Replication 2

Megan Ellertson

4/11/2021

1.

Data Prep

The following provides the code for the data formatting which is necessary prior to the analysis. Covariates which were to be included in both the Logit and OLS models were included in the first portion (plugged into the "control" dataset). The covariates specific to the Logit (cubed) and OLS (quadratic) were then separated into two different dataframe's to keep the analysis separate.

```
control <- combo %>%
  mutate(agesq = age^2,
         agecube = age^3,
         educsq = educ*educ,
         u74 = case_when(re74 == 0 ~ 1, TRUE ~ 0),
         u75 = case_when(re75 == 0 ~ 1, TRUE ~ 0),
         interaction1 = educ*re74,
         re74sq = re74^2,
         re75sq = re75^2,
         interaction2 = u74*hisp)
#Logit related dataframe
nsw_dw_cpscontrol_logit <- control %>%
  mutate(educcube = educ^3,
         re74cube=re74^3,
         re75cube=re75^3)
#OLS related dataframe
nsw_dw_cpscontrol_ols <- control
```

a. & b.

The following equations are for the logit and OLS versions of the basic equation provided in the mixtape nsw_pscore.do. This expands off the basic logit provided.

The following model is the Logit model which includes through the cubed polynomial of each non-binary variable from the basic equation provided.

$$logit(treat) = age+age^2+age^3+educ+educ^2+educ^3+marr+nodegree+black+hisp+re74+re74^2+re74^3+re75+re75^2+re75^3$$

Table 1: Logit Model

Predictor	B	SE	t	p
Intercept	-34.65	4.095	-8.46	<0.001
age	2.32	0.353	6.57	<0.001
agesq	-0.06	0.011	-5.73	<0.001
agecube	0.00	0.000	4.84	<0.001
educ	0.72	0.664	1.09	0.278
educsq	-0.03	0.074	-0.43	0.669
educcube	0.00	0.003	-0.30	0.761
marr	-1.54	0.255	-6.04	<0.001
nodegree	0.92	0.345	2.68	0.007
black	3.84	0.266	14.41	<0.001
hisp	1.69	0.408	4.14	<0.001
re74	0.00	0.000	1.74	0.081
re74sq	0.00	0.000	-2.15	0.031
re74cube	0.00	0.000	2.62	0.009
re75	0.00	0.000	1.04	0.300
re75sq	0.00	0.000	-2.33	0.020
re75cube	0.00	0.000	2.45	0.014
u74	2.15	0.415	5.17	<0.001
u75	0.46	0.352	1.30	0.193

```r
logit <- glm(treat ~ age + agesq + agecube + educ + educsq
                     + educcube + marr + nodegree
                     + black + hisp + re74 + re74sq +re74cube
                     + re75 + re75sq + re75cube + u74 + u75,
                      family = binomial(link =
                      "logit"), data =
               nsw_dw_cpscontrol_logit)
```

Running this Logit regression provides the following output table. Each of the included variables except for the education related variables are statistically significant.

The following model is the OLS regression and version which will be included along with the Logit model. This includes the covariates from the basic model up to the squared polynomial, and not including the binary covariates with a polynomial version. Note that the cube polynomial of the age covariate is kept in the analysis because it was included in the basic model.

$$ols(treat) = age+age^2+age^3+educ+educ^2+marr+nodegree+black+hisp+re74+re74^2+re75+re75^2+u74+u75+\epsilon$$

```r
ols <- lm(treat ~ age + agesq + agecube + educ + + educsq +
            marr + nodegree+ black + hisp + re74 + re74sq
          + re75 + re75sq + u74 + u75,
          data = nsw_dw_cpscontrol_ols)
```

```r
ols %>%
  tidy() %>%
  mutate(
    p.value = scales::pvalue(p.value),
```

Table 2: OLS Model

Predictor	B	SE	t	p
Intercept	-0.39	0.031	-12.34	<0.001
age	0.03	0.003	11.59	<0.001
agesq	0.00	0.000	-11.11	<0.001
agecube	0.00	0.000	10.44	<0.001
educ	0.01	0.001	5.90	<0.001
educsq	0.00	0.000	-6.39	<0.001
marr	-0.02	0.002	-10.53	<0.001
nodegree	0.02	0.003	8.15	<0.001
black	0.10	0.003	35.97	<0.001
hisp	0.01	0.003	2.37	0.018
re74	0.00	0.000	-0.51	0.609
re74sq	0.00	0.000	1.37	0.170
re75	0.00	0.000	-3.24	0.001
re75sq	0.00	0.000	3.02	0.003
u74	0.04	0.004	11.36	<0.001
u75	0.01	0.004	3.70	<0.001

```
    term = c("Intercept", "age", "agesq", "agecube", "educ",
             "educsq", "marr", "nodegree", "black", "hisp",
             "re74","re74sq", "re75", "re75sq",  "u74",
             "u75")
  ) %>%
  kable(
    caption= "OLS Model",
    col.names = c("Predictor", "B", "SE", "t", "p"),
    digits = c(0, 2, 3, 2, 3)
  )
```

In the OLS version of the model, more of the included variables are statistically significant compared to the Logit model. Only the “re74” covariate is not statistically significant.

c.

To map out the p-scores for the Logit and OLS models, the fitted values of the regressions are taken respectively. How this was done with the syntax is provided below:

```
nsw_dw_cpscontrol_logit <- nsw_dw_cpscontrol_logit %>%
  mutate(pscore = logit$fitted.values)


nsw_dw_cpscontrol_ols <- nsw_dw_cpscontrol_ols %>%
  mutate(pscore = ols$fitted.values)
```

The means of these p-scores for both the Logit and the OLS models are determined for the treated and control groups. The mean of these p-scores amung the two groups for each model are provided in the table below. The Logit treated is at about 0.4 while the control group is far lower. The OLS is also skewed far lower than the more centered Logit output.

