# *Problem Set 6*

Research Design for Causal Inference
Due: June 2, 2015

## Part I: Concepts behind RDD and IVE

**Question 1:** In the context of regression discontinuity designs, what is a forcing variable and how can you use one to identify a causal effect? Provide an example.

**Question 2:** Still thinking about RDDs, explain the concept of "bandwidth." How does using a larger bandwidth impact your estimate of treatment effects?

**Question 3:** Describe at least two strategies for checking the validity and robustness of RD estimates of causal effects. For each strategy, be sure to explain the potential threat that it addresses as well as the basic mechanics of how to perform the check.

**Question 4:** What are the properties of a good instrument (for performing valid instrumental variables estimation)? Feel free to use words, pictures, equations, diagrams, etc. in your response.

**Question 5:** Explain the "no third path" restriction. What is it? Why does it matter? How do you test for a third path? What is an example of a study we read about or discussed in which this exclusion rule might have been violated?

**Question 6:** Why is an IVE always LATE? Explain what this means and draw on at least one example to illustrate your explanation.

## Part II: RDD Analysis

For this part of the Problem Set, you'll analyze a subset of the data from an unpublished manuscript that I am preparing together with Benjamin Mako Hill.[1]

In the study, we analyze the effect of requiring contributors to online communities to adopt "cheap pseudonyms" (disposable usernames you might create on a website). We are interested in understanding the impact of requiring cheap pseudonyms on the quality of contributions coming into the wiki.

We take advantage a series of quasi-experiments that affected 137 wikis hosted by Wikia. All of these wikis underwent a policy change: from one day to the next, anyone who wanted to edit these wikis was suddenly required to login with a username. This shift was abrupt and (for the vast majority of those involved) both unannounced and unanticipated, making it the sort exogenous shock well-suited to quasi-experimental inference. The fact that the change was implemented in software means that end-users and potential participants in the wikis had no way of avoiding exposure to the "treatment" (in this case, the requirement of using a cheap pseudonym). We estimate the effect of this change within and across all of these wikis by comparing edits before and after the shock.

The overarching goals here are for you to reproduce a slightly-simplified part of our analysis by modeling the effect of the change on two measures of quality using an RD analysis. I also ask you to perform a pseudo-cutoff test to assess the robustness of the findings.

The dataset is available as an RData file at: `http://aaronshaw/teaching/2015/causal/data/ps6.RData`. Each row of the dataset corresponds to a "wiki-week" of observations, that is one week for one wiki. This is the unit of analysis. For almost all of the wikis, there are 24 weeks of data. The variables included in the dataset are described in Table . The DVs are `edits.reverted` (a count of bad edits) and `edits.non.reverted` (a count of good edits). Note that when you load the dataset you will also import a function called `reduced.summary()`. This will come in handy later.

**Question 1:** Summarize (provide descriptive statistics about) all of the substantively meaningful wiki-level independent variables at the week immediately before the software change (week -1) and the week immediately after (week 0).

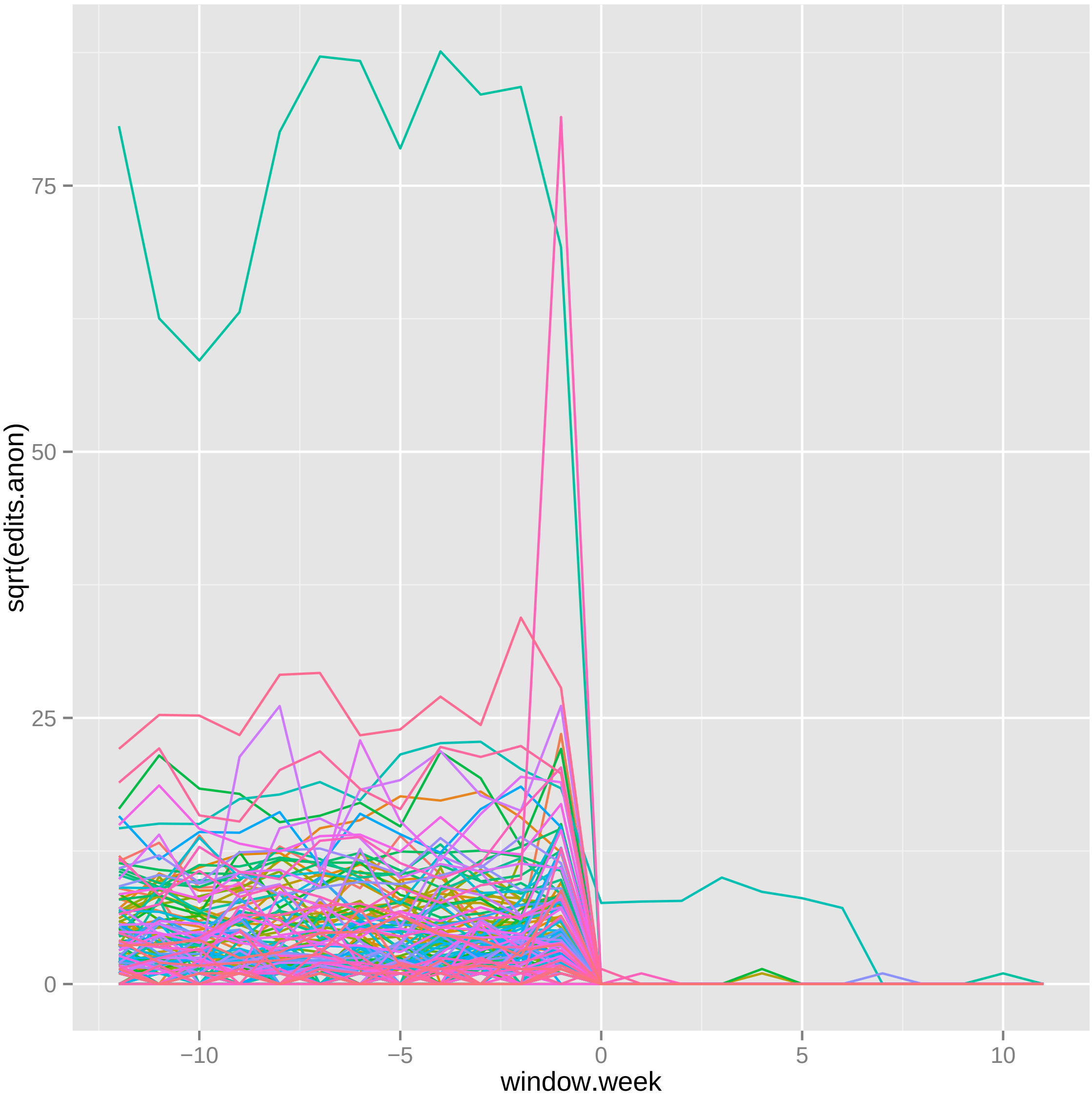
```r
# go get that dataset from wherever you stored it:
load('/tmp/ps6.RData')

# Describing covariates:
covariates <- c("age.weeks", "total.pages", "total.editors",
                "edits.anon")
```

