Problem Set 2

Gaurav Khanna

```
library(data.table)
```

```
## Warning: package 'data.table' was built under R version 3.4.4
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.4
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:data.table':
##
##     between, first, last
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(experiment)
```

```
## Warning: package 'experiment' was built under R version 3.4.4
```

```
## Loading required package: boot
```

```
## Loading required package: MASS
```

```
## Warning: package 'MASS' was built under R version 3.4.4
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
## experiment: R Package for Designing and Analyzing Randomized Experiments
## Version: 1.1-4
```

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.

```
# Dataframe from the CSV
d <- read.csv("./data/Clingingsmith.2009.csv", stringsAsFactors = FALSE)
```

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.

```
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
```

```
# Grouping data by success
# success <-group_by(d, success)
# TODO: Chaining the operations is more elegant but i'd like to see the intermediate
success <- group_by(d, success)
summarize(success, count=n(), viewSum=sum(views), viewMean=mean(views))
```

```
## # A tibble: 2 x 4
##   success count viewSum viewMean
##     <int> <int>   <int>    <dbl>
## 1       0   448     837     1.87
## 2       1   510    1195     2.34
```

Answer: The mean of views among lottery winners and losers is 2.34 and 1.86 respcetively. The **views for others generally look more positive among the lottery winners**

```
# Estimate ATE function
est.ate <- function(result, treat) {
  mean(result[treat==1]) - mean(result[treat==0])
  }
```

```
ate <- est.ate(success$views, success$success)
ate
```

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

** Answer: A positive ATE 0.4748337 adds more weight to above statement**

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.)

```
# Sharp null allows us to complete the data, y(0) and y(1) from the data we already have
# We just need to randomize the success variable
# Creating a new data frame to try randomization on the orgiginal data
# The original is kept around intact for reference
dr <- read.csv("./data/Clingingsmith.2009.csv", stringsAsFactors = FALSE)
head(dr)
```

```
##   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
```

```
success1 <- group_by(dr, success)
success.summary <- summarize(success1, count=n(), viewSum=sum(views), viewMean=mean(views))
success.summary
```

```
## # A tibble: 2 x 4
##   success count viewSum viewMean
##     <int> <int>   <int>    <dbl>
## 1       0   448     837     1.87
## 2       1   510    1195     2.34
```

```
g.randomize <- function(control, treatment) {
  sample(c(rep(0, control), rep(1, treatment)))
  }
```

```
# Randomizing the success variable
# We are not doing any blocking, so subjects in a group can be variable
# We are choosing to keep the ratios the same as the actual experiment
count.control <- success.summary[which(success.summary$success == 0),]$count
count.control
```

```
## [1] 448
```

```
count.treatment <- success.summary[which(success.summary$success == 1),]$count
count.treatment
```

```
## [1] 510
```

```
successr <- g.randomize(count.control, count.treatment)
head(successr)
```

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

```
# Some statistics on the classes after randomization
head(successr)
```

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

```
table(success1$success)
```

```
##
##   0   1
## 448 510
```

```
table(successr)
```

```
## successr
##   0   1
## 448 510
```

```
# Modifying est.ate to use the object of "randomize" class
# Just in case we use the randomize() function from the library
# For now, I've written my own
est.mate <- function(result, treat) {
  mean(result[treat=="Treat"]) - mean(result[treat=="Control"])
  }
```

```
# ATE for 1 randomization for testing
ateR <- est.ate(dr$views, g.randomize(count.control, count.treatment))
ateR
```

```
## [1] 0.01780462
```

```
# This is very different from our ATE
```

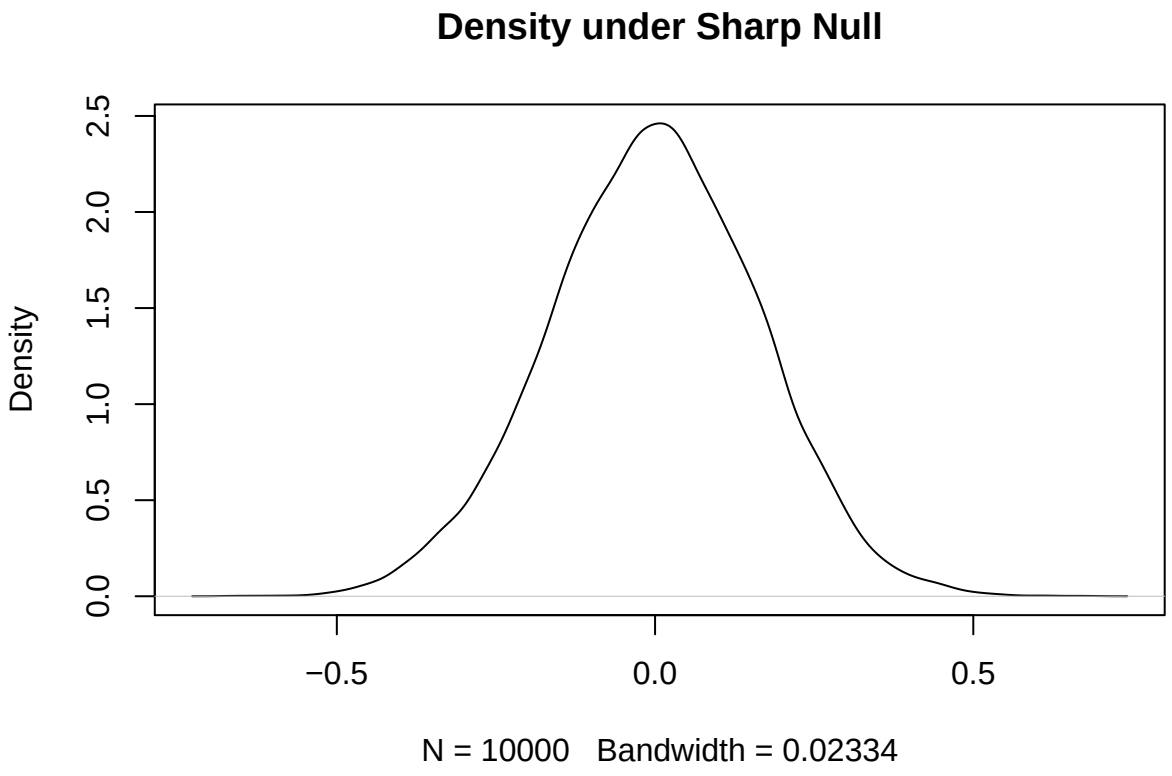
```
# Repeating the expriment 10,000 times
distribution.under.sharp.null <- replicate(10000, est.ate(dr$views, g.randomize(count.control, count.tr

# Average estimate for the ATE
mean(distribution.under.sharp.null)
```

```
## [1] -0.001774671
```

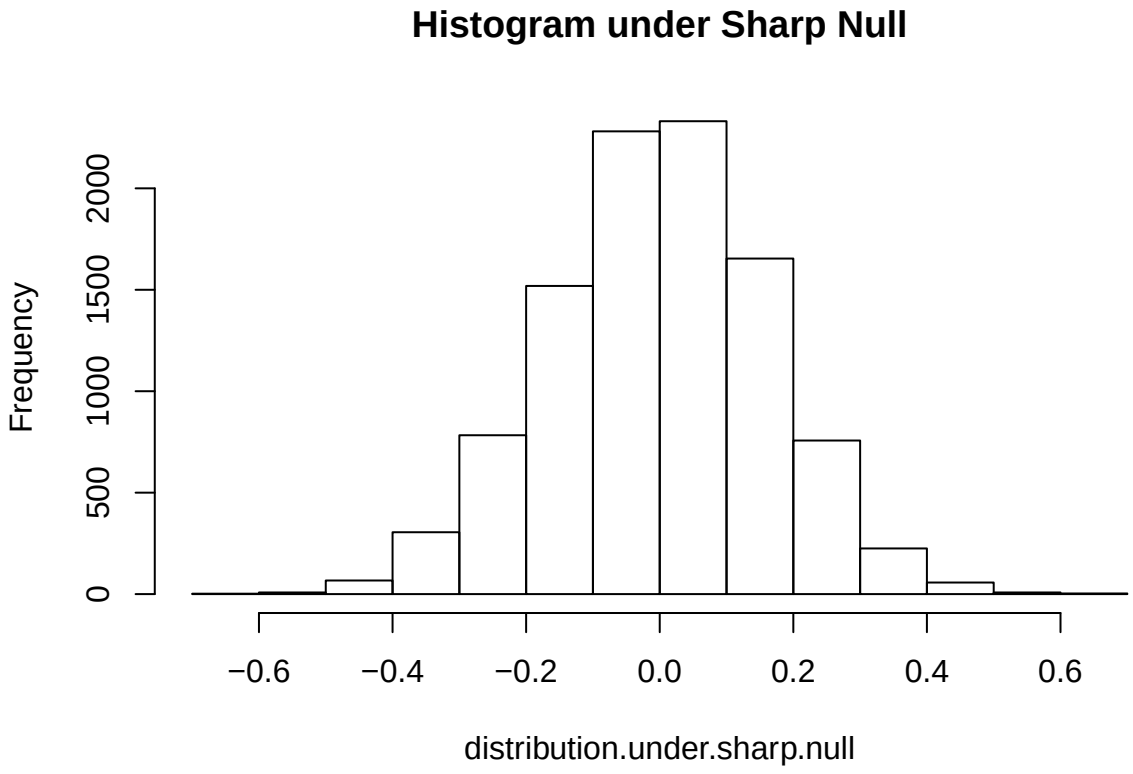
```
# Again, very different from ATE
```

```
# Graph for the estimates
plot(density(distribution.under.sharp.null),
     main = "Density under Sharp Null")
```

