Problem\_Set\_4

Amy Richardson

3/31/2021

#Initiate packages

library(tidyverse)

## Warning: package 'tidyverse' was built under R version 4.0.3

## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.3 v purrr 0.3.4  
## v tibble 3.0.5 v dplyr 1.0.3  
## v tidyr 1.1.2 v stringr 1.4.0  
## v readr 1.4.0 v forcats 0.5.0

## Warning: package 'ggplot2' was built under R version 4.0.3

## Warning: package 'tibble' was built under R version 4.0.3

## Warning: package 'tidyr' was built under R version 4.0.3

## Warning: package 'readr' was built under R version 4.0.3

## Warning: package 'purrr' was built under R version 4.0.3

## Warning: package 'dplyr' was built under R version 4.0.3

## Warning: package 'stringr' was built under R version 4.0.3

## Warning: package 'forcats' was built under R version 4.0.3

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(ggplot2)  
library(cluster)  
library(factoextra)

## Warning: package 'factoextra' was built under R version 4.0.4

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

library(dendextend)

## Warning: package 'dendextend' was built under R version 4.0.4

##   
## ---------------------  
## Welcome to dendextend version 1.14.0  
## Type citation('dendextend') for how to cite the package.  
##   
## Type browseVignettes(package = 'dendextend') for the package vignette.  
## The github page is: https://github.com/talgalili/dendextend/  
##   
## Suggestions and bug-reports can be submitted at: https://github.com/talgalili/dendextend/issues  
## Or contact: <tal.galili@gmail.com>  
##   
## To suppress this message use: suppressPackageStartupMessages(library(dendextend))  
## ---------------------

##   
## Attaching package: 'dendextend'

## The following object is masked from 'package:stats':  
##   
## cutree

library(dplyr)  
library(tidyr)  
library(psych)

## Warning: package 'psych' was built under R version 4.0.3

##   
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

library(pastecs)

## Warning: package 'pastecs' was built under R version 4.0.4

##   
## Attaching package: 'pastecs'

## The following objects are masked from 'package:dplyr':  
##   
## first, last

## The following object is masked from 'package:tidyr':  
##   
## extract

library(car)

## Warning: package 'car' was built under R version 4.0.4

## Loading required package: carData

## Warning: package 'carData' was built under R version 4.0.3

##   
## Attaching package: 'car'

## The following object is masked from 'package:psych':  
##   
## logit

## The following object is masked from 'package:dplyr':  
##   
## recode

## The following object is masked from 'package:purrr':  
##   
## some

library(ggpubr)

## Warning: package 'ggpubr' was built under R version 4.0.4

##   
## Attaching package: 'ggpubr'

## The following object is masked from 'package:dendextend':  
##   
## rotate

library(pgirmess)

## Warning: package 'pgirmess' was built under R version 4.0.4

##   
## Attaching package: 'pgirmess'

## The following object is masked from 'package:psych':  
##   
## shannon

library(gridExtra)

## Warning: package 'gridExtra' was built under R version 4.0.3

##   
## Attaching package: 'gridExtra'

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

##NYC Taxi Cab Data

taxi\_df <- read.csv("yellow\_tripdata\_2019-11.csv", header = TRUE)  
zonei\_df <- read.csv("taxi+\_zone\_lookup.csv", header = TRUE)  
  
taxi\_sub <- select(taxi\_df, PULocationID, trip\_distance, total\_amount)  
  
Loc\_df <- merge(taxi\_sub, zonei\_df, by.x = "PULocationID", by.y = "LocationID")  
Loc\_df <- select(Loc\_df, Zone, PULocationID, trip\_distance, total\_amount)  
Loc\_df <- filter(Loc\_df, trip\_distance > 0.1)  
Loc\_df <- filter(Loc\_df, trip\_distance < 100)  
Loc\_df<- filter(Loc\_df, total\_amount > 0 )  
  
mean\_df <- Loc\_df %>% group\_by(Zone) %>%  
 summarise\_at(vars(-PULocationID), funs(mean(., na.rm=TRUE)))

