Confidence Intervals

Nate Wells

Math 141, 3/15/21

Outline

In this lecture, we will. . .

Outline

In this lecture, we will. . .

- Introduce confidence intervals as a method for estimating a parameter
- Use bootstrapping as means of creating confidence intervals
- Implement the `infer` package to calculate confidence intervals

Section 1

Confidence Intervals

Point vs. Interval Estimates

- To estimate a population parameter, we can use a sample statistic.

Point vs. Interval Estimates

- To estimate a population parameter, we can use a sample statistic.
  - Suppose you are hosting a pizza party, and need to know what proportion of vegetarian pizza to order.

Point vs. Interval Estimates

- To estimate a population parameter, we can use a sample statistic.
  - Suppose you are hosting a pizza party, and need to know what proportion of vegetarian pizza to order.
  - You can ask a (random) sample of attendees about their pizza preference and use the proportion $\hat{p}$ in the sample as an estimate for the total proportion p.

Point vs. Interval Estimates

- To estimate a population parameter, we can use a sample statistic.
  - Suppose you are hosting a pizza party, and need to know what proportion of vegetarian pizza to order.
  - You can ask a (random) sample of attendees about their pizza preference and use the proportion $\hat{p}$ in the sample as an estimate for the total proportion p.
- The sample statistic is your best guess for the population parameter, but it isn't the whole story

Point vs. Interval Estimates

- To estimate a population parameter, we can use a sample statistic.
  - Suppose you are hosting a pizza party, and need to know what proportion of vegetarian pizza to order.
  - You can ask a (random) sample of attendees about their pizza preference and use the proportion $\hat{p}$ in the sample as an estimate for the total proportion p.
- The sample statistic is your best guess for the population parameter, but it isn't the whole story
  - Suppose after polling pizza party attendees, you find $p = 0.5$. How certain should you be in this estimate?

Point vs. Interval Estimates

- To estimate a population parameter, we can use a sample statistic.
    - Suppose you are hosting a pizza party, and need to know what proportion of vegetarian pizza to order.
    - You can ask a (random) sample of attendees about their pizza preference and use the proportion $\hat{p}$ in the sample as an estimate for the total proportion p.
- The sample statistic is your best guess for the population parameter, but it isn't the whole story
    - Suppose after polling pizza party attendees, you find $p = 0.5$. How certain should you be in this estimate?
    - If you asked just $n = 10$ people, you might not be very certain But if you asked $n = 100$, you might be very certain.

Point vs. Interval Estimates

- To estimate a population parameter, we can use a sample statistic.
    - Suppose you are hosting a pizza party, and need to know what proportion of vegetarian pizza to order.
    - You can ask a (random) sample of attendees about their pizza preference and use the proportion $\hat{p}$ in the sample as an estimate for the total proportion p.
- The sample statistic is your best guess for the population parameter, but it isn't the whole story
    - Suppose after polling pizza party attendees, you find $p = 0.5$. How certain should you be in this estimate?
    - If you asked just $n = 10$ people, you might not be very certain But if you asked $n = 100$, you might be very certain.
- It might be preferable to estimate the proportion using a range of values, with smaller intervals corresponding to larger samples.

Point vs. Interval Estimates

- To estimate a population parameter, we can use a sample statistic.
  - Suppose you are hosting a pizza party, and need to know what proportion of vegetarian pizza to order.
  - You can ask a (random) sample of attendees about their pizza preference and use the proportion $\hat{p}$ in the sample as an estimate for the total proportion p.
- The sample statistic is your best guess for the population parameter, but it isn't the whole story
  - Suppose after polling pizza party attendees, you find $p = 0.5$. How certain should you be in this estimate?
  - If you asked just $n = 10$ people, you might not be very certain But if you asked $n = 100$, you might be very certain.
- It might be preferable to estimate the proportion using a range of values, with smaller intervals corresponding to larger samples.
  - With just $n = 10$ people, you might give a range 0.2 to 0.8 for p.
  - But with $n = 100$, you might instead give the range 0.4 to 0.6.

Interval Estimates

An **interval estimate** for a parameter takes the form

$$\text{Statistic} \ \pm \ \text{Margin of Error}$$

Interval Estimates

An **interval estimate** for a parameter takes the form

$$\text{Statistic} \ \pm \ \text{Margin of Error}$$

- The interval estimate gives a range of plausible values for the parameter.

