script

Shrishti Vaish

03/12/2020

Importing packages

```
library(ggplot2)
library(factoextra)
```

```
## Warning: package 'factoextra' was built under R version 3.6.3
```

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

```
library(ggpubr)
```

```
## Warning: package 'ggpubr' was built under R version 3.6.3
```

```
## Loading required package: magrittr
```

Reading dataset

```
df <- read.csv("Iris.csv")
df$Id <- NULL

head(df)
```

```
##   SepalLengthCm SepalWidthCm PetalLengthCm PetalWidthCm     Species
## 1           5.1          3.5           1.4          0.2 Iris-setosa
## 2           4.9          3.0           1.4          0.2 Iris-setosa
## 3           4.7          3.2           1.3          0.2 Iris-setosa
## 4           4.6          3.1           1.5          0.2 Iris-setosa
## 5           5.0          3.6           1.4          0.2 Iris-setosa
## 6           5.4          3.9           1.7          0.4 Iris-setosa
```

K means Classification

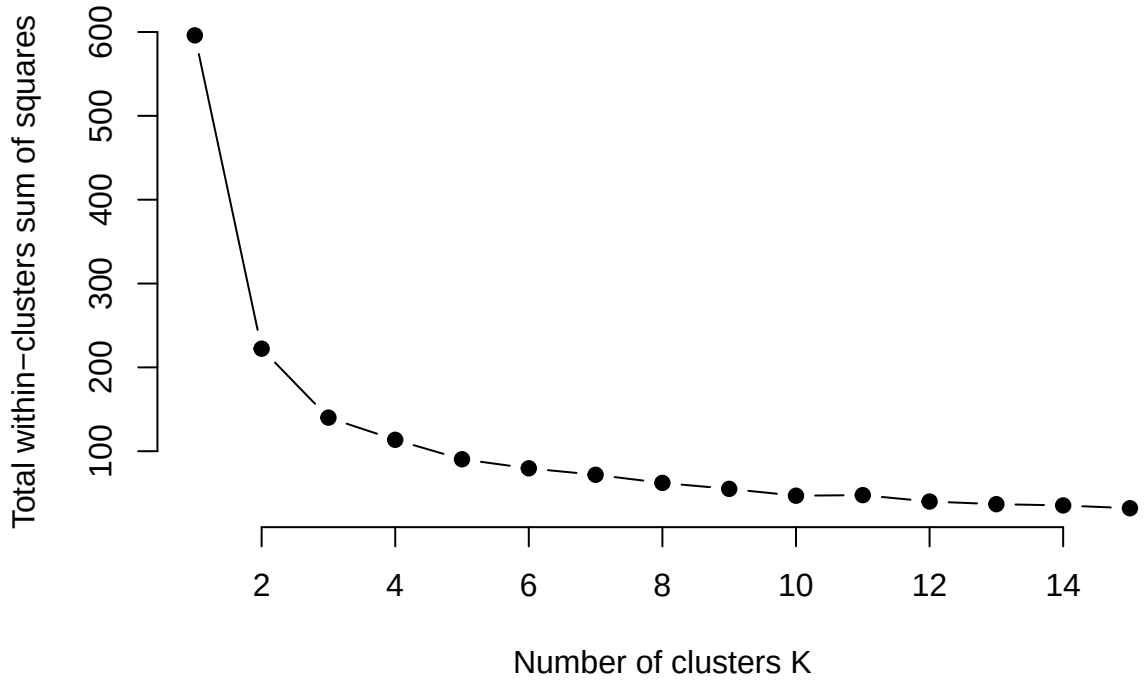
Checking the optimum no. of clusters

```r
x <- scale(df[, c(1:4)])

set.seed(173)
wss <- function(k){
  kmeans(x, k, nstart = 10)$tot.withinss
}

k.values <- 1:15

library(purrr)
```

```
##
## Attaching package: 'purrr'
```

