Multicollinearity

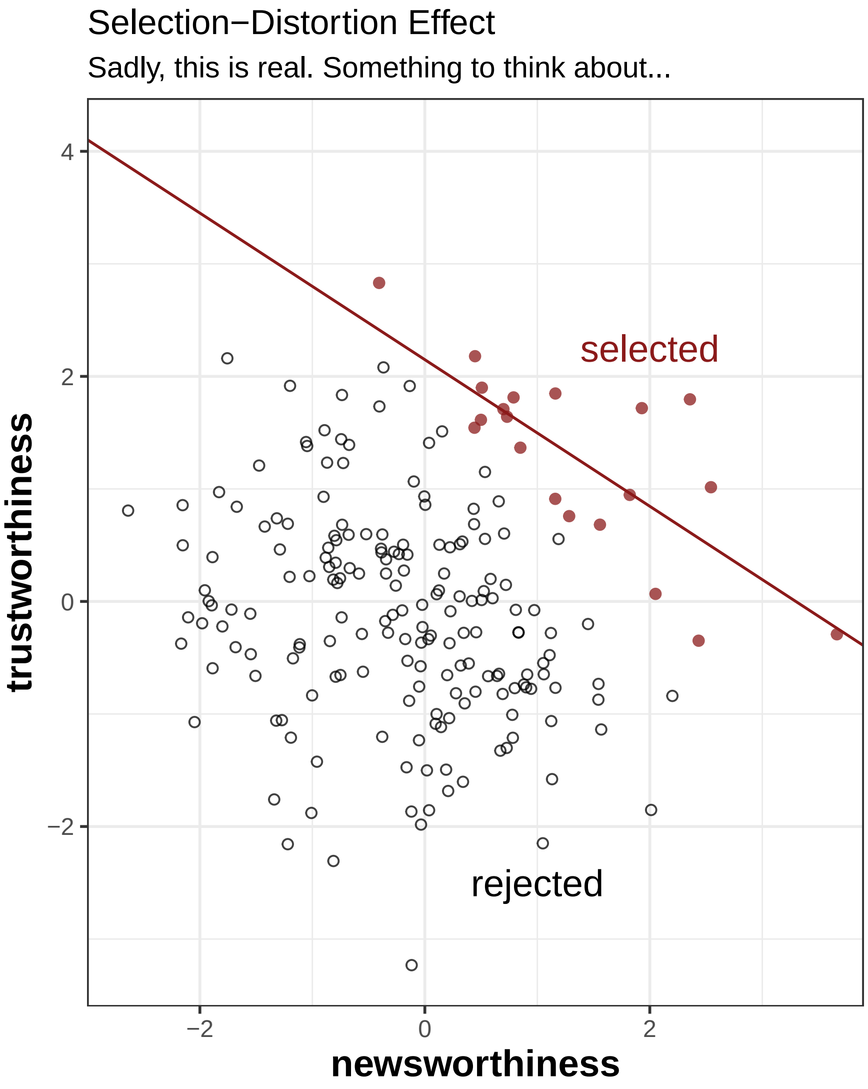
Jon Nations

1/19/23

Multicollinearity is important, so let's talk about it.

But first, I want to plot the graph from the opening of Chapter 6 on selection bias in science. As mentioned in the text, this is a real problem. I'm sure you can think of examples in your field of papers or funded projects that fall into these categories.

Selection Bias



Multicollinearity: Simulated Example

Multicollinearity is historically a big topic of conversation in applied statistics. Now that you've heard about it, you will start to see it pop up in a lot of places. Often, you will see it spoken of as this evil feature of data, the worst thing possible, and a reason why your analyses may be totally flawed!!

I think this book does a good job explaining what it is, and why it happens, and how to look out for it. It's not some evil demon lurking, rather, it's just a feature of regression. And, in this case, it results in a false negative rather than a false positive.

