Exam 1

Data Science for Studying Language & the Mind

Instructions

The exam is worth **100 points**. You have **1 hour and 30 minutes** to complete the exam.

- The exam is closed book/note/computer/phone except for the provided reference sheets
- If you need to use the restroom, leave your exam and phone with the TAs
- If you finish early, you may turn in your exam and leave early

(5 points) Preliminary questions

Please complete these questions *before* the exam begins.

(a) **(1 point)** What is your full name?

(b) **(1 point)** What is your penn ID number?

(c) **(1 point)** What is your lab section TA's name?

(d) **(1 point)** Who is sitting to your left?

(e) **(1 point)** Who is sitting to your right?

1. (6 points) R basics: general

(a) **(2 points)** Which of the following are expressions? Choose all that apply.

- ☐ `x <- 5`
- ☐ `mean(data)`
- ☐ `4 + 2`
- ☐ `"Hello, world!"`
- ☐ `y <- 4 + mean(2)`

(b) **(2 points)** Which of the following occur in the code block below? Choose all that apply

```r
x <- sum(c(5, 3))
```

- ☐ a message
- ☐ a comment
- ☐ a function
- ☐ a vector
- ☐ the assignment operator

(c) **(2 points)** What additional step do we need to take to start a new R notebook in Google Colab with `File > New notebook`? Choose one.

- ☐ `File > Download > Download .ipynb`
- ☐ `File > Download > Download .py`
- ☐ `Runtime > Change runtime type` and select R.
- ☐ No further action needed; R is the default notebook for Google Colab.

2. (16 points) R basics: vectors, operations, subsetting

(a) **(3 points)** Use the `seq()` function to write an expression that would return the vector `100 200 300 400 500` and store it as `my_vector`

(b) **(2 points)** Suppose you run the following code. What will `typeof(x)` return and why? Choose one.

```
x <- c(30, 40, 50, "sixty", 70, 80)
```

- ☐ `double` due to implicit coercion
- ☐ `double` due to explicit coercion
- ☐ `character` due to implicit coercion
- ☐ `character` due to explicit coercion
- ☐ `Error: vectors must be atomic`

(c) **(2 points)** What will the following code block return? Choose one.

```
x <- 50:60
typeof(x)
```

- ☐ `integer`
- ☐ `double`
- ☐ `matrix`
- ☐ `vector`

(d) **(3 points)** Suppose you run the following code. What will be returned? Choose one.

```
c(1, 2, 3) * c(1, 2, 3)
```

- ☐ `2 4 6`
- ☐ `1 4 9`
- ☐ `1 2 3 2 4 6 3 6 9`
- ☐ `2 3 4 3 4 5 3 5 6`
- ☐ Error: non-numeric argument to binary operator‘

(e) **(2 points)** Suppose you run the following code. What will `x[c(2,4)]` return? Write your answer in the box below and *show your work.*

```
x <- seq(1, 10, by = 2)
```

(f) **(2 points)** In R, how are complex objects like matrices or arrays built? Choose one.

- ☐ lists with named elements
- ☐ vectors with attributes
- ☐ arrays of numbers
- ☐ nested loops
- ☐ none of the above

(g) **(2 points)** Suppose we run the following code. What will `is.na(x)` return? Choose one.

```
x <- c("apple", NA, "na", "orange")
```

