Inference_for_Numerical_Data

Zachary Palmore

2020-10-18

```
library(tidyverse)
library(openintro)
library(infer)
library(statsr)
library(psych)
```

Pre-exercise

Loading the data for this lab.

```
data(yrbss)
```

Looking at the meaning of variables.

```
?yrbss
```

Exercise 1

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

```
glimpse(yrbss)
```

```
## Rows: 13,583
## Columns: 13
## $ age                      <int> 14, 14, 15, 15, 15, 15, 15, 14, 15, 15,
15...
## $ gender                   <chr> "female", "female", "female", "female",
"f...
## $ grade                    <chr> "9", "9", "9", "9", "9", "9", "9", "9",
"9...
## $ hispanic                 <chr> "not", "not", "hispanic", "not", "not",
"n...
## $ race                     <chr> "Black or African American", "Black or
Afr...
## $ height                   <dbl> NA, NA, 1.73, 1.60, 1.50, 1.57, 1.65,
1.88...
## $ weight                   <dbl> NA, NA, 84.37, 55.79, 46.72, 67.13,
131.54...
## $ helmet_12m               <chr> "never", "never", "never", "never", "did
n...
## $ text_while_driving_30d   <chr> "0", NA, "30", "0", "did not drive", "did
...
## $ physically_active_7d     <int> 4, 2, 7, 0, 2, 1, 4, 4, 5, 0, 0, 0, 4, 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,
...
## $ school_night_hours_sleep <chr> "8", "6", "<5", "6", "9", "8", "9", "6",
"...
```

There at 13,583 rows which is also the number of cases in this sample. The 13 cases in this data set were listed as:

```
* age
* gender
* grade
* hispanic
* race
* height
* weight
* helmet_12m
* text_while_driving_30d
* physically_active_7d
* hours_tv_per_school_day
* strength_training_7d
* school_night_hours_sleep
```

Exercise 2

How many observations are we missing weights from?

Altogether, there are 9,476 missing values.

```
sum(is.na(yrbss))
```

```
## [1] 9476
```

Under the observations of *weights* we have 1004 missing values.

```
sum(is.na(yrbss$weight))
```

```
## [1] 1004
```

We could also use the summary function which confirms these missing values as the total number of NA in each column and provides some basic statistics. Here, we selected the weight column from the cases.

```
summary(yrbss[,7])
```

```
##      weight
##  Min.   : 29.94
##  1st Qu.: 56.25
##  Median : 64.41
##  Mean   : 67.91
```

```
##  3rd Qu.: 76.20
##  Max.   :180.99
##  NA's   :1004
```

Exercise 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?

Creating the variable *physical_3plus* and filling in the case values with "yes" if the individual was physically active for at least 3 days in the week or "no" if they were not.

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

A side-by-side boxplot will be made but keep in mind the missing variables listed as "NA" in the data frame are also plotted.

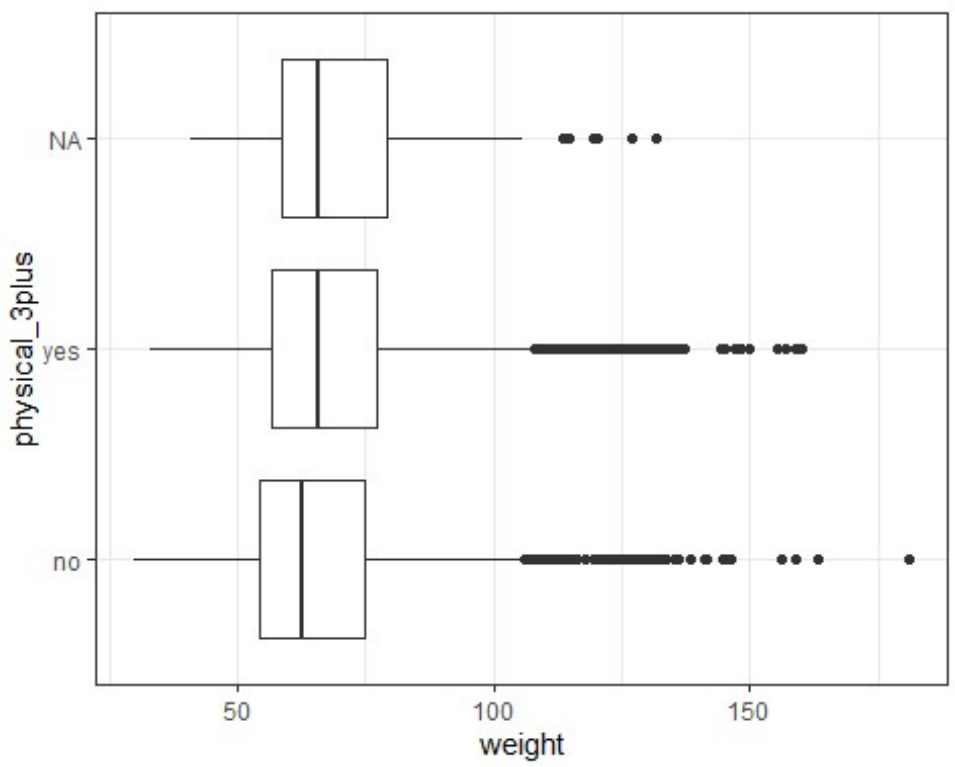
```
sum(is.na(yrbss$physical_3plus))
```

```
## [1] 273
```