## Warning: `funs()` is deprecated as of dplyr 0.8.0.  
## Please use a list of either functions or lambdas:   
##   
## # Simple named list:   
## list(mean = mean, median = median)  
##   
## # Auto named with `tibble::lst()`:   
## tibble::lst(mean, median)  
##   
## # Using lambdas  
## list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_warnings()` to see where this warning was generated.

mean\_df <- filter(mean\_df, Zone != "NV")  
mean\_df <- inner\_join(mean\_df,zonei\_df)

## Joining, by = "Zone"

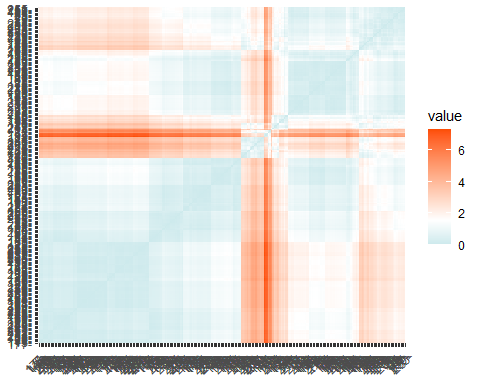
mean\_df <- select(mean\_df, Zone, Borough, trip\_distance, total\_amount, )  
  
mean\_df <- rename(mean\_df,  
 Distance = trip\_distance,  
 Total\_Cost = total\_amount  
 )

Standardize Trip Distance and Total Amount

mean\_df$trip\_distance <-scale(mean\_df$Distance)  
mean\_df$total\_amount <- scale(mean\_df$Total\_Cost)  
  
mean\_df <- select(mean\_df, Zone, Borough, trip\_distance, total\_amount, Distance, Total\_Cost)

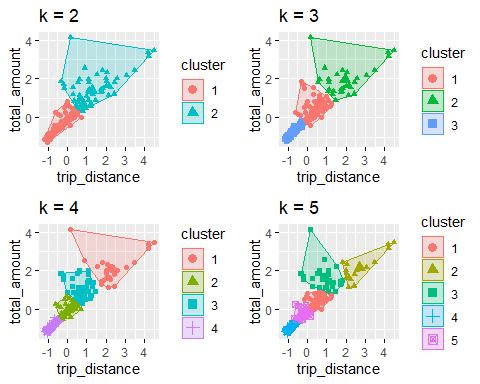
Initial visulization

distance <- get\_dist(mean\_df[,3:4])  
fviz\_dist(distance, gradient = list(low = "#00AFBB", mid = "white", high = "#FC4E07"))

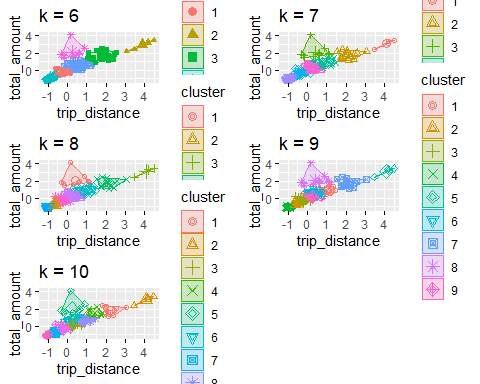


Can examine different number of groups

set.seed(123)  
k2 <- kmeans(mean\_df[,3:4], 2, nstart = 25)  
k3 <- kmeans(mean\_df[,3:4], 3, nstart = 25)  
k4 <- kmeans(mean\_df[,3:4], 4, nstart = 25)  
k5 <- kmeans(mean\_df[,3:4], 5, nstart = 25)  
k6 <- kmeans(mean\_df[,3:4], 6, nstart = 25)  
k7 <- kmeans(mean\_df[,3:4], 7, nstart = 25)  
k8 <- kmeans(mean\_df[,3:4], 8, nstart = 25)  
k9 <- kmeans(mean\_df[,3:4], 9, nstart = 25)  
k10 <- kmeans(mean\_df[,3:4], 10, nstart = 25)  
  
