Problem Set 2

Ted Dunmire

Thursday June 11, 2015

1. FE, exercise 3.6

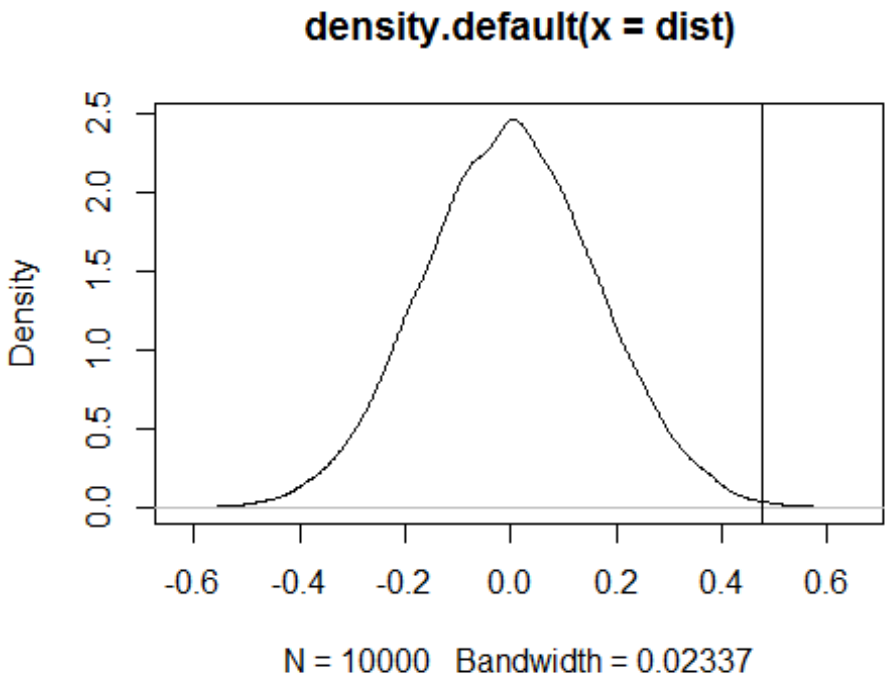
First I will retrieve the dataset and save it for future use. What I am now going to do is randomly generate a set of zeroes and ones to denote treatment or control groups. Then I will estimate the ATE for an experiment based on whether a subject was in the treatment or control group.

```
#inspiration for this code is credited to David Broockman and David Reily
data3.2 <- read.csv("http://hdl.handle.net/10079/6hdr852")

assign <- function(num) { #num will be the number of subjects in the
experiment
  sample(c(rep(0,num/2), rep(1,num/2)))
}

calc_ate <- function(result, treatment) {
  mean(result[treatment == 1]) - mean(result[treatment == 0])
}

dist <- replicate(10000, calc_ate(data3.2$views,
assign(length(data3.2$views))))

est.ate <- mean(data3.2$views[data3.2$success == 1]) -
mean(data3.2$views[data3.2$success == 0])
est.ate
```

```
## [1] 0.4748
```

Now let's plot the estimated ate from the experiment relative to the distribution I've assembled. Then I can see how many of my observations are greater than or equal to my estimated ate and how many absolute values are greater than or equal to my estimated ate. The mean of these counts is the implied p-value for a one and two tailed test respectively.

```
plot(density(dist))
abline(v = est.ate) #hmmm looks unlikely due to chance under the sharp null
```




```
#how many values in the simulation are greater than or equal to the est.ate
one_tail_sum = sum(dist >= est.ate)
one_tail_p = mean(dist>= est.ate) #one tailed p-value

abs_dist <- abs(dist)
two_tail_sum = sum(abs_dist >= est.ate)
two_tail_p = mean(abs_dist >= est.ate)
```

So I can say I have 18 observations where the value is greater than or equal to the estimated ate which yields an implied p-value of 0.0018. For the two-tailed version I used the absolute value of my distribution and found 31 which yielded 0.0031.

