# Data 606 Lab 7

Lu Beyer

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(openintro)
```

```
## Loading required package: airports
## Loading required package: cherryblossom
## Loading required package: usdata
```

```
library(infer)
```

```
## Warning: package 'infer' was built under R version 4.3.3
```

```
library(DATA606)
```

```
## Loading required package: shiny
##
## Attaching package: 'shiny'
##
## The following object is masked from 'package:infer':
##
##     observe
##
## Loading required package: markdown

##
## Welcome to CUNY DATA606 Statistics and Probability for Data Analytics
## This package is designed to support this course. The text book used
## is OpenIntro Statistics, 4th Edition. You can read this by typing
## vignette('os4') or visit www.OpenIntro.org.
##
```

```
## The getLabs() function will return a list of the labs available.
##
## The demo(package='DATA606') will list the demos that are available.

##
## Attaching package: 'DATA606'
##
## The following objects are masked from 'package:openintro':
##
##     calc_streak, present, qqnormsim
##
## The following object is masked from 'package:utils':
##
##     demo
```

```
set.seed(777)
```

```
data <- data('yrbss', package ='openintro')
```

```
?yrbss
```

```
## starting httpd help server ... done
```

Exercise 1 What are the cases in this data set? How many cases are there in our sample?

```
glimpse(yrbss)
```

```
## Rows: 13,583
## Columns: 13
## $ age                      <int> 14, 14, 15, 15, 15, 15, 15, 14, 15, 15, 15, 1~
## $ gender                   <chr> "female", "female", "female", "female", "fema~
## $ grade                    <chr> "9", "9", "9", "9", "9", "9", "9", "9", "9", ~
## $ hispanic                 <chr> "not", "not", "hispanic", "not", "not", "not"~
## $ race                     <chr> "Black or African American", "Black or Africa~
## $ height                   <dbl> NA, NA, 1.73, 1.60, 1.50, 1.57, 1.65, 1.88, 1~
## $ weight                   <dbl> NA, NA, 84.37, 55.79, 46.72, 67.13, 131.54, 7~
## $ helmet_12m               <chr> "never", "never", "never", "never", "did not ~
## $ text_while_driving_30d   <chr> "0", NA, "30", "0", "did not drive", "did not~
## $ physically_active_7d     <int> 4, 2, 7, 0, 2, 1, 4, 4, 5, 0, 0, 0, 4, 7, 7, ~
## $ hours_tv_per_school_day  <chr> "5+", "5+", "5+", "2", "3", "5+", "5+", "5+",~
## $ strength_training_7d     <int> 0, 0, 0, 0, 1, 0, 2, 0, 3, 0, 3, 0, 0, 7, 7, ~
## $ school_night_hours_sleep <chr> "8", "6", "<5", "6", "9", "8", "9", "6", "<5"~
```

There are 13,583 rows/cases in this dataset

```
summary(yrbss$weight)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   29.94   56.25   64.41   67.91   76.20  180.99    1004
```

Exercise 2 How many observations are we missing weights from?

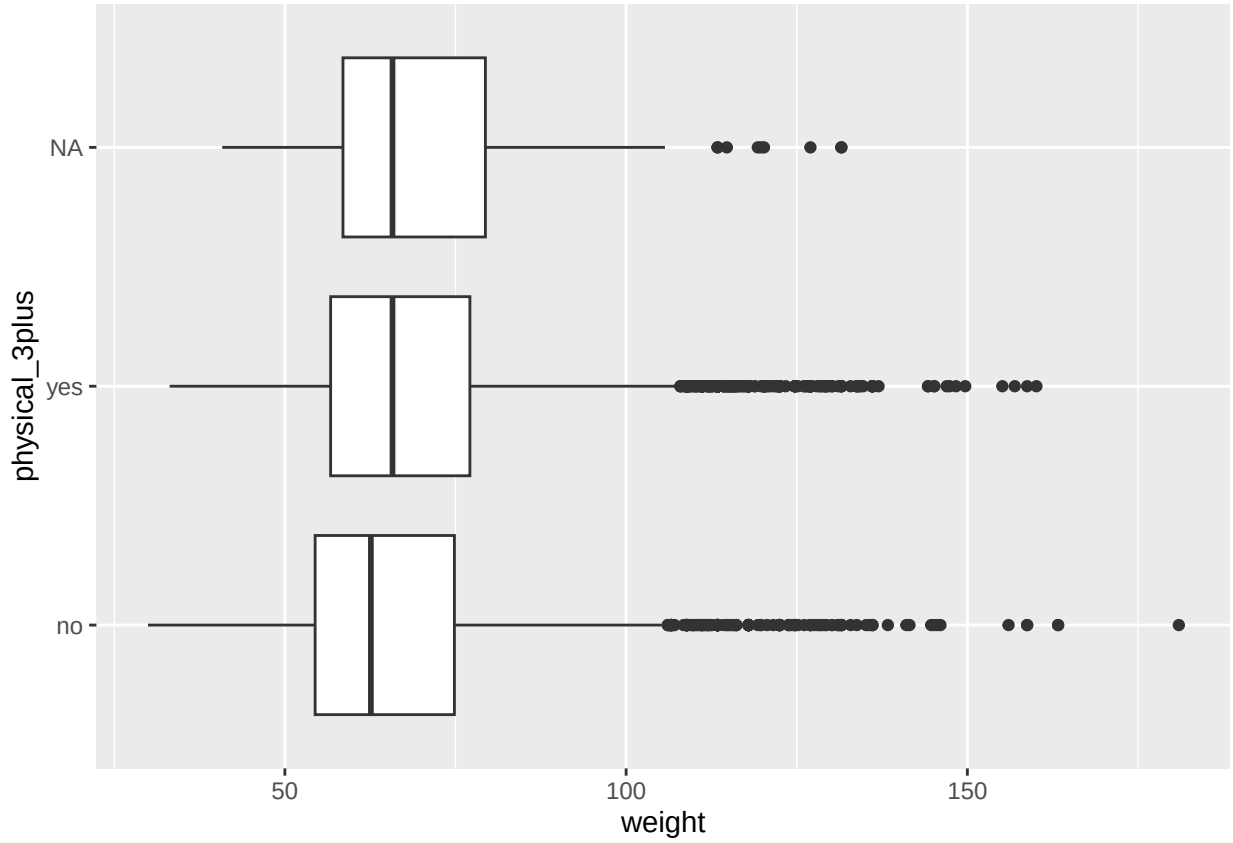
We're missing 1004 observations for weight.

