Problem Set #4

Experiments and Causality

```
# load packages
library(foreign)
library(data.table)
library(knitr)
library(dplyr)
```

1. Potential Outcomes

Consider the following hypothetical schedule of potential outcomes.

- Amy, Brian, and Chris are *compliers*. This means they actually get the treatment if they are assigned to the treatment group. Their potential outcomes in the untreated state of the world are 11, 10, and 11 respectively.
- David, Erin, and Felipe are never-takers. (I.e. they do not get the treatment even if they are assigned to the treatment group.) Their potential outcomes in the untreated state of the world is 3, 2, and 4 respectively.

1. Make up a set of potential outcomes in the treated state of the world (i.e. $Y_i(1)$ for each of the individuals listed above) that would make both the ATE and the CACE positive.

```
d.1a <- data.table(
  'Name' = c('Amy', 'Brian', 'Chris', 'David', 'Erin', 'Felipe'),
  'Type' = c('Complier', 'Complier', 'Complier', 'NonComplier', 'NonComplier', 'NonComplier'),
  'Y_{i}(0)'   = c(11, 10, 11, 3, 2, 4),
  'Y_{i}(1)'  = c(15, 15, 15, 15, 15, 15)
)
d.1a
```

```
##      Name        Type Y_{i}(0) Y_{i}(1)
## 1:    Amy    Complier       11       15
## 2:  Brian    Complier       10       15
## 3:  Chris    Complier       11       15
## 4:  David NonComplier        3       15
## 5:   Erin NonComplier        2       15
## 6: Felipe NonComplier        4       15
```

```
print("ATE")
```

```
## [1] "ATE"
```

```
ATE.1a <- mean(d.1a$`Y_{i}(1)`) - mean(d.1a$`Y_{i}(0)`)
ATE.1a
```

```
## [1] 8.166667
```

```
print("CACE")
```

```
## [1] "CACE"
```

```
CACE.1a <- mean(d.1a[d.1a$Type == 'Complier']$`Y_{i}(1)`) - mean(d.1a[d.1a$Type == 'Complier']$`Y_{i}(0)
CACE.1a
```

```
## [1] 4.333333
```

Note here, the CACE and ATE are both positive

2. Make up a set of potential outcomes in the treated state of the world that would make the ATE positive but the CACE *negative*.

```
d.1b <- data.table(
  'Name' = c('Amy', 'Brian', 'Chris', 'David', 'Erin', 'Felipe'),
  'Type' = c('Complier', 'Complier', 'Complier', 'NonComplier', 'NonComplier', 'NonComplier'),
  'Y_{i}(0)'   = c(11, 10, 11, 3, 2, 4),
  'Y_{i}(1)'  = c(7, 4, 15, 6, 10, 8)
)
d.1b
```

```
##      Name        Type Y_{i}(0) Y_{i}(1)
## 1:    Amy    Complier       11        7
## 2:  Brian    Complier       10        4
## 3:  Chris    Complier       11       15
## 4:  David NonComplier        3        6
## 5:   Erin NonComplier        2       10
## 6: Felipe NonComplier        4        8
```

```
print("ATE")
```

```
## [1] "ATE"
```

```
ATE.1b <- mean(d.1b$`Y_{i}(1)`) - mean(d.1b$`Y_{i}(0)`)
ATE.1b
```

```
## [1] 1.5
```

```
print("CACE")
```

```
## [1] "CACE"
```

```
CACE.1b <- mean(d.1b[d.1b$Type == 'Complier']$`Y_{i}(1)`) - mean(d.1b[d.1b$Type == 'Complier']$`Y_{i}(0)
CACE.1b
```

```
## [1] -2
```

Note in this case, CACE is negative but ATE is positive

3. Suppose that you are conducting a trial for a new feature to be released in a product. From a limited point of view, if you are the person who wrote the *creative* content that is in the new feature, do you care more about the CACE or the ATE?

CACE someone who is interested in the causaul impact, and thus if you are making the creative content, you care about the CACE. You are not responsible for who actually sees the product. Instead, you are responsible for the outcome once someone does see it.

4. Suppose that you are conducting a trial for a new feature to be released in the same product. From a limited point of view, compared to when you wrote the creative, if you are the product manager, do you care relatively **more** about the CACE or the ATE than before?

Here, we care a lot more about ATE. Ultimately, the product manager needs to look at the overall impact, regardless of whether or not there are noncompliers. The product manager can then understand what the impact of releasing a product actually is, because in the real world, there WILL be noncompliers. It is the PM's job to make sure as many people comply as possible, which will often be a function of how the product's trial works.

2. Noncompliance in Recycling Experiment

Suppose that you want to conduct a study of recycling behavior. A number of undergraduate students are hired to walk door to door and provide information about the benefits of recycling to people in the treatment group. 1,500 households are assigned to the treatment group. The undergrads tell you that they successfully managed to contact 700 households. The control group had 3,000 households (not contacted by any undergraduate students). The subsequent recycling rates (i.e. the outcome variable) are computed and you find that 500 households in the treatment group recycled. In the control group, 600 households recycled.

