NAME: Key UID:_____Psych 100A Spr 2019 Montoya

Please get in groups of 6! We're passing out one sheet of white paper and one of PURPLE. Fold each piece of paper so there are 6 sections. Each person should write their age in months on the white piece of paper (in a quadrant) and their height in inches on the PURPLE piece of paper (in a quadrant). Once everyone's height and age are on the pieces of paper, rip them up to create smaller pieces of paper.

Line up your pairs of observations in two columns (Height and Age) so that each person's height is next to their age. This is your data! It's even in tidy format!

217	71
218	58
122	80
204	62
220	72
216	60

1. Use R to create a dataset which looks like our dataset here. Estimate a linear model which predicts height based on age. Change the code below to correspond to your data.

```
data <- data.frame(height = c(71, 58, 80, 62, 72, 60), age = c(217,
218, 122, 204, 220, 216))
```

2. Below is the output from estimating the linear model $Y_i = b_0 + b_1 X_i + e_i$. Where Y_i is height, and X_i is age. Circle in the output where we find b_1 and write a one sentence interpretation of b_1 based on the output (Interpret for a family member, not your stats teacher!).

```
Call:
lm(formula = height ~ age, data = data)

Coefficients:
(Intercept)         age
    409.285      -3.123
```

Interpretation:

For each one month increase in age, we predict a 3.123 inch decrease in height.

3. Now try with your data. **Fill in the table** to the right, estimate a linear model predicting height from age in your dataset, and **provide the estimate of b_1 with an interpretation** of what it means in your data. Check out page 3, there is a graph that you'll fill out shortly. Draw a big thick vertical line to represent where the b1 from the original sample is.

Please note that your data will be different!
b1 = -0.01925; For each 1 month increase in age, we predict a .01925 inch decrease in height

```
lm(height~age, data = data)
 Call:
lm(formula = height ~ age, data = data)
Coefficients:
(Intercept)          age
   72.51616     -0.01925
```

Age	Height
234	73
337	68
367	62
350	65
334	71
253	60

4. When thinking about model comparison we want to imagine a world where the simple model is true. Write out the GLM equation for the simple model here:

   Y_i=b_0+e_i

5. Notice that the simple model is a version of the complex model, where $b_1 = 0$. We can interpret this as a situation where knowing a person's X **does not** help us guess their Y. Can you apply this interpretation to the current situation?

   Knowing a person's age does not help us guess their height.

   If knowing a person's age does not help us guess their height, then it should not matter that Person A's height is attached to Person A's age, rather it just as likely could have been Person B's height. We can use this understanding to generate a sampling distribution of b_1s which all come from samples where height and age are unrelated. We can do this by shuffling height and re-estimating our model!

6. Give it a try! Pick up your pile of 6 height cards, shuffle them up, and they lay them down again in a column next to the age variables. Fill in the table to the right with your new dataset.

