly-formed cluster **hi** and some other cluster **g** becomes equal to the smallest of the distance values previously computed between the two original clusters (**h** and **i**) and **g**.

Intermediate linkage clustering is not a combinatorial strategy. All along the clustering procedure, it is necessary to refer to the original association matrix in order to calculate the connectedness of pairs of clusters. This is why it cannot be obtained using the Lance & Williams general agglomerative clustering model.

10 — Flexible clustering

Lance & Williams (l966a, 1967c) proposed to vary parameter β (eq. 8.11 or 8.12) between –1 and +1 to obtain a series of intermediate solutions between single linkage chaining and the space dilation of complete linkage. The method is called *beta-flexible clustering* by some authors. Lance & Williams (*ibid*.) have shown that, if the other parameters are constrained a follows:

$$\alpha_h = \alpha_i = (1 - \beta)/2 \qquad \text{and} \qquad \gamma = 0$$

the resulting clustering is always ultrametric (no reversals; Section 8.6).

When β is close to 1, strong chaining is obtained. As β decreases and becomes negative, space dilation increases. The space properties are conserved for small negative values of β near –0.25. Figure 8.14 shows the effect of varying β in the clustering of 20 objects. Like weighted centroid clustering, flexible clustering is compatible with all association measures except correlation coefficients.

Ecological application 8.5

Pinel-Alloul *et al*. (1990) studied phytoplankton in 54 lakes of Québec to determine the effects of acidification, physical and chemical characteristics, and lake morphology on species assemblages. Phytoplankton was enumerated into five main taxonomic categories (microflagellates, chlorophytes, cyanophytes, chrysophytes, and pyrrophytes). The data were normalized using the generalized form of the Box-Cox method that finds the best normalizing transformation for all species (Subsection 1.5.6). A Gower (S_{19}) similarity matrix, computed among lakes, was subjected to flexible clustering with parameter $\beta = -0.25$. Six clusters were found, which were roughly distributed along a NE-SW geographic axis and corresponded to increasing concentrations of total phytoplankton, chlorophytes, cyanophytes, and microflagellates. Explanation of the phytoplankton-based lake typology was sought by comparing it to the environmental variables (Subsection 10.2.1).

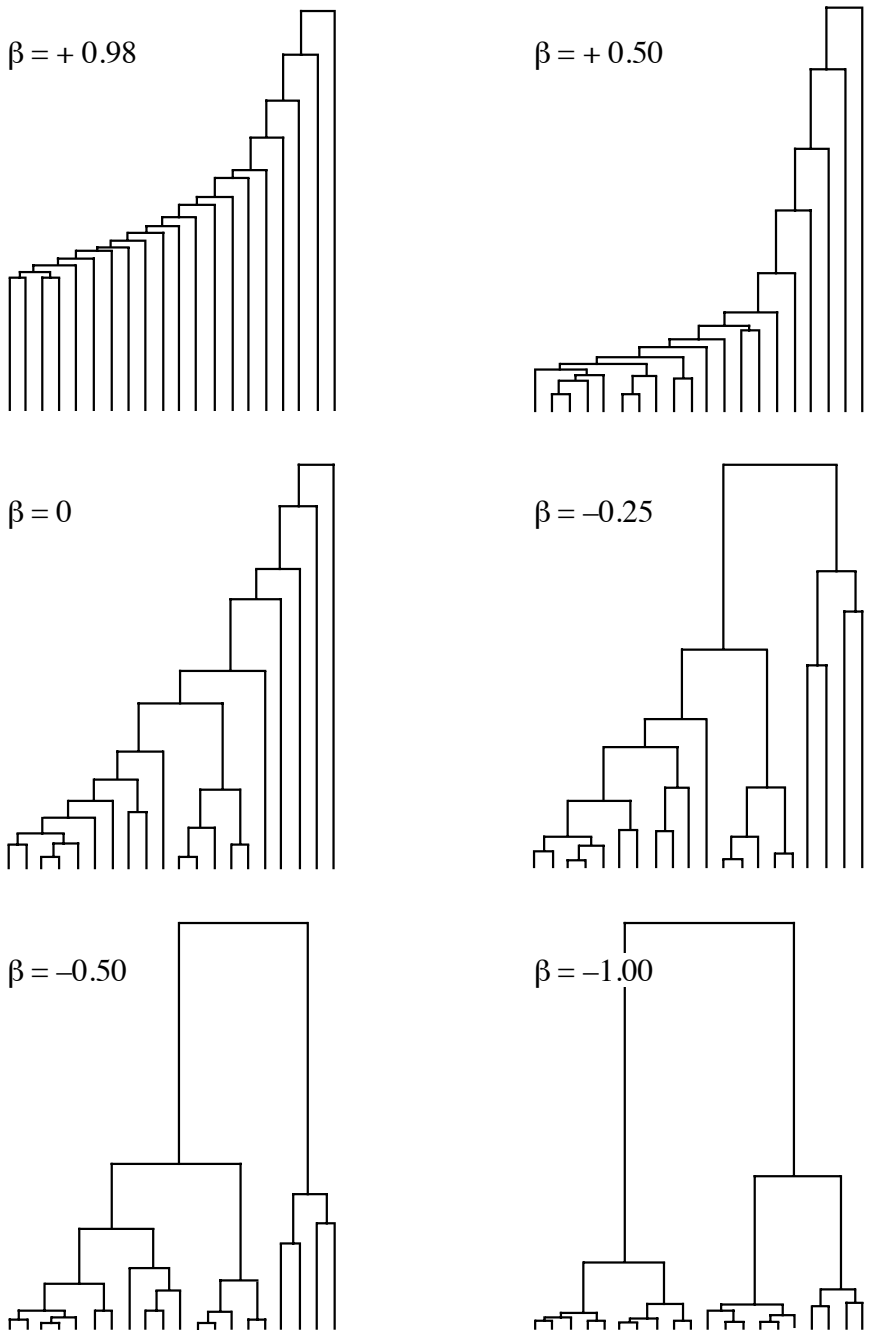


Figure 8.14 Flexible clustering of 20 objects for six values of β. The measure of association is the squared Euclidean distance D_1^2. Adapted from Lance & Williams (1967c: 376).

11 — Information analysis

Information analysis is a Q-mode clustering method developed for ecological purposes by Williams *et al*. (1966) and Lance & Williams (1966b). It does not go through the usual steps of distance calculation followed by clustering. It is a direct method of clustering based on information measures.

Entropy

Shannon's formula (eq. 6.1) can be used to measure the diversity or information in a frequency or probability distribution:

$$H = -\sum_{j=1}^{p} p_j \log p_j$$

Information analysis is a type of unweighted centroid clustering, adapted to species presence-absence data. At each step, the two objects or clusters causing the smallest gain in within-group diversity (or information) are fused. As a consequence, the clusters are as homogeneous as possible in terms of species composition.

The method could be applied to species abundance data divided into a small number of classes but, in practice, it is mostly used with presence-absence data. The information measure described below is not applicable to raw abundance data because the number of different states would then vary from one species to another, which would give them different weights in the overall measure.

To illustrate the method, the pond zooplankton counts used in Chapter 7 to illustrate the calculation of coefficient S_{23} (eq. 7.30) are transformed here into presence-absence data:

Species j	Ponds					p_j	$(1 - p_j)$
	212	214	233	431	432		
1	1	1	0	0	0	0.4	0.6
2	0	0	1	1	0	0.4	0.6
3	0	1	1	0	1	0.6	0.4
4	0	0	1	1	1	0.6	0.4
5	1	1	0	0	0	0.4	0.6
6	0	1	0	1	1	0.6	0.4
7	0	0	0	1	1	0.4	0.6
8	1	1	0	0	0	0.4	0.6

Total information in this group of ponds is computed using an information measure derived from the following reasoning (Lance & Williams, 1966b). The entropy of each species presence-absence descriptor j is calculated on the basis of the probabilities of presence p_j and absence $(1 - p_j)$ of species j, which are written in the right-hand part of

the table. The probability of presence is estimated as the number of ponds where species j is present, divided by the total number of ponds *in the cluster under consideration* (here, the group of five ponds). The probability of absence is estimated likewise, using the number of ponds where species j is absent. The entropy of species j is therefore:

$$H(j) = -[p_j \log p_j + (1 - p_j) \log(1 - p_j)] \quad \text{for } 0 < p_j < 1 \qquad \textbf{(8.13)}$$

The base of the logarithms is indifferent, as long as the same base is used throughout the calculations. Natural logarithms are used throughout the present example. For the first species, $H(1)$ would be:

$$H(1) = -[0.4 \log_e(0.4) + 0.6 \log_e(0.6)] = 0.673$$

The information of the conditional probability table can be calculated by summing the entropies per species, considering that all species have the same weight. Since the measure of *total information* in the group must also take into account the number of objects in the cluster, it is defined as follows:

$$I = -n \sum_{j=1}^{p} [p_j \log p_j + (1 - p_j) \log (1 - p_j)] \quad \text{for } 0 < p_j < 1 \qquad \textbf{(8.14)}$$

where p is the number of species represented in the group of n objects (ponds). For null probabilities, $\lim_{p \to 0} [-p \log (p)] = 0$. For the group of 5 ponds above,

$$I = -5 [8 (-0.673)] = 26.920$$

If I is to be expressed as a function of the number a_j of ponds with species j present, instead of a function of probabilities $p_j = a_j/n$, it can be shown that the following formula is equivalent to eq. 8.14:

$$I = np \log n - \sum_{j=1}^{p} [a_i \log a_j + (n - a_j) \log (n - a_j)] \qquad \textbf{(8.15)}$$

I is zero when all ponds in a group contain the exact same set of species. Like entropy H, I has no upper limit; its maximum value depends on the number of species present in the study.

At each clustering step, three series of values are computed: (a) the total information I in each group, which is 0 at the beginning of the process since each object (pond) then forms a distinct cluster; (b) the value of I for all possible combinations of groups taken two at a time; and (c) the increase of information ΔI resulting from each possible fusion. As recommended by Sneath & Sokal (1973), all these values can be written in a matrix, initially of dimension $n \times n$ which decreases as clustering proceeds. For the example data, values (a) of information in each group are

placed on the diagonal, values (b) of I in the lower triangle, and values (c) of ΔI in the upper triangle, in italics.

Ponds	Ponds				
	212	214	233	431	432
212	0	*2.773*	*8.318*	*9.704*	*9.704*
214	2.773	0	*8.318*	*9.704*	*6.931*
233	8.318	8.318	0	*4.159*	*4.159*
431	9.704	9.704	4.159	0	*2.773*
432	9.704	6.931	4.159	2.773	0

The value ΔI for two groups is found by subtracting the two corresponding values I, on the diagonal, from the value I of their combination in the lower triangle. Values on the diagonal are 0 in this first calculation matrix, so that values in the upper triangle are the same as in the lower triangle, but this will not be the case in subsequent matrices.

The first fusion is identified by the lowest ΔI value found in the upper triangle. This value is 2.773 for pairs (212, 214) and (431, 432), which therefore fuse. A new matrix of I values is computed:

Groups	Groups		
	212 214	233	431 432
212-214	2.773	*10.594*	*15.588*
233	13.367	0	*4.865*
431-432	21.134	7.638	2.773

This time, the ΔI values in the upper triangle differ from the I's in the lower triangle since there are now I values different from 0 on the diagonal. The ΔI corresponding to group (212, 214, 431, 432), for example, is computed as: 21.134 – 2.773 – 2.773 = 15.588. The lowest value of ΔI is for the group (233, 431, 432), which therefore fuses at this step at information level I = 7.638.

For the last clustering step, the only I value to calculate in the lower triangle is for the cluster containing the five ponds. This value, computed above after eq. 8.14, is 26.920. ΔI is then 26.920 – 2.773 – 7.638 = 16.509.

Groups	Groups	
	212 214	233 431-432
212-214	2.773	*16.509*
233-431-432	26.920	7.638

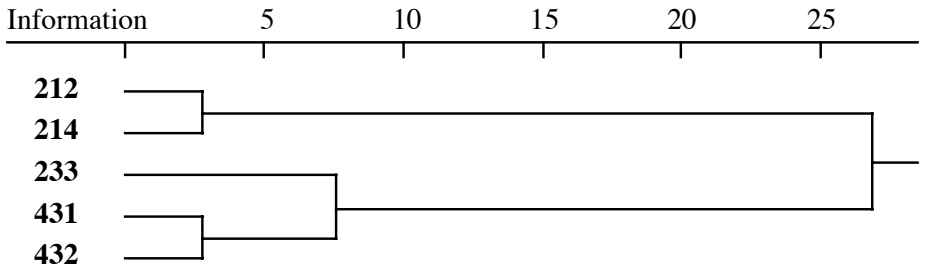


Figure 8.15 Clustering of the ponds from Ecological application 8.2, using information analysis.

The last fusion occurs at $I = 26.920$; computing ΔI is not necessary in this case. The values of I can be used as the scale for a dendrogram summarizing the clustering steps (Fig. 8.15). The same topology is obtained as in Figs. 8.3 to 8.11.

According to Williams *et al*. (1966), information analysis minimizes chaining and quickly delineates the main clusters, at least with ecological data. Field (1969) pointed out, however, that information analysis bases the similarity between objects on double absences as well as double presences. This method may therefore not be appropriate when a gradient has been sampled and the data matrix contains many zeros; see Subsections 7.2.2 and 9.2.5 for discussions of this problem.

Efficiency coefficient

The inverse of ΔI is known as the *efficiency coefficient* (Lance & Williams, 1966b). An analogue to the efficiency coefficient can be computed for dendrograms obtained using other agglomerative clustering procedures. In that case, the efficiency coefficient is still computed as $1/\Delta I$, where ΔI represents the amount by which the information in the classification is reduced due to the fusion of groups. The reduction is computed as the entropy in the classification before a fusion level minus the entropy after that fusion. In Fig. 8.2b for instance, the partition at $D = 0.60$ contains three groups of 2, 2, and 1 objects respectively; using natural logarithms, Shannon's formula (eq. 6.1) gives $H = 1.05492$. The next partition, at $D = 0.75$, contains two groups with 2 and 3 objects; Shannon's formula gives $H = 0.67301$. The difference is $\Delta = 0.38191$, hence the efficiency coefficient is $1/\Delta I = 2.61843$ for fusion level $D = 0.7$ of the dendrogram.

When $1/\Delta I$ is high, the procedure clusters objects that are mostly alike. The efficiency coefficient does not monotonically decrease as the clustering proceeds. With real data, it may decrease, reach a minimum, and increase again. If $1/\Delta I$ is plotted as a function of the successive fusion levels, the minima in the graph indicate the most informative partitions. If one wants to select a single cutting level in a dendrogram, this graph may help in deciding which partition should be selected. In Fig. 8.2b for example, one would choose the value $1/\Delta I = 1.48586$, which corresponds to the last fusion level ($D = 0.786$), as the most informative partition. The efficiency coefficient is not a rigorous decision criterion, however, since no test of significance is performed.

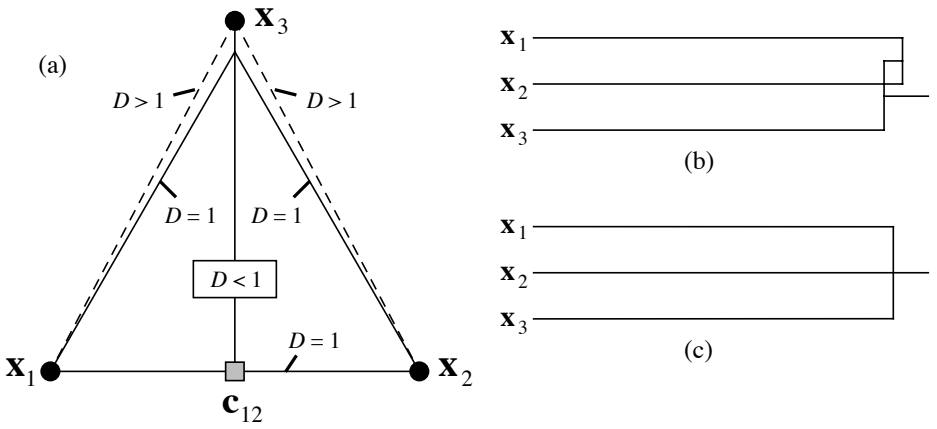


Figure 8.16 A reversal may occur in situations such as (a), where $\mathbf{x}_1$ and $\mathbf{x}_2$ cluster first because they represent the closest pair, although the distance from $\mathbf{x}_3$ to the centroid $\mathbf{c}_{12}$ is smaller than the distance from $\mathbf{x}_1$ to $\mathbf{x}_2$. (b) The result is usually depicted by a non-ultrametric dendrogram with reversal. (c) The reversal may also be interpreted as a trichotomy.

8.6 Reversals

Reversals may occasionally occur in the clustering structure when using UPGMC or WPGMC (Subsections 8.5.6 and 8.5.7), or with some unusual combinations of parameters in the general agglomerative model of Lance & Williams (Subsection 8.5.9). As an example, a reversal was produced in Fig. 8.7. Two types of situations lead to reversals:

- When $\mathbf{x}_1$ and $\mathbf{x}_2$ cluster first, because they represent the closest pair, although the distance from $\mathbf{x}_3$ to the centroid $\mathbf{c}_{12}$ is smaller than the distance from $\mathbf{x}_1$ to $\mathbf{x}_2$ (Fig. 8.16a).

- When $D(\mathbf{x}_1, \mathbf{x}_2) = D(\mathbf{x}_1, \mathbf{x}_3) = D(\mathbf{x}_2, \mathbf{x}_3)$. In such a situation, most computer programs use an arbitrary rule ("right-hand rule") and first cluster two of the three objects. A reversal appears when the third object is added to the cluster.

When this happens, the cophenetic matrix (Subsection 8.3.1) violates the ultrametric property (Subsection 8.3.2) and the dendrogram is more difficult to draw than in the no-reversal cases (Fig. 8.16b). However, departures from ultrametricity are never large in practice. For this reason, a reversal may be interpreted as nearly equivalent to a trichotomy in the hierarchical structure (Fig. 8.16c). They may also indicate true trichotomies, as discussed above; this can be checked by examination of the distance matrix.

