# Load the data

install.packages(c("factoextra","cluster","NbClust"))

library(factoextra)

library(cluster)

library(NbClust)

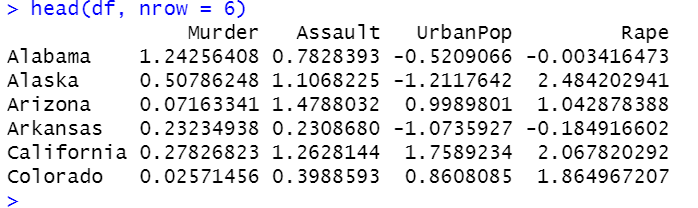
data("USArrests")

# Standardize the data

df <- scale(USArrests)

# Show the first 6 rows

head(df, nrow = 6)

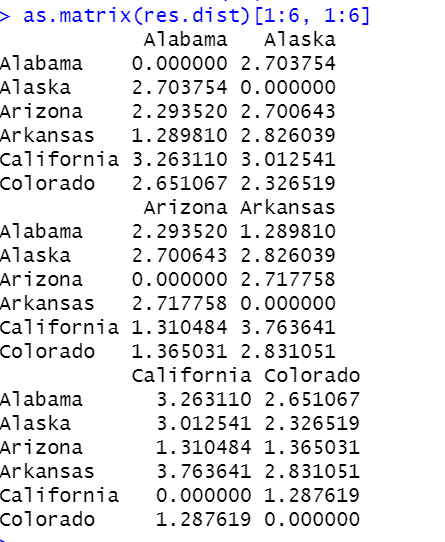


# Compute the dissimilarity matrix

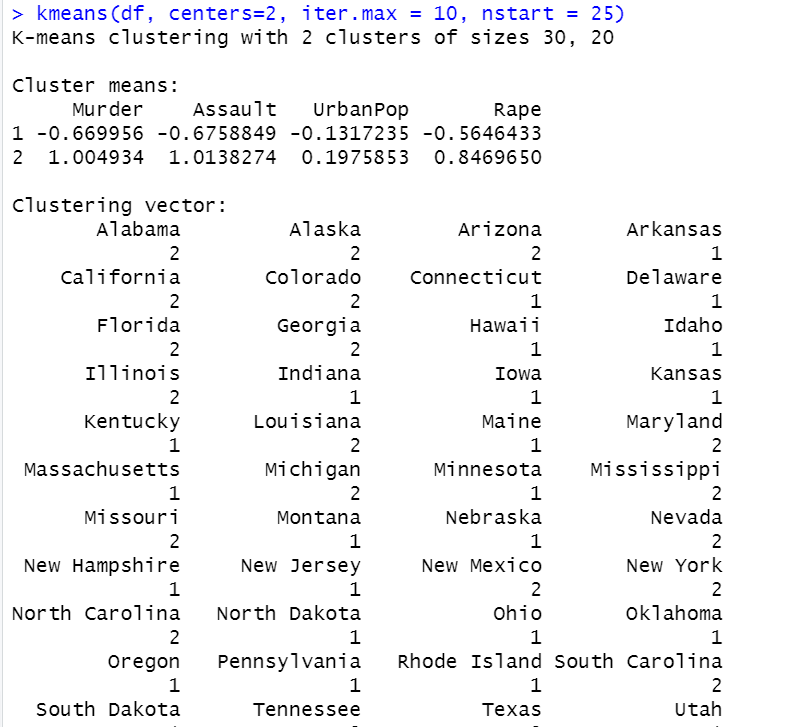
# df = the standardized data

res.dist <- dist(df, method = "euclidean")

