STAT 32950: Homework 6

Robert Winter

Table of Contents

1 Exercise 1: Hierarchical Clustering for Small Data

The vocabulary "richness" of a text can be quantitatively described by counting the words used once, the words used twice, and so forth. Based on these counts, a linguist proposed the following distances between chapters of an ancient book (the Hebrew Bible book *Lamentations*):

$$
\begin{array}{c} \\ Ch1 \\ Ch2 \\ Ch3 \\ Ch4 \\ Ch5 \end{array}
\begin{array}{c}
\begin{array}{ccccc} Ch1 & Ch2 & Ch3 & Ch4 & Ch5 \end{array} \\
\left[\begin{array}{ccccc}
0 & & & & \\
0.76 & 0 & & & \\
2.97 & 0.80 & 0 & & \\
4.88 & 4.17 & 0.21 & 0 & \\
3.86 & 1.96 & 1.51 & 0.51 & 0
\end{array}\right]
\end{array}
$$

Cluster the chapters of the book using the three linkage hierarchical methods we have discussed: Single linkage, Complete linkage, and Average linkage. Compare the dendrograms (do not hand in the plots). You may use R.

1.1 Part (a)

Are the results from the three methods similar? Are the vertical scales the same?

In some ways, the three linkage methods yield similar results. In particular, all three methods yield dendrograms that—from a "bottom-up" perspective—view Chapters 3/4/5 as closer to one another than they are to Chapters 1/2.

However, there are ways in which the three methods' dendrograms are also distinct. Firstly, their vertical scales vary. For the single linkage method, the greatest height of a split is around 0.8; for the complete linkage method, the greatest height of a split is around 5, and for the average linkage method, the greatest height of a split is around 3. The methods also vary in the orders in which splits occur. For example, from a "bottom-up" perspective, the single linkage method merges Chapters 3 and 4, and then merges Chapter 5 in with Chapters 3 and 4, all before merging Chapters 1 and 2 together. The complete linkage method, on the other hand, first merges Chapters 3 and 4, but then merges Chapters 1 and 2 before merging Chapter 5 in with Chapters 3 and 4.

1.2 Part (b)

If we decide to have $k = 3$ clusters, list the three clusters produced by the three linkage methods.

From a "bottom-up"/agglomerative perspective, with $k = 3$ clusters,

- the single linkage method yields clusters of (Chapters 3, 4, and 5), (Chapter 1), and (Chapter 2),
- the complete linkage method yields clusters of (Chapters 3 and 4), (Chapters 1 and 2), and (Chapter 5), and
- the average linkage method yields clusters of (Chapters 3 and 4), (Chapters 1 and 2), and (Chapter 5).

1.3 Part (c)

Compare and comment. What's your takeaway?

It is noteworthy that the general structures of the dendrograms using the complete and average linkage methods are very similar (although they are on slightly different height scales). The complete linkage method, of course, is more "strict" than the single linkage method, since the former requires that the furthest pair of members be close to one another, while the latter requires that only the nearest pair of members be close to one another. The average linkage method strikes a balance between these two extremes — but since its dendrogram is so similar to that of the complete linkage method, it lends support to this method's results. That is, we conclude that Chapters 3 and 4 are similar to one another, that Chapters 1 and 2 are similar to one another, and that Chapter 5 is more similar to Chapters 3 and 4 than it is to Chapters 1 and 2.

2 Exercise 2: K-means for Small Data

Suppose we measure two variables X_1 and X_2 for four items A, B, C, D. The data are

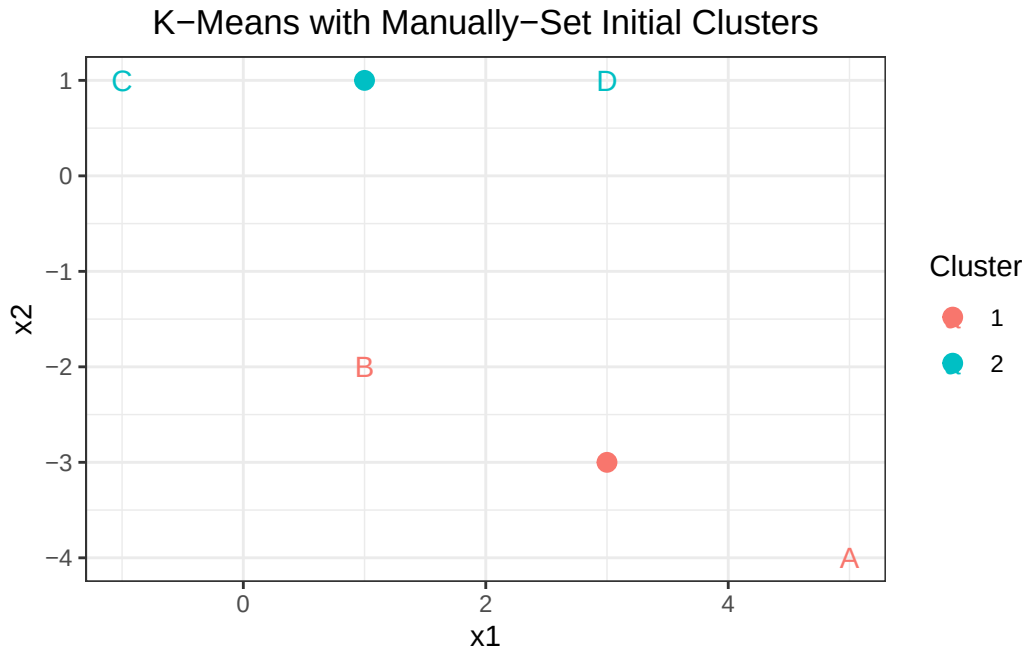
Item	x_1	x_2
A	5	-4
B	1	-2
C	-1	1
D	3	1

Use K-means clustering to divide the items into $K = 2$ clusters.

2.1 Part (a)

Start with the initial groups (AB) and (CD). Give the final clusters, cluster centroids, and squared distances to cluster centroids for each point.

