Lab 08: T-Tests

Nimita Gaggar (3039677409)

Today's date

Run this chunk of code to load the autograder package!

Instructions

- Due date: Tuesday, August 1st, at 10:00pm PST with a 2 hour grace period.
- Late penalty: 50% late penalty if submitted within 24 hours of due date, no marks for assignments submitted thereafter.
- This assignment is graded on **correct completion**, all or nothing. You must pass all public tests and submit the assignment for credit.
- Submission process: Follow the submission instructions on the final page. Make sure you do not remove any `\newpage` tags or rename this file, as this will break the submission.

Introduction

Part 1 of this lab focuses on two datasets sampled from data collected early in the HIV epidemic. Part 2 focuses on conducting a t-test, and compares results from a paired test vs. an independent test.

Section I: HIV data

- We have two data sets, both sampled from data collected relatively early in the HIV epidemic.
- Deeks, et al. (1999) performed a longitudinal study of HIV-infected adults undergoing Highly Active Anti-Retroviral Therapy (HAART) at San Francisco General Hospital (SFGH).
- Patients were included in this analysis if they received at least 16 weeks of continuous therapy with an anti-retroviral regimen.
- For both data, the outcome is a measure of severity of the disease, a count of an immune cell type called CD4.

More on data

- The first dataset, `deeks_ex1.csv`, has one response measurement per subject, which is their average CD4 count.

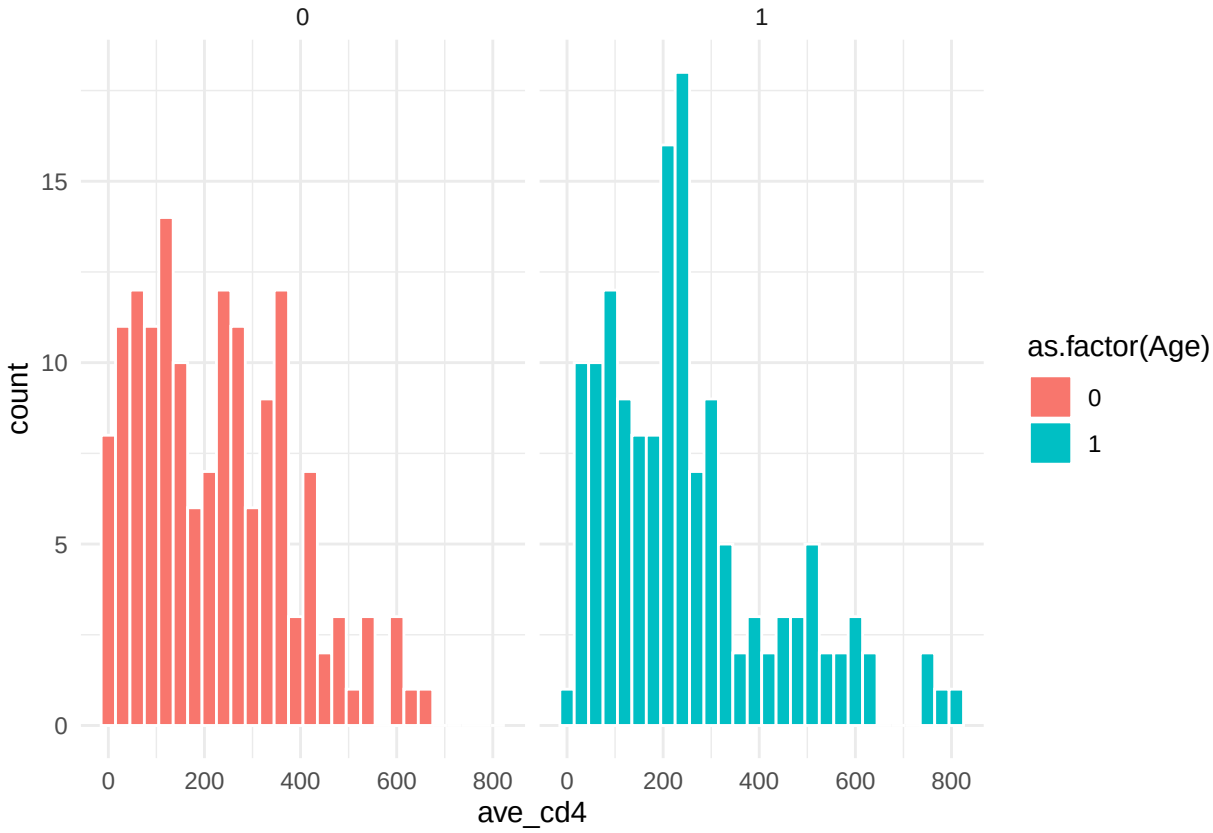
- The data set also contains a single binary covariate `age` (1 if $\geq 40 years$, 0 if ≤ 40).
- The second dataset, `deeks_ex2.csv`, has two measurements per individual, one at each level of the binary viral load (`vl` = 1 if ≥ 2000, `vl` = 0 if ≤ 2000).

