IRIS Clustering

Ryan Porter

8/31/2020

Setup

```
library(kernlab)
library(kknn)
library(tidyverse)
library(caret)
```

Use the R function kmeans to cluster the points as well as possible. Report the best combination of predictors, your suggested value of k, and how well your best clustering predicts flower type.

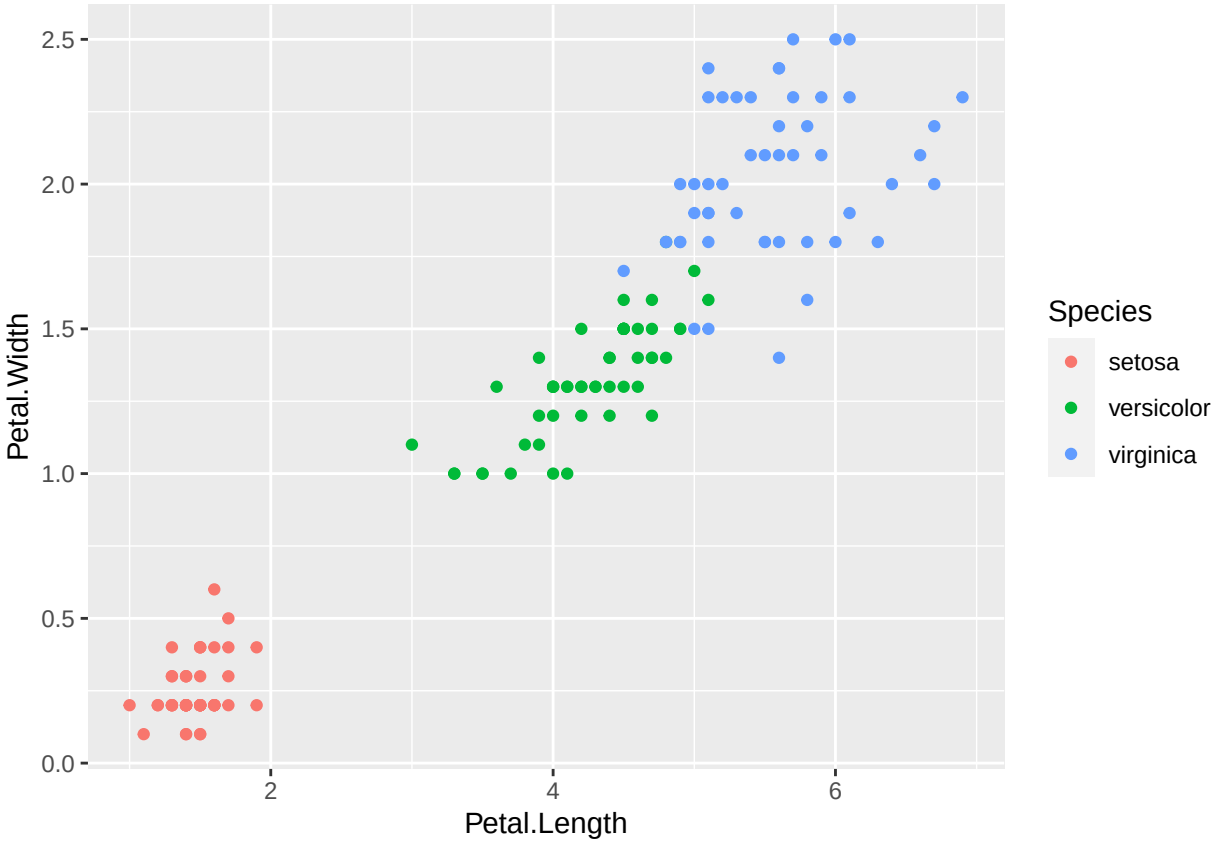
```
data(iris)  #load the iris data

summary(iris) #a quick look at the data
```