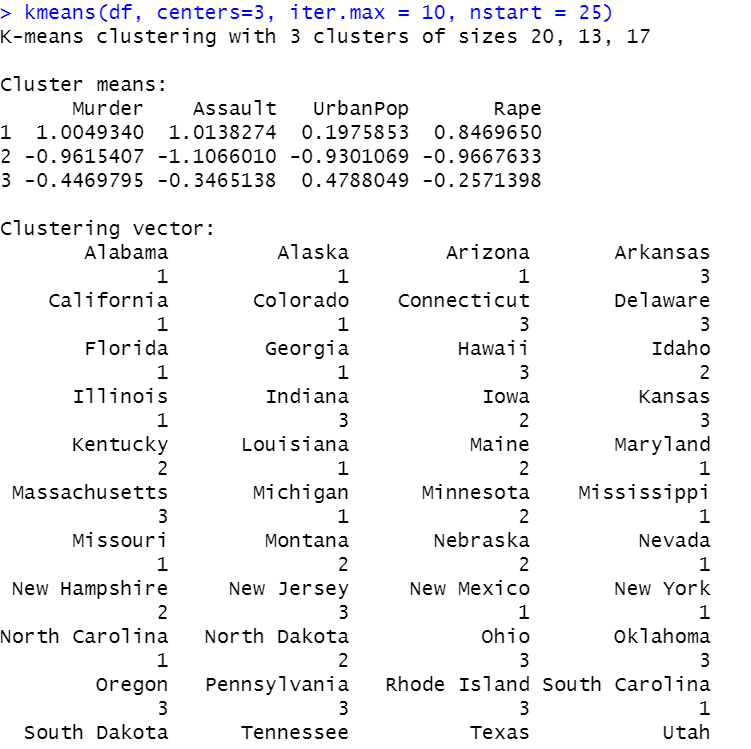
as.matrix(res.dist)[1:6, 1:6]



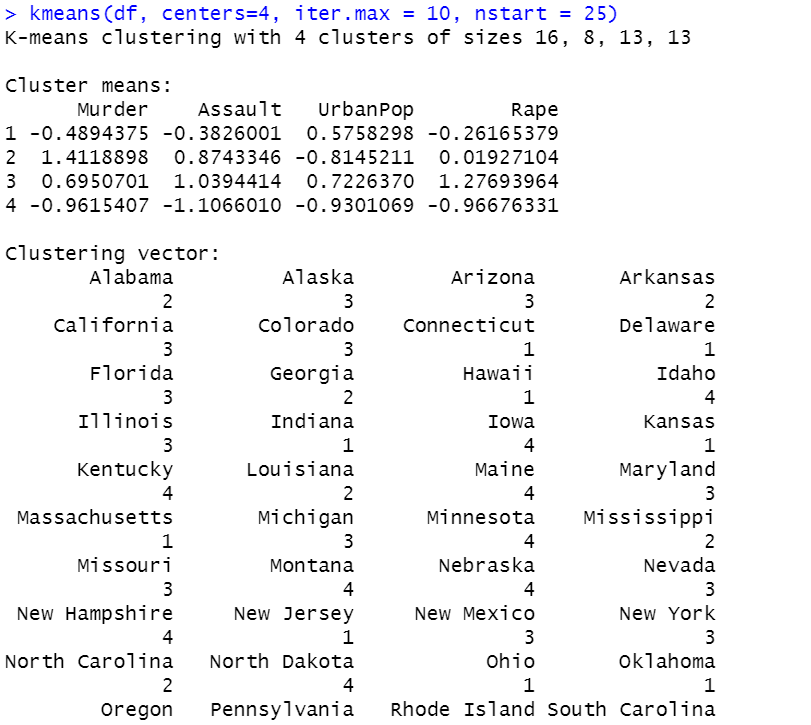
kmeans(df, centers=2, iter.max = 10, nstart = 25)



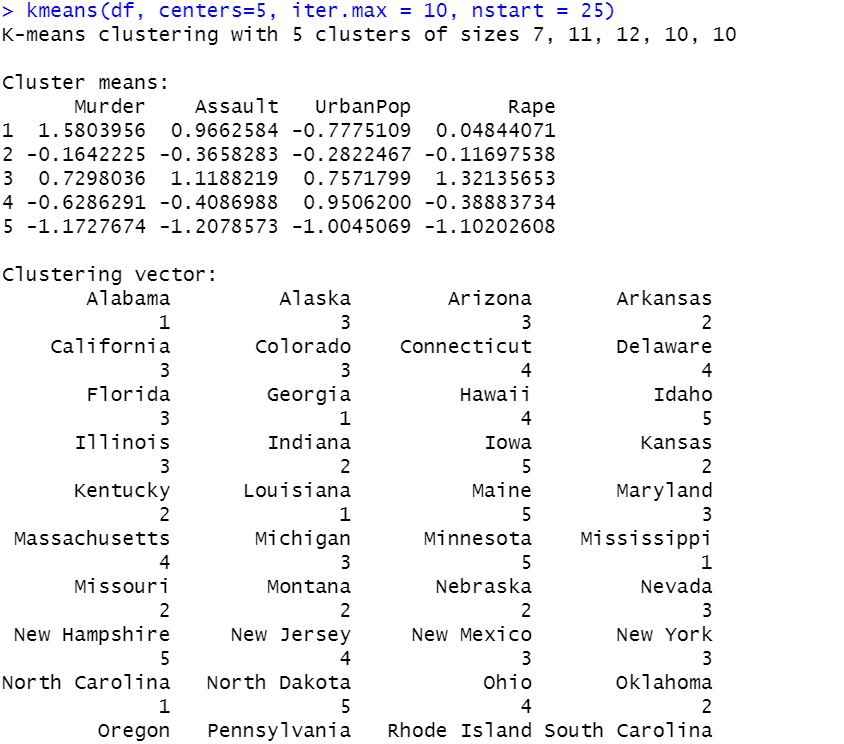
kmeans(df, centers=3, iter.max = 10, nstart = 25)