Interval Estimates

An **interval estimate** for a parameter takes the form

$$\text{Statistic} \ \pm \ \text{Margin of Error}$$

- The interval estimate gives a range of plausible values for the parameter.
- The Margin of Error indicates the precision of our estimate

Interval Estimates

An **interval estimate** for a parameter takes the form

$$\text{Statistic} \pm \text{Margin of Error}$$

- The interval estimate gives a range of plausible values for the parameter.
- The Margin of Error indicates the precision of our estimate
  - When sampling pizza preferences with $n = 100$, we estimate p using the interval

$$0.5 \pm 0.2 \qquad \text{or } 0.3 \text{ to } 0.7$$

Interval Estimates

An **interval estimate** for a parameter takes the form

$$\text{Statistic} \;\pm\; \text{Margin of Error}$$

- The interval estimate gives a range of plausible values for the parameter.
- The Margin of Error indicates the precision of our estimate
  - When sampling pizza preferences with $n = 100$, we estimate p using the interval

$$0.5 \pm 0.2 \qquad \text{or } 0.3 \text{ to } 0.7$$

- We also report a success rate (or confidence level) for the estimation technique.

Interval Estimates

An **interval estimate** for a parameter takes the form

$$\text{Statistic} \;\pm\; \text{Margin of Error}$$

- The interval estimate gives a range of plausible values for the parameter.
- The Margin of Error indicates the precision of our estimate
  - When sampling pizza preferences with $n = 100$, we estimate p using the interval

$$0.5 \pm 0.2 \qquad \text{or } 0.3 \text{ to } 0.7$$

- We also report a success rate (or confidence level) for the estimation technique.
- The confidence level corresponds to the proportion of sample statistics within the margin of error of the true parameter.

Interval Estimates

An **interval estimate** for a parameter takes the form

$$\text{Statistic} \pm \text{Margin of Error}$$

- The interval estimate gives a range of plausible values for the parameter.
- The Margin of Error indicates the precision of our estimate
  - When sampling pizza preferences with $n = 100$, we estimate p using the interval

$$0.5 \pm 0.2 \qquad \text{or } 0.3 \text{ to } 0.7$$

- We also report a success rate (or confidence level) for the estimation technique.
- The confidence level corresponds to the proportion of sample statistics within the margin of error of the true parameter.
  - When sampling pizza preference with $n = 100$, we estimate p using the interval

$$0.5 \pm 0.2 \qquad \text{with confidence } 95\%.$$

Interval Estimates

An **interval estimate** for a parameter takes the form

$$\text{Statistic} \ \pm \ \text{Margin of Error}$$

- The interval estimate gives a range of plausible values for the parameter.
- The Margin of Error indicates the precision of our estimate
  - When sampling pizza preferences with $n = 100$, we estimate p using the interval

$$0.5 \pm 0.2 \qquad \text{or } 0.3 \text{ to } 0.7$$

- We also report a success rate (or confidence level) for the estimation technique.
- The confidence level corresponds to the proportion of sample statistics within the margin of error of the true parameter.
  - When sampling pizza preference with $n = 100$, we estimate p using the interval

$$0.5 \pm 0.2 \qquad \text{with confidence } 95\%.$$

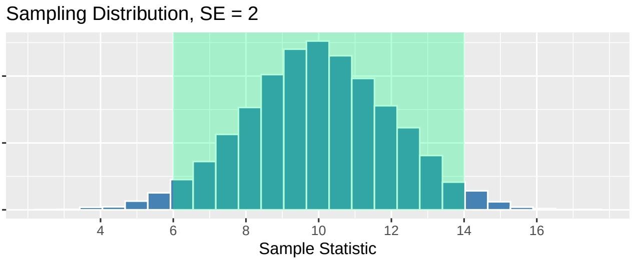
- To get the margin of error and the confidence level, we make use of the sampling distribution (or the bootstrap approximation).

The Sampling Distribution

- For sampling distributions that are approximately bell-shaped (usually true if $n \geq 30$), 95% of all sample statistics will be within 2 standard error units of the true parameter.

The Sampling Distribution

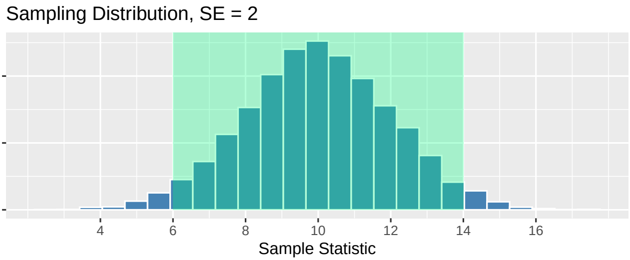
- For sampling distributions that are approximately bell-shaped (usually true if $n \geq 30$), 95% of all sample statistics will be within 2 standard error units of the true parameter.



