Homework 04 CSCI 036 Solutions

Lucas Welch

Due: Friday, 2022-09-30

Instructions

Please box your answers. For numerical answers, this can be done using something like $\boxed{34}$. For text answers, this can be done using something like $\boxed{\text{My answer}}$. The output of a code chunk is automatically boxed, so no need to do more.

a. What function do you use to draw a blank canvas in the tidyverse?

b. What function do you use to add a set of points to a canvas?

c. What function do you use to add lines connecting (x, y) values to a canvas?

a. ggplot

b. geom_point

c. geom_smooth

Consider the `cars` data set that is built in to R.

a. Create a new data set based on `cars` that has a column `ratio` that is the ratio between the speed of the car and the stopping distance of the car.

b. Create a scatter plot of `ratio` versus `speed`.

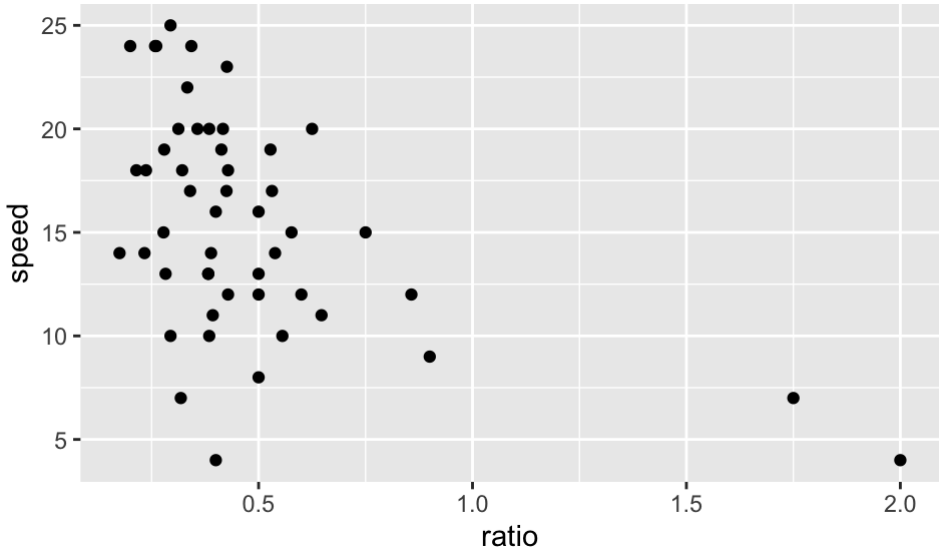
c. Add a least squares line fit to the plot.

a.

```
cars2 <- cars |>
  mutate(ratio = speed/dist)
```

b.

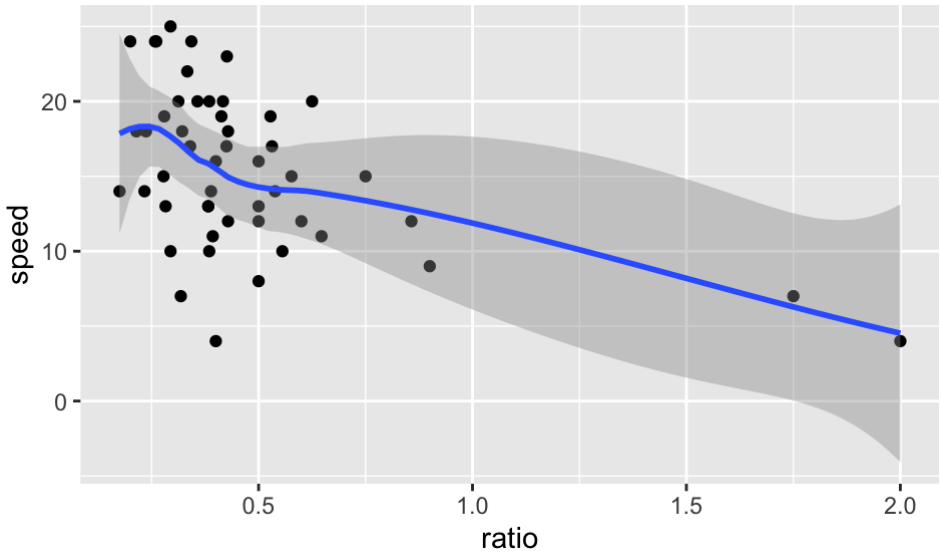
```
g1 <-
  cars2 |>
  ggplot(aes(x = ratio, y = speed)) +
  geom_point()
g1
```



c.

```
g1 <-
  cars2 |>
  ggplot(aes(x = ratio, y = speed)) +
  geom_point() +
  geom_smooth()
g1
```