```
x <- yrbss %>%
  mutate(physical_3plus = ifelse(yrbss$physically_active_7d > 2, "yes", "no"))
```

Exercise 3 Make a side-by-side boxplot of physical_3plus and weight. Is there a relationship between these two variables? What did you expect and why?

```
ggplot(x, aes(x=weight, y=physical_3plus)) +
  geom_boxplot()
```

```
## Warning: Removed 1004 rows containing non-finite values (`stat_boxplot()`).
```




The values for physically_active_7d yes/no, are very similar, although "yes" has a slightly higher median value. We generally would expect that those who exercise less have higher weights, but this is not the case, as shown by the median values.

```
x_summary <- x %>%
  group_by(physical_3plus) %>%
  summarise(mean_weight = mean(weight, na.rm = TRUE),
            count = n())
```

Exercise 4

Are all conditions necessary for inference satisfied? Comment on each. You can compute the group sizes with the summarize command above by defining a new variable with the definition n().

There is a large sample size, but we also see several outliers for each response.

Exercise 5

Write the hypotheses for testing if the average weights are different for those who exercise at least times a week and those who don't.

H1: Students who exercise three or more times a week will have a lower average weight than students who exercise less than three times a week.

H0: The weights of students who exercise three or more times a week do not differ from students who exercise less than three times a week.

