DATA 606 Lab 5a: Statistical inference: sampling distributions

Kevin Kirby

In this lab, you will investigate the ways in which the statistics from a random sample of data can serve as point estimates for population parameters. We're interested in formulating a *sampling distribution* of our estimate in order to learn about the properties of the estimate, such as its distribution.

Setting a seed: We will take some random samples and build sampling distributions in this lab, which means you should set a seed at the start of your lab. If this concept is new to you, review the lab on probability.

Getting Started

Load packages

In this lab, we will explore and visualize the data using the **tidyverse** suite of packages. We will also use the **infer** package for resampling.

Let's load the packages.

```
library(tidyverse)
library(openintro)
library(infer)
```

The data

A 2019 Gallup report states the following:

> The premise that scientific progress benefits people has been embodied in discoveries throughout the ages – from the development of vaccinations to the explosion of technology in the past few decades, resulting in billions of supercomputers now resting in the hands and pockets of people worldwide. Still, not everyone around the world feels science benefits them personally.
>
> **Source:** World Science Day: Is Knowledge Power?

The Wellcome Global Monitor finds that 20% of people globally do not believe that the work scientists do benefits people like them. In this lab, you will assume this 20% is a true population proportion and learn about how sample proportions can vary from sample to sample by taking smaller samples from the population. We will first create our population assuming a population size of 100,000. This means 20,000 (20%) of the population think the work scientists do does not benefit them personally and the remaining 80,000 think it does.

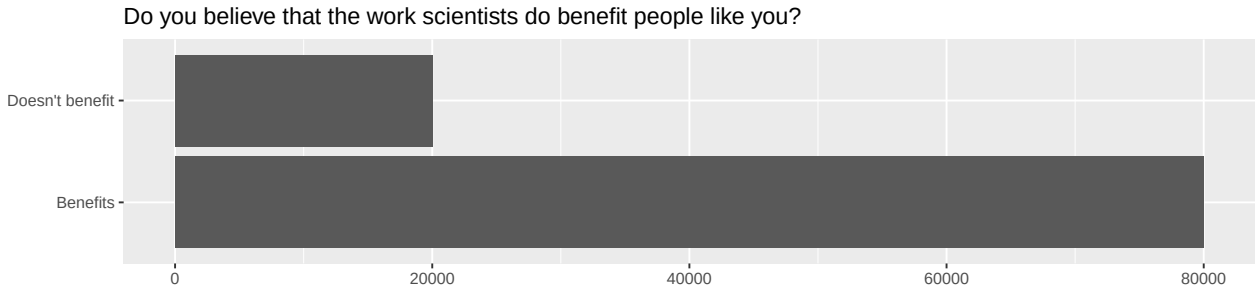
```
global_monitor <- tibble(
  scientist_work = c(rep("Benefits", 80000), rep("Doesn't benefit", 20000))
)
```

The name of the data frame is `global_monitor` and the name of the variable that contains responses to the question *"Do you believe that the work scientists do benefit people like you?"* is `scientist_work`.

We can quickly visualize the distribution of these responses using a bar plot.

```
ggplot(global_monitor, aes(x = scientist_work)) +
  geom_bar() +
```

```
  labs(
    x = "", y = "",
    title = "Do you believe that the work scientists do benefit people like you?"
  ) +
  coord_flip()
```



We can also obtain summary statistics to confirm we constructed the data frame correctly.

```
global_monitor %>%
  count(scientist_work) %>%
  mutate(p = n /sum(n))
```

```
## # A tibble: 2 x 3
##   scientist_work      n     p
##   <chr>           <int> <dbl>
## 1 Benefits        80000   0.8
## 2 Doesn't benefit 20000   0.2
```

The unknown sampling distribution

In this lab, you have access to the entire population, but this is rarely the case in real life. Gathering information on an entire population is often extremely costly or impossible. Because of this, we often take a sample of the population and use that to understand the properties of the population.

If you are interested in estimating the proportion of people who don't think the work scientists do benefits them, you can use the `sample_n` command to survey the population.

```
samp1 <- global_monitor %>%
  sample_n(50)
```

This command collects a simple random sample of size 50 from the `global_monitor` dataset, and assigns the result to `samp1`. This is similar to randomly drawing names from a hat that contains the names of all in the population. Working with these 50 names is considerably simpler than working with all 100,000 people in the population.

1. Describe the distribution of responses in this sample. How does it compare to the distribution of responses in the population. **Hint:** Although the `sample_n` function takes a random sample of observations (i.e. rows) from the dataset, you can still refer to the variables in the dataset with the same names. Code you presented earlier for visualizing and summarizing the population data will still be useful for the sample, however be careful to not label your proportion `p` since you're now calculating a sample statistic, not a population parameters. You can customize the label of the statistics to indicate that it comes from the sample.

