Bad Control

Alex

October 7, 2019

What goes wrong with bad controls? Everything!

A Very Simple Example

Let's make some data in just the same way that we typically make data.

First, let's make and `id` variable, an `epsilon` variable that represents everything in the world that we haven't measured, a `D` variable that represents the assignment to treatment, and a `tau` variable that represents each individuals response to treatment if he or she or it is exposed to the treatment.

```r
d <- data.table(
  id  = 1:1000,
  epsilon = rnorm(1000),
  D   = sample(0:1, 1000, replace = TRUE),
  tau = rnorm(n = 1000, mean = 2, sd = 1)
)
```

Now, lets make the *slightly* more complicated quantities, `x`, the bad control, and the potential outcomes. We're going to take some care to create the x-variable. As we show in the figure below, the x-variable is **caused** by both the treatment and other things in the system.

For example, consider the case of the white collar and blue collar jobs. If you're born the heir of a real-estate empire in NY (in this case represented by ϵ), you might be likely, no matter your college attendance or draft status, to end up in a white collar job (in this case represented in X). But, it may also be the case in the data that having been assigned to attend college (in this case represented in D) increases the probability that you get a white collar job.

```r
knitr::include_graphics("./coding_bad_controls_diagram.pdf")
```
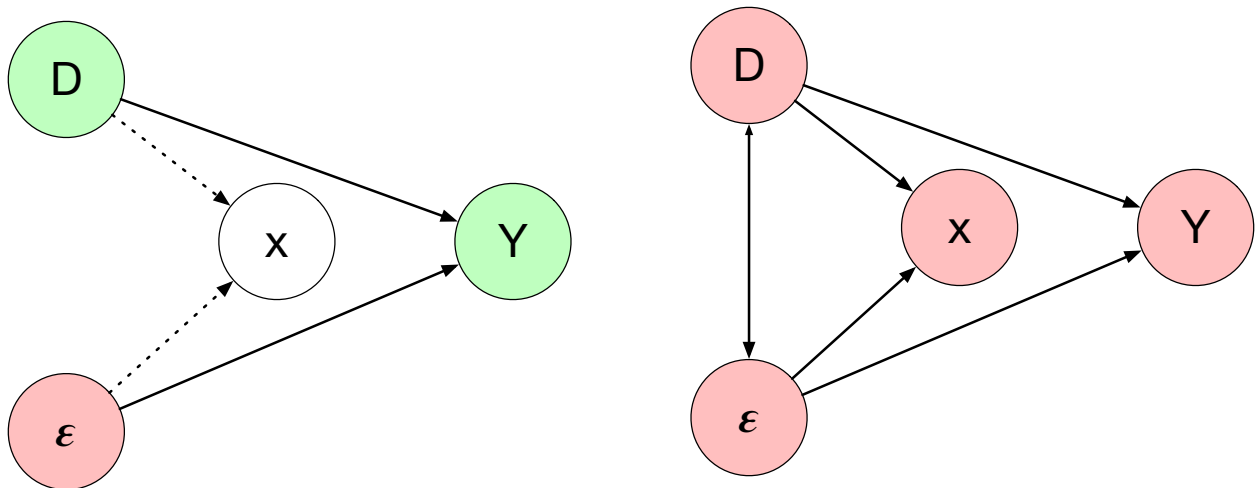

Suppose that we measure four variables, D, x, Y, ϵ that all have a positive relationship between one another, shown in the figures above. The one exception, and this is important to make this example easy-ish to understand, is that x actually has *no* relationship on Y.

The way we've built this data, if someone is in treatment ($D = 1$), then they're more likely to have a larger value for X and also a larger value for Y. Similarly, if someone has a high ϵ value, then they're more likely to have a higher value for X and also a higher value for Y.

In the figure on the left, where we've included dotted-line relationships between D, X, and ϵ, we're acknowledging that these relationships exist, but we're not going to condition on them. As a result, there is no covariance between D and ϵ, which we represent with no line. In the figure on the right, consider that we've conditioned on the X variable. As a result, there is a solid line relationship between ϵ and X; but so too is there a solid line relationship between ϵ and D **after we condition on** X.

