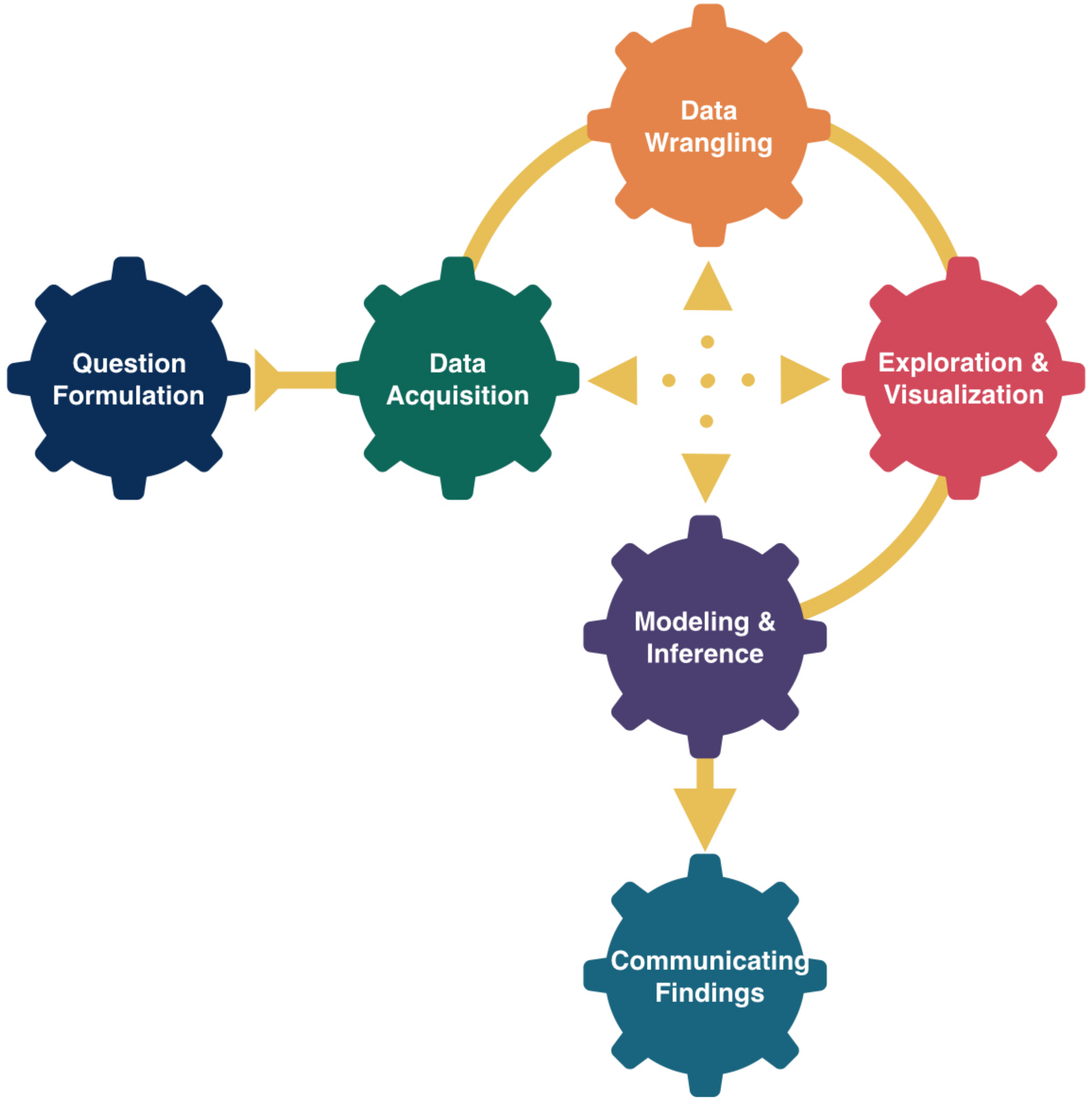


Sampling Distributions

Kelly McConville

Stat 100

Week 8 | Fall 2023

Announcements

- Oct 30th: Hex or Treat Day in Stat 100
    - Wear a Halloween costume and get either a hex sticker or candy!!

Goals for Today

- Modeling & Ethics: Algorithmic bias
- Sampling Distribution
    - Properties
    - Construction in R
- Estimation

Data Ethics: Algorithmic Bias

Return to the Americian Statistical Association's

"Ethical Guidelines for Statistical Practice"

Integrity of Data and Methods

> “The ethical statistical practitioner seeks to understand and mitigate known or suspected limitations, defects, or biases in the data or methods and communicates potential impacts on the interpretation, conclusions, recommendations, decisions, or other results of statistical practices.”

> “For models and algorithms designed to inform or implement decisions repeatedly, develops and/or implements plans to validate assumptions and assess performance over time, as needed. Considers criteria and mitigation plans for model or algorithm failure and retirement.”

Algorithmic Bias

Algorithmic bias: when the model systematically creates unfair outcomes, such as privileging one group over another.

