Notes 13 - Hypothesis Tests for a Single Value

STS 2300 (Fall 2024)

Updated: 2024-11-10

Table of Contents

[Reading for Notes 13 1](#_Toc182139420)

[Learning Goals for Notes 13 2](#_Toc182139421)

[Hypothesis tests for 2](#_Toc182139422)

[Step 1 - Setting up the hypotheses 3](#_Toc182139423)

[Step 2 - Gathering and summarizing data 3](#_Toc182139424)

[Step 3 - Simulating a null distribution and finding a p-value 4](#_Toc182139425)

[Step 4 - Conclusions for hypothesis tests about 5](#_Toc182139426)

[Hypothesis tests for 6](#_Toc182139427)

[Step 1 - Setting up the hypotheses 6](#_Toc182139428)

[Step 2 - Gathering and summarizing data 7](#_Toc182139429)

[Step 3 - Simulating a null distribution and finding a p-value 7](#_Toc182139430)

[Step 4 - Conclusions for hypothesis tests about 9](#_Toc182139431)

[Theory-based hypothesis tests for or 9](#_Toc182139432)

[Using prop.test() for hypotheses about 10](#_Toc182139433)

[Using t.test() for hypotheses about 10](#_Toc182139434)

[Revisiting the Learning Goals for Notes 13 11](#_Toc182139435)

# Reading for Notes 13

Read [Chapter 9 (Sections 9.3 and 9.4) of the Modern Dive textbook](https://moderndive.com/9-hypothesis-testing.html) to supplement Notes 13.

# Learning Goals for Notes 13

* Be able to set up the hypotheses for hypothesis tests for a single value (e.g., or ).
* Be able to use the infer package to conduct simulation-based hypothesis tests for a single value.
* Be able to use R to calculate p-values for theory-based hypothesis tests for and .
* Be able to write conclusions for hypothesis tests for a single value.

I’ll be using the following packages in this set of notes, so I’ll load them before I get started.

library(infer)  
library(ggplot2)  
library(dplyr)

# Hypothesis tests for

When our variable of interest is categorical, questions we have will be about proportions (or percentages). For example, “what proportion of American adults plan to vote in the upcoming election?” or “do a greater proportion of Elon students or Duke students plan to vote?” In each example above, my main variable of interest is whether people plan to vote or not. In the second example, I’m comparing this variable between two groups (which we will cover in Notes 14).

Suppose that public health officials in New York City are interested in studying drug resistant bacteria in mice in the city. Specifically, they want to know if they can claim that fewer than one fourth of mice in the city carry drug resistant bacteria. Clearly they won’t be able to study *every* mouse in New York City, so instead they collect a random sample of 416 mice. For each mouse, they can determine whether or not they carry the bacteria by studying their fecal matter. They plan to use as their significance level for a hypothesis test related to their question of interest.

(Note: This example is based on [this article](https://www.washingtonpost.com/news/speaking-of-science/wp/2018/04/17/1-in-4-new-york-city-mice-carry-drug-resistant-bacteria-study-finds/).)

## Step 1 - Setting up the hypotheses

When we have a hypothesis test for a single categorical variable, we will typically write our null hypothesis by saying our parameter of interest *equals* some number. Our alternative hypothesis is based on our question of interest. It will look **exactly** like our null hypothesis except that we replace the equals sign with a <, >, or ≠ sign.

**Practice:** For the example above, public health officials in New York City want to know if they can claim that fewer than one fourth of mice in the city carry drug resistant bacteria. What are our hypotheses? Define the parameter for this example.

vs.

where

=

## Step 2 - Gathering and summarizing data

We can find the data set on our class code repository. The code to read it into R is below:

mice <- read.csv("https://raw.githubusercontent.com/nbussberg/STS2300-Fall2024/refs/heads/main/Data/mice.csv")

If my parameter of interest is a population proportion (), I will calculate a sample proportion () to summarize my data.

**Practice:** Try using the infer package (or another method) to find the sample proportion of mice that have a drug resistant bacteria.

We can see that less than one quarter of the *sample* carries a drug resistant bacteria, but remember that the officials want to be able to say less than one quarter of **all** mice in NYC carry a drug resistant bacteria. To make this claim, we need a null distribution.

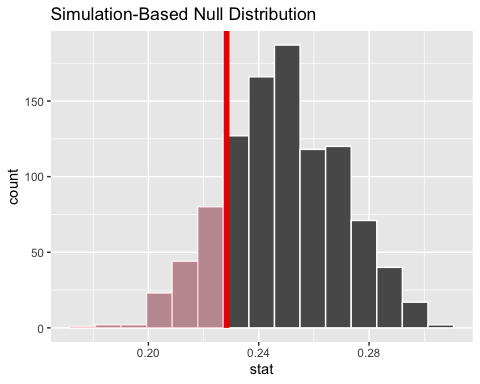
## Step 3 - Simulating a null distribution and finding a p-value

When our null hypothesis says \_\_\_, we can simulate data from this world by imagining we are flipping a weighted coin that comes up heads with probability equal to the value in our null hypothesis. In other words, we don’t permute the data like we did in Notes 12. Instead, we tell R what value to use for p (with the p = argument inside the hypothesize() function), and then can simulate data from that (using type = "draw" within the generate() function).