We plot the points A, B, C, and D, color-coded by their initial groups (AB) and (CD), below. The centroid of the initial (AB) cluster is $c_1 = (\frac{5+1}{2}, \frac{-4+(-2)}{2}) = (\frac{6}{2}, \frac{-6}{2}) = (3, -3)$. The centroid of the initial (CD) cluster is $c_2 = (\frac{-1+3}{2}, \frac{1+1}{2}) = (\frac{2}{2}, \frac{2}{2}) = (1, 1)$. We plot these centroids below as well.



Now, we update each point's cluster assignments based on their distances to the two centroids:

- Point A
  - $||A - c_1|| = \sqrt{(5-3)^2 + (-4-(-3))^2} = \sqrt{(2)^2 + (-1)^2} = \sqrt{4+1} = \sqrt{5} \approx 2.236$
  - $||A - c_2|| = \sqrt{(5-1)^2 + (-4-1)^2} = \sqrt{(4)^2 + (-5)^2} = \sqrt{16+25} = \sqrt{41} \approx 6.403$
  - $2.236 < 6.403$, so we assign A to Cluster 1.
- Point B
  - $||B - c_1|| = \sqrt{(1-3)^2 + (-2-(-3))^2} = \sqrt{(-2)^2 + (1)^2} = \sqrt{4+1} = \sqrt{5} \approx 2.236$
  - $||B - c2|| = \sqrt{(1-1)^2 + (-2-1)^2} = \sqrt{(0)^2 + (-3)^2} = \sqrt{0+9} = \sqrt{9} = 3$
  - $2.236 < 3$, so we assign B to Cluster 1.
- Point C
  - $||C - c_1|| = \sqrt{(-1-3)^2 + (1-(-3))^2} = \sqrt{(-4)^2 + (4)^2} = \sqrt{16+16} = \sqrt{32} \approx 5.657$
  - $||C - c_2|| = \sqrt{(-1-1)^2 + (1-1)^2} = \sqrt{(-2)^2 + (0)^2} = \sqrt{4+0} = \sqrt{4} = 2$
  - $2 < 5.657$, so we assign C to Cluster 2.
- Point D
  - $||D - c_1|| = \sqrt{(3-3)^2 + (1-(-3))^2} = \sqrt{(0)^2 + (4)^2} = \sqrt{0+16} = \sqrt{16} = 4$

- $||D - c_2|| = \sqrt{(3-1)^2 + (1-1)^2} = \sqrt{(2)^2 + (0)^2} = \sqrt{4+0} = \sqrt{4} = 2$
- $2 < 4$, so we assign D to Cluster 2.

Thus, our clusters are Cluster 1 = (AB) and Cluster 2 = (CD). Notice that our clusters haven't changed at all from the initial groups! Since our clusters stayed the same, our centroids stayed the same too. We've started our K-means algorithm at a fixed state: every iteration of the algorithm will leave the cluster assignments, hence centroids, unchanged.

Thus, our final clusters are:

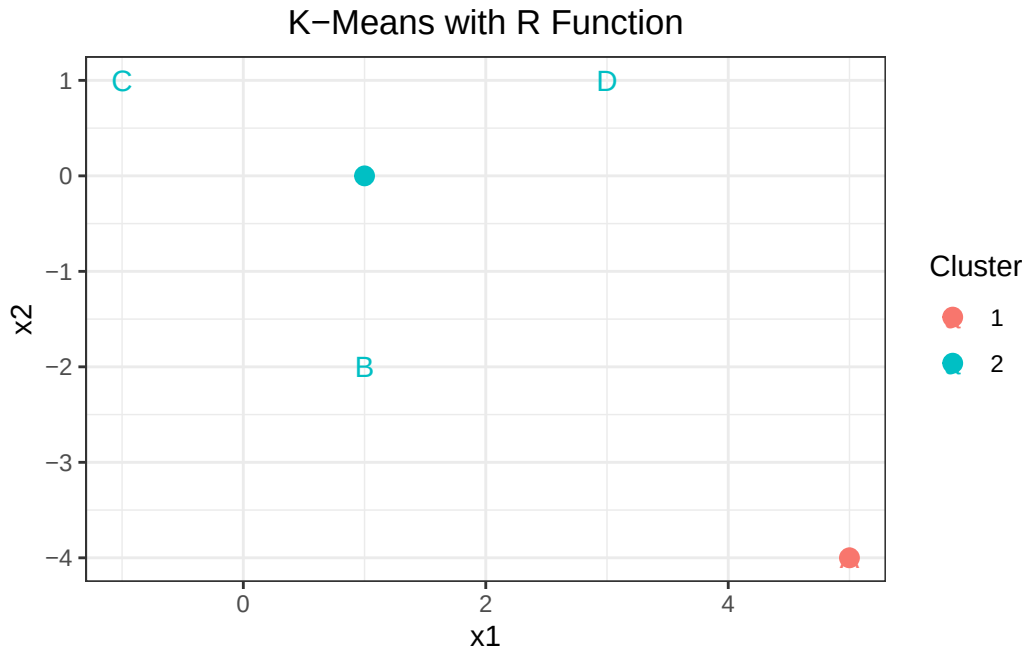
- (AB), with centroid $c_1 = (3, -3)$. The squared distance between A and c_1 is $||A - c_1||^2 = (\sqrt{5})^2 = 5$, and the squared distance between B and c_1 is $||B - c_1||^2 = (\sqrt{5})^2 = 5$.
- (CD), with centroid $c_2 = (1, 1)$. The squared distance between C and c_2 is $||C - c_2||^2 = (2)^2 = 4$, and the squared distance between D and c_2 is $||D - c_2||^2 = (2)^2 = 4$.

2.2 Part (b)

Use the `kmeans` function in R (which tries several initializations). Give the final clusters, cluster centroids, and squared distances to cluster centroids for each point.

```
set.seed(41)
q2kmeans = kmeans(abcd[,2:3], centers = 2)
```

Now, we color-code A, B, C, and D by their final cluster assignments using R's `kmeans` function.



