6
Cluster Analysis

6.1 Introduction

Cluster analysis is a generic term for a wide range of numerical methods for examining multivariate data with a view to uncovering or discovering groups or clusters of observations that are homogeneous and separated from other groups. In medicine, for example, discovering that a sample of patients with measurements on a variety of characteristics and symptoms actually consists of a small number of groups within which these characteristics are relatively similar, and between which they are different, might have important implications both in terms of future treatment and for investigating the aetiology of a condition. More recently cluster analysis techniques have been applied to microarray data (Alon et al., 1999) and image analysis (Everitt and Bullmore, 1999).

Clustering techniques essentially try to formalize what human observers do so well in two or three dimensions. Consider, for example, the scatterplot shown in Figure 6.1. The conclusion that there are two natural groups or clusters of dots is reached with no conscious effort or thought. Clusters are identified by the assessment of the relative distances between points and, in this example, the relative homogeneity of each cluster and the degree of their separation makes the task relatively simple.

Detailed accounts of clustering techniques are available in Everitt et al. (2001) and Gordon (1999). Here we concentrate on three types of clustering procedures.

- Agglomerative hierarchical methods;
- K-means type methods;
- Classification maximum likelihood methods.

6.2 Agglomerative Hierarchical Clustering

In a hierarchical classification the data are not partitioned into a particular number of classes or clusters at a single step. Instead the classification consists of a series of partitions that may run from a single "cluster" containing all individuals, to n clusters each containing a single individual. Agglomerative hierarchical clustering

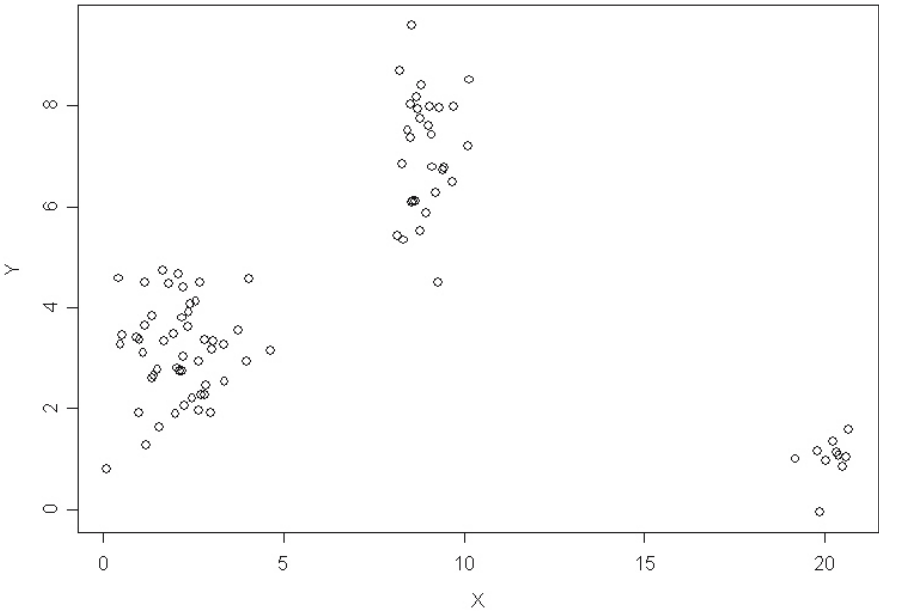

Figure 6.1 Bivariate data showing the presence of three clusters.

techniques produce partitions by a series of successive fusions of the n individuals into groups. Once made, however, such fusions are irreversible, so that when an agglomerative algorithm has placed two individuals in the same group, they cannot subsequently appear in different groups. Since all agglomerative hierarchical techniques ultimately reduce the data to a single cluster containing all the individuals, the investigator seeking the solution with the "best" fitting number of clusters will need to decide which division to choose. The problem of deciding on the "correct" number of clusters will be taken up later.

An agglomerative hierarchical clustering procedure produces a series of partitions of the data, P_n, P_{n-1}, ..., P_1. The first, P_n, consists of n single-member clusters, and the last, P_1, consists of a single group containing all n individuals. The basis operation of all methods is similar:

(START) Clusters $C_1, C_2, \ldots, C_n$ each containing a single individual.
(1) Find the nearest pair of distinct clusters, say C_i and C_j, merge C_i and C_j, delete C_j and decrease the number of clusters by one.
(2) If number of clusters equals one then stop, else return to 1.

At each stage in the process the methods fuse individuals or groups of individuals which are closest (or most similar). The methods begin with an interindividual distance matrix (e.g., one containing Euclidean distances as defined in Chapter 1), but as groups are formed, distance between an individual and a group containing several individuals or between two groups of individuals will need to be calculated. How such distances are defined leads to a variety of different techniques; see the next subsection.

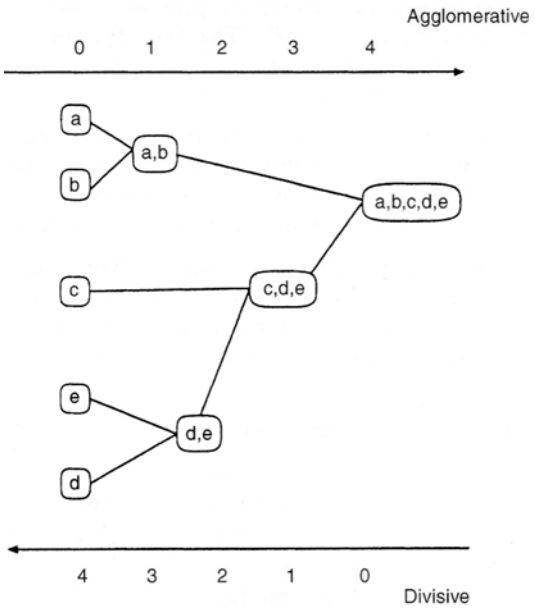

Figure 6.2 Example of a dendrogram. From *Finding Groups in Data: Introduction to Cluster Analysis*, Kaufman and Rousseeuw. Copyright © 1990. Reprinted with permission of John Wiley & Sons, Inc.

