R ASSIGNMENT 4

NWUDO CHIKAEZE FIDELIS JUNIOR

2022-11-17

```
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse
1.3.2 ──
## ✓ ggplot2 3.4.0      ✓ purrr   0.3.5
## ✓ tibble  3.1.8      ✓ dplyr   1.0.10
## ✓ tidyr   1.2.1      ✓ stringr 1.4.1
## ✓ readr   2.1.3      ✓ forcats 0.5.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✕ dplyr::filter() masks stats::filter()
## ✕ dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
library(fueleconomy)
```

Question 1

Load the movies.csv from your directory. head(movies)

1. Plot the side-by-side histograms of the movie scores for the top three genres.

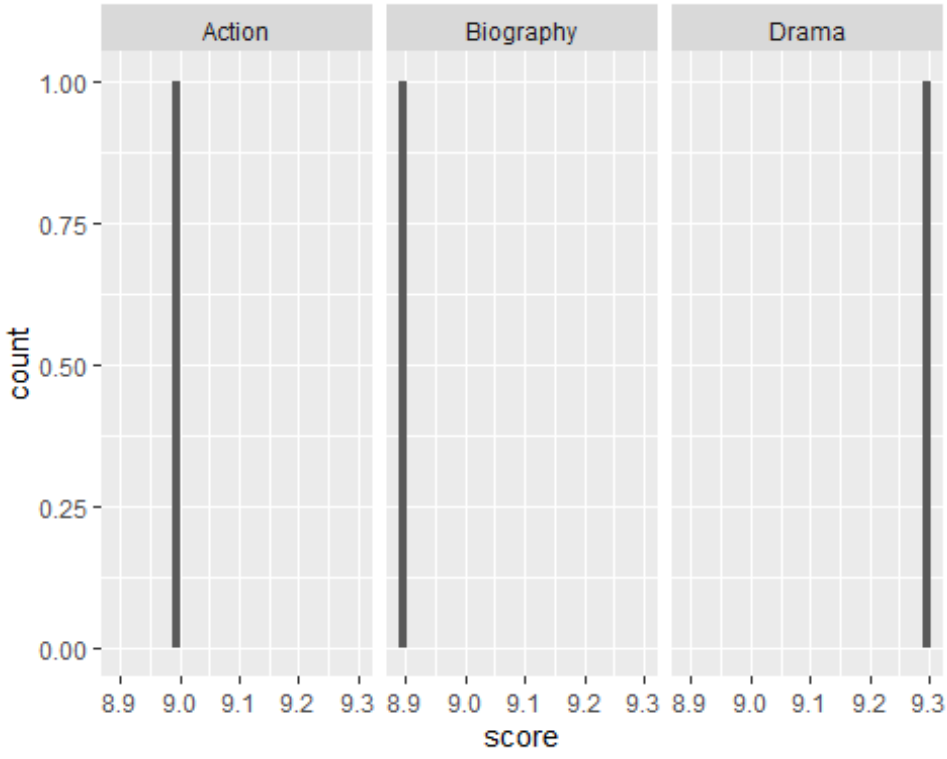
2. Plot the side-by-side boxplots of the movie scores for the top three genres.

QUESTION 1.1

```
movies <- read_csv("movies.csv")
```

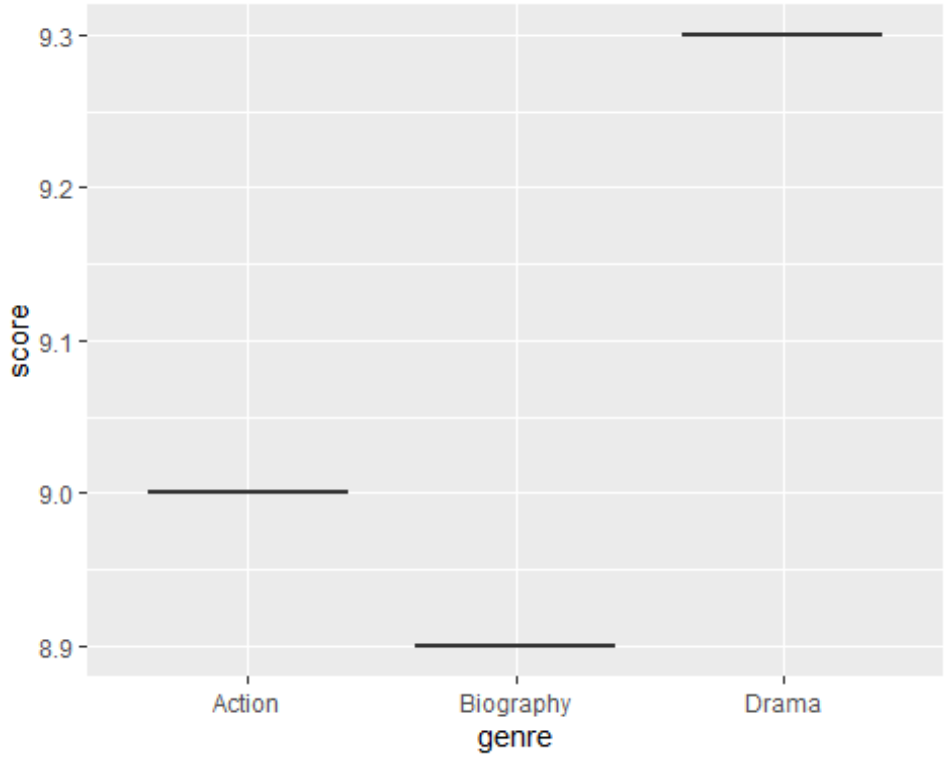
```
## Rows: 7668 Columns: 15
## ── Column specification ──────────────────────────────────────────────────────
## Delimiter: ","
## chr (9): name, rating, genre, released, director, writer, star, country,
com...
## dbl (6): year, score, votes, budget, gross, runtime
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

```
movies2 <- movies %>% group_by(genre) %>% select(genre,score) %>%
arrange(desc(score))
movies3 <- movies2[1:3,]
ggplot(data = movies3, mapping = aes(x=score)) + geom_histogram() +
facet_grid(~genre)

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



QUESTION 1.2

```
movies3 %>% ggplot(aes(x=genre, y=score)) + geom_boxplot()
```



Question 2

Load the ggplot2 and fueleconomy packages, as well as the vehicles dataset. Run the code below to extract just the first 1,000 rows of the dataset.

QUESTION 2.1

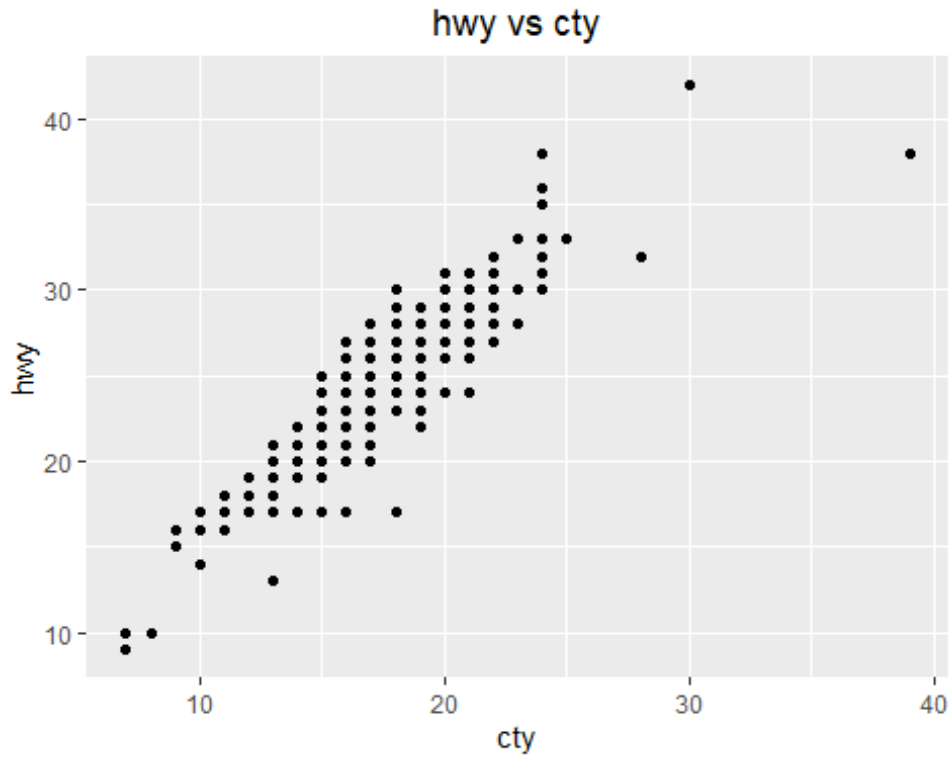
Make a scatterplot of hwy vs. cty. Give axis titles and a main title to the plot to make it more interpretable.

```
data("vehicles")
vehicle <- vehicles[1:1000,]
vehicle
```

```
## # A tibble: 1,000 × 12
##       id make  model        year class trans drive   cyl displ fuel    hwy
cty
##    <dbl> <chr> <chr>       <dbl> <chr> <chr> <chr> <dbl> <dbl> <chr> <dbl>
<dbl>
##  1 13309 Acura 2.2CL/3.0CL  1997 Subc… Auto… Fron…     4   2.2 Regu…    26
20
##  2 13310 Acura 2.2CL/3.0CL  1997 Subc… Manu… Fron…     4   2.2 Regu…    28
22
##  3 13311 Acura 2.2CL/3.0CL  1997 Subc… Auto… Fron…     6   3   Regu…    26
18
##  4 14038 Acura 2.3CL/3.0CL  1998 Subc… Auto… Fron…     4   2.3 Regu…    27
```

```
19
##  5 14039 Acura 2.3CL/3.0CL  1998 Subc… Manu… Fron…     4   2.3 Regu…    29
21
##  6 14040 Acura 2.3CL/3.0CL  1998 Subc… Auto… Fron…     6   3   Regu…    26
17
##  7 14834 Acura 2.3CL/3.0CL  1999 Subc… Auto… Fron…     4   2.3 Regu…    27
20
##  8 14835 Acura 2.3CL/3.0CL  1999 Subc… Manu… Fron…     4   2.3 Regu…    29
21
##  9 14836 Acura 2.3CL/3.0CL  1999 Subc… Auto… Fron…     6   3   Regu…    26
17
## 10 11789 Acura 2.5TL        1995 Comp… Auto… Fron…     5   2.5 Prem…    23
18
## # … with 990 more rows
```

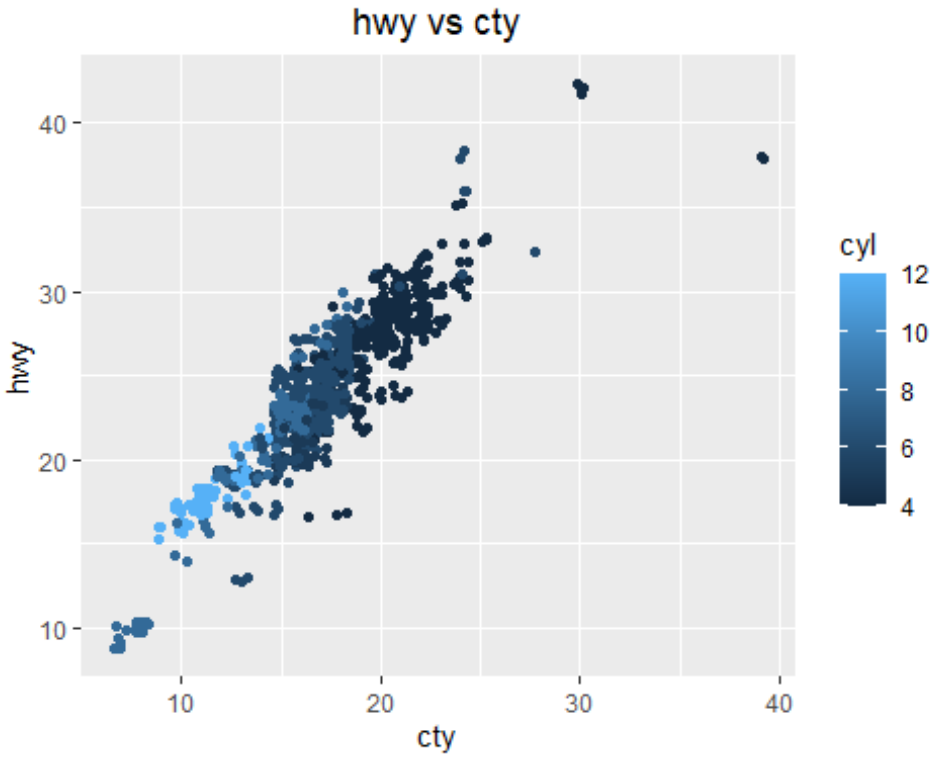
```
ggplot(data = vehicle, mapping = aes(x=cty, y=hwy)) + geom_point() +
ggtitle("hwy vs cty") + theme(plot.title = element_text(hjust = 0.5))
```



QUESTION 2.2

Modify the plot above such that the color of the dot represents cyl value. Also reduce the alpha of the points to an appropriate level and introduce jitter.

