Problem Set 2

Shelly Hsu

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.

```
d <- read.csv("./data/Clingingsmith.2009.csv", stringsAsFactors = FALSE)
head(d)
```

```
##   success views_saudi views_indonesian views_turkish views_african
## 1       0           1                1             0             0
## 2       0           1                1             0            -1
## 3       0           0                0             0             0
## 4       0           2                2             0             0
## 5       0           1                1             1             1
## 6       0           2                0             0             0
##   views_chinese views_european views
## 1             0              0     2
## 2             1             -1     1
## 3             0              0     0
## 4             1              0     5
## 5             1             -2     3
## 6             0              0     2
```

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.

```
s <- data.table(d)
s[, .(avg_views=mean(views)), by = success]
```

```
##    success avg_views
## 1:       0  1.868304
## 2:       1  2.343137
```

Views appear to be more positive among those who were sucessful at winning the lottery than those who did not win the lottery.

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.)

```
set.seed(1000)
# Calculate ATE
```

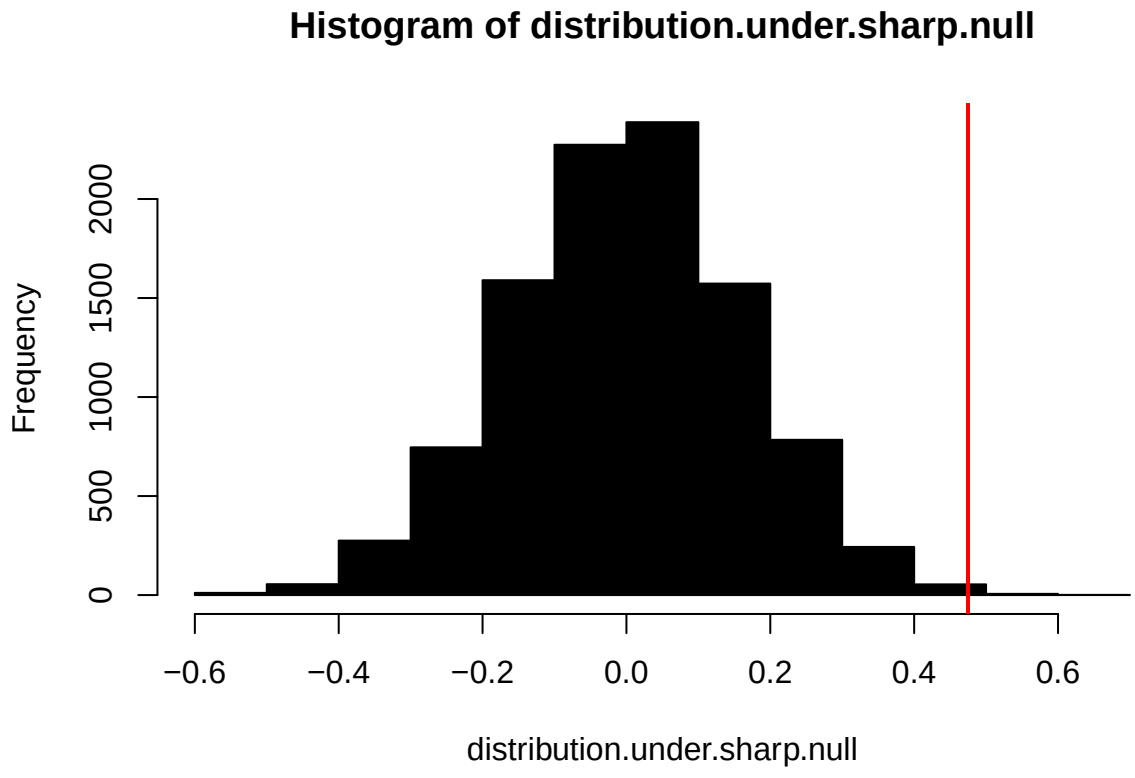
```r
ate <- mean(d$views[d$success==1]) - mean(d$views[d$success == 0])
ate
```

```
## [1] 0.4748337
```

```r
cases <- c(0,1)
randomize <- function(group) {
  sample(cases, size = group, replace = TRUE)
  }


estimate.ate <- function(d){
  treatment <- randomize(length(d$success))
  return(mean(d$views[treatment==1]) - mean(d$views[treatment == 0]))
}

distribution.under.sharp.null<-replicate(10000, estimate.ate(d))
m <- mean(ate <=distribution.under.sharp.null)
m
```

```
## [1] 0.001
```

```r
hist(distribution.under.sharp.null, col = 'black')
abline(v = ate, col = 'red', lwd = 2)
```



This is a meaningful difference. It seems likely that the value was not generated by randomization noise since the p-value is sufficiently small. The true effect is on the tail ends of the distribution suggesting that we can reject the sharp null hypothesis that there is no difference

in the treatment effect.

