Project 6: Randomization and Matching

Introduction

In this project, you will explore the question of whether college education causally affects political participation. Specifically, you will use replication data from Who Matches? Propensity Scores and Bias in the Causal Effects of Education on Participation by former Berkeley PhD students John Henderson and Sara Chatfield. Their paper is itself a replication study of Reconsidering the Effects of Education on Political Participation by Cindy Kam and Carl Palmer. In their original 2008 study, Kam and Palmer argue that college education has no effect on later political participation, and use the propensity score matching to show that pre-college political activity drives selection into college and later political participation. Henderson and Chatfield in their 2011 paper argue that the use of the propensity score matching in this context is inappropriate because of the bias that arises from small changes in the choice of variables used to model the propensity score. They use genetic matching (at that point a new method), which uses an approach similar to optimal matching to optimize Mahalanobis distance weights. Even with genetic matching, they find that balance remains elusive however, thus leaving open the question of whether education causes political participation.

You will use these data and debates to investigate the benefits and pitfalls associated with matching methods. Replication code for these papers is available online, but as you'll see, a lot has changed in the last decade or so of data science! Throughout the assignment, use tools we introduced in lab from the tidyverse and the MatchIt packages. Specifically, try to use dplyr, tidyr, purrr, stringr, and ggplot instead of base R functions. While there are other matching software libraries available, MatchIt tends to be the most up to date and allows for consistent syntax.

Data

The data is drawn from the Youth-Parent Socialization Panel Study which asked students and parents a variety of questions about their political participation. This survey was conducted in several waves. The first wave was in 1965 and established the baseline pre-treatment covariates. The treatment is whether the student attended college between 1965 and 1973 (the time when the next survey wave was administered). The outcome is an index that calculates the number of political activities the student engaged in after 1965. Specifically, the key variables in this study are:

- **college**: Treatment of whether the student attended college or not. 1 if the student attended college between 1965 and 1973, 0 otherwise.
- **ppnscal**: Outcome variable measuring the number of political activities the student participated in. Additive combination of whether the student voted in 1972 or 1980 (student_vote), attended a campaign rally or meeting (student_meeting), wore a campaign button (student_button), donated money to a campaign (student_money), communicated with an elected official (student_communicate), attended a demonstration or protest (student_demonstrate), was involved with a local community event (student_community), or some other political participation (student_other)

Otherwise, we also have covariates measured for survey responses to various questions about political attitudes. We have covariates measured for the students in the baseline year, covariates for their parents in the

baseline year, and covariates from follow-up surveys. **Be careful here**. In general, post-treatment covariates will be clear from the name (i.e. student_1973Married indicates whether the student was married in the 1973 survey). Be mindful that the baseline covariates were all measured in 1965, the treatment occurred between 1965 and 1973, and the outcomes are from 1973 and beyond. We will distribute the Appendix from Henderson and Chatfield that describes the covariates they used, but please reach out with any questions if you have questions about what a particular variable means.

```
# Load tidyverse and MatchIt
# Feel free to load other libraries as you wish
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.0     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(MatchIt)
library(purrr)
#install.packages("tinytex")
library("tinytex")

# Load ypsps data
ypsps <- read_csv('Computational-Social-Science-Training-Program/Projects/Project 6/data/ypsps.csv')
```

```
## Rows: 1254 Columns: 174
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (174): interviewid, college, student_vote, student_meeting, student_othe...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
head(ypsps)
```

```
## # A tibble: 6 x 174
##   interviewid college student_vote student_meeting student_other student_button
##         <dbl>   <dbl>        <dbl>           <dbl>         <dbl>          <dbl>
## 1           1       1            1               0             0              0
## 2           2       1            1               1             1              1
## 3           3       1            1               0             0              1
## 4           4       0            0               0             0              0
## 5           5       1            1               1             0              0
## 6           6       1            1               0             0              0
## # i 168 more variables: student_money <dbl>, student_communicate <dbl>,
## #   student_demonstrate <dbl>, student_community <dbl>, student_ppnscal <dbl>,
## #   student_PubAff <dbl>, student_Newspaper <dbl>, student_Radio <dbl>,
```

```
## #   student_TV <dbl>, student_Magazine <dbl>, student_FamTalk <dbl>,
## #   student_FrTalk <dbl>, student_AdultTalk <dbl>, student_PID <dbl>,
## #   student_SPID <dbl>, student_GovtOpinion <dbl>, student_GovtCrook <dbl>,
## #   student_GovtWaste <dbl>, student_TrGovt <dbl>, student_GovtSmart <dbl>, ...
```

