visual data in R

Making plots

Story

You are working with a professor of anthropology at Fancypants University. She has just discovered a new species of monkey that she wants you to collect data on. So you are off to the island of Naboombu, where you will be given the task of watching these primates.

Collecting your data, part 1

When you arrive you notice that there are a whole lot of monkeys here and you need to find a way to get an idea of what they look like. So you and your team decide to start by measuring their tails (they aren't apes!). you manage to measure 500 of these fellas (ok, this is a bit of a reach but lets say that you are *really* quick at measuring)

run the following R code to get the first monkey data!

```
library(tidyverse)
```

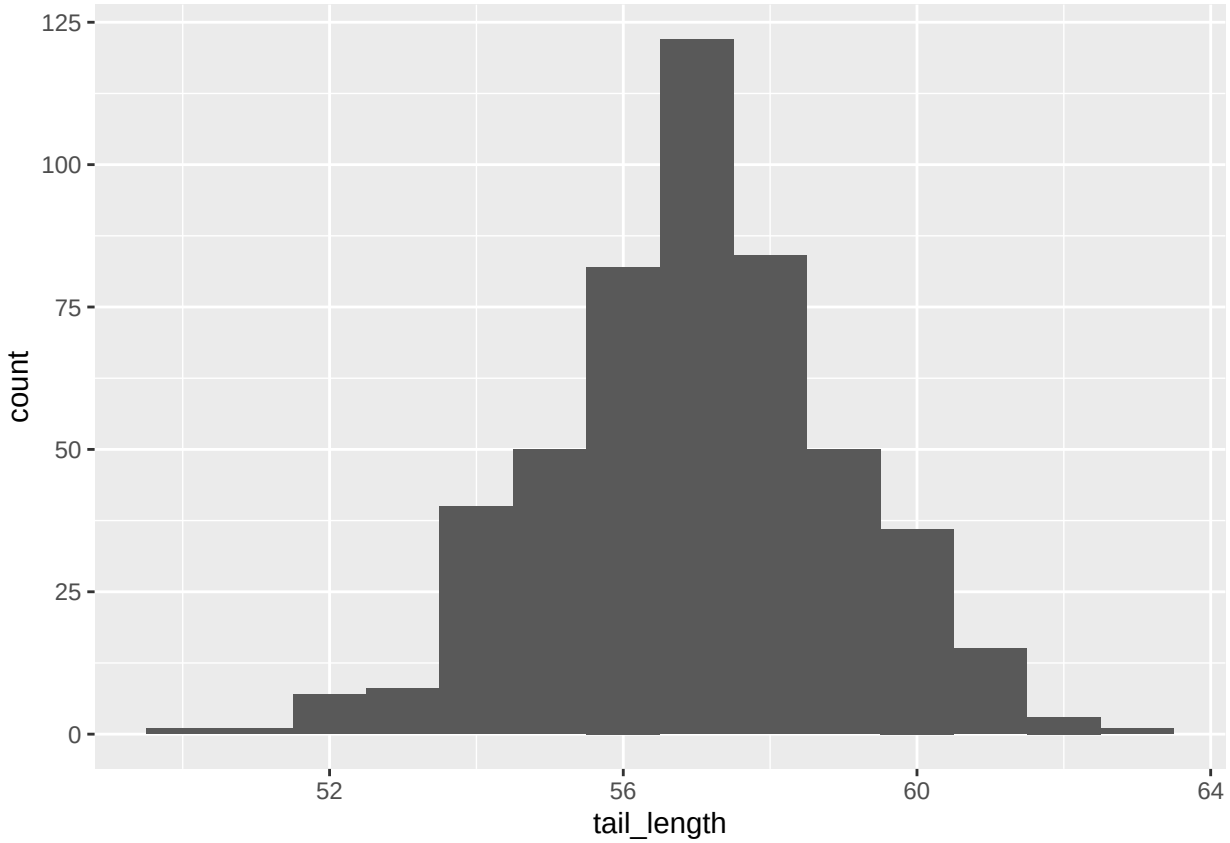
```
## Warning: package 'tidyverse' was built under R version 4.0.3
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.0
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

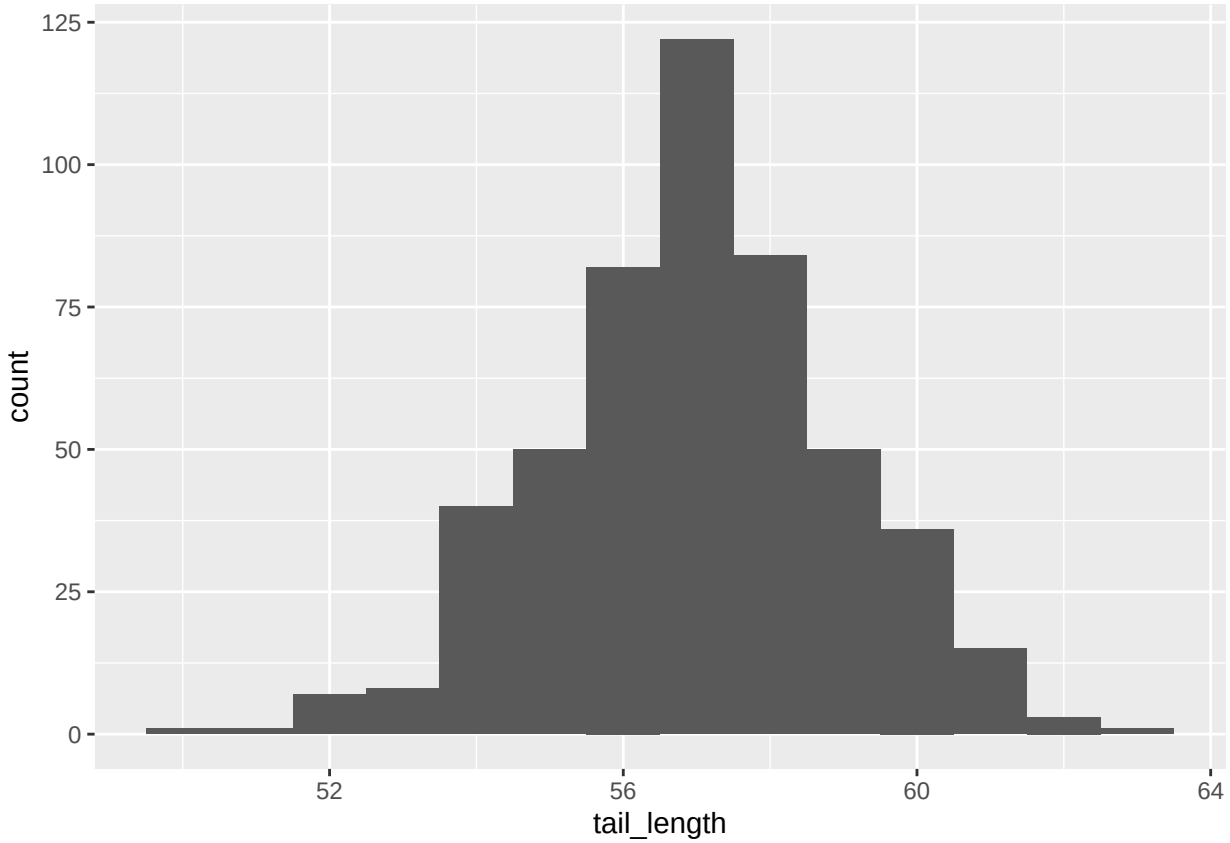
```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
tail_length <-rnorm(500,57,2)
sample_number <-1:500
monkey_pop <- rep(letters[1], 500)
monkey_data_1 <- tibble(sample_number, tail_length, monkey_pop)