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


20
10
0
speed
0.5
1.0
1.5
2.0
ratio

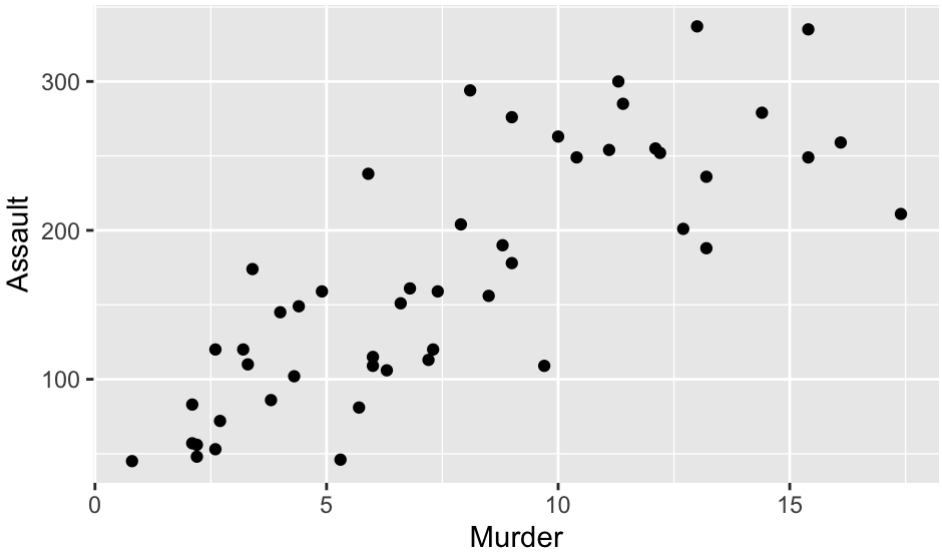
Consider the built in data set `USArrests` .

a. Using the help, what does the variable `Assault` measure in the data set?
b. Create a scatterplot of Murder arrests versus Assault arrests.
c. Add a loess line to your scatterplot.

a. numeric Assault arrests(per 100,000)

b.

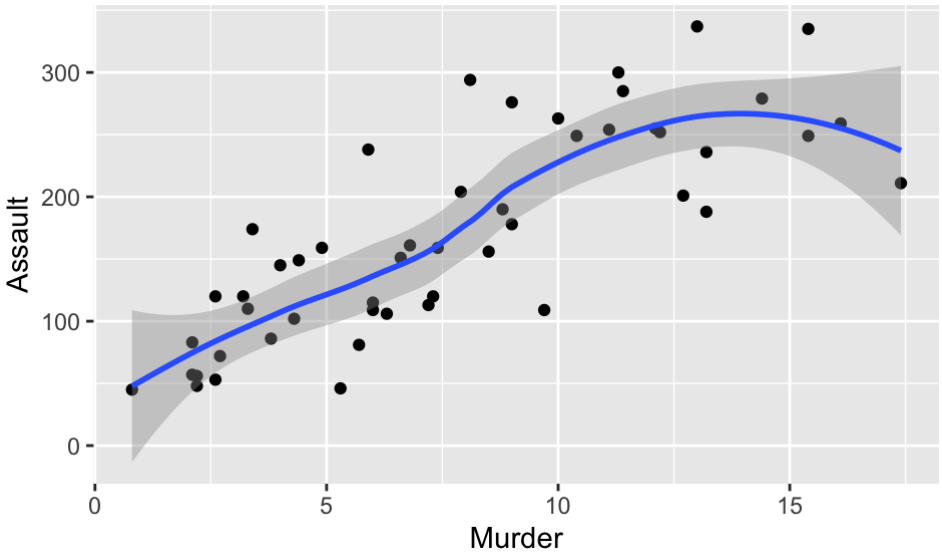
```
g1 <-
  USArrests |>
  ggplot(aes(x = Murder, y = Assault)) +
  geom_point()
g1
```



c.

```
g1 <- USArrests |>
  ggplot(aes(x = Murder, y = Assault)) +
  geom_point()+
  geom_smooth(method = "loess")
g1
```

