Infer package: Example

Case study: Is yawning contagious?

- From ModernDive

If you see someone else yawn, are you more likely to yawn? In an episode of the US show Mythbusters, the hosts conducted an experiment to answer this question. The episode is available to view in the United States on the Discovery Network website and more information about the episode is also available on IMDb.

Fifty adult participants who thought they were being considered for an appearance on the show were interviewed by a show recruiter. In the interview, the recruiter either yawned or did not. Participants then sat by themselves in a large van and were asked to wait. While in the van, the Mythbusters team watched the participants using a hidden camera to see if they yawned.

```
library(tidyverse)
library(moderndive)
library(infer)

mythbusters_yawn
```

```
## # A tibble: 50 x 3
##     subj group   yawn
##    <int> <chr>   <chr>
##  1     1 seed    yes
##  2     2 control yes
##  3     3 seed    no
##  4     4 seed    yes
##  5     5 seed    no
##  6     6 control no
##  7     7 seed    yes
##  8     8 control no
##  9     9 control no
## 10    10 seed    no
## # ... with 40 more rows
```

The variables are:

- subj: The participant ID with values 1 through 50.
- group: A binary treatment variable indicating whether the participant was exposed to yawning.
- **seed** indicates the participant was exposed to yawning while **control** indicates the participant was not.
- yawn: A binary response variable indicating whether the participant ultimately yawned.

```
mythbusters_yawn %>%
  group_by(group, yawn) %>%
  summarize(count = n()) # n() gives the current group size
```

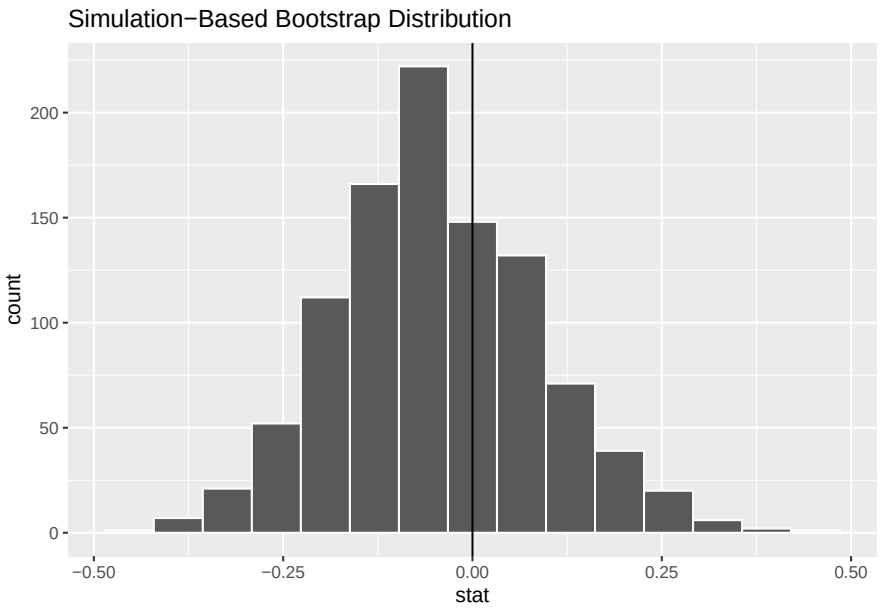
```
## # A tibble: 4 x 3
## # Groups:   group [2]
##   group   yawn  count
```

```
##   <chr>   <chr> <int>
## 1 control no       12
## 2 control yes       4
## 3 seed    no       24
## 4 seed    yes      10
```

```
set.seed(3)
# Constructing the CI for the difference between two pop props
bootstrap_distribution_yawning <- mythbusters_yawn %>%
  specify(formula = yawn ~ group, success = "yes") %>%
  generate(reps = 1000, type = "bootstrap") %>%
  calculate(stat = "diff in props", order = c("control", "seed"))

bootstrap_distribution_yawning
```

```
## # A tibble: 1,000 x 2
##    replicate    stat
##        <int>   <dbl>
##  1         1 -0.0857
##  2         2 -0.0677
##  3         3  0.0762
##  4         4  0.161
##  5         5  0.0515
##  6         6 -0.0873
##  7         7  0.0119
##  8         8  0.0913
##  9         9  0.0584
## 10        10  0.0221
## # ... with 990 more rows
```

```
visualize(bootstrap_distribution_yawning) +
  geom_vline(xintercept = 0)
```




```
# type = "se", the point estimate = 4.4%
obs_diff_in_proprs <- mythbusters_yawn %>%
  specify(formula = yawn ~ group, success = "yes") %>%
```

