Problem set 2

David Segovia

```
library(tidyverse)
```

```
## ── Attaching packages ─────────────
```

```
## ✓ ggplot2 3.3.2     ✓ purrr   0.3.4
## ✓ tibble  3.0.3     ✓ dplyr   1.0.2
## ✓ tidyr   1.1.1     ✓ stringr 1.4.0
## ✓ readr   1.3.1     ✓ forcats 0.5.0
```

```
## ── Conflicts ────────────────────
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
setwd("~/Downloads/PA 528 Public Program Evaluation/Problem Sets/PS2/data")
car <- read_csv("car_data.csv")
```

```
## Parsed with column specification:
## cols(
##   manufacturer = col_character(),
##   model = col_character(),
##   displ = col_double(),
##   year = col_double(),
##   cyl = col_double(),
##   trans = col_character(),
##   drv = col_character(),
##   cty = col_double(),
##   hwy = col_double(),
##   fl = col_character(),
##   class = col_character()
## )
```

Learning R

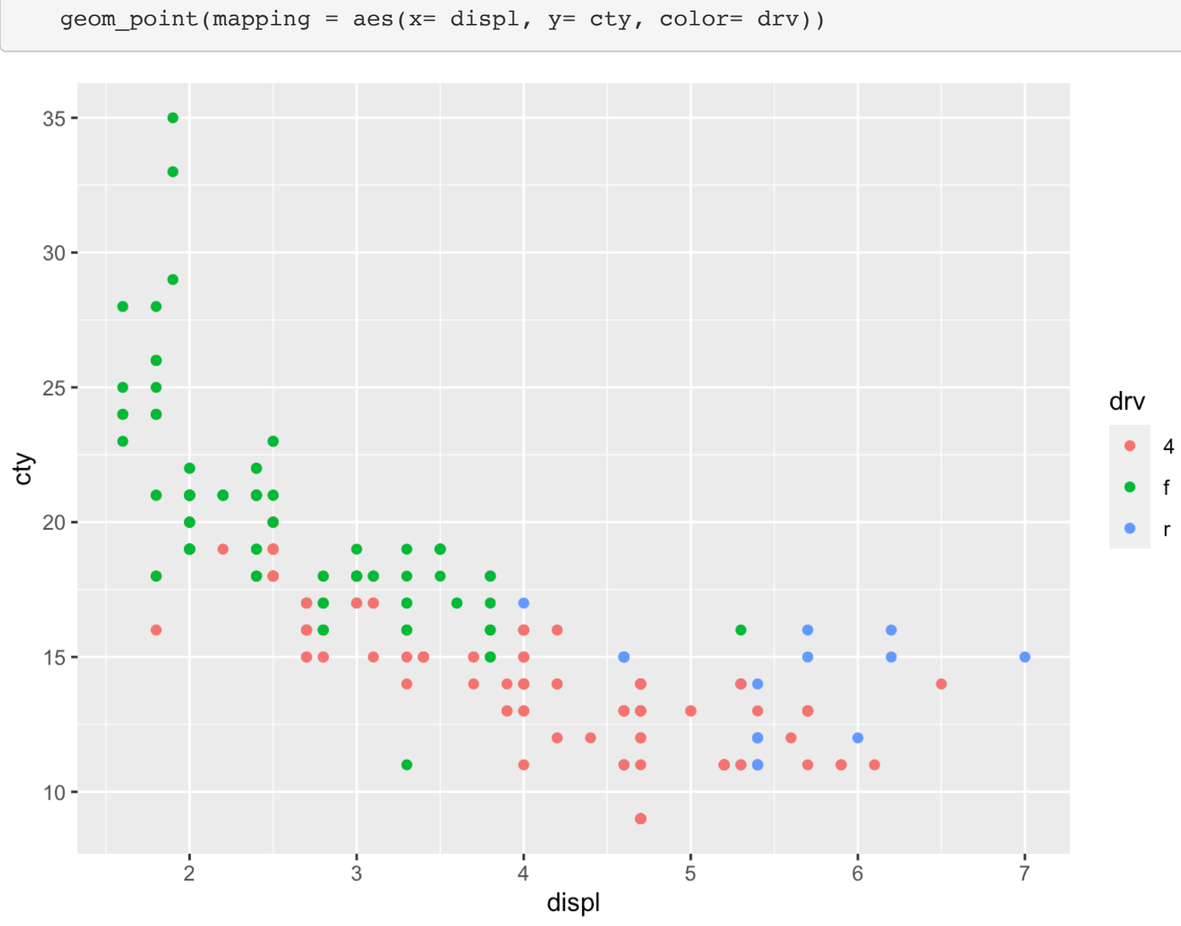
Tell me that you worked through the primers and videos and examples at the example page for this week:

I did all the primers and had the time of my life!

My first plots

Insert a chunk below and use it to create a scatterplot (hint: `geom_point()`) with diplacement (`displ`) on the x-axis, city MPG (`cty`) on the y-axis, and with the points colored by drive (`drv`).

```
ggplot(data= car) +
   geom_point(mapping = aes(x= displ, y= cty, color= drv))
```



Insert a chunk below and use it to create a histogram (hint: `geom_histogram()`) with highway MPG (`hwy`) on the x-axis. Do not include anything on the y-axis (`geom_histogram()` will do that automatically for you). Choose an appropriate bin width. If you're brave, facet by drive (`drv`).

```
ggplot(data = car) +
   geom_histogram(mapping =aes(x= hwy), binwidth = 1) + facet_wrap(~drv)
```


My first data manipulation

Insert a chunk below and use it to calculate the average city MPG (`cty`) by class of car (`class`). This won't be a plot—it'll be a table. Hint: use a combination of `group_by()` and `summarize()`.

```
car %>%
   group_by(class) %>%
   summarise(averagecity= mean(cty))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 7 x 2
##   class      averagecity
##   <chr>            <dbl>
## 1 2seater           15.4
## 2 compact           20.1
## 3 midsize           18.8
## 4 minivan           15.8
## 5 pickup            13
## 6 subcompact        20.4
## 7 suv               13.5
```