Chapter 13
Cluster Analysis

The next two chapters address classification issues from two varying perspectives. When considering groups of objects in a multivariate data set, two situations can arise. Given a data set containing measurements on individuals, in some cases we want to see if some natural groups or classes of individuals exist, and in other cases, we want to classify the individuals according to a set of existing groups. Cluster analysis develops tools and methods concerning the former case, that is, given a data matrix containing multivariate measurements on a large number of individuals (or objects), the objective is to build some natural sub-groups or clusters of individuals. This is done by grouping individuals that are "similar" according to some appropriate criterion. Once the clusters are obtained, it is generally useful to describe each group using some descriptive tool from Chaps. 1, 10 or 11 to create a better understanding of the differences that exist among the formulated groups.

Cluster analysis is applied in many fields such as the natural sciences, the medical sciences, economics, marketing, etc. In marketing, for instance, it is useful to build and describe the different segments of a market from a survey on potential consumers. An insurance company, on the other hand, might be interested in the distinction among classes of potential customers so that it can derive optimal prices for its services. Other examples are provided below.

Discriminant analysis presented in Chap. 14 addresses the other issue of classification. It focuses on situations where the different groups are known a priori. Decision rules are provided in classifying a multivariate observation into one of the known groups.

Section 13.1 states the problem of cluster analysis where the criterion chosen to measure the similarity among objects clearly plays an important role. Section 13.2

W.K. Härdle, L. Simar, *Applied Multivariate Statistical Analysis*,
DOI 10.1007/978-3-662-45171-7_13

shows how to precisely measure the proximity between objects. Finally, Sect. 13.3 provides some algorithms. We will concentrate on hierarchical algorithms only where the number of clusters is not known in advance.

13.1 The Problem

Cluster analysis is a set of tools for building groups (clusters) from multivariate data objects. The aim is to construct groups with homogeneous properties out of heterogeneous large samples. The groups or clusters should be as homogeneous as possible and the differences among the various groups as large as possible. Cluster analysis can be divided into two fundamental steps.

1. *Choice of a proximity measure:*
   One checks each pair of observations (objects) for the similarity of their values. A similarity (proximity) measure is defined to measure the "closeness" of the objects. The "closer" they are, the more homogeneous they are.
2. *Choice of group-building algorithm:*
   On the basis of the proximity measures the objects assigned to groups so that differences between groups become large and observations in a group become as close as possible.

In marketing, for example, cluster analysis is used to select test markets. Other applications include the classification of companies according to their organisational structures, technologies and types. In psychology, cluster analysis is used to find types of personalities on the basis of questionnaires. In archaeology, it is applied to classify art objects in different time periods. Other scientific branches that use cluster analysis are medicine, sociology, linguistics and biology. In each case a heterogeneous sample of objects are analysed with the aim to identify homogeneous sub-groups.

Summary

↪ Cluster analysis is a set of tools for building groups (clusters) from multivariate data objects.

↪ The methods used are usually divided into two fundamental steps: The choice of a proximity measure and the choice of a group-building algorithm.

13.2 The Proximity Between Objects

The starting point of a cluster analysis is a data matrix $\mathcal{X}(n \times p)$ with n measurements (objects) of p variables. The proximity (similarity) among objects is described by a matrix $\mathcal{D}(n \times n)$

$$\mathcal{D} = \begin{pmatrix} d_{11} & d_{12} & \dots & \dots & \dots & d_{1n} \\ \vdots & d_{22} & & & & \vdots \\ \vdots & \vdots & \ddots & & & \vdots \\ \vdots & \vdots & & \ddots & & \vdots \\ \vdots & \vdots & & & \ddots & \vdots \\ d_{n1} & d_{n2} & \dots & \dots & \dots & d_{nn} \end{pmatrix}. \tag{13.1}$$

The matrix $\mathcal{D}$ contains measures of similarity or dissimilarity among the n objects. If the values d_{ij} are distances, then they measure dissimilarity. The greater the distance, the less similar are the objects. If the values d_{ij} are proximity measures, then the opposite is true, i.e. the greater the proximity value, the more similar are the objects. A distance matrix, for example, could be defined by the L_2-norm: $d_{ij} = \|x_i - x_j\|_2$, where x_i and x_j denote the rows of the data matrix $\mathcal{X}$. Distance and similarity are of course dual. If d_{ij} is a distance, then $d'_{ij} = \max_{i,j}\{d_{ij}\} - d_{ij}$ is a proximity measure.

The nature of the observations plays an important role in the choice of proximity measure. Nominal values (like binary variables) lead in general to proximity values, whereas metric values lead (in general) to distance matrices. We first present possibilities for $\mathcal{D}$ in the binary case and then consider the continuous case.

Similarity of Objects with Binary Structure

In order to measure the similarity between objects we always compare pairs of observations (x_i, x_j) where $x_i^\top = (x_{i1}, \dots, x_{ip})$, $x_j^\top = (x_{j1}, \dots, x_{jp})$, and $x_{ik}, x_{jk} \in \{0, 1\}$. Obviously there are four cases:

$$\begin{aligned} x_{ik} &= x_{jk} = 1, \\ x_{ik} &= 0, x_{jk} = 1, \\ x_{ik} &= 1, x_{jk} = 0, \\ x_{ik} &= x_{jk} = 0. \end{aligned}$$

Define

$$a_1 = \sum_{k=1}^{p} \boldsymbol{I}(x_{ik} = x_{jk} = 1),$$

$$a_2 = \sum_{k=1}^{p} \boldsymbol{I}(x_{ik} = 0, x_{jk} = 1),$$

$$a_3 = \sum_{k=1}^{p} \boldsymbol{I}(x_{ik} = 1, x_{jk} = 0),$$

$$a_4 = \sum_{k=1}^{p} \boldsymbol{I}(x_{ik} = x_{jk} = 0).$$

Note that each $a_l, l = 1, \ldots, 4$, depends on the pair (x_i, x_j).

The following proximity measures are used in practice:

$$d_{ij} = \frac{a_1 + \delta a_4}{a_1 + \delta a_4 + \lambda(a_2 + a_3)} \tag{13.2}$$

where δ and λ are weighting factors. Table 13.1 shows some similarity measures for given weighting factors.