Age versus CD4 count

1. After importing `deeks_ex1.csv` into R, visually compare the distribution of CD4 counts between individuals where `age = 1` vs. `age = 0`. Note that these datasets are located in the data folder.

```
library(ggplot2)
library(readr)
library(dplyr)
library(tidyr)
library(tidyverse)
library(testthat)
```

```
deeks1 <- read.csv("../lab08/data/deeks_ex1.csv")
p1 <- ggplot(deeks1, aes(x = ave_cd4)) +
      geom_histogram( col= "white", aes(fill = as.factor(Age)),
                  binwidth = 30) + theme_minimal()+ facet_wrap(~as.factor(Age))
p1
```



2. [1 point] Which testing procedure can be used to test the difference between the mean CD4 counts across individuals with `age = 1` vs. `age = 0`? Perform the test using an R function. Note the estimated mean difference and the provided 95% confidence interval. Assign your p-value rounded to 2 decimal places to the object `pvalue_deeks`.

(If you have extra time, confirm that you can calculate the test statistic using dplyr functions only).

```
t.test(deeks1$ave_cd4 ~ deeks1$Age)
```

```
##
##  Welch Two Sample t-test
##
## data:  deeks1$ave_cd4 by deeks1$Age
## t = -1.2563, df = 286.52, p-value = 0.21
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -62.21788  13.73708
## sample estimates:
## mean in group 0 mean in group 1
##        225.9020        250.1424
```

```
pvalue_deeks <- 0.21
```