There are 273 of them which is a small proportion of the number of observations overall.

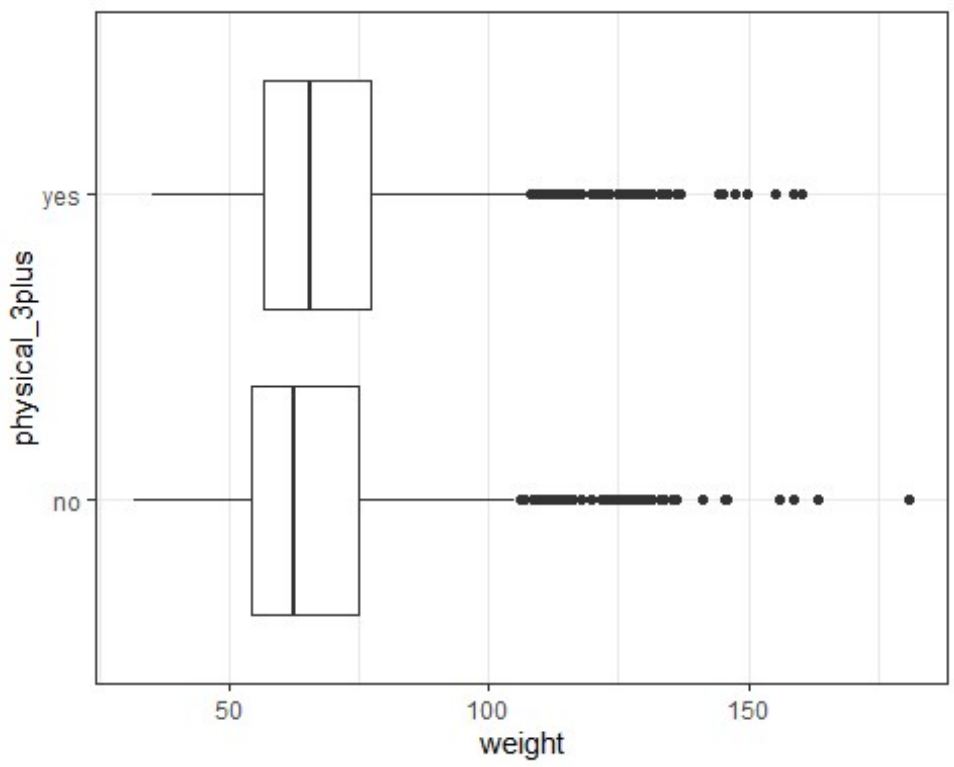
```
ggplot(yrbss, aes(x=weight, y=physical_3plus)) + geom_boxplot() + theme_bw()
```

```
## Warning: Removed 1004 rows containing non-finite values (stat_boxplot).
```



We could remove these from the data frame entirely by adding to a parameter to the chunk where the *physical_3plus* column was created then plot again.

```
yrbss2 <- yrbss %>%
  mutate(physical_3plus = ifelse(yrbss$physically_active_7d > 2, "yes",
"no")) %>%
  na.exclude()
ggplot(yrbss2, aes(x=weight, y=physical_3plus)) + geom_boxplot() + theme_bw()
```



The relationship between a student's weight and if they are physically active at least 3 times per week seems to show that those who are not physically active at least 3 times per week weigh less than those who are physically active at least 3 times per week. This is interesting as I would have expected those who were physically active at least 3 times per week to weigh less than those who were not physically active at least 3 times per week. My assumption comes from the idea that being physically active burns calories and fat, which over time, reduces a person's weight. Although, these results are contrary to that assumption.

