Hypothesis testing via simulation

Test for independence

Dr. Maria Tackett

11.07.19

STA 199

Click for PDF of slides

Announcements

- HW 04 **due Thu, Nov 14 at 11:59p**
- Project data analysis due December 3

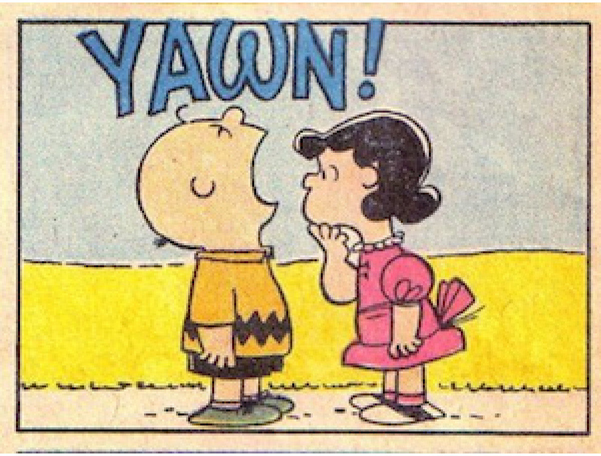
Hypothesis testing framework

- Start with a null hypothesis, H_0, that represents the status quo
- Set an alternative hypothesis, H_A, that represents the research question, i.e. what we're testing for
- Conduct a hypothesis test under the assumption that the null hypothesis is true and calculate a p-value (probability of observed or more extreme outcome given that the null hypothesis is true)
  - if the test results suggest that the data do not provide convincing evidence for the alternative hypothesis, stick with the null hypothesis
  - if they do, then reject the null hypothesis in favor of the alternative

Testing for independence

Is yawning contagious?

Do you think yawning is contagious?



Is yawning contagious?

An experiment conducted by the MythBusters tested if a person can be subconsciously influenced into yawning if another person near them yawns.

https://www.discovery.com/tv-shows/mythbusters/videos/is-yawning-contagious-2

Do we buy their conclusion?

Study description

In this study 50 people were randomly assigned to two groups: 34 to a group where a person near them yawned (treatment) and 16 to a control group where they didn't see someone yawn (control).

```r
mb_yawn <- read_csv("data/mb-yawn.csv")
```

```r
mb_yawn %>%
  count(group, outcome)
```

```
## # A tibble: 4 x 3
##   group     outcome       n
##   <chr>     <chr>     <int>
## 1 control   not yawn     12
## 2 control   yawn          4
## 3 treatment not yawn     24
## 4 treatment yawn         10
```

Proportion of yawners

```r
mb_yawn %>%
  count(group, outcome) %>%
  group_by(group) %>%
  mutate(p_hat = n / sum(n))
```

```
## # A tibble: 4 x 4
## # Groups:   group [2]
##   group     outcome       n p_hat
##   <chr>     <chr>     <int> <dbl>
## 1 control   not yawn     12 0.75
## 2 control   yawn          4 0.25
## 3 treatment not yawn     24 0.706
## 4 treatment yawn         10 0.294
```

- Proportion of yawners in the treatment group: $\frac{10}{34} = 0.2941$
- Proportion of yawners in the control group: $\frac{4}{16} = 0.25$
- Difference: $0.2941 - 0.25 = 0.0441$
- Our results match the ones calculated on the MythBusters episode.

Independence?

Based on the proportions we calculated, do you think yawning is really contagious, i.e. are seeing someone yawn and yawning dependent?

```
## # A tibble: 4 x 4
## # Groups:   group [2]
##   group     outcome       n p_hat
##   <chr>     <chr>     <int> <dbl>
## 1 control   not yawn     12 0.75 
## 2 control   yawn          4 0.25 
## 3 treatment not yawn     24 0.706
## 4 treatment yawn         10 0.294
```

Dependence, or another possible explanation?

- The observed differences might suggest that yawning is contagious, i.e. seeing someone yawn and yawning are dependent.
- But the differences are small enough that we might wonder if they might simple be **due to chance**.
- Perhaps if we were to repeat the experiment, we would see slightly different results.
- So we will do just that - well, somewhat - and see what happens.
- Instead of actually conducting the experiment many times, we will **simulate** our results.