- But this means that for 95% of all samples, the true parameter will be within a distance of 2 SE of the sample statistic.

The Sampling Distribution

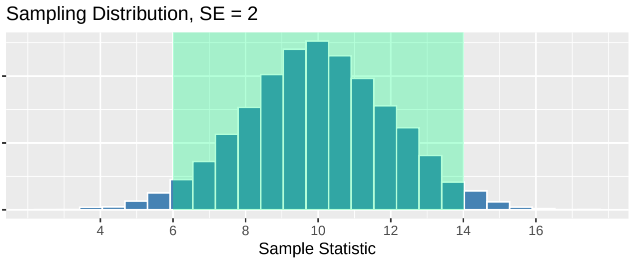
- For sampling distributions that are approximately bell-shaped (usually true if $n \geq 30$), 95% of all sample statistics will be within 2 standard error units of the true parameter.



- But this means that for 95% of all samples, the true parameter will be within a distance of 2 SE of the sample statistic.
- If you build an interval estimate using the sample statistic and a margin of error of 2 SE, the interval will contain the parameter for 95% of all samples.

The Sampling Distribution

- For sampling distributions that are approximately bell-shaped (usually true if $n \geq 30$), 95% of all sample statistics will be within 2 standard error units of the true parameter.



- But this means that for 95% of all samples, the true parameter will be within a distance of 2 SE of the sample statistic.
- If you build an interval estimate using the sample statistic and a margin of error of 2 SE, the interval will contain the parameter for 95% of all samples.
  - How do we get the SE? Bootstrap.

Reproduction Rate for Covid-19

Researchers are interested in the reproduction rate of COVID-19.

- We have a sample of 50 infected individuals and perform contract tracing to determine how many other individuals each infects.

Reproduction Rate for Covid-19

Researchers are interested in the reproduction rate of COVID-19.

- We have a sample of 50 infected individuals and perform contract tracing to determine how many other individuals each infects.

```
##   infected  n
## 1        0  5
## 2        1 13
## 3        2 14
## 4        3 12
## 5        4  5
## 6        6  1
```

```
##   mean_infected
## 1          2.06
```

Reproduction Rate for Covid-19

Researchers are interested in the reproduction rate of COVID-19.

- We have a sample of 50 infected individuals and perform contract tracing to determine how many other individuals each infects.

```
##   infected  n
## 1        0  5
## 2        1 13
## 3        2 14
## 4        3 12
## 5        4  5
## 6        6  1
```

```
##   mean_infected
## 1          2.06
```

- Is the true reproduction rate exactly 2.06?

Reproduction Rate for Covid-19

Researchers are interested in the reproduction rate of COVID-19.

- We have a sample of 50 infected individuals and perform contract tracing to determine how many other individuals each infects.

```
##   infected  n
## 1        0  5
## 2        1 13
## 3        2 14
## 4        3 12
## 5        4  5
## 6        6  1
```

```
##   mean_infected
## 1          2.06
```

- Is the true reproduction rate exactly 2.06?
    - Surely not! This is just one sample of size 50

Reproduction Rate for Covid-19

Researchers are interested in the reproduction rate of COVID-19.

- We have a sample of 50 infected individuals and perform contract tracing to determine how many other individuals each infects.

```
##   infected  n
## 1        0  5
## 2        1 13
## 3        2 14
## 4        3 12
## 5        4  5
## 6        6  1
```

```
##   mean_infected
## 1          2.06
```

- Is the true reproduction rate exactly 2.06?
    - Surely not! This is just one sample of size 50
- But how much does the reproduction rate vary from sample to sample?