monkey_data_1 %>%  ggplot(aes(tail_length)) + geom_histogram(binwidth =  1)
```



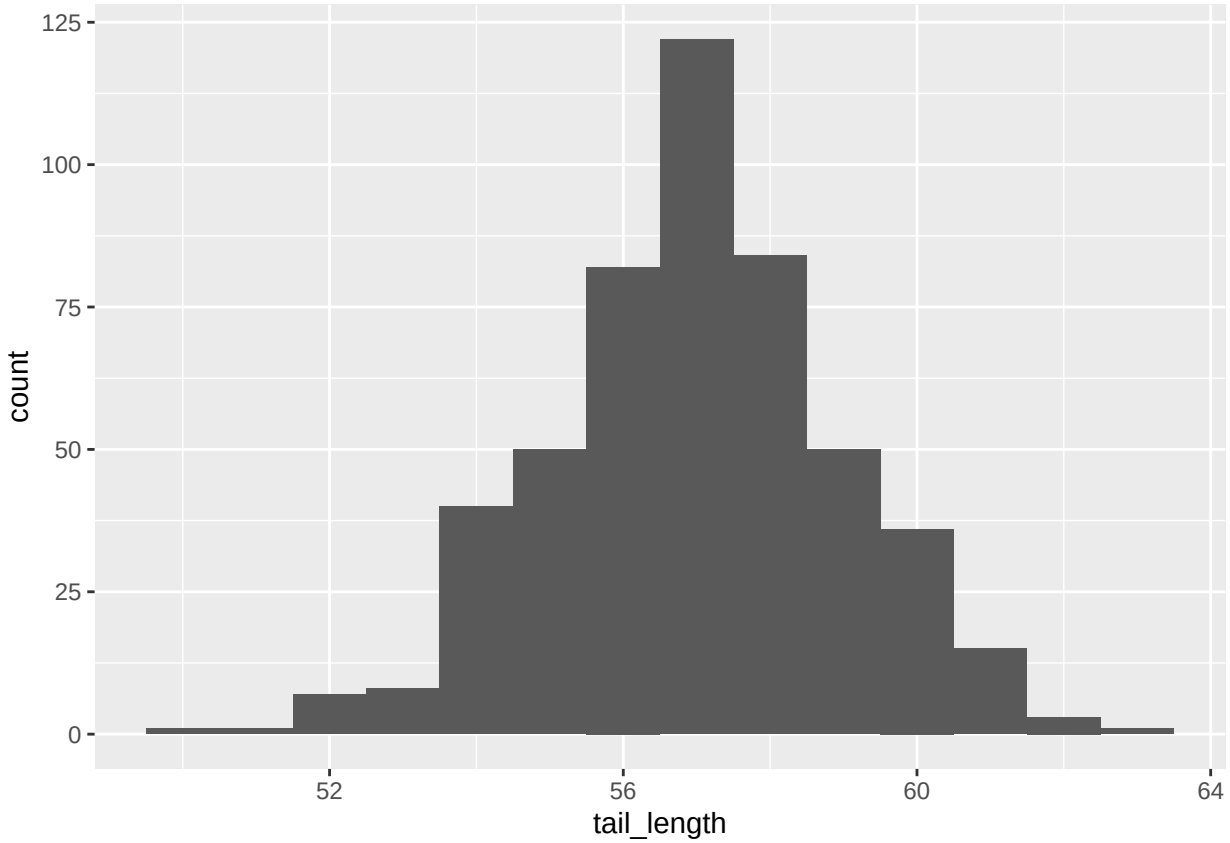
You call your professor and she is **so happy** to hear your good news. But she wants the data in be visualized. Run the following code to make a histogram.