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?

```
s_ate <- sum(distribution.under.sharp.null >= ate)
s_ate
```

```
## [1] 10
```

18 of the simulated random assignments generated an estimated ATE that was at least as large as the actual estimate of the ATE.

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

```
m <- mean(distribution.under.sharp.null >= ate)
m
```

```
## [1] 0.001
```

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?

```
s_abs_ate <- sum(abs(distribution.under.sharp.null) >= ate)
s_abs_ate
```

```
## [1] 29
```

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

```
abs_p_value <- mean(abs(distribution.under.sharp.null) >= ate)
abs_p_value
```

```
## [1] 0.0029
```

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)

d <- read.dta("./data/Titiunik.2010.dta")
head(d)
```

```
##   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
```

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?

```
b <- data.table(d)
b[, .(avg_bills=mean(bills_introduced)), by = texas0_arkansas1]
```

```
##    texas0_arkansas1 avg_bills
## 1:                0  68.77419
## 2:                1  25.51429
```

It appears as though Texas is more productive than Arkansas since it has has more bills on average compared to Arkansas when grouping by state.

```
b[, .(avg_bills=mean(bills_introduced)), by = term2year]
```

```
##    term2year avg_bills
## 1:         0  53.09091
## 2:         1  38.57576
```

When grouping by term year it appears that 4-year terms are more productive 2-year terms since it has more average bills than 2 year terms.

```
b[, .(avg_bills=mean(bills_introduced)), by = list(term2year,texas0_arkansas1)]
```

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

When grouping by state and term-length Texas is still more productive than Arkansas when comparing to the respective term lengths. The average treatment effect for Arkansas is smaller than the effect for Texas. The general direction of the treatment effect was not changed from what we learned after grouping by term length and state, however, more detail for the magnitude of the effects is revealed. For both Texas and Arkansas there were more average bills in the 4-year term than the 2-year term. Arkansas however, has a lower estimated ATE than Texas when comparing between term years. There is no enough data to determine whether there is a casual relationship between the variables. Based on the results we can only say that there is a possible association between term length and bills introduced and possibly state and number of bills introduced.

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

```
tex<-subset(d,texas0_arkansas1==0)
ark<-subset(d,texas0_arkansas1==1)

#table(d$term2year, d$texas0_arkansas1)

t_treat<-sum(tex$term2year==1)
#t_treat
t_control<-sum(tex$term2year==0)
#t_control
a_treat<-sum(ark$term2year==1)
```

```
#a_treat
a_control<-sum(ark$term2year==0)
#a_control

se_tex <-sqrt(var(tex$bills_introduced[tex$term2year==0])/
                t_control+var(tex$bills_introduced[tex$term2year==1])/t_treat)
se_ark <-sqrt(var(ark$bills_introduced[ark$term2year==0])/
                a_control+var(ark$bills_introduced[ark$term2year==1])/a_treat)
se_tex #se for texas
```

```
## [1] 9.345871
```

```
se_ark #se for arkansas
```

```
## [1] 3.395979
```

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

```
n <- sum(b$texas0_arkansas1==1)
#n
m <-sum(b$texas0_arkansas1==0)
#m
tot <-nrow(b)
#tot
tex_ate <-60.13333-76.87500
tex_ate
```

```
## [1] -16.74167
```

```
ark_ate <-20.61111-30.70588
ark_ate
```

```
## [1] -10.09477
```

```
overall_ate <- n/tot*ark_ate+m/tot*tex_ate
overall_ate
```

```
## [1] -13.2168
```

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.

Pooling the estimates together will affect the amount of variance in the sample. It will introduce more variance into the sample by not accounting for the different groupings that are found within the pooled sample. When the groupings are sparated out they have their own variances inherent within the grouping pooling does not account for all the variation. It may also be possible that pooling may confound the effects by aggregating the differences between the level and will obscure the true effect when one grouping is larger than the other so pooling biases the results and give an inaccurate ATE. *

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

