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
```

1. What are the cases in this data set? How many cases are there in our sample?

1 Insert your answer here

The word case in this context is a synonym for record, observation or row. Given this discussion starts from a data set file, it is probably better referred to as a record or a row. If we were discussing an individual person, we might call it a case.

Regardless, there are 13,583 records/rows in this dataset. Each record/case represents a teenager.

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
```

2. How many observations are we missing weights from?

2 Insert your answer here

1004

```
> 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

> yrbss %>%
  filter(is.na(weight)) %>%
  nrow()

# [1] 1004

# The summary function is nice!
```

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"))
```

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?

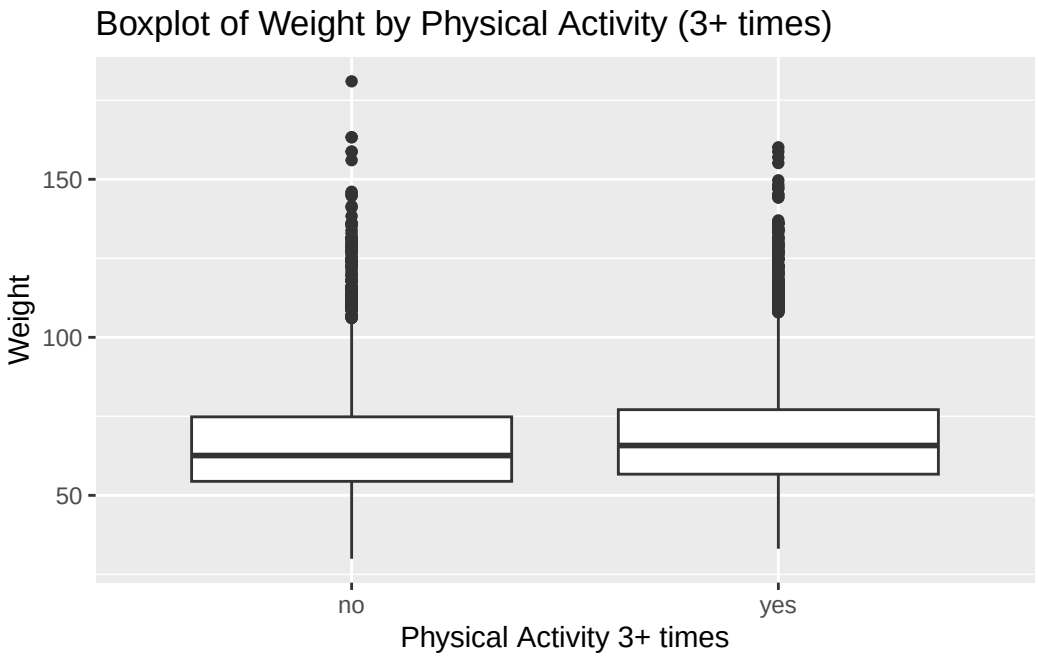
3 Insert your answer here

Logically, I would expect when comparing kids that are active versus inactive, at least with respect to weight, they would have very different IQR's. The were so similar I thought I did the chart wrong.

I went a little further and compared zero day of activity, and seven day worth of activity. It doesn't seem to make that much of a difference. I guess that's going to be what this lecture is about, finding that statistical significance threshold.

```
yrbss %>%
  filter(!is.na(weight)) %>%
  filter(!is.na(physical_3plus)) %>%
    ggplot(aes(x = physical_3plus, y = weight)) +
        geom_boxplot() +
```