- ☐ `TRUE`
- ☐ `FALSE`
- ☐ `FALSE TRUE FALSE FALSE`
- ☐ `FALSE TRUE TRUE FALSE`
- ☐ `2`

3. (14 points) Data visualization: basics

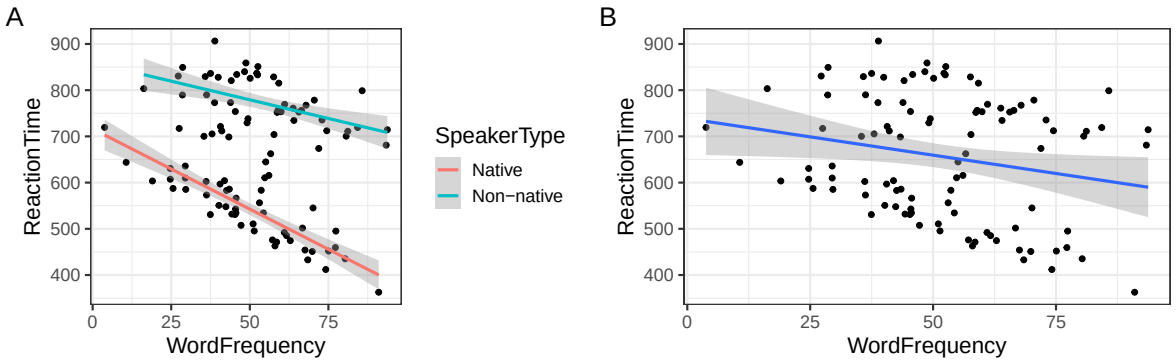
Suppose we measure the reaction times (in milliseconds) of both native and non-native speakers as they process words of varying frequency (measured as occurrences per million words). We store these data in a tibble called `rt_by_speaker`. The first 6 rows of this tibble are printed below for your reference.

```
# A tibble: 6 x 3
  WordFrequency ReactionTime SpeakerType
          <dbl>        <dbl> <chr>
1          38.8         773. Non-native
2          45.4         754. Non-native
3          81.2         711. Non-native
4          51.4         495. Native
5          52.6         851. Non-native
6          84.3         719. Non-native
```

Suppose we run the following code to visualize the data.

```
ggplot(
    rt_by_speaker,
    aes(x = WordFrequency, y = ReactionTime, color = SpeakerType)
) +
    geom_point(color = "black") +
    geom_smooth(method = "lm")
```

(a) **(2 points)** Which of the following plots will be returned? Choose one.



☐ A
☐ B
☐ There is not enough information to distinguish

(b) **(2 points)** In the code above, which of the following correctly describes which aesthetics are mapped and which are set? Choose one.

- ☐ Color is mapped to SpeakerType and the points are set to black.
- ☐ Color is set to SpeakerType and the points are mapped to black.
- ☐ Both aesthetics are mapped.
- ☐ Both aesthetics are set.

(c) **(2 points)** In the code above, which of the following correctly describes which aesthetics are global and which are local? Choose one.

- ☐ Color is mapped to the SpeakerType variable globally, and set to black locally.
- ☐ Color is mapped to the SpeakerType variable locally, and set to black globally.
- ☐ Both are global
- ☐ Both are local

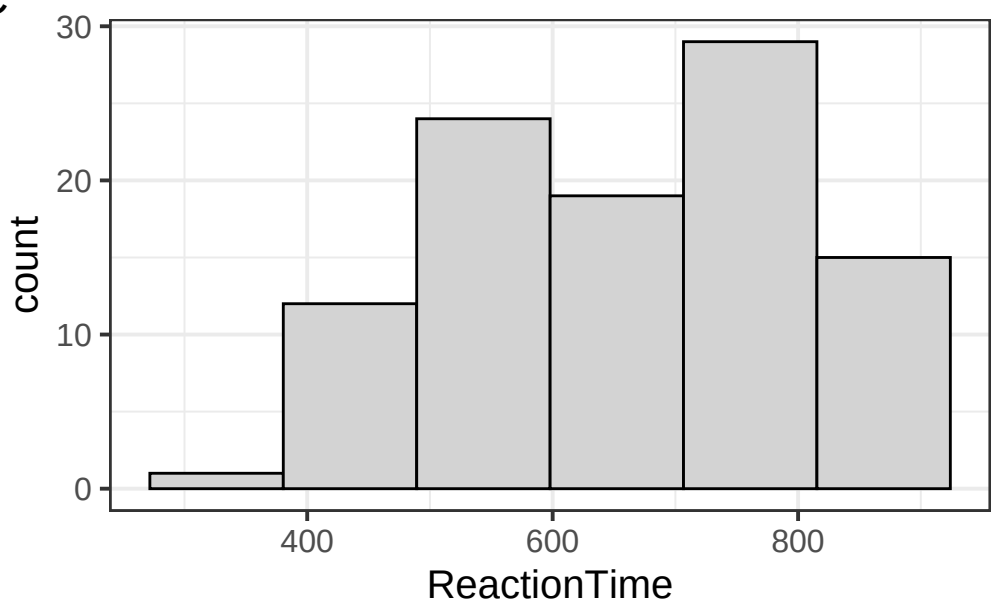
(d) **(2 points)** When `ggplot2` maps a categorical variable to an aesthetic, it automatically assigns a unique value of the aesthetic to each level of the variable. True or false, this process is called *scaling*.