Below is the general code template for examples where our hypotheses relate to a single population proportion. Other than the changes mentioned above, it should look very similar to what we did when we generated bootstrap distributions for confidence intervals for a single proportion.

null\_dist <- data %>%  
 specify(formula = response ~ NULL, success = "level") %>%  
 hypothesize(null = "point", p = nullvalue) %>%  
 generate(reps = 1000, type = "draw") %>%  
 calculate(stat = "prop")

**Practice:** Update the code above to generate a null distribution of sample proportions (n = 416) in a world where the population proportion is 0.25. Include the shaded p-value on your graph. Your graph should look similar to the one below (but may differ slightly if you use a different seed).



**Question:** How do we know which way to shade the null distribution for our p-value?

**Answer:**

**Question:** Which value is in the center of our null distribution and why does this make sense? (Hint: Think about where our sample proportions are coming from)

**Answer:**

We saw in Notes 12 that we can use the get\_pvalue() function to find the p-value from our simulated null distribution. We will use this value in step 4 to make a conclusion based on how compatible our data is with the null model.

mice\_pval <- mice\_null %>%  
 get\_pvalue(obs\_stat = mice\_phat, direction = "left")  
mice\_pval

## # A tibble: 1 × 1  
## p\_value  
## <dbl>  
## 1 0.174

This p-value is telling me that in a world where 25% of all mice in NYC have at least one drug resistant bacteria, I would see a sample proportion of 0.228 or lower in around 17.4% of random samples of 416 mice.

## Step 4 - Conclusions for hypothesis tests about

We started this problem by saying that the NYC officials would use a significance level of . This means that they will:

* reject the null hypothesis if (because their data wouldn’t be very compatible with ).
* *fail to reject* the null hypothesis if the (because in that case the data would fit with what we might see if were true).

Since our p-value of 0.174 is above our data is consistent with the null hypothesis and we will *fail to reject* the null hypothesis. This doesn’t mean we necessarily think the null hypothesis is true; we just can’t rule it out. Thus, we don’t have enough evidence to say our alternative hypothesis is true. We can word our conclusion as follows:

**Conclusion** - Because our p-value of 0.174 > 0.05, we fail to reject our null hypothesis. There is insufficient evidence to conclude that fewer than one fourth of all mice in NYC carry a drug resistant bacteria.

In this class, our hypothesis test conclusions (like the one above), should do each of the following **important things**…

* compare the p-value to the significance level
* state what this means for the null hypothesis (reject or don’t reject)
* include a sentence about the level of evidence for the alternative hypothesis *in context* of the problem

# Hypothesis tests for

For most questions involving quantitative data, we will focus on looking at population means. However, the methods described below could also be used for population medians (or even standard deviations!).

You may have heard at some point in your life that the average human body temperature is 98.6°F. Suppose that you’ve decided you’d rather not just accept this and instead would like to conduct a hypothesis test to see if you find evidence that the population mean adult body temperature is in fact something else. We will use for our hypothesis test.

(Note: This example is based on [this article](https://www.wired.com/story/98-degrees-is-a-normal-body-temperature-right-not-quite/).)

## Step 1 - Setting up the hypotheses

In this case, our hypotheses will be about a population mean, . Just as before, the null hypothesis will say “nothing’s going on” or will be the status quo. The alternative hypothesis will be based on my question of interest.

**Practice:** Set up the hypotheses to test whether the population mean adult body temperature is something other than the conventional wisdom. Be sure to define .

vs.

where

=

## Step 2 - Gathering and summarizing data

Suppose I collected a random sample of 30 adults and recorded the body temperature of each of them. Below is code to read in the data.

bodytemps <- data.frame(temperature = c(97.39, 97.45, 97.96, 97.35, 96.74, 99.66,  
 98.21, 99.02, 96.78, 97.70, 96.90, 97.29,  
 97.99, 97.73, 98.18, 97.78, 97.17, 97.34,  
 97.56, 98.13, 97.77, 97.07, 97.13, 96.74,   
 99.10, 96.76, 96.19, 97.84, 96.80, 98.09))

**Practice:** Generate an appropriate graph of this data and use the infer package (or another method) to find the sample mean body temperature.

