Lab 7

Insert Name

Math 141, Week 8

Due: Before your Week 9 lab meeting

Goals of this lab

1. Construct confidence intervals using both the se method and the percentile method.
2. Learn how to modify the confidence interval code, depending on the parameter of interest.
3. Practice interpreting confidence intervals.

Problems

```r
# Insert libraries here
library(tidyverse)
library(infer)
```

Problem 1

For this problem, we are going to return to the example we saw in class.

In a recent study, 23 rats showed compassion that surprised scientists. Twenty-three of the 30 rats in the study freed another trapped rat in their cage, even when chocolate served as a distraction and even when the rats would then have to share the chocolate with their freed companion. (Rats, it turns out, love chocolate.) Rats did not open the cage when it was empty or when there was a stuffed animal inside, only when a fellow rat was trapped. We wish to use the sample to estimate the proportion of rats that show empathy in this way.

a. We want to estimate the proportion of all rats that are empathetic. Is that a parameter or statistic? What is the symbol?

- Parameter
  - We are looking at the population
  - P

---

b. What is the correct point estimate? Include the appropriate symbol.

- $\hat{p} = 0.7666667$

---

c. Describe how to use 30 slips of paper to find one bootstrap statistic.

---

1. Put 23 *empathetic* slips of paper alond with 7 *non-empathetic* in a hat
2. Bootstrap
    - Draw a slip of paper
    - Make a note of wheter or not
    - Repeat the above 30 times and call it a bootstrap statistic!
3. Repeat step two until satisfied with standard deviation (*If you are doing more than one bootstrap*)

---

d. Describe the shape and center of the class's bootstrap distribution.

- Center: 23/30
- Spread: Narrow
- Skew: None
- Std: ≈ 0.05

---

e. Explain why we were able to construct (part of) a genuine sampling distribution for the sample proportion of GIFs with animals but are not able to construct a sampling distribution for this problem.

- We did not bootstrap for the entire sample (all gifs on the internet). Instead we created we sampled the population

---

f. Why are we able to construct a bootstrap distribution for this problem?

- We have a complete population (30 cases)

---

g. Since it takes a very long time to construct a bootstrap distribution by hand, let's now fully use r. The following code generates a bootstrap distribution.

