A Clustering Analysis of Iris Flower Data

Laura Schultz

30 May 2019

This report describes a k-means clustering analysis of a dataset consisting 150 data points, each with four predictor variables and one categorical response. The predictors are the sepal width, sepal length, petal width, and petal length of an iris flower, and the response is the iris species. I obtained this dataset from the UCI Machine Learning Repository (https://archive.ics.uci.edu/ml/datasets/Iris). Note that the response values make it possible to assess how well a specific clustering method performed; they were not used to build my clustering models.

```
#Clear the global environment
rm(list = ls())
#Load the iris dataset into a file named iris_data
iris_data <- read.table("4.2irisSummer2018.txt", header = TRUE)
#Take a look at the first few rows of the dataset
head(iris_data)
```

```
##   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
```

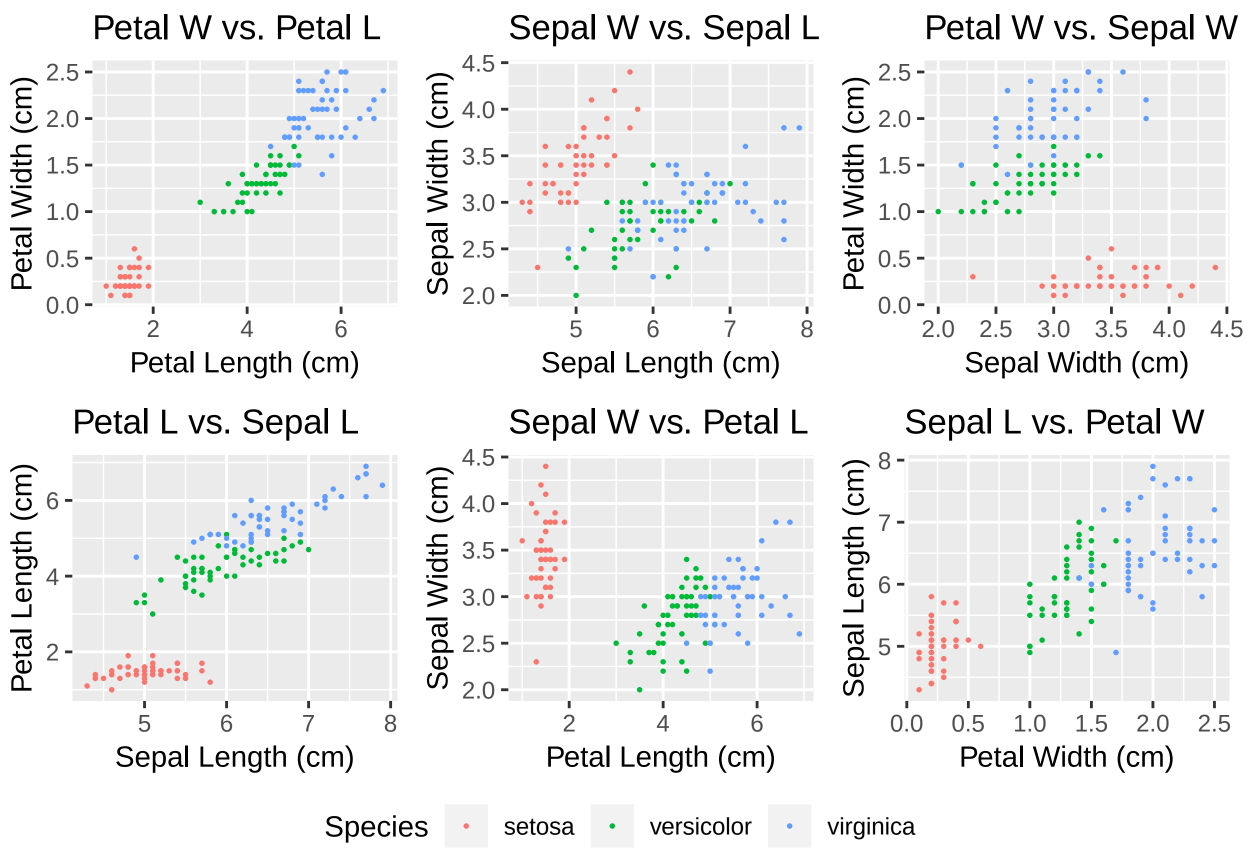
```
#Look at summary statistics for each of the variables in this dataset
summary(iris_data)
```

```
##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width   
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100  
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300  
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300  
##  Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199  
##  3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800  
##  Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500  
##        Species  
##  setosa    :50  
##  versicolor:50  
##  virginica :50  
##                 
##                 
##                 
```

Note that there are 50 irises of each of three different species (*setosa*, *versicolour*, and *virginica*) included in this dataset, for a total of 150 data points.

```
#Plot a variety of combinations of factors to look for clusters
#according to known species classifications

#Plot petal width vs. petal length (colors = existing species classifications)
 gg1 <- ggplot(iris_data, aes(Petal.Length, Petal.Width, color = Species)) +
   geom_point(size = 0.3) + ggtitle("Petal W vs. Petal L") +
```

```r
   labs(x = "Petal Length (cm)", y = "Petal Width (cm)")

#Plot sepal width vs. sepal length (colors = existing species classifications)
gg2 <- ggplot(iris_data, aes(Sepal.Length, Sepal.Width, color = Species)) +
  geom_point(size = 0.3) + ggtitle("Sepal W vs. Sepal L") +
  labs(x = "Sepal Length (cm)", y = "Sepal Width (cm)")

#Plot petal width vs. sepal width (colors = existing species classifications)
gg3 <- ggplot(iris_data, aes(Sepal.Width, Petal.Width, color = Species)) +
  geom_point(size = 0.3) + ggtitle("Petal W vs. Sepal W") +
  labs(x = "Sepal Width (cm)", y = "Petal Width (cm)")

#Plot petal length vs. sepal length (colors = existing species classifications)
gg4 <- ggplot(iris_data, aes(Sepal.Length, Petal.Length, color = Species)) +
  geom_point(size = 0.3) + ggtitle("Petal L vs. Sepal L") +
  labs(x = "Sepal Length (cm)", y = "Petal Length (cm)")

#Plot sepal width vs. petal length (colors = existing species classifications)
gg5 <- ggplot(iris_data, aes(Petal.Length, Sepal.Width, color = Species)) +
  geom_point(size = 0.3) + ggtitle("Sepal W vs. Petal L") +
  labs(x = "Petal Length (cm)", y = "Sepal Width (cm)")

#Plot sepal length vs. petal width (colors = existing species classifications)
gg6 <-ggplot(iris_data, aes(Petal.Width, Sepal.Length, color = Species)) +
  geom_point(size = 0.3) + ggtitle("Sepal L vs. Petal W") +
  labs(x = "Petal Width (cm)", y = "Sepal Length (cm)")

ggarrange(gg1, gg2, gg3, gg4, gg5, gg6, common.legend = TRUE, legend = "bottom")
```



Looking at this display, it appears that the different combinations of factors yielded clusters that differ in terms of how well the colored dots, which correspond to the existing iris species designations (*setosa*, *versicolor*, and *virginica*), are separated. It seems that the red dots, which correspond to *setosa* irises, are clearly separated from the other two species no matter what combination of factors I used to plot the data. There seems to be varying degrees of overlap between the blue and green dots based on which two factors I plotted, with the plot of sepal length vs. sepal width showing the most overlap between the points. When picking the factors to include in my clustering model, I decided that it would be best to pick only those factors that are correlated with each (i.e., those that result in a linear scatterplot). Judging from these scatterplots, sepal width is not strongly correlated with any of the other three factors; the scatterplots which include sepal width as a factor do not appear to be as linear as those plotting combinations of the other three factors. I confirmed this impression using the following R code.

```r
cor(iris_data[,1:4]) #Shows linear correlation between each pair of factors
```

```
##              Sepal.Length Sepal.Width Petal.Length Petal.Width
## Sepal.Length    1.0000000  -0.1175698    0.8717538   0.8179411
## Sepal.Width    -0.1175698   1.0000000   -0.4284401  -0.3661259
## Petal.Length    0.8717538  -0.4284401    1.0000000   0.9628654
## Petal.Width     0.8179411  -0.3661259    0.9628654   1.0000000
```

If I define “strong” positive correlations as those with r $\geq$ 0.7, it seems like there are indeed strong correlations between all of the possible pairings of the factors except those pairings that include sepal width as a factor, with the strongest correlation being that between petal length and petal width (r = 0.963). Hence, I am going to compare clustering models that include all four factors, three factors (petal length, petal width, and sepal length), and two factors (petal length and petal width) in my search for the “best” k-means clustering model.

I noticed that the petal width measurements (range: 0.1 to 2.5 cm) are quite a bit smaller than the other measurements (sepal length: 4.3 to 7.9 cm, sepal width: 2.0 to 4.4 cm, and petal length: 1.0 to 6.9 cm), so my next step was to scale the data and then re-plot it. I chose to scale the data using z-score standardization.

