Problem Set 4

Daniel Shapiro

9/22/2022

Question 1 Background:[1]

Download the demo.csv dataset from the course website. The dataset contains information from a sample of countries in the year 2000, taken from the Democracy Time-Series Dataset. It includes the following variables:

- **Nation**: country name
- **GDP**: GDP per capita in constant US dollars
- **FHouse**: Freedom House rating (a measure of the level of political and civil liberties in a country, on a scale from 1.0 (most free) to 7.0 (least free))
- **OECD**: a dummy variable indicating OECD status
- **regime**: a variable coded from the Freedom House rating that indicates whether a country is free (1.0-2.5), partly free (2.51-5.5), or not free (5.51-7.0).

1a) Using stargazer(), show the summary statistics for your dataset. Briefly interpret for the GDP variable.

```r
# I like putting this sort of thing in a separate chunk so I don't have to run it a billion times.

demo <- read.csv("demo.csv")
```

```r
# Had to put some extra things into the title brackets to get the table to show up as
# more than just Latex. Also put header = FALSE to suppress the initial lines.
stargazer(demo, header = FALSE)
```

Table 1:

Statistic	N	Mean	St. Dev.	Min	Max
GDP	161	8,468.658	9,516.346	463	50,564
FHouse	161	3.460	1.897	1.000	7.000
OECD	161	0.186	0.391	0	1

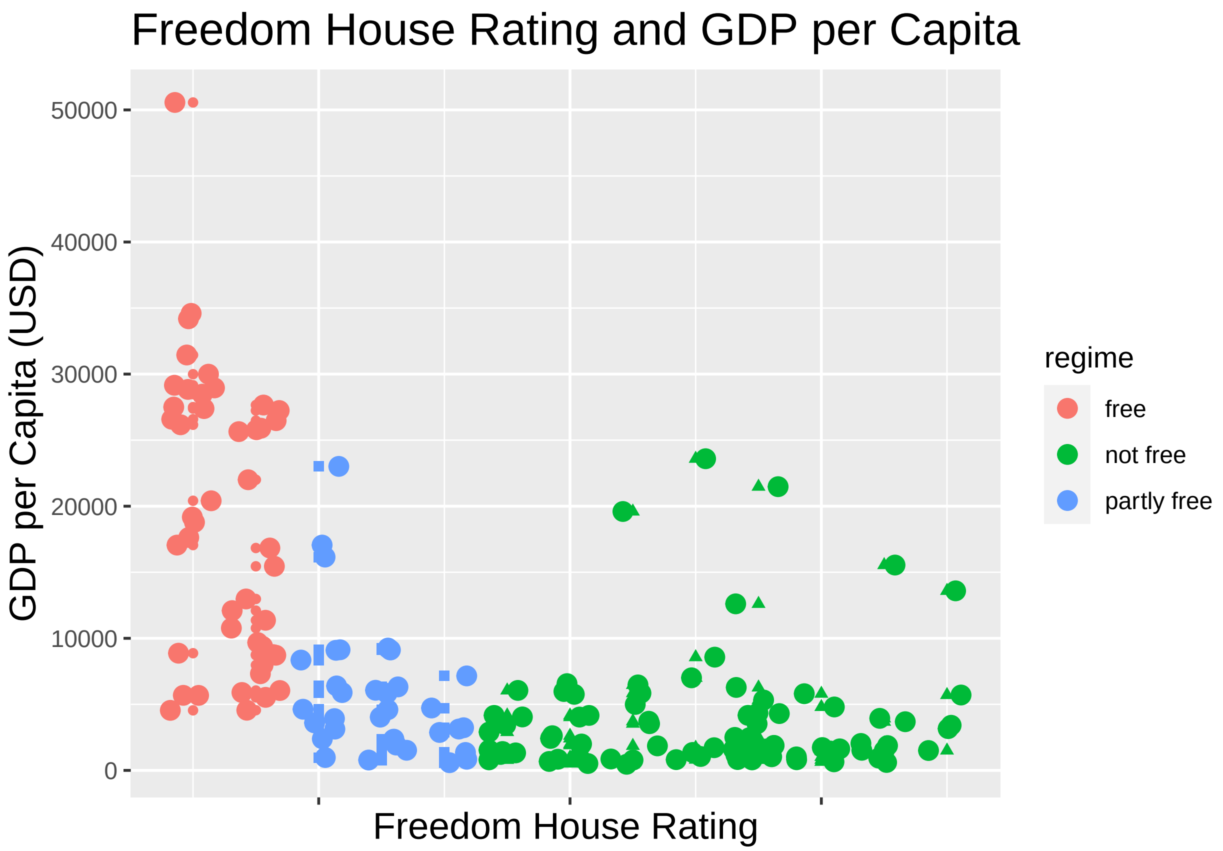
Brief interpretation of the GDP row: First, there's "N" which is just the number of observations – 161. The "Mean" is the mean GDP per capita across all observations. The standard deviation value is relatively high, so that means that the data is spread across a rather wide area. The "Min" and "Max" columns show the lowest GDP per capita measure (463) and the highest (50,564) in the dataset.

