# UML_hw2

*Bhargavi Ganesh*

*10/18/2019*

**Checking distance measures: Questions 1-3**

```
m <- matrix(1:4, nrow = 2, byrow=TRUE)
dist(m, method = "euclidean")[1]
```

```
## [1] 2.828427
```

```
dist(m, method = "manhattan")[1]
```

```
## [1] 4
```

```
dist(m, method = "canberra")[1]
```

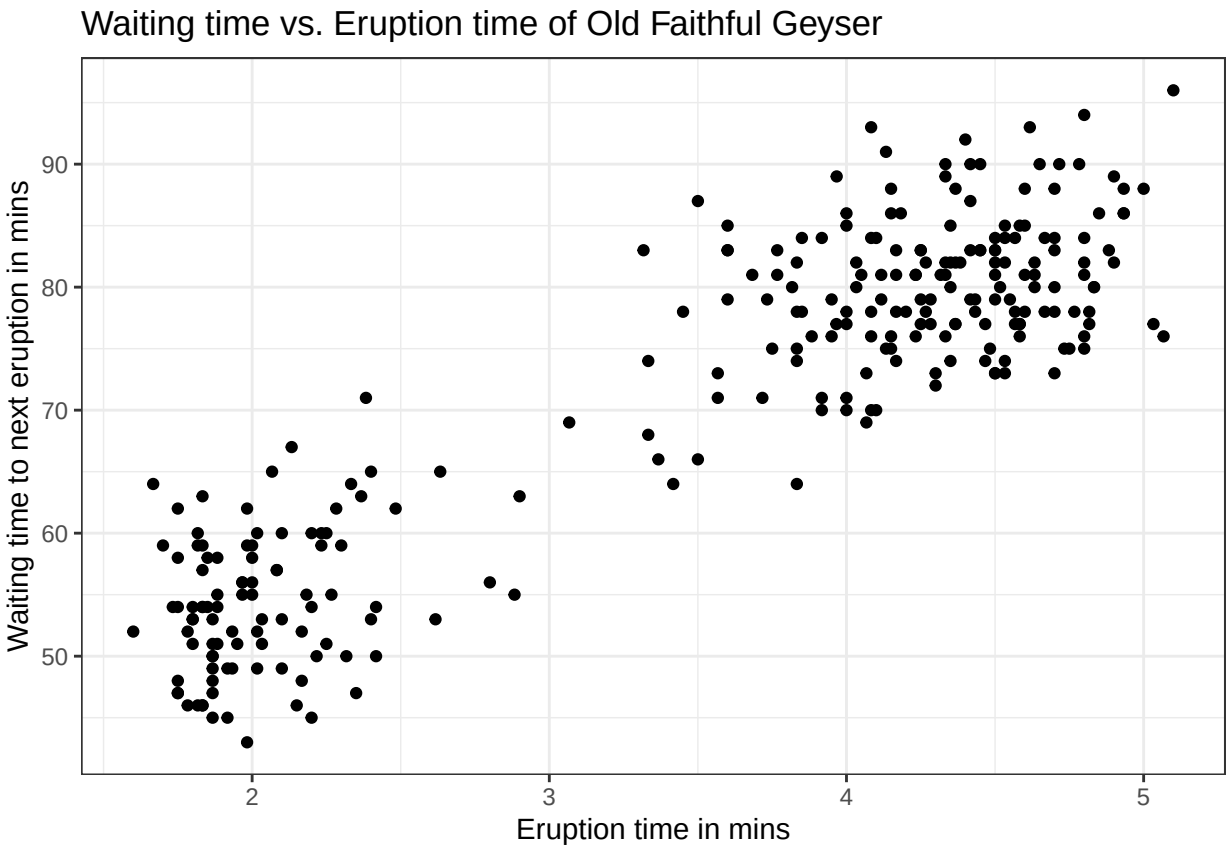
```
## [1] 0.8333333
```

The answers I got matched the handwritten calculations I had. The three distance measures are calculated in particularly different ways. The euclidean distance measures the straight line distance, the manhattan distance measures the "city block" distance, or distance in different directions in the coordinate plane, and the canberra distance is the weighted manhattan distance. The differences are important because they measure how similar two points are to each other, and therefore give us different clusters. We might see these differences in action by noting that if we consider a distance of 2 as being similar, for example, then one of the measures (canberra) would put both points in the same cluster, whereas the other two would not.