Example: The Coded Gaze

Joy Buolamwini

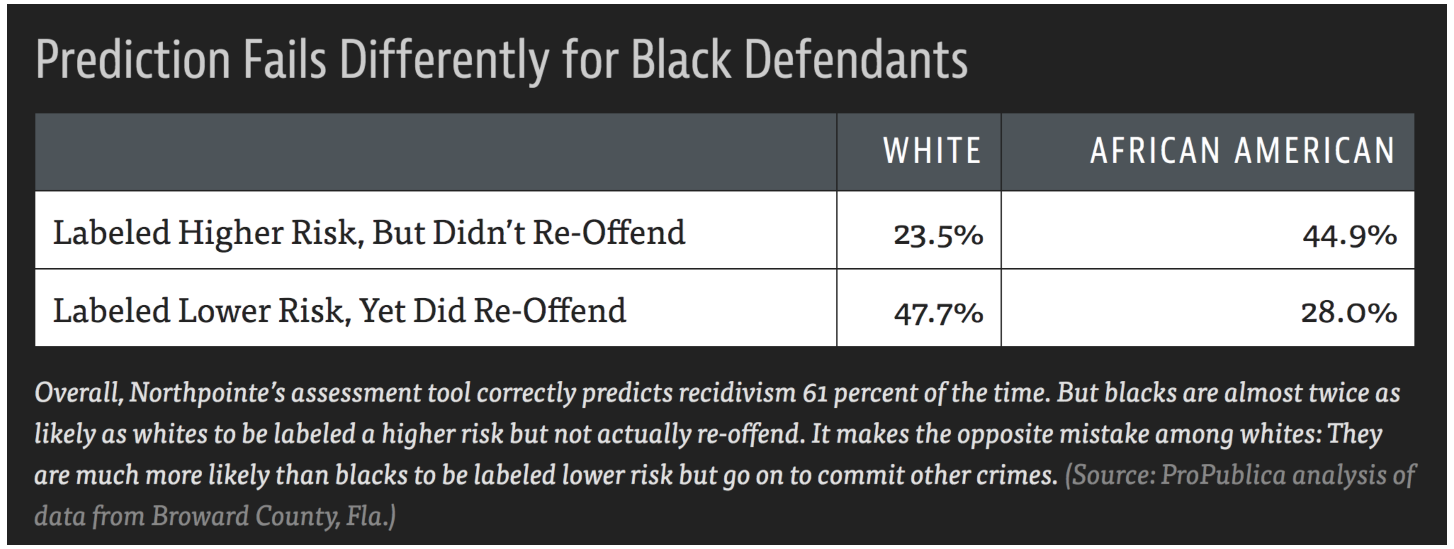
- Facial recognition software struggles to see faces of color.
- Algorithms built on a non-diverse, biased dataset.

Algorithmic Bias

Algorithmic bias: when the model systematically creates unfair outcomes, such as privileging one group over another.

Example: COMPAS model used throughout the country to predict recidivism

- Differences in predictions across race and gender



Prediction Fails Differently for Black Defendants

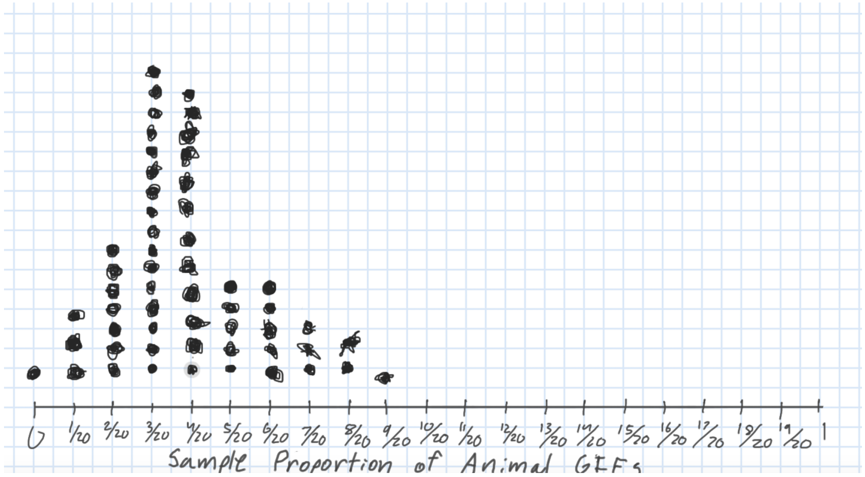
	WHITE	AFRICAN AMERICAN
Labeled Higher Risk, But Didn't Re-Offend	23.5%	44.9%
Labeled Lower Risk, Yet Did Re-Offend	47.7%	28.0%

Overall, Northpointe's assessment tool correctly predicts recidivism 61 percent of the time. But blacks are almost twice as likely as whites to be labeled a higher risk but not actually re-offend. It makes the opposite mistake among whites: They are much more likely than blacks to be labeled lower risk but go on to commit other crimes. *(Source: ProPublica analysis of data from Broward County, Fla.)*

ProPublica Analysis

Sampling Distribution of a Statistic

Steps to Construct an (Approximate)
Sampling Distribution:



1. Decide on a sample size, n.
2. Randomly select a sample of size n from the population.
3. Compute the sample statistic.
4. Put the sample back in.
5. Repeat Steps 2 - 4 many (1000+) times.

