# Problem Set 2

*Experiments and Causality*

## 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)
dt1 <- fread("./data/Clingingsmith.2009.csv")
```

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.

```
ATE <- dt1[ , .('group_mean' = mean(views)), keyby = .(success)] %>%
    .[ , diff(group_mean)]
ATE
```

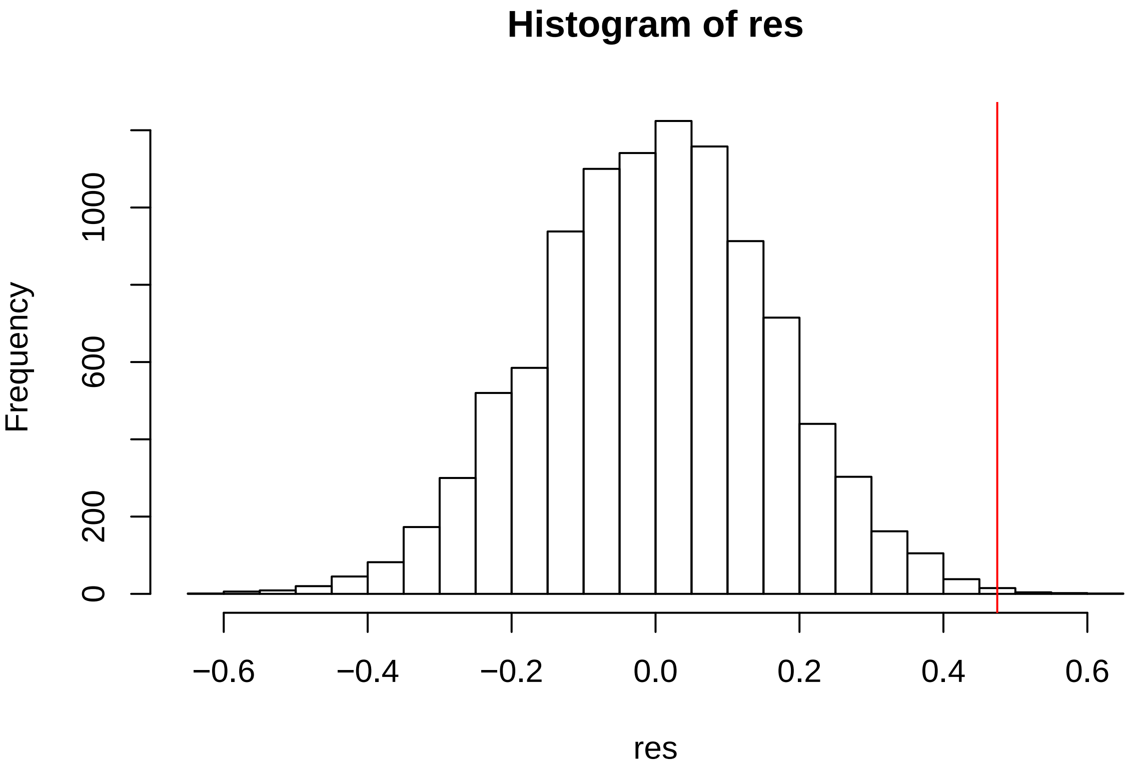
```
## [1] 0.4748337
```

**We see an ATE of about 0.475, meaning those who win the visa lottery see an average positive delta in views of about 0.475 versus those that do not win the visa 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.)

```
# specify number of simulations
n_sims = 10000
# init to NA values
res <- rep(NA , n_sims)
# loop and replicate the experiment
for(i in 1:n_sims) {
    res[i] <- dt1 %>%
        .[ , .('group_mean' = mean(views)), keyby = sample(success)] %>%
        .[ , diff(group_mean)]
}
```

```
# Plot the histogram
hist(res, breaks = 20, )
abline(v=ATE, col = "red")
```



**We see that when we replicate the experiment 10,000 times we see a relatively normal distribution centered around 0. The ATE of 0.475 that we were seeing is on an extreme tail end of the distribution. Hence, the ATE we computed may not be significant in the grand scheme of things.**

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?

```r
sum(res >= ATE)
```

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

**Of the simulated random assignments,** 14 **generate an estimated ATE which is at least as large as the ATE we estimated.**

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

```r
mean(res>=ATE)
```

```
## [1] 0.0014
```

**The implied one-tailed p-value is:** 0.0014

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?