**Exploratory Data Analysis: Question 4**

Next, we can do some exploratory data analysis to understand the nature of the data. A scatterplot below shows the clustering of the data:

```
faithful %>%
  ggplot() +
  geom_point(aes(x = eruptions, y = waiting), stat = "identity") +
  labs(x = "Eruption time in mins",
       y = "Waiting time to next eruption in mins",
       title = "Waiting time vs. Eruption time of Old Faithful Geyser") +
  theme_bw()
```



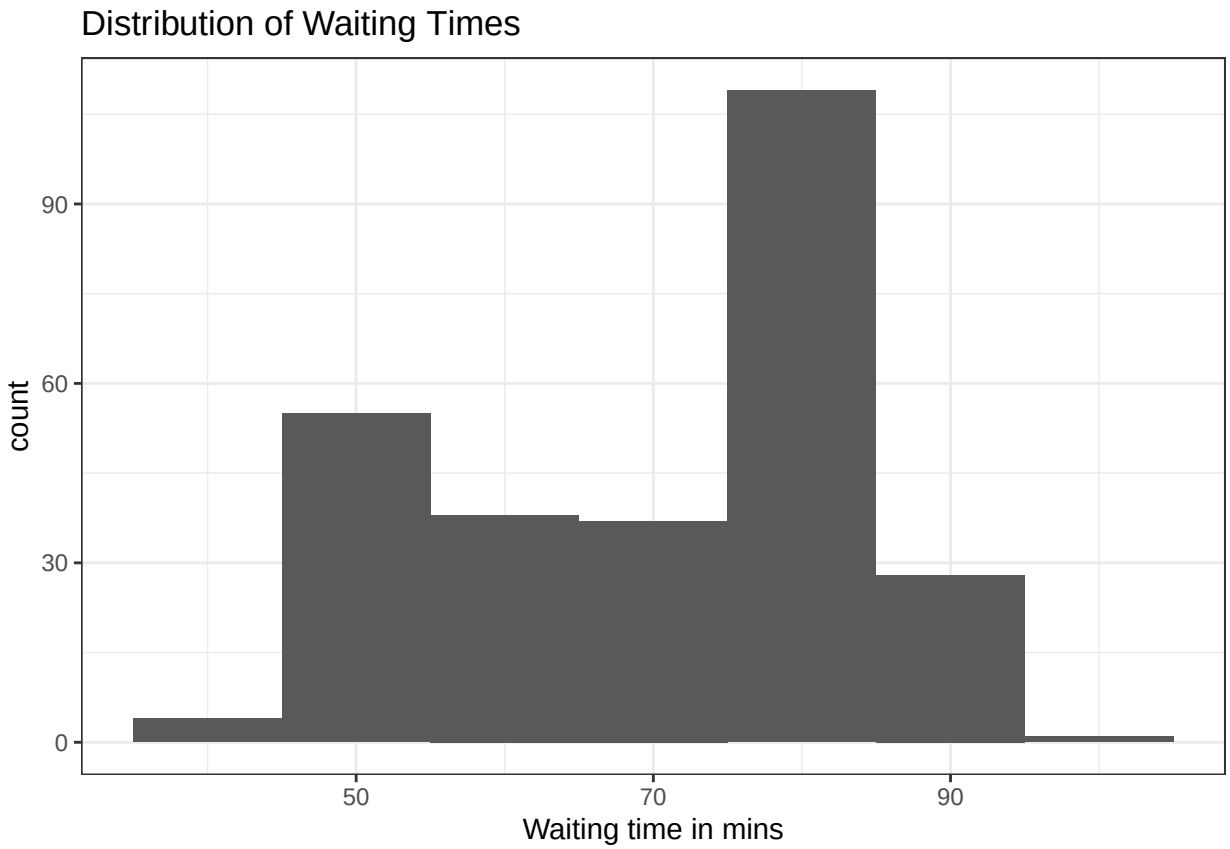

As we can see from the scatterplot above, visual inspection demonstrates that there are two possible clusters of the data. Below, we also look at the distribution of each of the variables, by using central tendency measures, as well as histograms.

```
summary(faithful)
```