1. What is the ITT?

```
IIT.2a <- 500/1500 - 600/3000
IIT.2a
```

```
## [1] 0.1333333
```

The ITT is 0.1333

2. What is the CACE?

```
CACE.2b <- IIT.2a / (700/1500)
CACE.2b
```

```
## [1] 0.2857143
```

The CACE is 0.2857143

3. There appear to be some inconsistencies regarding how the undergraduates actually carried out the instructions they were given. One of the students, Mike, tells you that they actually lied about the the number of contacted treatment households. The true number was 500. Another student, Andy, tells you that the true number was actually 600.

    a. What is the CACE if Mike is correct?

```
CACE.2c <- IIT.2a / (500/1500)
CACE.2c
```

```
## [1] 0.4
```

The CACE if Mike is correct is 0.4

`b. What is the CACE if Andy is correct?`

```
CACE.2cb <- IIT.2a / (600/1500)
CACE.2cb
```

```
## [1] 0.3333333
```

The CACE if Andy is correct is 0.333

4. Suppose that Mike is correct.

    a. What was the impact of the undergraduates's false reporting on our estimates of the treatment's effectiveness?

We understimate the impact of the treatment's effectiveness. It was 0.4 when it should have been 0.2857

`b. Does your answer change depending on whether you choose to focus on the ITT or the CACE?`

We don't change the value in the ITT, and thus we choose to focus on CACE. Note the ITT was the same in all the above problems. Answer chances with CACE, not with ITT

3. Fun with the placebo

The table below summarizes the data from a political science experiment on voting behavior. Subjects were randomized into three groups: a baseline control group (not contacted by canvassers), a treatment group (canvassers attempted to deliver an encouragement to vote), and a placebo group (canvassers attempted to deliver a message unrelated to voting or politics).

```
##    Assignment Treated?    N Turnout
## 1:   Baseline       No 2463  0.3008
## 2:  Treatment      Yes  512  0.3890
## 3:  Treatment       No 1898  0.3160
## 4:    Placebo      Yes  476  0.3002
## 5:    Placebo       No 2108  0.3145
```

1. Construct a data set that would reproduce the table.

```
baseline <- data.frame(Index = c(1:2463), Assignment = c("Baseline"), Treated = c("No"), Turnout=sample

placebo_no <- data.frame(Index = c(1:2108), Assignment = c("Placebo"), Treated = c("No"), Turnout=sampl

placebo_yes <- data.frame(Index = c(1:476), Assignment = c("Placebo"), Treated = c("Yes"), Turnout=samp

treatment_no <- data.frame(Index = c(1:1898), Assignment = c("Treatment"), Treated = c("No"), Turnout=sa

treatment_yes <- data.frame(Index = c(1:512), Assignment = c("Treatment"), Treated = c("Yes"), Turnout=


d3 <- rbind(baseline, placebo_no, placebo_yes, treatment_yes, treatment_no)
head(d3)
```

```
##   Index Assignment Treated Turnout
## 1     1   Baseline      No       1
## 2     2   Baseline      No       0
## 3     3   Baseline      No       0
## 4     4   Baseline      No       0
## 5     5   Baseline      No       1
## 6     6   Baseline      No       0
```

2. Estimate -the proportion of compliers by using the data on the treatment group.

```
compliers3b <- nrow(d3[d3$Assignment == 'Treatment' & d3$Treated == 'Yes', ])
noncompliers3b <- nrow(d3[d3$Assignment == 'Treatment' & d3$Treated == 'No', ])

compliers_treatment <- compliers3b/(noncompliers3b+compliers3b)
compliers_treatment
```

```
## [1] 0.2124481
```

3. Estimate the proportion of compliers by using the data on the placebo group.

```
compliers3c <- nrow(d3[d3$Assignment == 'Placebo' & d3$Treated == 'Yes', ])
noncompliers3c <- nrow(d3[d3$Assignment == 'Placebo' & d3$Treated == 'No', ])


compliers_placebo <- compliers3c/(noncompliers3c+compliers3c)
compliers_placebo
```

```
## [1] 0.1842105
```

4. Are the proportions in parts (1) and (2) statistically significantly different from each other? Provide *a test* and an description about why you chose that particular test, and why you chose that particular set of data.

```
library(MASS)
tbl = table(d3$Assignment, d3$Treated)
#Drop Baseline row
tbl = tbl[2:nrow(tbl), ]
tbl
```

```
##
##                No  Yes
##   Placebo    2108  476
##   Treatment 1898  512
```

```
chisq.test(tbl)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tbl
## X-squared = 6.0887, df = 1, p-value = 0.0136
```

At the 5% significance level, we can see that the proportions are different from each other (p value = 0.0136). I used a Chi Squared Test because we are dealing with categorical variables. I chose Assignment v Treated because we care about compliers and non compliers. By seeing them in a matrix for the Treatment and Placebo group, we can look for a difference in proportions.