We can check the statistics by comparing numeric values as well.

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

## `summarise()` ungrouping output (override with `.groups` argument)
```

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

With this, the relationship continues. Those who are physically active at least 3 days per week have a higher mean weight at 68.45 kg than those who are not physically active at least 3 times per week at 66.67 kg.

Exercise 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().

There are two conditions, independence and normality. Based on the information from the CDC, the data is a representative sample of many students across national, state, tribal, and local school systems and is independent. To determine normality we can look at the sample size and distribution of the boxplots. With a sample size well over 1000 (the threshold is 30) and no particularly extreme outliers, we can assume the normality condition is satisfied. The sample size of the weights is calculated by physical activity below.

```
yrbss %>%
  group_by(physical_3plus) %>%
  summarise(freq = table(weight)) %>%
  summarise(n = sum(freq))
```

```
## `summarise()` regrouping output by 'physical_3plus' (override with
`.groups` argument)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 3 x 2
##   physical_3plus     n
##   <chr>          <int>
## 1 no              4022
## 2 yes             8342
## 3 <NA>             215
```

Exercise 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.

Null hypothesis: Students who are physically active 3 or more days per week have the same average weight as those who are not physically active 3 or more days per week.

Alternative hypothesis: Students who are physically active 3 or more days per week have a different average weight when compared to those who are not physically active 3 or more days per week.

Exercise 6

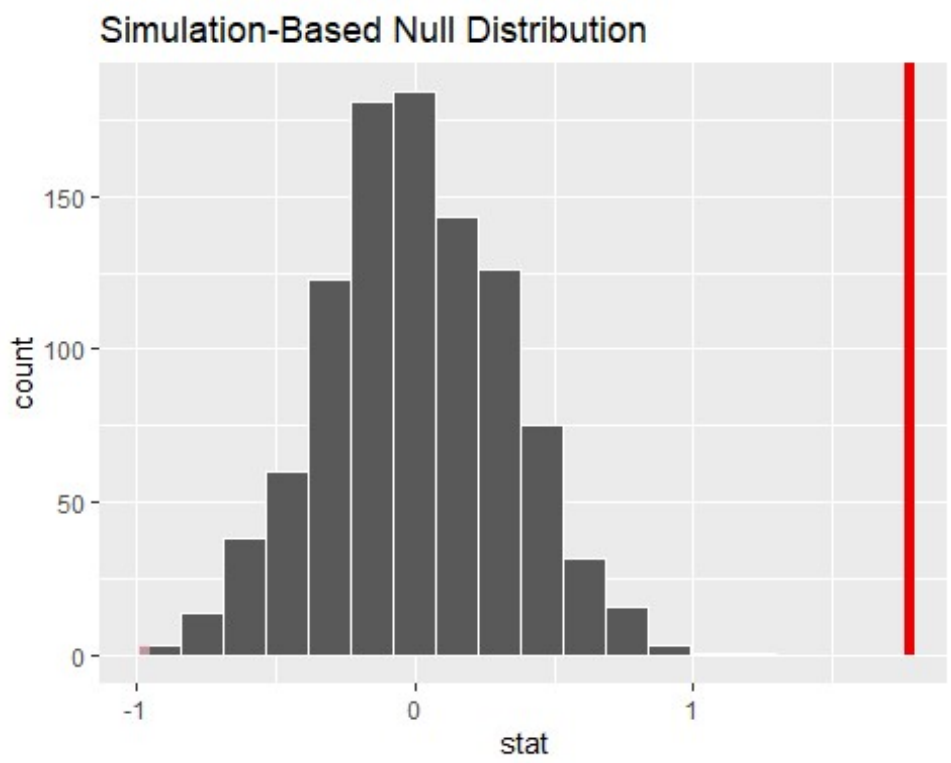
How many of these null permutations have a difference of at least obs_stat?

From lab we begin by initializing the test,

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

## Warning: Removed 1219 rows containing missing values.
```

simulating test on null and then visualizing the results.

```
set.seed(10142020)
null_dist <- yrbss %>%
  specify(weight ~ physical_3plus) %>%
  hypothesize(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "diff in means", order = c("yes", "no"))

## Warning: Removed 1219 rows containing missing values.

visualize(null_dist) +
  shade_p_value(obs_stat = obs_diff, direction = "two_sided")
```



Using the red line as a mark of the obs_stat, it appears to far from the data to have any values at or above it. To find the quantity of null permutations that have a difference of at least obs_stat we can filter the stat values in the null_dist data to show the total of those that are greater than or equal to obs_stat.