These measures provide alternative ways of weighting mismatching and positive (presence of a common character) or negative (absence of a common character) matchings. In principle, we could also consider the Euclidean distance. However, the disadvantage of this distance is that it treats the observations 0 and 1 in the same way. If $x_{ik} = 1$ denotes, say, knowledge of a certain language, then the contrary, $x_{ik} = 0$ (not knowing the language) should eventually be treated differently.

Example 13.1 Let us consider binary variables computed from the car data set (Table 22.7). We define the new binary data by

$$y_{ik} = \begin{cases} 1 & \text{if } x_{ik} > \overline{x}_k, \\ 0 & \text{otherwise,} \end{cases}$$

Table 13.1 The common similarity coefficients

Name	δ	λ	Definition
Jaccard	0	1	$\frac{a_1}{a_1 + a_2 + a_3}$
Tanimoto	1	2	$\frac{a_1 + a_4}{a_1 + 2(a_2 + a_3) + a_4}$
Simple matching (M)	1	1	$\frac{a_1 + a_4}{p}$
Russel and Rao (RR)	–	–	$\frac{a_1}{p}$
Dice	0	0.5	$\frac{2a_1}{2a_1 + (a_2 + a_3)}$
Kulczynski	–	–	$\frac{a_1}{a_2 + a_3}$

for $i = 1, \ldots, n$ and $k = 1, \ldots, p$. This means that we transform the observations of the k-th variable to 1 if it is larger than the mean value of all observations of the k-th variable. Let us only consider the data points 17 to 19 (Renault 19, Rover and Toyota Corolla) which lead to (3×3) distance matrices. The Jaccard measure gives the similarity matrix

$$\mathcal{D} = \begin{pmatrix} 1.000 & 0.000 & 0.400 \\ & 1.000 & 0.167 \\ & & 1.000 \end{pmatrix},$$

the Tanimoto measure yields

$$\mathcal{D} = \begin{pmatrix} 1.000 & 0.000 & 0.455 \\ & 1.000 & 0.231 \\ & & 1.000 \end{pmatrix},$$

whereas the Simple Matching measure gives

$$\mathcal{D} = \begin{pmatrix} 1.000 & 0.000 & 0.625 \\ & 1.000 & 0.375 \\ & & 1.000 \end{pmatrix}.$$

Distance Measures for Continuous Variables

A wide variety of distance measures can be generated by the L_r-norms, $r \geq 1$,

$$d_{ij} = ||x_i - x_j||_r = \left\{ \sum_{k=1}^{p} |x_{ik} - x_{jk}|^r \right\}^{1/r}. \tag{13.3}$$

Here x_{ik} denotes the value of the k-th variable on object i. It is clear that $d_{ii} = 0$ for $i = 1, \ldots, n$. The class of distances (13.3) for varying r measures the dissimilarity of different weights. The L_1-metric, for example, gives less weight to outliers than the L_2-norm (Euclidean norm). It is common to consider the squared L_2-norm.

Example 13.2 Suppose we have $x_1 = (0, 0)$, $x_2 = (1, 0)$ and $x_3 = (5, 5)$. Then the distance matrix for the L_1-norm is

$$\mathcal{D}_1 = \begin{pmatrix} 0 & 1 & 10 \\ 1 & 0 & 9 \\ 10 & 9 & 0 \end{pmatrix},$$

and for the squared L_2- or Euclidean norm

$$\mathcal{D}_2 = \begin{pmatrix} 0 & 1 & 50 \\ 1 & 0 & 41 \\ 50 & 41 & 0 \end{pmatrix}.$$

One can see that the third observation x_3 receives much more weight in the squared L_2-norm than in the L_1-norm.

An underlying assumption in applying distances based on L_r-norms is that the variables are measured on the same scale. If this is not the case, a standardisation should first be applied. This corresponds to using a more general L_2- or Euclidean norm with a metric $\mathcal{A}$, where $\mathcal{A} > 0$ (see Sect. 2.6):

$$d_{ij}^2 = \|x_i - x_j\|_{\mathcal{A}} = (x_i - x_j)^\top \mathcal{A}(x_i - x_j). \tag{13.4}$$

L_2-norms are given by $\mathcal{A} = \mathcal{I}_p$, but if a standardisation is desired, then the weight matrix $\mathcal{A} = \operatorname{diag}(s_{X_1X_1}^{-1}, \ldots, s_{X_pX_p}^{-1})$ may be suitable. Recall that $s_{X_kX_k}$ is the variance of the k-th component. Hence we have

$$d_{ij}^2 = \sum_{k=1}^{p} \frac{(x_{ik} - x_{jk})^2}{s_{X_kX_k}}. \tag{13.5}$$

Here each component has the same weight in the computation of the distances and the distances do not depend on a particular choice of the units of measure.

Example 13.3 Consider the French Food expenditures (Table 22.6). The Euclidean distance matrix (squared L_2-norm) is

$$\mathcal{D} = 10^4 \cdot \begin{pmatrix}
0.00 & 5.82 & 58.19 & 3.54 & 5.15 & 151.44 & 16.91 & 36.15 & 147.99 & 51.84 & 102.56 & 271.83 \\
 & 0.00 & 41.73 & 4.53 & 2.93 & 120.59 & 13.52 & 25.39 & 116.31 & 43.68 & 76.81 & 226.87 \\
 & & 0.00 & 44.14 & 40.10 & 24.12 & 29.95 & 8.17 & 25.57 & 20.81 & 20.30 & 88.62 \\
 & & & 0.00 & 0.76 & 127.85 & 5.62 & 21.70 & 124.98 & 31.21 & 72.97 & 231.57 \\
 & & & & 0.00 & 121.05 & 5.70 & 19.85 & 118.77 & 30.82 & 67.39 & 220.72 \\
 & & & & & 0.00 & 96.57 & 48.16 & 1.80 & 60.52 & 28.90 & 29.56 \\
 & & & & & & 0.00 & 9.20 & 94.87 & 11.07 & 42.12 & 179.84 \\
 & & & & & & & 0.00 & 46.95 & 6.17 & 18.76 & 113.03 \\
 & & & & & & & & 0.00 & 61.08 & 29.62 & 31.86 \\
 & & & & & & & & & 0.00 & 15.83 & 116.11 \\
 & & & & & & & & & & 0.00 & 53.77 \\
 & & & & & & & & & & & 0.00
\end{pmatrix}.$$