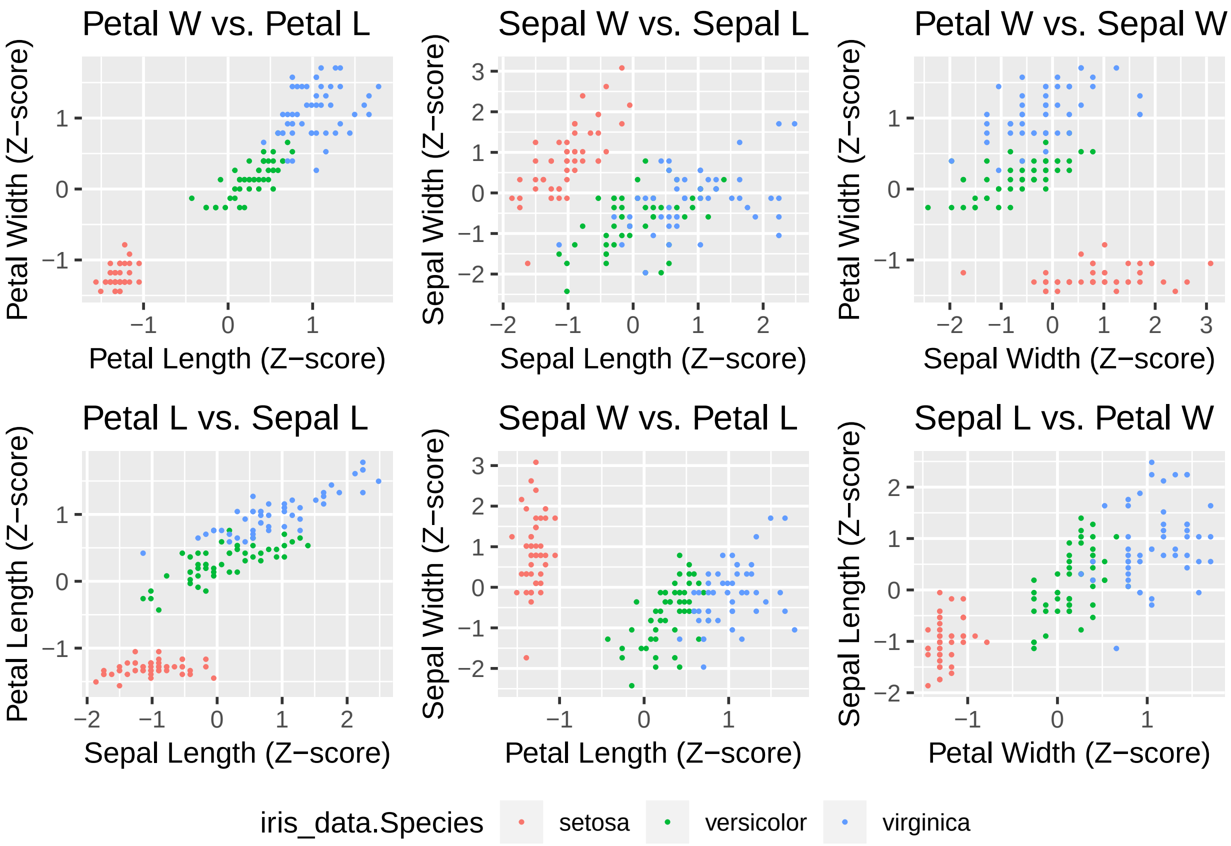
```
iris_data_NormZ <- as.data.frame(scale(iris_data[1:4])) #Scale the numeric data
#Attach the species name column to the new data frame containing the scaled data
iris_data_NormZ_species <-data.frame(iris_data_NormZ, iris_data$Species)
#Look at scaled data with species column added
head(iris_data_NormZ_species)
```

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

Having scaled the data, I re-did my graphical display and checked to see if anything changed in the visual appearance of the scatterplots when z-scores were used instead of the actual iris measurements to plot the data.

```
#Plot a variety of combinations of factors to look for clusters in the scaled data

#Plot petal width vs. petal length (colors = existing species classifications)
 gg7 <- ggplot(iris_data_NormZ_species, aes(Petal.Length, Petal.Width,
                                            color = iris_data.Species)) +
   geom_point(size = 0.3) + ggtitle("Petal W vs. Petal L") +
   labs(x = "Petal Length (Z-score)", y = "Petal Width (Z-score)")

#Plot sepal width vs. sepal length (colors = existing species classifications)
gg8 <- ggplot(iris_data_NormZ_species, aes(Sepal.Length, Sepal.Width,
                                           color = iris_data.Species)) +
  geom_point(size = 0.3) + ggtitle("Sepal W vs. Sepal L") +
  labs(x = "Sepal Length (Z-score)", y = "Sepal Width (Z-score)")

#Plot petal width vs. sepal width (colors = existing species classifications)
gg9 <- ggplot(iris_data_NormZ_species, aes(Sepal.Width, Petal.Width,
                                           color = iris_data.Species)) +
  geom_point(size = 0.3) + ggtitle("Petal W vs. Sepal W") +
  labs(x = "Sepal Width (Z-score)", y = "Petal Width (Z-score)")

#Plot petal length vs. sepal length (colors = existing species classifications)
gg10 <- ggplot(iris_data_NormZ_species, aes(Sepal.Length, Petal.Length,
                                            color = iris_data.Species)) +
  geom_point(size = 0.3) + ggtitle("Petal L vs. Sepal L") +
  labs(x = "Sepal Length (Z-score)", y = "Petal Length (Z-score)")

#Plot sepal width vs. petal length (colors = existing species classifications)
gg11 <- ggplot(iris_data_NormZ_species, aes(Petal.Length, Sepal.Width,
                                            color = iris_data.Species)) +
  geom_point(size = 0.3) + ggtitle("Sepal W vs. Petal L") +
  labs(x = "Petal Length (Z-score)", y = "Sepal Width (Z-score)")

#Plot sepal length vs. petal width (colors = existing species classifications)
gg12 <-ggplot(iris_data_NormZ_species, aes(Petal.Width, Sepal.Length,
                                           color = iris_data.Species)) +
  geom_point(size = 0.3) + ggtitle("Sepal L vs. Petal W") +
  labs(x = "Petal Width (Z-score)", y = "Sepal Length (Z-score)")
```

```r
ggarrange(gg7, gg8, gg9, gg10, gg11, gg12, common.legend = TRUE, legend = "bottom")
```



The scatterplots of the scaled data look essentially identical to the ones I made using the original unscaled data, suggesting that the iris petal and sepal dimensions are close enough in terms of the range of measurements (i.e., they were on the same order of magnitude) that scaling wasn't truly necessary. Also, I confirmed that the linear correlation coefficients didn't change when they were calculated using scaled data. I decided to use both scaled and unscaled data when building my k-means clustering models to see if the results were influenced based on whether or not I used scaled data.

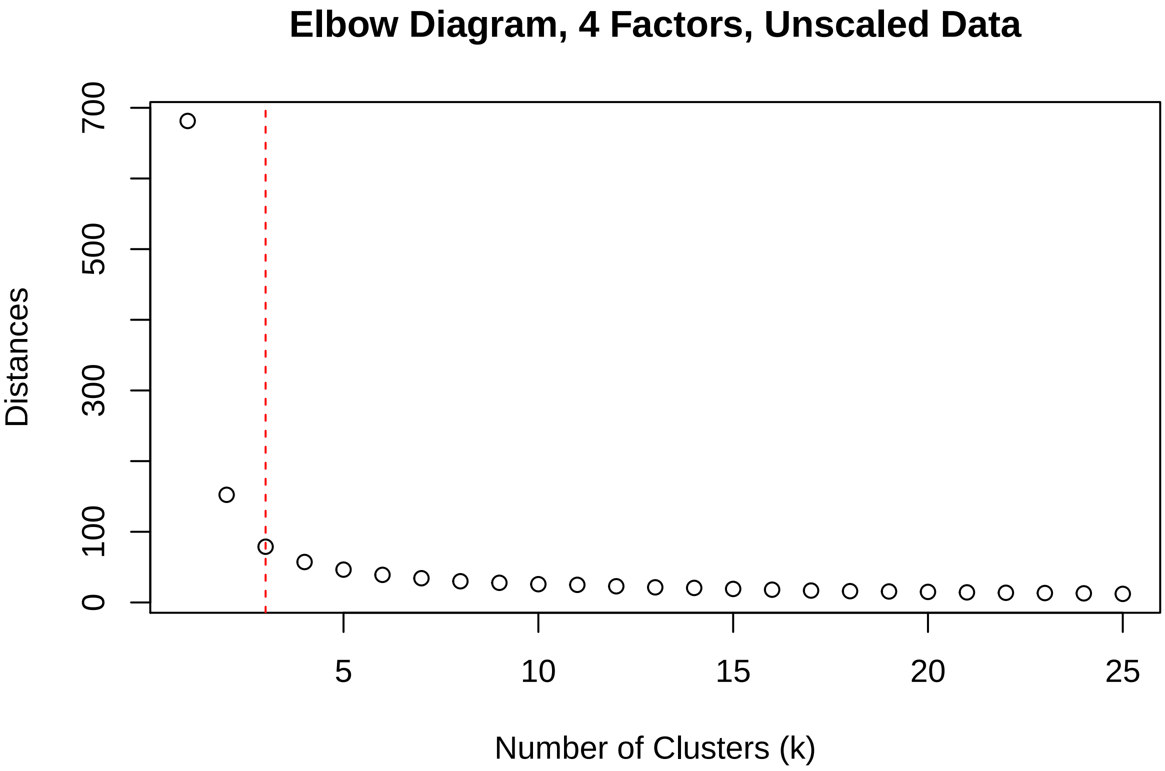
```r
#Shows correlation between each pair of factors using scaled data
cor(iris_data_NormZ_species[,1:4])
```

```
##              Sepal.Length Sepal.Width Petal.Length Petal.Width
## Sepal.Length    1.0000000  -0.1175698    0.8717538   0.8179411
## Sepal.Width    -0.1175698   1.0000000   -0.4284401  -0.3661259
## Petal.Length    0.8717538  -0.4284401    1.0000000   0.9628654
## Petal.Width     0.8179411  -0.3661259    0.9628654   1.0000000
```

