### Clustering

## Libraries

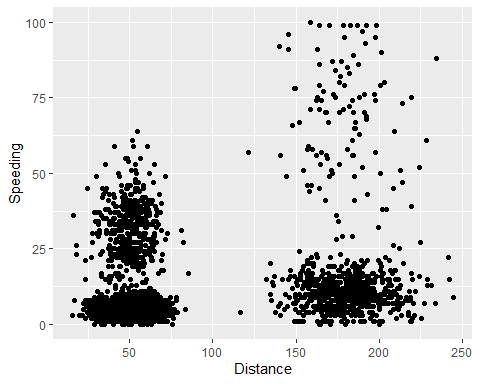
library(tidyverse)  
library(tidymodels)

trucks <- read\_csv("trucks.csv")  
  
#str(trucks)

## Task 1

There does appear to be some natural clustering in the data. There is a cluster of points traveling less than ~80 miles and speeding for less than ~12% of the time. Another cluster of those that were speeding for more than 12% of the time while driving 80 miles or less. There are 2 more clusters for those traveling 125+ miles. There is a group that was speeding between 0-25% of the trip and another, more scattered cluster for those that were speeding for more than 25% of the trip.

ggplot(trucks, aes(x=Distance, y=Speeding)) + geom\_point()



## Task 2

kmeans\_recipe = recipe(~ Distance + Speeding, trucks)  
  
trucks\_dummy = kmeans\_recipe %>%  
 step\_scale(all\_numeric()) %>%  
 step\_center(all\_numeric())  
  
trucks\_dummy = prep(trucks\_dummy, trucks)  
  
trucks\_cleaned = bake(trucks\_dummy, trucks)

### Task 3

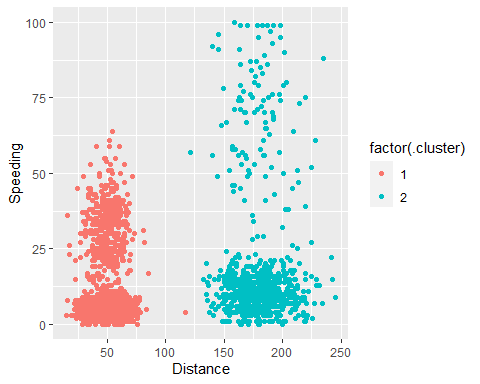
The clusters are predictable based on the initial scatter plot. With only 2 groups, they are grouped by distance traveled into those above 125 and those under 125 miles.

set.seed(64)  
clusters = kmeans(trucks\_cleaned, 2)

trucks = augment(clusters, trucks)  
str(trucks)

## tibble [4,000 x 4] (S3: tbl\_df/tbl/data.frame)  
## $ Driver\_ID: num [1:4000] 3.42e+09 3.42e+09 3.42e+09 3.42e+09 3.42e+09 ...  
## $ Distance : num [1:4000] 71.2 52.5 64.5 55.7 54.6 ...  
## $ Speeding : num [1:4000] 28 25 27 22 25 10 20 8 34 19 ...  
## $ .cluster : Factor w/ 2 levels "1","2": 1 1 1 1 1 1 1 1 1 1 ...

ggplot(trucks, aes(x=Distance, y=Speeding, color=factor(.cluster))) + geom\_point()



## Task 4

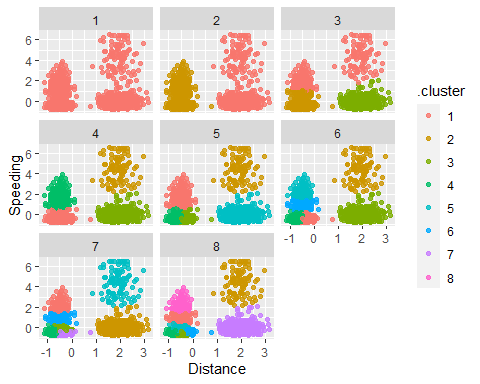
4 clusters seem appropriate for this data set based on the visualizations below.