Taking the weight matrix $\mathcal{A} = \text{diag}(s_{X_1X_1}^{-1}, \ldots, s_{X_7X_7}^{-1})$, we obtain the distance matrix (squared L_2-norm)

$$\mathcal{D} = \begin{pmatrix} 0.00 & 6.85 & 10.04 & 1.68 & 2.66 & 24.90 & 8.28 & 8.56 & 24.61 & 21.55 & 30.68 & 57.48 \\ & 0.00 & 13.11 & 6.59 & 3.75 & 20.12 & 13.13 & 12.38 & 15.88 & 31.52 & 25.65 & 46.64 \\ & & 0.00 & 8.03 & 7.27 & 4.99 & 9.27 & 3.88 & 7.46 & 14.92 & 15.08 & 26.89 \\ & & & 0.00 & 0.64 & 20.06 & 2.76 & 3.82 & 19.63 & 12.81 & 19.28 & 45.01 \\ & & & & 0.00 & 17.00 & 3.54 & 3.81 & 15.76 & 14.98 & 16.89 & 39.87 \\ & & & & & 0.00 & 17.51 & 9.79 & 1.58 & 21.32 & 11.36 & 13.40 \\ & & & & & & 0.00 & 1.80 & 17.92 & 4.39 & 9.93 & 33.61 \\ & & & & & & & 0.00 & 10.50 & 5.70 & 7.97 & 24.41 \\ & & & & & & & & 0.00 & 24.75 & 11.02 & 13.07 \\ & & & & & & & & & 0.00 & 9.13 & 29.78 \\ & & & & & & & & & & 0.00 & 9.39 \\ & & & & & & & & & & & 0.00 \end{pmatrix}. \tag{13.6}$$

When applied to contingency tables, a χ^2-metric is suitable to compare (and cluster) rows and columns of a contingency table.

If $\mathcal{X}$ is a contingency table, row i is characterised by the conditional frequency distribution $\frac{x_{ij}}{x_{i\bullet}}$, where $x_{i\bullet} = \sum_{j=1}^{p} x_{ij}$ indicates the marginal distributions over the rows: $\frac{x_{i\bullet}}{x_{\bullet\bullet}}$, $x_{\bullet\bullet} = \sum_{i=1}^{n} x_{i\bullet}$. Similarly, column j of $\mathcal{X}$ is characterised by the conditional frequencies $\frac{x_{ij}}{x_{\bullet j}}$, where $x_{\bullet j} = \sum_{i=1}^{n} x_{ij}$. The marginal frequencies of the columns are $\frac{x_{\bullet j}}{x_{\bullet\bullet}}$.

The distance between two rows, i_1 and i_2, corresponds to the distance between their respective frequency distributions. It is common to define this distance using the χ^2-metric:

$$d^2(i_1, i_2) = \sum_{j=1}^{p} \frac{1}{\left(\frac{x_{\bullet j}}{x_{\bullet\bullet}}\right)} \left(\frac{x_{i_1 j}}{x_{i_1\bullet}} - \frac{x_{i_2 j}}{x_{i_2\bullet}}\right)^2. \tag{13.7}$$

Note that this can be expressed as a distance between the vectors $x_1 = \left(\frac{x_{i_1 j}}{x_{\bullet\bullet}}\right)$ and $x_2 = \left(\frac{x_{i_2 j}}{x_{\bullet\bullet}}\right)$ as in (13.4) with weighting matrix $\mathcal{A} = \left\{\text{diag}\left(\frac{x_{\bullet j}}{x_{\bullet\bullet}}\right)\right\}^{-1}$. Similarly, if we are interested in clusters among the columns, we can define:

$$d^2(j_1, j_2) = \sum_{i=1}^{n} \frac{1}{\left(\frac{x_{i\bullet}}{x_{\bullet\bullet}}\right)} \left(\frac{x_{ij_1}}{x_{\bullet j_1}} - \frac{x_{ij_2}}{x_{\bullet j_2}}\right)^2.$$

Apart from the Euclidean and the L_r-norm measures one can use a proximity measure such as the Q-correlation coefficient

$$d_{ij} = \frac{\sum_{k=1}^{p}(x_{ik} - \overline{x}_i)(x_{jk} - \overline{x}_j)}{\left\{\sum_{k=1}^{p}(x_{ik} - \overline{x}_i)^2 \sum_{k=1}^{p}(x_{jk} - \overline{x}_j)^2\right\}^{1/2}}. \tag{13.8}$$

Here $\overline{x}_i$ denotes the mean over the variables $(x_{i1}, \ldots, x_{ip})$.

Summary

- ↪ The proximity between data points is measured by a distance or similarity matrix $\mathcal{D}$ whose components d_{ij} give the similarity coefficient or the distance between two points x_i and x_j.
- ↪ A variety of similarity (distance) measures exist for binary data (e.g. Jaccard, Tanimoto, Simple Matching coefficients) and for continuous data (e.g. L_r-norms).
- ↪ The nature of the data could impose the choice of a particular metric $\mathcal{A}$ in defining the distances (standardisation, χ^2-metric etc.).

13.3 Cluster Algorithms

There are essentially two types of clustering methods: hierarchical algorithms and partitioning algorithms. The hierarchical algorithms can be divided into agglomerative and splitting procedures. The first type of hierarchical clustering starts from the finest partition possible (each observation forms a cluster) and groups them. The second type starts with the coarsest partition possible: one cluster contains all of the observations. It proceeds by splitting the single cluster up into smaller sized clusters.

The partitioning algorithms start from a given group definition and proceed by exchanging elements between groups until a certain score is optimised. The main difference between the two clustering techniques is that in hierarchical clustering once groups are found and elements are assigned to the groups, this assignment cannot be changed. In partitioning techniques, on the other hand, the assignment of objects into groups may change during the algorithm application.

Hierarchical Algorithms, Agglomerative Techniques

Agglomerative algorithms are used quite frequently in practice. The algorithm consists of the following steps:

Algorithm Hierarchical algorithms-agglomerative technique

1: Construct the finest partition
2: Compute the distance matrix $\mathcal{D}$.
3: **repeat**
4: Find the two clusters with the closest distance
5: Put those two clusters into one cluster
6: Compute the distance between the new groups and obtain a reduced distance matrix $\mathcal{D}$
7: **until** all clusters are agglomerated into $\mathcal{X}$

If two objects or groups say, P and Q, are united, one computes the distance between this new group (object) $P + Q$ and group R using the following distance function:

$$d(R, P + Q) = \delta_1 d(R, P) + \delta_2 d(R, Q) + \delta_3 d(P, Q) + \delta_4 |d(R, P) - d(R, Q)|. \tag{13.9}$$

The δ_j's are weighting factors that lead to different agglomerative algorithms as described in Table 13.2. Here $n_P = \sum_{i=1}^{n} \boldsymbol{I}(x_i \in P)$ is the number of objects in group P. The values of n_Q and n_R are defined analogously.

