Inference for numerical data

Getting Started

Load packages

In this lab, we will explore and visualize the data using the **tidyverse** suite of packages, and perform statistical inference using **infer**. The data can be found in the companion package for OpenIntro resources, **openintro**.

Let's load the packages.

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

The data

Every two years, the Centers for Disease Control and Prevention conduct the Youth Risk Behavior Surveillance System (YRBSS) survey, where it takes data from high schoolers (9th through 12th grade), to analyze health patterns. You will work with a selected group of variables from a random sample of observations during one of the years the YRBSS was conducted.

Load the `yrbss` data set into your workspace.

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

There are observations on 13 different variables, some categorical and some numerical. The meaning of each variable can be found by bringing up the help file:

```
?yrbss
```

Exercise 1

1. What are the cases in this data set? How many cases are there in our sample? The cases are: helmet_12m, text_while_driving_30d , physically_active_7d, hours_tv_per_school_day, strength_training_7d, school_night_hours_sleep There are 13583 observations and 13 columns with 6 different cases.

Remember that you can answer this question by viewing the data in the data viewer or by using the following command:

```
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"~
```

Exploratory data analysis

You will first start with analyzing the weight of the participants in kilograms: `weight`.

Using visualization and summary statistics, describe the distribution of weights. The `summary` function can be useful.

```
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

2. How many observations are we missing weights from? There are 1004 missing data in weights columns

Next, consider the possible relationship between a high schooler's weight and their physical activity. Plotting the data is a useful first step because it helps us quickly visualize trends, identify strong associations, and develop research questions.

First, let's create a new variable `physical_3plus`, which will be coded as either "yes" if they are physically active for at least 3 days a week, and "no" if not.

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

Exercise 3

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?

```
yrbss %>% ggplot(aes(x = physical_3plus, y = weight)) + geom_boxplot()
```
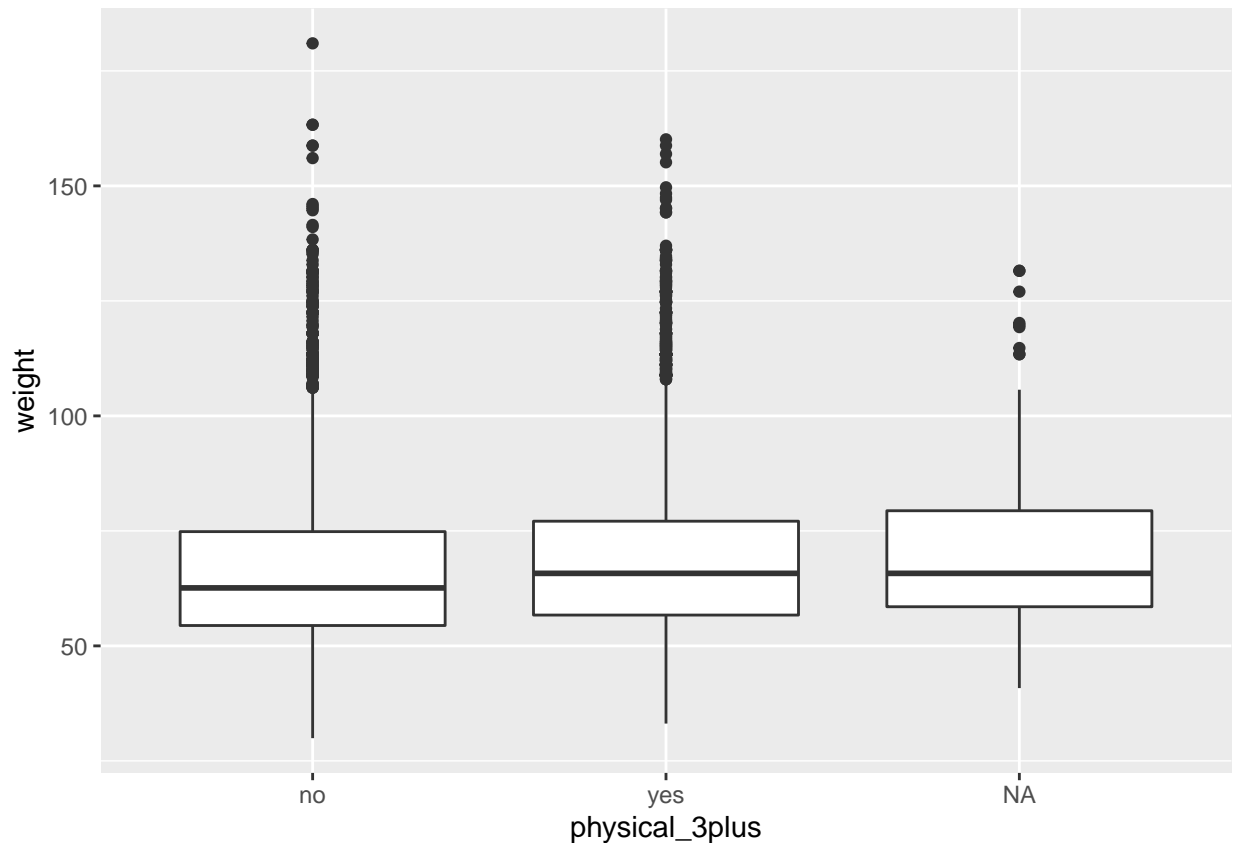
I would expected those who are physically less activate to be on the heavier side; however it looks like the mean for physical_3plus(yes) is higher than (no). The no category has more outliers.

The box plots show how the medians of the two distributions compare, but we can also compare the means of the distributions using the following to first group the data by the `physical_3plus` variable, and then calculate the mean `weight` in these groups using the `mean` function while ignoring missing values by setting the `na.rm` argument to `TRUE`.