A T-Test and Chi Squared test would give us the same value. We are seeing if the proprotions are fundamentall different.

e. What critical assumption does this comparison of the two groups' compliance rates test? *Porportion of nevertakers does not change. Compliance doesn't change based on assingment. This was brought up multiple times in class, and is a central assumption here as well.*

f. Estimate the CACE of receiving the placebo. Is the estimate consistent with the assumption that the placebo has no effect on turnout?

```
ITT.4f <- mean(d3$Turnout[d3$Assignment=='Placebo']) - mean(d3$Turnout[d3$Assignment=='Baseline'])

IITd.4f <- compliers_placebo

CACE.4f = ITT.4f/IITd.4f
CACE.4f
```

```
## [1] 0.01585653
```

The CACE is above. This is NOT consistent with the fact that placebo has no effect on turnout, because there is some sort of effect. We would have to run a test to see difference in the two groups. Note that based on the sampling, this CACE number can vary greatly.

g. Estimate the CACE by first estimating the ITT and then dividing by ITT_D.

```
ITT.4g <- mean(d3$Turnout[d3$Assignment=='Treatment']) - mean(d3$Turnout[d3$Assignment=='Baseline'])
ITT.4g
```

```
## [1] 0.0265422
```

```
IITd.4g <- compliers_treatment
```

```
CACE.4g = ITT.4g/IITd.4g
CACE.4g
```

```
## [1] 0.124935
```

I assume this means Treatment to Baseline. The CACE is above.

h. Estimate the CACE by comparing the turnout rates among the compliers in both the treatment and placebo groups. Interpret the results.

```
# We know who the compliers are, do we can look at the difference
ITT.4h <- mean(d3$Turnout[d3$Assignment=='Treatment' & d3$Treated=='Yes']) - mean(d3$Turnout[d3$Assignme
ITT.4h
```

```
## [1] 0.09020483
```

Compare treatment to placebo. This means that there is a difference in the treatment effect of the compliers between Treatment and Placebo.

i. In class we discussed that the rate of compliance determines whether one or another design is more efficient. (You can review the paper here). Given the compliance rate in this study, which design *should* provide a more efficient estimate of the treatment effect?

When compliance is more than 50%, run with control. Less than 50% run with placebo. Thus, we are less than 20% here, thus we run with placebo. Thus, subject in the placebo group, compared to the treatment group should give a more efficient estimate of the treatment effect. This way, we can measure the non compliance in both groups.

j. Does it?

Yes it does. Look at the results above.

4. Turnout in Dorms

Guan and Green report the results of a canvassing experiment conduced in Beijing on the eve of a local election. Students on the campus of Peking University were randomly assigned to treatment or control groups. Canvassers attempted to contact students in their dorm rooms and encourage them to vote. No contact with the control group was attempted. Of the 2,688 students assigned to the treatment group, 2,380 were contacted. A total of 2,152 students in the treatment group voted; of the 1,334 students assigned to the control group, 892 voted. One aspect of this experiment threatens to violate the exclusion restriction. At every dorm room they visited, even those where no one answered, canvassers left a leaflet encouraging students to vote.

```
d.4 <- fread('./data/Guan_Green_CPS_2006.csv')
head(d.4)
```

```
##    turnout treated  dormid treatment_group
## 1:       0       0 1010101               0
## 2:       0       0 1010101               0
## 3:       0       0 1010101               0
## 4:       0       0 1010102               0
## 5:       0       0 1010102               0
## 6:       0       1 1010103               1
```

```
IITd.4 <- 2380/2688
IITd.4
```

```
## [1] 0.8854167
```

Here's what is in that data:

- `turnout` did the person turn out to vote?
- `treated` did someone at the dorm open the door?
- `dormid` a unique ID for the door of the dorm
- `treatment_group` whether the dorm door was assigned to be treated or not

1. Using the data set from the book's website, estimate the ITT. First, estimate the ITT using the difference in two-group means. Then, estimate the ITT using a linear regression on the appropriate subset of data. *Heads up: There are two NAs in the data frame. Just na.omit to remove these rows.*

```
d.4 <- na.omit(d.4)

ITT4 <- mean(d.4$turnout[d.4$treatment_group=='1']) - mean(d.4$turnout[d.4$treatment_group=='0'])
ITT4
```

```
## [1] 0.1319296
```

```
lin.reg.1 <- lm(turnout ~ treatment_group, d.4)
summary(lin.reg.1)
```

```
##
## Call:
## lm(formula = turnout ~ treatment_group, data = d.4)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8006  0.1994  0.1994  0.1994  0.3313
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.66867    0.01162  57.521   <2e-16 ***
## treatment_group  0.13193    0.01422   9.278   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4246 on 4020 degrees of freedom
## Multiple R-squared:  0.02096,    Adjusted R-squared:  0.02072
## F-statistic: 86.08 on 1 and 4020 DF,  p-value: < 2.2e-16
```