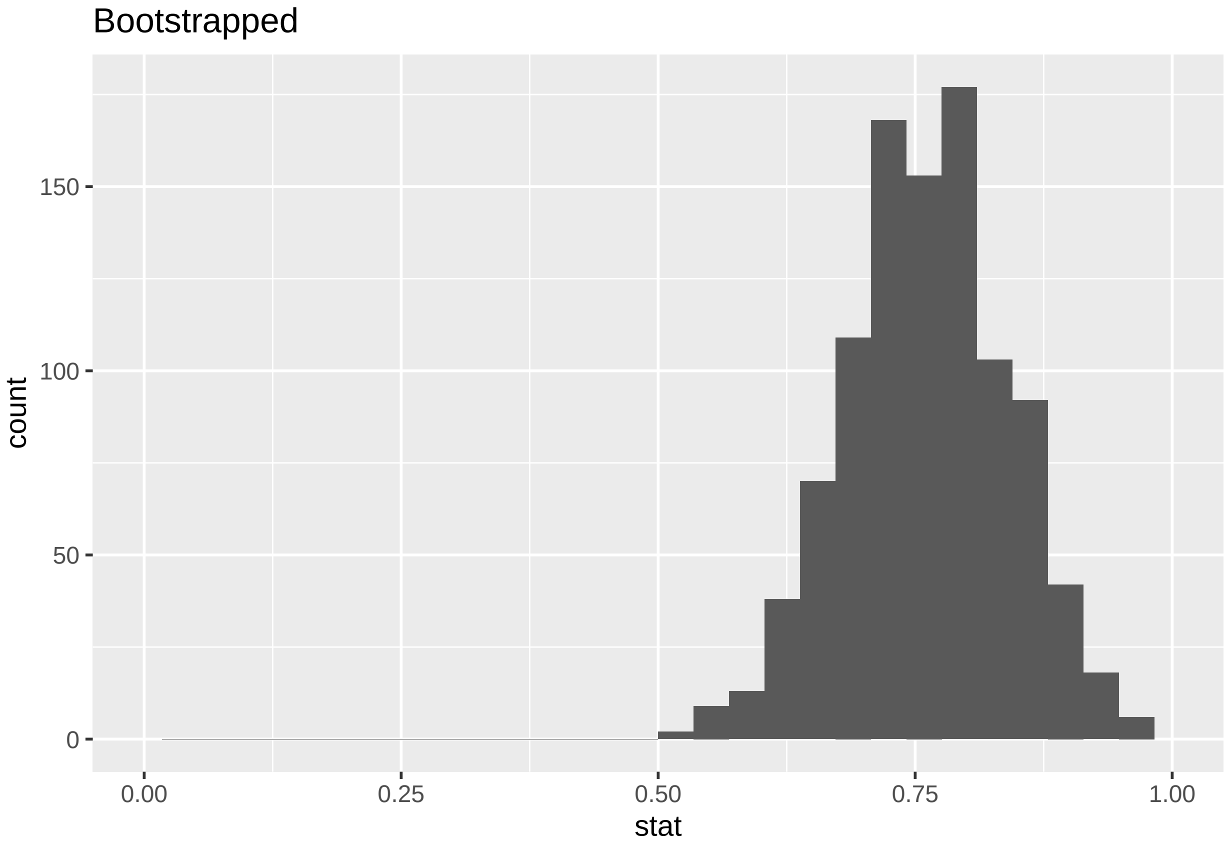
Explain what each line of code is doing.

```r
rats <- data.frame(empathy = c(rep("Yes", 23), rep("No", 7)))
#creates dataframe with 23 yes and 7 no

# Construct the bootstrap distribution
bootstrap_dist <- rats %>%
# Intializes the df made above to output to variable bootstrap_dist
  specify(response = empathy, success = "Yes") %>%
# Specifies response variable and what to calc for prop
  generate(reps =  1000, type = "bootstrap") %>%
# Creates 1000 bootsraps
  calculate(stat = "prop")
# Creates proportions of the bootstraps
```

- *See coments above*

h. Plot the bootstrap distribution. Describe the center and shape of the distribution.

```r
ggplot(bootstrap_dist, aes(stat)) +
  geom_histogram(bins=30) +
  labs(title="Bootstrapped") +
  xlim(0, 1)
```



```r
Mode <- function(x) {
  ux <- unique(x)
  ux[which.max(tabulate(match(x, ux)))]
}
# The above is from stack overflow
# why is there no mode IN A PROGRAMMING LANGUAGE
# FOR STATISTCIANS!?!?!!
```

- Centered around 0.8
- Narrow spread: 0.0770581
- Skew:
  - Using $\bar{x} - \tilde{x} = 0.0024$
  - None

i. Construct a 95% confidence interval for the parameter of interest. Will the whole class get the same interval? Why or why not?

```
#the no frills route
cat("Using the SD method Lower: ", mean(bootstrap_dist$stat)-2*sd(bootstrap_dist$stat))
```

```
## Using the SD method Lower:  0.6149504
```

```
cat("Using the SD method Upper: ", mean(bootstrap_dist$stat)+2*sd(bootstrap_dist$stat))
```

```
## Using the SD method Upper:  0.9231829
```

```
#using other method
quantile(bootstrap_dist$stat, c(0.025, 0.975))
```

```
##      2.5%     97.5%
## 0.6333333 0.9000000
```

- No, but they should get relatively close
- Standard deviation: 0.0770581
- As the standard deviation is so small, it is unlikely to vary greatly. It is possible to sample bootstraps. But this is highly unlikely going to change the outcome.

---

j. What does your confidence interval tell us about the population?

- The mean is within (0.6333333, 0.9) 95% of the time.