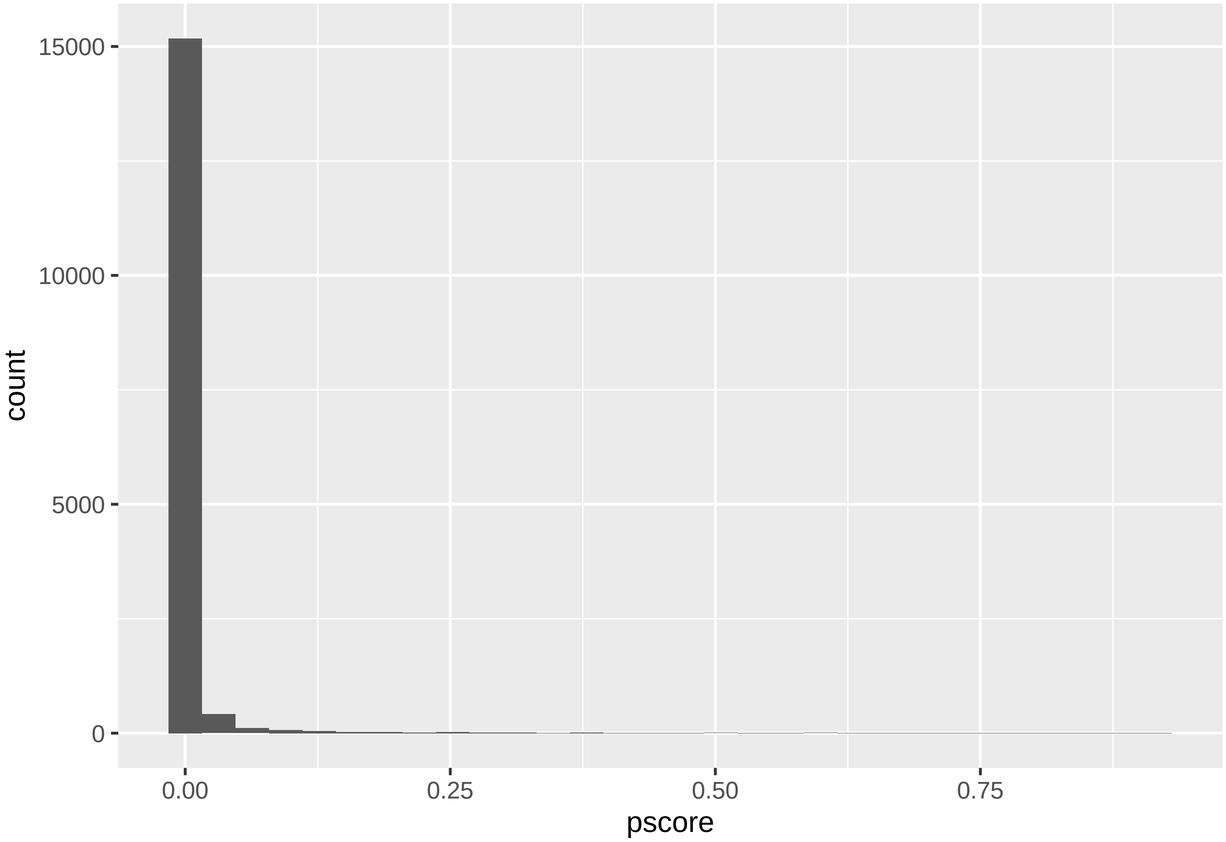
```
pscore_control_logit <- nsw_dw_cpscontrol_logit %>%
  filter(treat==0) %>%
  pull(pscore) %>%
  mean()
pscore_treat_logit <- nsw_dw_cpscontrol_logit %>%
  filter(treat==1) %>%
  pull(pscore) %>%
  mean()

pscore_control_ols <- nsw_dw_cpscontrol_ols %>%
  filter(treat==0) %>%
  pull(pscore) %>%
  mean()
pscore_treat_ols <- nsw_dw_cpscontrol_ols %>%
  filter(treat==1) %>%
  pull(pscore) %>%
  mean()
```