```
obs_diff <- x %>%
  drop_na(physical_3plus) %>%
  specify(weight ~ physical_3plus) %>%
  calculate(stat = "diff in means", order = c("yes", "no"))
```

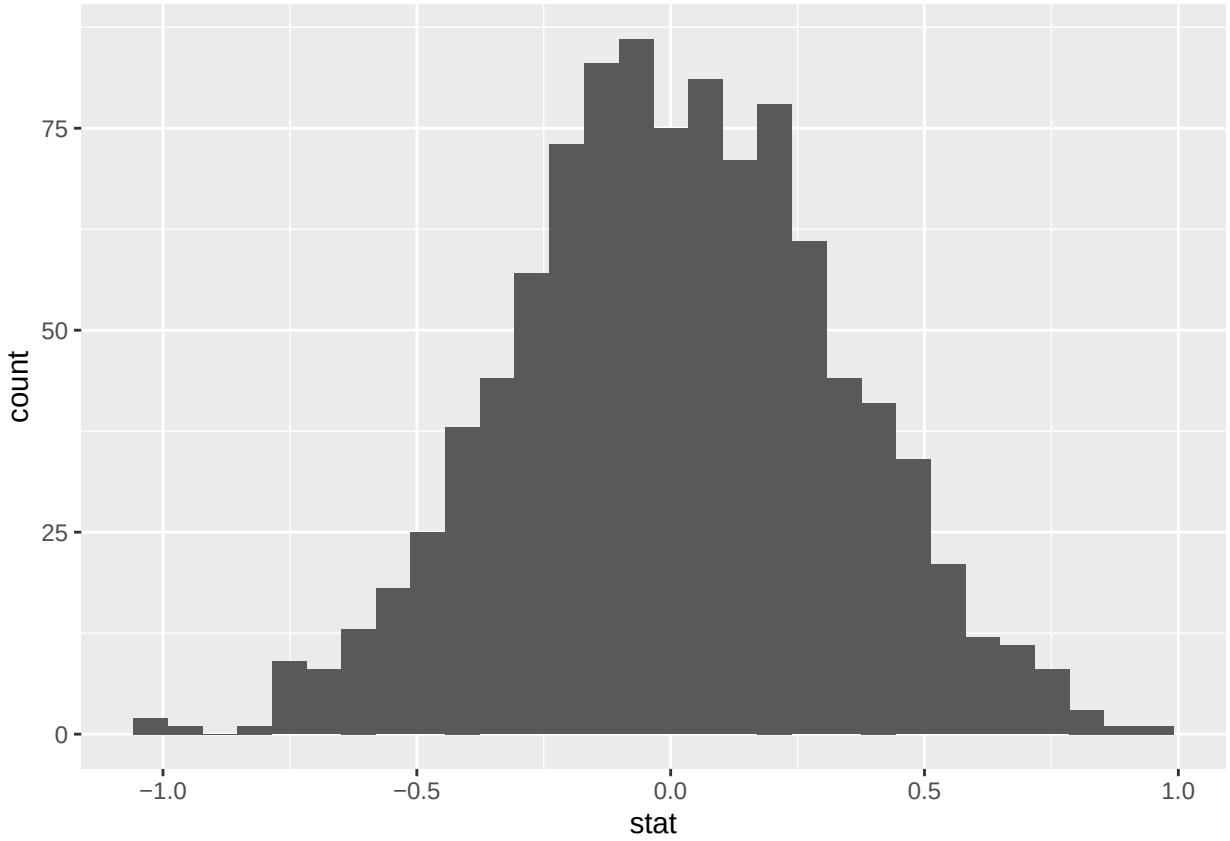
```
## Warning: Removed 946 rows containing missing values.
```

```
null_dist <- x %>%
  drop_na(physical_3plus) %>%
  specify(weight ~ physical_3plus) %>%
  hypothesize(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "diff in means", order = c("yes", "no"))
```

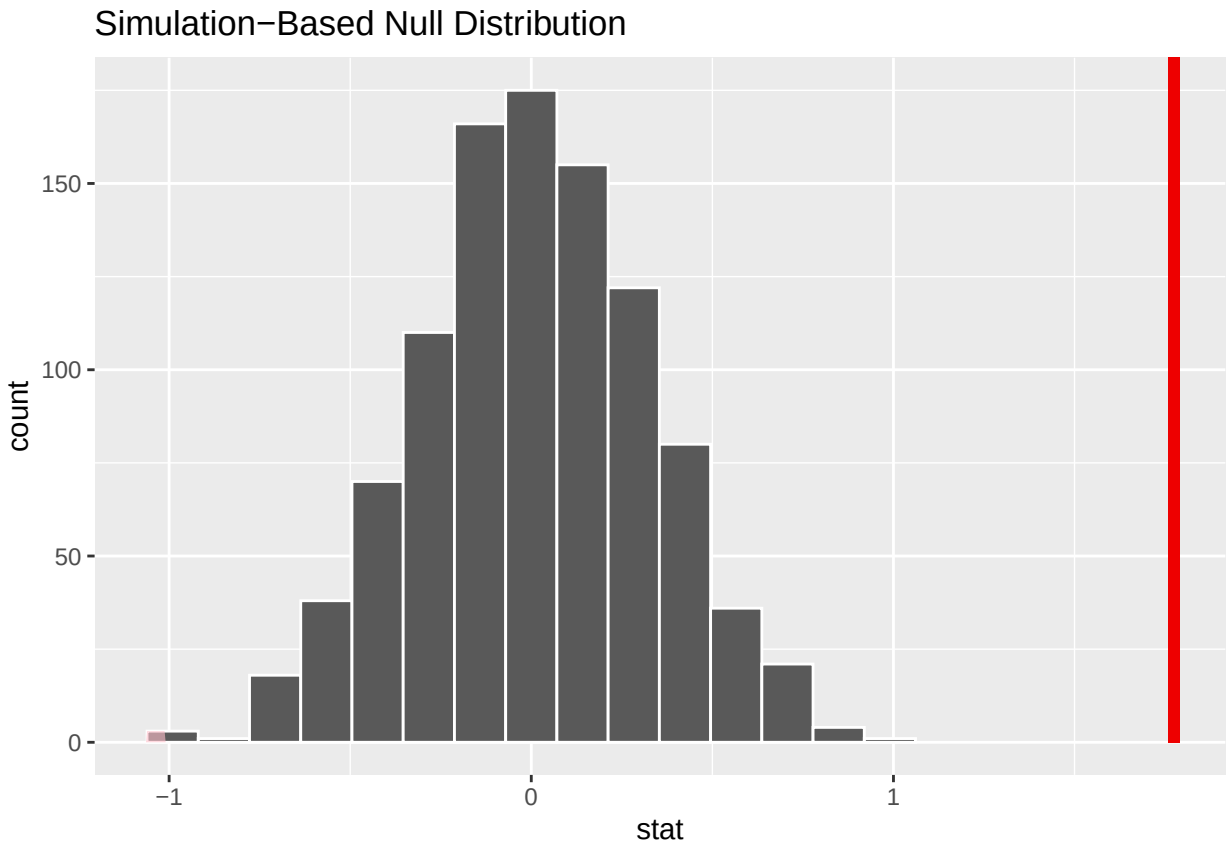
```
## Warning: Removed 946 rows containing missing values.
```