Answer:

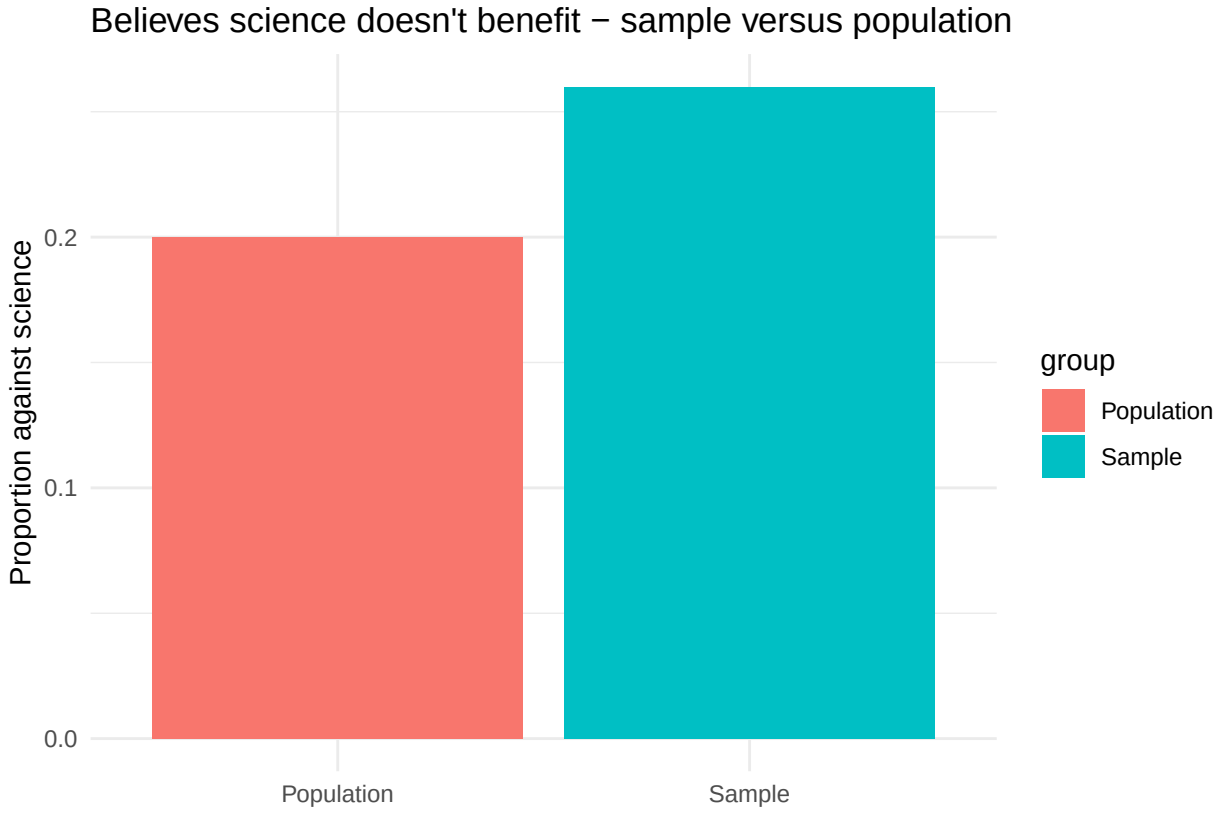
The sample has 26% saying that scientist's work doesn't benefit them, versus 20% in the population overall. In both cases, we have a shocking number of people saying scientist's work doesn't benefit them. However, I'm honestly surprised it's not higher. I would expect this number to shift closer and closer to partisan divides.

```r
science_bad_sample <- samp1 %>%
  summarize(prop_sample = mean(scientist_work == "Doesn't benefit"))

science_bad_pop <- global_monitor %>%
  summarize(prop_population = mean(scientist_work == "Doesn't benefit"))

sample_vs_population <- data.frame(
  group = c("Sample", "Population"),
  proportion = c(science_bad_sample$prop_sample, science_bad_pop$prop_population)
)

ggplot(sample_vs_population, aes(x = group, y = proportion, fill = group)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Believes science doesn't benefit - sample versus population",
       y = "Proportion against science",
       x = "") +
  theme_minimal()
```




```r
sample_props <- samp1 %>%
  summarize(
    prop_sample_benefits = mean(scientist_work == "Benefits"),
    prop_sample_doesnt_benefit = mean(scientist_work == "Doesn't benefit")
  )

pop_props <- global_monitor %>%
  summarize(
    prop_population_benefits = mean(scientist_work == "Benefits"),
    prop_population_doesnt_benefit = mean(scientist_work == "Doesn't benefit")
```

```
  )

cat("sample proportions: \n")
```

```
## sample proportions:
```

```
sample_props
```

```
## # A tibble: 1 x 2
##   prop_sample_benefits prop_sample_doesnt_benefit
##                  <dbl>                      <dbl>
## 1                 0.74                       0.26
```

```
cat("\npopulation proportions: \n")
```

```
##
## population proportions:
```

```
pop_props
```

```
## # A tibble: 1 x 2
##   prop_population_benefits prop_population_doesnt_benefit
##                      <dbl>                          <dbl>
## 1                      0.8                            0.2
```

If you're interested in estimating the proportion of all people who do not believe that the work scientists do benefits them, but you do not have access to the population data, your best single guess is the sample mean.

```
samp1 %>%
  count(scientist_work) %>%
  mutate(p_hat = n /sum(n))
```

```
## # A tibble: 2 x 3
##   scientist_work      n p_hat
##   <chr>           <int> <dbl>
## 1 Benefits           37  0.74
## 2 Doesn't benefit    13  0.26
```

Depending on which 50 people you selected, your estimate could be a bit above or a bit below the true population proportion of 0.26. In general, though, the sample proportion turns out to be a pretty good estimate of the true population proportion, and you were able to get it by sampling less than 1% of the population.