# plots to compare  
p1 <- fviz\_cluster(k2, geom = "point", data = mean\_df[,3:4]) + ggtitle("k = 2")  
p2 <- fviz\_cluster(k3, geom = "point", data = mean\_df[,3:4]) + ggtitle("k = 3")  
p3 <- fviz\_cluster(k4, geom = "point", data = mean\_df[,3:4]) + ggtitle("k = 4")  
p4 <- fviz\_cluster(k5, geom = "point", data = mean\_df[,3:4]) + ggtitle("k = 5")  
p5 <- fviz\_cluster(k6, geom = "point", data = mean\_df[,3:4]) + ggtitle("k = 6")  
p6 <- fviz\_cluster(k7, geom = "point", data = mean\_df[,3:4]) + ggtitle("k = 7")  
p7 <- fviz\_cluster(k8, geom = "point", data = mean\_df[,3:4]) + ggtitle("k = 8")  
p8 <- fviz\_cluster(k9, geom = "point", data = mean\_df[,3:4]) + ggtitle("k = 9")  
p9 <- fviz\_cluster(k10, geom = "point", data = mean\_df[,3:4]) + ggtitle("k = 10")  
  
  
grid.arrange(p1, p2, p3, p4, nrow = 2)

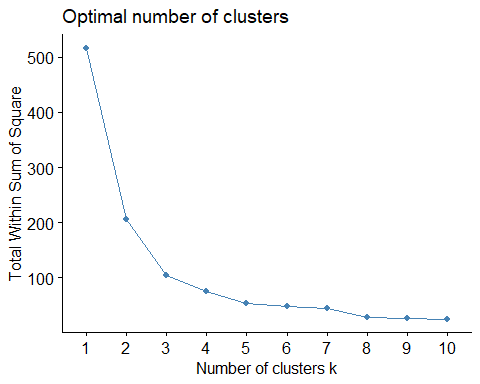


grid.arrange(p5, p6, p7, p8, p9, nrow = 3)



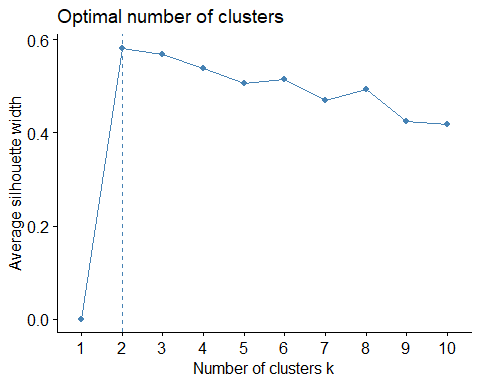
Using Elbow Method to choose the optimum number of clusters [k]

set.seed(123)  
  
fviz\_nbclust(mean\_df[,3:4], kmeans, method = "wss")



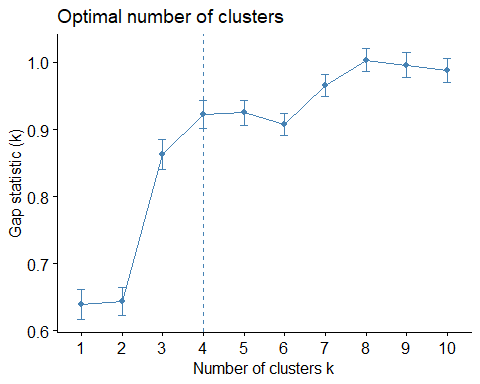
Use Average Silhouette Method

fviz\_nbclust(mean\_df[,3:4], kmeans, method = "silhouette")



Gap Statistic Method

set.seed(123)  
gap\_stat <- clusGap(mean\_df[,3:4], FUN = kmeans, nstart = 25,  
 K.max = 10, B = 50)  
fviz\_gap\_stat(gap\_stat)

