Assignment 5

Brody Coyne

2022-11-15

```
library(cluster)
library(stats)
library(factoextra)
```

```
## Loading required package: ggplot2
```

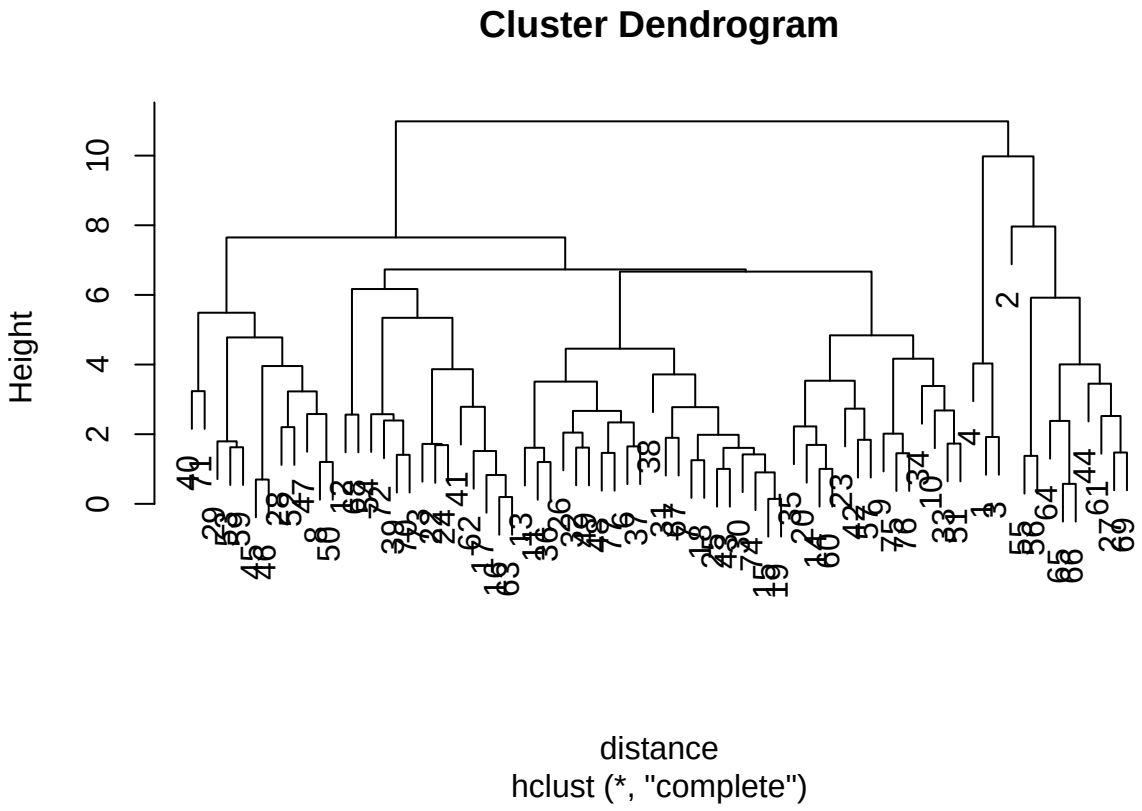
```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

Importing the csv file, pulling out ony the numerical data, getting rid of the missing data, and then normalizing the data. The data should be normalized because the weights of the data points are arbitrary and would have to be weighted manually to be useful.

```
cereals<-read.csv("cereals.csv")
numericaldata = data.frame(cereals[,4:16])
OmitMissing = na.omit(numericaldata)
Normalize = scale(OmitMissing)
```

Using the normalized data to do heirarchical clustering.