```
ggplot(data = null_dist, aes(x = stat)) +
  geom_histogram()
```

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




```
visualize(null_dist) +
  shade_p_value(obs_stat = obs_diff, direction = "two_sided")
```




Exercise 6

How many of these null permutations have a difference of at least obs_stat?

None of the null permutations have a difference of at least obs_stat

Exercise 7

Construct and record a confidence interval for the difference between the weights of those who exercise at least three times a week and those who don't, and interpret this interval in context of the data.

```
x %>%
  drop_na(physical_3plus) %>%
  specify(weight ~ physical_3plus) %>%
  hypothesise(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "diff in means", order = c("yes", "no")) %>%
  get_ci(level = 0.95)
```

```
## Warning: Removed 946 rows containing missing values.
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   -0.619    0.632
```

The weight difference between students who exercise 3 or more times a week and students who exercise less than three times a week falls within the range of -0.61lbs and 0.63lbs. Since this range includes 0, we can't say for sure if there is a difference in the average weights between these two groups.

Exercise 8

Calculate a 95% confidence interval for the average height in meters (height) and interpret it in context.

```
z <- 1.96
avg_height <- mean(x$height, na.rm = TRUE)
sd_height <- sd(x$height, na.rm = TRUE)
n_height <- x %>%
  summarise(freq = table(height)) %>%
  summarise(n = sum(freq, na.rm = TRUE))
```

```
## Warning: Returning more (or less) than 1 row per `summarise()` group was deprecated in
## dplyr 1.1.0.
## i Please use `reframe()` instead.
## i When switching from `summarise()` to `reframe()`, remember that `reframe()`
##   always returns an ungrouped data frame and adjust accordingly.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
upper_ci <- avg_height + z*(sd_height/sqrt(n_height))
lower_ci <- avg_height - z*(sd_height/sqrt(n_height))
```

```
print(lower_ci)
```

```
##          n
## 1 1.689411
```

```
print(upper_ci)
```

```
##          n
## 1 1.693071
```

at 95% confidence, the average height of students should fall between 1.6894 and 1.6930

Exercise 9

Calculate a new confidence interval for the same parameter at the 90% confidence level. Comment on the width of this interval versus the one obtained in the previous exercise.

```
z <- 1.645
avg_height <- mean(x$height, na.rm = TRUE)
sd_height <- sd(x$height, na.rm = TRUE)
n_height <- x %>%
  summarise(freq = table(height)) %>%
  summarise(n = sum(freq, na.rm = TRUE))
```

```
## Warning: Returning more (or less) than 1 row per `summarise()` group was deprecated in
## dplyr 1.1.0.
## i Please use `reframe()` instead.
## i When switching from `summarise()` to `reframe()`, remember that `reframe()`
##   always returns an ungrouped data frame and adjust accordingly.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
upper_ci <- avg_height + z*(sd_height/sqrt(n_height))
lower_ci <- avg_height - z*(sd_height/sqrt(n_height))
```

```
print(lower_ci)
```

```
##          n
## 1 1.689705
```

```
print(upper_ci)
```

```
##          n
## 1 1.692777
```

The 90% CI has a slightly more narrow width with a lower CI or 1.6897 and an upper CI of 1.6927 vs 1.6894 and 1.6930 at a 95% CI.

Exercise 10

Conduct a hypothesis test evaluating whether the average height is different for those who exercise at least three times a week and those who don't.

H1: Students who work out three or more times a week may have an average height that differs from those who do not. H0: The work out frequency does not impact a student's height.