First, I looked at what happened when I used all four predictors in a model using unscaled data.

```r
#Find a good number of clusters for a model using all 4 predictors (unscaled data)
Distances <- rep(0,25)
for (k_clusters in 1:25){
  set.seed(36)
  clusters <- kmeans(iris_data[,1:4], k_clusters, nstart = 20)
  Distances[k_clusters] <- clusters$tot.withinss
}
```

```
#Plot an elbow diagram
plot(Distances, xlab = "Number of Clusters (k)", main =
       "Elbow Diagram, 4 Factors, Unscaled Data")
abline(v = 3, lty = 2, col = "red") #Draw a vertical line at k = 3
```



The elbow diagram produced for k-means clustering models using all four predictors (sepal length, sepal width, petal length, and petal width) and unscaled measurements suggested that k = 3 was the best choice for the number of clusters to use in my model (which isn't surprising given that the dataset includes three known iris species). Next, I produced a table comparing the species classifications of the 150 data points with the three clusters that my model produced.

```
set.seed(36)
iris_clusters1 <- kmeans(iris_data[,1:4], 3, nstart = 20)
#See how clusters compare to three known iris species
table(iris_clusters1$cluster, iris$Species)
```

```
##
##     setosa versicolor virginica
##   1     50          0         0
##   2      0         48        14
##   3      0          2        36
```

```
cat("The total within SS for k = 3 is", iris_clusters1$tot.withinss, "\n")
```

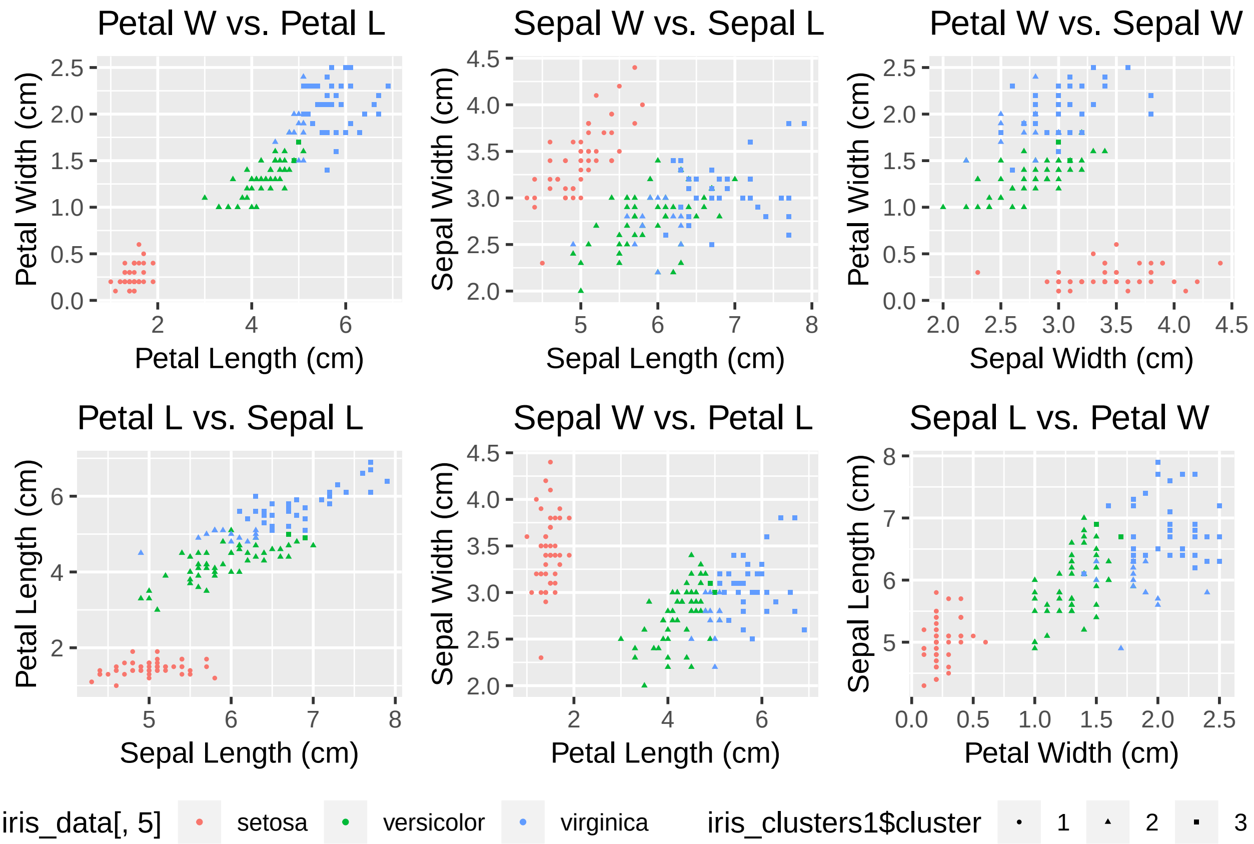
```
## The total within SS for k = 3 is 78.85144
```

```
cat("The k-means clustering model explains ",
    (iris_clusters1$betweenss / iris_clusters1$totss) *100,
    "% of the variability in the data.")
```

```
## The k-means clustering model explains  88.42753 % of the variability in the data.
```

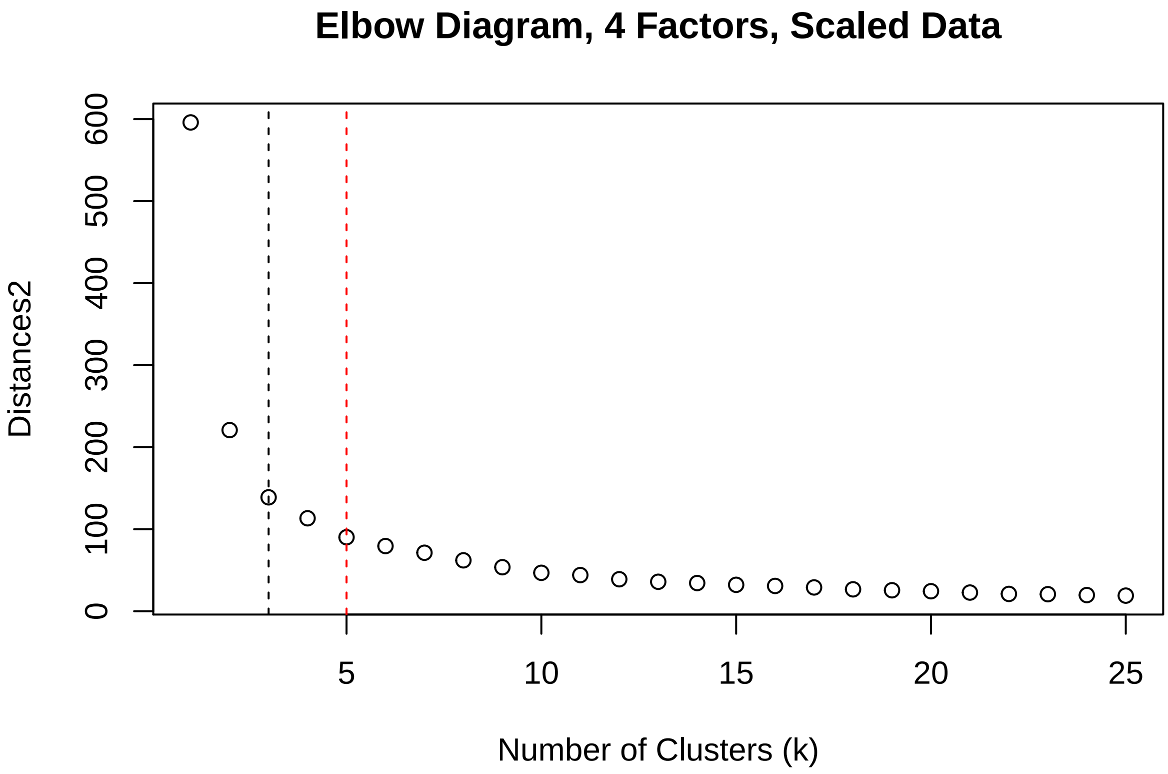
This table indicates that all 50 of the *setosa* irises in the dataset were assigned to cluster 1. The other two species were split between clusters 2 and 3, with cluster 2 including primarily *versicolor* irises and cluster 3 containing mostly *virginica* irises. Cluster 2 contained 48 *versicolor* irises and 14 *virginica* irises, whereas cluster 3 contained 36 *virginica* irises and 2 *versicolor* irises. My 3-means clustering model had a total within SS of 78.85 and explained 88.43% of the variability in the dataset. Next, I decided to re-do the plots I created earlier using colors corresponding to the three known species and plot symbols corresponding to the clusters identified by my 3-means clustering model. Doing so allowed me to visualize which data points were assigned to clusters that differed from the species classifications provided in the last column of the dataset.

```r
#Plot a variety of combinations of factors to look at results of my clustering model
#Color = species designation, plot symbol = cluster assignment, k = 3
#Unscaled data
#Designate the cluster numbers as factors
iris_clusters1$cluster <- as.factor(iris_clusters1$cluster)

#Plot petal width vs. petal length (colors = clusters assigned by model)
 gg1a <- ggplot(iris_data, aes(Petal.Length, Petal.Width, shape = iris_clusters1$cluster,
                               color = iris_data[,5])) +
   geom_point(size = 0.5) + ggtitle("Petal W vs. Petal L") +
   labs(x = "Petal Length (cm)", y = "Petal Width (cm)")

#Plot sepal width vs. sepal length (colors = clusters assigned by model)
gg2a <- ggplot(iris_data, aes(Sepal.Length, Sepal.Width, shape = iris_clusters1$cluster,
                               color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Sepal W vs. Sepal L") +
  labs(x = "Sepal Length (cm)", y = "Sepal Width (cm)")

#Plot petal width vs. sepal width (colors = clusters assigned by model)
gg3a <- ggplot(iris_data, aes(Sepal.Width, Petal.Width,  shape = iris_clusters1$cluster,
                               color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Petal W vs. Sepal W") +
  labs(x = "Sepal Width (cm)", y = "Petal Width (cm)")

#Plot petal length vs. sepal length (colors = clusters assigned by model)
gg4a <- ggplot(iris_data, aes(Sepal.Length, Petal.Length, shape = iris_clusters1$cluster,
                               color = iris_data[,5]))  +
  geom_point(size = 0.5) + ggtitle("Petal L vs. Sepal L") +
  labs(x = "Sepal Length (cm)", y = "Petal Length (cm)")

#Plot sepal width vs. petal length (colors = clusters assigned by model)
gg5a <- ggplot(iris_data, aes(Petal.Length, Sepal.Width, shape = iris_clusters1$cluster,
                               color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Sepal W vs. Petal L") +
  labs(x = "Petal Length (cm)", y = "Sepal Width (cm)")

#Plot sepal length vs. petal width (colors = clusters assigned by model)
gg6a <-ggplot(iris_data, aes(Petal.Width, Sepal.Length, shape = iris_clusters1$cluster,
                               color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Sepal L vs. Petal W") +
  labs(x = "Petal Width (cm)", y = "Sepal Length (cm)")

ggarrange(gg1a, gg2a, gg3a, gg4a, gg5a, gg6a, common.legend = TRUE, legend = "bottom")
```