2. Would you expect the sample proportion to match the sample proportion of another student's sample? Why, or why not? If the answer is no, would you expect the proportions to be somewhat different or very different? Ask a student team to confirm your answer.

Answer: I would expect different students to have slightly different sample proportions, although within a small range (maybe .18 to .28). This class has enough students that, through random chance, any two students could end up with the same sample proportions. However, there's a much smaller chance that it was the exact same sample members.

3. Take a second sample, also of size 50, and call it `samp2`. How does the sample proportion of `samp2` compare with that of `samp1`? Suppose we took two more samples, one of size 100 and one of size 1000. Which would you think would provide a more accurate estimate of the population proportion?

Answer: Lol, I ended up with a samp2 proportion of .16% not believing in the work of scientists, which is conveniently below what I confidently predicted as the possible ranges above. This sample proportion is also 10 percentage points below samp1.

I would expecr the larger sample size th provide the more accurate estimate, all else being equal. As the sample size approaches the population size, the sample proportions should have smaller and smaller deviations away from the population. The big IF here, though, is that the sample continues to represent the underlying population characteristics. If your population is 70% male but you only sample females while presenting it as a population sample, your numbers will look off.

```
samp2 <- global_monitor %>%
  sample_n(50)

samp2 %>%
  count(scientist_work) %>%
  mutate(p_hat = n /sum(n))
```

```
## # A tibble: 2 x 3
##   scientist_work      n p_hat
##   <chr>           <int> <dbl>
## 1 Benefits           42  0.84
## 2 Doesn't benefit     8  0.16
```

```
samp2_p_hat <- samp2 %>%
  count(scientist_work) %>%
  mutate(p_hat = n /sum(n)) %>%
  filter(scientist_work == "Doesn't benefit") %>%
  pull(p_hat) %>%
  round(2)
```

Not surprisingly, every time you take another random sample, you might get a different sample proportion. It's useful to get a sense of just how much variability you should expect when estimating the population mean this way. The distribution of sample proportions, called the *sampling distribution (of the proportion)*, can help you understand this variability. In this lab, because you have access to the population, you can build up the sampling distribution for the sample proportion by repeating the above steps many times. Here, we use R to take 15,000 different samples of size 50 from the population, calculate the proportion of responses in each sample, filter for only the *Doesn't benefit* responses, and store each result in a vector called `sample_props50`. Note that we specify that `replace = TRUE` since sampling distributions are constructed by sampling with replacement.

```
sample_props50 <- global_monitor %>%
                    rep_sample_n(size = 50, reps = 15000, replace = TRUE) %>%
                    count(scientist_work) %>%
                    mutate(p_hat = n /sum(n)) %>%
                    filter(scientist_work == "Doesn't benefit")
```

And we can visualize the distribution of these proportions with a histogram.

```
ggplot(data = sample_props50, aes(x = p_hat)) +
  geom_histogram(binwidth = 0.02) +
  labs(
    x = "p_hat (Doesn't benefit)",
    title = "Sampling distribution of p_hat",
    subtitle = "Sample size = 50, Number of samples = 15000"
  )
```