Perhaps think of it this way – if we don't condition on X then any value of ϵ is possible, and so too is any value of D. But, after we set the X to a value by conditioning on it, some arrangements are more likely to have been the case than others.

- If we don't condition on X, and if ϵ is low, it could still be the case that the unit was in treatment or was in control. We don't have any conditional information.
- But, if we condition so that the units we're considering have a $X = 1$, and if ϵ is low, then *we can know* that the person was more likely to be in the treatment group than the control group.

Let's make these, carefully. Create the variable x, the realization of the variable X to be a binary indicator that takes the values of either 1 or 0. And, suppose that the probability of being a 1 increases if you're in treatment and also if you've got a high value of epsilon. (I'm going to scale these probabilities using a `pnorm` call, just so that the values for the `prob` argument to `rbinom` conforms to be a valid probability statement.

```
d[ , x := rbinom(n = 1000, size = 1, prob = pnorm(2*D + epsilon))]
```

Questions for Understanding

1. If we examine the relationship between x and D with a regression, should the relationship be positive, negative, or we don't know?
2. If we examine the relationship between x and ϵ with a regression, should the relationship be positive, negative, or we don't know?
3. **Challenge**: If we examine the regression of `x ~ D + epsilon` what will be the nature of each variables' relationship with x? Will they be the same, different, and will they be positive, negative, or we don't know?

```
mod_d    <- d[ , lm(x ~ D)]
mod_x    <- d[ , lm(x ~ epsilon)]
mod_both <- d[ , lm(x ~ D + epsilon)]

## I've commented this out so that you can uncomment it after you've answered the
## questions for understanding.
##
stargazer(mod_d, mod_x, mod_both, type = 'text')
```

===
Dependent variable:

-- x

(1) (2) (3)
-- D 0.458*** 0.426***

(0.026) (0.022)

epsilon 0.212*** 0.195***
(0.013) (0.011)

Constant 0.464*** 0.691*** 0.484***
(0.018) (0.013) (0.016)

Observations 1,000 1,000 1,000
R2 0.243 0.210 0.420
Adjusted R2 0.242 0.209 0.418
Residual Std. Error 0.404 (df = 998) 0.413 (df = 998) 0.354 (df = 997)
F Statistic 319.980*** (df = 1; 998) 265.201*** (df = 1; 998) 360.307*** (df = 2; 997) ========================
Note: $p<0.1$; $\boldsymbol{p<0.05}$; $p<0.01$

Make Potential Outcomes

To this point, we haven't actually built any outcomes, we've been principally concerned with building the *upstream* data for our causal system. Let's make those potential outcomes now.

For everybody, their potential outcomes to control are a combination of some individual idiosyncratic value (represented by the random draw in this code), plus their x value, plus all the other features in the world that we haven't measured, their ϵ.

```
d[ , y0 := runif(1000, min = 0, max = 10) + epsilon]
```

Question for understanding

Do we **need** in a strict sense to define the potential outcomes to treatment on this data? Or, would that information simply be a linear combination of some data that we already have in hand?

Finally, let's produce a measured outcome value for Y.

```
d[ , Y := as.numeric(NA)]
d[D==0, Y := y0]
d[D==1, Y := y0 + tau]
```

Finally, Estimate Causal Effects

In the world that we've occupied to this point, whether we *do* or *do not* condition on a variable, an experiment that we've executed that has successfully randomized the treatment should produce a reliable causal estimate. That is, in this setup, **if we estimate a model, it should provide us with an unbiased estiamte of** τ, **or 2.**.

```
unconditional_model <- d[ , lm(Y ~ D)]
```

But, what is going to happen if we use a *cough cough bad...* control variable to "clean up our estimate". After all, if controls only *increase the accuracy of our predictions*, then this will improve our model right?

```
bad_control_model <- d[ , lm(Y ~ D + x)]
```