- ☐ True
- ☐ False

(e) **(6 points)** Suppose we wanted to plot a histogram of the `ReactionTime` variable in the `rt_by_speaker` dataset. Fill in the blanks below such that plot C is returned.

```
____a____ %>%
    ggplot(aes( ____b____ = ReactionTime) ) +
    geom_histogram(
        fill = "lightgray",
        color = "black",
        bins = _____c______
    )
```

C



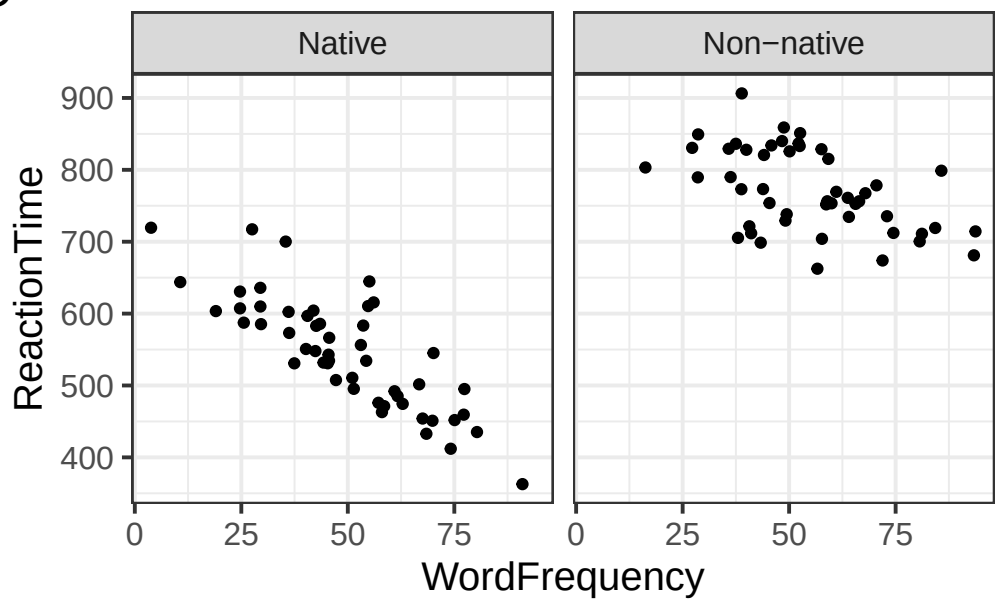
(i) **(2 points)** Fill in blank a.

(ii) **(2 points)** Fill in blank b.

(iii) **(2 points)** Fill in blank c.

4. (8 points) Data visualization: layers

D



(a) **(2 points)** Which of the following layers must be present in the code that generated plot D above? Choose one.

☐ `coord_flip()`
☐ `scale_group_manual(values = c("Native", "Non-native"))`
☐ `facet_grid(SpeakerType ~ .)`
☐ `facet_grid(. ~ SpeakerType)`
☐ `facet_grid(Native ~ Non-native)`

(b) **(2 points)** In plot D above, which of the folowing layers could be added to change the y-axis label to `RT`? Choose one.

