

4C-EffectSize

February 15, 2022

1 Lab 4C: Effect Size

1.0.1 PSYC 193L: Science of Learning Data Science

National Health and Nutrition Examination Survey (NHANES) The NHANES dataset contains survey data collected by the US National Center for Health Statistics (NCHS) which has conducted a series of health and nutrition surveys since the early 1960's. Since 1999 approximately 5,000 individuals of all ages are interviewed in their homes every year and complete the health examination component of the survey. The health examination is conducted in a mobile examination centre (MEC).

1.1 Part 0: Reminder about collaboration on lab assignments in PSYC 193L

We strongly believe in the value of collaborating with your peers for enhancing your learning experience in PSYC 193L. Being able to successfully collaborate with others is an important skill to have when you enter the workforce, and everyone can get better at collaboration with practice. However, for collaboration to be maximally valuable, we need to set some ground rules, building on the expectations laid out in the [course syllabus](#): Show Up, Try, Ask for Help When You Need It, Be Professional.

1.1.1 Guidelines

- Rotate responsibilities between group members.
- Choose a driver and a navigator.
- Discuss your thinking process openly with your group.
- Be supportive, respectful, and patient with one another.

Rotate responsibilities between group members. You will generally be working with the same people (from your discussion section) for each lab, and a new group of people for the next lab. Because you will be working with the same classmates for the next week or so, you will have the opportunity to share responsibilities with one another.

Choose a driver and a navigator. We suggest that one group member volunteer to act as the “driver” (and share their screen) while the other group members act as the “navigators.” Next time, it is a good idea to exchange roles, so that everyone gets a chance to act as the driver at least once, if possible.

Discuss your thinking process openly with your group. We suggest that you discuss the way you are thinking about each problem with your group. It is more important to us that you

gain practice explaining your reasoning to yourself and to your peers than it is to simply state what you think the “right answer” is, without explaining your reasoning.

Be supportive, respectful, and patient with one another. Try to give everyone an opportunity to play both a leading and supporting role. If you feel relatively comfortable with R, we encourage you to proactively encourage other members of your group who feel the least confident about writing R code to take a leading role. If you feel less confident about your R skills, please know that you are not alone! With practice and persistence over the course of the coming weeks, you will find your skills improving!

Every student is still responsible for submitting their own lab assignments. Although you are encouraged to work together on these lab assignments, please remember that everyone is responsible for submitting their own lab assignments.

```
[ ]: ## Run this code to load the required packages
suppressMessages(suppressWarnings(suppressPackageStartupMessages({
  require(tidyverse)
  require(supernova)
  require(ggformula)
  require(mosaic)
  require(lsr)
  require(NHANES)
})))
```

1.1.2 Learning objectives

The purpose of this lab is to get practice using R to fit statistical models that use one variable (i.e., explanatory variable) to explain variation in another (i.e., outcome variable).

In **Lab 4A**, we examine the empty model and compare it to a model with one binary explanatory variable (i.e. a variable where TRUE or FALSE are the only possible values).

In **Lab 4B** we consider *continuous* explanatory variables, exploring correlation and linear regression.

In **Lab 4C** we think about how to quantify (1) the size of an effect (e.g. the difference in heights between two groups) and (2) our confidence in parameter estimates (like the mean of a population, or the slope of a linear relationship between two continuous variables).

1.1.3 NHANES data preprocessing

```
[ ]: ## load in data and some column selection
Nall_ages <- NHANES %>%
  mutate("Height.inches" = Height / 2.54) %>% ## convert to inches
  select("ID", "Height.inches", "Age", "Gender") %>% ## just look at these
  ↪ columns
  distinct() %>% ## remove duplicate rows (same participant measured across
  ↪ multiple timepoints)
  drop_na()
```

```
## add column for child if Age < 18
Nall_ages <- Nall_ages %>%
  mutate(isChild = ifelse(Age < 18, TRUE, FALSE))

## Extract adult NHANES data
Nadult <- Nall_ages %>%
  filter(Age>=18)
```

1.2 Part 1: Using sampling distributions to estimate effect sizes

In previous parts of Lab 4, we have modeled `height` as a function of `ageGroup` and `height` as a function of `ageNum`. In Part 1 of this lab, we will be using a similar modeling approach to predict `height` based on `sex`. We will also be relating pieces of this model to the concept of **effect size** and **confidence interval**.