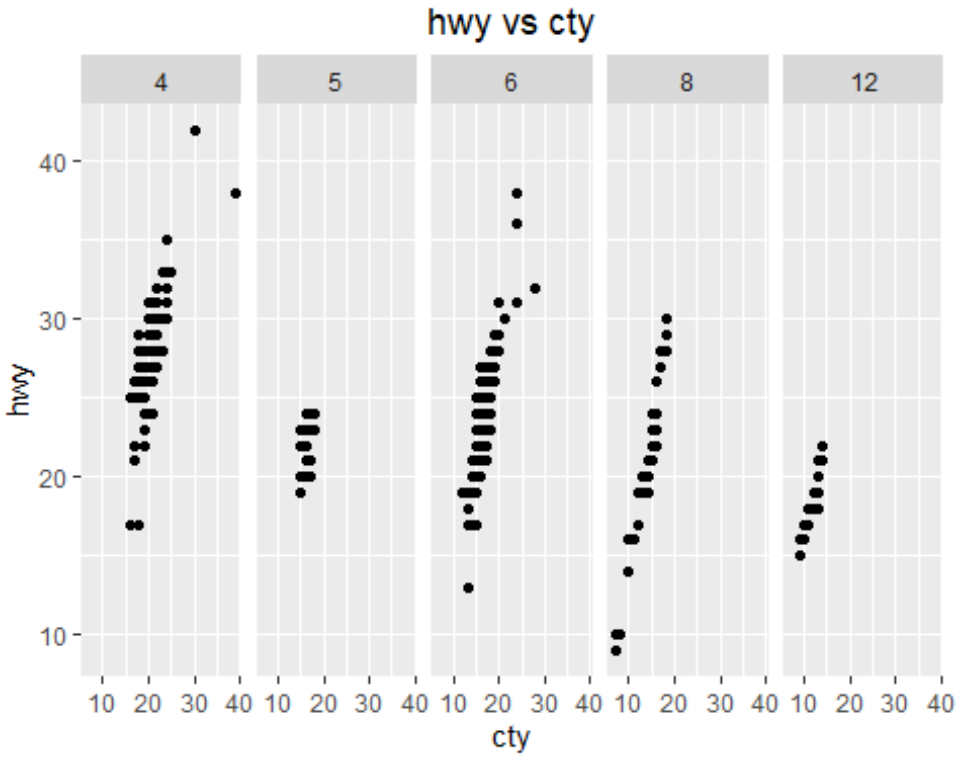
```
ggplot(data = vehicle, mapping = aes(x=cty, y=hwy)) + geom_point(aes(color =
cyl), position = "jitter") + ggtitle("hwy vs cty") + theme(plot.title =
element_text(hjust = 0.5))
```



QUESTION 2.3

Modify the plot above so that each value of cyl is in its own plot.

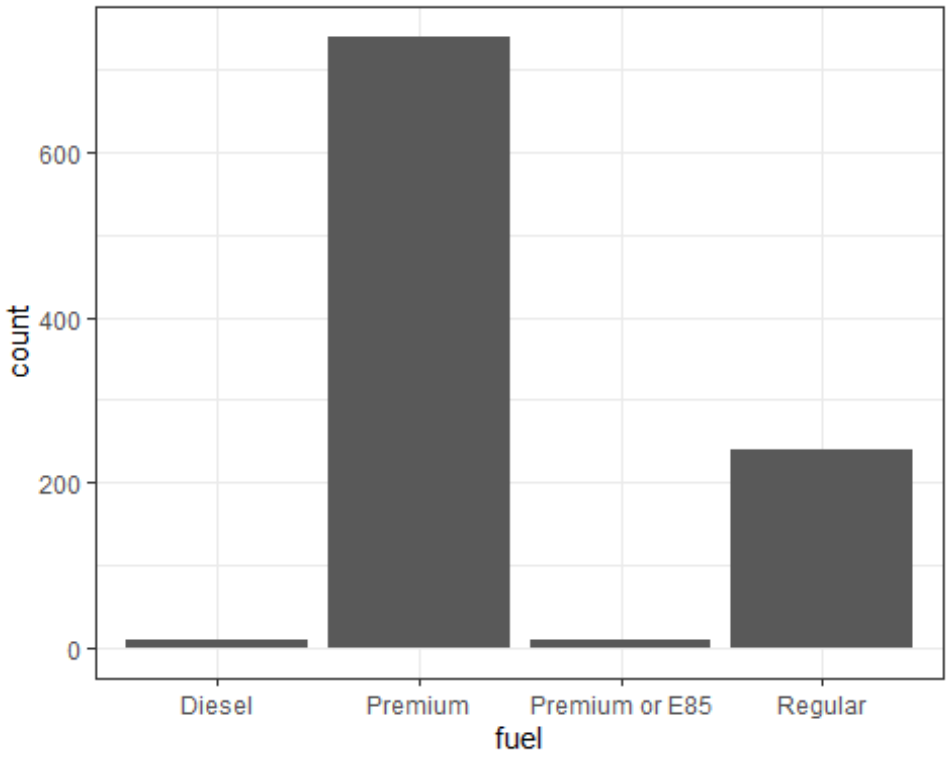
```
ggplot(data = vehicle, mapping = aes(x=cty, y=hwy)) + geom_point() +
ggtitle("hwy vs cty") + theme(plot.title = element_text(hjust = 0.5)) +
facet_grid(~ cyl)
```



QUESTION 2.4

Make a bar plot to show how many cars of each type of fuel there are in the dataset. (Use the geom_bar geom.) Change the theme to ggplot's black and white theme.

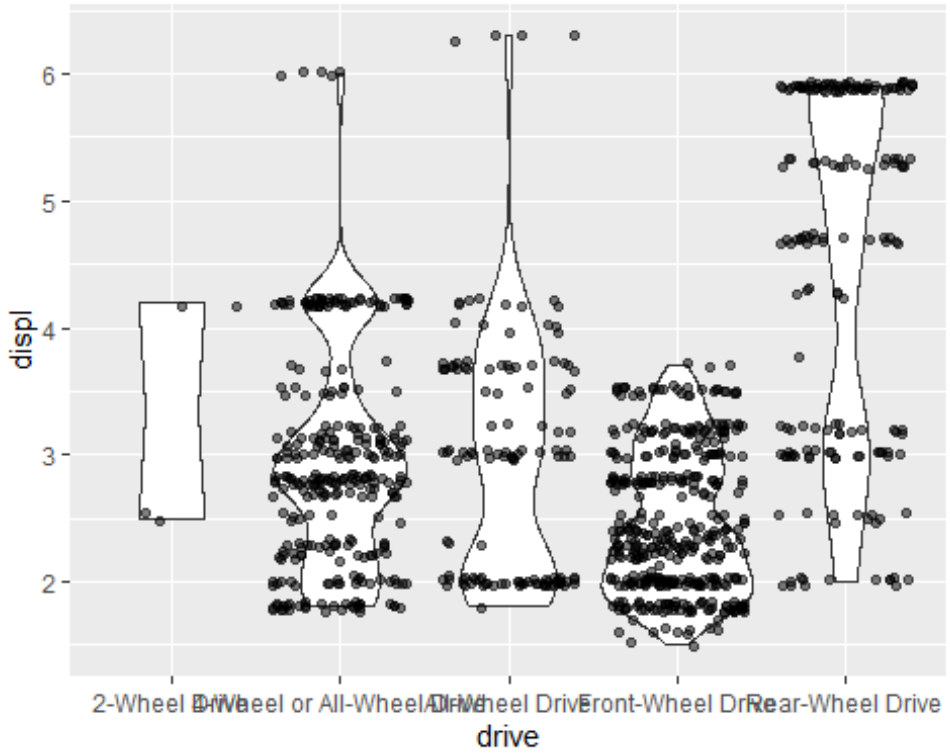
```
ggplot(data = vehicle, mapping = aes(x = fuel)) + geom_bar() + theme_bw()
```



QUESTION 2.5

Make a violin plot to show the distribution of displ for each value of drive. Overlay that with a scatterplot of displ vs. drive (with jitter and alpha). How does the scatterplot give the reader more information?

```
ggplot(data = vehicle, mapping = aes(x=drive, y=displ)) + geom_violin() +
geom_point(position = "jitter", alpha = 1/2)
```

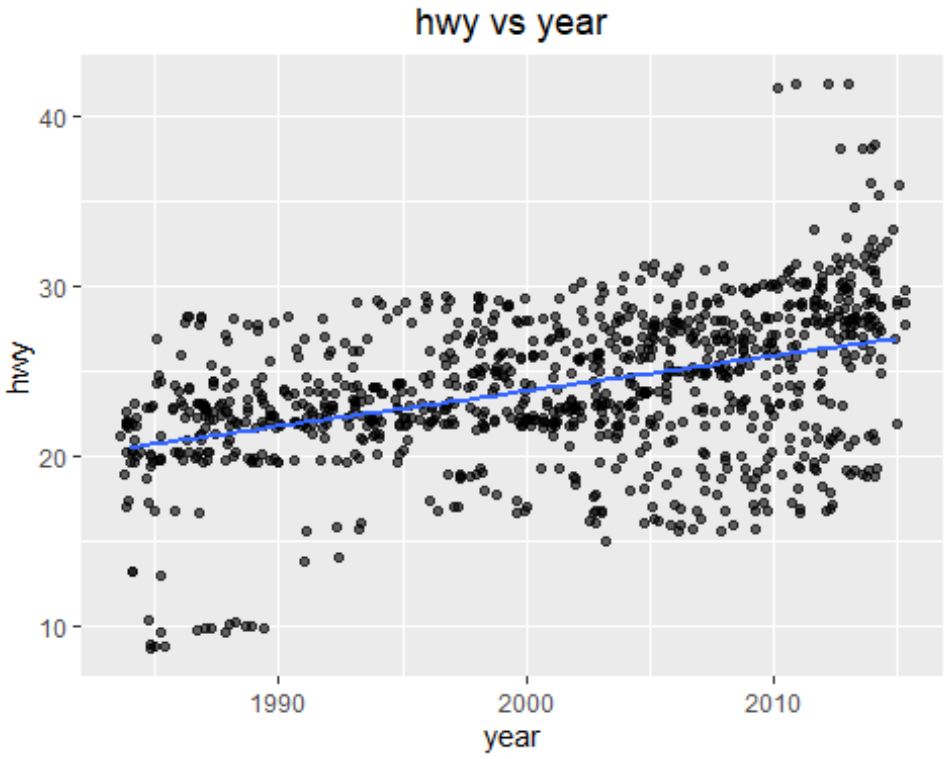


- Decreasing the alpha value, reduces the plots transparency.

QUESTION 2.6

Make a (jittered) scatterplot of hwy against year with alpha value 0.5. Add a geom_smooth layer with option method = “lm” and without the SE bands.

```
ggplot(data = vehicle, mapping = aes(x=year, y=hwy)) + geom_point(position =
"jitter", alpha = 0.6) + geom_smooth(method = "lm", se = FALSE) +
ggtitle("hwy vs year") + theme(plot.title = element_text(hjust = 0.5))

## `geom_smooth()` using formula = 'y ~ x'
```

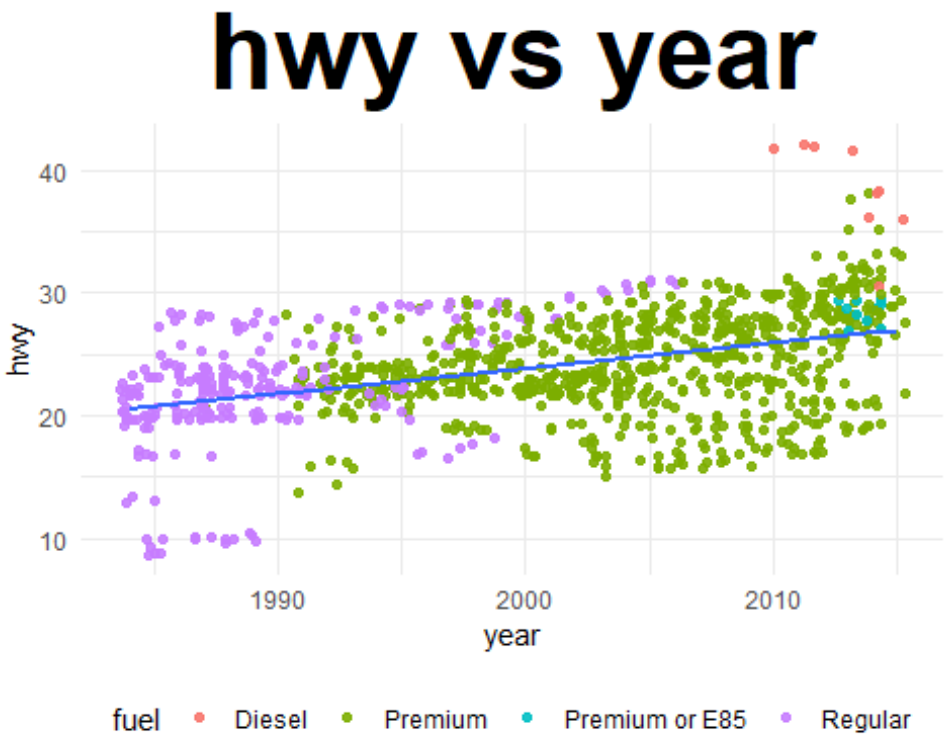


QUESTION 2.7

Modify the previous plot so that the color of the points depends on fuel. Also, change the theme to ggplot's minimal theme and move the legend to the bottom of the plot.

```
ggplot(data = vehicle, mapping = aes(x=year, y=hwy)) + geom_point(aes(color = fuel),position = "jitter", alpha = 0.9) + geom_smooth(method = "lm", se = FALSE) + theme_minimal() + theme(legend.position = "bottom") + ggtitle("hwy vs year") + theme(plot.title = element_text(hjust = 0.5, face="bold", size = 40))
```