```
obs_diff_height <- x %>%
  drop_na(physical_3plus) %>%
  specify(height ~ physical_3plus) %>%
  calculate(stat = "diff in means", order = c("yes", "no"))
```

```
## Warning: Removed 946 rows containing missing values.
```

```
nulldist_height <- x %>%
  drop_na(physical_3plus) %>%
  specify(height ~ physical_3plus) %>%
  hypothesise(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "diff in means", order = c("yes", "no"))
```

```
## Warning: Removed 946 rows containing missing values.
```

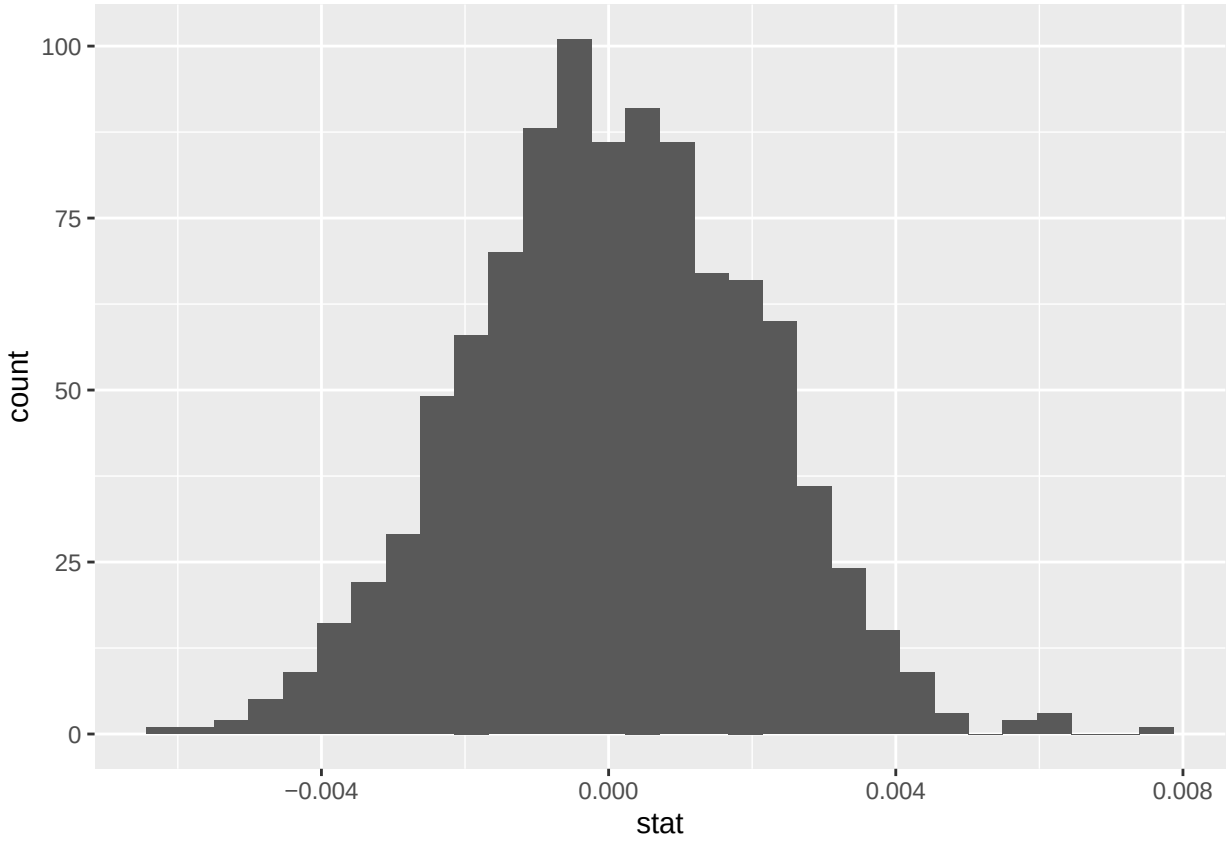
```
print(nulldist_height)
```