Hierarchic classifications may be represented by a two-dimensional diagram known as a *dendrogram*, which illustrates the fusions made a each stage of the analysis. An example of such a diagram is given in Figure 6.2. The structure of Figure 6.2 resembles an *evolutionary tree* (see Figure 6.3), and it is in biological applications that hierarchical classifications are most relevant and most justified

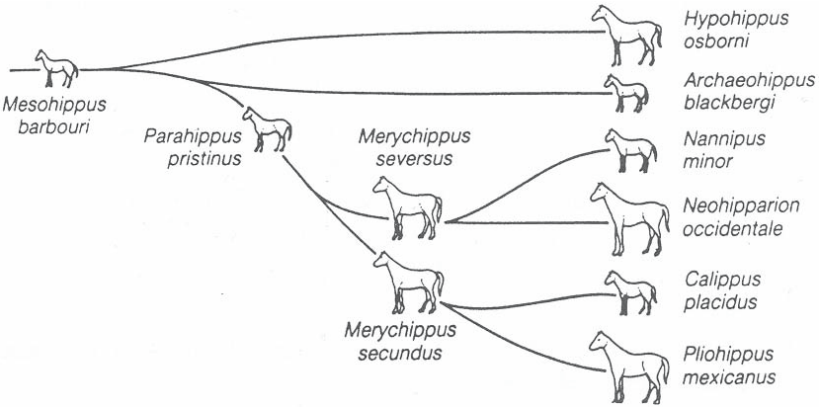

Figure 6.3 Evolutionary tree. From *Finding Groups in Data: Introduction to Cluster Analysis*, Kaufman and Rousseeuw. Copyright © 1990. Reprinted with permission of John Wiley & Sons, Inc.

(although this type of clustering has also been used in many other areas). According to Rohlf (1970), a biologist, "all things being equal," aims for a system of nested clusters. Hawkins et al. (1982), however, issue the following caveat: "users should be very wary of using hierarchic methods if they are not clearly necessary."

6.2.1 *Measuring Intercluster Dissimilarity*

Agglomerative hierarchical clustering techniques differ primarily in how they measure the distances between or similarity of two clusters (where a cluster may, at times, consist of only a single individual). Two simple intergroup measures are

$$d_{AB} = \min_{\substack{i \in A \\ i \in B}}(d_{ij}),$$
$$d_{AB} = \max_{\substack{i \in A \\ i \in B}}(d_{ij}),$$

where d_{AB} is the distance between two clusters A and B, and d_{ij} is the distance between individuals i and j. This could be Euclidean distance (see Chapter 1) or one of a variety of other distance measures; see Everitt et al., 2001, for details.

The first intergroup dissimilarity measure above is the basis of *single linkage* clustering, the second that of *complete linkage* clustering. Both these techniques have the desirable property that they are invariant under monotone transformations of the original interindividual dissimilarities or distances.

A further possibility for measuring intercluster distance or dissimilarity is

$$d_{AB} = \frac{1}{n_A n_B} \sum_{i \in A} \sum_{i \in B} d_{ij},$$

where n_A and n_B are the number of individuals in clusters A and B. This measure is the basis of a commonly used procedure known as *group average* clustering. All three intergroup measures described here are illustrated in Figure 6.4.

To illustrate the use of single linkage, complete linkage, and group average clustering we shall apply each method to the life expectancy data from the previous chapter (see Table 4.2). Here we assume that the eight life expectancies for each country are contained in the data frame `life` (see Chapter 4). The following R and S-PLUS code will calculate the Euclidean distance matrix for the countries, apply each of the clustering methods mentioned above, and then plot the resulting dendrograms, labelled with the country name:

R

```
#set up plotting area to take three side-by-side plots
country<-row.names(life)
par(mfrow=c(1,3))
#use dist to get Euclidean distance matrix, hclust to
#apply single linkage and plclust to plot dendrogram
plclust(hclust(dist(life),method="single"),
   labels=country,ylab="Distance")
```

```
title("(a) Single linkage")
plclust(hclust(dist(life),method="complete"),
   labels=country,ylab="Distance")
title("(b) Complete linkage")
plclust(hclust(dist(life),method="average"),
   labels=country,ylab="Distance")
title("(c) Average linkage")
```

S-PLUS

```
country<-row.names(life)
par(mfrow=c(1,3))
plclust(hclust(dist(life),method="connected"),
   labels=country,ylab="Distance")
title("(a) Single linkage")
plclust(hclust(dist(life),method="compact"),
   labels=country,ylab="Distance")
title("(b) Complete linkage")
plclust(hclust(dist(life),method="average"),
   labels=country,ylab="Distance")
title("(c) Average linkage")
```

The resulting diagram is shown in Figure 6.5.

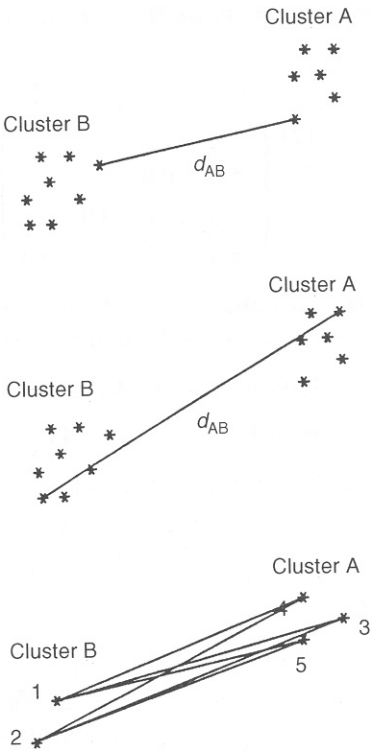

Figure 6.4 Intercluster distance measures.