[1] `http://mako.cc`

| Variable name | Definition |
|---|---|
| `wiki` | The name/url of the wiki. |
| `window.week` | Week relative to the cutoff. |
| `blocked` | Indicator for whether unregistered contributions were blocked. |
| `age.weeks` | The age (in weeks) of the wiki-week. |
| `total.pages` | The total (cumulative) # of pages in the wiki-week. |
| `total.editors` | The total (cumulative) # of editors in the wiki-week. |
| `edits.anon` | The # of edits from unregistered contributors in the wiki-week. |
| `edits.reverted` | The # of edits from that wiki-week which were deleted |
| `edits.non.reverted` | The # of edits from that wiki-week which were not deleted. |

Table 1: Variables in Shaw & Hill's "cheap pseudonyms" dataset

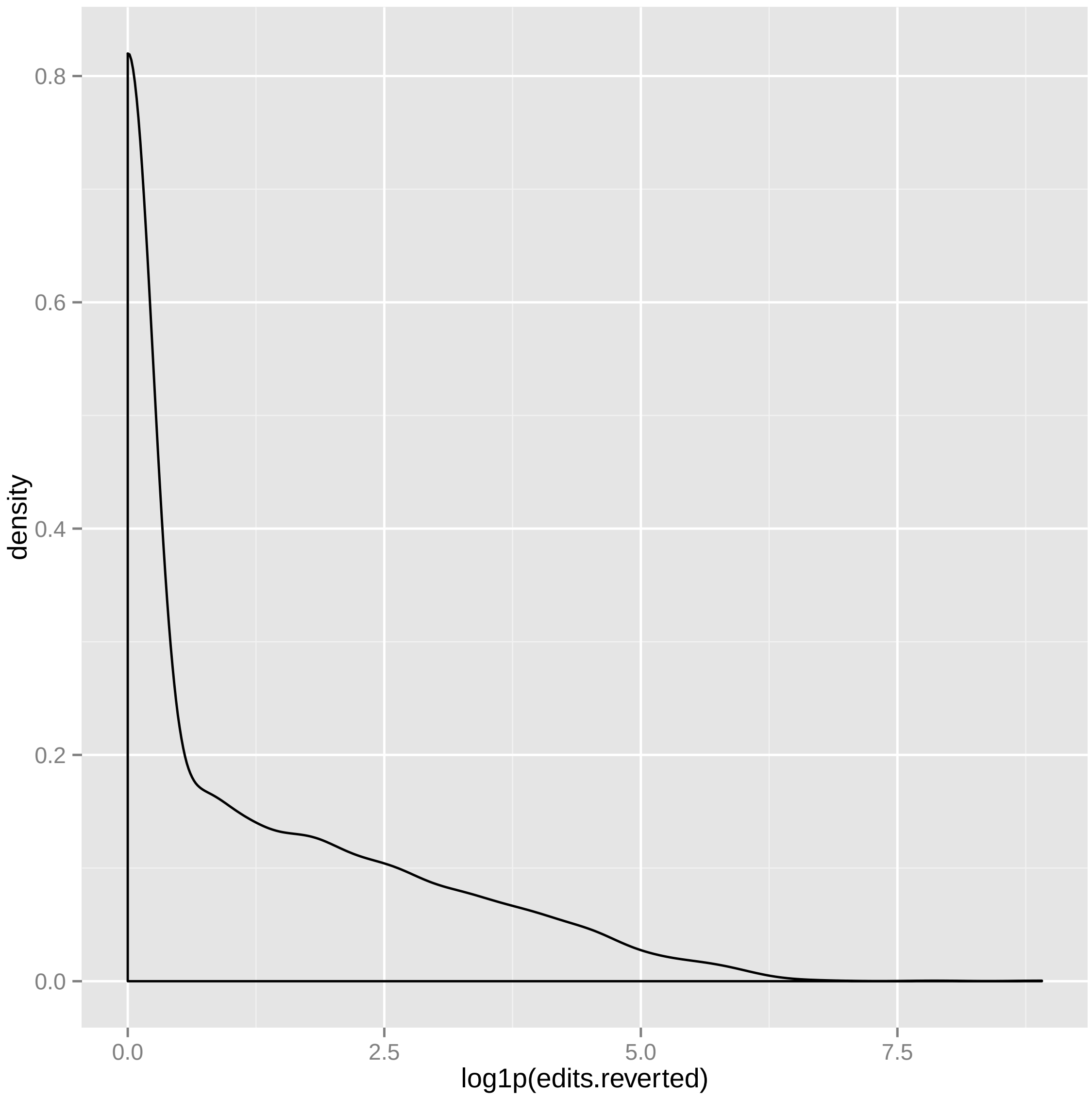
```
# I recently learned that R's builtin summary() function has some
# "issues" with rounding, so here's a quick/dirty replacement:
my.summary <- function(x){
    r <- NULL
    r[["Min"]] <- round(min(x),0)
    r[["Median"]] <- round(median(x),0)
    r[["Mean"]] <- round(mean(x), 2) # Don't need rounded means...
    r[["Max"]] <- round(max(x), 0)
    r
}

# Summaries across the weeks immediate pre/post treatment:
lapply(d[d$window.week == -1, covariates], my.summary)

## $age.weeks
##    Min Median   Mean    Max
##   3.00 140.00 145.85 437.00
##
## $total.pages
##       Min    Median      Mean       Max
##     64.00   2306.00  14064.82 747237.00
##
## $total.editors
##      Min   Median     Mean      Max
##     8.00   108.00   668.23 42669.00
##
## $edits.anon
##     Min  Median    Mean     Max
##    0.00   19.00  150.87 6634.00

lapply(d[d$window.week == 0, covariates], my.summary)

## $age.weeks
##    Min Median   Mean    Max
##   4.00 141.00 146.85 438.00
##
## $total.pages
##       Min    Median      Mean       Max
##    148.00   2354.00  14252.66 754883.00
##
## $total.editors
##      Min   Median     Mean      Max
##     9.00   111.00   677.09 42905.00
##
## $edits.anon
##    Min Median   Mean    Max
##   0.00   0.00   0.44  58.00

# Also, I should have asked you to summarize the forcing variable and
# to plot it!

# Summaries across the weeks immediate pre/post treatment:
lapply(d[d$window.week == -1, covariates], my.summary)

## $age.weeks
##    Min Median   Mean    Max
##   3.00 140.00 145.85 437.00
##
## $total.pages
##       Min    Median      Mean       Max
##     64.00   2306.00  14064.82 747237.00
##
## $total.editors
##      Min   Median     Mean      Max
##     8.00   108.00   668.23 42669.00
##
## $edits.anon
##     Min  Median    Mean     Max
##    0.00   19.00  150.87 6634.00

lapply(d[d$window.week == 0, covariates], my.summary) ## Hmm...

## $age.weeks
##    Min Median   Mean    Max
##   4.00 141.00 146.85 438.00
##
## $total.pages
##       Min     Median       Mean        Max
##    148.00   2354.00  14252.66 754883.00
##
## $total.editors
##      Min   Median      Mean      Max
##     9.00   111.00   677.09 42905.00
##
## $edits.anon
##    Min Median   Mean    Max
##   0.00   0.00   0.44  58.00

library(ggplot2)
ggplot(aes(y=sqrt(edits.anon), x=window.week, colour=wiki), data=d) +
    geom_line(show_guide=FALSE) + scale_colour_hue("clarity")
```