```
se_ate <- sqrt(se_tex^2*(m/tot)^2+se_ark^2*(n/tot)^2)
se_ate
```

```
## [1] 4.74478
```

f. Use randomization inference to test the sharp null hypothesis that the treatment effect is zero for senators in both states.

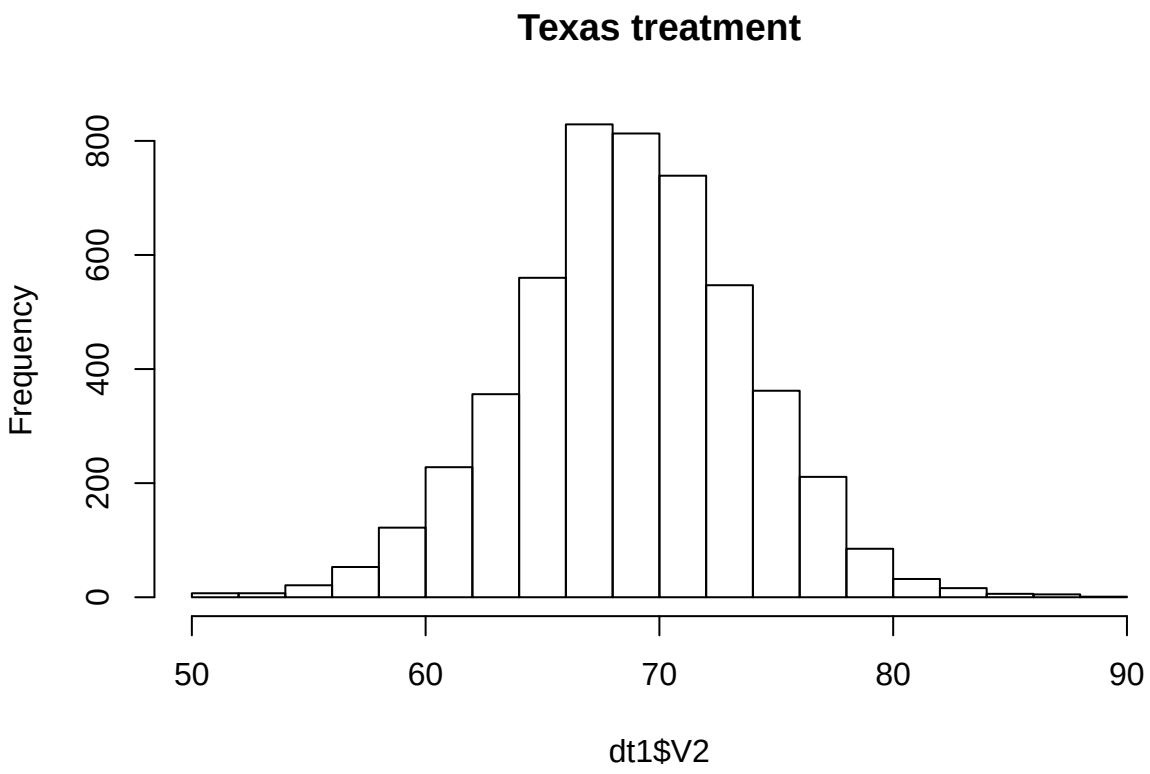
```
cases <- c(0,1)
randomize <- function(group) {
  sample(cases, size = group, replace = TRUE)
  }


r_ate<-function(b,N1,N2) {

  texas_treatment <- randomize(N1)
  arkansas_treatment <- randomize(N2)

  texas.treatment.po <- b[texas0_arkansas1==0,(bills_introduced)][texas_treatment==1]
  texas.control.po <- b[texas0_arkansas1==0,(bills_introduced)][texas_treatment==0]

  arkansas.treatment.po <- b[texas0_arkansas1==1,
                             (bills_introduced)][arkansas_treatment==1]
  arkansas.control.po <- b[texas0_arkansas1==1,
                           (bills_introduced)][arkansas_treatment==0]

  mean.texas.treatment.po <- mean(texas.treatment.po)
  mean.texas.control.po <- mean(texas.control.po)
  mean.arkansas.treatment.po<- mean(arkansas.treatment.po)
  mean.arkansas.control.po<-mean(arkansas.control.po)

  mean.texas.po <- mean.texas.treatment.po - mean.texas.control.po

  mean.arkansas.po <- mean.arkansas.treatment.po - mean.arkansas.control.po

  overall_ate <- mean.texas.po * N1/(N1+N2) + mean.arkansas.po * N2/(N1+N2)
  return (list(overall_ate,mean.texas.treatment.po,mean.texas.control.po,
               mean.arkansas.treatment.po,mean.arkansas.control.po))
}
```

