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 )  
  
Loc\_df <- inner\_join(Loc\_df,zonei\_df)

## Joining, by = "Zone"

Loc\_df <- select(Loc\_df, Zone, Borough, PULocationID, trip\_distance, total\_amount)  
Loc\_df <- filter(Loc\_df, Borough != "EWR")  
Loc\_df <- filter(Loc\_df, Borough != "Unknown")  
  
Loc\_df <- rename(Loc\_df,  
 Distance = trip\_distance,  
 Total\_Cost = total\_amount  
 )

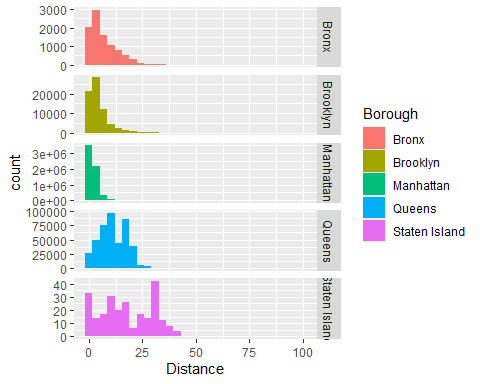
Standardize Trip Distance and Total Amount

Loc\_df$trip\_distance <-scale(Loc\_df$Distance)  
Loc\_df$total\_amount <- scale(Loc\_df$Total\_Cost)

Histogram by Borough

Loc\_df %>%  
 ggplot(aes(x = Distance, fill = Borough)) +  
 geom\_histogram () +  
 facet\_grid(Borough~., scales = "free")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



Describe by Borough

describe(Loc\_df$trip\_distance)

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 6722084 0 1 -0.35 -0.24 0.32 -0.75 25.15 25.89 3.24 14.36 0

by(Loc\_df$trip\_distance, Loc\_df$Borough, stat.desc)

## [1] 9446.00000000 0.00000000 0.00000000 -0.74571874 25.14667075  
## [6] 25.89238948 9354.90012427 0.45187247 0.99035572 0.01680949  
## [11] 0.03295022 2.66905262 1.63372354 1.64963307  
## ------------------------------------------------------------   
## [1] 7.039200e+04 0.000000e+00 0.000000e+00 -7.457187e-01 1.221476e+01  
## [6] 1.296048e+01 2.560526e+04 2.063572e-02 3.637524e-01 4.150945e-03  
## [11] 8.135843e-03 1.212879e+00 1.101308e+00 3.027630e+00  
## ------------------------------------------------------------   
## [1] 6.211878e+06 0.000000e+00 0.000000e+00 -7.457187e-01 2.512589e+01  
## [6] 2.587161e+01 -9.912490e+05 -3.742317e-01 -1.595732e-01 2.824619e-04  
## [11] 5.536152e-04 4.956129e-01 7.039978e-01 -4.411756e+00  
## ------------------------------------------------------------   
## [1] 4.301240e+05 0.000000e+00 0.000000e+00 -7.457187e-01 2.504016e+01  
## [6] 2.578588e+01 9.553417e+05 1.950810e+00 2.221084e+00 2.576565e-03  
## [11] 5.049989e-03 2.855459e+00 1.689810e+00 7.608042e-01  
## ------------------------------------------------------------   
## [1] 244.0000000 0.0000000 0.0000000 -0.7353275 10.2949811 11.0303086  
## [7] 947.1705067 3.6484798 3.8818463 0.1963701 0.3868047 9.4089342  
## [13] 3.0673986 0.7901906

Randomly choose 50,000 trips - to reduce running time of cluster

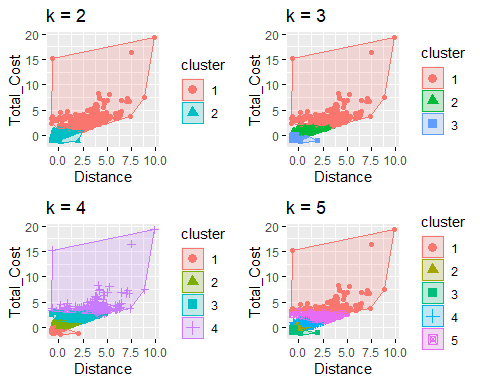
Loc\_scale <- sample\_n(Loc\_df, 10000)

Can examine different number of groups

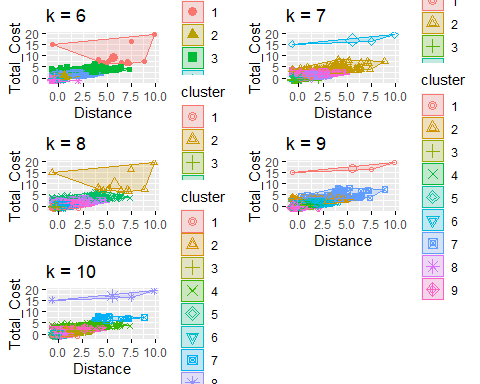
set.seed(123)  
k2 <- kmeans(Loc\_scale [,4:5], 2, nstart = 25)  
k3 <- kmeans(Loc\_scale [,4:5], 3, nstart = 25)  
k4 <- kmeans(Loc\_scale [,4:5], 4, nstart = 25)  
k5 <- kmeans(Loc\_scale [,4:5], 5, nstart = 25)  
k6 <- kmeans(Loc\_scale [,4:5], 6, nstart = 25)  
k7 <- kmeans(Loc\_scale [,4:5], 7, nstart = 25)  
k8 <- kmeans(Loc\_scale [,4:5], 8, nstart = 25)  
k9 <- kmeans(Loc\_scale [,4:5], 9, nstart = 25)  
k10 <- kmeans(Loc\_scale [,4:5], 10, nstart = 25)