```
yrbss %>%
  group_by(physical_3plus) %>%
  summarise(mean_weight = mean(weight, na.rm = TRUE))
```

```
## # A tibble: 3 x 2
##   physical_3plus mean_weight
##   <chr>                <dbl>
## 1 no                    66.7
## 2 yes                   68.4
## 3 <NA>                  69.9
```

There is an observed difference, but is this difference statistically significant? In order to answer this question we will conduct a hypothesis test.

Inference

Exercise 4

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()`.

```
yrbss %>%
  group_by(physical_3plus) %>%
  summarise(mean_weight = mean(weight, na.rm = TRUE), count=n())
```

```
## # A tibble: 3 x 3
##   physical_3plus mean_weight count
##   <chr>                <dbl> <int>
## 1 no                    66.7  4404
## 2 yes                   68.4  8906
## 3 <NA>                  69.9   273
```

I think we can drop the NA group since we have no data there and it's also the smallest

Exercise 5

5. Write the hypotheses for testing if the average weights are different for those who exercise at least 3 times a week and those who don't.

$H_o : \mu_{exercise} - \mu_{dont-exercise} = 0 \; H_a : \mu_{exercise} - \mu_{dont-exercise} \neq 0$

Next, we will introduce a new function, `hypothesize`, that falls into the `infer` workflow. You will use this method for conducting hypothesis tests.

But first, we need to initialize the test, which we will save as `obs_diff`.

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