```
dt1 <- data.table()
for(i in c(1:5000)){
  dt1<-rbindlist(list(dt1,(r_ate(b,b[texas0_arkansas1==0,.N],b[texas0_arkansas1==1,.N]))))
}

p_value <- mean(overall_ate >= dt1$V1) #p-value
p_value
```

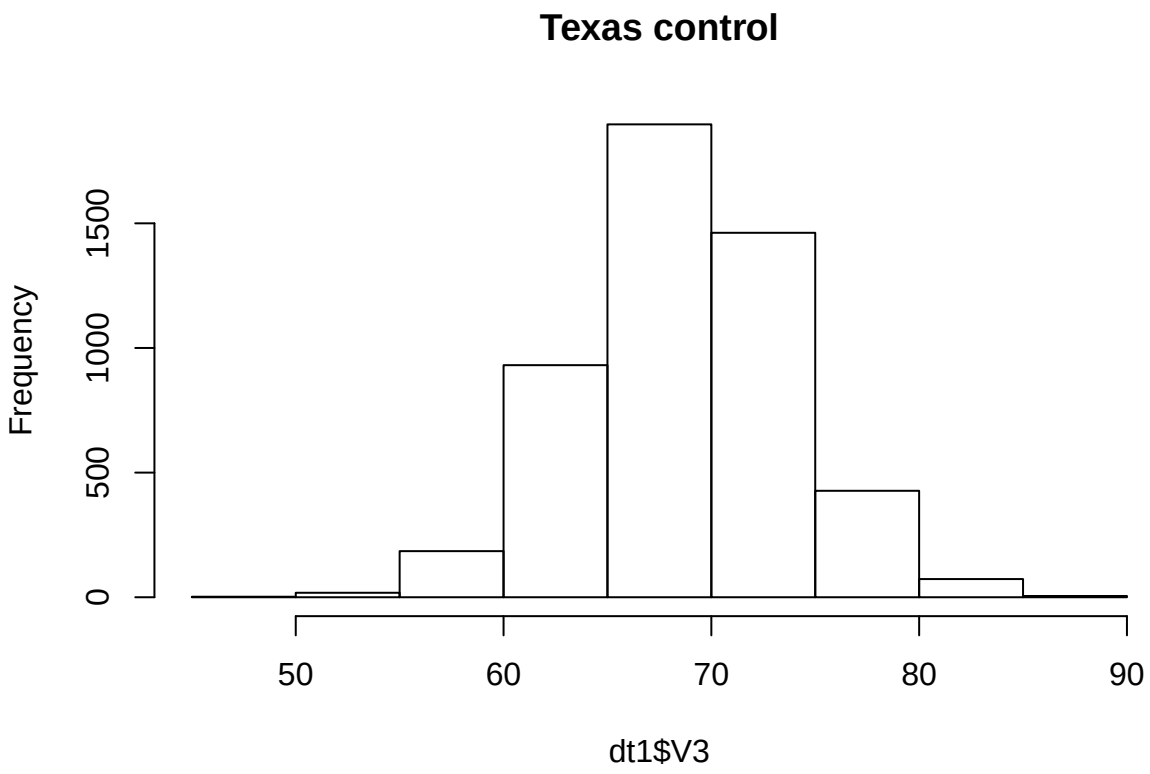
```
## [1] 0.0046
```