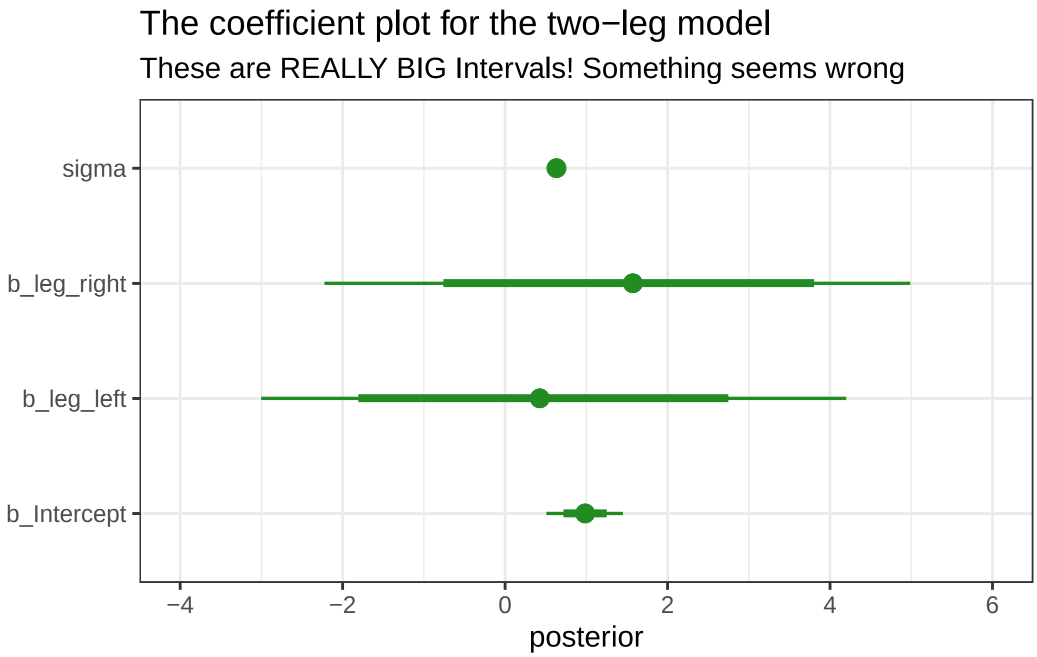
Simulate some data

```
#packages used in this example. Check out `pacman` if you haven't already!
pacman::p_load(tidyverse, brms, cmdstanr, tidybayes, patchwork)

n <- 100
set.seed(909)

d <-
  tibble(
    # height of person
    height   = rnorm(n, mean = 10, sd = 2),
    # proportion of leg to height, note tiny interval
    leg_prop = runif(n, min = 0.4, max = 0.5)) %>%
  #calculate left leg values and right leg values
  mutate(leg_left  = leg_prop * height + rnorm(n, mean = 0, sd = 0.02),
         leg_right = leg_prop * height + rnorm(n, mean = 0, sd = 0.02))

d
```