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

```
## # A tibble: 0 x 2
## # ... with 2 variables: replicate <int>, stat <dbl>
```

We could also sum the number of stat values in null_dist that are greater. Both produce the same result.

```
sum(null_dist$stat >= obs_diff$stat)
```

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

To check the p-value we can use the get_p_value function.

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

```
## 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. See
## `?get_p_value()` for more information.
```

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

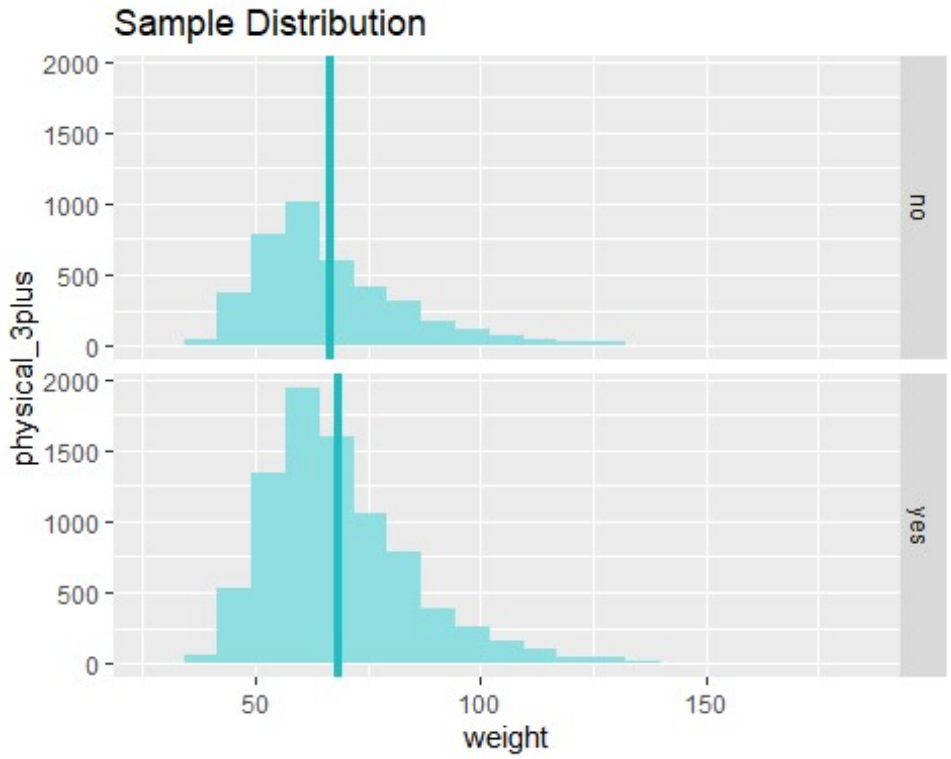
The result is a very small number, lower than a 0.001 significance level.

Exercise 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.

```
inference(data = yrbss, y = weight, x = physical_3plus,
          statistic = "mean",
          type = "ci",
          null = NULL,
          alternative = "twosided",
          method = "theoretical")
```

```
## Response variable: numerical, Explanatory variable: categorical (2 levels)
## n_no = 4022, y_bar_no = 66.6739, s_no = 17.6381
## n_yes = 8342, y_bar_yes = 68.4485, s_yes = 16.4783
## 95% CI (no - yes): (-2.4245 , -1.1246)
```



The confidence interval for the difference between the weights of those who exercise at least three times a week and those who don't can also be calculated on the null distribution with a Welch Two Sample t-test. This assumes the variances of the two are not equivalent.

```
t.test(data = yrbss, weight ~ physical_3plus)
```

```
##
##  Welch Two Sample t-test
##
## data:  weight by physical_3plus
## t = -5.353, df = 7478.8, p-value = 8.908e-08
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.424441 -1.124728
## sample estimates:
##  mean in group no mean in group yes
##          66.67389           68.44847
```

We can also calculate the intervals manually using the equation $\overline{x} \pm t_{df} * \frac{s}{\sqrt{n}}$ for comparison but to do so we need some parameters first.

Find the standard deviation of each category.

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

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 3 x 2
##   physical_3plus sd_weight
##   <chr>              <dbl>
## 1 no                  17.6
## 2 yes                 16.5
## 3 <NA>                17.6
```

The standard deviation is 17.638 for those who do are not physically active at least 3 days per week and 16.478 for those who are.

Find the mean of the weights in each category.

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

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

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

```
## 2 yes                    68.4
## 3 <NA>                   69.9
```

This agrees with the results from earlier that the mean weight is 66.674 for those who do are not physically active at least 3 days per week and 68.448 for those who are.

Lastly, the sample size of the whole was calculated as 13,583 with missing values. We want the sample sizes of each category.