g. **IN Addition:** Plot histograms for both the treatment and control groups in each state (for 4 histograms in total).

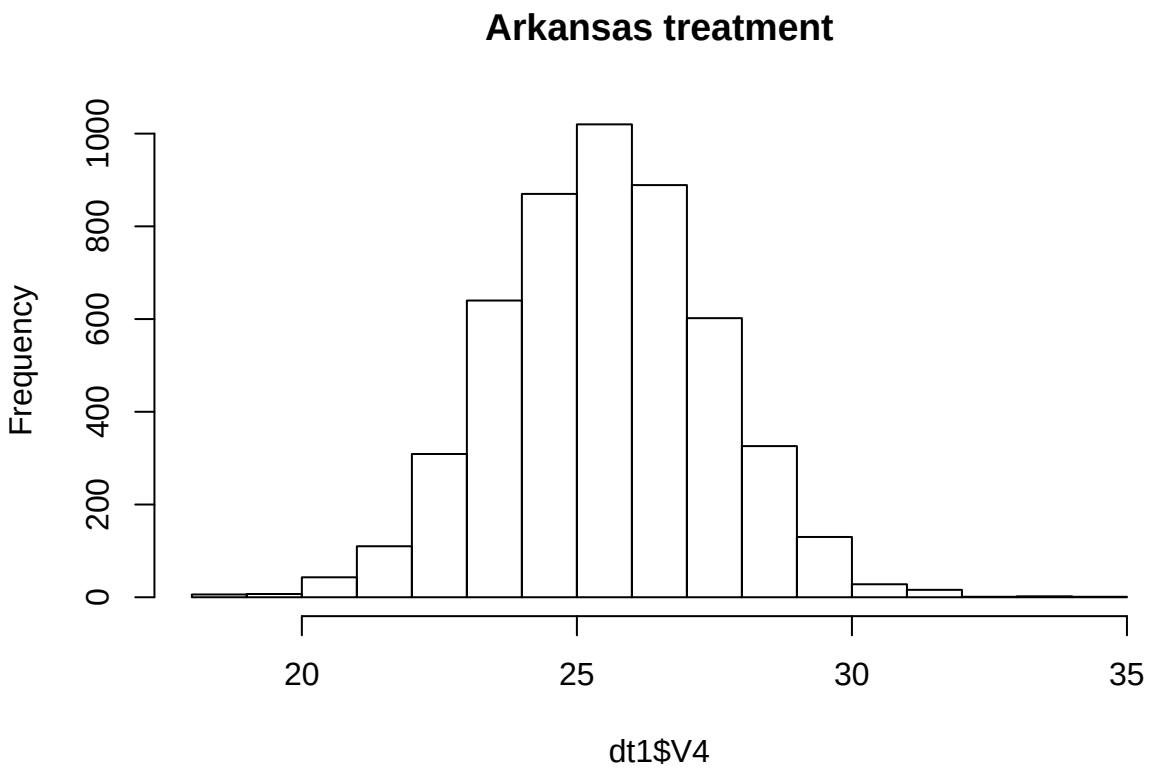
```
hist(dt1$V2, main = 'Texas treatment')
```



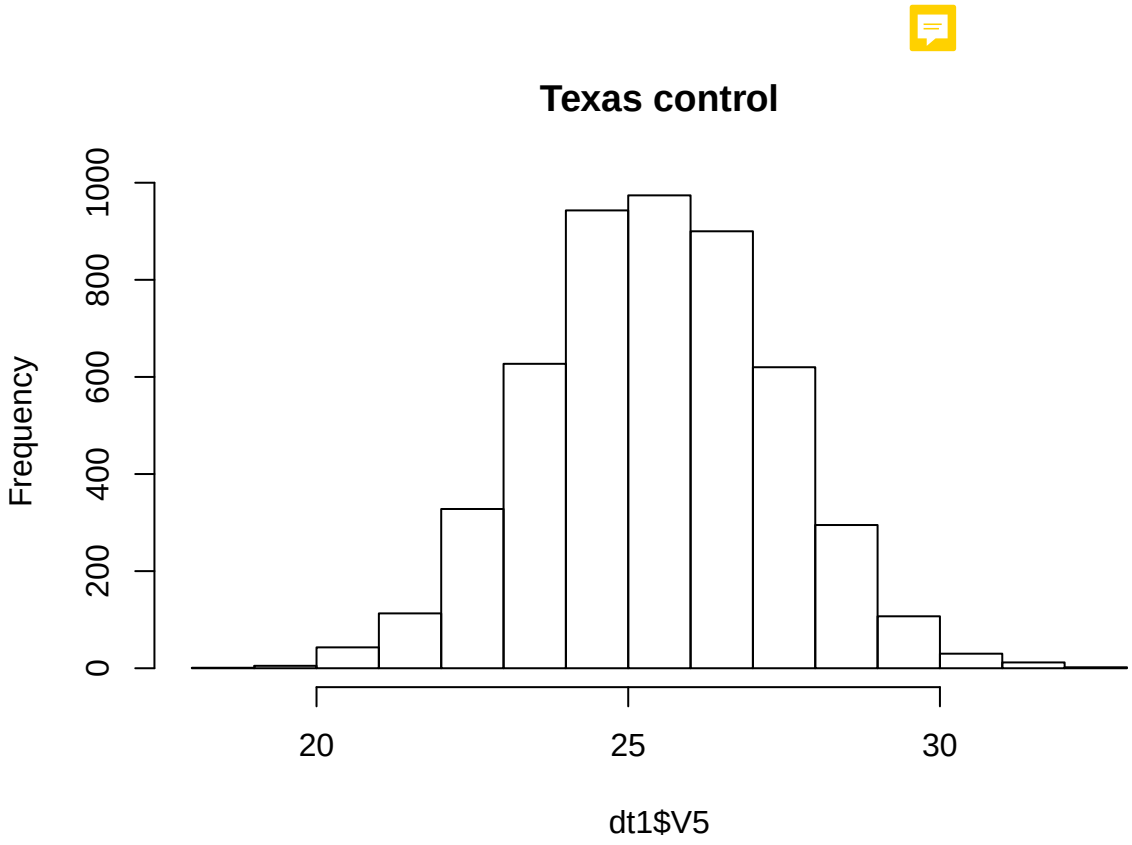