```
    labs(title = "Boxplot of Weight by Physical Activity (3+ times)",
        x = "Physical Activity 3+ times",
        y = "Weight")
```



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

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()`.

Insert your answer here

- *Must be independent between groups: true. Choosing one child doest not impact the next choice.*
- *Must be independent within groups: true. one group does not influence the other.*
- *Must have a large enough sample size: true*

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

Insert your answer here

Null Hypothesis (H0): The average weight is the same for children who exercise at least three times a week (`physical_3plus = yes`) and those who don't (`physical_3plus = no`).

Alternative Hypothesis (Ha): The average weight is different for children who exercise at least three times a week (`physical_3plus = yes`) compared to those who don't (`physical_3plus = no`).

$$H_0 : \mu_{yes} = \mu_{no}$$
$$Ha : \mu_{yes} \neq \mu_{no}$$

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 %>%
  drop_na(physical_3plus) %>%
  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`.

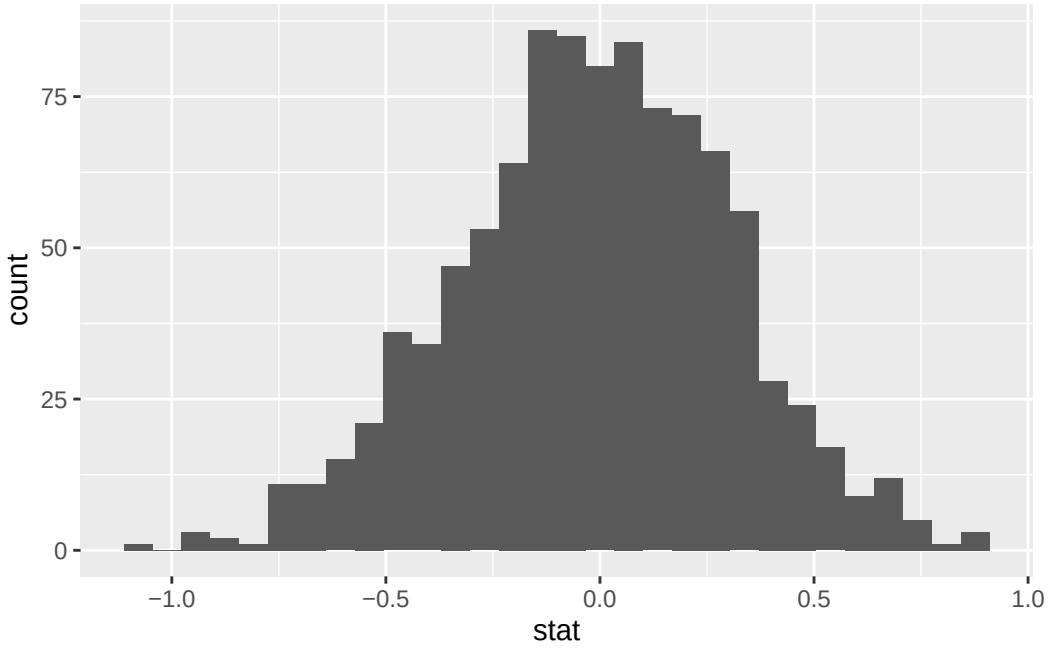
```r
null_dist <- yrbss %>%
  drop_na(physical_3plus) %>%
  specify(weight ~ physical_3plus) %>%
  hypothesize(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "diff in means", order = c("yes", "no"))
```

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()
```



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

6 Insert your answer here

I'm not sure what this question is asking, but I think I understand the problem. We are comparing the observed statistic (obs_stat) with the distribution of differences in means generated

from 1,000 permutation samples, under the null hypothesis that weight and physical3plus are independent. If the observed statistic falls within the range of the simulated distribution, it suggests that the observed data is consistent with the null hypothesis. However, if the observed statistic lies outside the range of the simulated values, it provides evidence against the null hypothesis, suggesting that weight and physical3plus are not independent of each other.

It does fall outside of the range, thus causing us to reject the null hypothesis and accept the alternative hypothesis, that weight and this physical_3plus variable are related. These two variables are not independent

Because it is totally outside of the range, we do not have to compute the p-value.

```
# > max(null_dist$stat)
# [1] 1.238921
# > min(null_dist$stat)
# [1] -0.9246731

# > summary(null_dist)
#    replicate          stat
#  Min.   :   1.0   Min.   :-0.963024
#  1st Qu.: 250.8   1st Qu.:-0.204757
#  Median : 500.5   Median : 0.011728
#  Mean   : 500.5   Mean   : 0.009297
#  3rd Qu.: 750.2   3rd Qu.: 0.235672
#  Max.   :1000.0   Max.   : 1.138529

# > obs_diff
# Response: weight (numeric)
# Explanatory: physical_3plus (factor)
# # A tibble: 1 × 1
#    stat
#   <dbl>

# 1  1.77 <- obs_diff -> this is outside
```

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`.