[1] For code, see my Github at https://github.com/dfshapir/ps4.

1b) Produce an appropriately named and labeled plot of GDP (on the y-axis) against FHouse (on the x-axis) using the ggplot() function (including a legend). Do the following:

1) Use a different color for data points representing different regimes (i.e. free, partly free, or not free)
2) In case some see your plot in black and white, use different point types for each regime category.
3) Adjust the size of the axis labels, axis titles, and title to make them more legible.
4) Increase the size of your points, and use geom_jitter() to make them more legible.

```
ggplot(demo, aes(x = FHouse, y = GDP, color = regime)) +
  geom_point(aes(shape = regime)) +
  labs(title = "Freedom House Rating and GDP per Capita",
       x = "Freedom House Rating",
       y = "GDP per Capita (USD)") +
  theme(plot.title = element_text(size = 17),
        axis.title = element_text(size = 14)) +
  geom_jitter(size = 3) +

# I removed the axis labels because they were non-nonsensical given the nature of the data.

  theme(axis.text.x=element_blank())
```



1c) Calculate the conditional expectation and the conditional standard deviation of GDP for the three regime types, using a function that takes as an input the type of regime and returns the conditional mean and standard deviation. What do the conditional summary statistics suggest about the relationship between democracy and wealth? Briefly explain.

To find the expected value of X, we can use the formula: $E[X] = \Sigma(x)P(x)$.

```
conditional <- function(type){

setup1 <- demo %>%
  filter(regime == "free")
exp1 <-  mean(setup1$GDP)
sd1 <- sd(setup1$GDP)

setup2 <- demo %>%
  filter(regime == "partly free")
exp2 <-  mean(setup2$GDP)
sd2 <- sd(setup2$GDP)

setup3 <- demo %>%
  filter(regime == "not free")
exp3 <-  mean(setup3$GDP)
sd3 <- sd(setup3$GDP)

  if(type == "free"){return <- c(exp1, sd1)}
  else if(type == "partly free"){return <- c(exp2, sd2)}
  else(return <- c(exp3, sd3))

return
}

conditional("free")
```

```
## [1] 19017.49 10671.70
```

```
conditional("partly free")
```

```
## [1] 5640.853 4975.443
```

