Problem Set #2

Andrew Walters

June 8, 2018

1. What happens when pilgrims attend the Hajj pilgrimage to Mecca?

On the one hand, participating in a common task with a diverse group of pilgrims might lead to increased mutual regard through processes identified in *Contact Theories*. On the other hand, media narritives have raised the spectre that this might be accompanied by "antipathy toward non-Muslims". Clingingsmith, Khwaja and Kremer (2009) investigates the question.

Using the data here, test the sharp null hypothesis that winning the visa lottery for the pilgrimage to Mecca had no effect on the views of Pakistani Muslims toward people from other countries. Assume that the Pakistani authorities assigned visas using complete random assignment. Use, as your primary outcome the `views` variable, and as your treatment feature `success`. If you're ambitious, write your fucntion generally so that you can also evaluate feeligns toward specific nationalities.

a. Using either 'dplyr' or 'data.table', group the data by 'success' and report whether views toward others are generally more positive among lottery winners or lottery non-winners.

lottery winners (success == 1) have more favorable views (2.3 on average) than non-lottery winners (1.9 on average).

```
p1.dt <- data.table::data.table(p1.d)
result <- p1.dt[,mean(views),by=success]
real.treatment.effect <- (as.numeric(result[2,2]-result[1,2]))
print(sprintf("Mean views of unccussessful visa applicants: %.2f and successful applicants: %.2f", as.nu
```

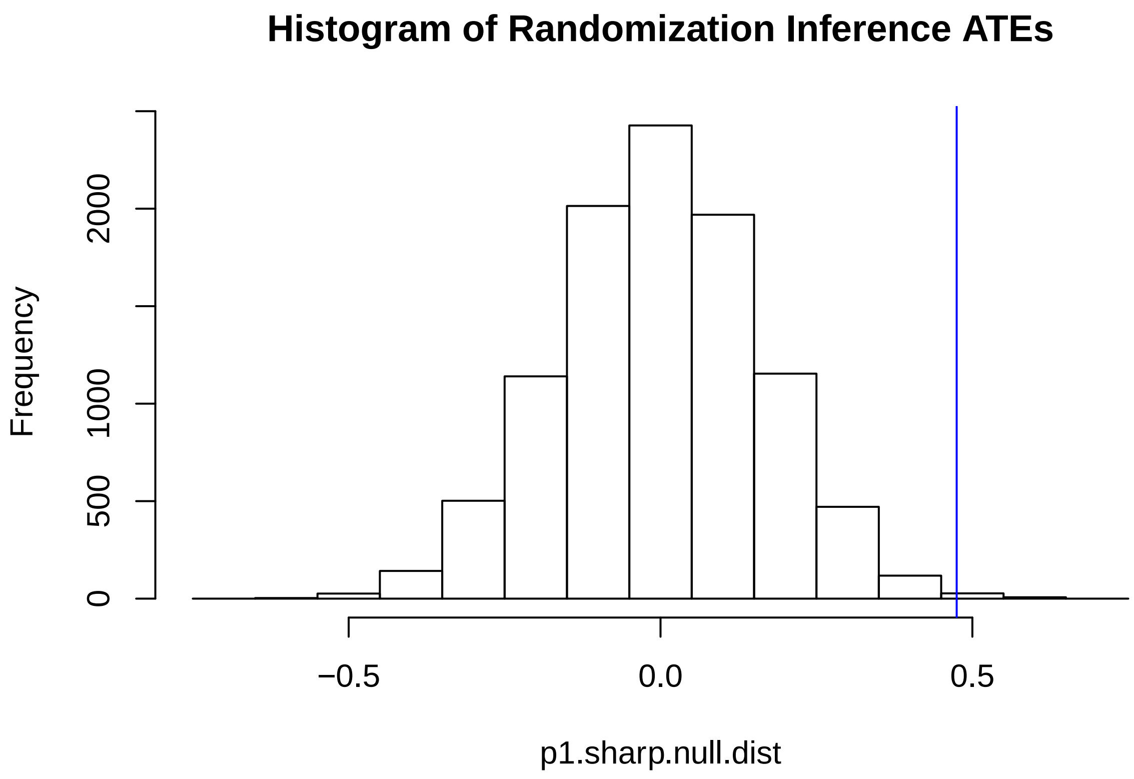
```
## [1] "Mean views of unccussessful visa applicants: 1.87 and successful applicants: 2.34"
```

```
print(sprintf("Experimental treatment effect: %.2f", real.treatment.effect))
```

```
## [1] "Experimental treatment effect: 0.47"
```

b. But is this a meaningful difference, or could it just be randomization noise? Conduct 10,000 simulated random assignments under the sharp null hypothesis to find out. (Don't just copy the code from the async, think about how to write this yourself.)

```
ri.binary <- function(dataTable,treatment,outcome) {
  rand.treatment <- rbinom(length(dataTable[,success]), 1, 0.5)
  experiment <- data.table::data.table(views=p1.dt[,views],success=rand.treatment)
  result <- experiment[,mean(views),by=success]
  ate <- (as.numeric(result[2,2]-result[1,2]))
  return (ate)
}
num_trials <- 10000
p1.sharp.null.dist <- replicate(num_trials,ri.binary(p1.dt,"success","views"))
hist(p1.sharp.null.dist,main = "Histogram of Randomization Inference ATEs",breaks = seq(-0.75,0.75,0.1))
abline(v = real.treatment.effect, col = "blue")
```



c. How many of the simulated random assignments generate an estimated ATE that is at least as large as the actual estimate of the ATE?

```
print(sprintf("Number of simulated ATEs as large as the ATE found experimentally: %.2f", sum(p1.sharp.nu
```

```
## [1] "Number of simulated ATEs as large as the ATE found experimentally: 24.00"
```

d. What is the implied *one-tailed* p-value?

```
print(sprintf("The one-tailed P-value for this experiment is: %.4f", mean(p1.sharp.null.dist>=real.trea
```

```
## [1] "The one-tailed P-value for this experiment is: 0.0024"
```

e. How many of the simulated random assignments generate an estimated ATE that is at least as large *in absolute value* as the actual estimate of the ATE?

```
print(sprintf("Number of simulated ATEs as large as the ATE found experimentally in absolute value: %.2
```

```
## [1] "Number of simulated ATEs as large as the ATE found experimentally in absolute value: 41.00"
```

f. What is the implied two-tailed p-value?

```
print(sprintf("The two-tailed P-value for this experiment is: %.4f", mean(abs(p1.sharp.null.dist)>=abs(
```

```
## [1] "The two-tailed P-value for this experiment is: 0.0041"
```

