Problem Set 07: Hypothesis Testing Solutions

Professor Bradley Warner

June, 2022

Documentation:

We used all the resources available to instructors from the authors of Modern Dive.

Introduction

In this problem set we will use a small dataset that contains grade point average, demographic data, and academic performance metrics. We will use our knowledge of hypothesis testing to make decisions about the population.

Setup

First load the necessary packages:

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

For this problem set you will work with some grade-point-average (GPA) data for college freshman. The following will read in the data:

```
sat_gpa <- read_csv("https://rudeboybert.github.io/SDS220/static/PS/sat_gpa.csv")
```

Each row or case in this data frame is a student. The data includes:

- the (binary) gender of each student
- the math, verbal and total SAT scores for each student
- the GPA range of each student in high school (categorized as "low" or "high")
- the GPA of each student their first year of college on a numeric scale.

We will use hypothesis testing to answer the following questions:

- Is there a difference in male and female freshman GPAs?
- Is there a difference in total SAT score for students with a "low" and "high" high-school GPA?

Note, if you get stuck as you are working through this, it will be helpful to review Chapter 9 in ModernDive.

Gender differences in first-year GPA?

For this question, let's use a pre-determined α significance-level of 0.05.

Exploratory data analysis

Exercise 1

Calculate the mean GPA score for each gender, using the `group_by` and `summarize` commands from the `dplyr` package.

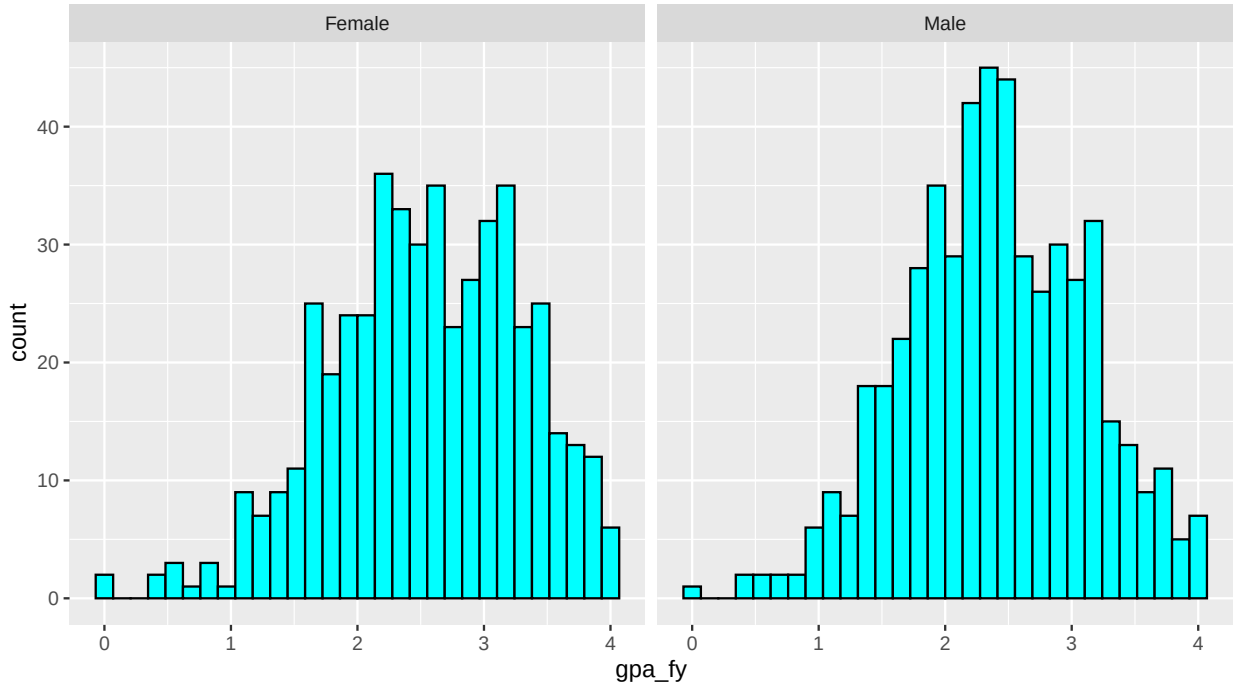
```
sat_gpa %>%
  group_by(sex) %>%
  summarize(gpa_fy = mean(gpa_fy),n=n())
```

```
## # A tibble: 2 x 3
##   sex    gpa_fy     n
##   <chr>   <dbl> <int>
## 1 Female   2.54   484
## 2 Male     2.40   516
```