A clustering method is said to be *monotonic* (i.e. without reversals) if

$$D(\mathbf{x}_1 \cup \mathbf{x}_2, \mathbf{x}_3) \geq D(\mathbf{x}_1, \mathbf{x}_2)$$

or

$$S(\mathbf{x}_1 \cup \mathbf{x}_2, \mathbf{x}_3) \leq S(\mathbf{x}_1, \mathbf{x}_2)$$

Assuming that $\alpha_h > 0$ and $\alpha_i > 0$ (Table 8.9), necessary and sufficient conditions for a clustering method to be monotonic in all situations are the following:

$$\alpha_h + \alpha_i + \beta \geq 1$$

and

$$\gamma \geq -\min(\alpha_h, \alpha_i)$$

(Milligan, 1979; Jain & Dubes, 1988). Some authors use the term *classification* only for hierarchies *without reversals* or for non-overlapping partitions of the objects (Table 8.1, Section 8.8).

8.7 Hierarchical divisive clustering

Contrary to the agglomerative methods of Section 8.5, hierarchical divisive techniques use the whole set of objects as the starting point. They divide it into two or several subgroups, after which they consider each subgroup and divide it again, until the criterion chosen to end the divisive procedure is met (Lance & Williams, 1967b).

In practice, hierarchical divisive clustering can only be achieved in the monothetic case or when working in an ordination space. In monothetic divisive methods, the objects are divided, at each step of the procedure, according to the states of a single descriptor. This descriptor is chosen because it best represents the whole set of descriptors (next subsection). Polythetic algorithms have been developed, but it will be seen that they are not satisfactory.

An alternative is to use a partitioning method (Section 8.8) for several numbers of groups from $K = 2$ and up and assemble the results into a graph. There is no guarantee, however, that the groups will be nested and form a hierarchy, unless the biological or ecological processes that have generated the data are themselves hierarchical.

1 — Monothetic methods

Association analysis

The clustering methods that use only one descriptor at a time are less than ideal, even when the descriptor is chosen after considering all the others. In ecology, the best-known monothetic method is Williams & Lambert's (1959) *association analysis*, originally described for species presence-absence data. Association analysis may actually be applied to any binary data table, not only species. The problem is to identify, at each step of the procedure, which descriptor is the most strongly associated

with all the others. First, X^2 (chi-square) statistics are computed for 2 × 2 contingency tables comparing all pairs of descriptors in turn. X^2 is computed using the usual formula:

$$X^2 = n\,(ad - bc)^2/[(a + b)\,(c + d)\,(a + c)\,(b + d)]$$

The formula may include Yates' correction for small sample sizes, as in similarity coefficient S_{25}. The X^2 values relative to each descriptor k are summed up:

$$\sum_{j=1}^{p} X_{jk}^2 \quad \text{for} \quad j \neq k \qquad \textbf{(8.16)}$$

The largest sum identifies the descriptor that is the most closely related to all the others. The first partition is made along the states of that descriptor; a first cluster is made of the objects coded 0 for the descriptor and a second cluster for the objects coded 1. The descriptor is eliminated from the study and the procedure is repeated, separately for each cluster. Division stops when the desired number of clusters is attained or when the sum of X^2 values no longer reaches a previously set threshold.

This method has been adapted by Lance & Williams (1968) to the information statistic I of Subsection 8.5.11. Lance & Williams (1965) also suggested using the point correlation coefficient $\varphi = \sqrt{X^2/n}$ (eq. 7.9) instead of X^2. This may prevent aberrant or unique objects in the study from determining the first partitions. This is analogous to the problem encountered with the higher powers of Minkowski's metric (D_6), which could give too much weight to the largest differences; this problem was less severe when using power 1, which is the Manhattan metric (D_7). One then looks for the descriptor that maximizes the sum $\Sigma\varphi_{jk}$ ($j \neq k$; see eq. 8.16). Gower (1967) suggested to use, for division, the species that has the largest R^2 with all the other species (eq. 10.20), instead of the one with the largest sum of simple correlations. He also suggested to use the largest variance inflation factor (*VIF*) as criterion, instead of the largest R^2, because *VIF* is monotonically related to R^2 (eq. 10.17). *VIF* can be computed by a single matrix operation for all species (sentence that follows eq. 10.17).

The principles of association analysis may be applied to descriptors with multiple states (semiquantitative or qualitative), by computing X^2 statistics between descriptors using the usual X^2 formulas (eqs. 6.5 and 6.6). Raw species abundance data should not be analysed in this way, however, because the large number of different abundance values makes the contingency tables meaningless.

Legendre & Rogers (1972) proposed a monothetic divisive method similar to association analysis, in which the choice of the descriptor best representing all the others is made with the help of an information statistic computed on contingency tables. For each descriptor k, two quantities developed by Christanson (*in* Brill *et al.*, 1972) are computed: SUMRAT (k) and SAMRAT (k) ("sum of ratios"). SUMRAT (k) is the sum of the fractions representing the amount of information that k has in common with

each descriptor j ($j \neq k$), divided by the amount of information in j. In SAMRAT (k), the divisor is the amount of information in k instead of j. Using the symbolism of Section 6.2:

$$\text{SUMRAT}\ (k) = \sum_{j=1}^{p} \frac{H(k) - H(k|j)}{H(j)} \quad \text{for} \quad j \neq k \qquad \textbf{(8.17)}$$

$$\text{SAMRAT}\ (k) = \sum_{j=1}^{p} \frac{H(k) - H(k|j)}{H(k)} \quad \text{for} \quad j \neq k \qquad \textbf{(8.18)}$$

which can be recognized as sums of asymmetric uncertainty coefficients, $\sum$ B/(B + C) and $\sum$ B/(A + B), respectively (Section 6.2). SUMRAT (k) and SAMRAT (k) both have the property of being high when k has much information in common with the other descriptors in the study. The descriptor that best represents the divisive power of all descriptors is expected to have the highest SUMRAT and SAMRAT values. However, SUMRAT (k) and SAMRAT (k) are also influenced by the number of states in k, which may unduly inflate $H(k)$, thus causing SUMRAT (k) to increase and SAMRAT (k) to decrease. This factor must be taken into account if there is conflict between the indications provided by SUMRAT and SAMRAT as to the descriptor that best represents the whole set. This peculiarity of the method requires the user's intervention at each division step, in the present state of development of the equations.

Since the information measures on which SUMRAT and SAMRAT are based are at the same exponent level as X^2 (Section 6.2), one could compute instead:

$$\text{SUMRAT}\ (k) = \sum_{j=1}^{p} \sqrt{\frac{H(k) - H(k|j)}{H(j)}} \quad \text{for} \quad j \neq k \qquad \textbf{(8.19)}$$

$$\text{SAMRAT}\ (k) = \sum_{j=1}^{p} \sqrt{\frac{H(k) - H(k|j)}{H(k)}} \quad \text{for} \quad j \neq k \qquad \textbf{(8.20)}$$

thus minimizing the effect of single objects on the first partitions, as indicated above.

Williams & Lambert (1961) have suggested using association analysis in the R mode for identifying species associations. This approach does not seem, however, to be based on an acceptable operational concept of association (see Section 8.9).

2 — Polythetic methods

There is no satisfactory algorithm for the hierarchical division of objects based on the entire set of descriptors.

The method of Edwards & Cavalli-Sforza (1965) tries all possible divisions of the set of objects into two clusters, looking for the division that maximizes the distance

between the centroids. Using sums of squared distances to centroids, one first computes *SS*, which is the sum of squares of the Euclidean distances of all objects to the centroid of the whole set of objects, divided by the number of objects *n*; this value is the total sum of squares of a single classification analysis of variance (eqs. 6.56 and 8.6). Then, for each possible partition of the objects into two groups **h** and **i**, the sums of squares of the distances to the centroids are computed within each cluster, using eq. 8.6, to obtain $SS(\mathbf{h})$ and $SS(\mathbf{i})$, respectively. The distance between the two clusters is therefore $SS - SS(\mathbf{h}) - SS(\mathbf{i})$. This is the quantity to be maximized for the first partition. Then each cluster is considered in turn and the operation is repeated to obtain subsequent divisions. Like *K*-means partitioning of Section 8.8, this method can only be applied to quantitative data because it is based on Euclidean distances.

This method may seem attractive but, apart from the theoretical objections that he raised about it, Gower (1967) noted that investigating all possible partitions to find the best one is a NP–hard computational problem (footnote in Section 8.8). He calculated that, before obtaining the first partition of a cluster of 41 objects, 54000 years of computing time would be required using a computer with an access time of 5 microseconds, to try all $(2^{40} - 1)$ possible partitions of 41 objects into two groups. 5 microseconds was the typical access time of computers in 1967. The problem remains with modern computers, even though they have much smaller access times (in the realm of nanoseconds at the beginning of the years 2010). The heuristic algorithms used to solve the *K*-means problem (Section 8.8) could, however, be applied here instead of the complete search through all possible solutions.

Dissimilarity analysis

The *dissimilarity analysis* of Macnaughton-Smith *et al*. (1964) first looks for the object that is the most different from all the others and removes it from the initial cluster. One by one, the most different objects are removed. Two groups are defined: the objects removed and the remaining ones, between which a distance is calculated. Objects are removed up to the point where the distance between clusters can no longer be increased. Each of the two clusters thus formed is subdivided again, using the same procedure. The first partition of a cluster of *n* objects requires at most $3n^2/4$ operations instead of the $(2^{n-1} - 1)$ operations required by the previous method. Other authors have developed special measures of distance to be used in dissimilarity analysis, such as Hall's (1965) *singularity index* and Goodall's (1966b) *deviant index*. Although attractive, dissimilarity analysis may produce strange results when many small clusters are present in the data, in addition to major clusters of objects.

A major disadvantage of all hierarchical divisive methods is that a division of the objects in two major clusters may also split the members of some minor cluster, which cannot be fused again unless special procedures are included in the algorithm for that purpose (Williams & Dale, 1965).

3 — Division in ordination space

Computer-efficient polythetic hierarchical divisive clustering can be obtained by partitioning the objects according to the axes of an ordination space. Using principal

component analysis (PCA, Section 9.1), the set of objects may be partitioned in two groups: those that have positive values along the first PCA axis and those that have negative values. The PCA analysis is repeated for each of the groups so obtained and a new partition of each group is performed. The process is repeated until the desired level of resolution is obtained (Williams, 1976b).

Following a similar suggestion by Piazza & Cavalli-Sforza (1975), Lefkovitch (1976) developed a hierarchical classification method for very large numbers of objects, based on principal coordinate analysis (PCoA, Section 9.3). The dendrogram is constructed from the successive principal coordinate axes, the signs of the objects on the coordinate axes indicating their membership in one of the two groups formed at each branching step. The objects are partitioned in two groups according to their signs along the first PCoA axis; each group is then divided according to the positions of the objects along the second axis; and so on. This differs from the method used with PCA above, where the analysis is repeated for each group before a new division takes place. To calculate the principal coordinates of a large number of objects, Lefkovitch proposed to first measure the similarity among objects by an equation which, like the covariance or correlation, is equivalent to the product of a matrix with its transpose. He described such a measure, applicable if necessary to combinations of binary, semiquantitative, and quantitative descriptors. The association matrix among objects is obtained by the matrix product $\mathbf{YY}'$ (order $n \times n$). In situations where there are many more objects than descriptors, computation of the eigenvalues and eigenvectors of the association matrix among *descriptors,* $\mathbf{Y}'\mathbf{Y}$, represents an important saving of computer time because $\mathbf{Y}'\mathbf{Y}$ (order $p \times p$) is much smaller than $\mathbf{YY}'$ (order $n \times n$). After Rao (1964) and Gower (1966), Lefkovitch showed that the principal coordinates $\mathbf{V}$ of the association matrix among *objects* can then be found, using the relation $\mathbf{V} = \mathbf{YU}$ where $\mathbf{U}$ is the matrix of the principal coordinates among *descriptors*. The principal coordinates thus calculated allow one to position the objects, numerous as they may be, in the reduced space. Principal coordinates can be used for the binary hierarchical divisive classification procedure that was Lefkovitch's goal.

A divisive algorithm of the same type is used in TWINSPAN (next subsection). It is based upon an ordination obtained by correspondence analysis instead of PCA or PCoA.

4 — TWINSPAN

Two Way INdicator SPecies ANalysis (TWINSPAN*) (Hill, 1979a) is fundamentally a method for hierarchical divisive classification of communities, based on progressive refinement of a single ordination axis obtained by correspondence analysis (CA) or

* Available as part of the package PC-ORD (distribution: footnote in Section 11.7). TWINSPAN is also available from Micro-computer Power: <http://www.microcomputerpower.com>. The TWINSPAN source code in FORTRAN and an executable version for Windows are available on Jari Oksanen's page <http://cc.oulu.fi/~jarioksa/softhelp/ceprog.html>. An executable program for Windows, WINTWINS, is available on the page http://www.canodraw.com/wintwins.htm.

detrended correspondence analysis (DCA) (Section 9.2) of a (sites × species) data matrix. Hill (1979a) also called the method a *dichotomized ordination analysis*.

An attractive feature of the output is a two-way table where the sites (columns) are sorted according to the splits of the hierarchical classification. The species (rows) are also sorted so as to form blocks corresponding to the groups of sites of the classification. A dendrogram representing the classification of the sites can easily be drawn, if required, from the TWINSPAN output table. In addition, the method computes an indicator values index (I) for the species for every split of the hierarchical classification of the sites.

Pseudo-species

To model the concept of *differential species* (i.e. species with clear ecological preferences), which is qualitative, TWINSPAN creates *pseudospecies*. Each species is recoded into a set of dummy variables (pseudospecies) corresponding to relative abundance levels; these classes are cumulative. If, for example, the pseudospecies cutting levels are 1%, 11%, 26%, 51% and 76%, a relative abundance of 18% at a site will fill the first and second dummy pseudospecies vectors with "1" (= presence). Cutting levels are arbitrarily decided by users. A (sites × pseudospecies) data table is thus created.

The TWINSPAN procedure is rather complex. A detailed description is given by Kent & Coker (1992). It may be summarized as follows.

1. After ordination by CA or DCA of the original (sites × species) data table, the objects are divided in two groups according to their signs along the first ordination axis. This is called the primary ordination.

2. TWINSPAN then computes an indicator values index (I) for the species, for every split of the hierarchical classification of the sites. According to Kent & Coker (1992), the index is computed as follows using the pseudospecies data:

Indicator value index

$$I_j = \frac{n_j^+}{n^+} - \frac{n_j^-}{n^-}$$

where n^+ and n^- are respectively the number of sites on the arbitrarily chosen positive and negative sides of the split, whereas n_j^+ and n_j^- are the number of sites on the positive and negative sides, respectively, that contain pseudospecies j. A pseudospecies present in every site on the positive side and in none of the sites on the negative side obtains $I_j = 1$, and -1 if it is found in every site on the negative side and in none on the positive side. A pseudospecies that occurs in all sites on both sides of the split obtains $I_j = 0$. In TWINSPAN, the indicator value describes the preference of a pseudospecies for one or the other side of the partition. The pseudospecies with the highest indicator absolute value is counted as the best indicator for that species. Then, only one pseudospecies of a single species is declared an indicator of a split, and that is the pseudospecies that has the highest absolute value of I. n_j^+/n^+ is the measure of *fidelity* to a group used in the *INDVAL* method described in Subsection 8.9.3.

Fidelity

3. Further steps lead to a refined ordination of the objects. After taking care of misclassifications, borderline cases, and other problems, a final division of the sites is obtained. Then, each subset is divided into smaller subsets by repeating the procedure. This goes on until groups become very small. Typically, groups of 4 objects or less are not partitioned further.

Problems with TWINSPAN are the following: (1) To identify species groups or compute indicator values, one cannot introduce some other classification of the sites in the program; only the classification produced by TWINSPAN, which is based on correspondence analysis (CA, Section 9.2) or detrended correspondence analysis (DCA, Subsection 9.2.5), can be used to delineate species groups. (2) The pseudospecies concept is based on species relative abundances. The relative abundance of a species depends on the absolute abundances of the other species present at a site. Such relative frequencies may be highly biased, in particular, when sampling mobile organisms: all species are not sampled with the same efficiency because of differences in behaviour. So, the coding of species abundances into pseudospecies may be highly unstable.

TWINSPAN has also been criticized by Belbin and McDonald (1993) on two grounds: (1) The method assumes the existence of a strong gradient dominating the data structure, so that it may fail to identify secondary gradients or other types of structures in data sets. (2) The cutting points along the dominant axis for the whole group, and then for subgroups, are always chosen to be the centroid of the group to be split instead of a point where a large gap occurs in the data. This problem has been alleviated by a modification to the method proposed by Rolecek *et al*. (2009).

An alternative method to obtain a reordered species-by-sites table is seriation (Section 8.10). In R (Section 8.15), a plot ("heat map") can be produced using functions ***hmap()*** of package SERIATION or ***heatmap()*** of STATS.

8.8 Partitioning by *K*-means

Partitioning consists in finding a single partition of a set of objects (Table 8.1). Jain & Dubes (1988) stated the problem in the following terms: given n objects in a p–dimensional space, determine a partition of the objects into K groups, or clusters, such that the objects within each cluster are more similar to one another than to objects in the other clusters. The number of groups, K, is determined by the user. This problem was first stated in statistical terms by Fisher (1958) who proposed solutions for a single variable (with or without contiguity constraint; see Sections 12.6 and 13.3). *K*-means partitioning is available in several R functions; see Section 8.15.

The difficulty is to define what 'more similar' means. Several criteria have been suggested; they can be divided into global and local criteria. A *global criterion* would be, for instance, to represent each cluster by a type-object (on *a priori* grounds, or

using the centroids obtained by agglomerative clustering, Subsections 8.5.6 and 8.5.7) without consideration for local densities of objects and assign each object to the nearest type-object. A type object representing a cluster is called a *medoid* (Kaufman & Rousseeuw, 1990). A *local criterion* uses the local structure of the data to delineate clusters; groups are formed by identifying high-density regions in the data represented in A-space (Fig. 7.2). The *K*-means method, described in the next paragraphs, is the most commonly used of the latter type. *K*-means belongs to a larger class of methods called *K*-centroid cluster analysis, which is briefly described in Section 8.15.