2. Use randomization inference to test the sharp null hypothesis that the ITT is zero for all observations, taking into account the fact that random assignment was clustered by dorm room. Interpret your results – in particular, are you surprised at your result when you compare it to the p-value in part (1)? (This is a 2 point question, because there's quite a bit of work here.)

```
afterTreatment <- function(a, b) {
  treament = a[b==1]
  control = a[b==0]
  return(mean(treament) - mean(control))
}

itt <- afterTreatment(d.4$turnout, d.4$treatment_group)
# itt

# Sample function takes amount of households in control/treatment, sample at a same rate
random <- function() {
  sample(0:1, size=1334+2688, replace=TRUE, prob=c(1334/(2688+1334), 2688/(2688+1334)))
```

```
}

N <- 10000
simulated.trial <- replicate(N, afterTreatment(d.4$turnout, random()))
simulated.trial.mean <- mean(simulated.trial)
# simulated.trial.mean

num_larger <- simulated.trial >= itt
number.simulated.assingments <- sum(num_larger)
p_value_one_tailed <- mean(num_larger)
p_value_one_tailed
```

```
## [1] 0
```

We can see the p values are quite similar: <2e-16 vs 0. This means we know there is a stat significant difference between the treatment and control group.

3. Assume that the leaflet had no effect on turnout. Estimate the CACE. Do this in two ways:

a. First, estimate the CACE using means.

```
ITTd4 <- nrow(d.4[d.4$treatment_group=='1' & d.4$treated == '1'])/nrow(d.4[d.4$treatment_group == '1'])

CACE4ca <- ITT4/ITTd4
CACE4ca
```

```
## [1] 0.1489402
```

b. Second, use some form of linear model to estimate this as well. If you use a 2SLS, then report the standard errors and draw inference about whether contact had any causal effect among compliers.

```
summary(lm(turnout ~ treated, data = d.4))
```

```
##
## Call:
## lm(formula = turnout ~ treated, data = d.4)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8043  0.1957  0.1957  0.3120  0.3120
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.68800    0.01050  65.535   <2e-16 ***
## treated      0.11629    0.01364   8.523   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4253 on 4020 degrees of freedom
## Multiple R-squared:  0.01775,    Adjusted R-squared:  0.0175
## F-statistic: 72.64 on 1 and 4020 DF,  p-value: < 2.2e-16
```

I get an estaimte of 0.11629(0.01364) with a p value of <2e-16. This tells us that contact did have a stat significant effect on the outcome.

5. Another Turnout Question

We're sorry; it is just that the outcome and treatment spaces are so clear!

Hill and Kousser (2015) report that it is possible to increase the probability that someone votes in the California *Primary Election* simply by sending them a letter in the mail. This is kind of surprising, because who even reads the mail anymore anyways? (Actually, if you talk with folks who work in the space, they'll say, "We know that everybody throws our mail away; we just hope they see it on the way to the garbage.")