2. Term Limits Aren't Good.

Naturally occurring experiments sometimes involve what is, in effect, block random assignment. For example, Rocio Titiunik , in this paper studies the effect of lotteries that determine whether state senators in TX and AR serve two-year or four-year terms in the aftermath of decennial redistricting. These lotteries are conducted within each state, and so there are effectively two distinct experiments on the effects of term length.

The "thoery" in the news (such as it is), is that legislators who serve 4 year terms have more time to slack off and not produce legislation. If this were true, then it would stand to reason that making terms shorter would increase legislative production.

One way to measure legislative production is to count the number of bills (legislative proposals) that each senator introduces during a legislative session. The table below lists the number of bills introduced by senators in both states during 2003.

```
library(foreign)

p2.d <- read.dta("./experiments-causality/assignments/PS2-upstream/data/Titiunik.2010.dta")
head(p2.d,n = 10)
```

```
##    term2year bills_introduced texas0_arkansas1
## 1          0               18                0
## 2          0               29                0
## 3          0               41                0
## 4          0               53                0
## 5          0               60                0
## 6          0               67                0
## 7          0               75                0
## 8          0               79                0
## 9          0               79                0
## 10         0               88                0
```

a. Using either ‘dplyr‘ or ‘data.table‘, group the data by state and report the mean number of bills introduced in each state. Does Texas or Arkansas seem to be more productive? Then, group by two- or four-year terms (ignoring states). Do two- or four-year terms seem to be more productive? **Which of these effects is causal, and which is not?** Finally, using ‘dplyr‘ or ‘data.table‘ to group by state and term-length. How, if at all, does this change what you learn?

From all of these results, we see that uniformily texas produced more bills than Oklahoma and that 4-year congresspeople regardless of state produced more bills per term than 2-year congresspeople.

```
# group by state
p2.dt <- data.table::data.table(p2.d)
result <- p2.dt[,mean(bills_introduced),by=texas0_arkansas1]
print(sprintf("Mean bills per congressperson introduced in Texas: %.2f and Arkansas: %.2f", as.numeric(r
```

```
## [1] "Mean bills per congressperson introduced in Texas: 68.77 and Arkansas: 25.51"
```

```
print(sprintf("Texas produced %.2f more bills in this term per congressperson.", (as.numeric(result[1,2
```

```
## [1] "Texas produced 43.26 more bills in this term per congressperson."
```

```
# group by term length
result <- p2.dt[,mean(bills_introduced),by=term2year]
print(sprintf("Mean bills of 4-year congresspeople: %.2f and 2-year congresspeople: %.2f", as.numeric(re
## [1] "Mean bills of 4-year congresspeople: 53.09 and 2-year congresspeople: 38.58"
```

```
print(sprintf("4-Year congresspeople produced %.2f more bills on average in this term.", as.numeric(resu
```

```
## [1] "4-Year congresspeople produced 14.52 more bills on average in this term."
```

```
# group by both state and term length
p2.result <- p2.dt[,mean(bills_introduced),by=c("term2year","texas0_arkansas1")]
print(p2.result)
```

```
##    term2year texas0_arkansas1       V1
## 1:         0                0 76.87500
## 2:         1                0 60.13333
## 3:         0                1 30.70588
## 4:         1                1 20.61111
```

b. For each state, estimate the standard error of the estimated ATE.

```
tx.ate <- (as.numeric(p2.result[1,3]-p2.result[2,3]))
ar.ate <- (as.numeric(p2.result[3,3]-p2.result[4,3]))
print(sprintf("The ATE for Texas is : %.2f and for Arkansas: %.2f",tx.ate,ar.ate))
```

```
## [1] "The ATE for Texas is : 16.74 and for Arkansas: 10.09"
```

c. Use equation (3.10) to estimate the overall ATE for both states combined.

```
ar.obs <- sum(p2.dt[,"texas0_arkansas1"])
overall.obs <- as.numeric(lengths(p2.dt[,"texas0_arkansas1"]))
tx.obs <- overall.obs - ar.obs
p2.overall.ate <- tx.ate * (tx.obs/overall.obs) + ar.ate * (ar.obs/overall.obs)
print(sprintf("The overall treatment effect for a 4-year term compared to a 2-year term is: %.2f",p2.ove
```

```
## [1] "The overall treatment effect for a 4-year term compared to a 2-year term is: 13.22"
```

d. Explain why, in this study, simply pooling the data for the two states and comparing the average number of bills introduced by two-year senators to the average number of bills introduced by four-year senators leads to biased estimate of the overall ATE.

In this experiment, the state acts as a confounding variable. Texas is a much larger and more populous state than Arkansas, so it is logical that they would produce more legistation regardless of the term length treatment. Using blocking is a logical way to eliminate bias that might arise if the treatment was distributed unevenly between states.