Our new final clusters are:

- (A), with centroid $c_1 = A = (5, -4)$. The squared distance between A and c_1 is of course 0.
- (BCD), with centroid $c_2 = (1, 0)$. The squared distance between B and c_2 is $(1-1)^2 + (-2-0)^2 = (0)^2 + (-2)^2 = 4$. The squared distance between C and c_2 is $(-1-1)^2 + (1-0)^2 = (-2)^2 + (1)^2 = 5$. The squared distance between D and c_2 is $(3-1)^2 + (1-0)^2 = (2)^2 + (1)^2 = 5$.

3 Exercise 3: Hierarchical and K-Means Cluster Analysis

The national track records data for women are used in previous assignments: `ladyrun24.dat`.

```
ladyrun =
 ↪  read.table("C:/Users/rewin/OneDrive/Documents/STAT_32950/Homework/HW6/ladyrun24.dat")
colnames(ladyrun) = c("Country", "d100m", "d200m", "d400m", "d800m",
 ↪  "d1500m", "d3000m", "Marathon")
```

3.1 Part (a)

Use the data to calculate the Euclidean distances between pairs of countries. Do not print the output. Give the two countries (translate into true names) with

the maximum distance and the two countries with the minimum distance, along with the distance values. (The `which` function in R can be useful.) (Note: the variables are in comparable scales; you may either use the original data or the normalized data.)

First, we scale the data so that national records for each distance have unit variance. We then calculate the pairwise Euclidean distances between countries based on these scaled records. For good measure, we also calculate the pairwise Euclidean distances between countries based on raw (non-normalized) national records.

```
# Pairwise Euclidean distances on data scaled to unit variance
ladyrun_sc = cbind(ladyrun[1], scale(ladyrun[2:8], center = FALSE,
 ↪  scale = apply(ladyrun[2:8], 2, sd, na.rm = TRUE)))
country_dists_sc = dist(ladyrun_sc[2:8], method = "euclidean")

# Pairwise Euclidean distances on raw data
country_dists = dist(ladyrun[2:8], method = "euclidean")
```

Based on scaled women's national track records, the two countries separated by the farthest Euclidean distance are Samoa ("SAM") and China ("CHN"), with a distance between them of approximately 12.031. Based on raw women's national track records, the two countries separated by the farthest Euclidean distance are Papua New Guinea ("PNG") and Kenya ("KEN"), with a distance between them of approximately 87.184.

```
# Max distance

# Scaled Data
max(country_dists_sc) # 12.031
which(as.matrix(country_dists_sc) == max(country_dists_sc), arr.ind =
 ↪  T) # (46, 9)
ladyrun$Country[46]; ladyrun$Country[9]  # SAM = Samoa; CHN = China

# Raw Data
max(country_dists) # 87.184
which(as.matrix(country_dists) == max(country_dists), arr.ind = T) #
 ↪  (29, 40)
ladyrun$Country[40]; ladyrun$Country[29] # PNG = Papua New Guinea; KEN
 ↪  = Kenya
```

Furthermore, based on scaled women's national track records, the two countries separated by the shortest Euclidean distance are Switzerland ("SUI") and Portugal ("POR"), with a distance between them of approximately 0.304. Based on raw women's national track records, the two countries separated by the shortest Euclidean distance are Canada ("CAN") and Spain ("ESP"), with a distance between them of approximately 0.513.

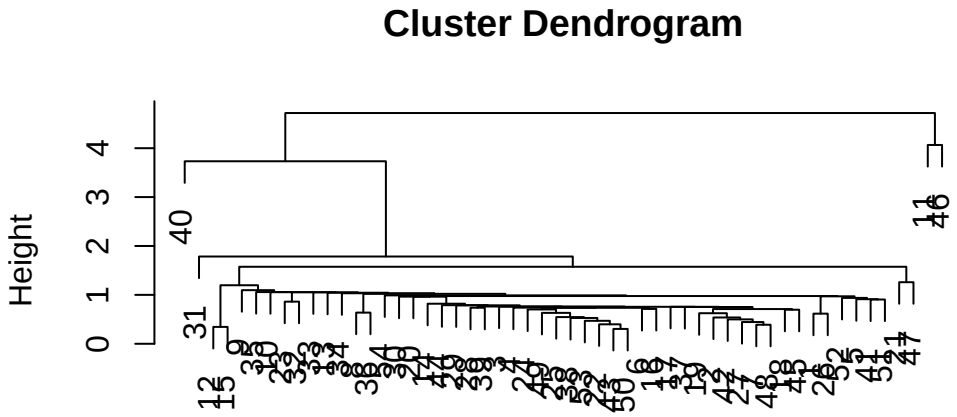
```r
# Min distance

# Scaled Data
min(country_dists_sc) # 12.031
which(as.matrix(country_dists_sc) == min(country_dists_sc), arr.ind =
 ↪  T) # (50, 43)
ladyrun$Country[50]; ladyrun$Country[43] # SUI = Switzerland; POR =
 ↪  Portugal

# Raw Data
min(country_dists) # 0.513
which(as.matrix(country_dists) == min(country_dists), arr.ind = T) #
 ↪  (7, 48)
ladyrun$Country[7]; ladyrun$Country[48] # CAN = Canada; ESP = Spain
```