Can you replicate their findings? Let's walk through them.

```
library(dplyr)
# d.5 <- fread('http://ischool.berkeley.edu/~d.alex.hughes/data/hill_kousser_analysisFile.csv')
# head(d.5)
# fwrite(d.5, file = '/Users/h/Documents/1H/Dropbox/MIDS/w241-ps4-fall2019-section1/data/raw/d5.csv')
#
d.5 <- fread('./data/raw/d5.csv')
head(d.5)
```

```
##    LocalityCode age.bin party.bin in.toss.up.dist minority.dist
## 1:            1       2         1               0             0
## 2:            1       5         3               0             0
## 3:            1       6         2               0             0
## 4:            1       5         1               1             1
## 5:            1       5         2               0             0
## 6:            1       6         3               0             0
##    vote.10.gen vote.08.gen Party age.in.14 Gender Dist1 Dist2 Dist3 Dist4
## 1:           0           0   REP        32      F CG015 SA020 SE002 SS010
## 2:           0           1   NPP        61      F CG013 SA018 SE002 SS009
## 3:           0           1   DEM        77      M CG013 SA015 SE002 SS009
## 4:           0           1   REP        62      M CG017 SA025 SE002 SS010
## 5:           1           1   DEM        60      M CG015 SA020 SE002 SS010
## 6:           1           1   NPP        81        CG015 SA020 SE002 SS010
##    Dist5 Dist6 Dist7 Dist8 reg.date.pre.08 reg.date.pre.10 vote.12.gen
## 1:    NA    NA    NA    NA               1               1           1
## 2:    NA    NA    NA    NA               0               0           1
## 3:    NA    NA    NA    NA               1               1           1
## 4:    NA    NA    NA    NA               0               0           1
## 5:    NA    NA    NA    NA               1               1           1
## 6:    NA    NA    NA    NA               1               1           1
##    vote.12.pre.pri vote.10.gen.pri vote.08.pre.pri vote.08.gen.pri
## 1:               0               0               0               0
## 2:               0               0               0               0
## 3:               0               0               0               0
## 4:               0               0               0               0
## 5:               0               0               0               0
## 6:               0               0               0               0
##    block.num leftover.case treatment.assign yvar matched.to.post
## 1:        91             0          Control    0               1
## 2:       316             0          Control    0               1
## 3:       364             0          Control    1               1
## 4:       296             0          Control    0               1
## 5:       302             0          Control    0               1
## 6:       382             0          Control    0               1
##    vote.14.gen
```

```
## 1:           0
## 2:           0
## 3:           1
## 4:           0
## 5:           1
## 6:           0
```

```
d.5.updated <- dplyr::select(d.5, treatment.assign, yvar)
d.5.blocks <- dplyr::select(d.5, block.num)
d.5.block_turnout <- dplyr::select(d.5, block.num, treatment.assign, yvar)
d.5.hetr <- dplyr::select(d.5, vote.14.gen, reg.date.pre.10, yvar, treatment.assign)

fwrite(d.5.updated, file = '/Users/h/Documents/1H/Dropbox/MIDS/w241-ps4-fall2019-section1/data/analysis

fwrite(d.5.blocks, file = '/Users/h/Documents/1H/Dropbox/MIDS/w241-ps4-fall2019-section1/data/analysis/

fwrite(d.5.hetr, file = '/Users/h/Documents/1H/Dropbox/MIDS/w241-ps4-fall2019-section1/data/analysis/he

fwrite(d.5.block_turnout, file = '/Users/h/Documents/1H/Dropbox/MIDS/w241-ps4-fall2019-section1/data/ana

fwrite(d.5.hetr, file = '/Users/h/Documents/1H/Dropbox/MIDS/w241-ps4-fall2019-section1/data/analysis/he

# nrow(d.5.blocks)

# summary(d.5.updated)
```

You'll note that this takes some time to download. Probably best to save a copy locally, and keep from reading this off the internet. In your project structure, create a folder called `./data/raw/` and write this file to the folder. Data that is in this raw folder *never* gets modified; instead, any changes that you make should be reflected into either an `./data/interim/` or `./data/analysis/` folder. You might consider using the function `fwrite` from data.table.

Here's what is in that data.

- `age.bin` a bucketed version of the `age.in.14` variable
- `party.bin` a bucketed version of the `Party` variable
- `in.toss.up.dist` whether the voter lives in a close race
- `minority.dist` whether the voter lives in a majority minority district
- `Gender` voter file reported gender
- `Dist1-8` congressional and data districts
- `reg.date.pre.08` whether the voter has been registered since before 2008
- `vote.xx.gen` whether the voter voted in the `xx` general election
- `vote.xx.gen.pri` whether the voter voted in the `xx` general primary election
- `vote.xx.pre.pri` whether the voter voted in the `xx` presidential primary election
- `block.num` a block indicator for blocked random assignment.
- `treatment.assign` either "Control", "Election Info", "Partisan Cue", or "Top-Two Info"
- `yvar` the outcome variable: did the voter vote in the 2014 primary election

These variable names are horrible. Do two things:

- Rename the smallest set of variables that you think you might use to something more useful
- For the variables that you think you might use; check that the data makes sense;

Then, save this data to `./data/analysis/`.

Well, while you're at it, you might as well also modify your `.gitignore` to ignore the data folder. Because you're definitely going to have the data rejected when you try to push it to github.