Bootstrap Reproduction Rate

Create the bootstrap samples:

```
bootstrap_samples <- covid %>%
  rep_sample_n(size = 50, replace = TRUE, reps = 2000)

head(bootstrap_samples)
```

```
## # A tibble: 6 x 2
## # Groups:   replicate [1]
##   replicate infected
##       <int>    <int>
## 1         1        2
## 2         1        1
## 3         1        1
## 4         1        0
## 5         1        1
## 6         1        0
```

Bootstrap Reproduction Rate

Compute bootstrap statistics:

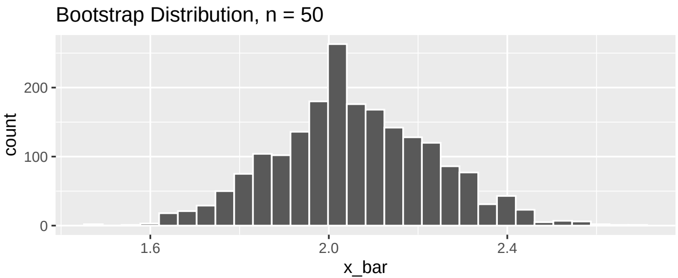
```
bootstrap_stats <- bootstrap_samples %>%
  group_by(replicate) %>%
  summarize(x_bar = mean(infected))

head(bootstrap_stats)
```

```
## # A tibble: 6 x 2
##   replicate x_bar
##       <int> <dbl>
## 1         1  1.86
## 2         2  2.36
## 3         3  2.22
## 4         4  1.86
## 5         5  1.88
## 6         6  1.6
```

Bootstrap Reproduction Rate

Graph the bootstrap distribution:

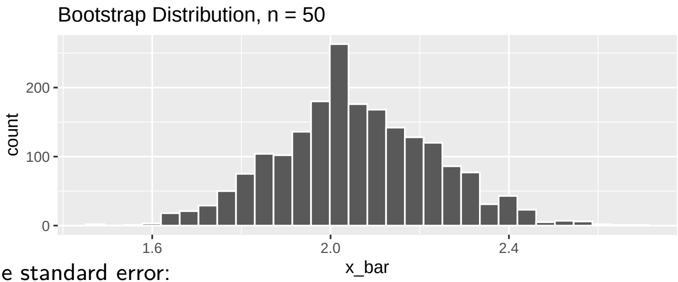
```
ggplot(bootstrap_stats, aes(x = x_bar))+
  geom_histogram(bins = 30, color = "white")+
  labs(title = "Bootstrap Distribution, n = 50")
```



Bootstrap Reproduction Rate

Graph the bootstrap distribution:

```
ggplot(bootstrap_stats, aes(x = x_bar))+
  geom_histogram(bins = 30, color = "white")+
  labs(title = "Bootstrap Distribution, n = 50")
```



Estimate the standard error:

```
bootstrap_stats %>% summarize(SE = sd(x_bar))
```

```
## # A tibble: 1 x 1
##       SE
##   <dbl>
## 1 0.177
```

Confidence Interval for Reproduction Rate

- Our sample reproduction rate was $\bar{x} = 2.04$.

Confidence Interval for Reproduction Rate

- Our sample reproduction rate was $\bar{x} = 2.04$.
- Based on the bootstrap distribution, this statistic has a standard error of $SE = 0.177$.

Confidence Interval for Reproduction Rate

- Our sample reproduction rate was $\bar{x} = 2.04$.
- Based on the bootstrap distribution, this statistic has a standard error of $SE = 0.177$.
- Our 95% confidence interval for the true reproduction rate of COVID-19 is

$$2.06 \pm 2 \cdot 0.177$$

Confidence Interval for Reproduction Rate

- Our sample reproduction rate was $\bar{x} = 2.04$.
- Based on the bootstrap distribution, this statistic has a standard error of $SE = 0.177$.
- Our 95% confidence interval for the true reproduction rate of COVID-19 is

$$2.06 \pm 2 \cdot 0.177$$

- Our best guess for the reproduction rate is between 1.705 and 2.415. This method has a success rate of 95%.

Confidence Interval for Reproduction Rate

- Our sample reproduction rate was $\bar{x} = 2.04$.
- Based on the bootstrap distribution, this statistic has a standard error of $SE = 0.177$.
- Our 95% confidence interval for the true reproduction rate of COVID-19 is

$$2.06 \pm 2 \cdot 0.177$$

- Our best guess for the reproduction rate is between 1.705 and 2.415. This method has a success rate of 95%.
- For reference, this interval matches the one provided by the WHO on 1/23/20.