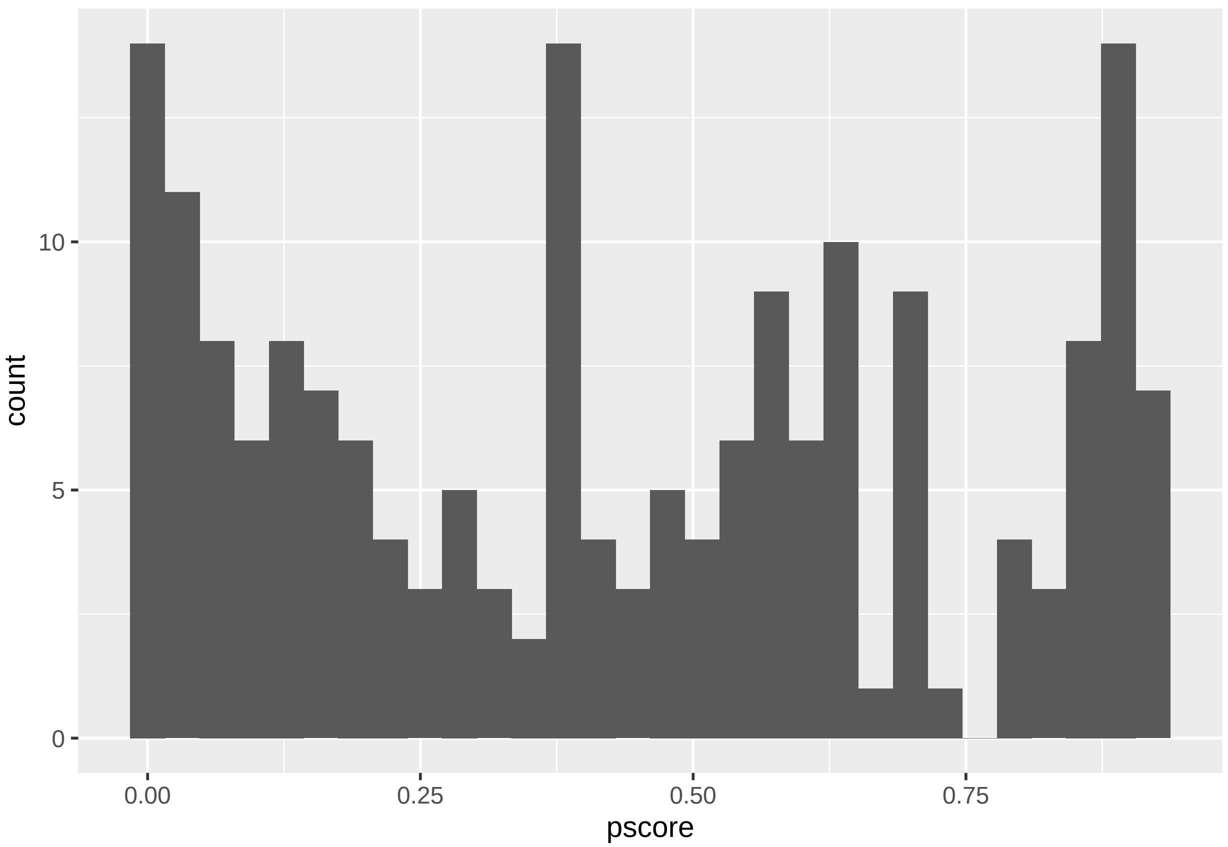
```
##          Groups       Means
## 1 Logit Control 0.006582229
## 2   Logit Treat 0.431010791
## 3   OLS Control 0.009867421
## 4     OLS Treat 0.147028133
```

To see further into what the p-score distribution looks like for the treated and control groups of each model the following histograms provide a more easily understood visual representation.

```
nsw_dw_cpscontrol_logit %>%
  filter(treat == 0) %>%
  ggplot() +
  geom_histogram(aes(x = pscore))
```



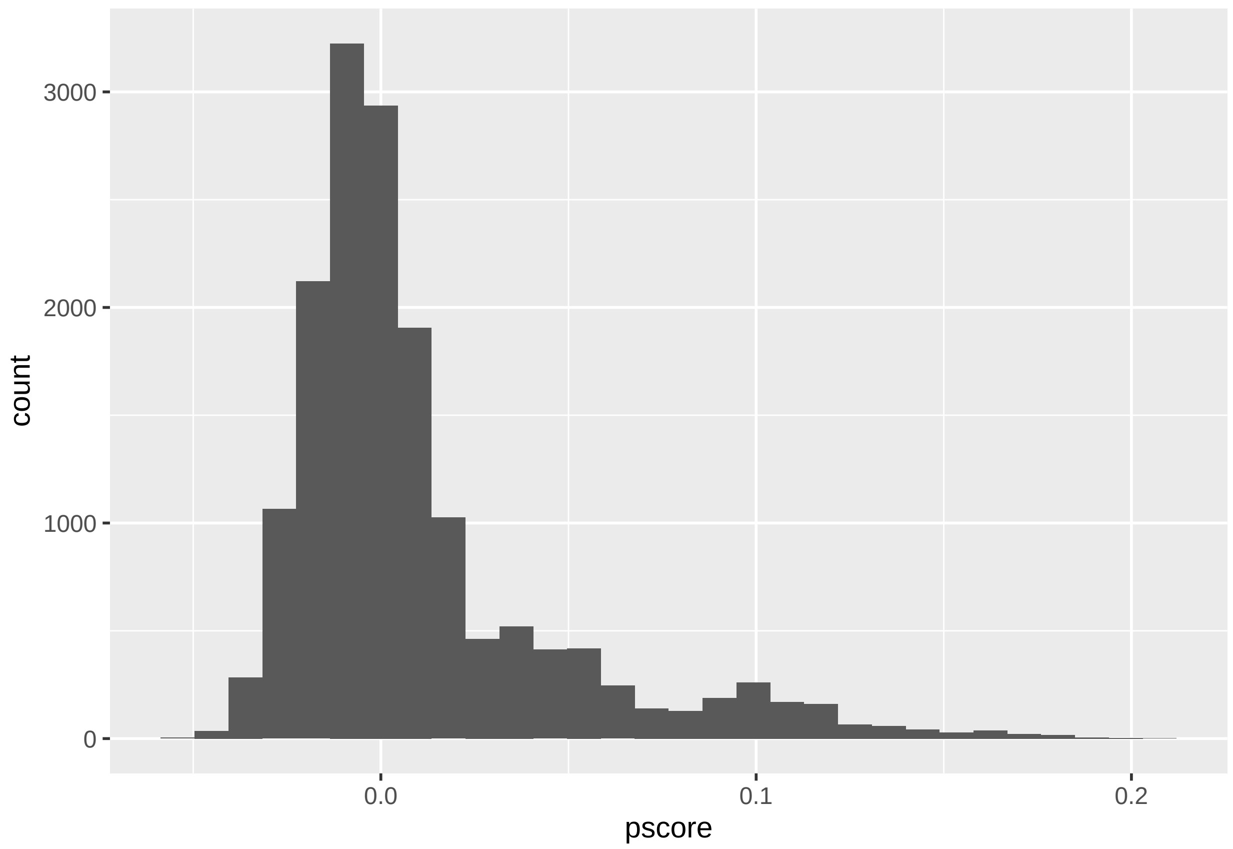
```
nsw_dw_cpscontrol_logit %>%
  filter(treat == 1) %>%
  ggplot() +
  geom_histogram(aes(x = pscore))
```