Medoid

Objective function

In *K*-means, the objective function that the partition should minimize is the same as in Ward's agglomerative clustering method (Subsection 8.5.8): the total error sum of squares (E_K^2, or TESS). The major problem encountered by the algorithms is that the solution on which the computation eventually converges depends to some extent on the initial positions of the centroids. This problem does not exist in Ward's method, which proceeds iteratively by hierarchical agglomeration. However, even though Ward's algorithm guarantees that the *increase* in sum of squared errors ($\Delta E_{\mathbf{hi}}^2$, eq. 8.8) is minimized at each step of the agglomeration (so that any order of entry of the objects should lead to the same solution, except in cases of equal distances where a "right-hand" programming rule may prevail), there is no guarantee that any given Ward's partition is optimal in terms of the E_K^2 criterion — surprising at this may seem. This same problem occurs with all stepwise statistical methods.

Local minimum

The problem of the final solution depending on the initial positions of the centroids is known as the "local minimum" problem in algorithms. The concept is illustrated in Fig. 8.17, by reference to a *solution space*. It may be explained as follows. Solutions to the *K*-means problem are the different ways to partition n objects into, say, $K = 4$ groups. If a single object is moved from one group to another, the corresponding two solutions will have slightly different values for the criterion to be minimized (E_K^2). Imagine that all possible solutions form a "space of solutions". The different solutions can be plotted as a graph with the E_K^2 criterion as the ordinate. It is not essential to accurately describe the abscissa to understand the concept; it would actually be a multidimensional space. A *K*-means algorithm starts at some position in that space, the initial position being assigned by the user (see below). It then tries to navigate the space to find the solution that minimizes the objective criterion (E_K^2). The space of solutions is not smooth, however. It may contain *local minima* from which the algorithm may be unable to escape. When this happens, the algorithm has not found the overall minimum and the partition is not optimal in terms of the objective criterion.

Overall minimum

Several approaches may be used to help a *K*-means algorithm converge towards the overall minimum of the objective criterion E_K^2. They involve either selecting specific objects as "group seeds" at the beginning of the run, or attributing the objects to the K groups in some special way. Here are some commonly-used approaches:

- Provide an initial configuration corresponding to an (ecological) hypothesis. The idea is to start the algorithm in a position in the solution space that is, hopefully, close to the final solution sought. This ideal situation is seldom encountered in real studies.

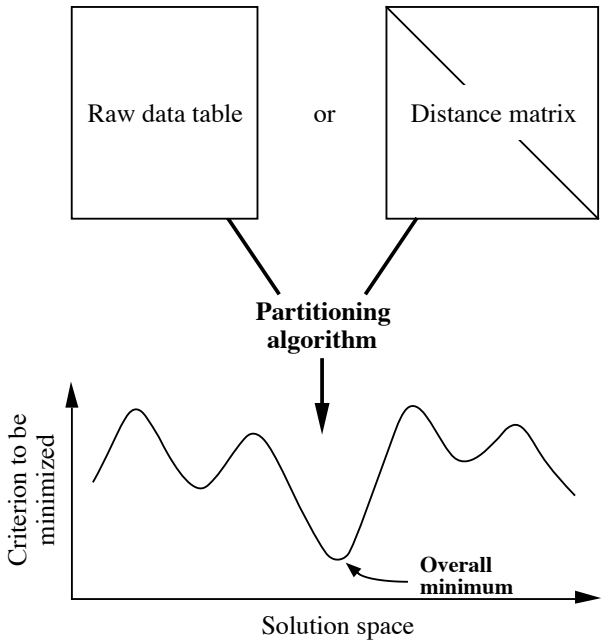


Figure 8.17 *K*-means algorithms search the space of solutions, trying to find the overall minimum (arrow) of the objective criterion to be minimized, while avoiding local minima (troughs).

• Provide an initial configuration corresponding to the result of a hierarchical clustering, obtained from a space-conserving method (Table 8.9). One simply chooses the partition into K groups found on the dendrogram and lists the objects pertaining to each group. The K-means algorithm will then be asked to rearrange the group membership and look for a better overall solution (lower E_K^2 statistic).

• If the program allows it, select as "group seed", for each of the K groups to be delineated, some object located near the centroid of that group. For very large problems, Lance & Williams (1967d) suggested to use as starting point the result of a hierarchical clustering of a *random subset of the objects*, using as "group seeds" either the centroids of K clusters, or objects located near these centroids.

• Attribute the objects at random to the various groups. All K-means computer programs offer this option. Find a solution and note the $E_{K_2}^2$ value. It is possible that the solution found corresponds to a local minimum of E_K^2. So, repeat the whole procedure a number of times (for example, 100 times), starting every time from a different random configuration. Retain the solution that minimizes the E_K^2 statistic. One is more confident that this solution corresponds to the overall minimum when the corresponding value of E_K^2 is found several times across the runs.

NP-hard

Several algorithms have been proposed to solve the *K*–means problem, which is but one of a family of problems known in computer sciences as the *NP–complete* or *NP–hard problems*[*]. In all these problems, the only way to be sure that the optimal solution has been found would be to try all possible solutions in turn. This is impossible, of course, for real-size problems, even with modern-day computers, as explained in Subsection 8.7.2. Classical references to *K*-means algorithms are Anderberg (1973), Hartigan (1975), Späth (1975, 1980), Everitt (1980), Jain & Dubes (1988) and Kaufman & Rousseeuw (1990). Milligan & Cooper (1987) reviewed the most commonly used algorithms and compared them for structure recovery, using artificial data sets. One of the best algorithms available is the following; it frequently converges to the solution representing the overall minimum for the E_K^2 statistic. It is a very simple alternating least-squares algorithm, which iterates between two steps:

- Compute cluster centroids and use them as new cluster seeds.
- Assign each object to the nearest seed.

At the start of the program, K observations are selected as "group seeds". Each iteration reduces the sum of squared errors E_K^2, if possible. Since only a finite number of partitions are possible, the algorithm eventually reaches a partition from which no improvement is possible; iterations stop when E_K^2 can no longer be improved. The FASTCLUS procedure of the SAS package, mentioned here because it can handle very large numbers of objects, uses this algorithm. Options of the program can help deal with outliers if this is a concern. The SAS manual (SAS Institute, 2011) provides more information on the algorithm and the available options.

K-means partitioning was originally proposed in a pioneering paper by MacQueen (1967) who gave the method its name: *K*-means. Lance & Williams made it popular by recommending it in their review paper (1967d). In the MacQueen paper, group centroids are recomputed after each addition of an object; this is also an option in SAS. MacQueen's algorithm contains procedures for the fusion of clusters, if centroids become very close, and for creating new clusters if an object is very distant from existing centroids.

K-means partitioning may be computed from either a table of raw data or a distance matrix, because the total error sum of squares E_K^2 (eq. 8.7) is equal to the sum of squares of the distances from the points to their respective centroids (eq. 8.5; Fig. 8.18a) and to the sum (over groups) of the mean squared within-group distances[†] (eq. 8.6; Fig. 8.18b). It is especially advantageous to compute it on raw data when the number of objects is large because, in such a situation, the distance matrix may

[*] *NP* stands for *Non-deterministic Polynomial*. In theory, these problems can be solved in polynomial time (i.e. some polynomial function of the number of objects) on a (theoretical) non-deterministic computer. NP-hard problems are probably not solvable by efficient algorithms.

[†] As shown in eq. 8.6, the mean squared distance within group k is computed as the sum of the squared within-group distances divided by the number of objects n_k in the group.

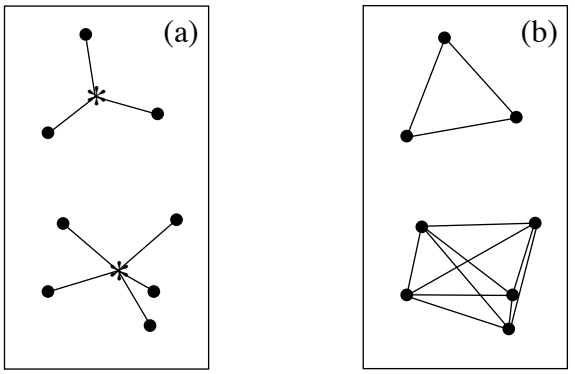


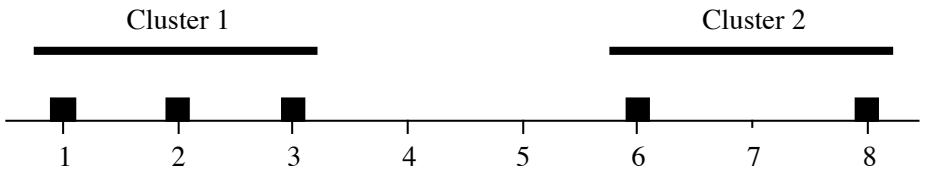
Figure 8.18 The total error sum of squares (TESS, E_K^2, eq. 8.7)) is equal (a) to the sum of squares of the distances from the points to their respective centroids (eq. 8.5). (b) It is also equal to the sum (over groups) of the mean squared within-group distances (eq. 8.6).

become very cumbersome or even impossible to store and search. In contrast, when using a table of original data, one only needs to compute the distance of each object to each group centroid, rather than to all other objects.

The disadvantage of using a table of raw data is that the only distance function among points, available during *K*-means partitioning, is the Euclidean distance (D_1, Chapter 7) in A-space. This is not suitable for species counts and other types of frequency data (Fig. 7.8). Two solutions are possible when the Euclidean distance is unsuitable: (1) one may transform the species data using one of the transformations described in Section 7.7 and use the transformed data in the *K*-means analysis; or (2) one may first compute a suitable distance matrix among objects (see Tables 7.4 and 7.5), decompose the distance matrix into eigenvectors by principal coordinate analysis (PCoA, Section 9.3), and run *K*-means partitioning using the table of eigenvectors (principal coordinates).

Following are two numerical examples that illustrate the behaviour of the E_K^2 criterion computed using eqs. 8.5 and 8.6.

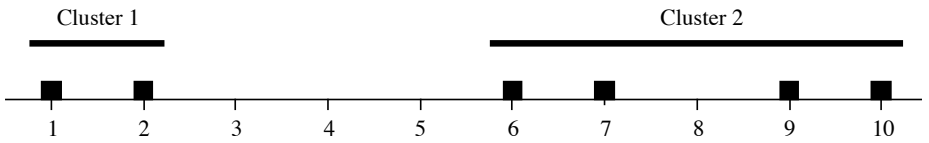
Numerical example 1. For simplicity, consider a single variable. The best partition of the following five objects (dark squares) in two clusters (boldface horizontal lines) is obviously to put objects with values 1, 2 and 3 in one group, and objects with values 6 and 8 in the other:



This example is meant to illustrate that the E_K^2 criterion can be computed from either raw data (eq. 8.5) or distances among objects (eq. 8.6). Using raw data (left-hand column, below), the group centroids are at positions 2 and 7 respectively; deviations from the centroids are calculated for each object, squared, and added within each cluster. Distances among objects (right-hand column, below) are easy to calculate from the object positions along the axis; the numbers of objects (n_k), used in the denominators, are 3 for cluster 1 and 2 for cluster 2.

$$e_1^2 = (1^2 + 0^2 + (-1)^2) = 2 \qquad e_1^2 = (2^2 + 1^2 + 1^2)/3 = 2$$
$$e_2^2 = (1^2 + (-1)^2) = 2 \qquad e_2^2 = 2^2/2 = 2$$
$$E_K^2 = 4 \qquad E_K^2 = 4$$

Numerical example 2. Considering a single variable again, this example examines the effect on the E_K^2 statistic of changing the cluster membership. There are six objects and they are to be partitioned into $K = 2$ clusters. The optimal solution is that represented by the boldface horizontal lines:



Calculations are as above. Using raw data (left-hand column, below), the group centroids are at positions 1.5 and 8 respectively; deviations from the centroids are calculated for each object, squared, and added within each cluster. Distances among objects (right-hand column, below) are easy to calculate from the object positions along the axis; the numbers of objects (n_k), used in the denominators, are 2 for cluster 1 and 4 for cluster 2.

$$e_1^2 = (0.5^2 + (-0.5)^2) = 0.5 \qquad e_1^2 = 1^2/2 = 0.5$$
$$e_2^2 = (2^2 + 1^2 + (-1)^2 + (-2)^2) = 10.0 \qquad e_2^2 = (1^2 + 3^2 + 4^2 + 2^2 + 3^2 + 1^2)/4 = 10.0$$
$$E_K^2 = 10.5 \qquad E_K^2 = 10.5$$

Consider now a sub-optimal solution where the clusters would contain the objects located at positions (1, 2, 6, 7) and (9, 10), respectively. The centroids are now at positions 4 and 9.5 respectively. Results are the following:

$$e_1^2 = (3^2 + 2^2 + (-2)^2 + (-3)^2) = 26.0 \qquad e_1^2 = (1^2 + 5^2 + 6^2 + 4^2 + 5^2 + 1^1)/4 = 26.0$$
$$e_2^2 = (0.5^2 + (-0.5)^2) = 0.5 \qquad e_2^2 = 1^2/2 = 0.5$$
$$E_K^2 = 26.5 \qquad E_K^2 = 26.5$$

This example shows that the E_K^2 criterion quickly increases when the cluster membership departs from the optimum.

In some studies, the number of clusters K to be delineated is determined by the ecological problem, but this it is not often the case. The problem of determining the most appropriate number of clusters has been extensively discussed in the literature. Over 30 different indices, called "stopping rules", have been proposed to do so. Milligan & Cooper (1985) compared them through an extensive series of simulations using artificial data sets with known numbers of clusters; the results of that study are also reported in Milligan (1996). Some of these rules recover the correct number of clusters in most instances, but others are appallingly inefficient. The best of the criteria investigated in that paper is the Calinski-Harabasz index (C-H, Calinski & Harabasz, 1974), which is the multivariate F-statistic (eq. 11.7) of a RDA in which $m = (K - 1)$ dummy variables are used to represent a partition into K groups. When the groups identified by a clustering method are well separated in A-space, the F-statistic becomes large. This statistic cannot be tested for significance, however, because the groups are derived from the same data that would be used for testing.

C-H index

SAS has implemented two among the best rules studied by Milligan: the Calinski-Harabasz F-statistic (called pseudo-F in SAS manuals) and the cubic clustering criterion. Fourteen stopping indices are available in function ***clustIndex()*** of package CCLUST in R. Cross-validation, used in multivariate regression tree analysis (MRT, Section 8.11) to decide about the size of trees, can also be used to determine the optimal number of clusters found in a series of K-means analyses involving different numbers of groups. However, none of these indices correctly identifies the correct solution when a single cluster is present in the data.

8.9 Species clustering: biological associations

Most of the methods discussed in the previous sections may be applied to clustering descriptors as well as objects. When searching for species associations, however, it is important to cluster species using methods that model as precisely as possible a clearly formulated concept of association. The present section (1) attempts to define an operational concept of association and (2) shows how to identify species associations in that framework.

Several concepts of species association have been developed since the nineteenth century; Whittaker (1962) wrote a remarkable review about them. These concepts are not always operational, however. In other words, they cannot always be translated into a series of well-defined analytical steps that would lead to the same result if they were applied by two independent researchers, using the same data. In general, the concept of association refers to a group of species that are "significantly" found together, without this implying necessarily any positive interaction among them. An association, in the statistical sense, is a recurrent group of co-occurring (presence-absence data) or correlated (abundance data) species (Legendre & Legendre, 1978). Associations of taxa belonging to categories other than species may also be defined.

Species association

Several procedures have been proposed for the identification of species associations. Quantitative algorithms have progressively replaced the empirical methods, as they have in other areas of science. All these methods, whether simple or elaborate, have two goals: first, identify the species that occur together and, second, minimize the likelihood that the co-occurrences so identified be fortuitous. The search for valid associations obviously implies that the sampling be random and planned in accordance with the source of variability under study (i.e. geographical, temporal, experimental), which defines the framework within which the groups of species, found repeatedly along the sampling axes, are called associations; one then speaks of association of species over geographic space, or in time, etc. The criterion is the recurrence of a group of species along the study axes.

Species distributions may be correlated through space or time because they have common (or opposite) environmental requirements, or as the result of biotic interactions. These two families of processes produce identifiable spatial or temporal patterns, as described in Subsection 1.1.1. For the first type, positive associations occur when species have the same ecological requirements, and negative associations when their requirements differ. The second type refers to biotic interactions among species, which include predator-prey relationships, competition, and mutualism; it can also lead to positive or negative associations among species. These processes provide grounding theory and hypotheses for the search of species associations, which is one of the classical problems of community ecology (Roxburgh & Chesson, 1998).

Correlation analysis in one form or another, for presence-absence or abundance data, has proven useful to identify species associations (Greig-Smith, 1983; O'Connor & Aarssen, 1987; Myster & Pickett, 1992; Roxburgh & Chesson, 1998). Interspecific associations are recognized when two or more species co-occur (for presence-absence data) either more or less frequently than expected by chance, or when their quantitative variation is correlated. One cannot, however, distinguish between the hypotheses of environmental control and biotic interactions from the results of an association analysis alone (Rejmánek & Leps, 1996). Finer analyses using multiscale correlation methods, e.g. multiscale codependence analysis (Subsection 14.5.2), may help decide between competing hypotheses about to the causes of species associations.

Under the hypothesis of environmental control, when associations have been found, one can concentrate on finding the ecological requirements common to most or all species of an association instead of having to describe the biology and habitat of each species individually. In an inverse approach, species associations may be used to predict environmental characteristics or as indicators of environmental quality (Legendre, 2005). Associations may be better predictors of environmental quality than individual species because they are less subject to sampling error. In certain cases, trophic groups or size classes may also be used for the same purpose.

As mentioned at the beginning of this section, a simple and operational statistical definition is that a species association is *a recurrent group of co-occurring or correlated species*. Using this definition, one can select clustering methods that are

appropriate to delineate species associations. Appropriate measures of resemblance in R mode were described in Chapter 7 (Table 7.6). A great variety of clustering methods have been used for the identification of associations, although the choice of a given method often appears to have been based on the availability of a program on the local computer instead of a good understanding of the properties and limitations of the various techniques. An alternative to standard clustering techniques was proposed by Lambshead & Paterson (1986) who used numerical cladistic methods to delineate species associations. Among the ordination methods, principal component and correspondence analyses may not produce clearly identifiable clusters of species except in the most simple cases (e.g. Fig. 8.20), even though these analyses may be very useful to investigate other multivariate ecological problems (Chapter 9).

