K-Means Clustering

Mike Levine

Statistical Learning and Data Mining (STAT 5603-401, Spring 2019)

K means clustering on the iris dataset

The iris dataset, popularized by British statistician and biologist Ronald Fisher, contains data about *sepal length*, *sepal width*, *petal length*, and *petal width* of flowers of 3 different species. Let's pretend we don't know the species; we'll attempt to cluster the dataset into the correct species by using the variables available to us.

First, let's load the dataset into R:

```
library(datasets)
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

First, we can establish that the Euclidian distance is appropriate for measuring the distance between two points in this dataset. We know this because our values ('Sepal.Length', 'Sepal.Width', etcetera) are continuous variables. This is a precondition for performing k-means clustering.

Next, let's perform k-means clustering with k=2, 3 and 4 on the data and plot the resulting clusters. We'll also make sure we don't include the species variable in our clustering!

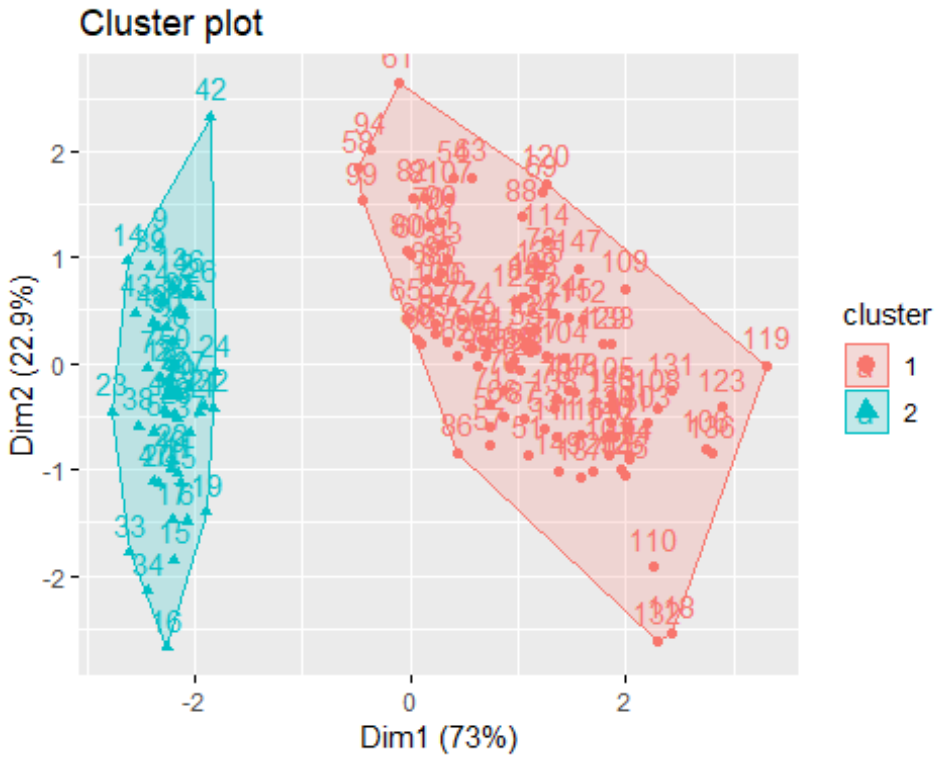
```
library(tidyverse)

library(cluster)

library(factoextra)

rm(list=ls())
df <- iris
df <- df[,1:4]
df = na.omit(df)
df <- scale(df)
head(df)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1   -0.8976739  1.01560199    -1.335752   -1.311052
## 2   -1.1392005 -0.13153881    -1.335752   -1.311052
## 3   -1.3807271  0.32731751    -1.392399   -1.311052
## 4   -1.5014904  0.09788935    -1.279104   -1.311052
## 5   -1.0184372  1.24503015    -1.335752   -1.311052
## 6   -0.5353840  1.93331463    -1.165809   -1.048667
```

```
k2 <- kmeans(df, centers = 2, nstart = 25)
k3 <- kmeans(df, centers = 3, nstart = 25)
k4 <- kmeans(df, centers = 4, nstart = 25)

str(k2)

str(k3)