3.2 Part (b)

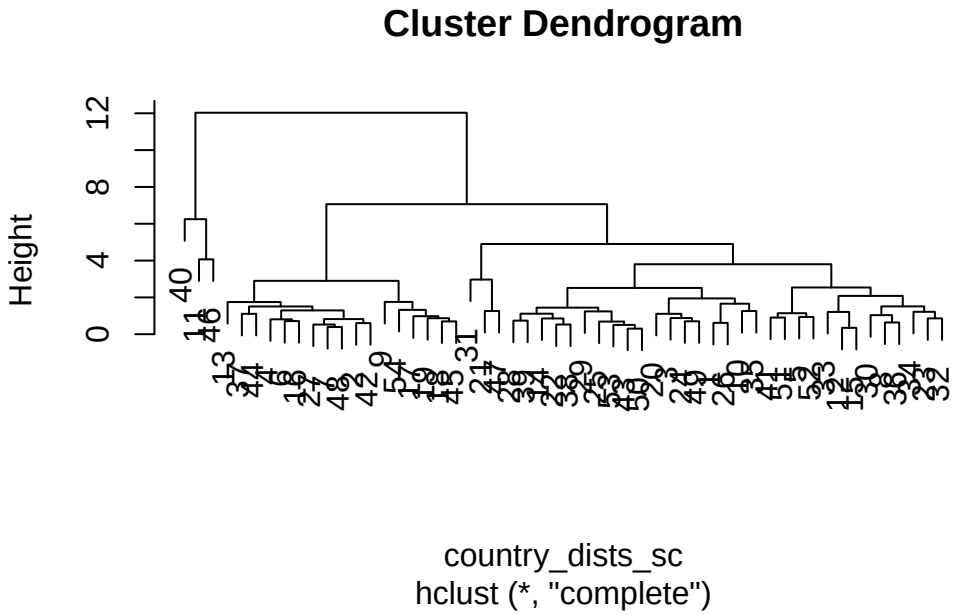
Treating the distances in Part (a) as measures of dissimilarity, cluster the countries using the single linkage and complete linkage hierarchical procedures. Construct dendrograms and compare the results. When $k = 8$ (or 7), provide the three smallest clusters (using country abbreviations) according to the complete linkage.

Using the scaled data, we plot dendrograms of the distances between countries using the single and complete linkage methods below. One of the most striking differences between the two dendrograms is the relative heights and orders in which merges take place. In the single linkage dendrogram, a single cluster of *every* country except for the Cook Islands (11), Papua New Guinea (40), and Samoa (46) forms before a height of 2, while these small island countries don't even merge into non-singleton clusters before heights of 3 or 4. In the complete linkage dendrogram, on the other hand, merges between these small island countries happen relatively earlier — at similar heights as merges of clusters of other countries. For example, when the Cook Islands and Samoa merge into a cluster of two countries, there are still four other clusters, unlike in the single linkage case, where just one other cluster remains once the Cook Islands and Samoa have merged.

```r
plot(hclust(country_dists_sc, method = "single"))
```



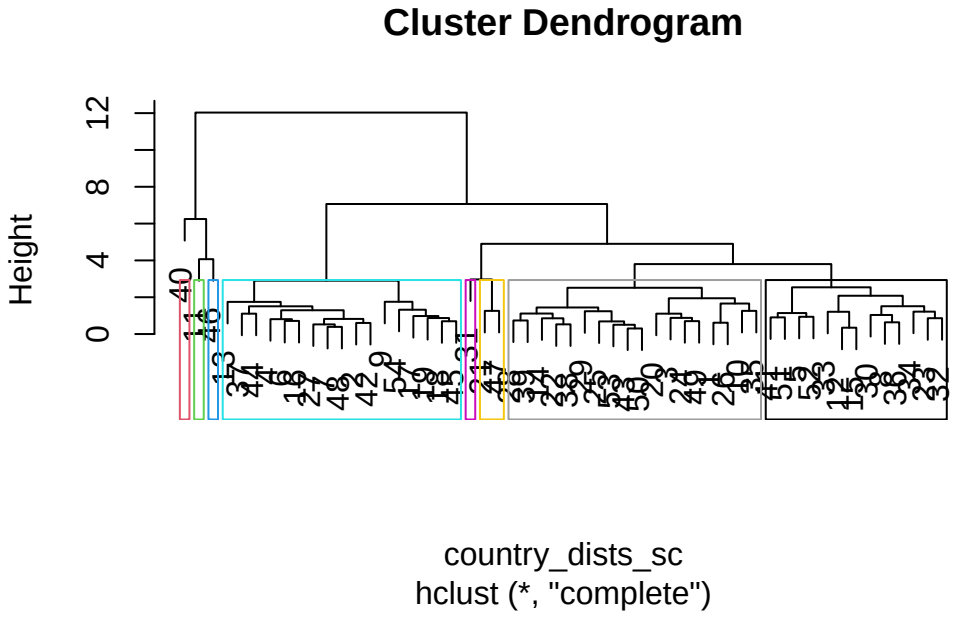

```
plot(hclust(country_dists_sc, method = "complete"))
```



Using the complete linkage, when there are $k = 8$ clusters, there is a four-way tie for the smallest cluster, all of which are singletons: (COK), (PNG), (SAM), and (KORN), as shown

below.

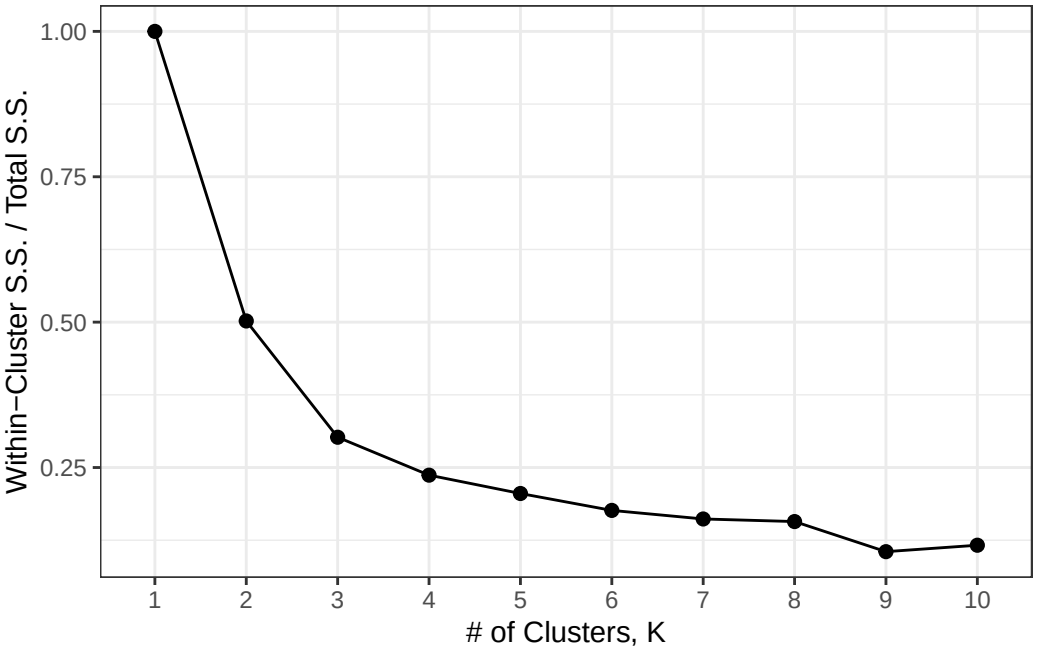
```
# k=8 clusters
plot(hclust(country_dists_sc, method = "complete"))
rect.hclust(hclust(country_dists_sc, method = "complete"), k=8, border
 ↪  = 2:9)
```