## Warning: Quick-TRANSfer stage steps exceeded maximum (= 500000)

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

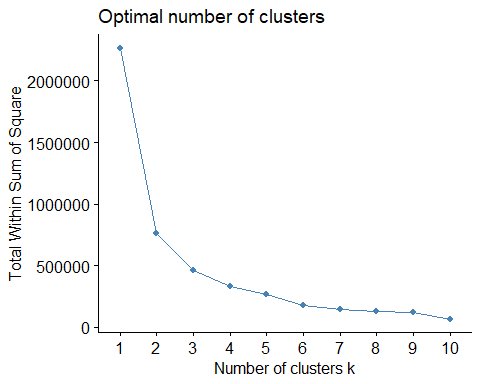


grid.arrange(p5, p6, p7, p8, p9, nrow = 3)



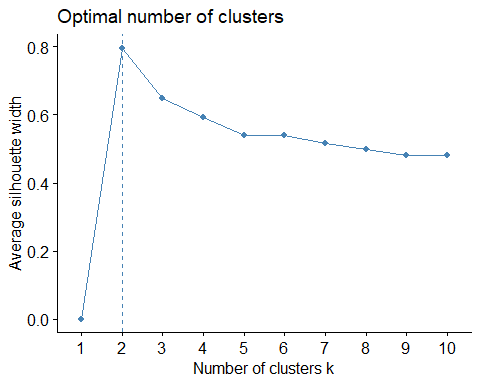
Using Elbow Method to choose the optimum number of clusters [k]

set.seed(123)  
  
fviz\_nbclust(Loc\_scale [,4:5], kmeans, method = "wss")



Use Average Silhouette Method

fviz\_nbclust(Loc\_scale [,4:5], kmeans, method = "silhouette")

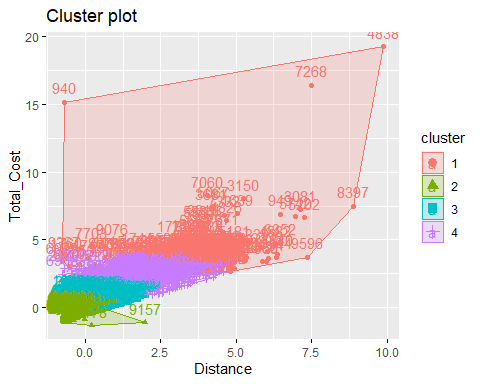


Extract Results

set.seed(123)  
final <- kmeans(Loc\_scale [,4:5], 4, nstart = 25)

Plot Final Groups

fviz\_cluster(final, data = Loc\_scale [,4:5])



Add Cluster numbers in new Dataframe

k4\_2 <- kmeans(Loc\_scale [,4:5], 4)  
kmeans\_basic\_table <- data.frame(k4\_2$size, k4\_2$centers)  
kmeans\_basic\_df <- data.frame(Cluster = k4\_2$cluster, Loc\_scale)  
head(kmeans\_basic\_df)

## Cluster Zone Borough PULocationID Distance  
## 1 3 East Village Manhattan 79 3.10  
## 2 1 Central Park Manhattan 43 1.24  
## 3 3 Times Sq/Theatre District Manhattan 230 1.82  
## 4 1 TriBeCa/Civic Center Manhattan 231 1.30  
## 5 3 Midtown East Manhattan 162 3.30  
## 6 3 Meatpacking/West Village West Manhattan 158 5.20  
## Total\_Cost trip\_distance total\_amount  
## 1 20.75 0.03102697 0.10008160  
## 2 12.96 -0.45216601 -0.44660513  
## 3 18.59 -0.30149293 -0.05150291  
## 4 11.30 -0.43657914 -0.56310064  
## 5 19.55 0.08298321 0.01586798  
## 6 20.80 0.57656743 0.10359050

Find descriptive statistics for the 4 clusters

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

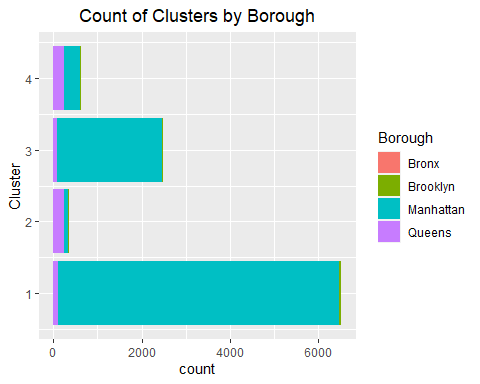
## Warning in mean.default(Zone): argument is not numeric or logical: returning NA  
  
## Warning in mean.default(Zone): argument is not numeric or logical: returning NA  
  
## Warning in mean.default(Zone): argument is not numeric or logical: returning NA  
  
## Warning in mean.default(Zone): argument 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 8  
## Cluster Zone Borough PULocationID Distance Total\_Cost trip\_distance  
## \* <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1 NA NA 139. 17.5 74.7 3.78   
## 2 2 NA NA 166. 1.31 12.6 -0.434  
## 3 3 NA NA 159. 3.67 23.5 0.180  
## 4 4 NA NA 153. 9.94 45.0 1.81   
## # ... with 1 more variable: total\_amount <dbl>

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