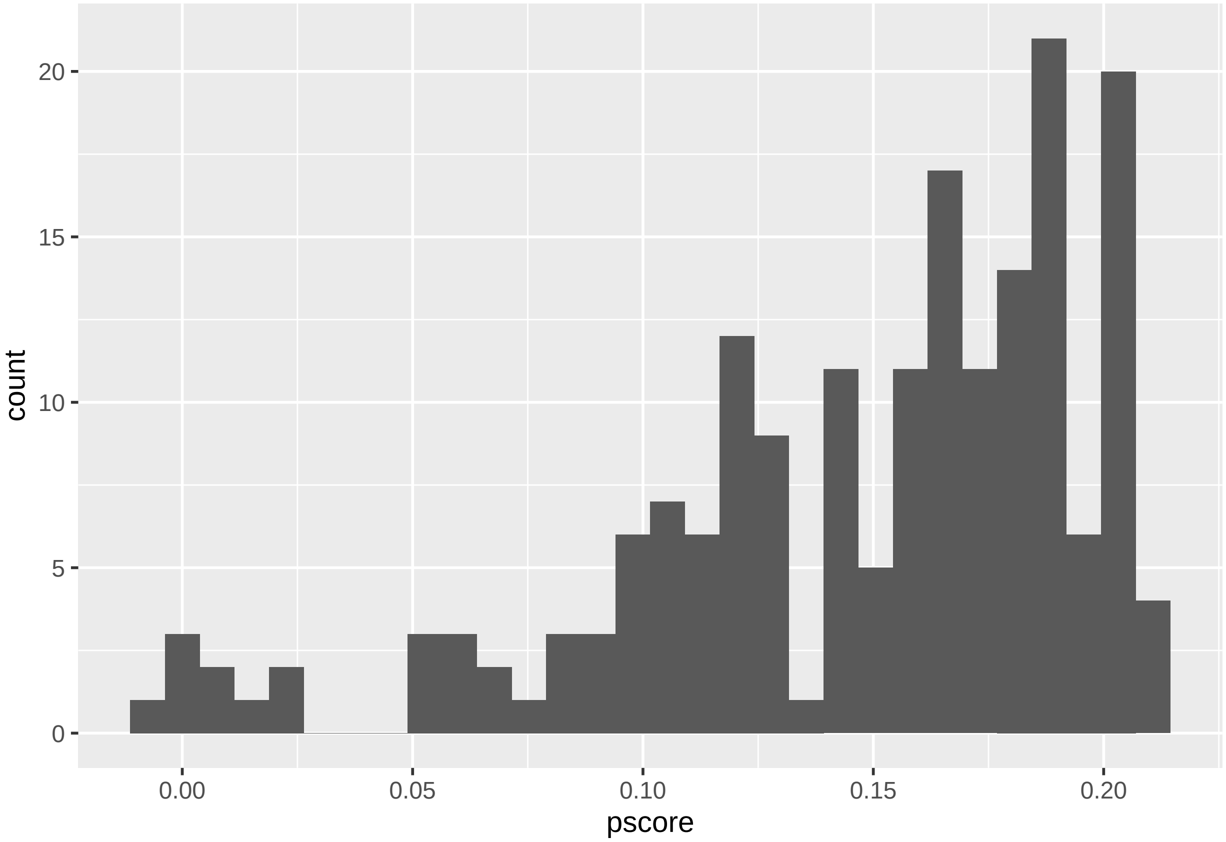
The two histograms above represent first the control p-score distribution and the treated distribution. The control group for the Logit model is highly skewed to the left. There is a vast amount of bunching around 0. Where the minimum value appears to go below 0 but really sit right at 0. The maximum value is around the 0.25. The treatment p-score distribution is far more evenly distributed across the frame. The histogram ranges from just below 0 to 1 and is distributed across the range in a non-uniform manner.

The following provides the histograms for the treated and control OLS model p-scores.

```
nsw_dw_cpscontrol_ols %>%
  filter(treat == 0) %>%
  ggplot() +
  geom_histogram(aes(x = pscore))
```



```
nsw_dw_cpscontrol_ols %>%
  filter(treat == 1) %>%
  ggplot() +
  geom_histogram(aes(x = pscore))
```



The first histogram represents the control distribution. This is centered around 0 and has a loosely normal shaped distribution. There are far more values below 0 in this histogram an the prior histograms. The minimum appears to be around -0.05 and the maximum is just below 0.2. The second histogram is the treatment group distribution. This distribution appears to be skewed to the right side. The maximum values appear to be just above 0.2 (around 0.21) while the minimum is just below 0. The distribution is more weighted toward the higher end of the distribution with a peak around 0.18.

d.

The p-scores which are less than 0.1 and greater than 0.9 can then be dropped and the histograms recalibrated with the trimmed p-scores. The following will go through the same process as in part c. Additionally, the total number of observations in this trimmed dataset for both Logit and OLS models will be stored for later to use in the weighting for the ATT. Once the trimmed datasets are calculated they will be used for the rest of the problems. The first chuck of code trims the data and stores the number of observations (n) for each dataset.

```
nsw_dw_cpscontrol_logit <- nsw_dw_cpscontrol_logit %>%
  filter(!(pscore >= 0.9)) %>%
  filter(!(pscore <= 0.1))

Nl <- nrow(nsw_dw_cpscontrol_logit)

nsw_dw_cpscontrol_ols <- nsw_dw_cpscontrol_ols %>%
  filter(!(pscore >= 0.9)) %>%
  filter(!(pscore <= 0.1))
```

```
No <- nrow(nsw_dw_cpscontrol_ols)
```

The following chunk provides the code for recalculating the p-score means for the control and treatment group for OLS and Logit models.