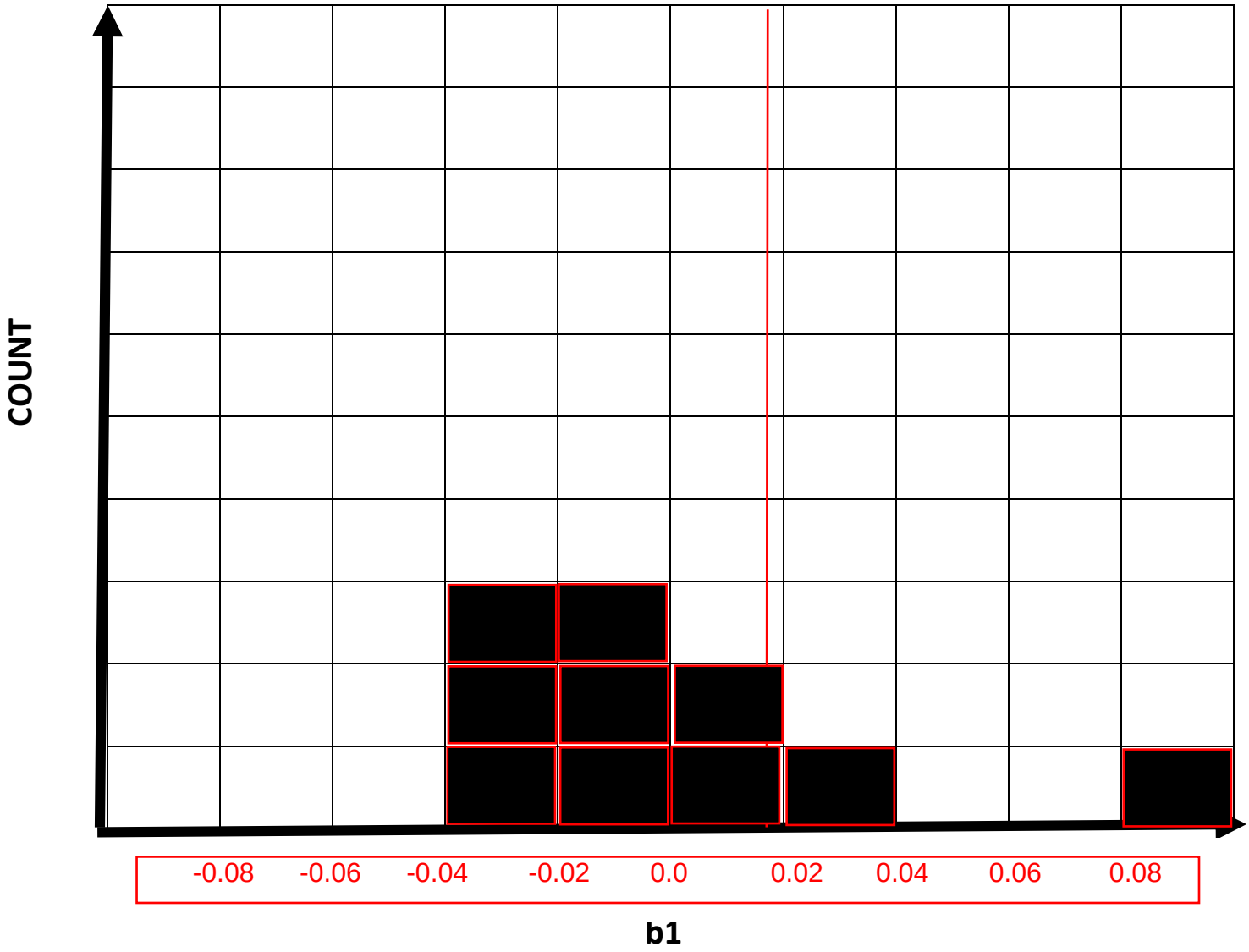
Age	Height
234	73
337	68
367	71
350	60
334	62
253	65

7. Estimate a linear model predicting height from age using this new shuffled dataset. What is b_1 in this new dataset?

```
> data2 <- data.frame(height = c(73, 68, 71, 60, 62, 65), age = c(234, 337, 367, 350, 334, 253))
> lm(height~age, data = data2)
Call:
lm(formula = height ~ age, data = data2)
Coefficients:
(Intercept)          age
   76.39623     -0.03167
```

8. You're going to do step 6 and 7 a few times. When you do fill in b_1 from each shuffled sample on the sampling distribution on the next page. Do it at least 10 times! Break up the task within your team for efficiency, have people doing different parts: shuffle, put data in R and run linear model, add b1 to histogram.

Mark from 3



9. Is your sampling distribution centered around the original estimate from the original sample? Explain why this distribution is or is not centered around the original estimate?

If sampling distribution is centered around original estimate (like above):

The sampling distribution is centered around the original estimate because our original estimate is close to zero. The sampling distribution will always be centered around zero, because shuffling generates b1s under the simple model, where beta1 is zero.