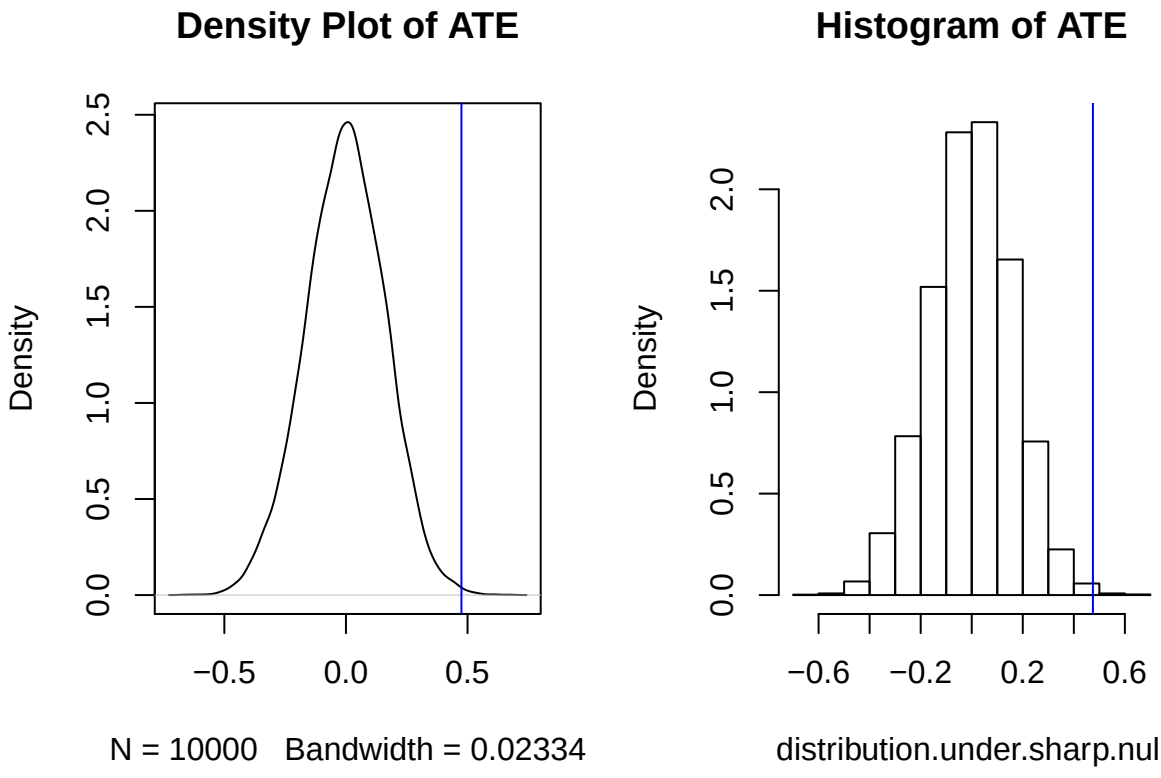



```
# Almost a normal distribution
```

```
# Histogram for the distribution
hist(distribution.under.sharp.null,
     main = "Histogram under Sharp Null")
```



```
# More analysis on the distribution. Plotting the p value
par(mfrow = c(1,2))
plot(density(distribution.under.sharp.null),
     main = "Density Plot of ATE")
abline(v = ate, col = "blue")
hist(distribution.under.sharp.null,
     main = "Histogram of ATE",
     freq = FALSE)
abline(v = ate, col = "blue")
```



Answer: Randomization done above. Graphs show the normal distribution of ATE estimates

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?

```
# ate was
ate
```

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

```
# num of assignments that generate an estimated ATE at least as large as the actual
n <- sum(distribution.under.sharp.null >= ate)
n
```

```
## [1] 17
```

Answer: 18 assignments generate an estimated ATE at least as large as the actual

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

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

```
## [1] 0.0017
```

Answer: One tailed p-value is .0018

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?

```
# num of simulated random assignments with an estimated ATE that is at least as large *in absolute valu
na <- sum(abs(distribution.under.sharp.null) >= abs(ate))
na
```

```
## [1] 34
```

Answer: 33 simulated random assignments generate an estimated ATE that is at least as large "in absolute value" as the actual estimate of the ATE

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

```
# 2 tailed p-value
ma <- mean(abs(distribution.under.sharp.null) >= ate )
ma
```

```
## [1] 0.0034
```

Answer: The 2 tailed p-value is .0033 ** The p value is less than .05. We should be able to reject the SHARP NULL hypothesis (No treatment effect)**

---

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)
```

```
## Warning: package 'foreign' was built under R version 3.4.4
```

```
d2 <- read.dta("./data/Titiunik.2010.dta")
head(d2)
```

```
##   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?

```
# Grouping the data by state
state <- group_by(d2, texas0_arkansas1)
head(state)
```

```
## # A tibble: 6 x 3
## # Groups:   texas0_arkansas1 [1]
##   term2year bills_introduced texas0_arkansas1
##       <int>            <int>             <int>
## 1         0               18                 0
## 2         0               29                 0
## 3         0               41                 0
## 4         0               53                 0
## 5         0               60                 0
## 6         0               67                 0
```

```
# Mean number of bills introduced in each state
summarize(state, count=n(), meanBills=mean(bills_introduced))
```

```
## # A tibble: 2 x 3
##   texas0_arkansas1 count meanBills
##              <int> <int>     <dbl>
## 1                0    31      68.8
## 2                1    35      25.5
```

```
# Which state looks more productive?
```

Answer: Texas has a mean of 68.77 compared to 25.51 for Arkansas. Texas's legislative sessions seem to produce more bills in average/session There is significant difference (significant as in value on average) among the legislative performance that one has to wonder about state/location being causal. A Question would be that if a Senator starts producing more bill when she moves from Arkansas to Texas? ** The mechanics of changing the state for the Senator and the answer to this Q or her producing more bills in Texas (and building a counterfactual argumet) does not seem very viable or even interesting. We'll not investigate state(texas0_arkansas1) as causal**

```
# Grouping by terms
terms <- group_by(d2, term2year)
```

```
head(terms)
```

```
## # A tibble: 6 x 3
## # Groups:   term2year [1]
##   term2year bills_introduced texas0_arkansas1
##       <int>            <int>            <int>
## 1         0               18                0
## 2         0               29                0
## 3         0               41                0
## 4         0               53                0
## 5         0               60                0
## 6         0               67                0
```

```
# Mean number of bills by terms
summarize(terms, meanBills=mean(bills_introduced))
```

```
## # A tibble: 2 x 2
##   term2year meanBills
##       <int>     <dbl>
## 1         0      53.1
## 2         1      38.6
```

```
# Which term is more effective?
```

Answer: 4 year terms seem to produce more bills (53.09) per legislative session on an average, when compared to average number of bills for a 2 year term (38.57) Again, a significant difference in performance among the 2 groups. Motivates us to look at cuasality We can change the term limits and observe performance. In a way, that's what the experiment does within each state. Terms could be causal and we should be able to measure the effect with experiments

Which of this effect is causal?

Answer: It looks more intersting to investigate "Term2year" as Causal. Its a parameter we can change and then observe the impact while controlling for other factors in a random sample

```
# Grouping by state and term term
stateTerms <- group_by(d2, texas0_arkansas1, term2year)
head(stateTerms)
```

```
## # A tibble: 6 x 3
## # Groups:   texas0_arkansas1, term2year [1]
##   term2year bills_introduced texas0_arkansas1
##       <int>            <int>            <int>
## 1         0               18                0
## 2         0               29                0
## 3         0               41                0
## 4         0               53                0
## 5         0               60                0
## 6         0               67                0
```

```
# Mean number of bills by grouping
summarize(stateTerms, meanBills=mean(bills_introduced))
```

```
## # A tibble: 4 x 3
## # Groups:   texas0_arkansas1 [?]
```

```
##   texas0_arkansas1 term2year meanBills
##              <int>     <int>     <dbl>
## 1                0         0      76.9
## 2                0         1      60.1
## 3                1         0      30.7
## 4                1         1      20.6
```

```
# Which groupings indicate more effectiveness?
```

Answer: 2 year terms on an average produce less bills(in a session) in Texas Same is the case in Arkansas So within a state, different terms show different legislative output

This should motivate us to look at the lenght of term (term2year) as causal

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

```
# Separating out the states

# Texas
texas <- state[which(state$texas0_arkansas1==0),]
head(texas)
```

```
## # A tibble: 6 x 3
## # Groups:   texas0_arkansas1 [1]
##   term2year bills_introduced texas0_arkansas1
##       <int>            <int>            <int>
## 1         0               18                0
## 2         0               29                0
## 3         0               41                0
## 4         0               53                0
## 5         0               60                0
## 6         0               67                0
```

```
# Finding out the control/treatment ratio
texas.group <- group_by(texas, term2year)
texas.summary <- summarize(texas.group, count=n())
texas.summary
```

```
## # A tibble: 2 x 2
##   term2year count
##       <int> <int>
## 1         0    16
## 2         1    15
```

```
# Arkansas
arkansas <- state[which(state$texas0_arkansas1==1),]
head(arkansas)
```

```
## # A tibble: 6 x 3
## # Groups:   texas0_arkansas1 [1]
##   term2year bills_introduced texas0_arkansas1
##       <int>            <int>            <int>
## 1         0               11                1
## 2         0               15                1
## 3         0               17                1
## 4         0               23                1
```

```
## 5         0               24                1
## 6         0               25                1
```

```
# Control/Treatment ratio
arkansas.group <- group_by(arkansas, term2year)
arkansas.summary <- summarize(arkansas.group, count=n())
arkansas.summary
```

```
## # A tibble: 2 x 2
##   term2year count
##       <int> <int>
## 1         0    17
## 2         1    18
```

```
# ATE estimate for Texas (based on the actual expriment)
texas_ate <- est.ate(texas$bills_introduced, texas$term2year)
texas_ate
```

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

```
# Considering the distribution with other random assignments
# We'll randomize the term variable. "2 years" is our treatment
# Since the grouping is binary, randomization of the entire dataset should work

# Ratio of treatment and control
texas.control <- texas.summary[which(texas.summary$term2year == 0), ]$count
texas.control
```

```
## [1] 16
```

```
texas.treatment <- texas.summary[which(texas.summary$term2year == 1), ]$count
texas.treatment
```