2. FE exercise 3.8 (also plot histograms for treatment and control in each state)

   Download the data using the foreign package in R

```
library(foreign)

## Warning: package 'foreign' was built under R version 3.1.3

lottery <- read.dta("Titiunik_WorkingPaper_2010.csv.dta")
```

   a. Estimate effect of having a two-year term on the number of bills by state

```
texas <- lottery[lottery$texas0_arkansas1 == 0,c(1,2)]
texasATE <- mean(texas$bills_introduced[texas$term2year == 1]) -
mean(texas$bills_introduced[texas$term2year == 0])
```

```
arkansas <- lottery[lottery$texas0_arkansas1 == 1, c(1,2)]
arkansasATE <- mean(arkansas$bills_introduced[arkansas$term2year == 1]) -
mean(arkansas$bills_introduced[arkansas$term2year == 0])
```

So what I have done is subsetted each state and then taken the mean number of bills passed for when the senators have two year terms and subtracted the mean number of bills for when the senators have 4 year terms. What I have found is that in Texas the effect of having a two year term is -16.7417 fewer bills and in Arkanasas it is -10.0948 fewer bills when a senator has a two year term.

b. Estimate standard error of estimated ATE

```
#Doing Texas first
m = length(texas[texas$term2year == 1,1])
avg <- mean(texas$bills_introduced[texas$term2year == 1])
var2year <- (1/(m-1)) * sum((texas$bills_introduced[texas$term2year == 1]
- avg)^2)

avg = mean(texas$bills_introduced[texas$term2year == 0])
var4year <- (1/(length(texas[,1]) - m -1)) *
sum((texas$bills_introduced[texas$term2year == 0] - avg)^2)

texasSE <- sqrt(var4year/(length(texas[,1]) - m) + var2year/m)
```

This leads me to conclude the standard error for Texas's estimated ATE is 9.3459

```
#repeat for Arkansas
m = length(arkansas[arkansas$term2year == 1,1])
avg <- mean(arkansas$bills_introduced[arkansas$term2year == 1])
var2year <- (1/(m-1)) * sum((arkansas$bills_introduced[arkansas$term2year
== 1] - avg)^2)

avg = mean(arkansas$bills_introduced[arkansas$term2year == 0])
var4year <- (1/(length(arkansas[,1]) - m -1)) *
sum((arkansas$bills_introduced[arkansas$term2year == 0] - avg)^2)

arkansasSE <- sqrt(var4year/(length(arkansas[,1]) - m) + var2year/m)
```

So I can say the standard error for Arkansas's estimated ATE is 3.396.

c. Use equation 3.10 to esimate the overall ATE

```
estATE <- length(lottery[lottery$texas0_arkansas1 == 0,
1])/length(lottery[,1]) * texasATE +
length(lottery[lottery$texas0_arkansas1 == 1, 1])/length(lottery[,1]) *
arkansasATE
```

This shows that the overall ATE for both states is -13.2168.

d. Why does poooling the data of the two states lead to a biased estimate of the overall ATE?

The reason this generates a biased estimate of the ATE is that the probability of a random subject being assigned to the treatment group is not the same across the blocks. In Texas, there is a 15/31 or 48.4% chance of being in the treatment group versus in Arkansas where there is an 18/35 or 51.4% chance of being in the treatment. The probability of assignment to the treatment would have to be the same across all blocks for the overall ATE to be unbiased. To demonstrate this I will pool the two states together and show an ATE

```
pooledATE <- mean(lottery$bills_introduced[lottery$term2year == 1]) -
mean(lottery$bills_introduced[lottery$term2year == 0])
```

Which shows an ATE of -14.5152 versus the -13.2168 from part c.

e. Use equation 3.12 to estimate the standard error for the overall ATE

```
overallSE <- sqrt(arkansasSE^2 * (35/66)^2 + texasSE^2 * (31/66)^2)
```

From this equation I can see the overall standard error for the ATE is 4.7448.

f. Use randomization inference to test the sharp null that the true treatment effect is zero.