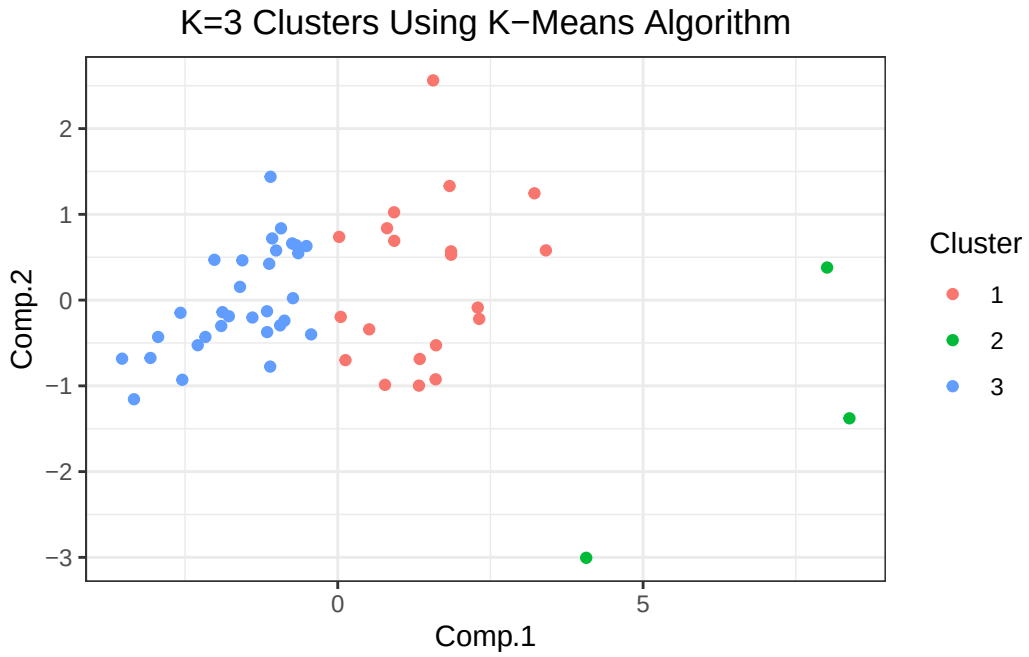
3.3 Part (c)

Input the data into a K-means clustering program. Cluster the countries into groups using several values of K. Compare the results with those in Part (b). Which K would you choose? Plot your K clusters (by colors or symbols) on the plane with the first two principal components as axes.

To determine how many clusters we will use, we begin with a scree plot depicting the proportion of variation in the scaled data that is *not* attributable to between-cluster variation. Once we have $K = 3$ clusters, adding more clusters does not decrease the proportion of unexplained variation in the data by much, so we settle on three clusters.

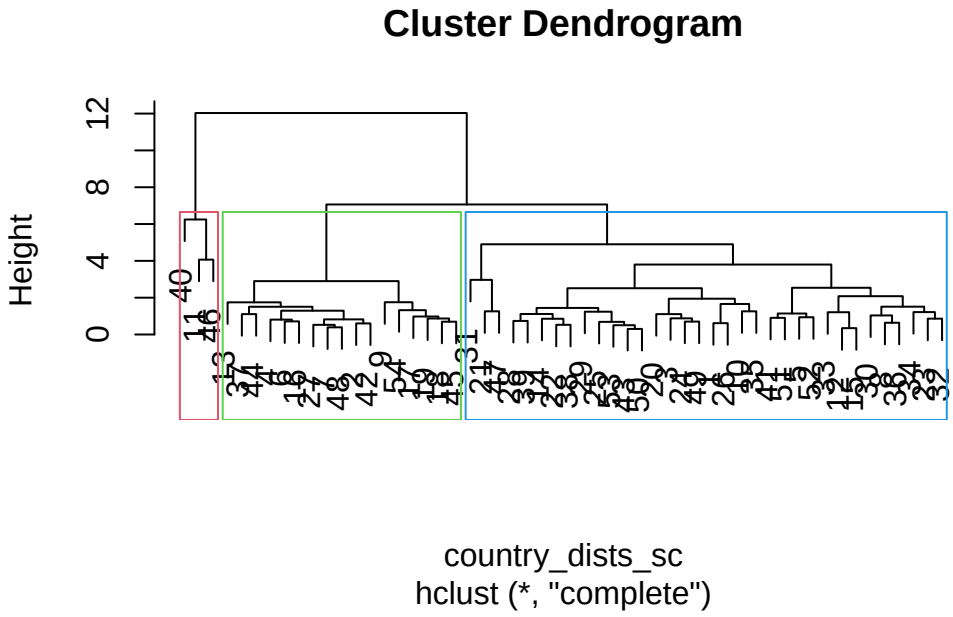


Below, we plot the scaled data in the plane of their first two principal components, color-coded by cluster with $K = 3$ clusters total. We see that our choice of $K = 3$ has an intuitive justification in addition to the scree plot above. In Problem Set 2, we concluded that the first principal component of these data captures a country's overall standing across all distances, with lower scores corresponding to faster national records across all distances. Our three clusters seem to group the data based on this overall standing. One cluster contains the three small island countries of Samoa, Papua New Guinea, and the Cook Islands, whose national records across distances are slow outliers. The remaining two clusters then separate the non-outlier countries into a faster group and a slower group.



The complete linkage method explored in Part (b) seems to cluster the data in a similar way, as cutting its dendrogram to have $K = 3$ clusters also groups the three small island countries into their own cluster.

