

Lab 3: Relationship between global cesarean delivery rates and GDP

Your name and student ID

today's date

Instructions

- Due date: Friday, September 11 at 11:59pm PST.
- 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.

Start by loading the required libraries, reading in the data and adding on a variable:

```
library(dplyr)
library(ggplot2)
library(readr)
library(broom)
```

```
CS_data <- read_csv("cesarean.csv")
```

```
## Parsed with column specification:
## cols(
##   Country_Name = col_character(),
##   CountryCode = col_character(),
##   Births_Per_1000 = col_double(),
##   Income_Group = col_character(),
##   Region = col_character(),
##   GDP_2006 = col_double(),
##   CS_rate = col_double()
## )
```

```
# This code re-orders the variable Income_Group in the specified order.
# Note that it *does not* change the order of the data frame (like arrange() does)
# Rather, it specifies the order the data will be plotted.
# This will make more sense when we plot the data using Income_Group, and then
# again using Income_Group_order
CS_data$Income_Group <- forcats::fct_relevel(CS_data$Income_Group,
                                             "Low income", "Lower middle income",
                                             "Upper middle income", "High income: nonOECD",
                                             "High income: OECD")
```

```
CS_data <- CS_data %>% mutate(CS_rate_100 = CS_rate*100)
```

1. [1 point] Make a scatter plot between CS_rate_100 and GDP_2006:

```
p1 <- "<<<<YOUR CODE HERE>>>>"
p1
```

```
## [1] "<<<<YOUR CODE HERE>>>>"
```

```
check_problem1()
```

```
## [1] "Checkpoint 1 Error: You did not define a ggplot."
## [1] "Checkpoint 2 Error: Did you use the right dataset?"
## [1] "Checkpoint 3 Error: Did you plot the right variable?"
## [1] "Checkpoint 4 Error: Did you plot the right variable?"
## [1] "Checkpoint 5 Error: Did you define a scatterplot in ggplot?"
##
## Problem 1
## Checkpoints Passed: 0
## Checkpoints Errored: 5
## 0% passed
## -----
## Test: FAILED
```

In your plot, you might notice that many of the points are condensed towards the lower left corner. And you might recall from the lab and assignment that the distributions of both cesarean delivery rate and GDP covered a wide range of values. Both of these variables are good candidates for log transformations to spread out the range of data at the lowest levels.

2.[1 point] Using the `mutate()` function, add two new logged variables to the data set `CS_data` and assign this new data set to `CS_data_log`. Call the variables `log_CS` and `log_GDP`. Use base `e`, also known as natural logarithms, to create the logged variables:

```
CS_data_log <- "<<<<YOUR CODE HERE>>>>"
```

```
check_problem2()
```

```
## [1] "Checkpoint 1 Error: Did you name your new data set CS_data_log?"
```

```
## [1] "Checkpoint 2 Error: Did you transform CS_rate_100 correctly?"
```

```
## [1] "Checkpoint 3 Error: Did you transform GDP_2006 correctly?"
```

```
##
```

```
## Problem 2
```

```
## Checkpoints Passed: 0
```

```
## Checkpoints Errored: 3
```

```
## 0% passed
```

```
## -----
```

```
## Test: FAILED
```

3. [1 point] Remake the scatter plot using the logged variables:

```
p3 <- "<<<<YOUR CODE HERE>>>>"
p3
```

```
## [1] "<<<<YOUR CODE HERE>>>>"
```

```
check_problem3()
```

```
## [1] "Checkpoint 1 Error: You did not define a ggplot."
## [1] "Checkpoint 2 Error: Did you use the right dataset?"
## [1] "Checkpoint 3 Error: Did you plot the right variable?"
## [1] "Checkpoint 4 Error: You plotted the right variable?"
## [1] "Checkpoint 5 Error: Did you define a scatterplot in ggplot?"
##
## Problem 3
## Checkpoints Passed: 0
## Checkpoints Errored: 5
## 0% passed
## -----
## Test: FAILED
```

4. [1 point] A geom that you have not yet learnt is `geom_smooth()`. This geom can fit a curve to the data. Extend your `ggplot()` code by adding `geom_smooth()` to it:

```
p4 <- "<<<<YOUR CODE HERE>>>>"
p4
```

```
## [1] "<<<<YOUR CODE HERE>>>>"
```

```
check_problem4()
```

```
## [1] "Checkpoint 1 Error: You did not define a ggplot."
## [1] "Checkpoint 2 Error: Did you use the right dataset?"
## [1] "Checkpoint 3 Error: Did you plot the right variable?"
## [1] "Checkpoint 4 Error: Did you plot the right variable?"
## [1] "Checkpoint 5 Error: Did you define a scatterplot in ggplot?"
## [1] "Checkpoint 6 Error: Did you define a geom_smooth in ggplot?"
##
## Problem 4
## Checkpoints Passed: 0
## Checkpoints Errored: 6
## 0% passed
## -----
## Test: FAILED
```

Does the relationship between logged GDP and logged CS look linear?