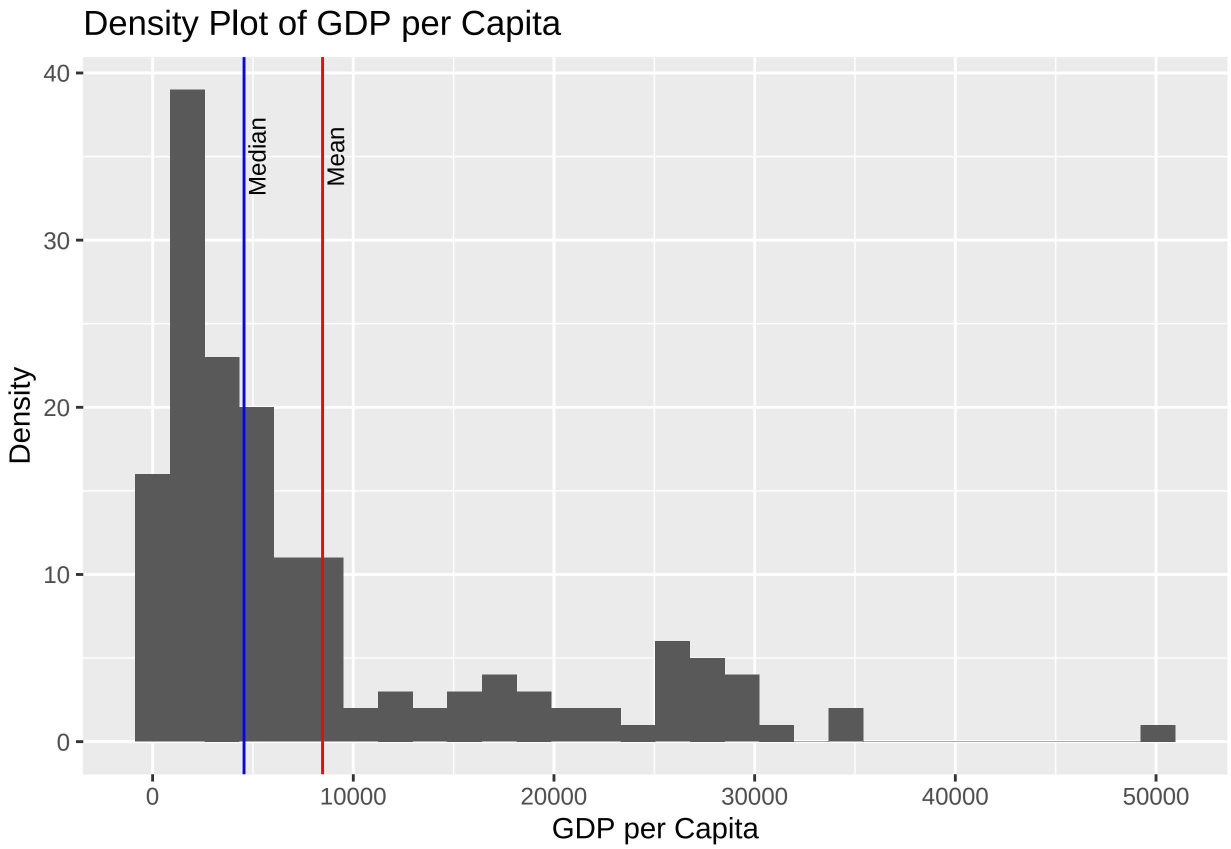
```
conditional("not free")
```

```
## [1] 3852.171 4492.659
```

According to these statistics, a higher level of democracy tends to correlate with a higher expected GDP per capita. Regimes labeled “free” have a higher GDP per capita than regimes labeled “partly free,” which in turn have a higher GDP per capita than regimes labeled “not free.”

1d) Using the geom_histogram() command in ggplot(), produce a density plot of GDP per capita. Overlay two vertical lines, in different colors, for the mean and the median of that variable. Annotate the graph to mark these lines informatively using geom_text() (hint: geom_text takes a dataframe as an input, so start by making a dataframe of your labels and their desired position). What does the relationship between the mean and the median, as shown on the plot, tell you about the variable GDP per capita?

```r
xint <- mean(demo$GDP)
medint <- median(demo$GDP)

textdata <- data.frame(xint, medint)

ggplot(demo, aes(x = GDP)) +
  geom_histogram(bins = 30) +
  geom_vline(xintercept = textdata$xint, color = "red") +
  geom_vline(xintercept = textdata$medint, color = "blue") +
  geom_text(data = textdata, x = xint, y = 35, label = "Mean", size = 3, angle = 90, vjust = 1.25) +
  geom_text(data = textdata, x = medint, y = 35, label = "Median", size = 3, angle = 90, vjust = 1.25) +
  labs(title = "Density Plot of GDP per Capita",
       x = "GDP per Capita",
       y = "Density")
```



Here we see that the median is farther to the left (lower) than the mean. So that means that there are some serious outliers on the right – on the richer side. Most countries are grouped around lower GDP per capitas, but there are a few countries that are significantly higher.

