Daniel Gomes

John Gomes

Blake Simmons

CIS490 Sectional Project III Report

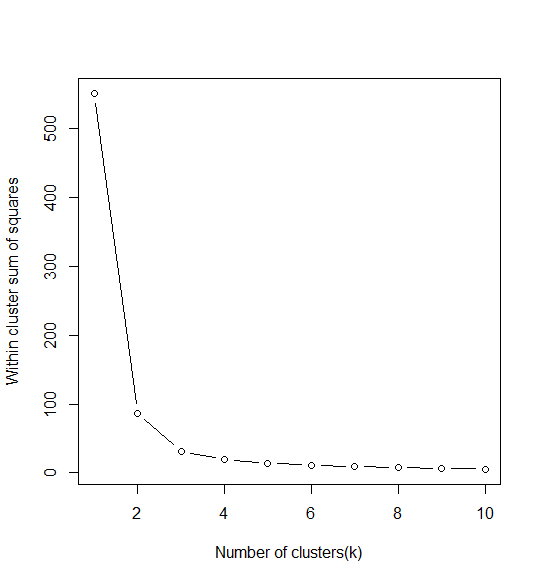
**Code and output located at end of report.**

**1. The clustering methods**

We chose to implement K-Means and Hierarchical clustering. We chose these two because the Iris dataset is simple enough where K-Means and Hierarchical clustering resulted in pure enough clustering. GMM was not necessary because the clusters were not overlapping. We were also familiar with both K-Means and Hierarchical clustering from another class’s coursework and have implemented it in Python.

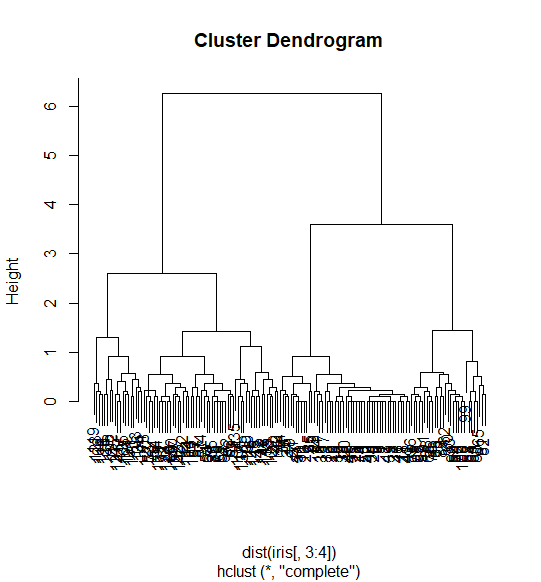
**2. Optimal number of clusters**

For K-Means, in order to find the optimal number of clusters we made an elbow method plot:



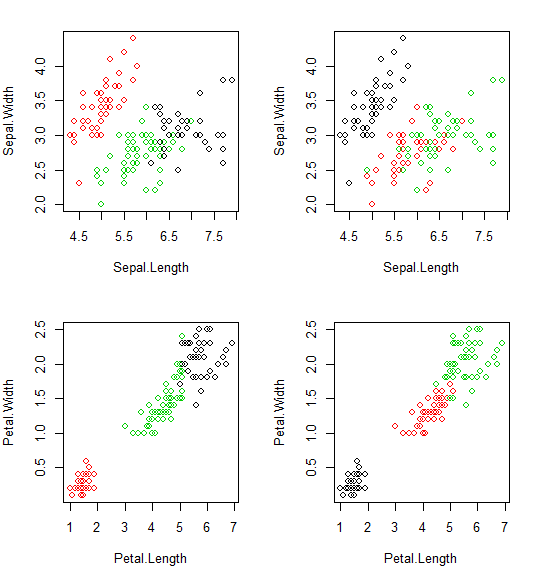
From the plot, we concluded that the optimal number of clusters is 3. That is the point where we begin to see the plot begin to stabilize, that is why we chose it.