Notice how you can use the functions `specify` and `calculate` again like you did for calculating confidence intervals. Here, though, the statistic you are searching for is the difference in means, with the order being `yes - no != 0`.

After you have initialized the test, you need to simulate the test on the null distribution, which we will save as `null`.

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

Here, `hypothesize` is used to set the null hypothesis as a test for independence. In one sample cases, the `null` argument can be set to "point" to test a hypothesis relative to a point estimate.

Also, note that the `type` argument within `generate` is set to `permute`, whichis the argument when generating a null distribution for a hypothesis test.

We can visualize this null distribution with the following code:

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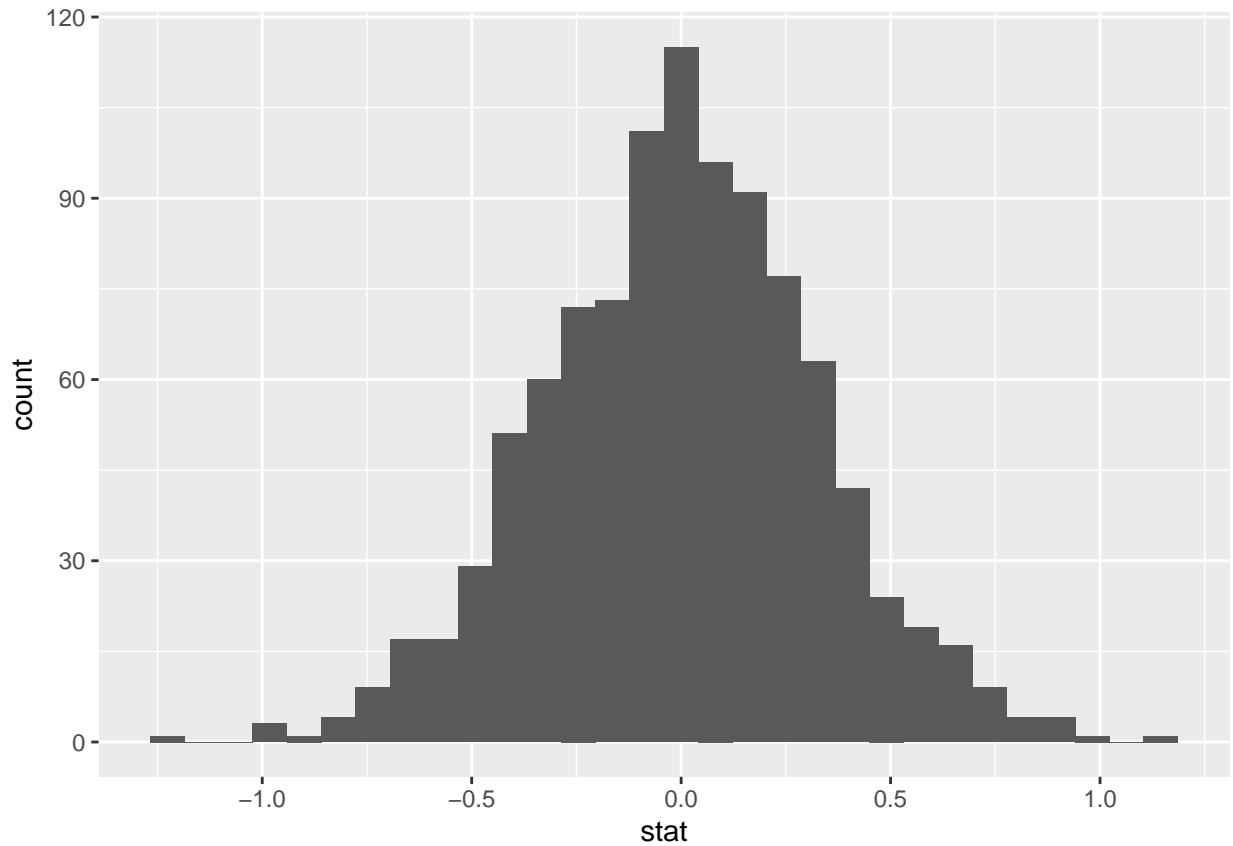

Exercise 6

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

```r
nrow(null_dist %>% filter(stat >= obs_diff))
```

```
## [1] 0
```

None of the null values have a difference of 1.77.

Now that the test is initialized and the null distribution formed, you can calculate the p-value for your hypothesis test using the function `get_p_value`.

```r
null_dist %>%
  get_p_value(obs_stat = obs_diff, direction = "two_sided")
```

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

This the standard workflow for performing hypothesis tests.

Exercise 7

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.