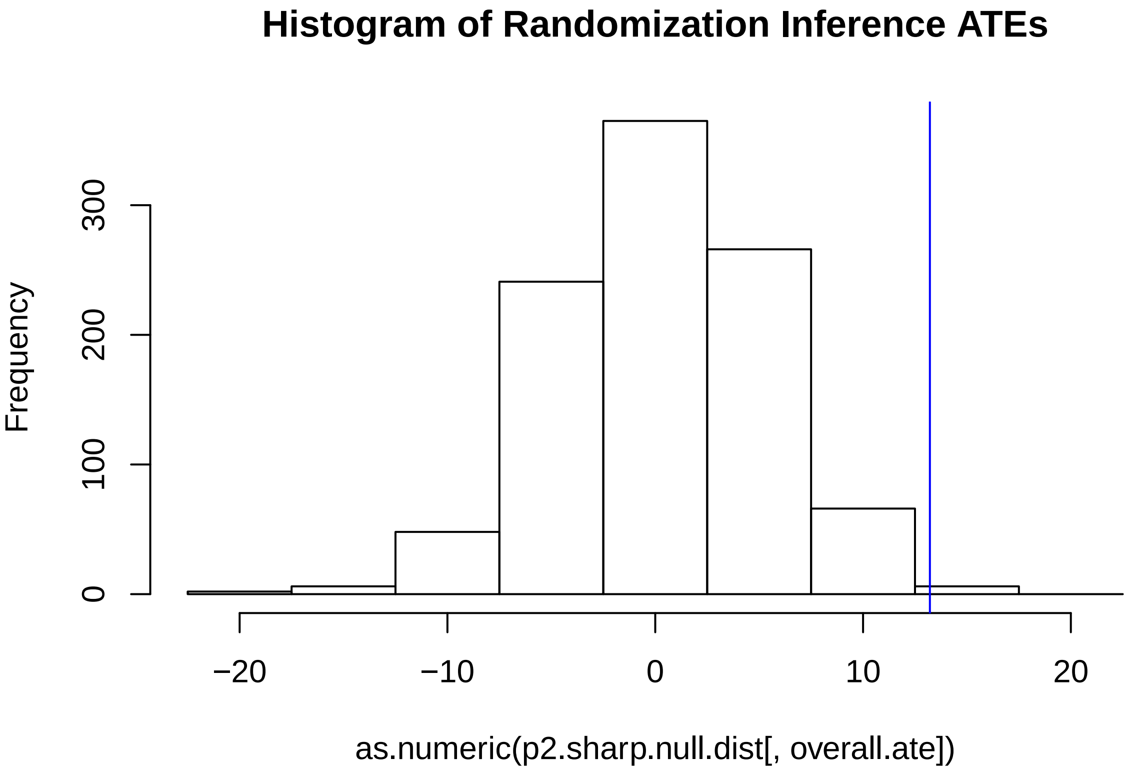
e. Insert the estimated standard errors into equation (3.12) to estimate the standard error for the overall ATE.

```
tx.twoyear <- p2.dt[term2year==1 & texas0_arkansas1==0]
tx.fouryear <- p2.dt[term2year==0 & texas0_arkansas1==0]
tx.twoyear.obs <- as.numeric(tx.twoyear[,sum(term2year)])
tx.se <- ((var(tx.twoyear[,bills_introduced])/tx.twoyear.obs)+(var(tx.fouryear[,bills_introduced])/(tx.

ar.twoyear <- p2.dt[term2year==1 & texas0_arkansas1==1]
ar.fouryear <- p2.dt[term2year==0 & texas0_arkansas1==1]
ar.twoyear.obs <- as.numeric(ar.twoyear[,sum(term2year)])
ar.se <- ((var(ar.twoyear[,bills_introduced])/ar.twoyear.obs)+(var(ar.fouryear[,bills_introduced])/(ar.
overall.se <- ( tx.se^2 * (tx.obs/overall.obs)^2 + ar.se^2 * (ar.obs/overall.obs)^2 )^0.5
print(sprintf("The overall standard error for the experiment is: %.2f", overall.se))
```

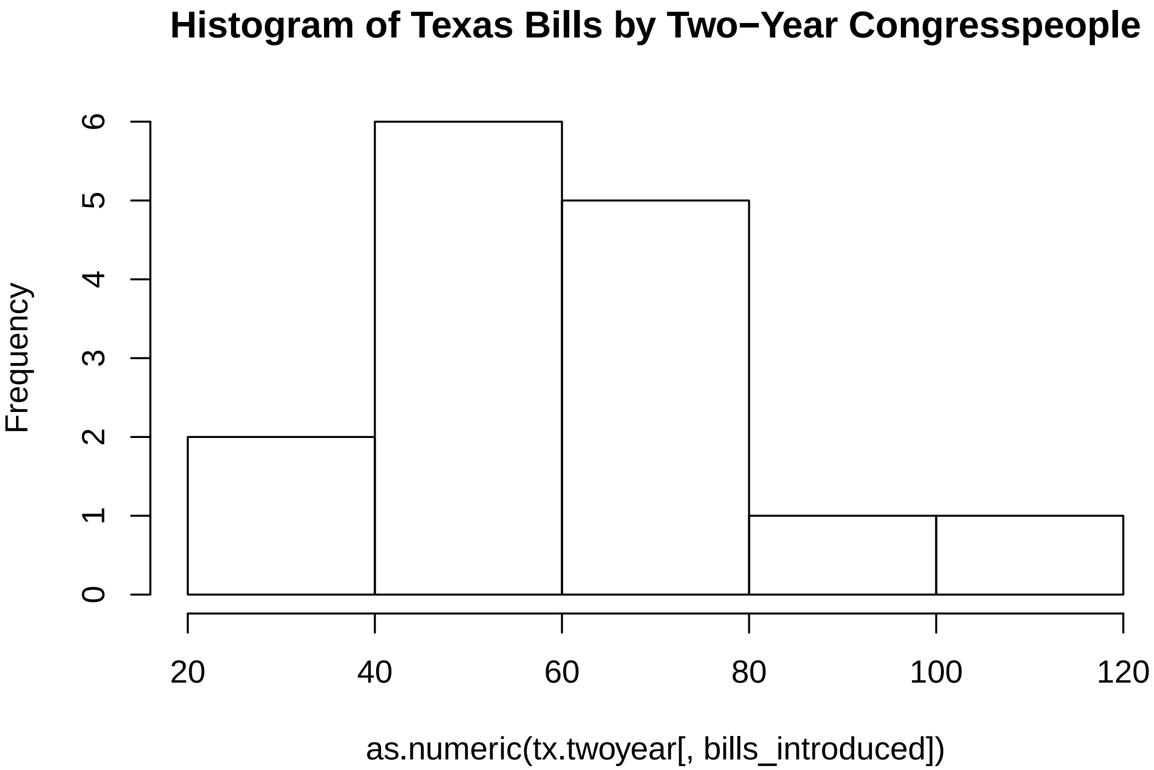
```
## [1] "The overall standard error for the experiment is: 4.74"
```

