ANSWERS: Problem Set #2

June 15, 2018

```
library(data.table)
library(stargazer)
library(foreign)
library(magrittr)

nReps = 10000 # this is for the final run.
              # i'd decrease this while your writing
              # unless you like getting coffee.

# define functions

est.ate <- function(outcomes, treatment) {
  # estimates ates
  mean(outcomes[treatment==1]) - mean(outcomes[treatment==0])
  }
assign.treatment <- function(n) {
  sample(0:1, n, replace=TRUE)
  }
ri.sim.once <- function(outcomes) {
  est.ate(outcomes, assign.treatment(length(outcomes)))
  }
calc_est_ate <- function(group, outcome) {
  return(mean(outcome[group==1], na.rm = TRUE) - mean(outcome[group==0], na.rm = TRUE))
  }
```

FE exercise 3.6

The Clingingsmith, Khwaja, and Kremer study discussed in section 3.5 may be be used to 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.

a. Conduct 10,000 simulated random assignments under the sharp null hypothesis.

And so, the average treatment effect is **0.4748337**.

```
distribution.under.sharp.null <- replicate(
  nReps,
  ri.sim.once(d3.6$views)
  )
```

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

With those functions defined, we can answer our question. How many are bigger?

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

```
## [1] 19
```

And so, there are **19**. To calculate a simple p-val estimate, let's just use some simple "count and divide" probability. If there are 19, the the probability can be pretty easily calculated as that quantity divided by the total number of instances, which we can figure out using `length`. This value is 0.0019.

c. What is the implied one-tailed p-value?

```
n.bigger / length(distribution.under.sharp.null)
```

```
## [1] 0.0019
```

And so, our estimate for the p-value here, under the sharp null is **0.0019**.

d. 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?

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

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

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

That is: there are **35** which are larger in absolute terms, and so, the p-val can be found by calculating `n.bigger.abs / length(distribution.under.sharp.null)`, which is **0.0035**.

FE exercise 3.8

Naturally occurring experiments sometimes involve what is, in effect, block random assignment. For example, Titunik 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. An interesting outcome variable is 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.

```
d3.8 <- read.dta("http://hdl.handle.net/10079/s1rn910")
names(d3.8) <- c("group", "bills", "st")
dt3.8 <- data.table(d3.8)
```

a. For each state, estimate the effect of having a two-year term on the number of bills introduced.

```
calc_est_ate <- function(group, outcome) {
  ate = mean(outcome[group==1], na.rm = TRUE) -
    mean(outcome[group==0], na.rm = TRUE)

    return(ate)

  }
dt3.8[ ,  .(ate = calc_est_ate(group, bills)), by = st]
```

```
##    st       ate
## 1:  0 -16.74167
## 2:  1 -10.09477
```

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

```
calc_est_se <- function(group, outcome) {
  ## This is from Equation 3.6 in the FE book.
  y0 <- outcome[group == 0]
  y1 <- outcome[group == 1]
  m <- length(y1)
  N <- m + length(y0)
  v0 <- var(y0)
  v1 <- var(y1)
  return(sqrt(v0 / (N - m) + v1 / m))
  }
dt3.8[ , .(se = calc_est_se(group, bills)), by = st]
```

```
##    st       se
## 1:  0 9.345871
## 2:  1 3.395979
```

So, we could print these next to each other, and probably also calculate a test.statistic to go along with it using a little bit of chaining.

```
dt3.8[ , .(ate = calc_est_ate(group, bills),
           se  = calc_est_se(group, bills)),
      by = st] %>%
  .[ , .(ate, se, t = ate/se), by = st] %>%
  .[ , .(ate , se, t , p_val = pt(q = t, df = nrow(dt3.8)))]
```

```
##          ate       se         t      p_val
## 1: -16.74167 9.345871 -1.791344 0.03891149
## 2: -10.09477 3.395979 -2.972566 0.00205935
```

```
dt3.8[st==0, summary(lm(bills ~ group))]
```

```
##
## Call:
## lm(formula = bills ~ group)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -58.875 -16.004   0.867  15.496  54.125
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   76.875      6.588  11.670 1.78e-12 ***
## group        -16.742      9.470  -1.768   0.0876 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 26.35 on 29 degrees of freedom
## Multiple R-squared:  0.09728,    Adjusted R-squared:  0.06615
## F-statistic: 3.125 on 1 and 29 DF,  p-value: 0.08761
```

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

```
overallAte <- dt3.8[ , .(ate  = calc_est_ate(group, bills),
           prop.st = .N/nrow(dt3.8)),
      by = st] %>%
  .[ , .(weighted = sum(ate*prop.st))]
overallAte
```

```
##    weighted
## 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.