My next step was to look at an elbow diagram for k-means clustering models produced using all four predictors again, but using scaled data instead of the original measurements.

```
#Find a good number of clusters for a model using
#all 4 predictors (scaled data)
Distances2 <- rep(0,25)
for (k_clusters in 1:25){
  set.seed(36)
  clusters <- kmeans(iris_data_NormZ_species[,1:4], k_clusters, nstart = 20)
  Distances2[k_clusters] <- clusters$tot.withinss
}

#Plot an elbow diagram
plot(Distances2, xlab = "Number of Clusters (k)",
     main = "Elbow Diagram, 4 Factors, Scaled Data")
abline(v = 3, lty = 2, col = "black") #Draw a vertical line at k = 3
abline(v = 5, lty = 2, col = "red") #Draw a vertical line at k = 5
```



Interestingly, there was a difference in the appearance of my two elbow diagrams, with the "kink" being tougher to identify in the second elbow diagram. Based on the elbow diagram for k-means clustering models built using scaled data and all four predictors, I initially decided that k = 3 was the best choice (see the black dashed line on the elbow diagram above).

```
set.seed(36)
iris_clusters2 <- kmeans(iris_data_NormZ_species[,1:4], 3, nstart = 20)  #model with k = 3
#See how clusters compare to three known iris species
table(iris_clusters2$cluster, iris_data$Species)
```

```
##
##     setosa versicolor virginica
##   1      0         39        14
##   2     50          0         0
##   3      0         11        36
```

```
cat("The total within SS for k = 3 is", iris_clusters2$tot.withinss, "\n")
```

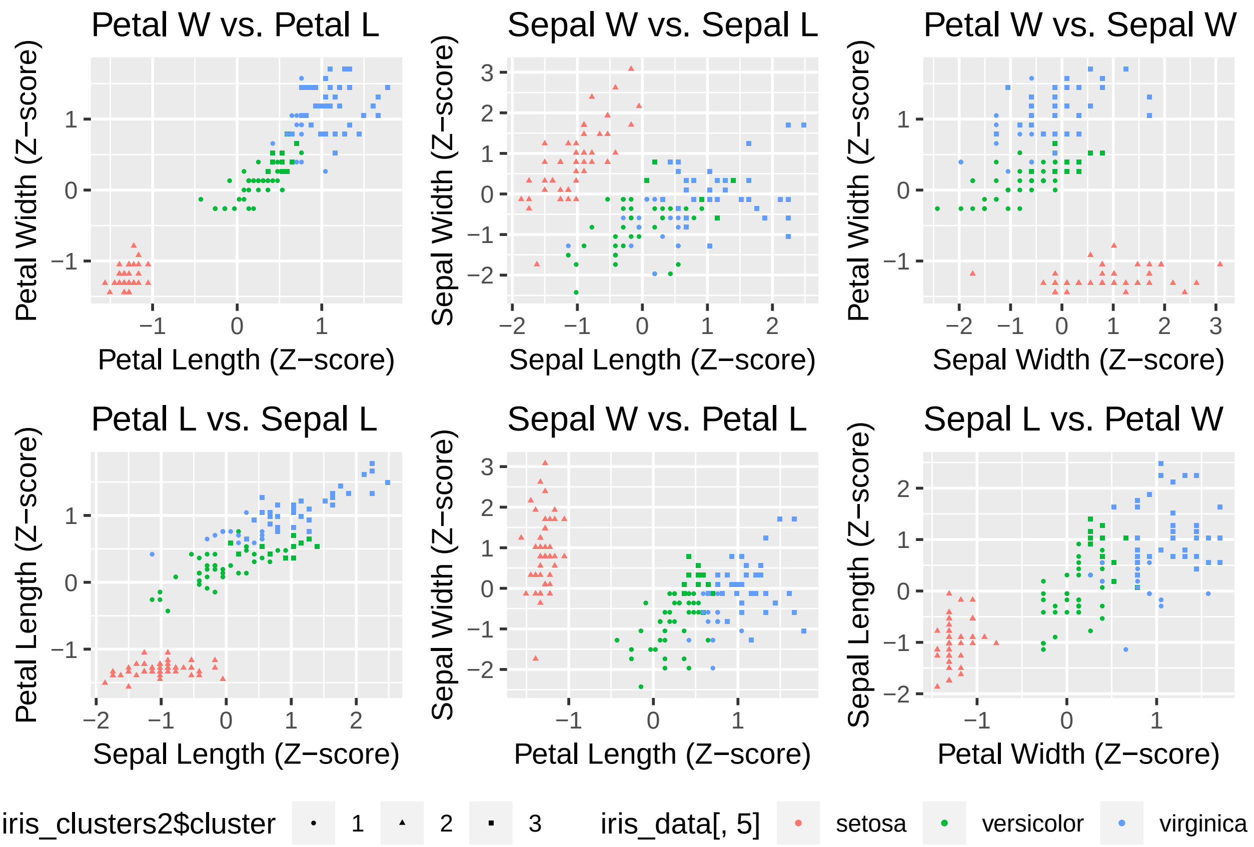
```
## The total within SS for k = 3 is 138.8884
```

```
cat("The 3-means clustering model explains ",
    (iris_clusters2$betweenss / iris_clusters2$totss) *100,
    "% of the variability in the data.")
```

```
## The 3-means clustering model explains  76.69658 % of the variability in the data.
```

The 3-means clustering model produced using the scaled iris data as not be as good as the one produced using the unscaled data. The total within SS this time was 138.89 (compared to 78.85), and the model explained only 76.70% (compared to 88.43%) of the variability in the scaled data. All 50 *setosa* irises were assigned to cluster 2 in the model using scaled data. The *versicolor* irises were split between clusters 1 and 3, with more of them assigned to cluster 1 (39) than to cluster 3 (11). The *virginica* irises were also split between clusters 1 and 3, but with more of them assigned to cluster 3 (36) than to cluster 1 (14). Finally, I plotted the scaled data with the points colored according to the species classification given in the last column of the data table and the plot symbols corresponding to the clusters that the data points were assigned by my 3-means clustering model of the scaled data. This graphical display clearly demonstrates the inferiority of my second 3-means clustering model; the points are not as cleanly separated as they were in the previous display that I generated for the unscaled data.

```r
#Plot a variety of combinations of factors to look at clusters for scaled data
#Color = species designation, plot symbol = cluster assignment, k = 3
#Designate cluster numbers as factors
iris_clusters2$cluster <- as.factor(iris_clusters2$cluster)

#Plot petal width vs. petal length
 gg7a <- ggplot(iris_data_NormZ_species, aes(Petal.Length, Petal.Width,
                shape = iris_clusters2$cluster, color = iris_data[,5])) +
   geom_point(size = 0.5) + ggtitle("Petal W vs. Petal L") +
   labs(x = "Petal Length (Z-score)", y = "Petal Width (Z-score)")

#Plot sepal width vs. sepal length
gg8a <- ggplot(iris_data_NormZ_species, aes(Sepal.Length, Sepal.Width,
                shape = iris_clusters2$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Sepal W vs. Sepal L") +
  labs(x = "Sepal Length (Z-score)", y = "Sepal Width (Z-score)")

#Plot petal width vs. sepal width
gg9a <- ggplot(iris_data_NormZ_species, aes(Sepal.Width, Petal.Width,
                shape = iris_clusters2$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Petal W vs. Sepal W") +
  labs(x = "Sepal Width (Z-score)", y = "Petal Width (Z-score)")

#Plot petal length vs. sepal length
gg10a <- ggplot(iris_data_NormZ_species, aes(Sepal.Length, Petal.Length,
                shape = iris_clusters2$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Petal L vs. Sepal L") +
  labs(x = "Sepal Length (Z-score)", y = "Petal Length (Z-score)")

#Plot sepal width vs. petal length
gg11a <- ggplot(iris_data_NormZ_species, aes(Petal.Length, Sepal.Width,
                shape = iris_clusters2$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Sepal W vs. Petal L") +
  labs(x = "Petal Length (Z-score)", y = "Sepal Width (Z-score)")

#Plot sepal length vs. petal width
gg12a <-ggplot(iris_data_NormZ_species, aes(Petal.Width, Sepal.Length,
                shape = iris_clusters2$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Sepal L vs. Petal W") +
  labs(x = "Petal Width (Z-score)", y = "Sepal Length (Z-score)")

ggarrange(gg7a, gg8a, gg9a, gg10a, gg11a, gg12a, common.legend = TRUE, legend = "bottom")
```