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

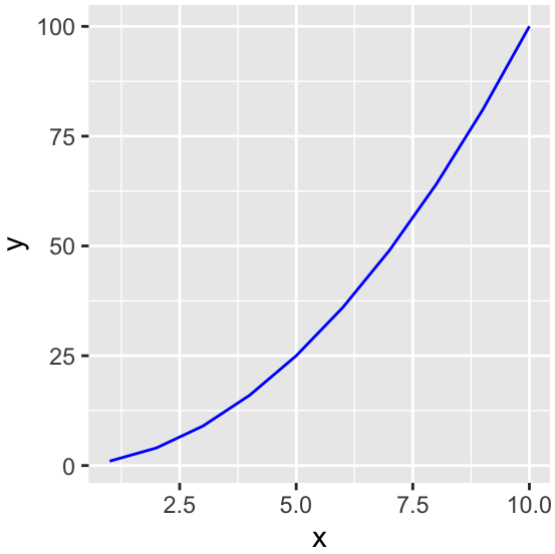

300
200
100
0
Assault
0
5
10
15
Murder

Consider the following tibble

```
x <- 1:10
y <- x^2
df <- tibble(x = x, y = y)
```

If we plot this tibble using `geom_line`, the default axis values are set automatically by default.

```
ggplot(df) + geom_line(aes(x, y), color = "blue")
```

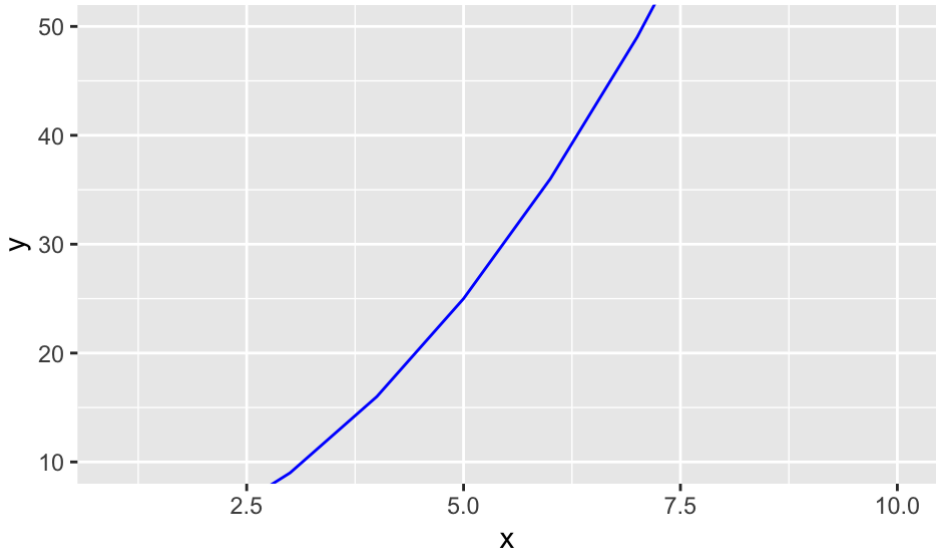


a. Use the `coord_cartesian` function to restrict the y values from 10 to 50.

b. In the previous part the `ylim` parameter inside the `coord_cartesian` function was used to set the limits. It turns out that [ggplot2]{class=PackageName] has a *separate* function also called ylim. Use `?ylim` to see help and some examples. In this part, use the `ylim` function to restrict the y values from 10 to 50 in your plot.

a.

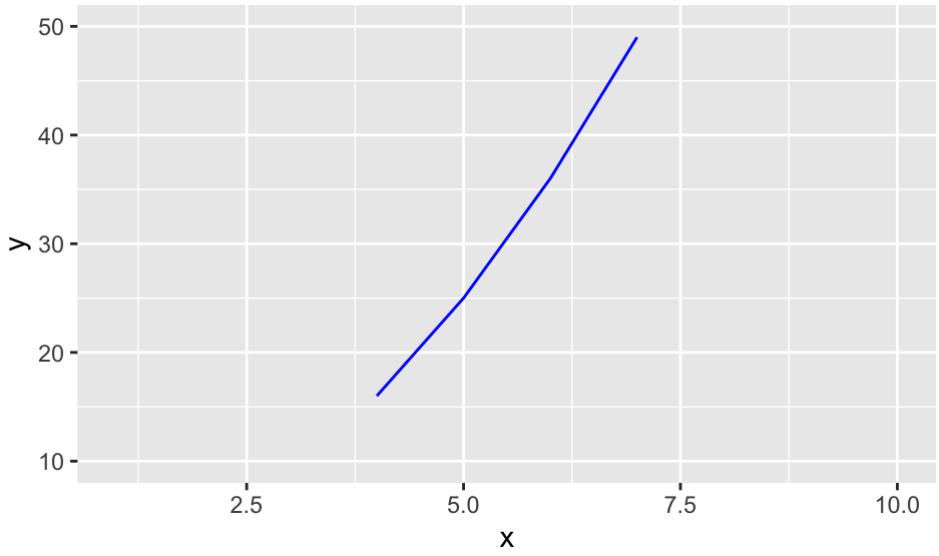
```
ggplot(df) + geom_line(aes(x, y), color = "blue") +
  coord_cartesian(ylim = c(10,50))
```