kmeans(df, centers=4, iter.max = 10, nstart = 25)



kmeans(df, centers=5, iter.max = 10, nstart = 25)



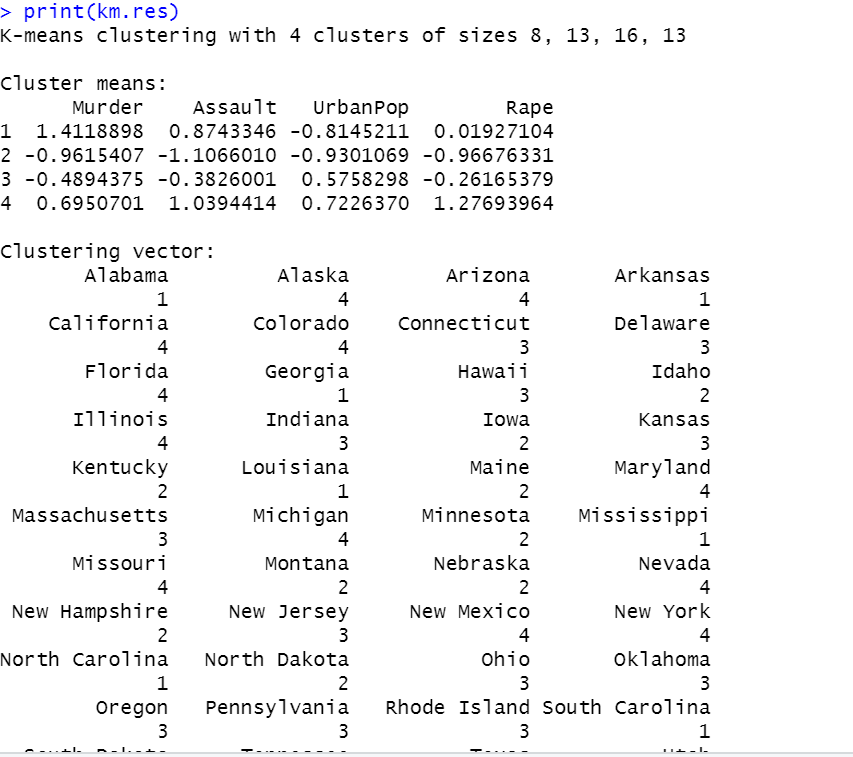
fviz\_nbclust(df, kmeans, method = "wss")+ geom\_vline(xintercept = 4, linetype = 2)

set.seed(123)

km.res <- kmeans(df, 4, nstart = 25)

# Print the results

print(km.res)

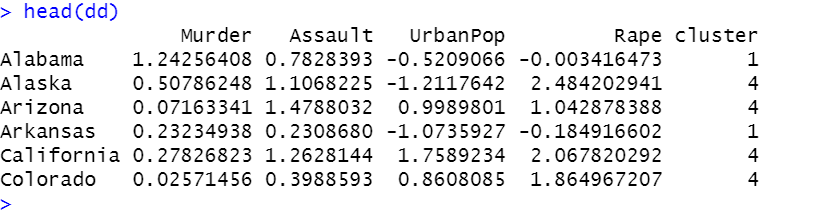


#To find mean of the each variable

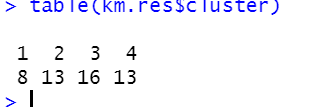
aggregate(df, by=list(cluster=km.res$cluster), mean)

dd <- cbind(df, cluster = km.res$cluster)

head(dd)



table(km.res$cluster)



km.res$size



#To represent in diagram

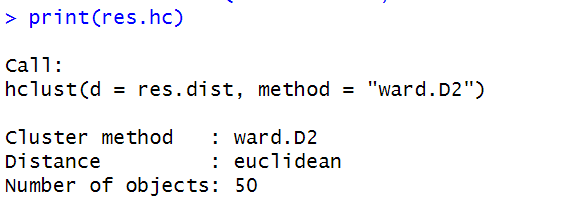
fviz\_cluster(km.res, data = df, palette = c("#2E9FDF", "#00AFBB", "#E7B800", "#FC4E07"), ellipse.type = "euclid", star.plot = TRUE, repel = TRUE, ggtheme = theme\_minimal() )

fviz\_cluster(km.res, data = df, palette = c("set2"), ellipse.type = "euclid", star.plot = TRUE, repel = TRUE, ggtheme = theme\_minimal() )