```r
sum(abs(res) >= ATE)
```

```
## [1] 38
```

**Of the simulated random assignments,** 38 **generate an estimated ATE whose absolute value is at least as large as the ATE we estimated.**

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

```r
mean(abs(res) >= ATE)
```

```
## [1] 0.0038
```

**The implied two-tailed p-value is:** 0.0038

# 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.

```r
library(foreign)
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?

```r
# create a data table
dt2 <- data.table::data.table(d2)
```

```r
# report mean number of bills in each state
d2 %>%
  group_by(texas0_arkansas1) %>%
  summarize(mean_bills = mean(bills_introduced))
```

```
## # A tibble: 2 x 2
##   texas0_arkansas1 mean_bills
##              <int>      <dbl>
## 1                0       68.8
## 2                1       25.5
```

**We see that Texas introduced significantly more bills than Arkansas. However, is this a good measure of productivity? The bills introduced by Texas could be very minor. We cannot say for sure if Texas was more productive unless we know the impact the bills had. Going solely by average number of bills, Texas was far more productive.**

```r
# report mean number of bills by term
d2 %>%
  group_by(term2year) %>%
  summarize(mean_bills = mean(bills_introduced))
```

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

**We see that the difference in bills introduced by each state is lower when grouped by term. However, if you normalized the data using time, 2 year terms produce more bills as per the data. This does seem to suggest that more bills get passed during two year terms if you normalize by time. Term length is more likely to be causal rather than the state.**

```
# report mean number of bills by term and year
d2 %>%
  group_by(texas0_arkansas1, term2year) %>%
  summarize(mean_bills = mean(bills_introduced))
```

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

**The overall number of bills introduced in four year terms is higher. This makes sense given that there is more time to pass more bills. However, the delta between bills passed in two year terms versus four year terms is small. If we consider productivity a function of time, two year terms are more productive than four year terms.**

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

```
var0_tx = var(dt2[term2year == 0 & texas0_arkansas1 == 0, bills_introduced])
var1_tx = var(dt2[term2year == 1 & texas0_arkansas1 == 0, bills_introduced])
N_tx = dt2[texas0_arkansas1 == 0, .N]
m_tx = dt2[texas0_arkansas1 == 0 & term2year == 1, .N]
se_tx = sqrt((var0_tx/(N_tx - m_tx) + var1_tx/m_tx))
cat("The standard error of the estimated ATE for Texas is: ", se_tx)
```

```
## The standard error of the estimated ATE for Texas is:  9.345871
```

```
var0_ak = var(dt2[term2year == 0 & texas0_arkansas1 == 1, bills_introduced])
var1_ak = var(dt2[term2year == 1 & texas0_arkansas1 == 1, bills_introduced])
N_ak = dt2[texas0_arkansas1 == 1, .N]
m_ak = dt2[texas0_arkansas1 == 1 & term2year == 1, .N]
se_ak = sqrt((var0_ak/(N_ak - m_ak) + var1_ak/m_ak))
cat("The standard error of the estimated ATE for Arkansas is: ", se_ak)
```

```
## The standard error of the estimated ATE for Arkansas is:  3.395979
```

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

```
# compute the individual ATEs
ATE <- dt2[ , .('group_mean' = mean(bills_introduced)), keyby = .(term2year, texas0_arkansas1)]
ate_tx <- ATE[3, group_mean] - ATE[1, group_mean]
ate_ak <- ATE[4, group_mean] - ATE[2, group_mean]
cat("The ATE for Texas is: ", ate_tx, "\n")
```

```
## The ATE for Texas is:  -16.74167
```

```
cat("The ATE for Arkansas is: ", ate_ak)
```

```
## The ATE for Arkansas is:  -10.09477
```

```
# compute the overall ate
N_tot = dt2[ , .N]
ate_overall = ate_tx*(N_tx/N_tot) + ate_ak*(N_ak/N_tot)
cat("The overall ATE for both states combines is: ", ate_overall)
```

```
## The overall ATE for both states combines 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.

**Since the data is not normalized by time more often than not four year terms will have more bills simply because there is more time to pass them. Hence pooling data this way will bias results in favor of four year terms.**

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