```
distance<-dist(Normalize, method = "euclidian")
clustering_heirarchical = hclust(distance, method = "complete")
plot(clustering_heirarchical)
```



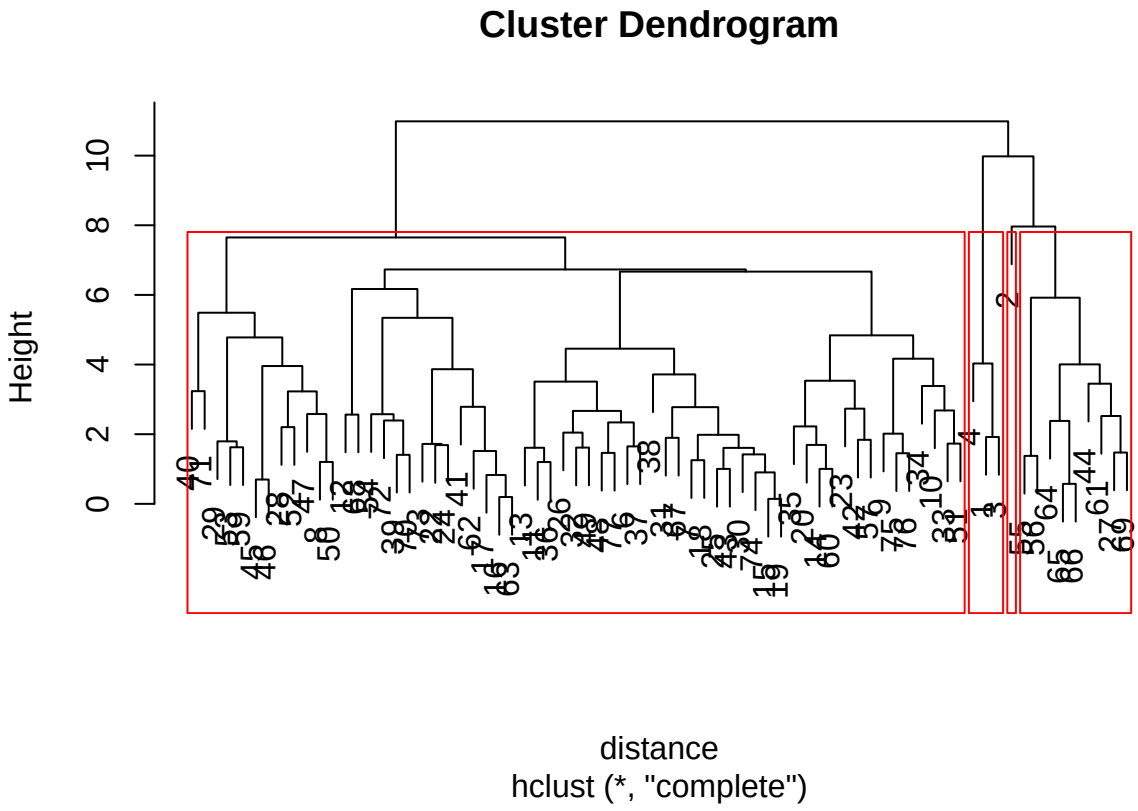
Rounding all of the values to 3 decimal places.

```
round(clustering_heirarchical$height, 3)
```

```
##  [1]  0.143  0.196  0.575  0.698  0.828  0.904  1.003  1.004  1.201  1.203
## [11]  1.254  1.378  1.408  1.421  1.454  1.463  1.474  1.517  1.608  1.611
## [21]  1.616  1.625  1.650  1.687  1.692  1.720  1.730  1.795  1.839  1.897
## [31]  1.919  1.982  2.015  2.046  2.203  2.224  2.339  2.381  2.394  2.522
## [41]  2.563  2.574  2.579  2.668  2.682  2.734  2.776  2.787  3.229  3.236
## [51]  3.385  3.451  3.510  3.535  3.717  3.866  3.957  4.005  4.031  4.168
## [61]  4.456  4.779  4.839  5.342  5.488  5.920  6.169  6.669  6.731  7.650
## [71]  7.964  9.979 10.984
```

-Using hierarchical clustering and seperating the values into 4 groups.

```
plot(clustering_heirarchical)
rect.hclust(clustering_heirarchical, k = 4, border = "red")
```



Using agnes to compare the different clustering methods. Because ward's method is the highest value at .90, that means that it is the best of the 4 methods.