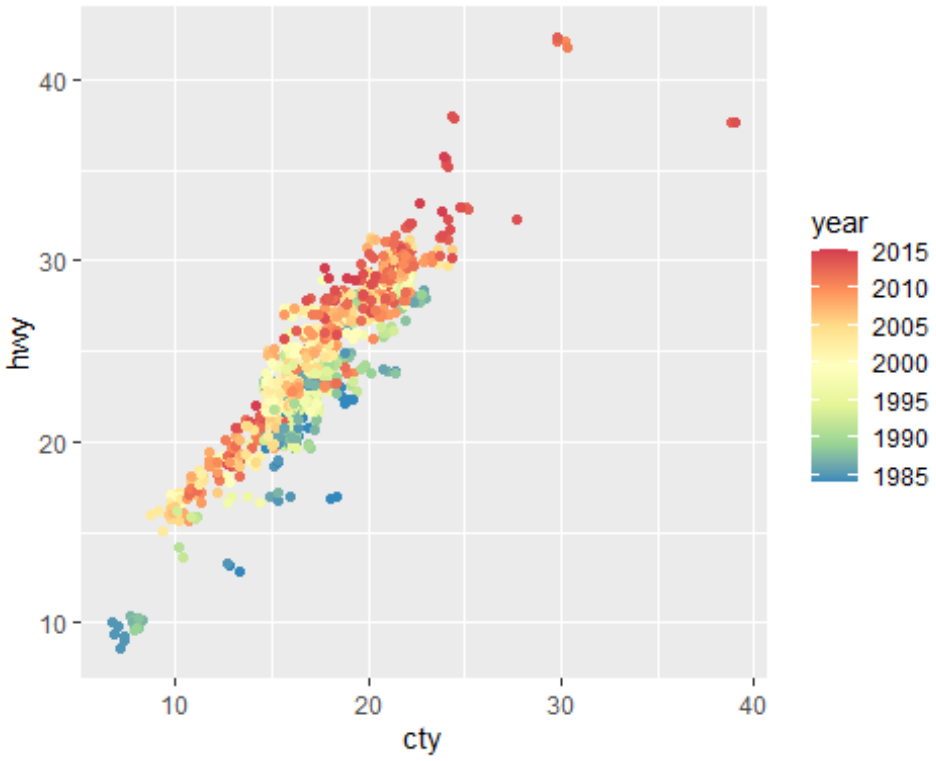
```
## `geom_smooth()` using formula = 'y ~ x'
```



QUESTION 2.8

Make a (jittered) scatterplot of hwy vs. cty, with the color of the point depending on year. Change the color scale to "Spectral". Do you see a trend?

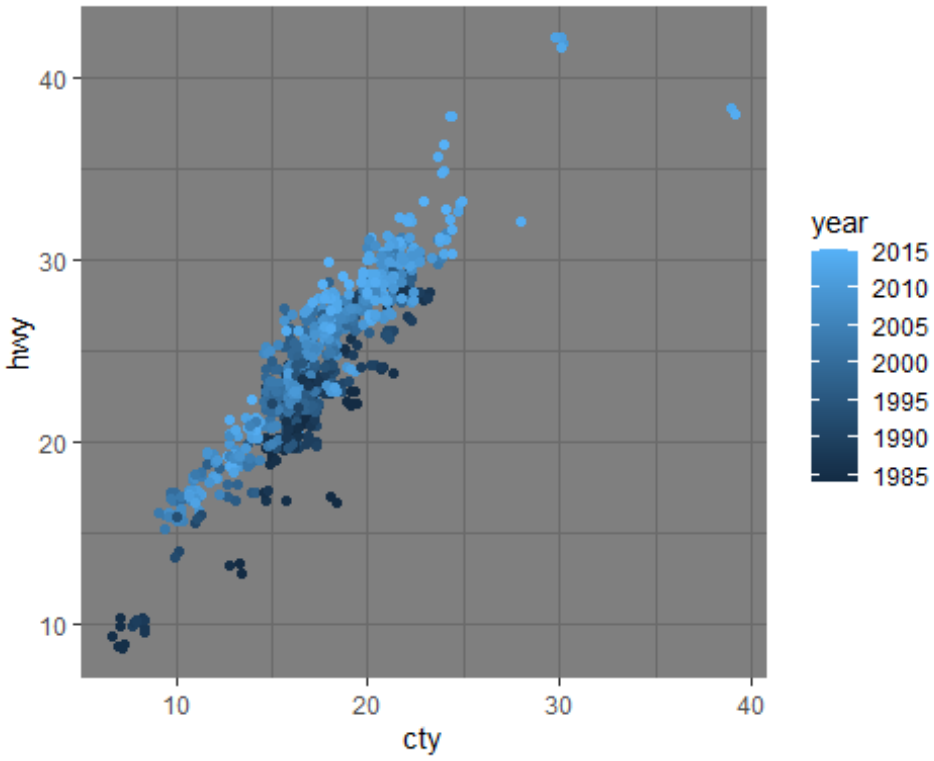
```
ggplot(data = vehicle, mapping = aes(x=cty, y=hwy)) + geom_point(aes(color =
year), position = "jitter") + scale_color_distiller(palette = "Spectral")
```



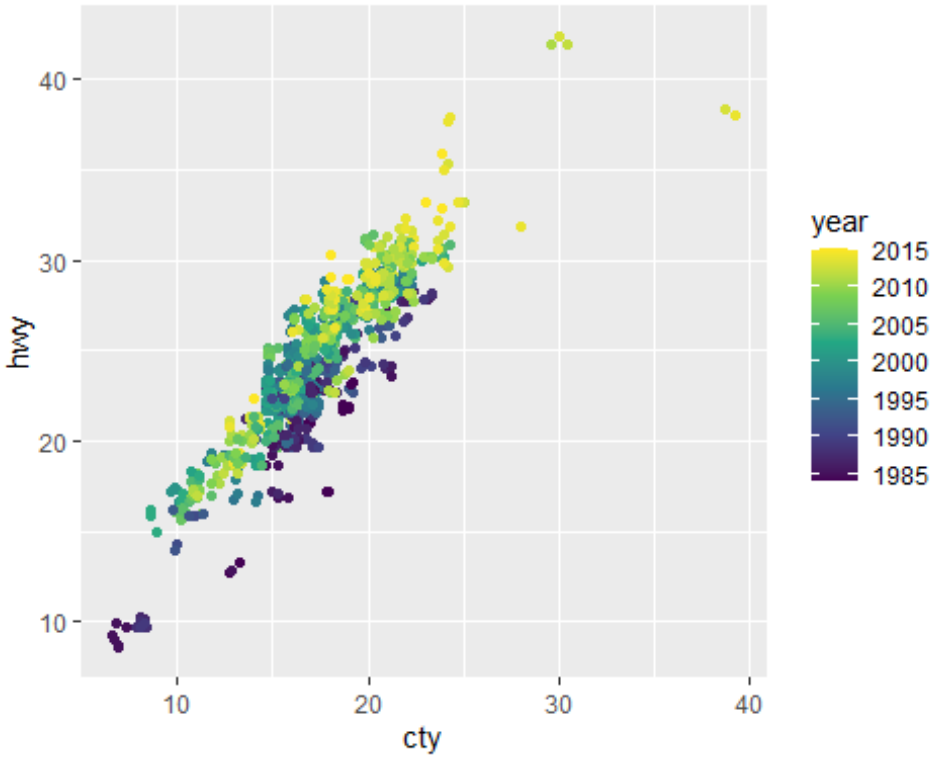
QUESTION 2.9

Modify the theme of the plot above to a theme you like and try a different color scale. Also, give the plot a title and make it bigger, bold and centralized.

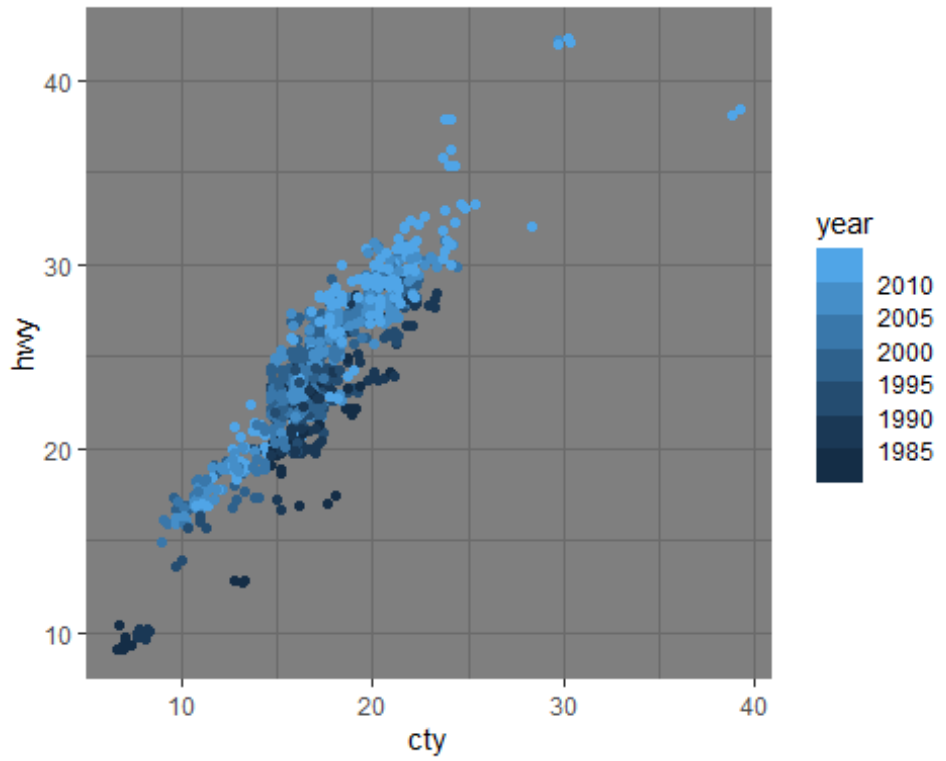
```
ggplot(data = vehicle, mapping = aes(x=cty, y=hwy)) + geom_point(aes(color = year), position = "jitter") + theme_dark() + scale_color_continuous(type = "gradient")
```



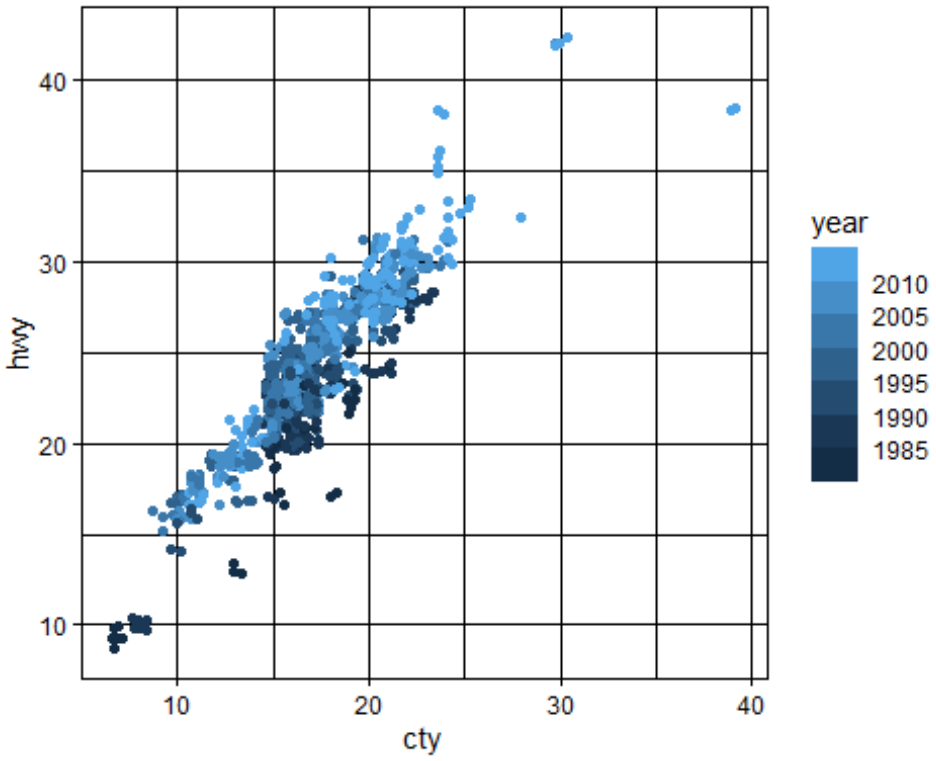

```
ggplot(data = vehicle, mapping = aes(x=cty, y=hwy)) + geom_point(aes(color = year), position = "jitter") + theme_gray() + scale_color_continuous(type = "viridis")
```



```
ggplot(data = vehicle, mapping = aes(x=cty, y=hwy)) + geom_point(aes(color =
year), position = "jitter") + theme_dark() + scale_color_binned(type =
"gradient")
```




```
ggplot(data = vehicle, mapping = aes(x=cty, y=hwy)) + geom_point(aes(color =
year), position = "jitter") + theme_linedraw() + scale_color_binned(type =
"gradient")
```