☐ `labs(y = "RT")`
☐ `labs(x = "RT")`
☐ `t + annotate(geom="text", y = ReactionTime, text = "RT")`
☐ `t + labs(y = "RT")`

(c) **(2 points)** In plot D above, which of the following aesthetics should we set to make the points more transparent? Choose one.

☐ `color = "transparent"`
☐ `fill = "none"`
☐ `alpha = 0.5`
☐ `shape = lighter`
☐ None of the above

(d) **(2 points)** Plot D above makes use of `theme_minimal()`, one of R's built-in themes. Which of the following would adjust the size of the font to 15? Choose one.

☐ change to `theme_minimal(base_size = 15)`
☐ change to `theme_minimal(font_family = 15)`
☐ add a layer `scale_size(15)` after `theme_minimal()`
☐ add a layer `font(size = 15)` after `theme_minimal()`
☐ None of the above.

5. (11 points) Tidy data and importing

(a) **(3 points)** Fill in the blanks with `cell`, `column`, or `row` to describe the `tidy data` way of organizing datasets:

1. Each _____ forms a variable.
2. Each _____ forms an observation.
3. Each _____ is a single value.

(i) **(1 point)** Fill in blank 1.

(ii) **(1 point)** Fill in blank 2.

(iii) **(1 point)** Fill in blank 3.

(b) **(2 points)** True or false, we need to import both `readr` and the `tidyverse` in order to import data with the `read_*` functions.

☐ True
☐ False

(c) **(2 points)** Which of the following would convert the dataframe `df` to a tibble? Choose one.

☐ `as.data.frame(df, type = "tibble")`
☐ `convert_to_tibble(df)`
☐ `df %>% as_tibble()`
☐ None. Dataframes cannot be converted to tibbles.

(d) **(4 points)** Suppose we have a dataset called `my_file.csv` that looks like the following. Fill in the blanks to import the data and ensure R understands the missing values as `NA`.

```
Name,Class_Year,Favorite_Ice_Cream,GPA
Alice,Freshman,Vanilla,3.5
Bob,Sophomore,Chocolate,na
Charlie,Junior,Strawberry,3.2
Diana,Senior,Mint,3.9
Eve,Freshman,Cookies and Cream,3.6
Frank,Junior,Rocky Road,n\a
```

```
data <- ____a____(file = 'my_file.csv',
    na = _____b_____
)
```

(i) **(2 points)** Fill in blank a.

(ii) **(2 points)** Fill in blank b.

6. (11 points) Data transformation

The following questions refer to our `rt_by_speaker` dataset introduced in question 3. The first 6 rows of this tibble are printed again here for your reference.

```
# A tibble: 6 x 3
  WordFrequency ReactionTime SpeakerType
          <dbl>        <dbl> <chr>
1          38.8         773. Non-native
2          45.4         754. Non-native
3          81.2         711. Non-native
4          51.4         495. Native
5          52.6         851. Non-native
6          84.3         719. Non-native
```

(a) **(2 points)** How would you filter the dataset to keep only `Native` speakers with `ReactionTime` greater than 800 ms? Choose one.

- ☐ `rt_by_speaker %>% filter(ReactionTime < 800 & SpeakerType == "Native")`
- ☐ `rt_by_speaker %>% filter(ReactionTime < 800 & SpeakerType = "Native")`
- ☐ `rt_by_speaker %>% filter(ReactionTime > 800 & SpeakerType == "Native")`
- ☐ `rt_by_speaker %>% filter(ReactionTime > 800 & SpeakerType = "Native")`
- ☐ `rt_by_speaker %>% filter(ReactionTime = 800 & SpeakerType = "Native")`

(b) **(3 points)** Fill in the blanks to return the *median* `ReactionTime` by `SpeakerType`?

```
rt_by_speaker %>%
    group_by(___a____) %>%
    summarise(
        n = n(),
        medianRT = ______b______
    )
```

(i) **(1 point)** Fill in blank a.