```
hist(dt1$V3, main = 'Texas control')
```




```
hist(dt1$V4, main = 'Arkansas treatment')
```




```
hist(dt1$V5, main = 'Texas control')
```



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 mean****run 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
d1 <- read.csv("./data/ggChapter3.csv", stringsAsFactors = FALSE)
summary(d1)
```

```
##     Village            Y               D             Block
##  Min.   : 1.00   Min.   : 0.000   Min.   : 0.000   Min.   :1.000
##  1st Qu.: 4.25   1st Qu.: 4.000   1st Qu.: 0.250   1st Qu.:1.000
##  Median : 7.50   Median : 7.500   Median : 2.500   Median :1.000
##  Mean   : 7.50   Mean   : 9.143   Mean   : 5.286   Mean   :1.429
##  3rd Qu.:10.75   3rd Qu.:15.750   3rd Qu.: 8.750   3rd Qu.:2.000
##  Max.   :14.00   Max.   :18.000   Max.   :17.000   Max.   :2.000
```

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.

```
#Forming clusters

c_list <- list(c(1,2), c(3,4), c(5,6) , c(7,8), c(9,10), c(11,12) , c(13,14))
```

```
# Set size of clusters
num_clusters = 7
num_treatment_clusters =3
num_control_clusters =4
num_total_records=14
num_treatment_records = 2*3


#Get outcomes for control and treatment from datasets

control_list=list()
treatment_list = list()

index=0
for (ele in c_list){
  index = index +1

  v_index1 = ele[1:1]
  v_index2 = ele[2:2]

  po_y0_v1 = d1[v_index1,"Y"]
  po_y0_v2 = d1[v_index2,"Y"]

  mean_po_y0 = mean(po_y0_v1,po_y0_v2)

  control_list[[index]] = mean(mean_po_y0)

  po_y1_v1 = d1[v_index1,"D"]
  po_y1_v2 = d1[v_index2,"D"]

  mean_po_y1 = mean(po_y1_v1,po_y1_v2)

  treatment_list[[index]] = mean(mean_po_y1)

}


### Calculating variance of clusters for control
var_control = var(unlist(control_list))
var_treatment = var(unlist(treatment_list))

#Calculating SE(ATE)

m_var_y0 = num_treatment_records*var_control
n_minus_m = num_total_records - num_treatment_records
n_minus_m_var_y1 = n_minus_m * var_treatment

#Cluster1
cov_y0_y1 = cov(unlist(control_list),unlist(treatment_list))
se_ate1 = sqrt(
  (1/(num_clusters-1))* (m_var_y0/n_minus_m +
                           n_minus_m*var_treatment/num_treatment_clusters +
                           2 * cov_y0_y1)
```

```
)

se_ate1
```

```
## [1] 4.94072
```

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.

```
c_list <- list(c(1,14), c(2,13), c(3,12) , c(4,11), c(5,10), c(6,9) , c(7,8))


# Set size of clusters
num_clusters = 7
num_treatment_clusters =3
num_control_clusters =4
num_total_records=14
num_treatment_records = 2*3

#Get outcomes for control and treatment from datasets

control_list=list()
treatment_list = list()

index=0
for (ele in c_list){
  index = index +1

  v_index1 = ele[1:1]
  v_index2 = ele[2:2]

  po_y0_v1 = d1[v_index1,"Y"]
  po_y0_v2 = d1[v_index2,"Y"]

  mean_po_y0 = mean(po_y0_v1,po_y0_v2)

  control_list[[index]] = mean(mean_po_y0)

  po_y1_v1 = d1[v_index1,"D"]
  po_y1_v2 = d1[v_index2,"D"]

  mean_po_y1 = mean(po_y1_v1,po_y1_v2)

  treatment_list[[index]] = mean(mean_po_y1)

}


### Calculating variance of clusters for control
var_control = var(unlist(control_list))
var_treatment = var(unlist(treatment_list))

#Calculating SE(ATE)
```

```
m_var_y0 = num_treatment_records*var_control
n_minus_m = num_total_records - num_treatment_records
n_minus_m_var_y1 = n_minus_m * var_treatment

#Cluster1
cov_y0_y1 = cov(unlist(control_list),unlist(treatment_list))
se_ate2 = sqrt(
  (1/(num_clusters-1))* (m_var_y0/n_minus_m +
                           n_minus_m*var_treatment/num_treatment_clusters +
                           2 * cov_y0_y1)
)

se_ate2
```

```
## [1] 1.186621
```

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?