```
se_overall = sqrt(se_tx^2 * (N_tx/N_tot)^2 + se_ak^2 * (N_ak/N_tot)^2)
cat("The estimated standard error for overall ATE is: ", se_overall)
```

```
## The estimated standard error for overall ATE is:  4.74478
```

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

```
# specify number of simulations
n_sims = 10000
# init to NA values
tex_res <- rep(NA , n_sims)
ak_res <- rep(NA , n_sims)
# loop and replicate the experiment
for(i in 1:n_sims) {
    tex_res[i] <- dt2[texas0_arkansas1 == 0] %>%
        .[ , .('group_mean' = mean(bills_introduced)), keyby = sample(term2year)] %>%
        .[ , diff(group_mean)]
    ak_res[i] <- dt2[texas0_arkansas1 == 1] %>%
        .[ , .('group_mean' = mean(bills_introduced)), keyby = sample(term2year)] %>%
        .[ , diff(group_mean)]
}
```

```
cat("The average treatment effect for Texas is: ", mean(tex_res), "\n")
```

```
## The average treatment effect for Texas is:  -0.06553958
```

```
cat("The average treatment effect for Arkansas is: ", mean(ak_res))
```

```
## The average treatment effect for Arkansas is:  -0.0006683007
```

```
cat("The two-tailed p-value using randomization inference for Texas is: ", mean(abs(tex_res) >= abs(ate_
```

```
## The two-tailed p-value using randomization inference for Texas is:  0.0854
```

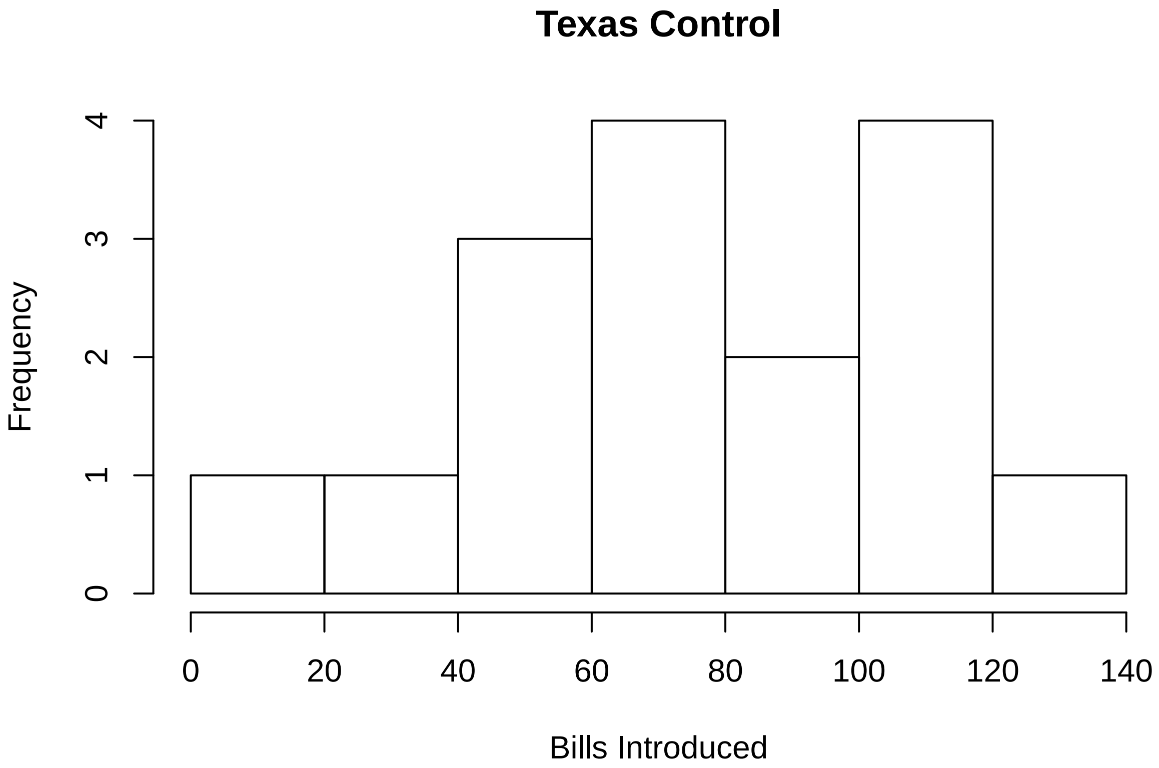
```
cat("The two-tailed p-value using randomization inference for Arkansas is: ", mean(abs(ak_res) >= abs(a
```