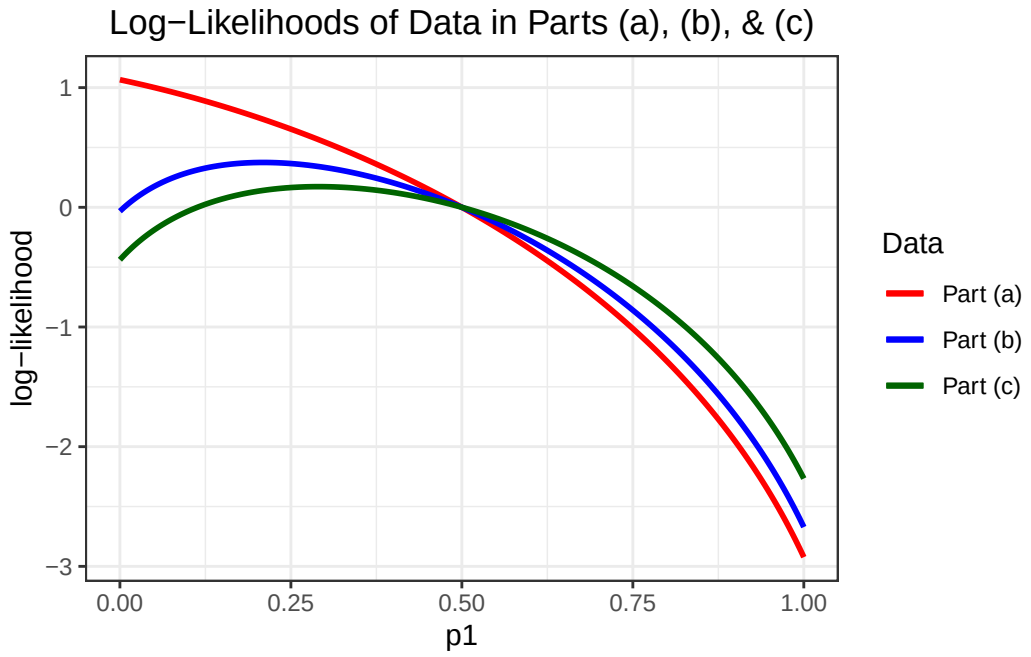
```
plot(hclust(country_dists_sc, method = "complete"))
rect.hclust(hclust(country_dists_sc, method = "complete"), k=3, border
 ↪  = 2:4)
```


Cluster Dendrogram
Height
0
4
8
12
country_dists_sc
hclust (*, "complete")

4 Exercise 4: Two Component Mixture Model for Small Data

4.1 Part (d)

Plot the log-likelihood functions in cases (a), (b), and (c) on the same graph, as functions of p_1.



4.2 Part (e)

Based on your plots, what are the estimated $(\hat{p}_1, \hat{p}_2)$ in cases (a), (b), and (c)? Are the estimates reasonable in every case?

Below, we add vertical lines to the plot indicating where each log-likelihood is maximized.

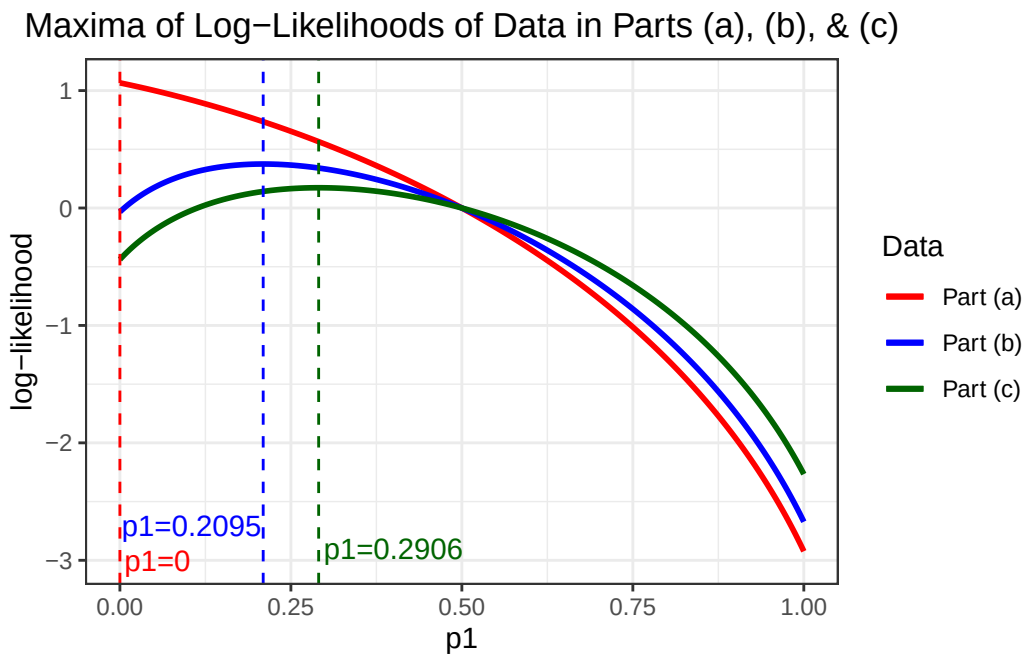
For Part (a), the log-likelihood is maximized at $p_1 = 0$, so the maximum likelihood estimates of the mixture proportion parameters are $(\hat{p}_1, \hat{p}_2) = (0, 1-0) = (0, 1)$.

For Part (b), the log-likelihood is maximized at $p_1 = 0.2095$, so the maximum likelihood estimates of the mixture proportion parameters are $(\hat{p}_1, \hat{p}_2) = (0.2095, 1-0.2095) = (0.2095, 7095)$.

For Part (c), the log-likelihood is maximized at $p_1 = 0.2906$, so the maximum likelihood estimates of the mixture proportion parameters are $(\hat{p}_1, \hat{p}_2) = (0.2906, 1-0.2906) = (0.2906, 0.7094)$.