```
stats <- yrbss %>%
  group_by(physical_3plus) %>%
  summarise(count = n(),
            means = mean(weight, na.rm = TRUE),
            stdev = sd(weight, na.rm = TRUE)) %>% drop_na()
```

```
s1 = stats$stdev[1]
s2 = stats$stdev[2]
n1 = stats$count[1]
n2 = stats$count[2]

xbar = diff(stats$means)
se = sqrt(s1^2/n1 + s2^2/n2)
t = 1.96
c(xbar - t * se, xbar + t * se)
```

```
## [1] 1.151287 2.397881
```

The difference between the weights of those who exercise at least three times a week is between 1.15 and 2.4
* * *

More Practice

Exercise 8

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

```
heights = yrbss %>% drop_na %>% pull(height)
mn = mean(heights)
stdev = sd(heights)
n = length(heights)
t = 1.96

c(mn - t * stdev / sqrt(n), mn + t * stdev / sqrt(n))
```

```
## [1] 1.694811 1.699298
```

The average height is pretty much 1.7, since the range is (1.695 and 1.699) and anything past the second digit isn't necessary when talking about heights.

Exercise 9

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.

```
t = 1.645
c(mn - t * stdev / sqrt(n), mn + t * stdev / sqrt(n))
```

```
## [1] 1.695171 1.698937
```

As one would expect, the width get's narrower; however the realistic conclusion would still be the same 1.7

Exercise 10

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.

$H_o : \mu_{height} - \mu_{height_dont-exercise} = 0 \ H_a : \mu_{height} - \mu_{height_dont-exercise} \neq 0$

```
stats <- yrbss %>%
  group_by(physical_3plus) %>%
  summarise(count = n(),
            means = mean(height, na.rm = TRUE),
            stdev = sd(height, na.rm = TRUE)) %>% drop_na()

sy <- stats %>% filter(physical_3plus == 'yes') %>% pull(stdev)
sn <- stats %>% filter(physical_3plus == 'no') %>% pull(stdev)

my <- stats %>% filter(physical_3plus == 'yes') %>% pull(means)
mn <- stats %>% filter(physical_3plus == 'no') %>% pull(means)

ny <- stats %>% filter(physical_3plus == 'yes') %>% pull(count)
nn <- stats %>% filter(physical_3plus == 'no') %>% pull(count)

xbar = my-mn
se = sqrt(sy^2/ny + sn^2/nn)
t = 1.96

c(xbar - t * se, xbar + t * se)
```

```
## [1] 0.03390685 0.04134492
```

Exercise 11

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.

```
count = unique(yrbss$hours_tv_per_school_day)
count
```

```
## [1] "5+"           "2"           "3"           "do not watch" "<1"
## [6] "4"           "1"           NA
```

```
length(count)
```

```
## [1] 8
```

7 in total with the 8th set of values being NA

Exercise 12

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 α level, and conclude in context.

Getting more than 5 hrs of sleep is considered good else bad. I don't believe height has any relationship to the number of hours one sleeps

```
sdf <- yrbss %>%
  mutate(sleeping=ifelse(yrbss$school_night_hours_sleep %in% c('<5', '5'), 'no', 'yes')) %>%
  drop_na()

stats <- sdf %>%
          group_by(sleeping) %>%
          summarise(count = n(),
                    means = mean(height, na.rm = TRUE),
                    stdev = sd(height, na.rm = TRUE))

sy <- stats %>% filter(sleeping == 'yes') %>% pull(stdev)
sn <- stats %>% filter(sleeping == 'no') %>% pull(stdev)

my <- stats %>% filter(sleeping == 'yes') %>% pull(means)
mn <- stats %>% filter(sleeping == 'no') %>% pull(means)

ny <- stats %>% filter(sleeping == 'yes') %>% pull(count)
nn <- stats %>% filter(sleeping == 'no') %>% pull(count)

xbar = my-mn
se = sqrt(sy^2/ny + sn^2/nn)
t = 1.96

c(xbar - t * se, xbar + t * se)
```

```
## [1] 0.0000522319 0.0113863194
```

The 95% confidence interval shows a very small difference * * *