Two competing claims

- "There is nothing going on." Yawning and seeing someone yawn are **independent**, yawning is not contagious, observed difference in proportions is simply due to chance. → Null hypothesis

- "There is something going on." Yawning and seeing someone yawn are **dependent**, yawning is contagious, observed difference in proportions is not due to chance. → Alternative hypothesis

Simulation setup

1. A regular deck of cards is comprised of 52 cards: 4 aces, 4 of numbers 2-10, 4 jacks, 4 queens, and 4 kings.

2. Take out two aces from the deck of cards and set them aside.

   - Take out Jokers if you have them.

3. The remaining 50 playing cards to represent each participant in the study:

   - 14 face cards (including the 2 aces) represent the people who yawn.
   - 36 non-face cards represent the people who don't yawn.

Running the simulation

1. Shuffle the 50 cards at least 7 times[1] to ensure that the cards counted out are from a random process.

2. Count out the top 16 cards and set them aside. These cards represent the people in the control group.

3. Out of the remaining 34 cards (treatment group) count the number of face cards (the number of people who yawned in the treatment group).

4. Calculate the difference in proportions of yawners (treatment - control).

5. Send one group member to the front of the room to input your group's difference in proportions.

[1] http://www.dartmouth.edu/~chance/course/topics/winning_number.html

Simulation by hand - Your results

Do the simulation results suggest that yawning is contagious, i.e. does seeing someone yawn and yawning appear to be dependent?

Simulation by hard - Your results

```
## # A tibble: 12 x 1
##    diff_in_prop
##           <dbl>
##  1            0
##  2            0
##  3            0
##  4            0
##  5            0
##  6            0
##  7            0
##  8            0
##  9            0
## 10            0
## 11            0
## 12            0
```

```
## # A tibble: 12 x 1
##    diff_in_prop
##           <dbl>
##  1            0
##  2            0
##  3            0
##  4            0
##  5            0
##  6            0
##  7            0
##  8            0
##  9            0
## 10            0
## 11            0
## 12            0
```

Simulation by computation

```r
set.seed(110719)
null_dist <- mb_yawn %>%
  specify(response = outcome, explanatory = group,
          success = "yawn") %>%
  hypothesize(null = "independence") %>%
  generate(1000, type = "permute") %>%
  calculate(stat = "diff in props",
            order = c("treatment", "control"))
```

Simulation by computation - 1

- Start with the data frame
- **Specify the variables**
    - **Since the response variable is categorical, specify the level which should be considered as "success"**

```
mb_yawn %>%
{{  specify(response = outcome, explanatory = group,
          success = "yawn") }}
```

Simulation by computation - 2

- Start with the data frame
- Specify the variables
  - Since the response variable is categorical, specify the level which should be considered as "success"
- **State the null hypothesis (yawning and whether or not you see someone yawn are independent)**

```r
mb_yawn %>%
  specify(response = outcome, explanatory = group,
          success = "yawn") %>%
  hypothesize(null = "independence")
```

Simulation by computation - 3

- Start with the data frame
- Specify the variables
  - Since the response variable is categorical, specify the level which should be considered as "success"
- State the null hypothesis (yawning and whether or not you see someone yawn are independent)
- **Generate simulated differences via permutation**

```
mb_yawn %>%
  specify(response = outcome, explanatory = group,
          success = "yawn") %>%
  hypothesize(null = "independence") %>%
  generate(1000, type = "permute")
```

Simulation by computation - 4

- Start with the data frame
- Specify the variables
    - Since the response variable is categorical, specify the level which should be considered as "success"
- State the null hypothesis (yawning and whether or not you see someone yawn are independent)
- Generate simulated differences via permutation
- **Calculate the sample statistic of interest (difference in propotions)**
    - **Since the explanatory variable is categorical, specify the order in which the subtraction should occur for the calculation of the sample statistic,** $(\hat{p}_{treatment} - \hat{p}_{control})$.

```
mb_yawn %>%
  specify(response = outcome, explanatory = group,
          success = "yawn") %>%
  hypothesize(null = "independence") %>%
  generate(1000, type = "permute") %>%
```