(ii) **(2 points)** Fill in blank b.

(c) **(2 points)** Suppose we run the following code block. What will `n()` do? Choose one.

```
rt_by_speaker %>%
    summarise(n = n())
```

- ☐ remove all `NA`s from the dataset
- ☐ count of the number of rows in the dataset
- ☐ add the string `n` before each value in `SpeakerType`
- ☐ Nothing, because it requires a grouping
- ☐ Throw `Error: Missing arguments to n()`

(d) **(2 points)** True or false, the following code blocks are equivalent

```
# option 1
rt_by_speaker %>% select() %>% glimpse()

# option 2
select() %>% rt_by_speaker %>% glimpse()
```

- ☐ True
- ☐ False

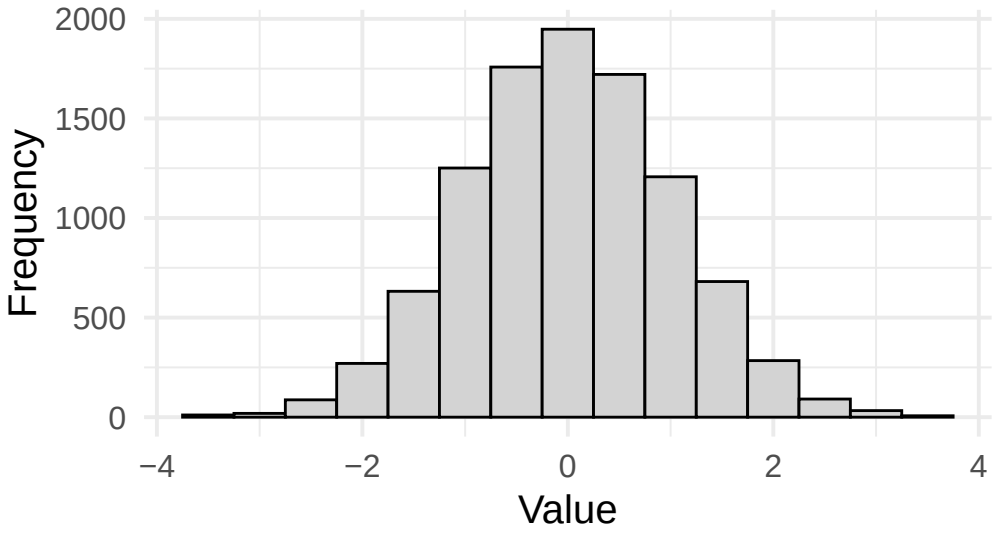
(e) **(2 points)** Suppose we want to create a new column `ReactionTimeSeconds` which converts every value in the `ReactionTime` column from milliseconds to seconds. Which of the following `dplyr` functions could accomplish this? Choose one.

- ☐ `group_by` with `summarise()`
- ☐ `filter()`
- ☐ `select()`
- ☐ `mutate()`
- ☐ `rename()`

7. (18 points) Sampling distribution

Suppose we visualize the frequency distribution of a value in a dataset, shown below.

E



(a) **(2 points)** Which of the following could best summarise the spread of these data? Choose all that apply.

- ☐ mean
- ☐ median
- ☐ sd
- ☐ interquartile range
- ☐ p-value

(b) **(2 points)** Which of the following could summarise the central tendency of these data? Choose all that apply.

- ☐ mean
- ☐ median
- ☐ sd
- ☐ interquartile range
- ☐ p-value

(c) **(2 points)** Suppose we visualize the probability density function of the distribuition that generated these data (below) What could be the height of the probability density function at a value of 2? Choose one.

☐ 0.981
☐ 0.501
☐ 0.053
☐ Not enough information to determine this

(d) **(2 points)** True or false, the probability density function shown in plot F is given by the following equation.