b.

```
ggplot(df) + geom_line(aes(x, y), color = "blue") +ylim(10,50)
```

```
## Warning: Removed 6 row(s) containing missing values (geom_path).
```



This problem requires that you use data from the maps package.

a. Plot a map of the state of Iowa using the correct aspect ratio.

b. The capital of Iowa is Des Moines, at 41.5868°, -93.6250° in latitude and longitude. Place a point here at that location.

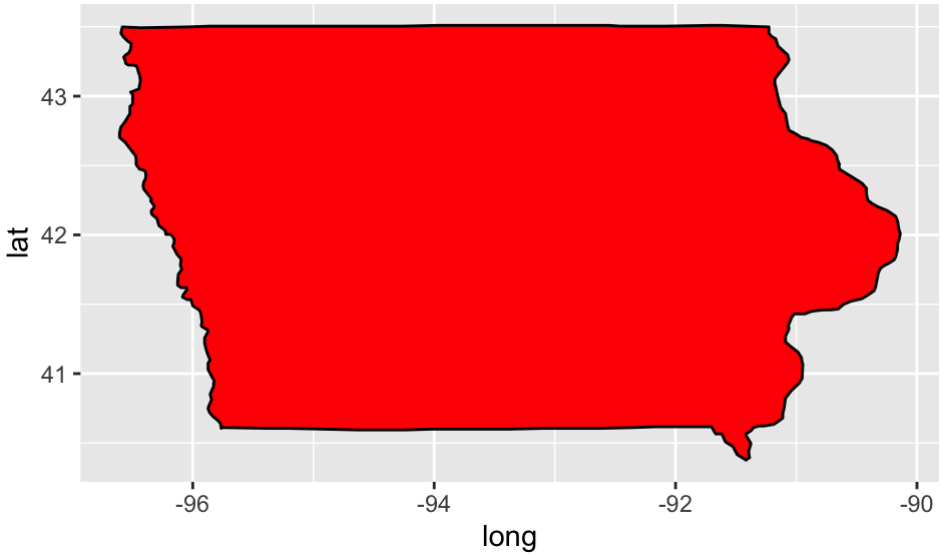
```
library(maps)
```

```
##
## Attaching package: 'maps'
```

```
## The following object is masked from 'package:purrr':
##
##     map
```

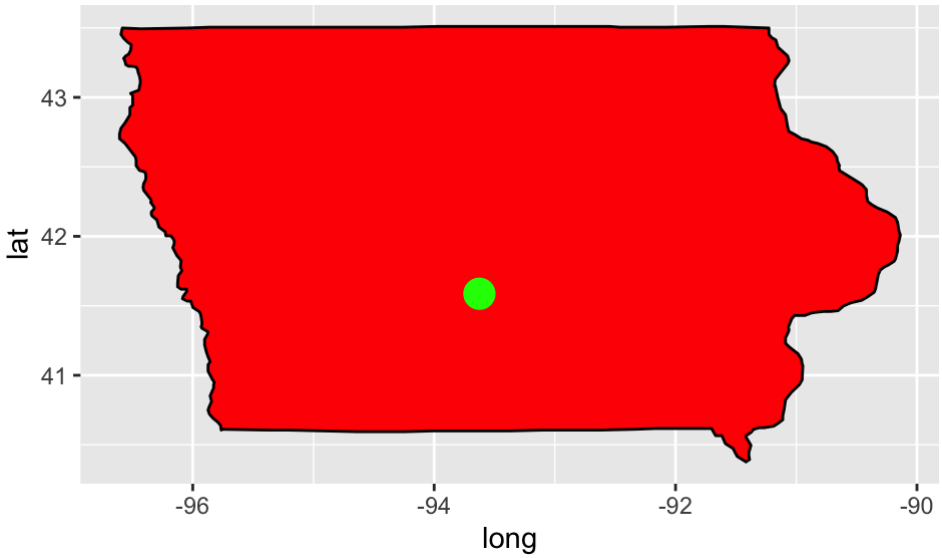
a.

```
Iowa <- map_data("state") |>
  filter(region == "iowa")
ggplot() + geom_polygon(data = Iowa, aes(x=long, y=lat), fill = "red", color = "black")
```