---

k. Is there evidence that a majority of rats will show empathy? Justify your answer.

```
sum(bootstrap_dist$stat>0.5)/length(bootstrap_dist$stat)
```

Using the above

- Percent bootstraps with greater than 50% empathetic rats: 1
- Yes

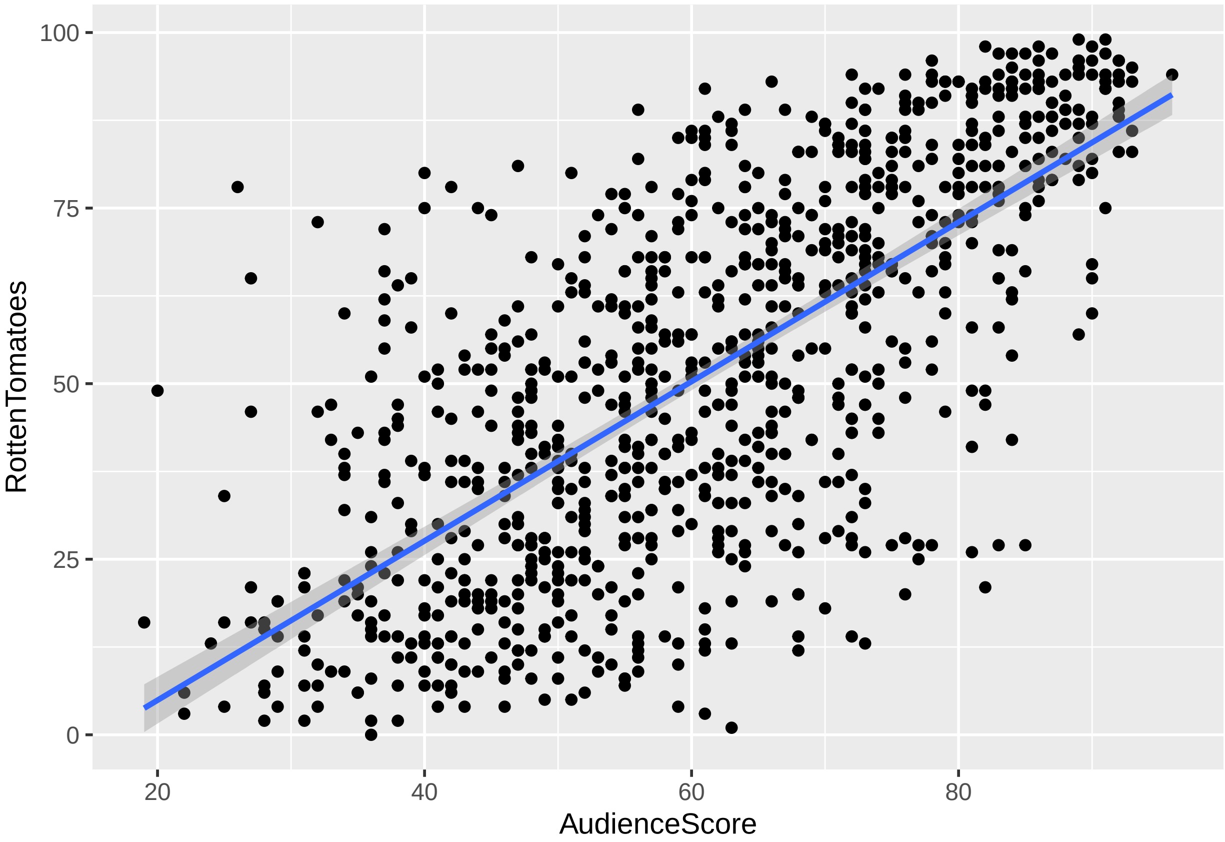
Problem 2

Let's return to the movies dataset and revisit our explorations of the correlation between the critics' ratings and audience ratings.

```
# Read in data
movies <- read_csv("/home/courses/math141f19/Data/HollywoodMovies.csv")
```

a. Plot the `AudienceScore` and the `RottenTomatoes` score and include the line of best fit. Does there appear to be a linear relationship between these two variables? Does the trend seem to be that both groups give the same score? (In other words, if the `AudienceScore` is a 60, is the `RottenTomatoes` score also about 60?)

```
ggplot(movies, aes(AudienceScore, RottenTomatoes)) +
  geom_point() + geom_smooth(method="lm")
```



```r
lm(movies$RottenTomatoes ~ movies$AudienceScore)
```

```
##
## Call:
## lm(formula = movies$RottenTomatoes ~ movies$AudienceScore)
##
## Coefficients:
##          (Intercept)  movies$AudienceScore
##              -17.767                 1.135
```

- There is a line of best fit with a high coorelation.
- But there is not a 1 for 1 relation. The cooeficient is 1.135.