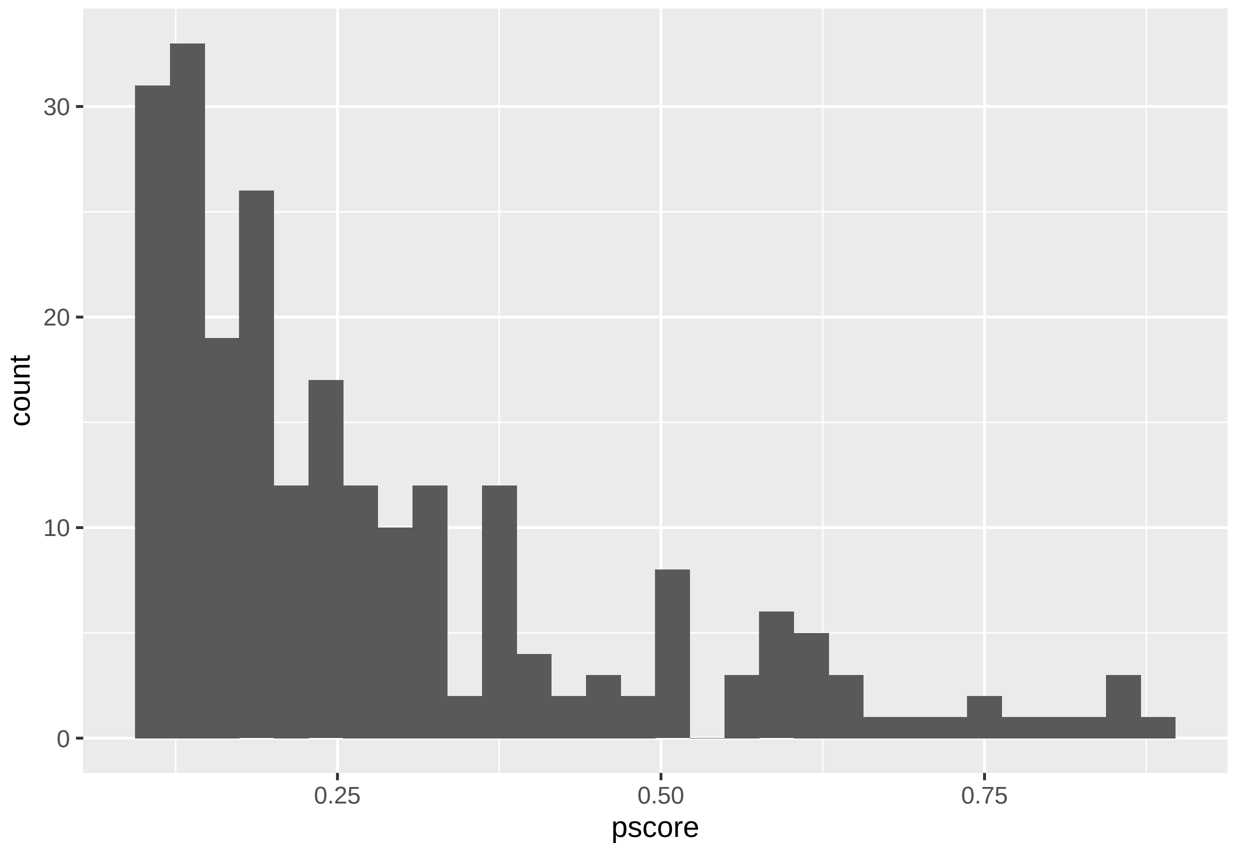
```
pscore_control_logit2 <- nsw_dw_cpscontrol_logit %>%
  filter(treat==0) %>%
  pull(pscore) %>%
  mean()
pscore_treated_logit2 <- nsw_dw_cpscontrol_logit %>%
  filter(treat==1) %>%
  pull(pscore) %>%
  mean()

pscore_control_ols2 <- nsw_dw_cpscontrol_ols %>%
  filter(treat==0) %>%
  pull(pscore) %>%
  mean()
pscore_treated_ols2 <- nsw_dw_cpscontrol_ols %>%
  filter(treat==1) %>%
  pull(pscore) %>%
  mean()
```