```
## The two-tailed p-value using randomization inference for Arkansas is:  0.0036
```

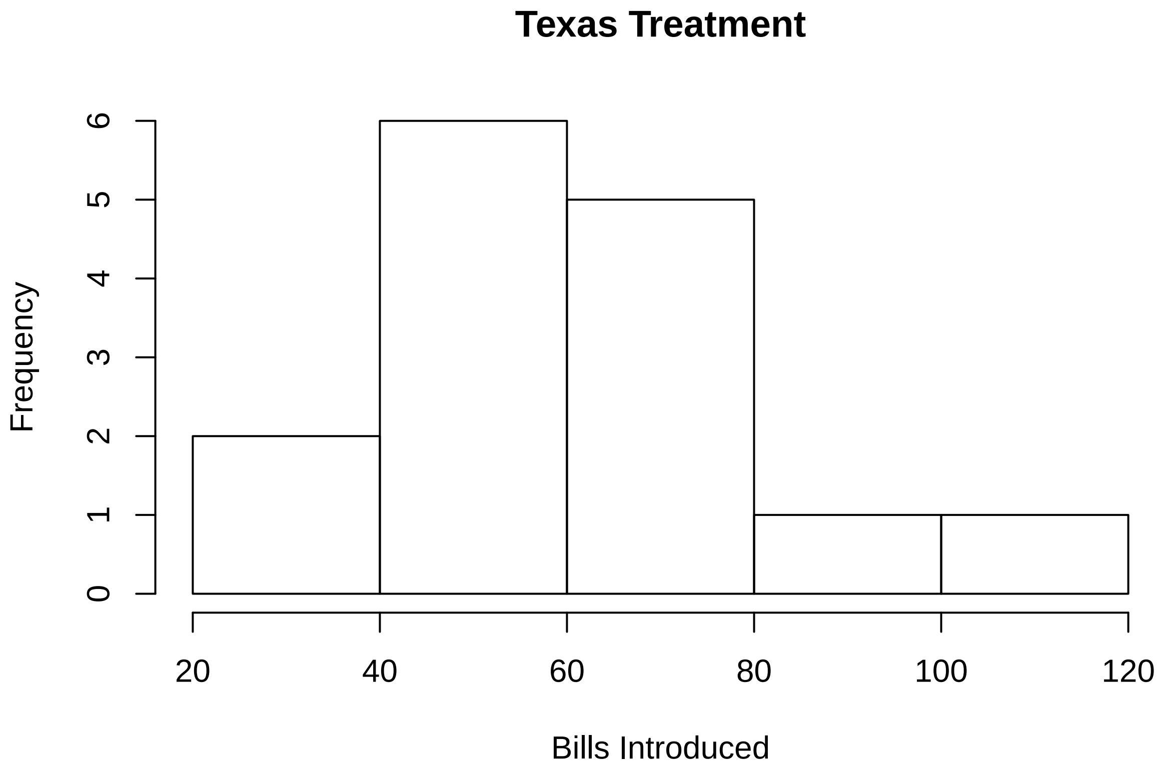
**As you can see the treatment effect under the sharp null is very close to zero for both states.**

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