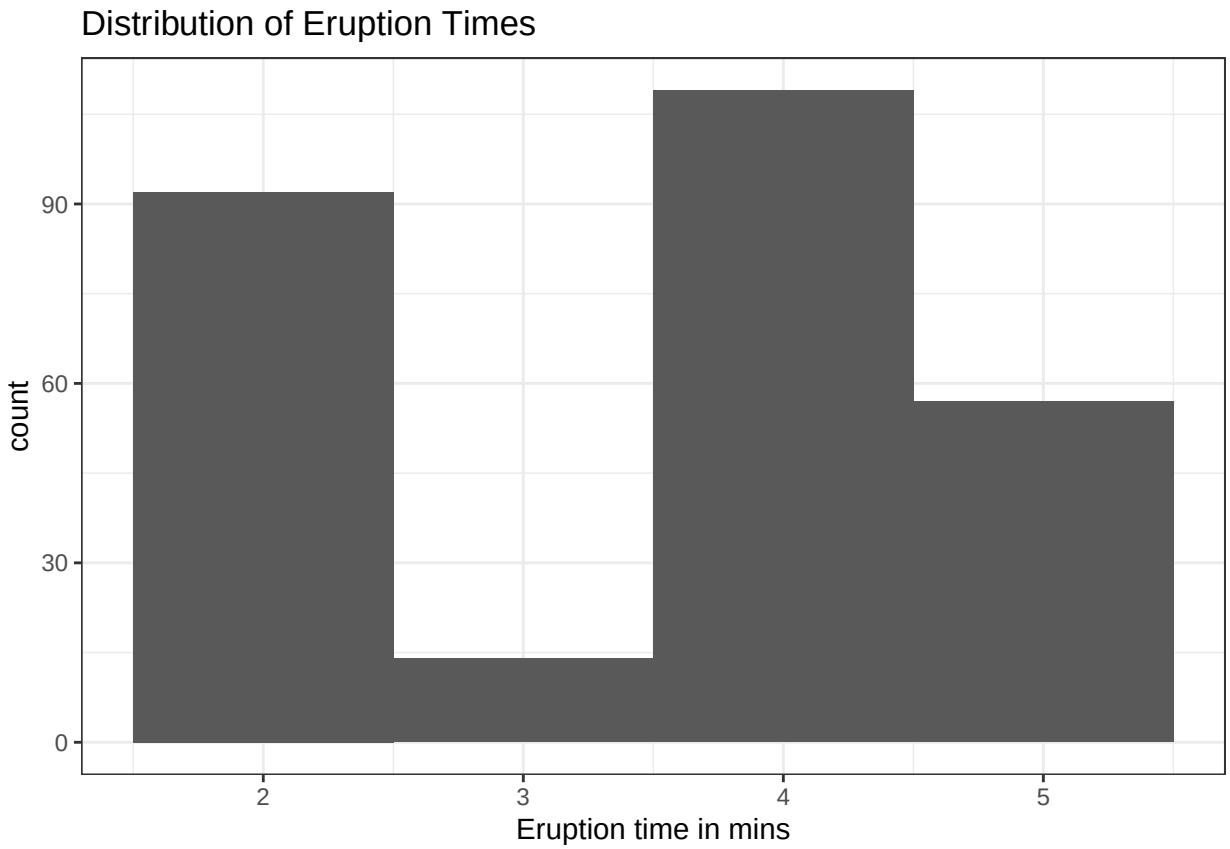
```
##    eruptions        waiting
##  Min.   :1.600   Min.   :43.0
##  1st Qu.:2.163   1st Qu.:58.0
##  Median :4.000   Median :76.0
##  Mean   :3.488   Mean   :70.9
##  3rd Qu.:4.454   3rd Qu.:82.0
##  Max.   :5.100   Max.   :96.0
```

The mean waiting time is 70.9 minutes, whereas the median waiting time is 76 minutes. The mean eruption time is 3.49 minutes, whereas the median eruption time is 4.0 minutes. This suggests some skew in both distributions, which can be seen in the histograms.

```
ggplot(data = faithful) +
  geom_histogram(aes(x = waiting), binwidth = 10) +
  labs(x = "Waiting time in mins", title= "Distribution of Waiting Times") +
  theme_bw()
```