```
dt3.8[ , mean(group), by = st]
```

```
##    st        V1
## 1:  0 0.4838710
## 2:  1 0.5142857
```

The treatment assignment probabilities are different in Arkansas and Texas, such that the treatment group over represents Arkansas. Therefore, if outcomes were higher in the treatment group, it might reflect differences between Texas and Arkansas rather than an effect of the treatment.

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

```
dt3.8[ , .(se      = calc_est_se(group, bills),
           prop.st = .N/nrow(dt3.8)),
      by = st] %>%
  .[ , .(overall.se = sqrt(sum(se^2 * prop.st^2)))]
```

```
##    overall.se
## 1:    4.74478
```

```
dt3.8[ , .(se      = calc_est_se(group, bills),
           prop.st = .N/nrow(dt3.8)),
      by = st] %>%
  .[ , .(x = se^2 * prop.st^2)] %>%
  .[ , .(x = sum(x))] %>%
  .[ , .(x = sqrt(x))]
```

```
##          x
## 1: 4.74478
```

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

Let's think about what we've got to do here, there are a few steps, but this is a useful exercise.

1. We need to block random assign. This means: complete random assignment w/in each block.
2. We need to calculate an ATE within each block.
3. We need to calculate the overall ATE from these block ATEs.
4. Repeat several times.

```
d <- data.table(read.dta("http://hdl.handle.net/10079/s1rn910"))
setnames(d, c("group", "bills", "st"))

overallAte <- d[ , .(
  ate = calc_est_ate(group = group, outcome = bills),
  prop_st = .N/nrow(d)), by = st] %>%
  .[ , .(overallAte = sum(ate *prop_st))]

# block random assign, which is just shuffling things around.
d <- d[ , group := sample(group), by = st]

# from here, we've done everything before...
d[ , group := sample(group),
   by = st] %>%
```

```r
  .[ , .(ate  = calc_est_ate(group, bills),
         prop.st = .N/nrow(dt3.8)),
     by = st ] %>%
  .[ , sum(ate*prop.st)]
```

```
## [1] -5.209745
```

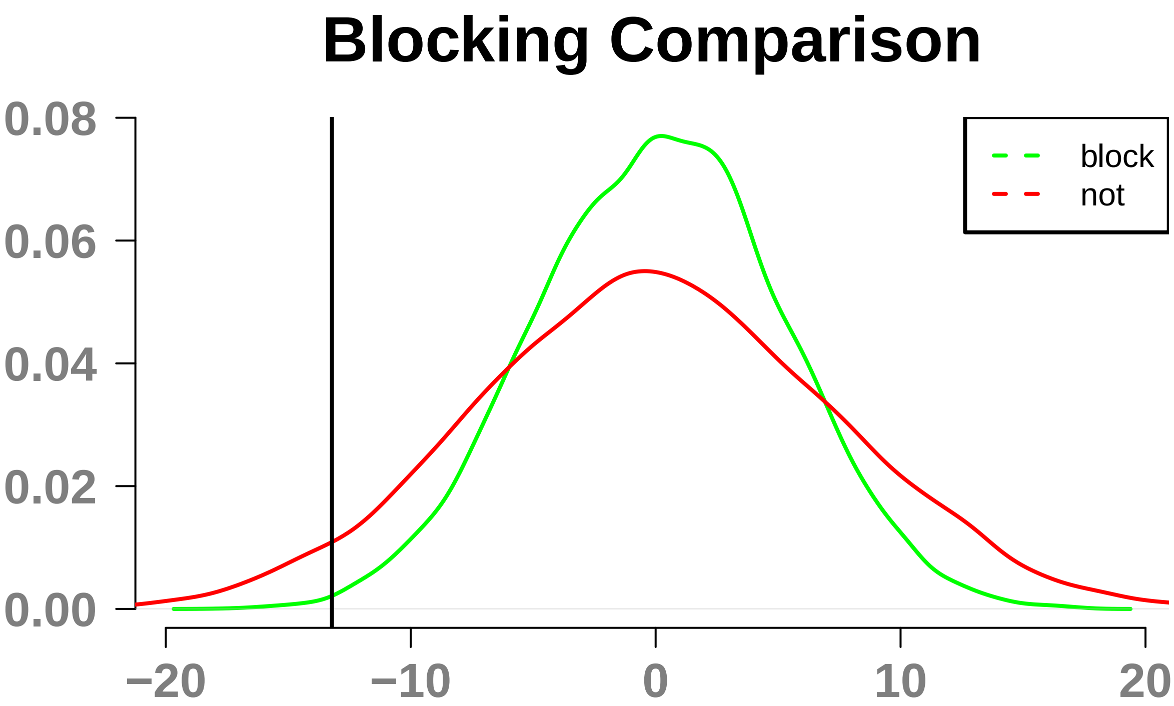
```r
# so, to do it a bunch of times, just wrap that in a function.
block <- function() {
  d[ , group := sample(group), by = st][ ,
    # calculate the ate and prop.st
    .(ate  = calc_est_ate(group, bills),
      prop.st = .N/nrow(dt3.8)), by = st ][ ,
    # and calculate the weighted ATE
    sum(ate*prop.st)
  ]
}