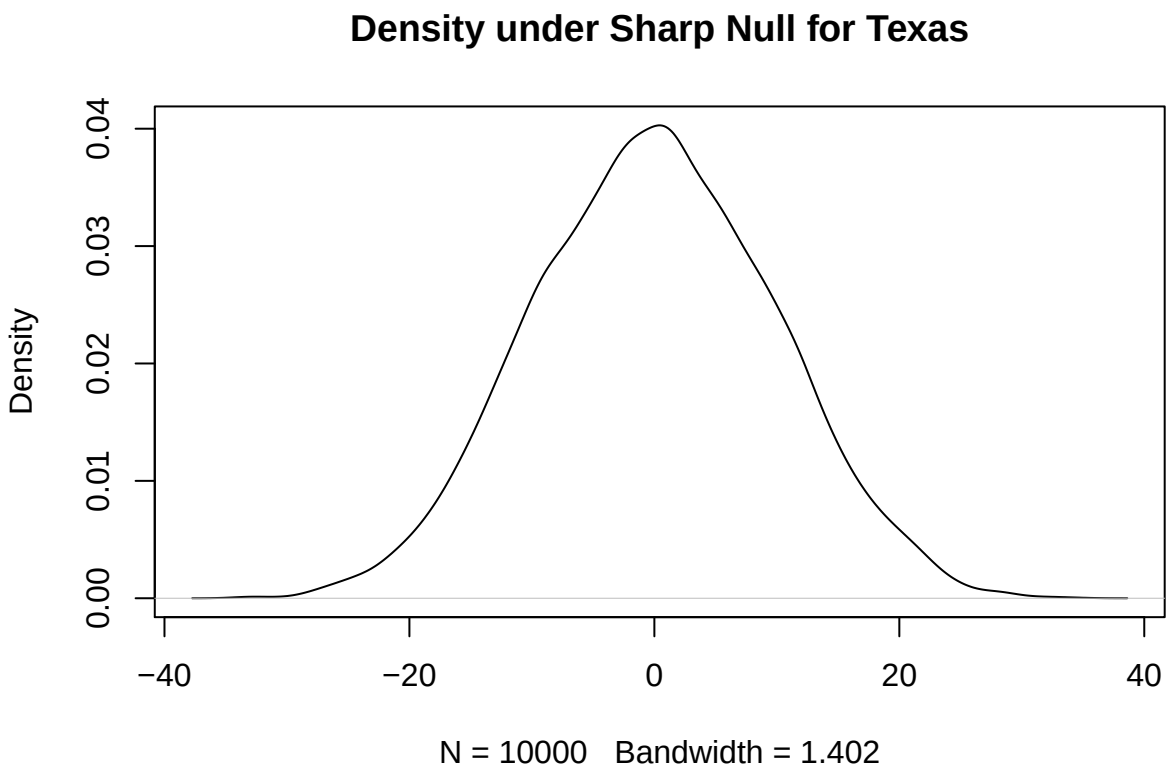
```
## [1] 15
```

```
# We'll create random sets in the same ratio
texas_r <- g.randomize(texas.control, texas.treatment)
table(texas_r)
```

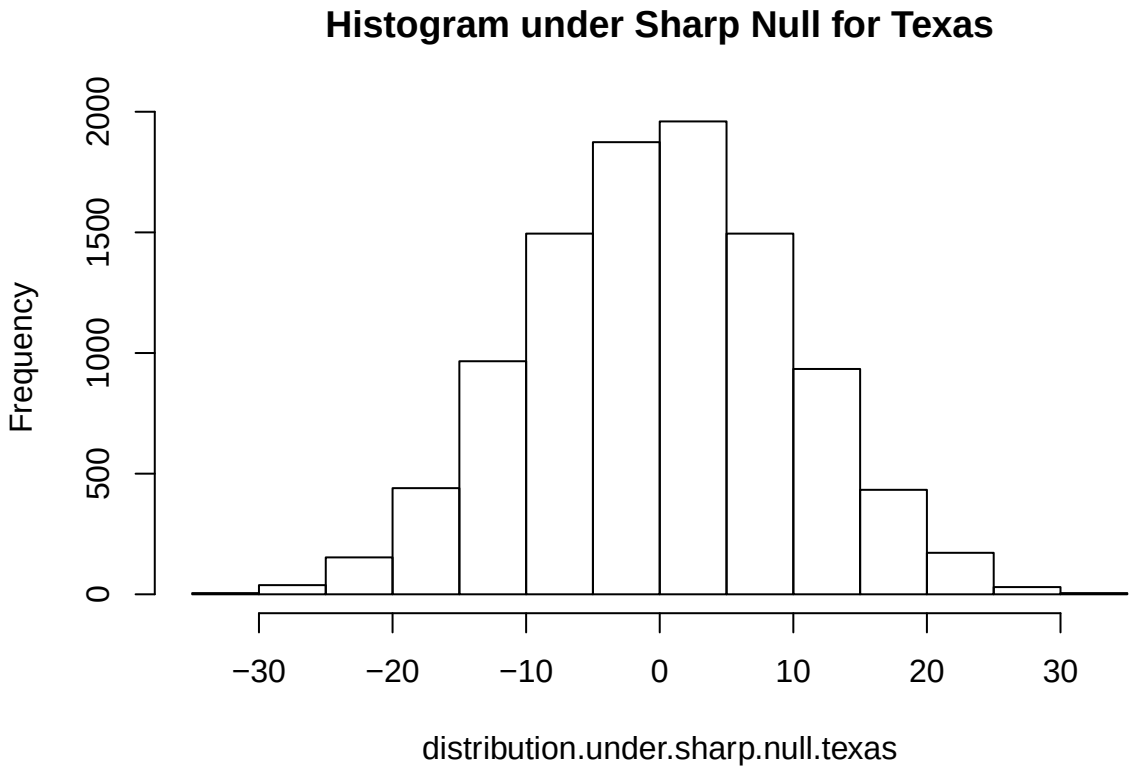
```
## texas_r
##  0  1
## 16 15
```

```
# Repeating the expriment 10,000 times
distribution.under.sharp.null.texas <- replicate(10000, est.ate(texas$bills_introduced, g.randomize(texa
```

```
# Visualizing the distribution of ATE estimates
plot(density(distribution.under.sharp.null.texas),
     main = "Density under Sharp Null for Texas")
```




```
# Histogram of the ATE estimates
hist(distribution.under.sharp.null.texas,
     main = "Histogram under Sharp Null for Texas")
```



```r
# average estimated ate for texas
texas.ate.mean <- mean(distribution.under.sharp.null.texas)
texas.ate.mean
```

```
## [1] -0.02403833
```

```r
# Standard error is the square root of the averaged squared deviation (from the average EST estimate)
# Debug head(distribution.under.sharp.null.texas)
se_texas <- sqrt(mean((distribution.under.sharp.null.texas-mean(distribution.under.sharp.null.texas))^2)
se_texas
```

```
## [1] 9.825166
```

Answer: The standard error for ATE estimates in Texas is 9.7734

```r
# ATE for arkansas
arkansas_ate <- est.ate(arkansas$bills_introduced, arkansas$term2year)
arkansas_ate
```

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

```r
# Considering the distribution with other random assignments
# We'll randomize the term variable. "2 years" is our treatment
# Since the grouping is binary, randomization of the entire dataset should work
# Ratio of treatment and control
arkansas.control <- arkansas.summary[which(arkansas.summary$term2year == 0), ]$count
arkansas.control
```

```
## [1] 17
```

```r
arkansas.treatment <- arkansas.summary[which(arkansas.summary$term2year == 1), ]$count
arkansas.treatment
```

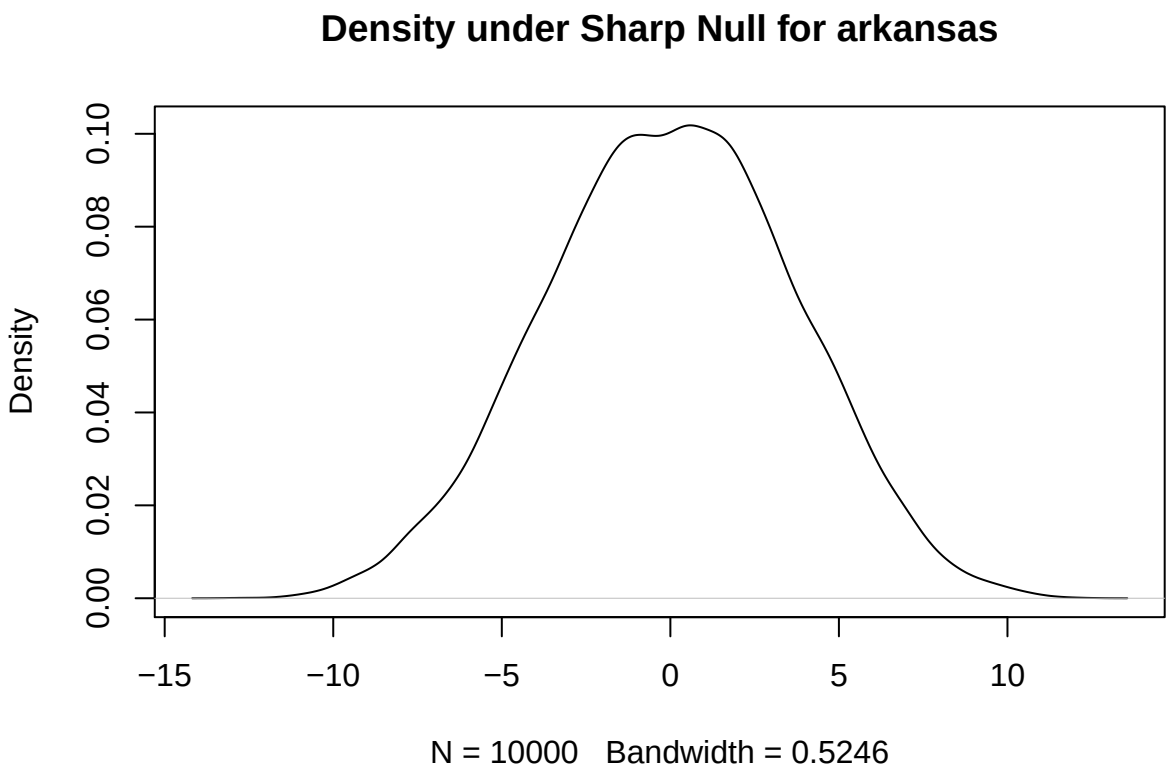
```
## [1] 18
```

```r
# We'll create random sets in the same ratio
arkansas_r <- g.randomize(arkansas.control, arkansas.treatment)
table(arkansas_r)
```