$p(x) = \frac{1}{max - min}$

☐ True
☐ False

(d) **(6 points)** Suppose we want to generate the bootstrap sampling distribution for the *median* of `value` in our dataset. Fill in the blanks to accomplish this with the `infer` package, generating 500 bootstrapped samples. Note that `data` has 10000 rows.

```
library(____a_____)

data %>%
    specify(response = value) %>%
    generate(reps = _____b_____, type = _____c_____) %>%
    calculate(stat = _____d_____)
```

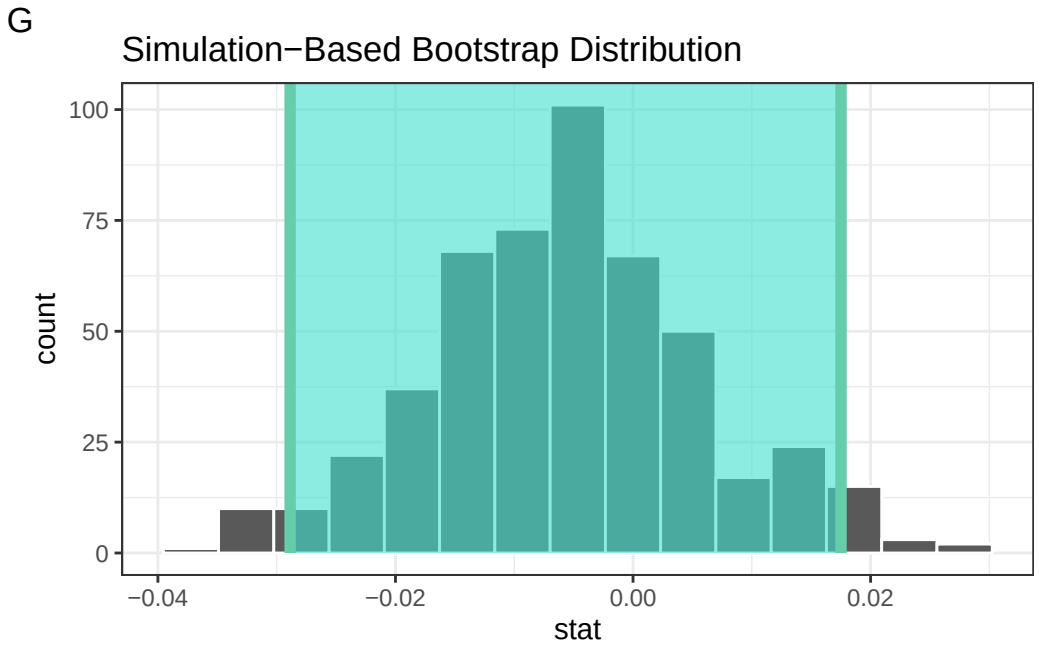
(i) **(1 point)** Fill in blank a.

(ii) **(2 points)** Fill in blank b.

(iii) **(2 points)** Fill in blank c.

(iii) **(1 points)** Fill in blank d.

(e) **(2 points)** The shaded area of the figure shows the 95% confidence interval. If we were to decrease the `level` of confidence to 68%, the confidence interval would become (choose one):



☐ Narrower
☐ Wider
☐ Unchanged
☐ Not enough information to determine this

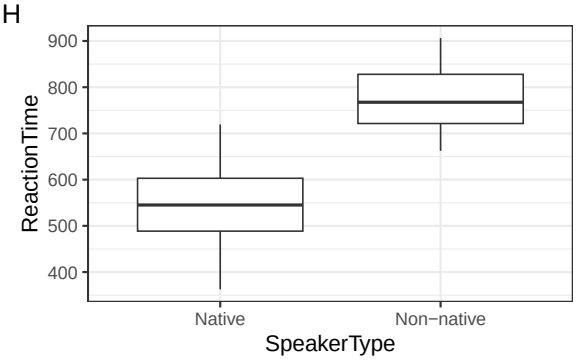
(f) **(2 points)** Which of the following best describes the standard error? Choose one.