For the most common used Single and Complete linkages, below are the modified agglomerative algorithm steps:

As instead of computing new distance matrixes every step, a linear search in the original distance matrix is enough for clustering in the modified algorithm, it is more efficient in practice.

Table 13.2 Computations of group distances

Name	δ_1	δ_2	δ_3	δ_4
Single linkage	1/2	1/2	0	$-1/2$
Complete linkage	1/2	1/2	0	1/2
Average linkage (unweighted)	1/2	1/2	0	0
Average linkage (weighted)	$\frac{n_P}{n_P + n_Q}$	$\frac{n_Q}{n_P + n_Q}$	0	0
Centroid	$\frac{n_P}{n_P + n_Q}$	$\frac{n_Q}{n_P + n_Q}$	$-\frac{n_P n_Q}{(n_P + n_Q)^2}$	0
Median	1/2	1/2	$-1/4$	0
Ward	$\frac{n_R + n_P}{n_R + n_P + n_Q}$	$\frac{n_R + n_Q}{n_R + n_P + n_Q}$	$-\frac{n_R}{n_R + n_P + n_Q}$	0

Algorithm Modified hierarchical algorithms-agglomerative technique

1: Construct the finest partition
2: Compute the distance matrix $\mathcal{D}$.
3: **repeat**
4: Find the smallest (Single linkage)/ largest (Complete linkage) value d (between objects m and n) in $\mathcal{D}$
5: If m and n are not in the same cluster, combine the clusters m and n belonging to together, and delete the smallest value
6: **until** all clusters are agglomerated into $\mathcal{X}$ or the value d exceeds the preset level

Example 13.4 Let us examine the agglomerative algorithm for the three points in Example 13.2, $x_1 = (0, 0)$, $x_2 = (1, 0)$ and $x_3 = (5, 5)$, and the squared Euclidean distance matrix with single linkage weighting. The algorithm starts with $N = 3$ clusters: $P = \{x_1\}$, $Q = \{x_2\}$ and $R = \{x_3\}$. The distance matrix $\mathcal{D}_2$ is given in Example 13.2. The smallest distance in $\mathcal{D}_2$ is the one between the clusters P and Q. Therefore, applying step 4 in the above algorithm we combine these clusters to form $P + Q = \{x_1, x_2\}$. The single linkage distance between the remaining two clusters is from Table 13.2 and (13.9) equal to

$$\begin{aligned} d(R, P + Q) &= \frac{1}{2}d(R, P) + \frac{1}{2}d(R, Q) - \frac{1}{2}|d(R, P) - d(R, Q)| \\ &= \frac{1}{2}d_{13} + \frac{1}{2}d_{23} - \frac{1}{2} \cdot |d_{13} - d_{23}| \\ &= \frac{50}{2} + \frac{41}{2} - \frac{1}{2} \cdot |50 - 41| \\ &= 41. \end{aligned} \tag{13.10}$$

The reduced distance matrix is then $\begin{pmatrix} 0 & 41 \\ 41 & 0 \end{pmatrix}$. The next and last step is to unite the clusters R and $P + Q$ into a single cluster $\mathcal{X}$, the original data matrix.

When there are more data points than in the example above, a visualisation of the implication of clusters is desirable. A graphical representation of the sequence of clustering is called a *dendrogram*. It displays the observations, the sequence of clusters and the distances between the clusters. The vertical axis displays the indices of the points, whereas the horizontal axis gives the distance between the clusters. Large distances indicate the clustering of heterogeneous groups. Thus, if we choose to "cut the tree" at a desired level, the branches describe the corresponding clusters.

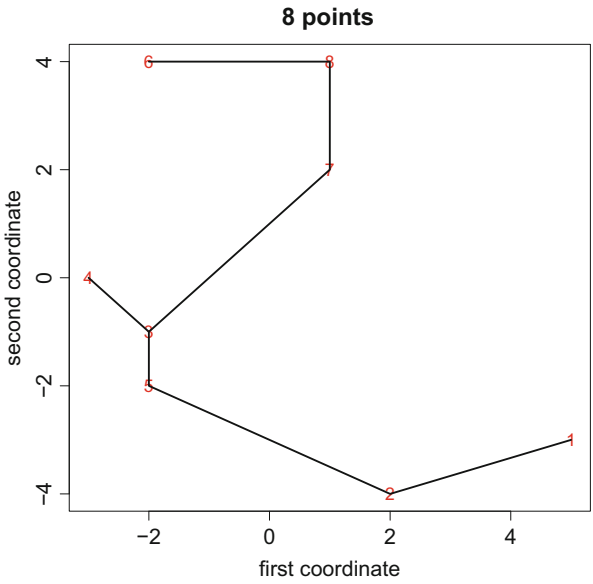

Fig. 13.1 The 8-point example MVAclus8p

Example 13.5 Here we describe the single linkage algorithm for the eight data points displayed in Fig. 13.1. The distance matrix (L_2-norms) is

$$\mathcal{D} = \begin{pmatrix} 0 & 10 & 53 & 73 & 50 & 98 & 41 & 65 \\ & 0 & 25 & 41 & 20 & 80 & 37 & 65 \\ & & 0 & 2 & 1 & 25 & 18 & 34 \\ & & & 0 & 5 & 17 & 20 & 32 \\ & & & & 0 & 36 & 25 & 45 \\ & & & & & 0 & 13 & 9 \\ & & & & & & 0 & 4 \\ & & & & & & & 0 \end{pmatrix}$$

and the dendrogram is shown in Fig. 13.2.

If we decide to cut the tree at the level 10, three clusters are defined: $\{1, 2\}$, $\{3, 4, 5\}$ and $\{6, 7, 8\}$.