---

b. What is the value of the correlation coefficient for these data? Is it a statistic (r) or a parameter (ρ)?

```r
no_nan_movies <- movies %>%
  drop_na(RottenTomatoes) %>%
  drop_na(AudienceScore)

cor(no_nan_movies$AudienceScore,
    no_nan_movies$RottenTomatoes)
```

```
## [1] 0.7029077
```

- It is a statistic, it refers to the entire dataset which is a sample of a population.

____

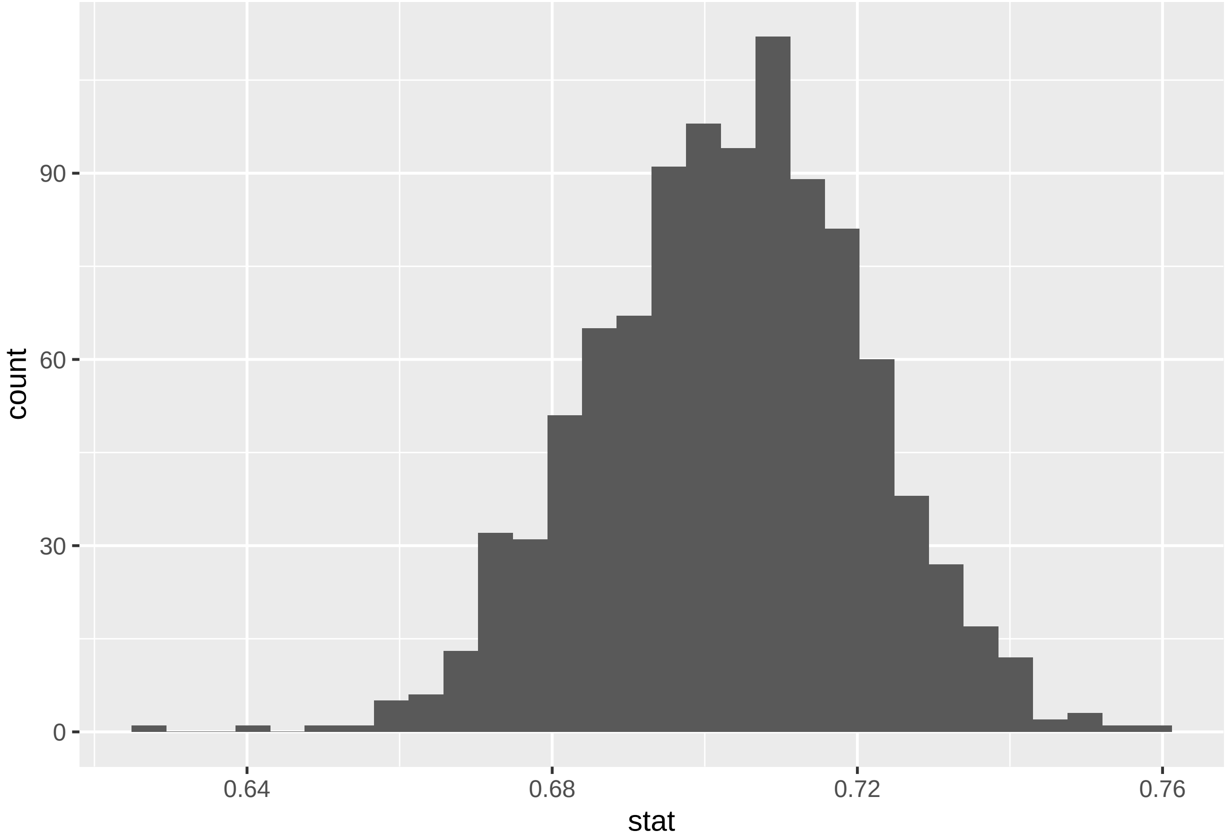
c. Generate the bootstrap distribution.

```
# Construct the bootstrap distribution
# Hint 1: For the specify step:
boot_movie<- no_nan_movies %>%
specify(RottenTomatoes~ AudienceScore) %>%
generate(reps =  1000, type = "bootstrap") %>%
# Hint 2: For the calculate step, use
calculate(stat = "correlation")



sample_n(boot_movie, 5)
```

```
## # A tibble: 5 x 2
##   replicate  stat
##       <int> <dbl>
## 1       168 0.723
## 2       607 0.714
## 3       374 0.700
## 4       871 0.737
## 5       377 0.717
```

