K-means Clustering and Agglomerative Clustering

2024-03-19

#K-means clustering  
library(tidyverse) #data manipulation

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.2 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.4.3 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.3 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.1   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(cluster) #clustering algorithm  
library(ggplot2)  
library(factoextra) #clustering algorithm & visual

## Warning: package 'factoextra' was built under R version 4.3.3

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

df = USArrests  
head(df)

## Murder Assault UrbanPop Rape  
## Alabama 13.2 236 58 21.2  
## Alaska 10.0 263 48 44.5  
## Arizona 8.1 294 80 31.0  
## Arkansas 8.8 190 50 19.5  
## California 9.0 276 91 40.6  
## Colorado 7.9 204 78 38.7

df = na.omit(df)  
df = scale(df)  
head(df)

## Murder Assault UrbanPop Rape  
## Alabama 1.24256408 0.7828393 -0.5209066 -0.003416473  
## Alaska 0.50786248 1.1068225 -1.2117642 2.484202941  
## Arizona 0.07163341 1.4788032 0.9989801 1.042878388  
## Arkansas 0.23234938 0.2308680 -1.0735927 -0.184916602  
## California 0.27826823 1.2628144 1.7589234 2.067820292  
## Colorado 0.02571456 0.3988593 0.8608085 1.864967207

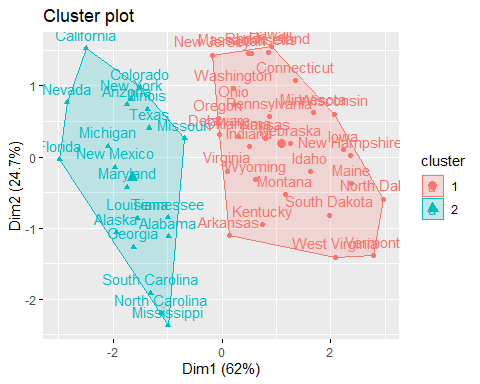
k2 = kmeans(df, centers = 2, nstart = 25)  
str(k2)

## List of 9  
## $ cluster : Named int [1:50] 2 2 2 1 2 2 1 1 2 2 ...  
## ..- attr(\*, "names")= chr [1:50] "Alabama" "Alaska" "Arizona" "Arkansas" ...  
## $ centers : num [1:2, 1:4] -0.67 1.005 -0.676 1.014 -0.132 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : chr [1:2] "1" "2"  
## .. ..$ : chr [1:4] "Murder" "Assault" "UrbanPop" "Rape"  
## $ totss : num 196  
## $ withinss : num [1:2] 56.1 46.7  
## $ tot.withinss: num 103  
## $ betweenss : num 93.1  
## $ size : int [1:2] 30 20  
## $ iter : int 1  
## $ ifault : int 0  
## - attr(\*, "class")= chr "kmeans"

k2

## K-means clustering with 2 clusters of sizes 30, 20  
##   
## Cluster means:  
## Murder Assault UrbanPop Rape  
## 1 -0.669956 -0.6758849 -0.1317235 -0.5646433  
## 2 1.004934 1.0138274 0.1975853 0.8469650  
##   
## Clustering vector:  
## Alabama Alaska Arizona Arkansas California   
## 2 2 2 1 2   
## Colorado Connecticut Delaware Florida Georgia   
## 2 1 1 2 2   
## Hawaii Idaho Illinois Indiana Iowa   
## 1 1 2 1 1   
## Kansas Kentucky Louisiana Maine Maryland   
## 1 1 2 1 2   
## Massachusetts Michigan Minnesota Mississippi Missouri   
## 1 2 1 2 2   
## Montana Nebraska Nevada New Hampshire New Jersey   
## 1 1 2 1 1   
## New Mexico New York North Carolina North Dakota Ohio   
## 2 2 2 1 1   
## Oklahoma Oregon Pennsylvania Rhode Island South Carolina   
## 1 1 1 1 2   
## South Dakota Tennessee Texas Utah Vermont   
## 1 2 2 1 1   
## Virginia Washington West Virginia Wisconsin Wyoming   
## 1 1 1 1 1   
##   
## Within cluster sum of squares by cluster:  
## [1] 56.11445 46.74796  
## (between\_SS / total\_SS = 47.5 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss" "tot.withinss"  
## [6] "betweenss" "size" "iter" "ifault"