Exercise 2

Generate a data visualization, faceted histograms, that displays the GPAs of the two groups. Be sure to include a title and label your axes.

```
ggplot(sat_gpa, aes(x = gpa_fy)) +
  geom_histogram(fill = "cyan",color="black") +
  facet_wrap(~sex)
```




```
  labs(title = "Grade Point Averages for first year college students",
       y = "GPA score")
```

```
## $y
## [1] "GPA score"
##
## $title
## [1] "Grade Point Averages for first year college students"
##
## attr(,"class")
## [1] "labels"
```

Stating a null hypothesis

We will now test the null hypothesis that there's no difference in population mean GPA between the genders at the population level. We can write this out in mathematical notation

$$H_0 : \mu_{male} = \mu_{female}$$
$$\text{vs } H_A : \mu_{male} \neq \mu_{female}$$

or expressed differently, that the difference is 0 or not:

$$H_0 : \mu_{male} - \mu_{female} = 0$$
$$\text{vs } H_A : \mu_{male} - \mu_{female} \neq 0$$

Testing the hypothesis

Here's how we use the `infer` package to conduct this hypothesis test:

Step 1: Calculate the observed difference

Exercise 3

Complete the code below to find the observed difference. Note that the order we choose does not matter here (female then male)... but we want you to use male then female!

```
 obs_diff_gpa_sex <- sat_gpa %>%
   specify(gpa_fy ~ sex) %>%
   calculate(stat = "diff in means", order = c("Male", "Female"))

obs_diff_gpa_sex
```

```
## Response: gpa_fy (numeric)
## Explanatory: sex (factor)
## # A tibble: 1 x 1
##     stat
##    <dbl>
## 1 -0.149
```

Step 2. Calculate the differences between male and females under the null

This step involves generating simulated values *as if* we lived in a world where there's no difference between the two groups. Going back to the idea of permutation, and tactile sampling, this is akin to shuffling the GPA scores between male and female labels (i.e. removing the structure to the data) just as we could have done with index cards.

Exercise 4

Complete the code to generate the simulated sampling distribution.

```
set.seed(8)
gpa_diff_under_null <- sat_gpa %>%
   specify(gpa_fy ~ sex) %>%
   hypothesize(null = "independence") %>%
   generate(reps = 5000, type = 'permute') %>%
   calculate(stat = "diff in means", order = c("Male", "Female"))

gpa_diff_under_null %>%
   slice(1:5)
```