Clustering leads to confounding of effects since there is some other factor not accounted for within the cluster that may effects the potential outcome and will depending on the characteristics inherent to the cluster that cannot be adjusted for by individual randomization. The factor that is associated with certain clusters may be unmeasured and cannot be accounted for or adjusted in the analysis so it can lead to different standard errors. The implications for the design of cluster randomized experiments may not be as reliable as a pure randomization experiment and may not capture the true underlying effects present within the experiment.

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)?

```
#(.5+x)/100*1000000*100=profit+ad
profit<-0.5/100*1000000*100
ad <-1000000*.1

x<-(profit+ad)/1000000-0.5
x
```

```
## [1] 0.1
```

A 0.1% increase of visitors is needed for advertising to be worth it

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?

```r
#50/50 split of total
n1<-1000000*.5
n2<-1000000*.5

#calculate successes for each
x1<-n1*(.5/100)
x2<-n2*(.7/100)

#calculate p
p<-(x1+x2)/(n1+n2)
p
```

```
## [1] 0.006
```

```r
#calculate SE
se<-sqrt(p*(1-p)*(1/n1+1/n2))
se
```

```
## [1] 0.0001544539
```

```r
#Calculate 95% CI
0.002+1.96*se #upperbound
```

```
## [1] 0.00230273
```

```r
0.002-1.96*se #lowerbound
```

```
## [1] 0.00169727
```

- **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% confidence interval is calculated by multiplying the standard error by 1.96.

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

The confidence interval is fairly precise so assuming that the data is reliable and the underlying assumptions are correct, it can be recommended to run the experiment. However, caution is needed since there are underlying assumptions in the analysis which may not apply to real world situations. The calculations for the confidence interval is assuming the underlying distribution is normal. If this condition is not met the confidence interval is not a valid measure of the true confidence interval.

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% of users. What would be the width of the confidence interval for this experiment if only 1% of users were placed in the control group?

```r
#99/1 split of total
n1<-1000000*.01 # Control
n1
```

```
## [1] 10000
```

```r
n2<-1000000*.99 #Treatment
n2
```

```
## [1] 990000
```

```r
#calculate successes for each
x1<-n1*.5/100

x2<-n2*.5/100+.2/100*n2

#calculate p
p<-(x1+x2)/(n1+n2)
p
```

```
## [1] 0.00698
```

```r
#calculate SE
se<-sqrt(p*(1-p)*(1/n1+1/n2))

#Calculate 95% CI
0.002+1.96*se #upperbound
```

```
## [1] 0.003640005
```

```r
0.002-1.96*se #lowerbound
```

```
## [1] 0.000359995
```

5. Sports Cards

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

```r
d2 <- read.csv("./data/listData.csv", stringsAsFactors = FALSE)
head(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% confidence interval for the difference between the treatment mean and the control mean, using analytic formulas for a two-sample t-test from your earlier statistics course.

```r
t.test(d2$bid[d2$uniform_price_auction==1],
       d2$bid[d2$uniform_price_auction==0], alterative='two.tailed')
```

```
##
##  Welch Two Sample t-test
##
## data:  d2$bid[d2$uniform_price_auction == 1] and d2$bid[d2$uniform_price_auction == 0]
## 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:
##  -20.854624  -3.557141
## sample estimates:
```

```
## mean of x mean of y
##  16.61765  28.82353
```

b. In plain language, what does this confidence interval mean? **The confidence interval of -20.854 to -3.557 is the range that contains the true value of the test statistic value of whether or not to reject the null hypothesis 95% of the time if repeated sampled over many times.**

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.