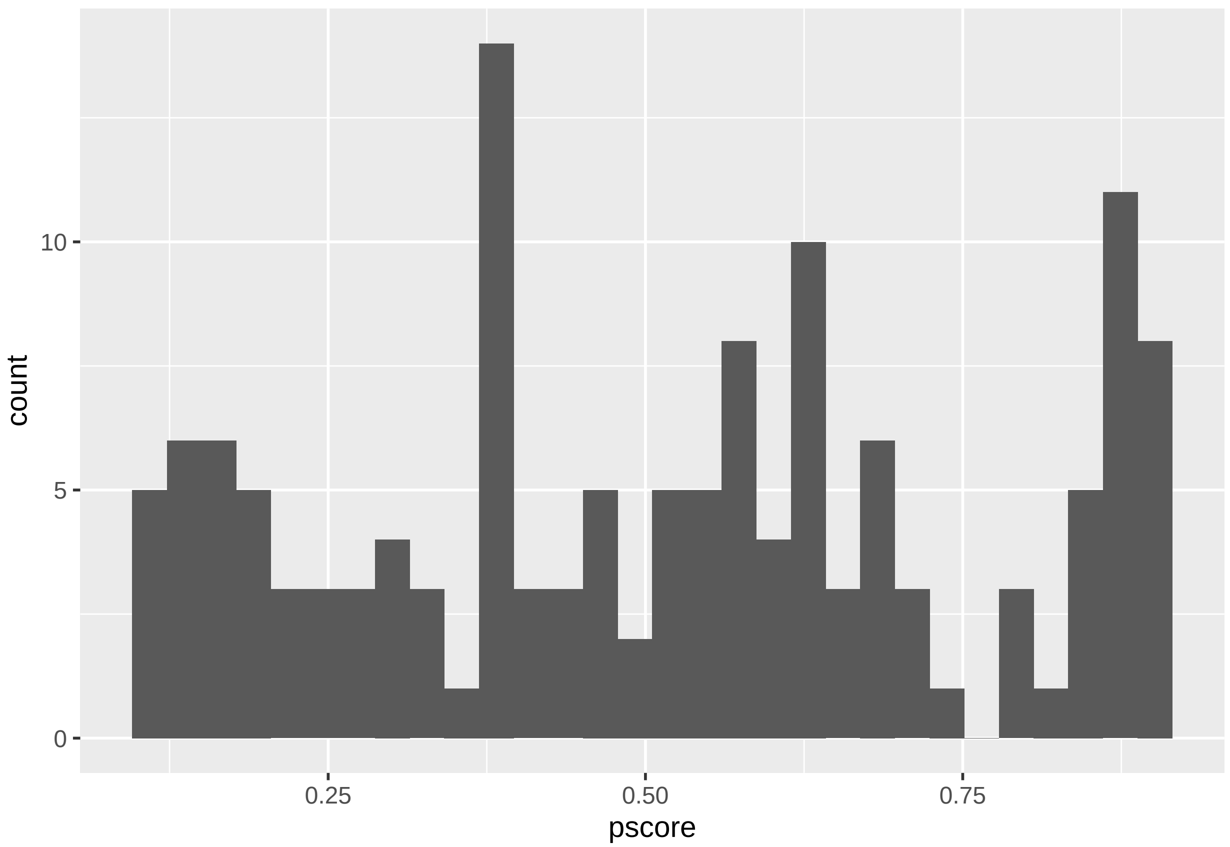
The means are then provided in the following table (as previously provided). All of the means are a bit larger than the prior means form the first table. This change makes sense given the data now will not conclude the lower values which was weighting down the distribution in majority of the groups. Only the treatment group of the Logit model had a distribution which was near the higher end of the trimming (0.9), considering that and the many negative or close to 0 p-score's are gone, then the increase in these values is justified.

```
##          Groups     Means
## 1 Logit Control 0.2825155
## 2   Logit Treat 0.5112145
## 3   OLS Control 0.1249108
## 4     OLS Treat 0.1635168
```

```
nsw_dw_cpscontrol_logit %>%
  filter(treat == 0) %>%
  ggplot() +
  geom_histogram(aes(x = pscore))
```

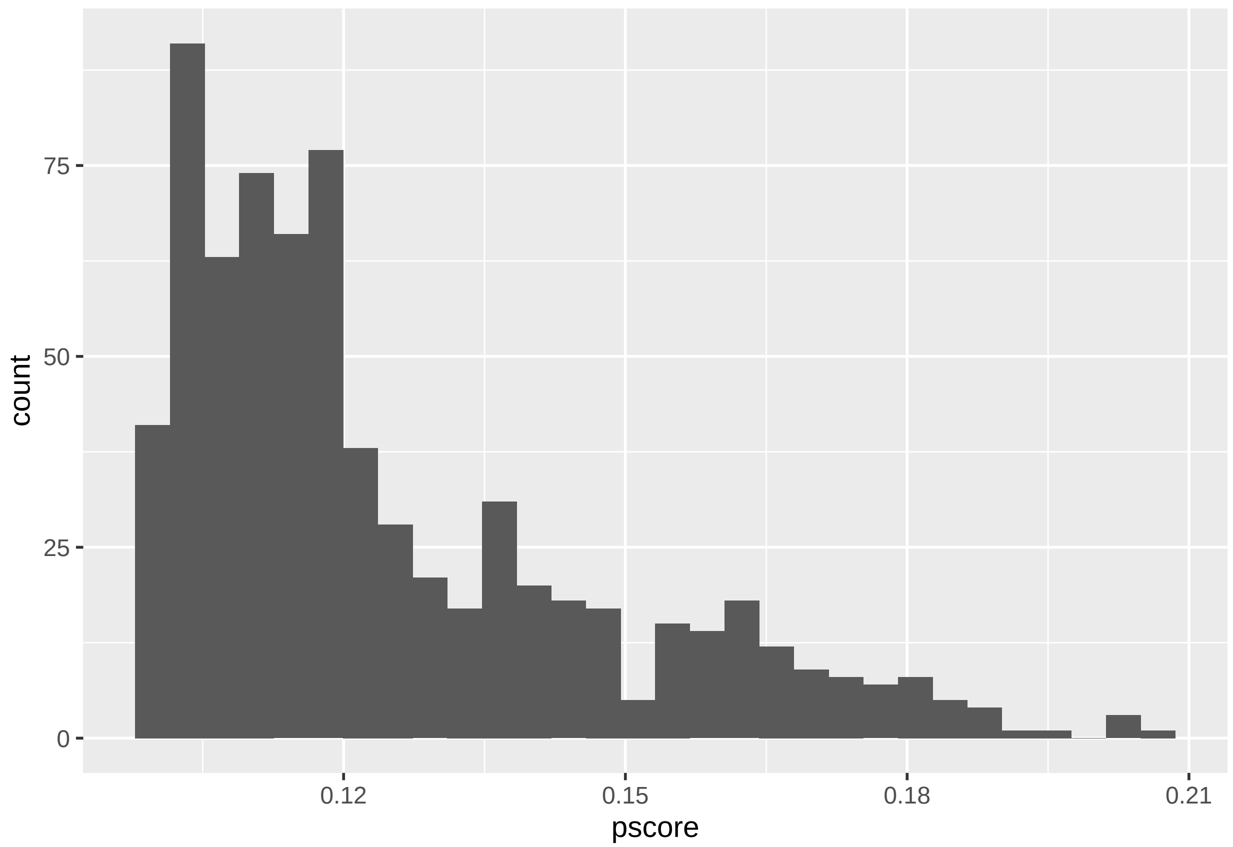


```
nsw_dw_cpscontrol_logit %>%
  filter(treat == 1) %>%
  ggplot() +
  geom_histogram(aes(x = pscore))
```