After seeing these results, I decided to revisit the elbow diagram for the scaled data. In retrospect, I decided that my initial interpretation was biased by my knowledge that there are three species of irises included in the dataset. This time, I decided that the bend might actually occur at k = 5 (see the red dashed line on the elbow diagram on page 9), although I still am not sure; the "kink" really isn't all that obvious in this graph. When I checked out the performance of a 5-means clustering model of the scaled data using all four predictors, I observed that the model performance improved. This time, the total within SS was 90.20 (which is lower than 138.89), and the percentage of variability in the scaled data that was explained by the model increased to 84.87% (from 76.70%). Plotting the data again with the five clusters identified using different plot symbols shows an improvement in the separation of the plot symbols, too.

```
set.seed(36)
iris_clusters3 <- kmeans(iris_data_NormZ_species[,1:4], 5, nstart = 20)
#See how clusters compare to three known iris species
table(iris_clusters3$cluster, iris_data$Species)
```

```
##
##     setosa versicolor virginica
##   1     28          0         0
##   2      0          2        27
##   3      0         21         2
##   4     22          0         0
##   5      0         27        21
```

```
cat("The total within SS for k = 5 is", iris_clusters3$tot.withinss, "\n")
```

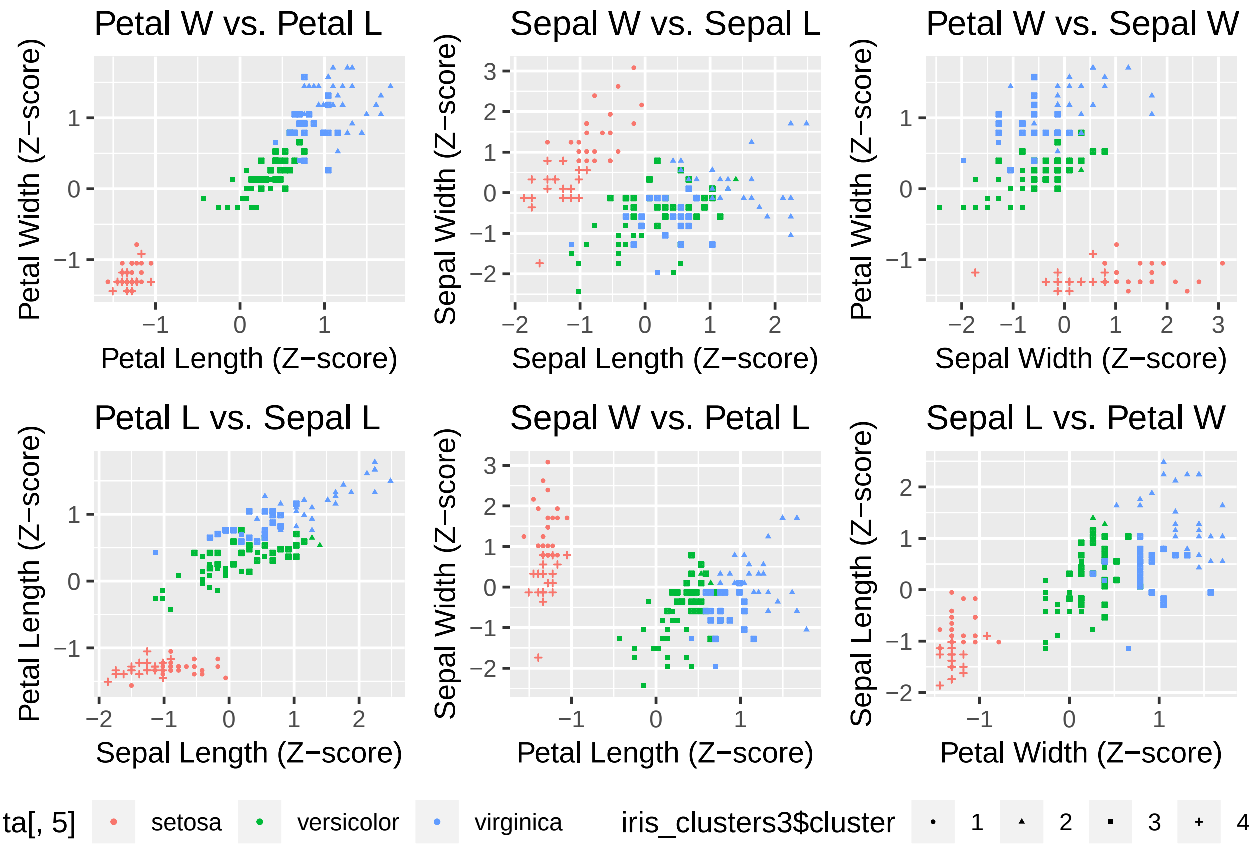
```
## The total within SS for k = 5 is 90.20221
```

```
cat("The 5-means clustering model explains ", (iris_clusters3$betweenss /
                                                 iris_clusters3$totss) *100,
```

```r
    "% of the variability in the data.")
```

```
## The 5-means clustering model explains  84.8654 % of the variability in the data.
```

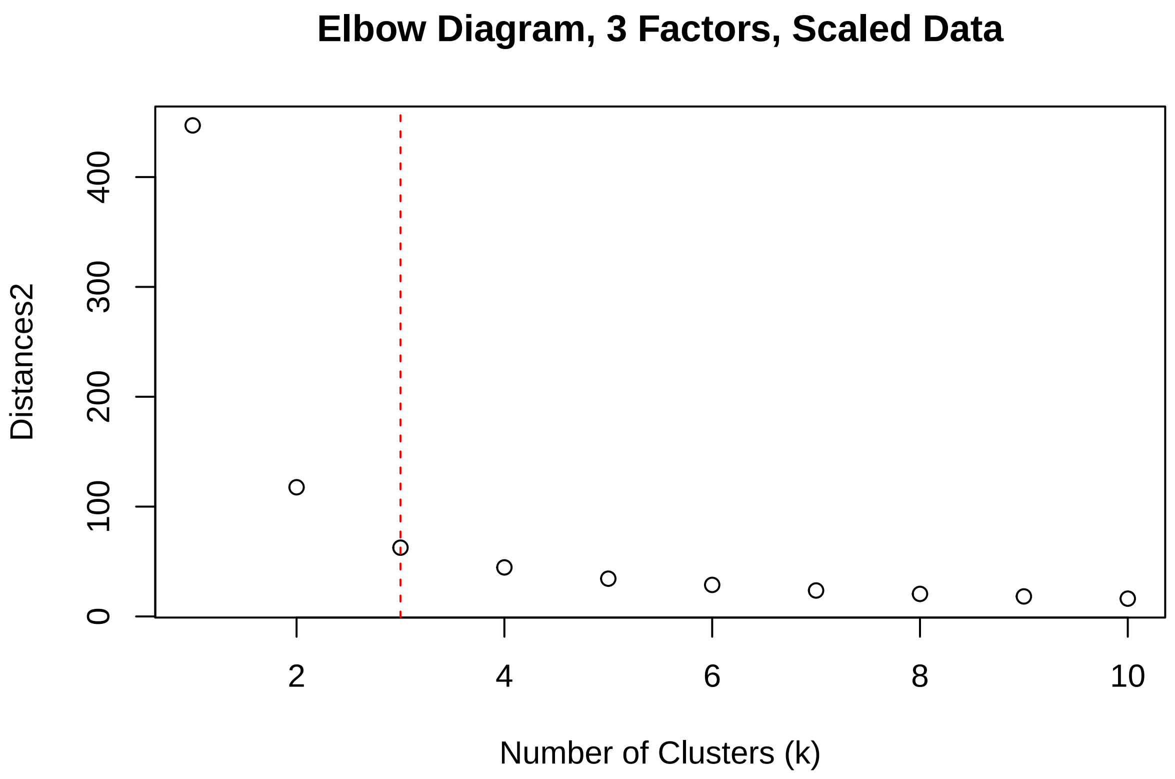
```r
#Plot a variety of combinations of factors to look at clusters
#All 4 predictors used in model
#Color = species, plot shape = cluster assignment, k = 5
iris_clusters3$cluster <- as.factor(iris_clusters3$cluster) #Designate clusters as factors

#Plot petal width vs. petal length
 gg7b <- ggplot(iris_data_NormZ_species, aes(Petal.Length, Petal.Width,
                shape = iris_clusters3$cluster, color = iris_data[,5])) +
   geom_point(size = 0.5) + ggtitle("Petal W vs. Petal L") +
   labs(x = "Petal Length (Z-score)", y = "Petal Width (Z-score)")

#Plot sepal width vs. sepal length
gg8b <- ggplot(iris_data_NormZ_species, aes(Sepal.Length, Sepal.Width,
                shape = iris_clusters3$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Sepal W vs. Sepal L") +
  labs(x = "Sepal Length (Z-score)", y = "Sepal Width (Z-score)")

#Plot petal width vs. sepal width
gg9b <- ggplot(iris_data_NormZ_species, aes(Sepal.Width, Petal.Width,
                shape = iris_clusters3$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Petal W vs. Sepal W") +
  labs(x = "Sepal Width (Z-score)", y = "Petal Width (Z-score)")

#Plot petal length vs. sepal length
gg10b <- ggplot(iris_data_NormZ_species, aes(Sepal.Length, Petal.Length,
                shape = iris_clusters3$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Petal L vs. Sepal L") +
  labs(x = "Sepal Length (Z-score)", y = "Petal Length (Z-score)")

#Plot sepal width vs. petal length
gg11b <- ggplot(iris_data_NormZ_species, aes(Petal.Length, Sepal.Width,
                shape = iris_clusters3$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Sepal W vs. Petal L") +
  labs(x = "Petal Length (Z-score)", y = "Sepal Width (Z-score)")

#Plot sepal length vs. petal width
gg12b <-ggplot(iris_data_NormZ_species, aes(Petal.Width, Sepal.Length,
                shape = iris_clusters3$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Sepal L vs. Petal W") +
  labs(x = "Petal Width (Z-score)", y = "Sepal Length (Z-score)")

ggarrange(gg7b, gg8b, gg9b, gg10b, gg11b, gg12b, common.legend = TRUE, legend = "bottom")
```



In summary, a 3-means clustering model worked best when I used unscaled petal and sepal measurements, whereas a 5-means clustering model was preferable when I scaled the data. In both cases, I included all four predictors supplied in the dataset.