1.0 Please inspect the `Nadult` dataframe using `str()` to get an overview of this dataframe.

```
[ ]: # your code here
fail() # No Answer - remove if you provide an answer
```

1.1 Please construct a linear model predicting `Height.inches` on the basis of `Gender` in `Nadult`. Save the model as `height_sex_lm`.

```
[ ]: # your code here
fail() # No Answer - remove if you provide an answer
```

1.2 Using your model from 1.1, please make your best guess as to the average difference between men's and women's heights in the *population from which this data was sampled*. Save that estimate as `male_minus_female_ht`.

Note: Your answer should be expressed as the amount by which men are taller than women on average, so a positive value means men are taller, and a negative means women are taller.

```
[ ]: # your code here
fail() # No Answer - remove if you provide an answer
```

In 1.2 you reported your best guess as to the average difference between men's and women's heights in the population. This "best guess" also happens to be the mean difference between men's and women's in our sample!

Think back to Lab 3 and the relationship you observed between **sample size** (i.e., how many observations included in each sample) and the sampling distribution of the mean (i.e., how much the sample mean varies, or "wobbles," from sample to sample). You can think of the **variability** in this **sampling distribution** as reflecting our uncertainty in our estimate – the higher the variability (i.e., the more spread out), the greater the uncertainty; the lower the variability (i.e., the more tightly clustered), the lower the uncertainty. A common way to quantify this variability is to compute the **standard error**, which is defined as the *standard deviation of the sampling distribution*.

1.3 Suppose we instead derived our best estimate of the true population height difference between men and women from a sample that was 4x larger than the one we are currently using in this lab. Imagine you could keep resampling the same 4x larger sample from the same population and estimating the mean height difference each time. Using this 4x larger sample, do you expect the standard error on this estimate to be LARGER or SMALLER compared to the standard error of the estimate derived using the original sample size? Why do you think so?

YOUR ANSWER HERE

We know that our *parameter estimates* (like the parameter `male - female height difference`) are not perfect. Because of sampling variability, some samples will just happen to have lots of tall women and short men (leading to a small estimate of the height difference) and some will happen to have lots of short women and tall men (leading to a large estimate). But if we're sometimes too low, and sometimes too high, those effects will eventually cancel each other out *on average* (meaning in the imaginary case of taking lots and lots of samples, and calculating the difference in mean heights for each one).

1.4 Using R, we can actually verify that this is the case by conducting a computer simulation! We can pretend that NHANES is the population we are interested in (when in reality it is itself a *sample* from a larger population), and take lots of samples from it, from which we calculate the mean height difference. Let's do that! In Lab 3, we wrote some code for you that took lots of samples from `aliens`. Now, you get to write that code yourself! (Note that the code we ask you to write here is similar in end result to what was used in Lab 3, but use different functions - it probably won't be too helpful to refer back to the code itself from Lab 3.)

Please use `resample()` to simulate drawing a single random sample of size `n=200` from the `Nadult` dataframe, then refitting the same model from 1.1 to this smaller sample using the `lm()` function, then extracting the estimated mean height difference between men and women using the `b1()` function. There is need to save the result as a variable, just display the output of applying the `b1()` function.

Hint: Coursekata chapter 10.7 discusses how to use `resample()` to do this in a single line of code, so it might be a good idea to review that module for help with this question.

```
[ ]: # your code here
fail() # No Answer - remove if you provide an answer
```

1.5 Now please use the `do()` function to simulate drawing 1000 different random samples, all of them of size `n=200`, then refitting the same model from 1.1 to this smaller sample using the `lm()` function, then extracting the estimated mean height difference between men and women using the `b1()` function. In other words, please do exactly what you did in 1.4 but use the `do()` function to repeat the same operations 1000 times. Please save the result of this as a variable called `heightDiff_boot`.

Note: This kind of resampling with replacement in order to construct parameter estimates is known as “bootstrapping”, a technique that is widely used in modern statistics.

```
[ ]: # your code here
fail() # No Answer - remove if you provide an answer
```

1.6 Using your bootstrapped sample height differences in `heightDiff_boot`, construct a 95%

confidence interval on the mean height difference between men and women in `Nadult`.

Hint 1: Remember that a 95% confidence interval is the range of values such that 95% of the time, that interval contains the true mean. Thus, we want our interval to include the middle 95% of sampled mean height differences (with 2.5% of `b1s` remaining in each of the two tails).