Generalized Confidence Intervals

- In the previous example, we used the fact for approximately bell-shaped sampling distributions, 95% of of sample statistics are within 2 SE of the population parameter

Generalized Confidence Intervals

- In the previous example, we used the fact for approximately bell-shaped sampling distributions, 95% of of sample statistics are within 2 SE of the population parameter
  - But suppose we instead want a different success rate for our estimation method

Generalized Confidence Intervals

- In the previous example, we used the fact for approximately bell-shaped sampling distributions, 95% of of sample statistics are within 2 SE of the population parameter
  - But suppose we instead want a different success rate for our estimation method
  - Or suppose we want to create interval estimates for sampling distributions that are not bell-shaped

Generalized Confidence Intervals

- In the previous example, we used the fact for approximately bell-shaped sampling distributions, 95% of of sample statistics are within 2 SE of the population parameter
  - But suppose we instead want a different success rate for our estimation method
  - Or suppose we want to create interval estimates for sampling distributions that are not bell-shaped
- We can make these modifications again using the Bootstrap approximation to the sampling distribution

Generalized Confidence Intervals

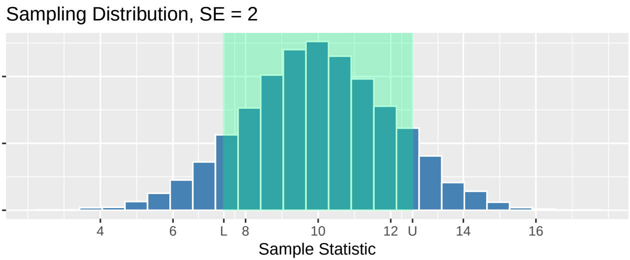
- In the previous example, we used the fact for approximately bell-shaped sampling distributions, 95% of of sample statistics are within 2 SE of the population parameter
  - But suppose we instead want a different success rate for our estimation method
  - Or suppose we want to create interval estimates for sampling distributions that are not bell-shaped
- We can make these modifications again using the Bootstrap approximation to the sampling distribution

General Confidence Intervals

The $C\%$ confidence interval for a parameter is an interval estimate that is computed from sample data by a method that captures the parameter for $C\%$ of all samples.

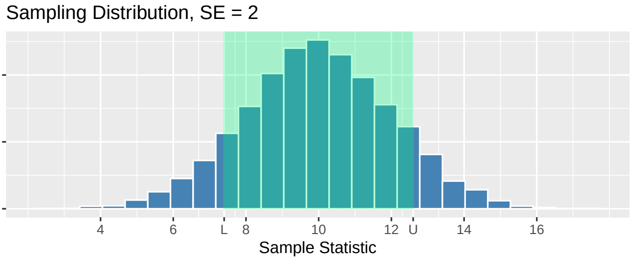
The Percentile Method

Suppose we want to construct an 80% confidence interval:



The Percentile Method

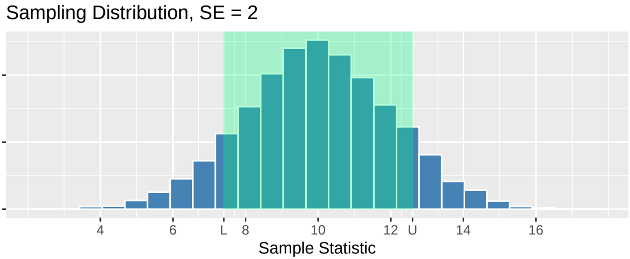
Suppose we want to construct an 80% confidence interval:



- Our goal is to find values L and U along the horizontal axis so that 10% of sample means are less than L and 10% of sample means are greater than U
  - Then 80% of sample means are between L and U
  - L is called the 10th percentile and U is called the 90th percentile

The Percentile Method

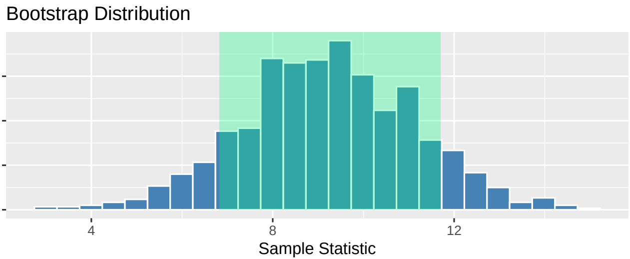
Suppose we want to construct an 80% confidence interval:



- Our goal is to find values L and U along the horizontal axis so that 10% of sample means are less than L and 10% of sample means are greater than U
  - Then 80% of sample means are between L and U
  - L is called the 10th percentile and U is called the 90th percentile
- If we build an interval around each sample mean $\bar{x}$ of length $U - L$, then our interval will capture the true parameter for 80% of all samples