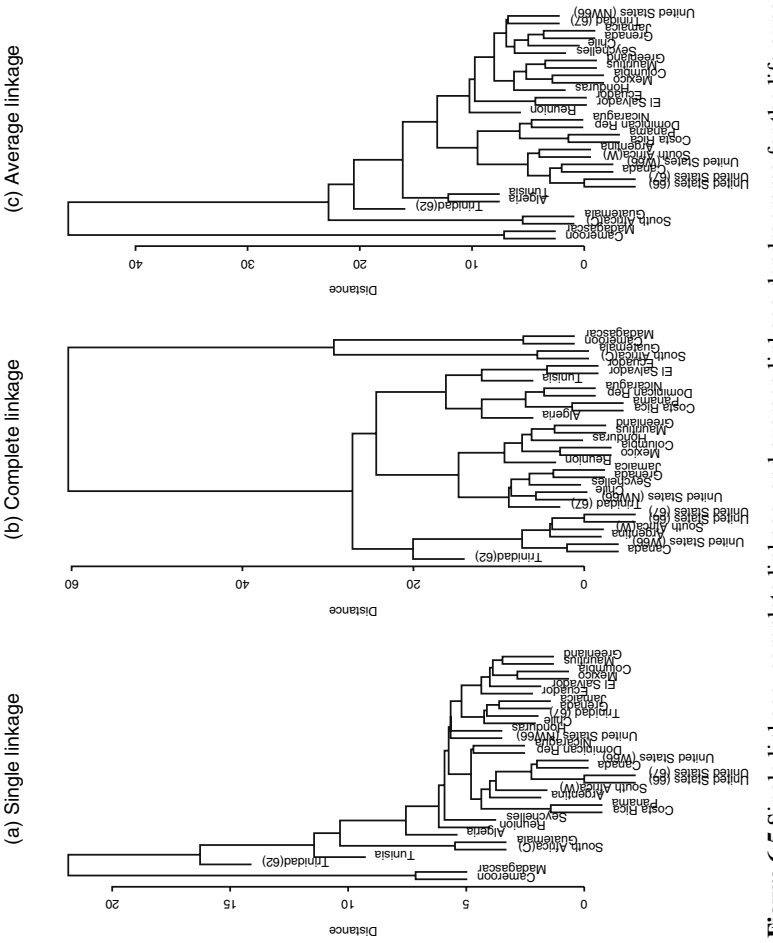

Figure 6.5 Single linkage, complete linkage, and average linkage dendrograms for the life expectancy data.

There are differences and similarities between the three dendrograms. Here we shall concentrate on the results given by complete linkage and we will examine the clustering found by "cutting" the complete linkage dendrogram at height 21 using the following R and S-PLUS® code:

R

```
four<-cutree(hclust(dist(life),method="complete"),h=21)
```

S-PLUS

```
four<-cutree(hclust(dist(life),method="compact"),h=21)
```

The resulting clusters in terms of country labels can be found from

```
#
country.clus<-lapply(1:5,function(nc)country[four==nc])
country.clus
```

The results from S-PLUS are shown in Table 6.1. (The group order differs in R, although the groups are the same.)

The means for the countries in each cluster can be found as follows:

```
country.mean<-lapply(1:5,function(nc)
   apply(life[four==nc,],2,mean))
country.mean
```

The results for the S-PLUS order of clusters are shown in Table 6.2. The S-PLUS clusters can be shown on a scatterplot matrix of the data using

```
pairs(life,panel=function(x,y) text(x,y,four))
```

The resulting plot is shown in Figure 6.6. This diagram suggests that the evidence for five distinct clusters in the data is not convincing.

Table 6.1 Clustering Solution from Complete Linkage

Cluster 1
South Africa (W), Canada, Trinidad (62), USA (66)
USA (W66), USA (67), Argentina
Cluster 2
Algeria, Tunisia, Costa Rica, Dominican Republic
El Salvador, Nicaragua, Panama, Ecuador
Cluster 3
Mauritius, Reunion, Seychelles, Greenland
Grenada, Honduras, Jamaica, Mexico
Trinidad (67), USA (NW66), Chile, Columbia
Cluster 4
Cameroon, Madagascar
Cluster 5
South Africa (C), Guatemala

Table 6.2 Mean Life Expectancies for the Five Clusters from Complete Linkage

	m0	m25	m50	m75	w0	w25	w50	w75
Cluster 1	66.4	48.0	22.9	7.9	72.7	50.7	27.7	9.7
Cluster 2	61.4	47.6	26.9	10.8	65.0	50.8	29.3	12.6
Cluster 3	60.1	42.8	22.0	7.6	64.9	46.8	25.3	9.7
Cluster 4	36.0	29.5	15.0	6.0	38.0	33.0	18.5	6.5
Cluster 5	49.5	39.5	21.0	8.0	53.0	42.0	23.0	8.0

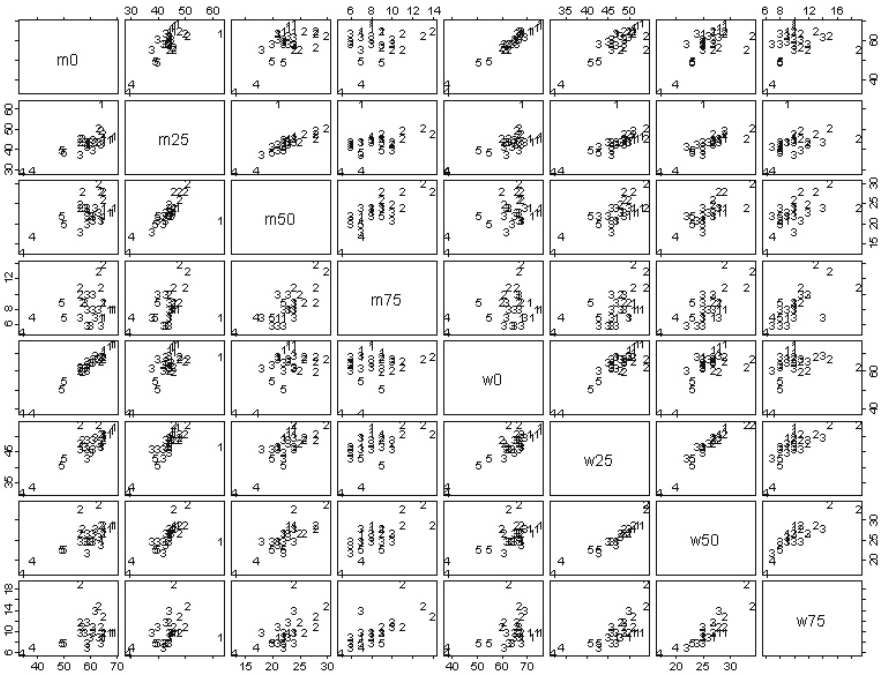

Figure 6.6 Scatterplot of life expectancy data showing five cluster solution from complete linkage.