Each of these pairs of estimates make sense. In particular, as we moved from Part (a) to Part (b) to Part (c), our dataset contained more points further to the right of the $[0, 1]$

interval. These points have greater densities under mixtures that assign greater weight to $f_1(x)$ — which is why the weights assigned to $f_1(x)$ steadily increase from Part (a) to Part (b) to Part (c).



5 Exercise 5: Mixture Model Analysis Practice

The dataset `heart.dat.txt` consists of 270 observations.

```
heart =
 ↪  read.table("C:/Users/rewin/OneDrive/Documents/STAT_32950/Homework/HW6/heart.dat.txt")
colnames(heart) = c("age", "sex", "chest", "bp", "chl", "sugar", "ecg",
 ↪  "rate", "angina", "peak", "slope", "vssl", "thal", "ill")
```

The 14 variables are:

```
-- 1. age
-- 2. sex
-- 3. chest pain type  (4 values)
-- 4. resting blood pressure
-- 5. serum cholestoral in mg/dl
-- 6. fasting blood sugar > 120 mg/dl
-- 7. resting electrocardiographic results  (values 0,1,2)
-- 8. maximum heart rate achieved
-- 9. exercise induced angina
-- 10. oldpeak = ST depression induced by exercise relative to rest
-- 11. the slope of the peak exercise ST segment
-- 12. number of major vessels (0-3) colored by flourosopy
-- 13.  thal: 3 = normal; 6 = fixed defect; 7 = reversible defect
-- 14. illness: 1 = absence; 2 = presence of heart disease
```

The variable types are:

```
Real: 1,4,5,8,10,12
Ordered:11
Binary: 2,6,9
Nominal:7,3,13,14
```

Conduct a mixture analysis on the data using the real variables or a subset of them. Show your main steps, plots, and results. Interpret (and comment on the goodness of fit of the model, the clusters, etc.).

We use the expectation-maximization algorithm to cluster the patients based on their six real-valued variables. The optimal model has $k = 5$ clusters, $k^* = 44$ parameters, a log-likelihood of $\log(L) \approx -5207.073$, and a "VEI"—or "diagonal, varying volume, equal shape"—covariance structure. The BIC of this model is therefore approximately

$$\begin{aligned} BIC &= -2\log(L) + k^* \log(n) \\ &\approx -2(-5207.073) + 44\log(270) \\ &\approx 10660.48. \end{aligned}$$

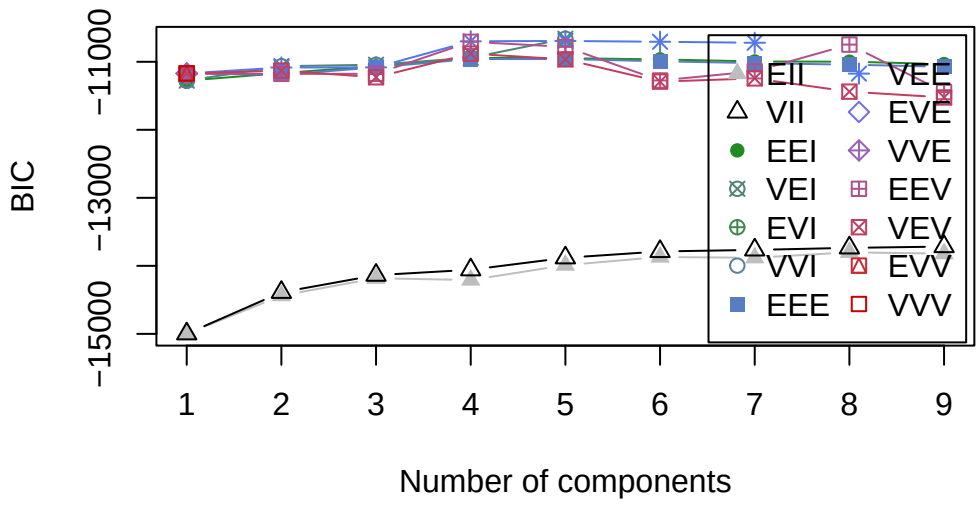
```r
heartreal = dplyr::select(heart, c("age", "bp", "chl", "rate", "peak",
 ↪  "vssl")) # subset to real-valued variables
mclust_heart = Mclust(heartreal)
summary(mclust_heart)
```

```
----------------------------------------------------
Gaussian finite mixture model fitted by EM algorithm
```

```
----------------------------------------------------

Mclust VEI (diagonal, equal shape) model with 5 components:

 log-likelihood   n df       BIC       ICL
      -5207.073 270 44 -10660.48 -10698.41

Clustering table:
 1  2  3  4  5
52 42 54 58 64
```

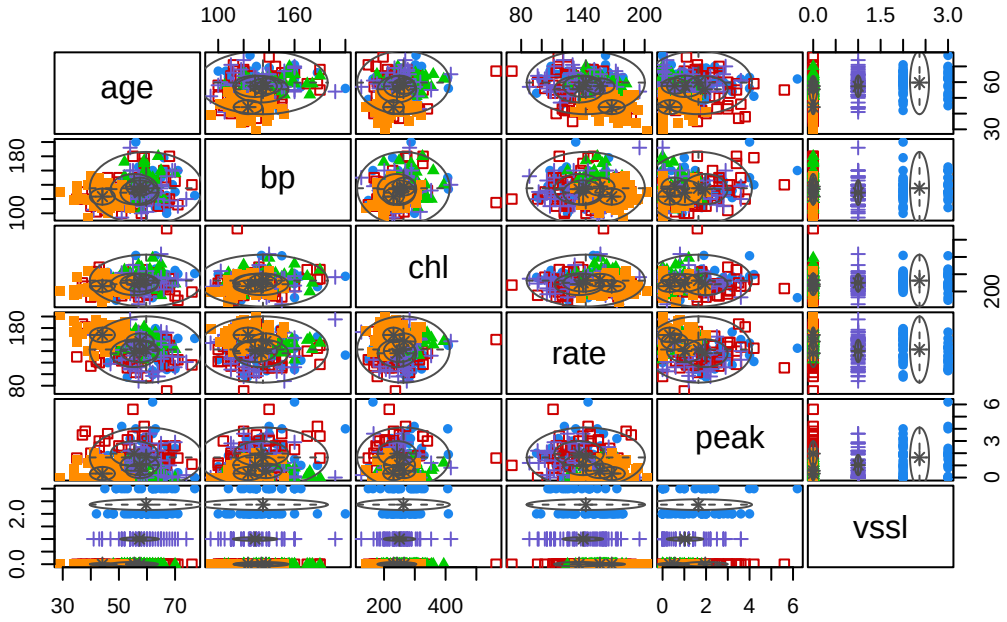
It is somewhat difficult to make out in the plot below, but we can also tell that this is the optimal model because the highest $-BIC$ is achieved by the VEI model with 5 components, marked with a green $\otimes$ symbol.

```r
plot(mclust_heart, what = c("BIC"))
```