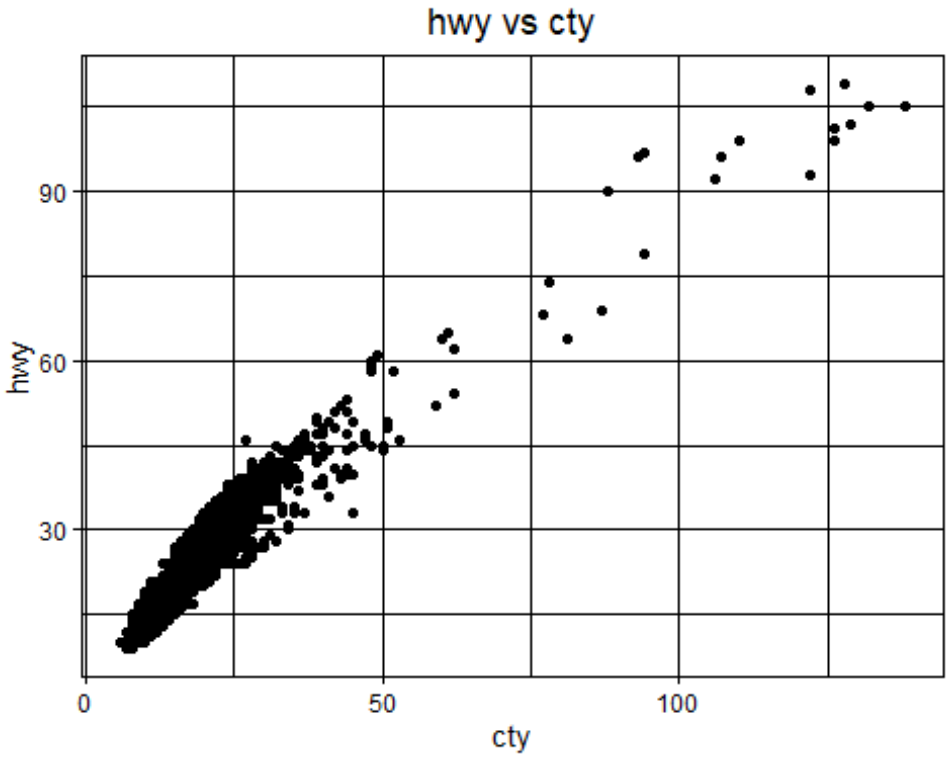
Question 3

Load the ggplot2 and fueleconomy packages, as well as the vehicles dataset.

QUESTION 3.1

Make a scatterplot of hwy vs. cty.

```
veh <- vehicles
ggplot(data = veh, mapping = aes(x=cty, y=hwy)) + geom_point() +
ggtitle(label = "hwy vs cty") + theme_linedraw() + theme(plot.title =
element_text(hjust = 0.5))
```



QUESTION 3.2

Convert the cyl column to a factor.

```
veh$cyl <- as.factor(veh$cyl)
veh
```

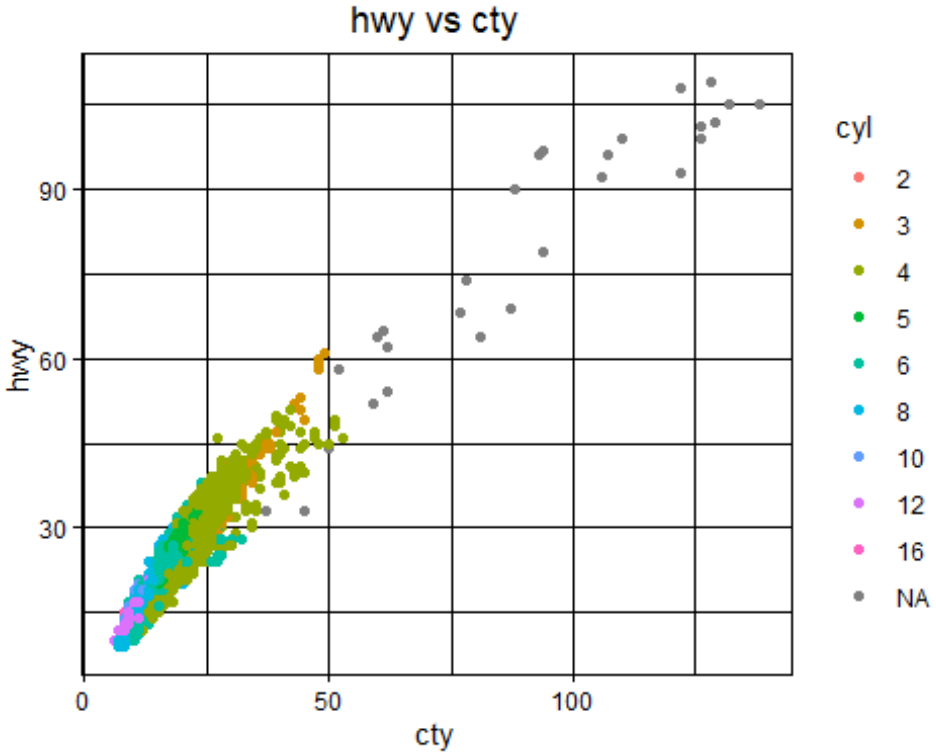
```
## # A tibble: 33,442 × 12
##       id make  model        year class trans drive cyl   displ fuel    hwy
cty
##    <dbl> <chr> <chr>       <dbl> <chr> <chr> <chr> <fct> <dbl> <chr> <dbl>
<dbl>
##  1 13309 Acura 2.2CL/3.0CL  1997 Subc… Auto… Fron… 4       2.2 Regu…    26
20
##  2 13310 Acura 2.2CL/3.0CL  1997 Subc… Manu… Fron… 4       2.2 Regu…    28
22
##  3 13311 Acura 2.2CL/3.0CL  1997 Subc… Auto… Fron… 6       3   Regu…    26
18
##  4 14038 Acura 2.3CL/3.0CL  1998 Subc… Auto… Fron… 4       2.3 Regu…    27
19
##  5 14039 Acura 2.3CL/3.0CL  1998 Subc… Manu… Fron… 4       2.3 Regu…    29
21
##  6 14040 Acura 2.3CL/3.0CL  1998 Subc… Auto… Fron… 6       3   Regu…    26
17
##  7 14834 Acura 2.3CL/3.0CL  1999 Subc… Auto… Fron… 4       2.3 Regu…    27
20
##  8 14835 Acura 2.3CL/3.0CL  1999 Subc… Manu… Fron… 4       2.3 Regu…    29
```

```
21
##  9 14836 Acura 2.3CL/3.0CL  1999 Subc… Auto… Fron… 6       3    Regu…    26
17
## 10 11789 Acura 2.5TL        1995 Comp… Auto… Fron… 5       2.5 Prem…    23
18
## # … with 33,432 more rows
```

QUESTION 3.3

Modify the plot from part (1) such that the color of the dot represents cyl value.

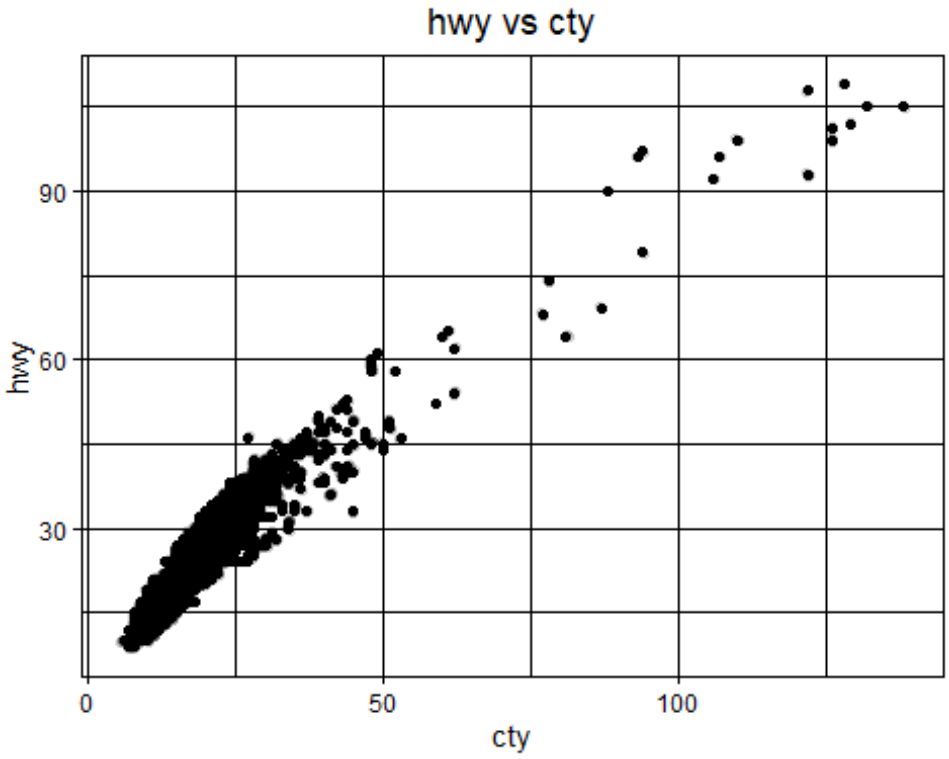
```
ggplot(data = veh, mapping = aes(x=cty, y=hwy)) + geom_point(aes(color =
cyl)) + theme_linedraw() + ggtitle(label = "hwy vs cty") + theme(plot.title =
element_text(hjust = 0.5))
```



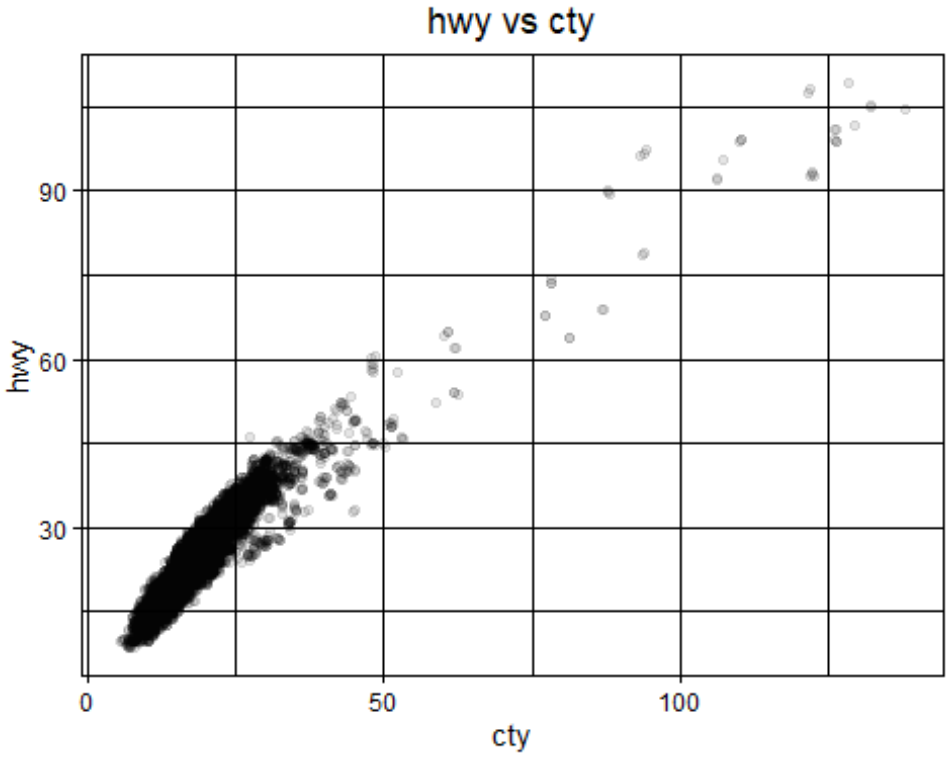
QUESTION 3.4

There is a lot of overplotting in the plot above. Remove the color scale and modify the previous plot so that alpha = 0.1

```
ggplot(data = veh, mapping = aes(x=cty, y=hwy)) + geom_point() +
geom_jitter(alpha = 0.1) + theme_linedraw() + ggtitle(label = "hwy vs cty") +
theme(plot.title = element_text(hjust = 0.5))
```