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

```
# > null_dist %>%
#   get_p_value(obs_stat = obs_diff, direction = "two_sided")
# # A tibble: 1 × 1
#   p_value
#     <dbl>
# 1       0
```

Insert Personal Comments Here

I didn't understand how to interpret that p-value, so here are my notes.

- *p-value is a measure of the evidence against a specified null hypothesis. Smaller values suggest weaker evidence of the null hypothesis. Larger imply that the null hypothesis might be plausible.*
- *A p-value less than 0.05 is often considered to provide "significant" evidence against the null hypothesis in many fields.*
- *A p-value less than 0.01 might be considered to provide "strong evidence" against the null hypothesis.*
- *A p-value less than 0.001 might be seen as "very strong evidence" against the null hypothesis.*

This the standard workflow for performing hypothesis tests.

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.

7 Insert your answer here

The confidence interval below will be roughly between 1.1 and 2.5. Because it does not contain 0, it aligns with the p-value work above, leading the the rejection of the null hypothesis. Cool. Same answer, but two different ways.

```
ci_diff_weights <- yrbss %>%
  drop_na(physical_3plus) %>%
  specify(weight ~ physical_3plus) %>%
  generate(reps = 1000, type = "bootstrap") %>%
  calculate(stat = "diff in means", order = c("yes", "no")) %>%
  get_ci(level = 0.95)

print(sprintf("Difference in weights confidence interval: [%s,%s]",
              round(ci_diff_weights$lower_ci, 4),
              round(ci_diff_weights$upper_ci, 4)))
```

```
[1] "Difference in weights confidence interval: [1.1587,2.4362]"
```

More Practice

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

Insert your answer here

I would expect height is independent, but it appears they are not. There is a relationship between people who work out a couple of times per week and height. It might not be causation, but there is correlation.

Null Hypothesis (H0): The average height is the same for children who exercise at least three times a week (`physical_3plus = yes`) and those who don't (`physical_3plus = no`).

Alternative Hypothesis (Ha): The average height is different for children who exercise at least three times a week (`physical_3plus = yes`) compared to those who don't (`physical_3plus = no`).

$$H_0 : \mu_{yes} = \mu_{no}$$
$$Ha : \mu_{yes} \neq \mu_{no}$$

```r
calculate_ci_height <- function(level) {

  ci_result <- yrbss %>%
    drop_na(physical_3plus) %>%
    specify(height ~ physical_3plus) %>%
    generate(reps = 1000, type = "bootstrap") %>%
    calculate(stat = "diff in means", order = c("yes", "no")) %>%
    get_ci(level = level)

  lci = round(ci_result$lower_ci, 4)
  uci = round(ci_result$upper_ci, 4)
  differ = round(uci - lci, 4)
  return(list(percentage = level,
              lower_ci = lci,
              upper_ci = uci,
              diff = differ
              ))
```

```
}

p <- c(.95)
results <- purrr::map_df(p, calculate_ci_height)
print(results)
```

```
# A tibble: 1 x 4
  percentage lower_ci upper_ci   diff
       <dbl>    <dbl>    <dbl>  <dbl>
1       0.95    0.034   0.0413 0.0073
```

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.

Insert your answer here

If I understand this lesson correctly, I am supposed to see that even though the 95% interval above is very close to zero, it does not overlap zero. And that should stay the same even if I push the confidence intervals wide open. Let's test that theory. I expect that if I open it wide enough, eventually it'll overlap zero and that level of confidence it will overlap zero and support the null hypothesis.

That doesn't happen. As you approach zero as a confidence level, the upper and lower CI lock into almost the same number. As you approach 1, the interval is at the widest. Regardless of confidence interval, these two variables never wrap zero, and therefore are not independent.