Simulation by computation - 0

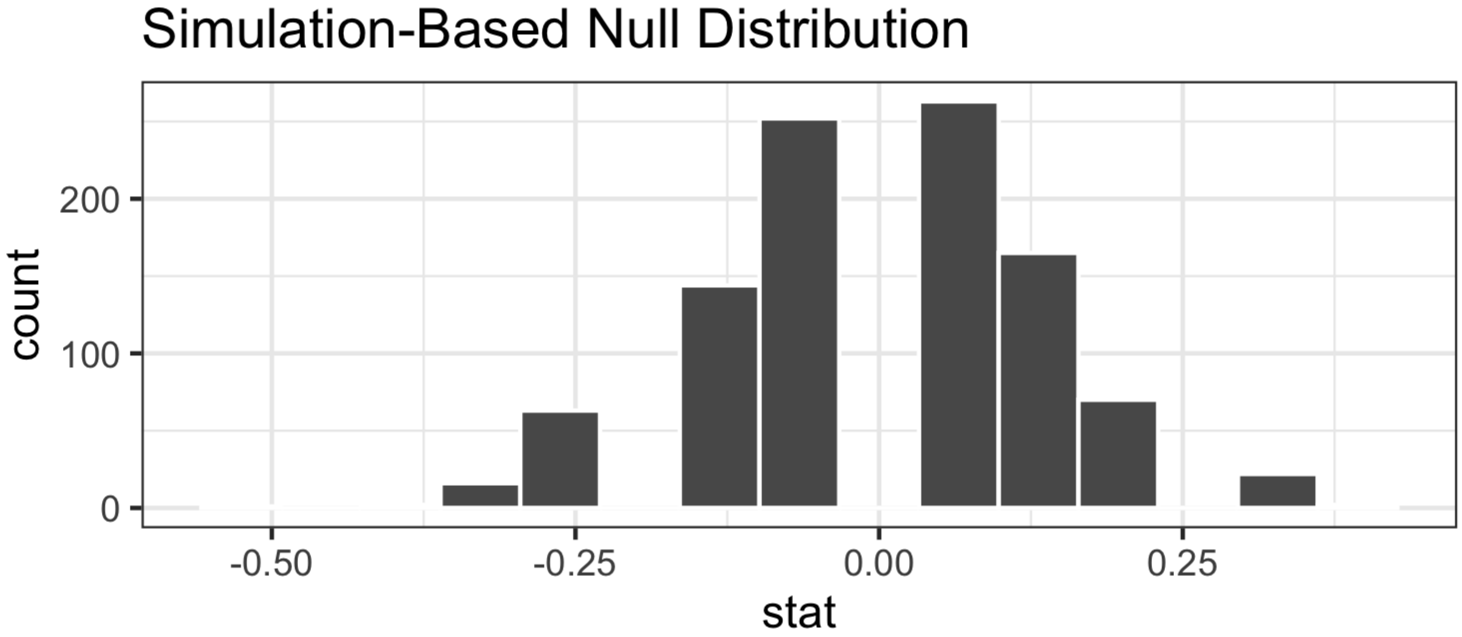
- **Save the result**
- Start with the data frame
- Specify the variables
- State the null hypothesis (yawning and whether or not you see someone yawn are independent)
- Generate simulated differences via permutation
- Calculate the sample statistic of interest (difference in proportions)

```r
null_dist <- mb_yawn %>%
  specify(response = outcome, explanatory = group,
          success = "yawn") %>%
  hypothesize(null = "independence") %>%
  generate(1000, type = "permute") %>%
  calculate(stat = "diff in props",
            order = c("treatment", "control"))
```

Visualizing the null distribution

What would you expect the center of the null distribution to be?

```r
visualize(null_dist)
```



Calculating the p-value, visually

What is the p-value, i.e. in what % of the simulations was the simulated difference in sample proportion at least as extreme as the observed difference in sample proportions?

```r
visualize(null_dist) +
  shade_p_value(obs_stat = 0.0441, direction = "greater")
```

Calculating the p-value, directly

```
null_dist %>%
  filter(stat >= 0.0441) %>%
  summarise(p_value = n()/nrow(null_dist))
```

```
## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1   0.522
```

```
get_p_value(null_dist, obs_stat = 0.0441, direction = "greater")
```

```
## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1   0.522
```