f. Use randomization inference to test the sharp null hypothesis that the treatment effect is zero for senators in both states. g. **IN Addition:** Plot histograms for both the treatment and control groups in each state (for 4 histograms in total).

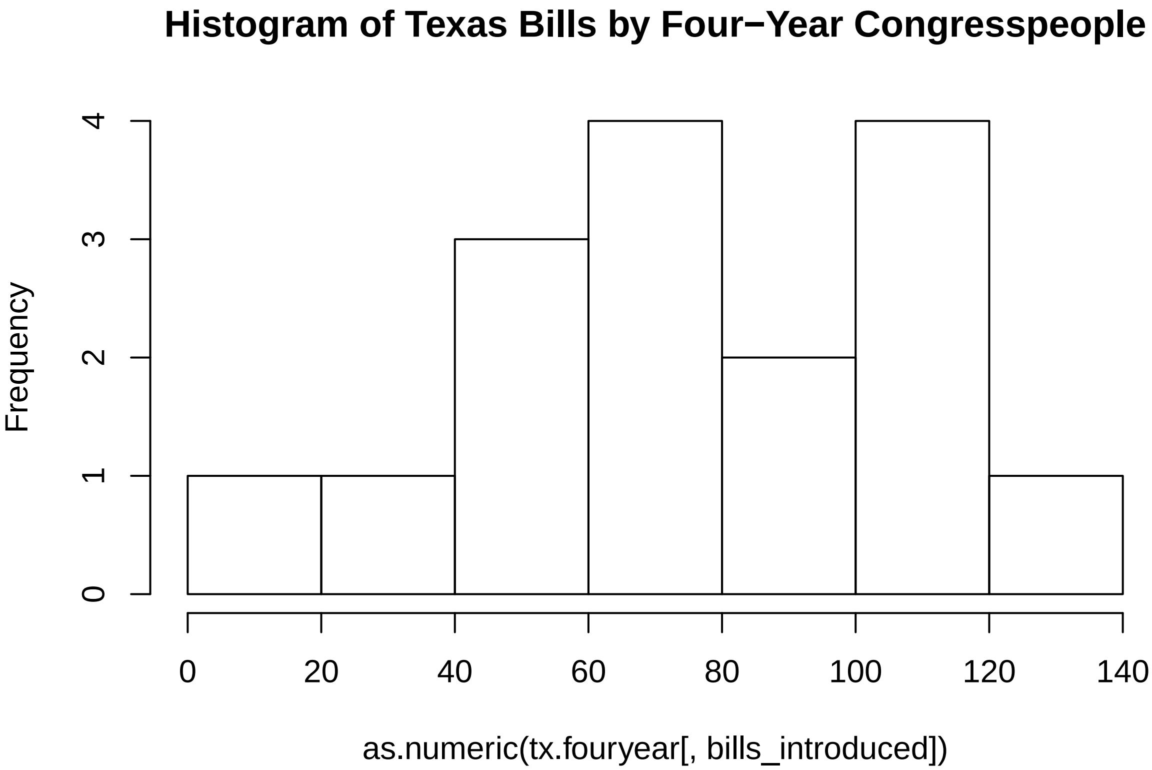
```
ri.blocking <- function(block1,block2) {
  b1.n <- nrow(block1)
  b1.sample <- sample(b1.n,b1.n/2)
  b1.t1 <- block1[b1.sample,mean(bills_introduced)]
  b1.t2 <- block1[!b1.sample,mean(bills_introduced)]

  b2.n <- nrow(block2)
  b2.sample <- sample(b2.n,b2.n/2)
  b2.t1 <- block2[b2.sample,mean(bills_introduced)]
  b2.t2 <- block2[!b2.sample,mean(bills_introduced)]

  overall.n <- b1.n + b2.n
  b1.ate <- b1.t2 - b1.t1
  b2.ate <- b2.t2 - b2.t1
  overall.ate <- b1.ate * (b1.n/overall.n) + b2.ate * (b2.n/overall.n)
  return (list("b1.t1" = b1.t1, "b1.t2" = b1.t2, "b2.t1" = b2.t1, "b2.t2" = b2.t2, "overall.ate" = overa
}

num_trials <- 1000
tx.block <- p2.dt[texas0_arkansas1==0]
ar.block <- p2.dt[texas0_arkansas1==1]
p2.sharp.null.dist <- replicate(num_trials,ri.blocking(block1 = tx.block, block2 = ar.block))
p2.sharp.null.dist <- data.table::data.table(t(p2.sharp.null.dist))
hist(as.numeric(p2.sharp.null.dist[,overall.ate]),main = "Histogram of Randomization Inference ATEs",bre
abline(v = p2.overall.ate, col = "blue")
```



