library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.0 ──

## ✓ ggplot2 3.3.2 ✓ purrr 0.3.4  
## ✓ tibble 3.0.4 ✓ dplyr 1.0.2  
## ✓ tidyr 1.1.2 ✓ stringr 1.4.0  
## ✓ readr 1.4.0 ✓ forcats 0.5.0

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

library(tidymodels)

## ── Attaching packages ────────────────────────────────────── tidymodels 0.1.2 ──

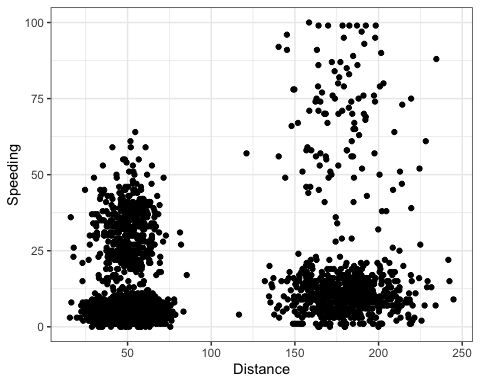
## ✓ broom 0.7.2 ✓ recipes 0.1.15  
## ✓ dials 0.0.9 ✓ rsample 0.0.8   
## ✓ infer 0.5.4 ✓ tune 0.1.2   
## ✓ modeldata 0.1.0 ✓ workflows 0.2.1   
## ✓ parsnip 0.1.5 ✓ yardstick 0.0.7

## ── Conflicts ───────────────────────────────────────── tidymodels\_conflicts() ──  
## x scales::discard() masks purrr::discard()  
## x dplyr::filter() masks stats::filter()  
## x recipes::fixed() masks stringr::fixed()  
## x dplyr::lag() masks stats::lag()  
## x yardstick::spec() masks readr::spec()  
## x recipes::step() masks stats::step()

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

##   
## ── Column specification ────────────────────────────────────────────────────────  
## cols(  
## Driver\_ID = col\_double(),  
## Distance = col\_double(),  
## Speeding = col\_double()  
## )

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



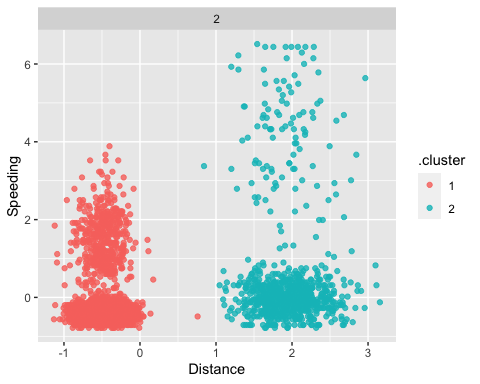
Describe the relationship? Does there appear to be any natural clustering of driver? Based on the plot, the speed is between 0 and 70 of distances 0 to 100. While distances of 101 to 250, speed ranges from 0 to 100. Yes, there appears to be natural clustering of drivers.

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)

set.seed(64)  
clusts =   
 tibble(k=2) %>%  
 mutate(  
 kclust = map(k, ~kmeans(trucks\_cleaned, .x)),  
 tidied = map(kclust, tidy),  
 glanced = map(kclust, glance),  
 augmented = map(kclust, augment, trucks\_cleaned)  
 )

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 = .8) +  
 facet\_wrap(~k)  
p1

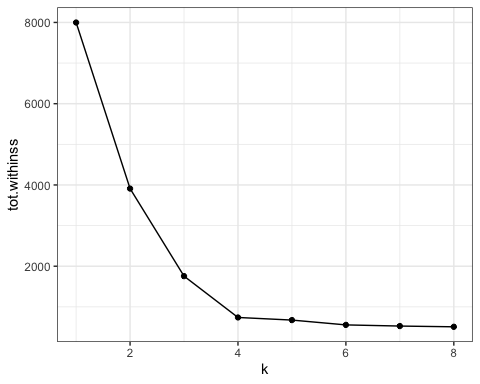


Comment on the clusters: The clusters appear to be similar to the clusters we saw in the first plot. There are two points from both clusters that seem to line up similarly in the middle.

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)  
 )

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

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

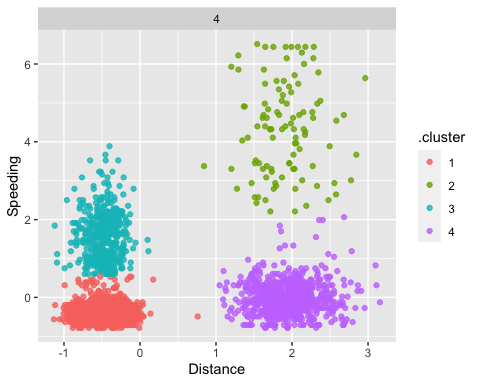


The value of k that appears to be the most appropriate is 4.

set.seed(64)  
clusts =   
 tibble(k=4) %>%  
 mutate(  
 kclust = map(k, ~kmeans(trucks\_cleaned, .x)),  
 tidied = map(kclust, tidy),  
 glanced = map(kclust, glance),  
 augmented = map(kclust, augment, trucks\_cleaned)  
 )

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

p2 =  
 ggplot(assignments, aes(x=Distance, y=Speeding)) +  
 geom\_point(aes(color = .cluster), alpha = .8) +  
 facet\_wrap(~k)  
p2



Now the 4 clusters are split between the low distance with low speed, low distance with higher speed, high distance with low speed, and high distance with high speed.