str(k4)
```

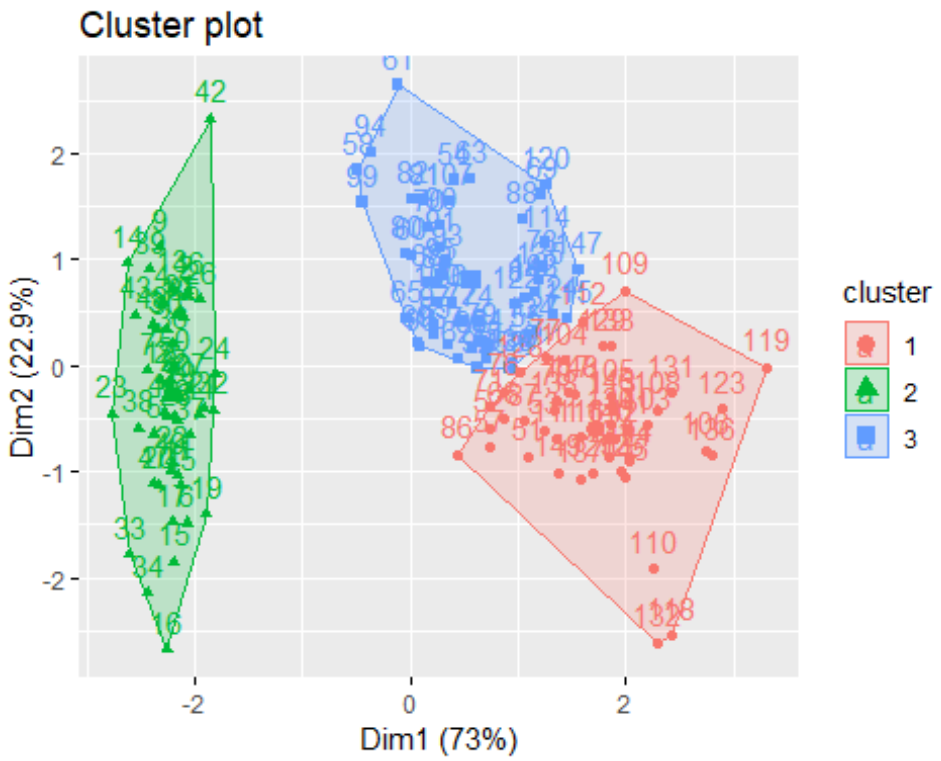
Now that we've setup our code appropriately, let's see our clusterings for k=2.

```
k2

fviz_cluster(k2, data = df)
```



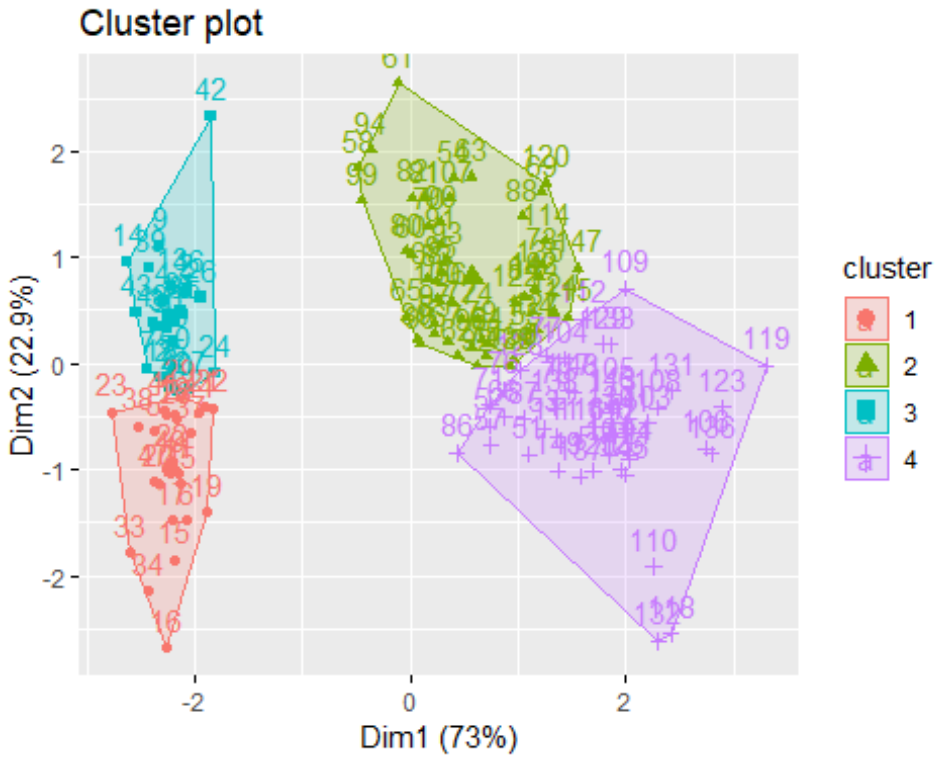
Next, let's take a look at our clustering with k=3.