#Hierarchical clustering

res.hc <- hclust(d = res.dist, method = "ward.D2")

print(res.hc)



#cex: label size

install.packages("factoextra")

library("factoextra")

fviz\_dend(res.hc, cex = 0.5)

# Compute cophentic distance

res.coph <- cophenetic(res.hc)

# Correlation between cophenetic distance and

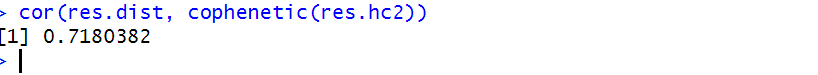
# the original distance

cor(res.dist, res.coph)



res.hc2 <- hclust(res.dist, method = "average")

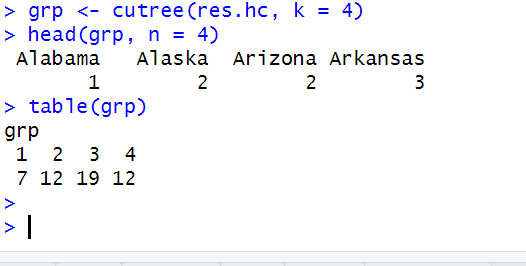
cor(res.dist, cophenetic(res.hc2))



grp <- cutree(res.hc, k = 4)

head(grp, n = 4)

table(grp)



SALES PRODUCT DATA SET

install.packages(c("factoextra","cluster","NbClust"))

install.packages("lifecycle", version = "1.0.3")

library(broom)

library(lifecycle)

library(factoextra)

library(cluster)

library(NbClust)

library(ggplot2)

df = read.csv("Sales\_Product\_Details.csv")

df$Product\_Description <- as.numeric(as.factor(df$Product\_Description))

df$Product\_Category <- as.numeric(as.factor(df$Product\_Category))

df$Product\_Line<- as.numeric(as.factor(df$Product\_Line))

df$Raw\_Material<- as.numeric(as.factor(df$Raw\_Material))

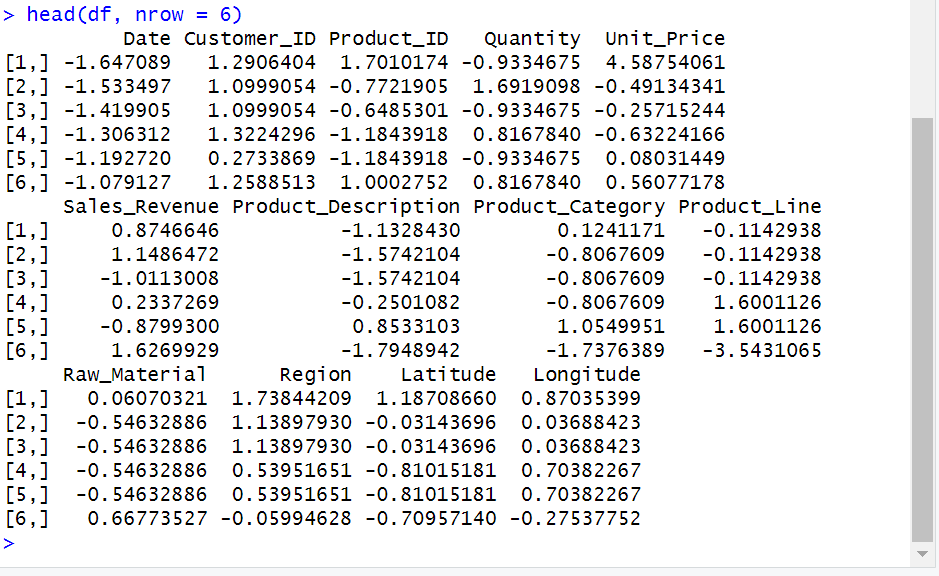
df$Region <- as.numeric(as.factor(df$Region))