My next step was to repeat my analysis using only three factors (sepal length, petal length, and petal width). In the interest of saving time, I decided to only use scaled data for this analysis. I started by generating an elbow diagram so that I could pick an appropriate value for k.

```
#Find a good number of clusters for a model using
#petal length, petal width, and sepal length as predictors (scaled data)
Distances2 <- rep(0,10)
cols = c(1, 3:4)
for (k_clusters in 1:10){
  set.seed(36)
  clusters <- kmeans(iris_data_NormZ_species[cols], k_clusters, nstart = 20)
  Distances2[k_clusters] <- clusters$tot.withinss
}

#Plot an elbow diagram
plot(Distances2, xlab = "Number of Clusters (k)",
     main = "Elbow Diagram, 3 Factors, Scaled Data")
abline(v = 3, lty = 2, col = "red") #Draw a vertical line at k = 3
```



This elbow diagram suggests that I should use k = 3 clusters when building a model using scaled data with petal length, petal width, and sepal length as the three predictors.

```
set.seed(36)
iris_clusters6 <- kmeans(iris_data_NormZ_species[cols], 3, nstart = 20)
#See how clusters compare to three known iris species
table(iris_clusters6$cluster, iris_data$Species)
```

```
##
##     setosa versicolor virginica
##   1     50          1         0
##   2      0         44        14
##   3      0          5        36
```

```
cat("Using scaled data with petal length, pedal width, and sepal length as factors:", "\n")
```

```
## Using scaled data with petal length, pedal width, and sepal length as factors:
```

```
cat("The total within SS for k = 3 is", iris_clusters6$tot.withinss, "\n")
```

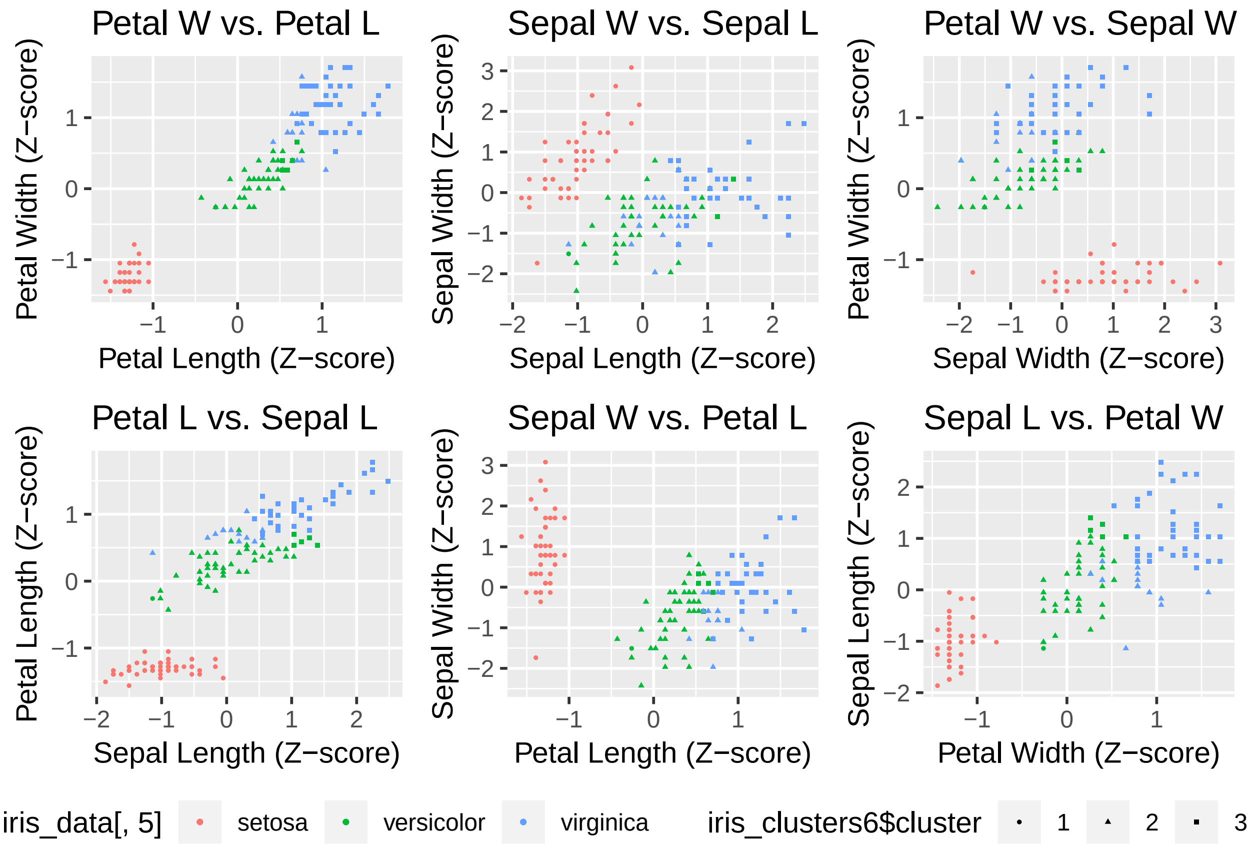
```
## The total within SS for k = 3 is 62.62093
```

```
cat("The 3-means clustering model explains ", (iris_clusters6$betweenss /
                                                iris_clusters6$totss) *100,
    "% of the variability in the data.")
```

```
## The 3-means clustering model explains  85.99084 % of the variability in the data.
```

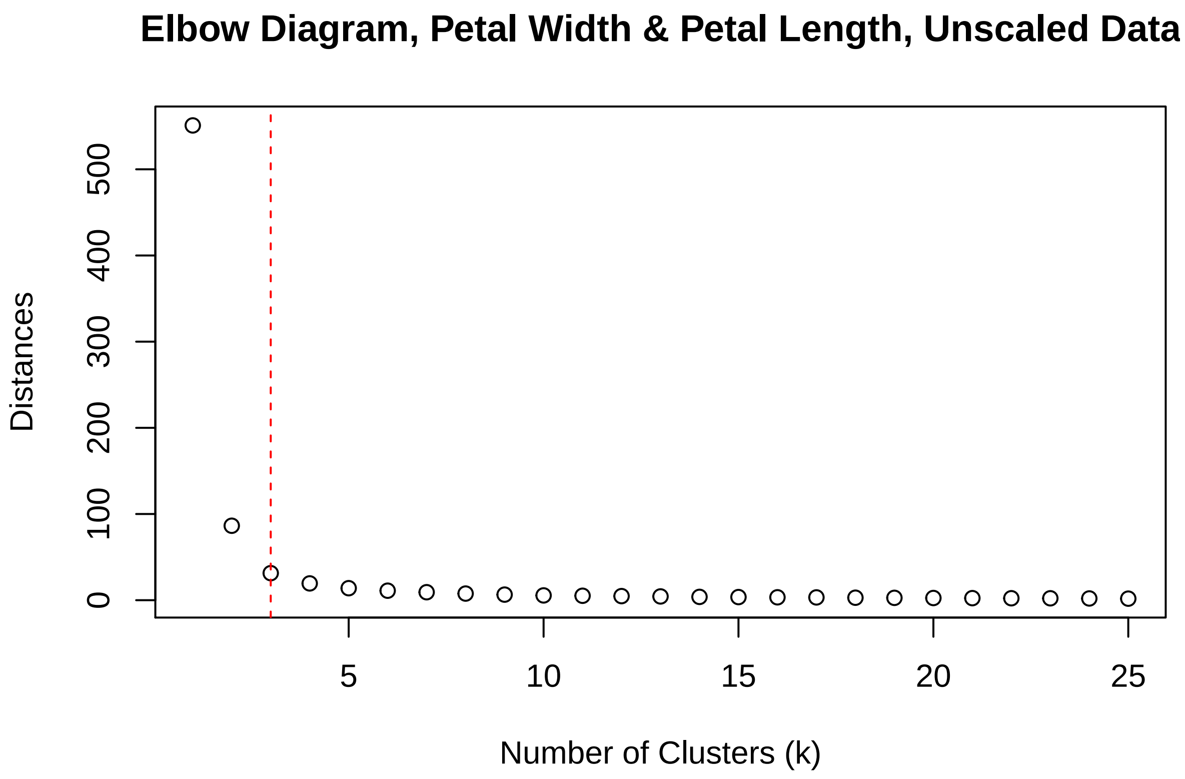
When I used scaled data, the model generated with three predictors and k = 3 had a much lower total within SS (62.62) than the model of the scaled data using all four predictors and k = 3 (138.89), suggesting an improvement in terms of internal model accuracy when sepal width was not included as a predictor. Likewise, this model explained more of the variability in the scaled data (85.99%) than did the model using all four predictors (76.70%). However, this model was less accurate in terms of the cluster assignments aligning with the given species classifications. All 50 *setosa* irises plus 1 *veriscolor* iris were assigned to cluster 1 in this model. Cluster 2 contained 44 *versicolor* irises and 14 *virginica* irises, and cluster 3 contained 36 *virginica* irises and 5 *versicolor* irises. Removing sepal width from the model seemed to result in the *versicolor* irises being split up more between the clusters than what I observed in my earlier models. The following graphical display shows how the given species designations compared to the clusters assigned by my 3-means clustering model of the scaled data using petal length, petal width, and sepal length as predictors.

```r
#Plot a variety of combinations of factors to look at clusters
#Petal length, petal width, and sepal length used as predictor
#Color = species, plot shape = cluster assignment, k = 3
iris_clusters6$cluster <- as.factor(iris_clusters6$cluster) #Designate clusters as factors

#Plot petal width vs. petal length
 gg7c <- ggplot(iris_data_NormZ_species, aes(Petal.Length, Petal.Width,
                shape = iris_clusters6$cluster, color = iris_data[,5])) +
   geom_point(size = 0.5) + ggtitle("Petal W vs. Petal L") +
   labs(x = "Petal Length (Z-score)", y = "Petal Width (Z-score)")

#Plot sepal width vs. sepal length
gg8c <- ggplot(iris_data_NormZ_species, aes(Sepal.Length, Sepal.Width,
                shape = iris_clusters6$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Sepal W vs. Sepal L") +
  labs(x = "Sepal Length (Z-score)", y = "Sepal Width (Z-score)")

#Plot petal width vs. sepal width
gg9c <- ggplot(iris_data_NormZ_species, aes(Sepal.Width, Petal.Width,
                shape = iris_clusters6$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Petal W vs. Sepal W") +
  labs(x = "Sepal Width (Z-score)", y = "Petal Width (Z-score)")

#Plot petal length vs. sepal length
gg10c <- ggplot(iris_data_NormZ_species, aes(Sepal.Length, Petal.Length,
                shape = iris_clusters6$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Petal L vs. Sepal L") +
  labs(x = "Sepal Length (Z-score)", y = "Petal Length (Z-score)")

#Plot sepal width vs. petal length
gg11c <- ggplot(iris_data_NormZ_species, aes(Petal.Length, Sepal.Width,
                shape = iris_clusters6$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Sepal W vs. Petal L") +
  labs(x = "Petal Length (Z-score)", y = "Sepal Width (Z-score)")

#Plot sepal length vs. petal width
gg12c <-ggplot(iris_data_NormZ_species, aes(Petal.Width, Sepal.Length,
                shape = iris_clusters6$cluster, color = iris_data[,5])) +
  geom_point(size = 0.5) + ggtitle("Sepal L vs. Petal W") +
  labs(x = "Petal Width (Z-score)", y = "Sepal Length (Z-score)")

ggarrange(gg7c, gg8c, gg9c, gg10c, gg11c, gg12c, common.legend = TRUE, legend = "bottom")
```