What are other ways that we could see this relationship? Well, earlier we talked about "setting x to be equal to 1". Let's try that on this data. Among the people who have an x value that is one, what is the causal relationship between D and Y?

```
subset_model <- d[x==1, lm(Y ~ D)]
```

```
stargazer(unconditional_model, bad_control_model, subset_model,
          type = 'latex', header = FALSE, table.placement = 'h',
          add.lines = list(c("Data Subset", "All", "All", "$x==1$")))
```

```
nRows <- 1000
set.seed(2)
```

Table 1:

	(1)	(2)	(3)
	Dependent variable:		
	Y		
D	2.073***	1.515***	1.630***
	(0.203)	(0.230)	(0.258)
x		1.220***	
		(0.248)	
Constant	4.878***	4.311***	5.457***
	(0.141)	(0.181)	(0.209)
Data Subset	All	All	$x == 1$
Observations	1,000	1,000	686
R^2	0.095	0.116	0.055
Adjusted R^2	0.094	0.115	0.054
Residual Std. Error	3.204 (df = 998)	3.167 (df = 997)	3.224 (df = 684)
F Statistic	104.585*** (df = 1; 998)	65.616*** (df = 2; 997)	39.791*** (df = 1; 684)

Note: *p<0.1; **p<0.05; ***p<0.01

```
d <- data.table(id = 1:nRows)
setkeyv(d, cols = "id")
```

Create Potential Outcomes to control

```
d[ , y0  := runif(min=-10, max=10, n = .N)]
```

Create a Treatment Effect with a mean **4**. This is the value that we're looking to recover with our estimator.

```
d[ , tau := rnorm(n = .N, mean = 4)]
```

Randomly Assign Treatment

```
d[ , D   := sample(0:1, size = .N, replace = TRUE)]
```

Create a bad control that is correlated with treatment

```
d[ , bc  := 2 + 3 * D + rnorm(n = .N, mean = 0)]
```

And, finally, reate the observed outcomes.

```
d[D == 0, Y := y0 + bc]
d[D == 1, Y := y0 + bc + tau]
```

What is the causal model we hold?

When we are thinking about the causal model here, we're saying, "I think that the conditional expectation of Y depends on the treatment status". But, maybe I think I want to also control for the variable bc.

In fact, we can estimate a reliable causal effect for *either* D on Y, *or* D on bc, but not the two togheter.

First, the caual effect of D on Y in Table 2.

```
m1 <- lm(Y ~ D, data = d)
stargazer(m1, type = 'latex', label = 'goodModel', header = F)
```

Table 2:

	Dependent variable:
	Y
D	6.700***
	(0.385)
Constant	2.200***
	(0.264)
Observations	1,000
R^2	0.233
Adjusted R^2	0.232
Residual Std. Error	6.081 (df = 998)
F Statistic	302.328*** (df = 1; 998)
Note:	*p<0.1; **p<0.05; ***p<0.01

Then, the causal effect of `D` on `bc` in Table 3.

```
m2 <- lm(bc ~ D, data = d)
stargazer(m2, type = 'latex', label = 'goodModelOnControl', header = F)
```

Table 3:

	Dependent variable:
	bc
D	3.036***
	(0.065)
Constant	2.049***
	(0.044)
Observations	1,000
R^2	0.689
Adjusted R^2	0.688
Residual Std. Error	1.019 (df = 998)
F Statistic	2,209.019*** (df = 1; 998)
Note:	*p<0.1; **p<0.05; ***p<0.01

So does this. But what happens if we attempt to control for the `bc` indicator at the same time that we estimate the effect of `D` on `Y`? This isn't going to work-out, as we can see in Table 4

```
m3 <- lm(Y ~ D + bc, data = d)
stargazer(m3, type = 'latex', label = 'badModel', header = F)
```

Now we're left with an under-estiamte of the causal effect of `D` on `Y`, and we've got some estimate of the effect of `bc` on `Y`. But, when we built the data, there wasn't such an effect! In fact, what is in here is the relationship between `y0` and `Y`, but through a mangled causal pipeline.