I will use my assign() and calc_ate() functions from problem 1 to run a randomization inference and then see where the esimated ate falls.

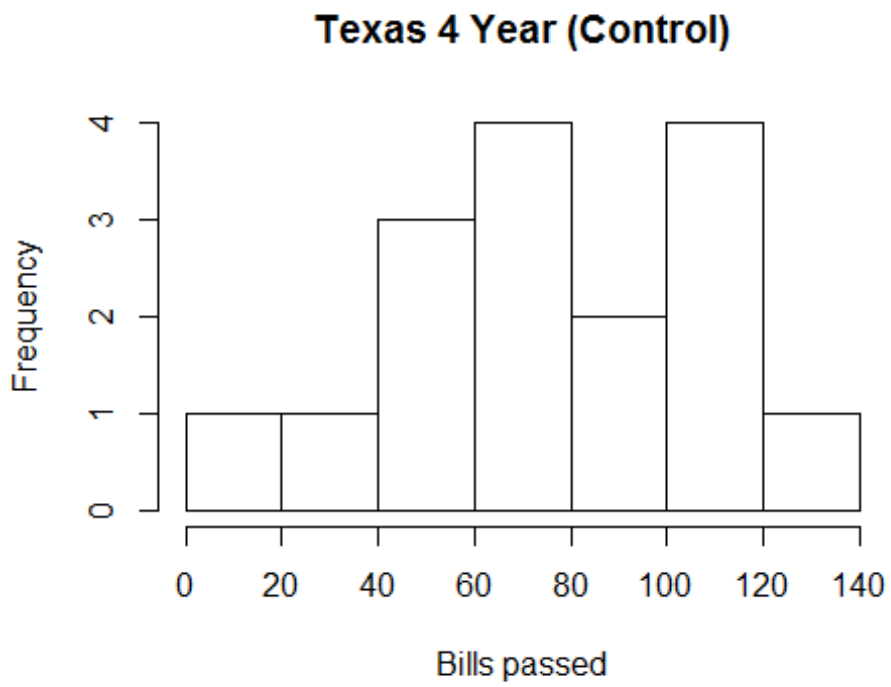
```
dist2 <- replicate(10000, calc_ate(lottery$bills_introduced,
assign(length(lottery$bills_introduced))))
#conduct a one-tailed "t test" because my estimate is negative
lottery_p <- mean(dist2 <= estATE)
```

This shows a pretty small p-value 0.0348 for a one-tailed test.

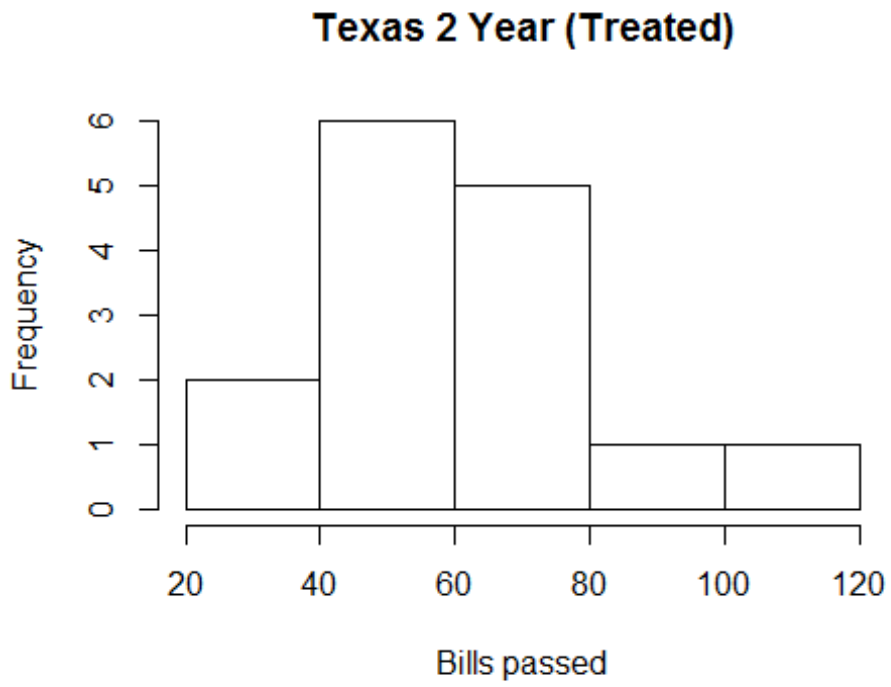
g. plot histogrms for the treatment and control in each state

I will plot the Texas histograms first and then the Arkansas histograms.