The p value is 0.21

```
. = ottr::check("tests/p2.R")
```

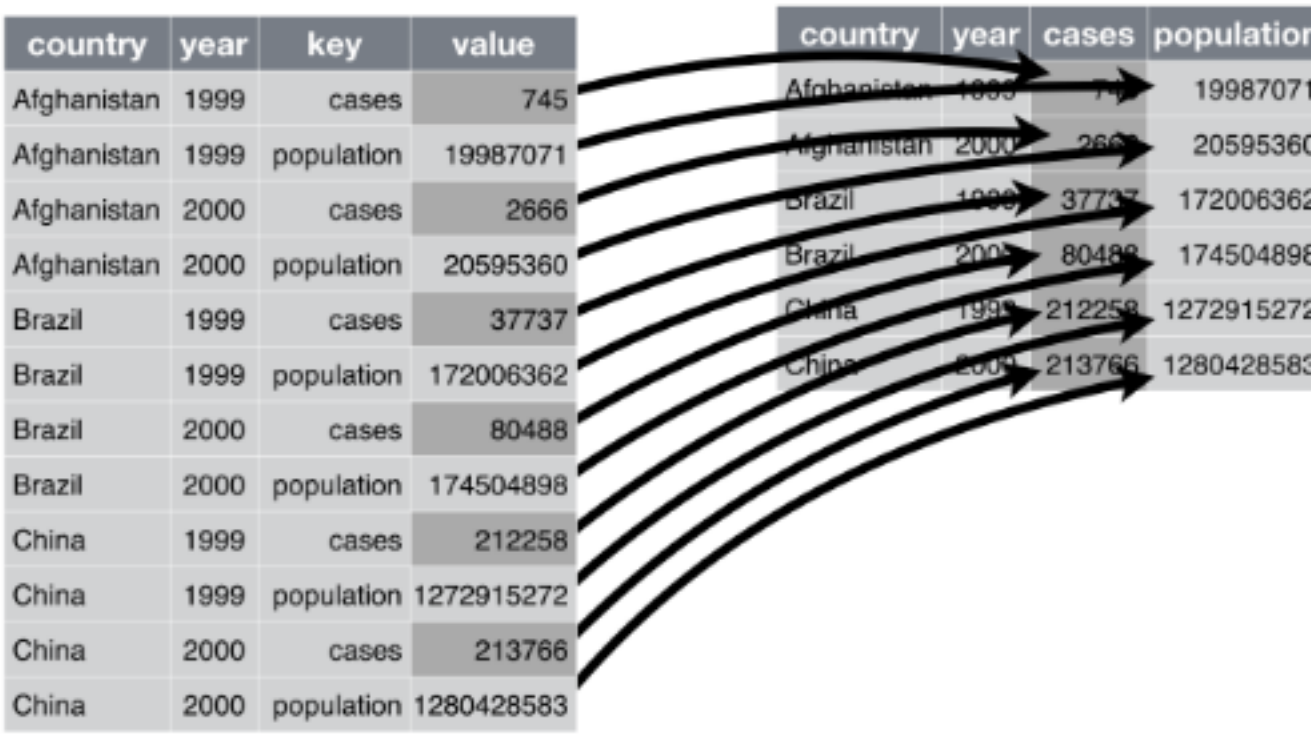
```
##
## All tests passed!
```

CD4 count and viral load

3. [1 point] Read in the `deeks_ex2.csv` dataset and assign it to an object called `deeks2`. The data is in "long" format (with two rows per individual, one for each level of `medvl`). Use the `pivot_wider()` function from `tidyr` to convert the data into "wide" format so the CD4 measures at `medvl = 0` and `medvl = 1` are contained in the same row for each individual and assign this new dataset to an object called `deeks_wide`. Try using the help window to figure out how to use this function!

Here is an illustration of how spread works:

```
knitr::include_graphics("src/lab08-spread-function.png")
```



country	year	key	value
Afghanistan	1999	cases	745
Afghanistan	1999	population	19987071
Afghanistan	2000	cases	2666
Afghanistan	2000	population	20595360
Brazil	1999	cases	37737
Brazil	1999	population	172006362
Brazil	2000	cases	80488
Brazil	2000	population	174504898
China	1999	cases	212258
China	1999	population	1272915272
China	2000	cases	213766
China	2000	population	1280428583

table2

```
deeks2 <-read.csv("../lab08/data/deeks_ex2.csv")
deeks2
```

```
##       id cd4 medvl
## 1    16 449     0
## 2    16 226     1
## 3    18 294     0
## 4    18 138     1
## 5    21 132     1
## 6    21 132     0
## 7    26 324     1
## 8    26 500     0
## 9    30 216     1
## 10   30 254     0
## 11   33 219     1
## 12   33 318     0
## 13   36 318     0
## 14   36 251     1
## 15   41  13     0
## 16   41   9     1
## 17   49 216     1
## 18   49 308     0
## 19   50 740     0
```

```
## 20   50 564     1
## 21   52  61     0
## 22   52  74     1
## 23   68 151     0
## 24   68   9     1
## 25   78 471     0
## 26   78 485     1
## 27   91 172     1
## 28   91  97     0
## 29   97 239     0
## 30   97 290     1
## 31  111 993     0
## 32  111 467     1
## 33  118  50     0
## 34  118 218     1
## 35  124 190     1
## 36  124 286     0
## 37  138  28     1
## 38  138  87     0
## 39  141 310     0
## 40  141 170     1
## 41  155 448     1
## 42  155 320     0
## 43  156 250     1
## 44  156 243     0
## 45  163 554     0
## 46  163 353     1
## 47  165 512     0
## 48  165 584     1
## 49  168  14     1
## 50  168 109     0
## 51  178 321     0
## 52  178 211     1
## 53  183 401     0
## 54  183 397     1
## 55  