```
monkey_data_1 %>%  ggplot(aes(tail_length)) + geom_histogram(binwidth =  1)
```



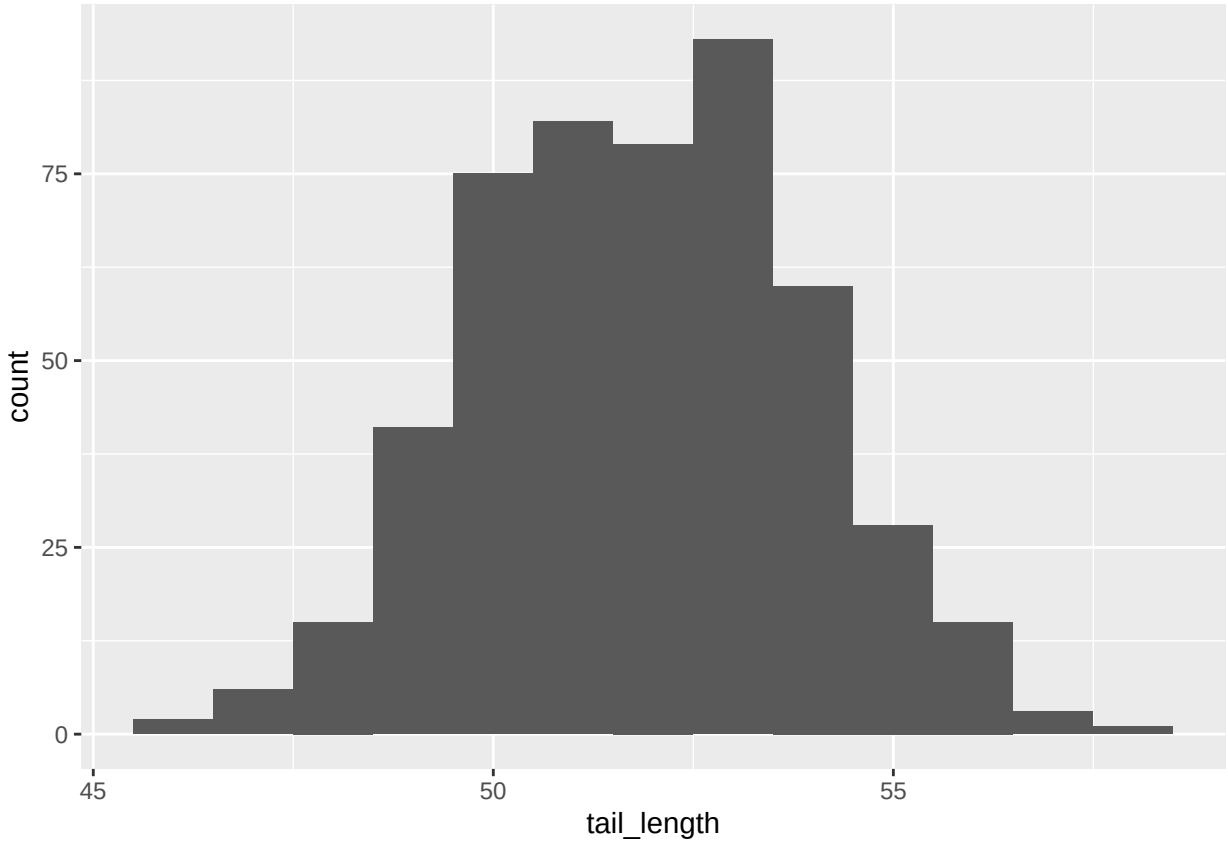
Ok, but she wants the histogram to be blue. How can you add to the code below to make it blue?