Table 4:

	Dependent variable:
	Y
D	3.368***
	(0.679)
bc	1.098***
	(0.186)
Constant	−0.050
	(0.461)
Observations	1,000
R^2	0.258
Adjusted R^2	0.257
Residual Std. Error	5.980 (df = 997)
F Statistic	173.774*** (df = 2; 997)
Note:	*p<0.1; **p<0.05; ***p<0.01

The thing of it is, depending on the relationship between the bad control and the treatment, and the bad control and the treatment, there's *just no structured way to know* which direction the bias is going to be when you include a bad control.

```
stargazer(m1, m2, m3, type = 'latex', header = F)
```

Table 5:

	Dependent variable:		
	Y	bc	Y
	(1)	(2)	(3)
D	6.700***	3.036***	3.368***
	(0.385)	(0.065)	(0.679)
bc			1.098***
			(0.186)
Constant	2.200***	2.049***	−0.050
	(0.264)	(0.044)	(0.461)
Observations	1,000	1,000	1,000
R^2	0.233	0.689	0.258
Adjusted R^2	0.232	0.688	0.257
Residual Std. Error	6.081 (df = 998)	1.019 (df = 998)	5.980 (df = 997)
F Statistic	302.328*** (df = 1; 998)	2,209.019*** (df = 1; 998)	173.774*** (df = 2; 997)
Note:			*p<0.1; **p<0.05; ***p<0.01

I can't get this written before class Fall 2017. Let me know if you figure it out.

An example with Education

What if we build an example that is a little more complicated, but that fits with a story that we believe about the world.

Suppose that people have potential outcomes to earnings, and they are deciding whether they want to pursue a MIDS degree or not. We've been working on this one for a while.

The Dean, in her wisdom, agrees to let current students and 241 faculty run an expriment where they randomly admit people to the program. Hurrah! We can evaluate this! And we can make a causal claim.

Suppose we have the following as a setup:

- Some students live in Bay Area. Because wages are stupid here (so are mortages), these students are paid *on average* $25,000 more than people who do not live in the Bay Area.
- The *true* state of the world is that graduating with a MIDS degree leads individuals to earn $50,000 more, on average, if they graduate with a MIDS degree than if they didn't. This isn't a selection effect, but instead is actually the causal effect.
- There are some students who are *really* motivated to get that job at Pintrest. And, they think that actually have a shot at it once they're in MIDS! So, of the folks who are admitted, suppose that 25% of them move to the Bay Area. None of the people who are not admitted to MIDS move to the Bay Area.

We could make data that is consistent with this story, right? Suppose that we have 500 students, at baseline the mean earnings for people who live in SF is $75,000 and the mean earnings for people who do not live in SF is $50,000. Furthermore, at baseline, 25% of the students live in SF.

First, create baseline earnings for people.

```
d <- data.table(id = 1:500)
d[ , y0 := rnorm(.N, mean = 50000, 10000)]
d[ , lives_in_sf_before := sample(c(0,1), size = .N, replace = TRUE, prob = c(0.75, 0.25))]

d[ , y0 := y0 + lives_in_sf_before * rnorm(.N, 25000, 10000)]
```

Then, randomly assign people to receive the MIDS admission.

```
d[ , MIDS_admission := sample(c(0,1), size = .N, replace = TRUE)]
```

For the people who are admitted to MIDS and don't live in Bay Area, suppose the hardest working among them move to the Bay Area (and in doing so, they earn $25,000 more just because wages are crazy). They the folks who just *needed to catch a break!*

```
d[lives_in_sf_before == 1, lives_in_sf_after := 1]
d[lives_in_sf_before == 0 & MIDS_admission == 0, lives_in_sf_after := 0]
d[lives_in_sf_before == 0 & MIDS_admission == 1, lives_in_sf_after := sample(c(0,1), size = .N, prob = 
#d[lives_in_sf_before == 0 & MIDS_admission == 1][y0 < quantile(y0, probs = 0.25),
#  lives_in_sf_after := 1 ]
```