set.seed(412)  
clusts =  
 tibble(k= 1:8) %>%  
 mutate(  
 kclust = map(k, ~kmeans(trucks\_cleaned, .x)),  
 tidied = map(kclust, tidy),  
 glanced = map(kclust, glance),  
 augmented = map(kclust, augment, trucks\_cleaned)  
 )  
  
clusts

## # A tibble: 8 x 5  
## k kclust tidied glanced augmented   
## <int> <list> <list> <list> <list>   
## 1 1 <kmeans> <tibble [1 x 5]> <tibble [1 x 4]> <tibble [4,000 x 3]>  
## 2 2 <kmeans> <tibble [2 x 5]> <tibble [1 x 4]> <tibble [4,000 x 3]>  
## 3 3 <kmeans> <tibble [3 x 5]> <tibble [1 x 4]> <tibble [4,000 x 3]>  
## 4 4 <kmeans> <tibble [4 x 5]> <tibble [1 x 4]> <tibble [4,000 x 3]>  
## 5 5 <kmeans> <tibble [5 x 5]> <tibble [1 x 4]> <tibble [4,000 x 3]>  
## 6 6 <kmeans> <tibble [6 x 5]> <tibble [1 x 4]> <tibble [4,000 x 3]>  
## 7 7 <kmeans> <tibble [7 x 5]> <tibble [1 x 4]> <tibble [4,000 x 3]>  
## 8 8 <kmeans> <tibble [8 x 5]> <tibble [1 x 4]> <tibble [4,000 x 3]>

clusters =   
 clusts %>%  
 unnest(cols = c(tidied))  
  
assignments =   
 clusts %>%  
 unnest(cols = c(augmented))  
  
clusterings =   
 clusts %>%  
 unnest(cols = c(glanced))

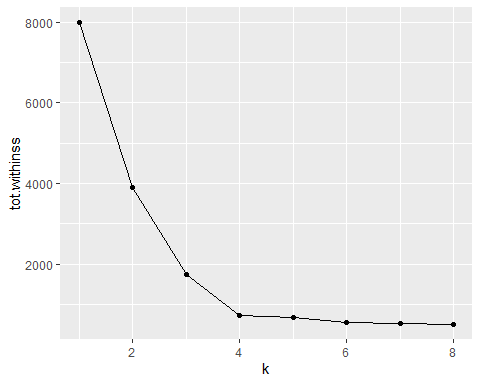
p1 =   
 ggplot(assignments, aes(x=Distance, y=Speeding)) + geom\_point(aes(color=.cluster), alpha = 0.8) + facet\_wrap(~k)  
  
p1



## Task 5

Based on the plot, k = 4 is at the “elbow” of the line and seems to be most appropriate.

ggplot(clusterings, aes(k, tot.withinss)) + geom\_line() + geom\_point()



## Task 6

These clusters seem most appropriate for this data based on the analysis above. It was predicted based on the initial visualization that 4 clusters could be made from this data. The analysis agrees with the inital prediction.

set.seed(64)  
clusters = kmeans(trucks\_cleaned, 4)

trucks = augment(clusters, trucks)  
str(trucks)

## tibble [4,000 x 4] (S3: tbl\_df/tbl/data.frame)  
## $ Driver\_ID: num [1:4000] 3.42e+09 3.42e+09 3.42e+09 3.42e+09 3.42e+09 ...  
## $ Distance : num [1:4000] 71.2 52.5 64.5 55.7 54.6 ...  
## $ Speeding : num [1:4000] 28 25 27 22 25 10 20 8 34 19 ...  
## $ .cluster : Factor w/ 4 levels "1","2","3","4": 3 3 3 3 3 1 3 1 3 3 ...

ggplot(trucks, aes(x=Distance, y=Speeding, color=factor(.cluster))) + geom\_point()