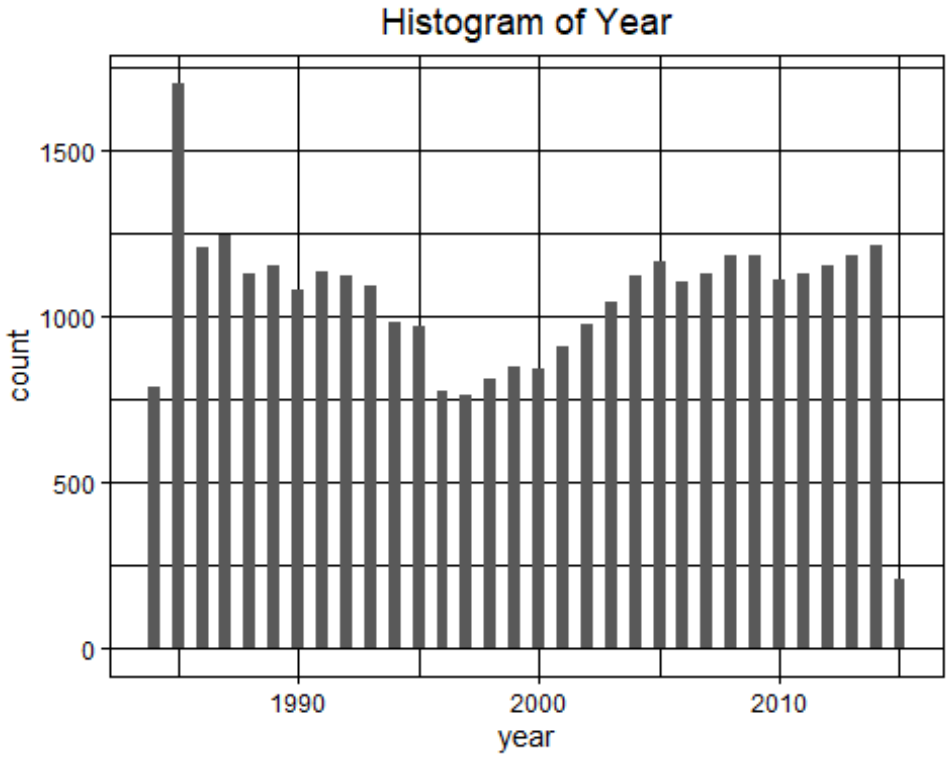

```
ggplot(data = veh, mapping = aes(x=cty, y=hwy)) + geom_point( alpha = 0.1,
position = "jitter") + theme_linedraw() + ggtitle(label = "hwy vs cty") +
theme(plot.title = element_text(hjust = 0.5))
```



QUESTION 3.5

Make a histogram of year.

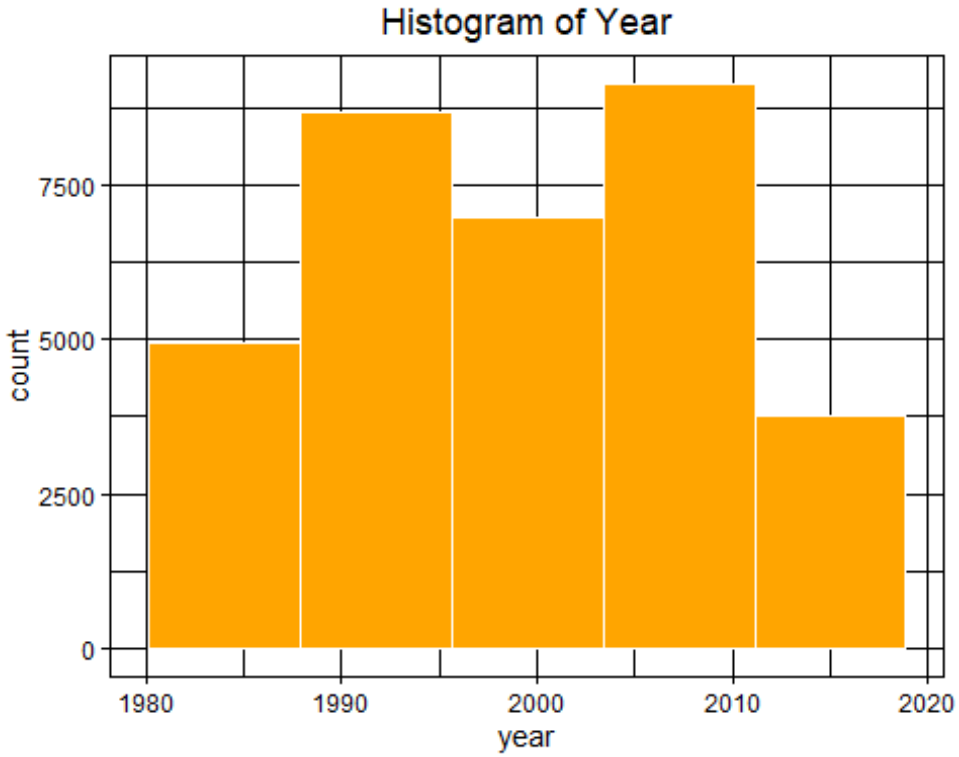
```
ggplot(data = veh, mapping = aes(x=year)) + geom_histogram(binwidth = 0.5) +
theme_linedraw() + ggtitle(label = "Histogram of Year") + theme(plot.title =
element_text(hjust = 0.5))
```



QUESTION 3.6

Make a histogram of year with just 5 bins.

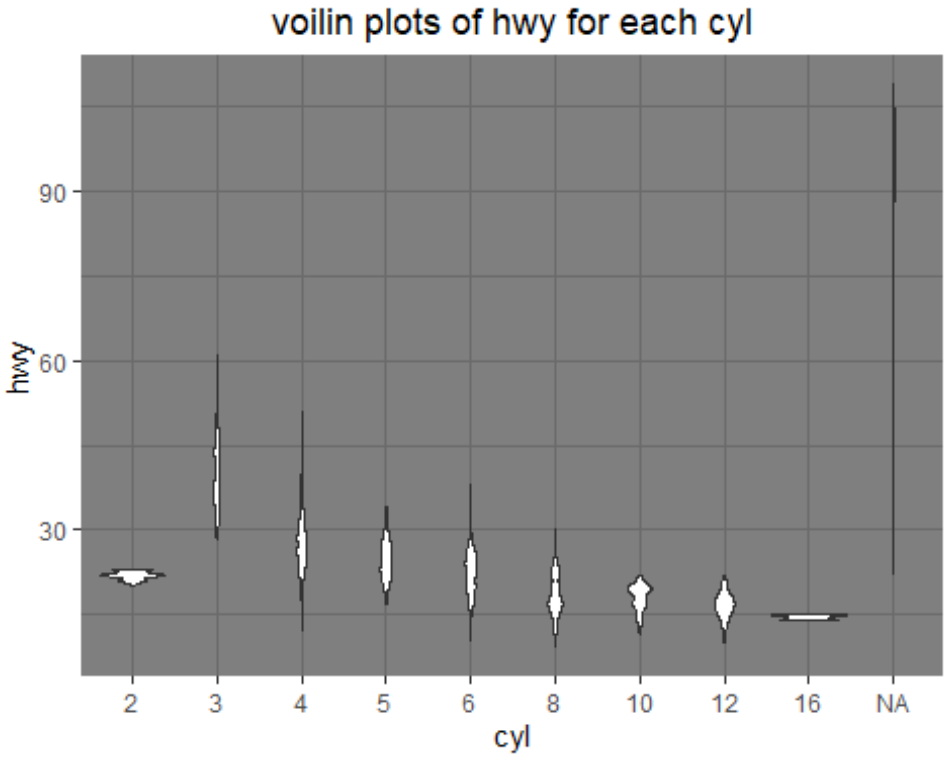
```
ggplot(data = veh, mapping = aes(x=year)) + geom_histogram(bins = 5, color =
"white", fill = "orange") + theme_linedraw() + ggtitle(label = "Histogram of
Year") + theme(plot.title = element_text(hjust = 0.5))
```



QUESTION 3.7

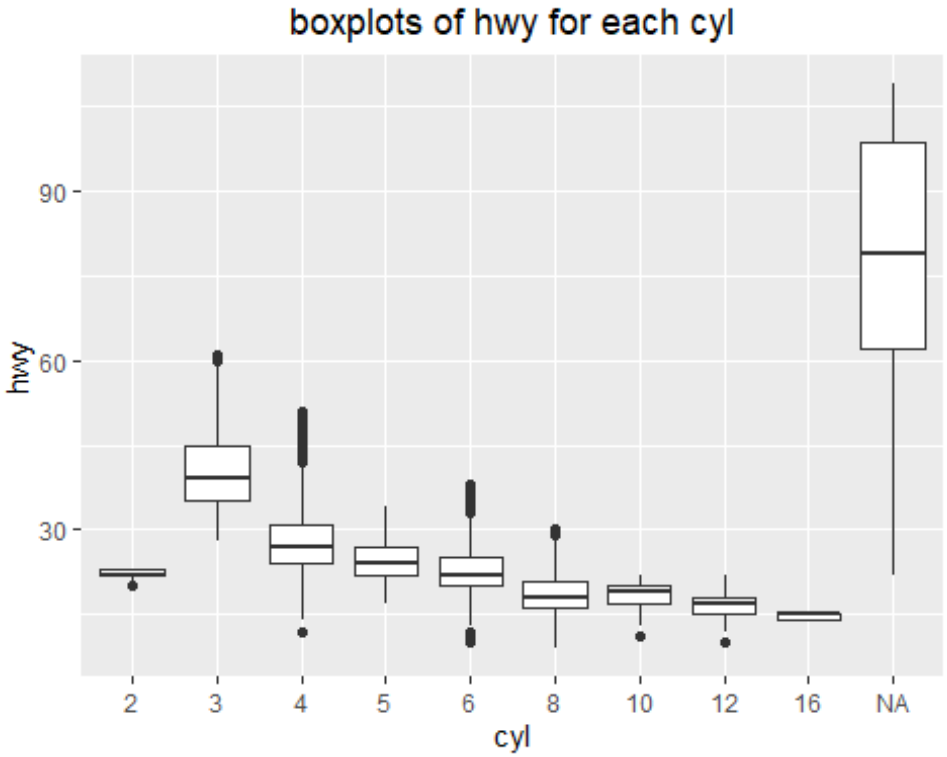
For each value of cyl, make a violin plot of hwy values.

```
ggplot(data = veh, mapping = aes(x=cyl, y=hwy)) + geom_violin(fill = "white")
+ theme_dark() + ggtitle(label = "voilin plots of hwy for each cyl") +
theme(plot.title = element_text(hjust = 0.5))
```



QUESTION 3.8 For each value of cyl, make a boxplot of hwy values.

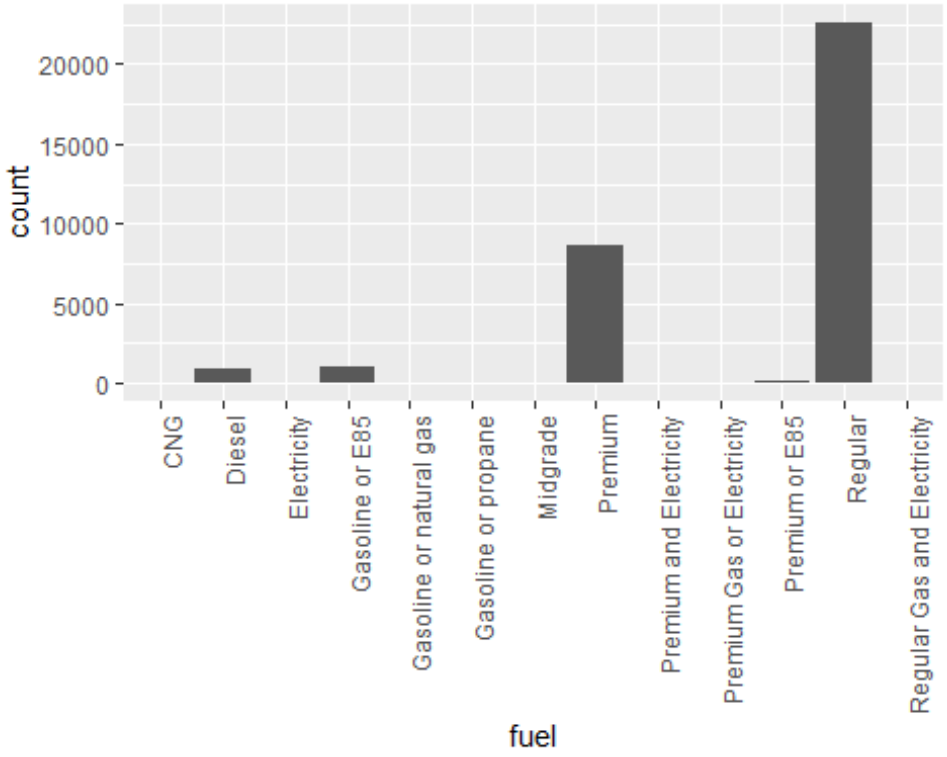
```
ggplot(data = veh, mapping = aes(x=cyl, y=hwy)) + geom_boxplot(fill =
"white") + ggtitle(label = "boxplots of hwy for each cyl") + theme(plot.title
= element_text(hjust = 0.5))
```



QUESTION 3.9

Make a barplot to show how many cars of each type of fuel there are in the dataset. (Hint: Use the geom_bar geom.)

```
ggplot(data = veh, mapping = aes(x=fuel)) + geom_bar() + theme(axis.text.x =
element_text(angle = 90, hjust = 1))
```