After selecting the most appropriate coefficient of dependence for the data at hand (Table 7.6), one must next make a choice among the usual hierarchical clustering methods discussed in the previous sections of this chapter, including TWINSPAN (Subsection 8.7.4). Partitioning by *K*-means (Section 8.8) should also be considered after transformation of the species data (Section 7.7). In addition, there are two specialized methods to delineate species associations described in Subsections 8.9.1 and 8.9.2 below. When the analysis aims at identifying hierarchically-related associations, hierarchical clustering methods are appropriate. When one simply looks for species associations without implying that they should form a hierarchy, partitioning methods are in order. Hierarchical clustering may also be used in that case but one must decide, using a dendrogram or another graphical representation, which level of partition in the hierarchy best corresponds to the associations to be identified. One must take into account the level of detail required and the limits of significance or interpretability of the species clusters found. In any case, space-conserving or space-dilating methods should be preferred to single linkage, especially when one is trying to delimit groups of species from data sampled along an ecological continuum.

The search for species associations is based on the often untested assumption that species have non-random patterns of association, these associations being due to environmental control or biotic interactions. Jackson *et al*. (1992) discussed several null models that may be used to test the non-randomness of species co-occurrence across sites.

Ecological application 8.9a

Thorrington-Smith (1971) identified 237 species of phytoplankton in water samples from the West Indian Ocean. 136 of the species were clustered into associations by single linkage hierarchical clustering of a Jaccard (S_7) association matrix among species. The largest of the 11 associations contained 50 species; its distribution mostly corresponded to the equatorial subsurface water. This association was dominant at all sites and was considered typical of the endemic flora of the West Indian Ocean. Other phytoplankton associations represented seasonal or regional differences, or characterized currents or nutrient-rich regions. Since phytoplankton associations did not lose their identities even when they were mixed, the study of associations in zones of water mixing seemed a good way of tracing back the origins of water masses.

1 — Non-hierarchical complete linkage clustering

Fager's (1957) *non-hierarchical complete linkage clustering* is a specialized partitioning method designed for discovering species associations. It is well-adapted to probabilistic measures of dependence among species and other measures of dependence for which a critical or significance level can be set. This method differs from hierarchical complete linkage clustering in that one looks for clusters formed at a stated threshold of similarity without taking into account the hierarchical cluster structure that may exist at other similarity levels. For probabilistic similarity coefficients, e.g. S_{25} (eq. 7.62), the threshold is usually the significance level $\alpha = 0.05$ or $\alpha = 0.01$, which corresponds to $S \geq 0.95$ or $S \geq 0.99$. With the non-probabilistic similarity coefficient S_{24} (eq. 7.60), Fager & McGowan (1963) used $S \geq 0.5$ as the clustering threshold.

Computer programs that make the method operational have been written, but it is also possible to implement it without a special program. If a similarity coefficient was used to compute the resemblance among the species, select a threshold similarity level and draw a graph (as in Fig. 8.2a) of the species with link edges corresponding to all values of $S \geq$ (threshold). Then, delineate the species associations on the graph as the groups meeting the complete-linkage criterion, i.e. the groups in which all objects are linked to all others at the stated similarity level (Subsection 8.5.2). In case of conflicts, use the following decision rules.

Fager and Krylov rules

1. Complete-linkage clusters of species, obtained by this method, must be independent of one another, i.e. they must have no species in common. Between two possible species partitions, *form first the clusters containing as many species as possible*. For instance, if a cluster of 8 species has two species in common with another cluster of 5 species, create clusters of 8 and 3 species instead of clusters of 6 and 5 species. Krylov (1968) adds that no association should be recognized that contains fewer than three species.

If non-independent clusters remain (i.e. clusters with objects in common), consider rules 2 and 3, in that order.

2. Between several non-independent clusters *containing the same number of species*, choose the partition that maximizes the size of the resulting independent clusters. For example, if there are three clusters of 5 species each where clusters 1 and 2 have one species in common and clusters 2 and 3 also have one species in common, select clusters 1 and 3 with five species each, leaving 3 species into cluster 2. One thus creates three clusters with membership 5, 3, and 5, instead of three clusters with membership 4, 5, and 4.

3a. If the above two criteria do not solve the problem, between two or more non-independent clusters having about the same number of species, select the one *found at the largest number of sites* (Fager, 1957). One has to go back to the original data matrix in order to use this criterion.

3b. Krylov (1968) suggested replacing this last criterion with the following one: among alternative species, the species to include in a cluster is the one *that has the least affinity* with all the other species that are not members of that cluster, i.e. the species that belongs to the cluster more exclusively. This criterion may be decided from the graph of link edges among the species.

This form of non-hierarchical complete linkage clustering led Fager (1957), Fager & McGowan (1963), and Krylov (1968) to identify meaningful and reproducible plankton associations. Venrick (1971) explains an interesting additional step of Fager's computer program; this step answers an important problem of species association studies. After having recognized independent clusters of completely linked species, the program associates the remaining species, by single linkage clustering, to one or several of the main clusters. These *satellite species* do not have to be associated with all members of an association. They may also be satellites of several associations. This reflects adequately the organizational complexity of biological communities.

This last point shows that *overlapping* clustering methods could be applied to the problem of delineating species associations. The mathematical bases of these methods have been established by Jardine & Sibson (1968, 1971) and Day (1977).

Ecological application 8.9b

Fager's non-hierarchical complete linkage clustering was used by Legendre & Beauvais (1978) to identify fish associations in 378 catches from 299 lakes of northwestern Québec. Their computer program provided the list of all possible complete linkage clusters formed at a user-selected similarity level. Species associations were determined using the criteria listed above. The similarity between species was established by means of the probabilistic measure S_{25} (Subsection 7.5.2), based on presence-absence data.

At similarity level $S_{25} \geq 0.989$, the program identified 25 non-independent species clusters, involving 26 of the 29 species in the study. Each subgroup of at least three species could eventually become an association since the clustering method was complete linkage. Many of these clusters overlapped. The application of Fager's decision rules (with rule 3b of Krylov) led to the identification of five fish associations, each one completely formed at the similarity level indicated to the right. Stars indicate the internal strength of the associations (*** all links ≥ 0.999, ** all links ≥ 0.99, * all links ≥ 0.95).

1)	Lake whitefish	*Coregonus clupeaformis*	$S_{25} \geq 0.999$ ***
	Longnose sucker	*Catostomus catostomus*	
	Lake trout	*Salvelinus namaycush*	
	Round whitefish	*Prosopium cylindraceum*	
	Lake chub	*Couesius plumbeus*	
2)	Northern pike	*Esox lucius*	$S_{25} \geq 0.995$ **
	White sucker	*Catostomus commersoni*	
	Walleye	*Stizostedion vitreum*	
	Shallowwater cisco	*Coregonus artedii*	
	Yellow perch	*Perca fluviatilis*	

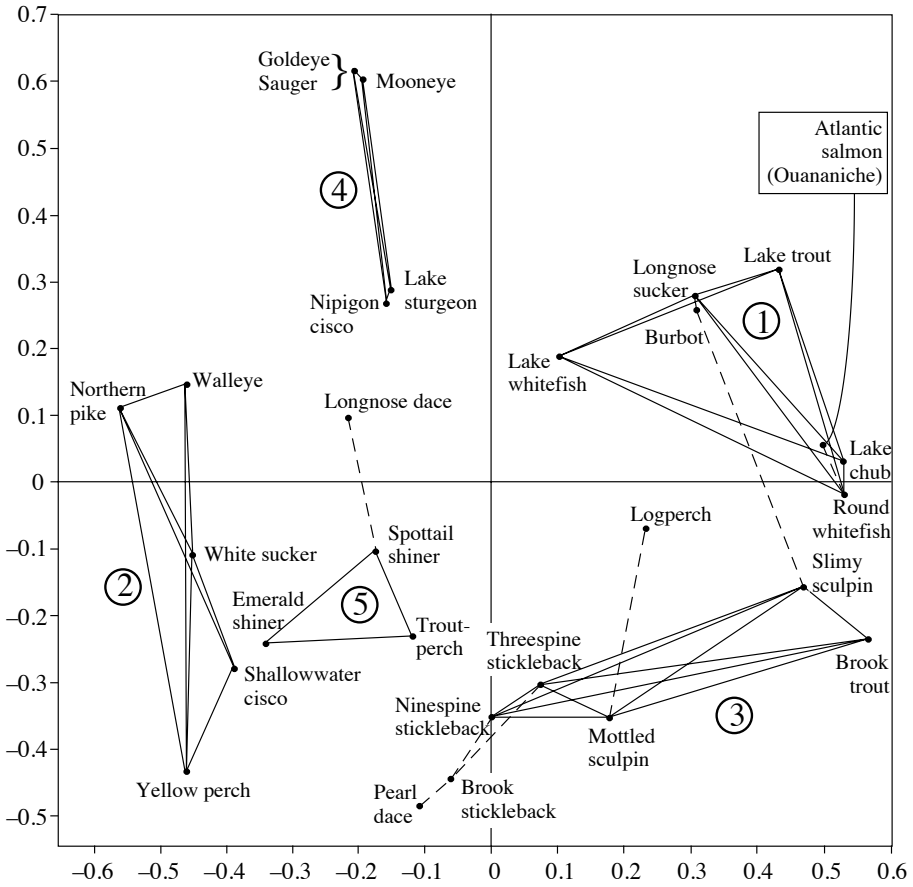


Figure 8.19 Fish associations drawn on a two-dimensional principal coordinate ordination of the species. Axes I (abscissa) and II (ordinate) explain together 55% of the variability among species. Full lines link species that are members of associations identified by non-hierarchical complete linkage clustering at $S \geq 0.989$. Dashed lines attach satellite species to the most closely related species that is a member of an association. The five associations are identified by circled numbers. Redrawn from Legendre & Beauvais (1978).

3)	Brook trout	*Salvelinus fontinalis*	$S_{25} \geq 0.991$ **
	Ninespine stickleback	*Pungitius pungitius*	
	Mottled sculpin	*Cottus bairdi*	
	Threespine stickleback	*Gasterosteus aculeatus*	
	Slimy sculpin	*Cottus cognatus*	

4)	Nipigon cisco	*Coregonus nipigon*	$S_{25} \geq 0.991$ **
	Lake sturgeon	*Acipenser fulvescens*	
	Goldeye	*Hiodon alosoides*	
	Mooneye	*Hiodon tergisus*	
	Sauger	*Stizostedion canadense*	
5)	Trout-perch	*Percopsis omiscomaycus*	$S_{25} \geq 0.989$ *
	Spottail shiner	*Notropis hudsonius*	
	Emerald shiner	*Notropis atherinoides*	

The six remaining species were attached as satellites, by single linkage chaining, to the association containing the closest species. Figure 8.19 shows the species associations drawn on a two-dimensional principal coordinate ordination (Section 9.3) of the species. Three of these associations can be interpreted ecologically. Association 1 was characteristic of the cold, clear, low-conductivity lakes of the Laurentide Shield. Association 2 characterized lakes with warmer and more turbid waters, found in the lowlands. Association 4 contained species that were all at the northern limit of their distributions; they were found in the southern part of the study area.

2 — Concordance analysis

Concordance analysis, which is based upon Kendall's coefficient of concordance (Section 5.4), is useful to delineate groups of species that form statistically significant associations. Described by Legendre (2005), the method proceeds in three steps.

1. Perform a correlation analysis to identify groups of positively correlated species. The most widely used method is to compute Ward's agglomerative clustering (Subsection 8.5.8) of a matrix of correlations among the species. In detail:

1.1. Transform the species abundances using one of the transformations described in Section 7.7. Several transformations may be tried in turn and the results compared.

1.2. Compute a Pearson or Spearman correlation matrix $\mathbf{R} = [r_{hi}]$ among the species. This is done to make the clustering results compatible with concordance analysis, which is based on correlations. Turn matrix **R** into a distance matrix by computing $\mathbf{D} = [1 - r_{hi}]$.

1.3. Carry out Ward's hierarchical clustering of that matrix.

1.4. Cut the dendrogram in two groups and retrieve the vector of species membership.

1.5. After steps 2 and 3 below, one may have to come back and try divisions of the species into 3, 4, 5, ... groups.

In simple cases, a principal component analysis (PCA, Section 9.1) of the standardized transformed species abundance data may be sufficient to delineate species groups on which steps 2 and 3 can be carried out. Because the transformed species data are standardized by columns, the PCA will be computed on the correlation matrix, making the results compatible with concordance analysis, which is based on correlations.

2. Compute global tests of significance of the concordance within the two (or more) groups (Subsection 5.4.2) using the matrix of transformed species abundances (e.g. after Hellinger transformation). Groups that are not globally significant must be refined (step 1.5) or abandoned.

3. Compute *a posteriori* tests of the contribution of individual species to the concordance of their group (Subsection 5.4.3). If the mean of the Spearman correlations of a species with all the other species of its group is negative, this indicates that this species clearly does not belong to the group, hence that group is too inclusive. Go back to step 1.5 and cut the dendrogram more finely. Groups can be refined (i.e. cut into smaller groups) separately from other groups, independently of the levels along the dendrogram.

Use corrections for multiple testing (Box 1.3) at all steps of this analysis. R functions to carry out these tests are described in Section 8.15.

This method should only be applied to species abundance data because Kendall's concordance analysis is meaningless for presence-absence data. Other methods should be developed to deal with presence-absence data. The Kendall concordance approach is useful in environmental studies where researchers are interested in identifying groups of concordant species that are indicators of some property of the environment. In some applications, significantly concordant species can be combined into environmental quality indices (Siegel, 1956), in particular in situations of pollution or contamination, and used to produce indicator maps.

Ecological application 8.9c

Legendre (2005) used the Kendall coefficient of concordance (W) to identify species associations in a multi-species community of oribatid mites (35 species, 70 soil cores[*]; Borcard & Legendre, 1994). The mite data were subjected to the Hellinger transformation (eq. 7.69) at the beginning of the analysis. Ward's agglomerative clustering (Subsection 8.5.8) and K-means partitioning (Section 8.8) both suggested the presence of two groups of mites, one including 24 species and the other 11 (Fig. 8.20). Kendall coefficients of concordance computed over each group separately indicated that both groups had significant concordance. *A posteriori* tests showed that 20 species of the first group and 8 of the second group significantly contributed to the concordance of their group, at the 5% significance level and after Holm correction for multiple testing (Box 1.3). The abundances of the species that were significant members of a group were summed over each group and the sums were plotted on maps of the study area. These indices were related to environmental variables by multiple regression (Section 10.3), which produced highly significant environmental models.

The PCA ordination diagram (Section 9.1) shown in Fig. 8.20 was computed after standardizing the Hellinger-transformed species vectors (eq. 1.12); the two mite associations identified by clustering followed by concordance analysis are represented by symbols on the plot. An alternative graphical presentation of the clustering results would be a heat map (Section 8.10) with dendrograms added to the sides; R functions to produce heat maps are listed in Section 8.15. When environmental descriptors are available, computing an RDA (Section 11.1) instead of a PCA will produce a plot providing an interpretation of the differences among the groups of species. Another example (fish associations) is presented in Section 4.10.2 of Borcard *et al*. (2011).

[*] The mite data are available on the Web page of the Borcard *et al*. (2011) book, http://numericalecology.com/NEwR.

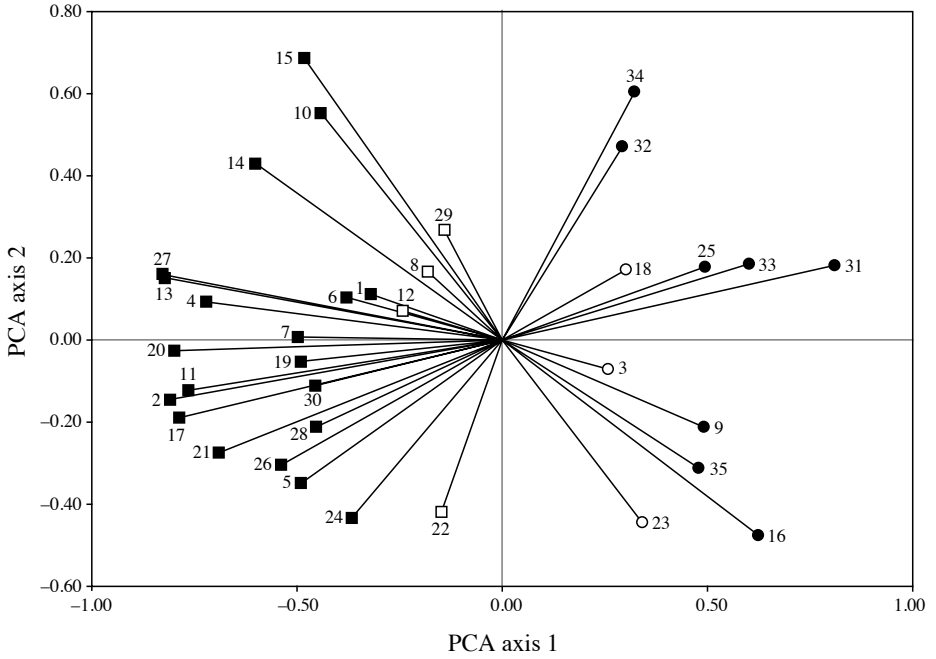


Figure 8.20 Principal component ordination diagram showing the species vectors projected in the space formed by PCA axes 1 (28.8% of the variation) and 2 (9.2%). Mite group 1 resulting from the preliminary Ward clustering is represented by squares and group 2 by circles. Solid symbols: species that are significant members of their respective associations. Modified from Legendre (2005, Fig. 4).