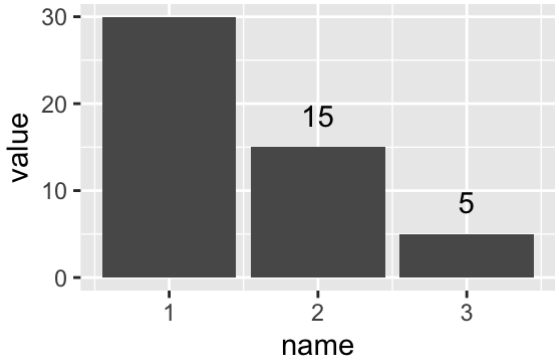
b.

```
Iowa <- map_data("state") |>
  filter(region == "iowa")
ggplot() + geom_polygon(data = Iowa, aes(x=long, y=lat), fill = "red", color = "black")
 +
  geom_point(aes(x =  -93.6250, y = 41.5868), color = "green", size = 5)
```


43
42
41
lat
-96
-94
-92
-90
long

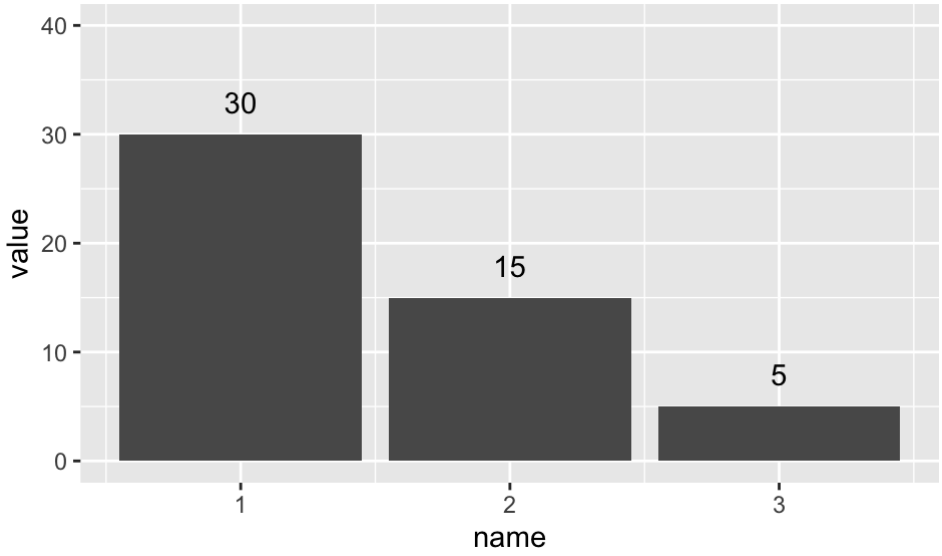
Consider the following plot, which creates a bar plot using `stat = "identity"`, and then puts the height of each bar as a number that sits above the bar.

```
df <- enframe(c(30,15,5))
ggplot(df, aes(x = name, y = value)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label = value), vjust = -1)
```



Change the y limits so that the label above the first bar is not cut off.

```
df <- enframe(c(30,15,5))
ggplot(df, aes(x = name, y = value)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label = value), vjust = -1) + ylim(0,40)
```



Consider the `starwars` data set from the `dplyr` package. A *boxplot* gives a way of seeing where the bulk of values for a variable lie. The line in the middle of the box is the sample median, and the width of the box depends on the spread in the variable. Points outside the box indicate *outliers*, values very far away from the center.

a. Create a boxplot of character height versus gender using geom_boxplot.