- What happens to the center/spread/shape as we increase the sample size?
- What happens to the center/spread/shape if the true parameter changes?

Let's Construct Some Sampling Distributions using R!

Important Notes

- To construct a sampling distribution for a statistic, we need access to the entire population so that we can take repeated samples from the population.
  - Population = Harvard trees
- But if we have access to the entire population, then we know the value of the population parameter.
  - Can compute the exact mean diameter of trees in our population.
- The sampling distribution is needed in the exact scenario where we can't compute it: the scenario where we only have a single sample.
- We will learn how to estimate the sampling distribution soon.
- Today, we have the entire population and are constructing sampling distributions anyway to study their properties!

New R Package: `infer`

```r
library(infer)
```

- Will use `infer` to conduct statistical inference.

Our Population Parameter

Create data frame of Harvard trees:

```r
library(tidyverse)
library(bosTrees)
harTrees <- camTrees %>%
  filter(Ownership == "Harvard", SiteType == "Tree") %>%
  drop_na(SpeciesShort)
```

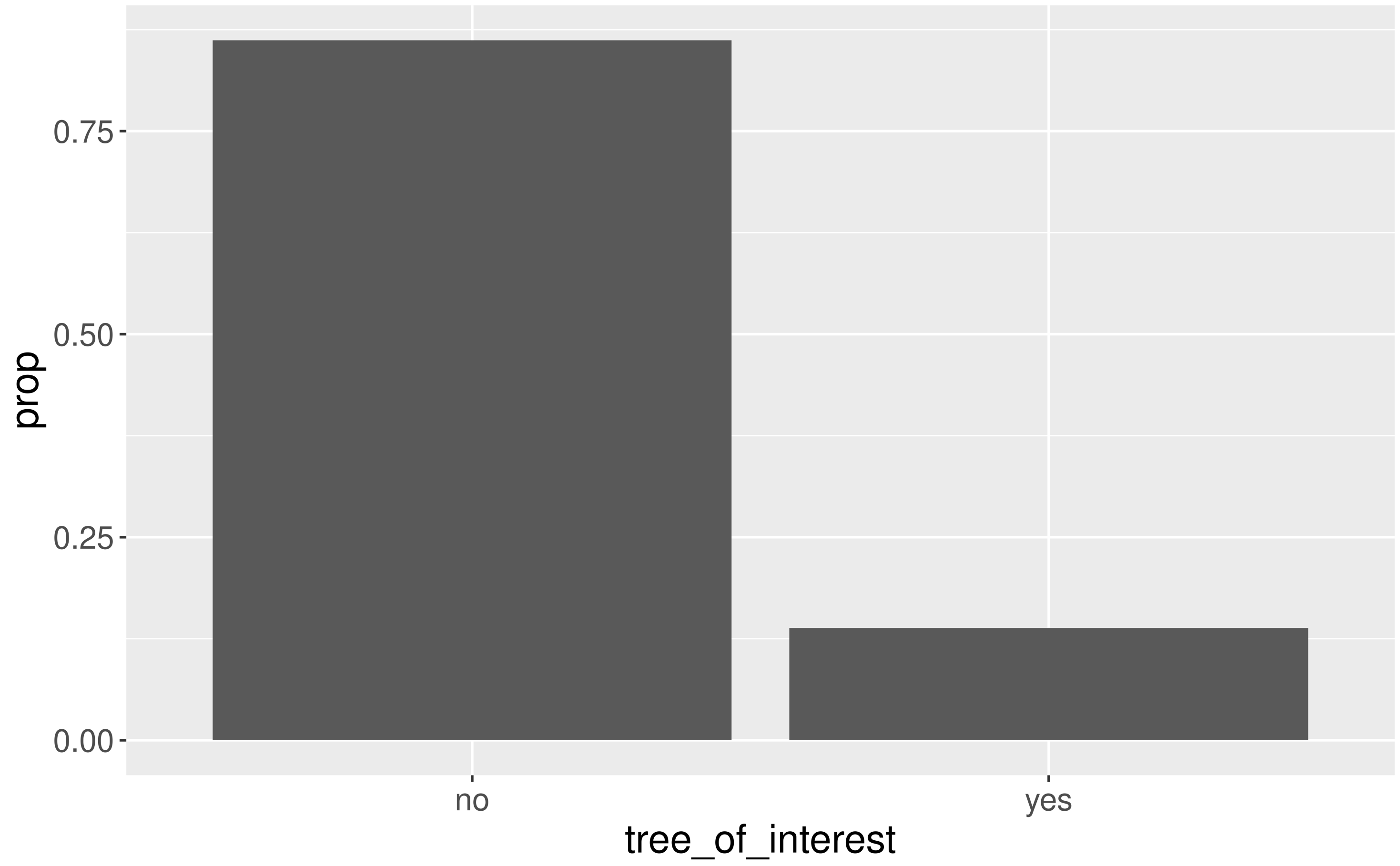
Add variable of interest:

```r
harTrees <- harTrees %>%
  mutate(tree_of_interest = case_when(
    SpeciesShort == "Maple" ~ "yes",
    SpeciesShort != "Not Maple" ~ "no"
  ))
count(harTrees, tree_of_interest)
```

```
# A tibble: 2 × 2
  tree_of_interest     n
  <chr>            <int>
1 no                2707
2 yes                434
```