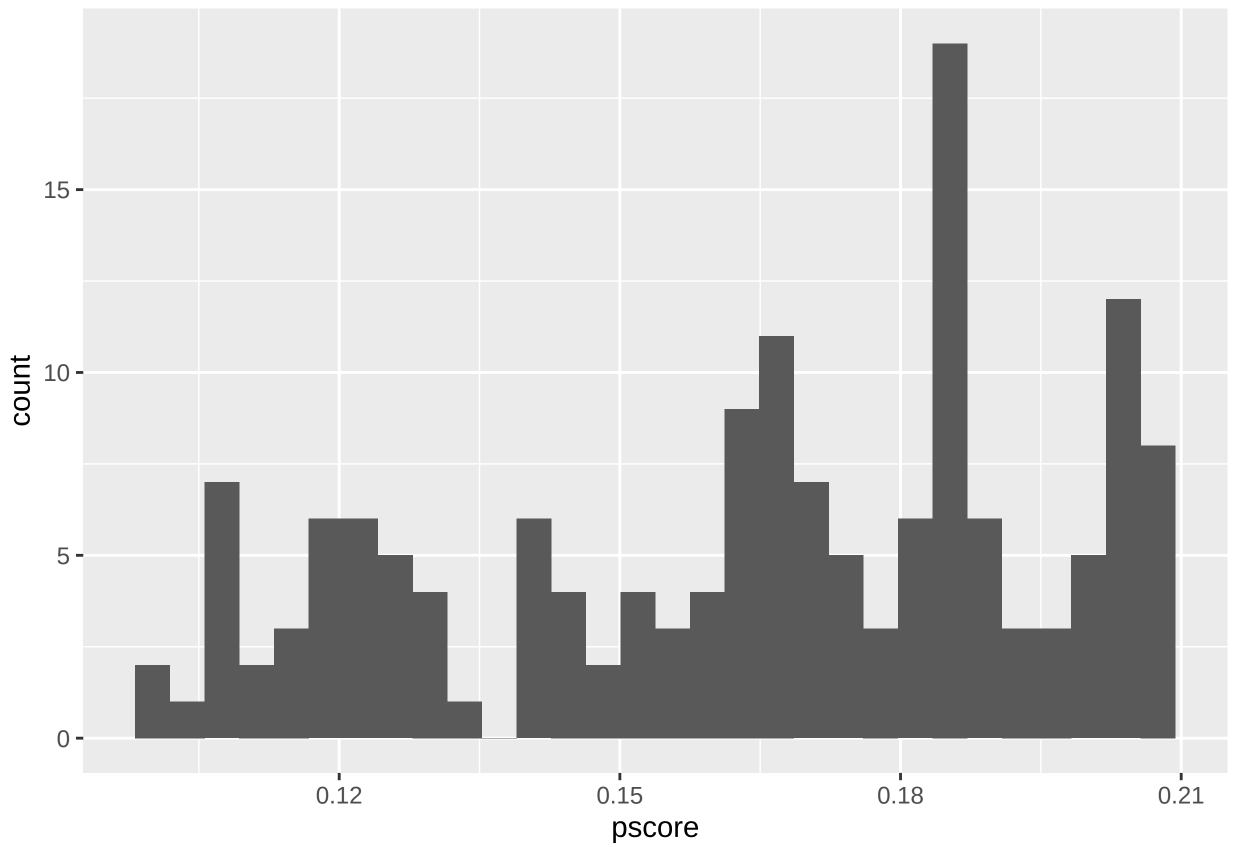


The histograms above represent the trimmed version of the p-score distributions which were generated by the Logit model. This changes the distribution of the histograms from the untrimmed version of the data. Now the minimum values for both the control and treatment groups are 0.1 and maximums of 0.9. For the treatment group the distribution looks very similar. Just values at the very ends of the fairly even distribution have been cut therefore it would make sense that the histogram looks similar. However, the control group distribution is significantly different than the untrimmed version. This is due to the fact that the previous was surrounding right where the cutoff of the trimming exists. Therefore a lot of items were dropped. The distribution is skewed in the same direction but less extreme.

```
nsw_dw_cpscontrol_ols %>%
  filter(treat == 0) %>%
  ggplot() +
  geom_histogram(aes(x = pscore))
```



```
nsw_dw_cpscontrol_ols %>%
  filter(treat == 1) %>%
  ggplot() +
  geom_histogram(aes(x = pscore))
```



The histograms above represent the trimmed OLS control and treatment group p-score distributions. The control group now has a minimum of 0.1 and a maximum of just below 0.21. This distribution is skewed at the lower end. This makes sense with the process of the trimming. Much of the data was at or below the 0.1 cutoff. Therefore, much of the values in the control group were dropped with the trimming. The shape of the distribution remains the same. The treatment group is now has a minimum value of 0.1 and a maximum value of just below 0.21 similar to the control group. The untrimmed version was skewed to the higher end, while this distribution is a little more spread across the range.

2.

The rest of the analysis will be conducted with the trimmed data as specified above which is important for ensuring the covariate balance and the sample. The first differencing compares the outcomes of the trimmed data provides the ATE or in the SDO output. This is the first differencing for both the Logit model and the OLS model.