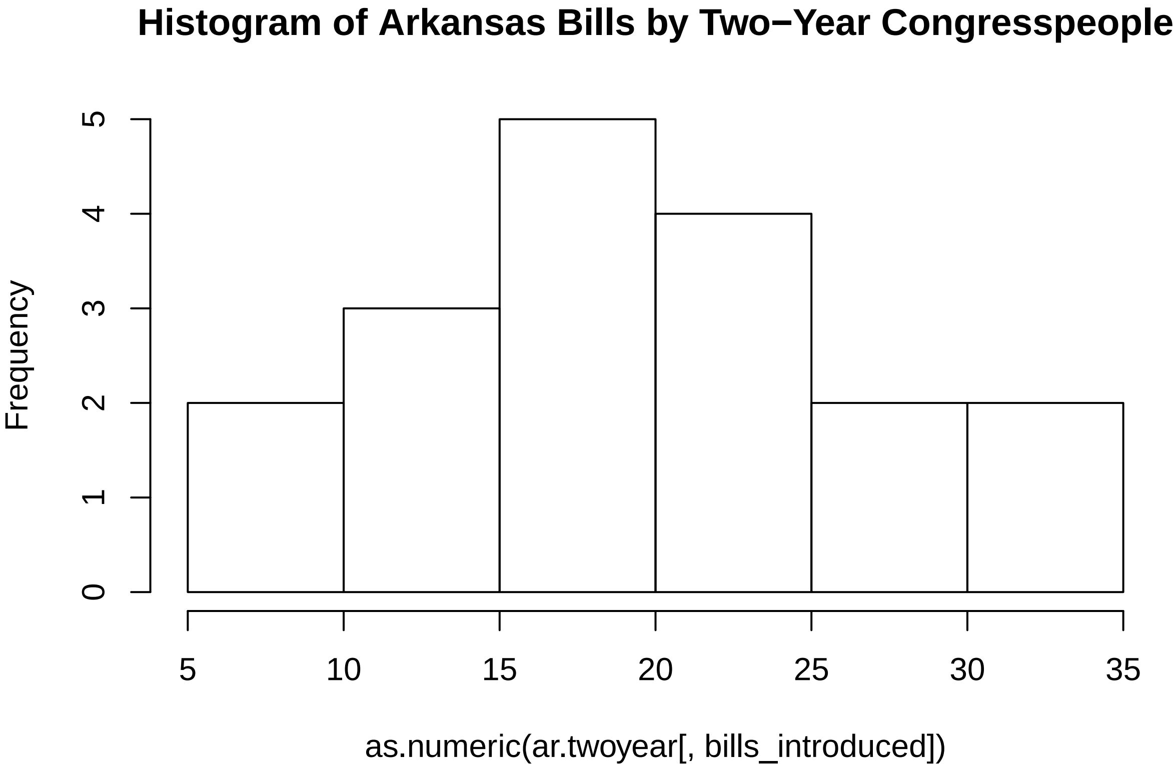
```
hist(as.numeric(tx.twoyear[,bills_introduced]),main = "Histogram of Texas Bills by Two-Year Congresspeop
```



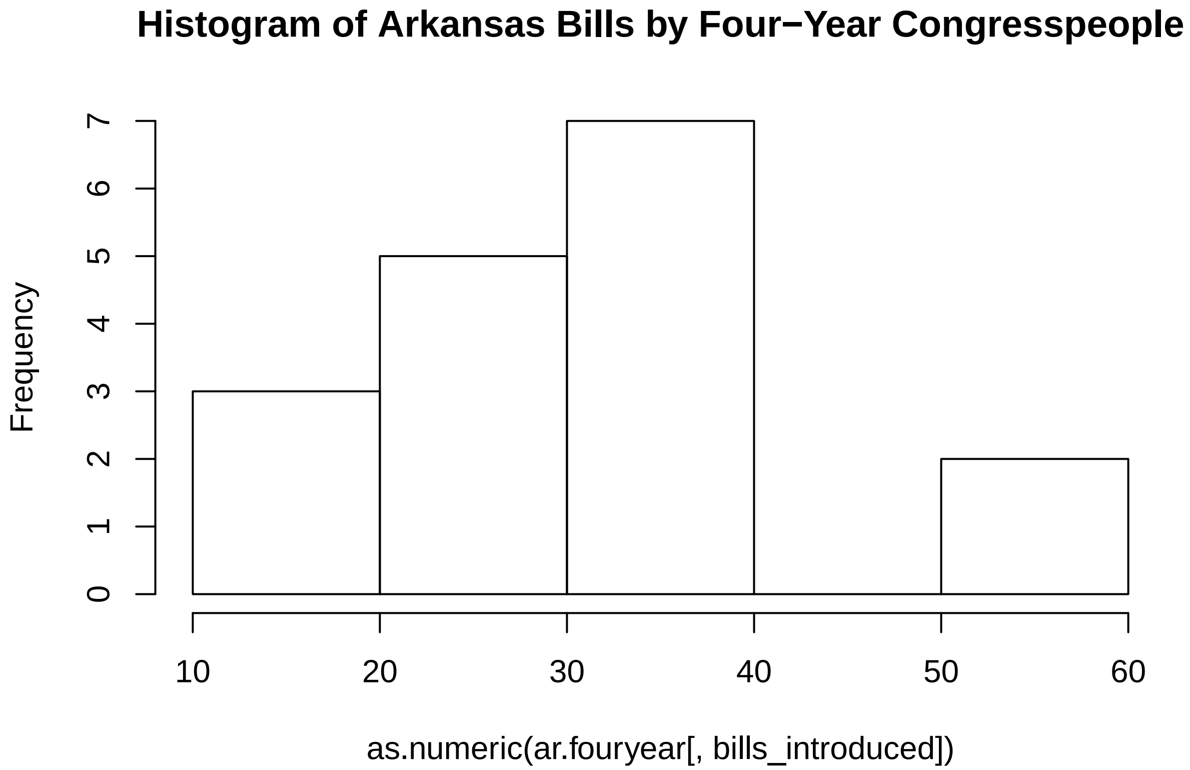
```
hist(as.numeric(tx.fouryear[,bills_introduced]),main = "Histogram of Texas Bills by Four-Year Congresssp
```



```
hist(as.numeric(ar.twoyear[,bills_introduced]),main = "Histogram of Arkansas Bills by Two-Year Congressp
```



```
hist(as.numeric(ar.fouryear[,bills_introduced]),main = "Histogram of Arkansas Bills by Four-Year Congres
```