```
## Response: height (numeric)
## Explanatory: physical_3plus (factor)
## Null Hypothesis: independence
## # A tibble: 1,000 x 2
##    replicate      stat
##        <int>     <dbl>
##  1         1 -0.000776
##  2         2 -0.000216
##  3         3  0.00205
```

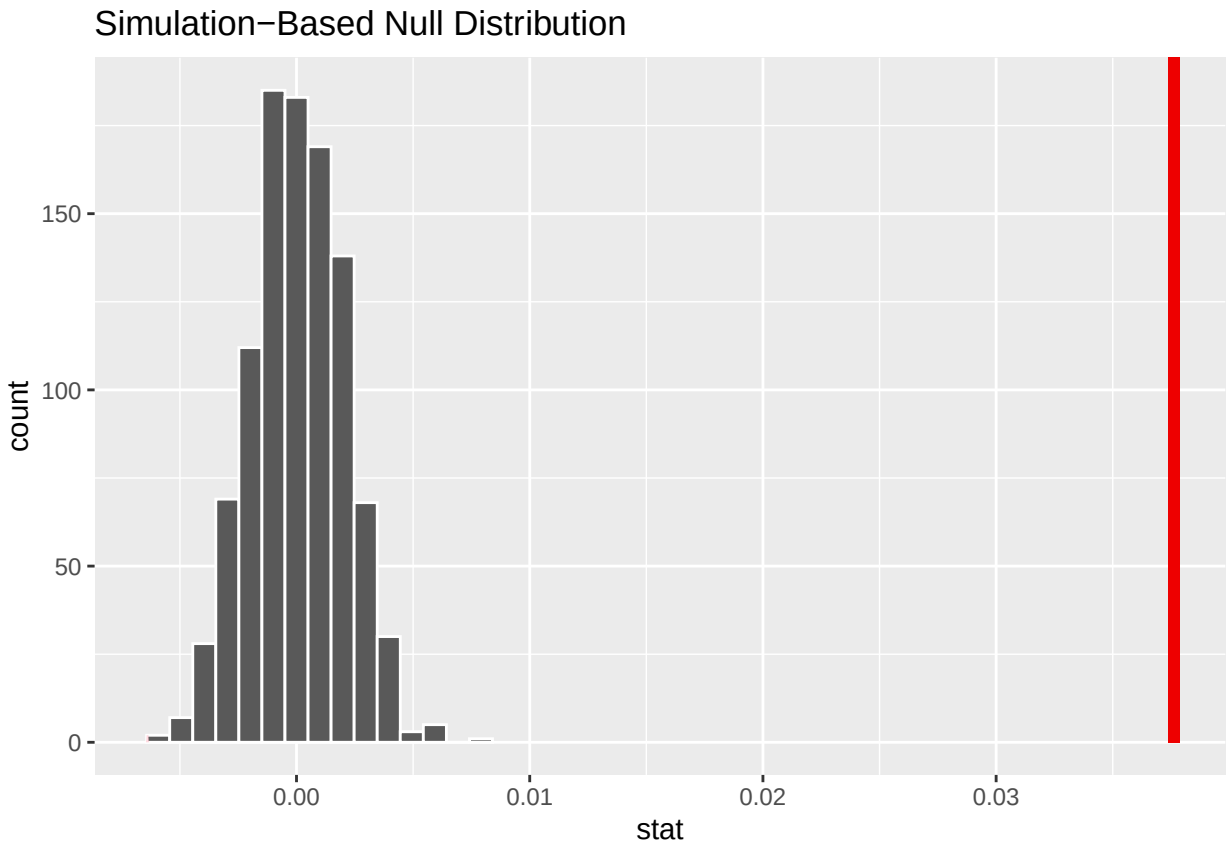
```
##  4         4  0.00210
##  5         5  0.00194
##  6         6 -0.00222
##  7         7  0.00237
##  8         8 -0.00224
##  9         9  0.000495
## 10        10  0.00196
## # i 990 more rows
```

```
ggplot(data = nulldist_height, aes(x = stat)) +
  geom_histogram()
```

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




```
visualize(nulldist_height) +
  shade_p_value(obs_stat = obs_diff_height, direction = "two_sided")
```




```
nulldist_height %>%
  get_p_value(obs_stat = obs_diff_height, direction = "two_sided")
```

```
## Warning: Please be cautious in reporting a p-value of 0. This result is an approximation
## based on the number of 'reps' chosen in the 'generate()' step.
## i See 'get_p_value()' ('?infer::get_p_value()') for more information.
```

```
## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1       0
```

Because the p value is below .05, we can reject the null hypothesis.

Exercise 11

Now, a non-inference task: Determine the number of different options there are in the dataset for the hours_tv_per_school_day there are.

```
table(x$hours_tv_per_school_day)
```

```
##
##           <1            1            2            3            4           5+
##         2168         1750         2705         2139         1048         1595
## do not watch
##         1840
```

There are 7 options

Exercise 12

Come up with a research question evaluating the relationship between height or weight and sleep. Formulate the question in a way that it can be answered using a hypothesis test and/or a confidence interval. Report the statistical results, and also provide an explanation in plain language. Be sure to check all assumptions, state your [p-value] level, and conclude in context

H0: The amount of sleep a student gets does not have an impact on the student's averge weight. H1: students who get five or more hours of sleep per school night will have a lower average weight than students who get less than five hours of sleep a night.