```
yrbss %>%
  group_by(physical_3plus) %>%
  summarise(freq = table(weight)) %>%
  summarise(n = sum(freq))
```

```
## `summarise()` regrouping output by 'physical_3plus' (override with
`.groups` argument)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 3 x 2
##   physical_3plus     n
##   <chr>          <int>
## 1 no              4022
## 2 yes             8342
## 3 <NA>             215
```

For example, the sample size of those who were physically active for at least 3 days per week is 8,342 while the sample size of those who were not physically active for at least 3 days per week is 4,022. Missing values were not included since they do not convey any meaning here.

We can now calculate the confidence interval of each category using a 95% confidence level.

```
x_not3plus <- 66.67389
n_not3plus <- 4022
s_not3plus <- 17.63805
x_3plus <- 68.44847
n_3plus <- 8342
s_3plus <- 16.47832
# At 95% confidence level where n is so large it is ~= z* of
# normal distribution
t = 1.96

# Not physically active 3 plus days per week
upper_ci_not <- x_not3plus + t*(s_not3plus/sqrt(n_not3plus))
lower_ci_not <- x_not3plus - t*(s_not3plus/sqrt(n_not3plus))

# physically active 3 plus days per week
upper_ci <- x_3plus + t*(s_3plus/sqrt(n_3plus))
lower_ci <- x_3plus - t*(s_3plus/sqrt(n_3plus))
```

```
upper_ci_not

## [1] 67.219

lower_ci_not

## [1] 66.12878

upper_ci

## [1] 68.80209

lower_ci

## [1] 68.09485
```

We can be 95% confident that those students who exercise at least three times a week have an average weight between 68.095 kg and 68.802 kg. We can also be confident that those students who do not exercise at least three times a week have an average weight between 66.129 kg and 67.219 kg.

Exercise 8

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

```
# Verifying sum of frequency / counts = n without NAs
table_height <- as.data.frame(table(yrbss$height))
freq_height <- sum(table_height$Freq)

x_height <- mean(yrbss$height, na.rm = TRUE)
sd_height <- sd(yrbss$height, na.rm = TRUE)
n_height <- yrbss %>%
  summarise(freq = table(height)) %>%
  summarise(n = sum(freq, na.rm = TRUE))

upper_ci_height <- x_height + t*(sd_height/sqrt(n_height))
lower_ci_height <- x_height - t*(sd_height/sqrt(n_height))
upper_ci_height

##          n
## 1 1.693071

lower_ci_height

##          n
## 1 1.689411
```

We can be 95% confident that the average height of the students in this population is between 1.689m and 1.693m.

Exercise 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.

```
# At 90% confidence level where n is so large it is ~= z-score of normal
distribution
t_90 <- 1.645
upper_ci_height_90 <- x_height + t_90*(sd_height/sqrt(n_height))
lower_ci_height_90 <- x_height - t_90*(sd_height/sqrt(n_height))
upper_ci_height_90

##          n
## 1 1.692777

lower_ci_height_90

##          n
## 1 1.689705
```

The new confidence interval is 1.689705 to 1.692777. Our intervals at a 95% confidence level were 1.689411 and 1.693071. We can find the difference in these two confidence intervals and compare 90% to 95% confidence.

```
rng_hgt_95 <- (upper_ci_height - lower_ci_height)
rng_hgt_90 <- (upper_ci_height_90 - lower_ci_height_90)
rng_hgt_95