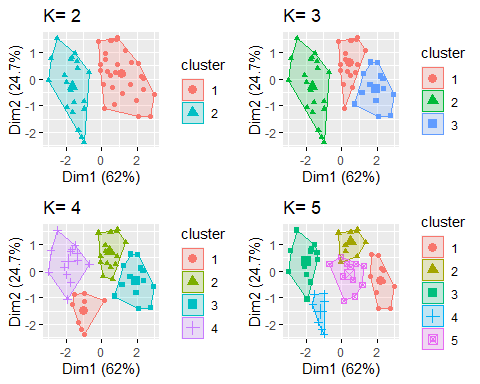
fviz\_cluster(k2, data = df)



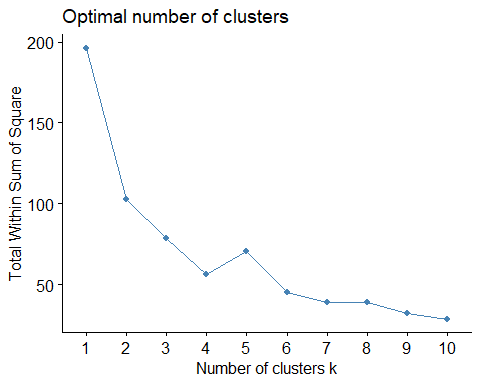
k3 = kmeans(df,centers = 3, nstart = 25)  
k4 = kmeans(df,centers = 4, nstart = 25)  
k5 = kmeans(df,centers = 5, nstart = 25)  
  
#plotting to compare  
p1 = fviz\_cluster(k2, geom = "point", data = df)+ggtitle("K= 2")  
p2 = fviz\_cluster(k3, geom = "point", data = df)+ggtitle("K= 3")  
p3 = fviz\_cluster(k4, geom = "point", data = df)+ggtitle("K= 4")  
p4 = fviz\_cluster(k5, geom = "point", data = df)+ggtitle("K= 5")  
  
library(gridExtra)

##   
## Attaching package: 'gridExtra'  
##   
## The following object is masked from 'package:dplyr':  
##   
## combine

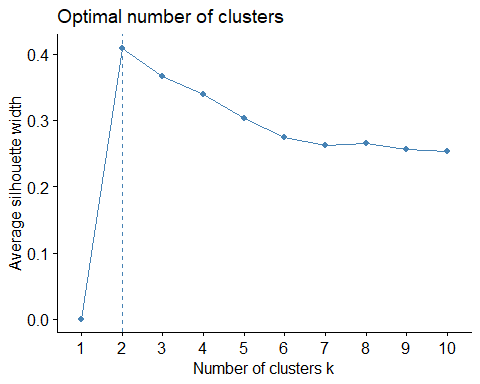
grid.arrange(p1,p2,p3,p4, nrow = 2)



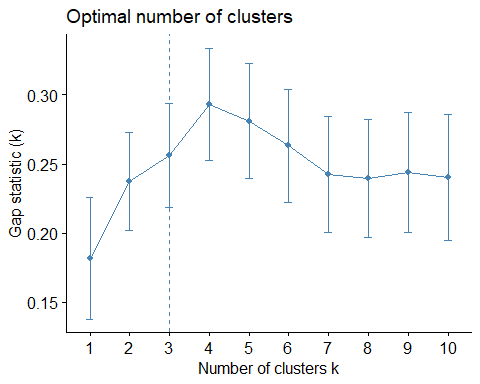
#selecting optimal number of clusters  
set.seed(123)  
fviz\_nbclust(df,kmeans, method = "wss")