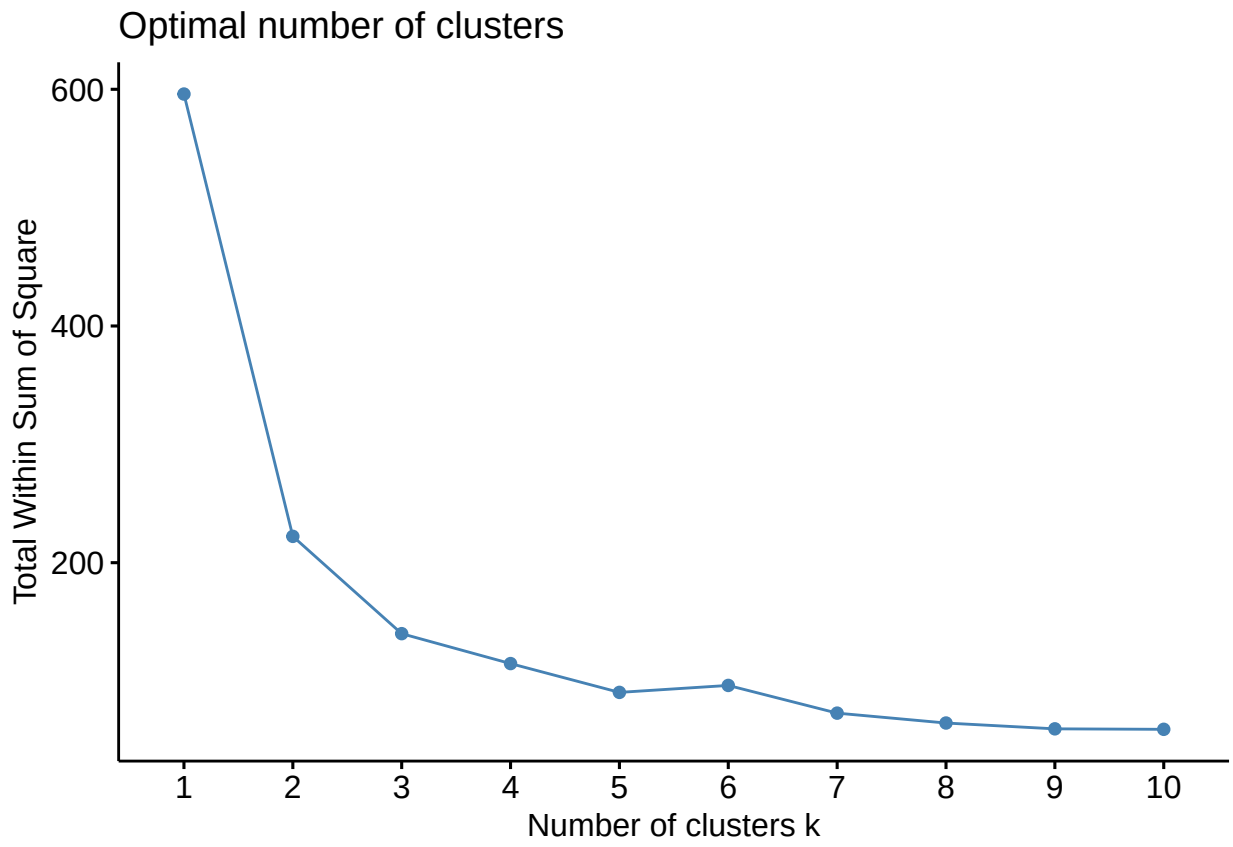
```
## The following object is masked from 'package:magrittr':
##
##     set_names
```

```r
wss_values <- map_dbl(k.values, wss)

plot(k.values, wss_values,
       type="b", pch = 19, frame = FALSE,
       xlab="Number of clusters K",
       ylab="Total within-clusters sum of squares")
```