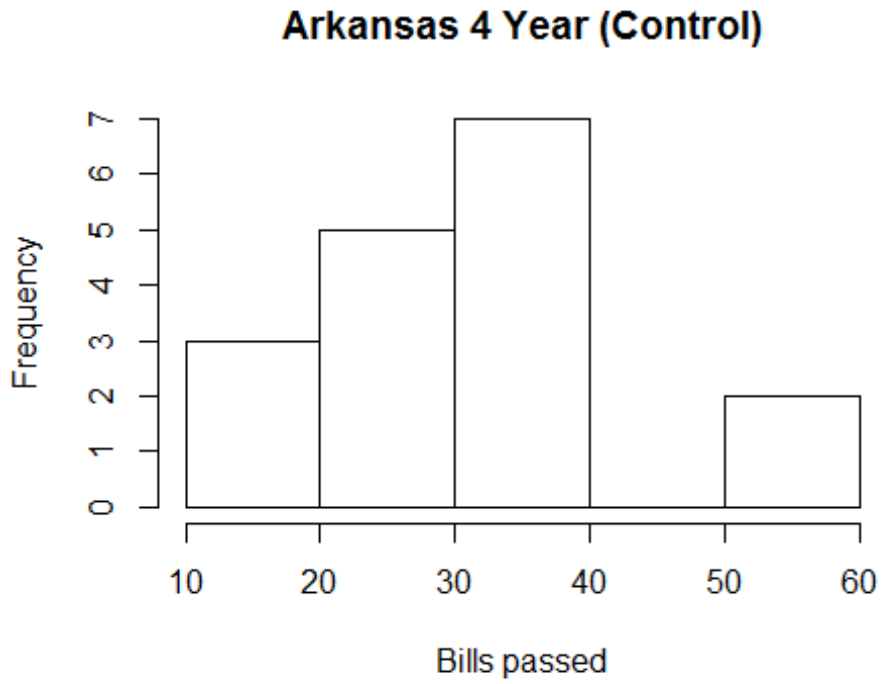
```
plot1 <- hist(lottery$bills_introduced[lottery$texas0_arkansas1 == 0 &
lottery$term2year == 0], main = "Texas 4 Year (Control)", xlab = "Bills
passed")
```