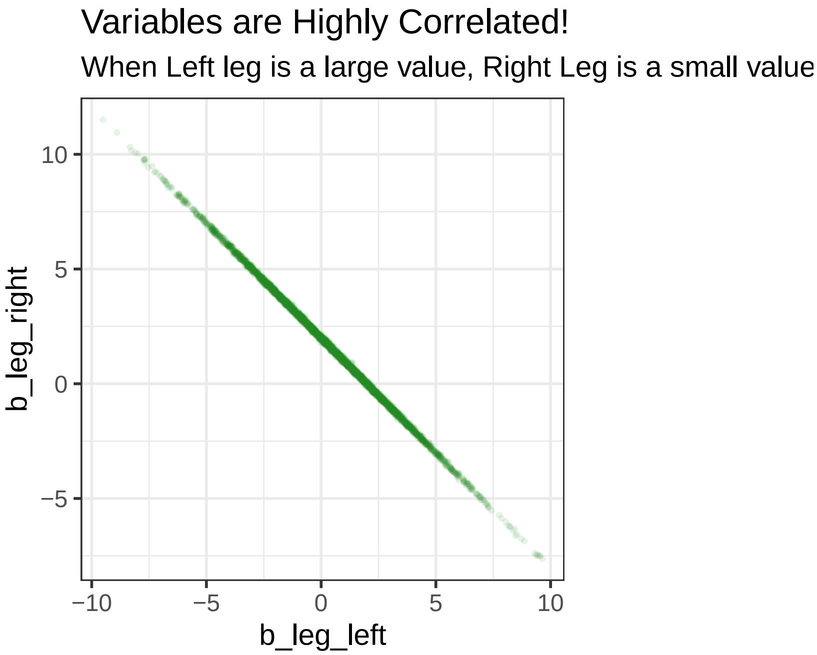
```
# A tibble: 100 x 4
   height leg_prop leg_left leg_right
    <dbl>    <dbl>    <dbl>     <dbl>
 1   5.93    0.454     2.68      2.71
 2   6.51    0.412     2.68      2.68
 3   9.35    0.422     3.93      3.98
 4   9.23    0.431     3.96      3.99
 5  10.4     0.429     4.43      4.42
 6  10.1     0.494     4.96      4.97
 7  13.1     0.416     5.40      5.40
 8   7.50    0.474     3.54      3.58
 9  13.8     0.456     6.30      6.30
10   7.33    0.444     3.23      3.27
# ... with 90 more rows
```

What happens when we use these two highly correlated variables in our analysis?

$$\begin{aligned}\text{Height}_i &\sim \text{Normal}(\mu_i, \sigma)\\ \mu_i &= \alpha + \beta_1 \text{Left-Leg}_i + \beta_2 \text{Right-Leg}_i.\end{aligned}$$



Why does this happen?



What can't be seen from this plot however is that the correlation is related by draw. When the model estimates a high value for right leg, it estimates an almost equally low value for right leg, with the mean value as the center point!

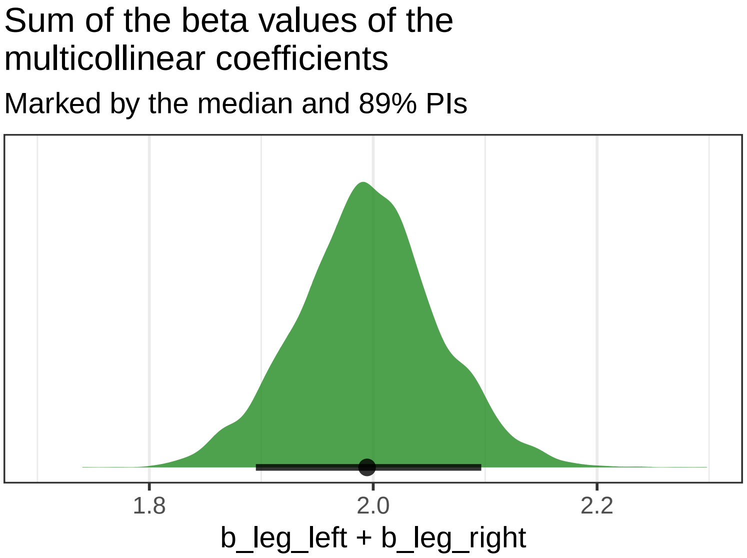
See if you can spot that here:

```
post %>%
  select(b_Intercept, b_leg_left, b_leg_right) %>%
  # calculate the mean value of the left and right leg estimates
  mutate(mean_leg = (b_leg_left + b_leg_right)/2)
```

```
# A tibble: 4,000 x 4
   b_Intercept b_leg_left b_leg_right mean_leg
         <dbl>      <dbl>       <dbl>    <dbl>
 1       1.12       1.85       0.0988    0.973
 2       1.12       1.87       0.0731    0.973
 3       1.26      -4.06       6.01      0.979
 4       0.568      3.28      -1.22      1.03
 5       1.23       0.796      1.14      0.970
 6       0.891      1.69       0.334     1.01
 7       1.05      -0.807      2.78      0.988
 8       0.712      0.914      1.13      1.02
 9       1.16       0.850      1.09      0.970
10       0.604      1.81       0.263     1.04
# ... with 3,990 more rows
```

Look at how those mean values are similar!!!