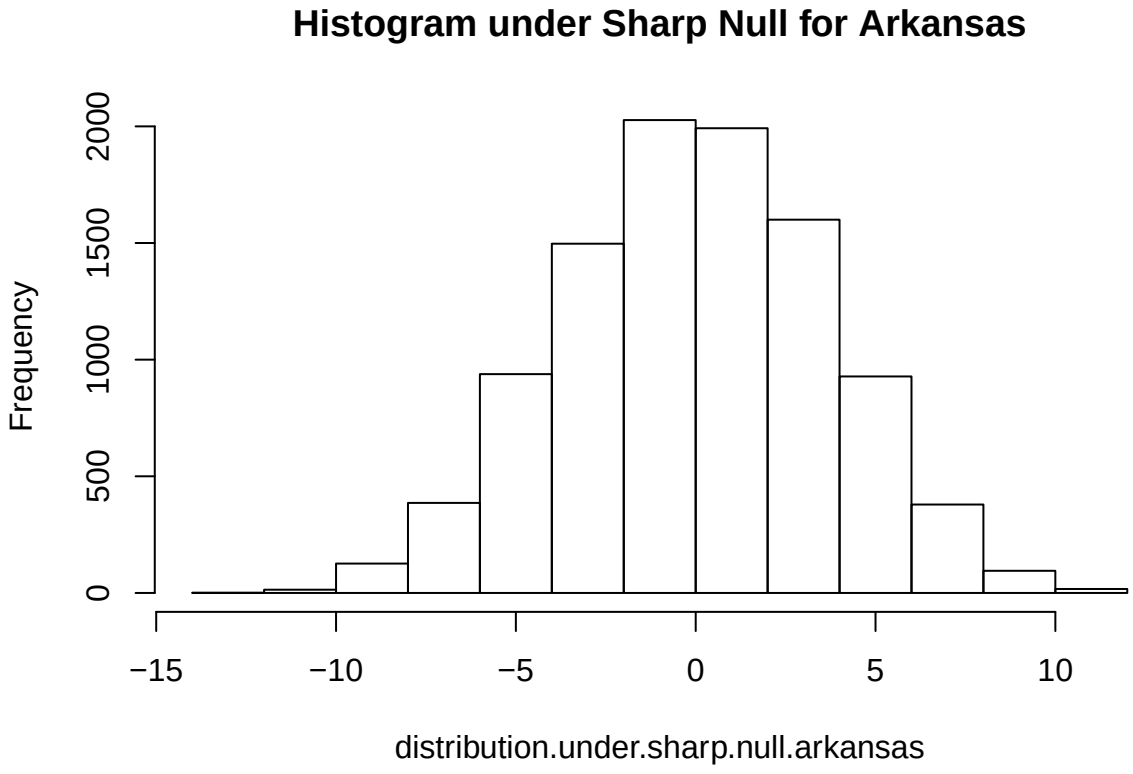
```
## arkansas_r
##  0  1
## 17 18
```

```r
# Repeating the expriment 10,000 times
distribution.under.sharp.null.arkansas <- replicate(10000, est.ate(arkansas$bills_introduced, g.randomi
```

```r
plot(density(distribution.under.sharp.null.arkansas),
     main = "Density under Sharp Null for arkansas")
```




```
hist(distribution.under.sharp.null.arkansas,
     main = "Histogram under Sharp Null for Arkansas")
```




```
# average estimated ate for arkansas
arkansas.ate.mean <- mean(distribution.under.sharp.null.arkansas)
```

```
arkansas.ate.mean
```

```
## [1] -0.007851307
```

```
# Standard error is the square root of the averaged squared deviation (from the average EST estimate)
# head(distribution.under.sharp.null.arkansas)
se_arkansas <- sqrt(mean((distribution.under.sharp.null.arkansas-mean(distribution.under.sharp.null.ark
se_arkansas
```

```
## [1] 3.677486
```

Answer: The standard error for ATE estimate for arkansas is 3.707

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

```
# ATE for both sides combined
# Recalling previous values and then using 3.10

# ATE for texas from the experiment
texas_ate
```

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

```
# ATE for arkansas from the experiment
arkansas_ate
```

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

```
# Number of observations in texas
nrow(texas)
```

```
## [1] 31
```

```
# Number of observations in Arkansas
nrow(arkansas)
```

```
## [1] 35
```

```
# Total number of observations
nrow(d2)
```

```
## [1] 66
```

```
# Combined ATE
comb_ate <- (texas_ate*nrow(texas)/nrow(d2)) + (arkansas_ate*nrow(arkansas)/nrow(d2))
comb_ate
```

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

Answer: The overall ATE is -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.

```
# The summary below reveals a lof of why pooling this data would produce a biased estimate of ATE. We s

# Mean number of bills by grouping
summarize(stateTerms, meanBills=mean(bills_introduced))
```

```
## # A tibble: 4 x 3
## # Groups:   texas0_arkansas1 [?]
##   texas0_arkansas1 term2year meanBills
##              <int>     <int>     <dbl>
## 1                0         0      76.9
## 2                0         1      60.1
## 3                1         0      30.7
## 4                1         1      20.6
```

```
# What if we pooled data for both states to calculate the ATE (Instead of the method above)?
# Value of ATE from the experiment with pooled data
pooled.ate <- est.ate(d2$bills_introduced, d2$term2year)
pooled.ate
```

```
## [1] -14.51515
```

Its quite similar to the value we get from the calculation of the overall ATE

Answer: The pooled data produces a biased estimate of ATE due to the different baselines (of bills produced in one session) among the 2 states. Texas average is 76.8 (bills in a session) for control, while Arkansas is at 30.70, which is less than half the baseline for Texas. Same is observed via the standard error of ate estimates from the 2 states. SE for Texes is 9.77. SE for Arkansas is 3.77561

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

```
# Recalling previous values and then putting them in 3.12 for combined SE

# Standard ATE error for Texas
se_texas
```

```
## [1] 9.825166
```

```
# Standard ATE error for Arkansas
se_arkansas
```

```
## [1] 3.677486
```

```
# Standar error for the 2 combined
se_comb <- sqrt((nrow(texas)*se_texas/nrow(d2))^2 + (nrow(arkansas)*se_arkansas/nrow(d2))^2)
se_comb
```

```
## [1] 5.009996
```

Answer: Standard Error for the overall ATE is 4.99

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

```
# Randomization inference to test the sharp null hypothesis (treatment effect is 0 in Texas)

# We have the results of 10,000 randomizations above in distribution.under.sharp.null.texas
head(distribution.under.sharp.null.texas)
```

```
## [1] -2.9208333 -5.1166667  5.6041667  5.9916667 -6.1500000  0.3083333
```

```
# We also have the ate from the actual expriment
texas_ate
```

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

```
# Two tailed p value
p_two_texas <- mean(abs(distribution.under.sharp.null.texas) >= abs(texas_ate))
p_two_texas
```

```
## [1] 0.0882
```

Answer: p value for ATE for Texas is 0.0863 p value is close but not less than .05. We cannot reject the SHARP NULL hypothesis Means that the observed effects of term changes could happen by chance

```
# Randomization inference to test the sharp null hypothesis (treatment effect is 0 in Arkansas)

# We have the results of 10,000 randomizations above in distribution.under.sharp.null.texas
head(distribution.under.sharp.null.arkansas)
```

```
## [1]  0.8856209 -2.6601307 -3.8039216 -1.5163399  3.1732026 -0.6013072
```

```
# We also have the ate from the actual expriment
arkansas_ate
```

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

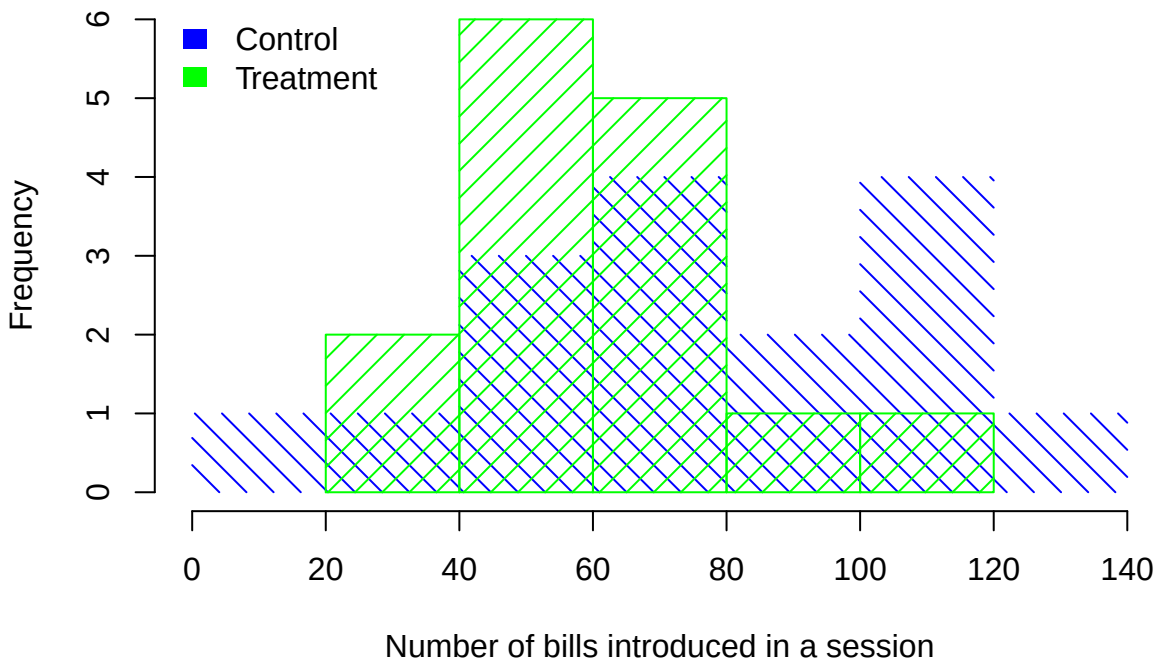
```
# Two tailed p value
p_two_arkansas <- mean(abs(distribution.under.sharp.null.arkansas) >= abs(arkansas_ate))
p_two_arkansas
```

```
## [1] 0.003
```