6.3 *K*-Means Clustering

The *k*-means clustering technique seeks to partition a set of data into a specified number of groups, *k*, by minimizing some numerical criterion, low values of which are considered indicative of a "good" solution. The most commonly used approach, for example, is to try to find the partition of the *n* individuals into *k* groups, which minimizes the within-group sum of squares over all variables. The problem then appears relatively simple; namely, consider every possible partition of the *n* individuals into *k* groups, and select the one with the lowest within-groupsum of squares.

Unfortunately, the problem in practice is not so straightforward. The numbers involved are so vast that complete enumeration of *every* possible partition remains impossible even with the fastest computer. To illustrate the scale of the problem:

n	k	Number of possible partitions
15	3	2, 375, 101
20	4	45, 232, 115, 901
25	8	690, 223, 721, 118, 368, 580
100	5	10^{68}

The impracticability of examining every possible partition has led to the development of algorithms designed to search for the minimum values of the clustering criterion by rearranging existing partitions and keeping the new one only if it provides an improvement. Such algorithms do not, of course, guarantee finding the global minimum of the criterion. The essential steps in these algorithms are as follows:

1. Find some initial partition of the individuals into the required number of groups. (Such an initial partition could be provided by a solution from one of the hierarchical clustering techniques described in the previous section.)
2. Calculate the change in the clustering criterion produced by "moving" each individual from its own to another cluster.
3. Make the change that leads to the greatest improvement in the value of the clustering criterion.
4. Repeat steps (2) and (3) until no move of an individual causes the clustering criterion to improve.

To illustrate the *k*-means approach with minimization of the within-clusters sum of squares criterion we shall apply it to the data shown in Table 6.3 which shows the chemical composition of 48 specimens of Romano-British pottery, determined by atomic absorption spectrophotometry, for nine oxides (Tubb et al., 1980).

Because the variables are on very different scales they will need to be standardized in some way before applying *k*-means clustering. In what follows we will divide each variable's values by the range of the variable. Assuming that the data are contained in a matrix `pottery.data`, this standardization can be applied in R and S-PLUS as follows:

```
rge<-apply(pottery.data,2,max)-apply(pottery.data,2,min)
pottery.dat<-sweep(pottery.data,2,rge,FUN="/")
```

The *k*-means approach can be used to partition the states into a prespecified number of clusters set by the investigator. In practice, solutions for a range of values for number of groups are found, but the question remains as to the "optimal" number of clusters for the data. A number of suggestions have been made as to how to tackle this question (see Everitt et al., 2001), but none is completely satisfactory. Here, we shall examine the value of the within-group sum of squares associated

Table 6.3 Results of Chemical Analyses of Romano British Pottery from Tubb et al. (1980) reprinted by kind permission of Blackwell Publishing

No	Kiln	Al_2O_3	Fe_2O_3	MgO	CaO	Na_2O	K_2O	TiO_2	MnO	BaO
1	1	18.8	9.52	2.00	0.79	0.40	3.20	1.01	0.077	0.015
2	1	16.9	7.33	1.65	0.84	0.40	3.05	0.99	0.067	0.018
3	1	18.2	7.64	1.82	0.77	0.40	3.07	0.98	0.087	0.014
4	1	16.9	7.29	1.56	0.76	0.40	3.05	1.00	0.063	0.019
5	1	17.8	7.24	1.83	0.92	0.43	3.12	0.93	0.061	0.019
6	1	18.8	7.45	2.06	0.87	0.25	3.26	0.98	0.072	0.017
7	1	16.5	7.05	1.81	1.73	0.33	3.20	0.95	0.066	0.019
8	1	18.0	7.42	2.06	1.00	0.28	3.37	0.96	0.072	0.017
9	1	15.8	7.15	1.62	0.71	0.38	3.25	0.93	0.062	0.017
10	1	14.6	6.87	1.67	0.76	0.33	3.06	0.91	0.055	0.012
11	1	13.7	5.83	1.50	0.66	0.13	2.25	0.75	0.034	0.012
12	1	14.6	6.76	1.63	1.48	0.20	3.02	0.87	0.055	0.016
13	1	14.8	7.07	1.62	1.44	0.24	3.03	0.86	0.080	0.016
14	1	17.1	7.79	1.99	0.83	0.46	3.13	0.93	0.090	0.020
15	1	16.8	7.86	1.86	0.84	0.46	2.93	0.94	0.94	0.20
16	1	15.8	7.65	1.94	0.81	0.83	3.33	0.96	0.112	0.019
17	1	18.6	7.85	2.33	0.87	0.39	3.17	0.98	0.081	0.018
18	1	16.9	7.87	1.83	1.31	0.53	3.09	0.95	0.092	0.023
19	1	18.9	7.58	2.05	0.83	0.13	3.29	0.98	0.072	0.015
20	1	18.0	7.50	1.94	0.69	0.12	3.14	0.93	0.035	0.017
21	1	17.8	7.28	1.92	0.81	0.18	3.15	0.90	0.067	0.017
22	2	14.4	7.00	4.30	0.15	0.51	4.25	0.79	0.160	0.019
23	2	13.8	7.08	3.43	0.12	0.17	4.14	0.77	0.144	0.020
24	2	14.6	7.09	3.88	0.13	0.20	4.36	0.81	0.124	0.019
25	2	11.5	6.37	5.64	0.16	0.14	3.89	0.69	0.087	0.009
26	2	13.8	7.06	5.34	0.20	0.20	4.31	0.71	0.101	0.021
27	2	10.9	6.26	3.47	0.17	0.22	3.40	0.66	0.109	0.010
28	2	10.1	4.26	4.26	0.20	0.18	3.32	0.59	0.149	0.017
29	2	11.6	5.78	5.91	0.18	0.16	3.70	0.65	0.082	0.015
30	2	11.1	5.49	4.52	0.29	0.30	4.03	0.63	0.080	0.016
31	2	13.4	6.92	7.23	0.28	0.20	4.54	0.69	0.163	0.017
32	2	12.4	6.13	5.69	0.22	0.54	4.65	0.70	0.159	0.015
33	2	13.1	6.64	5.51	0.31	0.24	4.89	0.72	0.094	0.017
34	3	11.6	5.39	3.77	0.29	0.06	4.51	0.56	0.110	0.015
35	3	11.8	5.44	3.94	0.30	0.04	4.64	0.59	0.085	0.013

(*Continued*)

Table 6.3 (*Continued*)