Fortunately, this process to compute the "Elbow method" has been wrapped up in a single function (fviz_nbclust):

```
set.seed(173)

#install.packages("factoextra")

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



The results suggest that 3 is the optimal number of clusters as it appears to be the bend in the knee (or elbow)

K means

Now, applying kmeans to the dataset

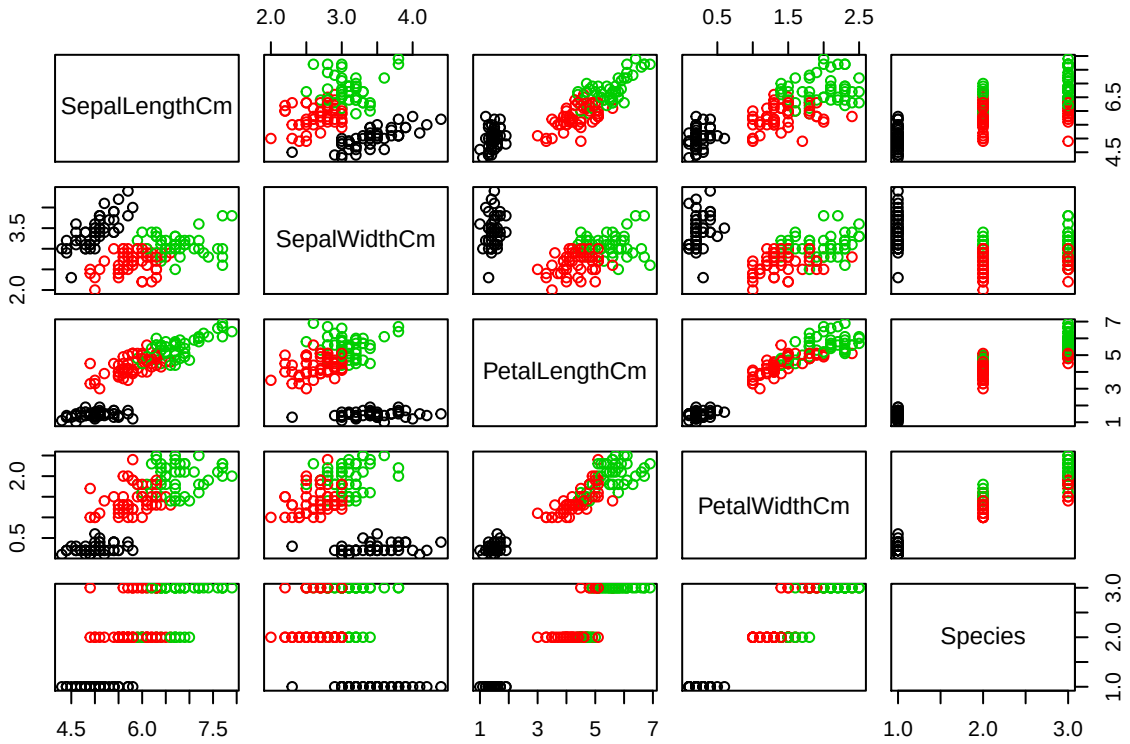
```
fitK <- kmeans(x,3)
fitK
```

```
## K-means clustering with 3 clusters of sizes 50, 53, 47
##
## Cluster means:
##   SepalLengthCm SepalWidthCm PetalLengthCm PetalWidthCm
## 1   -1.01119138    0.8394944    -1.3005215   -1.2509379
```

```
## 2   -0.05005221   -0.8773526     0.3463713    0.2811215
## 3    1.13217737    0.0962759     0.9929445    1.0137756
##
## Clustering vector:
##   [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##  [38] 1 1 1 1 1 1 1 1 1 1 1 1 1 3 3 3 2 2 2 3 2 2 2 2 2 2 2 3 2 2 2 2 3 2 2 2
##  [75] 2 3 3 3 2 2 2 2 2 2 2 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 3 3 3 3 2 3 3 3 3
## [112] 3 3 2 2 3 3 3 3 2 3 2 3 2 3 3 2 3 3 3 3 3 2 2 3 3 3 2 3 3 3 2 3 3 3 2 3
## [149] 3 2
##
## Within cluster sum of squares by cluster:
## [1] 48.15831 44.25778 47.60995
##  (between_SS / total_SS =  76.5 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

Visualizations

```
plot(df, col = fitK$cluster)
```



Actual vs Predicted Classifications

```
table(Predicated = fitK$cluster, Actual = df$Species)
```

```
##            Actual
## Predicated Iris-setosa Iris-versicolor Iris-virginica
##          1          50               0              0
##          2           0              39             14
##          3           0              11             36
```

Clustering Visualization

```
fviz_cluster(fitK, x, palette = c("red", "blue", "green"), geom = "point", ellipse.type = "convex", ggtl
```