d. Plot the bootstrap distribution. Comment on the center and shape.

```
ggplot(boot_movie, aes(stat)) +
  geom_histogram()
```



---

e. Construct a 99% confidence interval using the percentile method.

```r
quantile(boot_movie$stat, c(0.005, 0.995))
```

```
##      0.5%     99.5% 
## 0.6574615 0.7453782
```

f. Construct a 90% confidence interval using the percentile method.

```r
quantile(boot_movie$stat, c(0.05, 0.95))
```

```
##        5%       95% 
## 0.6735748 0.7318305
```

g. Compare the 90% and 99% confidence intervals. Which one is narrower and why is it narrower?

- 90th
- Because it ranges from the 5th to 95th percentile opposed ot the 99% CI which ranges from 0.05 to 99.5 percentiles.

---

h. Based on your intervals, do we have evidence of a positive linear relationship between the critic and audience ratings?

- Zero bootstrapped values were less than 0. Making it extremely unlikely that a negative coorelation is possible

---

Problem 3

Gas is pricey these days. In this problem, we want to compare the miles per gallon (MPG) of family sedans to the MPG of sports utility vehicles (SUV) to see if family sedans really do get better gas mileage (on average). I obtained the following sample of cars (in 2011) from the website http://www.fueleconomy.gov/feg/findacar.shtml#findacar.

```r
gasmileage <- read_csv("/home/courses/math141f19/Data/gasmileage.csv")
```

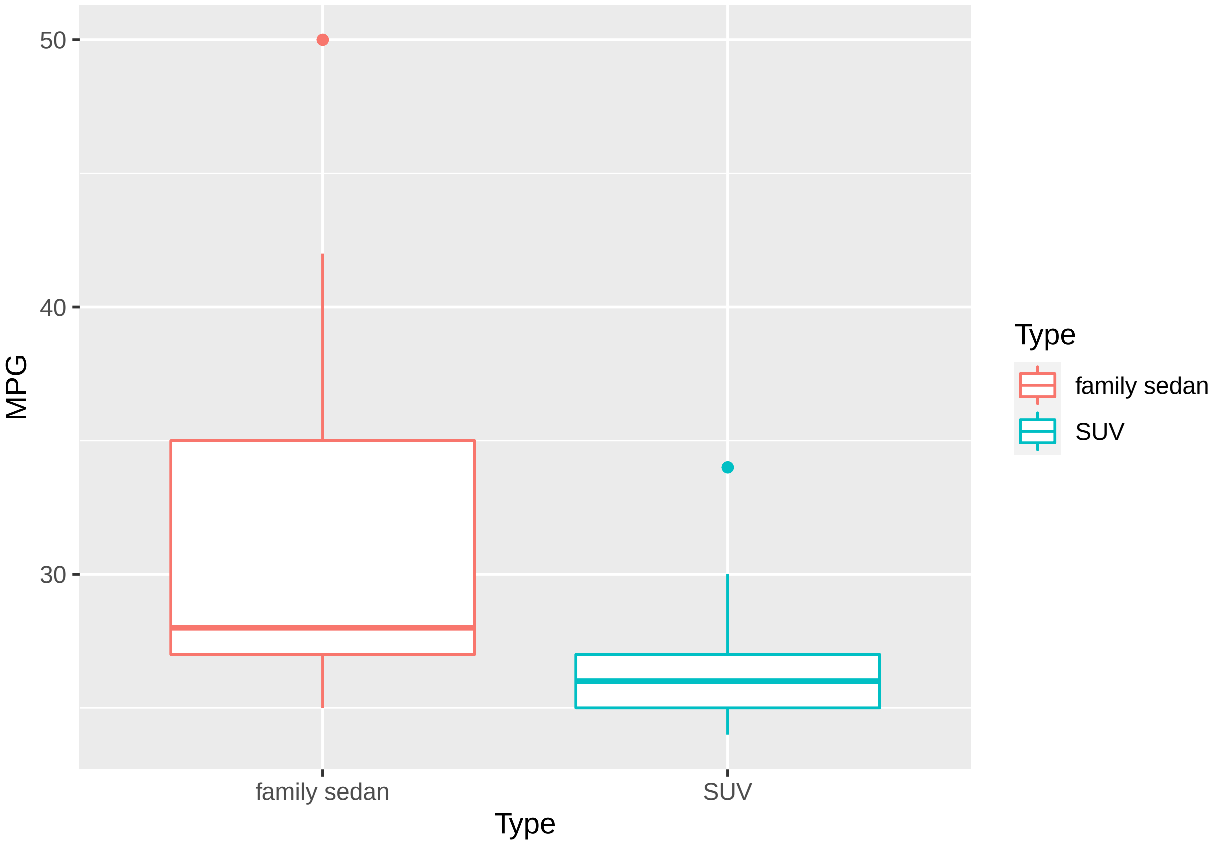
a. What is the parameter of interest? Include the correct symbol. (Here's a useful example μ_1.)