191 112     0
## 56  191 139     1
## 57  194 141     0
## 58  194   7     1
## 59  195 132     1
## 60  195 118     0
## 61  200 153     1
## 62  200 181     0
## 63  207 563     0
## 64  207 515     1
## 65  210 242     0
## 66  210 187     1
## 67  218 773     1
## 68  218 855     0
## 69  223 400     1
## 70  223 354     0
## 71  233 381     0
## 72  233 187     1
## 73  242 443     1
```

```
## 74  242 286     0
## 75  244 259     1
## 76  244 471     0
## 77  257 690     0
## 78  257 520     1
## 79  264 409     0
## 80  264 299     1
## 81  272 270     1
## 82  272 348     0
## 83  275 309     0
## 84  275 442     1
## 85  280 513     1
## 86  280 600     0
## 87  285 410     0
## 88  285 185     1
## 89  302 271     1
## 90  302 206     0
## 91  308 297     1
## 92  308  95     0
## 93  310 284     0
## 94  310 258     1
## 95  313 312     0
## 96  313 316     1
## 97  322 339     0
## 98  322 467     1
## 99  325 465     0
## 100 325 234     1
## 101 333 144     1
## 102 333 163     0
## 103 343 418     0
## 104 343  42     1
## 105 359  56     0
## 106 359 363     1
## 107 382 219     1
## 108 382  86     0
## 109 386 351     0
## 110 386 243     1
## 111 388 140     0
## 112 388 137     1
## 113 392 136     1
## 114 392 158     0
## 115 398  74     1
## 116 398 305     0
## 117 406 144     1
## 118 406 190     0
## 119 411 401     1
## 120 411 409     0
## 121 415  88     1
## 122 415 111     0
## 123 419 378     0
## 124 419 382     1
## 125 434 209     0
## 126 434 292     1
## 127 444 693     0
```

```
## 128 444 459     1
## 129 445 202     0
## 130 445 219     1
## 131 449 103     0
## 132 449  59     1
## 133 454  33     1
## 134 454  18     0
## 135 474 375     1
## 136 474 470     0
## 137 475 333     1
## 138 475 207     0
## 139 478 185     1
## 140 478 168     0
## 141 481 379     1
## 142 481 520     0
```

```
deeks_wide <-pivot_wider(deeks2,id_cols =id,names_from = medv1,values_from=cd4)
deeks_wide
```

```
## # A tibble: 71 x 3
##       id   `0`   `1`
##    <int> <int> <int>
##  1    16   449   226
##  2    18   294   138
##  3    21   132   132
##  4    26   500   324
##  5    30   254   216
##  6    33   318   219
##  7    36   318   251
##  8    41    13     9
##  9    49   308   216
## 10    50   740   564
## # i 61 more rows
```

```
# YOUR CODE HERE
```

```
. = ottr::check("tests/p3.R")
```

```
##
## All tests passed!
```