```
# change the code to make the histogram blue
monkey_data_1 %>%  ggplot(aes(tail_length)) + geom_histogram(binwidth =  1)
```



After sending your data to your professor you hear about another group of monkeys on a different part of the island run the code below to get that data and make a histogram!

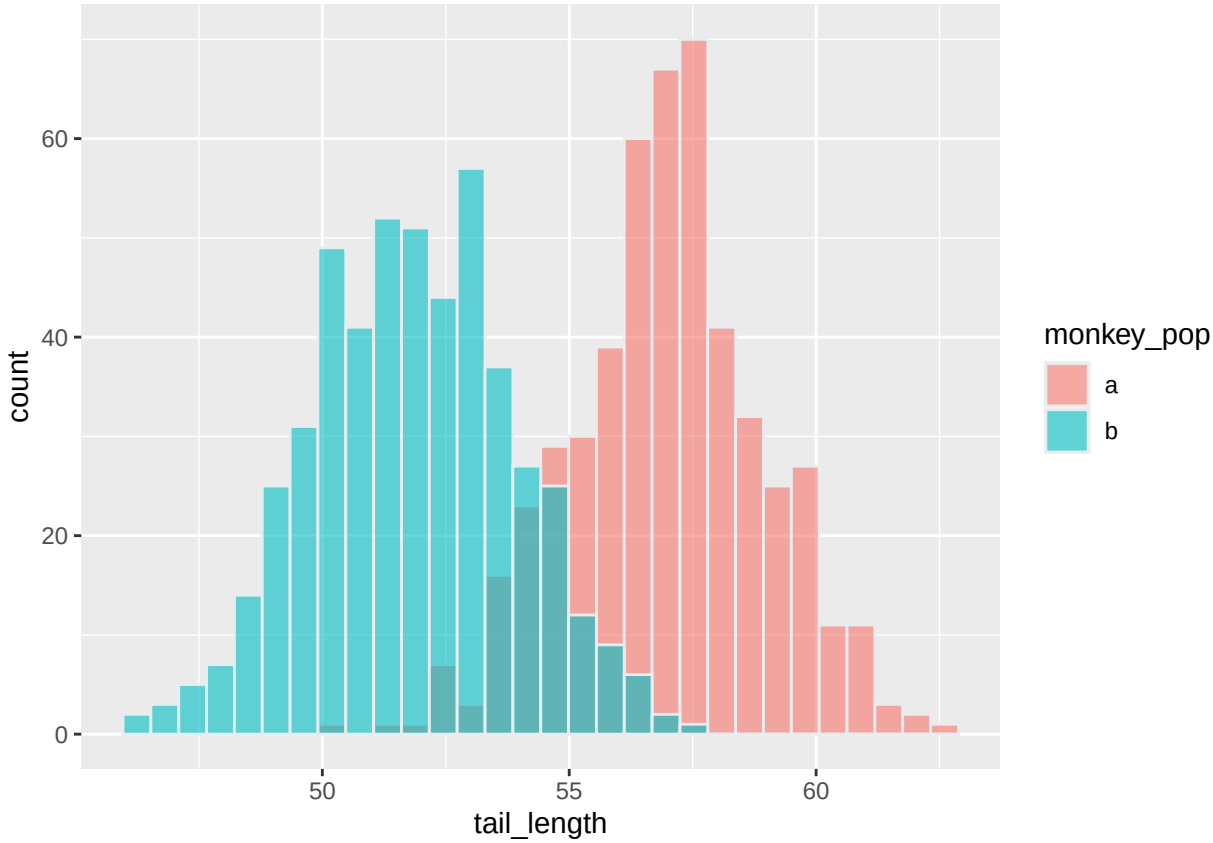
```
tail_length <-rnorm(500,52,2)
sample_number <-501:1000
monkey_pop <- rep(letters[2], 500)
monkey_data_2 <- tibble(sample_number, tail_length, monkey_pop)

monkey_data_2 %>%  ggplot(aes(tail_length)) + geom_histogram(binwidth =  1)
```



Now you want to look at both of the populations at the same time! Thankfully you have taken this class and have some R code ready to do this:

```
all_data <- bind_rows(monkey_data_1, monkey_data_2)

all_data %>% ggplot(aes(tail_length, fill=monkey_pop )) + geom_histogram(color="#e9ecef", alpha=0.6, pos
```