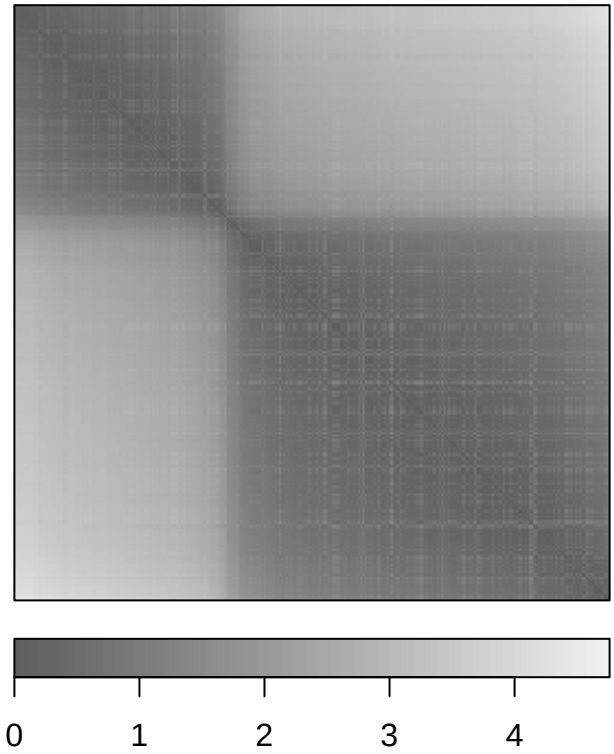

```
ggplot(data = faithful) +
  geom_histogram(aes(x = eruptions), binwidth = 1) +
  labs(x = "Eruption time in mins", title= "Distribution of Eruption Times") +
  theme_bw()
```




The histogram of the waiting variable shows that the greatest number of observations occurred at the waiting time of 75-85 minutes. The histogram of the eruptions variable shows that the greatest number of observations occurred at an eruption time of 3.5-4.5 minutes.

**Dissimilarity Matrix and ODI Plot: Questions 5 and 6**

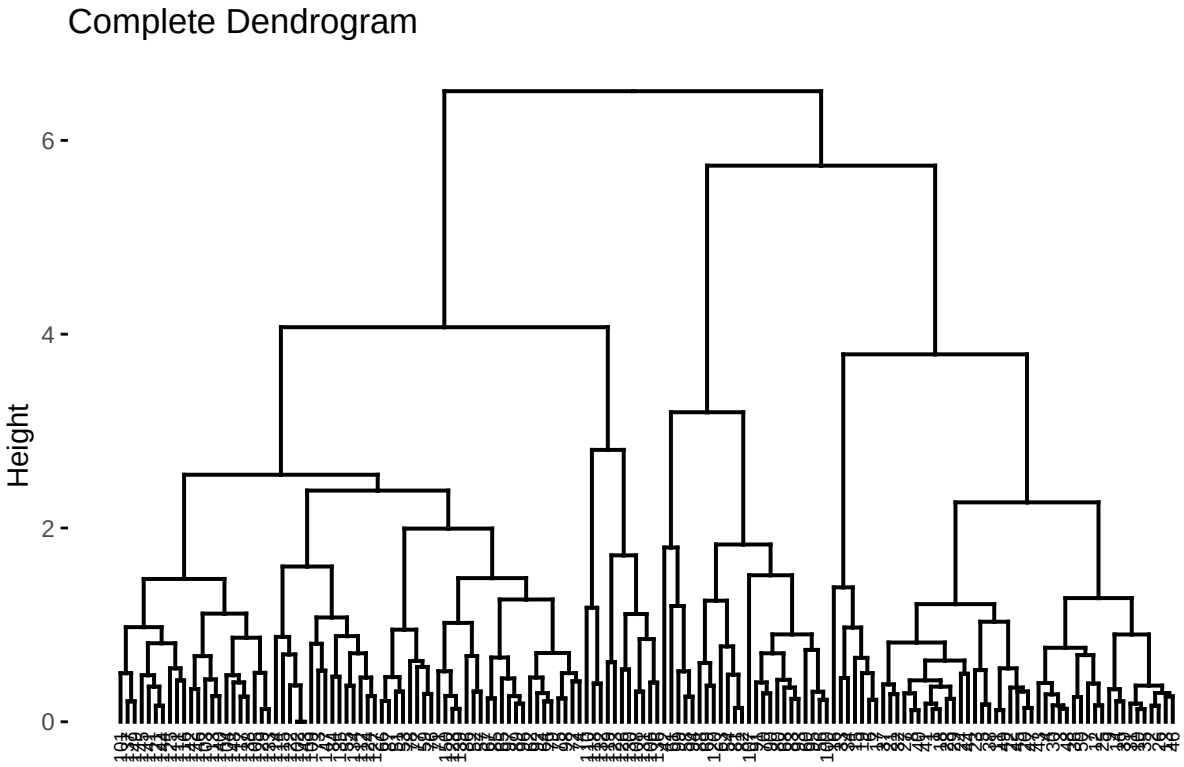
```
scaled_faithful <- scale(faithful)
dist_faithful <- dist(scaled_faithful, method = "euclidean")
dissplot(dist_faithful)
```




The figure above shows two clusters of data, with the black squares clearly indicating these clusters in the top left and top righthand corners.