Interestingly, if we simply sum the beta values together…

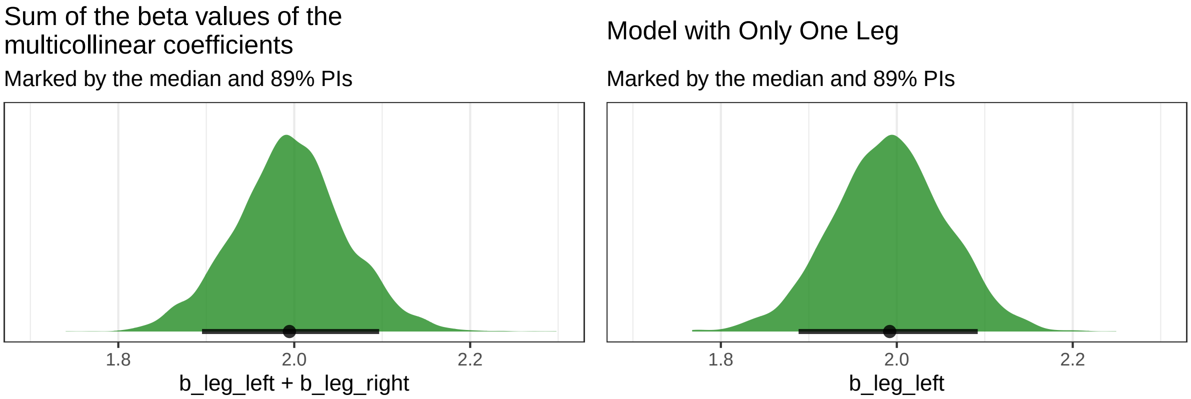

Sum of the beta values of the multicollinear coefficients

Marked by the median and 89% PIs

This is weird - What happens in a model with just one leg?

$$\begin{aligned}\text{Height}_i &\sim \text{Normal}(\mu_i, \sigma)\\ \mu_i &= \alpha + \beta_1 \text{Left-Leg}_i\end{aligned}$$

```
 #patchwork plotting - the best figure-making package ever!
p1 + p2
```



> The basic lesson is only this: When two predictor variables are very strongly correlated (conditional on other variables in the model), including both in a model may lead to confusion. The posterior distribution isn't wrong, in such cases. It's telling you that the question you asked cannot be answered with these data. And that's a great thing for a model to say, that it cannot answer your question. And if you are just interested in prediction, you'll find that this leg model makes fine predictions. It just doesn't make any claims about which leg is more important. (p. 166)

How do we deal with multicollinearity?

1. Explore your data!
2. Think about a **Scientific Model**, preferably using causal reasoning and Directed Acyclic Graphs, to see the effects of one variable on the other, so you know what to "control for."

3. There are various methods of variable selection which can identify co-linear variables. Variance Inflation Factor (VIF) is an older but still commonly used method. Lasso or horseshoe priors are another interesting way to do variable selection. These are really tight priors with really long narrow tails. The idea is that any variable with a large effect size has enough influence to end up in posterior space far away from the mean. A newer method that works with modern Bayesian Multilevel Modeling is Projection Prediction, or ProjPred, which uses cross validation and model accuracy to determine

the importance of different variables. However, these are often easy ways to bypass thoughtful consideration of variables and use "causal salad" to throw it at the wall and see what sticks. ***Nothing can take the place of a well thought out scientific model and causal reasoning that determines the causal relationships between variables!***