3 — Indicator species

The identification of indicator (or characteristic) species is a traditional question in ecology and biogeography. Field studies describing sites or habitats usually mention one or several species that characterize each habitat. For many years, the most widely used statistical method for identifying indicator species was TWINSPAN (Hill, 1979a; Subsection 8.7.4). There is clearly a need for the identification of characteristic or indicator species in the fields of monitoring, conservation, and management, as discussed below. Because indicator species add ecological meaning to groups of sites discovered by clustering, they provide criteria to compare typologies derived from data analysis, to identify where to stop dividing clusters into subsets, and to point out the main levels in a hierarchical classification of sites. *Indicator species* differ from *species associations* in that they are indicative of particular groups of sites. Good indicator species should be found mostly in a single group of a typology and be present

at most of the sites belonging to that group. This duality is of ecological interest; yet it is seldom exploited in indicator species studies.

Dufrêne & Legendre (1997) proposed an alternative to TWINSPAN in the search for indicator species and species assemblages characterizing groups of sites. Like TWINSPAN, their method is *asymmetric*, meaning that species are analysed on the basis of a *prior* partition of the sites. The first original characteristic of the method is that it derives indicator species from any hierarchical or non-hierarchical classification of the objects (sampling sites), contrary to TWINSPAN where indicator species can only be derived for classifications obtained by splitting sites along correspondence analysis (CA) or detrended correspondence analysis (DCA) axes (Subsection 8.7.4). The second original characteristic lies in the way the indicator value of a species is measured for a group of sites. The *indicator value index* (*INDVAL*) is based only on within-species abundance and occurrence comparisons; its value is not affected by the abundances of other species. The significance of the indicator value of each species is assessed by a randomization procedure (Section 1.2).*

The *indicator value* (*INDVAL*) index is defined as follows. For each species j in each cluster of sites k, one computes the product of two values, A_{kj} and B_{kj}. A_{kj} is a measure of *specificity* based on abundance values whereas B_{kj} is a measure of *fidelity* computed from presence data:

Specificity $$A_{kj} = Nindividuals_{kj} / Nindividuals_{+k}$$

Fidelity $$B_{kj} = Nsites_{kj} / Nsites_{k+}$$

$$INDVAL_{kj} = A_{kj} B_{kj} \qquad \textbf{(8.21)}$$

• In the formula for specificity (A_{kj}), $Nindividuals_{kj}$ is the mean abundance of species j across the sites pertaining to cluster k and $Nindividuals_{+k}$ is the sum of the mean abundances of species j within the various clusters. The *mean* number of individuals in each cluster is used, instead of summing the individuals across all sites of a cluster, because this removes any effect of variations in the number of sites belonging to the various clusters. Differences in abundance among sites of a cluster are not taken into account in the calculation. A_{kj} is maximum when species j is present in cluster k only.

• In the formula for fidelity (B_{kj}), $Nsites_{kj}$ is the number of sites in cluster k where species j is present and $Nsites_{k+}$ is the total number of sites in that cluster. B_{kj} is maximum when species j is present at all sites of cluster k.

• Quantities A and B must be combined by multiplication because they represent independent information (i.e. specificity and fidelity) about the distribution of species j.

* How to compute *INDVAL* in R is described in Section 8.15. The *INDVAL* index is also available in package PC-ORD; distribution: see footnote in Section 11.7.

In De Cáceres & Legendre (2009), *specificity* is called *positive predictive value* and *fidelity* is called *sensitivity*.

The indicator value of species j for a partition of sites is the largest value of $INDVAL_{kj}$ observed over all clusters k of that partition:

Indicator value

$$INDVAL_j = \max [INDVAL_{kj}] \quad \textbf{(8.22)}$$

The index is maximum (its value is 1) when the individuals of species j are observed at all sites belonging to a single cluster. A random permutation procedure of the sites among the site groups is used to test the significance of $INDVAL_j$ (Section 1.2). A correction for multiple testing (Box 1.3) is in order before reporting the results since multiple tests (i.e. for p species) are conducted. The index can be computed for any given partition of sites, or for all levels of a hierarchical classification of sites.

Numerical example. Table 8.10 describes the example given by Dufrêne & Legendre (1997) to illustrate the computation of the *INDVAL* index, slightly modified. The data represent three species observed at 25 sites, which are divided into 5 groups. To facilitate comparisons, the sums of the mean group abundances are 20 for all three species. For species 1, $INDVAL_{k1}$ has the highest value (0.30) for group $k = 3$, so $INDVAL_1 = 0.30$. Following similar reasoning, $INDVAL_2 = 0.40$ and $INDVAL_3 = 0.90$. The permutational p-values computed by functions ***indval()*** of LABDSV or ***multipatt()*** of INDICSPECIES in R are significant in all three cases.

De Cáceres & Legendre (2009) described several other statistics that can be used to measure the indicator value of species. They are divided into *correlation indices*, which are used for determining the ecological preferences of species among a set of alternative site groups or site group combinations, and *indicator value indices*, including *INDVAL*, which are used for assessing the predictive values of species as indicators of the conditions prevailing in site groups, e.g. for field determination of community types or ecological monitoring. Each of these categories of indices comes in different types: there are indices for *presence-absence* and for *quantitative* species data; there are also *non-equalized indices* that give equal weights to individual sites and *group-equalized indices* that give equal weights to all groups whatever the number of sites they contain. For studies involving several groups of sites, De Cáceres *et al*. (2010) showed how to improve the interpretation of indicator value analysis by computing the statistics for all possible combinations of site groups. An application of that method is found in Moretti *et al*. (2010).

De Cáceres *et al*. (2010) present a detailed discussion of the limitations of indicator value analysis. In particular, they point out that more indicator species will be found than expected by chance when the classification of sites has been obtained from the same species composition data that are used for *INDVAL* analysis. In that case, p-values must be interpreted with caution: they are not the result of a genuine test of significance, where the classification of sites has to be independent of the species data used in the test.

Table 8.10 Numerical example: abundance of three species at 25 sites divided into 5 groups. Modified from Dufrêne & Legendre (1997). Top panel: data. Bottom panel: calculation of the specificity (A_{kj}), fidelity (B_{kj}) and $INDVAL_{kj}$ index for each species (j) in each group of sites (k). The maximum value of $INDVAL_{kj}$ for each species is in bold.

Groups	Group 1					Group 2					Group 3					Group 4					Group 5				
Sites	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Species 1	4	4	4	4	4	5	5	5	5	5	6	6	6	6	6	3	3	3	3	3	2	2	2	2	2
Species 2	8	8	8	8	8	4	4	4	4	4	6	6	6	6	6	4	4	2	0	0	0	0	0	0	0
Species 3	18	18	18	18	18	2	2	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	Group 1	Group 2	Group 3	Group 4	Group 5
Species 1					
A_{k1}	4/20 = 0.20	5/20 = 0.25	6/20 = 0.30	3/20 = 0.15	2/20 = 0.10
B_{k1}	5/5 = 1	5/5 = 1	5/5 = 1	5/5 = 1	5/5 = 1
$INDVAL_{k1}$	0.20	0.25	**0.30**	0.15	0.10
Species 2					
A_{k2}	8/20 = 0.40	4/20 = 0.20	6/20 = 0.30	2/20 = 0.10	0/20 = 0.00
B_{k2}	5/5 = 1	5/5 = 1	5/5 = 1	3/5 = 0.6	0/5 = 0
$INDVAL_{k2}$	**0.40**	0.20	0.30	0.06	0.00
Species 3					
A_{k3}	18/20 = 0.90	2/20 = 0.10	0/20 = 0.00	0/20 = 0.00	0/20 = 0.00
B_{k3}	5/5 = 1	5/5 = 1	0/5 = 0	0/5 = 0	0/5 = 0
$INDVAL_{k3}$	**0.90**	0.10	0.00	0.00	0.00

Podani & Csányi (2010) proposed variants of the *INDVAL* index. Instead of using specificity and fidelity alone, they proposed to define the indicator value of a species as the product of two among three quantities: specificity A_{kj} (that they renamed concentration), specificity (new equation, with allowance for positive or negative species preferences), and fidelity B_{kj}. They provided formulas based on either presence-absence or abundance data for each of these three quantities.

McGeoch & Chown (1998) found the indicator value method important to conservation biology because it is conceptually straightforward and allows researchers to identify bioindicators for any combination of habitat types or areas of interest, e.g. existing conservation areas, or groups of sites based on the outcome of a classification procedure. In addition, it may be used to identify bioindicators for groups of sites classified using the target taxa, as in Ecological application 8.9d, or using non-target taxa, e.g. insect bioindicators of plant community classifications.

Because each *INDVAL* index is calculated independently of other species in the assemblage, comparisons of indicator values can be made between taxonomically unrelated taxa, taxa in different functional groups, or those in different communities. Comparisons across taxa are robust to differences in abundance that may or may not be due to differences in detectability or visibility, or to sampling methods. The method is also robust to differences in the numbers of sites between site groups, to differences in abundance among sites within a particular group, and to differences in the absolute abundances of very different taxa that may exhibit similar trends.

When a group of sites for which indicator species are sought corresponds to a delimited geographic area, superposition of the distribution maps for the indicator species of that group should help delineate the core conservation areas for these species, even when little other biological information is available. McGeoch & Chown (1998) also consider the *indicator measure of a species absence* to be of value. The species absence *IndVal* provides a method for improving the objectivity with which species transient to an assemblage can be identified. Species with high values for this absence index may also be of ecological interest as indicators of peculiar ecological conditions where the species is seldom or never present.

Taxa proposed as bioindicators in the literature are often merely the favourite taxa of their proponents; ornithologists prefer birds, lepidopterists butterflies, and coleopterists beetles. According to McGeoch & Chown (1998), *IndVal* provides an objective method for addressing this problem by enabling the assessment of the relative merits of different taxa as bioindicators for a given study area. The species that do emerge from this procedure as the most useful indicators for a group of sites should prove useful in practical conservation for monitoring site change and disturbance. Two groups of species collected at the same sites can be compared by co-inertia analysis (CoIA, Section 11.5, see Ecological application 11.5) and several groups by multiple factor analysis (MFA, Section 11.5).

Ecological application 8.9d

In order to illustrate the indicator value method, Dufrêne & Legendre (1997) used a large data set of Carabid beetle distributions in open habitats of Belgium (189 species collected in pitfall traps, for a total of 39984 specimens). The data represented 123 year-catch cycles at 69 locations; a year-catch cycle cumulates catches at a site during a full year; 54 sites were studied during two years and 15 sites were sampled during a single year. The typology of sites was computed by distance-based *K*-means partitioning computed as follows: first, a distance matrix

(percentage difference D_{14}, eq. 7.58) was computed from the log-transformed species abundance data; this distance matrix was subjected to principal coordinate analysis (PCoA, Section 9.3); all principal coordinates were then used as input data into *K*-means partitioning. Although the clusters produced by *K*-means had not been forced to be hierarchically nested, they showed a strong hierarchical structure for $K = 2$ to 10 groups. This allowed the authors to represent the relationships among partitions as a dendrogram. The $K = 10$ level corresponded to the main types of habitat, recognized *a priori*, where sampling had been conducted.

Indicator values were computed for each species and partitioning level. Some species were found to be stenotopic (narrow niches) while others were eurytopic (species with wide niches, present in a variety of habitats). Others characterized intermediate levels of the hierarchy. The best indicator species ($INDVAL > 0.25$) were assembled into a two-way indicator table; this tabular representation displayed the hierarchical relationships among species.

Results of the indicator value method were compared to TWINSPAN. Note that the partitions of sites used in the two methods were not the same; the TWINSPAN typology was obtained by partitioning correspondence analysis ordination axes (Subsection 8.7.4). TWINSPAN identified, as indicators, pseudospecies pertaining to very low cut-off levels. These species were not particularly useful for prediction because they were simply known to be present at all sites of a group. Several species identified by TWINSPAN as indicators also received a high indicator value from the *INDVAL* procedure, for the same or a closely related habitat class. The *INDVAL* method identified several other indicator species, with rather high indicator values, that also contributed to the specificity of the groups of sites but had been missed by TWINSPAN. So, the *INDVAL* method appeared to be more sensitive than TWINSPAN to the fidelity and specificity of species.

Here are some more examples of the many applications of indicator species analysis found in the literature. Borcard (1996) and Borcard & Vaucher-von Ballmoos (1997) present applications of the indicator value method to the identification of the Oribatid mite species that characterize well-defined zones in a peat bog of the Swiss Jura. The indicator values of beetle species characterizing different types of forests have been studied by Barbalat & Borcard (1997). Tuomisto *et al*. (2003) used constrained clustering (Subsection 12.6.4) to group 86 sampling units, each 500 m long, forming a 43-km long transect in the Amazonian rain forest into spatial clusters, on the basis of satellite image pixel values. They also surveyed in the field the ferns and *Melastomaceae* observed in the 86 sampling units. Then they used the *INDVAL* method to determine the species that were good indicators of the spatial clusters.

Legendre *et al*. (2009) used multivariate regression tree analysis (MRT, Section 8.11) to identify habitat types that were similar in topographic conditions and in species composition in a Chinese permanent forest plot divided in 20×20 m quadrats; then they used the *IndVal* method to identify the nine, among 159 tree species, that were statistically significant indicators of the five main habitat types. De Cáceres *et al*. (2010) carried out indicator species analysis of the vegetation of the Barro Colorado Island (BCI) permanent forest plot in Panama, also divided in 20×20 m quadrats, grouped into seven habitat types identified in a previous paper. Among 307 tree species, they identified 44 indicator species of individual habitats and 64 for habitat combinations. In the first of these papers, the species used for *IndVal* analysis had been used to obtain the classification of the sites, so that the p-values had to be

interpreted with caution. In the second paper, the classification of the sites was independent of the species analysed for indicator value.

8.10 Seriation

Before clustering methods were developed, the structure of an ecological resemblance matrix was often studied by *matrix rearrangement* (Orlóci, 1978). In this approach, the order of the objects is modified in such a way as to concentrate the lowest distances (or the highest similarities) near the main diagonal of the resemblance matrix. This is a special case of an approach called *seriation* in archaeology, where the rows and columns of a *rectangular* matrix of (artefacts × descriptors) are rearranged in such a way as to bring the highest values near the main diagonal, in order to evidence the temporal seriation of the artefacts; see Kendall (1988) for a review. This technique was developed by anthropologists Petrie (1899) and Czekanowski (1909) and was first applied to ecological data by Kulczynski (1928). An interesting aspect of seriation for ecologists, nowadays, lies in the fact that the technique can be applied to the special case of similarity or distance matrices that are not symmetric, as explained below.

The statistical theory of seriation is now well developed. Papers and syntheses are found mostly in the archaeological literature, e.g. Renfrew & Bahn (2008). Hahsler *et al*. (2008) describe different seriation methods that can be used to visualize (objects × descriptors) or distance (**D**) matrices, reordered or not according to clustering results, and cite the relevant literature. These methods can only be used with small data sets. That paper is also an introduction to the R package SERIATION (Section 8.15).

Trellis diagram
Heat map

A rearranged resemblance matrix can be represented by a *trellis diagram*, called a *heat map* in recent software, which is a shaded or colour-coded matrix. Figure 8.21a gives an example where half of the matrix is represented by shades of gray corresponding to distance values, and Fig. 8.21c shows a heat map of the same distance matrix computed by an R function. Heat maps provide an interesting representation of a raw data or distance matrix, before or after clustering, when the number of objects is small, e.g. 30 or less. In R (Section 8.15), heat maps without or with dendrograms can be produced by functions ***heatmap()*** of package STATS and ***hmap()*** of SERIATION. Function ***coldiss()*** (Section 8.15) plots side by side an original and reordered **D** matrix without dendrogram. Examples are given in Borcard *et al*. (2011, Subsections 3.3.2 and 4.7.3.7).

Seriation works best when there is a single gradient in the data. For symmetric matrices, the order of the objects in any agglomerative clustering dendrogram can be used as the seriation order. A minimum spanning tree (Section 8.2) computed for the **D** matrix (Fig. 8.21b) provides details about the structure and shows if the single-gradient assumption holds for at least part of the objects ordered in a trellis diagram. At the end of the seriation procedure (Fig. 8.21a), the lowest distances, which are now found close to the diagonal, indicate the first important clusters of objects. The first

axis of an ordination diagram (Chapter 9) provides another optimal order of objects, which can be used in a trellis diagram.

Non-symmetric matrix

Seriation is an interesting approach for the analysis of non-symmetric distance matrices. Non-symmetric matrices, in which $D(\mathbf{x}_1, \mathbf{x}_2) \neq D(\mathbf{x}_2, \mathbf{x}_1)$, are rare in ecology. They may, however, be encountered in cases where the resemblance is a direct measure of the influence of an organism on another, or in behavioural studies where the attraction of an organism for another can be used as a similarity measure. They are also common in taxonomic and phylogenetic analysis (e.g. serological data, DNA pairing data).

An analytical solution to seriation that can be applied to non-symmetric as well as symmetric matrices was proposed by Beum & Brundage (1950). The algorithm starts with a similarity matrix (**S**) among objects, provided in any order. The diagonal values are excluded from the calculation, so that the "similarities" in the matrix can be any quantitative indications of preference, not necessarily with a maximum value of 1. In each column j, the products of the elements s_{ij} by the *inverse order numbers of the rows* are summed and divided by the sum of the elements in column j. These average weights are used to determine the new order of the rows and columns, from which the procedure starts over again until convergence is reached. The algorithm may at times end up alternating between two equally optimal final solutions. An R function is available to carry out the Beum-Brundage seriation procedure; see Section 8.15.

Besides seriation, non-symmetric matrices can be decomposed into symmetric and skew-symmetric components, as described in Subsection 2.3, before analysis by clustering and/or ordination methods.

Ecological application 8.10a

Kulczynski (1928) studied the phytosociology of a region in the Carpathian Mountains, southeastern Poland. He recognized 37 plant associations, listed the species found in each, and computed a similarity matrix among them. Part of that similarity matrix, turned into a **D** matrix, is reproduced in Fig. 8.21a. The order of the associations shown in the figure is the one that Kulczynski found when he performed seriation by hand. He interpreted that order as representing a series from association 22 (*Varietum pinetosum czorsztynense*) to association 13 (*Seslerietum variae normale*). The blocs of higher (darker) values near the diagonal allow one to recognize two main groups: associations (22, 21) and (11, 15, 14, 17, 18). Association 13 and 16 seem less related with the others. A minimum spanning tree computed for the same **D** matrix (Fig. 8.21b) provides more detail about the structure of the data. The corresponding heat map is shown in Fig. 8.21c. The dendrogram shown along both axes was obtained by complete linkage clustering of matrix **D**. *Note*: seriation produces a clear one-dimensional ordination when there is a single gradient in the data, which is not the case here.