```
x <- x %>%
  mutate(sleep_adjusted = case_when(
    school_night_hours_sleep == "<5" ~ "4",
    school_night_hours_sleep %in% c("5", "6", "7", "8", "9") | grepl("^10+", school_night_hours_sleep) ~
    TRUE ~ school_night_hours_sleep
  )) %>%
  mutate(sleep_5plus = ifelse(as.numeric(sleep_adjusted) > 4, "yes", "no"))
```

```
obs_diff_sleep <- x %>%
  drop_na(sleep_5plus) %>%
  specify(weight ~ sleep_5plus) %>%
  calculate(stat = "diff in means", order = c("yes", "no"))
```

```
## Warning: Removed 854 rows containing missing values.
```

```
nulldist_sleep5 <- x %>%
  drop_na(sleep_5plus) %>%
  specify(weight ~ sleep_5plus) %>%
  hypothesise(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "diff in means", order = c("yes", "no"))
```

```
## Warning: Removed 854 rows containing missing values.
```

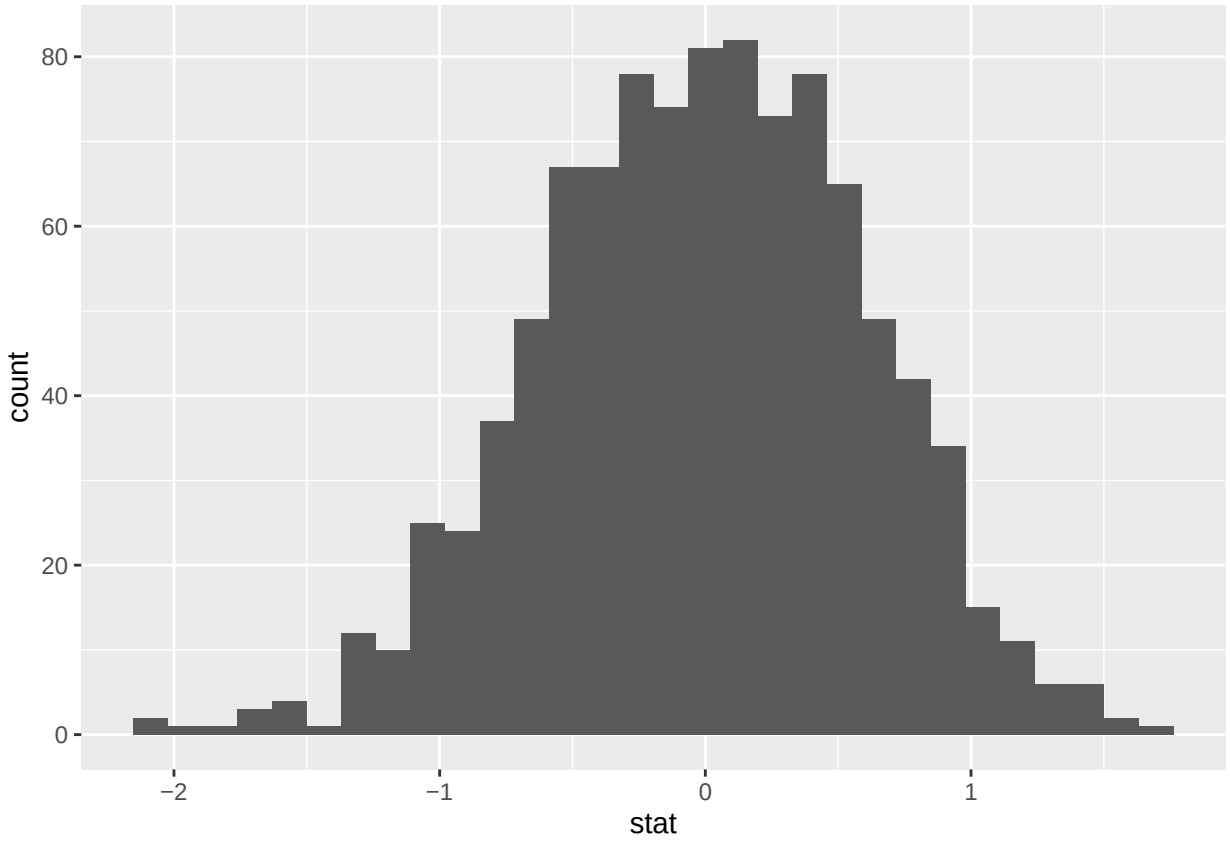
```
print(nulldist_sleep5)
```