Population Parameter

```r
# Population distribution
ggplot(data = harTrees,
       mapping = aes(x = tree_of_interest)) +
  geom_bar(aes(y = ..prop.., group = 1),
           stat = "count")
```



```r
# True population parameter
summarize(harTrees,
          parameter = mean(tree_of_interest == "yes"))
```

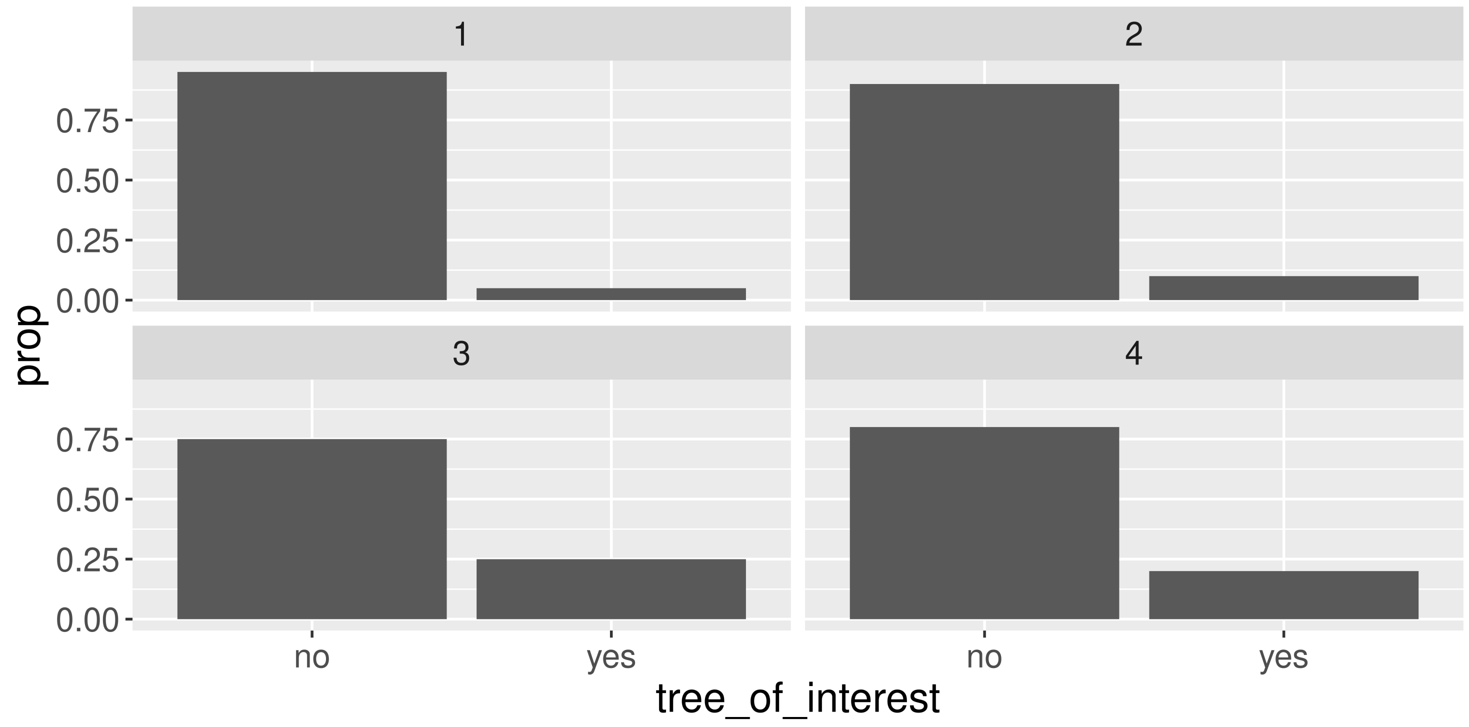
```
# A tibble: 1 × 1
  parameter
      <dbl>
1     0.138
```

Random Samples

Let’s look at 4 random samples.

```
# Draw random samples
samples <- harTrees %>%
  rep_sample_n(size = 20, reps = 4)

# Graph the samples
ggplot(data = samples,
       mapping = aes(x = tree_of_interest)) +
  geom_bar(aes(y = ..prop.., group = 1),
           stat = "count") +
  facet_wrap( ~ replicate)
```