1e) Write a function that returns the amount of GDP data that falls within 1, 1.96, and 3 standard deviations of the mean. Compare these results with what we would expect if the data were perfectly normally distributed.

```
# Created new column for standard deviation

demo1 <- demo %>%
  mutate("sd" = (GDP - mean(GDP)))

# Set up my function. I use absolute value to show the +/- aspect.

gdpdeviation <- function(data){

initial1 <- data %>%
    dplyr::filter(abs(sd) <= abs(sd(GDP)))
initial2 <- data %>%
    dplyr::filter(abs(sd) <= 1.96*abs(sd(GDP)))
initial3 <- data %>%
    dplyr::filter(abs(sd) <= 3*abs(sd(GDP)))

standard <- nrow(initial1)/nrow(data)
midstandard <- nrow(initial2)/nrow(data)
largestandard <- nrow(initial3)/nrow(data)

c(standard, midstandard, largestandard)

}

gdpdeviation(demo1)
```

```
## [1] 0.8322981 0.9192547 0.9937888
```

In a perfect normal distribution, 68% of the population falls within 1 standard deviation of the mean, while 95% falls within 1.96 standard deviations and 99.7% falls within 3 standard deviations. In this data, it's 83%, 92%, and 99.4%. The 99.4% figure is quite close, 92% is close, and 83% is relatively far away. So the data look rather different here than in a perfect normal distribution.

1f) Now draw 100, 1,000, and 10,000 samples the length of the dataframe, with replacement, from the GDP data (bootstrap). Plot a histogram of the sample means (you can use the above ggplot() code or the hist() function). How well do these sampling distributions approximate the normal distribution? How close are they to the mean value of GDP?

First, I set everything up.