4. [1 point] Rename the `medv1 = 0` and `medv1 = 1` columns as "high" and "low", respectively. Then calculate the difference in CD4 counts (high - low) for each individual and save this value in a new column `diff`.

```
deeks_wide <- deeks_wide%>% rename(high = "0", low = "1" )
deeks_wide <- deeks_wide %>% mutate (diff = high - low)

deeks_wide
```

```
## # A tibble: 71 x 4
```

```
##       id  high   low  diff
##    <int> <int> <int> <int>
##  1    16   449   226   223
##  2    18   294   138   156
##  3    21   132   132     0
##  4    26   500   324   176
##  5    30   254   216    38
##  6    33   318   219    99
##  7    36   318   251    67
##  8    41    13     9     4
##  9    49   308   216    92
## 10    50   740   564   176
## # i 61 more rows
```

```
# YOUR CODE HERE
```

```
. = ottr::check("tests/p4.R")
```

```
##
## All tests passed!
```

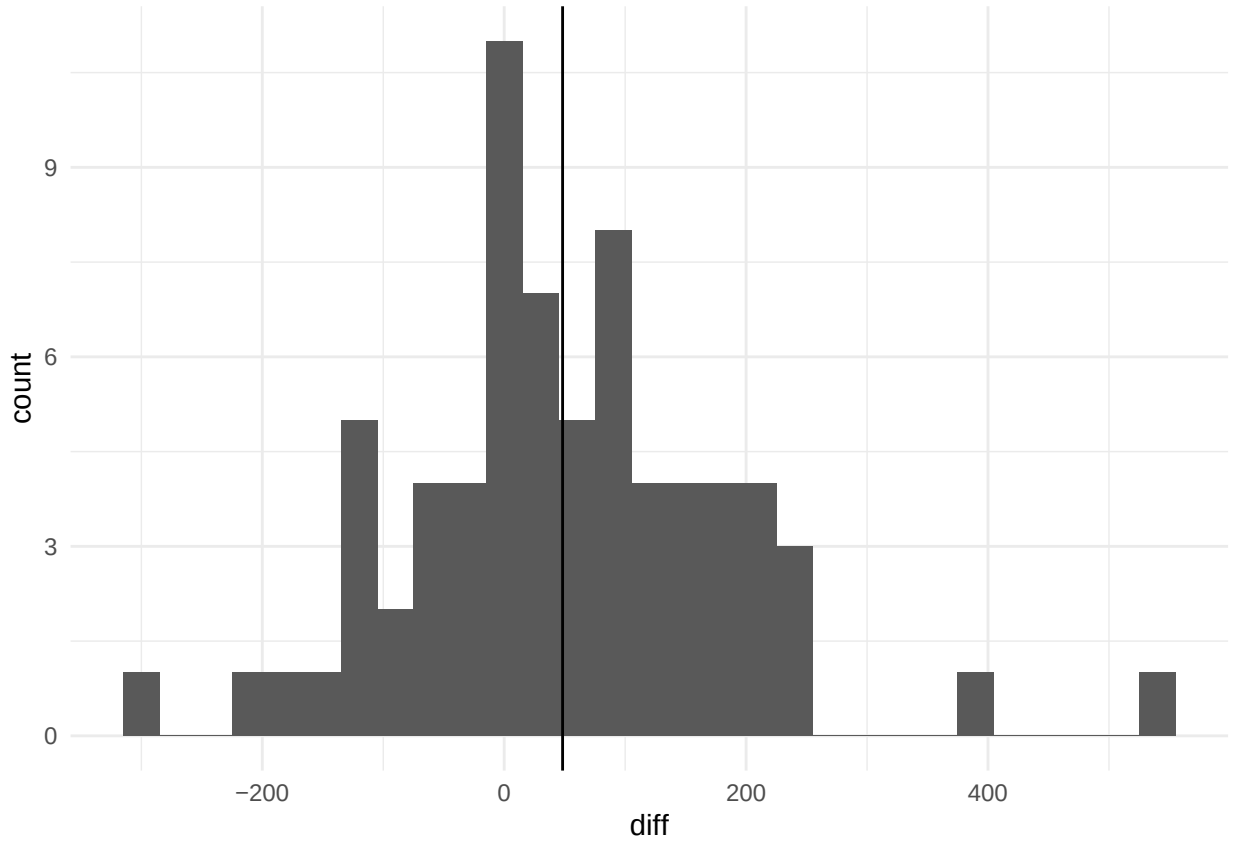
5. [1 point] Visualize the distribution of the *individual differences* in CD4 counts in `deeks_wide` and add an x-intercept line at the mean value of the `diff` variable.

```
mean(deeks_wide$diff)
```

```
## [1] 48.28169
```

```
p5 <- ggplot(deeks_wide, aes(x = diff)) + geom_histogram(binwidth = 30)+
      geom_vline(aes(xintercept = (mean(deeks_wide$diff))))+ theme_minimal()
p5
```

```
## Warning: Use of ‘deeks_wide$diff‘ is discouraged.
## i Use ‘diff‘ instead.
```




```
# YOUR CODE HERE
```

```
. = ottr::check("tests/p5.R")
```

```
##
## All tests passed!
```

6. [1 point] Which of the testing procedures that we've learned so far can be used to test the difference between each individual's CD4 count during a time of high vs. low viral load? Perform the test using an R function. Note the estimated mean difference and the provided 95% confidence interval. Report your p-value rounded to 4 decimal places.

```
p6 <- t.test(deeks_wide$high, deeks_wide$low, paired = TRUE)
p6 <- 0.0033


# YOUR CODE HERE
```

The p value is 0.033

```
. = ottr::check("tests/p6.R")
```

```
##
## All tests passed!
```

Section II: Coin Flip Game.

If you are doing this lab before your lab section, please answer the questions using the sample Googlesheet.

Go to this website

The game: See how many dots you can hit in the grid within 30 seconds. We will each try this once with our dominant hand and once with our non-dominant hand (**where your dominant hand is the one you prefer to operate a computer mouse or track pad with**).

Instructions:

Flip a coin to see which hand to play the game with first: - Heads = dominant hand first - Tails = non-dominant hand first

Don't have a coin near you? That's okay! How do you simulate flipping a coin in R?