```
singleCH = agnes(Normalize, method = "single")
completeCH = agnes(Normalize, method = "complete")
averageCH = agnes(Normalize, method = "average")
wardCH = agnes(Normalize, method = "ward")
print(singleCH$ac)
```

```
## [1] 0.6067859
```

```
print(completeCH$ac)
```

```
## [1] 0.8353712
```

```
print(averageCH$ac)
```

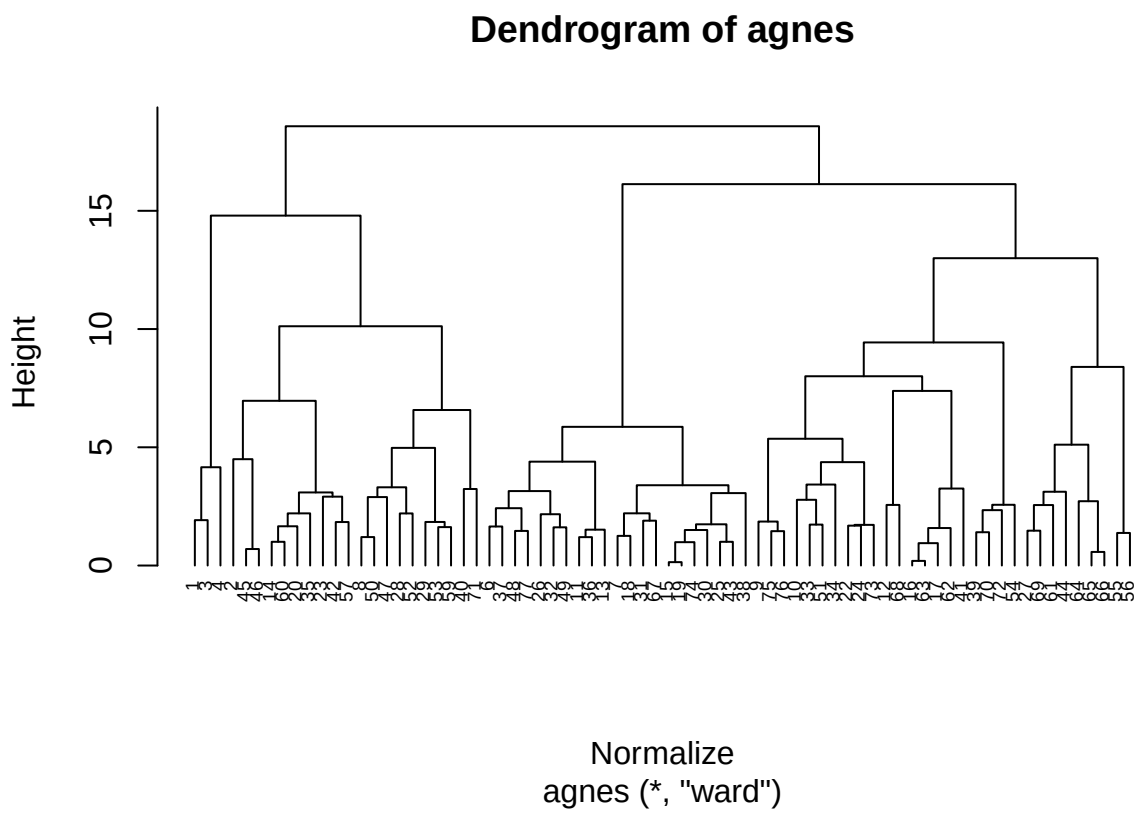
```
## [1] 0.7766075
```

```
print(wardCH$ac)
```

```
## [1] 0.9046042
```

Using ward's method for clustering.