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



So you look at this and think to yourself "hey, they overlap a bit but there are difference at the tails. I am going to send this to my adviser and she will be soooooo happy!"

A few hours later you get an email with these requests. Your adviser wants you to do all of the following to the figure you have made

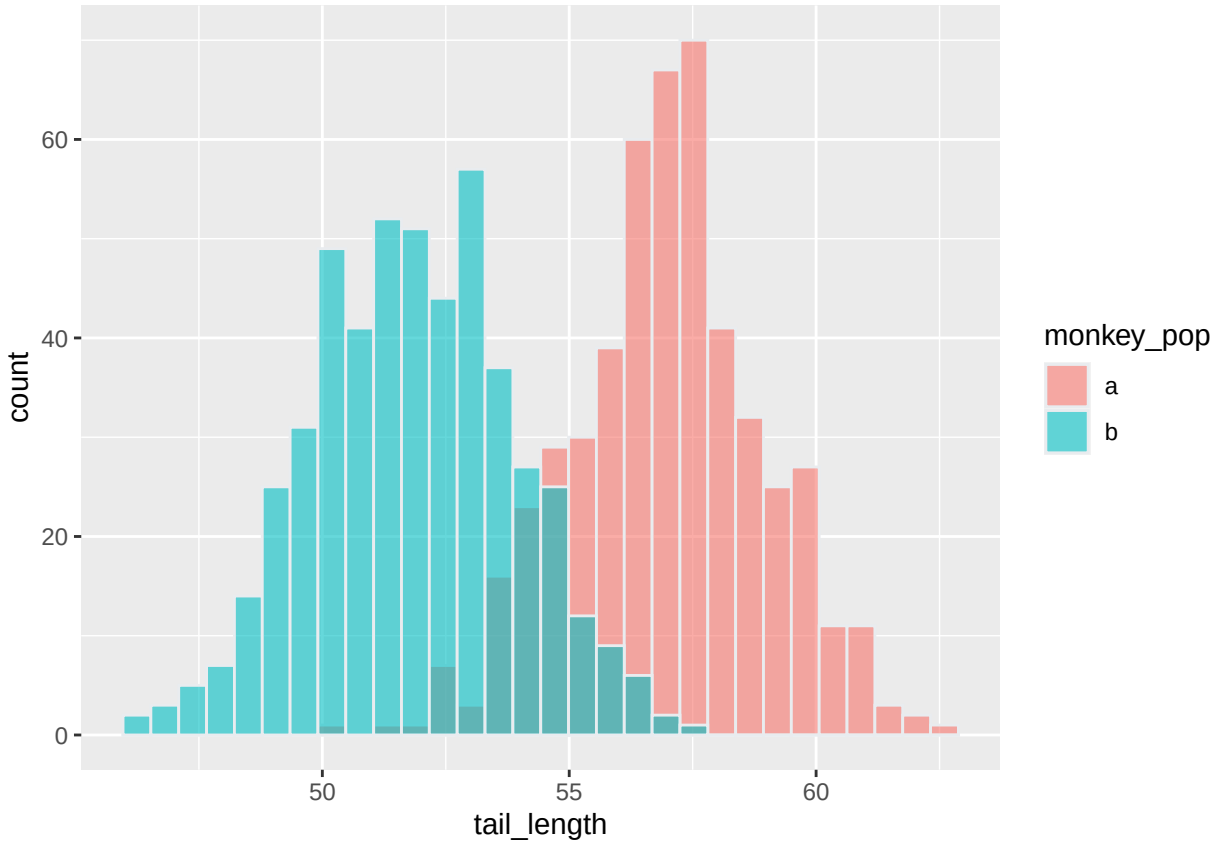
Set one

1. Put the legend on the bottom rather than on the right
2. Label the x axis "tail length (cm)"
3. Label the y axis "number of observations"
4. Give it a title that is meaningful
5. Put a credit somewhere that says who made the figure
6. Change the legend so that it says "Monkey groups"
7. Xhange the labels from "a" and "b" to "Group one" and "Group 2"
8.

lets take this step by step. copy the below R code and use each step to build the new graph. If you need help use the class readings or ask on Discord!