# Standardize the data

df <- scale(df)

# Show the first 6 rows

head(df, nrow = 6)

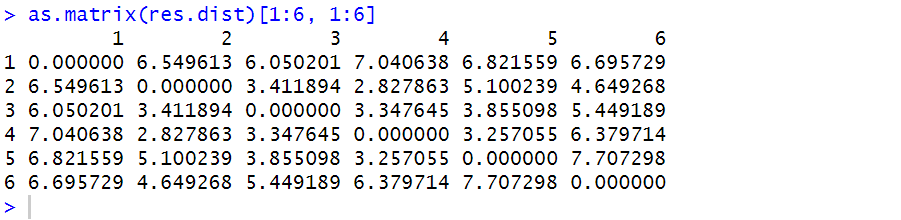


# Compute the dissimilarity matrix

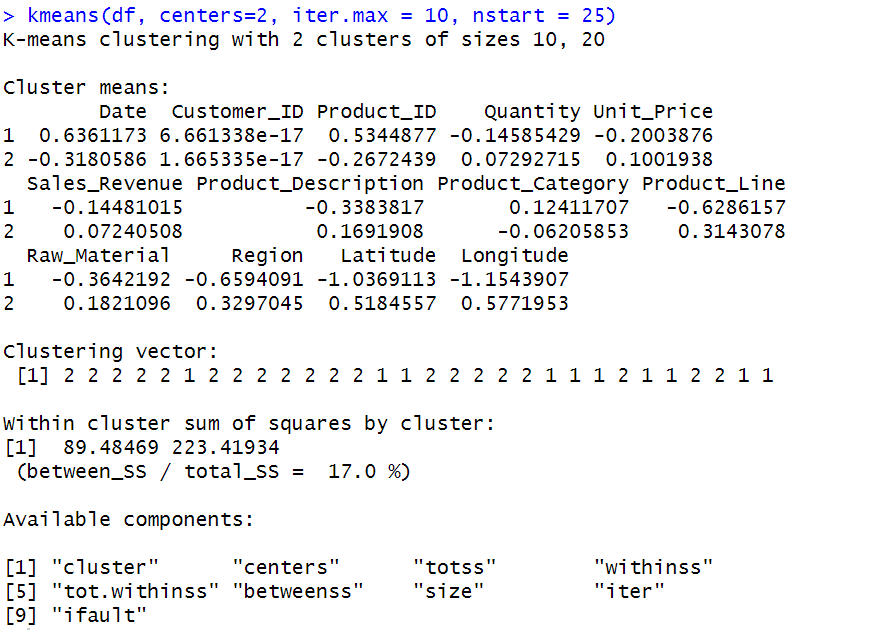
# df = the standardized data

res.dist <- dist(df, method = "euclidean")

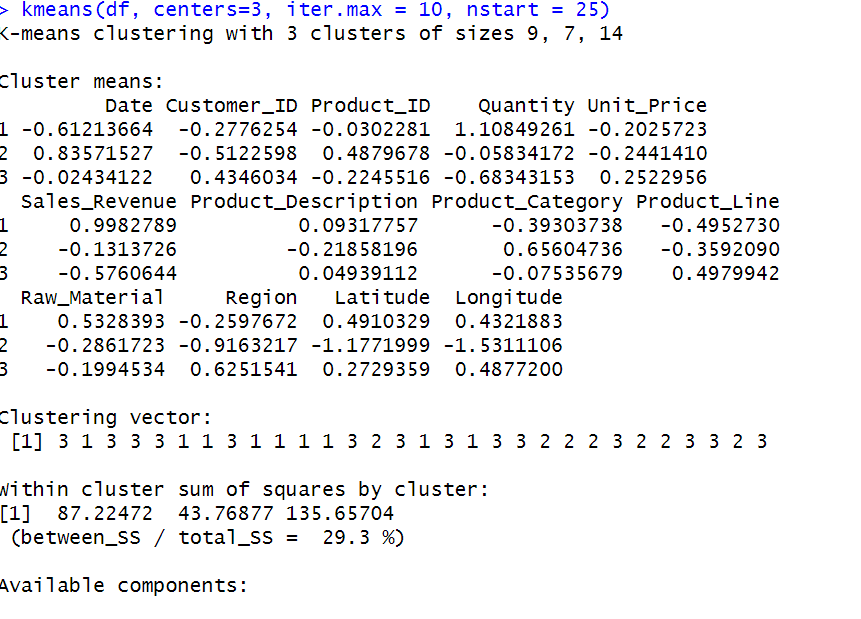
as.matrix(res.dist)[1:6, 1:6]