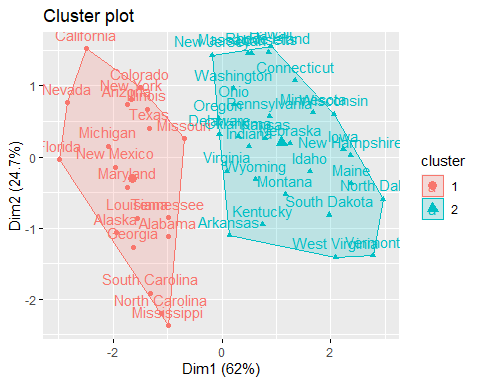
set.seed(123)  
fviz\_nbclust(df,kmeans, method = "silhouette")



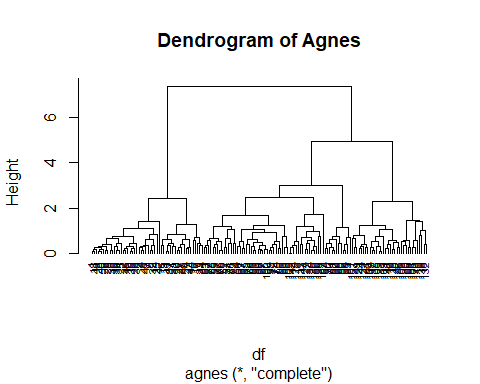
set.seed(123)  
gap\_stat = clusGap(df, FUNcluster = kmeans, nstart = 25, K.max = 10, B = 50)  
fviz\_gap\_stat(gap\_stat)



set.seed(123)  
final = kmeans(df,2,nstart = 25)  
fviz\_cluster(final, data = df)



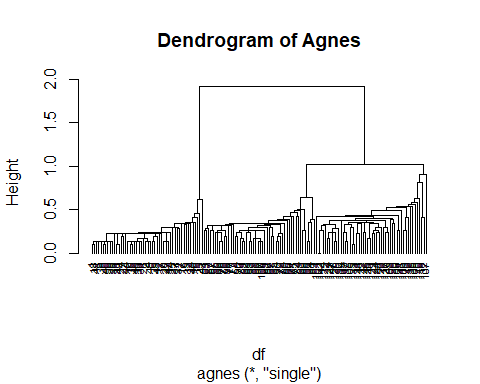
#Hierarchical Clustering-Agnes  
library(tidyverse) #data manipulation  
library(cluster) #clustering algorithm  
library(factoextra) #clustering algorithm & visual  
  
df = iris # or use USArrests  
hcal = agnes(df, method = "complete")  
pltree(hcal, cex = 0.6, hang = -1, main = "Dendrogram of Agnes")



#cut into 3 groups  
dvcut = cutree(as.hclust(hcal), k = 3)  
table(dvcut)

## dvcut  
## 1 2 3   
## 50 66 34

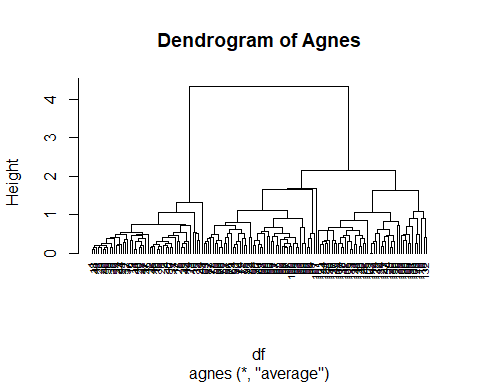
hcal2 = agnes(df, method = "single")  
pltree(hcal2, cex = 0.6, hang = -1, main = "Dendrogram of Agnes")



#cut into 3 groups  
dvcut = cutree(as.hclust(hcal2), k = 3)  
table(dvcut)

## dvcut  
## 1 2 3   
## 50 50 50

hcal3 = agnes(df, method = "average")  
pltree(hcal3, cex = 0.6, hang = -1, main = "Dendrogram of Agnes")



#cut into 3 groups  
dvcut = cutree(as.hclust(hcal3), k = 3)  
table(dvcut)

## dvcut  
## 1 2 3   
## 50 51 49