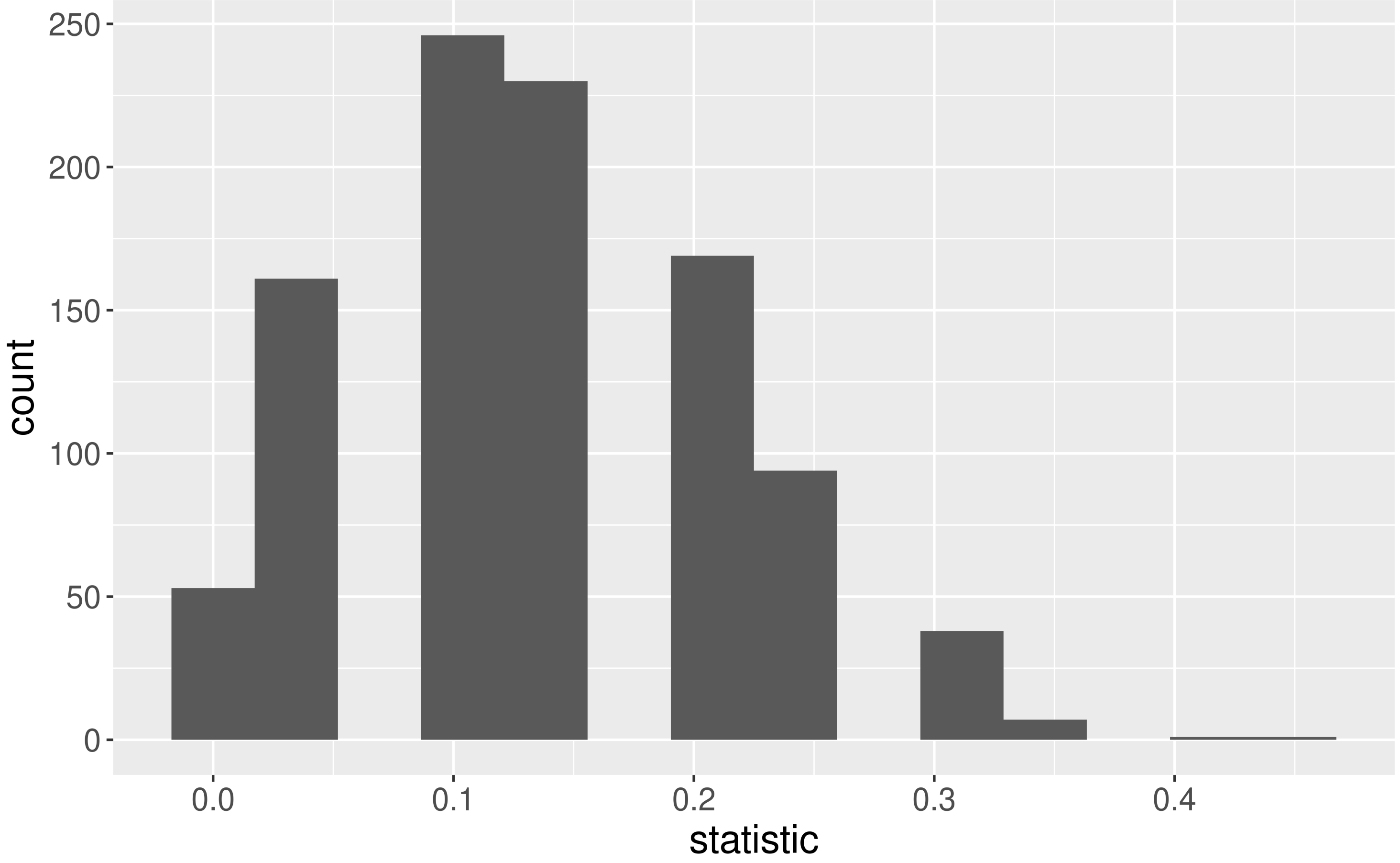


Constructing the Sampling Distribution

Now, let’s take 1000 random samples.