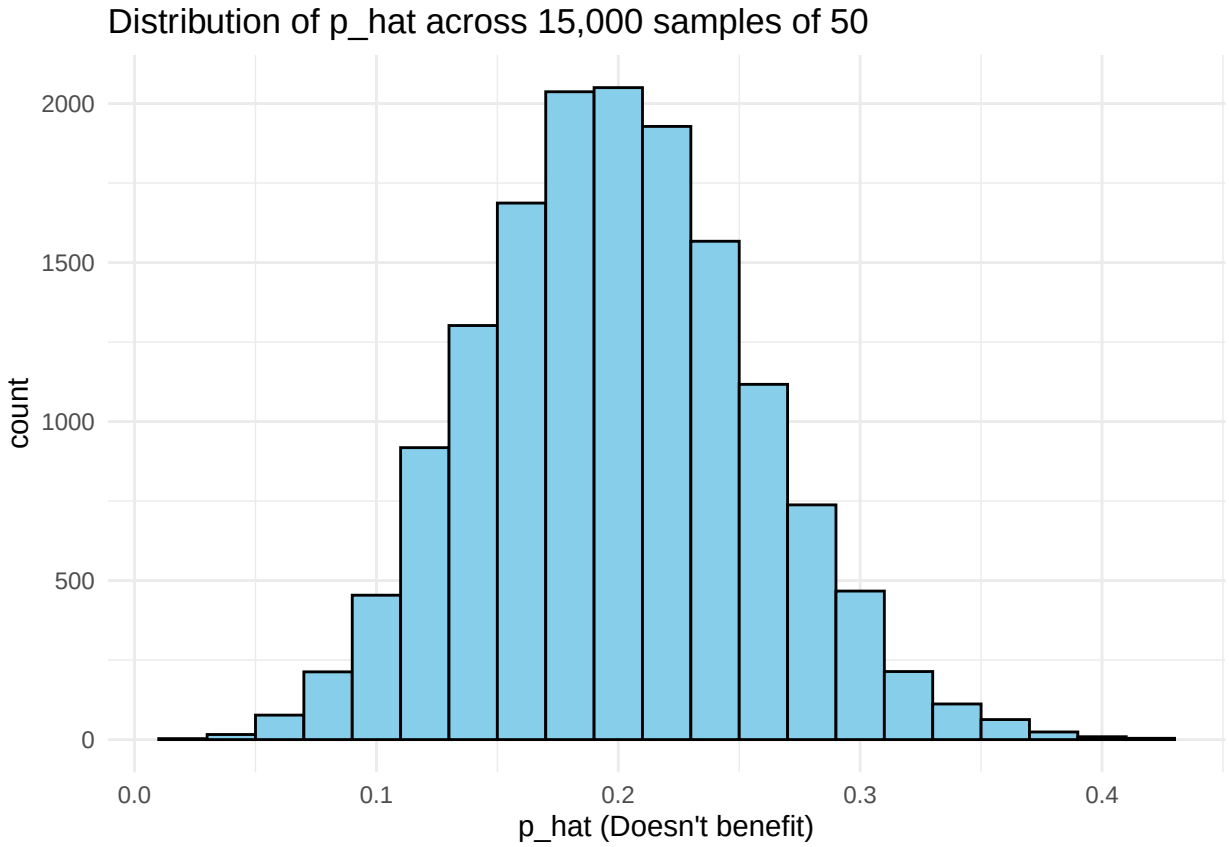
Next, you will review how this set of code works.

4. How many elements are there in `sample_props50`? Describe the sampling distribution, and be sure to specifically note its center. Make sure to include a plot of the distribution in your answer.

Answer: The histogram shows the proportion of respondents in the sample that believe scientists work does

not benefit them; it's based on 15,000 samples of 50 results. The distribution is close to normal, with a center peak around .2, which is good because the known population proportion is .2. As I got close to predicting above, a lot of the proportions fall between 0.1 and 0.3.

```
ggplot(data = sample_props50, aes(x = p_hat)) +
  geom_histogram(binwidth = 0.02, fill = "skyblue", color = "black") +
  labs(
    x = "p_hat (Doesn't benefit)",
    y = "count",
    title = "Distribution of p_hat across 15,000 samples of 50"
  ) +
  theme_minimal()
```



Interlude: Sampling distributions

The idea behind the `rep_sample_n` function is *repetition*. Earlier, you took a single sample of size `n` (50) from the population of all people in the population. With this new function, you can repeat this sampling procedure `rep` times in order to build a distribution of a series of sample statistics, which is called the **sampling distribution**.

Note that in practice one rarely gets to build true sampling distributions, because one rarely has access to data from the entire population.

Without the `rep_sample_n` function, this would be painful. We would have to manually run the following code 15,000 times

```
global_monitor %>%
  sample_n(size = 50, replace = TRUE) %>%
  count(scientist_work) %>%
```

```
  mutate(p_hat = n /sum(n)) %>%
  filter(scientist_work == "Doesn't benefit")
```

```
## # A tibble: 1 x 3
##   scientist_work      n p_hat
##   <chr>           <int> <dbl>
## 1 Doesn't benefit     6  0.12
```

as well as store the resulting sample proportions each time in a separate vector.

Note that for each of the 15,000 times we computed a proportion, we did so from a **different** sample!

5. To make sure you understand how sampling distributions are built, and exactly what the `rep_sample_n` function does, try modifying the code to create a sampling distribution of **25 sample proportions** from **samples of size 10**, and put them in a data frame named `sample_props_small`. Print the output. How many observations are there in this object called `sample_props_small`? What does each observation represent?

Answer: Each observation is a proportion from a sample of 10 people, with only proportions with non-zero values surfacing in sample_props_small. This leads to 23 observations that print out, with 2 that didn't yield a "doesn't benefit" result in the sample.

```
sample_props_small <- global_monitor %>%
  rep_sample_n(size = 10, reps = 25, replace = TRUE) %>%
  count(scientist_work) %>%
  mutate(p_hat = n / sum(n)) %>%
  filter(scientist_work == "Doesn't benefit")

sample_props_small
```

```
## # A tibble: 23 x 4
## # Groups:   replicate [23]
##    replicate scientist_work      n p_hat
##        <int> <chr>           <int> <dbl>
##  1         1 Doesn't benefit     2   0.2
##  2         2 Doesn't benefit     2   0.2
##  3         3 Doesn't benefit     1   0.1
##  4         4 Doesn't benefit     3   0.3
##  5         5 Doesn't benefit     1   0.1
##  6         6 Doesn't benefit     1   0.1
##  7         8 Doesn't benefit     2   0.2
##  8         9 Doesn't benefit     2   0.2
##  9        10 Doesn't benefit     1   0.1
## 10        11 Doesn't benefit     1   0.1
## # i 13 more rows
```