☐ The mean of the parameter
☐ The standard deviation of the population
☐ The standard deviation of the sampling distribution
☐ The mean of the parameter estimate

8. (11 points) Hypothesis testing

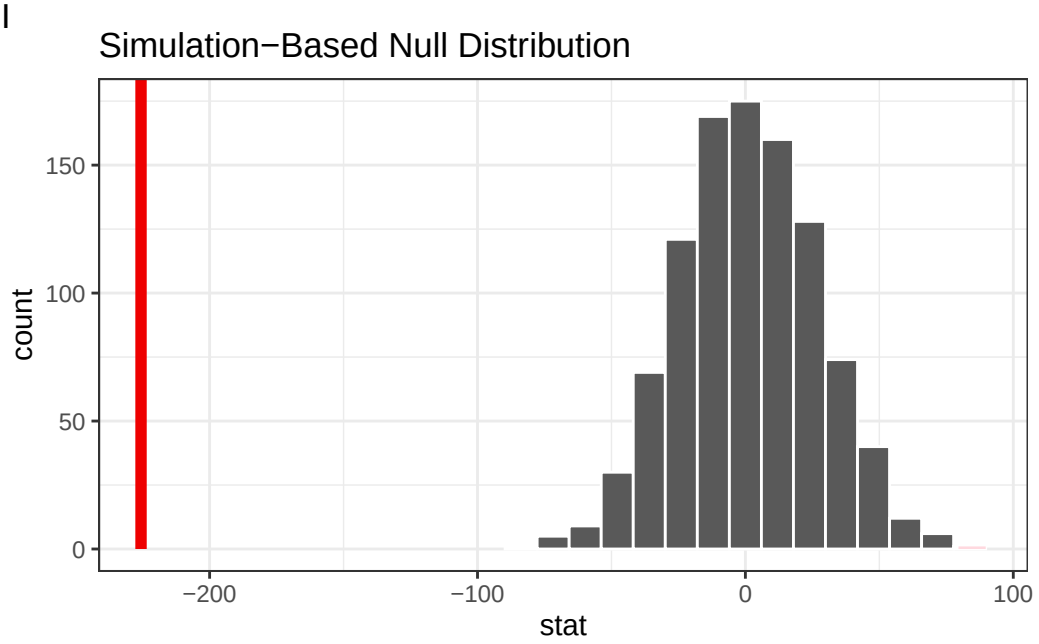
Suppose we want to determine whether there is a difference in mean `ReactionTime` between our Native and Non-native speakers in the `rt_by_speaker` dataset. We can visualize these data with a boxplot.

H



Then we use `infer` to generate the sampling distribution for the difference in mean `ReactionTime` between the Native and Non-native speakers. We've visualized this distribution here and called `shade_p_value()` to generate the vertical line.

I



(a) **(3 points)** Step 1 of the 3-step hypothesis testing framework is to pose the null hypothesis. State the null hypothesis here in the box below.

(b) **(2 points)** Step 2 is to ask, if the null hypothesis is true, how likely is our observed pattern of results? Given the figures above, which of the following could be the p-value? Choose one.

☐ 0
☐ -100 to 100
☐ -220
☐ There is not enough information to determine this.

(c) **(2 points)** Step 3 is to decide whether to reject the null hypothesis. True or false, we failed to reject the null hypothesis. Assume our threshold for rejection is $p < 0.05$.

☐ True
☐ False

(d) **(2 points)** Why do we pose a null hypothesis? Choose one.

☐ It is the hypothesis most likely to be true.
☐ It allows us to generate predictions based on prior beliefs.
☐ It is the hypothesis for which we can simulate data.
☐ It ensures that the alternative hypothesis is proven false.

(e) **(2 points)** True or false, we can compute a p-value by counting the number values in our null distribution that are more extreme than the actual observed value and then dividing by the total number of simulations that we generated.

☐ True
☐ False