No	Kiln	Al_2O_3	Fe_2O_3	MgO	CaO	Na_2O	K_2O	TiO_2	MnO	BaO
36	4	18.3	1.28	0.67	0.03	0.03	1.96	0.65	0.001	0.014
37	4	15.8	2.39	0.63	0.01	0.04	1.94	1.29	0.001	0.014
38	4	18.0	1.50	0.67	0.01	0.06	2.11	0.92	0.001	0.016
39	4	18.0	1.88	0.68	0.01	0.04	2.00	1.11	0.006	0.022
41	4	20.8	1.51	0.72	0.07	0.10	2.37	1.26	0.002	0.016
42	5	17.7	1.12	0.56	0.06	0.06	2.06	0.79	0.001	0.013
43	5	18.3	1.14	0.67	0.06	0.05	2.11	0.89	0.006	0.019
44	5	16.7	0.92	0.53	0.01	0.05	1.76	0.91	0.004	0.013
45	5	14.8	2.74	0.67	0.03	0.05	2.15	1.34	0.003	0.015
46	5	19.1	1.64	0.60	0.10	0.03	1.75	1.04	0.007	0.018

with solutions for a range of values of k, the number of groups. As k increases this value will necessarily decrease but some "sharp" change may be indicative of the best solution. To obtain a plot of the within-group sum of squares for the one to six group solutions we can use the following R and S-PLUS code:

```
n<-length(pottery.dat[,1])
#find within group ss for all the data
wss1<-(n-1)*sum(apply(pottery.dat,2,var))
wss<-numeric(0)
#calculate within group ss for 2 to 6 group partitions given
   by k-means clustering
for(i in 2:6) {
       W<-sum(kmeans(pottery.dat,i)$withinss)
       wss<-c(wss,W)
}
wss<-c(wss1,wss)
plot(1:6,wss,type="l",xlab="Number of groups",
   ylab="Within groups sum of squares",lwd=2)
```

The resulting diagram is shown in Figure 6.7. The plot suggests looking at the two- or three-cluster solution. Details of the latter can be obtained using

```
pottery.kmean <- kmeans(pottery.dat, 3)
pottery.kmean
```

The output is shown in Table 6.4. The means from the code above are for the standardized data; to get the cluster means for the raw data we can use

```
lapply(1:3,function(nc)
   apply(pottery.dat[pottery.kmeans$cluster==nc,],2,mean))
```

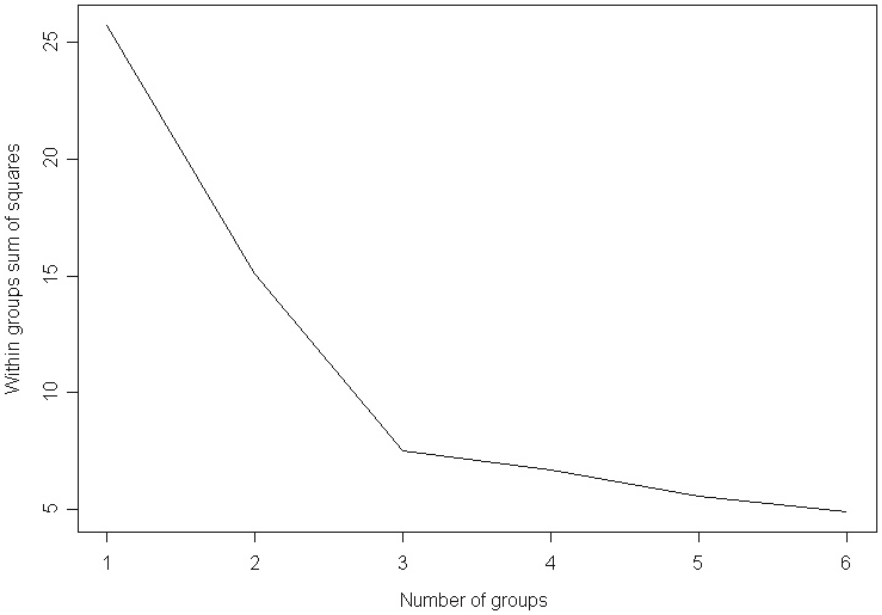

Figure 6.7 Plot of within-cluster sum of squares against number of clusters.

These means are also shown in Table 6.4. The means of each of the nine variables for each of the three clusters show that:

- Cluster three is characterized by a high aluminium oxide value and low iron oxide and calcium oxide values.
- Cluster two has a very high manganese oxide value and a high potassium oxide value.
- Cluster one has high calcium oxide value.

In addition to the chemical composition of the pots, the kiln site at which the pottery was found is known for these data (see Table 6.3). An archaeologist might be interested in assessing whether there is any association between the site and the distinct compositional groups found by the cluster analysis. To look at this we can cross-tabulate the kiln site against cluster label as follows:

```
table(kiln,pottery.kmean$cluster)
```

The resulting cross classification is shown in Table 6.5. Cluster 1 contains all 21 pots from kiln number one, cluster 2 contains pots from kilns 2 and 3, and cluster 3 pots from kilns 4 and 5. In fact, the five kiln sites are from three different regions defined by 1, (2, 3), (4, 5), so the clusters actually correspond to pots from three different regions.

Table 6.4 Details of Three-Group Solution for the Pottery Data

Means for standardized data

```
Centers:
          AL2O3       FE2O3         MGO         CAO        NA2O        K2O        TIO2         MNO        BAO
[1,]     1.5812     0.86379    0.274982    0.545958     0.43214    0.98817     1.20208    0.439153     1.2245
[2,]     1.1622     0.72184    0.713113    0.124585     0.28214    1.33371     0.87546    0.726190     1.1378
[3,]     1.6589     0.18744    0.095522    0.022674     0.06375    0.64363     1.30769    0.019753     1.1429

Clustering vector:
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3
[38] 3 3 3 3 3 3 3 3

Within cluster sum of squares:
[1] 3.1644 2.8748 1.4667

Cluster sizes:
[1] 21 14 10
```