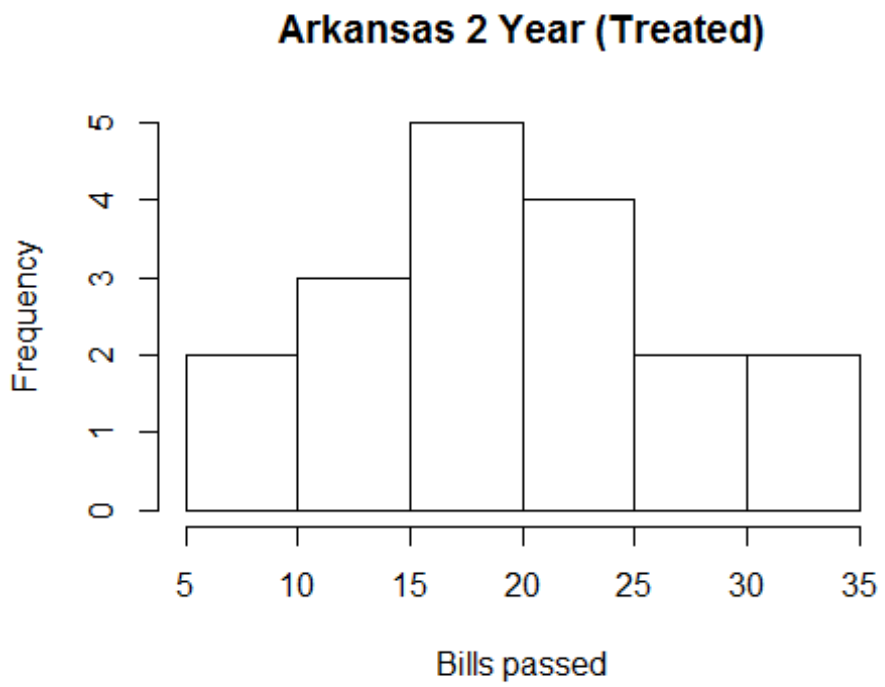

```
plot2 <- hist(lottery$bills_introduced[lottery$texas0_arkansas1 == 0 &
lottery$term2year == 1], main = "Texas 2 Year (Treated)", xlab = "Bills
passed")
```



```r
  plot3 <- hist(lottery$bills_introduced[lottery$texas0_arkansas1 == 1 &
lottery$term2year == 0], main = "Arkansas 4 Year (Control)", xlab =
"Bills passed")
```




```r
plot4 <- hist(lottery$bills_introduced[lottery$texas0_arkansas1 == 1 &
lottery$term2year == 1], main = "Arkansas 2 Year (Treated)", xlab =
"Bills passed")
```



3. FE exercise 3.11, assuming 3 clusters in treatment and 4 in control

   a. Standard error grouped by {1,2}, {3,4} . . .

```
cluster <- c(1:7)
Yi_0 <- c(.5, 3, 5, 7.5, 14.5, 16, 17.5)
Yi_1 <- c(0, 1.5, 0, 5/2, 21/2, 11.5, 11)
m = 3
N = 7
x = (m * var(Yi_0))/(N-m) + (((N-m) * var(Yi_1))/m) + 2*cov(Yi_0, Yi_1)
x = x * (1/6)
x = sqrt(x)
```

   This gives me a standard error of 4.919.

   b. Standard error of clusters formed by {1,14}, {2,13} . . .

```
Yi_02 <- c(9, 9, 9, 10, 19/2, 10, 15/2)
Yi_12 <- c(17/2, 5/2, 8, 5, 9/2, 6, 5/2)
y = (m * var(Yi_02))/(N-m) + (((N-m) * var(Yi_12))/m) + 2*cov(Yi_02,
Yi_12)
y = y * (1/6)
y = sqrt(y)
```

   Which gives me a standard error of 1.2649.

   c. Why do these lead to different standard errors and what are the implications

Gerber and Green address this issue on page 82 of their textbook "Field experiments" which says "the penalty associated with clustering depends on the variability of the cluster-level means". If we examine the separate clusters it is easy to see there is much more variability in part a than in part b. So in part a the potential outcomes have variability 46.3095 and 29.4048 for the control and treatment groups respectiviley, which are significantly higher than the potential outcomes for 0.7262 and 5.7381. The implications here are that when forming clusters it is extremely important to take the variability of the subjects into account going into those clusters as this can have a massive effect on the standard error that results.

4. iPhone ads sold at newspaper

    a. If all users could see the ads then potentially Apple would be spending $100,000 on ads. This means they would need to sell at least 100 iPhones (at $100 profit per phone) to generate a positive ROI. So Apple needs to sell at least 1001 iPhones beyond the baseline to make the investment worthwhile. If 0.5% of the 1,000,000 users would buy an iPhone anyway that means in the abscence of advertising Apple would sell 5,000 iPhones. That would mean that the probability of a user buying an iPhone must be 6001/1000000 or 0.6001%. Therefore the advertising campaign needs to shift the probability of purchase by 0.1001%.