For hierarchical clustering, we could also use the elbow method. But we decided to construct a dendrogram using complete linkage to see if it followed the elbow method plot. Here’s the dendrogram:



From the dendrogram, we can also see that around height 3 - 4 is where it begins to split. So it verifies again that we should probably use 3 clusters.

**3. Cluster Accuracy Comparison**

We can compare the two clustering methods by computing accuracy based on each clustering methods confusion matrix. K-means clustering resulted in 88.74% accuracy and Hierarchical clustering resulted in 96% accuracy. Obviously Hierarchical clustering was more accurate in this case.

**K - Means:**

setosa versicolor virginica

1 0 2 36

2 50 0 0

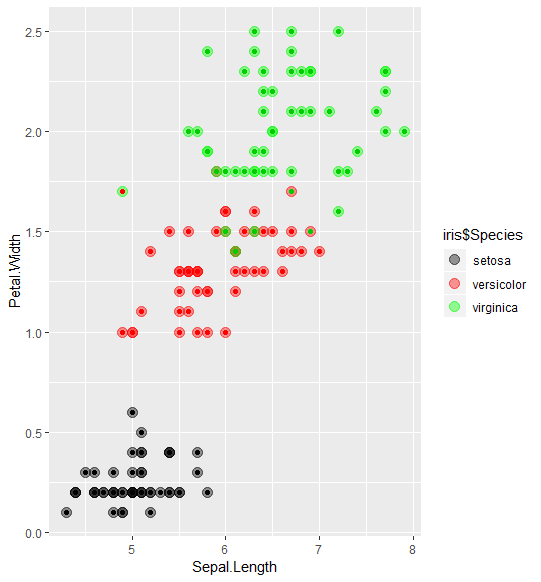
3 0 48 14

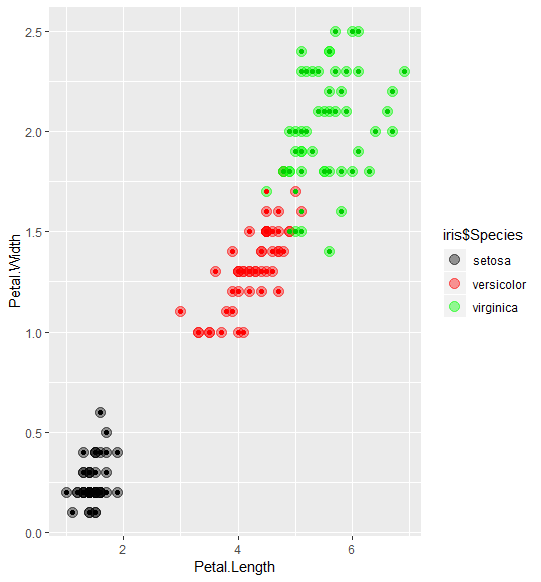
Compute accuracy for K-Means:

50 + 48 + 36 = 134

2 + 14 = 16

accuracy = 134 / (134 + 16) = 0.8874 or **88.74%** accurate

**Hierarchical Clustering:**

setosa versicolor virginica

1 50 0 0

2 0 45 1

3 0 5 49

Compute accuracy for Hclust:

50 + 45 + 49 = 144

5 + 1 = 6

accuracy = 144 / (144 + 6) = 0.96 or **96%** accurate

**4. Pseudocode**

K-means Pseudocode

clusters <- kmeans(data, number of clusters)

**5. Citations**

“Iris Dataset - Clustering Using K Means.” *RPubs*, rpubs.com/AnanyaDu/361293.

Kodali, Teja. “Hierarchical Clustering in R.” *DataScience* , 28 Apr. 2017, datascienceplus.com/hierarchical-clustering-in-r/.