Sample size and the sampling distribution

Mechanics aside, let's return to the reason we used the `rep_sample_n` function: to compute a sampling distribution, specifically, the sampling distribution of the proportions from samples of 50 people.

```
ggplot(data = sample_props50, aes(x = p_hat)) +
  geom_histogram(binwidth = 0.02)
```

The sampling distribution that you computed tells you much about estimating the true proportion of people who think that the work scientists do doesn't benefit them. Because the sample proportion is an unbiased estimator, the sampling distribution is centered at the true population proportion, and the spread of the

distribution indicates how much variability is incurred by sampling only 50 people at a time from the population.

In the remainder of this section, you will work on getting a sense of the effect that sample size has on your sampling distribution.

6. Use the app below to create sampling distributions of proportions of *Doesn't benefit* from samples of size 10, 50, and 100. Use 5,000 simulations. What does each observation in the sampling distribution represent? How does the mean, standard error, and shape of the sampling distribution change as the sample size increases? How (if at all) do these values change if you increase the number of simulations? (You do not need to include plots in your answer.)

Answer: I used "Number of samples" = 5000 for all I'm about to write:

Each observation is one full cycle of the sample size. If the sample size is ten, then each observation is a random sample of ten people.

Increasing from sample size 10 to 50 lead to noticeably more filled out histogram that has a normal-ish curve. The mean decreased from .22 to .2 while the standard error was cut in half, from .11 to .06. A lot of benefit for a relatively small change.

Increasing from 50 to 100 lead to a tighter normal curve but with a mean still around .2. The standard error did drop another 33%, down to .04 from .06

Overall, sample size ten is too small, with the curve sparse the data potentially heavily skewed. Depending on the dataset, I would aim for 1% sample overall, with that giving way to a raw goal when the overall population is small enough.

More Practice

So far, you have only focused on estimating the proportion of those you think the work scientists doesn't benefit them. Now, you'll try to estimate the proportion of those who think it does.

Note that while you might be able to answer some of these questions using the app, you are expected to write the required code and produce the necessary plots and summary statistics. You are welcome to use the app for exploration.

7. Take a sample of size 15 from the population and calculate the proportion of people in this sample who think the work scientists do enhances their lives. Using this sample, what is your best point estimate of the population proportion of people who think the work scientists do enhances their lives?

Answer: The code is below. The proportion of people who believe the work of scientists benefit them from my sample of 15 was .73. Using this value alone, I would guesa the population proportion is around .75 or so.

```
samp15 <- global_monitor %>%
  sample_n(15)

samp15 %>%
  count(scientist_work) %>%
  mutate(p_hat = n /sum(n))
```

```
## # A tibble: 2 x 3
##   scientist_work      n p_hat
##   <chr>           <int> <dbl>
## 1 Benefits           10 0.667
## 2 Doesn't benefit     5 0.333
```

```
samp15_p_hat <- samp15 %>%
  count(scientist_work) %>%
```

```
  mutate(p_hat = n /sum(n)) %>%
  filter(scientist_work == "Benefits") %>%
  pull(p_hat) %>%
  round(2)
```

8. Since you have access to the population, simulate the sampling distribution of proportion of those who think the work scientists do enchances their lives for samples of size 15 by taking 2000 samples from the population of size 15 and computing 2000 sample proportions. Store these proportions in as `sample_props15`. Plot the data, then describe the shape of this sampling distribution. Based on this sampling distribution, what would you guess the true proportion of those who think the work scientists do enchances their lives to be? Finally, calculate and report the population proportion.