Hint 2: We currently have 1000 (unsorted) sampled mean height differences. When they are sorted, the bottom $.025 \times 1000$ and the top $.025 \times 1000$ values should be excluded from your `CI95`.

Please save the lower bound of the CI as `CI95_heightDiff_lower` and the upper bound as `CI95_heightDiff_upper`.

```
[ ]: # your code here
      fail() # No Answer - remove if you provide an answer
```

1.7 Please create a histogram of these height differences. Please add vertical lines representing the “true” population mean (which in this case refers to the actual mean height difference computed for all observations in our `Nadult` dataframe). Please also add vertical lines representing the upper and lower bounds of your 95% confidence interval.

```
[ ]: # your code here
      fail() # No Answer - remove if you provide an answer
```

This visualization shows how a parameter estimate (like our men-women height difference) varies, but is centered about the mean of the population. Moreover, the probability of being off by a lot is lower than the probability of being off by just a bit. In fact, this distribution of sample `b1s` is approximately normal. Now that we’ve demonstrated this is the case, we can use that insight, plus some math from the Central Limit Theorem, to characterize the error of our estimate without needing to run simulations.

1.3 Part 2: Constructing a summary visualization (with error bars!)

The histogram in 1.7 visualizes how much the average difference in height between men and women is expected to vary across samples of size `n=200`.

According to this visualization, it looks like men are on average approximately 5.4 inches taller than women, and 95% of those simulated differences fell within 4.6 and 6.2 inches, so we can be fairly confident the true population parameter falls in that range.

But focusing on the *difference in height* doesn’t tell us how tall men and women are in absolute terms. So let’s create a summary visualization that displays the mean height of women and the mean height of men. Because there are two groups of interest that we are comparing to each other, a **bar plot** is a reasonable default choice. The goal of Part 2 is to create a bar plot that makes it easy to see both how tall men and women are on average in absolute terms (i.e., in units of inches), and how statistically reliable the difference in average difference in height is between the two groups.

To visually communicate the quantitative uncertainty we have in our estimate of the mean height difference between groups, we will also include **error bars** that represent the range of estimates that we would not be especially surprised to see, because we *expect* there to be random variation across different samples due to sampling variability.

2.1 Please use `favstats()` and `select()` pull out the means, sds, and counts of men’s and women’s `Height.inches` in `Nadult`. Save the result as `heights_summary`. **Hint:** Your result should be a data frame with two rows (one each for `Gender == female` and `male`) and four columns (`Gender`, `mean`, `sd`, `n`).

```
[ ]: # your code here
      fail() # No Answer - remove if you provide an answer
```

We’ll need to calculate a value for our error bars. Above, we used bootstrapping to construct a 95% confidence interval for the mean difference in height between men and women.

Now, let’s use the Central Limit Theorem to calculate the standard error of the mean for each group. Remember, the **standard error of the mean** (SEM) is shorthand for “*the standard deviation of the sampling distribution (of the sample mean)*.” This may be a mouthful to say, but each word plays an important part!

You can interpret the SEM for each `Gender` group as telling you about the average error made by an empty model fit to each group separately. In other words, the SEM for `men` would tell you how far off the empty model would be on average when predicting the height of men; the SEM for `women` would tell you how far off the empty model would be on average when predicting the height of women.

Note that neither of these group-specific SEMs would tell you about the average error when predicting the *difference in mean heights between groups*. Obtaining this latter SEM requires fitting a model predicting `Height.inches` from `Gender`, as we did in Part 1.1. Nevertheless, it can be useful to still construct summary visualizations that give the viewer a sense for how much variation there is within each group, prior to conveying information about how reliable the mean difference is between groups.

2.2 Please add a column to `heights_summary` called `sem`. This should contain the standard error of the mean for female and male heights. **Hint:** `heights_summary` already contains all the values you need for this calculation.

```
[ ]: # your code here
      fail() # No Answer - remove if you provide an answer
```

2.3 Let’s add two more columns to `heights_summary`, one for the lower bound of our error bars, and one for the upper bound. Let’s call them `errbar_low` and `errbar_up`, respectively. Generally, there are three common choices for what to represent with errorbars: (1) a confidence interval (usually 95%), (2) extending from one SEM below to one SEM above the mean, and (3; less common) one standard deviation below to one standard deviation above the mean.

Let’s get some practice constructing error bars that represent 1 SEM.