```
all_data %>% ggplot(aes(tail_length, fill=monkey_pop )) + geom_histogram(color="#e9ecef", alpha=0.6, pos
```

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



OK, phew! you have this all ready to go... but then your prof calls you and says "can you do one more thing for me..."

Bonus (this is hard so do it if you want the challenge and have the time)

1. put a background image on the plot
2. make a table of the data to print in R

New visualizations

Set two

New data

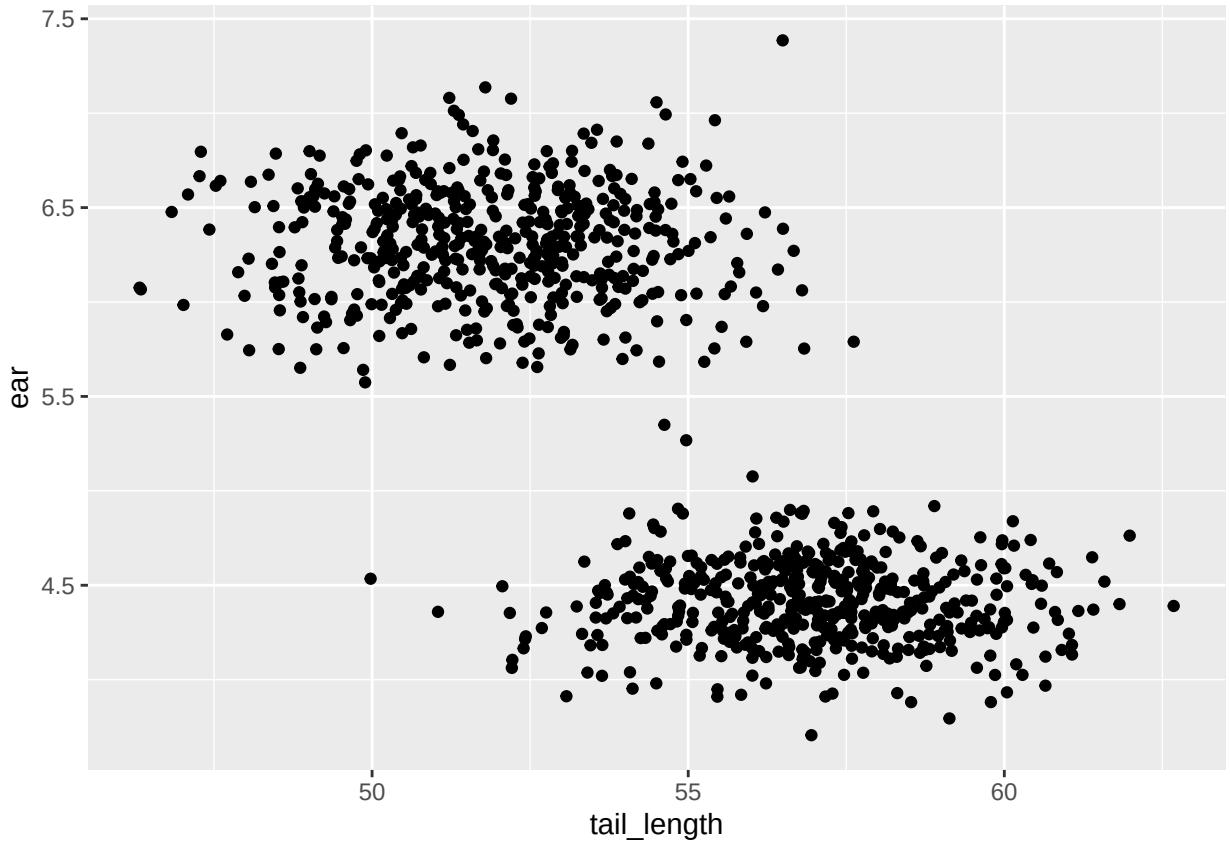
You and your professor submit your data to a journal. After a few weeks you get your peer review back and the reviewers say "wow, this is cool! but ya know what, we think you need more measurements before we can conclude these are 2 different populations. can you go and measure ear length?"

Run the following code to get the new_data

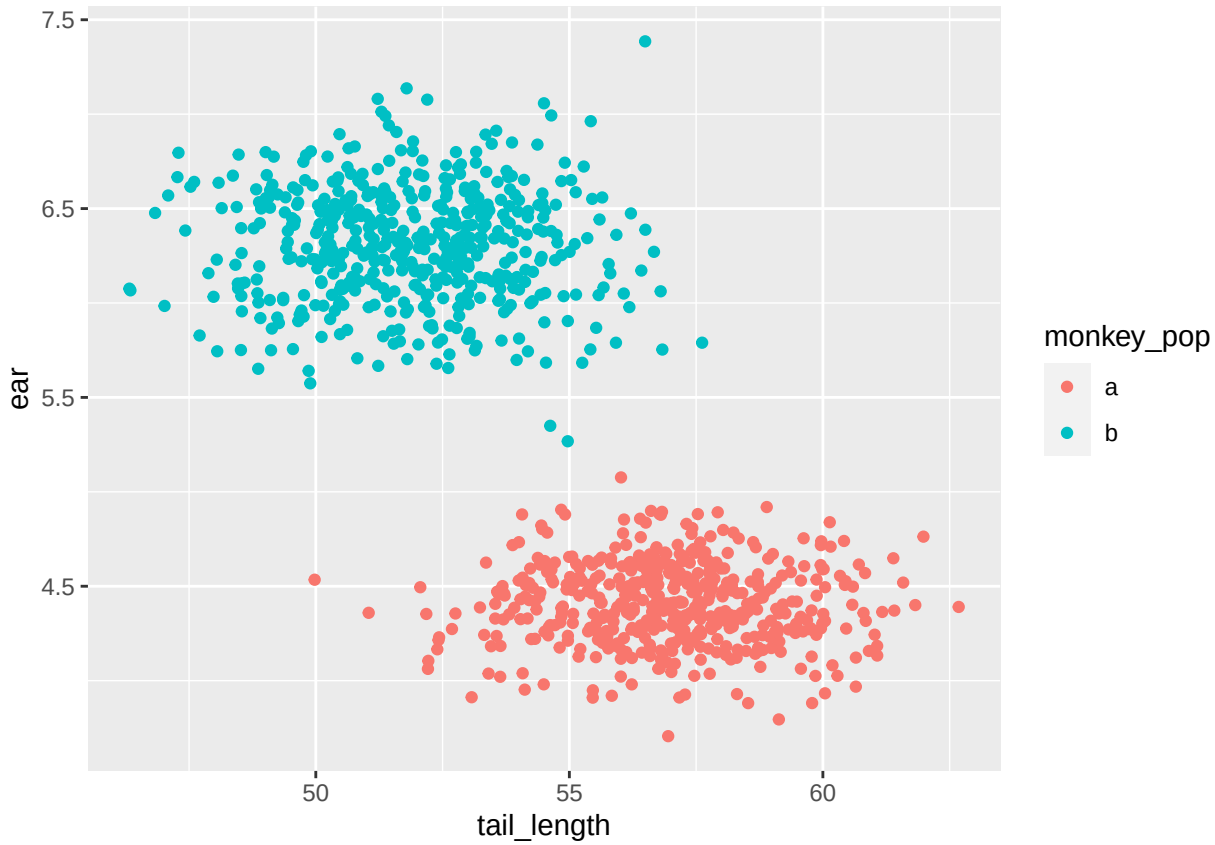
```
ear1 <- rnorm(500, 4.4, .22)
ear2 <- rnorm(500, 6.3, .31)