not.block <- function() {
  d[ , group := sample(group)][ , calc_est_ate(group, bills)]
}


res.block <- replicate(nReps, block())
res.notblock <- replicate(nReps, not.block())

mean(abs(res.block) > abs(as.numeric(overallAte)))
```

```
## [1] 0.0069
```

```r
mean(abs(res.notblock) > abs(as.numeric(overallAte)))
```

```
## [1] 0.0724
```

```r
source("http://ischool.berkeley.edu/~d.alex.hughes/code/pubPlot.R")
plot(density(res.block), col = "green",
     main = "Blocking Comparison",
     xlab = NA, ylab = NA)
lines(density(res.notblock), col = "red")
abline(v=overallAte)
legend("topright", legend = c("block", "not"), col = c("green", "red"),
       lty = 2)
```



```
mean(abs(res.block) > abs(as.numeric(overallAte)))
```

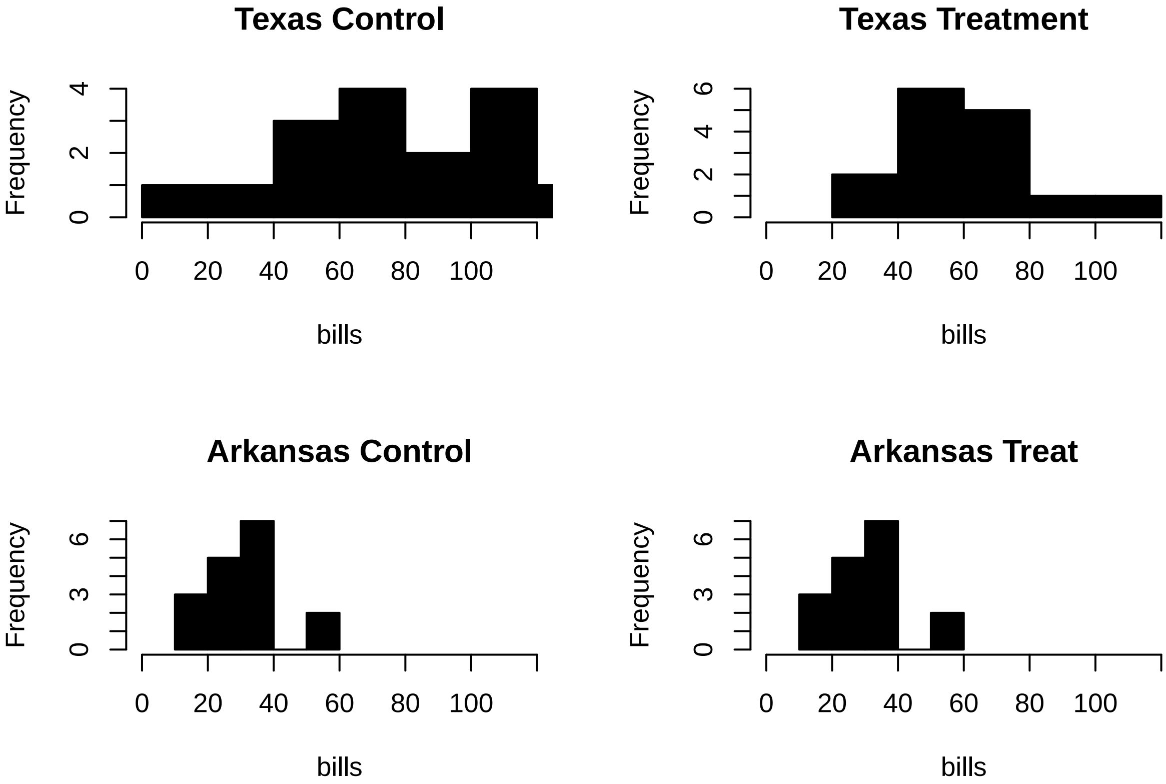
```
## [1] 0.0069
```