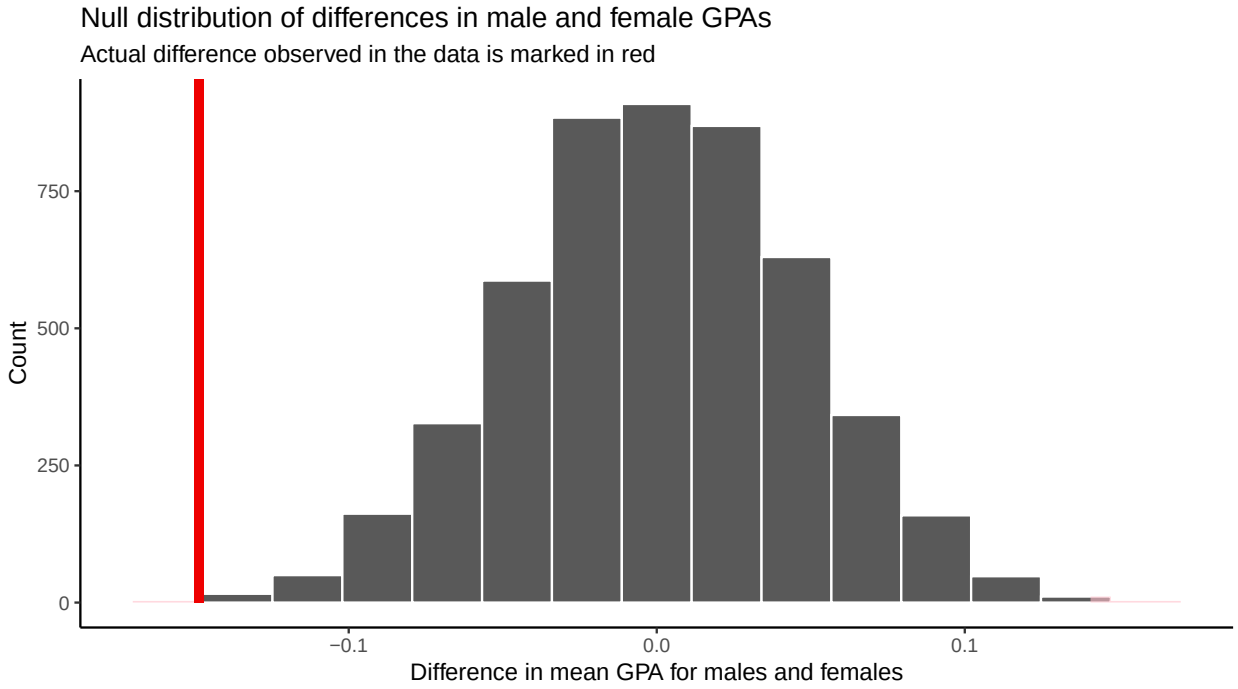
```
## Response: gpa_fy (numeric)
## Explanatory: sex (factor)
## Null Hypothesis: independence
## # A tibble: 5 x 2
##   replicate     stat
##       <int>    <dbl>
## 1         1  0.0486
## 2         2  0.0346
## 3         3 -0.00477
## 4         4 -0.0751
## 5         5 -0.0513
```

Step 3. Visualize how the observed difference compares to the null distribution

Exercise 5

When completed, the following plots the difference in means values we calculated for each of the different "shuffled" replicates. This is the null distribution of the difference in means. The red line will show the observed difference between male and female scores in the data.

```
visualize(gpa_diff_under_null) +
   shade_p_value(obs_stat = obs_diff_gpa_sex, direction = "both") +
   labs(x = "Difference in mean GPA for males and females", y = "Count",
        title = "Null distribution of differences in male and female GPAs",
        subtitle = "Actual difference observed in the data is marked in red") +
   theme_classic()
```



Note that zero is the center of this null distribution. The null hypothesis is that there is no difference between males and females in GPA score. In the permutations, zero was the most common difference, because observed GPA values were re-assigned to males and females **at random**.

Step 4: Calculate a p-value

Exercies 6

Calculate the p-value.

```
gpa_diff_under_null %>%
  get_pvalue(obs_stat = obs_diff_gpa_sex, direction = "both")
```

```
## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1  0.0012
```

Exercise 7

Fill in the blanks below to write up the results & conclusions for this test:

The mean GPA scores for females in our sample ($\bar{x}$ = **2.544587**) was **greater** than that of males ($\bar{x}$ = **2.396066**). This difference **was** statistically significant at $\alpha = 0.05$, (p = *_____**0.0012**). Given this I **would** reject the Null hypothesis and conclude that **females** have higher GPAs than **males** at the population level.

Step 5: Calculate a confidence interval for the difference

Exercise 8

Complete the code to generate a bootstrap sampling distribution for the difference between mean GPA scores for males and females.

```
boot_gpa_means <- sat_gpa %>%
   specify(gpa_fy ~ sex) %>%
   generate(reps = 5000, type = "bootstrap") %>%
   calculate(stat = "diff in means", order = c("Male", "Female"))

head(boot_gpa_means)
```

```
## Response: gpa_fy (numeric)
## Explanatory: sex (factor)
## # A tibble: 6 x 2
##   replicate    stat
##       <int>   <dbl>
## 1         1 -0.149
## 2         2 -0.150
## 3         3 -0.0756
## 4         4 -0.174
## 5         5 -0.135
## 6         6 -0.0672
```

Exercise 9

Find the 95% percentile bootstrap confidence interval.