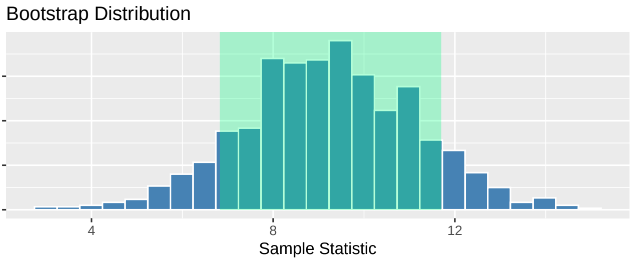
The Percentile Method

- In practice, we won't be able to look at the sampling distribution to find L and U.
  - Instead, we find the corresponding percentiles from the bootstrap distribution.



The Percentile Method

- In practice, we won't be able to look at the sampling distribution to find L and U.
  - Instead, we find the corresponding percentiles from the bootstrap distribution.



- We can use the `quantile` function in R to calculate L and U:

```
my_boot %>% summarise(L = quantile(x_bar, .1),
                      U = quantile(x_bar, .9))
```

```
## # A tibble: 1 x 2
##       L     U
##   <dbl> <dbl>
## 1  6.83  11.7
```

Section 2

The `infer` package

The `infer` Package

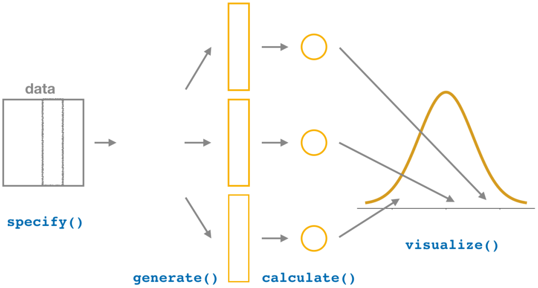
- The `infer` package makes efficient use of the %>% operator perform statistical inference.

The `infer` Package

- The `infer` package makes efficient use of the `%>%` operator perform statistical inference.
- The `infer` package makes use of several verbs-like functions:
    - `specify`, `generate`, `calculate`, `visualize`, `get_ci`

The infer Package

- The `infer` package makes efficient use of the %>% operator perform statistical inference.
- The `infer` package makes use of several verbs-like functions:
  - `specify`, `generate`, `calculate`, `visualize`, `get_ci`



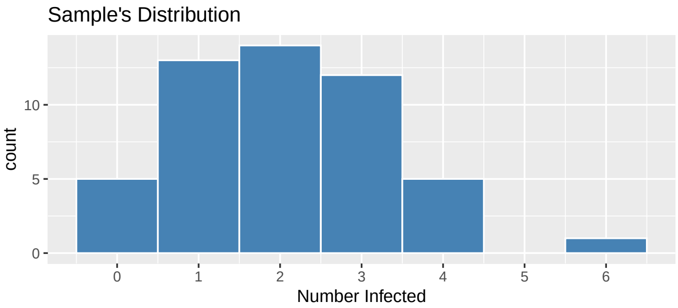
COVID Reproduction Rate

We'll use the previous COVID-19 data to explore the `infer` package:

COVID Reproduction Rate

We'll use the previous COVID-19 data to explore the `infer` package:

```
##   infected  n
## 1        0  5
## 2        1 13
## 3        2 14
## 4        3 12
## 5        4  5
## 6        6  1
```



`specify` the variables

- Every statistical investigation begins with a sample data frame (i.e. `friday`)

specify the variables

- Every statistical investigation begins with a sample data frame (i.e. `friday`)
- The sample may contain many variables of interest

`specify` the variables

- Every statistical investigation begins with a sample data frame (i.e. `friday`)
- The sample may contain many variables of interest
- We must first specify which variable(s) will be the focus of our investigation by designating a response variable

specify the variables

- Every statistical investigation begins with a sample data frame (i.e. `friday`)
- The sample may contain many variables of interest
- We must first specify which variable(s) will be the focus of our investigation by designating a response variable
- To investigate the infection rate

```
covid %>%
  specify(response = infected)
```

generate replicates

- In order to create a bootstrap distribution, we need to resample many times from the OG sample

generate replicates

- In order to create a bootstrap distribution, we need to resample many times from the OG sample
- After selecting variables, pipe results into the generate function to create replicates

```
covid %>%
  specify(response = infected) %>%
  generate( reps = 2000, type = "bootstrap")
```

generate replicates

- In order to create a bootstrap distribution, we need to resample many times from the OG sample
- After selecting variables, pipe results into the generate function to create replicates

```
covid %>%
  specify(response = infected) %>%
  generate( reps = 2000, type = "bootstrap")
```

