Chapter 7
Clustering

Clustering is used to classify the n cases into k groups. Unlike discriminant analysis, it is not known to which group the cases in the training data belong, and often the number of clusters k is unknown. Discriminant analysis is a type of supervised classification while clustering is a type of unsupervised classification. Factor analysis groups highly correlated variables X_j together (columns of the data matrix $\boldsymbol{W}$). Clustering groups cases $\boldsymbol{x}_i$ together (rows of the data matrix).

7.1 Hierarchical and k-Means Clustering

Two common methods of clustering are k-means clustering and hierarchical clustering. A wide variety of distances or similarities have been suggested. We will focus on Euclidean distances.

For the simplest version of k-means clustering, there are 4 steps.
1) Partition the n cases into k initial groups and find the means of each group. Alternatively, choose k initial seed points. These are groups of size 1 so the mean is equal to the seed point.
2) Compute distances between each case and each mean. Assign each case to the cluster whose mean is the nearest.
3) Recalculate the mean of each cluster.
4) Go to 2) and repeat until no more reassignments occur.

Two problems with k-means clustering are i) there could be more or less than k clusters, and ii) two initial means could belong to the same cluster. Then the resulting clusters may be poorly differentiated. It is often useful to run the k-means clustering program with several randomly drawn partitions or seeds, and to use several values of k.

Hierarchical clustering also has several steps. A distance is needed. Single linkage (or nearest neighbor) is the minimum distance between cases in cluster i and cases in cluster j. Complete linkage is the maximum distance between cases in cluster i and cases in cluster j. The average distance between clusters is also sometimes used.

1) Start with m $= n$ clusters. Each case forms a cluster. Compute the distance matrix for the n clusters. Let $d_{U,V}$ be the smallest distance. Combine clusters U and V into a single cluster and set $m = n - 1$.

2) Repeat step 1) with the new m. Continue until there is a single cluster.

3) Plot the resulting clusters as a dendrogram. Use the dendrogram to select k reasonable clusters of cases.

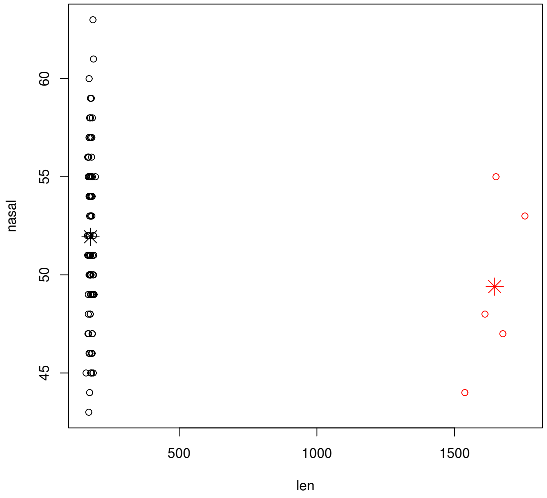

Fig. 7.1 Two Clusters From k-means Clustering With $k = 2$

Example 7.1. Often the clean data and outliers form two clusters. The R function kmeans was used on the Buxton (1920) data to produce Figure 7.1. See the R commands below.

```
x <- cbind(buxx,buxy)
out<-kmeans(x,2,nstart=25)
plot(x, col = out$cluster)
points(out$centers, col = 1:2, pch = 8, cex=2)
```

Using 5 clusters does not change the appearance of the plot much. Try the commands below.

```
out5<-kmeans(x,5,nstart=25)
plot(x, col = out5$cluster)
points(out5$centers, col = 1:5, pch = 8, cex=2)
```

Removing the outliers and trying 5 clusters seems to show one cluster. Try the commands below.

```
xc <-x[-c(61,62,63,64,65),]
out<-kmeans(xc,5,nstart=25)
plot(xc, col = out$cluster)
points(out$centers, col = 1:5, pch = 8, cex=2)
```

The following commands suggest that the clustering was done using values of buxy = height.

```
plot(xc[,c(1,5)],col = out$cluster)
points(out$centers[,c(1,5)],col=1:5,pch=8,cex=2)
```