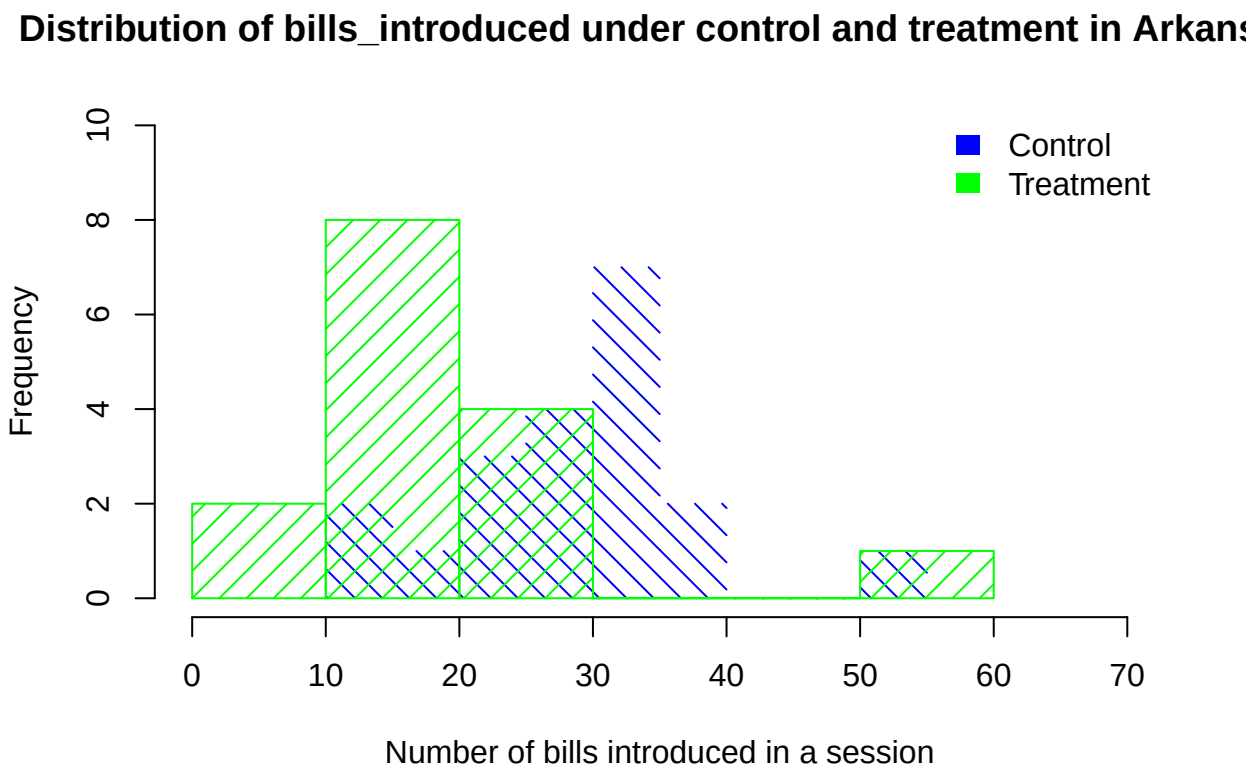
Answer: p value for ATE for Arkansas is 0.0028 p value is less than .05. We can reject the SHARP NULL hypothesis Means the effect of term change is statistically significant in Arkansas

g. **IN Addition:** Plot histograms for both the treatment and control groups in each state (for 4 histograms in total). **Answer: Tough to compare with 2 different histograms. Trying to overlap**

```
# Histogram for control and Treatment in Texas
hist(texas[which(state$term2year == 0),]$bills_introduced, xlim = c(0,140), ylim = c(0,6), col = "blue"
hist(texas[which(state$term2year == 1),]$bills_introduced, xlim = c(0, 140), ylim = c(0,6), col= "green
legend('topleft',c('Control','Treatment'),
       fill = c("blue", "green"), bty = 'n',
       border = NA)
```




```
# Histogram for control and treatment in Arkansas
hist(arkansas[which(state$term2year == 0),]$bills_introduced, xlim = c(0,70), ylim = c(0,10), col = "blu
hist(arkansas[which(state$term2year == 1),]$bills_introduced, xlim = c(0, 70), ylim = c(0,10), col= "gr
legend('topright',c('Control','Treatment'),
       fill = c("blue", "green"), bty = 'n',
       border = NA)
```



---

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
d3 <- read.csv("./data/ggChapter3.csv", stringsAsFactors = FALSE)
```

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.

```
# Add a cluster ID to data
d3$cluster <- NA
# Assign the clusters as stated in "a"
d3$cluster <- c(1,1,2,2,3,3,4,4,5,5,6,6,7,7)
head(d3)
```

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

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

```
# pull out the cluster ID's
all_clusters <- unique(d3$cluster)
all_clusters
```

```
## [1] 1 2 3 4 5 6 7
```

```
# Number of clusters in treatment
clusters.in.treatment <- 3
# Generic function to randomly(pseudo) pick the cluster ID's
randomize.clusters <- function(d) {
  treat.clusters <- sample(x = all_clusters,
                           size = clusters.in.treatment,
                           replace = FALSE)
  # returns 1 if a cluster ID in those that are picked
  return(as.numeric(d$cluster %in% treat.clusters))
}

# Trying out one randomization. Now we have the data prepared for the experiment
d3$treatment <- randomize.clusters(d3)

# QUESTION & TODO More randomizations to get an estimate. For now just treating it as one expriment

# Grouping data by clusters and adding the treatment column(treatment-1, control-0 for the cluster)
clusters <- group_by(d3, cluster)
clusters.summary <- summarize(clusters, count=n(), mean_y=mean(Y), mean_d=mean(D), treatment=sum(treatm
```

```
## Warning: package 'bindrcpp' was built under R version 3.4.4
```

```
clusters.summary
```

```
## # A tibble: 7 x 5
##   cluster count mean_y mean_d treatment
##     <dbl> <int>  <dbl>  <dbl>     <dbl>
## 1       1     2    0.5    0           0
## 2       2     2    3      1.5         0
## 3       3     2    5      0           1
## 4       4     2    7.5    2.5         1
## 5       5     2   14.5   10.5         0
## 6       6     2   16     11.5         1
## 7       7     2   17.5   11           0
```

```
# QUESTION When calculating variances, are we calculating among all the clusters or clusters in control

# Variance in y(0) clustered
variance_y <- (sum((clusters.summary$mean_y - mean(clusters.summary$mean_y))^2))/nrow(clusters.summary)
variance_y
```

```
## [1] 39.69388
```

```
# Variance in y(1) clustered
variance_d <- (sum((clusters.summary$mean_d - mean(clusters.summary$mean_d))^2))/nrow(clusters.summary)
variance_d
```

```
## [1] 25.20408
```

```
# Covariance among y(0) and Y(1)
covariance <- sum((clusters.summary$mean_y - mean(clusters.summary$mean_y))*(clusters.summary$mean_d - m
covariance
```

```
## [1] 30.53061
```

```
# applying 3.22

# m is subjects in treatment (not clusters). N is count of subjects, number of clusters is 7
m <- clusters.in.treatment * 2
m
```

```
## [1] 6
```

```
N <- nrow(d3)
N
```

```
## [1] 14
```

```
nrow(clusters.summary)
```

```
## [1] 7
```

```
se <- sqrt(((m*variance_y/N-m) + ((N-m)*variance_d/m) + 2*covariance)/(nrow(clusters.summary) - 1))
se
```

```
## [1] 4.196791
```

Answer: The standard error for one randomization (3 groups in treatment) is 4.196791

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.

```
# repeating by grouping the observations in a different way
d4 <- read.csv("./data/ggChapter3.csv", stringsAsFactors = FALSE)
# Add the cluster ID to data
d4$cluster <- NA
# Assign the clusters for "a"
d4$cluster <- c(1,2,3,4,5,6,7,7,6,5,4,3,2,1)
head(d4)
```

```
##   Village Y D Block cluster
## 1       1 0 0     1       1
## 2       2 1 0     1       2
## 3       3 2 1     1       3
## 4       4 4 2     1       4
## 5       5 4 0     1       5
## 6       6 6 0     1       6
```

```
# Trying out one randomization. Now we have the data prepared for the experiment
d4$treatment <- randomize.clusters(d4)
# Grouping by the cluster and creating the summary
clusters <- group_by(d4, cluster)
clusters.summary <- summarize(clusters, count=n(), mean_y=mean(Y), mean_d=mean(D), treatment=sum(treatm
clusters.summary
```

```
## # A tibble: 7 x 5
##   cluster count mean_y mean_d treatment
##     <dbl> <int>  <dbl>  <dbl>     <dbl>
## 1       1     2    9      8.5         0
## 2       2     2    9      2.5         0
## 3       3     2    9      8           1
## 4       4     2   10      5           0
## 5       5     2    9.5    4.5         1
## 6       6     2   10      6           1
## 7       7     2    7.5    2.5         0
```

```
# Variance in y(0) clustered
variance_y <- (sum((clusters.summary$mean_y - mean(clusters.summary$mean_y))^2))/nrow(clusters.summary)
variance_y
```

```
## [1] 0.622449
```

```
# Variance in y(1) clustered
variance_d <- (sum((clusters.summary$mean_d - mean(clusters.summary$mean_d))^2))/nrow(clusters.summary)
variance_d
```

```
## [1] 4.918367
```

```
# Covariance among y(0) and Y(1)
covariance <- sum((clusters.summary$mean_y - mean(clusters.summary$mean_y))*(clusters.summary$mean_d - m
covariance
```

```
## [1] 0.6020408
```

```
# applying 3.22

# m is subjects in treatment (not clusters). N is count of subjects, number of clusters is 7
m <- clusters.in.treatment * 2
m
```

```
## [1] 6
```

```
N <- nrow(d4)
N
```

```
## [1] 14
```

```
nrow(clusters.summary)
```

```
## [1] 7
```

```
se1 <- sqrt(((m*variance_y/N-m) + ((N-m)*variance_d/m) + 2*covariance)/(nrow(clusters.summary) - 1))
se1
```

```
## [1] 0.5814735
```

Answer: The standard error for one randomization (3 groups in treatment) is 0.5814735

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?

Answer: The methods lead to different standard errors due to differences in the cluster means (for outcomes in control and treatment) which indicate the differences among the subjects (villages in this case, that become part of a cluster). In this problem, the first method was

akin to clustering based on geography. Villages next to each other (in Sr. no), had similar outcomes and ended up in the same cluster, leading to large differences among the clusters. Method 2 mixed things up and clustered villages that had "more" different outcomes. This sharply reduced the differences among the clusters

Answer: The implication is that cluster design has to account for averages of outcomes among the subjects, or we'll get imprecise estimates for the treatment effect. The **best scenario would be to bring down the variance among clusters by grouping unlike subjects** (like method 2 where dissimilar villages were clustered together). This may not be possible as clustering is forced in most cases (like villages, schools by geography). **The next best idea could be to increase the number of clusters and then simulate all possible random assignments of clusters** to obtain the sampling distribution
**

---

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)?