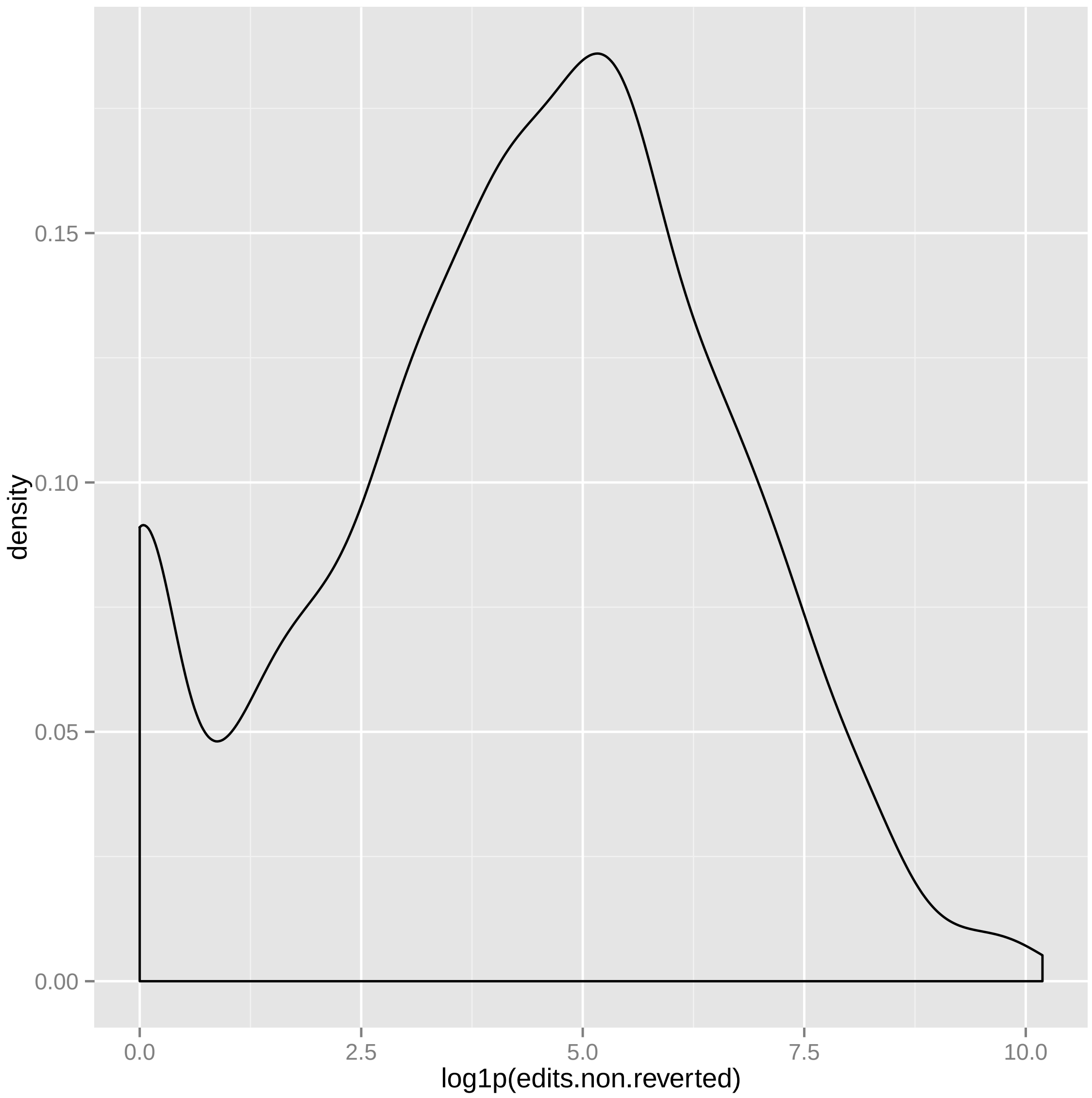
```
# Is this potentially troubling? Why yes? Why no? Think about it
# and we'll discuss in class.
```