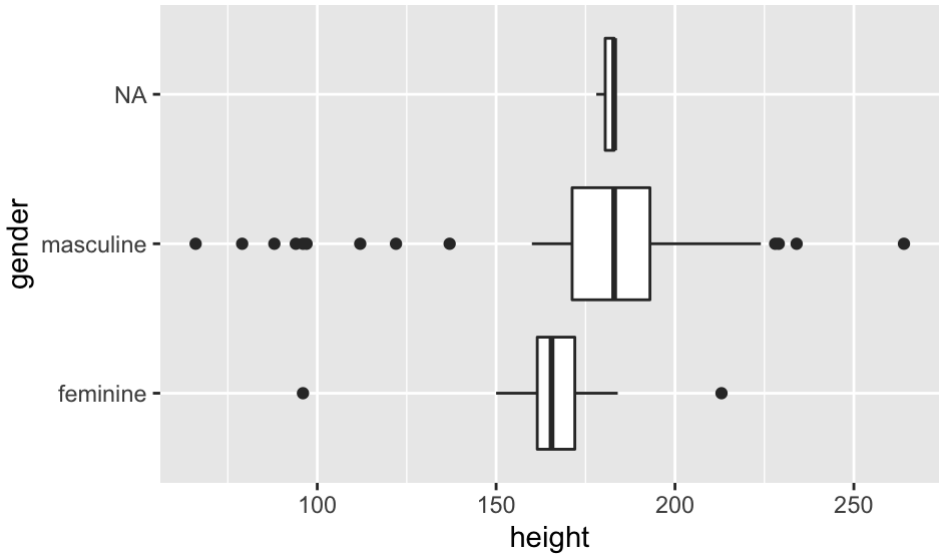
b. Which gender tends to have a larger height, male or female.

c. Which has greater variation, male or female?

a.

```
g1 <- ggplot(starwars, aes(height, gender)) +
  geom_boxplot()
g1
```

```
## Warning: Removed 6 rows containing non-finite values (stat_boxplot).
```



b. masculine

c. masculine

In the 2018 US Census American Community Survey (Table B03002), the following breakdown of the population of New York City by Race was given:

```
library(tibble)
df <- tibble(
  race = c("White", "Black or African American", "Some Other Race", "Asian", "Two or Mor
e Races", "American Indian and Alaska Native", "Native Hawaiian and Other pacific Island
er"),
  pop = c(3603057, 2049418, 1277050, 1177700, 296074, 36075, 4339)
)
```

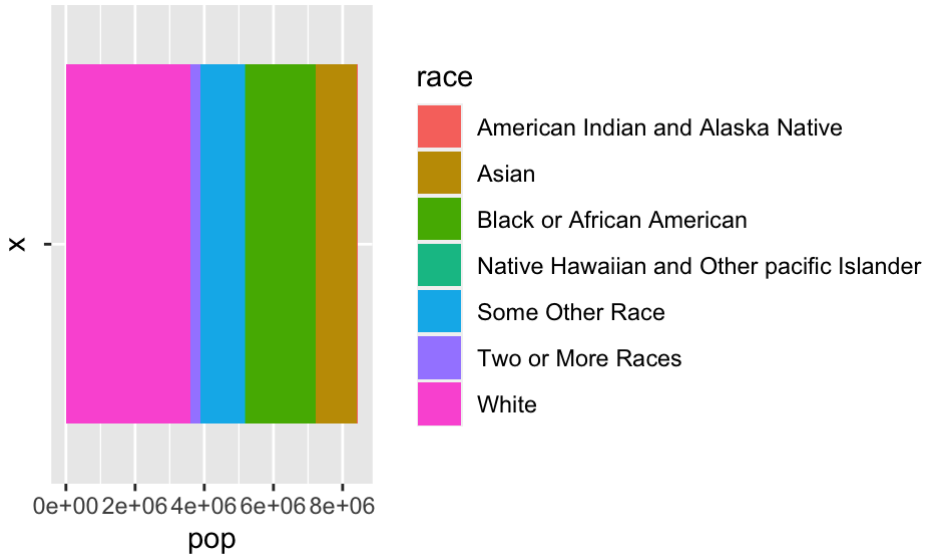
This can be printed as a table as follows:

```
knitr::kable(df)
```

race	pop
White	3603057
Black or African American	2049418
Some Other Race	1277050
Asian	1177700
Two or More Races	296074
American Indian and Alaska Native	36075
Native Hawaiian and Other pacific Islander	4339