Example 7.2. *R* functions for hierarchical clustering include `hclust` and `agnes`. See MathSoft (1999b, ch. 4) and Kaufman and Rousseeuw (1990, ch. 5). One problem with hierarchical clustering is that it can be hard to read the labels on the dendrogram unless n is small. The dendrogram for the Buxton (1920) data is shown in Figure 7.2. The very top of the dendrogram is a cluster containing all of the data. Then two clusters are formed, one containing the 5 outlying cases (the five cases furthest to the left on the bottom of the plot) and one cluster containing all of the remaining cases. Outliers often appear among the last clusters formed in the dendrogram, corresponding to the clusters near the top of the dendrogram.

```
x <- cbind(buxx,buxy)
out <- hclust(dist(x),"complete")
#complete is the default
plot(out)
plot(out,hang=-1)
```

Following James et al. (2014, pp. 391-392), to interpret the dendrogram, each *leaf* on the bottom of Figure 7.2 represents one of the 87 cases of the Buxton data. As we move up the tree, some leaves begin to fuse into branches corresponding to cases that are similar to each other. Moving further up the tree causes branches to fuse with other branches or leaves. The lower in the tree that the fusions occur, the more similar the group of cases are to each other. Cases that fuse near the top of the tree can be quite different. The outliers fused together quickly, and the clean cases fused together quickly. The outliers and clean cases fused together last since the outliers and clean cases are quite different.

Example 7.3. Following James et al. (2014, pp. 392-393), observations that are close together horizontally are not necessarily similar. Case 5 and 7 are similar and cases 1 and 6 are similar since they fuse together at the lowest