**Dendrograms: Question 8**

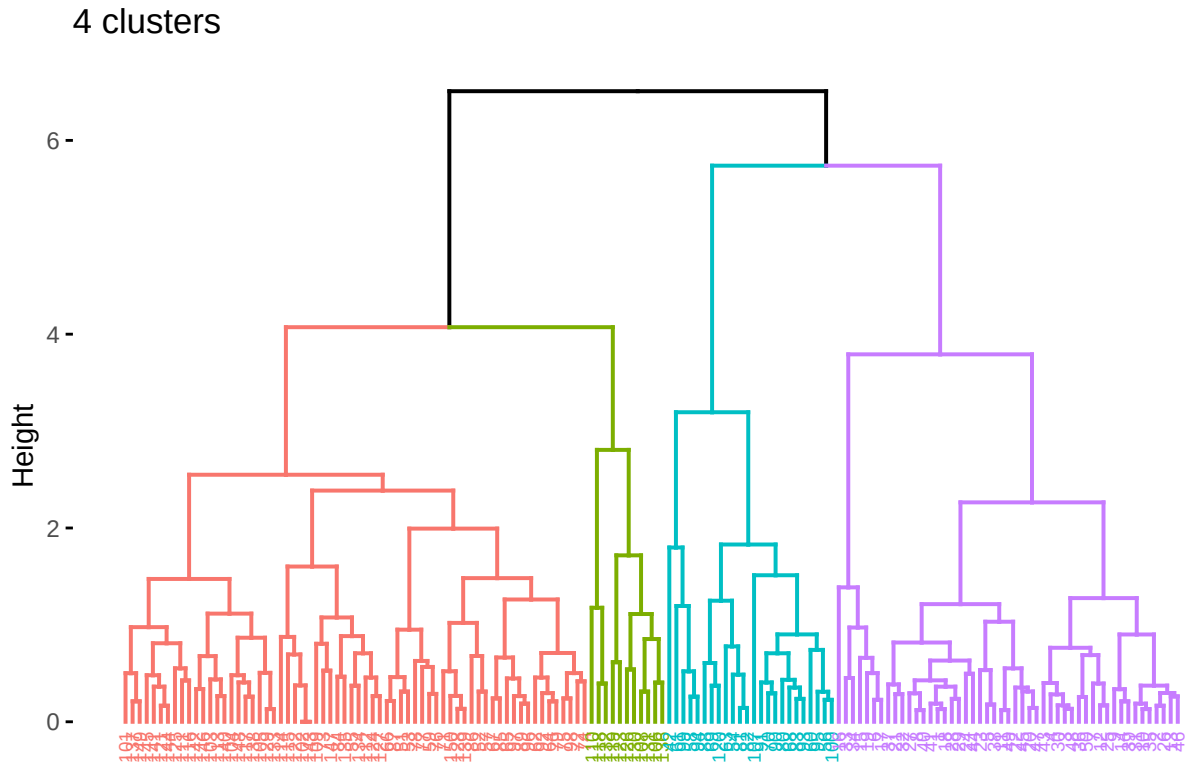
```
scaled_subset <- iris %>%
  select(-Species) %>%
  scale() %>%
  dist()

iris.hc <- hclust(scaled_subset, method = "complete")
fviz_dend(iris.hc, cex=0.5, main= "Complete Dendrogram")
```