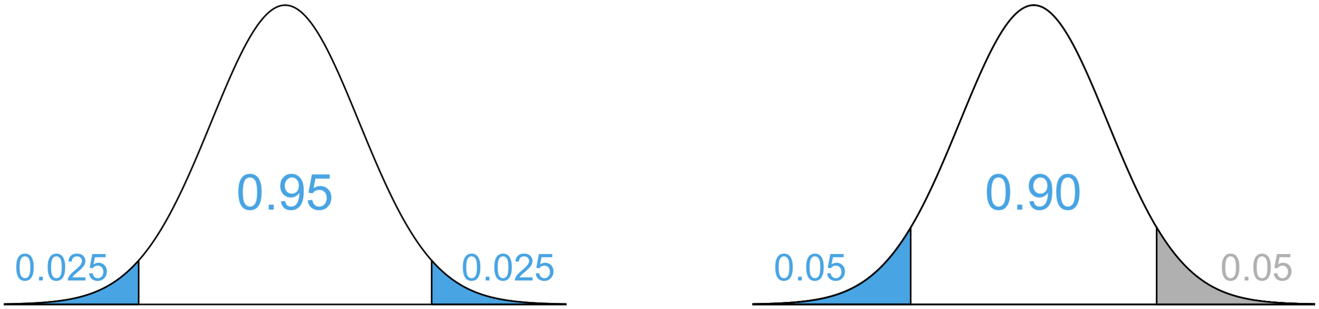
Conclusion

What is the conclusion of the hypothesis test?

Do you buy this conclusion?

Equivalency of confidence and significance levels

- Two sided alternative HT with $\alpha \rightarrow CL = 1 - \alpha$
- One sided alternative HT with $\alpha \rightarrow CL = 1 - (2 \times \alpha)$



Testing errors

- Type 1: Reject H_0 when you shouldn't have
  - P(Type 1 error) = α
- Type 2: Fail to reject H_0 when you should have
  - P(Type 2 error) is harder to calculate, but increases as α decreases

In a court of law

- Null hypothesis: Defendant is innocent
- Alternative hypothesis: Defendant is guilty

Which is worse: Type 1 or Type 2 error?

Probabilities of testing errors

- P(Type 1 error) = α
- P(Type 2 error) = 1 - Power
- Power = P(correctly rejecting the null hypothesis)

Fill in the blanks in terms of correctly/incorrectly rejecting/failing to reject the null hypothesis:

- α is the probability of __.
- 1 - Power is the probability of __.

Hypothesis testing for a single numeric variable

Hypothesis testing for a single numeric variable

Let's go back to the price to rent a one-bedroom apartment in Manhattan.

```r
library(tidyverse)
manhattan <- read_csv("data/manhattan.csv")
```

```r
manhattan %>% slice(1:10)
```

```
## # A tibble: 10 x 1
##     rent
##    <dbl>
##  1  3850
##  2  3800
##  3  2350
##  4  3200
##  5  2150
##  6  3267
##  7  2495
##  8  2349
##  9  3950
## 10  1795
```

```r
manhattan %>% slice(11:20)
```

```
## # A tibble: 10 x 1
##     rent
##    <dbl>
##  1  2145
##  2  2300
##  3  1775
##  4  2000
##  5  2175
##  6  2350
##  7  2550
##  8  4195
##  9  1470
## 10  2350
```

Rent in Manhattan

```r
manhattan %>% summarise(mean=mean(rent))
```

```
## # A tibble: 1 x 1
##    mean
##   <dbl>
## 1 2626.
```

According to the site Rent Jungle, the average price to rent an apartment in LA is around $2400. Is the average rent for a one-bedroom in Manhattan significantly different from $2400?

Simulation process

We will use bootstrapping to generate a sampling distribution under the assumption of the null hypothesis being true. Then, calculate the p-value to make a decision on the hypotheses.

1. Take a bootstrap sample - a random sample taken with replacement from the original sample, of the same size as the original sample.

2. Calculate the mean of the bootstrap sample.

3. Repeat steps (1) and (2) many times to create a bootstrap distribution - a distribution of bootstrap means.

4. Shift the bootstrap distribution to be centered at the null value by subtracting/adding the difference between the center of the bootstrap distribution and the null value to each bootstrap mean.