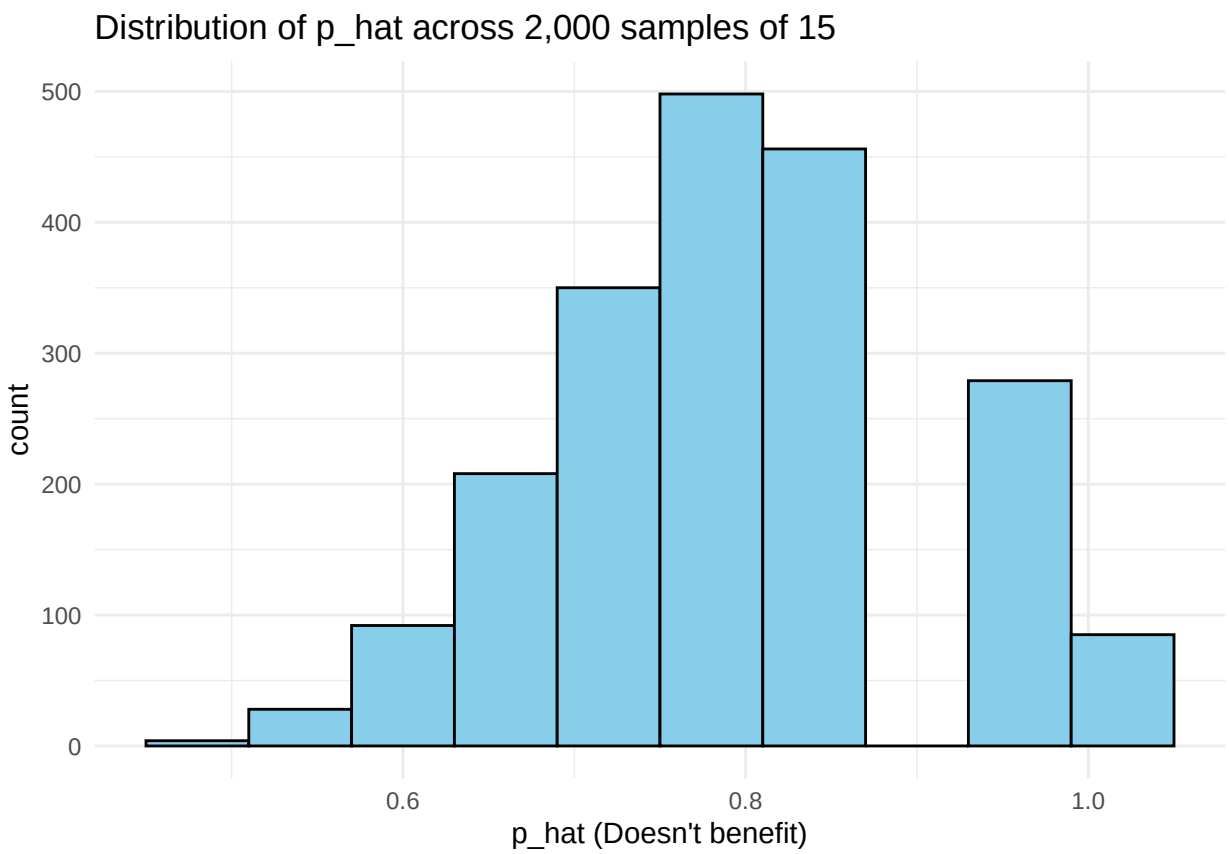
Answer: The shape of the distribution is unreliable but showing signs of a normal curve. Depending how aggressive you're willing to be with the bid width, you could find a very random and specific value that completely fills in the gaps. Based on this sampling distribution, I would guess the population proportion is around .78.

```
sample_props15 <- global_monitor %>%
  rep_sample_n(size = 15, reps = 2000, replace = TRUE) %>%
  count(scientist_work) %>%
  mutate(p_hat = n / sum(n)) %>%
  filter(scientist_work == "Benefits")

sample_props15
```

```
## # A tibble: 2,000 x 4
## # Groups:   replicate [2,000]
##    replicate scientist_work     n p_hat
##        <int> <chr>          <int> <dbl>
##  1         1 Benefits          11 0.733
##  2         2 Benefits          13 0.867
##  3         3 Benefits          13 0.867
##  4         4 Benefits          13 0.867
##  5         5 Benefits          10 0.667
##  6         6 Benefits          13 0.867
##  7         7 Benefits          14 0.933
##  8         8 Benefits          11 0.733
##  9         9 Benefits          14 0.933
## 10        10 Benefits          12 0.8
## # i 1,990 more rows
```

```
ggplot(data = sample_props15, aes(x = p_hat)) +
  geom_histogram(binwidth = 0.06, fill = "skyblue", color = "black") +
  labs(
    x = "p_hat (Doesn't benefit)",
    y = "count",
    title = "Distribution of p_hat across 2,000 samples of 15"
  ) +
  theme_minimal()
```