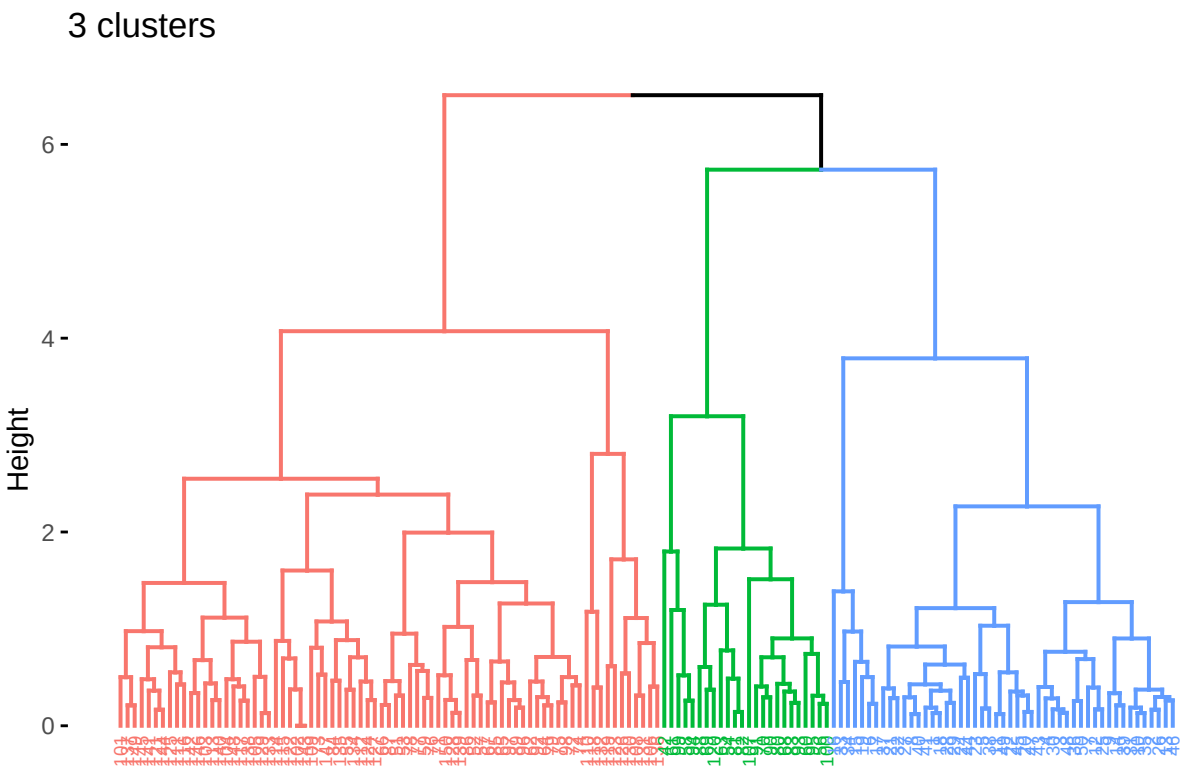

The dendrogram above shows 4 clusters at about the same level. Because it is hard to see on the graph, the labels from left to right are: c(101, 137, 149, 145, 141, 121, 144, 125, 111, 116, 142, 146 103, 113, 140, 104, 148, 117, 138, 105, 129, 133, 114, 115, 122, 102, 143, 109, 73, 147, 84, 135, 55, 134, 112, 124, 127, 66, 87, 51, 53, 78, 77, 59, 76, 71, 150, 128, 139, 86, 52, 57, 67, 85, 65, 97, 89, 96, 62, 92, 64, 79, 75, 98, 72, 74, 110, 118, 132, 119, 123, 126, 130, 108, 131, 106, 136, 42, 61, 99, 58, 94, 88, 69, 120, 63, 54, 81, 82, 107, 91, 70, 90, 80, 68, 83, 93, 60, 95, 56, 100, 16, 33, 34, 15, 19, 6, 17, 37, 21, 32, 27, 8, 40, 41, 1, 18, 28, 29, 24, 44, 23, 5, 38, 11, 49, 22, 45, 20, 47, 43, 4, 30, 3, 48, 36, 50, 7, 12, 25, 9, 14, 39, 31, 10, 35, 2, 26, 13, 46). These labels represent the row position/order of the observations. The height of the dendogram is a little over 6. There are many different sub-partitions of the clusters.