Means for original data

```
  [[1]]:
   AL2O3       FE2O3         MGO         CAO        NA2O        K2O        TIO2         MNO        BAO
16.91905    7.428571    1.842381   0.9390476   0.3457143   3.102857    0.937619   0.07114286   0.01714286

  [[2]]:
   AL2O3       FE2O3         MGO         CAO        NA2O        K2O        TIO2         MNO        BAO
12.43571    6.207857    4.777857   0.2142857   0.2257143   4.187857   0.6828571   0.1176429   0.01592857

  [[3]]:
   AL2O3       FE2O3         MGO         CAO        NA2O        K2O        TIO2         MNO        BAO
   17.75       1.612        0.64       0.039       0.051       2.021       1.02      0.0032       0.016
```

Table 6.5 Cross-Tabulation of Cluster Label and Kiln

	1	2	3
1	21	0	0
2	0	12	0
3	0	2	0
4	0	0	5
5	0	0	5

6.4 Model-Based Clustering

The agglomerative hierarchical and k-means clustering methods described in the previous two sections are based largely in heuristic but intuitively reasonable procedures. But they are not based on formal models—those making problems such as deciding on a particular method, estimating the number of clusters, etc., particularly difficult. And, of course, without a reasonable model, formal inference is precluded. In practice, these may not be insurmountable objections to the use of the techniques since cluster analysis is essentially an "exploratory" tool. But model-based cluster methods do have some advantages, and a variety of possibilities have been proposed. The most successful approach has been that proposed by Scott and Symons (1971) and extended by Banfield and Raftery (1993) and Fraley and Raftery (2002), in which it is assumed that the population from the population from which the observations arise consists of c subpopulations, each corresponding to a cluster, and that the density of a q-dimensional observation from the jth subpopulation is $f_j(\mathbf{x}, \boldsymbol{\theta}_j)$ for some unknown vector of parameters, $\boldsymbol{\theta}_j$. They also introduce a vector $\boldsymbol{\gamma}' = [\gamma_1, \ldots, \gamma_n]$, where $\gamma_i = k$ if x_i is from the kth subpopulation; the γ_i label the subpopulation of each observation. The clustering problem now becomes that of choosing $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_c)$ and γ to maximize the likelihood function associated with such assumptions. This classification maximum likelihood procedure is described briefly in Display 6.1.

Display 6.1
Classification Maximum Likelihood

- Assume the population consists of c subpopulations, each corresponding to a cluster of observations, and that the density function of a q-dimensional observation from the jth subpopulation is $f_i(\mathbf{x}; \boldsymbol{\theta}_j)$ for some unknown vector of parameters, $\boldsymbol{\theta}_j$.
- Also, assume that $\boldsymbol{\gamma}' = [\gamma_1, \ldots, \gamma_n]$ gives the labels of the subpopulation to which each observation belongs. So $\gamma_i = j$ if $\mathbf{x}_i$ is from the jth population.

- The clustering problem becomes that of choosing $\boldsymbol{\theta}' = [\theta_1, \theta_2, \ldots, \theta_c]$ and $\boldsymbol{\gamma}$ to maximize the likelihood

$$L(\boldsymbol{\theta}, \boldsymbol{\gamma}) = \prod_{i=1}^{n} f_{\gamma_i}(\mathbf{x}_i; \boldsymbol{\theta}_{\gamma_i})$$

- If $f_j(\mathbf{x}; \boldsymbol{\theta}_j)$ is taken as a multivariate normal density with mean vector $\boldsymbol{\mu}_j$ and covariance matrix $\boldsymbol{\Sigma}_j$ this likelihood has the form

$$L(\boldsymbol{\theta}; \boldsymbol{\gamma}) = \text{const} \prod_{k=1}^{c} \prod_{i \in E_k} |\Sigma_k|^{1/2} \exp\left\{ -\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_k)' \sum\nolimits_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) \right\}$$

  where $E_j = \{i : \gamma_i = j\}$.
- The maximum likelihood estimator of $\boldsymbol{\mu}_j$ is $\bar{\mathbf{x}}_j = n_j^{-1} \Sigma_{i \in E_j} \mathbf{x}_i$ where n_j is the number of elements in E_j. Replacing $\boldsymbol{\mu}_j$ in (2) with this maximum likelihood estimator yields the following log-likelihood:

$$l(\theta, \gamma) = \text{const} - \frac{1}{2} \sum_{i=1}^{c} \text{trace}(\mathbf{W}_j \boldsymbol{\Sigma}_j^{-1} + n \log |\boldsymbol{\Sigma}_j|)$$

  where $\mathbf{W}_j$ is the $p \times p$ matrix of sums of squares and cross-products of the variables for subpopulation j.
- Banfield and Raftery (1992) demonstrate the following:

  1. If $\boldsymbol{\Sigma}_k = \sigma^2 \mathbf{I}$ $(k = 1, 2, \ldots, c)$, then the likelihood is maximized by choosing γ to minimize trace $(\mathbf{W})$, where $\mathbf{W} = \boldsymbol{\Sigma}_{k=1}^{c} \mathbf{W}_k$, that is, minimization of the written group sum of squares. Use of this criterion in a cluster analysis will tend to produce spherical clusters of largely equal sizes.
  2. If $\boldsymbol{\Sigma}_k = \boldsymbol{\Sigma} (k = 1, 2, \ldots, c)$, then the likelihood is maximized by choosing γ to minimize $|\mathbf{W}|$, a clustering criterion discussed by Friedman and Rubin (1967) and Mariott (1982). Use of this criterion in a cluster analysis will tend to produce clusters with the same elliptical slope.
  3. If $\boldsymbol{\Sigma}_k$ is not constrained, the likelihood is maximized by choosing γ to minimize $\boldsymbol{\Sigma}_{k=1}^{c} n_k \log |\mathbf{W}_k / n_k|$.

- Banfield and Raftery (1992) also consider criteria that allow the shape of clusters to be less constrained than with the minimization of trace $(\mathbf{W})$ and $|\mathbf{W}|$ criteria, but which remain more parsimonious then the completely unconstrained model. For example, constraining clusters to be spherical but not to have the same volume, or constraining clusters to have diagonal covariance matrices but allowing their shapes, sizes, and orientations to vary.

- The EM algorithm (see Dempster et al., 1977), is used for the maximum likelihood estimation; details are given in Fraley and Raftery (2002).
- Model selection is a combination of choosing the appropriate clustering model and the optimal number of clusters. A Bayesian approach is used (see Fraley and Raftery, 2002), using what is known as the *Bayesian Information Criterion* (BIC).