ear1 <-tibble(ear =rnorm(500, 4.4, .22), monkey_pop = rep(letters[1], 500), sample_number = 1:500)


ear2 <-tibble(ear =rnorm(500, 6.3, .31), monkey_pop = rep(letters[2], 500), sample_number = 501:1000)

ear_all <- bind_rows(ear1, ear2)

new_data <- all_data %>% left_join(ear_all, by= c("monkey_pop", "sample_number"))
```

Since you know have two measurements on each monkey you need a new way to visualize these data. Remembering what you learned at AppState you go and make a scatterplot of these data:

```
new_data %>%   ggplot(aes(tail_length, ear)) + geom_point()
```




```
new_data %>%   ggplot(aes(tail_length, ear, color = monkey_pop)) + geom_point()
```



She comes back and asks for the following updates:

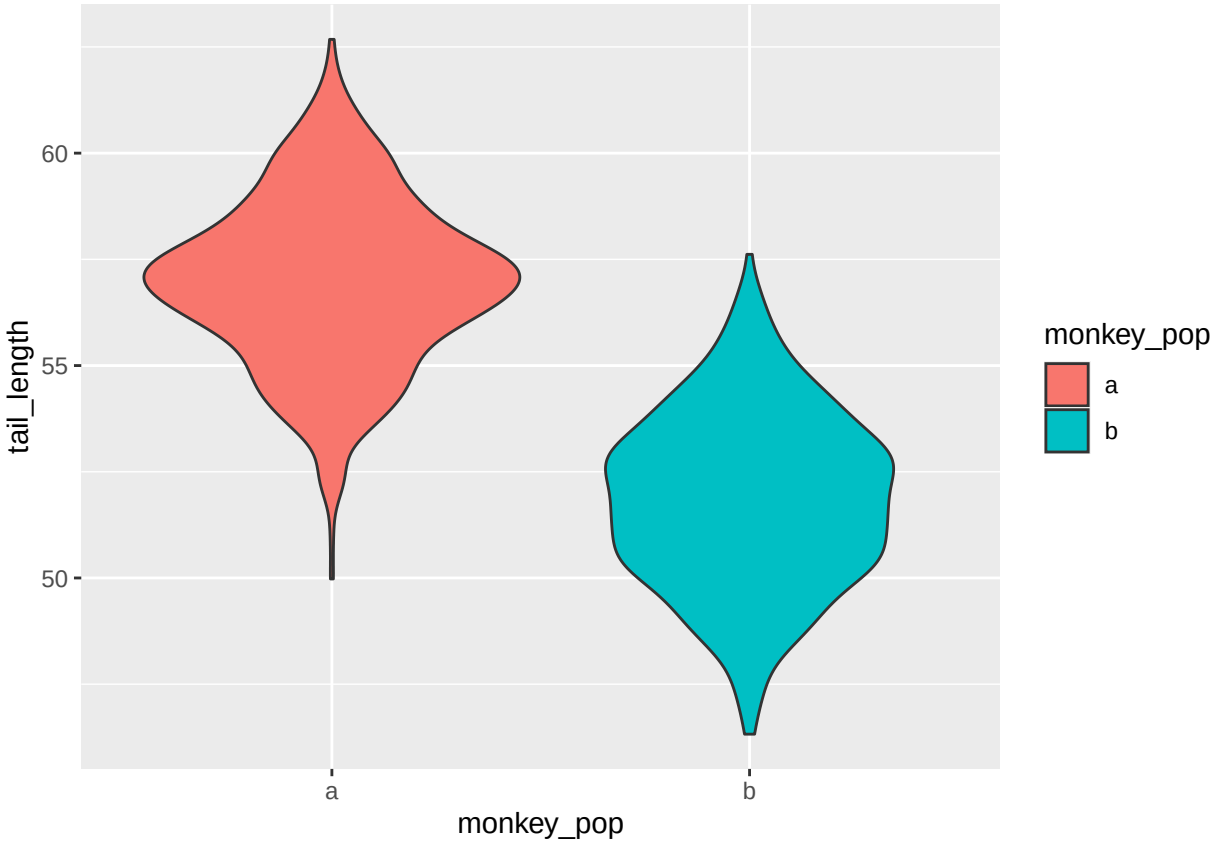
Set three:

1. Get rid of the legend
2. Add a regression line for each group
3. Add a transparency o deal with over plotting
4. Change the shape so that each group has a different shape
5. Add an ellipse to the groups (s**tat_ellipse()**

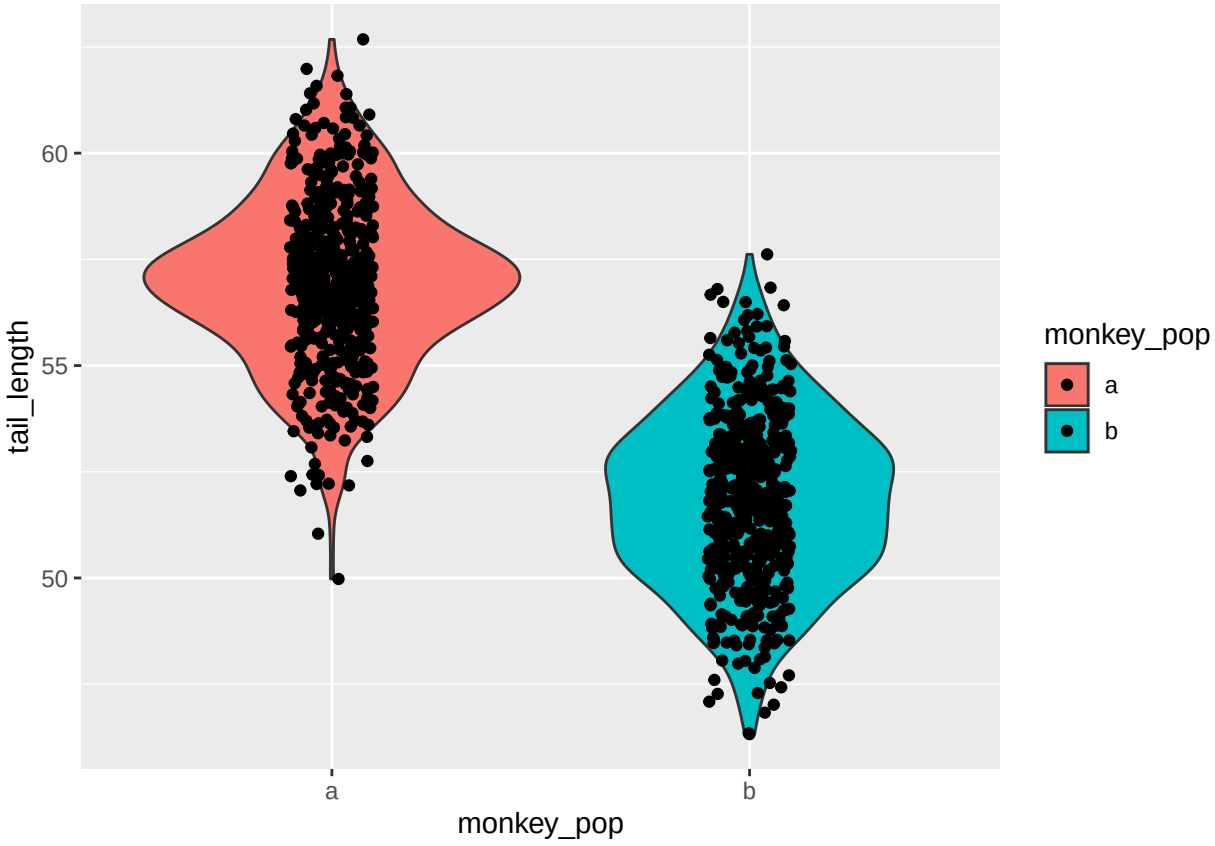
Your adviser now asks if you can make the plots with a different geom

1. make a density plot
2. make a ridgeline plot
3. make a boxplot
4. make a violin plot