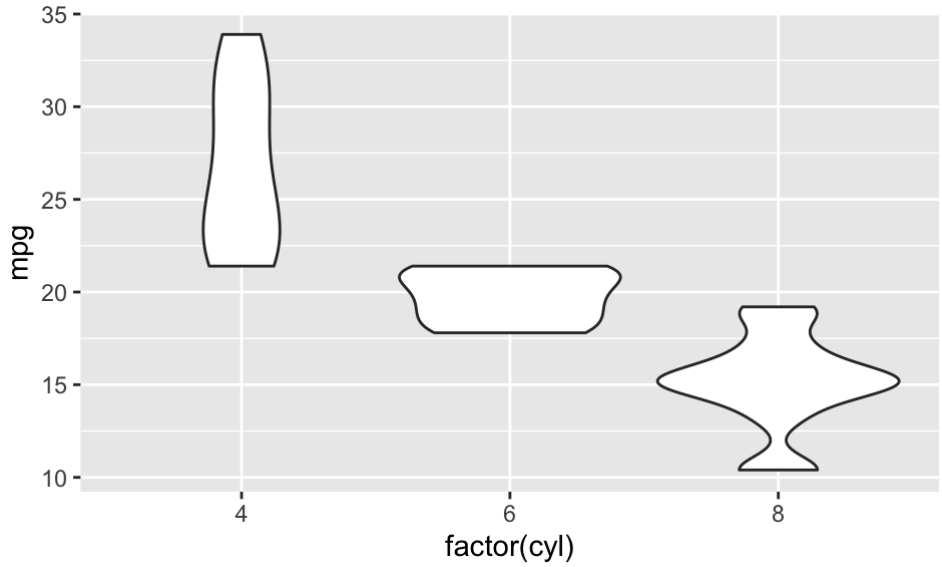
Create a composition plot that is a horizontal bar that is colored with a legend according to the different races represented in the survey.

```
g1 <- ggplot(df) +
  geom_bar(aes(x = "", y = pop, fill = race), stat="identity") +
  coord_flip()
g1
```


x
0e+00
2e+06
4e+06
6e+06
8e+06
pop
race
American Indian and Alaska Native
Asian
Black or African American
Native Hawaiian and Other pacific Islander
Some Other Race
Two or More Races
White

A *violin plot* is a method of displaying variation in a continuous variable that combines aspects of a boxplot and a histogram. Consider the following plot of data from the `mtcars` dataset.

```
ggplot(mtcars, aes(factor(cyl), mpg)) +
  geom_violin()
```



Note that in the aesthetic we used `factor(cyl)` rather than just `cyl`. This forces the plot to treat `cyl` as a categorical variable rather than a continuous one.

The histogram for each factor is mirrored with vertical symmetry, giving it a look in a very general sense like a violin. That is the reason behind the name.

a. From this plot, what happens to miles per gallon as the number of cylinders increases?

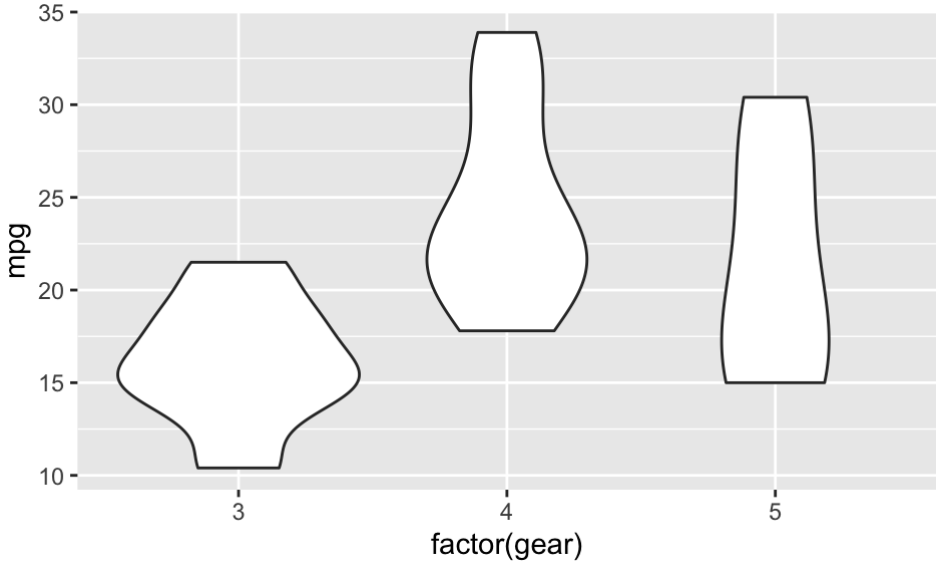
b. Create a violin plot for `mpg` versus `gear`.

c. For which number of forward gears is the mpg the best?

a. decreases

b.

```
ggplot(mtcars, aes(factor(gear), mpg)) +
  geom_violin()
```



c. $\boxed{4}$

Consider the `starwars` data set from package `dplyr`.

a. Create a scatter plot of the mass versus the height with a least squares prediction line.

b. What observation has the outlier among mass?

c. Repeat part a after removing this outlier.