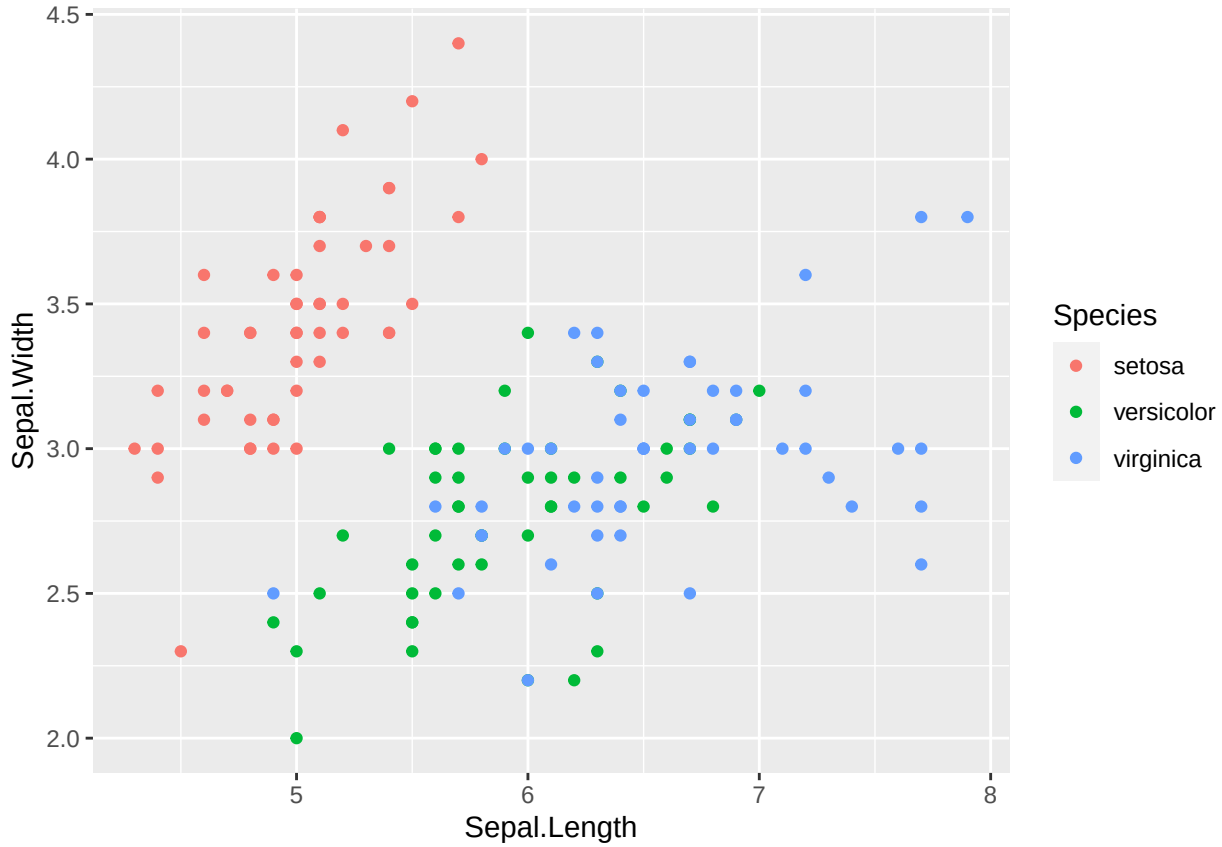
```
##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300
##  Mean   :5.843   Mean   :3.057   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
##
##
##
```

```
set.seed(0222)

#must load tidyverse in order to use %>%
iris %>% ggplot(aes(Petal.Length, Petal.Width, color = Species)) +
  geom_point()
```




```
iris %>% ggplot(aes(Sepal.Length, Sepal.Width, color = Species)) +
  geom_point()
```




```
iris %>%
  group_by(Species) %>%
  count()
```

```
## # A tibble: 3 x 2
## # Groups:   Species [3]
##   Species        n
##   <fct>      <int>
## 1 setosa        50
## 2 versicolor    50
## 3 virginica     50
```

The first graph is of petal length by the petal width and I have it color cordinated by the species of flower. The second graph is very similiar to the first but instead of petal we are using the sepal lenth and width. Last, is a simple table of the number of each species of flower which is 50 for each of them.

```
set.seed(0222)
#cluster <- kmeans(iris[, 2:4], 3, nstart = 20)
#predicts virginica less accurately by one flower each time

#testing out different numbers of clusters. My guess is going to be that 3 is optimal
#because we are talking about three different species of flower.
cluster_2 <- kmeans(iris[, 3:4], 2, nstart = 10)
cluster_3 <- kmeans(iris[, 3:4], 3, nstart = 10)
cluster_4 <- kmeans(iris[, 3:4], 4, nstart = 10)
```

```r
cluster_5 <- kmeans(iris[, 3:4], 5, nstart = 10)
table(cluster_2$cluster, iris$Species)
```

```
##
##     setosa versicolor virginica
##   1      0         49        50
##   2     50          1         0
```

```r
table(cluster_3$cluster, iris$Species)
```

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

```r
table(cluster_4$cluster, iris$Species)
```

```
##
##     setosa versicolor virginica
##   1     50          0         0
##   2      0         24        15
##   3      0          0        35
##   4      0         26         0
```

```r
table(cluster_5$cluster, iris$Species)
```

```
##
##     setosa versicolor virginica
##   1      0          0        13
##   2      0         22         0
##   3      0          0        30
##   4     50          0         0
##   5      0         28         7
```

You can see from each of the tables that k values bigger than 3 seem to start miss cateogizing them a lot more. However, less than 3 you are trying to fit two different groups of species into one group which doesnt cluster than correctly. I decided to go with k=3. I tried different values of nstart (5,20) but didnt seem to get much of a variation when changing it.

```r
set.seed(0222)
cluster1 <- kmeans(iris[, 3:4], 3, nstart = 10)
# best prediction from the models
cluster1
```

```
## K-means clustering with 3 clusters of sizes 48, 50, 52
##
## Cluster means:
##   Petal.Length Petal.Width
## 1     5.595833    2.037500
```

```
## 2     1.462000    0.246000
## 3     4.269231    1.342308
##
## Clustering vector:
##   [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##  [38] 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
##  [75] 3 3 3 1 3 3 3 3 3 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 1 1 1 1 1 3 1 1 1 1
## [112] 1 1 1 1 1 1 1 1 3 1 1 1 1 1 1 3 1 1 1 1 1 1 1 1 1 1 3 1 1 1 1 1 1 1 1 1
## [149] 1 1
##
## Within cluster sum of squares by cluster:
## [1] 16.29167  2.02200 13.05769
##  (between_SS / total_SS =  94.3 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
cluster2 <- kmeans(iris[, 1:2], 3, nstart = 10)

table(cluster1$cluster, iris$Species)
```

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

```
table(cluster2$cluster, iris$Species)
```

```
##
##     setosa versicolor virginica
##   1      0         38        15
##   2     50          0         0
##   3      0         12        35
```

The best combination of predictors was petal length and petal width (cluster1). This makes sense because each petal is very specific to the flower where sepal length is not as specific to each of them. My model predicted all 50 setosa correctly, 38 of 50 correct for versicolor, and 35 of 50 correct for virginica. From the table you can see that versicolor and virginica seem to miss clustered for each other but never setosa.