**Question 2:** Summarize both of the dependent variables across the entire dataset for all wikis. Summarize these same variables at the week immediately before the software change (week -1) and the week immediately after (week 0). Provide a visual comparison of these pre-/post- weeks.

```
dvs <- c("edits.reverted", "edits.non.reverted")

# Whole dataset
lapply(d[, dvs], my.summary)

## $edits.reverted
##     Min  Median    Mean     Max
##    0.00    0.00   16.71 7389.00
##
## $edits.non.reverted
##      Min   Median     Mean      Max
##     0.00    90.00   584.91 26596.00

# Seems skewed, let's plot them with a log transformation:
qplot(log1p(edits.reverted), data=d, geom="density")
```

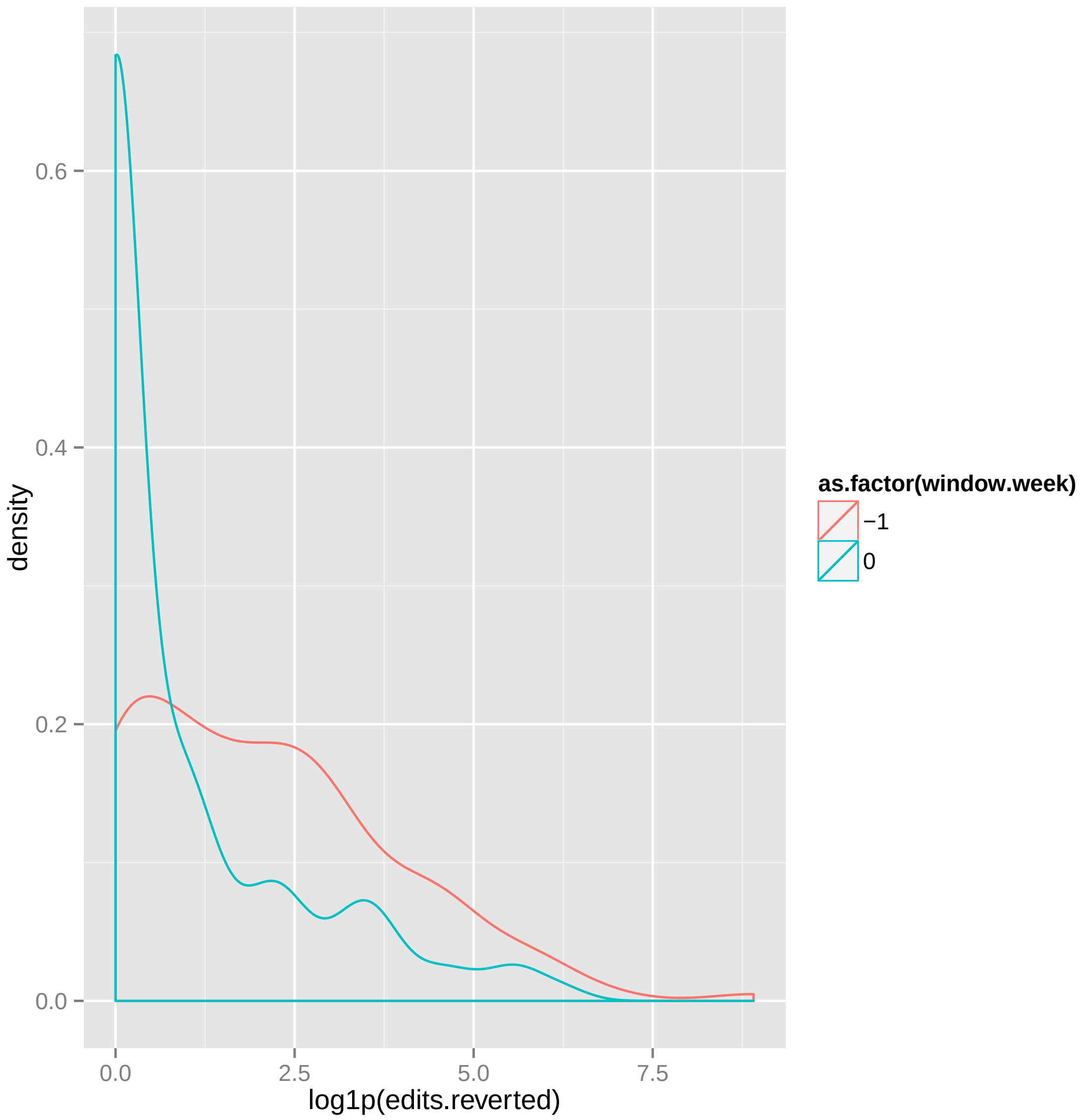


```
qplot(log1p(edits.non.reverted), data=d, geom="density")
```