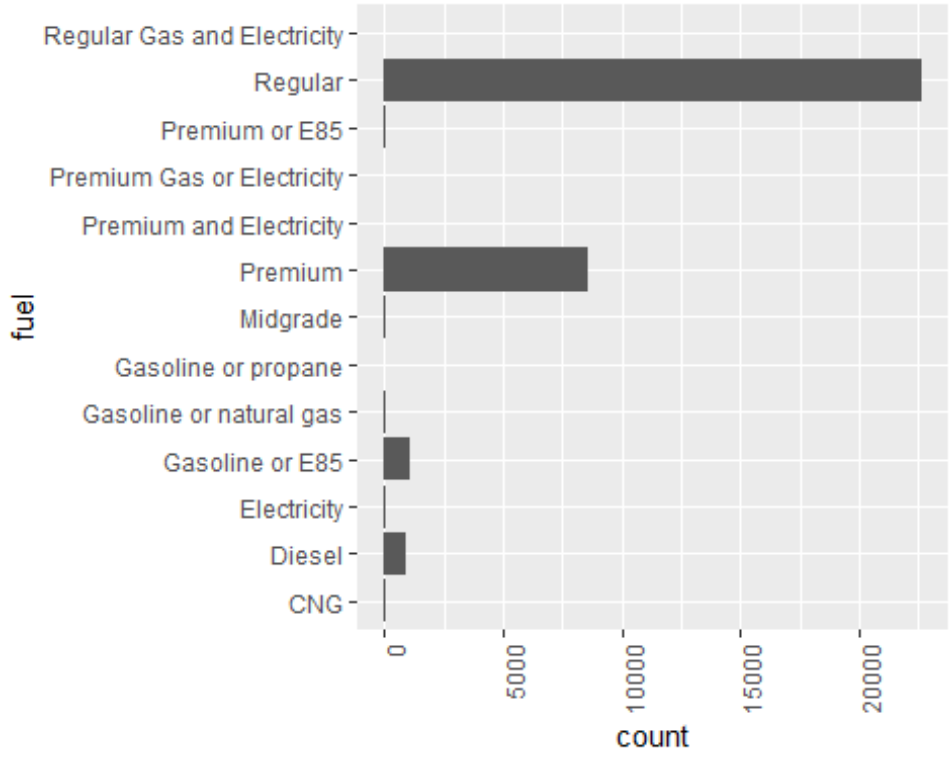



```
# to view values for the very low bars, you can adjust the scaling, and input
into coord_cartesian(ylim= c(0,60))
```

QUESTION 3.10

Add a coord_flip() layer to the previous plot to make a horizontal barplot.

```
ggplot(data = veh, mapping = aes(x=fuel)) + geom_bar() + theme(axis.text.x =
element_text(angle = 90, hjust = 1)) + coord_flip()
```



Question 4

Load the life expectancy dataset

QUESTION 4.1

Filter the data for the Americas in 2007, deselect all other variables.

```
gapminder <- read_csv("gapminder.csv")
```

```
## Rows: 1704 Columns: 6
## ── Column specification
──────────────────────────────────────────────────────
## Delimiter: ","
## chr (2): country, continent
## dbl (4): year, pop, lifeExp, gdpPercap
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

```
gapminder
```

```
## # A tibble: 1,704 × 6
##    country      year      pop continent lifeExp gdpPercap
```

```
##    <chr>       <dbl>    <dbl> <chr>       <dbl>     <dbl>
##  1 Afghanistan  1952  8425333 Asia         28.8      779.
##  2 Afghanistan  1957  9240934 Asia         30.3      821.
##  3 Afghanistan  1962 10267083 Asia         32.0      853.
##  4 Afghanistan  1967 11537966 Asia         34.0      836.
##  5 Afghanistan  1972 13079460 Asia         36.1      740.
##  6 Afghanistan  1977 14880372 Asia         38.4      786.
##  7 Afghanistan  1982 12881816 Asia         39.9      978.
##  8 Afghanistan  1987 13867957 Asia         40.8      852.
##  9 Afghanistan  1992 16317921 Asia         41.7      649.
## 10 Afghanistan  1997 22227415 Asia         41.8      635.
## # … with 1,694 more rows
```

```
gapminder %>% filter(continent == "Americas" , year == 2007) %>%
select(continent, year)
```

```
## # A tibble: 25 × 2
##    continent  year
##    <chr>     <dbl>
##  1 Americas   2007
##  2 Americas   2007
##  3 Americas   2007
##  4 Americas   2007
##  5 Americas   2007
##  6 Americas   2007
##  7 Americas   2007
##  8 Americas   2007
##  9 Americas   2007
## 10 Americas   2007
## # … with 15 more rows
```

QUESTION 4.2

Create the variable gdp, defined as the product of population size and gdp per person.

```
p1 <- gapminder %>% mutate(gdp = pop * gdpPercap)
p1
```

```
## # A tibble: 1,704 × 7
##    country      year      pop continent lifeExp gdpPercap          gdp
##    <chr>       <dbl>    <dbl> <chr>       <dbl>     <dbl>        <dbl>
##  1 Afghanistan  1952  8425333 Asia         28.8      779.  6567086330.
##  2 Afghanistan  1957  9240934 Asia         30.3      821.  7585448670.
##  3 Afghanistan  1962 10267083 Asia         32.0      853.  8758855797.
##  4 Afghanistan  1967 11537966 Asia         34.0      836.  9648014150.
##  5 Afghanistan  1972 13079460 Asia         36.1      740.  9678553274.
##  6 Afghanistan  1977 14880372 Asia         38.4      786. 11697659231.
##  7 Afghanistan  1982 12881816 Asia         39.9      978. 12598563401.
##  8 Afghanistan  1987 13867957 Asia         40.8      852. 11820990309.
##  9 Afghanistan  1992 16317921 Asia         41.7      649. 10595901589.
```

```
## 10 Afghanistan  1997 22227415 Asia         41.8      635. 14121995875.
## # … with 1,694 more rows
```

QUESTION 4.3

Identify the observation with lowest gdp per person.

```
print(arrange(p1, gdpPercap))
```

```
## # A tibble: 1,704 × 7
##    country          year      pop continent lifeExp gdpPercap          gdp
##    <chr>           <dbl>    <dbl> <chr>       <dbl>     <dbl>        <dbl>
##  1 Congo Dem. Rep.  2002 55379852 Africa       45.0      241. 13355730548.
##  2 Congo Dem. Rep.  2007 64606759 Africa       46.5      278. 17931726045.
##  3 Lesotho          1952   748747 Africa       42.1      299.   223760205.
##  4 Guinea-Bissau    1952   580653 Africa       32.5      300.   174108987.
##  5 Congo Dem. Rep.  1997 47798986 Africa       42.6      312. 14922290060.
##  6 Eritrea          1952  1438760 Africa       35.9      329.   473266516.
##  7 Myanmar          1952 20092996 Asia         36.3      331   6650781676
##  8 Lesotho          1957   813338 Africa       45.0      336.   273279222.
##  9 Burundi          1952  2445618 Africa       39.0      339.   829789527.
## 10 Eritrea          1957  1542611 Africa       38.0      344.   530907911.
## # … with 1,694 more rows
```

QUESTION 4.4

Identify all observations with above average life expectancy, stratified for each continent.

```
by_continent <- p1 %>% group_by(continent)
ses <- summarize(
  by_continent,
  averagelifeExp =  mean(lifeExp)
)
ses
```

```
## # A tibble: 5 × 2
##   continent averagelifeExp
##   <chr>              <dbl>
## 1 Africa              48.9
## 2 Americas            64.7
## 3 Asia                60.1
## 4 Europe              71.9
## 5 Oceania             74.3
```

QUESTION 4.6

Compute the mean life expectancy (the grand mean; i.e., across all observations).

```
gapminder %>% summarize(meanlifeexp = mean(lifeExp))
```

```
## # A tibble: 1 × 1
##   meanlifeexp
##         <dbl>
## 1        59.5
```

QUESTION 4.7

Compute the mean life expectancy for each year.

```
by_year <- gapminder %>% group_by(year)
sum_meanlifeexp <- summarize(by_year,
                 mle = mean(lifeExp))
sum_meanlifeexp
```

```
## # A tibble: 12 × 2
##     year   mle
##    <dbl> <dbl>
##  1  1952  49.1
##  2  1957  51.5
##  3  1962  53.6
##  4  1967  55.7
##  5  1972  57.6
##  6  1977  59.6
##  7  1982  61.5
##  8  1987  63.2
##  9  1992  64.2
## 10  1997  65.0
## 11  2002  65.7
## 12  2007  67.0
```

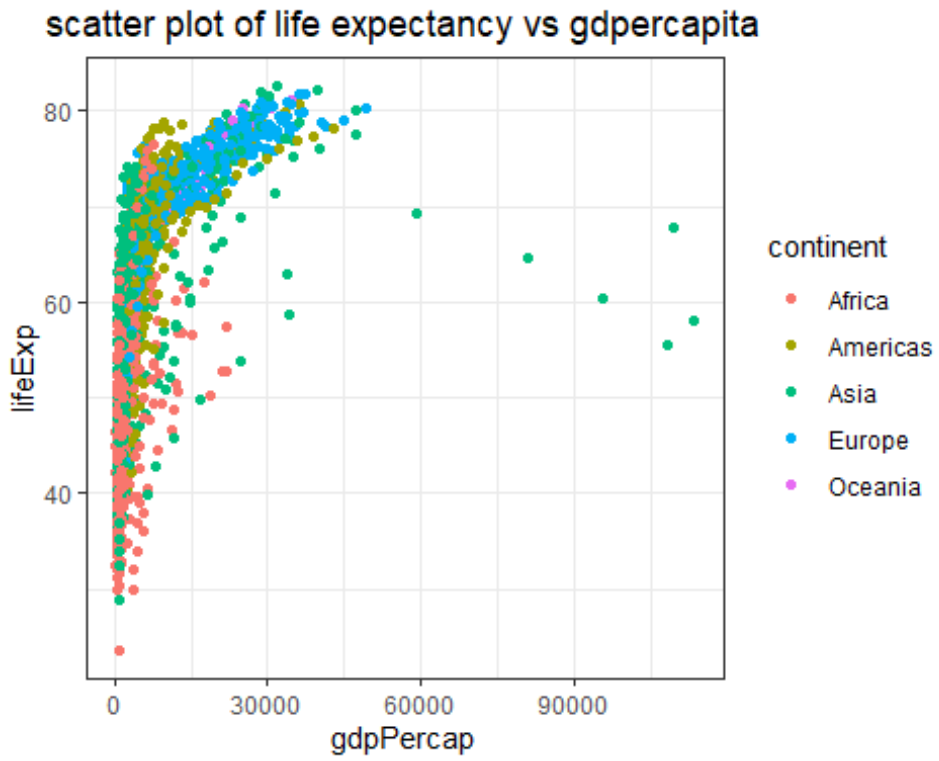
Question 5

Consider again the life expectancy dataset

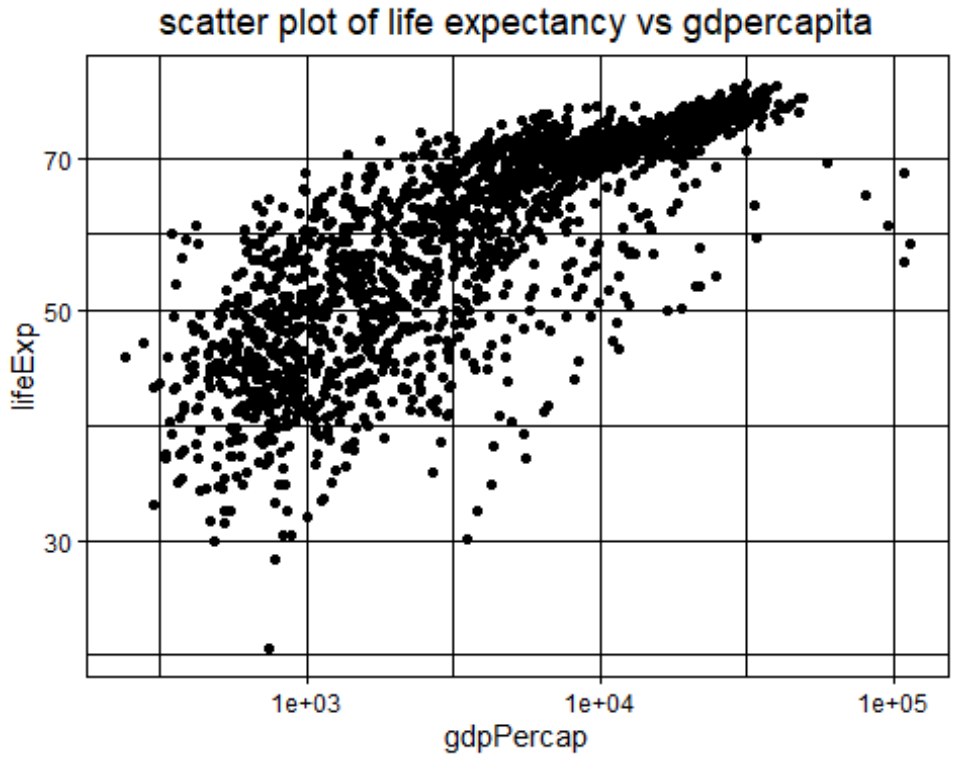
QUESTION 5.1

Create a scatter plot showing the association of gdp per person and life expectancy. Put the putative cause on the X axis and the putative effect on the y axis.