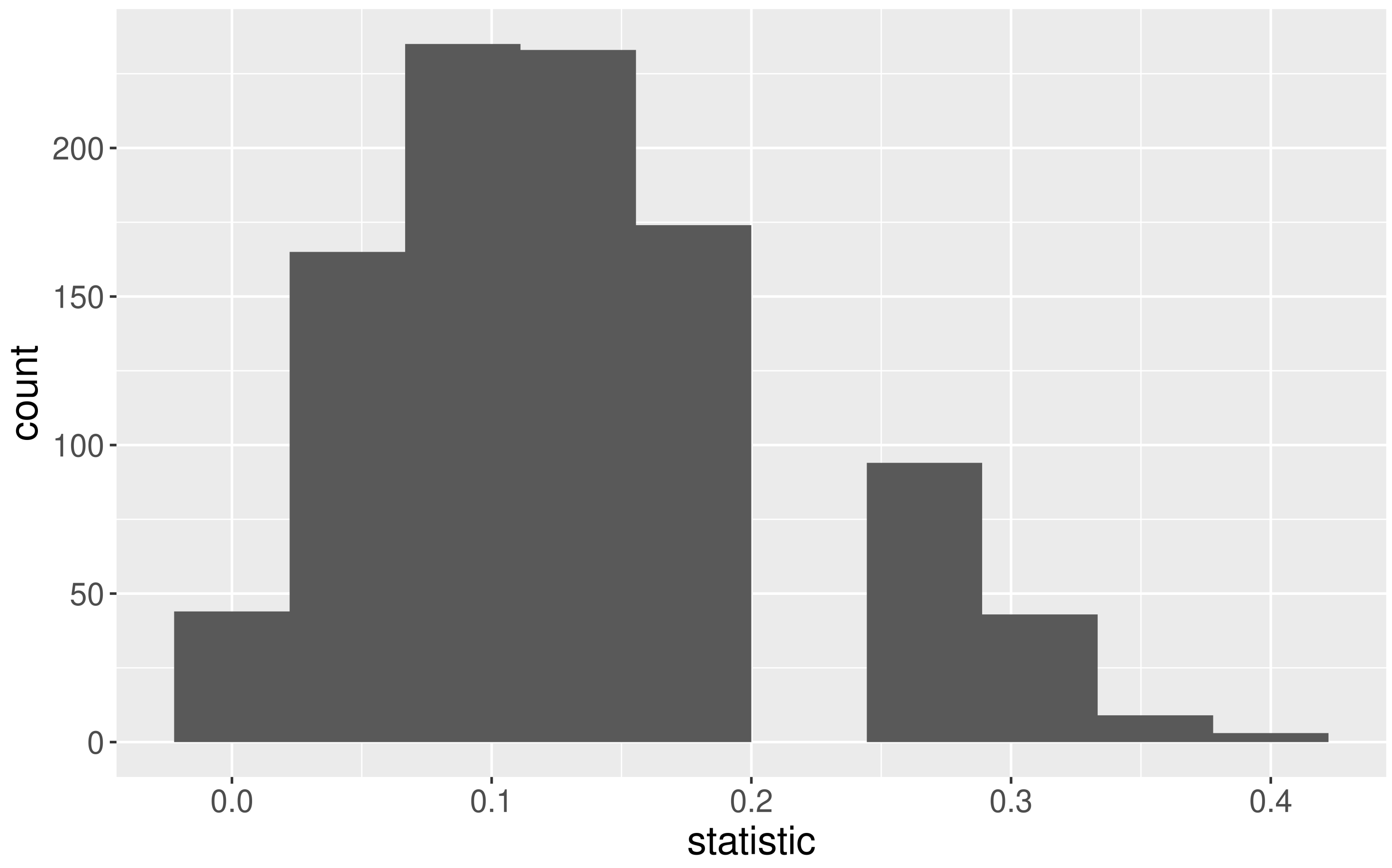
```r
# Construct the sampling distribution
samp_dist <- harTrees %>%
  rep_sample_n(size = 20, reps = 1000) %>%
  group_by(replicate) %>%
  summarize(statistic =
              mean(tree_of_interest == "yes"))

# Graph the sampling distribution
ggplot(data = samp_dist,
       mapping = aes(x = statistic)) +
  geom_histogram(bins = 14)
```



- Shape?
- Center?
- Spread?

Properties of the Sampling Distribution



```r
summarize(samp_dist, mean(statistic))
```

```
# A tibble: 1 × 1
  `mean(statistic)`
              <dbl>
1             0.142
```

```r
summarize(samp_dist, sd(statistic))
```