- `MPG` is the only quanitative variable
-

---

b. Create a useful graph of miles per gallon by type of car. What can you infer from the graphic?

```r
ggplot(gasmileage, aes(Type, MPG, color=Type)) +
  geom_boxplot()
```



- Sedans have a higher average MPG as well as spread.

c. Using either bootstrap method, construct a 90% confidence interval for the difference in mean gas mileage.

```
boot_sedan<- gasmileage %>%
  filter(Type=="family sedan") %>%
  specify(response = MPG) %>%
  generate(reps =  1000, type = "bootstrap") %>%
  calculate(stat = "mean")

boot_suv<- gasmileage %>%
  filter(Type!="family sedan") %>%
  specify(response = MPG) %>%
  generate(reps =  1000, type = "bootstrap") %>%
  calculate(stat = "mean")

dif_mean_sedan_suv = boot_sedan$stat-boot_suv$stat
quantile(dif_mean_sedan_suv, c(0.005, 0.995))
```

```
##   0.5%  99.5%
## 2.1996 9.0402
```

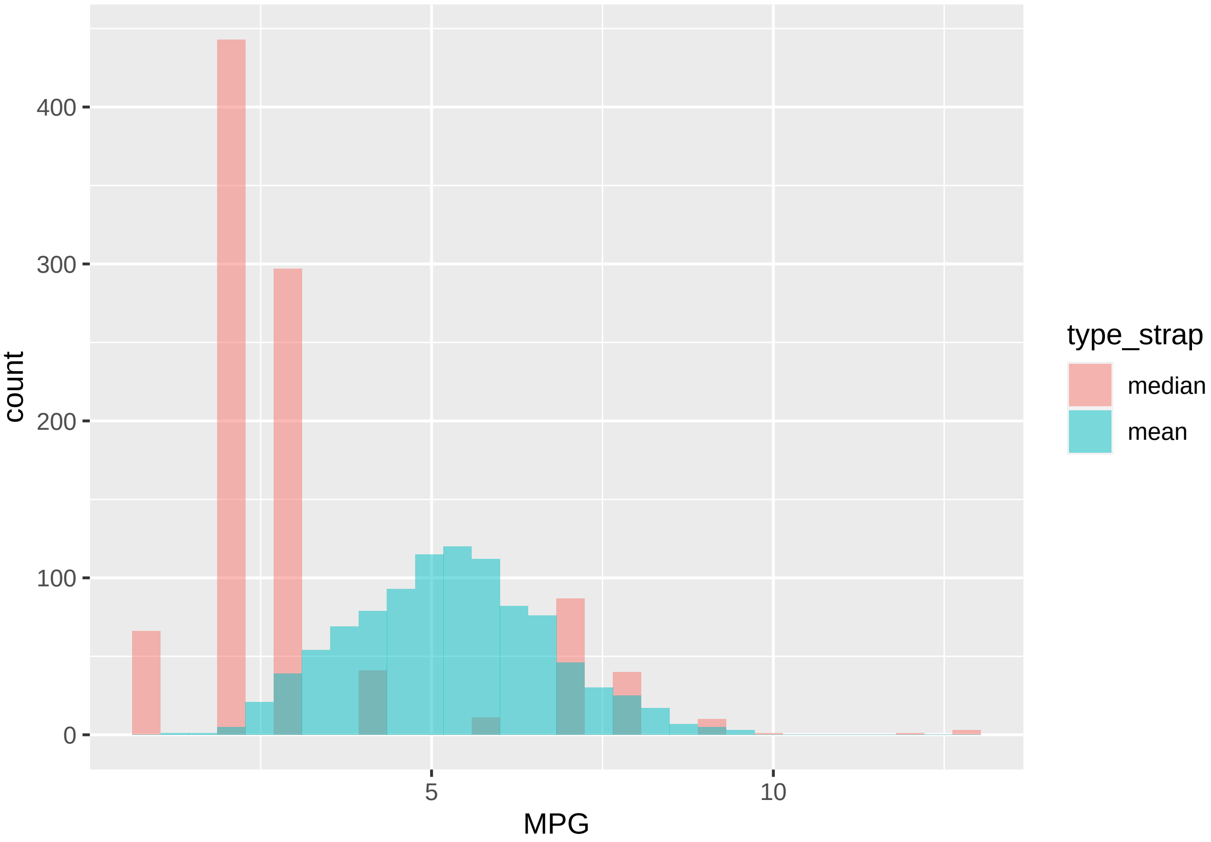
d. Let's change our parameter of interest to the difference in medians, instead of means. Using either bootstrap method, construct a 90% confidence interval for the difference in median gas mileage.

```
boot_sedan<- gasmileage %>%
  filter(Type=="family sedan") %>%
  specify(response = MPG) %>%
  generate(reps =  1000, type = "bootstrap") %>%
  calculate(stat = "median")

boot_suv<- gasmileage %>%
  filter(Type!="family sedan") %>%
  specify(response = MPG) %>%
  generate(reps =  1000, type = "bootstrap") %>%