```
ggplot(data = gapminder, mapping = aes(x=gdpPercap,y=lifeExp)) +
geom_point(aes(color = continent)) + theme_bw() + ggtitle(label = "scatter
plot of life expectancy vs gdpercapita") + theme(plot.title =
element_text(hjust = 0.5))
```



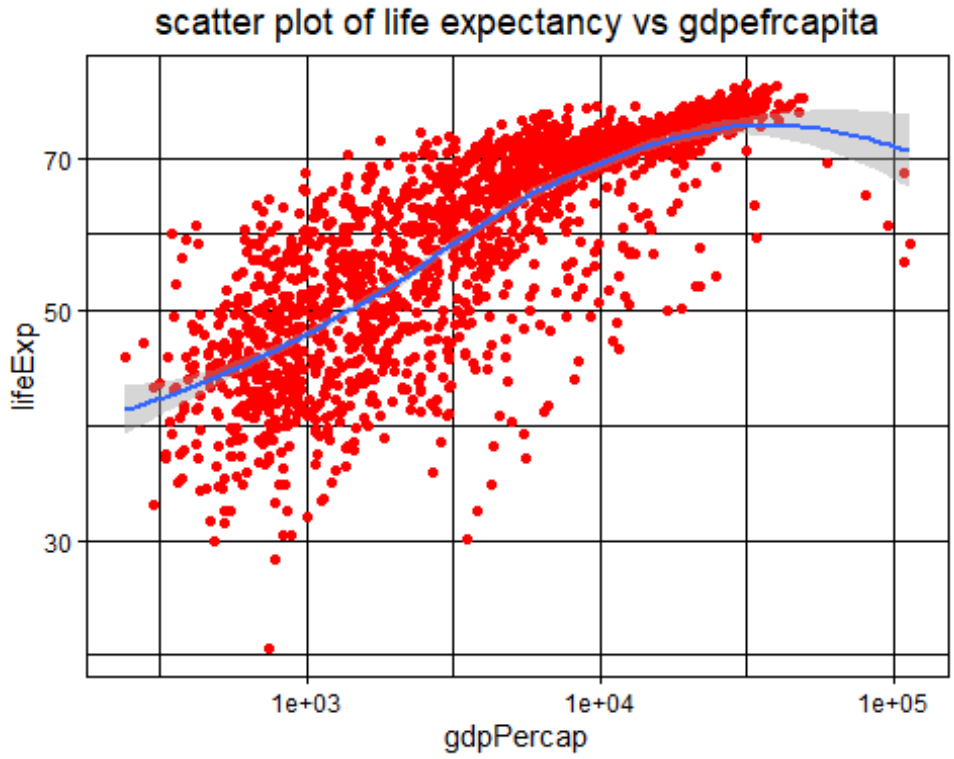

```
# carrying out log transformation on both x and y axes

ggplot(data = gapminder, mapping = aes(x=gdpPercap,y=lifeExp)) + geom_point()
+ theme_linedraw() + ggtitle(label = "scatter plot of life expectancy vs
gdpercapita") + theme(plot.title = element_text(hjust = 0.5)) +
scale_x_log10() + scale_y_log10()
```



QUESTION 5.2

Add a rolling average line (also known as LOESS smoother).

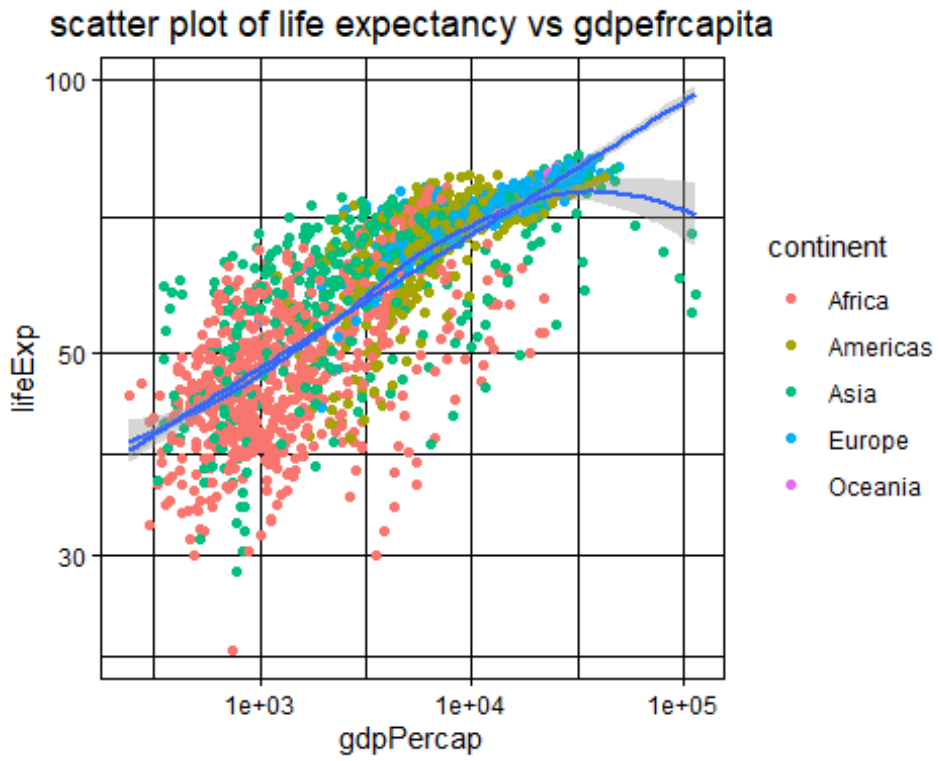
```
ggplot(data = gapminder, mapping = aes(x=gdpPercap,y=lifeExp)) +
geom_point(color = "red") + theme_linedraw() + ggtitle(label = "scatter plot
of life expectancy vs gdpefrcapita") + theme(plot.title = element_text(hjust
= 0.5)) + scale_x_log10() + scale_y_log10() + geom_smooth(method = "loess")

## `geom_smooth()` using formula = 'y ~ x'
```



QUESTION 5.3

Add a linear model line.

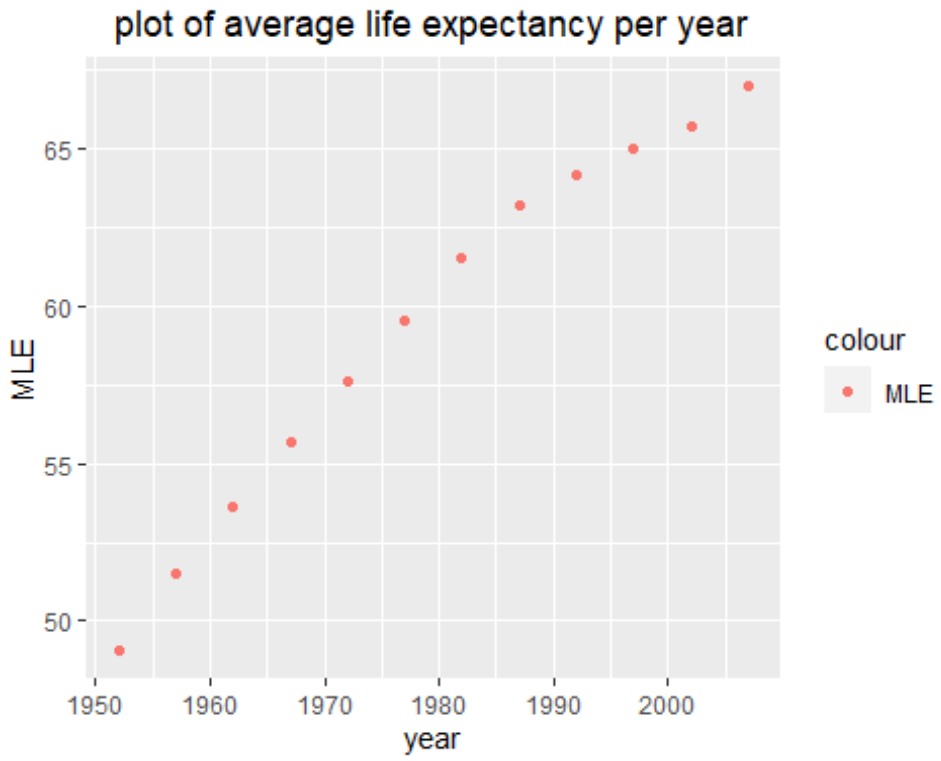
```
ggplot(data = gapminder, mapping = aes(x=gdpPercap,y=lifeExp)) +
geom_point(aes(color = continent)) + theme_linedraw() + ggtitle(label =
"scatter plot of life expectancy vs gdpefrcapita") + theme(plot.title =
element_text(hjust = 0.5)) + scale_x_log10() + scale_y_log10() +
geom_smooth(method = "loess") + geom_smooth(method = "lm")

## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
```



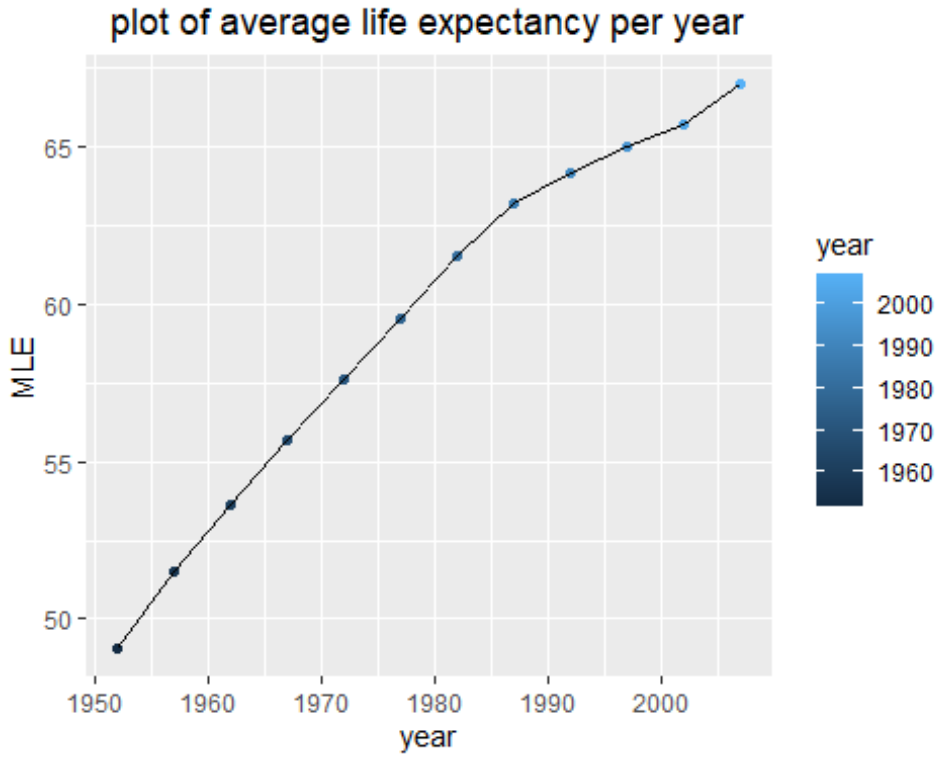
QUESTION 5.4

Create a scatter plot with year on the x axis, and life expectancy on the y axis. Each point should indicate the average life expectancy per year. Connect the dots with a line.