3. Cluster Randomization

Use the data in *Field Experiments* Table 3.3 to simulate cluster randomized assignment. (*Notes: (a) Assume 3 clusters in treatment and 4 in control; and (b) When Gerber and Green say* simulate'', they do not meanrun *simulations with R code’‘, but rather, in a casual sense “take a look at what happens if you do this this way.” There is no randomization inference necessary to complete this problem.*)

```
## load data
p3.d <- read.csv("./experiments-causality/assignments/PS2-upstream/data/ggChapter3.csv")
head(p3.d,n=5)
```

```
##   Village Y D Block
## 1       1 0 0     1
## 2       2 1 0     1
## 3       3 2 1     1
## 4       4 4 2     1
## 5       5 4 0     1
```

```
p3.dt <- data.table::data.table(p3.d)
```

a. Suppose the clusters are formed by grouping observations 1,2, 3,4, 5,6, ... , 13,14. Use equation (3.22) to calculate the standard error assuming half of the clusters are randomly assigned to treatment.

```
p3.clusters <- list(1,1,2,2,3,3,4,4,5,5,6,6,7,7)
p3.sample <- sample(unique(p3.clusters),length(unique(p3.clusters))/2)
p3.treatment <- is.element(p3.clusters, p3.sample)
p3.k <- as.numeric(length(unique(p3.clusters)))
p3.N <- as.numeric(length(p3.clusters))
```

```
p3.m <- as.numeric(sum(p3.treatment))
p3.dt.a <- cbind(p3.dt,data.table::data.table("cluster"=p3.clusters,"treatment"=p3.treatment))
p3.outcomes.treatment <- p3.dt.a[p3.treatment,c("Y","cluster")]
p3.outcomes.treatment.means <- tapply(p3.outcomes.treatment$Y, as.numeric(p3.outcomes.treatment$cluster
p3.outcomes.control <- p3.dt.a[p3.treatment,c("Y","cluster")]
p3.outcomes.control.means <- tapply(p3.outcomes.control$Y, as.numeric(p3.outcomes.control$cluster), mea
p3.se.a <- ( ( 1/(p3.k-1) ) * ( ( p3.m*var(p3.outcomes.control.means) / (p3.N-p3.m) ) + ( ( (p3.N-p3.m)
print(sprintf("The standard error for the clustering described in part a is: %.2f",p3.se.a))
```

```
## [1] "The standard error for the clustering described in part a is: 5.38"
```

b. Suppose that clusters are instead formed by grouping observations 1,14, 2,13, 3,12, ... , 7,8. Use equation (3.22) to calculate the standard error assuming half of the clusters are randomly assigned to treatment.

```
p3.clusters <- list(1,2,3,4,5,6,7,7,6,5,4,3,2,1)
p3.sample <- sample(unique(p3.clusters),length(unique(p3.clusters))/2)
p3.treatment <- is.element(p3.clusters, p3.sample)
p3.k <- as.numeric(length(unique(p3.clusters)))
p3.N <- as.numeric(length(p3.clusters))
p3.m <- as.numeric(sum(p3.treatment))
p3.dt.b <- cbind(p3.dt,data.table::data.table("cluster"=p3.clusters,"treatment"=p3.treatment))
p3.outcomes.treatment <- p3.dt.b[p3.treatment,c("Y","cluster")]
p3.outcomes.treatment.means <- tapply(p3.outcomes.treatment$Y, as.numeric(p3.outcomes.treatment$cluster
p3.outcomes.control <- p3.dt.b[p3.treatment,c("Y","cluster")]
p3.outcomes.control.means <- tapply(p3.outcomes.control$Y, as.numeric(p3.outcomes.control$cluster), mea
p3.se.b <- ( ( 1/(p3.k-1) ) * ( ( p3.m*var(p3.outcomes.control.means) / (p3.N-p3.m) ) + ( ( (p3.N-p3.m)
print(sprintf("The standard error for the clustering described in part b is: %.2f",p3.se.b))
```

```
## [1] "The standard error for the clustering described in part b is: 0.00"
```

c. Why do the two methods of forming clusters lead to different standard errors? What are the implications for the design of cluster randomized experiments?

The difference in standard error is due the outcome variable Y, which roughly increases linearly with the index. In grouping opposite indexes together, the difference between $\bar{Y}_j(0)$ and $\bar{Y}_j(1)$ is minimized.

4. Sell Phones?

You are an employee of a newspaper and are planning an experiment to demonstrate to Apple that online advertising on your website causes people to buy iPhones. Each site visitor shown the ad campaign is exposed to $0.10 worth of advertising for iPhones. (Assume all users could see ads.) There are 1,000,000 users available to be shown ads on your newspaper's website during the one week campaign.

Apple indicates that they make a profit of $100 every time an iPhone sells and that 0.5% of visitors to your newspaper's website buy an iPhone in a given week in general, in the absence of any advertising.

a. By how much does the ad campaign need to increase the probability of purchase in order to be "worth it" and a positive ROI (supposing there are no long-run effects and all the effects are measured within that week)?