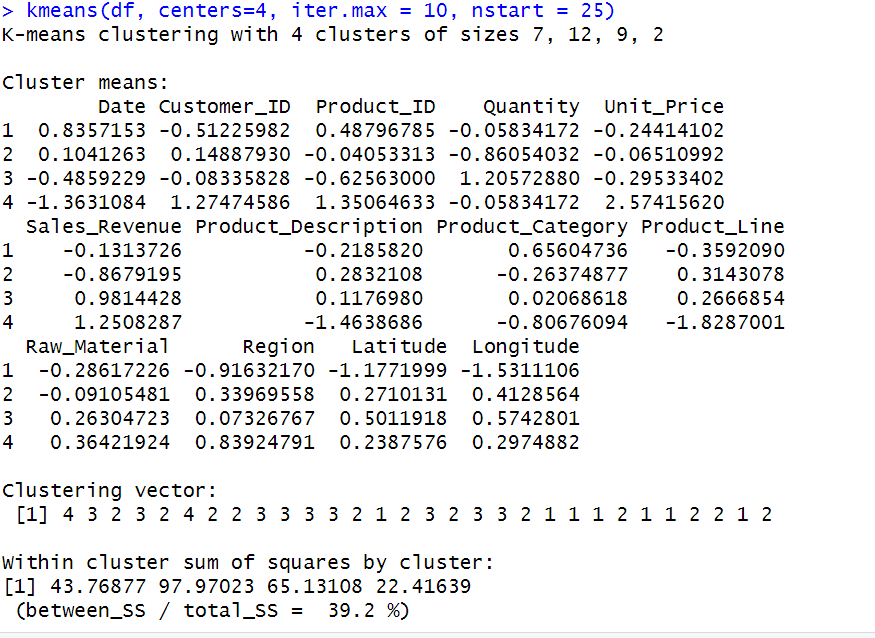
kmeans(df, centers=2, iter.max = 10, nstart = 25)



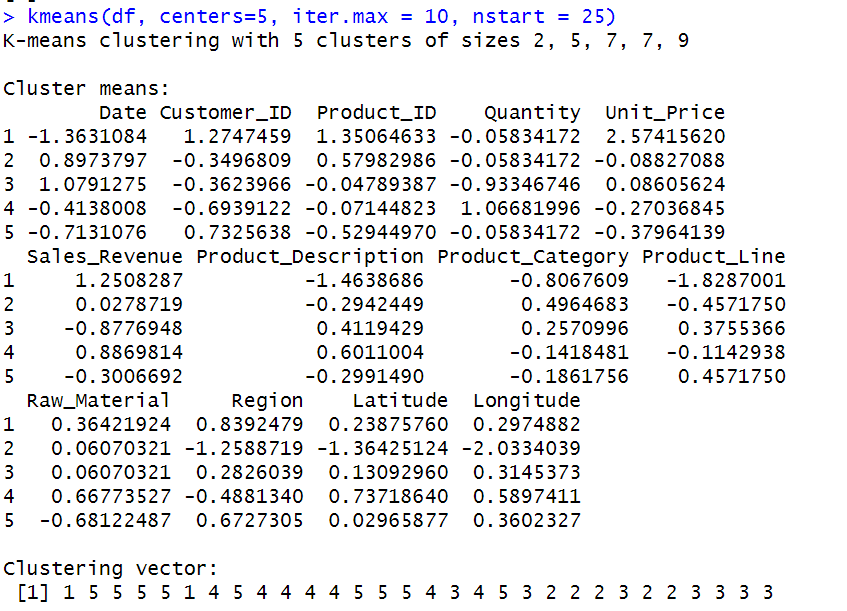
kmeans(df, centers=3, iter.max = 10, nstart = 25)