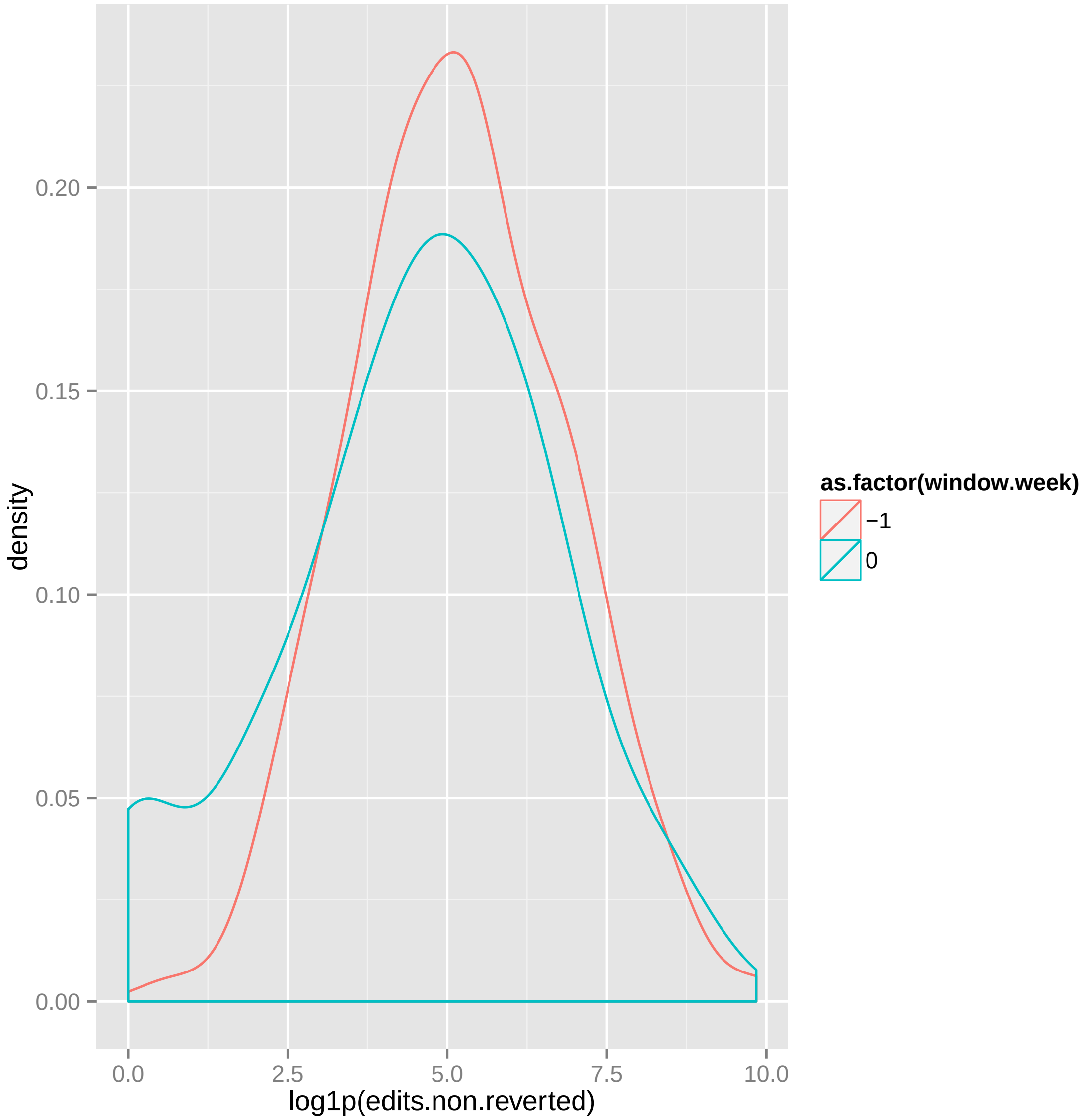


```
# Just pre/post weeks:
lapply(d[d$window.week == -1, dvs], my.summary)

## $edits.reverted
##     Min  Median    Mean     Max
##    0.00    5.00   91.38 7389.00
##
## $edits.non.reverted
##     Min  Median    Mean     Max
##     1.0   172.0   654.5 18784.0
```

```
lapply(d[d$window.week == 0, dvs], my.summary)

## $edits.reverted
##    Min Median   Mean    Max
##   0.00   0.00  15.82 515.00
##
## $edits.non.reverted
##      Min   Median     Mean      Max
##     0.00   113.00   576.95 12793.00

# Visual comparison of pre/post weeks:
# Note that because the data is super skewed, I'm going to
# log-transform everything. The "log1p()" function adds 1 and then
# takes the natural logarithm (why? try taking the natural logarithm of
# zero).

qplot(log1p(edits.reverted), geom="density",
      colour=as.factor(window.week), data=d[d$window.week >=-1 &
                                        d$window.week <= 0,])
```



```
qplot(log1p(edits.non.reverted), geom="density",
      colour=as.factor(window.week), data=d[d$window.week >=-1 &
                                           d$window.week <= 0,])
```



**Question 3:** For both dependent variables, calculate a naive estimate of the ATE across all wikis by comparing the average in the week immediately before the block (week -1) and the week immediately after (week 0). Perform a statistical test (of your choosing) to determine whether this difference seems different than what you might expect by chance.

```
# Note that I use the log values again because the variables are super skewed:
t.test(log1p(d$edits.reverted[d$window.week == -1]),
       log1p(d$edits.reverted[d$window.week == 0]))


##
```

```
##  Welch Two Sample t-test
##
## data:  log1p(d$edits.reverted[d$window.week == -1]) and log1p(d$edits.reverted[d$wind
## t = 5.6099, df = 262.637, p-value = 5.128e-08
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.7302706 1.5201528
## sample estimates:
## mean of x mean of y
## 2.0730550 0.9478433

t.test(log1p(d$edits.non.reverted[d$window.week == -1]),
       log1p(d$edits.non.reverted[d$window.week == 0]))

##
##  Welch Two Sample t-test
##
## data:  log1p(d$edits.non.reverted[d$window.week == -1]) and log1p(d$edits.non.reverte
## t = 2.6964, df = 252.061, p-value = 0.007482
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.168395 1.080813
## sample estimates:
## mean of x mean of y
##  5.166561  4.541957
```

**Question 4:** Summarize your findings from Questions 1-3. Based on what you know about the study, are your naive estimates (from Question 3) biased? Why might you find them credible or not?

> The summary statistics illustrate that the wikis vary quite widely in terms of their age, editor population size, and the number of pages they contain.
>
> Most important for the sake of this analysis, the communities indeed experience a shock to the number of unregistered edits before and after the cutoff. Interestingly (see the colorful first plot above), it looks like one wiki did not experience a complete shut-off of unregistered contributions. This is interesting from the point of view of understanding the specific character of the treatment.
>
> In terms of the dependent variables, both are heavily skewed distributions with quite a lot of variance. The cutoff appears to have induced a sudden decline in

both the number of reverted (bad) edits as well as the number of non-reverted (good) edits incoming per week. The difference is supported by the naive estimates and t-statistics calculated in Question 3. While the naive estimates are not biased, they may not provide a precise estimate of treatment effects because of the existence of underyling "secular trends" in the overall trajectory of the dependent variables (bad and good contributions) along the forcing variable (time).

**Question 5:** Explain in words how you might model the effect of the software change using a regression discontinuity approach.