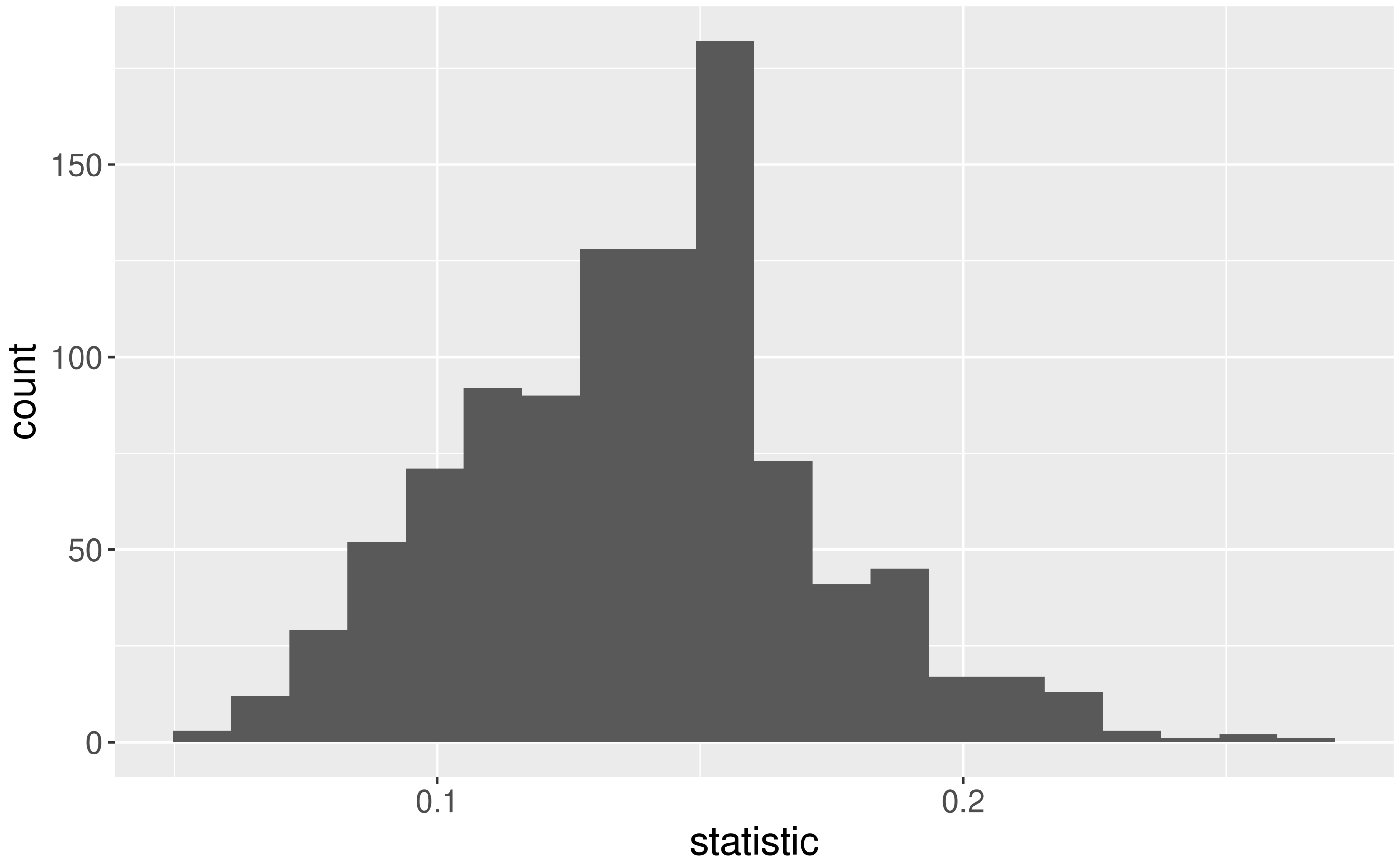
```
# A tibble: 1 × 1
  `sd(statistic)`
            <dbl>
1          0.0779
```

The standard deviation of a sample statistic is called the **standard error**.

What happens to the sampling distribution if we change the sample size from 20 to 100?

```r
# Construct the sampling distribution
samp_dist <- harTrees %>%
  rep_sample_n(size = 100, reps = 1000) %>%
  group_by(replicate) %>%
  summarize(statistic =
                  mean(tree_of_interest == "yes"))

# Graph the sampling distribution
ggplot(data = samp_dist,
       mapping = aes(x = statistic)) +
  geom_histogram(bins = 20)
```



```r
summarize(samp_dist, mean(statistic))
```

```
# A tibble: 1 × 1
  `mean(statistic)`
              <dbl>
1             0.139
```

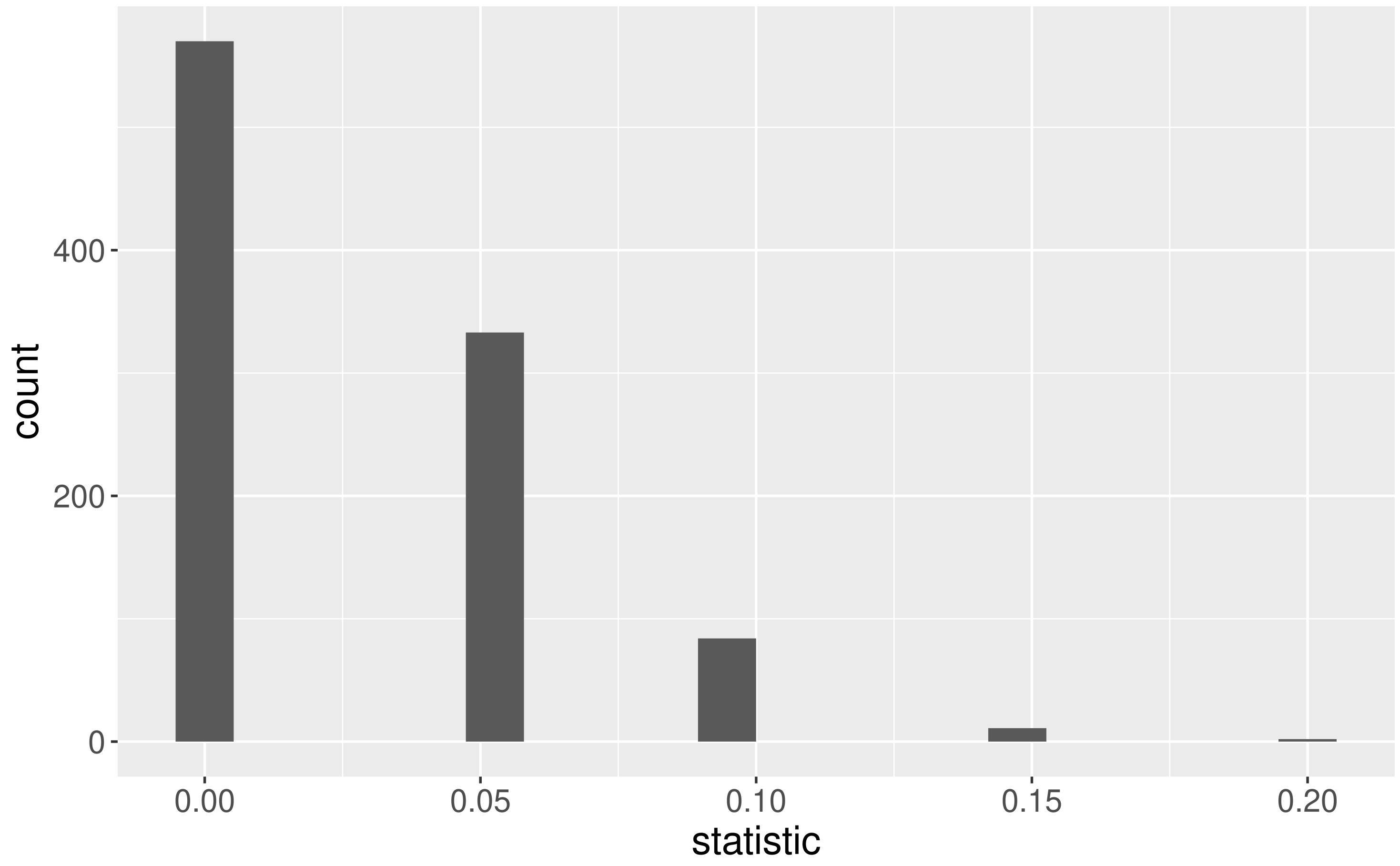
```r
summarize(samp_dist, sd(statistic))
```

```
# A tibble: 1 × 1
  `sd(statistic)`
            <dbl>
1          0.0339
```