Ecological application 8.10b

Wieser (1960) studied the meiofauna (small benthic metazoans) at three sites (6 or 7 cores per site) in Buzzards Bay, Massachusetts, USA. After representing the resemblance among cores as

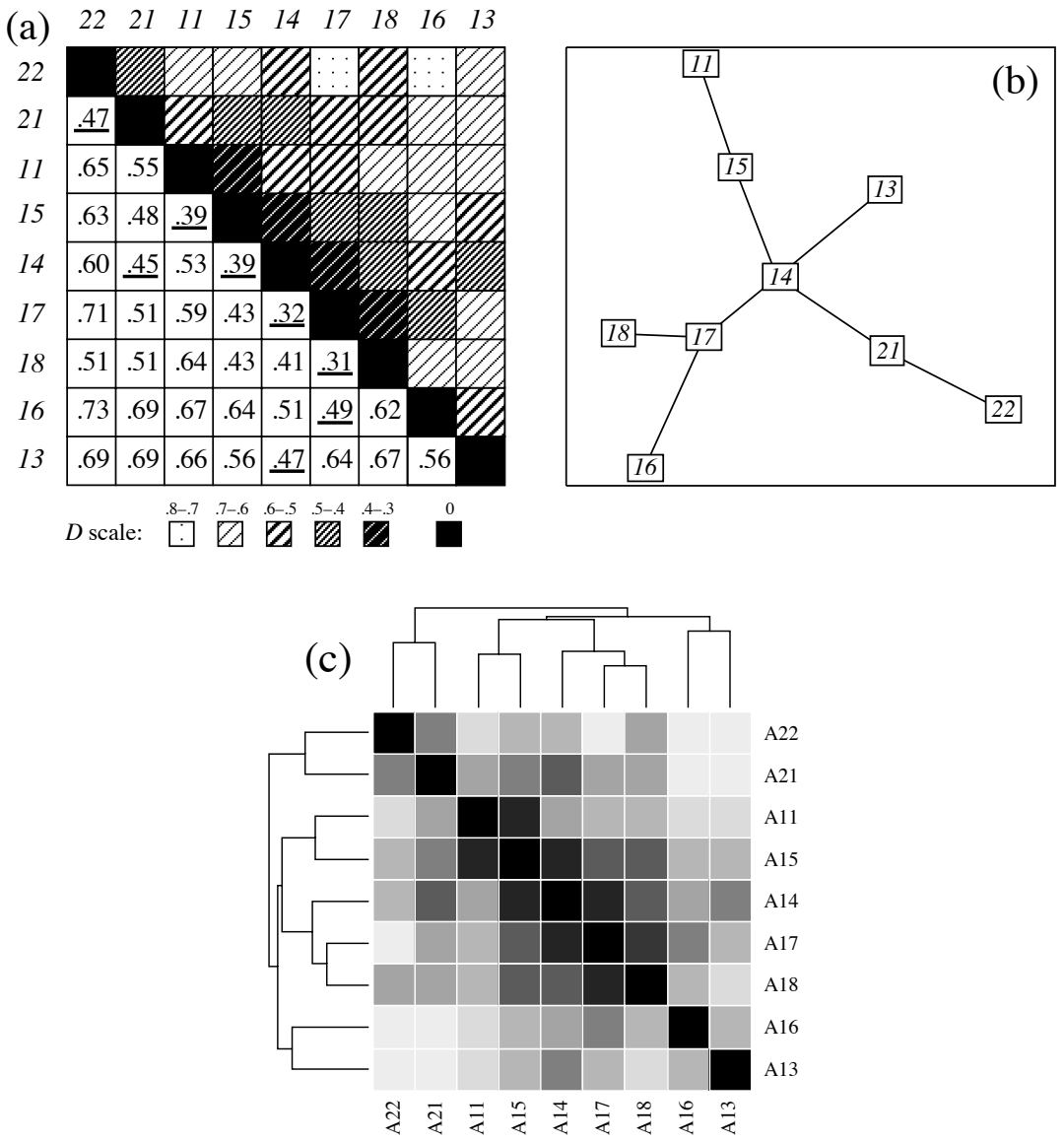


Figure 8.21 (a) Distance matrix (lower half) and trellis diagram (upper half) for part of Kulczynski's (1928) plant associations of the Carpathian Mountains. Numbers in italics, in the margins, identify the associations. In the trellis diagram, the distances are represented by shadings, as indicated underneath the matrix. (b) Minimum spanning tree computed for the same **D** matrix with function ***spantree()*** of the VEGAN R package. The edges of the tree correspond to the underscored distance values in (a). (c) Heat map of the distance matrix with dendrograms shown along both axes. The picture produced by function ***heatmap()*** was in colour.

a similarity matrix (using Whittaker's index of association, $1 - D_9$) and a trellis diagram, he found that although the three sites differed in species composition, the two sandy sites were more similar to each other than they resembled the third site where the sediment contained high concentrations of fine deposits.

The classical study reported in Ecological application 8.10b encouraged other applications of trellis diagrams in benthic ecology. Among these is Sanders' (1960) representation of an ecological time series, also from Buzzards Bay, using a trellis diagram. Inspired by these applications to benthic ecology, Guille (1970) and Soyer (1970) used the method of trellis diagrams to delineate benthic communities (macrofauna and harpacticoid copepods, respectively) along the French Catalonian coast of the Mediterranean Sea near Banyuls-sur-Mer.

Wieser's (1960) study offers an opportunity to come back to the warning of Section 8.0, that not all problems of data analysis belong to the clustering approach. Nowadays, one would not have to seriate or cluster the sites before comparing the species to the sediment data. One would directly compare the species abundance to the sediment data, or to a factor representing the three study sites, using canonical analysis (Chapter 11).

8.11 Multivariate regression trees (MRT)

Univariate *classification tree* analysis (CT) refers to situations where a qualitative response variable is to be predicted by a decision tree (defined below), whereas in *regression tree* analysis (RT) the response variable is quantitative. *Classification and regression tree* analysis (CART, Breiman *et al.,* 1984) combines these two procedures. A decision tree is a forecasting or predictive tree-like diagram resulting from recursive partitioning of the response data, with indication of the influence of the explanatory variables at each split; examples are given below. These univariate forms of analysis are not discussed further in the present chapter.

Proposed by De'ath in 2002 and Larsen & Speckman in 2004, *multivariate regression tree* analysis (MRT) is an extension of CART to multivariate response data. The method could have been presented in Chapter 11 devoted to canonical analysis since, like RDA and CCA, it involves a response and an explanatory data set. It is presented here instead because its output is a tree.

Figure 8.22a shows a simple example with a multivariate response data set **Y** on the left and a matrix of explanatory variables on the right. There are three explanatory variables in **X**; $\mathbf{x}_1$ and $\mathbf{x}_2$ are quantitative in this example and $\mathbf{x}_3$ is qualitative (three levels or states: A, B and C). For the first split, the analysis will search for the best partition of **Y** in two groups, constrained by each variable **x** in turn.

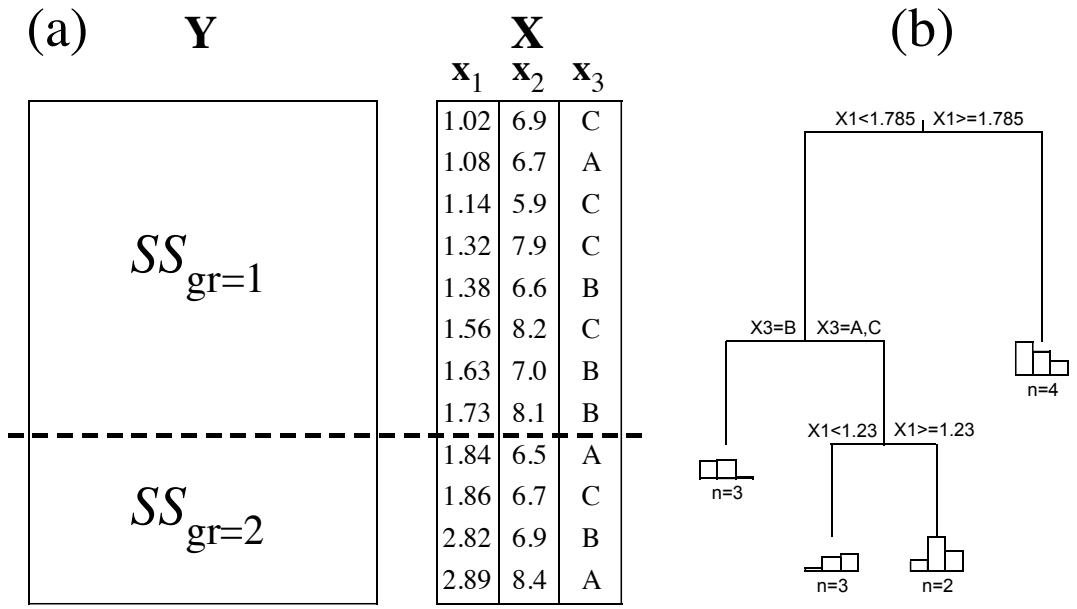


Figure 8.22 Schematic description of MRT analysis. (a) Data: **Y** is the response data set. There are three explanatory variables in **X**: $\mathbf{x}_1$ and $\mathbf{x}_2$ are quantitative in this example and $\mathbf{x}_3$ is qualitative (three factor levels or qualitative states). The dashed horizontal line indicates a cut-point along the values of $\mathbf{x}_1$. The line is extended across **Y**, which is thus divided into two groups. (b) Multivariate regression tree computed by function ***mvpart()*** of the MVPART package; there were 3 species in **Y** (not shown) in this analysis. Variable $\mathbf{x}_1$ controls the first split (the split occurs at the position of the dashed line in panel a); variable $\mathbf{x}_3$ controls the second split (objects with level 1 on the left, those with levels 2 and 3 on the right); variable $\mathbf{x}_1$ is used again for the third split. The number of objects in each group is shown underneath each leaf (terminal group) of the tree, together with a histogram showing the relative abundances of the three species in **Y**.

• For variable $\mathbf{x}_1$, imagine that the rows of the two data sets, **Y** and **X**, are ordered by increasing $\mathbf{x}_1$ values, as in Fig. 8.22a (the actual programming may differ from the description that follows). The program tries in turn all possible cut-points of variable $\mathbf{x}_1$. For each cut-point between successive but different values of $\mathbf{x}_1$, imagine a line drawn across **Y** (dashed line in Fig. 8.22a); it divides **Y** in two groups. $SS_{gr=1}$ is the sum of within-group sums-of squares (also called squared error) for the top group (gr=1), computed using eq. 8.5, and $SS_{gr=2}$ is the sum of within-group sums-of squares for the bottom group (gr=2). So the total within-group sum-of-squares, or total error, for that split of the objects is $E^2 = SS_{gr=1} + SS_{gr=2}$ (eq. 8.7). Because of the equivalence of eqs. 8.5 and 8.6 for the computation of squared error, one can compute MRT from a raw data file **Y** or from a distance matrix **D** computed from **Y**.

• The function tries in turn all possible cut-points along $\mathbf{x}_1$, making no cut between identical (tied) values, and computes $E^2_{\mathbf{x}1}$. It notes the position of the cut where E^2 is minimum for variable $\mathbf{x}_1$ and the value of $E^2_{\mathbf{x}1}$ at that point.

• The process is repeated for variable $\mathbf{x}_2$: the rows of the two data matrices are reordered in such a way that the values of $\mathbf{x}_2$ are in increasing order, all possible cut-points between non-identical values are tried in turn, and the cut that produces the smallest value of $E^2_{\mathbf{x}2}$ is noted.

• The third variable in Fig. 8.22a is a qualitative variable or ANOVA factor. All possible combinations of factor levels are tried in turn. In this example, only three solutions need to be studied: the group defined by state A *versus* the other objects, the group defined by state B, and finally the group defined by state C. The combination that produces the smallest value of $E^2_{\mathbf{x}3}$ is noted. (In the example, the second split separated the rows with level B from those with levels A and C.)

• All values of $E^2_{\mathbf{x}}$ (three in this illustration) are compared: min($E^2_{\mathbf{x}1}$), min($E^2_{\mathbf{x}2}$), and min($E^2_{\mathbf{x}3}$). The smallest of these values is used to draw the first split of the regression tree (Fig. 8.22b, top), which is the first split of data set **Y**.

• Each branch of the tree is then analysed separately (a branch is a group formed by a split). The search for a meaningful split is first carried out for the left branch of the tree. All explanatory variables in **X** are tried in turn and the variable that produces the split with the smallest value of $E^2_{\mathbf{x}}$ is used for the next split of the left-hand side of the tree. Similarly, the search is carried out for the objects in the right branch of the tree and the variable of **X** that produces the split with the smallest value of $E^2_{\mathbf{x}}$ is used for the next split of the right-hand side of the tree. Any variable may be used for several splits. Figure 8.22b shows a tree that was produced for a data set **Y** with 3 species (data not shown).

The process could go on until the tree is fully resolved and individual objects form the terminal groups (leaves of the tree). Users, however, are usually not interested in the fully resolved tree, but instead in a tree that has informative partitions. That shorter tree is found by pruning the tree, an operation that consists in removing the smallest branches. The optimal size of the tree is decided by a resampling analysis called cross-validation. In that analysis, the data are randomly divided into a number of approximately equal-sized *test groups*, e.g. 10% of the objects. Each test group is left aside in turn while the tree is reconstructed using the remaining objects, e.g. 90%. Then, distances are computed from each object of the test group to the multivariate centroids of the groups forming the *leaves* (terminal groups of objects) of the tree. The objects of the test group are attributed to the closest leaf of the reconstructed tree.

Leaf

The objects in the test group are attributed to the closest leaf of a tree with 2 groups, 3 groups, etc., considering the distances of the objects to the centroids of the groups. An overall relative error statistic (cross-validation relative error, *CVRE*) is

computed as follows for each partition size using all n objects (that is done by using the predictions made for the members of all test groups in the formula):

Cross-validation error

$$CVRE = \frac{\sum_{i=1}^{n}\sum_{j=1}^{p}\left(y_{ij(k)} - \hat{y}_{j(k)}\right)^2}{\sum_{i=1}^{n}\sum_{j=1}^{p}\left(y_{ij} - \bar{y}_j\right)^2} \quad \textbf{(8.23)}$$

where $y_{ij(k)}$ is the value of variable j for object i belonging to test group k, $\hat{y}_{j(k)}$ is the value of that same variable at the centroid of the leaf that is closest to object i, whereas the denominator is the overall sum of squares of the **Y** data. *CVRE* is then the ratio of the variation unexplained by the tree to the total variation in **Y**. *CVRE* varies from zero for a perfect set of predictors chosen for the splits of a tree, to close to one for poor predictors; its value can actually exceed 1.

Cross-validation is repeated a number of times, e.g. 100 times, for successive and independent divisions of the objects into random test groups. Then, for each partition size (number of groups), the mean and standard error of all *CVRE* estimates is computed. The cross-validation procedure is described in more details by Borcard *et al*. (2011, Section 4.11) and Ouellette *et al*. (2012).

Should one retain a tree with a single split (2 groups), 2 splits (3 groups), or more splits? *CVRE* is used to indicate the optimal size of the tree. One can select the tree that has the smallest *CVRE* value; alternatively, and following Breiman *et al*. (1984), one may prefer a more parsimonious solution (i.e. a tree with fewer splits) whose *CVRE* value is within one standard error of the smallest *CVRE* value. In any case, *CVRE* is simply a criterion that helps researchers select the optimal tree; in the end, one can opt for a tree with fewer or more leaves (groups) than proposed by the *CVRE* criterion.

MRT belongs to the family of Euclidean methods because it is based on sums of squared deviations from means, just like ANOVA and *K*-means partitioning. The appropriateness of MRT analysis for the analysis of species data tables containing many zeros may be enhanced by transforming them following Section 7.7; this could improve the interpretability and usefulness of the trees as explanatory models of community response data.

Cascade multivariate regression tree analysis (CascadeMRT) is an extension of MRT developed by Ouellette *et al*. (2012). Users can assess their explanatory hypotheses in a hierarchical (nested) manner, carrying out MRT analyses using explanatory data matrices in the order corresponding to the hierarchy of their hypotheses. The nested hypotheses may, for example, correspond to processes operating at different spatial or temporal scales. An R package implementing cascade MRT is described in Section 8.15.