The Logit first differnce:

```
mean1log <- nsw_dw_cpscontrol_logit %>%
  filter(treat == 1) %>%
  pull(re78) %>%
  mean()

nsw_dw_cpscontrol_logit$y1 <- mean1log
```

```r
mean0log <- nsw_dw_cpscontrol_logit %>%
  filter(treat == 0) %>%
  pull(re78) %>%
  mean()

nsw_dw_cpscontrol_logit$y0 <- mean0log

atelog <- unique(nsw_dw_cpscontrol_logit$y1 - nsw_dw_cpscontrol_logit$y0)
```

```
## [1] 1724.773
```

The SDO of the logit model provides the same result as the ATE calculation as provided below, just around 1700 difference between the treated and the control.

```r
sdologit <- nsw_dw_cpscontrol_logit %>%
  mutate(d = case_when(treat == 1 ~ 1, TRUE ~ 0))

ey1logit <- sdologit %>%
  filter(d == 1) %>%
  pull(re78) %>%
  mean()

ey0logit <- sdologit %>%
  filter(d == 0) %>%
  pull(re78) %>%
  mean()

sdologit <- ey1logit - ey0logit
sdologit
```

```
## [1] 1724.773
```

The OLS ATE calculation is below:

```r
mean1ols <- nsw_dw_cpscontrol_ols %>%
  filter(treat == 1) %>%
  pull(re78) %>%
  mean()

nsw_dw_cpscontrol_ols$y1 <- mean1ols


mean0ols <- nsw_dw_cpscontrol_ols %>%
  filter(treat == 0) %>%
  pull(re78) %>%
  mean()

nsw_dw_cpscontrol_ols$y0 <- mean0ols

ateols <- unique(nsw_dw_cpscontrol_ols$y1 - nsw_dw_cpscontrol_ols$y0)
```

```
## [1] -4128.57
```

The SDO of the OLS model also produces the same result of a larger negative difference between the treated and the control group.

```
sdools <- nsw_dw_cpscontrol_ols %>%
  mutate(d = case_when(treat == 1 ~ 1, TRUE ~ 0))

ey1ols <- sdools %>%
  filter(d == 1) %>%
  pull(re78) %>%
  mean()

ey0ols <- sdools %>%
  filter(d == 0) %>%
  pull(re78) %>%
  mean()

sdools <- ey1ols - ey0ols
sdools
```

```
## [1] -4128.57
```

3.

The following is the code syntax for the Logit model weighted with normalized and non normalized ATT calculations. The equation provided for the Abadie (2005) estimations is the non-normalized weights. This also uses the trimmed data.

```
nsw_dw_cpscontrol_logit <- nsw_dw_cpscontrol_logit %>%
  mutate(d1 = treat/pscore,
         d0 = (1-treat)/(1-pscore))

s1 <- sum(nsw_dw_cpscontrol_logit$d1)
s0 <- sum(nsw_dw_cpscontrol_logit$d0)
# Non-Normalized Weights
nsw_dw_cpscontrol_logit <- nsw_dw_cpscontrol_logit %>%
  mutate(y1 = treat * re78/pscore,
         y0 = (1-treat) * re78/(1-pscore),
         ht = y1 - y0)
#Normalized Weights
nsw_dw_cpscontrol_logit <- nsw_dw_cpscontrol_logit %>%
  mutate(y1 = (treat*re78/pscore)/(s1/N1),
         y0 = ((1-treat)*re78/(1-pscore))/(s0/N1),
         norm = y1 - y0)

nsw_dw_cpscontrol_logit %>%
  pull(ht) %>%
  mean()
```

```
## [1] 1744.356
```

```r
nsw_dw_cpscontrol_logit %>%
  pull(norm) %>%
  mean()
```

```
## [1] 1635.115
```

The first value is the non-normalized weight ATT (with trimmed data), which is the version of interest. But the normalized weights are also provided.

The following is the OLS model of the normalized and non-normalized weighted ATT calculation. Abadie (2005) is the non-normalized version.

```r
nsw_dw_cpscontrol_ols <- nsw_dw_cpscontrol_ols %>%
  mutate(d1 = treat/pscore,
         d0 = (1-treat)/(1-pscore))

s1 <- sum(nsw_dw_cpscontrol_ols$d1)
s0 <- sum(nsw_dw_cpscontrol_ols$d0)

#Non-normalized weights
nsw_dw_cpscontrol_ols <- nsw_dw_cpscontrol_ols %>%
  mutate(y1 = treat * re78/pscore,
         y0 = (1-treat) * re78/(1-pscore),
         ht = y1 - y0)

#Normalized weights
nsw_dw_cpscontrol_ols <- nsw_dw_cpscontrol_ols %>%
  mutate(y1 = (treat*re78/pscore)/(s1/No),
         y0 = ((1-treat)*re78/(1-pscore))/(s0/No),
         norm = y1 - y0)

nsw_dw_cpscontrol_ols %>%
  pull(ht) %>%
  mean()
```

```
## [1] -2322.159
```

```r
nsw_dw_cpscontrol_ols %>%
  pull(norm) %>%
  mean()
```

```
## [1] -3938.138
```

The first is the OLS ATT (trimmed data) with non-normalized weights, as specified by Abadie (2005) and the second is the ATT (trimmed data) with the normalized weights. The outcome of interest is the first from the non-normalized weights.

The estimates determined by this analysis are slightly different than the ones generated by Dr. Cunningham in the Mixtape due to the fact that this model is a little more complicated in that it includes additional covariates. However the estimates are close and match the direction (i.e. the logit model is positive). The OLS model provides negative estimates. This indicates that the Logit model is more effective in the analysis of this situation.

The version control and files relevant to this replication can be found in my github repository: https://github.com/meganellertson/Replication-2-.git