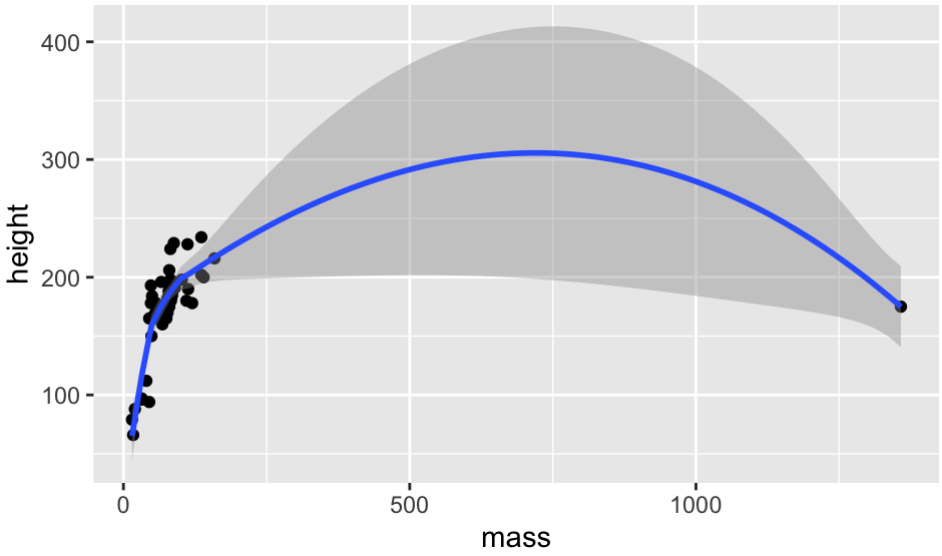
a.

```
starwars |> ggplot(aes(x = mass, y = height)) +
  geom_point() +
  geom_smooth()
```

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

```
## Warning: Removed 28 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 28 rows containing missing values (geom_point).
```

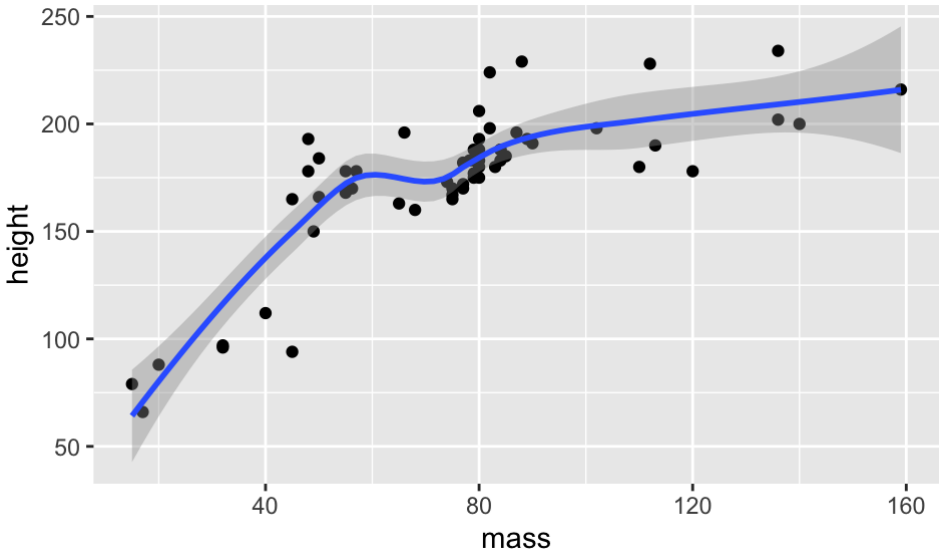


b. Jabba Desilijic Tiure

c.

```
starwars |> filter(mass<1000) |> ggplot(aes(x = mass, y = height)) +
  geom_point() +
  geom_smooth()
```

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


250
200
150
100
50
height
40
80
120
160
mass