The single linkage algorithm defines the distance between two groups as the smallest value of the individual distances. Table 13.2 shows that in this case

$$d(R, P + Q) = \min\{d(R, P), d(R, Q)\}. \tag{13.11}$$

This algorithm is also called the *Nearest Neighbour* algorithm. As a consequence of its construction, single linkage tends to build large groups. Groups that differ but are not well separated may thus be classified into one group as long as they have

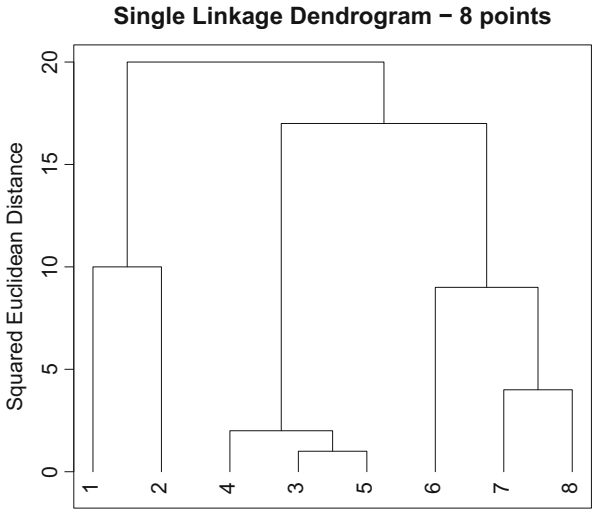

Fig. 13.2 The dendrogram for the 8-point example, single linkage algorithm `MVAclus8p`

two approximate points. The *complete linkage* algorithm tries to correct this kind of grouping by considering the largest (individual) distances. Indeed, the complete linkage distance can be written as

$$d(R, P + Q) = \max\{d(R, P),\ d(R, Q)\}. \tag{13.12}$$

It is also called the *Farthest Neighbour* algorithm. This algorithm will cluster groups where all the points are proximate, since it compares the largest distances. The *average linkage* algorithm (weighted or unweighted) proposes a compromise between the two preceding algorithms, in that it computes an average distance:

$$d(R, P + Q) = \frac{n_P}{n_P + n_Q} d(R, P) + \frac{n_Q}{n_P + n_Q} d(R, Q). \tag{13.13}$$

The *centroid* algorithm is quite similar to the average linkage algorithm and uses the natural geometrical distance between R and the weighted centre of gravity of P and Q (see Fig. 13.3):

$$d(R, P + Q) = \frac{n_P}{n_P + n_Q} d(R, P) + \frac{n_Q}{n_P + n_Q} d(R, Q) - \frac{n_P n_Q}{(n_P + n_Q)^2} d(P, Q). \tag{13.14}$$

The *Ward clustering* algorithm computes the distance between groups according to the formula in Table 13.2. The main difference between this algorithm and the linkage procedures is in the unification procedure. The Ward algorithm does not put together groups with smallest distance. Instead, it joins groups that do not increase a given measure of heterogeneity "too much". The aim of the Ward procedure is

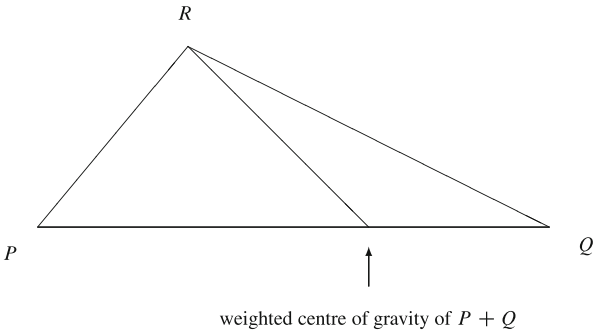

Fig. 13.3 The centroid algorithm

to unify groups such that the variation inside these groups does not increase too drastically: the resulting groups are as homogeneous as possible.

The heterogeneity of group R is measured by the inertia inside the group. This inertia is defined as follows:

$$I_R = \frac{1}{n_R} \sum_{i=1}^{n_R} d^2(x_i, \overline{x}_R) \tag{13.15}$$

where $\overline{x}_R$ is the centre of gravity (mean) over the groups. I_R clearly provides a scalar measure of the dispersion of the group around its centre of gravity. If the usual Euclidean distance is used, then I_R represents the sum of the variances of the p components of x_i inside group R.

When two objects or groups P and Q are joined, the new group $P + Q$ has a larger inertia I_{P+Q}. It can be shown that the corresponding increase of inertia is given by

$$\Delta(P, Q) = \frac{n_P n_Q}{n_P + n_Q} d^2(P, Q). \tag{13.16}$$

In this case, the Ward algorithm is defined as an algorithm that "joins the groups that give the smallest increase in $\Delta(P, Q)$". It is easy to prove that when P and Q are joined, the new criterion values are given by (13.9) along with the values of δ_i given in Table 13.2, when the centroid formula is used to modify $d^2(R, P + Q)$. So, the Ward algorithm is related to the centroid algorithm, but with an "inertial" distance Δ rather than the "geometric" distance d^2.

As pointed out in Sect. 13.2, all the algorithms above can be adjusted by the choice of the metric $\mathcal{A}$ defining the geometric distance d^2. If the results of a clustering algorithm are illustrated as graphical representations of individuals in spaces of low dimension (using principal components (normalised or not) or using a correspondence analysis for contingency tables), it is important to be coherent in the choice of the metric used.