To illustrate this approach to clustering, we shall apply it to the data shown in Table 6.6. These data, taken with permission from Mayor and Frei (2003) give the values of three variables for the exoplanets discovered up to October 2002 (an exoplanet is a planet located outside the solar system). We assume the data are available as the data frame `planet.dat`.

R and S-PLUS functions for model-based clustering are available at http://www.stat.washington.edu/mclust. In R, the package can be installed from CRAN and then loaded in the usual way. Here we use the `Mclust` function since this selects both the most appropriate model for the data *and* the optimal number of groups based on the values of the BIC (see Display 6.1) computed over several models and a range of values for number of groups. The necessary code is

```
library(mclust)
planet.clus<-Mclust(planet.dat)
```

We can first examine a plot of BIC values using

```
plot(planet.clus,planet.dat)
```

and selecting the BIC option (option number 1). The resulting diagram is shown in Figure 6.8. In this diagram the numbers refer to different model assumptions about the shape of clusters:

1. Spherical, equal volume;
2. Spherical, unequal volume;
3. Diagonal equal volume, equal shape;
4. Diagonal varying volume, varying shape;
5. Ellipsoidal, equal volume, shape and orientation;
6. Ellipsoidal, varying volume, shape and orientation.

The BIC selects model 4 and three clusters as the best solution. This solution can be shown graphically on scatterplot matrix of the three variables constructed by using

Table 6.6 Data on Exoplanets, from Mayor et al. (2003), reprinted by kind permission of Cambridge University Press

Mass (in Jupiter mass)	Period (in Earth days)	Eccentricity
0.12	4.95	0
0.197	3.971	0
0.21	44.28	0.34
0.22	75.8	0.28
0.23	6.403	0.08
0.25	3.024	0.02
0.34	2.985	0.08
0.4	10.901	0.498
0.42	3.5097	0
0.47	4.229	0
0.48	3.487	0.05
0.48	22.09	0.3
0.54	3.097	0.01
0.56	30.12	0.27
0.68	4.617	0.02
0.685	3.524	0
0.76	2594	0.1
0.77	14.31	0.27
0.81	828.95	0.04
0.88	221.6	0.54
0.88	2518	0.6
0.89	64.62	0.13
0.9	1136	0.33
0.93	3.092	0
0.93	14.66	0.03
0.99	39.81	0.07
0.99	500.73	0.1
0.99	872.3	0.28
1	337.11	0.38
1	264.9	0.38
1.01	540.4	0.52
1.01	1942	0.4
1.02	10.72	0.044
1.05	119.6	0.35

(*Continued*)

Table 6.6 (*Continued*)

Mass (in Jupiter mass)	Period (in Earth days)	Eccentricity
1.12	500	0.23
1.13	154.8	0.31
1.15	2614	0
1.23	1326	0.14
1.24	391	0.4
1.24	435.6	0.45
1.282	7.1262	0.134
1.42	426	0.02
1.55	51.61	0.649
1.56	1444.5	0.2
1.58	260	0.24
1.63	444.6	0.41
1.64	406.0	0.53
1.65	401.1	0.36
1.68	796.7	0.68
1.76	903	0.2
1.83	454	0.2
1.89	61.02	0.1
1.9	6.276	0.15
1.99	743	0.62
2.05	241.3	0.24
0.05	1119	0.17
2.08	228.52	0.304
2.24	311.3	0.22
2.54	1089	0.06
2.54	627.34	0.06
2.55	2185	0.18
2.63	414	0.21
2.84	250.5	0.19
2.94	229.9	0.35
3.03	186.9	0.41
3.32	267.2	0.23
3.36	1098	0.22
3.37	133.71	0.511
3.44	1112	0.52
3.55	18.2	0.01

(*Continued*)

Table 6.6 (*Continued*)

Mass (in Jupiter mass)	Period (in Earth days)	Eccentricity
3.81	340	0.36
3.9	111.81	0.927
4	15.78	0.046
4	5360	0.16
4.12	1209.9	0.65
4.14	3.313	0.02
4.27	1764	0.353
4.29	1308.5	0.31
4.5	951	0.45
4.8	1237	0.515
5.18	576	0.71
5.7	383	0.07
6.08	1074	.011
6.292	71.487	0.1243
7.17	256	0.7
7.39	1582	0.478
7.42	116.7	0.4
7.5	2300	0.395
7.7	58.116	0.529
7.95	1620	0.22
8	1558	0.314
8.64	550.65	0.71
9.7	653.22	0.41
10	3030	0.56
10.37	2115.2	0.62
10.96	84.03	0.33
11.3	2189	0.34
11.98	1209	0.37
14.4	8.428	0.277
16.9	1739.5	0.228
17.5	256.03	0.429

```
plot(planet.clus,planet.dat)
```

and selecting the pairs option (option number 2). The plot is shown in Figure 6.9.

Mean vectors of the three clusters can be found from

```
planet.clus$mu
```

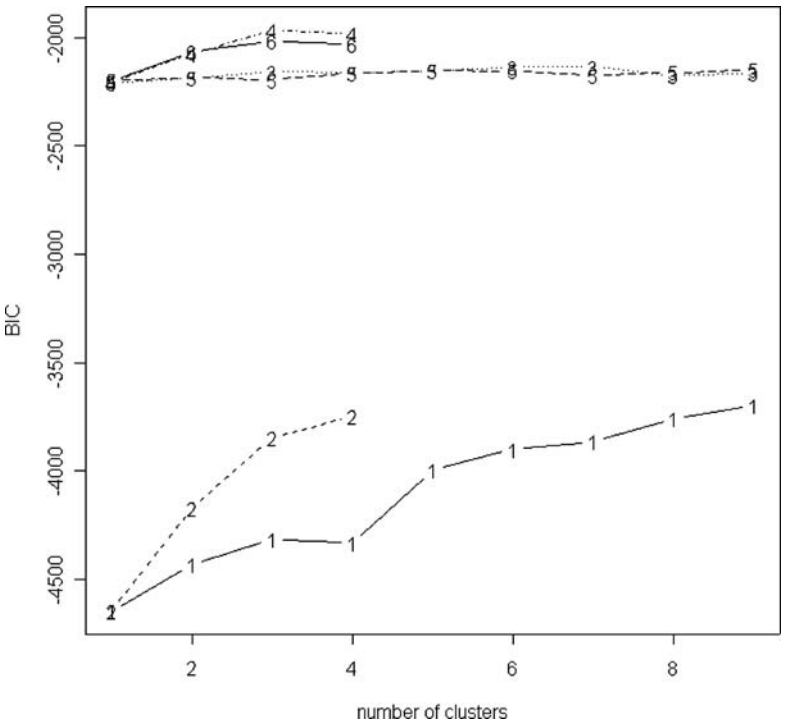

Figure 6.8 Plot of BIC values for a variety of models and a range of number of clusters.

and these are shown in Table 6.7. Cluster 1 consists of the "small" exoplanets (but still, on average, with a mass greater than Jupiter), with very short periods and eccentricities. The second cluster consists of large planets with very long periods and large eccentricities. The third cluster contains planets approximately the same mass as Jupiter, but with moderate periods and eccentricities.