```
by_year <- gapminder %>% group_by(year) %>%  summarise( MLE = mean(lifeExp))
ggplot(data = by_year, mapping = aes(x=year, y=MLE)) + geom_point(aes(color =
"MLE")) + ggtitle(label = "plot of average life expectancy per year") +
theme(plot.title = element_text(hjust = 0.5))
```



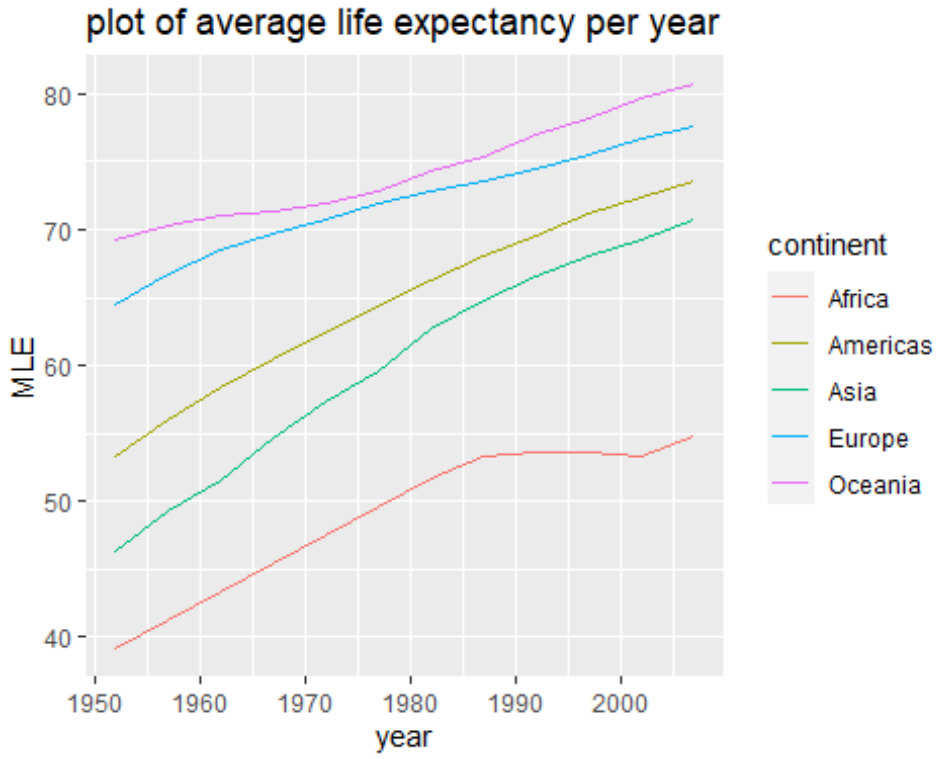

```
by_year <- gapminder %>% group_by(year) %>%  summarise( MLE = mean(lifeExp))
ggplot(data = by_year, mapping = aes(x=year, y=MLE)) + geom_point(aes(color =
year)) + geom_line() + ggtitle(label = "plot of average life expectancy per
year") + theme(plot.title = element_text(hjust = 0.5))
```



QUESTION 5.5

Modify the last plot so that there is a line for each continent (ie., group by continent).

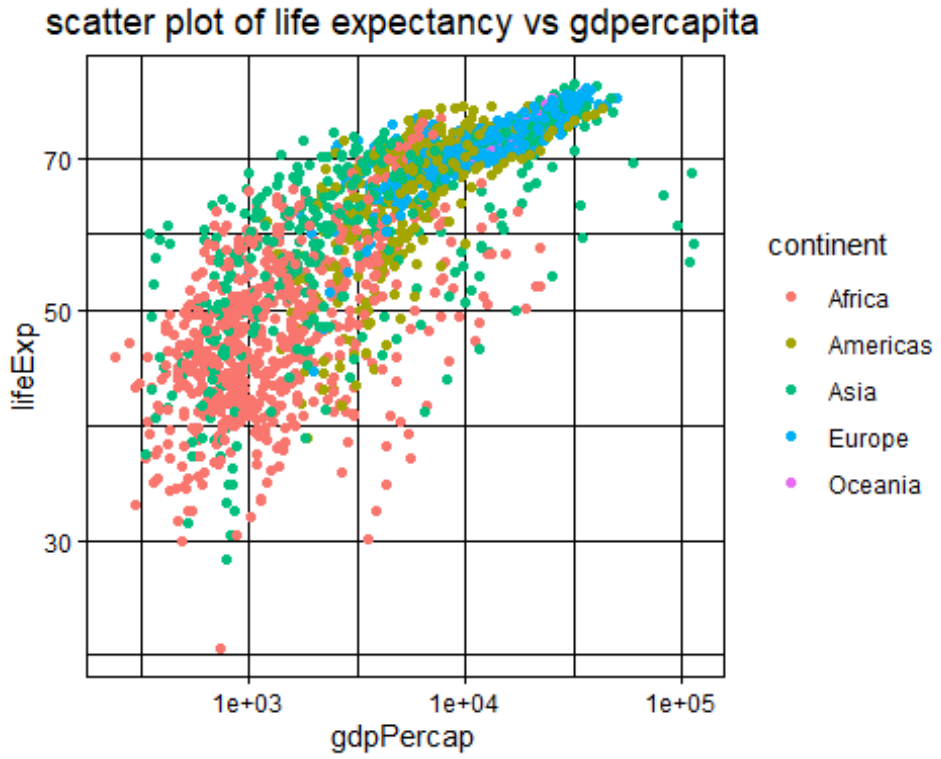
```
by_year <- gapminder %>% group_by(year, continent) %>%  summarise( MLE =
mean(lifeExp))
ggplot(data = by_year, mapping = aes(x=year, y=MLE)) + geom_line(aes(color =
continent))+ ggtitle(label = "plot of average life expectancy per year") +
theme(plot.title = element_text(hjust = 0.5))
```



QUESTION 5.6

Create a scatter plot showing the association of gdp per person and life expectancy. Put the putative cause on the X axis and the putative effect on the y axis. The color of the dots should map to the respective continent.

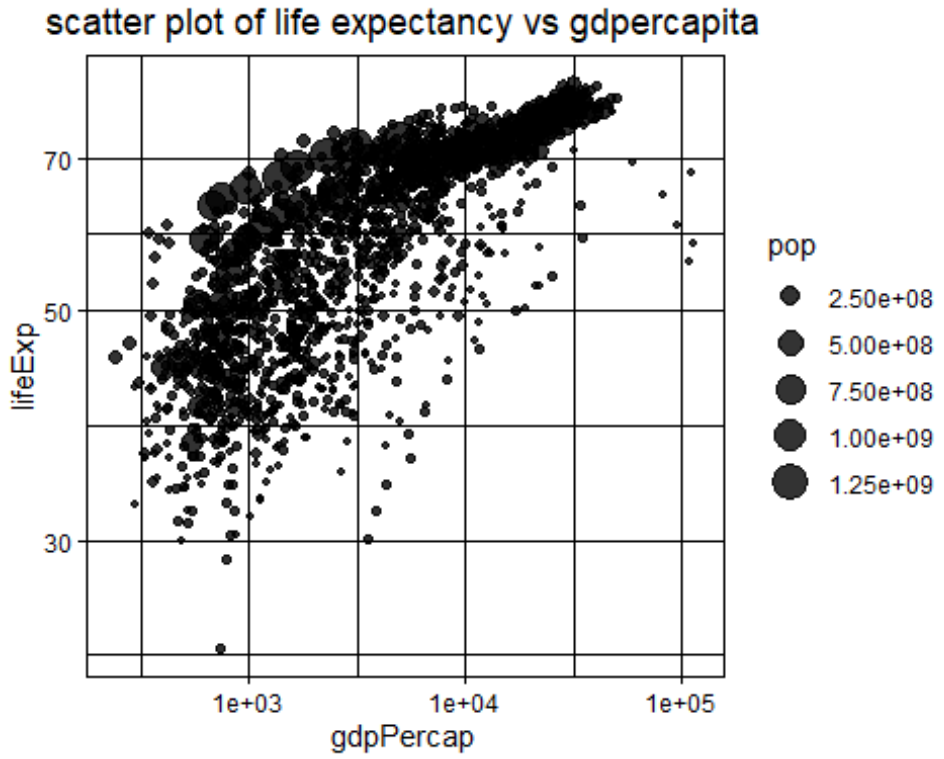
```
ggplot(data = gapminder, mapping = aes(x=gdpPercap,y=lifeExp)) +
geom_point(aes(color = continent)) + theme_linedraw() + ggtitle(label =
"scatter plot of life expectancy vs gdpercapita") + theme(plot.title =
element_text(hjust = 0.5)) + scale_x_log10() + scale_y_log10()
```



QUESTION 5.7

Modify the last plot so that the size of the dots represents the population size. In addition, increase the transparency of the dots in order to mitigate overplotting.

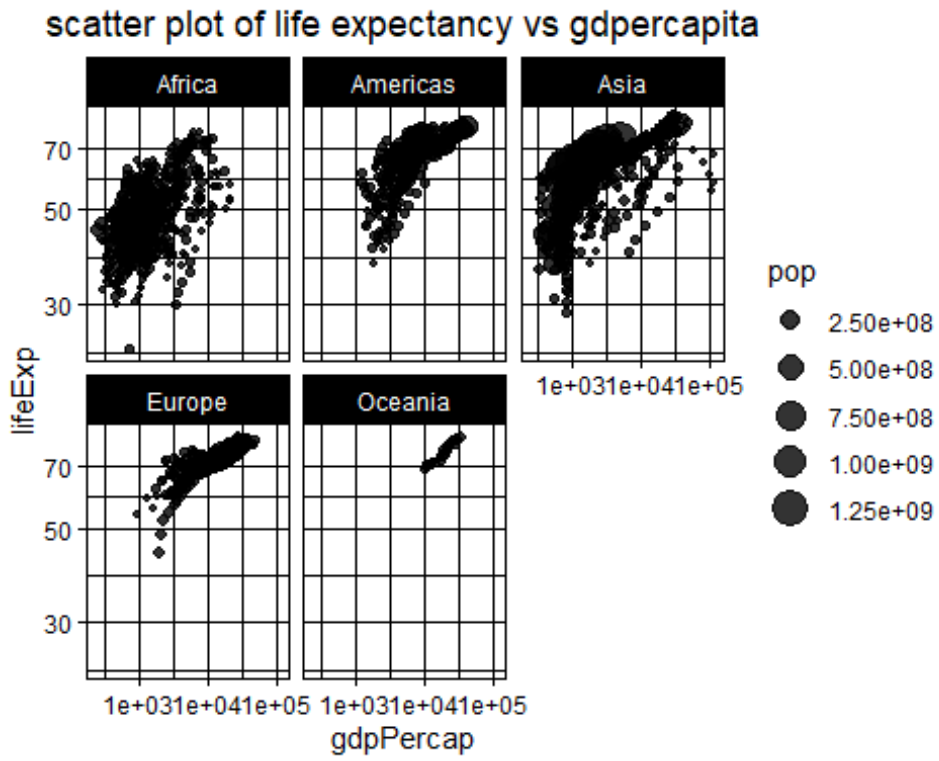
```
ggplot(data = gapminder, mapping = aes(x=gdpPercap,y=lifeExp)) +
geom_point(aes(size = pop), alpha = 0.8) + theme_linedraw() + ggtitle(label =
"scatter plot of life expectancy vs gdpercapita") + theme(plot.title =
element_text(hjust = 0.5)) + scale_x_log10() + scale_y_log10()
```



QUESTION 5.8

Modify the last plot so that there's a facet (sub-plot) for each continent.

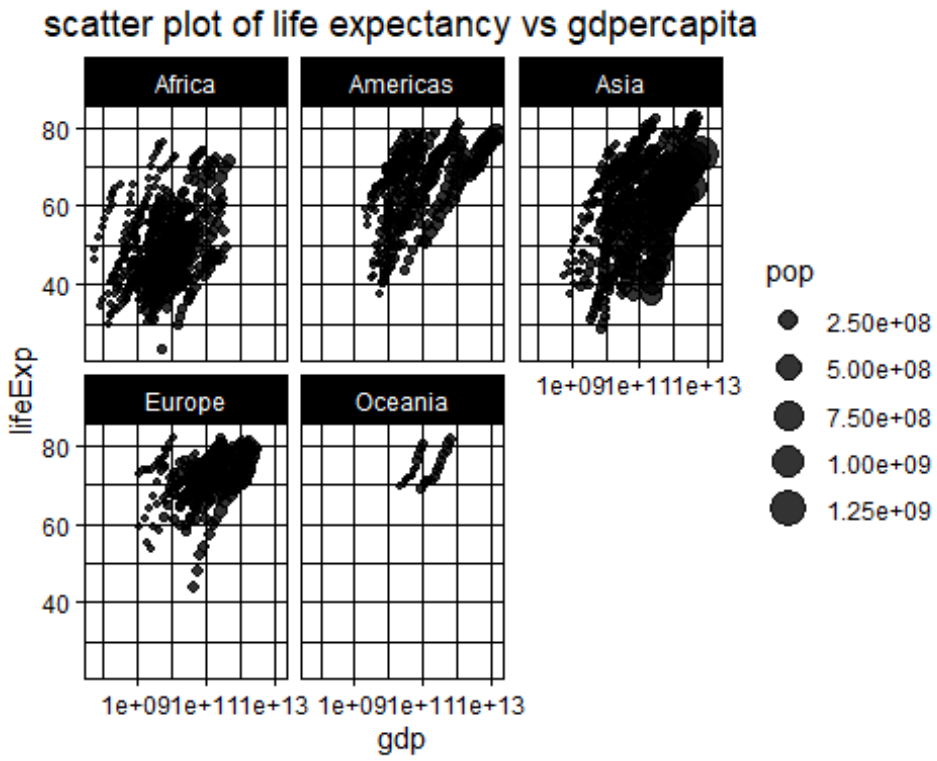
```
ggplot(data = gapminder, mapping = aes(x=gdpPercap,y=lifeExp)) +
geom_point(aes(size = pop), alpha = 0.8) + theme_linedraw() + ggtitle(label =
"scatter plot of life expectancy vs gdpercapita") + theme(plot.title =
element_text(hjust = 0.5)) + scale_x_log10() + scale_y_log10() + facet_wrap(~
continent)
```



QUESTION 5.9

Modify the last plot so that GDP is log transformed.

```
ggplot(data = p1, mapping = aes(x=gdp,y=lifeExp)) + geom_point(aes(size =
pop), alpha = 0.8) + theme_linedraw() + ggtitle(label = "scatter plot of life
expectancy vs gdpercapita") + theme(plot.title = element_text(hjust = 0.5)) +
scale_x_log10() + facet_wrap(~ continent)
```


scatter plot of life expectancy vs gdpercapita
Africa
Americas
Asia
Europe
Oceania
lifeExp
80
60
40
1e+09
1e+11
1e+13
gdp
pop
2.50e+08
5.00e+08
7.50e+08
1.00e+09
1.25e+09