```
# load library
library(ggplot2)
library(gridExtra)
library(readr)
library(ISLR)
library(fpc)
library(dplyr)
library(faraway)
library(cluster)
library(factoextra)
library(tidyverse)
library(NbClust)
library(mvoutlier)

# import data
data <- read_csv("customers.csv")
data <- data.frame(data)
data_0 <- data colSums(is.na(data)) ## check existence of no value
sum(duplicated(data)) ## check duplicate observations

# detect possible outliers
outlier = mvoutlier.CoDa(data)$outliers
sum(outlier) ## 45 outliers detected

# check distribution of features
data %>%psych::describe()
hist(data$Fresh)
hist(data$Milk)
hist(data$Grocery)
hist(data$Frozen)
hist(data$Detergents_Paper)
hist(data$Delicassen)

# transform dataset
data = log(data)
# explore dataset
dim(data)
summary(data)

apply(data,2,var) ## variances of variables are not vastly different from each other
apply(data,2,mean) ## mean of variables are not vastly different from each other

### use K-mean clustering with Euclidean distance
## unscaled data
# visualize data with different k
set.seed(4) # choose the best number of clusters
fviz_nbclust(data, kmeans, method = "wss") + labs(subtitle = "Elbow method") ## using total
within sum of squares
fviz_nbclust(data, kmeans, method = "silhouette") + labs(subtitle = "Silhouette method") ##
using average silhouette
```

```
set.seed(4)
## using prediction strength
ps_data <- prediction.strength(data,Gmin=2,Gmax=10, M=100,clustermethod=kmeansCBI)


ps_data
# k=2 is the best value
# check stability of clusters
boot_kmean_2 <-clusterboot(data=data,bscompare=T, multipleboot=F,bootmethod="boot",
B=100, clustermethod=kmeansCBI, count=T, showplots=T, krange=2, seed = 4)


boot_kmean_2
set.seed(4)
k_means_2=kmeans(data,2) ## clustering data with k=2

# visualize data for k=2
fviz_cluster(k_means_2, geom = "point", data = data) + ggtitle("k = 2")

clusplot(data, k_means_2$cluster, main='2D representation of the Cluster solution',
color=TRUE, shade=TRUE, labels=2, lines=0)


# analyze original data
data_index <- k_means_2$cluster %>% as.factor() ## get labels of observations
data_kmean <- data_0 %>%mutate(total_spend=rowSums(data_0))%>%
mutate(cluster=k_means_2$cluster)
table(data_index)
group_summary = data_kmean %>% group_by(cluster)%>%
summarise_all(funs(round(mean(.))))
group_summary


## scaled data
data.sc <- scale(data) %>% data.frame()


# choose the best number of clusters
set.seed(4)
fviz_nbclust(data.sc, kmeans, method = "wss") + labs(subtitle = "Elbow method") ## using
total within sum of squares
fviz_nbclust(data.sc, kmeans, method = "silhouette") + labs(subtitle = "Silhouette method")
## using average silhouette

set.seed(4)
# using prediction strength
ps_data_sc <- prediction.strength(data.sc,Gmin=2,Gmax=10,
M=100,clustermethod=kmeansCBI) ps_data_sc
# k=2 is the best value
```

```
## check stability of clusters
boot_kmean_2_sc <-clusterboot(data=data.sc,bscompare=T,
multipleboot=F,bootmethod="boot", B=100, clustermethod=kmeansCBI, count=T,
showplots=T, krange=2, seed = 4)


set.seed(4)
k_means_2_sc <- kmeans(data.sc,2) ## clustering with k=2
kmean_index_sc <- k_means_2_sc$cluster %>% as.factor() ## get labels of observation


# visualize data for k=2
fviz_cluster(k_means_2_sc, geom = "point", data = data) + ggtitle("k = 2")
clusplot(data.sc, k_means_2_sc$cluster, main='2D representation of the Cluster solution',
color=TRUE, shade=TRUE, labels=2, lines=0)

# check the matching of clusters between scaled and unscaled data
data.sc_index <- k_means_2_sc$cluster %>% as.factor()
matching <- mean(data_index==data.sc_index)
matching
which(data_index!=data.sc_index) ## unmatched observations

# analyze original data
data_sc_kmean <- data_0 %>%mutate(total_spend=rowSums(data_0))%>%
mutate(cluster=k_means_2_sc$cluster)
table(kmean_index_sc)
group_summary_sc = data_sc_kmean %>% group_by(cluster)%>%
summarise_all(funs(round(mean(.))))
group_summary_sc

# using the principle component for k-means
data.pc <-prcomp(data.sc, scale. = T)
biplot(data.pc,scale=0)
pr.var=data.pc$sdev^2
pve=pr.var/sum(pr.var)
pve

cor(data.sc) ## correlation matrix
```