1. **A Simple Treatment Effect**: Load the data from `./data/analysis/` and estimate a model that compares the rates of turnout in the control group to the rate of turnout among *anybody* who received a letter. Report robust standard errors.

```
library(sandwich)
library(lmtest)
get_VCOC_HC <- function(lm) {
  vcovHC <- vcovHC(lm)
  return (vcovHC)
}

get_ROBUST_SE <- function(lm) {
  print("Robust SE")
  return(sqrt(diag(get_VCOC_HC(lm))))
}

d5.filtered.a <- fread('./data/analysis/d5filtered.csv')
# head(d5.filtered)

nrow(d5.filtered.a[d5.filtered.a$treatment.assign == 'Control'])
```

```
## [1] 3722672
```

```
nrow(d5.filtered.a[d5.filtered.a$treatment.assign == 'Election info'])
```

```
## [1] 29885
```

```
nrow(d5.filtered.a[d5.filtered.a$treatment.assign == 'Partisan'])
```

```
## [1] 59857
```

```
nrow(d5.filtered.a[d5.filtered.a$treatment.assign == 'Top-two info'])
```

```
## [1] 59854
```

```
# Now weve made anything else into treatment
d5.filtered.a$treatment.assign[d5.filtered.a$treatment.assign == "Control"] <- "0"
d5.filtered.a$treatment.assign[d5.filtered.a$treatment.assign == "Election info"] <- "1"
d5.filtered.a$treatment.assign[d5.filtered.a$treatment.assign == "Partisan"] <- "1"
d5.filtered.a$treatment.assign[d5.filtered.a$treatment.assign == "Top-two info"] <- "1"

summary(lm(yvar ~ treatment.assign, data = d5.filtered.a))
```

```
##
## Call:
## lm(formula = yvar ~ treatment.assign, data = d5.filtered.a)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.09802 -0.09312 -0.09312 -0.09312  0.90688
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.0931248  0.0001508 617.722  < 2e-16 ***
## treatment.assign1 0.0048992  0.0007670   6.388 1.69e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.2909 on 3872266 degrees of freedom
## Multiple R-squared:  1.054e-05,  Adjusted R-squared:  1.028e-05
## F-statistic:  40.8 on 1 and 3872266 DF,  p-value: 1.686e-10
```

```
get_ROBUST_SE(lm(yvar ~ treatment.assign, data = d5.filtered.a))
```

```
## [1] "Robust SE"
```

```
##       (Intercept) treatment.assign1
##      0.0001506188      0.0007834035
```

For treatment, we have 0.0048992(0.0007834035), and a p value of 1.69e-10.

2. **Specific Treatment Effects**: Suppose that you want to know whether different letters have different effects. To begin, what are the effects of each of the letters, as compared to control? Report robust standard errors on a linear model.

```
d5.filtered.b <- fread('./data/analysis/d5filtered.csv')
summary(lm(yvar ~ treatment.assign, data = d5.filtered.b))
```

```
##
## Call:
## lm(formula = yvar ~ treatment.assign, data = d5.filtered.b)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.09838 -0.09312 -0.09312 -0.09312  0.90688
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   0.0931248  0.0001508 617.722  < 2e-16 ***
## treatment.assignElection info 0.0049846  0.0016893   2.951 0.003171 **
## treatment.assignPartisan      0.0052597  0.0011984   4.389 1.14e-05 ***
## treatment.assignTop-two info  0.0044961  0.0011984   3.752 0.000176 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2909 on 3872264 degrees of freedom
## Multiple R-squared:  1.059e-05,  Adjusted R-squared:  9.816e-06
## F-statistic: 13.67 on 3 and 3872264 DF,  p-value: 6.508e-09
```

```
get_ROBUST_SE(lm(yvar ~ treatment.assign, data = d5.filtered.b))
```

```
## [1] "Robust SE"
```

```
##                   (Intercept) treatment.assignElection info
##                  0.0001506188                   0.0017273388
##      treatment.assignPartisan  treatment.assignTop-two info
##                  0.0012266555                   0.0012224977
```

For Election info, we have 0.0049846(0.0017273388), for Partisan we have 0.0011984(0.0012266555), and for Top 2 we have 0.0044961(0.0012224977)

3. Then, test, using an F-test, whether the increased flexibility of the model estimated in part (2) has improved the performance of the model over that estimated in part (1). What does the evidence suggest?

```
anova(lm(yvar ~ treatment.assign, data = d5.filtered.a), lm(yvar ~ treatment.assign, data = d5.filtered
```

```
## Analysis of Variance Table
```

```
##
## Model 1: yvar ~ treatment.assign
## Model 2: yvar ~ treatment.assign
##    Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1 3872266 327616
## 2 3872264 327616  2  0.017723 0.1047 0.9006
```