**Tree Cutting: Question 9**

```
fviz_dend(iris.hc, cex= 0.5, k=4, color_labels_by_k = TRUE, main="4 clusters")
```




```
fviz_dend(iris.hc, cex= 0.5, k=3, color_labels_by_k = TRUE, main= "3 clusters")
```

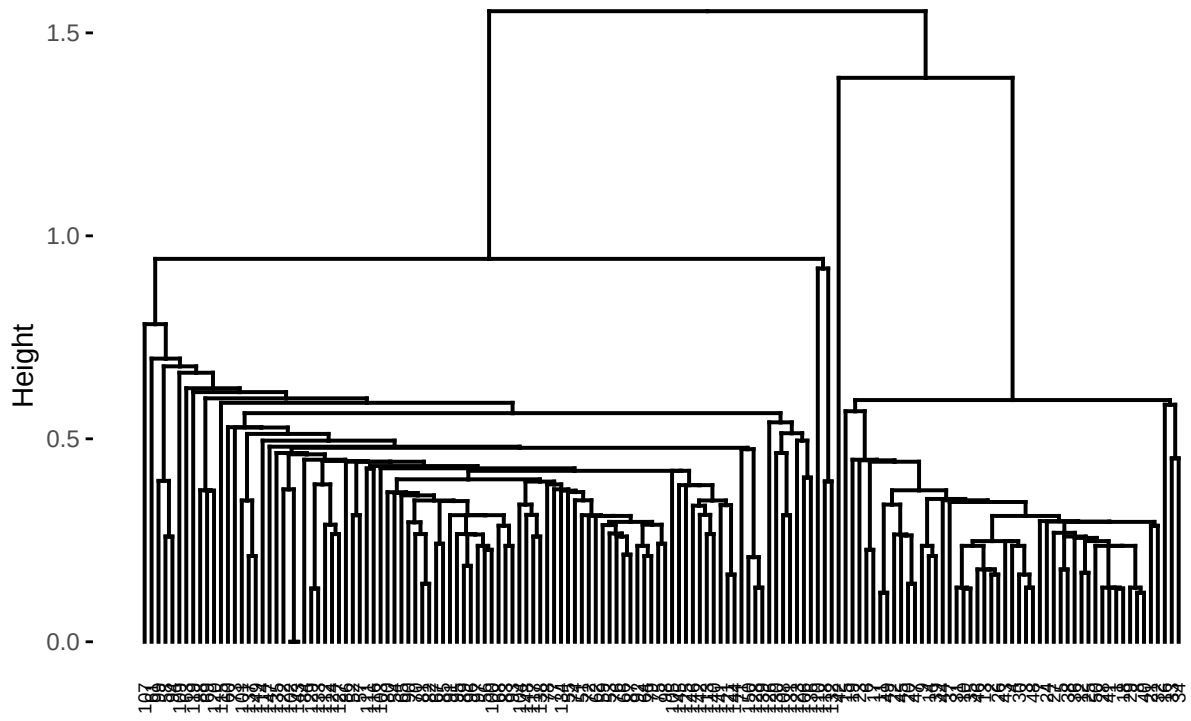



```
# note I did not put them directly side by side because
# the labels were impossible to read in that case.
```