```
## Response: weight (numeric)
## Explanatory: sleep_5plus (factor)
## Null Hypothesis: independence
## # A tibble: 1,000 x 2
##    replicate    stat
##        <int>   <dbl>
##  1         1  1.06
##  2         2 -0.171
##  3         3  0.868
##  4         4  1.02
##  5         5 -0.0234
##  6         6 -0.397
##  7         7  0.0324
##  8         8  0.739
```

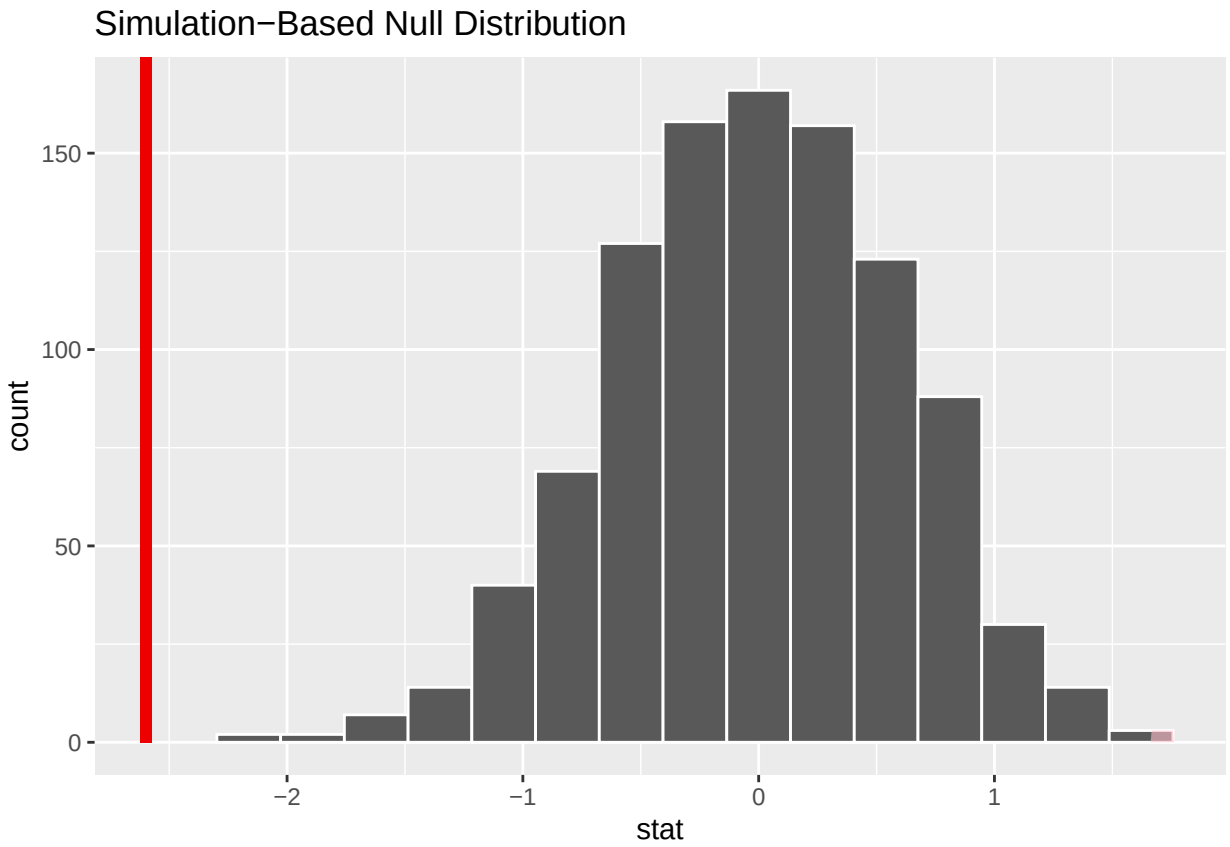
```
##  9         9  0.653
## 10        10  0.328
## # i 990 more rows
```

```
ggplot(data = nulldist_sleep5, aes(x = stat)) +
  geom_histogram()
```

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




```
visualize(nulldist_sleep5) +
  shade_p_value(obs_stat = obs_diff_sleep, direction = "two_sided")
```




```
nulldist_sleep5 %>%
  get_p_value(obs_stat = obs_diff_sleep, direction = "two_sided")
```

```
## Warning: Please be cautious in reporting a p-value of 0. This result is an approximation
## based on the number of ‘reps‘ chosen in the ‘generate()‘ step.
## i See ‘get_p_value()‘ (‘?infer::get_p_value()‘) for more information.
```

```
## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1       0
```

Because the p-value is below .05, we may reject the null hypothesis.

```
x %>%
  drop_na(sleep_5plus) %>%
  specify(weight ~ sleep_5plus) %>%
  hypothesise(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "diff in means", order = c("yes", "no")) %>%
  get_ci(level = 0.95)
```

```
## Warning: Removed 854 rows containing missing values.
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1    -1.20     1.14
```

However, looking at 95% CI, our range holds a value of 0, so it is possible that the average weights of those who sleep 5 or more hours a night aligns with the average weights of students who do not.