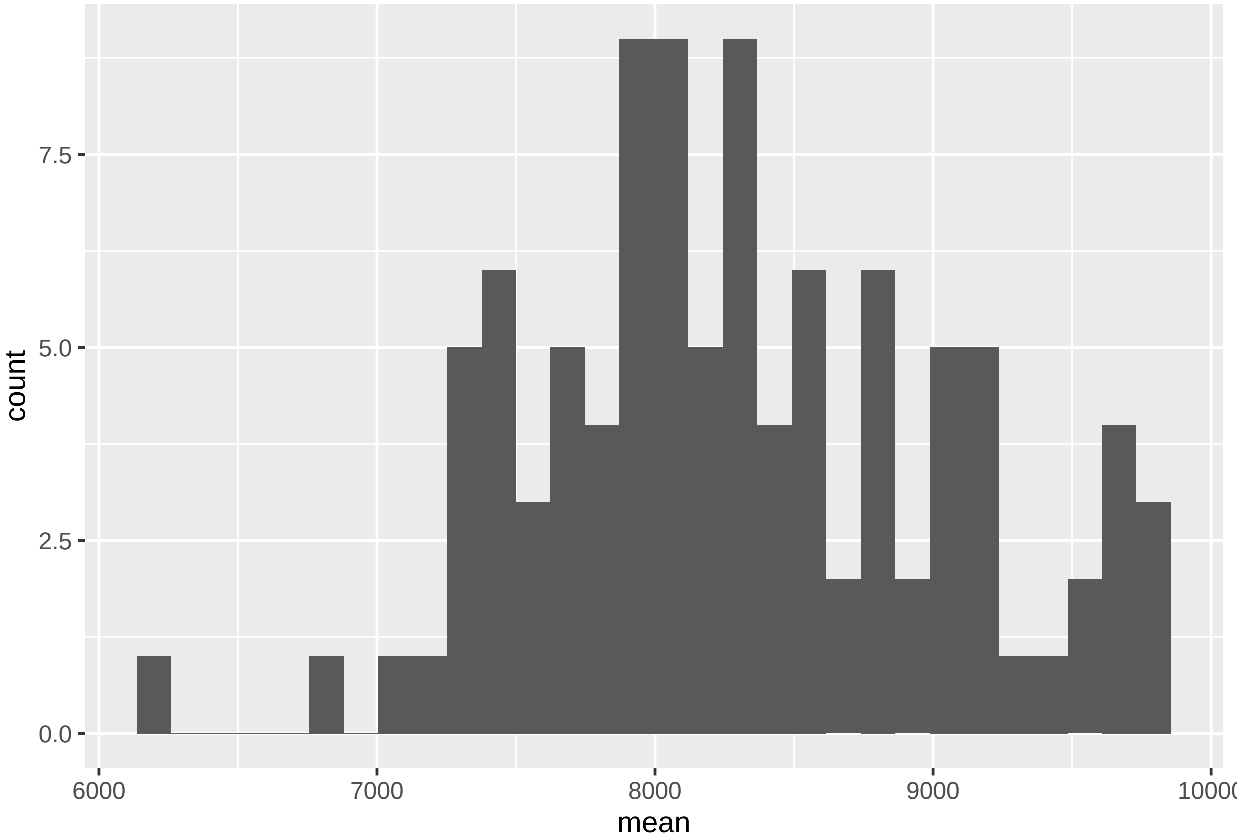
```
demonew <- demo %>%
  select(GDP)

# Not sure if this is a popular choice, but I've used the rep_sample_n() function from
# the infer package before for bootstrapping -- used it again here.

first <- demonew %>% rep_sample_n(size = 161, replace = TRUE, reps = 100)
```

```r
# replace = TRUE ensures that we are bootstrapping.

second <- demonew %>% rep_sample_n(size = 161, replace = TRUE, reps = 1000)
third <- demonew %>% rep_sample_n(size = 161, replace = TRUE, reps = 10000)

# Here, I used nesting and mapping to apply the mean function to all.

firstmean <- first %>%
  group_by(replicate) %>%
  nest() %>%
  mutate(mean = map(.x = data, .f = ~mean(.x$GDP, na.rm = TRUE))) %>%
  select(-data) %>%
  ungroup()

secondmean <- second %>%
  group_by(replicate) %>%
  nest() %>%
  mutate(mean = map(.x = data, .f = ~mean(.x$GDP, na.rm = TRUE))) %>%
  select(-data) %>%
  ungroup()

thirdmean <- third %>%
  group_by(replicate) %>%
  nest() %>%
  mutate(mean = map(.x = data, .f = ~mean(.x$GDP, na.rm = TRUE))) %>%
  select(-data) %>%
  ungroup()
```

Next, I create histograms.