Because the cutoff is an exogenous shock that occurred at a specific location along the forcing variable, I can use regression to capture the causal effect of the change. I will construct two models that regress the dependent variables on the cutoff as well as several controls variables. In a generic form, both of my models will look like the following:

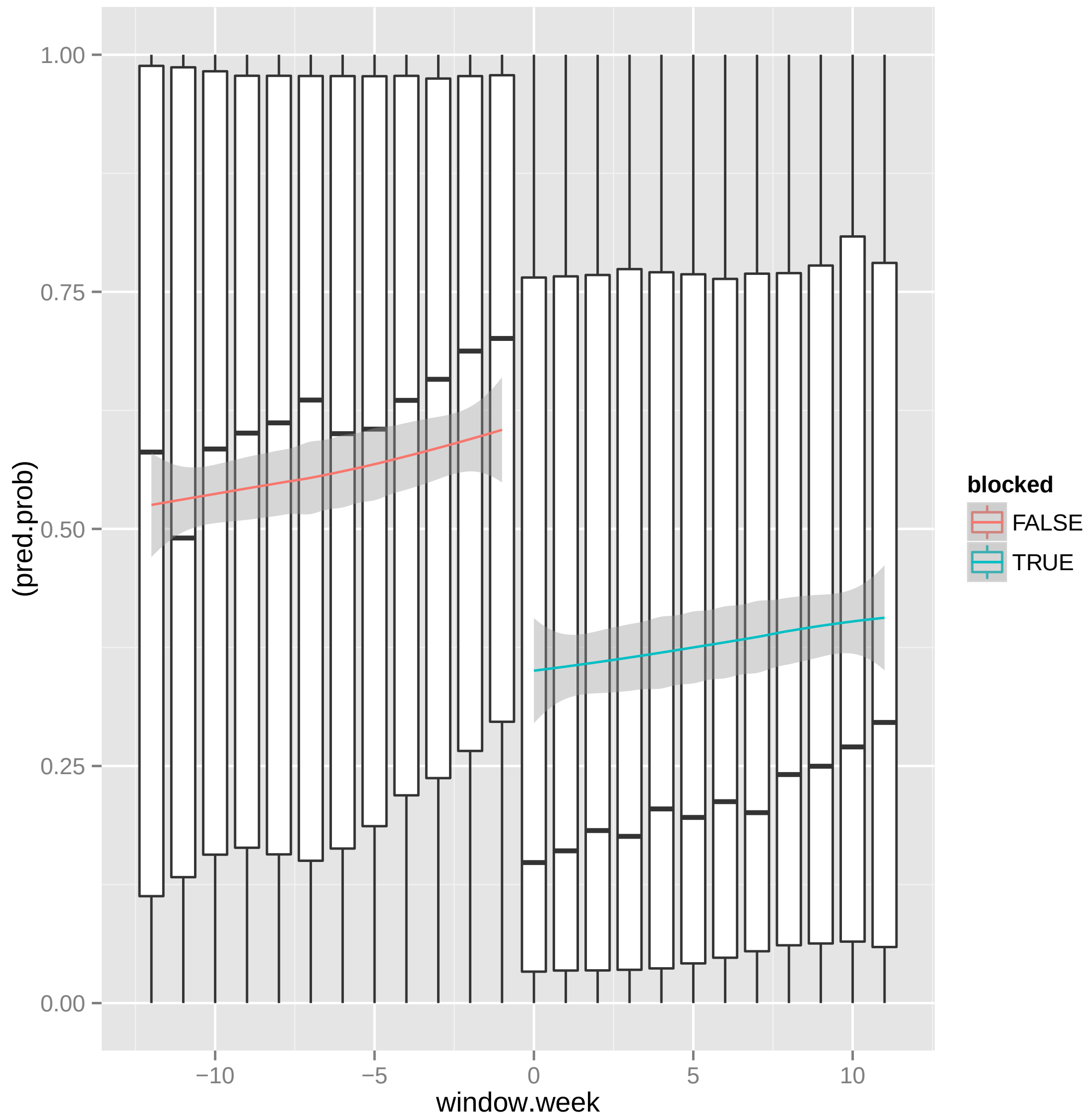
$$Y = Block + Window.week + Wiki + Window.week * Wiki + \epsilon \quad (1)$$

The controls are $Window.week$, $Wiki$, and the interaction terms between the two. Note that these controls are intended to capture for any underlying relationships between variation in the dependent variables and (1) the week relative to the cutoff, as well as (2) any wiki-specific factors. These are often called "fixed effects," a term that summarizes the idea I have no substantive hypotheses about these relationships and indeed treat them as just capturing the noise in my dataset that would otherwise obscure the underlying estimate of treatment effects. *I will not interpret the coefficients for the fixed effects at all and will not even report them.* The coefficient on $Block$ will provide an estimate of the treatment effects — that's the one we care about!