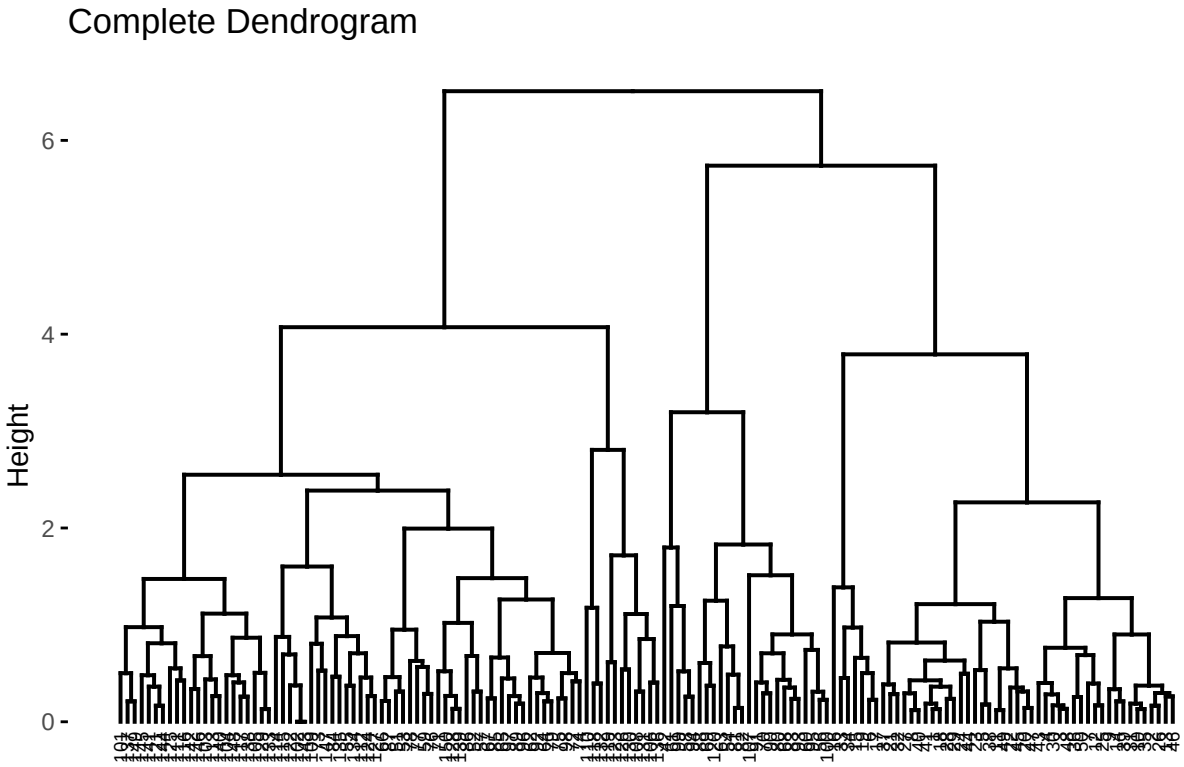
Looking at the dendrogram for cutting of 3 versus 4 branches, we can see that the graph with four clusters splits up the first partition into two sub-partitions, whereas the graph with 3 clusters identifies the three groups at around the same level of height 6. The main difference is that the colored dendrogram with 3 clusters includes the green portion in the red portion, whereas the colored dendrogram with 4 clusters seperates them out.

**Single vs. Complete Linkage: Question 10**

```
iris.hc_single <- hclust(scaled_subset, method = "single")
fviz_dend(iris.hc_single, cex = 0.5, main="Single Dendrogram")
```




```
fviz_dend(iris.hc, cex= 0.5, main="Complete Dendrogram")
```




```
# note I did not put them directly side by side
# because the labels were impossible to read in that case.
```

When comparing the dendrogram produced by the single versus complete method, it is possible to see that the height of the dendogram is much shorter for the single method. The height for the single dendrogram is about 1.5, compared to about 6 for the complete dendrogram. There also seems to be a difference in the number of clusters, as the single dendrogram seems to have two compact clusters, compared to the 3 or 4 clusters in the complete dendrogram. There also appears to be some moving around of the x-axis labels, as the leftmost label for the single dendrogram is now 107 and the rightmost label is 34, whereas the leftmost label for the complete dendrogram was 101 and rightmost label was 46.

**Critical Thinking: Questions 1 and 2**

1. One way to diagnose clusterability is by visual inspection. This can be done informally using scatterplots and ordered dissimilarity images (ODIs) as shown above. For this technique, I would be looking to see if visually I can see distinct groups of the data. A more formal method of diagnosing clusterability would be to test spatial randomness using the hopkins statistic, which calculates whether the data was generated following a random distribution, or if it was generated non-randomly. If it is non-random that may suggest clustering. For this technique, I would compare the actual data to a synthetically created random sample which has the same standard deviation as the actual data. Based on this, I would either reject or accept the null hypothesis that the data are uniformly distributed. These techniques can be used together to motivate clustering. If we do not find clusterability using any of these techniques, it would probably be unwise to proceed with clustering, because at that point we would be making an unfair assumption about the distribution of the data.

2. The paper I read was "Unsupervised SIFT-based Face Recognition Using an Automatic Hierarchical Agglomerative Clustering Solution" by Tudor Barbu. The author's approach was to use hierarchical agglomerative clustering to classify features from images of faces. This was done for the purpose of facial recognition in the absence of a training set of faces. The first step the author took was to extract the facial features from each image and store them into a matrix. Then the author used a novel distance measure to determine which clusters the features belong to. The author noted that traditional distance measures could not be used because of the nature of the dataset, so they proposed a new distance measure be calculated. They did not mention any efforts taken to diagnose clusterability before applying the clustering algorithm. I am not entirely sure how that would have impacted their results because it is unclear to me how they would go about diagnosing clusterability for this type of dataset. The authors note that a potential future extension of their research would be to use the clustering algorithm for face indexing in databases, and then subsequently retrieve faces based on this index.