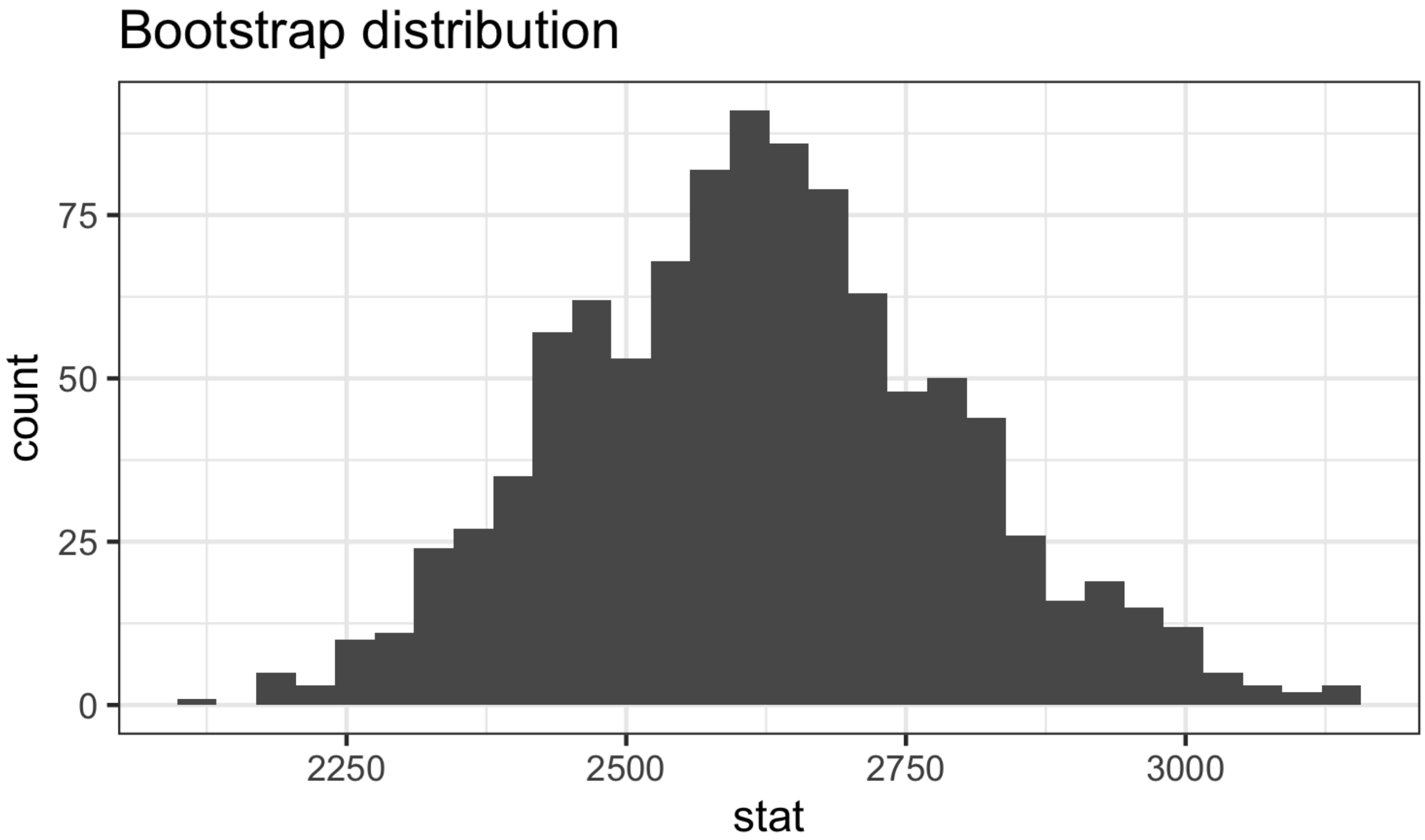
5. Calculate the p-value as the proportion of simulations that yield a sample mean at least as extreme as the observed sample mean.

Part 1: Take Bootstrap Sample

Take a bootstrap sample - a random sample taken with replacement from the original sample, of the same size as the original sample

```
rent_bootstrap <- manhattan %>%
  specify(response = "rent") %>%
  generate(reps = 1000, type = "bootstrap") %>%
  calculate(stat = "mean")
```