```
new_data %>%   ggplot(aes( monkey_pop, tail_length, fill = monkey_pop)) + geom_violin()
```




```
new_data %>%   ggplot(aes(monkey_pop, tail_length, fill = monkey_pop)) + geom_violin()+ geom_jitter(hei
```



export the data

she wants the whole dataset sent to her so she can take a look at it herself

```
write_csv(new_data, "my_monkeydata.csv")
```

Set Four

1. change the name of the csv file so it is called "new_primate_data"?
2. change the format of the exported file to an Excel sheet (this is hard)

What else can you do with R visualizations

Now that you have learned a lot about how to work with visualization data i want you to take a moment and think about how to approach another dataset

The palmerpenguins package contains two datasets.

One is called `penguins`, and is a simplified version of the raw data; see `?penguins` for more info

The second dataset is `penguins_raw`, and contains all the variables and original names as downloaded; see `?penguins_raw` for more info.

Both datasets contain data for 344 penguins. There are 3 different species of penguins in this dataset, collected from 3 islands in the Palmer Archipelago, Antarctica. The culmen is the upper ridge of a bird's

bill. In the simplified penguins data, culmen length and depth are renamed as variables bill_length_mm and bill_depth_mm to be more intuitive.

```
#install.library("palmerpenguins")
library(palmerpenguins)
```

```
## Warning: package 'palmerpenguins' was built under R version 4.0.3
```

```
penguins %>%
  count(species)
```

```
## # A tibble: 3 x 2
##   species       n
##   <fct>     <int>
## 1 Adelie      152
## 2 Chinstrap    68
## 3 Gentoo      124
```

```
#> # A tibble: 3 x 2
#>   species       n
#>   <fct>     <int>
#> 1 Adelie      152
#> 2 Chinstrap    68
#> 3 Gentoo      124
penguins %>%
  group_by(species) %>%
  summarize(across(where(is.numeric), mean, na.rm = TRUE))
```

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

```
## # A tibble: 3 x 6
##   species   bill_length_mm bill_depth_mm flipper_length_mm body_mass_g  year
##   <fct>              <dbl>         <dbl>             <dbl>       <dbl> <dbl>
## 1 Adelie              38.8          18.3              190.       3701. 2008.
## 2 Chinstrap           48.8          18.4              196.       3733. 2008.
## 3 Gentoo              47.5          15.0              217.       5076. 2008.
```

think about what the above code is doing?

```
install.packages("palmerpenguins")
```

```
## Warning: package 'palmerpenguins' is in use and will not be installed
```

```
library(palmerpenguins)
glimpse(penguins)
```

```
## Rows: 344
## Columns: 8
## $ species           <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, A...
## $ island            <fct> Torgersen, Torgersen, Torgersen, Torgersen, Torge...
```

```
## $ bill_length_mm    <dbl> 39.1, 39.5, 40.3, NA, 36.7, 39.3, 38.9, 39.2, 34....
## $ bill_depth_mm     <dbl> 18.7, 17.4, 18.0, NA, 19.3, 20.6, 17.8, 19.6, 18....
## $ flipper_length_mm <int> 181, 186, 195, NA, 193, 190, 181, 195, 193, 190, ...
## $ body_mass_g       <int> 3750, 3800, 3250, NA, 3450, 3650, 3625, 4675, 347...
## $ sex               <fct> male, female, female, NA, female, male, female, m...
## $ year              <int> 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2...
```

take a look at these data. what sort of figures might you want to make? think about how you could tell a story about these penguins via an image. then write some code that will let you do this. when you are done, take a moment to explain in prose why you made this figure and how you used R to do it