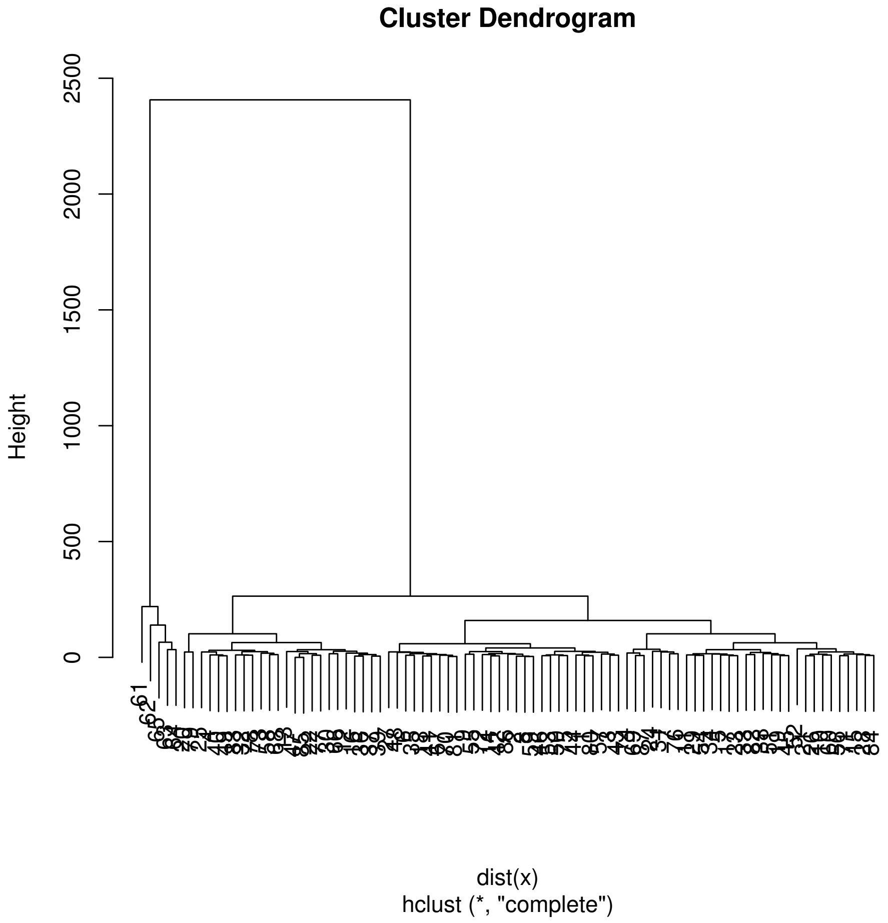

Fig. 7.2 Dendrogram for Buxton (1920) Data

points in the dendrogram shown in Figure 7.3. Cases 9 and 2 are located close together horizontally, cases 2, 5, 7, and 8 fuse with case 9 at the same height. Hence case 9 is about as similar to cases 5, 7, and 8 as case 9 is to case 2. Plot the raw data to help see this. See Problem 7.3. The height of the fusion determines similarity. A horizontal line at 1.5 gives two clusters, while a horizontal line at 1.0 gives 5 clusters: i) 1, 6, and 4; ii) 3; iii) 2; iv) 5, 7, and 8; and v) 9. See the *R* code shown below to produce Figure 7.3.

```
x1 <- c(-0.6,0.1,-1.5,-1.4,1.1,-0.9,1.4,0.6,0)
x2 <- c(-1,-0.75,-0.4,-1.6,-0.3,-1.2,0,-0.2,0.7)
x <- cbind(x1,x2)
```

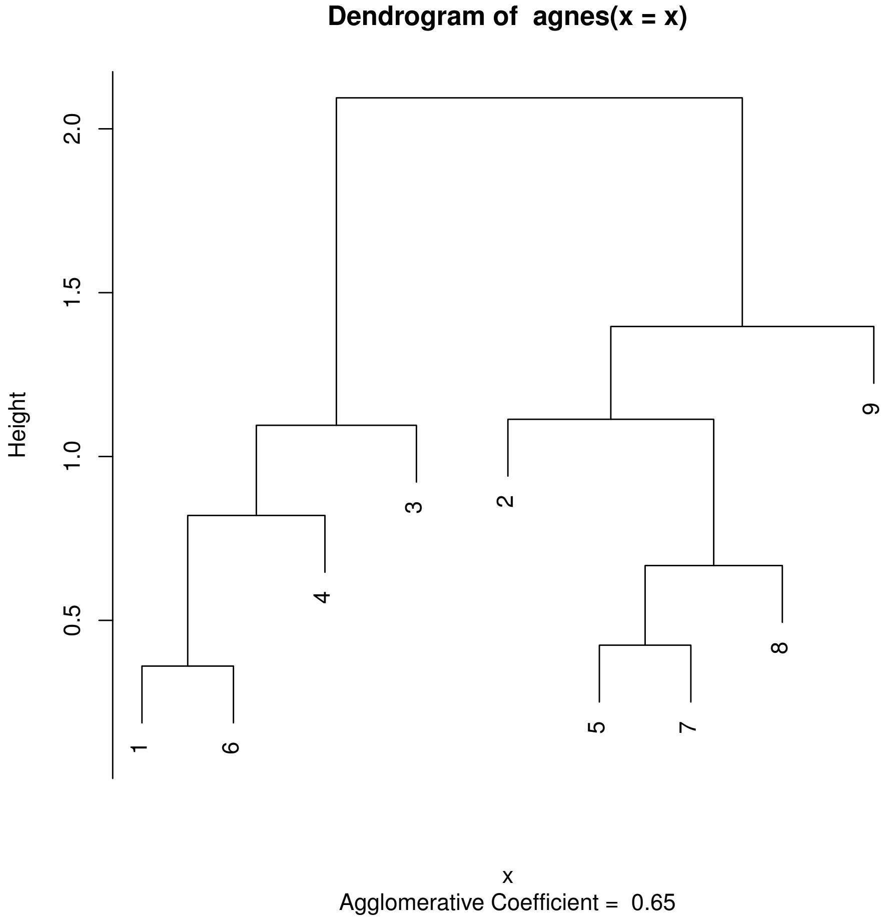

Fig. 7.3 9 and 2 are close in horizontal distance, but 2, 5, 7, and 8 fuse with 9 at the same height

```
##out<-hclust(x) #errors
out <- hclust(dist(x))
plot(out)
plot(x[,1],x[,2])
library(cluster)
out<-agnes(x)
plot(out) #right click twice
```

7.2 Complements

This chapter follows Olive (2017b, ch. 13) closely. Atkinson et al. (2004, ch. 7) has some interesting ideas. Also see Kaufman and Rousseeuw (1990), Farcomeni and Greco (2015), and Ritter (2014). A good review for robust methods is García-Escudero et al. (2010). For high dimensional clustering, see Jin and Wang (2016).