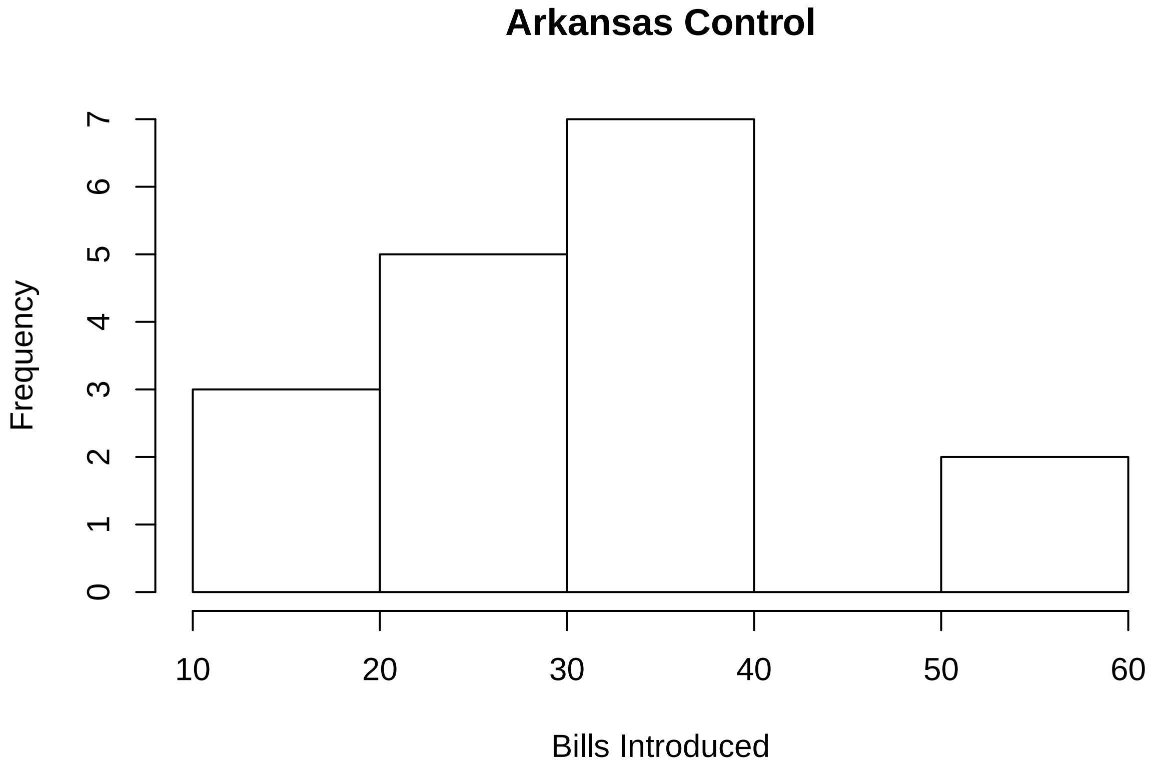
```r
hist(dt2[texas0_arkansas1 == 0 & term2year == 0, bills_introduced], breaks = 5, main = "Texas Control",
```



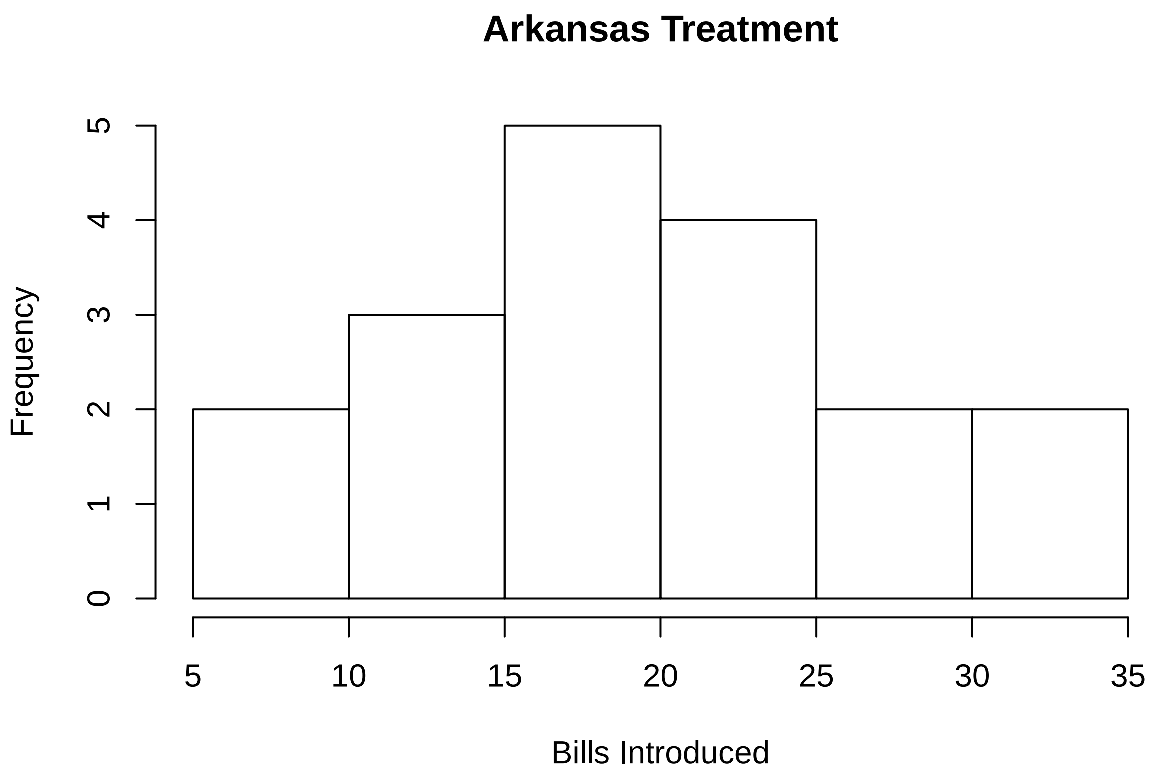
```r
hist(dt2[texas0_arkansas1 == 0 & term2year == 1, bills_introduced], breaks = 5, main = "Texas Treatment
```