## Step 3 - Simulating a null distribution and finding a p-value

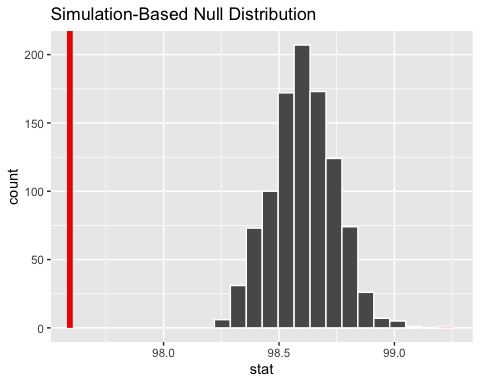
Below is generic code to generate a null distribution for a question involving a population mean.

null\_dist <- data %>%  
 specify(formula = response ~ NULL) %>%  
 hypothesize(null = "point", mu = nullvalue) %>%  
 generate(reps = 1000, type = "bootstrap") %>%  
 calculate(stat = "mean")

Notice that I am again using null = "point" but now I have a mu = \_\_ argument instead of a p = \_\_ argument in the hypothesize() function. This will help me to imagine a world where is some specified value (from my null hypothesis).

When we were creating a bootstrap distribution for a proportion, we could simulate from our null world by “flipping a coin.” That doesn’t work with quantitative variables that can take on many different values. Instead, we use type = "bootstrap" in generate() to create a bootstrap distribution (which is meant to mimic a sampling distribution). The distribution will be shifted at the end to be centered on the nullvalue.

**Practice:** Create a null distribution for the body temperature example. Shade your p-value on the graph and then calculate the value. The graph should look similar to this one.



## 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.

**Review:** What does it mean when we get a p-value of 0 using a simulation-based method like this?

**Answer:**

## Step 4 - Conclusions for hypothesis tests about

Recall that our hypothesis test conclusions, should do each of the following:

* compare the p-value to the significance level
* state what this means for the null hypothesis (reject or don’t reject)
* include a sentence about the level of evidence for the alternative hypothesis *in context* of the problem

**Practice:** Write a conclusion for our hypothesis test in context of this example.

**Conclusion:**

This is a fairly remarkable thing to be able to conclude so confidently based on only 30 people. However, hypothesis testing gives us a framework to do this (as long as our data is a random sample). Essentially we’ve seen that a random sample like this would be incredibly unlikely if . It seems only reasonable that a better explanation for what we saw then is that the null hypothesis is not true and equals something besides .

(Note: The idea of an average body temperature of 98.6°F actually comes from 1868, and it just sort of “stuck”. There is some information about the origins of that number [here](https://www.yalemedicine.org/stories/temperature-checks-covid/). In reality, our body temperatures change throughout the day and are affected by many factors. If you’re interested in diving into the issue more, you can check out the article I linked at the top of the section, too.)

# Theory-based hypothesis tests for or

We will use the same R functions and same conditions/assumptions for theory-based hypothesis tests as we did for theory-based confidence intervals.

This means that for hypothesis tests about proportions we use prop.test() and for hypothesis tests about means we use t.test(). Since we are not making a confidence interval, we do not need to use the conf.level argument.

## Using prop.test() for hypotheses about

Let’s start with our example about a population proportion. In addition to specifying x (number of successes) and n (sample size), we will also add:

* p - the value from our null hypothesis (default is 0.5)
* alternative - which way to shade based on (either “two.sided”, “greater”, or “less” - default is “two.sided”)

**Practice:** Use the output from the table() function below to write code that uses prop.test() to calculate a p-value for a theory-based hypothesis test of our mice in NY example.

table(mice$drugresist) # to find values for x and n

##   
## No Yes   
## 321 95

We should get at the bottom and a p-value of 0.1679, which should be similar to our simulation-based p-value.

**Note**: At this point in the notes, we’ve talked about , , and p-values. Each of these are discussed differently and represent different quantities.

* is the population proportion (called p)
* is the *sample* proportion (called p hat)
* p-value is the probability of getting data at least as extreme as what we saw if we lived in a world where the null hypothesis were true (called a p value)

## Using t.test() for hypotheses about

In addition to specifying our vector of quantitative values, we will also add:

* mu - the value from our null hypothesis (default is 0)
* alternative - which way to shade based on (either “two.sided”, “greater”, or “less” - default is “two.sided”)

**Practice:** Use the t.test() function to test vs.  for our bodytemps data. You should get the results below:

##   
## One Sample t-test  
##   
## data: bodytemps$temperature  
## t = -7.1709, df = 29, p-value = 6.813e-08  
## alternative hypothesis: true mean is not equal to 98.6  
## 95 percent confidence interval:  
## 97.30708 97.88092  
## sample estimates:  
## mean of x   
## 97.594

# Revisiting the Learning Goals for Notes 13

* Be able to set up the hypotheses for hypothesis tests for a single value (e.g., or ).
* Be able to use the infer package to conduct simulation-based hypothesis tests for a single value.
* Be able to use R to calculate p-values for theory-based hypothesis tests for and .
* Be able to write conclusions for hypothesis tests for a single value.