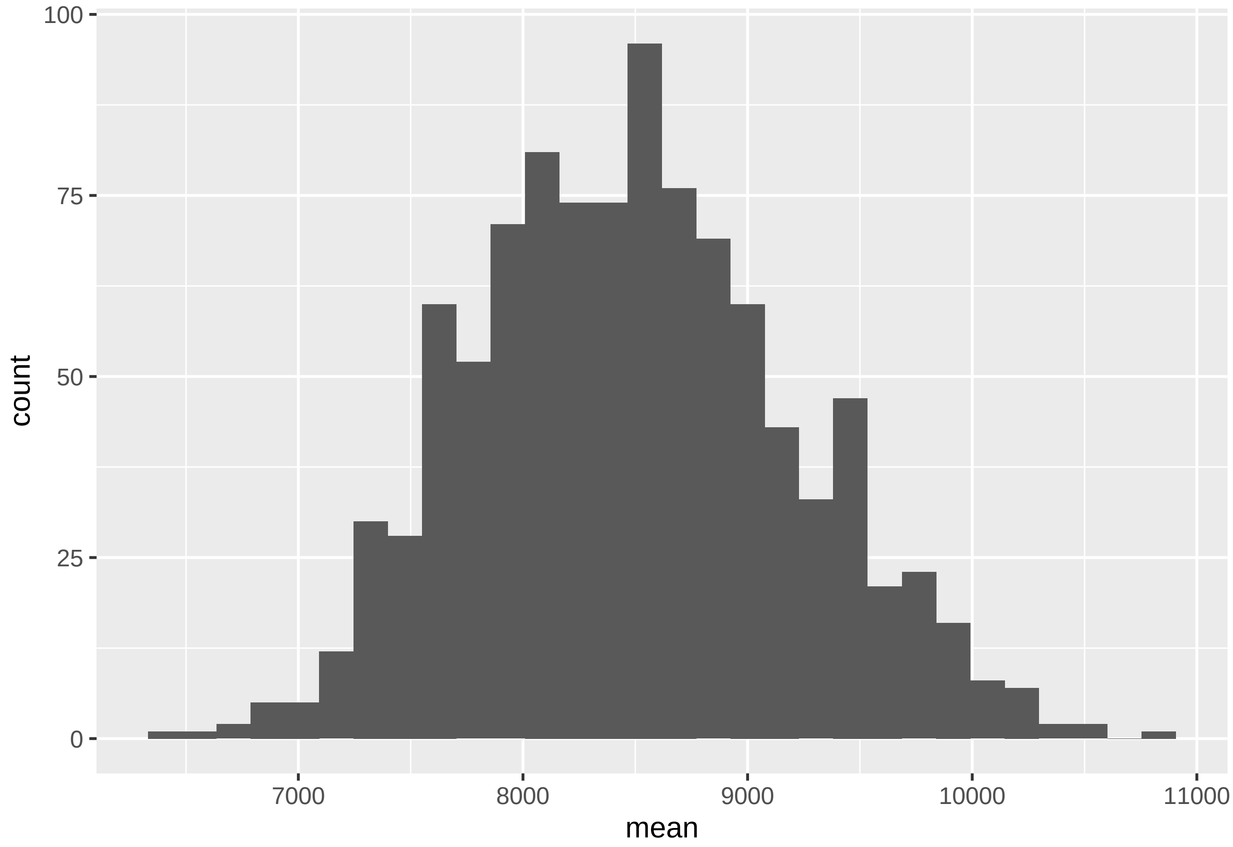
```r
# First, I need to set columns as numeric -- otherwise I will get errors.

firstmean$mean <- as.numeric(firstmean$mean)
secondmean$mean <- as.numeric(secondmean$mean)
thirdmean$mean <- as.numeric(thirdmean$mean)

# Now, for plots.

ggplot(firstmean, aes(x = mean)) +
  geom_histogram()
```

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



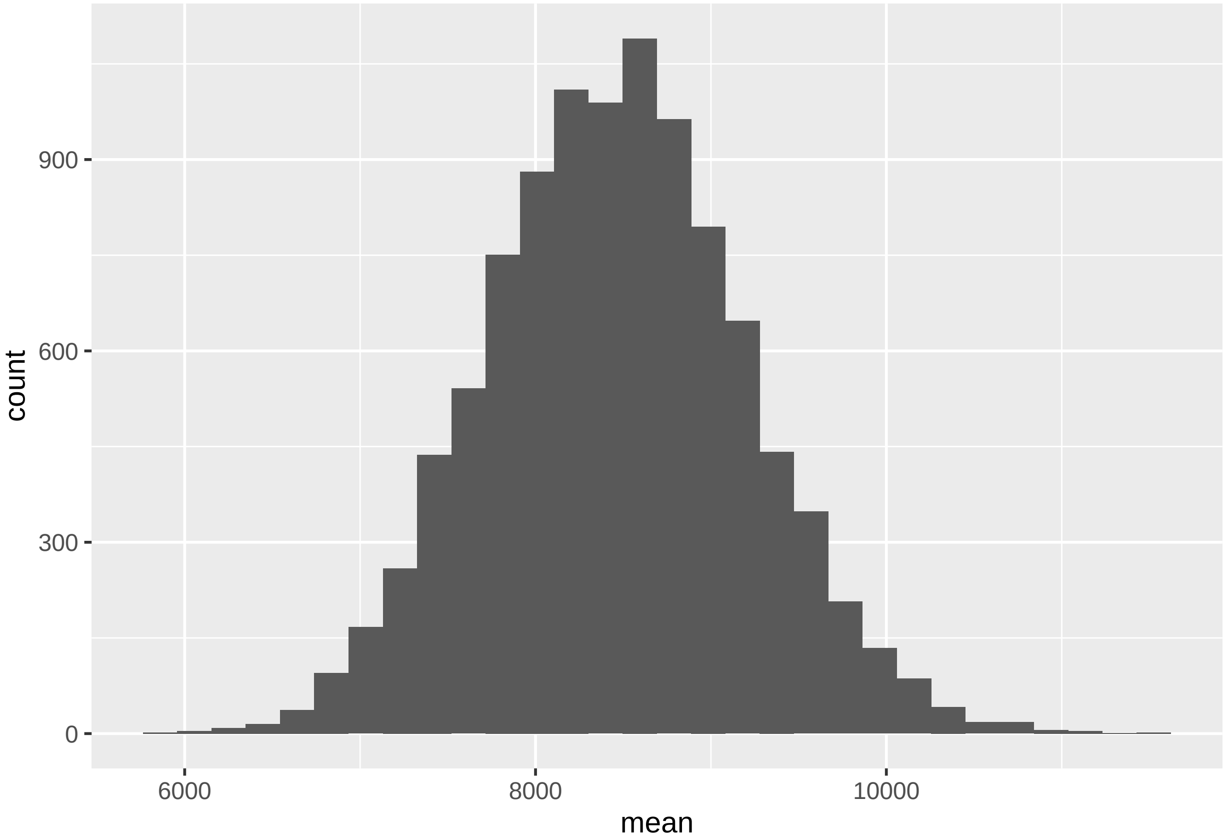
```
ggplot(secondmean, aes(x = mean)) +
  geom_histogram()
```