```
mean(abs(res.notblock) > abs(as.numeric(overallAte)))
```

```
## [1] 0.0724
```

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

```
par(mfrow = c(2,2))
dt3.8[st==0 & group==0,
      hist(bills, col = "black",
           main = "Texas Control",
           xlim = c(0, 120))]
dt3.8[st==0 & group==1,
      hist(bills, col = "black",
           main = "Texas Treatment",
           xlim = c(0, 120))]
dt3.8[st==1 & group==0,
      hist(bills, col = "black",
           main = "Arkansas Control",
           xlim = c(0, 120))]
dt3.8[st==1 & group==0,
      hist(bills, col = "black",
           main = "Arkansas Treat",
           xlim = c(0, 120))]
```



FE exercise 3.11

Use the data in 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.*)

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.

```
library(data.table)
d <- fread("http://isps.its.yale.edu/isps/public/Gerber_Green_FEDAI_2012/Chapter-3/GerberGreenBook_Chap

calc_cluster_se <-function(d){
   k <- d[ , length(unique(cluster))]
   N <- nrow(d)
   m <- 6 # this is just baked into the problem

   cluster_means <- d[ , .(y0 = mean(Y), y1 = mean(D)), by = .(cluster)]

   var1 <- cluster_means[ , var(y1)]
   var0 <- cluster_means[ , var(y0)]
   cov  <- cluster_means[ , cov(y0, y1)]

   # Equation 3.22.
   return(sqrt( 1/(k-1) * ( m/(N-m)*var0 + (N-m)/m*var1 + 2*cov)) )
}
```

```
calc_cluster_se(d[ , cluster := rep(1:7, each=2)])
```

```
## [1] 4.918953
```

```
## one could use the population rather than the
## sample variance; this is an easy toggle.
gg.var <- function(x) {
  sum((x-mean(x))^2)/(length(x))
  }
gg.cov <- function(x,y) {
  sum( ((x - mean(x))*(y - mean(y))) / length(x) )
}

gg_calc_cluster_se <-function(d){
   k <- d[ , length(unique(cluster))]
   N <- nrow(d)
   m <- 6 # this is just baked into the problem

   cluster_means <- d[ , .(y0 = mean(Y), y1 = mean(D)), by = .(cluster)]

   var1 <- cluster_means[ , gg.var(y1)]
   var0 <- cluster_means[ , gg.var(y0)]
   cov  <- cluster_means[ , gg.cov(y0, y1)]

   # Equation 3.22.
   return(sqrt( 1/(k-1) * ( m/(N-m)*var0 + (N-m)/m*var1 + 2*cov)) )
}

d[ , cluster := rep(1:7, each =2)]
d[ , cluster:= c(1,2,3,4,5,6,7,7,6,5,4,3,2,1)]
gg_calc_cluster_se(d)
```

```
## [1] 1.171092
```

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.

```
calc_cluster_se(d[ , cluster := c(1:7,7:1)])
```

```
## [1] 1.264924
```

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 implications for cluster randomized experiments fits with our discussion in the last class. When we are able to construct clusters with high variance inside the cluster and relatively small variance between the clusters, we generate more efficient estimates!

Another way to think of it is that the variation between clusters was extremely high in the first clustering scheme. In other words, in the first scheme, the cluster was highly predictive of the level of potential outcomes. This means that one very high or low cluster being included in the treatment or control group had a large effect on the ATE estimate, resulting in a high standard error because the ATE is likely to differ by a great deal from assignment to assignment. In the second scheme, the average composition of each cluster was more similar across clusters, meaning there will be less variation in the ATE from assignment to assignment. The lesson for experimental design is that you ideally want your cluster means to be similar, and that experiments where clusters differ dramatically will have much higher standard errors.

More Practice #1

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)? **A user seeing an ad needs to increase profitability by at least 10 cents for the ad campaign to be worth the coast. If Apple makes $100 on each iPhone, an increased probability of sale of 0.1 percent (0.001) is worth an increase of 10 cents.**

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?

- **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.