Break even point: add spending <= revenue gain p4.n_users * p4.add_spend <= p4.n_users * (p4.conversion.final - p4.conversion.initial) * p4.profit_margin

```
p4.n_users <- 1000000
p4.conversion.initial <- 0.5
p4.profit_margin <- 100
p4.add_spend <- 0.10
p4.conversion.final <- (p4.n_users * p4.add_spend) / (p4.n_users * p4.profit_margin) + p4.conversion.ini
print(sprintf("Apple would need the rate at which users purchase iPhones to increase from %.4f to %.4f
```

```
## [1] "Apple would need the rate at which users purchase iPhones to increase from 0.5000 to 0.5010 in
```

b. Assume the measured effect is 0.2 percentage points. If users are split 50:50 between the treatment group (exposed to iPhone ads) and control group (exposed to unrelated advertising or nothing; something you can assume has no effect), what will be the confidence interval of your estimate on whether people purchase the phone?

```
p4.conversion.b <- p4.conversion.initial + 0.2
p4.n_control.b <- p4.n_users*0.5
p4.n_treatment.b <- p4.n_users*0.5
p4.p.b <- ( p4.conversion.initial*p4.n_control.b + p4.conversion.b*p4.n_treatment.b ) / (p4.n_control.b
p4.se.b <- ( p4.p.b*(1-p4.p.b) * ( (1/p4.n_control.b) + (1/p4.n_treatment.b) ) )^0.5
print(sprintf("The standard error for the experiment described in part b is : %.6f", p4.se.b))
```

```
## [1] "The standard error for the experiment described in part b is : 0.000980"
```

```
print(sprintf("The 95 percent confidence interval for the experiment described in part b is : %.6f to %
```

```
## [1] "The 95 percent confidence interval for the experiment described in part b is : 0.698080 to 0.70
```

+ **Note:** The standard error for a two-sample proportion test is $\sqrt{p(1-p) * (\frac{1}{n_1} + \frac{1}{n_2})}$ where $p = \frac{x_1 + x_2}{n_1 + n_2}$, where x and n refer to the number of "successes" (here, purchases) over the number of "trials" (here, site visits). The length of each tail of a 95

c. Is this confidence interval precise enough that you would recommend running this experiment? Why or why not?

The lower bound of the confidence interval is well above the break-even point for this advertising campaign, and the interval is very narrow which indicates a high degree of confidence in the projection. It is very very unlikely that this advertising campaign would lose the company money, so I would strongly reccomend that we follow through on it.

d. Your boss at the newspaper, worried about potential loss of revenue, says he is not willing to hold back a control group any larger than 1

```
p4.conversion.d <- p4.conversion.initial + 0.2
p4.n_control.d <- p4.n_users*0.01
p4.n_treatment.d <- p4.n_users*0.99
p4.p.d <- ( p4.conversion.initial*p4.n_control.d + p4.conversion.d*p4.n_treatment.d ) / (p4.n_control.d
p4.se.d <- ( p4.p.d*(1-p4.p.d) * ( (1/p4.n_control.d) + (1/p4.n_treatment.d) ) )^0.5
print(sprintf("The standard error for the experiment described in part b is : %.6f", p4.se.d))
```

```
## [1] "The standard error for the experiment described in part b is : 0.004614"
```

```
print(sprintf("The 95 percent confidence interval for the experiment described in part b is : %.6f to %
```

```
## [1] "The 95 percent confidence interval for the experiment described in part b is : 0.690956 to 0.70
```

5. Sports Cards

Here you will find a set of data from an auction experiment by John List and David Lucking-Reiley (2000).

```
p5.d2 <- read.csv("./experiments-causality/assignments/PS2-upstream/data/listData.csv")
head(p5.d2)
```

```
##   bid uniform_price_auction
## 1   5                     1
## 2   5                     1
## 3  20                     0
## 4   0                     1
## 5  20                     1
## 6   0                     1
```

In this experiment, the experimenters invited consumers at a sports card trading show to bid against one other bidder for a pair trading cards. We abstract from the multi-unit-auction details here, and simply state that the treatment auction format was theoretically predicted to produce lower bids than the control auction format. We provide you a relevant subset of data from the experiment.

a. Compute a 95

```
t.test(p5.d2$bid[p5.d2$uniform_price_auction==0],p5.d2$bid[p5.d2$uniform_price_auction==1])
```

```
##
##  Welch Two Sample t-test
##
## data:  p5.d2$bid[p5.d2$uniform_price_auction == 0] and p5.d2$bid[p5.d2$uniform_price_auction == 1]
## t = 2.8211, df = 61.983, p-value = 0.006421
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   3.557141 20.854624
## sample estimates:
## mean of x mean of y
##  28.82353  16.61765
```

b. In plain language, what does this confidence interval mean?

We can reject the null hypothesis with greater than 99% confidence. Additionally we see that the 95% confidence interval is that uniform price auction increases bids between 3.5 and 20.9 dollars. While this is a useful result, the large confidence interval suggests that we should conduct a larger experiment to determine the effect of uniform price auctions more precisely.

c. Regression on a binary treatment variable turns out to give one the same answer as the standard analytic formula you just used. Demonstrate this by regressing the bid on a binary variable equal to 0 for the control auction and 1 for the treatment auction.

```
p5.model <- lm(formula = bid ~ uniform_price_auction, data = p5.d2)
summary(p5.model)
```

```
##
## Call:
## lm(formula = bid ~ uniform_price_auction, data = p5.d2)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max 
## -28.824 -11.618  -3.221   8.382  58.382 
## 
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)    
## (Intercept)            28.824      3.059   9.421 7.81e-14 ***
## uniform_price_auction  -12.206      4.327  -2.821  0.00631 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 17.84 on 66 degrees of freedom
## Multiple R-squared:  0.1076, Adjusted R-squared:  0.09409 
## F-statistic: 7.959 on 1 and 66 DF,  p-value: 0.006315
```

d. Calculate the 95

```
confint(p5.model, "uniform_price_auction", level=0.95)
```

```
##                           2.5 %    97.5 %
## uniform_price_auction -20.84416 -3.567603
```

e. On to p-values. What p-value does the regression report? Note: please use two-tailed tests for the entire problem.