7. Play the game and record the number of dots you hit: 35 with the dominant

8. Re-do the game, this time with the other hand. Record the results below. 28 with the non-dominant

9. Now we need to record this data by appending it to the dataset `our_sheet` in R. Fill in the code with the number of dot hits by your dominant and non dominant hand, as well as whether your dominant hand went first. (This dataset already contains previously simulated data from former students.)

```
our_sheet <- read_csv("data/our_sheet.csv")
```

```
## Rows: 23 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): Student_name
## dbl (2): Dominant_num_dots_hit, Non_dominant_num_dots_hit
## lgl (1): Dominant_hand_first
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
dom_num_dots_hit <- 35
non_dom_num_dots_hit <- 28
dom_hand_first <- TRUE
name <- "Nimita"

our_sheet <- rbind(our_sheet, list(dom_num_dots_hit, non_dom_num_dots_hit,
                                   dom_hand_first, name))
our_sheet
```

```
## # A tibble: 24 x 4
##    Dominant_num_dots_hit Non_dominant_num_dot~1 Dominant_hand_first Student_name
##                    <dbl>                  <dbl> <lgl>               <chr>
##  1                    36                     24 TRUE                Ellen
##  2                    39                     33 TRUE                Jennifer
##  3                    37                     32 TRUE                Ivan
##  4                    25                     20 TRUE                Annette
##  5                    30                     27 TRUE                Paula
##  6                    22                     28 TRUE                Annette
```

```
##  7                    22                    19 TRUE                Dee
##  8                    29                    24 TRUE                Alex
##  9                    32                    27 FALSE               Sherry
## 10                    27                    18 FALSE               Max
## # i 14 more rows
## # i abbreviated name: 1: Non_dominant_num_dots_hit
```

10. These data are very naturally paired. Add a variable to `our_sheet` called `diff` that is the difference between the number of dots hit with the dominant and non-dominant hands and assign this new dataset to an object called `our_sheet_diff`. What two assumptions do we need to make to use a paired t-test? For each assumption, write why you think the assumption is met (or not met). Create a plot and assign it to an object called `p10` to investigate one of the assumptions and comment on whether the plot supports the assumption.