```
reg <-lm(bid ~ uniform_price_auction, data=d2)
summary(reg)
```

```
##
## Call:
## lm(formula = bid ~ uniform_price_auction, data = 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% confidence interval you get from the regression.

```
# Can also be manually calcuated by using bid estimate + or - 2*(Std. Error)
confint(reg, '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 t value p-value reported is 0.00631.
anova(reg)$'Pr(>F)'[1]
```

```
## [1] 0.006314796
```

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

```
cases <- c(0,1)

ate <- mean(d2$bid[d2$uniform_price_auction==1]) - mean(d2$bid[d2$uniform_price_auction == 0])
ate
```

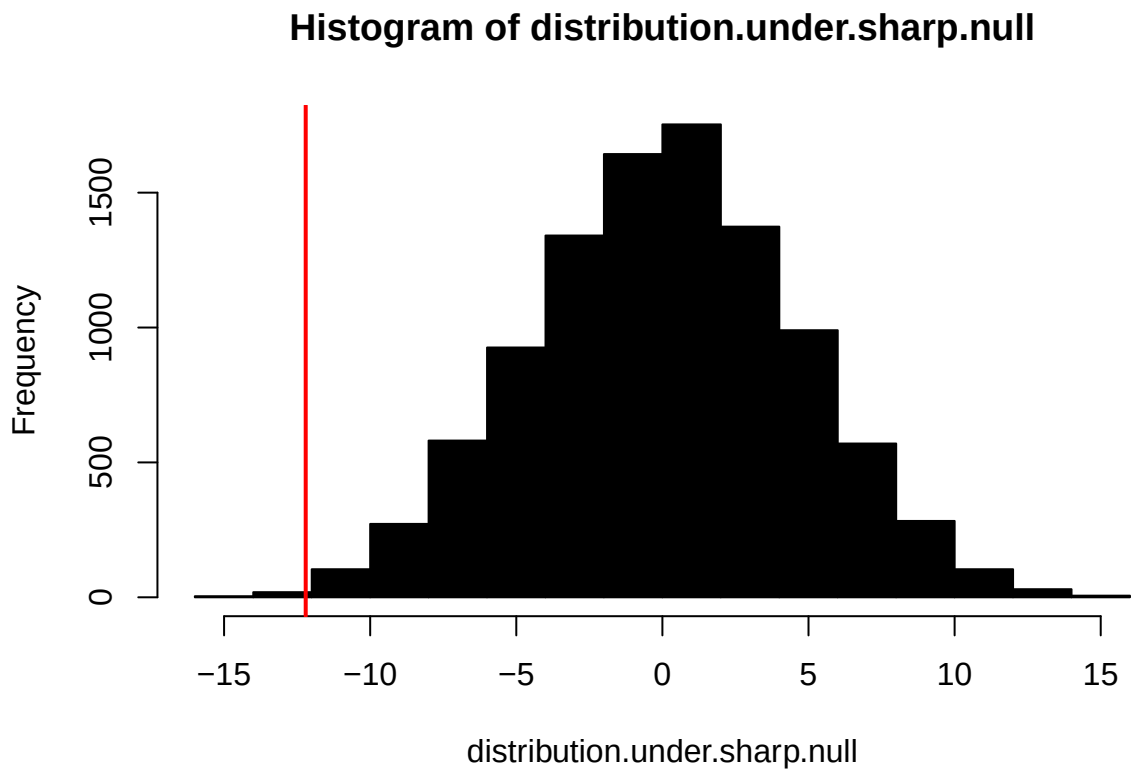
```
## [1] -12.20588
```

```
randomize <- function(group) {
  sample(cases, size = group, replace = TRUE)
```

```
  }

estimate.ate <- function(d2){
  treatment <- randomize(length(d2$uniform_price_auction))
  return(mean(d2$bid[treatment==1]) - mean(d2$bid[treatment == 0]))
}

distribution.under.sharp.null<-replicate(10000, estimate.ate(d2))

p_value <- mean(abs(distribution.under.sharp.null) >= abs(ate))
p_value
```

```
## [1] 0.0052
```

```
hist(distribution.under.sharp.null, col = 'black')
abline(v = ate, col = 'red', lwd = 2)
```



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).)

```
t.test(d2$bid[d2$uniform_price_auction==1],
       d2$bid[d2$uniform_price_auction==0], alterative='two.tailed')
```

```
##
##  Welch Two Sample t-test
##
## data:  d2$bid[d2$uniform_price_auction == 1] and d2$bid[d2$uniform_price_auction == 0]
```

```
## 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:
##  -20.854624  -3.557141
## sample estimates:
## mean of x mean of y
##  16.61765  28.82353
```

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-values in e and f are very similar to each other but not exactly the same. The p-value should all be small since the sharp null hypothesis should be rejected in this case. They are not that different because regression on a large sample size tends to follow the central limit theorem especially when sample sizes are large. Randomization also follows the central limit theory and produces a normalized curve after repeated samples even if the underlying distribution was not normal at the beginning. If the sample size was smaller the p-values would not be similar since large samples are needed to satisfy the approximate normal distribution under the central limit theory.