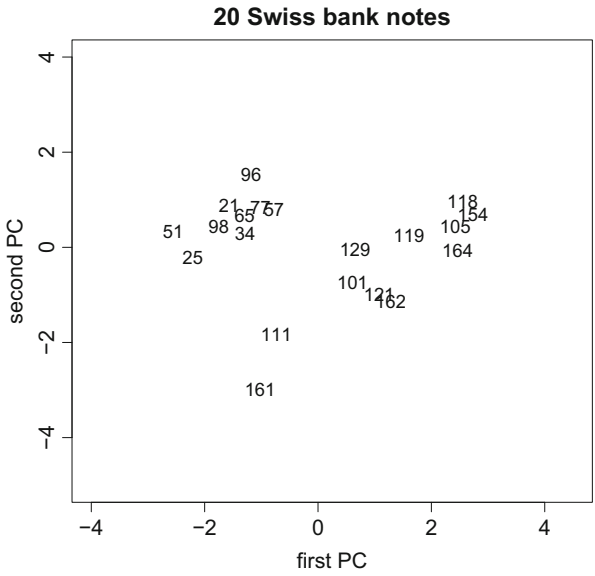

Fig. 13.4 PCA for 20 randomly chosen bank notes MVAclusbank

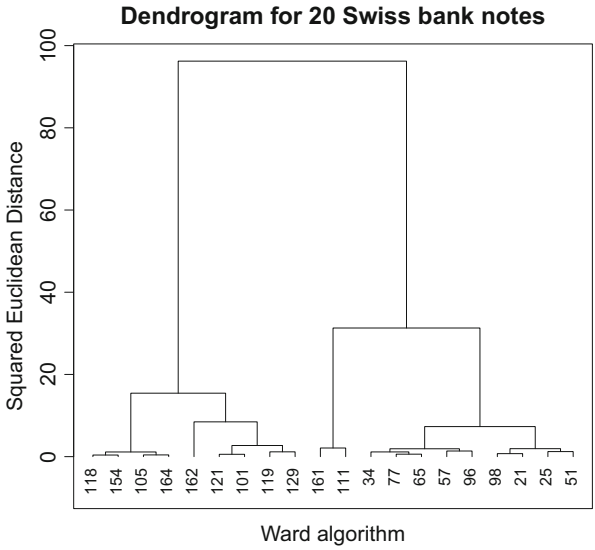

Fig. 13.5 The dendrogram for the 20 bank notes, Ward algorithm MVAclusbank

Example 13.6 As an example we randomly select 20 observations from the bank notes data and apply the Ward technique using Euclidean distances. Figure 13.4 shows the first two PCs of these data, Fig. 13.5 displays the dendrogram.

Example 13.7 Consider the French food expenditures. As in Chap. 11 we use standardised data which is equivalent to using $\mathcal{A} = \text{diag}(s_{X_1X_1}^{-1}, \ldots, s_{X_7X_7}^{-1})$ as the weight matrix in the L_2-norm. The NPCA plot of the individuals was given

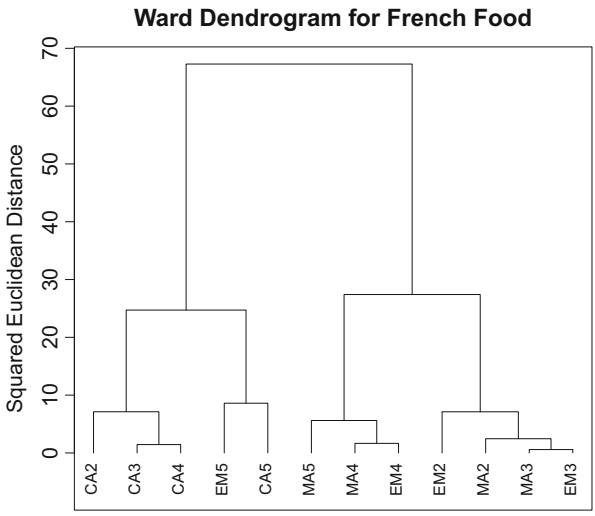

Fig. 13.6 The dendrogram for the French food expenditures, Ward algorithm
`MVAclusfood`

in Fig. 11.7. The Euclidean distance matrix is of course given by (13.6). The dendrogram obtained by using the Ward algorithm is shown in Fig. 13.6.

If the aim was to have only two groups, as can be seen in Fig. 13.6, they would be {CA2, CA3, CA4, CA5, EM5} and {MA2, MA3, MA4, MA5, EM2, EM3, EM4}. Clustering three groups is somewhat arbitrary (the levels of the distances are too similar). If we were interested in four groups, we would obtain {CA2, CA3, CA4}, {EM2, MA2, EM3, MA3}, {EM4, MA4, MA5} and {EM5, CA5}. This grouping shows a balance between socio-professional levels and size of the families in determining the clusters. The four groups are clearly well represented in the NPCA plot in Fig. 11.7.

Summary

↪ The class of clustering algorithms can be divided into two types: hierarchical and partitioning algorithms. Hierarchical algorithms start with the finest (coarsest) possible partition and put groups together (split groups apart) step by step. Partitioning algorithms start from a preliminary clustering and exchange group elements until a certain score is reached.

↪ Hierarchical agglomerative techniques are frequently used in practice. They start from the finest possible structure (each data point forms a cluster), compute the distance matrix for the clusters and join the clusters that have the smallest distance. This step is repeated until all points are united in one cluster.

Summary (continued)
↪ The agglomerative procedure depends on the definition of the distance between two clusters. Single linkage, complete linkage, and Ward distance are frequently used distances.
↪ The process of the unification of clusters can be graphically represented by a dendrogram.

13.4 Boston Housing

Presented multivariate techniques are now applied to the Boston Housing data. We focus our attention to 14 transformed and standardised variables, see e.g. Fig. 13.7 that provides descriptive statistics via boxplots for two clusters, as discussed in the

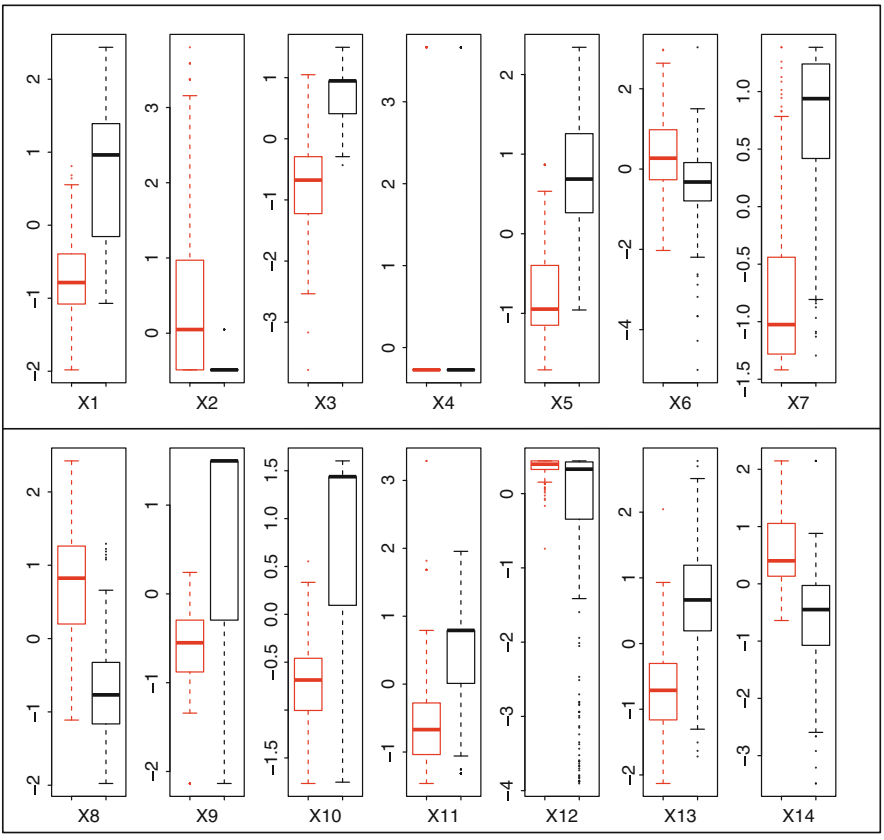

Fig. 13.7 Boxplots of the 14 standardised variables of the Boston housing data `MVAclusbh`