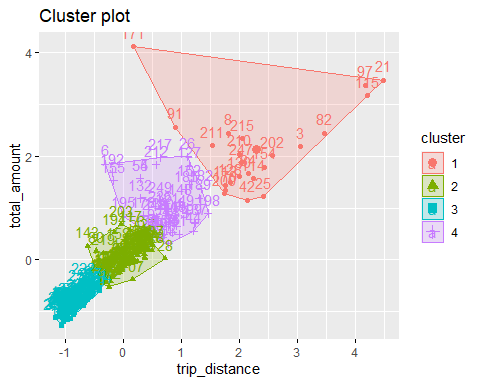


Extract Results

set.seed(123)  
final <- kmeans(mean\_df[,3:4], 4, nstart = 25)

Plot Final Groups

fviz\_cluster(final, data = mean\_df[,3:4])



Find descriptive statistics for the 4 clusters

mean\_df %>%  
 mutate(Cluster = final$cluster) %>%  
 group\_by(Cluster) %>%  
 summarise\_all("mean")

## Warning in mean.default(Zone): argument is not numeric or logical: returning NA  
  
## Warning in mean.default(Zone): argument is not numeric or logical: returning NA  
  
## Warning in mean.default(Zone): argument is not numeric or logical: returning NA  
  
## Warning in mean.default(Zone): argument is not numeric or logical: returning NA

## Warning in mean.default(Borough): argument is not numeric or logical: returning  
## NA  
  
## Warning in mean.default(Borough): argument is not numeric or logical: returning  
## NA  
  
## Warning in mean.default(Borough): argument is not numeric or logical: returning  
## NA  
  
## Warning in mean.default(Borough): argument is not numeric or logical: returning  
## NA

## # A tibble: 4 x 7  
## Cluster Zone Borough trip\_distance total\_amount Distance Total\_Cost  
## \* <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1 NA NA 2.29 2.12 16.4 62.2  
## 2 2 NA NA 0.0322 -0.0222 6.83 31.1  
## 3 3 NA NA -0.855 -0.883 3.09 18.6  
## 4 4 NA NA 0.712 0.987 9.70 45.7

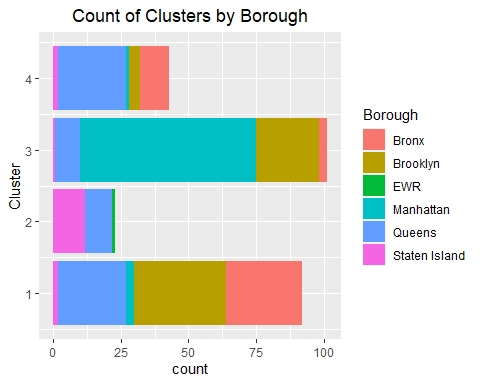
Add Cluster numbers in new Dataframe

# Cluster Analysis - kmeans  
k4\_2 <- kmeans(mean\_df[,3:4], 4)  
kmeans\_basic\_table <- data.frame(k4\_2$size, k4\_2$centers)  
kmeans\_basic\_df <- data.frame(Cluster = k4\_2$cluster, mean\_df)  
# head of df  
head(kmeans\_basic\_df)

## Cluster Zone Borough trip\_distance total\_amount  
## 1 1 Allerton/Pelham Gardens Bronx 0.3909584 0.2290112  
## 2 3 Alphabet City Manhattan -0.9435514 -0.9337817  
## 3 2 Arden Heights Staten Island 3.0478086 2.1859477  
## 4 1 Arrochar/Fort Wadsworth Staten Island -0.2226912 -0.4155193  
## 5 3 Astoria Queens -0.9120129 -1.1389667  
## 6 4 Astoria Park Queens -0.3210172 1.8472411  
## Distance Total\_Cost  
## 1 8.346726 34.73504  
## 2 2.710992 17.85424  
## 3 19.566800 63.14480  
## 4 5.755238 25.37810  
## 5 2.844182 14.87547  
## 6 5.340000 58.22764

Example ggplot

ggplot(data = kmeans\_basic\_df, aes(y = Cluster)) +  
 geom\_bar(aes(fill = Borough)) +  
 ggtitle("Count of Clusters by Borough") +  
 theme(plot.title = element\_text(hjust = 0.5))



##END