If sampling distribution is not centered around original estimate:

The sampling distribution is not centered around our original estimate because shuffling generates b1s under the simple model, where beta1 is zero. Our estimate from our original sample seems to have a much larger/smaller estimate of b1 than ones generated from the shuffling procedure

10. Compare where the shuffled samples ended up as compared to the original sample estimate (marked on your graph with a vertical line). Does it look like your data could have been generated from the simple model? Explain why or why not.

If original b1 is contained in distribution (like above):

It seems like our data could have been generated by the simple model since the b1 that we observed is very similar to the b1s generated under the simple mode.

If original b1 is not contained in distribution:

If seems that our data seems unlikely under the simple model. The original b1 estimate is very far away from the generated estimates from the simple model, so it seems like our data is unlikely if the simple model is true.

YOU CAN WORK AT YOUR OWN PACE FROM HERE ON OUT, BUT I RECOMMEND CONTINUING TO CHECK IN WITH YOUR GROUP

11. Let's start to do this using R, since shuffling by hand can be pretty arduous. Try running the command `shuffle(data$height)`. Describe what you get? What is this equivalent to doing with your cards?

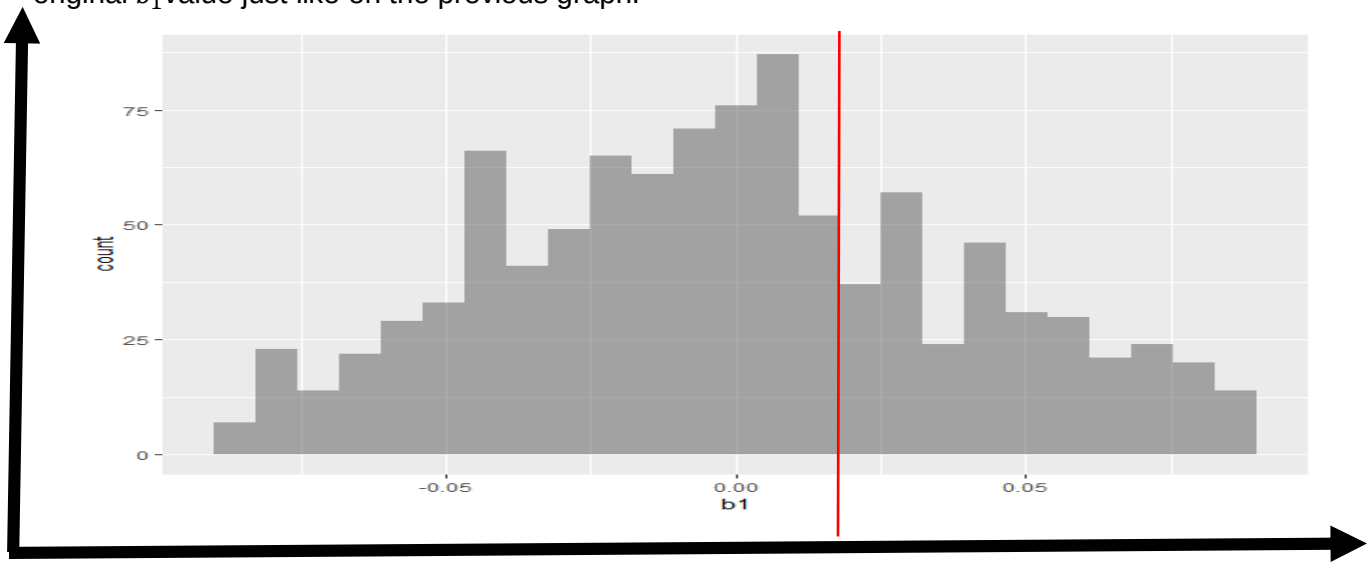
We get the heights from the original dataset in a random order. This is equivalent to taking the height cards, shuffling them up, and laying them down in order

12. Now try estimating a model with shuffled height values and pulling out the b1 coefficient! Below is the command for the original analysis, how would we change the code to use shuffled height values instead (edit the text below)?

```
b1(shuffle(height)~age, data = data)
```

```
b1(height~age, data = data)
```