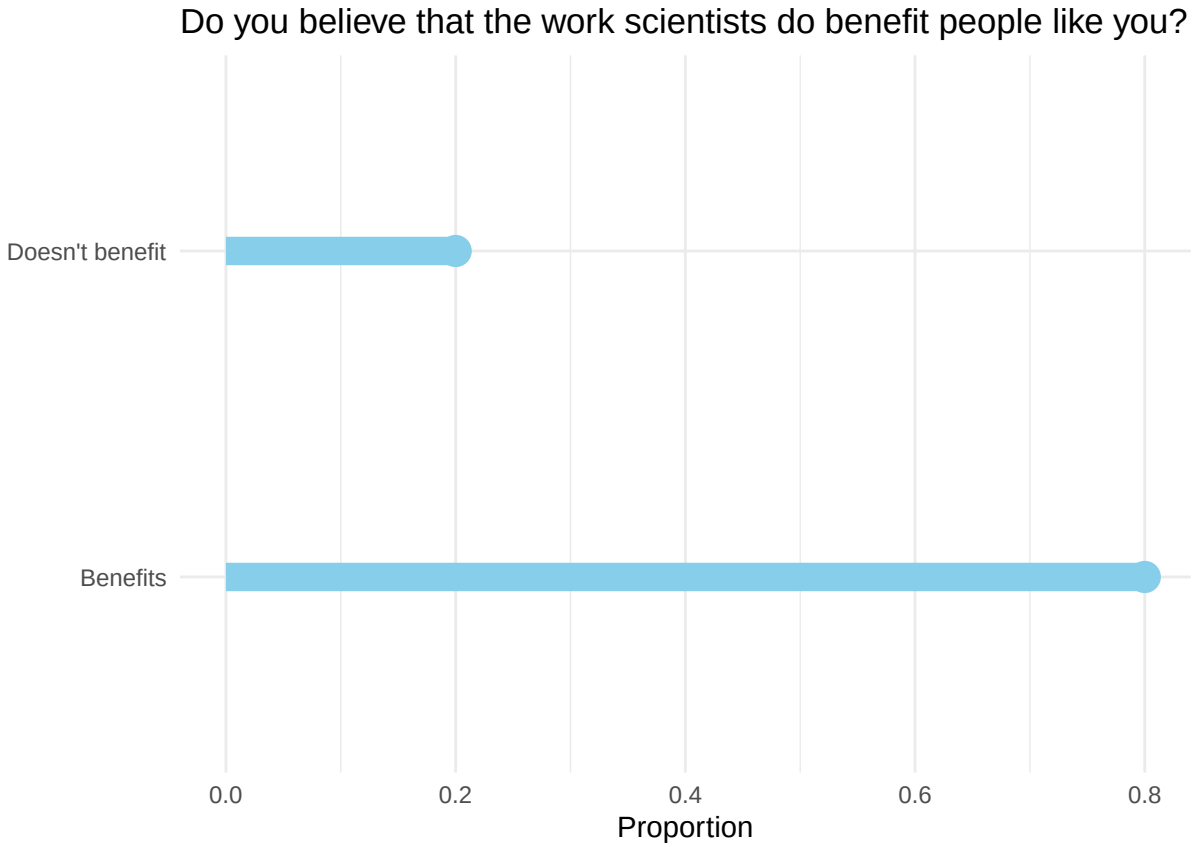
Plotting the overall population: The overall proportion who believe the work of scientists benefits them 0.8.

```
global_science_reality <- global_monitor %>%
  group_by(scientist_work) %>%
  summarise(count = n()) %>%
  mutate(proportion = count / sum(count))

global_science_reality
```

```
## # A tibble: 2 x 3
##   scientist_work  count proportion
##   <chr>           <int>      <dbl>
## 1 Benefits        80000        0.8
## 2 Doesn't benefit 20000        0.2
```

```
ggplot(global_science_reality, aes(x = scientist_work, y = proportion)) +
  geom_point(size = 5, color = "skyblue") +
  geom_segment(aes(x = scientist_work, xend = scientist_work, y = 0, yend = proportion), color = "skybl
  coord_flip() +
  labs(
    x = "",
    y = "Proportion",
    title = "Do you believe that the work scientists do benefit people like you?"
  ) +
  theme_minimal()
```



9. Change your sample size from 15 to 150, then compute the sampling distribution using the same method as above, and store these proportions in a new object called `sample_props150`. Describe the shape of this sampling distribution and compare it to the sampling distribution for a sample size of 15. Based on this sampling distribution, what would you guess to be the true proportion of those who think the work scientists do enchances their lives?

Answer: The shape of this distribution is the closest to a perfect normal curve that I've seen all lab. Unlike with the 15 where there was a gap on downslope from the top towards the right tail, the 150 is basically perfectly proportioned. Based on this sampling distribution, I would guess the population proportion to be right on .8, with a general possible range of .77 to .83.