[TODO: YOUR ANSWER HERE]

5. [1 point] Modify your scatter plot by linking the color of the points to the variable `Income_Group`.

```
p5 <- "<<<<YOUR CODE HERE>>>>"
p5
```

```
## [1] "<<<<YOUR CODE HERE>>>>"
```

```
check_problem5()
```

```
## [1] "Checkpoint 1 Error: You did not define a ggplot."
## [1] "Checkpoint 2 Error: Did you use the right dataset?"
## [1] "Checkpoint 3 Error: Did you plot the right variable?"
## [1] "Checkpoint 4 Error: Did you plot the right variable?"
## [1] "Checkpoint 5 Error: Did you define a scatterplot in ggplot?"
## [1] "Checkpoint 6 Error: Did you define a geom_smooth in ggplot?"
## [1] "Checkpoint 7 Error: Did you set the plot to color by Income_Group?"
##
## Problem 5
## Checkpoints Passed: 0
## Checkpoints Errored: 7
## 0% passed
## -----
## Test: FAILED
```

Based on this colored scatter plot, it looks like the relationship is linear for those countries that are not categorized as one of the two high income categories.

6. [1 point] For this lab, we would like to use linear regression. To do this, use a dplyr function to make a new data set called CS_data_sub that only contains the low-, lower-middle, and upper-middle income countries (hint: You might want to look at the data to see exactly what these levels are called in the data set):

```
CS_data_sub <- "<<<<YOUR CODE HERE>>>>"
```

```
check_problem6()
```

```
## [1] "Checkpoint 1 Error: Did you name your new data set CS_data_sub?"
## [1] "Checkpoint 2 Error: Did you filter Income_Group correctly?"
## [1] "Checkpoint 3 Error: Did you filter Lower middle income Group correctly?"
## [1] "Checkpoint 4 Error: Did you filter Upper middle income Group correctly?"
## [1] "Checkpoint 5 Error: Did you filter High income: nonOECD group correctly?"
## [1] "Checkpoint 6 Error: Did you filter High income: OECD group correctly?"
##
## Problem 6
## Checkpoints Passed: 0
## Checkpoints Errored: 6
## 0% passed
## -----
## Test: FAILED
```


7. [1 point] Remake the last scatter plot, this time using `CS_data_sub` to see if the relationship looks approximately linear between the logged variables:

```
p7 <- "<<<<YOUR CODE HERE>>>>"
p7
```

```
## [1] "<<<<YOUR CODE HERE>>>>"
```

```
check_problem7()
```

```
## [1] "Checkpoint 1 Error: You did not define a ggplot."
## [1] "Checkpoint 2 Error: Did you use the right dataset?"
## [1] "Checkpoint 3 Error: Did you plot the right variable?"
## [1] "Checkpoint 4 Error: Did you plot the right variable?"
## [1] "Checkpoint 5 Error: Did you define a scatterplot in ggplot?"
## [1] "Checkpoint 6 Error: Did you define a geom_smooth in ggplot?"
## [1] "Checkpoint 7 Error: Did you set the plot to color by Income_Group?"
##
## Problem 7
## Checkpoints Passed: 0
## Checkpoints Errored: 7
## 0% passed
## -----
## Test: FAILED
```

8. [1 point] Given that the relationship is approximately linear, use linear regression to model the relationship between `log_CS` as the response variable and `log_GDP` as the explanatory variable. Don't forget to specify the correct data set!:

```
p8 <- "<<<<YOUR CODE HERE>>>>"
p8
```

```
## [1] "<<<<YOUR CODE HERE>>>>"
```

```
check_problem8()
```

```
## [1] "Checkpoint 1 Error: Please choose the correct variable for the model."
## [1] "Checkpoint 2 Error: Please choose the correct variable for the model."
##
## Problem 8
## Checkpoints Passed: 0
## Checkpoints Errored: 2
## 0% passed
## -----
## Test: FAILED
```

9. Interpret the slope estimate:

[TODO: YOUR ANSWER HERE]

10. Estimate what the cesarean delivery rate would be for a country with a GDP of 2000. Outline the steps you take to calculate your answer and provide an interpretation. Round your final answer to one decimal place.

[TODO: YOUR ANSWER HERE]

11. Is it appropriate to use the model to predict the cesarean delivery rate for a country with a GDP of 50,000? Why or why not? Based on the relationship in the full data set, would you expect the linear model to over or under predict?

[TODO: YOUR ANSWER HERE]

Check your score

Click on the middle icon on the top right of this code chunk (with the downwards gray arrow and green bar) to run all your code in order. Then, run this chunk to check your score.

```
# Just run this chunk.  
total_score()
```

##	Test	Points_Possible	Type
## Problem 1	FAILED	1	autograded
## Problem 2	FAILED	1	autograded
## Problem 3	FAILED	1	autograded
## Problem 4	FAILED	1	autograded
## Problem 5	FAILED	1	autograded
## Problem 6	FAILED	1	autograded
## Problem 7	FAILED	1	autograded
## Problem 8	FAILED	1	autograded
## Problem 9	FAILED	0	autograded
## Problem 10	FAILED	0	autograded
## Problem 11	FAILED	0	autograded

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-fa20/lab/lab03; 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 output.
5. If the submission fails, try to diagnose the issue using the error messages—if you have problems, post on Piazza under the post "Submission 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.