```
N = 1e6
p1 = .007
p2 = .005  # Baseline of .005 + .002 measured treatment effect.

calc_ci <- function(p1,p2,n1,n2){
   x1 = p1*n1
   x2 = p2*n2
   xbar = abs(p1-p2)
   p = (x1+x2)/(n1+n2)
   se = sqrt(p*(1-p)*(1/n1 + 1/n2))
   ci = c(xbar - se*1.96, xbar + se*1.96)
   return(list(ci = ci, xbar = xbar, se = se))
   }

n1 = N*.5
n2 = N*.5

ci1 <- calc_ci(p1,p2,n1,n2)
ci1
```

```
## $ci
## [1] 0.00169727 0.00230273
##
## $xbar
## [1] 0.002
##
## $se
## [1] 0.0001544539
```

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

We noted in part (a) that an effect of 0.1% would be sufficient for the campaign to be profitable. The lower tail of the 95% confidence interval in part (b) is larger than this value, meaning that Apple should look at this result and be very confident that the ad campaign is profitable. Therefore, as the newspaper attempting to argue that our ads work, we would recommend running this 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?

```
n1 = 0.99*N
n2 = 0.01*N
ci2 <- calc_ci(p1,p2,n1,n2)
ci2
```

```
## $ci
## [1] 0.000359995 0.003640005
##
## $xbar
## [1] 0.002
##
## $se
## [1] 0.0008367373
```

And so the width of the ci will be 0.00328.

More Practice #2

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

```
d2 <- read.csv("https://docs.google.com/spreadsheets/d/1M2wSeI4xB1YWoHAjzTNvmJHBLwbaewOf8FHMSWRQSb8/pub
dt2 <- data.table(d2)
```

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.

```
res3 <- dt2[ , .(m = mean(bid),
                 v = var(bid),
                 n = .N),
             by = uniform_price_auction] %>%
  .[ , .(ATE = m[1] - m[2],
         SE  = sqrt(v[1]/n[1] + v[2]/n[2]))]

res3$ATE + qt(c(0.025, 0.975),df = nrow(dt2) - 1) %o% res3$SE
```

```
##            [,1]
## [1,] -20.841756
## [2,]  -3.570009
```

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

This confidence interval indicates there is a 95% probability that the true ATE falls within this range. **In more CORRECT language, this means that this confience interval will overlap with the**

population level treatment effect in 95 of 100 times that the procedue is repeated, under the same sampling regime.

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
m1 <- lm(bid ~ uniform_price_auction, data = dt2)
stargazer(m1, type = "latex", header = FALSE)
```

Table 1:

	Dependent variable:
	bid
uniform_price_auction	-12.206^{***}
	(4.327)
Constant	28.824^{***}
	(3.059)
Observations	68
R^2	0.108
Adjusted R^2	0.094
Residual Std. Error	17.839 (df = 66)
F Statistic	7.959^{***} (df = 1; 66)
Note:	*p<0.1; **p<0.05; ***p<0.01

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

```r
summary(m1)$coef[2,1] + qt(c(0.025, 0.975), df = nrow(dt2) - 1) %o% summary(m1)$coef[2,2]
```

```
##             [,1]
## [1,] -20.841756
## [2,]  -3.570009
```

```r
# or use built in methods
confint(m1, level = .95)
```

```
##                           2.5 %    97.5 %
## (Intercept)            22.71534 34.931716
## 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.

Here, the p-value that we're interested in is the p-value that is associated with the F-statistic for the overall significance of the entire ensemble of regressors. In this case, the ensemble is just the intercept (the outcomes in the control group) and the treatment effect. Together these recover the mean outcome in the treatment group.

We can recover the p-value of this F-statistic from the regression object.

The F statistic for this regression model is 8, which has an associated p-value of 0.0063148.

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

```
setnames(dt2, "uniform_price_auction", "treat")
true.ate <- dt2[ , mean(bid[treat==1]) - mean(bid[treat==0])]

dt2[, treat := sample(treat)][ ,
      mean(bid[treat==1]) - mean(bid[treat==0])
      ]
```

```
## [1] -7.911765
```

```
# to do it a bunch of times... wrap in a function and replicate
bar <- function() {
  dt2[, treat := sample(treat)][ ,
      mean(bid[treat==1]) - mean(bid[treat==0])
      ]
  }

dist.under.null <- replicate(nReps, bar())
mean(abs(dist.under.null) > abs(true.ate))
```

```
## [1] 0.0061
```

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).)

```
2 * pt(res3$ATE / res3$SE, df = nrow(dt2))
```

```
## [1] 0.00626697
```

```
summary(m1)$coefficients[2,4]
```

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

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 aren't much different actually, which is typical when we have around this many observations and the data is well-behaved. If we had more data, then no matter the behavior, we would expect the distributions, and hence the p-values, to be very similar.