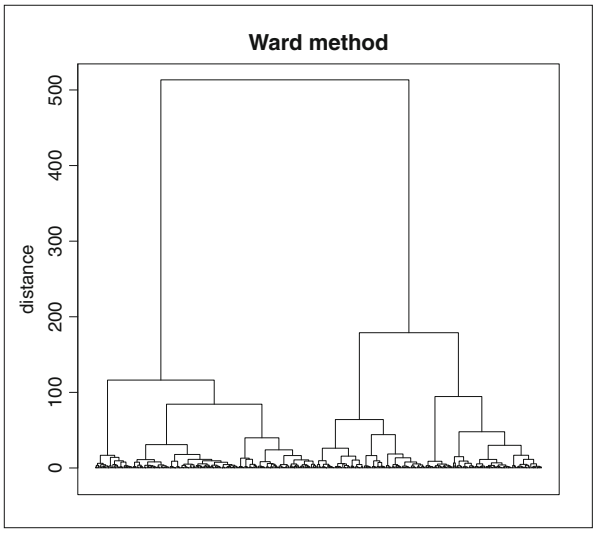

Fig. 13.8 Dendrogram of the Boston housing data using the Ward algorithm
`MVAclusbh`

Table 13.3 Means and standard errors of the 13 standardised variables for Cluster 1 (251 observations) and Cluster 2 (255 observations)
`MVAclusbh`

Variable	Mean C1	*SE* C1	Mean C2	*SE* C2
1	−0.7105	0.0332	0.6994	0.0535
2	0.4848	0.0786	−0.4772	0.0047
3	−0.7665	0.0510	0.7545	0.0279
5	−0.7672	0.0365	0.7552	0.0447
6	0.4162	0.0571	−0.4097	0.0576
7	−0.7730	0.0429	0.7609	0.0378
8	0.7140	0.0472	−0.7028	0.0417
9	−0.5429	0.0358	0.5344	0.0656
10	−0.6932	0.0301	0.6823	0.0569
11	−0.5464	0.0469	0.5378	0.0582
12	0.3547	0.0080	−0.3491	0.0824
13	−0.6899	0.0401	0.6791	0.0509
14	0.5996	0.0431	−0.5902	0.0570

sequel. A dendrogram for 13 variables(excluding the dummy variable $\tilde{X}_4$—Charles River indicator) using the Ward method is displayed in Fig. 13.8. One observes two dominant clusters. A further refinement of say, four clusters, could be considered at a lower level of distance.

To interpret the two clusters, we present the mean values and their respective standard errors of the 13 $\tilde{\mathcal{X}}$ variables by groups in Table 13.3. Comparison of the mean values for both groups shows that all the differences in the means are individually significant. Moreover, cluster one corresponds to housing districts with better living quality and higher house prices, whereas cluster two corresponds to less favored districts in Boston. This can be confirmed, for instance, by a lower crime rate, a higher proportion of residential land, lower proportion of African American,

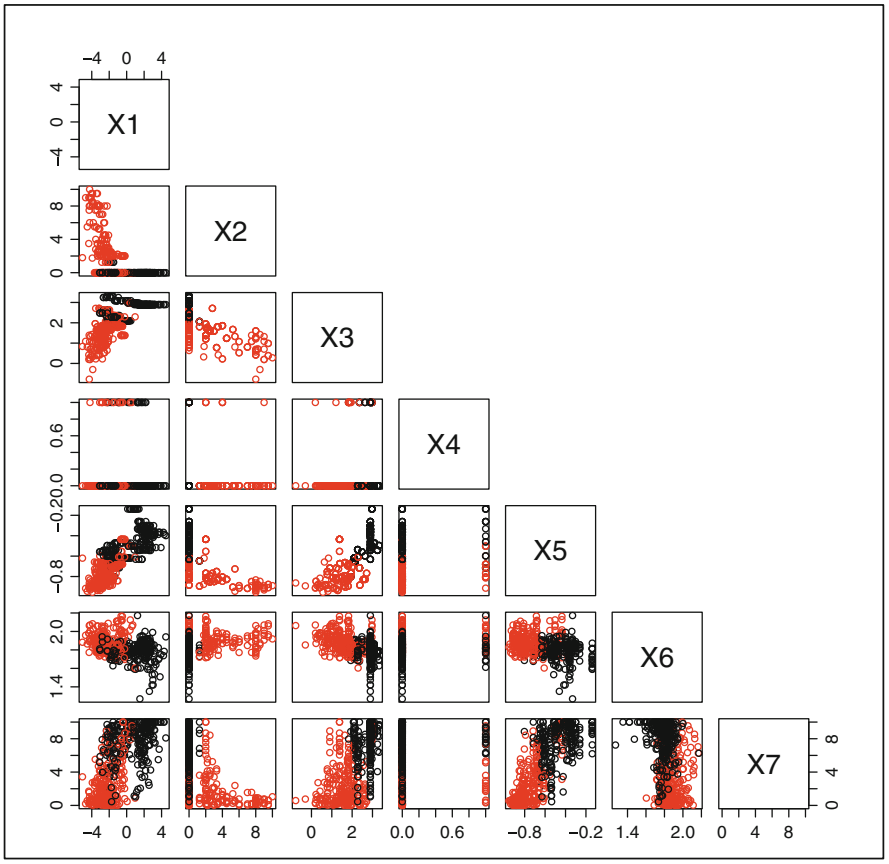

Fig. 13.9 Scatterplot matrix for variables $\tilde{X}_1$ to $\tilde{X}_7$ of the Boston housing data MVAclusbh

etc. for cluster one. Cluster two is identified by a higher proportion of older houses, a higher pupil/teacher ratio and a higher percentage of the lower status population.