4. **More Specific Treatment Effects** Is one message more effective than the others? The authors have drawn up this design as a full-factorial design. Write a *specific* test for the difference between the *Partisan* message and the *Election Info* message. Write a *specific* test for the difference between *Top-Two Info* and the *Election Info* message. Report robust standard errors on both tests.

```
d5.filtered.d <- fread('./data/analysis/d5filtered.csv')


d5.filtered.d <- d5.filtered.d[d5.filtered.d$treatment.assign == "Election info" | d5.filtered.d$treatme

d5.filtered.d$treatment.assign[d5.filtered.d$treatment.assign == "Partisan"] <- "0"
d5.filtered.d$treatment.assign[d5.filtered.d$treatment.assign == "Election info"] <- "1"

summary(lm(yvar ~ treatment.assign, data = d5.filtered.d))
```

```
##
## Call:
## lm(formula = yvar ~ treatment.assign, data = d5.filtered.d)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.09838 -0.09838 -0.09838 -0.09811  0.90189
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.0983845  0.0012169   80.85   <2e-16 ***
## treatment.assign1 -0.0002751  0.0021087   -0.13    0.896
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2977 on 89740 degrees of freedom
## Multiple R-squared:  1.896e-07,  Adjusted R-squared:  -1.095e-05
## F-statistic: 0.01702 on 1 and 89740 DF,  p-value: 0.8962
```

```
get_ROBUST_SE(lm(yvar ~ treatment.assign, data = d5.filtered.d))
```

```
## [1] "Robust SE"
```

```
##       (Intercept) treatment.assign1
##       0.001217373       0.002107845
```

```
d5.filtered.d2 <- fread('./data/analysis/d5filtered.csv')
d5.filtered.d2 <- d5.filtered.d2[d5.filtered.d2$treatment.assign == "Election info" | d5.filtered.d2$tre

d5.filtered.d2$treatment.assign[d5.filtered.d2$treatment.assign == "Top-two info"] <- "0"
d5.filtered.d2$treatment.assign[d5.filtered.d2$treatment.assign == "Election info"] <- "1"

summary(lm(yvar ~ treatment.assign, data = d5.filtered.d2))
```

```
##
```

```
## Call:
## lm(formula = yvar ~ treatment.assign, data = d5.filtered.d2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.09811 -0.09811 -0.09762 -0.09762  0.90238
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.0976209  0.0012141  80.407   <2e-16 ***
## treatment.assign1 0.0004885  0.0021038   0.232    0.816
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.297 on 89737 degrees of freedom
## Multiple R-squared:  6.009e-07,  Adjusted R-squared:  -1.054e-05
## F-statistic: 0.05392 on 1 and 89737 DF,  p-value: 0.8164
```

```
get_ROBUST_SE(lm(yvar ~ treatment.assign, data = d5.filtered.d2))
```

```
## [1] "Robust SE"
```

```
##       (Intercept) treatment.assign1
##       0.001213184       0.002105428
```

For Election Info and Partisan, Partisan was better than election by 0.0002751(0.002107845). For Election Info vs Top Two, Election was better by 0.0004885(0.002105428). Note the SE are such that if we take a confidence interval, it could very well be the case there is no difference between election info/partisan and top 2/election info.

5. **Blocks?** There are a *many* of blocks in this data. How many?

```
d5.filtered.e <- fread('./data/analysis/blocks.csv')
length(unique(d5.filtered.e$block.num))
```

```
## [1] 382
```

There are 382 blocks

6. Create a new indicator that is the *average turnout within a block* and attach this back to the data.table. Use this new indicator in a regression that predicts the difference between Control and Any Letter. Then, using an F-test, does the increased information from all these blocks improve the performance of the *causal* model? Use an F-test to check.

```
d5.filtered.f <- fread('./data/analysis/d.5.block_turnout.csv')

# head(d5.filtered.f)

# Now we have turnout in each block
test <- aggregate(d5.filtered.f[, 3], list(d5.filtered.f$block.num), mean)

# head(test)
# Make key value pairs
DF2 <- setNames(test$yvar, test$Group.1)

# Now find that value from DF2
d5.filtered.f$Indicator <- d5.filtered.f$block.num
d5.filtered.f$Indicator <- as.character(d5.filtered.f$Indicator)
```

```r
d5.filtered.f$Indicator <- DF2[d5.filtered.f$Indicator]

# Here we have any letter
d5.filtered.f$treatment.assign[d5.filtered.f$treatment.assign == "Control"] <- "0"
d5.filtered.f$treatment.assign[d5.filtered.f$treatment.assign == "Election info"] <- "1"
d5.filtered.f$treatment.assign[d5.filtered.f$treatment.assign == "Partisan"] <- "1"
d5.filtered.f$treatment.assign[d5.filtered.f$treatment.assign == "Top-two info"] <- "1"

#Lin Reg
summary(lm(yvar ~ treatment.assign + Indicator, data = d5.filtered.f))
```

```
##
## Call:
## lm(formula = yvar ~ treatment.assign + Indicator, data = d5.filtered.f)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38038 -0.10342 -0.07671 -0.04433  0.97938
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -0.0001904  0.0002812  -0.677    0.499
## treatment.assign1  0.0049238  0.0007524   6.545 5.97e-11 ***
## Indicator          1.0000014  0.0025635 390.090  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2853 on 3872265 degrees of freedom
## Multiple R-squared:  0.03782,    Adjusted R-squared:  0.03782
## F-statistic: 7.611e+04 on 2 and 3872265 DF,  p-value: < 2.2e-16
```

```r
# F test
anova(lm(yvar ~ treatment.assign, data = d5.filtered.a), lm(yvar ~ treatment.assign + Indicator, data =
```

```
## Analysis of Variance Table
##
## Model 1: yvar ~ treatment.assign
## Model 2: yvar ~ treatment.assign + Indicator
##    Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1 3872266 327616
## 2 3872265 315228  1     12388 152170 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The F value is 152170, and the p value is 2.2e-16. We thus know there is a stat significant difference, and that the blocks improve the performance of the model. An F value would be 0 when there is no predictive capability. This is clearly not the case.

Spoke to Alex in OH and confirmed you can use LM to test difference in groups.

7. **HTES?** Do you think that there are features of the data that might systematically predict that people will respond strongly or weakly to the treatment effect? List two that you think might be there, in the order that you would like to test them. Then, test for these heterogeneities. What do you learn? What is the right way to adjust your p-values, given that you're testing twice?

