

Computational Statistics

Solutions, Comments

16.1 a) Choosing the number of clusters is somewhat subjective. Figure 16.5 on page 524 illustrates the subjectivity. We look at the differences in the lengths of the vertical line segments. The horizontal line segments indicate groupings. The longer the line segments that connect the horizontal, the greater the separation between the groups, the longer are the vertical line segments joining them. A specific number of groups is formed by drawing a horizontal line as illustrated in Figure 16.5. It may be easier to draw such a horizontal line on a cluster tree drawn as in Figure 16.5, instead of the way the trees are drawn in Figure 16.7, because in Figure 16.5, the number of vertical lines crossed is the number of groups. In Figure 16.7, the horizontal line like the dotted lines in Figure 16.5 would not cross all vertical lines, but all leaves of the tree above the horizontal line would represent separate groups.

In R the appearance of the cluster tree produced by `hclust` is controlled by the `hang` argument. If `hang=-1`, the display has the appearance of Figure 16.5. The `axes` argument determines whether or not the vertical axis is shown as in Figure 16.7. With `hang=-1` and `axes=F`, the trees in Figure 16.7 would appear as shown below.

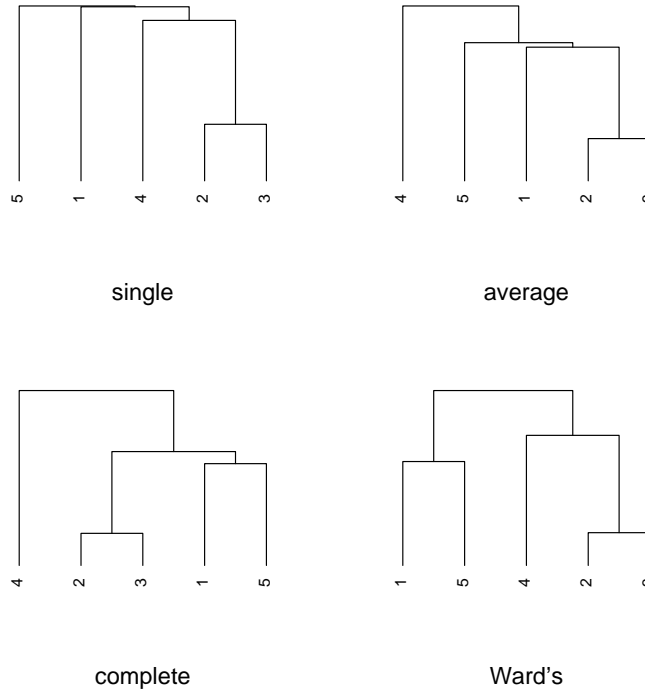


Figure 1: Cluster Trees of Figure 16.7

Although it is not necessary to draw the trees this way, it does make it easier to see the best separation.

It appears that single linkage suggests 4 groups: $\{1\}$, $\{2,3\}$, $\{4\}$, and $\{5\}$. (I've listed them in ascending order of the observation labels.)

It appears that average linkage gives the same 4 groups: $\{1\}$, $\{2,3\}$, $\{4\}$, and $\{5\}$.

It appears that complete linkage gives 3 groups: $\{1,5\}$, $\{2,3\}$, and $\{4\}$.

It appears that Ward's method suggests the same 3 groups as complete linkage: $\{1,5\}$, $\{2,3\}$, and $\{4\}$.

b) The two-way table to compare the contents of the 4 clusters formed by the two methods is

	C_{11}	C_{12}	C_{13}	C_{14}	
C_{21}	1	0	0	0	1
C_{22}	0	2	0	0	2
C_{23}	0	0	1	0	1
C_{24}	0	0	0	1	1
	1	2	1	1	5

So we have

$$R = 1 - \frac{7 - 14 + 7}{20} = 1.$$

In the modified Rand statistic, there are several terms that involve the number of ways 2 items can be selected from a fixed number of items, for example, $\binom{n_i}{2}$. If the number to choose from is less than 2, as it is with many terms in the modified Rand statistic for this problem, then this value is zero.

$$R_{HA} = \frac{1 - 1/20}{1 - 1/20} = 1.$$

- c) The two-way table to compare the contents of the 4 clusters in average linkage and 3 clusters in complete linkage is

	C_{11}	C_{12}	C_{13}	C_{14}	
C_{21}	1	0	0	0	1
C_{22}	0	0	0	2	2
C_{23}	0	1	1	0	2
	1	1	1	2	5

So we have

$$R = 1 - \frac{9 - 14 + 7}{20} = 0.9,$$

and

$$R_{HA} = \frac{1 - 2/20}{3/2 - 2/20} = 9/14$$

- d) This is similar to the previous parts.