13. What if we shuffle many many times, each time pulling out a b1 to represent a random sample from a population where the simple model is true. Do this using a do()* statement! Draw a histogram of the sampling distribution you get (no need to be perfect, just the relative shape), make sure to label the X and Y axis. Draw a vertical line on this plot as well, at the place of the original $b_1$value just like on the previous graph.




```
> SDOb1 <- do(1000)*b1(shuffle(height)~age, data = data)
> gf_histogram(~b1, data = SDOb1)
```

14. Compare where the shuffled samples ended up as compared to the original sample estimate (marked on your graph with a vertical line). Does it look like your data could have been generated from the simple model? Explain why or why not.

If original b1 is contained in distribution (as above):
It seems like our data could have been generated by the simple model since the b1 that we observed is very similar to the b1s generated under the simple mode.

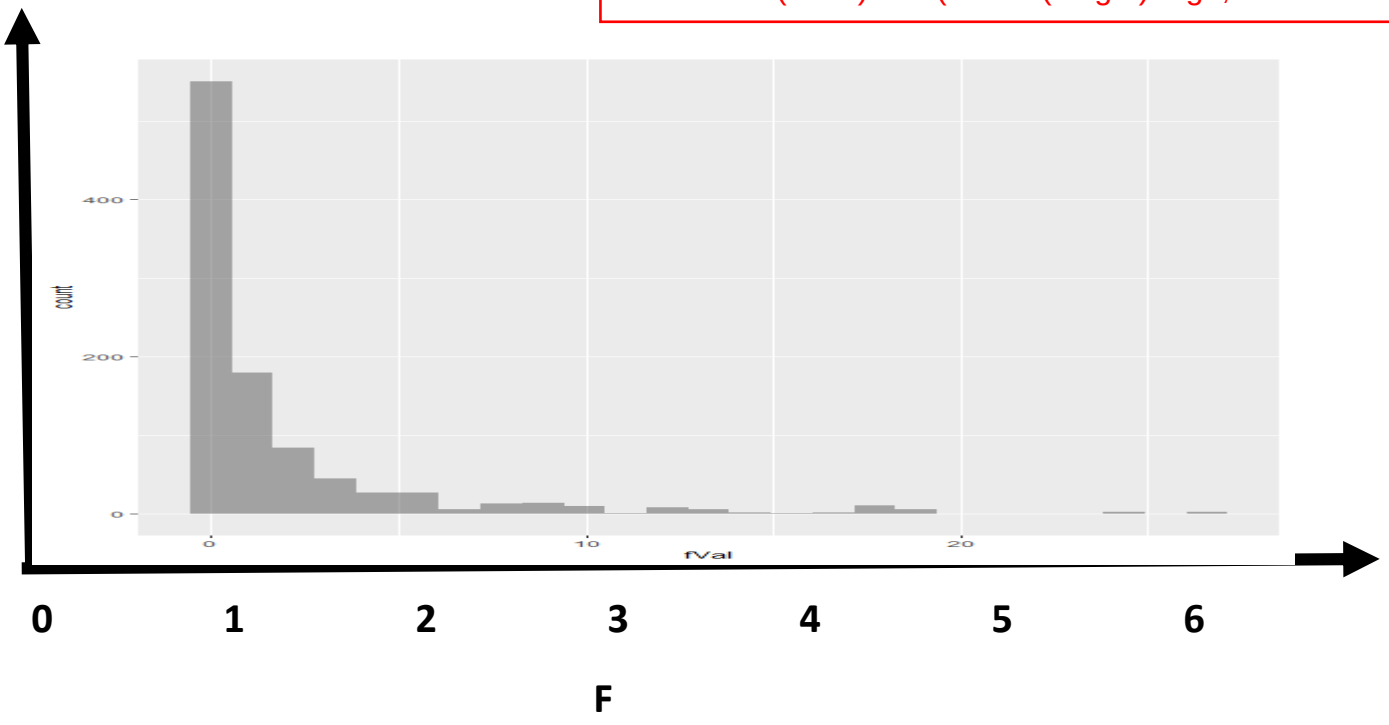
If original b1 is not contained in distribution:
If seems that our data seems unlikely under the simple model. The original b1 estimate is very far away from the generated estimates from the simple model, so it seems like our data is unlikely if the simple model is true.