6.5 Summary

Cluster analysis techniques provide a rich source of possible stategies for exploring complex multivariate data. They have been used widely in medical investigations; examples include Everitt et al. (1971) and Wastell and Gray (1987). Increasingly, model-based techniques such as finite mixture densities (see Everitt et al., 2001) and classification maximum likelihood, as described in this chapter, are superseding older methods, such as the single linkage, complete linkage, and average linkage methods described in Section 6.2. Two recent references are Fraley and Raftery (1998, 1999).

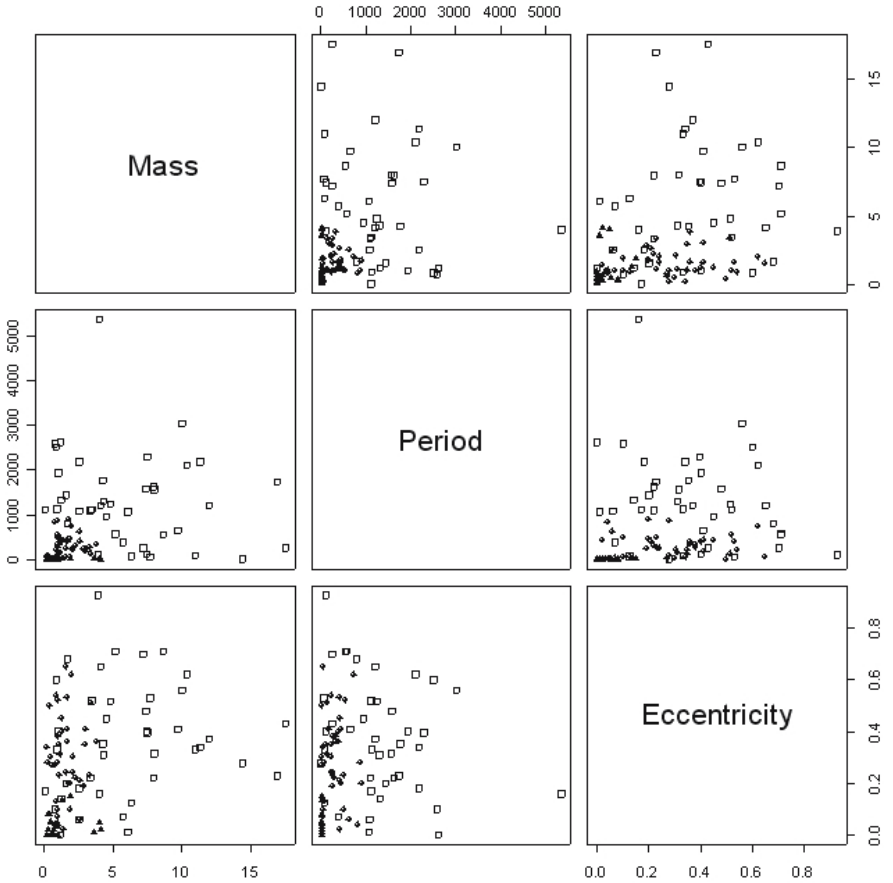

Figure 6.9 Scatterplot matrix of planets data showing three cluster solution from `Mclust`.

Table 6.7 Means for the Three-Group Solution for the Exoplanets Data

	Mass	Period	Eccentricity
Cluster 1: $n = 19$	1.16	6.45	0.035
Cluster 2: $n = 41$	5.81	1263.01	0.363
Cluster 3: $n = 15$	1.54	303.82	0.308

Exercises

6.1 Show that the intercluster distances used by single linkage, complete linkage, and group average clustering satisfy the following formula:

$$d_{k(ij)} = \alpha_i d_{ki} + \alpha_j d_{kj} + \gamma |d_{ki} - d_{kj}|,$$

where

$$\alpha_i = \alpha_j, \gamma = -\frac{1}{2} \qquad \text{(single linkage)},$$

$$\alpha_i = \alpha_j, \gamma = \frac{1}{2} \qquad \text{(complete linkage)},$$

$$a_i = \frac{n_i}{n_i + n_j}, \alpha_j = \frac{n_j}{n_i + n_j}, \gamma = 0 \qquad \text{(group average)}.$$

($d_{k(ij)}$ is the distance between a group k and a group (ij) formed by the fusion of groups i and j, and d_{ij} is the distance between groups i and j; n_i and n_j are the number of observations in groups i and j.)

6.2 Ward (1963) proposed an agglomerative hierarchical clustering procedure in which, at each step, the union of every possible pair of clusters is considered and the two clusters whose fusion results in the minimum increase in an error sum-of-squares criterion, *ESS*, are combined. For a single variable, *ESS* for a group with n individuals is simply $ESS = \sum_{i=1}^{n}(x_i - \bar{x})^2$.

(a) If ten individuals with variable values {2, 6, 5, 6, 2, 2, 2, 0, 0, 0} are considered as a single group, calculate *ESS*. If the individuals are grouped into two groups with individuals 1, 5, 6, 7, 8, 9, 10 in one group and individuals 2, 3, 4 in the other, what does *ESS* become?

(b) Can you fit Ward's method into the general equation given in Exercise 5.1?

6.3 Reanalyze the pottery data using `Mclust`. To what model in `Mclust` does the k-mean approach approximate?

6.4 Construct a three-dimensional drop-line scatterplot of the planets data in which the points are labelled with a suitable cluster label.

6.5 Reanalyze the life expectancy data by clustering the countries on the basis on differences between the life expectancies of men and women at corresponding ages.