```
# Each site visitor shown the ad is exposed to $.10 of ads
# There are 1,000,000 users in one week
# Profit is $100 on every phone
# y(0) - .5% of visitors to news site buy an iphone in a week (in absence of ads)

# Users for the website in a wee
week.users <- 1000000
week.users
```

```
## [1] 1e+06
```

```
# Profit per phone sold
profit_per_phone <- 100
profit_per_phone
```

```
## [1] 100
```

```
# % of visitors that buy without ads. This can be considered "control"
week.percent_users_buy <- .5
week.percent_users_buy
```

```
## [1] 0.5
```

```
# Number of users that buy (irrespective of the ads)
week.number_users_buy_gen <- (.5/100)*1000000
week.number_users_buy_gen
```

```
## [1] 5000
```

```
# Total profit in a week (irrespective of the ads)
week.profit_gen <- week.number_users_buy_gen * profit_per_phone
week.profit_gen
```

```
## [1] 5e+05
```

```
# ad spend per user and per week for all users. This assumes all users are exposed to the ads
adspend_per_user <- .10
adspend_per_week <- adspend_per_user * week.users
adspend_per_week
```

```
## [1] 1e+05
```

```
# We should atleast make up the money spend on ads in a week. That means we have to sell +inc number of
inc <- adspend_per_week/profit_per_phone
inc
```

```
## [1] 1000
```

```
# so we have to sell 1000 additional phones
# Which is obs % points based on the original population eligible for ads and purchases
inc_per <- (inc/week.users)*100
inc_per
```

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

```
# This is breakeven effect of the advertising. Anything more would be positive ROI
# Also, we assumed that all users were shown the ad so

# Say we show the ads to only half the population. The ad spend would be half, but then we'll measure t
```

Answer: The ads would have to convert an additional .1% website user to buyers This'll sell 1000 additional phones, which creates 1e5 of incremental profits, which covers the (1e-1)(1e6) cost of advertising

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?

```
# Measured effect is .2% points
inc_measured <- .2
inc_measured
```

```
## [1] 0.2
```

```
# users in control and treatment
week.users.control <- week.users/2
week.users.treatment <- week.users/2
week.users.control
```

```
## [1] 5e+05
```

```
week.users.treatment
```

```
## [1] 5e+05
```

```
# p by the formula in the question
p <- (week.users.control*week.percent_users_buy/100 + week.users.treatment*(week.percent_users_buy+inc_
p
```

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

```
# se by the formula in the question
se <- sqrt(p*(1-p)*((1/week.users.control) + (1/week.users.treatment)))
se
```

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

```
# tail of a 95% confidence interval
ci_tail <- se*1.96
ci_tail
```

```
## [1] 0.0003027296
```

```
# Upper and Lower limits of the confidence interval
ci1 <- .002 + ci_tail
ci1
```

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

```
ci2 <- .002 - ci_tail
ci2
```

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

```
# Width of the confidence interval
ci.wid1 <- ci1 - ci2
ci.wid1
```

```
## [1] 0.0006054592
```

```
# % size of the CI
(ci.wid1/(inc_measured/100))*100
```

```
## [1] 30.27296
```

Answer: The 95% confidence interval would be [0.00169727, 0.00230273]

- **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?

Answer: Yes, the confidence interval is precise enough to run the experiment. With equal sized groups we know that the interval is conservative, i.e. we have more than 95% chance of the estimate falling in this range. ** The 95% confidence level calculated above also meets our needs. Since profitability is a concern, its reassuing that the lower bound is above the customer conversion rate(visitor buying phones on seeing the AD) for positive ROI (Our campaign has positive ROI above .1% ad conversion

rate) The interval is narrow but includes both the "minimum requirment for ROI" (.1%) and the observed value of the expriment(.2%)**

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?

```
# what if only 1% of users were in control

# users in control and treatment
week.users.control <- .01*week.users
week.users.treatment <- week.users - week.users.control
week.users.control
```

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

```
week.users.treatment
```

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

```
# p by the formula in the question
p <- (week.users.control*week.percent_users_buy/100 + week.users.treatment*(week.percent_users_buy+inc_
p
```

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

```
# se by the formula in the question
se <- sqrt(p*(1-p)*((1/week.users.control) + (1/week.users.treatment)))
se
```

```
## [1] 0.0008367373
```

```
# tail of a 95% confidence interval
ci_tail <- se*1.96
ci_tail
```

```
## [1] 0.001640005
```

```
# Upper and Lower limits of the confidence interval
ci1 <- .002 + ci_tail
ci1
```

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

```
ci2 <- .002 - ci_tail
ci2
```

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

```
# Width of the confidence interval
ci.wid2 <- ci1 - ci2
ci.wid2
```

```
## [1] 0.00328001
```

Answer: The 95% confidence interval with 1% in control is would be [0.000359995, 0.003640005] Width of the confidence interval is 0.00328001 which is about 5 times the width of the confidence interval with 50% of the sample in control

---

5. Sports Cards

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

```
d5 <- read.csv("./data/listData.csv", stringsAsFactors = FALSE)
head(d5)
```

```
##   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.

```
# 95% confidence interval for the difference between the treatment mean and the control mean

# Grouping the data by treatment (& control)
by.treatment <- group_by(d5, uniform_price_auction)
# Summary
by.treatment.summary <- summarize(by.treatment, count=n(), mean.bid=mean(bid), sd.bid=sd(bid))
by.treatment.summary
```

```
## # A tibble: 2 x 4
##   uniform_price_auction count mean.bid sd.bid
##                   <int> <int>    <dbl>  <dbl>
## 1                     0    34     28.8   20.0
## 2                     1    34     16.6   15.4
```

```
# Seems we have 2 equally sized groups (control and treatment) with very different average bid sizes bu

# Getting an idea of normality (an essential for the T test)

# Extract the control and treatment separately
bid.control <- d5[which(d5$uniform_price_auction == 0),]
nrow(bid.control)
```

```
## [1] 34
```

```
head(bid.control)
```

```
##    bid uniform_price_auction
## 3   20                     0
## 8    2                     0
## 11  50                     0
## 12  25                     0
```

```
## 13   0                       0
## 14  70                       0
```

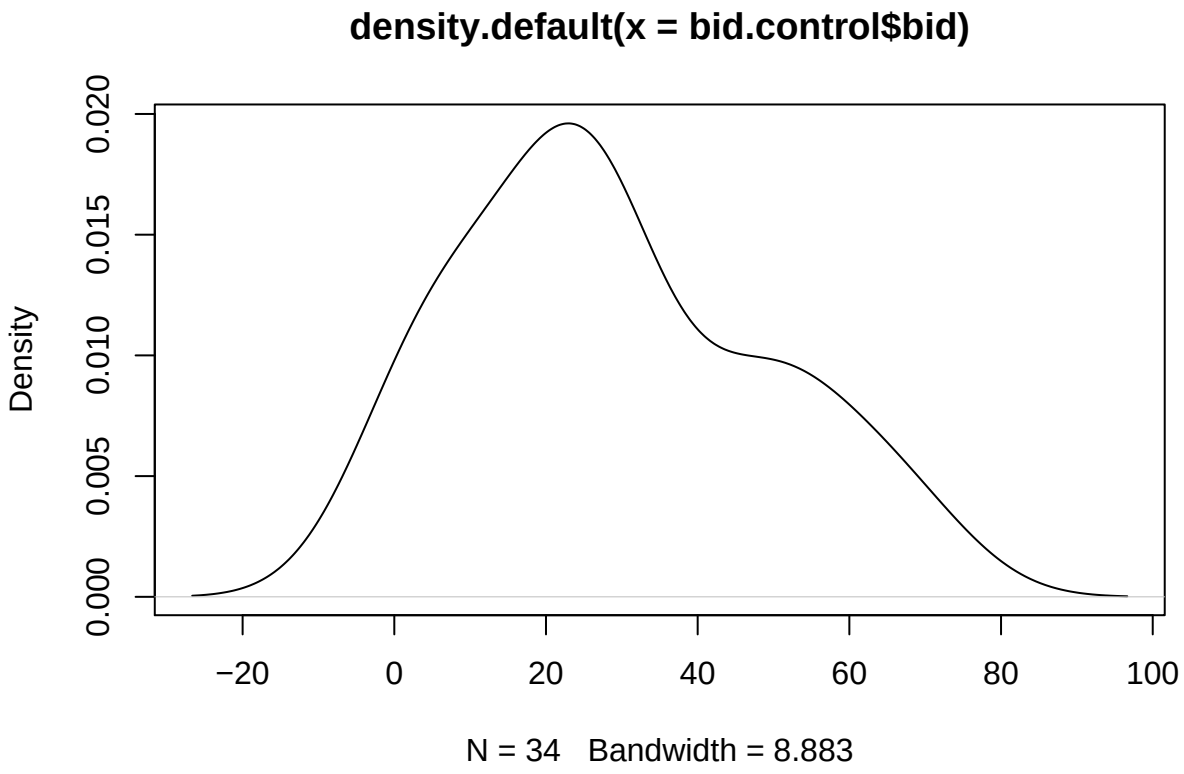
```
bid.treatment <- d5[which(d5$uniform_price_auction == 1),]
nrow(bid.treatment)
```

```
## [1] 34
```