15. What if we looked at the F-value instead of b_1. Below is the `supernova` output from my original data. Run `supernova` on your original data, and draw a vertical line on the graph below where the original F is. Use a `do` statement in combination with the `fVal` function to create a sampling distribution of F-values under the assumption that the simple model is true. Draw the distribution of F-values you find on the graph below.

```
> supernova(lm(height~age, data = data))
 Analysis of Variance Table (Type III SS)
 Model: height ~ age

                              SS df       MS     F    PRE     p
----- ----------------- -------- -- -------- ----- ------ -----
Model (error reduced)  | 3559.050  1 3559.050 3.738 0.4831 .1253
Error (from model)     | 3808.450  4  952.112
----- ----------------- -------- -- -------- ----- ------ -----
Total (empty model)    | 7367.500  5 1473.500
```

```
SDOF <- do(1000)*fVal(shuffle(height)~age, data = data)
```



15. Use R to calculate the probability that a shuffled sample had an F which was greater than or equal to the F observed in your original sample.

```
> tally(~(fVal >= fVal(height~age, data = data)), data = SDOF, format = "proportion")
(fVal >= fVal(height ~ age, data = data))
 TRUE FALSE
0.645 0.355
```

16. Write an interpretation of this probability (also called a p-value). Remember to interpret for a family member, not your stats teacher!

The probability of getting an explained variance as big or bigger than F = 0.181 from a model where the predictor explains no variance in the population is 64.5%. This mean the odds of getting an effect (i.e., slope) this big or bigger is pretty high even when there is no relationship between age and height.

17. How does this probability compare to the p-value from your supernova output, is it similar? Should it be?

The probability should be fairly similar to the p-value from the supernova output. They don't necessarily need to be exactly the same since the shuffle method does not use the mathematical F-distribution to calculate this probability, but the p-value in the supernova output uses the mathematical F-distribution to do this calculation.

18. People often get confused between bootstrapping and shuffling. Shuffling is a method for creating samples from a population where the simple model is true. Bootstrapping is a method for creating samples from a population which looks like our original sample. Describe at least 2 similarities and 2 differences between bootstrapping and shuffling.

Similarities:
- Both are methods for creating sampling distributions
- Both methods only rely on the data in the sample to generate the distributions

Differences:
- Shuffling assumes the simple model is true, whereas bootstrapping does not make an assumption about which model is true.
- Shuffling breaks up the X's and the Y's in the data, whereas bootstrapping keeps them together.

19. If we had bootstrapped the sampling distribution of F or b_1 do you think it would have looked similar or very different from the shuffled distributions you drew on the previous pages. Will this always be the case?

If original b1 is contained in shuffle distribution:
Bootstrapping assumes that the population looks very similar to the sample collected. In this case our estimate of b1 was pretty small (near zero) so bootstrapping and shuffling will result in similar looking distributions for the F-ratio

If original b1 is not contained in distribution:
Bootstrapping assumes that the population looks very similar to the sample collected. In this case our estimate of b1 was not near zero so bootstrapping will result in a distribution of F's which is generally higher than the distribution of F's from the shuffling approach. This is because when we bootstrap we're generating data as if there is a relationship between height and age, so the F-ratios will be larger in general.

20. We shuffled height, but we could have shuffled age. Do you think this would make a difference in what we find? Try it with age and reflect on whether the results are different.

We can shuffle either X or Y or both, ultimately any of these methods break up the relationship between X and Y (i.e., John's X doesn't necessarily go with John's Y, but rather could go with Steve's Y). The results are not notably different depending on which variable you shuffle.