```
hist(dt2[texas0_arkansas1 == 1 & term2year == 0, bills_introduced], breaks = 5, main = "Arkansas Contro
```



```
hist(dt2[texas0_arkansas1 == 1 & term2year == 1, bills_introduced], breaks = 5, main = "Arkansas Treatm
```


Arkansas Treatment
Frequency
0
1
2
3
4
5
5
10
15
20
25
30
35
Bills Introduced

## 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
d <- read.csv("./data/ggChapter3.csv", stringsAsFactors = FALSE)
dt3 <- fread("./data/ggChapter3.csv")
```

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 cluster assignments to the data table
dt3[, clusters := rep(1:7, each=2)]
```

```
# define and compute the information we need
sample <- sample(unique(dt3[["clusters"]]),length(unique(dt3[["clusters"]]))/2)
treatment <- is.element(dt3[["clusters"]], sample)
k = length(unique(dt3[["clusters"]]))
N = dt3[ , .N]
m = sum(treatment)
# append treatment to the data table
cbind(dt3, treatment)
```

```
##     Village  Y  D Block clusters treatment
##  1:       1  0  0     1        1     FALSE
##  2:       2  1  0     1        1     FALSE
##  3:       3  2  1     1        2      TRUE
##  4:       4  4  2     1        2      TRUE
##  5:       5  4  0     1        3     FALSE
##  6:       6  6  0     1        3     FALSE
##  7:       7  6  2     1        4      TRUE
##  8:       8  9  3     1        4      TRUE
##  9:       9 14 12     2        5     FALSE
## 10:      10 15  9     2        5     FALSE
## 11:      11 16  8     2        6     FALSE
## 12:      12 16 15     2        6     FALSE
## 13:      13 17  5     2        7      TRUE
## 14:      14 18 17     2        7      TRUE
```

```
# compute treatment means and control means
treat_out = dt3[treatment,c("Y","clusters")]
treat_out_mean = tapply(treat_out$Y, as.numeric(treat_out$cluster), mean)
control_out = treat_out
control_out_mean = tapply(control_out$Y, as.numeric(control_out$cluster), mean)
```

```
# compute the standard error
se = sqrt((1/(k-1))*((m*var(control_out_mean))/(N-m) + (N-m)*var(treat_out_mean)/m + 2*cov(control_out_
cat("The standard error is: ", se)
```

```
## The standard error is:  6.122685
```

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.

```
# re-init the data table
dt3 <- fread("./data/ggChapter3.csv")
# add cluster assignments to the data table
dt3[, clusters := c(1,2,3,4,5,6,7,7,6,5,4,3,2,1)]
```

```
# define and compute the information we need
sample <- sample(unique(dt3[["clusters"]]),length(unique(dt3[["clusters"]]))/2)
treatment <- is.element(dt3[["clusters"]], sample)
k = length(unique(dt3[["clusters"]]))
N = dt3[ , .N]
m = sum(treatment)
# append treatment to the data table
cbind(dt3, treatment)
```