```
k3

fviz_cluster(k3, data = df)
```



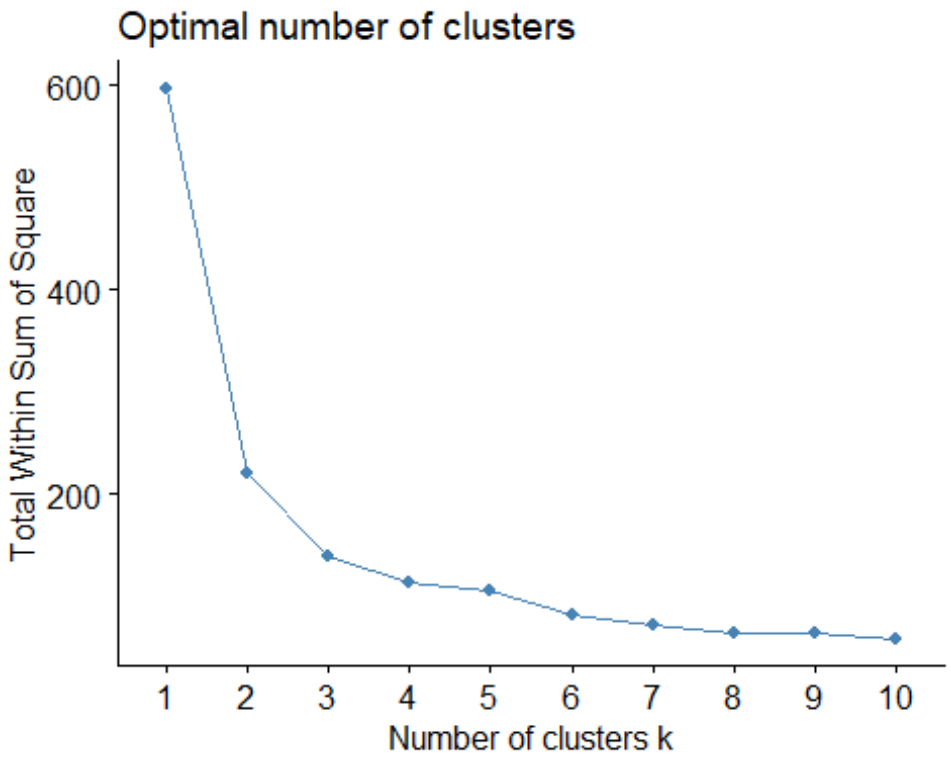
And lastly, let's look at our clustering with k=4.

```
k4

fviz_cluster(k4, data = df)
```



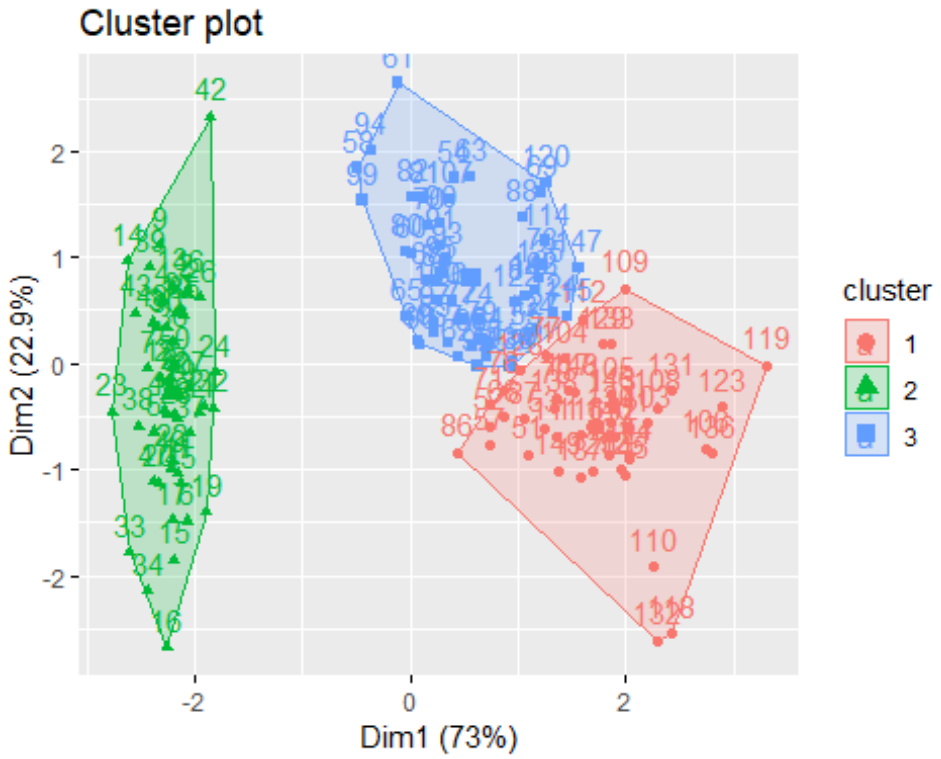
With three different options, it can seem a little difficult to choose what 'k' should be! Luckily, we have a way to pick the optimal value of 'k.' To do this, we'll use a method called the Elbow Method.

```
set.seed(123)

fviz_nbclust(df, kmeans, method = "wss")
```



The above graph illustrates our optimal 'k' value using the Elbow Method (the term 'elbow' is used because we essentially pick the value at the 'elbow' of the curve). Picking our k value can occasionally be subjective depending on where we believe the elbow is. In this case, we can safely argue that the elbow is at three.

```
k3

fviz_cluster(k3, data = df)
```



Thus, we can say that there are three distinct species of flowers in the dataset. If we do some research on the 'iris' dataset, we would see that this is the correct answer.