Finally, I considered k-means clustering models that used only petal length and petal width as predictors. This time, I built models using both scaled and unscaled data so that I could see if the model performance differed based on whether I used scaled or unscaled data. Using the unscaled data with petal length and petal width as predictors, I started by generating an elbow diagram. Once again, it appears that k = 3 clusters (marked with a dashed red line) is the best choice when the iris petal measurements are not scaled.

```
#Find a good number of clusters for a model using petal
#length and petal width (unscaled data)
Distances <- rep(0,25)
for (k_clusters in 1:25){
  set.seed(36)
  clusters <- kmeans(iris_data[,3:4], k_clusters, nstart = 20)
  Distances[k_clusters] <- clusters$tot.withinss
}

#Plot an elbow diagram
plot(Distances, xlab = "Number of Clusters (k)", main =
"Elbow Diagram, Petal Width & Petal Length, Unscaled Data")
abline(v = 3, lty = 2, col = "red") #Draw a vertical line at k = 3
```



```
set.seed(36)
iris_clusters4 <- kmeans(iris_data[,3:4], 3, nstart = 20)
table(iris_clusters4$cluster, iris$Species) #See how clusters compare to three known iris species
```

```
##
##     setosa versicolor virginica
##   1      0          2        46
##   2      0         48         4
##   3     50          0         0
```

```
cat("For a clustering model using only petal length and petal width as predictors:", "\n")
```

```
## For a clustering model using only petal length and petal width as predictors:
```

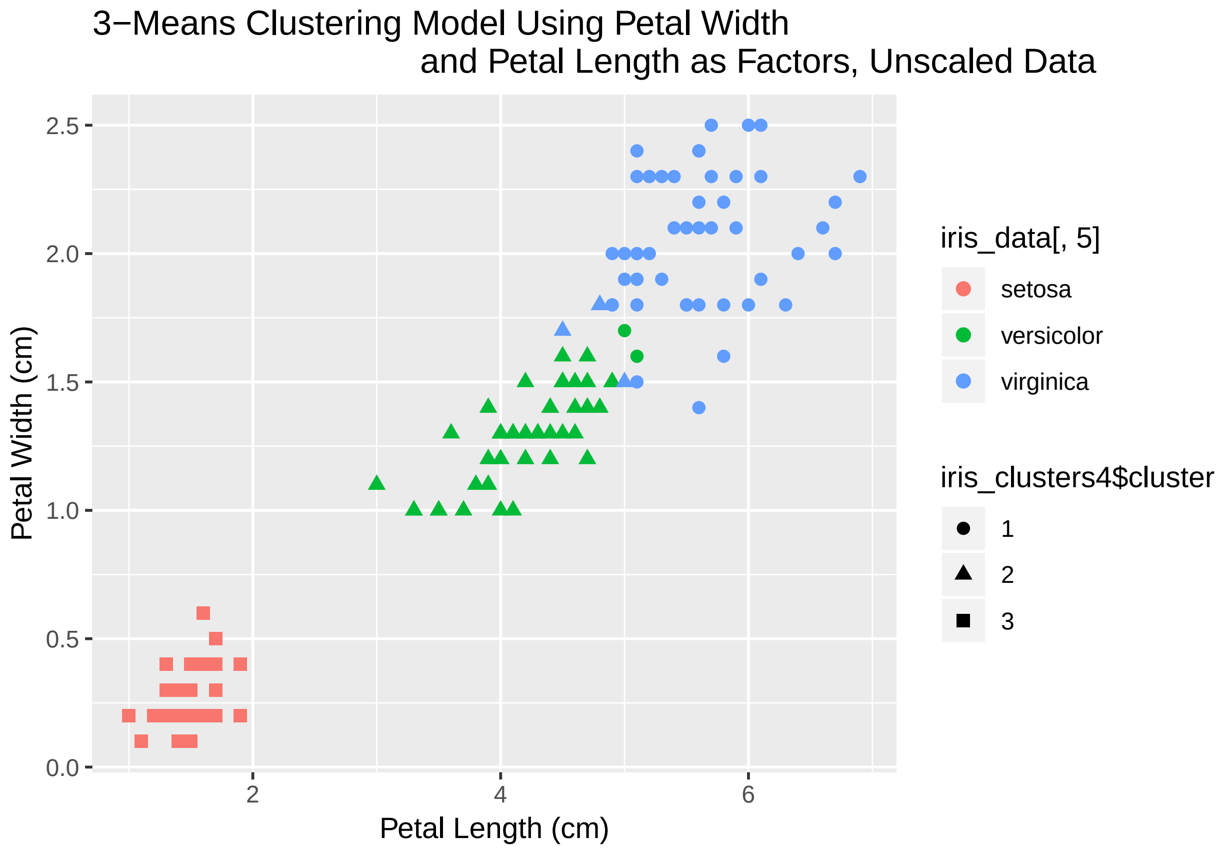
```
cat("The total within SS for k = 3 is", iris_clusters4$tot.withinss, "\n")
```

```
## The total within SS for k = 3 is 31.37136
```

```
cat("The 3-means clustering model explains", (iris_clusters4$betweenss /
                                                iris_clusters1$totss)*100,
    "% of the variability in the data.")
```

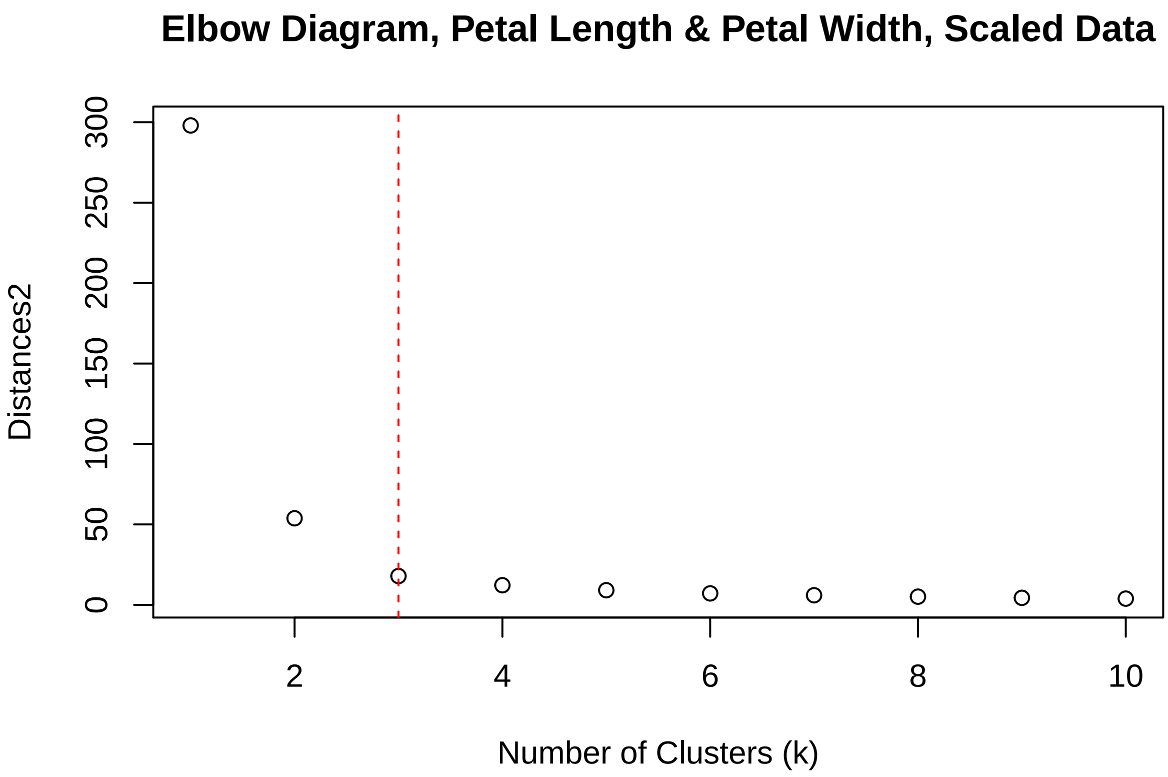
```
## The 3-means clustering model explains 76.2469 % of the variability in the data.
```

```
#Plot petal width vs. petal length (colors = clusters assigned by model)
iris_clusters4$cluster <- as.factor(iris_clusters4$cluster)
ggplot(iris_data, aes(Petal.Length, Petal.Width, color = iris_data[,5],
                      shape = iris_clusters4$cluster)) +
   geom_point(size = 2) + ggtitle("3-Means Clustering Model Using Petal Width
                                  and Petal Length as Factors, Unscaled Data") +
   labs(x = "Petal Length (cm)", y = "Petal Width (cm)")
```