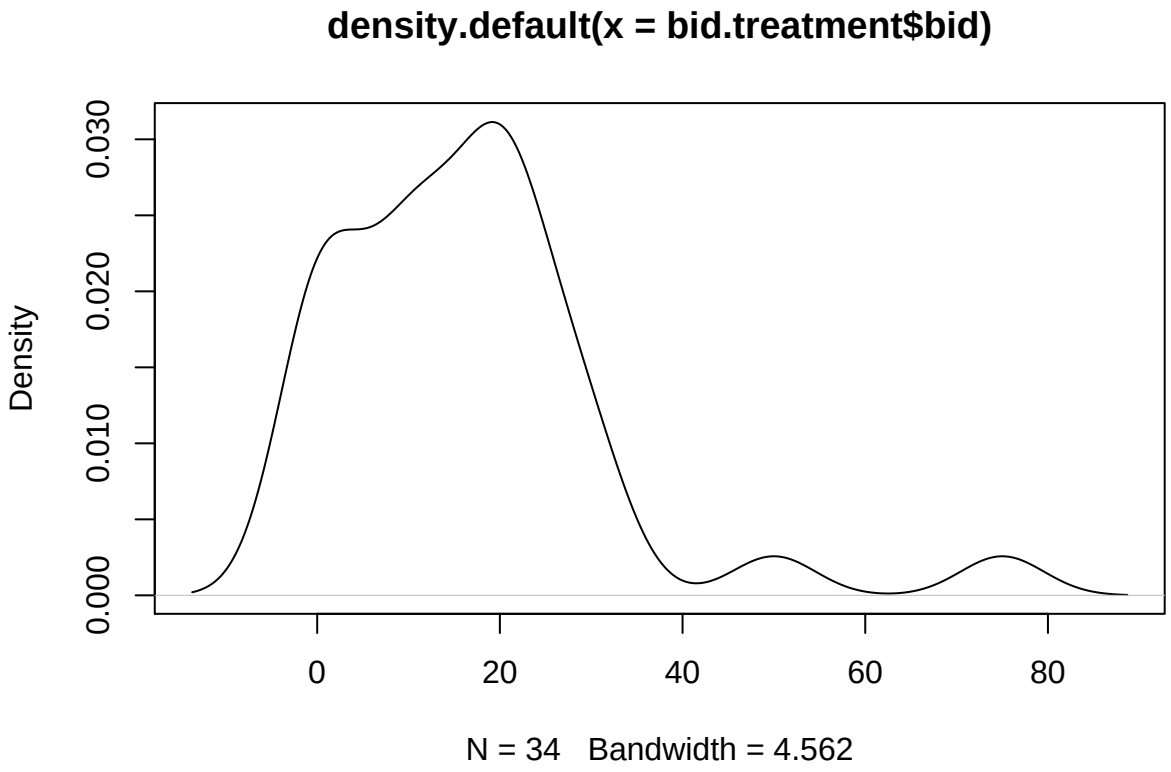
```
head(bid.treatment)
```

```
##   bid uniform_price_auction
## 1   5                     1
## 2   5                     1
## 4   0                     1
## 5  20                     1
## 6   0                     1
## 7   0                     1
```

```
# plots
plot(density(bid.control$bid))
```




```
plot(density(bid.treatment$bid))
```



Visual test for normality (essential for the T test): The distribution of bids in control is close to normal. Bids in treatment have a bias.

```
# T test using the analytic formula

# Mean of the 2 samples (control and test)
bid.control.mean <- mean(bid.control$bid)
bid.control.mean
```

```
## [1] 28.82353
```

```
bid.treatment.mean <- mean(bid.treatment$bid)
bid.treatment.mean
```

```
## [1] 16.61765
```

```
# Difference of the means
stat.mean.diff <- bid.treatment.mean - bid.control.mean
stat.mean.diff
```

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

```
# degrees of freedom
stat.df <- nrow(bid.control) + nrow(bid.treatment) -2
stat.df
```

```
## [1] 66
```

```
# sample variances
stat.control.variance <- sum((bid.control$bid - bid.control.mean)^2)/nrow(bid.control)
stat.control.variance
```

```
## [1] 387.4983
```

```r
stat.treatment.variance <- sum((bid.treatment$bid - bid.treatment.mean)^2)/nrow(bid.treatment)
stat.treatment.variance
```

```
## [1] 230.2362
```

```r
# Pooled standard deviation
sdpool.small <- sqrt(((nrow(bid.control)-1)*stat.control.variance + (nrow(bid.treatment)-1)*stat.treatm
sdpool.small
```

```
## [1] 17.57462
```

```r
# Pooled standard deviation with formula used for sample sizes > 30
sdpool.large <-sqrt((stat.control.variance/nrow(bid.control)) + (stat.treatment.variance/nrow(bid.treatr
sdpool.large
```

```
## [1] 4.262471
```

```r
# SSQ
ssq <- (sum((bid.control$bid - bid.control.mean)^2)+ sum((bid.treatment$bid - bid.treatment.mean)^2))/(
ssq
```

```
## [1] 318.2268
```

```r
sd <- sqrt(ssq)
sd
```

```
## [1] 17.83891
```

```r
# We'll use the small sample 34 is very close to the "30"
# Boundaries of the confidence interval
# Taking the t value (t.95) from the t tables for 66 degress of freedom
# The value is 2.00

ci1 <- stat.mean.diff + 2.00 * sdpool.small * (sqrt(1/nrow(bid.control) + 1/nrow(bid.treatment)))
ci1
```

```
## [1] -3.68094
```

```r
ci2 <- stat.mean.diff - 2.00 * sdpool.small * (sqrt(1/nrow(bid.control) + 1/nrow(bid.treatment)))
ci2
```

```
## [1] -20.73082
```

```r
# t statistic
stat.t <- (bid.treatment.mean - bid.control.mean)/sqrt((ssq/nrow(bid.treatment)) + (ssq/nrow(bid.contro
stat.t
```

```
## [1] -2.821144
```

```r
# Verified with the R calculations below
```

Answer: The 95% confidence interval from the calculations is: [-20.73082, -3.68094]

```r
# T test using R to verify the calculations
res <- t.test(bid.treatment$bid, bid.control$bid, var.equal = TRUE)
res
```

```
##
##  Two Sample t-test
##
## data:  bid.treatment$bid and bid.control$bid
## t = -2.8211, df = 66, p-value = 0.006315
```

```
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -20.844162  -3.567603
## sample estimates:
## mean of x mean of y
##  16.61765  28.82353
```

p value is .0063, which is less than the significance level alpha = .05 95% confidence interval from the "R" t test is [-20.844162, -3.567603]

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

Answer: The confidence interval [-20.844162, -3.567603] means that if we do a large number of measurements, there's a 95% chance that the difference in means would be in the range [-20.844162, -3.567603] This makes us reject the NULL hypothesis that the means are the same

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.

```r
# Regression
reg <- lm(bid ~ uniform_price_auction,
                data=d5)
summary(reg)
```

```
##
## Call:
## lm(formula = bid ~ uniform_price_auction, data = d5)
##
## 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
```

Answer We get exactly the same results

p value (0.006315) The equation is bid = 28.824 -12.206(uniform_price_auction)

d. Calculate the 95% confidence interval you get from the regression.

```
# The 95% confidence interval is the slope (coeff for "uniform_price_auction") +- 2 standard errors
# Could also be calculated using confint
confint(reg, 'uniform_price_auction', level=.95)
```

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

Answer: The 95% confidence interval for the coeff of "uniform_price_auction" is the coeff +- 2 standard errors(.4327 from the summary) Answer: We get the same interval as the T test

2.5 % 97.5 % -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.

```
# Coefficients and p values from the regression
summary(reg)
```

```
##
## Call:
## lm(formula = bid ~ uniform_price_auction, data = d5)
##
## 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
```

Answer: The regression reports a p value (for 2 sided hypotheses): 0.006315

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

```
# Some analysis of the data
bids.group <- group_by(d5, uniform_price_auction)
head(bids.group)
```

```
## # A tibble: 6 x 2
## # Groups:   uniform_price_auction [2]
##     bid uniform_price_auction
##   <int>                 <int>
## 1     5                     1
## 2     5                     1
```

```
## 3    20                     0
## 4     0                     1
## 5    20                     1
## 6     0                     1
```

```
bids.summary <- summarize(bids.group, count=n())
bids.summary
```

```
## # A tibble: 2 x 2
##   uniform_price_auction count
##                   <int> <int>
## 1                     0    34
## 2                     1    34
```

```
# p value using the randomization inference

# ATE observed
# Using the ATE formula from a
bids.ate <- est.ate(d5$bid, d5$uniform_price_auction)
bids.ate
```

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

```
# Randomizing the bids (among control and treatment).
# uniform_price_auction is binary so we can just assume 0, 1 as control and treatment
# We'll keep the ratio control:treatment the same as observed (34:34)
bids.control <- bids.summary[which(bids.summary$uniform_price_auction == 0), ]$count
bids.control
```

```
## [1] 34
```

```
bids.treatment <- bids.summary[which(bids.summary$uniform_price_auction == 1), ]$count
bids.treatment
```

```
## [1] 34
```

```
bids.random <- g.randomize(bids.control, bids.treatment)

# DEBUG: For curiosity figuring out the est ATE of 2 of the randomization done above
# Using the modified ATE formula done for Q a
bids.rate <- est.ate(d5$bid, g.randomize(bids.control, bids.treatment))
bids.rate
```

```
## [1] 1.205882
```

```
bids.random <- g.randomize(bids.control, bids.treatment)
bids.rate <- est.ate(d5$bid, g.randomize(bids.control, bids.treatment))
bids.rate
```

```
## [1] -0.2058824
```

```
# So we get quite a different value from the actual and  quite different values each time
```