```
pltree(wardCH, cex = 0.6, hang = -1, main = "Dendrogram of agnes")
```



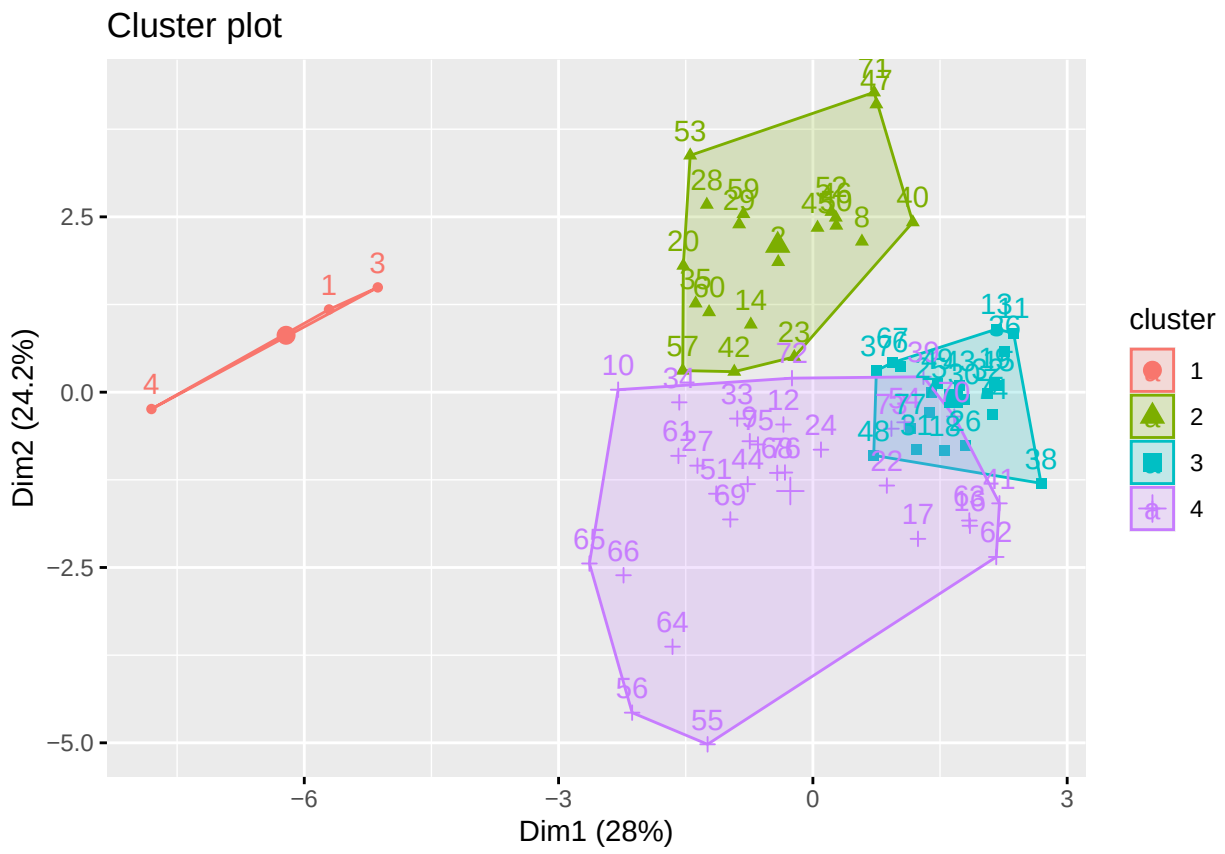
Using ward's method to cluster the data into 4 subgroups.

```
Hcluster1 <- hclust(distance, method = "ward.D2")
subgrp <- cutree(Hcluster1, k = 4)
table(subgrp)
```

```
## subgrp
##  1  2  3  4
##  3 20 21 30
```

A vizualization of the clusters.

```
cereals <- as.data.frame(cbind(Normalize, subgrp))
fviz_cluster(list(data = Normalize, cluster = subgrp))
```



Adding an extra row to show which clustering group the different cereals are a part of.

```
Newcereals = numericaldata
Newcereals_omit = na.omit(Newcereals)
Clust = cbind(Newcereals_omit, subgrp)
Clust[Clust$subgrp==1,]
```

```
##   calories protein fat sodium fiber carbo sugars potass vitamins shelf weight
## 1       70       4   1    130    10     5      6    280       25     3      1
## 3       70       4   1    260     9     7      5    320       25     3      1
## 4       50       4   0    140    14     8      0    330       25     3      1
##   cups    rating subgrp
## 1 0.33 68.40297      1
## 3 0.33 59.42551      1
## 4 0.50 93.70491      1
```

Finding the average of the rating category for each cluster. This shows that subgroup 1 is significantly higher that the other 3 groups. This means that if an elementary school was looking to choose a cluster for healthy cereals, I would suggest subgroup 1. This subgroup includes 100% Bran, 100% Natural Bran, and All-Bran with Extra Fiber.

```
mean(Clust[Clust$subgrp==1, "rating"])
```

```
## [1] 73.84446
```

```r
mean(Clust[Clust$subgrp==2, "rating"])
```

```
## [1] 38.26161
```

```r
mean(Clust[Clust$subgrp==3, "rating"])
```

```
## [1] 28.84825
```

```r
mean(Clust[Clust$subgrp==4, "rating"])
```

```
## [1] 51.43111
```