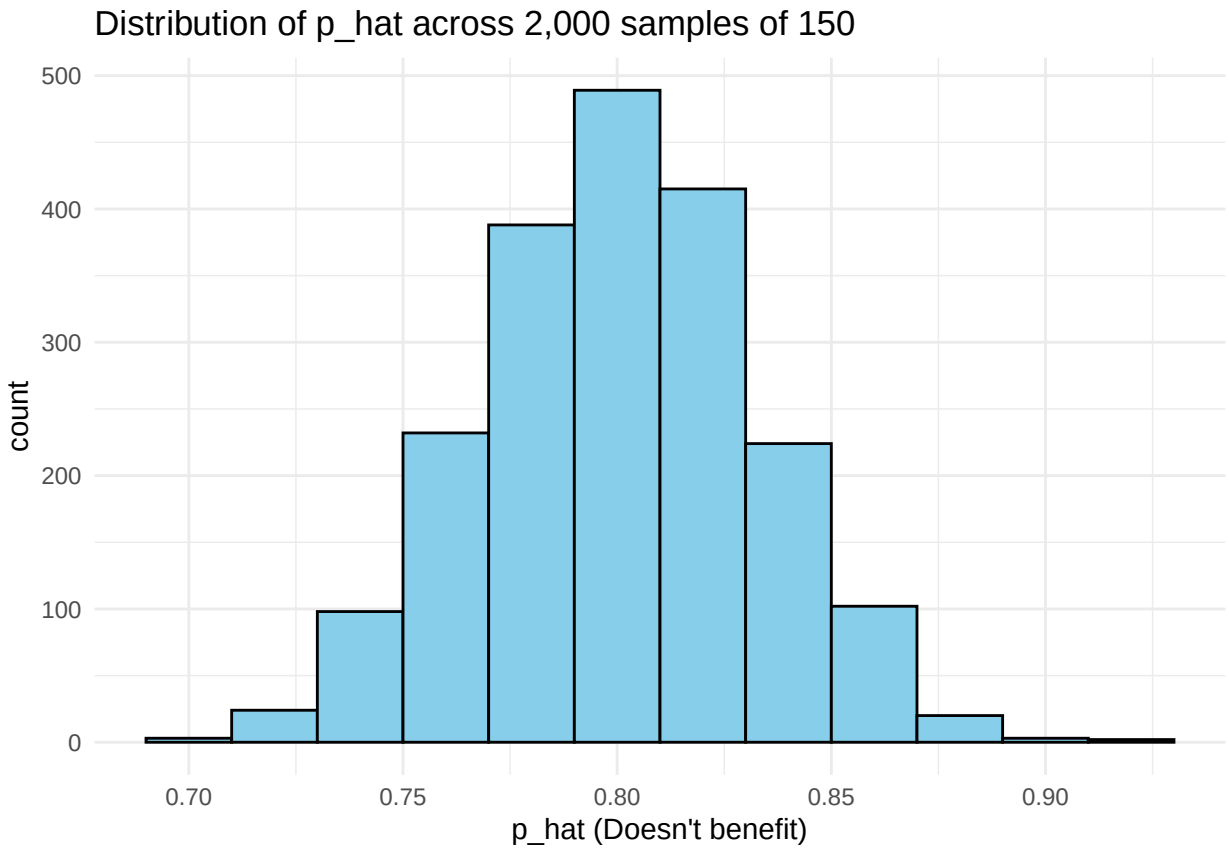
```
sample_props150 <- global_monitor %>%
  rep_sample_n(size = 150, reps = 2000, replace = TRUE) %>%
  count(scientist_work) %>%
  mutate(p_hat = n / sum(n)) %>%
  filter(scientist_work == "Benefits")

sample_props150
```

```
## # A tibble: 2,000 x 4
## # Groups:   replicate [2,000]
##    replicate scientist_work     n p_hat
##        <int> <chr>          <int> <dbl>
##  1         1 Benefits         129 0.86
##  2         2 Benefits         125 0.833
##  3         3 Benefits         124 0.827
##  4         4 Benefits         117 0.78
##  5         5 Benefits         128 0.853
```

```
##  6         6 Benefits         121 0.807
##  7         7 Benefits         114 0.76
##  8         8 Benefits         115 0.767
##  9         9 Benefits         125 0.833
## 10        10 Benefits         123 0.82
## # i 1,990 more rows
```

```r
ggplot(data = sample_props150, aes(x = p_hat)) +
  geom_histogram(binwidth = 0.02, fill = "skyblue", color = "black") +
  labs(
    x = "p_hat (Doesn't benefit)",
    y = "count",
    title = "Distribution of p_hat across 2,000 samples of 150"
  ) +
  theme_minimal()
```



10. Of the sampling distributions from 2 and 3, which has a smaller spread? If you're concerned with making estimates that are more often close to the true value, would you prefer a sampling distribution with a large or small spread?

Answer:

I think this question meant to say "from 8 and 9" since question 2 was about guessing what another student's proportion would look like. I'm going to pick from the answers generated by 8 and 9.

The sample size of 150 has a smaller spread, with its value coming in at .032 compared to .133 for sample size 15. To get a distribution closer to the population, I would prefer a smaller spread so long as other metrics were also trending favorably. For instance, the standard error for the sample size 15 was .02658 versus .002632 for the sample size 150. While the spread was opnly cut in half, the SE came in at 10% of the SE for 15.

```r
spread_15 <- sd(sample_props15$p_hat)
iqr_15 <- IQR(sample_props15$p_hat)
se_15 <- spread_15 / sqrt(15)

spread_150 <- sd(sample_props150$p_hat)
iqr_150 <- IQR(sample_props150$p_hat)
se_150 <- spread_150 / sqrt(150)

cat("Spread, IQR, and standard error for sample size 15:\n")
```

```
## Spread, IQR, and standard error for sample size 15:
```

```r
cat("spread: ", spread_15, "\n")
```

```
## spread:  0.1042746
```

```r
cat("IQR: ", iqr_15, "\n")
```

```
## IQR:  0.1333333
```

```r
cat("SE: ", se_15, "\n")
```

```
## SE:  0.02692358
```

```r
cat("\nSpread, IQR, and standard error for sample size 150:\n")
```

```
##
## Spread, IQR, and standard error for sample size 150:
```

```r
cat("spread: ", spread_150, "\n")
```

```
## spread:  0.03213333
```

```r
cat("IQR: ", iqr_150, "\n")
```

```
## IQR:  0.04
```

```r
cat("SE: ", se_150, "\n")
```

```
## SE:  0.002623675
```