16.5 The problem does not tell us what sample size to use. Let's just arbitrarily choose $n = 100$. Instead of using a function for the scree plot, I'll just write it directly.

```
n <- 100
a1 <- 5
a2 <- 1
sig <- 1
set.seed(3)
x1 <- runif(n)
x2 <- runif(n)
x3 <- a1*x1+a2*x2+sig*rnorm(n)
x <- cbind(x1,x2,x3)
xcmp<-prcomp(x)
x <- xcmp$sdev/sum(xcmp$sdev)
plot(x,type="l",ylab="Relative Size of Eigenvalue",xlab="Index of Eigenvalue")
```

The scree plot seems to suggest use of 2 principal components.

The standard deviations are 1.87, 0.271, and 0.164, which are the square roots of the eigenvalues of the sample variance/covariance matrix.

By adjusting the values of **a1**, **a2**, and **sig** in the code above, we can get varying scree plots suggesting varying numbers of principal components.

16.7 We use

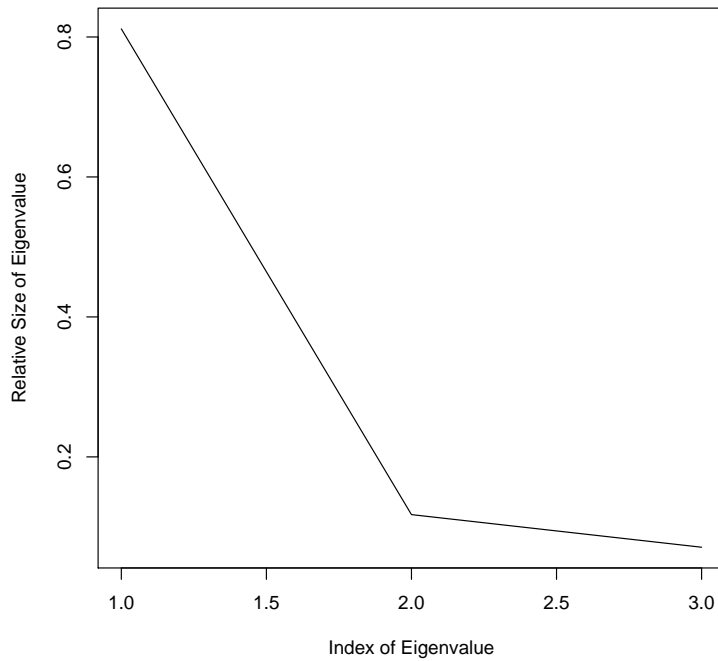


Figure 2: Scree Plot of Simulated Data

```

n <- 200
set.seed(3)
x <- rnorm(n)
y <- rnorm(n)
xx <- 10*x + y
yy <- 2*y + x
n2 <- n/2
yy[1:n2] <- yy[1:n2] + 5
yy[(n2+1):n] <- yy[(n2+1):n] - 5
data <- cbind(xx,yy)
# sphere data and plot
datas <- data%%solve(chol(var(data)))
plot(datas, ylab=expression(z[2]),xlab=expression(z[1]),cex=1.2,
      xlim=c(min(datas),max(datas)),ylim=c(min(datas),max(datas)))

```

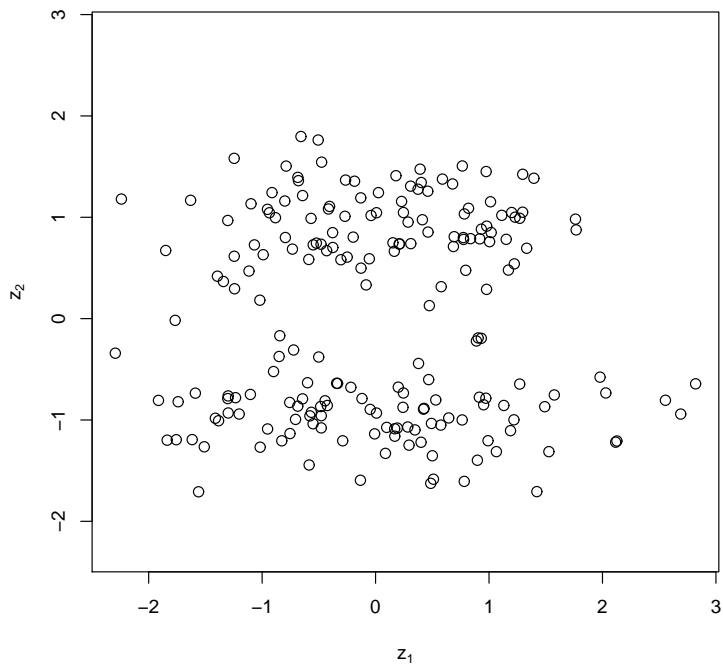


Figure 3: Plot of Sphered Simulated Data