Recitation 4 Randomization and Selection Bias

Seamus Wagner

September 17, 2021

#Selection Bias For the first section of recitation, I will go over selection bias and lab 2 question 1 as there was a bit of confusion between what you were supposed to take from it versus what most of you did takeaway. Note, I did alter the grades on 1.7 since the question states to mention your expectations and why, not whether they were grounded in what we hoped or not.

The main culprit for this question was 1.7, where you were asked to interpret what happened in 1.6 as you increased the number of observations and observed the bias. Nearly everyone mentioned the law of large numbers as why the bias was going to 0. What we need to remember though, is that the ATE was 0 for this case. If the ATE were positive or negative, the law of large numbers should not trend toward 0 if the sample is biased, as the underlying distribution is no longer normal in this sense. If we observe selection bias, we should see steady estimates as the number of observations increases.

Below is the example I put on many of your responses for how to check this (I forgot to add the -.1 and -(-.1) for your examples, so they probably didn't work). The first few simulations have too few observations to trend toward what we would expect the bias to be. It is important to remember that the error is the estimate - ATE. The ATE for the question at the time was 0. So the error = estimate.

```
## [1] 1795694184
```

```
## [1] -0.04304698
```

```
## [1] -0.04118014
```

```
## [1] -0.02504797
```

```
## [1] -0.02790532
```

```
## [1] -0.02650942
```

See here that once we get to a larger number of observations, but the bias seems pretty consistent here, and we are getting underestimates. Also note that we subtract out the known ATE at the end of our difference in means. So the NDIM minus the known difference in means produces underestimates when selection bias is included. Now let's try this with a negative ATE.

```
## [1] -0.003145136
```

```
## [1] 0.04133125
```

```
## [1] 0.02287491
```

```
## [1] 0.02276578
```

```
## [1] 0.02660381
```

Here we see pretty consistent bias and overestimates this time. Note that we also accounted for the negative ATE at the end of our NDIM code. These estimates should be 0 if no bias were present given that we account for the ATE not being zero in our NDIM calculation.

What this leaves us with is the knowledge that with enough number of observations, we can see that bias is influencing our estimations, and that where the weka law of large numbers comes in, it is that a decent number of observations is needed to see that the estimates do not trend toward zero when bias is present. We observe bias because the assignment to treatment is no longer independent of the values of Y1 and Y0. With higher Y1 values will be disproportionately assigned to treatment compared to lower Y1 values. Further, higher Y0 values disproportionately assigned to control compared to lower Y0 values.

#Randomization

Moving into the lab specific stuff for lab three. This should be a shorter lab and much of the code comes from the demos and slides. We will start with complete versus simple randomization then move onto p values.

Simple randomization takes forever to run as the number of observations get remotely interesting. We start with resetting a seed (not necessary, I just do it sometimes). Then we create a matrix of complete randomization for all combinations of 0 and 1 for eight participants. We then create a subset matrix of only those cases where exactly half of the observations are treated. One way to check that it worked properly is to check the dimensions of the matrix next to the R command for combinations.

```
## [1] 70  8
```

```
## [1]  4 70
```

```
##    user  system elapsed
##       0       0       0
```

```
##    user  system elapsed
##    2.89    5.77    8.34
```

```
##           used (Mb) gc trigger   (Mb)  max used   (Mb)
## Ncells  787931 42.1    1504927   80.4   1154850   61.7
## Vcells 1436797 11.0  467589065 3567.5 479590969 3659.0
```

As we mentioned in class, these get excessively large quickly and adding another eight participants exceeds the maximum integer size for number of rows that my laptop can take.

The options we tend to take is sampling from all possible randomizations by either taking exactly half or using some probability selection. If we go with simple randomization (exactly half), the sample function is best for matricies.

```
## [1] 256   8
```

```
## [1] 128   8
```

```
## [1] 128   8
```