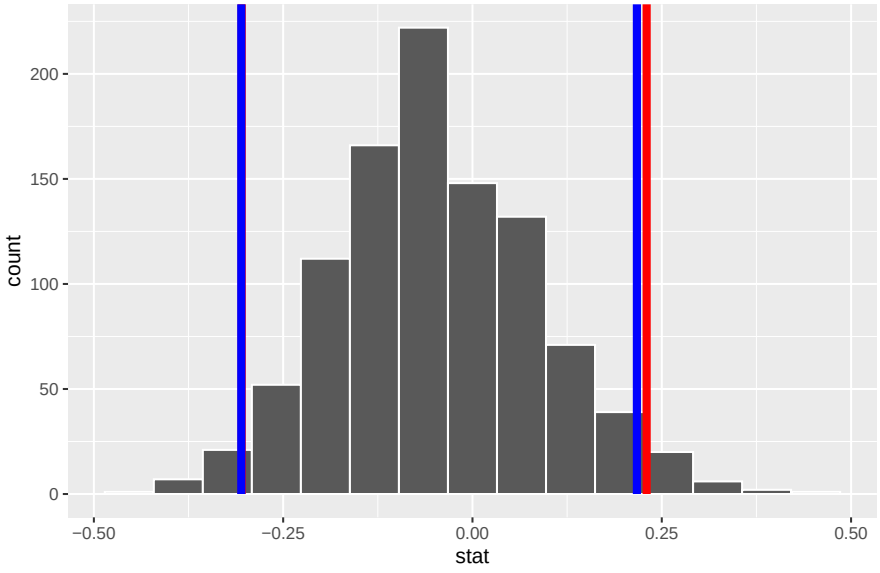
```
  calculate(stat = "diff in props", order = c("control", "seed"))

myth_ci_se <- bootstrap_distribution_yawning %>%
  get_confidence_interval(level = 0.95, type = "se", point_estimate = obs_diff_in_proprs)

myth_ci_perc <- bootstrap_distribution_yawning %>%
  get_confidence_interval(level = 0.95, type = "percentile")

myth_ci_se
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   -0.305    0.217
```

```
myth_ci_perc
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   -0.305    0.230
```

```
visualize(bootstrap_distribution_yawning) +
  ggtitle("") +
  shade_ci(endpoints = myth_ci_perc, fill = NULL, color = "red") +
  shade_ci(endpoints = myth_ci_se, fill = NULL, color = "blue")
```



Law Data

```
library(bootstrap)

data("law")

obs_corr <- law %>%
  specify(formula = LSAT ~ GPA) %>%
  calculate(stat = "correlation")
```

```
obs_corr
```

```
## # A tibble: 1 x 1
##    stat
##   <dbl>
## 1 0.776
```

```
bootstrap_distribution_law <- law %>%
  specify(LSAT ~ GPA) %>%
  generate(reps = 1000, type = "bootstrap") %>%
  calculate(stat = "correlation")

bootstrap_distribution_law
```

```
## # A tibble: 1,000 x 2
##    replicate  stat
##        <int> <dbl>
##  1         1 0.683
##  2         2 0.856
##  3         3 0.691
##  4         4 0.508
##  5         5 0.785
##  6         6 0.873
##  7         7 0.694
##  8         8 0.799
##  9         9 0.934
## 10        10 0.822
## # ... with 990 more rows
```

```
visualize(bootstrap_distribution_law) +
  geom_vline(xintercept = 0.776, col = "red")


bootstrap_distribution_law %>%
  get_confidence_interval(level = 0.95, type = "percentile")
```