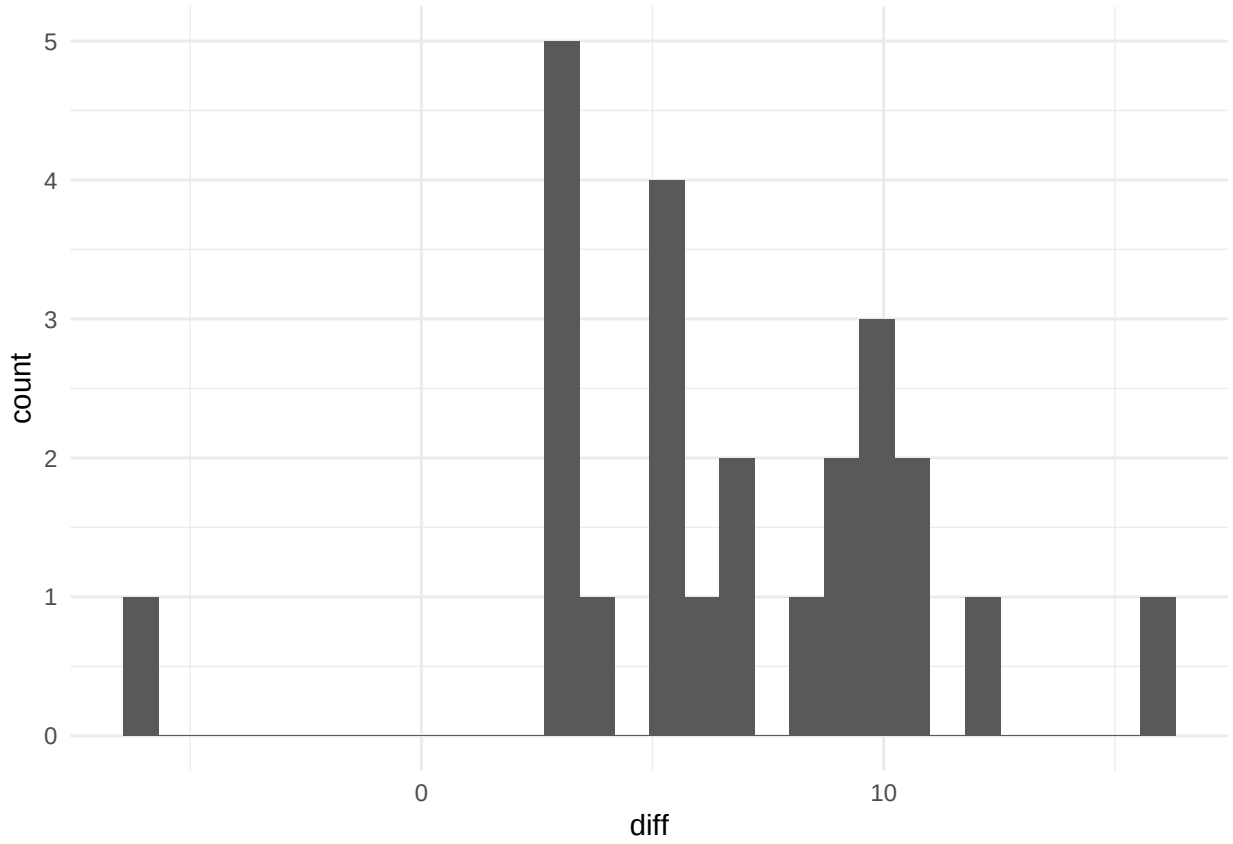
The observation folllows are independent and follows a normal disturbution

```
our_sheet_diff <-our_sheet%>% mutate(diff=Dominant_num_dots_hit-
                                       Non_dominant_num_dots_hit)
our_sheet_diff
```

```
## # A tibble: 24 x 5
##    Dominant_num_dots_hit Non_dominant_num_dot~1 Dominant_hand_first Student_name
##                    <dbl>                  <dbl> <lgl>               <chr>
##  1                    36                     24 TRUE                Ellen
##  2                    39                     33 TRUE                Jennifer
##  3                    37                     32 TRUE                Ivan
##  4                    25                     20 TRUE                Annette
##  5                    30                     27 TRUE                Paula
##  6                    22                     28 TRUE                Annette
##  7                    22                     19 TRUE                Dee
##  8                    29                     24 TRUE                Alex
##  9                    32                     27 FALSE               Sherry
## 10                    27                     18 FALSE               Max
## # i 14 more rows
## # i abbreviated name: 1: Non_dominant_num_dots_hit
## # i 1 more variable: diff <dbl>
```

```
  p10 <- ggplot(our_sheet_diff, aes(x= diff)) +
    geom_histogram()+theme_minimal()
p10
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



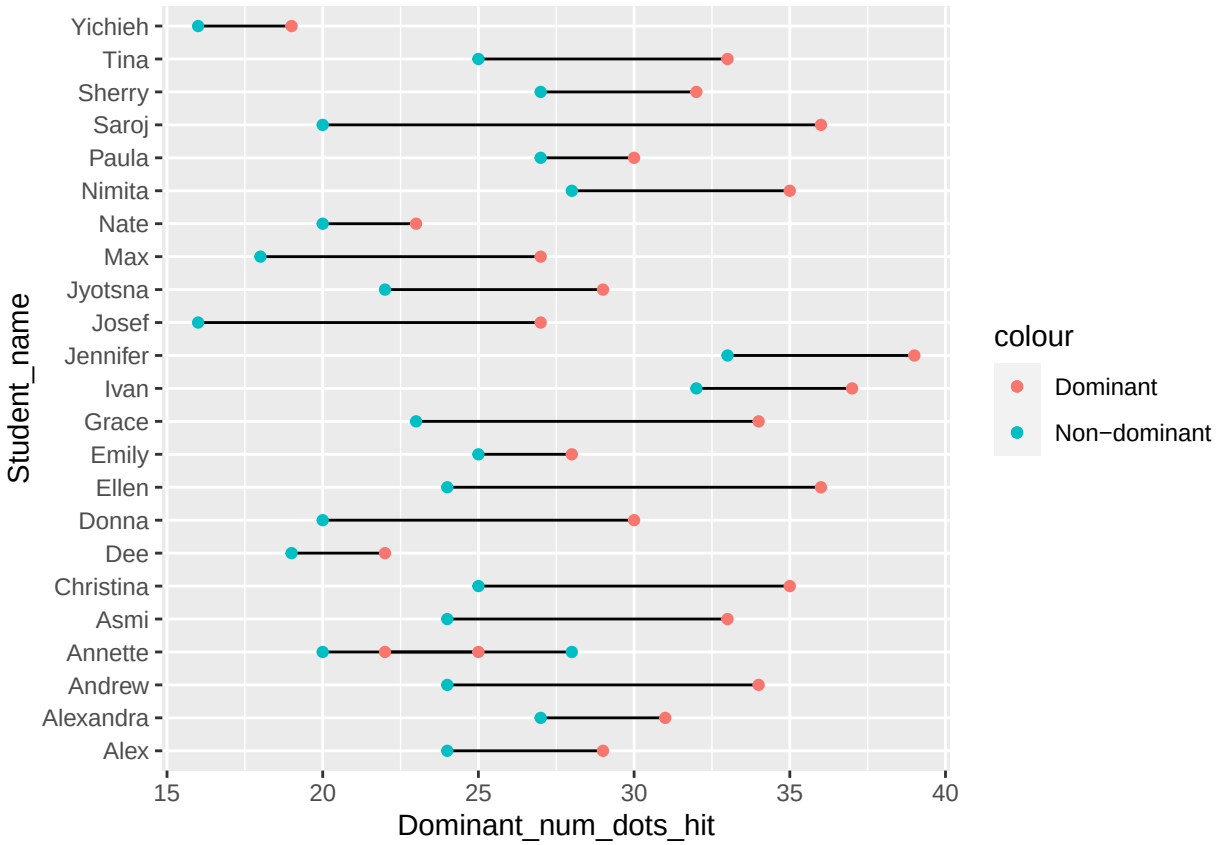