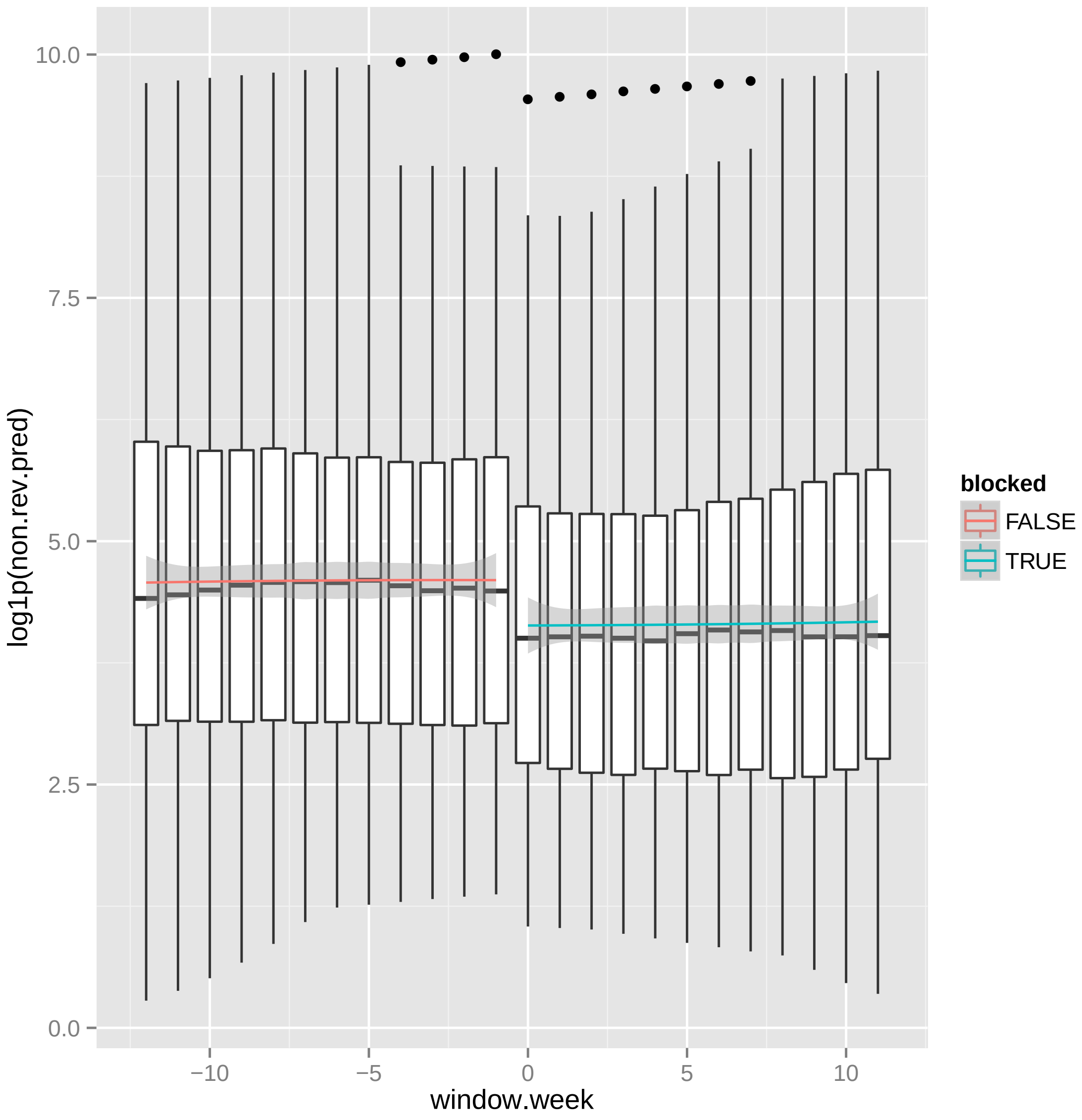
**Question 6:** Use an RD approach to estimate the Average Treatment Effect for both of the dependent variables (use any kind of regression model you like, but if you're not sure what to go with, I'd recommend good old OLS). In addition to whatever variables you need to model the effect of the block, I recommend you include only the covariates `window.week` and `wiki` as well as their interaction term, `window.week*wiki`. Let's call these "fixed effects" — they control for any variations related to week-specific and wiki-specific factors respectively (I'll explain more in class next week). Note that when you include fixed effects you don't really need to interpret or even report any of the resulting coefficients from your model. The `reduced.summary()` function that Mako and I wrote helps you do this by only displaying the coefficients for the variables you care about in this dataset.

```
# Given the super-skewed DVs, let's start by log-transforming the
# dependent variables:
d$l.edits.reverted <- log1p(d$edits.reverted)
d$l.edits.non.reverted <- log1p(d$edits.non.reverted)

# I'll just use OLS (a linear model)
# Here are formulas I can use for each model:
f.reverted <-  l.edits.reverted ~ blocked + window.week + wiki +
    (window.week*wiki)
f.non.rev <- update.formula(f.reverted, l.edits.non.reverted ~ .)

# First, the model of incoming bad edits:
m.reverted <- lm(f.reverted, data=d)
# remember that reduced.summary() function I mentioned? Here's where
# you'll want it:
reduced.summary(m.reverted)

##                Estimate Std. Error    t value      Pr(>|t|) Lower.95
## (Intercept)  3.60356099 0.15854296  22.729240 1.279665e-105     3.29
## blockedTRUE -0.78010982 0.05342260 -14.602618  1.058821e-46    -0.88
## window.week -0.08915404 0.02272253  -3.923597  8.920233e-05    -0.13
##             Upper.95
## (Intercept)     3.91
## blockedTRUE    -0.68
## window.week    -0.04

# That's fine, but note that because of how skewed the edits.reverted
# variable is, I think it's preferable to run this as a # logistic
# model (we can discuss why in class), so here's that result too:
d$any.edits.reverted <- d$edits.reverted > 0
table(d$any.edits.reverted)

##
## FALSE  TRUE
##  1732  1530

f.logit.reverted <- update.formula(f.reverted, any.edits.reverted ~ .)
m.logit.reverted <- glm(f.logit.reverted, data=d,
                        family=binomial(link=logit))

## Warning:  glm.fit:  fitted probabilities numerically 0 or 1 occurred
```

```
# Also, we should be using "cluster robust" standard
# errors (More details in class). Here’s how to do that (link:
# http://stats.stackexchange.com/questions/117052/replicating-statas-robust-option-in-r)

library(sandwich)
library(lmtest)

## Loading required package:  zoo
##
## Attaching package:  ’zoo’
##
## The following objects are masked from ’package:base’:
##
##     as.Date, as.Date.numeric

m.logit.reverted.adjusted <- coeftest(m.logit.reverted, vcov =
                                        vcovHC(m.logit.reverted, "HC1"))
reduced.summary(m.logit.reverted.adjusted) # The results:

##               Estimate Std. Error   z value     Pr(>|z|) Lower.95 Upper.95
## (Intercept)  3.518350 0.79067556  4.449803 8.594922e-06     1.97     5.07
## blockedTRUE -2.635111 0.28282420 -9.317134 1.195254e-20    -3.19    -2.08
## window.week  0.115834 0.07126422  1.625416 1.040740e-01    -0.02     0.26

# Onwards to the model of good incoming edits. Here’s the linear
# model:
m.non.rev <- lm(f.non.rev, data=d)
reduced.summary(m.non.rev)

##               Estimate Std. Error   t value      Pr(>|t|) Lower.95
## (Intercept)  5.7074039 0.21853397 26.116781 1.335242e-135     5.28
## blockedTRUE -0.4896162 0.07363716 -6.649037  3.494279e-11    -0.63
## window.week -0.1525994 0.03132049 -4.872191  1.161203e-06    -0.21
##             Upper.95
## (Intercept)     6.14
## blockedTRUE    -0.35
## window.week    -0.09

# This is fine. In reality, I think a negative binomial model is more
# appropriate, but it takes waaay to long to converge on a laptop, so
# I’ll skip it here.
# For presenting results, I really like to use marginal effects (a.k.a.
# predicted probabilities). This involves simulating predicted values
# for "typical" individuals in your model.

fake.data <- data.frame(blocked=d$blocked,
                        window.week=d$window.week,
                        wiki=d$wiki)

fake.data.rev <- cbind(fake.data, predict(m.reverted, type="response", se.fit=TRUE))


fake.data.rev <- cbind(fake.data, predict(m.logit.reverted, type="link", se=TRUE))

fake.data.rev <- within(fake.data.rev, {
    pred.prob <- plogis(fit)
    l95 <- exp(fit - 1.96 * se.fit)
    u95 <- exp(fit + 1.96 * se.fit)
})

# Visualize it:
ggplot(data=fake.data.rev, aes(x=window.week, y=(pred.prob),
           colour=blocked, group=window.week)) +
    geom_boxplot(aes(colour=NULL)) + geom_smooth(method="loess",
                         aes(group=NULL))
```



```
# Here's the ATE in those terms
mean(fake.data.rev$pred.prob[fake.data.rev$window.week==0])-mean(fake.data.rev$pred.prob

## [1] -0.2536707

# And now for the other DV:
fake.data.non.rev <- cbind(fake.data, predict(m.non.rev, type="response", se.fit=TRUE))

fake.data.non.rev <- within(fake.data.non.rev, {
    non.rev.pred <- exp(fit)
    l95 <- exp(fit - 1.96 * se.fit)
    u95 <- exp(fit + 1.96 * se.fit)
})

# Visualize it again:
ggplot(data=fake.data.non.rev, aes(x=window.week, y=log1p(non.rev.pred),
           colour=blocked, group=window.week)) +
    geom_boxplot(aes(colour=NULL)) + geom_smooth(method="loess",
                         aes(group=NULL))
```



```
# Here's an estimate of the ATE:
median(fake.data.non.rev$non.rev.pred[fake.data.non.rev$window.week==0])-
median(fake.data.non.rev$non.rev.pred[fake.data.non.rev$window.week==-1])

## [1] -34.19148
```