```
##     Village  Y  D Block clusters treatment
##  1:       1  0  0     1        1     FALSE
##  2:       2  1  0     1        2      TRUE
##  3:       3  2  1     1        3      TRUE
##  4:       4  4  2     1        4     FALSE
##  5:       5  4  0     1        5     FALSE
##  6:       6  6  0     1        6      TRUE
##  7:       7  6  2     1        7     FALSE
##  8:       8  9  3     1        7     FALSE
##  9:       9 14 12     2        6      TRUE
## 10:      10 15  9     2        5     FALSE
## 11:      11 16  8     2        4     FALSE
## 12:      12 16 15     2        3      TRUE
## 13:      13 17  5     2        2      TRUE
## 14:      14 18 17     2        1     FALSE
```

```
# compute treatment means and control means
treat_out = dt3[treatment,c("Y","clusters")]
treat_out_mean = tapply(treat_out$Y, as.numeric(treat_out$cluster), mean)
control_out = treat_out
control_out_mean = tapply(control_out$Y, as.numeric(control_out$cluster), mean)
```

```
# compute the standard error
se = sqrt((1/(k-1))*((m*var(control_out_mean))/(N-m) + (N-m)*var(treat_out_mean)/m + 2*cov(control_out_
cat("The standard error is: ", se)
```

```
## The standard error is:  0.4762897
```

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 second method of clustering has a lower standard error because it better randomizes the Y variable when pairing villages. This implies that a cluster design should be selected such that it randomizes the data properly and has an overall low standard error.**

## 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)?

```r
# store the data we are given
cost_per_ad = 0.1
users = 1000000
profit_per_phone = 100
users_buy_rate_no_ads = 0.5
phones_sold_no_ads = (users_buy_rate_no_ads/100)*users
```

```r
# assuming all users are shown ads
ad_cost = cost_per_ad * users
# number of phones needed to be sold to cover ad cost
num_phones_cover_ad_cost = ad_cost/profit_per_phone
# convert to percentage of users
percent_users = num_phones_cover_ad_cost/users
cat("The increase in user buy rate needs to be greater than: ", percent_users*100)
```

```
## The increase in user buy rate needs to be greater than:  0.1
```

**More than 0.6% of all visitors to our newspaper's website need to purchase a phone in order to produce a positive ROI.**

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?

```
meas_eff = 0.2/100
users_control = users/2
users_treat = users/2
p = (users_buy_rate_no_ads*users_control/100 + (users_buy_rate_no_ads/100+meas_eff)*users_treat)/users
se = sqrt(p*(1-p)*(1/users_treat + 1/users_control))
cat("The standard error for part b is: ", se, "\n")
```

```
## The standard error for part b is:  0.0001544539
```

```
ci = 1.96*se
upper_ci = meas_eff + ci
lower_ci = meas_eff - ci
cat("The 95% CI is between: ", lower_ci*100, " and ", upper_ci*100, " percentage points.")
```

```
## The 95% CI is between:  0.169727  and  0.230273  percentage points.
```

- **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 we calculated is well beyond the minimum we calculated in part (a). Thus, we would recomment running the experiment.**

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?

```
users_control = users*0.01
users_treat = users*0.99
p = (users_buy_rate_no_ads*users_control/100 + (users_buy_rate_no_ads/100+meas_eff)*users_treat)/users
se = sqrt(p*(1-p)*(1/users_treat + 1/users_control))
cat("The standard error for part b is: ", se, "\n")
```

```
## The standard error for part b is:  0.0008367373
```

```
ci = 1.96*se
upper_ci = meas_eff + ci
lower_ci = meas_eff - ci
cat("The 95% CI is between: ", lower_ci*100, " and ", upper_ci*100, " percentage points.")
```

```
## The 95% CI is between:  0.0359995  and  0.3640005  percentage points.
```

# 5. Sports Cards

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

```
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.

```
dt5 = fread("./data/listData.csv")
```

```
control = dt5[uniform_price_auction == 0, bid]
treat = dt5[uniform_price_auction == 1, bid]
t.test(treat, control)
```

```
## 
##  Welch Two Sample t-test
## 
## data:  treat and control
## 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, -3.557] implies that if this experiment was repeated several times, there is a 95% chance that the ATE would fall in this range.**

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.

```
model = lm(bid ~ uniform_price_auction, data = dt5)
summary(model)
```

```
##
## Call:
## lm(formula = bid ~ uniform_price_auction, data = dt5)
##
## 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.

```
confint(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.