  calculate(stat = "median")

dif_median_sedan_suv = boot_sedan$stat-boot_suv$stat
quantile(dif_median_sedan_suv, c(0.005, 0.995))
```

```
##  0.5% 99.5%
## 1.000 9.005
```

e. As we discussed in class, the accuracy of these bootstrap methods requires that the bootstrap distribution of the statistic is fairly bell-shaped and symmetric. Graph the bootstrap distribution of the difference in sample means and the bootstrap distribution of the difference in sample medians. Are they both roughly bell-shaped and symmetric? If one is not, why do you think that is?

```
convert_df <- data.frame(dif_median_sedan_suv,
                         rep(c("median"), times = length(dif_median_sedan_suv)))

convert_df<- setNames(convert_df,
         c('MPG','type_strap'))

convert_df_2 <- data.frame(dif_mean_sedan_suv,
                         rep(c("mean"), times = length(dif_mean_sedan_suv)))
convert_df_2<- setNames(convert_df_2,
         c('MPG','type_strap'))

grouped_samples <- rbind(convert_df, convert_df_2)


ggplot(grouped_samples, aes(MPG, fill=type_strap)) +
  geom_histogram(alpha=0.5, position="identity")
```



- The difference in the two comes from the fact that median is a single value (in this case an integer) and mean is the average value of a bootstrap.
- As a result mean can be in between values, while median can only be an integer that exists in the data frame.
- Mean is also dragged by outliers
