My Code:

library(dplyr)

library(DataExplorer)

library(ggplot2)

library(cluster)

library(factoextra)

library(gridExtra)

library(purrr)

library(tidyverse)

data = read.csv("~/Desktop/R/Project-2/Projects for Submission/Ecommerce/Project 2\_Dataset.csv")

head(data)

str(data)

colSums(is.na(data))

#Remove column 'X' which has all the rows wit NA

data <-select(data, -c(X))

#Remove rows of CustomerIDvwhich has NA data

data<-na.omit(data)

head(data)

data$InvoiceDate <- as.Date(data$InvoiceDate, "%d-%b-%y")

View(data)

#Computing the line total

data <- data %>% mutate(LineTotal = Quantity \* UnitPrice)

head(data)

#Country summary

countrySummary <- data %>% group\_by(Country) %>% summarise(revenue = sum(LineTotal), transactions = n\_distinct(InvoiceNo)) %>% mutate(aveOrdVal = (round((revenue/transactions), 2))) %>% ungroup() %>% arrange(desc(revenue))

head(countrySummary)

#Total revenue generated and total item purchased by each customer

customerData <- data %>% group\_by(CustomerID, Country) %>% summarise(TotalRevenue = sum(LineTotal), TotalItemsSold = sum(Quantity))

head(customerData)

n\_occur <- data.frame(table(customerData$CustomerID))

single\_ResidentsIds = (n\_occur[n\_occur$Freq == 1,])$Var

customerData <- subset(customerData, (customerData$CustomerID %in% single\_ResidentsIds))

remove(n\_occur)

remove(single\_ResidentsIds)

set.seed(123) #To ensure the same result every time

#Categorical variable - Country

length(unique(data$Country))

#Removing CustomerId and COuntry Columns

customerData <- customerData[-c(1:2)]

#Visualizing outliers

boxplot(customerData)$out

boxplot(customerData, horizontal = TRUE)$out

#Eliminating outliers-1-TotalRevenue

iqr <- IQR(customerData$TotalRevenue)

Q <- quantile(customerData$TotalRevenue, probs = c(.25, .75), na.rm = FALSE)

eliminated <- subset(customerData, customerData$TotalRevenue > (Q[1] - 1.5\*iqr) & customerData$TotalRevenue < (Q[2]+1.5\*iqr))

#Emliminating outliers-2-TotalRevenue

iqr <- IQR(customerData$TotalItemsSold)

Q <- quantile(eliminated$TotalItemsSold, probs = c(.25,.75), na.rm = FALSE)

customerData <- subset(eliminated, eliminated$TotalItemsSold > (Q[1]-1.5\*iqr) & eliminated$TotalItemsSold < (Q[2]+1.5\*iqr))

remove(eliminated)

#Scaling the data before applying the clustering algorithms

customerData <- scale(customerData)

head(customerData)

#Determine optimal number of clusters using Elbow Method

set.seed(123)

#function to compute total within-cluster sum of square

wss <- function(k){

kmeans(customerData, k, nstart = 10)$tot.withinss

}

#Compute and plot wss for k=1 to k=15

k.values <- 1:15

#extract wss for 2-15 clusters

wss\_values <- map\_dbl(k.values, wss)

print(wss\_values)

plot(k.values, wss\_values,

type = "b", pch = 19, frame =FALSE,

xlab = "Number of clusters K",

ylab = "Total within-clusters sum of squares")

##Aggloromative hierachical clustering

#Dissimilarity matrix

d <- dist(customerData, method = "euclidean")

#Hierarchical clustering using complete linkage

hc1 <- hclust(d, method = "complete")

#Plot the obtained dendrogram

plot(hc1, cex = 0.6, hang = -1)

##Diverse Heirachical clustering

#compute divisive heirarchical clustering

hc2 <- diana(customerData)

#Divise coefficient; amount of clustering structure found

hc2$dc

#plot dendrogram

pltree(hc2, cex=0.6, hang=-1, main="Dendrogram of diana")

#Methods to asses

m <- c("average", "single", "complete","ward")

names(m) <- c("average", "single", "complete", "ward")

#function to compute coefficient

ac <- function(x){

agnes(customerData, method = x)$ac

}

map\_dbl(m, ac)

#Ward's method

hc3 <- hclust(d, method = "ward.D2")

#cut tree into 3 groups

sub\_grp <- cutree(hc3, k = 3)

str(sub\_grp)

head(sub\_grp)

table(sub\_grp)

fviz\_cluster(list(data = customerData, cluster = sub\_grp))

#Dendogram with border around 3 clusters

plot(hc3, cex = 0.6)

rect.hclust(hc3, k=3, border = 5:10)

#Determining Optimal clusters in Hierachical Clustering using Elbow Method

fviz\_nbclust(customerData, FUN = hcut, method = "wss")

#compute WSS for given k value-- Hierachical Clustering

wssForHierachical <- function(k){

sub\_grp <- cutree(hc3, k)

df = as.data.frame(customerData)

df2 = df %>%

mutate(cluster = sub\_grp) %>%

group\_by(cluster) %>%

summarize(TR=mean(TotalRevenue), TS=mean(TotalItemsSold))

df1=df %>%

mutate(cluster = sub\_grp)

df3=left\_join(df1, df2, by="cluster")

D=(df3$TotalRevenue-df3$TR)^2 + (df3$TotalItemSold-df3$TS)^2

df4=cbind(df3, D)

wss=sum(df4$D)

WSS

}

#calculating WSS at hcut=3

wssForHierachical(3)