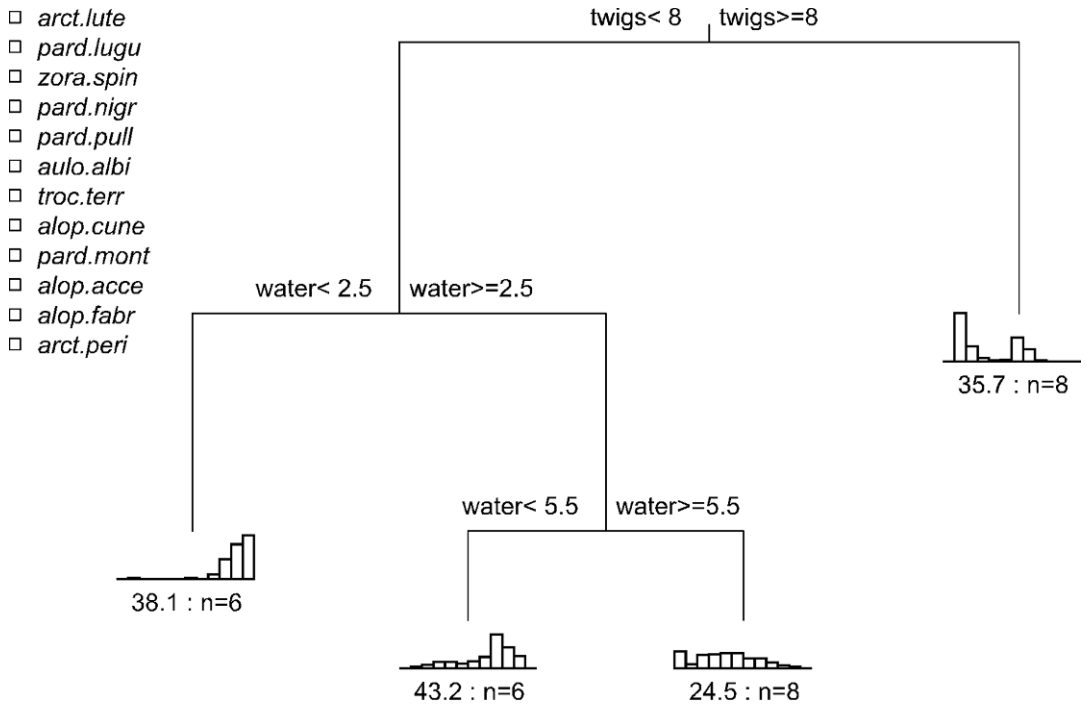


Figure 8.23 Multivariate regression tree for the hunting spider data analysed by De'ath (2002). The relative abundances of the 12 species are shown in histograms positioned at the tips of the branches, with the species in the same order as in the **Y** input file; the species names are shown in the upper-left portion of the plot as they appear in the **Y** data file. The squares in the species list and bars in the histograms have colours in the R-produced ***mvpart()*** plot. Under each histogram, *n* is the number of sites in the leaf (group); the value before *n* is the sum of squared errors for the group (eq. 8.5).

Ecological application 8.11

De'ath (2002) reanalysed the hunting spider data of Aart & Smeenk-Enserink (1975), using the spider and environmental data transformed and recoded by ter Braak (1986, Table 3); ter Braak had used these data to illustrate canonical correspondence analysis in his seminal paper. The recoded data are available in a data file of package MVPART (De'ath, 2011): 28 sites, 12 species and 6 environmental variables (water, sand, moss, light reflection, twigs, and herbs, transformed into classes from 0 to 9). Following De'ath (2002), the species data were transformed by dividing each abundance value by its column mean, then by the row mean recomputed on the resulting file. The size of the tree was selected by cross-validation: the minimum value of the cross-validation error (*CVRE* = 0.483) was used to decide on the size of the tree (4 groups, Fig. 8.23). The R^2 of that tree (1 – relative error) was 0.788. The first split separated a group of 8 sites that had more twigs (≥ 8) than the other sites; that group had higher abundances of species 2 and 7 than the other sites. The second split isolated a group of 6 sites found on dryer ground

(water < 2.5); it had higher abundances of the last two species. The last split separated two groups (n = 6 and 8 respectively) according to soil humidity (water < 5.5 *versus* ≥ 5.5); the left-hand group is dominated by species 9, while the right hand group is the only one to show substantial abundances of species 1, 4, 5 and 6. De'ath (2002) confirmed the predictive values of these spider species to the groups by indicator value analysis (Subsection 8.9.3). An identical partition of the sites into four groups was obtained by applying MRT to the chi-square transformed spider data (eq. 7.70).

8.12 Clustering statistics

This section is devoted to clustering statistics. These include connectedness and isolation, and the correlation between a cophenetic matrix and the original distance matrix.

1 — Connectedness and isolation

The connectedness within clusters and their degree of isolation can be quantified using clustering statistics. Some of these measures are described here.

Connectedness

The basic statistic of a cluster k is its *number of objects*, n_k. In linkage clustering, a measure of link density of a cluster in A-space is obtained by comparing the number of objects to the number of links among them. Link density increases with the *degree of connectedness* of a cluster. Connectedness can be measured as follows (Estabrook, 1966):

$$Co = \frac{\text{number of links in a cluster}}{\text{maximum possible number of links}} \qquad \textbf{(8.24)}$$

Cohesion index

where the maximum possible number of links is $n_k(n_k - 1)/2$, with n_k being the number of objects in cluster k. This measure of connectedness varies between 0 and 1. Day (1977) proposed other related measures. One of them is the *cohesion index*, which considers only the links that exceed the minimum number of links necessary for the cluster to be connected. If this minimum number is called m, the cohesion index can be written as follows:

$$\frac{\text{No. links} - m}{[n_k (n_k - 1)/2] - m} \qquad \textbf{(8.25)}$$

For single linkage clustering, the minimum number of links necessary for n_k objects to be connected is $n_k - 1$, so that the cohesion index becomes:

$$\frac{2\,(\text{No. links} - n + 1)}{(n-1)\,(n-2)} \qquad \textbf{(8.26)}$$

which is Estabrook's (1966) normalized connectedness index. Other possible measures of cluster density are the maximum distance or minimum similarity within a cluster, and the mean distance or similarity (Estabrook, 1966).

Isolation

The degree of isolation of clusters in metric A-space (Fig. 7.2) can be measured as the distance between the two closest objects in different clusters. It may also be measured as the mean distance between all objects in one cluster and all objects in another, or else the ratio of the distance between the two closest objects to the distance between the centroids of the two clusters. These measures are ways of quantifying the distances between clusters; a clustering or ordination *of clusters* can be computed using these distances. In the context of linkage clustering without reference to a metric A-space, Wirth *et al*. (1966) used as measure of isolation the difference between the similarity at which a cluster is formed and the similarity at which it fuses with another cluster.

2 — Cophenetic correlation and related measures

Pearson's correlation coefficient, computed between the values in a cophenetic matrix (Subsection 8.3.1) and those in the original resemblance matrix (excluding the values on the diagonal), is called the *cophenetic correlation* (Sokal & Rohlf, 1962), *matrix correlation* (Sneath & Sokal, 1973) or *standardized Mantel* (1967) *statistic* (Subsection 10.5.1). It measures the extent to which the clustering result corresponds to the original resemblance matrix. When the clustering perfectly corresponds to the coefficients in the original matrix, the cophenetic correlation is 1. In R, the cophenetic distance matrix corresponding to a hierarchical clustering is computed by function ***cophenetic()*** of the STATS package. Following that, the cophenetic correlation between the original and cophenetic distance matrices can be computed using ***cor()***.

Besides the cophenetic correlation, which compares the original distances [or similarities] to those in a cophenetic matrix, matrix correlations are useful in the following situations:

- To compare any pair of resemblance matrices, such as the original distance matrix **D** of Ecological application 8.2, and a matrix of distances among the objects in a space of reduced dimension obtained from **D** by principal coordinate analysis (Section 9.3).

- To compare two distance [or similarity] matrices obtained by computing different resemblance measures on the same data.

- To compare the results of two clustering methods applied to a resemblance matrix.

- To compare different clustering levels in a dendrogram. The ultrametric matrix representing a given clustering level only contains zeros and ones in that case, as shown in Subsection 8.3.1.

Correlations take values between –1 and +1. The cophenetic correlation is expected to be positive if the original distances are compared to cophenetic distances (or similarities to similarities) and negative if distances are compared to similarities. The higher the absolute value of the cophenetic correlation, the better the correspondence between the two matrices that are compared. Ecologists might prefer to use a non-parametric correlation coefficient (Kendall's τ or Spearman's r) instead of Pearson's r, if the interest lies more in the geometric structure of the dendrogram than the actual lengths of its branches.

A cophenetic correlation cannot be tested for significance because the cophenetic matrix is not independent of the original distance or similarity matrix; one comes from the other through the clustering algorithm. In order to test the significance of a cophenetic correlation, one would have to pretend that, under H_0, the two matrices may be independent of each other, i.e. that the clustering algorithm is likely to have a null efficiency. On the contrary, the relationship between two hierarchical classifications of *different data sets* about the same objects, e.g. community composition and environmental, measured by matrix correlation or some other measure of consensus (Rohlf, 1974, 1982b), can be tested for significance (Section 10.2, Fig. 10.4).

Other coefficients have been proposed to measure the goodness-of-fit between matrices. For instance, Gower's (1983) distance is the sum of the squared differences between values in the original distance matrix and the cophenetic distance matrix:

Gower distance

$$D_{\text{Gower}} = \sum_{i,j} (original\ D_{ij} - cophenetic\ D_{ij})^2 \qquad \textbf{(8.27)}$$

This measure, also called *stress 1* (Kendall, 1938), takes values in the interval $[0, \infty)$; it is used in standardized form as a measure of goodness-of-fit in nonmetric multidimensional scaling (eq. 9.49). Small values indicate high fit. Like the cophenetic correlation, this measure only has relative value when comparing clustering results obtained from the same original distance matrix. Several other such functions are listed by Rohlf (1974).

Other measures have been proposed for comparing different *partitions* of the same objects. Consider in turn all pairs of objects and determine, for each one, whether the two objects are placed in the same group, or not, by the partition. One can construct a 2×2 contingency table, similar to the one shown at the beginning of Subsection 7.3.1, comparing the pair assignments made by two partitions. The simple matching coefficient (eq. 7.1), computed on this contingency table, is called the Rand index (1971). Hubert & Arabie (1985) suggested a modified form that corrects the Rand index as follows: if the relationship between two partitions is comparable to that of partitions picked at random, the corrected Rand index returns a value near 0. The *modified Rand index* is widely used for comparing partitions.

Modified Rand index

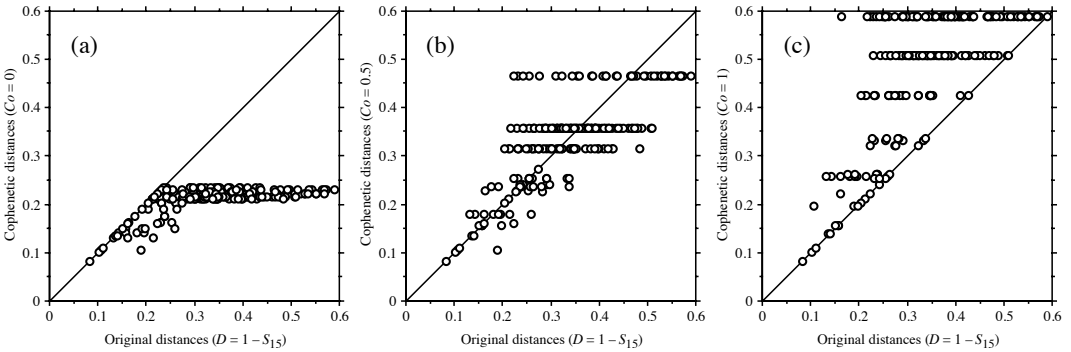


Figure 8.24 Shepard-like diagrams comparing cophenetic distances to the original distances for 21 objects analysed using three clustering methods: (a) single linkage ($Co = 0$, cophenetic $r = 0.64$, $\tau = 0.45$), (b) proportional link linkage ($Co = 0.5$, cophenetic $r = 0.75$, $\tau = 0.58$), and (c) complete linkage ($Co = 1$, cophenetic $r = 0.68$, $\tau = 0.51$). Co is the connectedness of the linkage clustering method (Subsection 8.5.3). There are 210 points (i.e. 210 distance pairs) in each graph. The diagonal lines are visual references.

Shepard-like diagram

A Shepard diagram is a scatter plot comparing distances in a space of reduced dimension, obtained by ordination methods, to distances in the original association matrix (Fig. 9.1). This type of diagram has been proposed by Shepard (1962) in the paper where he first described nonmetric multidimensional scaling (Section 9.4). Shepard-like diagrams can be constructed to compare the distances (or similarities) of the cophenetic matrix (Section 8.3) to the distances (or similarities) of the original resemblance matrix (Fig. 8.24). Such a plot may help choose between parametric and nonparametric cophenetic correlation coefficients: if the relationship between the original and cophenetic distances is curvilinear in the Shepard-like diagram, as it is the case in Figs. 24a and c, a nonparametric correlation coefficient should be used.

Figure 8.24 also helps in understanding the space-contraction effect of single linkage clustering, where the cophenetic distances are always smaller than or equal to the original distances; the space-conservation effect of intermediate linkage clustering with connectedness values around $Co = 0.5$; and the space-dilation effect of complete linkage clustering, in which cophenetic distances can never be smaller than the original distances. There are $(n - 1)$ clustering levels in a dendrogram. This limits to $(n - 1)$ the number of different values that can be found in a cophenetic matrix and, hence, along the ordinate of a Shepard-like diagram. This is why points form horizontal bands in Fig. 8.24.

Following are three measures of goodness-of-fit between the single linkage clustering results and the original distance matrix, for the pond example of Ecological application 8.2:

Pearson r cophenetic correlation = 0.9409
Kendall τ_b cophenetic correlation = 0.7736
Gower distance (D_{Gower}) = 0.1906

8.13 Cluster validation

Users of clustering methods may wonder whether the result of a clustering program run is valid or not, i.e. whether the clusters are "real", or simply artefacts of the clustering algorithm. Indeed, clustering algorithms may produce misleading results. except in simple situations where the clusters are well separated. On the one hand, most hierarchical clustering (or partitioning) algorithms will give rise to a hierarchy (or a partition), whether the objects are, or not, hierarchically interrelated (or pertaining to distinct clusters). On the other hand, different clustering algorithms may produce markedly different results because clustering methods impose different models onto the data, as shown in the present chapter: compare the dendrograms of Figs. 8.2, 8.7, 8.9, 8.11 and 8.15. Finally, different clustering methods are variously sensitive to noise (error) in the data. A simulation study comparing several clustering and partitioning methods under different levels of noise can be found in Milligan (1980); see also the review paper of Milligan (1996).

It is important to validate the results of cluster analyses. One has to show that a clustering structure departs from what may be expected from unstructured data. Unfortunately, most of the validation methods summarized below are not presently available in standard clustering packages or in R functions. Readers are referred to Chapter 4 of Jain & Dubes (1988) for details, and to the review papers of Perruchet (1983a, b), Bock (1989, 1996), Gordon (1994, 1996a, 1996b) and Milligan (1996). Lapointe (1998) reviewed the validation methods used in phylogenetic studies.

Validation may be carried out in nonstatistical or statistical ways. Statistical ways involve tests of hypotheses, whereas nonstatistical assessment accepts weaker evidence for the presence of clusters. Commonly-used nonstatistical methods are:

- Plot the clusters onto an ordination diagram and look for separation of the clusters (Section 10.1). This method is often used to assess the degree of refinement of hierarchical clustering results that one should consider for interpretation.

- Compare the results of several clustering algorithms, either informally (using visual examination, identify the partition levels that are found in most or all trees being compared), or formally (calculate consensus indices or construct a compromise "consensus" tree: below).

Different issues can be considered in cluster validation:

• The most general hypothesis is that of complete absence of classification structure in the data. In principle, such tests should be carried out before cluster analysis is attempted. Several methods have been proposed to assess the positions of the objects distributed in multidimensional space (random position hypothesis) and test for either uniform or unimodal distributions (i.e. greater density of objects near the centre of the distribution); see Gordon (1996a, 1996b). There are also tests that are carried out on graphs linking the objects, and others that involve only the object labels.

• Other methods are available to test (1) for the presence of a hierarchical structure in the data, (2) for partitions (are there distinct clusters in the data? how many clusters?), or (3) for the validity of individual clusters.

For any one of these hypotheses, validation may be carried out at different conceptual levels.

1. *Internal validation using* **Y**. — *Internal validation* methods allow the assessment of the *consistency* of a clustering topology. Internal validation consists in using the original data (i.e. matrix **Y** containing the data originally used for clustering) to assess the clustering results. One approach is to resample the original data set. One repeatedly draws subsets of objects at random, using sampling with or without replacement, to verify that the original clusters of objects are found by the clustering method for the different subsets. Nemec & Brinkhurst (1988) present an ecological application of this method to species abundance data. Another approach is to randomize the original data set, or generate random simulated data with similar distribution parameters, and compute the classification a large number of times to obtain a null distribution for some clustering statistic of interest, which can be tested using the null distribution; one may use one of the statistics discussed in Subsection 8.12.2, or the *U* statistic of Gordon (1994) described at the end of Subsection 10.5.3. The test of cluster fusion in chronological clustering (Subsection 12.6.5) is an example of an internal validation criterion. Using simulations, Milligan (1981) compared 30 internal validation criteria that may be used in this type of study. One *must not*, however, use a standard hypothesis testing procedure such as ANOVA or MANOVA on the variables used to determine the clusters. This approach would be incorrect because the *alternative hypothesis* of the test would be constructed to fit the group structure since it would be computed from the same data that would now be used for testing the null hypothesis. As a consequence, such a test would almost necessarily (subject to type II error) result in significant differences among the groups. To illustrate this point, one can generate multivariate data at random using the uniform distribution and carry out clustering: a MANOVA comparing the clusters to the original data would produce a significant result in most cases even though the data are random and thus have no structure.

2. *External validation comparing* **Y** *to* **X**. — *External validation* methods involve the comparison of two different data tables. The clustering results derived from data matrix **Y**, e.g. species, are compared to a matrix of explanatory variables,

e.g. environmental, which is called **X** in the contexts of regression (Chapter 10) and canonical analysis (Chapter 11). Comparisons can be made at different levels. One may compare a partition of the objects based on **Y** to matrix **X** using linear discriminant analysis (Table 10.1; Section 11.3). Else, the whole hierarchical tree structure may be coded using binary variables (Baum, 1992; Ragan, 1992), in the same way as nested factors in ANOVA; this matrix is then compared to the explanatory matrix **X** using RDA or CCA (Sections 11.1 and 11.2). A third way is to compare the cophenetic matrix (Section 8.3) that represents the hierarchical tree structure to a distance or similarity matrix computed from matrix **X**, using a Mantel test (Subsection 10.5.1; Hubert & Baker, 1977). Contrary to the cophenetic correlations considered in Subsection 8.12.2, testing is legitimate here because matrix **X** is independent of the data matrix **Y** used to construct the classification, but note that the Mantel test has low power compared to the other methods mentioned above.