Did that seem to work? Yes.

```
d[ , table(lives_in_sf_before, lives_in_sf_after)]
```

```
##                   lives_in_sf_after
## lives_in_sf_before   0   1
##                  0 325  56
##                  1   0 119
```

Make the after MIDS earnings data.

```
d[MIDS_admission == 0, Y := y0]
d[MIDS_admission == 1 & lives_in_sf_before == 1, Y := y0 + rnorm(.N, 50000, 10000)]
```

```
d[MIDS_admission == 1 & lives_in_sf_before == 0 & lives_in_sf_after == 1,
  Y := y0 + rnorm(.N, 50000, 10000) + rnorm(.N, 25000, 10000)]
d[MIDS_admission == 1 & lives_in_sf_before == 0 & lives_in_sf_after == 0,
  Y := y0 +  rnorm(.N, 50000, 10000)]
```

```
mod1 <- d[ , lm(Y ~ MIDS_admission)]
mod2 <- d[ , lm(Y ~ MIDS_admission + lives_in_sf_before)]
mod3 <- d[ , lm(Y ~ MIDS_admission + lives_in_sf_after)]
```

```
stargazer(mod1, mod2, mod3, type = 'text')
```

```
##
## ==================================================================================================
##                                             Dependent variable:
##                       ----------------------------------------------------------------------------
##                                                      Y
##                              (1)                     (2)                      (3)
## -------------------------------------------------------------------------------------------------
## MIDS_admission           56,209.760***           55,515.650***            50,049.570***
##                          (1,533.294)             (1,343.733)              (1,243.839)
##
## lives_in_sf_before                               19,483.750***
##                                                  (1,577.471)
##
## lives_in_sf_after                                                         23,399.010***
##                                                                           (1,303.731)
##
## Constant                 55,760.640***           51,465.020***            50,601.810***
##                          (1,075.494)             (1,003.876)               (886.482)
##
## -------------------------------------------------------------------------------------------------
## Observations                 500                     500                      500
## R2                          0.730                   0.793                    0.836
## Adjusted R2                 0.729                   0.792                    0.835
## Residual Std. Error  17,140.560 (df = 498)    15,008.320 (df = 497)     13,364.900 (df = 497)
## F Statistic          1,343.917*** (df = 1; 498) 952.729*** (df = 2; 497) 1,266.312*** (df = 2; 497)
## ==================================================================================================
## Note:                                                        *p<0.1; **p<0.05; ***p<0.01
```

```
## The people who newly have moved to SF now also get the SF pay-bump.
## People who newly moved to SF now get the SF bump, and the MIDS bump
d[lives_in_sf_before == 0 & lives_in_sf_after == 1,
  y0 :=  y0 + rnorm(.N, 25000, 10000) + rnorm(.N, 50000, 10000)]

## For the people who didn't move, they get the treatment effect
d[lives_in_sf_before == 0 & lives_in_sf_after == 0,
  y1 := y0 + rnorm(.N, 50000, 10000)]

## And for the people who already lived in SF, they get the treatmetn effect
d[lives_in_sf_before == 1,
  y1 := y0 + rnorm(.N, 50000, 10000)]

## For everyone, if they were to be admitted to MIDS, they would realize
## a $50,000 increase
```

```
## d[ , y1 := y0 + rnorm(.N, 50000, 10000)]

## The people who newly have moved to SF now also get the SF pay-bump.
## People who newly moved to SF now get the SF bump
d[lives_in_sf_before == 0 & lives_in_sf_after == 1,
  y0 :=  y0 + rnorm(.N, 25000, 10000)]

d[ , y1 := y0 + rnorm(.N, 50000, 10000)]

## For the people who didn't move, they get the treatment effect
##d[lives_in_sf_before == 0 & lives_in_sf_after == 0,
##   y1 := y0 + rnorm(.N, 50000, 10000)]

## And for the people who already lived in SF, they get the treatmetn effect
##d[lives_in_sf_before == 1,
##   y1 := y0 + rnorm(.N, 50000, 10000)]

## For everyone, if they were to be admitted to MIDS, they would realize
## a $50,000 increase

## d[ , y1 := y0 + rnorm(.N, 50000, 10000)]
```