- We need to indicate how many replicates we want, and what type of method we'll use to create them.

generate replicates

- In order to create a bootstrap distribution, we need to resample many times from the OG sample
- After selecting variables, pipe results into the generate function to create replicates

```
covid %>%
  specify(response = infected) %>%
  generate( reps = 2000, type = "bootstrap")
```

- We need to indicate how many replicates we want, and what type of method we'll use to create them.
- For bootstrap confidence intervals, choose `type = "bootstrap"`, and almost always use at least `reps = 2000`

generate replicates

- In order to create a bootstrap distribution, we need to resample many times from the OG sample
- After selecting variables, pipe results into the generate function to create replicates

```
covid %>%
  specify(response = infected) %>%
  generate( reps = 2000, type = "bootstrap")
```

- We need to indicate how many replicates we want, and what type of method we'll use to create them.
- For bootstrap confidence intervals, choose `type = "bootstrap"`, and almost always use at least `reps = 2000`
- The resulting data frame has a number of rows equal `reps` × `sample_size`

`calculate` summary statistics

- Once we have our bootstrap samples, we need to compute the corresponding statistics

`calculate` summary statistics

- Once we have our bootstrap samples, we need to compute the corresponding statistics
- Use the `calculate` function, whose first argument is `stat`

calculate summary statistics

- Once we have our bootstrap samples, we need to compute the corresponding statistics
- Use the `calculate` function, whose first argument is `stat`
- Many statistics are available: `"mean"`, `"sum"`, `"sd"`, `"median"`, `"prop"`, `"diff in mean`, `"correlation"`, `"slope"`, and more!

calculate summary statistics

- Once we have our bootstrap samples, we need to compute the corresponding statistics
- Use the `calculate` function, whose first argument is `stat`
- Many statistics are available: `"mean"`, `"sum"`, `"sd"`, `"median"`, `"prop"`, `"diff in mean`, `"correlation"`, `"slope"`, and more!

```
covid %>%
  specify(response = infected) %>%
  generate( reps = 2000, type = "bootstrap") %>%
  calculate(stat = "mean")
```

calculate summary statistics

- Once we have our bootstrap samples, we need to compute the corresponding statistics
- Use the `calculate` function, whose first argument is `stat`
- Many statistics are available: `"mean"`, `"sum"`, `"sd"`, `"median"`, `"prop"`, `"diff in mean`, `"correlation"`, `"slope"`, and more!

```
covid %>%
  specify(response = infected) %>%
  generate( reps = 2000, type = "bootstrap") %>%
  calculate(stat = "mean")
```

- After applying `calculate` the resulting data frame consists of one bootstrap statistic for each replicate (saved to the variable `stat`)

Sample Statistic

- Suppose you want to just calculate summary statistics of the OG sample

Sample Statistic

- Suppose you want to just calculate summary statistics of the OG sample
- By using specify and calculate (and omitting generate) we can do just that, paralleling similar calculation for the bootstrap statistics

```
covid_stat<- covid %>%
  specify(response = infected) %>%
  calculate(stat = "mean")
```

Sample Statistic

- Suppose you want to just calculate summary statistics of the OG sample
- By using `specify` and `calculate` (and omitting `generate`) we can do just that, paralleling similar calculation for the bootstrap statistics

```
covid_stat<- covid %>%
  specify(response = infected) %>%
  calculate(stat = "mean")
```

- Note: we saved the value of this calculation as `covid_stat` so we could use it later

Sample Statistic

- Suppose you want to just calculate summary statistics of the OG sample
- By using specify and calculate (and omitting generate) we can do just that, paralleling similar calculation for the bootstrap statistics

```
covid_stat<- covid %>%
  specify(response = infected) %>%
  calculate(stat = "mean")
```

- Note: we saved the value of this calculation as covid_stat so we could use it later
- Since we also will want to make frequent use of the bootstrap statistics, it's worth saving them as a variable too:

Sample Statistic

- Suppose you want to just calculate summary statistics of the OG sample
- By using `specify` and `calculate` (and omitting `generate`) we can do just that, paralleling similar calculation for the bootstrap statistics

```
covid_stat<- covid %>%
  specify(response = infected) %>%
  calculate(stat = "mean")
```