So: `errbar_low` should be one `sem` below the mean for each `Gender` and `errbar_up` should be one `sem` above the mean for each `Gender`.

```
[ ]: # your code here
      fail() # No Answer - remove if you provide an answer
```

2.4 Please use `gf_col()` to create a bar plot representing the mean heights of men and women in our sample (it works just like `gf_bar()` but doesn’t summarize over groups— we’ve already done

that!).

Also, use `gf_errorbar()` to represent our uncertainty in that estimate by adding error bars. Error bars should stretch from one `sem` below the mean of each group to one `sem` above the mean for each group, as calculated in 2.3.

Hint: `gf_errorbar()` takes a bit of a weird formula as an argument. The syntax is like this (sub in the necessary values): `gf_errorbar(lower_bound + upper_bound ~ group)`. It's often a good idea to also use the `width` argument and set it to something smaller than 1 (e.g. .5, .333), so the error bars aren't as wide as the bars representing the means.

```
[ ]: # your code here
fail() # No Answer - remove if you provide an answer
```

2.5 Please refine this plot, giving it an informative y-axis label, a title, and a caption briefly explaining what the errorbars represent (to provide clarity for readers, who might not assume we're using standard errors vs. a confidence interval).

Notice also that the error bars are quite narrow! To get a better look at their span, we can also tweak the y-axis limits. Feel free to change the color of the error bars to increase visibility against the grey bars, too. **Note:** it is often considered questionable practice not to start a y-axis at 0! It can make group differences look much larger than they are.

Still, for our current purposes, it's a useful trick. Here's a line of code you can modify and add to your plot code: `gf_refine(scale_y_continuous(limits = c(lower_y, upper_y), oob=rescale_none))`. Just replace `lower_y` and `upper_y` with bounds that you think make the errorbars visible enough, without distorting the size of the height difference too much. We have to use this fancier trick than the normal `gf_lims()` because `gf_col()` *really* likes to start at 0 (for the reason just mentioned) and we need to convince it not to.

```
[ ]: # your code here
fail() # No Answer - remove if you provide an answer
```

2.6 If instead of using the entire `Nadult` dataframe to derive these estimate of SEM and we had instead used a smaller sample of size `n=200` as in Part 1, would you expect the error bars to be larger (i.e., taller) or smaller (i.e., shorter) than they appear in 2.5? Why do you think so?

YOUR ANSWER HERE

At this point, we have a couple ways of talking about the difference in average height for men and women. We've built a model and examined `b1`, and also characterized the sampling distribution of that difference, so we can understand its variability. We can also visually display group means, with error bars representing our uncertainty.

One last very common measure of effect size: **Cohen's D**. Cohen's D is a standardized measure of the difference between two group means. Just like Z-scores measure deviation from the mean *in units of standard deviations* and correlations report the slope of the regression line through the Z-scores, Cohen's D measures how far apart two means are, *in units of standard deviations*. But our two variables (here, male and female heights), might have different standard deviations, so which do we use?

Neither, actually. Since the two standard deviations could be very different (though here they're

actually pretty close, as seen in 2.2), we need a way to *pool* them together. Without going into the details (see Coursekata chapter 7!), Cohen’s D basically takes a “weighted average” of the two standard deviations, where the “weight” for each standard deviation is related to the size of the sample.

2.7 We should understand more or less what Cohen’s D does, but there is a handy R function for calculating it. Please use `CohensD()` to calculate this measure of effect size. Its syntax is exactly the same as for `tally()` – first a formula like `variable ~ group`, then a `data=` argument.

Once you’ve run that calculation, please interpret what this effect size means in the context of our variables of interest (use the markdown cell below the code cell).

```
[ ]: # your code here  
fail() # No Answer - remove if you provide an answer
```

YOUR ANSWER HERE

1.3.1 LAB 4 Reflection

As part of each lab we ask that you look back at the three parts that you completed and write a short reflection about the experience with the lab.

- What parts did you find most challenging?
- What was the most interesting part?
- How does this lab connect to the CourseKata readings?
- What concepts did this lab cover?
- Is there anything else you would like us to know about your experience with the lab? (Note: this response is a large part of your grade on the overall lab, please take time to give us thoughtful feedback, as it will help us make the course better for future students!)

WRITE YOUR REFLECTION HERE. PLEASE DO YOUR BEST TO ADDRESS EACH OF THE QUESTIONS ABOVE.

Before submitting this lab via DataHub, remember to consult the pre-submission checklist on the course website.