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



```
ggplot(thirdmean, aes(x = mean)) +
  geom_histogram()
```

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



So after the histograms, we see that the higher the number of samples, the more even the distribution of means is.

Now, I want to look at the standard deviation and see how well the data approximates the normal distribution. Just looking at the graphs, they certainly look much closer to the normal distribution, because they have more data. This is due to the Central Limit Theorem. To see if this bears true in the data, I run my gdpdeviation() function that I created earlier. As we can see below, the distributions of the means do tend to get closer and closer to the normal distribution. Additionally, the means do tend to be rather close to the initial mean, which was 8468.6583851. We can see this graphically and in the tables.

Below is my code.

```
# I have to change the names of the "mean" column for it to work in my function.

setnames(firstmean, "mean", "GDP")
setnames(secondmean, "mean", "GDP")
setnames(thirdmean, "mean", "GDP")
```

```
# Here, I will use my gdpdeviation() function. I have to create a SD column for each.

firstmean1 <- firstmean %>%
  mutate("sd" = (GDP - mean(GDP)))

secondmean1 <- secondmean %>%
  mutate("sd" = (GDP - mean(GDP)))

thirdmean1 <- thirdmean %>%
  mutate("sd" = (GDP - mean(GDP)))

gdpdeviation(firstmean1)
```

```
## [1] 0.63 0.95 1.00
```

```
gdpdeviation(secondmean1)
```

```
## [1] 0.665 0.957 0.999
```

```
gdpdeviation(thirdmean1)
```

```
## [1] 0.6856 0.9498 0.9963
```

1g) Calculate the standard error of the mean, using the mathematical formula from lecture and using the three sampling distributions you created above. Explain what the standard error means here.

Here, I take the three samples, labeled first, second, and third, and find the standard errors of all three.

```
se1 <- sd(first$GDP)/sqrt(nrow(first))
se2 <- sd(second$GDP)/sqrt(nrow(second))
se3 <- sd(third$GDP)/sqrt(nrow(third))

se1
```

```
## [1] 73.67602
```

```
se2
```

```
## [1] 23.69746
```

```
se3
```

```
## [1] 7.466441
```

Standard error is the standard deviation of a sampling distribution. It shows how different the population is likely to be from a given sample. The lower the standard error, the more representative the sample is. Honestly, all three of these are relatively decent, because while 73 technically looks high, when we remember that we're dealing with numbers well into the thousands, 73 could be a lot worse. But, notably, as we sample more and more, the standard error decreases to the point that it is at just around 7.5 for the last one – a tiny (not a technical term!) standard of error considering the size of the numbers we're dealing with.