    b. Measured effect is 0.2% (users split 50:50).

```
p = (0.007*500000 + 0.005 * 500000)/1000000
SE = p*(1-p) * ((1/500000) + (1/500000))
SE = sqrt(SE)
marER = SE * 1.96
upper = 0.002 + marER
lower = 0.002 - marER
```

Here I have used the formula from the assignment and found the standard error for this two sample proportion is 1.5445 × 10-4. So what this will indicate is a lower and upper bound for my confidence interval of 0.1697% and 0.2303%

    c. Yes, I am confident that this interval allows me to recommend this experiment. Based on my confidence interval, the lower bound is above the target of 0.0101% that would make this advertising campaign a positive return on investment.

    d. Control group if only 1% of population is in control

I will repeat the code I used above:

```
p2 = (0.007*990000 + 0.005 * 10000)/1000000
SE2 = p2*(1-p2) * ((1/990000) + (1/10000))
SE2 = sqrt(SE2)
marER2 = SE2 * 1.96
upper2 = 0.002 + marER2
lower2 = 0.002 - marER2
```

This gives me a confidence interval of 0.036% to 0.364%. This interval contains the amount we need to shift the probability of purchase by (recall 0.101%) so I would now not recommend this ad campaign.

5. Auction data downloaded from web

```
auctionData <- read.csv("C:/Users/tdunmire/Downloads/PS 2 data -
list_luckingreiley_auction_data.csv - PS 2 data -
list_luckingreiley_auction_data.csv.csv")
```

   a. Find a 95% confidence interval for the difference between treatment and control means.

```
val1 <- var(auctionData$bid[auctionData$uniform_price_auction ==
1])/length(auctionData$bid[auctionData$uniform_price_auction == 1])
val2 <- var(auctionData$bid[auctionData$uniform_price_auction ==
0])/length(auctionData$bid[auctionData$uniform_price_auction == 0])
SE <- sqrt(val1 + val2)
marER <- SE * 1.96
diff <- mean(auctionData$bid[auctionData$uniform_price_auction == 1]) -
mean(auctionData$bid[auctionData$uniform_price_auction == 0])
lower <- diff - marER
upper <- diff + marER
```

   b. What does this mean?

A confidence interval is the chance the true interval is within some range. Another way of expressing this is to say that if one repeated this experiment (under the same conditions as the original) then 95 out of 100 of these experiments would yield an ATE within the range presented above.

   c. Confidence interval from regression

```
regression = summary(lm(bid ~ uniform_price_auction, data = auctionData))
estimate <- regression$coefficients[2,1]
standard.error <- regression$coefficients[2,2]
lower2 = estimate - standard.error*1.96
upper2 = estimate + standard.error*1.96
```

This clearly shows a confidence interval of -20.686 to -3.7258.

   d. I can get the p-value this way:

```
p.value <- regression$coefficients[2,4]
```

Which shows a p-value of 0.0063

   e. p-value using randomization inference

I am going to use my previously defined functions assign() and calc_ate from problem 1.

```
dist <- replicate(10000, calc_ate(auctionData$bid,
assign(length(auctionData$bid))))
abs_dist <- abs(dist)
abs_diff = abs(diff)
two_tail_sum2 = sum(abs_dist >= abs_diff)
two_tail_p2 = mean(abs_dist >= abs_diff)
```

Using randomization inference I find that this gives me a two-tailed p-value of 0.0049.

f. Compute the same p-value using analytic formulas

I am going to use a two sample t test using R's built in functionality.

```
tTest <- t.test(auctionData$bid[auctionData$uniform_price_auction == 1],
auctionData$bid[auctionData$uniform_price_auction ==0], paired = FALSE)
p.value2 <- tTest$p.value
```

Here I can see a p-value of 0.0064 which is slightly different from 0.0049 (depends on randomization).

g. The two p-values in part e and part f are slightly different (how much different will depend on the particular run of randomization). This will depend a lot of the particular randomization, although on average this will build a sampling distribution it may not be perfect. As the sample size increases the standard errors should decrease which means the p-value from the randomization will become more precise and more closely resemble the p-value from part f.