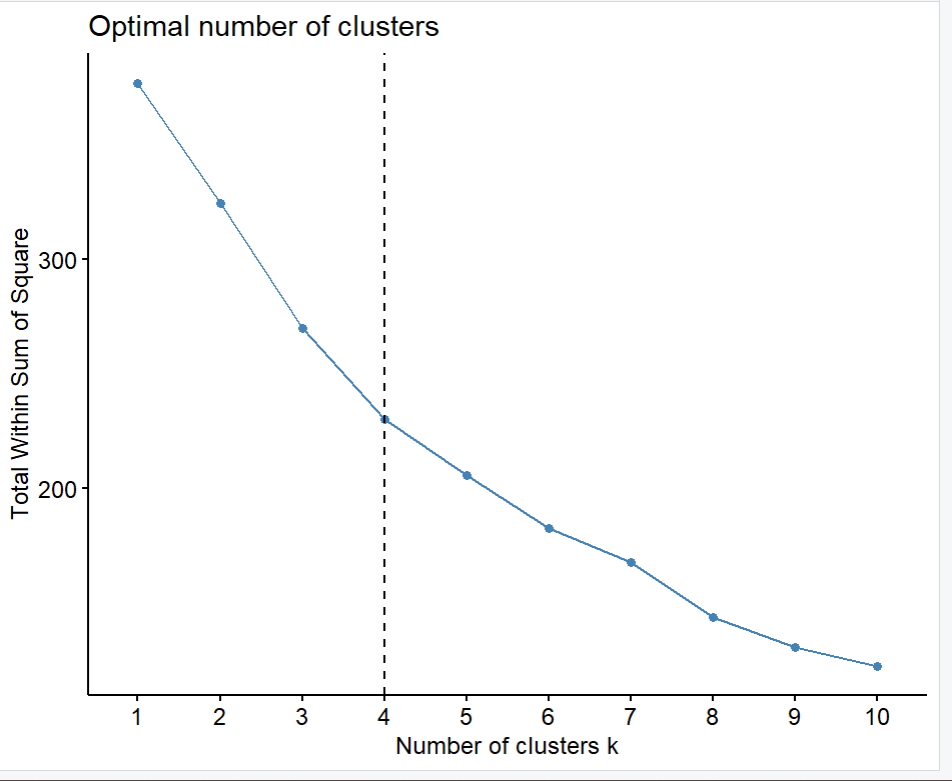
kmeans(df, centers=4, iter.max = 10, nstart = 25)



kmeans(df, centers=5, iter.max = 10, nstart = 25)



fviz\_nbclust(df, kmeans, method = "wss")+ geom\_vline(xintercept = 4, linetype = 2)

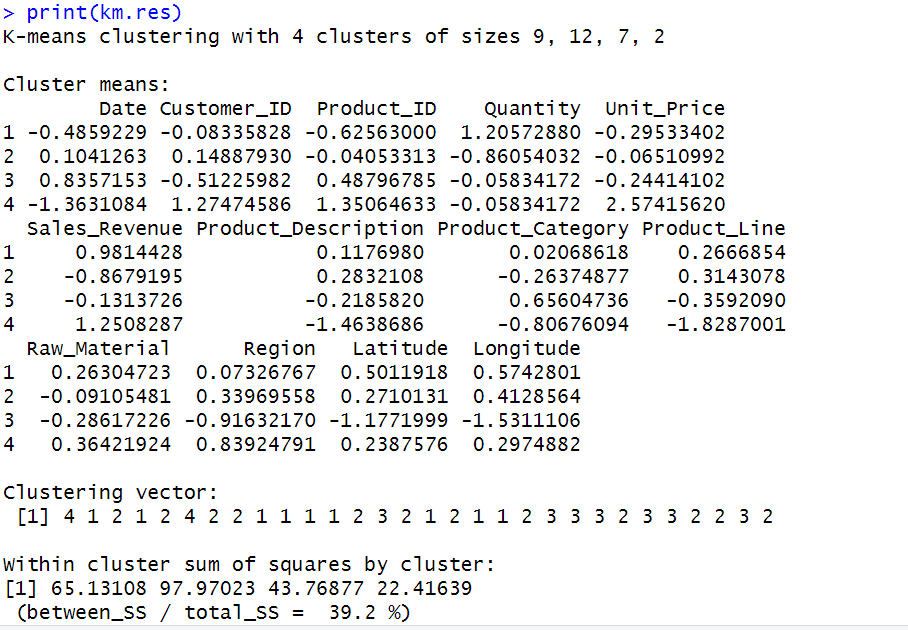


set.seed(123)

km.res <- kmeans(df, 4, nstart = 25)