Question 2 Background:

Imagine that every person in the United States has a fixed preference for redistribution, which is defined as a continuous variable, where smaller values mean that individual favors less redistribution, and larger values mean they agree with more redistribution. Let N equal the size of the entire population of the US. Let Y be the vector (of length N) of preferences for each person in the population, such that Y_i refers to the level of this political stance of person i. We will treat this as a fixed attribute of individual i. As political scientists, we would like to make inferences about the aggregated preference for redistributive policies of the entire population. Suppose that the average population view is μ and the variance is σ^2.

We can't go out and measure every single person's view on this issue, so instead, we're going to randomly sample n people from the population, and measure their preferences for redustribution (imagine for now that every person who is sampled responds, and that we measure their views correctly each time). Each person will either be sampled or not, with a constant probability p between 0 and 1.[2] *S is an indicator variable for being sampled, so S_i is equal to 1 if individual i was sampled, and 0 otherwise. Let $\hat{\mu}$ be the sample mean, $\frac{\sum_{i=1}^{N} S_i \cdot Y_i}{n}$. The "hat'' notation $(\hat{\cdot})$ indicates that we are estimating some property of the population $(\cdot)$ using our sample.*

2a) Conceptually, what does $E[Y]$ refer to?

$E[Y]$ is the expected value of Vector [Y]. The idea is that if you make infinite draws of Y_i with replacement, $E[Y]$ would be the mean value of those draws. We can expect that this value $E[Y]$ would be equal to μ, the population mean.

2b) Conceptually, what do $E[\hat{\mu}]$ and $Var[\hat{\mu}]$ refer to?

$E[\hat{\mu}]$ refers to the expected sample mean, while $Var[\hat{\mu}]$ refers do the variance of the sample mean.

2c) What is $Var[S_i]$, if $E[S_i] = \frac{n}{N}$?

It should be $\frac{n}{N} * (1 - \frac{n}{N})$. First, we know that $Var[S_i] = E[S_i^2] - E[S_i]^2$. This gives us $E[S_i^2] - (\frac{n}{N})^2$. Then, we know the two possible values for $S_i - 1$ and 0. $1^2 = 1$ and $0^2 = 0$, so $S_i^2 = S_i$. So now we get $\frac{n}{N} - \frac{n}{N}^2$, which can also be written as $\frac{n}{N} * (1 - \frac{n}{N})$.

2d) What is $E[\hat{\mu}]$? (hint: remember that S_i and Y_i are independent.)

$E[\hat{\mu}]$ should be equal to μ. We know this logically, but we can prove it as well. The proof:

$E[\hat{\mu}] = E[\sum_{i=1}^{N} \frac{S_i Y_i}{n}]$

$= \frac{1}{n} * \sum_{i=1}^{N} E[S_i Y_i]$

$= \frac{1}{n} * \sum_{i=1}^{N} E[S_i]E[Y_i]$

$= \frac{1}{n} * \frac{n}{N} * \sum_{i=1}^{N} E[Y_i]$

$= \frac{1}{N} * N * \mu$

$= \mu$

[2] Technically, if we're sampling exactly n people without replacement, then the observations are not independent—when the n-th person joins the sample, that means that all subsequent individuals must be excluded—but we'll neglect this for now and assume that they are independent.

2e) If S_i and Y_i weren't independent, would we necessarily get the same result as in 4.d? Practically, what does this mean for surveys? Can you describe a scenario where S_i and Y_i wouldn't be independent?

Not necessarily. S_i and Y_i not being independent imply that there could be bias in the sample, so the sample would potentially not be representative of the sample as a whole. Ergo, the sample mean ($\hat{\mu}$) may also not be equal to μ.

An example of this could be a survey that the organizers decided to run on Instagram. While such a survey could reach a wide audience, 50% of Instagram users are between the ages of 25-34, meaning that any results would certainly not be representative of the population as a whole.