The 3-means clustering model of the unscaled data using only petal width and petal length as predictors had a total within SS of 31.37, which was an improvement over what I found for my 3-means clustering model using all four variables as predictors and unscaled data (78.85). However, this model only explained 76.25% of the variability in the data, which is not as good as the model built using all four predictors (88.43%). Nonetheless, the three clusters corresponded more closely to the three known iris species this time around. All 50 *setosa* irises were assigned to cluster 3. Cluster 1 contained 46 *virginica* irises and only 2 *versicolor* irises, and cluster 2 contained 48 *versicolor* irises and only 4 *virginica* irises. The graphical display at the top of this page shows a scatterplot of petal width vs. petal length with the colors of the data points representing the given species classification of each iris and the plot symbols representing the clusters assigned by this model.

```r
#Find a good number of clusters for a model using petal length and
#petal width as predictors (scaled data)
Distances2 <- rep(0,10)
for (k_clusters in 1:10){
  set.seed(36)
  clusters <- kmeans(iris_data_NormZ_species[,3:4], k_clusters, nstart = 20)
  Distances2[k_clusters] <- clusters$tot.withinss
}

#Plot an elbow diagram
plot(Distances2, xlab = "Number of Clusters (k)",
     main = "Elbow Diagram, Petal Length & Petal Width, Scaled Data")
abline(v = 3, lty = 2, col = "red") #Draw a vertical line at k = 3
```



For k-means clustering models of the scaled data using only petal length and petal width as predictors, the elbow diagram (shown above) indicated that k = 3 clusters was the best choice once again.

```r
set.seed(36)
iris_clusters7 <- kmeans(iris_data_NormZ_species[,3:4], 3, nstart = 20)  #model with k = 3
#See how clusters compare to three known iris species
table(iris_clusters7$cluster, iris_data$Species)
```

```
##
##     setosa versicolor virginica
##   1      0          2        46
##   2     50          0         0
##   3      0         48         4
```

```r
cat("For a clustering model using only petal length and petal width as predictors & scaled data:", "\n")
```

```
## For a clustering model using only petal length and petal width as predictors & scaled data:
```

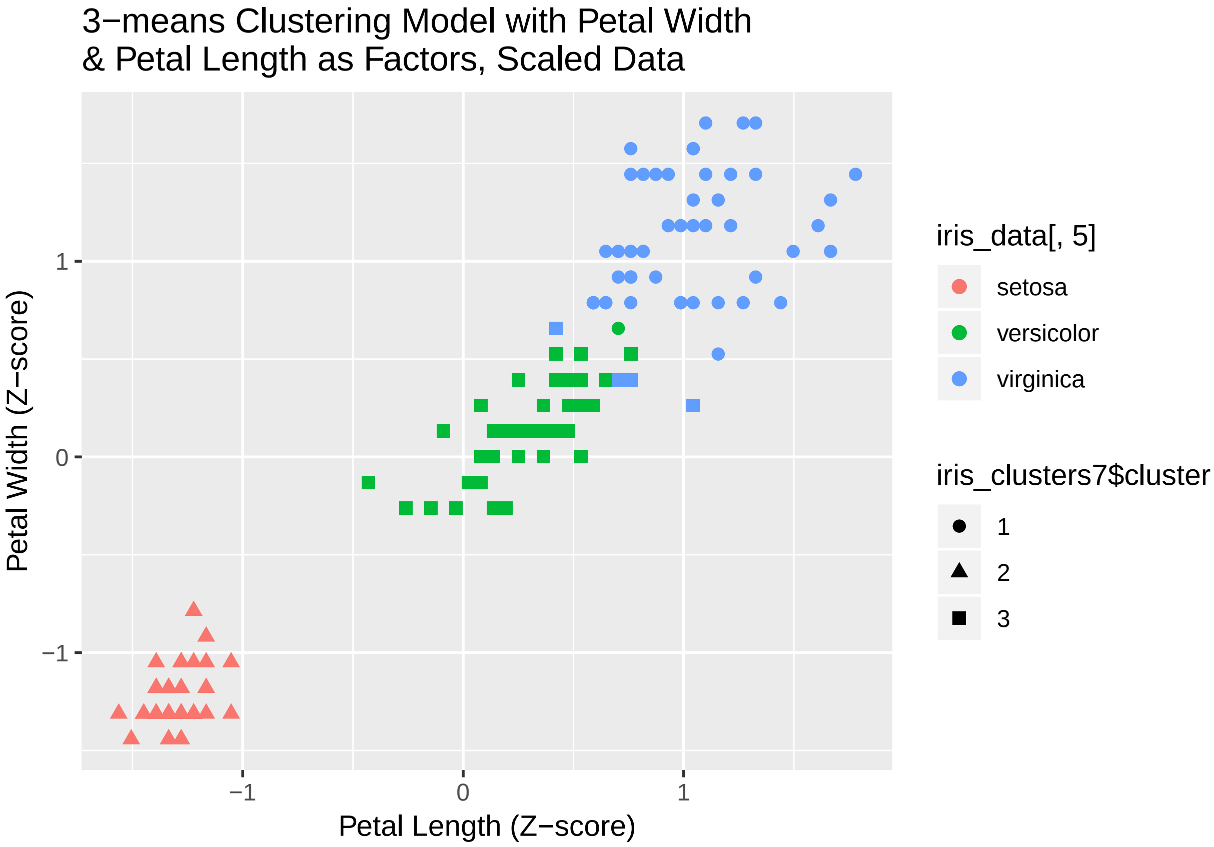
```r
cat("The total within SS for k = 3 is", iris_clusters7$tot.withinss, "\n")
```

```
## The total within SS for k = 3 is 17.90678
```

```r
cat("The 3-means clustering model explains ",
    (iris_clusters7$betweenss / iris_clusters7$totss) *100,
    "% of the variability in the data.")
```

```
## The 3-means clustering model explains  93.99101 % of the variability in the data.
```

```r
#Plot the clusters generated using scaled data, k = 3, petal width vs. petal length
#color = species, plot symbol = cluster
iris_clusters7$cluster <- as.factor(iris_clusters7$cluster)
ggplot(iris_data_NormZ_species, aes(Petal.Length, Petal.Width, color = iris_data[,5],
                      shape = iris_clusters7$cluster)) +
  geom_point(size = 2) + ggtitle("3-means Clustering Model with Petal Width
& Petal Length as Factors, Scaled Data") +
  labs(x = "Petal Length (Z-score)", y = "Petal Width (Z-score)")
```



The 3-means model of the scaled iris data using only petal length and petal width as predictors had a total within SS of 17.91, which is the lowest value that I observed for the various models that I generated. Also, this model explained 93.99% of the variability in the data, which was also the best performance by this metric. Finally, the correspondence between my clusters and the actual species classifications of the irises in the dataset was quite good. Cluster 2 contained all 50 *setosa* irises, cluster 1 contained 46 *virginica* irises and only 2 *versicolor* irises, and cluster 3 contained 48 *versicolor* irises and only 4 *virginica* irises. The fact that this model yielded three clusters that are quite similar in composition to the actual species classifications of the irises in the data set further provides an external confirmation of its strength. Specifically, 144/150 = 96% of the cluster assignments made by this model matched the known species classifications of the data points (which was the same for the model using unscaled data). By every measure, the best model that I generated for clustering the iris data was a 3-means clustering model of the scaled data using only petal length and petal width as factors (i.e., predictors) in the model. Also, I discovered that it does indeed make a difference when I use scaled data vs. unscaled data in building k-means models.

The table at the top of the next page summarizes the results of my experiments building k-means clustering models of the iris data.

Data Type	Factors Included	Number of Clusters (k)	Total Within SS	Total Explained Variation
Unscaled	All four	3	78.85	88.43%
Scaled	All four	3	138.89	76.70%
Scaled	All four	5	90.20	84.87%
Scaled	Petal Length, Petal Width, & Sepal Length	3	62.62	85.99%
Unscaled	Petal Length & Petal Width	3	31.37	76.25%
Scaled	Petal Length & Petal Width	3	17.91	93.99%

Built with R version 3.5.1