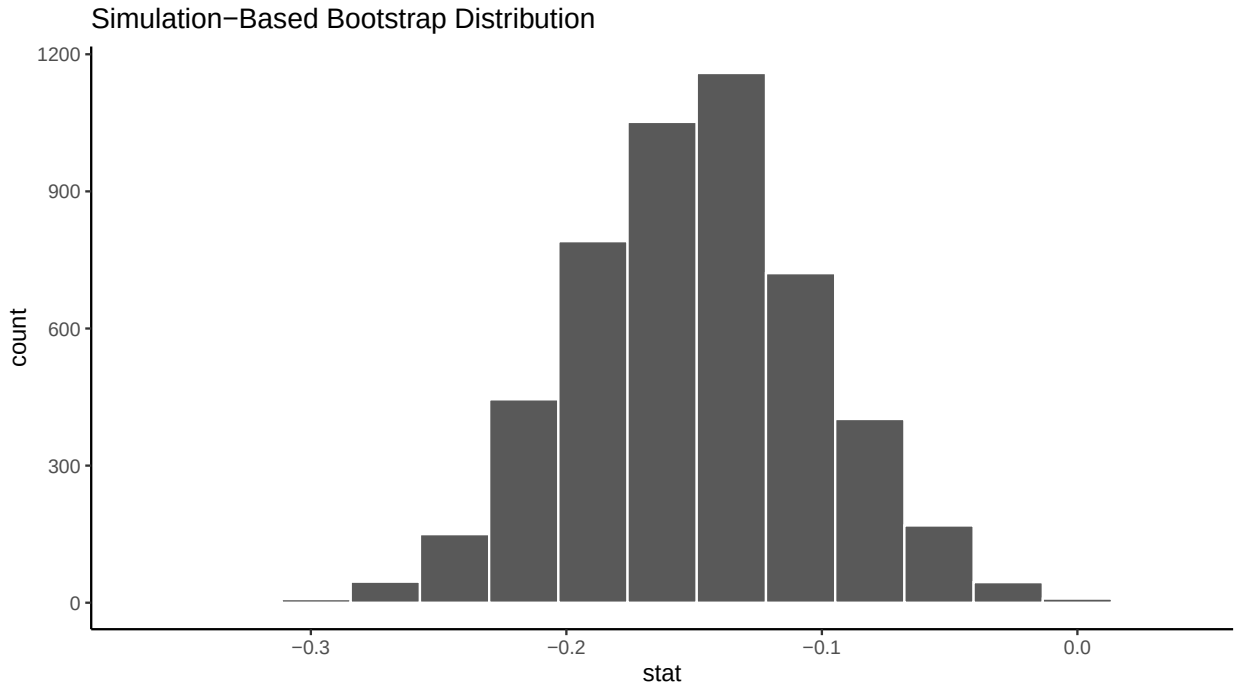
```
get_ci(boot_gpa_means,type="percentile")
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   -0.240  -0.0557
```

Exercise 10

Plot the bootstrap sampling distribution and comment on whether you think it is appropriate to use a confidence interval based on the standard error method.

```
visualize(boot_gpa_means) +
  theme_classic()
```



Answer:

Yes, the distribution appears to be symmetric and approximately normal. Thus we should confidence finding a confidence interval using the standard error method.

Traditional t-test

Note that all the above steps can be done with one line of code **if a slew of assumptions** like normality and equal variance of the groups are met.

```
t.test(gpa_fy ~ sex, var.equal = TRUE, data = sat_gpa)
```

```
##
##  Two Sample t-test
##
## data:  gpa_fy by sex
## t = 3.1828, df = 998, p-value = 0.001504
## alternative hypothesis: true difference in means between group Female and group Male is not equal to
## 95 percent confidence interval:
##  0.05695029 0.24009148
## sample estimates:
## mean in group Female   mean in group Male
##             2.544587             2.396066
```

Documenting software

- File creation date: 2022-06-20
- R version 4.1.3 (2022-03-10)
- `tidyverse` package version: 1.3.1
- `infer` package version: 1.0.0