This interpretation is underlined by visual inspection of all the variables via scatterplot matrices, see e.g. Figs. 13.9 and 13.10. For example, the lower right boxplot of Fig. 13.7 and the correspondingly coloured clusters in the last row of Fig. 13.10 confirm the role of each variable in determining the clusters. This interpretation perfectly coincides with the previous PC analysis (Fig. 11.11). The quality of life factor is clearly visible in Fig. 13.11, where cluster membership is

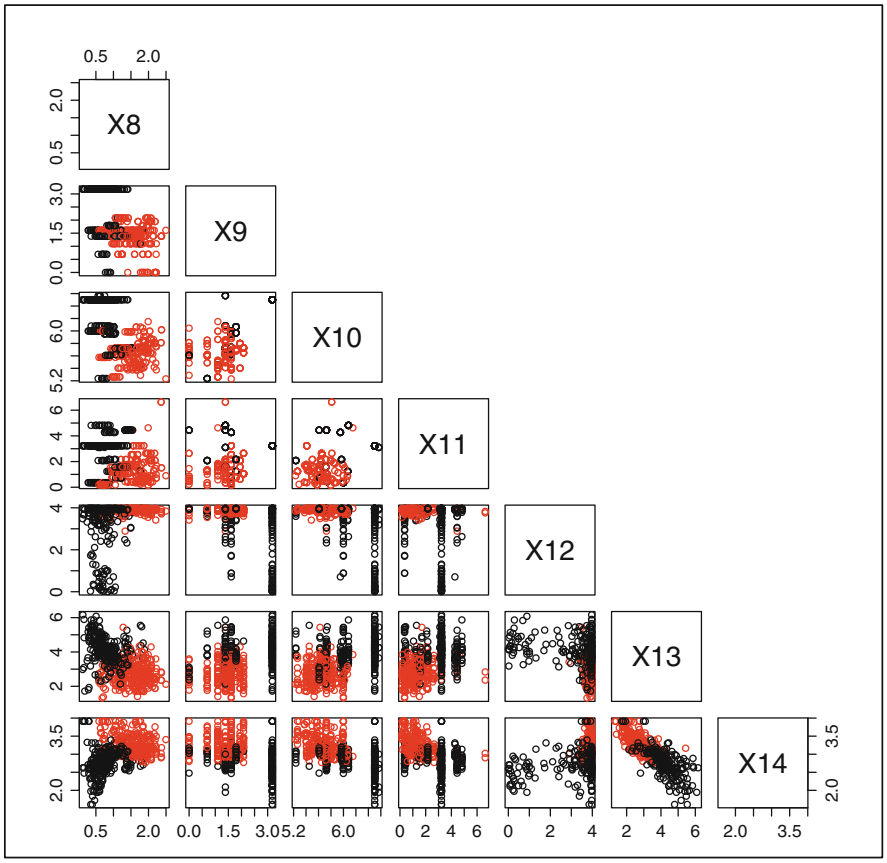

Fig. 13.10 Scatterplot matrix for variables $\tilde{X}_8$ to $\tilde{X}_{14}$ of the Boston housing data MVAclusbh

distinguished by the shape and colour of the points graphed according to the first two principal components. Clearly, the first PC completely separates the two clusters and corresponds, as we have discussed in Chap. 11, to a quality of life and house indicator.

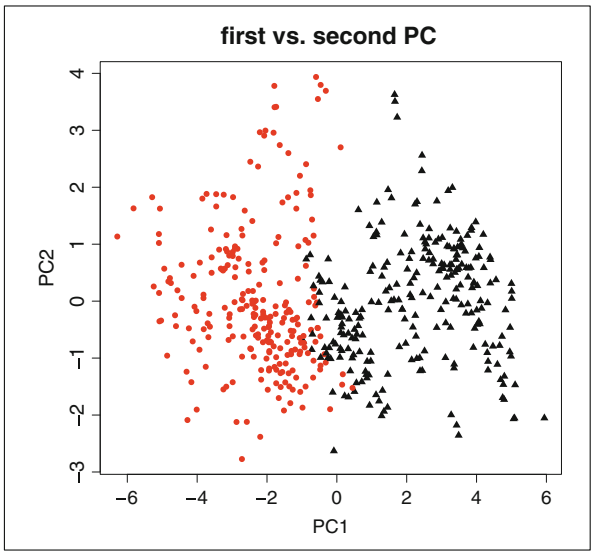

Fig. 13.11 Scatterplot of the first two PCs displaying the two clusters MVAclusbh

13.5 Exercises

Exercise 13.1 *Prove formula (13.16).*

Exercise 13.2 *Prove that* $I_R = \mathrm{tr}(\mathcal{S}_R)$*, where* $\mathcal{S}_R$ *denotes the empirical covariance matrix of the observations contained in* R*.*

Exercise 13.3 *Prove that*

$$\Delta(R, P + Q) = \frac{n_R + n_P}{n_R + n_P + n_Q}\,\Delta(R, P) + \frac{n_R + n_Q}{n_R + n_P + n_Q}\,\Delta(R, Q) - \frac{n_R}{n_R + n_P + n_Q}\,\Delta(P, Q),$$

when the centroid formula is used to define $d^2(R, P + Q)$*.*

Exercise 13.4 *Repeat the 8-point example (Example 13.5) using the complete linkage and the Ward algorithm. Explain the difference to single linkage.*

Exercise 13.5 *Explain the differences between various proximity measures by means of an example.*

Exercise 13.6 *Repeat the bank notes example (Example 13.6) with another random sample of 20 notes.*

Exercise 13.7 *Repeat the bank notes example (Example 13.6) with another clustering algorithm.*

Exercise 13.8 *Repeat the bank notes example (Example 13.6) or the 8-point example (Example 13.5) with the L_1-norm.*

Exercise 13.9 *Analyse the US companies example (Table 22.5) using the Ward algorithm and the L_2-norm.*

Exercise 13.10 *Analyse the US crime data set (Table 22.10) with the Ward algorithm and the L_2-norm on standardised variables (use only the crime variables).*

Exercise 13.11 *Repeat Exercise 13.10 with the US health data set (use only the number of deaths variables).*

Exercise 13.12 *Redo Exercise 13.10 with the χ^2-metric. Compare the results.*

Exercise 13.13 *Redo Exercise 13.11 with the χ^2-metric and compare the results.*