What if we change the true parameter value?

```r
# Construct the sampling distribution
samp_dist <- harTrees %>%
  rep_sample_n(size = 20, reps = 1000) %>%
  group_by(replicate) %>%
  summarize(statistic =
              mean(SpeciesShort == "Cherry"))

# Graph the sampling distribution
ggplot(data = samp_dist,
       mapping = aes(x = statistic)) +
    geom_histogram(bins = 20)
```



```r
summarize(samp_dist, mean(statistic))
```

```
# A tibble: 1 × 1
  `mean(statistic)`
              <dbl>
1            0.0271
```

```r
summarize(samp_dist, sd(statistic))
```

```
# A tibble: 1 × 1
  `sd(statistic)`
            <dbl>
1          0.0356
```

On P-Set 5, will investigate what happens when we change the parameter of interest to a mean or a correlation coefficient!

Key Features of a Sampling Distribution

What did we learn about sampling distributions?

- Centered around the true population parameter.
- As the sample size increases, the **standard error** (SE) of the statistic decreases.
- As the sample size increases, the shape of the sampling distribution becomes more bell-shaped and symmetric.
- **Question:** How do sampling distributions help us **quantify uncertainty**?
- **Question:** If I am estimating a parameter in a real example, why won't I be able to construct the sampling distribution??

Estimation

Goal: Estimate the value of a population parameter using data from the sample.

- Question: How do I know which population parameter I am interesting in estimating?
- Answer: Likely depends on the research question and structure of your data!
- Point Estimate: The corresponding statistic
    - Single best guess for the parameter

```r
library(tidyverse)
ce <- read_csv("data/fmli.csv")
summarize(ce, meanFINCBTAX = mean(FINCBTAX))
```

```
# A tibble: 1 × 1
  meanFINCBTAX
         <dbl>
1       62480.
```

Potential Parameters and Point Estimates

Confidence Intervals

It is time to move beyond just point estimates to interval estimates that quantify our uncertainty.

```r
summarize(ce, meanFINCBTAX = mean(FINCBTAX))
```

```
# A tibble: 1 × 1
  meanFINCBTAX
         <dbl>
1       62480.
```

- Confidence Interval: Interval of plausible values for a parameter
- Form: $\text{statistic} \pm \text{Margin of Error}$
- Question: How do we find the Margin of Error (ME)?
- Answer: If the sampling distribution of the statistic is approximately bell-shaped and symmetric, then a statistic will be within 2 SEs of the parameter for 95% of the samples.
- Form: $\text{statistic} \pm 2\text{SE}$
- Called a 95% confidence interval (CI). (Will discuss the meaning of confidence soon)

Confidence Intervals

95% CI Form:

$$\text{statistic} \pm 2\text{SE}$$

Let’s use the `ce` data to produce a CI for the average household income before taxes.

```r
summarize(ce, meanFINCBTAX = mean(FINCBTAX))
```

```
# A tibble: 1 × 1
  meanFINCBTAX
         <dbl>
1       62480.
```

What else do we need to construct the CI?

- **Problem**: To compute the SE, we need many samples from the population. We have 1 sample.
- **Solution**: Approximate the sampling distribution using **ONLY OUR ONE SAMPLE!**

Reminders:

- Oct 30th: Hex or Treat Day in Stat 100
  - Wear a Halloween costume and get either a hex sticker or candy!!