##             n
## 1 0.003659302

rng_hgt_90

##           n
## 1 0.0030712
```

As expected, the 95% confidence interval has a slightly larger range than the confidence interval 90%. This larger range is necessary to be more certain about the population parameter.

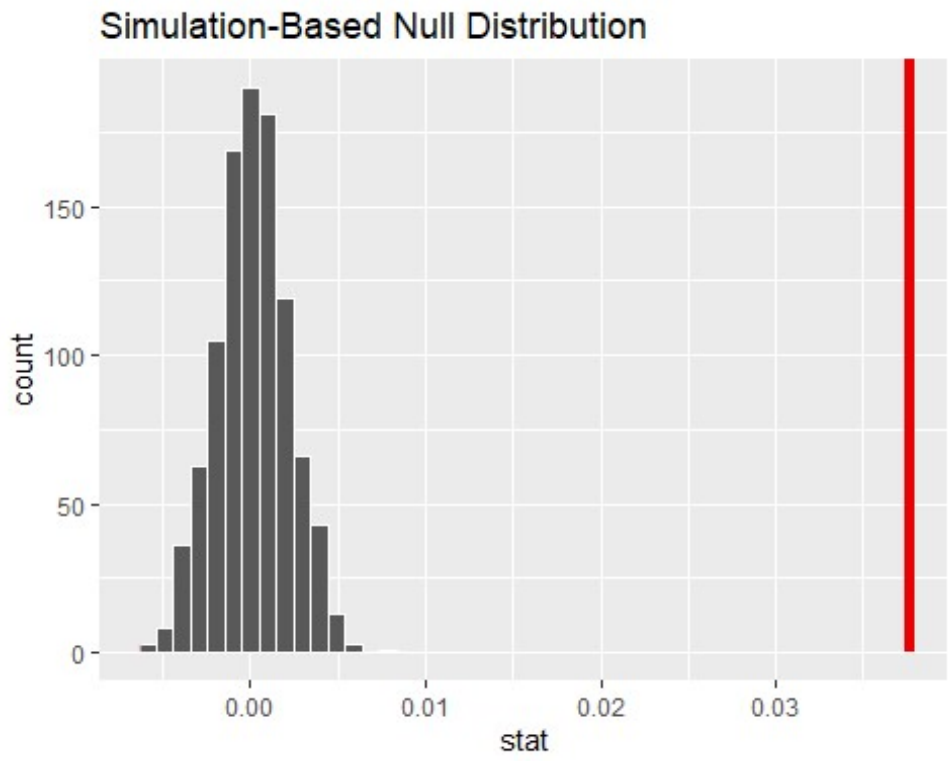
Exercise 10

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.

Null hypothesis: There is no difference in the average height of those who are physically active at least 3 days per week and those who are not.

Alternative hypothesis: There is a difference in the average height of those who are physically active at least 3 days per week and those who are not.

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

## Warning: Removed 1219 rows containing missing values.

set.seed(10152020)
null_dist_hgt <- yrbss %>%
  specify(height ~ physical_3plus) %>%
  hypothesize(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "diff in means", order = c("yes", "no"))

## Warning: Removed 1219 rows containing missing values.

visualize(null_dist_hgt) +
  shade_p_value(obs_stat = obs_diff_hgt, direction = "two_sided")
```



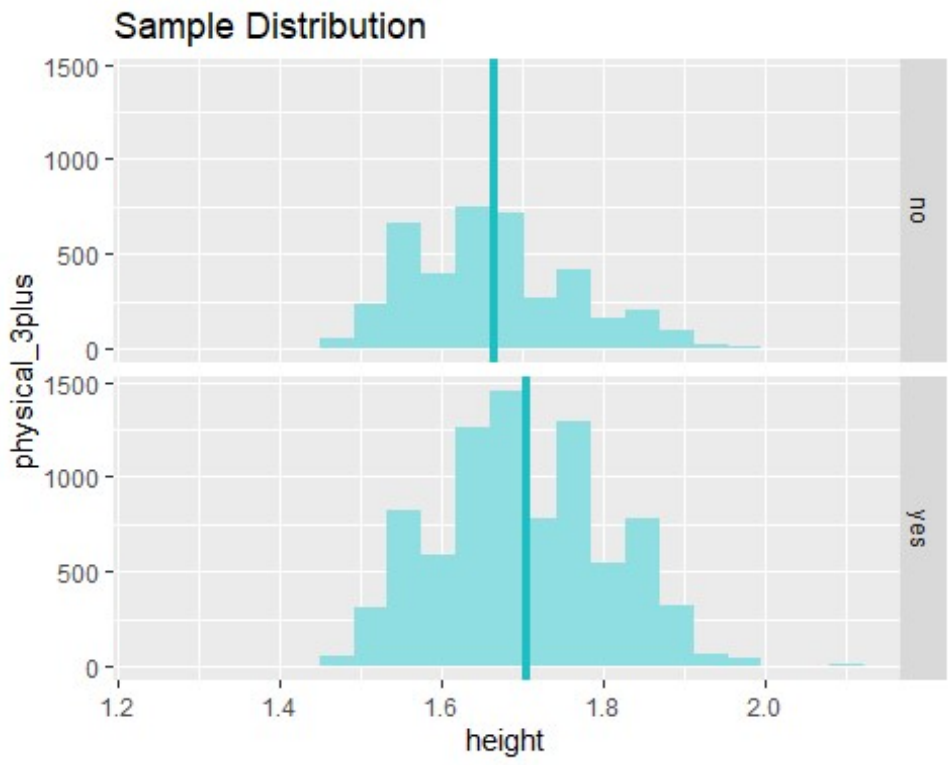

```r
null_dist_hgt %>%
  get_p_value(obs_stat = obs_diff_hgt, direction = "two_sided")

## 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. See
## `?get_p_value()` for more information.

## # A tibble: 1 x 1
##   p_value
```

```
##     <dbl>
## 1       0
```