**Question 7:** Interpret your findings from Question 6. What is the effect of requiring cheap pseudonyms of the number of reverted (bad) edits coming into these wikis? What is the effect on the number of non-reverted (good) edits? What sorts of conclusions should Mako and I draw? What limitations do you see based on this analysis?

> Let's start with the effect of the change on the "bad" edits: Both the linear model and the logit model with cluster-robust standard errors suggest a sizeable and significant decline in incoming edits that get reverted. Interpreting coefficients when dependent variables are log-transformed or in logit models is a bit crazy, so I generally just don't do it.[2] The predicted values provide a more intuitive interpretation: On average, blocking unregistered edits led to a 25% decline in the probability of incoming edits being reverted.
>
> For the "good" edits, the story is similar, but the effect size is much smaller. Here, we only have the linear model to work with. The model results suggest a significant decline in the number of incoming good edits. Calculating marginal effects in the same way as we did with the earlier models suggests an estimated decline of about 34 fewer non-reverted edits in the week immediately following the cutoff (out of an average of about 90 non-reverted edits per week).
>
> On the basis of these results, I plan to conclude that imposing a software-based requirement of cheap pseudonyms causes a decrease in the number of both bad and good edits in these wikis. This suggests that the wikis reduce incoming vandalism at the cost of also losing a substantial number of high quality contributions. There are lots of limitations, but I'll be curious to see what you observe on the basis of your analysis!

**Question 8:** Test whether the finding is valid using a "pseudo-cutoff." Our options are somewhat constrained by the amount of data I've given you here, but you can create a `fake.blocked` variable and move the block out to a different week than the actual one in which it occurred. Then run the same analysis you did in Question 6 substituting `fake.blocked` for `blocked` in the model. Report and interpret the results of this validity check in light of your findings from Question 6.

[2] See: http://www.ats.ucla.edu/stat/mult_pkg/faq/general/log_transformed_regression.htm

```
d$fake.blocked <- d$window.week > -3

f.logit.reverted.fake <- any.edits.reverted ~ fake.blocked + window.week + wiki + (windo

m.logit.reverted.fake <- glm(f.logit.reverted.fake,
                             data=d[d$window.week < 5,],
                        family=binomial(link=logit))

## Warning:  glm.fit:  fitted probabilities numerically 0 or 1 occurred

reduced.summary(m.logit.reverted.fake)

##                      Estimate Std. Error    z value   Pr(>|z|) Lower.95
## (Intercept)       1.4767750  0.8500787  1.7372216 0.08234806    -0.19
## fake.blockedTRUE  0.2286159  0.2583324  0.8849680 0.37617389    -0.28
## window.week      -0.1471610  0.1718277 -0.8564451 0.39175163    -0.48
##                  Upper.95
## (Intercept)          3.14
## fake.blockedTRUE     0.73
## window.week          0.19

# And for the non.reverted edits:
f.non.rev.fake <- update.formula(f.non.rev, ~ . -blocked +
                                 fake.blocked)
m.non.rev.fake <- lm(f.non.rev.fake, data=d[d$window.week <5,])
reduced.summary(m.non.rev.fake)

##                      Estimate Std. Error   t value     Pr(>|t|) Lower.95
## (Intercept)       5.1463866 0.32665700 15.754711 7.607493e-53     4.51
## window.week      -0.2261599 0.05118419 -4.418551 1.046051e-05    -0.33
## fake.blockedTRUE  0.1981508 0.08330332  2.378667 1.746748e-02     0.03
##                  Upper.95
## (Intercept)          5.79
## window.week         -0.13
## fake.blockedTRUE     0.36
```

With the reverted ("bad") edits, the pseudo-cutoff analysis suggests no significant relationship between a fake discontinuity and the outcome. This lends some further credibility to our estimates and the validity of our discontinuity design.

Withe the non-reverted ("good") edits, the pseudo-cutoff analysis suggests a smaller and only marginally significant *positive* shock. This may reflect a variety of factors, but it does call into question the validity of our findings with the true cutoff.