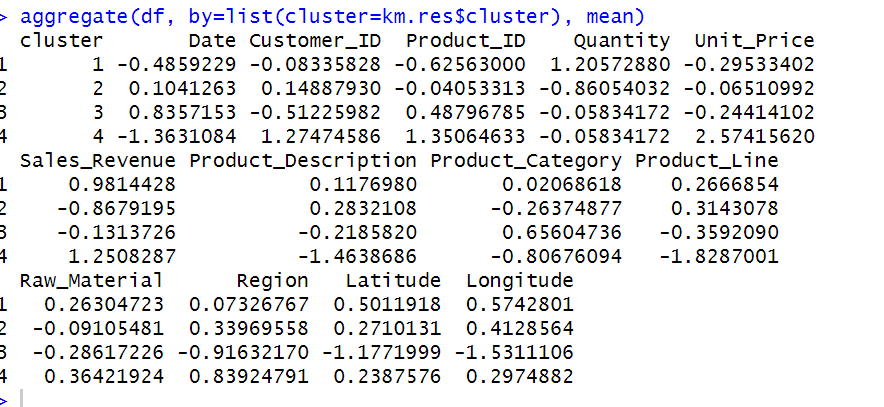
# Print the results

print(km.res)

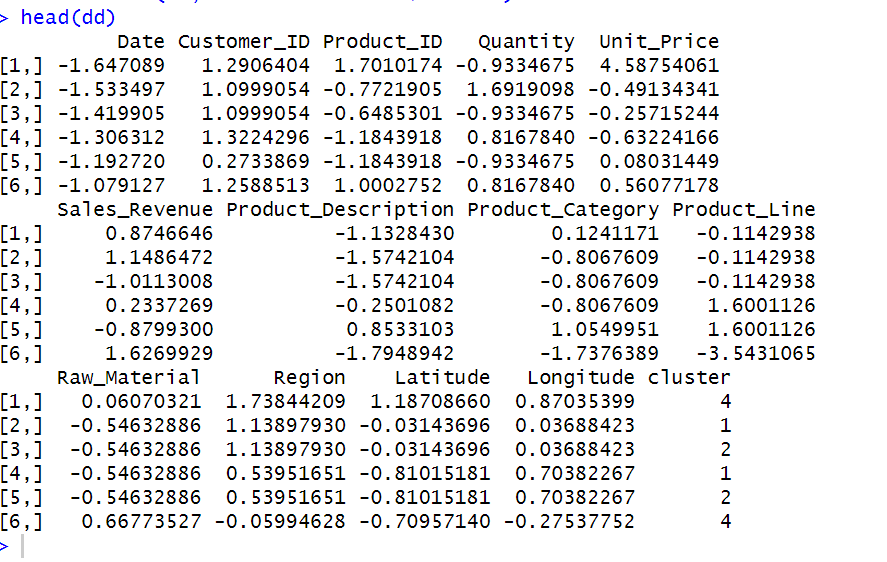


#To find mean of the each variable

aggregate(df, by=list(cluster=km.res$cluster), mean)



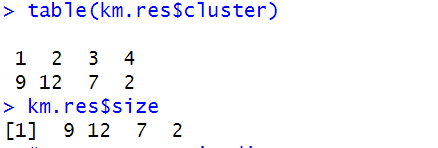
dd <- cbind(df, cluster = km.res$cluster)



head(dd)

table(km.res$cluster)

km.res$size



#To represent in diagram

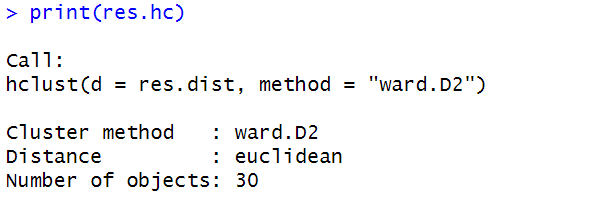
fviz\_cluster(km.res, data = df, palette = c("#2E9FDF", "#00AFBB", "#E7B800", "#FC4E07"), ellipse.type = "euclid", star.plot = TRUE, repel = TRUE, ggtheme = theme\_minimal() )

fviz\_cluster(km.res, data = df, palette = c("set2"), ellipse.type = "euclid", star.plot = TRUE, repel = TRUE, ggtheme = theme\_minimal() )

#Hierarchical clustering

res.hc <- hclust(d = res.dist, method = "ward.D2")

print(res.hc)



#cex: label size

install.packages("factoextra")

library("factoextra")

fviz\_dend(res.hc, cex = 0.5)

# Compute cophentic distance

res.coph <- cophenetic(res.hc)

# Correlation between cophenetic distance and

# the original distance

cor(res.dist, res.coph)

res.hc2 <- hclust(res.dist, method = "average")

cor(res.dist, cophenetic(res.hc2))

grp <- cutree(res.hc, k = 4)

head(grp, n = 4)

table(grp)