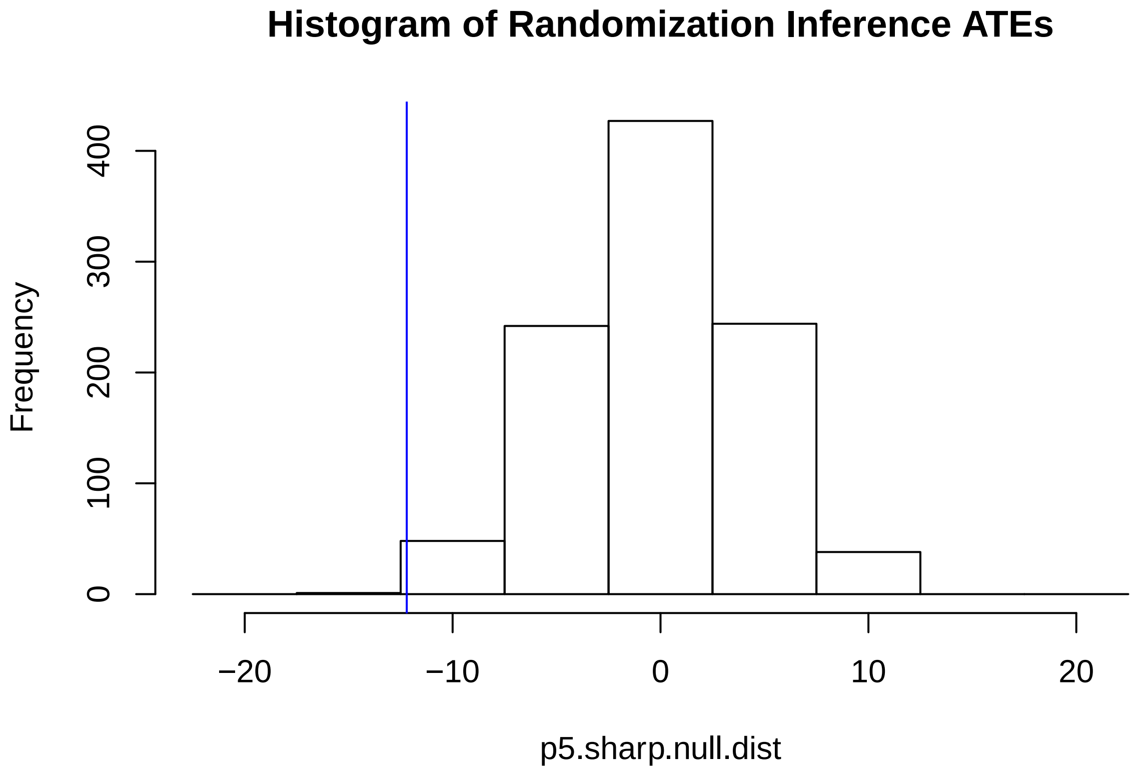
The linear regression reported the a p-value within one ten-thousandnth of the t-test: 0.006315

```
print(sprintf("The p-value for the linear regression model is: %.6f",as.numeric(summary(p5.model)$coeff
```

```
## [1] "The p-value for the linear regression model is: 0.006315"
```

f. Now compute the same p-value using randomization inference.

```
p5.ri <- function(data,outcome) {
  n <- nrow(data)
  s <- sample(n,n/2)
  ind <- logical(n)
  ind[s] <- TRUE
  treatment <- data[ind,outcome]
  control <- data[!ind,outcome]
  ate <- as.numeric(mean(treatment)-mean(control))
  return (ate)
}

p5.real.ate <- as.numeric(mean(p5.d2[p5.d2$uniform_price_auction==1,"bid"])-mean(p5.d2[p5.d2$uniform_pr
num_trials <- 1000 #change to 10,000 before final knit
p5.sharp.null.dist <- replicate(num_trials, p5.ri(data = p5.d2, outcome = "bid"))
hist(p5.sharp.null.dist, main = "Histogram of Randomization Inference ATEs",breaks = seq(-22.5,22.5,5))
abline(v = p5.real.ate, col = "blue")
```



```
print(sprintf("The two-tailed P-value for this experiment is: %.4f", mean(abs(p5.sharp.null.dist)>=abs(
```

```
## [1] "The two-tailed P-value for this experiment is: 0.0050"
```

g. Compute the same p-value again using analytic formulas for a two-sample t-test from your earlier statistics course. (Also see part (a).)

Not sure why my t-statistic calculation doesn't match up with the value produced by the t-test, but I ran out of time to troubleshoot it.

```
p5.t <- (mean(p5.d2$bid) - abs(p5.real.ate))/ (sd(p5.d2$bid) / sqrt(nrow(p5.d2)))
p5.manual_p <- 2 * pt(p5.t, nrow(p5.d2)-1, lower.tail = FALSE)
p5.t.from_t_test <- 2.8211
p5.semi_manual_p <- 2 * pt(p5.t.from_t_test, df = nrow(p5.d2)-1, lower.tail = FALSE)
print(sprintf("The manually computed p-value is: %.6f",p5.manual_p))
```

```
## [1] "The manually computed p-value is: 0.000018"
```

```
print(sprintf("The manually computed p-value borrowing the t-value from the t-test is: %.6f",p5.semi_man
```

```
## [1] "The manually computed p-value borrowing the t-value from the t-test is: 0.006291"
```

h. Compare the two p-values in parts (e) and (f). Are they much different? Why or why not? How might your answer to this question change if the sample size were different?

The p-value produced by randomization inference is about 30% larger than the p-value computed analytically. By design, the randomization inference is more sensitive to outliers or otherwise non-symetcial data. A larger sample size would improve the accuraccy of the analytical p-value by the cental limit theorem.