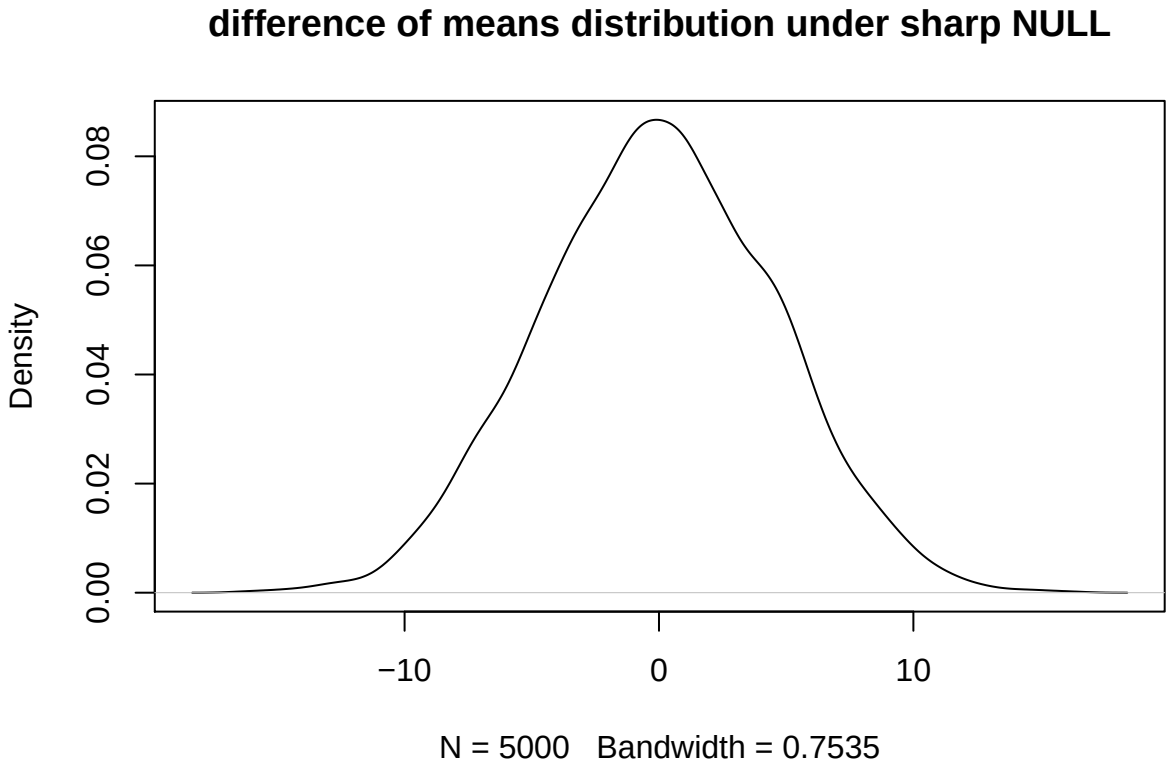
```
# Repeating the experiment 5k times and getting the distribution under sharp null
# We are most interested in the differences in mean statistic
dom.distribution.under.sharp.null <- replicate(5000, est.ate(d5$bid, g.randomize(bids.control, bids.trea


# Average estimate for the ATE
mean(dom.distribution.under.sharp.null)
```

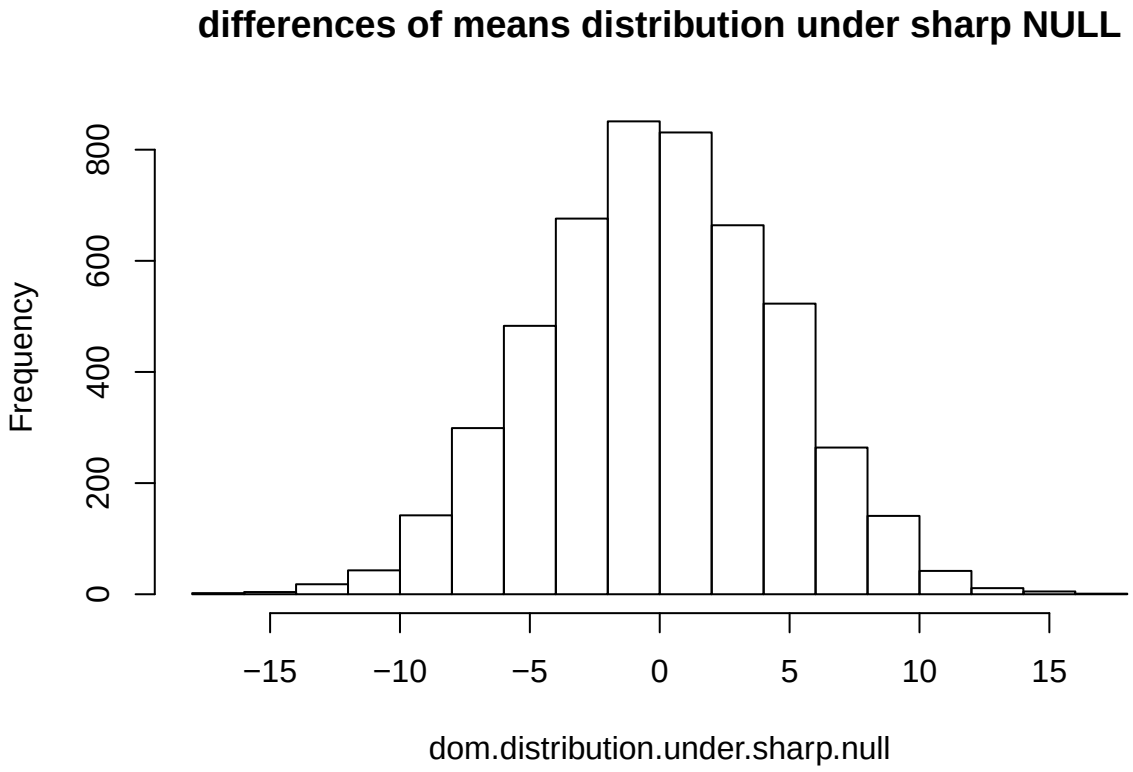
```
## [1] -0.04121176
```

```
# Again, very different from ATE
```

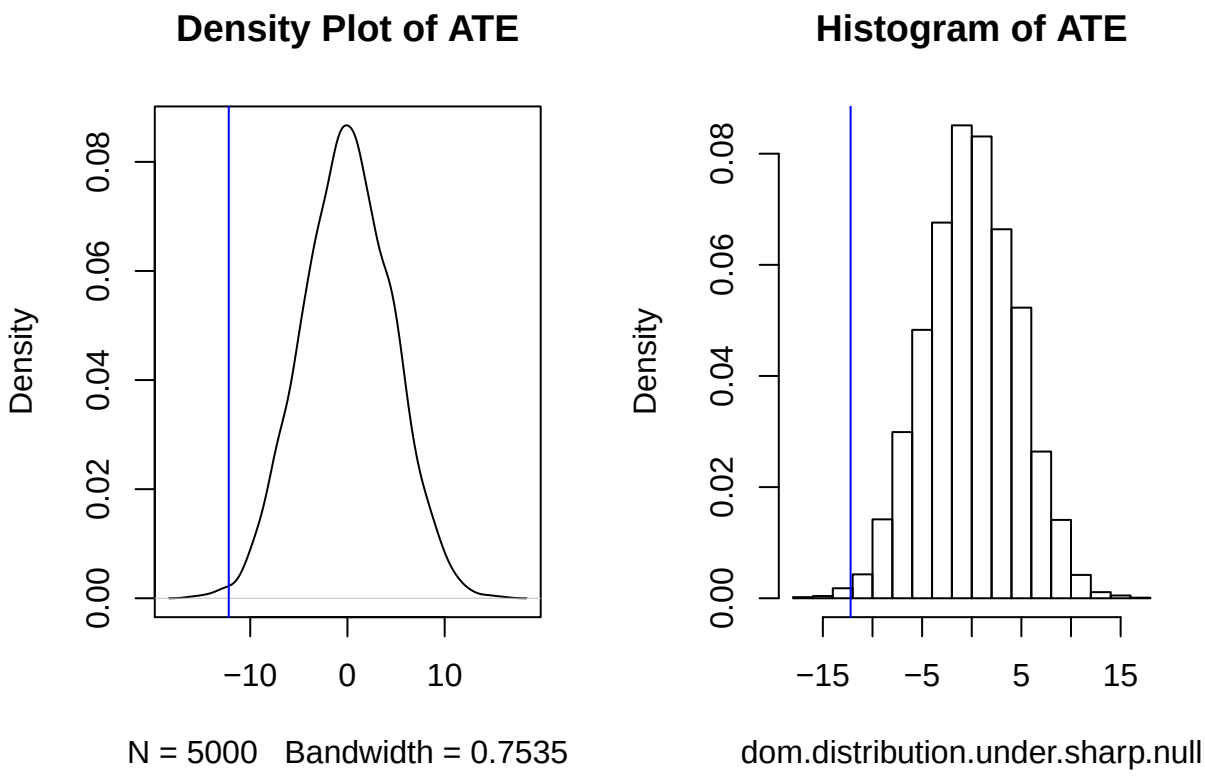
```
# visualizing the distribution
plot(density(dom.distribution.under.sharp.null), main = "difference of means distribution under sharp NU
```




```
hist(dom.distribution.under.sharp.null, main = "differences of means distribution under sharp NULL")
```




```
# More analysis on the distribution. Plotting the p value
par(mfrow = c(1,2))
plot(density(dom.distribution.under.sharp.null),
     main = "Density Plot of ATE")
abline(v = bids.ate, col = "blue")
hist(dom.distribution.under.sharp.null,
     main = "Histogram of ATE",
     freq = FALSE)
abline(v = bids.ate, col = "blue")
```




```
bids.ate
```

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

```
# observations with absolute value >= ate
sum(abs(dom.distribution.under.sharp.null) >= abs(bids.ate))
```

```
## [1] 39
```

```
# 2 tailed p-value
p2 <- mean(abs(dom.distribution.under.sharp.null) >= abs(bids.ate))
p2
```

```
## [1] 0.0078
```

```
# observations with value >= ate
sum(dom.distribution.under.sharp.null >= abs(bids.ate))
```

```
## [1] 17
```

```
# single tail p value
p1 <- mean(dom.distribution.under.sharp.null >= abs(bids.ate))
p1
```

```
## [1] 0.0034
```

Answer: The 2 sided p value from randomization inference is .0058

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).)

```
# p value using the analytic formulas

# p value is the area of the t distribution > t statistic (calculated above in a and b)
# We double it to get to the 2 sided value
# We'll look up a table to confirm

# Value from the table corresponding to t statistic of -2.8211 and degrees of freedom = 66
table.p1 <- .003159
table.p2 <- table.p1 * 2
table.p2
```

```
## [1] 0.006318
```

Answer: The p value from the t test is 0.006318

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?

p value from (e)(Regression): 0.006315 p value from (f)(Randomization inference): .0058

Answer: The p values from regression and randomization inference are different (.0063, .0058). The sample size is not big. The distribution of outcomes in control are normal, but not in treatment. Randomization inference should give us a more accurate result in this situation.

We may see more difference among the results if sample size were smaller. The result from RI would be more trustworthy in that case. With a larger sample size and normally ditributed outcomes, the difference in p values may not be much