The p-value is very small, smaller than 0.05. At this level, we should reject the null hypothesis.

```
inference(data = yrbss, y = height, x = physical_3plus,
          statistic = "mean",
          type = "ci",
          null = NULL,
          alternative = "twosided",
          method = "theoretical")
```

```
## Response variable: numerical, Explanatory variable: categorical (2 levels)
## n_no = 4022, y_bar_no = 1.6656, s_no = 0.1029
## n_yes = 8342, y_bar_yes = 1.7032, s_yes = 0.1033
## 95% CI (no - yes): (-0.0415 , -0.0337)
```




```
x_nhgt <- 1.6665
n_nhgt <- 4022
s_nhgt <- 0.1029
x_yhgt <- 1.7032
n_yhgt <- 8342
s_yhgt <- 0.1033
# At 95% confidence level where n is so large it is ~= z* of
# normal distribution
t = 1.96
```

```r
# Not physically active 3 plus days per week
upper_ci_nhgt <- x_nhgt + t*(s_nhgt/sqrt(n_nhgt))
lower_ci_nhgt <- x_nhgt - t*(s_nhgt/sqrt(n_nhgt))

# physically active 3 plus days per week
upper_ci_yhgt <- x_yhgt + t*(s_yhgt/sqrt(n_yhgt))
lower_ci_yhgt <- x_yhgt - t*(s_yhgt/sqrt(n_yhgt))

upper_ci_nhgt
```

```
## [1] 1.66968
```

```r
lower_ci_nhgt
```

```
## [1] 1.66332
```

```r
upper_ci_yhgt
```

```
## [1] 1.705417
```

```r
lower_ci_yhgt
```

```
## [1] 1.700983
```

We can be 95% confident that the average height of students who are physically active at least 3 days per week is between 1.705 and 1.701 while the average height of students who are not physically active at least 3 days per week is between 1.670 and 1.663.

Exercise 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.

If the question is referring to the number of combinations (or options) of the variable hours_tv_per_school_day, then the answer would depend on whether or not the order of the options matter and if we should repeat or replace the variables once they are used. We can easily calculate the number of different variables including hours_tv_per_school_day at 14 and see each of their labels below.

```r
ncol(yrbss)
```

```
## [1] 14
```

```r
colnames(yrbss)
```

```
##  [1] "age"                     "gender"
##  [3] "grade"                   "hispanic"
##  [5] "race"                    "height"
##  [7] "weight"                  "helmet_12m"
##  [9] "text_while_driving_30d"  "physically_active_7d"
```

```
## [11] "hours_tv_per_school_day"  "strength_training_7d"
## [13] "school_night_hours_sleep" "physical_3plus"
```

There are also only 14 different options if the variable hours_tv_per_school_day can only be paired with exactly one other variable from the dataset only once and including itself.

If the question is referring to the number of options within the variable hours_tv_per_school_day then we can calculate the quantity of unique values for this particular variable in the dataset.

```
unique(yrbss$hours_tv_per_school_day)

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

length(unique(yrbss$hours_tv_per_school_day))

## [1] 8
```

We can see the options here are “do not watch”, “<1”, 1, 2, 3, 4, 5+, and NA. If we are to include missing values as an option within this dataset then we are left with 8 options of the variable hours_tv_per_school_day. If we were to remove the missing values as an option in the variable then the answer is 7 options without NA.

Exercise 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.

Setup:

Null hypothesis: The average weight of students has no affect on the average number of hours of sleep students receive on school nights.

Alternative hypothesis: The average weight of students has an affect on the average number of hours of sleep students receive on school nights.