7.3 Problems

R Problems

For some of the following problems, the *R* commands can be copied and pasted from (http://parker.ad.siu.edu/Olive/slrhw.txt) into *R*.

7.1. Enter the commands for Example 7.1 to reproduce Figure 7.1.

7.2. Enter the commands for Example 7.2 to reproduce Figure 7.2.

7.3. Enter the commands for Example 7.3 to reproduce Figure 7.3. Also plot X_1 versus X_2 to see that case 9 is about as similar to case 2 as case 9 is to cases 5, 7, and 8.

7.4. a) Obtain the file `mbb1415.csv` from (http://parker.ad.siu.edu/Olive/slearnbk.htm), and save it on a flash drive (F, say). This file contains comma separated variables. The commands for this problem show how to read the file into *R*.

The file, obtained and analyzed by Nicole Staples and Philip Kains, contains variables on male basketball players from the Missouri Valley conference 2014–2015 season. The first variable $x_1 = position$ where 0 means position is unknown, 1 for guard, 2 for guard-forward, 3 for forward, 4 for forward-center, and 5 for center. The variable x_2 is games played, x_3 is number of minutes played, x_4 is sst (an efficiency rating), x_5 is sst.ex.pts (an efficiency rating excluding points), x_6 is points, x_7 is assists, x_8 is turnovers, x_9 is assists to turn over ratio, x_{10} is steals, x_{11} is stl.pos (stolen possessions, a ball handling rating), x_{12} is blocks, x_{13} is rebounds, x_{14} is offensive rebounds, x_{15} is defensive rebounds, x_{16} is games played $= x_2$, x_{17} is field goal (FG) attempts, x_{18} is field goals made, x_{19} is FGs missed, x_{20} is field goal percentage, x_{21} is adjusted field goal percentage, x_{22} is two point field goal attempts, x_{23} is two point field goals made, x_{24} is two point FGs missed, x_{25} is two point field goal percentage, x_{26} is three point field goal attempts, x_{27} is three point field goals made, x_{28} is three point FGs missed, x_{29} is three point field goal percentage, x_{30} is free throws attempted, x_{31} is free throws made, x_{32} is free throws missed, x_{33} is free throw percentage, x_{34} is related to the number of "and one plays" (free throw after a made shot), x_{35} is personal fouls taken, and x_{36} is personal fouls committed.

Note that $\boldsymbol{X}$ will not be full rank since, for example $x_{16} = x_2$, and offensive rebounds + defensive rebounds = rebounds.

b) Sometimes the classes are known and you want to see how well clustering works. The commands for this problem use assists and rebounds to form the clusters. The second dendrogram uses positions as labels. We would like each cluster to have one position or neighboring positions (all labels are i's or all labels are i's and $(i+1)$'s). Include the second plot in *Word*.

c) Many basketball players do not play much so all of their statistics are near zero (and could be regarded as near point mass outliers). The commands for this problem deletes about 25% of the players who had the fewest minutes, and then uses assists and rebounds to form the clusters. Include the plot in *Word*.

7.5. a) Obtain the file `wbb1415.csv` from (http://parker.ad.siu.edu/Olive/slearnbk.htm), and save it on a flash drive (F, say). This file contains comma separated variables. The commands for this problem show how to read the file into *R*.

The file, obtained and analyzed by Nicole Staples and Philip Kains, contains variables on female basketball players from the Missouri Valley conference 2014–2015 season.

The variables are almost the same as those in Problem 7.4. The only difference is that this file does not have two games played variables. Hence variables $x_1, ..., x_{15}$ are the same, but x_i for the `wbb1415` data set are variables x_{i+1} for the `mbb1415` data set for $i = 16, ..., 35$.

b) Sometimes the classes are known and you want to see how well clustering works. The commands for this problem use assists and rebounds to form the clusters. The second dendrogram uses positions as labels. We would like each cluster to have one position or neighboring positions (all labels are i's or all labels are i's and $(i+1)$'s). Include the second plot in *Word*.

c) Many basketball players do not play much so all of their statistics are near zero (and could be regarded as near point mass outliers). The commands for this problem deletes about 25% of the players who had the fewest minutes, and then uses assists and rebounds to form the clusters. Include the plot in *Word*.