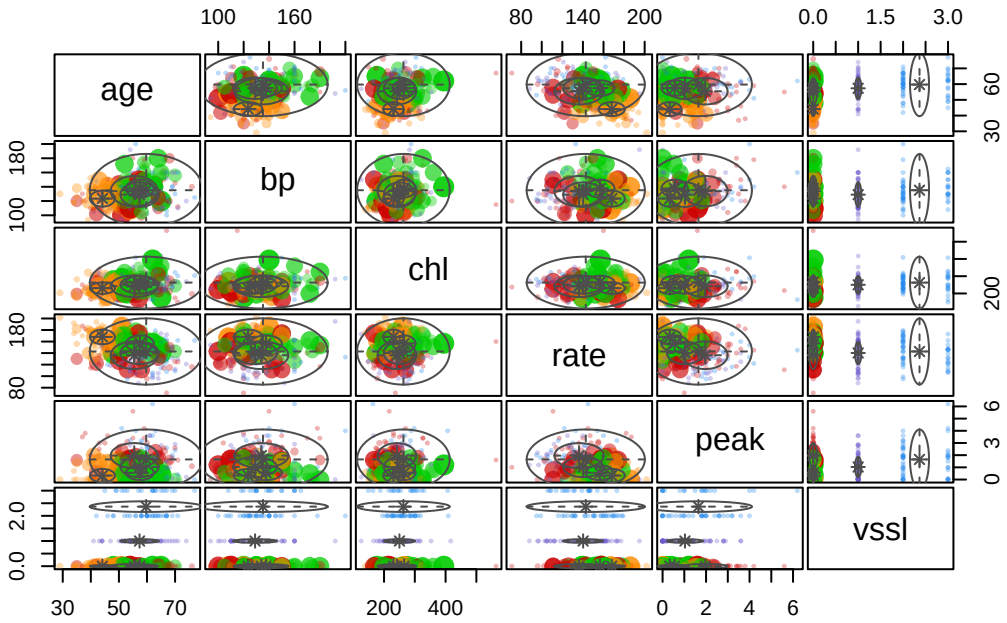
Using this model, we can visualize the five clusters in each of the $\text{nCr}(6, 2) = 15$ planes of variable pairs. The clusters are admittedly difficult to make out, especially in certain covariate planes. At minimum, the fact that the clusters seem to respect a patient's number of major vessels colored by flourosopy—which are constrained to the integers $0, 1, 2, 3$—suggests that meaningful clusters have been identified.

```
plot(mclust_heart, what = c("classification"))
```



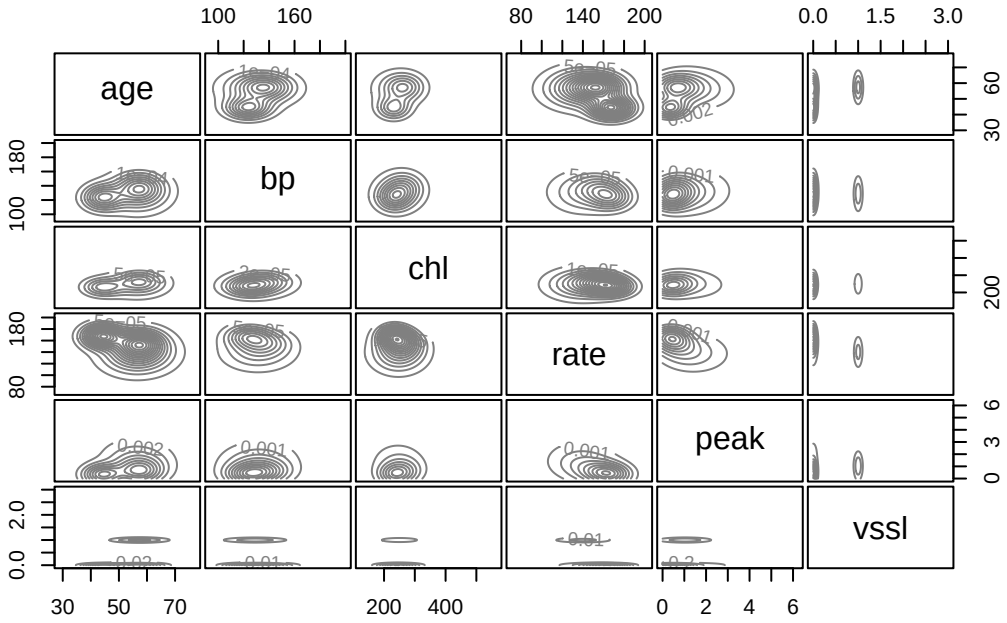
We also plot the uncertainty associated with the model's classifications:

```
plot(mclust_heart, what = c("uncertainty"))
```



Below are also the densities of the six variables in each variable pair plane:

```
plot(mclust_heart, what = c("density"))
```



The estimated mixture proportions are $\hat{p}_1 \approx 0.193, \hat{p}_2 \approx 0.167, \hat{p}_3 \approx 0.193, \hat{p}_4 \approx 0.215, \hat{p}_5 \approx 0.233$. Since these are all around 0.2, it seems that the joint distribution of the six real-valued variables in the data is a pretty even mixture of five multivariate normal distributions.

```
# Estimated mixture proportions
mclust_heart$parameters$pro
```

```
[1] 0.1925926 0.1666624 0.1932443 0.2148148 0.2326859
```