I also just discovered purrr!

```
percentages <- c(.000000001, .3, 0.50, 0.66, 0.90, 0.95, 0.99, 0.99999999999999)
results2 <- purrr::map_df(percentages, calculate_ci_height)
print(results2)
```

```
# A tibble: 8 x 4
   percentage lower_ci upper_ci   diff
        <dbl>    <dbl>    <dbl>  <dbl>
1 0.000000001   0.0376   0.0376 0
2 0.3           0.037    0.0385 0.0015
3 0.5           0.0363   0.039  0.0027
4 0.66          0.0357   0.0394 0.0037
5 0.9           0.0344   0.0409 0.0065
6 0.95          0.0335   0.0417 0.0082
7 0.99          0.0328   0.0421 0.0093
8 1.00          0.0305   0.0441 0.0136
```

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.

Insert your answer here

These two variables are not independent. The ***observed_diff_height*** *variable is well outside the null hypothesis distribution range.*

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

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

# > summary(null_dist_height)
#    replicate           stat
#  Min.   :   1.0   Min.   :-7.653e-03
#  1st Qu.: 250.8   1st Qu.:-1.271e-03
#  Median : 500.5   Median :-1.719e-05
#  Mean   : 500.5   Mean   :-2.195e-05
#  3rd Qu.: 750.2   3rd Qu.: 1.273e-03
#  Max.   :1000.0   Max.   : 6.583e-03
# > obs_diff_height
# Response: height (numeric)
# Explanatory: physical_3plus (factor)
# # A tibble: 1 × 1
#     stat
#    <dbl>
# 1 0.0376
# >
```

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.

Insert your answer here

There are seven.

- *1 <1 2168*
- *2 1 1750*
- *3 2 2705*
- *4 3 2139*
- *5 4 1048*
- *6 5+ 1595*
- *7 do not watch 1840*

```
print(table(yrbss$hours_tv_per_school_day))

#           <1            1            2            3            4           5+
#         2168         1750         2705         2139         1048         1595
# do not watch
#         1840
# >
```

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.

Insert your answer here

Let's go with let's assume weight is not related to getting less than eight hours of sleep per night.

We find our observation within a probability range that could support the null hypothesis, so we finish by looking up the p-value for this data set.

The p-value for this test is .002 which is well past an alpha of .05. We should reject the null hypothesis and assume these two variables are not independent.

```
library(stringr)
library(infer)

yrbss <- yrbss %>%
    mutate(sleep_int = as.integer(str_extract(school_night_hours_sleep, "\\d+"))) %>%
    mutate(sleep_less_than_8 = ifelse(sleep_int < 8, "yes", "no")) %>%
    drop_na(sleep_less_than_8)

obs_diff_sleep <- yrbss %>%
  specify(weight ~ sleep_less_than_8) %>%
  calculate(stat = "diff in means", order = c("yes", "no"))
```

```r
null_dist_sleep <- yrbss %>%
  specify(weight ~ sleep_less_than_8) %>%
  hypothesize(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "diff in means", order = c("yes", "no"))

summary(null_dist)
```

```
   replicate           stat          
 Min.   :   1.0   Min.   :-1.051116  
 1st Qu.: 250.8   1st Qu.:-0.217072  
 Median : 500.5   Median :-0.002187  
 Mean   : 500.5   Mean   :-0.014299  
 3rd Qu.: 750.2   3rd Qu.: 0.207942  
 Max.   :1000.0   Max.   : 0.903240  
```

```r
obs_diff_sleep
```

```
Response: weight (numeric)
Explanatory: sleep_less_than_8 (factor)
# A tibble: 1 x 1
   stat
  <dbl>
1 0.984
```

```r
# We have to compute the P value here, because obs_diff_sleep falls between the range of
# null_dist_sleep

p_val <- get_p_value(x = null_dist_sleep, obs_stat = obs_diff_sleep$stat, direction = "two
print(p_val)
```

```
# A tibble: 1 x 1
  p_value
    <dbl>
1   0.008
```