3. *External validation comparing two or several matrices* **Y**, *same variables*. — Confirmation of the presence of a clustering structure in the data can be obtained by repeating the cluster analysis using different sets of objects (data matrices $\mathbf{Y}_1$, $\mathbf{Y}_2$, etc., all with the same descriptors) and comparing the results. Consider the situation where replicate data are available. If, for example, lakes can be selected at random from different geographic regions, one can conduct independent cluster analyses of the regions using one lake per region, different lakes being used in the separate runs, followed by a comparison of the resulting partitions or dendrograms representing the classifications of regions. Methods are available for comparing independently-obtained dendrograms representing the same objects (Fig. 10.4 and references in Section 10.2). A second approach is to take the classification of regions obtained from the first set of lakes (matrix $\mathbf{Y}_1$) as a model to be validated, using discriminant analysis, by comparing it to a second, independent set of lakes (matrix $\mathbf{Y}_2$) representing the same regions.

A third approach is *replication analysis*, where external validation is carried out for data that are not replicate observations of the same objects. One finds a classification using matrix $\mathbf{Y}_1$, determines group centroids, and assigns the data points in $\mathbf{Y}_2$ to the nearest centroid (McIntyre & Blashfield, 1980). Then, the data in $\mathbf{Y}_2$ are clustered without considering the result from $\mathbf{Y}_1$. The independently obtained classification of $\mathbf{Y}_2$ is compared to the first one using some appropriate measure of consensus (point 4 below).

In studies where data are costly to obtain, this approach is, in most cases, not appealing to researchers who are more interested in using all the available information in a single cluster analysis, instead of dividing the data set into two or several analyses. This approach is only feasible when the objects are numerous.

4. *External validation comparing two or several matrices* **Y**, *same objects*. — Several groups of descriptors may be available about the same objects, and one may wish to conduct separate cluster analyses on them. An example would be sites where data are available about several groups of arthropods (e.g. matrices $\mathbf{Y}_1$ = acarians, $\mathbf{Y}_2$ = insects,

and $\mathbf{Y}_3$ = spiders), besides physical or other variables of the environment which would form a matrix **X** of explanatory variables. Classifications may be obtained independently for each matrix **Y**. Measures of resemblance between trees, called *consensus indices* (Rohlf, 1982b), may be calculated. The cophenetic correlation coefficient of the previous subsection can be used as a consensus index; other indices are available, that only take the classification topologies into account. Alternatively, one may compute a compromise tree, called a *consensus tree*, which represents the areas of agreement among trees. Several criteria have been proposed for constructing consensus trees: majority rule, strict consensus, average consensus, etc. (Leclerc & Cucumel, 1987). Tests of significance are available for comparing independently-obtained dendrograms that describe relationships among the same objects (Fig. 10.4 and references in Section 10.2).

Cluster validation has progressed in important ways during the last decade, with new methods and packages being made available. Summarizing these developments, Rendón *et al*. (2011) described and compared a large number of internal and external cluster validation indexes. Because cluster validity indices (CVI) are numerous and no single CVI always outperforms the others, Kryszczuk & Hurley (2010) proposed composite validation indices combining different approaches. Brock *et al*. (2008) wrote the R package CLVALID for cluster validation; see Section 8.15.

8.14 Cluster representation and choice of a method

This section summarizes the most usual graphical representations of clustering results. More complete reviews of the subject are found in Sneath & Sokal (1973) and Chambers & Kleiner (1982).

Hierarchical clustering results are represented, in most cases, as dendrograms, e.g. Fig. 8.2b. They can also be represented as plots of connected subgraphs, e.g. Fig. 8.2a; the construction of these informative graphs, which would be difficult to draw by computer, was explained in Section 8.2. The branches of dendrograms may point upwards, downwards or sideways; the horizontal representation is an easier way of plotting a dendrogram that contains a large number of objects and fitting it into a page. Dendrograms are graduated in distances or similarities; the branching pattern indicates the distance or similarity of bifurcating branches. Usually, the names of the objects (or descriptors when descriptors are clustered), or their code numbers, are written at the tips of the branches. The ordinate (on horizontal dendrograms) has no specified ordering, except in TWINSPAN. Bifurcating branches are not fixed; they may be swivelled as required by the presentation of results without altering the nature of the ultrametric information in the dendrogram.

Dendrogram

Dendrograms clearly illustrate the clusters formed at each partition level, but in linkage clustering they do not allow the identification of the exact links among objects that generate cluster fusions. With some clustering methods, this information is not

directly available and must be found *a posteriori* when needed; how to compute the chain of primary connections was described at the end of Subsection 8.5.4 for the UPGMA clustering case. In synoptic clustering, which only aims at recognizing major clusters of objects, connecting links are not required.

Connected subgraphs

Series of connected subgraphs, as drawn in Fig. 8.2a, may be used to represent all the information of the distance or similarity matrix. Complex information may be represented by different types of lines; colours may also be used. When they become numerous, objects can be placed at the rim of a circle; distance links are drawn as lines between them. In each subgraph, the relative positions of the objects are of little importance. They are merely arranged in such a way as to simplify the paths of the links connecting them. The objects may have been positioned beforehand in a two-dimensional ordination space, which may be obtained by principal coordinate analysis or nonmetric scaling of the association matrix (Sections 9.3 and 9.4). Figures of connected subgraphs, informative as they may be, are quite time consuming to draw and difficult to publish.

Skyline plot
Tree
Icicle plot

Some programs still use "skyline plots" (Ward, 1963; Wirth *et al*., 1966), which may also be called "trees" or "icicle plots". These plots may be imagined as negative pictures of dendrograms. They contain the same information as dendrograms, but they are rather odd to read and interpret. In Fig. 8.25a for instance (UPGMA clustering of the pond data, see Fig. 8.5), the object names are sitting on the lines *between* columns of X's; the ordinate of the plot is a scale of distances or similarities. Since the value $D = 0$ is at the bottom of the graph, this is where the hierarchical agglomeration begins. The first clustering step is materialized by the first horizontal row of X's, at distance $D = 0.4$, which joins objects 212 and 214. It is drawn like the lintel of a door. The surface above the lintel of X's is filled with X's; these are without meaning. The next clustering step is at distance $D = 0.5$; it consists in a row of X's joining ponds 431 and 432. The third clustering step is more interesting. Note how a new lintel of X's, at $D = 0.75$, goes from pond 233, and right across the column of X's already joining ponds 431 and 432. The final clustering step is at $D = 0.942$. This new lintel crosses the last remaining gap, uniting the two columns of X's corresponding to the two already-formed clusters.

A skyline plot can be directly transformed into a dendrogram (Fig. 8.25b, c). Working from the bottom to the top, proceed as follows:

- Identify lintels and draw lines across the column of X's. The lintel lines should not extend beyond the row of X's.

- When all the horizontal lintel lines have been drawn, draw vertical lines from individual objects up to the first lintel encountered, and from the *centre* of a lower lintel up to the one above. Erase the overhanging part of the upper lintel. Repeat the operation for the next lintel up.

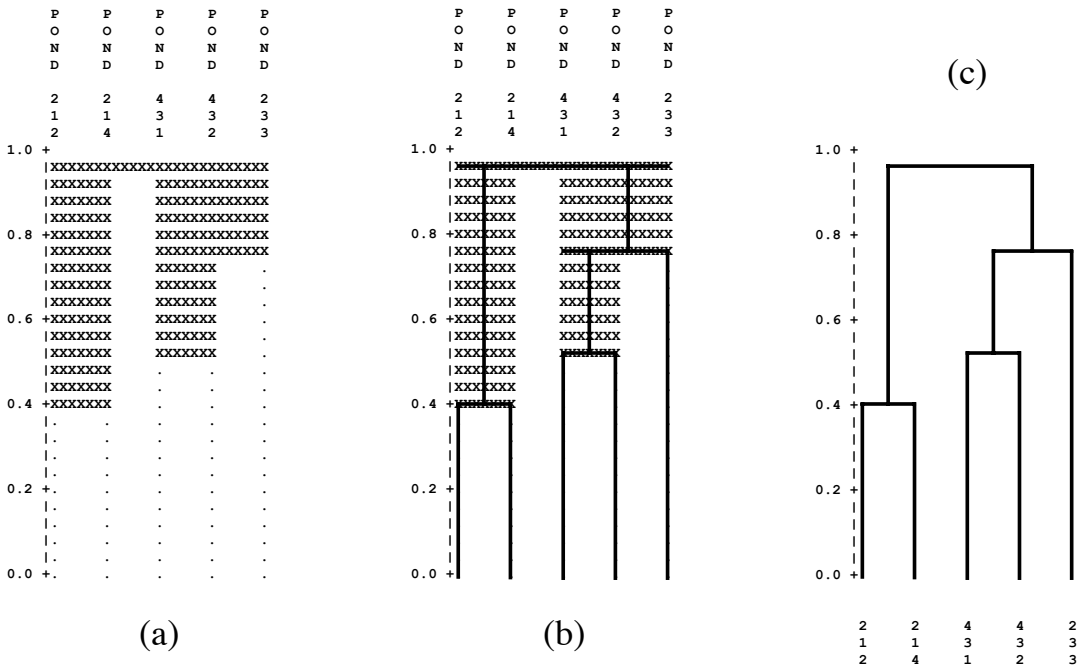


Figure 8.25 A skyline plot (a) can be transformed into a dendrogram (c) by going through the drawing steps described in (b). Vertical scale: distances. The skyline plot was computed using SAS.

• Erase the X's. The result is a standard dendrogram (Fig. 8.25c). It is identical to the dendrogram representing the same clustering results in Fig. 8.5, but it is drawn here vertically instead of horizontally, and pond 233 is swivelled to the right instead of being between ponds 214 and 431.

Heat maps (see Section 8.15) can be used to represent **D** matrices graphically, before or after clustering, e.g. Fig. 8.21. Dendrograms can be represented along the axes of heat maps as in Fig. 8.21c.

Section 10.1 shows how to superimpose clustering results onto an ordination of the same objects. This often helps evidence the structure when ecological objects form a continuum. When it comes to representing the results of a partition, the objects are represented in an ordination space and symbols can be used to represent the groups; else, envelopes can be drawn around points corresponding to the groups.

Table 8.11 summarizes, in a comparative way, the various clustering methods discussed in the present chapter. Some advantages and disadvantages of each method are pointed out.

Table 8.11 Synoptic summary of the clustering methods presented in Chapter 8.

Method	Pros & cons	Use in ecology
Hierarchical agglomeration: linkage clustering	Pairwise relationships among the objects are known.	
Single linkage	Computation simple; contraction of space (chaining); combinatorial method.	Good complement to ordination.
Complete linkage (see also: species associations)	Dense nuclei of objects; space expansion; many objects cluster at high distance; arbitrary rules to resolve conflicts; combinatorial method.	To increase the contrast among clusters.
Intermediate linkage	Preservation of reference space A; non-combinatorial: not included in Lance & Williams' general model.	Preferable to the above two methods if only one clustering method is to be used.
Hierarchical agglomeration: average clustering	Preservation of reference space A; pairwise relationships between objects are lost; combinatorial method.	
Unweighted arithmetic average (UPGMA)	Fusion of clusters when the distance reaches the mean inter-cluster distance value.	For a collection of objects obtained by simple random or systematic sampling.
Weighted arithmetic average (WPGMA)	As UPGMA, with adjustment for group sizes.	Preferable to the previous method in all other sampling situations.
Unweighted centroid (UPGMC)	Fusion of clusters with closest centroids; may produce reversals.	For simple random or systematic samples of objects.
Weighted centroid (WPGMC)	As UPGMC, with adjustment for group sizes; may produce reversals.	Preferable to the previous method in all other sampling situations.
Ward's method	Minimizes the within-group sum of squares.	When looking for hyperspherical clusters in space A.
Hierarchical agglomeration: flexible clustering	Allows contraction, conservation, or dilation of space A; pairwise relationships between objects are lost; combinatorial method.	All combinatorial methods, including this one, are implemented using the simple Lance & Williams algorithm.
Hierarchical agglomeration: information analysis	Minimal chaining; only for Q-mode clustering based upon presence-absence of species.	Ecological use is unclear: distances reflect double absences as well as double presences.

Table 8.13 Continued.

Method	Pros & cons	Use in ecology
Hierarchical division	Danger of incorrect separation of members of minor clusters near the beginning of clustering.	
Monothetic	Division of the objects following the states of the "best" descriptor at each step of the procedure.	Useful to split data into large clusters, inside which clustering depends on different phenomena.
Polythetic	For small number of objects only.	Computation impossible for sizable data sets.
Division in ordination space	Binary division along each axis of ordination space; no search is done for high concentrations of objects in space A.	Efficient algorithms for large data sets, when a coarse division of the objects is sought.
TWINSPAN	Dichotomized ordination analysis; ecological justification of several steps unclear.	Produces an ordered two-way table classifying sites and species.
***K*-means partitioning**	Minimizes within-group sum of squares; different rules may suggest different optimal numbers of clusters.	Produces a partition of the objects into K groups, K being determined by the user.
Species associations	Non-hierarchical methods; clustering at a pre-selected level of similarity or probability.	Concept of association based on co-occurring or correlated species.
Non-hierarchical complete linkage	Species associated by complete linkage (no overlap); satellite species joined by single linkage (possible overlap).	Straightforward concept; no easily available software.
Concordance analysis	Find groups of species that form statistically significant associations.	Clear, easy to apply method; R functions available.
Multivariate regression tree	A multivariate response table is constrained by a table of explan. variables, producing a tree.	Two-matrix method related to canonical analysis.
Seriation	One-dimensional ordination along the main diagonal of a distance matrix.	Useful to analyse non-symmetric association matrices.
Indicator species		
TWINSPAN	Only for classifications of sites obtained by splitting CA axes; justification of some steps unclear.	Gives indicator values for the pseudospecies.
Indicator value index	For any hierarchical or non-hierarchical classification of sites; *IndVal* for a species is not affected by the other species in the study.	Gives indicator values for the species under study; the *IndVal* index is tested by permutation.

8.15 Software

Several, but not all statistical packages offer clustering capabilities: SAS, SPSS, SYSTAT, JMP, STATISTICA, and NTSYSPC offer clustering among their methods for data analysis. All packages with clustering procedures offer at least a Lance & Williams algorithm capable of carrying out the clustering methods listed in Table 8.9. Many also have a *K*-means partitioning algorithm. Few offer proportional-link linkage or additional forms of clustering. Some methods are available in specialized packages only: clustering with constraints of temporal (Section 12.6) or spatial contiguity (Section 13.3); fuzzy clustering (algorithms described e.g. in Bezdek, 1987); or clustering by neural network algorithms (algorithms described e.g. in Fausett, 1994).

Functions in the R language are available to carry out all analyses described in this chapter.

1. Several R functions are devoted to clustering. Hierarchical clustering is computed using ***hclust()*** in STATS and ***agnes()*** in CLUSTER using the Lance & Williams general agglomerative algorithm. Functions for constrained hierarchical clustering are listed in Sections 12.8 and 13.6. A cophenetic distance matrix corresponding to a hierarchical clustering is computed by function ***cophenetic()*** of STATS.

Minimum spanning trees can be computed by several functions including ***mstree()*** in ADE4, ***mst()*** in APE, ***mstree()*** in SPDEP and ***spantree()*** in VEGAN. Function ***cophenetic()*** in STATS computes the cophenetic matrix corresponding to a hierarchical clustering. Function ***clustIndex()*** of CCLUST computes stopping indices for clustering.

2. *K*-means partitioning is available in functions ***kmeans()*** of STATS, ***cclust()*** of CCLUST, ***kkmeans()*** of KERNLAB, ***KMeans()*** of RCMDR and ***cascadeKM()*** of VEGAN; the latter function automatically repeats *K*-means partitioning using a range of values of *K*.

Heat map

3. Seriation is obtained by function ***seriate()*** of package SERIATION, which offers several calculation methods. *Heat maps* in colour can be obtained using function ***heatmap()*** of STATS, or by function ***hmap()*** of SERIATION, which calls ***heatmap()*** to produce the plot. Heat maps are also produced by function ***coldiss()*** available on the Web page of the Borcard *et al*. (2011) book, http://numericalecology.com/NEwR. With ***coldiss()***, a **D** matrix is represented by an unordered and a reordered colour heat maps, the new ordering being the result of single linkage chaining. Function ***seriation()***[*] carries out the Beum-Brundage seriation procedure for non-symmetric or symmetric matrices.

4. Multivariate regression tree analysis is available in ***mvpart()*** of package MVPART. Package MVPARTWRAP contains additional functions for multivariate regression tree

[*] Available on the Web page http://numericalecology.com/rcode.

analysis, including ***CascadeMRT()*** that carries out two MRT analyses in sequence, using explanatory matrices in the order specified by the researcher.

5. Other clustering methods have been described in the statistical literature. For instance, *K*-means partitioning is a member of a larger class of methods called *K*-centroids, where the Euclidean distance is replaced by other distances; for example, using the Manhattan distance instead of the Euclidean produces *K*-medians clustering. Package FLEXCLUST offers different types of clustering, including function ***kcca()*** that computes various types of *K*-centroid cluster analysis (*K*-means, *K*-medians and others). For distances other than the Euclidean, the *K*-centroid approach is also called *partitioning around medoids* (Kaufman & Rousseeuw, 1990); it is implemented in function ***pam()*** of the CLUSTER package. An example of partitioning around medoids is presented in Subsection 4.8.2 of the Borcard *et al*. (2011) book.

6. Fuzzy partitioning is available in functions ***fanny()*** of package CLUSTER and ***cmeans()*** of package E1071. An example of analysis in Q mode is presented in Subsection 4.12.1 of Borcard *et al*. (2011). Function ***vegclust()*** of package VEGCLUST offers three forms of fuzzy partitioning (fuzzy *c*-means, probabilistic *c*-means, and noise clustering) in addition to hard *K*-means.

7. Concordance analysis to search for species associations is available in functions ***kendall.global()*** and ***kendall.post()*** of the VEGAN package.

8. Indicator value indices (*INDVAL*, EQ. 8.21) can be computed by functions ***strassoc()*** and ***multipatt()*** of INDICSPECIES and function ***indval()*** of LABDSV. The functions in INDICSPECIES offer a choice of several different indicator statistics described in De Cáceres & Legendre (2009).

9. Function ***clValid()*** of the CLVALID package computes validation measures for clustering results, including internal validation and stability measures, plus biological measures for genetic data. The package is described in Brock *et al*. (2008). Function ***randIndex()*** computes the Rand and modified Rand indices quantifying the agreement of two partitions.