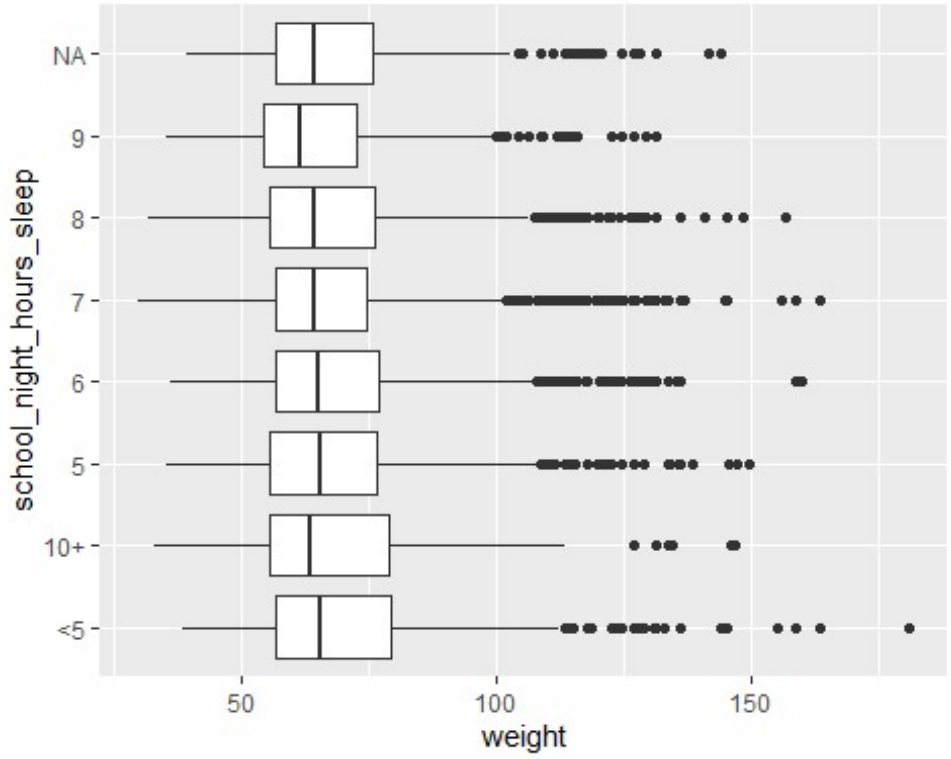
An analysis of variance(ANOVA) would work best for this to determine if the means of all the groups are different from the null. We know the data are independent already and are approximately normal. Since we have more than two means to compare, this is a good place to start. To be more specific, there are 8 groups to find means of.

```
unique(yrbss$school_night_hours_sleep)

## [1] "8"   "6"   "<5"  "9"   "10+" "7"   "5"   NA

length(unique(yrbss$school_night_hours_sleep))

## [1] 8
```

We can call these eight options in the school_night_hours_sleep variable the sleeping groups. For each sleeping group we will find the mean and other statistics. First, let's look at a boxplot for outliers.

```
ggplot(yrbss, aes(x = weight, y = school_night_hours_sleep)) + geom_boxplot()
```

```
## Warning: Removed 1004 rows containing non-finite values (stat_boxplot).
```



Based on the boxplot, all of the medians appear similar with some subtle variations. Each sleeping group also has similar IQRs but we should take a closer look.

```
desc <- describeBy(yrbss$weight, yrbss$school_night_hours_sleep,
mat=TRUE)[,c(2,4,5,6)]
desc$Var <- desc$sd^2
print(desc, row.names=FALSE)
```

```
##  group1    n     mean       sd      Var
##      <5  859 70.29700 19.47970 379.4586
##     10+  255 69.29251 19.92961 397.1895
##       5 1378 68.41806 17.47753 305.4639
##       6 2496 68.33318 17.12553 293.2838
##       7 3283 67.43457 16.12185 259.9140
##       8 2505 67.45745 16.52393 273.0401
##       9  705 65.55898 15.87743 252.0929
```

The sample sizes of each sleeping group are greater than 30 and based on the boxplots and means calculated for each sleeping group, there are very few particularly extreme values.

There is clear variation in sleeping groups, however, they are close enough to perform an anova.

```
aov.out <- aov(data=yrbss, weight ~ school_night_hours_sleep )
summary(aov.out)
```

```
##                              Df  Sum Sq Mean Sq F value  Pr(>F)
## school_night_hours_sleep      6   11333    1889   6.581 5.9e-07 ***
## Residuals                 11474 3293032     287
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 2102 observations deleted due to missingness
```

This produces a p-value that is significant to the 0.001 level. This provides strong evidence that we should reject the null hypothesis in favor of the alternative. In other words, the average weight of students appears to have an affect on the average number of hours of sleep students receive on school nights. …