```
# YOUR CODE HERE
```

11. Before performing the test, take a look at the data by making a "dumbbell" plot. This type of plot has the student name on the y-axis, and the number of dots hit on the x-axis. For each student you put a point at the two reaction times and connect them with a line. Below is the code to make the plot. We can also color the points by hand dominance. Based on the plot, comment on whether there appears to be a significant difference between the number of points hit with the dominant or non-dominant hand.

Here is the code to make the dumbbell chart. You will need to change `data` to the name of your saved dataset.

STOP: Remove eval = F before continuing

```
# This code is provided to students because it is a bit advanced.
# You are not expected to know how to make this plot yourself!
ggplot(data = our_sheet_diff, aes(x = Dominant_num_dots_hit, y = Student_name))+
  geom_segment(aes(xend = Non_dominant_num_dots_hit, yend = Student_name)) +
  geom_point(aes(col = "Dominant")) +
  geom_point(aes(x = Non_dominant_num_dots_hit, col = "Non-dominant"))
```



Yes

12. [1 point] Use an R function to conduct a paired two-sided t-test on the data, and note the 95% confidence interval for the test. Assign your p-value rounded to 2 decimal places to the object called `p12`. Interpret the p-value and the confidence interval for the test.

```
  p12 <- t.test(our_sheet$Non_dominant_num_dots_hit,
                our_sheet$Dominant_num_dots_hit,paired = TRUE)



p12 <- round(1.688e-07,2)
p12
```

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

```
# YOUR CODE HERE
```

p value is 0 and CI is (-8.48, -477)

```
. = ottr::check("tests/p12.R")
```

```
##
## All tests passed!
```

13. [1 point] Re-run the code for the test, but this time set `paired = F`, which is incorrect. We want to run the incorrect test to compare the p-value from this test to the p-value from

the paired t-test. Determine whether the p-value is smaller or larger and assign "smaller" or "larger" to p13. Why do you think that is?

```
p13 <- t.test(our_sheet$Non_dominant_num_dots_hit,
              our_sheet$Dominant_num_dots_hit,paired = FALSE)
p13 <-"larger"
p13
```

```
## [1] "larger"
```

```
# YOUR CODE HE'RE
```

CI (-9.49, -3.76), P value is 2.83

```
. = ottr::check("tests/p13.R")
```

```
##
## All tests passed!
```

14. Lastly, we didn't use the data on the last column in the data frame, which recorded whether you were randomized to use your dominant hand first. Why might this matter? What could we have done to investigate whether it mattered?

Analyzing the data based on the randomization of hand usage order can help us understand whether any observed differences between dominant and non-dominant hand hits are due to the hand dominance or simply the order of usage.

Submission

For assignments in this class, you'll be submitting using the **Terminal** tab in the pane below. In order for the submission to work properly, make sure that:

1. Any image files you add that are needed to knit the file are in the `src` folder and file paths are specified accordingly.
2. You **have not changed the file name** of the assignment.
3. The file knits properly.

Once you have checked these items, you can proceed to submit your assignment.

1. Click on the **Terminal** tab in the pane below.
2. Copy-paste the following line of code into the terminal and press enter.

cd; cd ph142-su23/lab/lab08; python3 turn_in.py

3. Follow the prompts to enter your Gradescope username and password.
4. If the submission is successful, you should see "Submission successful!" appear as the output. **Check your submission on the Gradescope website to ensure that the autograder worked properly and you received credit for your correct answers. If you think the autograder is incorrectly grading your work, please post on Ed!**
5. If the submission fails, try to diagnose the issue using the error messages–if you have problems, post on Ed under the post "Datahub Issues".

The late policy will be strictly enforced, **no matter the reason**, including submission issues, so be sure to submit early enough to have time to diagnose issues if problems arise.

END