- Note: we saved the value of this calculation as `covid_stat` so we could use it later
- Since we also will want to make frequent use of the bootstrap statistics, it's worth saving them as a variable too:

```
covid_boot<- covid %>%
  specify(response = infected) %>%
  generate( reps = 2000, type = "bootstrap") %>%
  calculate(stat = "mean")
```

`visualize` Bootstrap Distribution

- In order to perform any statistical inference, we need to ensure appropriate shape conditions on bootstrap distribution are met

visualize Bootstrap Distribution

- In order to perform any statistical inference, we need to ensure appropriate shape conditions on bootstrap distribution are met
- Use the `visaulize` verb to quickly generate a reasonably nice-looking histogram of the bootstrap distribution.

visualize Bootstrap Distribution

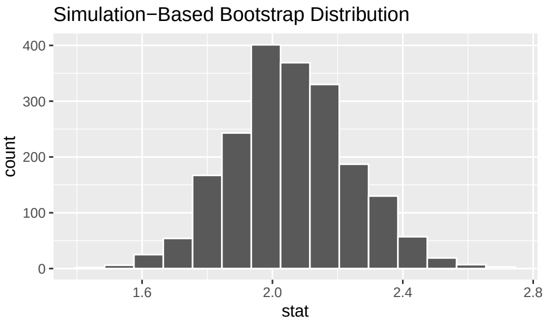
- In order to perform any statistical inference, we need to ensure appropriate shape conditions on bootstrap distribution are met
- Use the visaulize verb to quickly generate a reasonably nice-looking histogram of the bootstrap distribution.

```
covid_boot %>% visualize()
```

visualize Bootstrap Distribution

- In order to perform any statistical inference, we need to ensure appropriate shape conditions on bootstrap distribution are met
- Use the visaulize verb to quickly generate a reasonably nice-looking histogram of the bootstrap distribution.

```
covid_boot %>% visualize()
```



get_confidence_interval to... Get Confidence Interval

- To compute a confidence interval, pipe the calculated data frame into get_confidence_interval (you can use get_ci for brevity)

`get_confidence_interval` to... Get Confidence Interval

- To compute a confidence interval, pipe the `calculated` data frame into `get_confidence_interval` (you can use `get_ci` for brevity)
- We need to specify the `type` of interval we want (either `"percentile"` or `"se"`), along with the confidence `level`

get_confidence_interval to... Get Confidence Interval

- To compute a confidence interval, pipe the `calculated` data frame into `get_confidence_interval` (you can use `get_ci` for brevity)
- We need to specify the `type` of interval we want (either `"percentile"` or `"se"`), along with the confidence `level`
- It's useful to save the resulting data frame for later use

get_confidence_interval to... Get Confidence Interval

- To compute a confidence interval, pipe the calculated data frame into get_confidence_interval (you can use get_ci for brevity)
- We need to specify the type of interval we want (either "percentile" or "se"), along with the confidence level
- It's useful to save the resulting data frame for later use

```
percentile_ci<-covid_boot %>%
  get_ci(level = .95, type = "percentile")
percentile_ci
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1      1.7     2.42
```

get_confidence_interval to... Get Confidence Interval

- To compute a confidence interval, pipe the calculated data frame into get_confidence_interval (you can use get_ci for brevity)
- We need to specify the type of interval we want (either "percentile" or "se"), along with the confidence level
- It's useful to save the resulting data frame for later use

```
percentile_ci<-covid_boot %>%
  get_ci(level = .95, type = "percentile")
percentile_ci
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1      1.7     2.42
```

- When using the percentile type, the first value printed is the lower and the second is the upper bound.

get_confidence_interval to... Get Confidence Interval

- To compute a confidence interval, pipe the calculated data frame into get_confidence_interval (you can use get_ci for brevity)
- We need to specify the type of interval we want (either "percentile" or "se"), along with the confidence level
- It's useful to save the resulting data frame for later use

```
percentile_ci<-covid_boot %>%
  get_ci(level = .95, type = "percentile")
percentile_ci
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1      1.7     2.42
```

- When using the percentile type, the first value printed is the lower and the second is the upper bound.
  - The headings indicate that these are the corresponding percentiles in the bootstrap distribution

Shade Confidence Intervals

- Once you've used get_ci to obtain endpoints of the confidence interval, you can shade the sampling distribution with the confidence interval region.

```
covid_boot %>% visualize()+shade_ci(endpoints = percentile_ci)
```