```r
d5.filtered.g <- fread('./data/analysis/hetr.csv')

d5.filtered.g$treatment.assign[d5.filtered.g$treatment.assign == "Control"] <- "0"
d5.filtered.g$treatment.assign[d5.filtered.g$treatment.assign == "Election info"] <- "1"
d5.filtered.g$treatment.assign[d5.filtered.g$treatment.assign == "Partisan"] <- "1"
d5.filtered.g$treatment.assign[d5.filtered.g$treatment.assign == "Top-two info"] <- "1"

# We test seperately
summary(lm(yvar ~ treatment.assign + reg.date.pre.10, data = d5.filtered.g))
```

```
##
## Call:
## lm(formula = yvar ~ treatment.assign + reg.date.pre.10, data = d5.filtered.g)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.12663 -0.12175 -0.07046 -0.07046  0.92954
##
## Coefficients:
##                     Estimate Std. Error  t value Pr(>|t|)
## (Intercept)        0.1217542  0.0002235  544.851  < 2e-16 ***
## treatment.assign1  0.0048724  0.0007641    6.377 1.81e-10 ***
## reg.date.pre.10   -0.0512976  0.0002965 -173.010  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2898 on 3872265 degrees of freedom
## Multiple R-squared:  0.007681,   Adjusted R-squared:  0.007681
## F-statistic: 1.499e+04 on 2 and 3872265 DF,  p-value: < 2.2e-16
```

```r
summary(lm(yvar ~ treatment.assign + vote.14.gen, data = d5.filtered.g))
```

```
##
## Call:
## lm(formula = yvar ~ treatment.assign + vote.14.gen, data = d5.filtered.g)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.2579 -0.0329 -0.0329 -0.0329  0.9671
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.0329005  0.0001656 198.636  < 2e-16 ***
## treatment.assign1 0.0048792  0.0007220   6.758  1.4e-11 ***
## vote.14.gen       0.2201077  0.0003121 705.194  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2738 on 3872265 degrees of freedom
## Multiple R-squared:  0.1138, Adjusted R-squared:  0.1138
## F-statistic: 2.487e+05 on 2 and 3872265 DF,  p-value: < 2.2e-16
```

```r
# If we combine them
summary(lm(yvar ~ treatment.assign + reg.date.pre.10 + vote.14.gen, data = d5.filtered.g))
```

```
##
## Call:
## lm(formula = yvar ~ treatment.assign + reg.date.pre.10 + vote.14.gen,
##     data = d5.filtered.g)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.28131 -0.05811 -0.05811 -0.01381  0.98619
##
## Coefficients:
##                      Estimate Std. Error  t value Pr(>|t|)
## (Intercept)         0.0581060  0.0002292  253.493  < 2e-16 ***
## treatment.assign1   0.0048562  0.0007197    6.747  1.5e-11 ***
## reg.date.pre.10    -0.0442989  0.0002795 -158.511  < 2e-16 ***
## vote.14.gen         0.2183458  0.0003113  701.367  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2729 on 3872264 degrees of freedom
## Multiple R-squared:  0.1195, Adjusted R-squared:  0.1195
## F-statistic: 1.752e+05 on 3 and 3872264 DF,  p-value: < 2.2e-16
```

The first covariate tested was reg.date.pre.10, and the second one was vote.14.gen. It makes sense becauase we want to see who was been registered since 2010, and then we want to see who voted in the year's general election. The F stats were 1.499e+04 and 2.487e+05 respectively, with both p values being 2e-16. It is clear to see that more than the treatment itsself, whether or not someone has been registered since 2010, and whether or not they voted in the 2014 general election, would determine if they would vote in the 2014 primary election.

When testing the two features together, we see p values of 2e-16. Again, both are statistically significant. We can see the F stat is now 1.752e+05. It seems as if vote.14.gen is the strongest predictor of voting in the primary.

Given we are testing 2x, we should use a Bonferri Correction, which notes that one must divide the p value but # of tests they are running. Thus, we should divide the p value threshold by 2.

8. Summarize these results in a short paragraph that includes inline reports from your estimated models. (This can be integrated into your last response, if that works better for you.)

Check last part

9. Cheating? Suppose that you didn't write down your testing plan. How risky is the false discovery problem in this data set?

The false discovery problem is extremely risky. Note there are 31 features in this data set – all of which we can test for. Given our p value is 0.05, after we test more than 20 of the features, atleast 1 will be significant due to chance. Thus, we must adjust the p value using the Bonferri Correction.