**Code and Output**

**K-Means Clustering**

**# Daniel Gomes, John Gomes, Blake Simmons**

**# CIS 490: Sectional Project 3**

**# K-Means algorithm**

**# Load iris dataset**

**require("datasets")**

**data("iris") # load Iris Dataset**

**summary(iris)**

**set.seed(490)**

**# Plot elbow method to find best k**

**k.max <- 10**

**wss<- sapply(1:k.max,function(k){kmeans(iris[,3:4],k,nstart = 20,iter.max = 20)$tot.withinss})**

**wss**

**plot(1:k.max,wss, type= "b", xlab = "Number of clusters(k)", ylab = "Within cluster sum of squares")**

**# remove class label and store to new variables**

**iris.new<- iris[,c(1,2,3,4)]**

**iris.class<- iris[,"Species"]**

**# do k means algorithm**

**iris.kmeans <- kmeans(iris.new,3) #apply k-means algorithm with no. of centroids(k)=3**

**par(mfrow=c(2,2), mar=c(5,4,2,2))**

**plot(iris.new[c(1,2)], col=iris.kmeans$cluster)# Plot to see how Sepal.Length and Sepal.Width data points have been distributed in clusters**

**plot(iris.new[c(1,2)], col=iris.class)# Plot to see how Sepal.Length and Sepal.Width data points have been distributed originally as per "class" attribute in dataset**

**plot(iris.new[c(3,4)], col=iris.kmeans$cluster)# Plot to see how Petal.Length and Petal.Width data points have been distributed in clusters**

**plot(iris.new[c(3,4)], col=iris.class)**

**table(iris.kmeans$cluster,iris.class)**

**# 50 + 48 + 36 = 134**

**# 2 + 14 = 16**

**# accuracy = 134 / (134 + 16) = 0.8874 or 88.74% accurate**

**Hierarchical Clustering**

**> # Daniel Gomes, John Gomes, Blake Simmons**

**> # CIS 490: Sectional Project 3**

**> # Hclust algorithm**

**>**

**> # Load iris dataset**

**> require("datasets")**

**> data("iris") # load Iris Dataset**

**>**

**> library(mclust)**

**> library(ggplot2)**

**>**

**> set.seed(77)**

**>**

**> # Hierarchical clustering using Complete Linkage**

**> iris.hclust <- hclust(dist(iris[,3:4]))**

**> plot(iris.hclust) # Plot the obtained dendrogram**

**>**

**> # determined the denogram, we want to use 3 clusters**

**> iris.clusterCut <- cutree(iris.hclust, 3)**

**> table(iris.clusterCut, iris$Species)**

**iris.clusterCut setosa versicolor virginica**

**1 50 0 0**

**2 0 21 50**

**3 0 29 0**

**> # 50 + 21 + 50 = 121**

**> # 29**

**> # accuracy = 121 / (121 + 29) = 0.806 or 80.6% accurate**

**>**

**> # Hierarchical clustering using average linkage**

**> iris.hclust <- hclust(dist(iris[, 3:4]), method = 'average')**

**> plot(iris.hclust) # Plot the obtained dendrogram**

**>**

**> # determined the denogram, we want to use 3 clusters**

**> iris.clusterCut <- cutree(iris.hclust, 3)**

**> table(iris.clusterCut, iris$Species)**

**iris.clusterCut setosa versicolor virginica**

**1 50 0 0**

**2 0 45 1**

**3 0 5 49**

**> # 50 + 45 + 49 = 144**

**> # 5 + 1**

**> # accuracy = 144 / (144 + 6) = 0.96 or 96% accurate**

**>**

**> # plot petal.length and petal.width**

**> ggplot(iris, aes(Petal.Length, Petal.Width, color = iris$Species)) +**

**+ geom\_point(alpha = 0.4, size = 3.5) + geom\_point(col = iris.clusterCut) +**

**+ scale\_color\_manual(values = c('black', 'red', 'green'))**

**>**

**> # plot sepal.length and petal.width**

**> ggplot(iris, aes(Sepal.Length, Petal.Width, color = iris$Species)) +**

**+ geom\_point(alpha = 0.4, size = 3.5) + geom\_point(col = iris.clusterCut) +**

**+ scale\_color\_manual(values = c('black', 'red', 'green'))**