Then, the last thing is just to reveal these potential outcomes.

```
d[ , Y := ifelse(MIDS_admission == 1, y1, y0)]

mod1 <- d[ , lm(Y ~ MIDS_admission)]
mod2 <- d[ , lm(Y ~ MIDS_admission + lives_in_sf_before)]
mod3 <- d[ , lm(Y ~ MIDS_admission + lives_in_sf_after)]

stargazer(mod1, mod2, mod3, type = 'text')
```

===

Dependent variable:

—— Y

(1) (2) (3)

———- MIDS_admission 74,128.790***
73,894.710*** 62,540.350***
(2,899.300) (2,893.886) (2,359.770)

lives_in_sf_before 6,570.597*
(3,397.269)

lives_in_sf_after 44,017.750***
(2,473.394)

Constant 55,760.640*** 54,312.010*** 46,055.940***
(2,033.647) (2,161.965) (1,681.803)

————————————————————————————————————

Observations 500 500 500
R2 0.568 0.571 0.736
Adjusted R2 0.567 0.569 0.735
Residual Std. Error 32,411.010 (df = 498) 32,322.190 (df = 497) 25,355.420 (df = 497)
F Statistic 653.714*** (df = 1; 498) 330.526*** (df = 2; 497) 692.431*** (df = 2; 497) =========================
Note: $p<0.1$; $p<0.05$; $p<0.01$

People who didn't move to SF as a result of MIDS_admission just get the MIDS bump

d[lives_in_sf_before == 0 & lives_in_sf_after == 0, y1 := y0 + rnorm(.N, 50000, 10000)] ## People who already live in SF get the MIDS bump. d[lives_in_sf_before == 1 & MIDS_admission == 1, y1 := y0 + rnorm(.N, 50000, 10000)]

People who didn't get admitted to MIDS just show their potential outcomes to control

d[MIDS_admission == 0, Y := y0] "'

Phew! That's a lot of data to have made!

```
mod <- d[ , lm(Y ~ MIDS_admission)]
summary(mod)
```

```
##
## Call:
## lm(formula = Y ~ MIDS_admission)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -66437 -18866  -5173   9837 129470
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)        55761       2034   27.42   <2e-16 ***
## MIDS_admission     74129       2899   25.57   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32410 on 498 degrees of freedom
## Multiple R-squared:  0.5676, Adjusted R-squared:  0.5667
## F-statistic: 653.7 on 1 and 498 DF,  p-value: < 2.2e-16
```

```
mod2 <- d[ , lm(Y ~ MIDS_admission + lives_in_sf_before)]
summary(mod2)
```

```
##
## Call:
## lm(formula = Y ~ MIDS_admission + lives_in_sf_before)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -64755 -18316  -4385   8105 131153
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)           54312       2162  25.122   <2e-16 ***
## MIDS_admission        73895       2894  25.535   <2e-16 ***
## lives_in_sf_before     6571       3397   1.934   0.0537 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 32320 on 497 degrees of freedom
## Multiple R-squared:  0.5708, Adjusted R-squared:  0.5691
## F-statistic: 330.5 on 2 and 497 DF,  p-value: < 2.2e-16
```

```r
mod3 <- d[ , lm(Y ~ MIDS_admission + lives_in_sf_after)]
summary(mod3)
```

```
##
## Call:
## lm(formula = Y ~ MIDS_admission + lives_in_sf_after)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -76072 -13988  -1246  10319 106746
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)          46056       1682   27.39   <2e-16 ***
## MIDS_admission       62540       2360   26.50   <2e-16 ***
## lives_in_sf_after    44018       2473   17.80   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25360 on 497 degrees of freedom
## Multiple R-squared:  0.7359, Adjusted R-squared:  0.7348
## F-statistic: 692.4 on 2 and 497 DF,  p-value: < 2.2e-16
```