```
cat("The p-value is: ", summary(model)$coefficients[8] )
```

```
## The p-value is:  0.006314796
```

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

```
# compute the ate
ATE5 <- dt5[ , .('group_mean' = mean(bid)), keyby = .(uniform_price_auction)] %>%
    .[ , diff(group_mean)]
ATE5
```

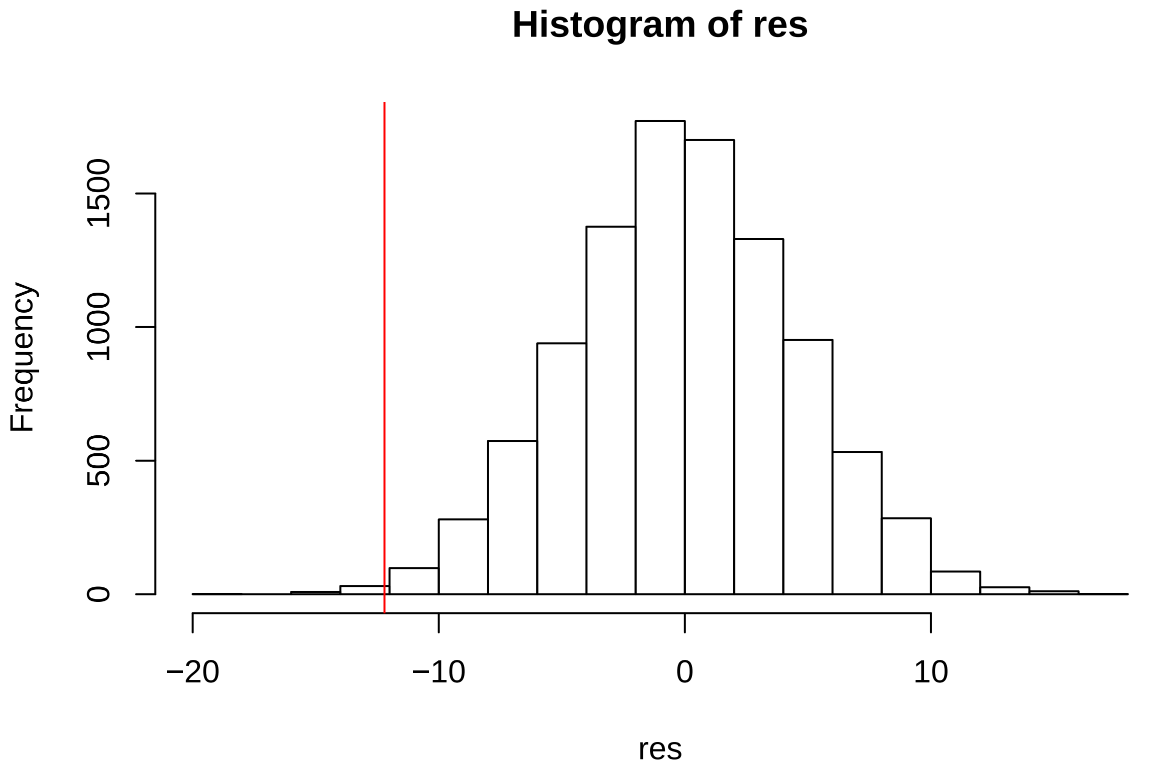
```
## [1] -12.20588
```

```
# specify number of simulations
n_sims = 10000
# init to NA values
res <- rep(NA , n_sims)
# loop and replicate the experiment
for(i in 1:n_sims) {
    res[i] <- dt5 %>%
        .[ , .('group_mean' = mean(bid)), keyby = sample(uniform_price_auction)] %>%
        .[ , diff(group_mean)]
}
```

```
mean(abs(res) >= abs(ATE5))
```

```
## [1] 0.0073
```

```
hist(res, breaks = 20)
abline(v=ATE5, col = "red")
```



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).)

```
tstat = abs(t.test(treat, control)$statistic)
df = t.test(treat, control)$parameter
analytic_pval = 2*pt(tstat, df, lower=FALSE)
cat("The analytic p-value is: ", analytic_pval)
```

```
## The analytic p-value is:  0.006420778
```

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?

**There is a small delta between the values. This is most likely due to the sample size. Increasing the sample size would likely reduce the delta even further.**