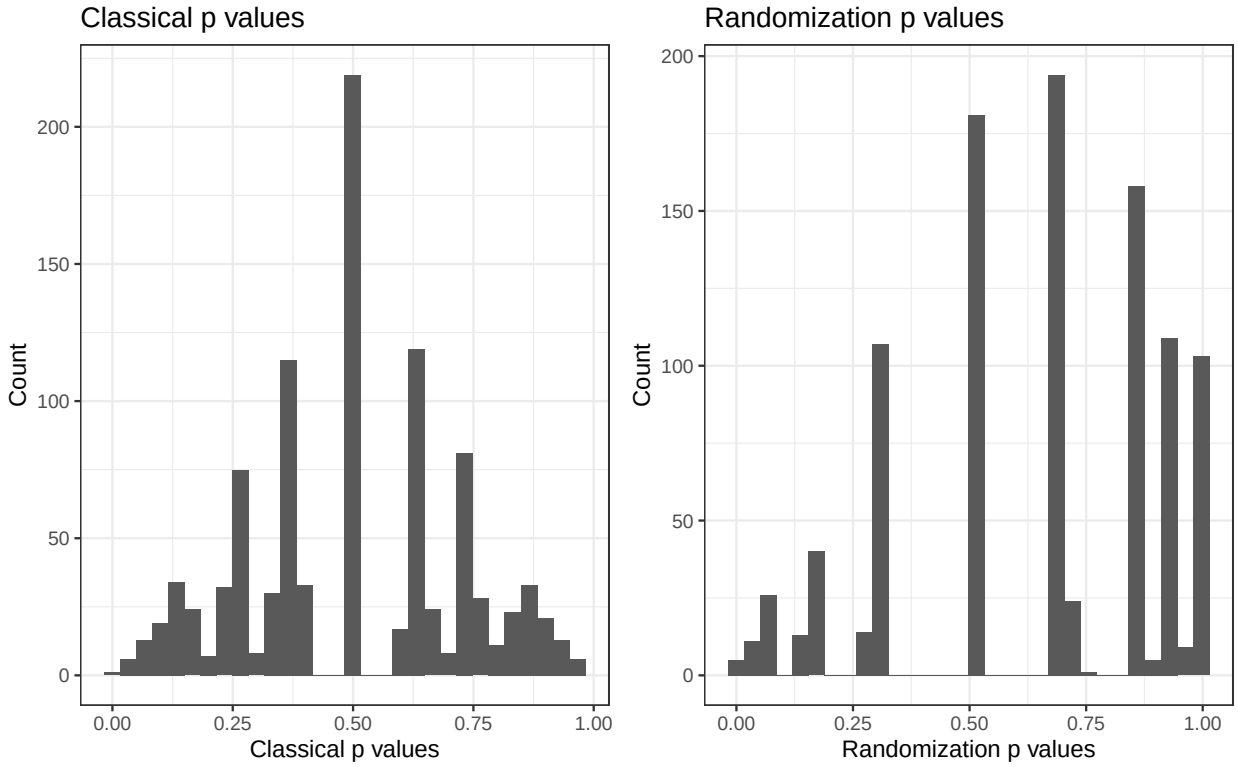
For the p-values, t-tests, at standard errors and such, take a close look at the code from William's demo. This is one test of the function writing we have been practicing, as you will start nesting functions you wrote within others and gain a grasp on what each function does. I am happy to go over these with you all. There are also canned functions for most of these, but test them against your written ones to illustrate what William went over on Friday to see how they may differ and why canned functions are something to make sure you fully understand before using them and then defending their use in your own research.

I am happy to go over any code people have for this question or do some live coding with suggestions from the group.

I am going to bring in some data using some of the functions I am not sharing as recitation but to go over what you sohuld be getting and how to interpret them.

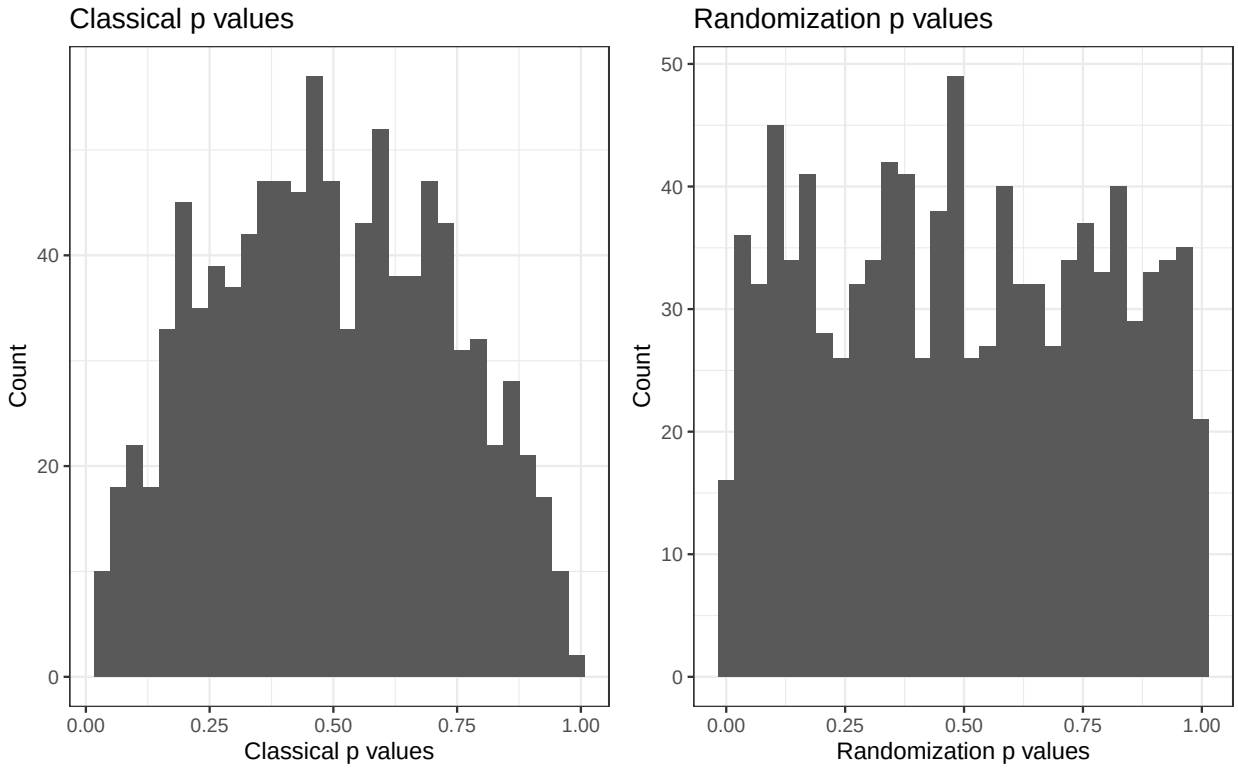
```
## [1] 0.007
```

```
## [1] 0.016
```




```
## [1] 0.011
```

```
## [1] 0.05
```


Classical p values
Count
0
20
40
0.00
0.25
0.50
0.75
1.00
Classical p values
Randomization p values
Count
0
10
20
30
40
50
0.00
0.25
0.50
0.75
1.00
Randomization p values