Part 1: Take Bootstrap Sample



Where is the center of the distribution? What should it be under the null

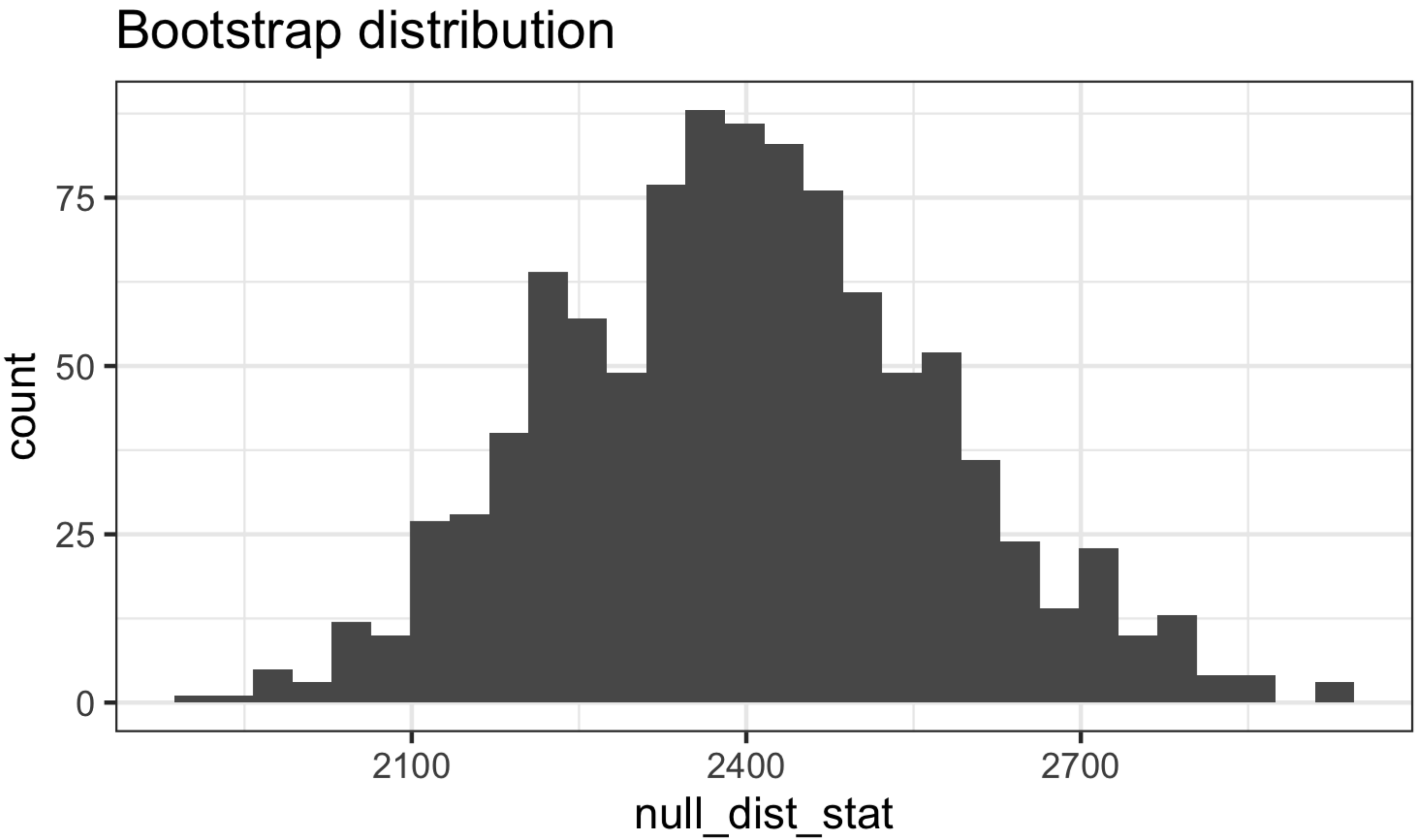
Part 2: Shift Distribution

Shift the bootstrap distribution to be centered at the null value by subtracting/adding the difference between the center of the bootstrap distribution and the null value to each bootstrap mean.

```r
rent_boot_mean <- rent_bootstrap %>%
  summarise(mean = mean(stat)) %>% pull()


rent_bootstrap <- rent_bootstrap %>%
  mutate(null_dist_stat = stat - (rent_boot_mean - 2400))
```

Part 2: Shift Distribution



Part 3: Calculate P-value

How should we calculate the p-value?

Part 3: Calculate p-value

Use the results below to calculate p-value:

```
rent_bootstrap %>%
  count(null_dist_stat > 2625.8)
```

```
## # A tibble: 2 x 2
##   `null_dist_stat > 2625.8`     n
##   <lgl>                     <int>
## 1 FALSE                       904
## 2 TRUE                         96
```