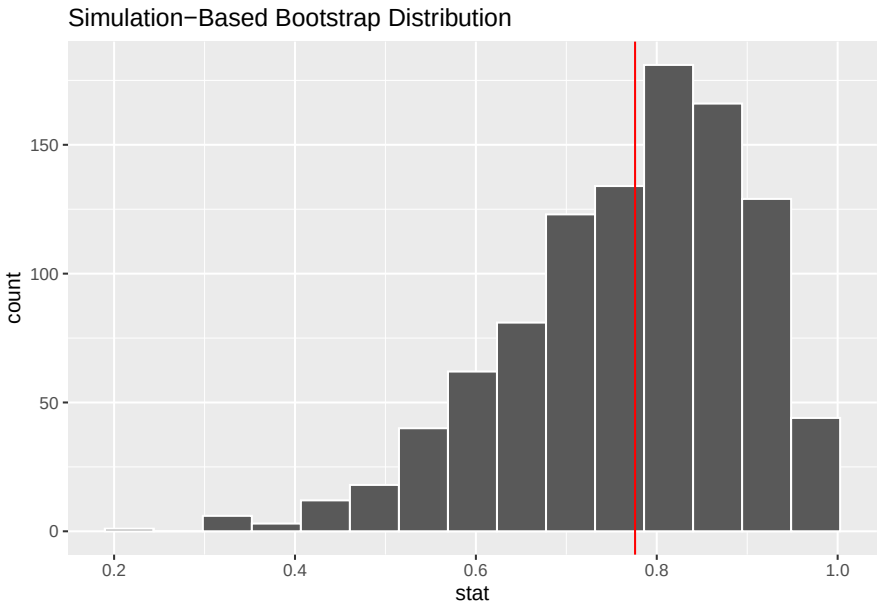
```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1    0.469    0.961
```

```
bootstrap_distribution_law %>%
  get_confidence_interval(level = 0.95, point_estimate = obs_corr, type = "se")
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1    0.522     1.03
```

```
bootstrap_distribution_law %>%
  get_confidence_interval(level = 0.95, point_estimate = obs_corr, type = "bias-corrected")
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1    0.436    0.950
```



Mouse Data

```
trt <- c(94, 197, 16, 38, 99, 141, 23)
ctrl <- c(52, 104, 146, 10, 51, 30, 40, 27, 46)

total <- c(trt, ctrl)
id <- as.factor(c(rep("x", length(trt)), rep("y", length(ctrl))))
mouse <- tibble(id, total)

bootstrap_distribution_mouse <- mouse %>%
  specify(formula = total ~ id) %>%
  generate(reps = 999, type = "bootstrap") %>%
  calculate(stat = "diff in means", order = c("x", "y"))

visualize(bootstrap_distribution_mouse)

mean(bootstrap_distribution_mouse$stat)
```

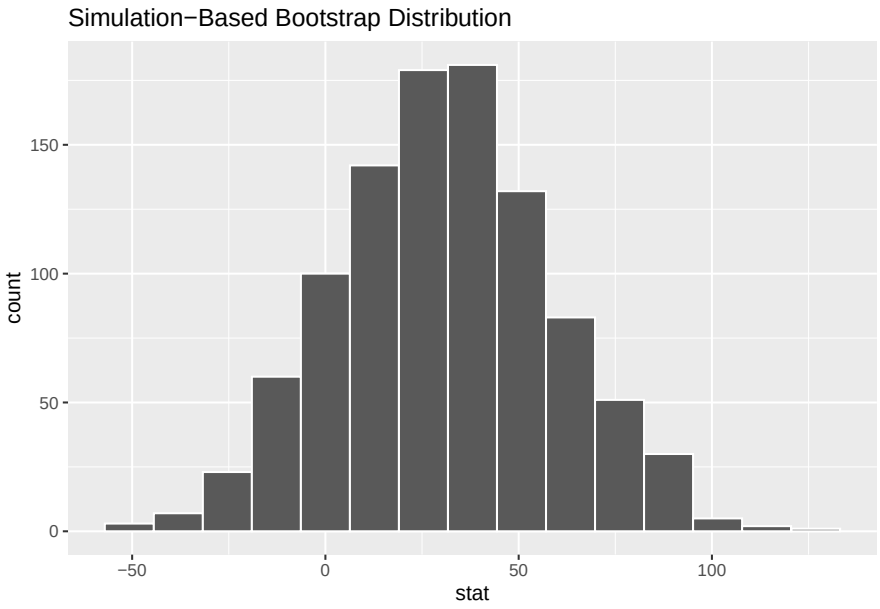
```
## [1] 31.03789
```

```
sd(bootstrap_distribution_mouse$stat)
```

```
## [1] 27.90157
```

```
bootstrap_distribution_mouse %>%
  get_confidence_interval(level = 0.95, type = "percentile")
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1    -22.2     87.0
```


Simulation−Based Bootstrap Distribution
count
150
100
50
0
−50
0
50
100
stat