Randomization

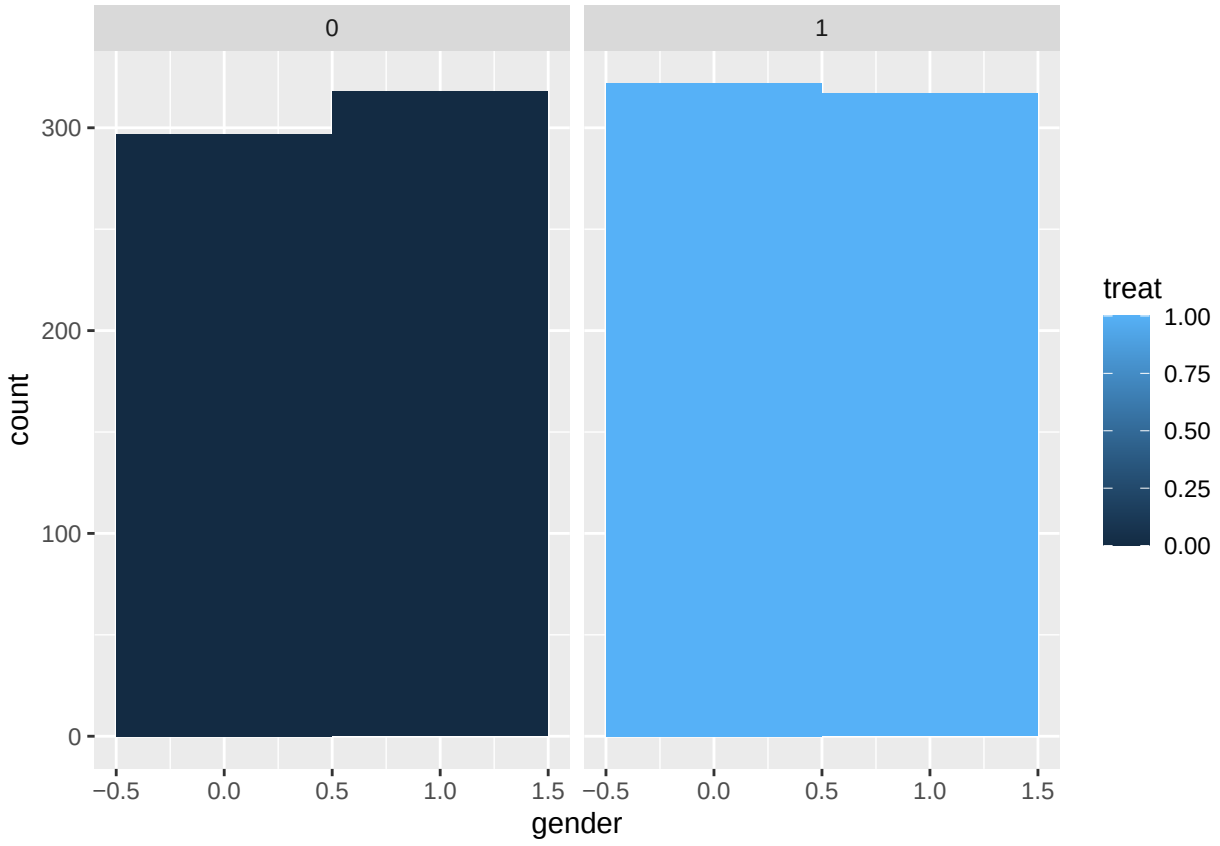
Matching is usually used in observational studies to to approximate random assignment to treatment. But could it be useful even in randomized studies? To explore the question do the following:

1. Generate a vector that randomly assigns each unit to either treatment or control
2. Choose a baseline covariate (for either the student or parent). A binary covariate is probably best for this exercise.
3. Visualize the distribution of the covariate by treatment/control condition. Are treatment and control balanced on this covariate?
4. Simulate the first 3 steps 10,000 times and visualize the distribution of treatment/control balance across the simulations.

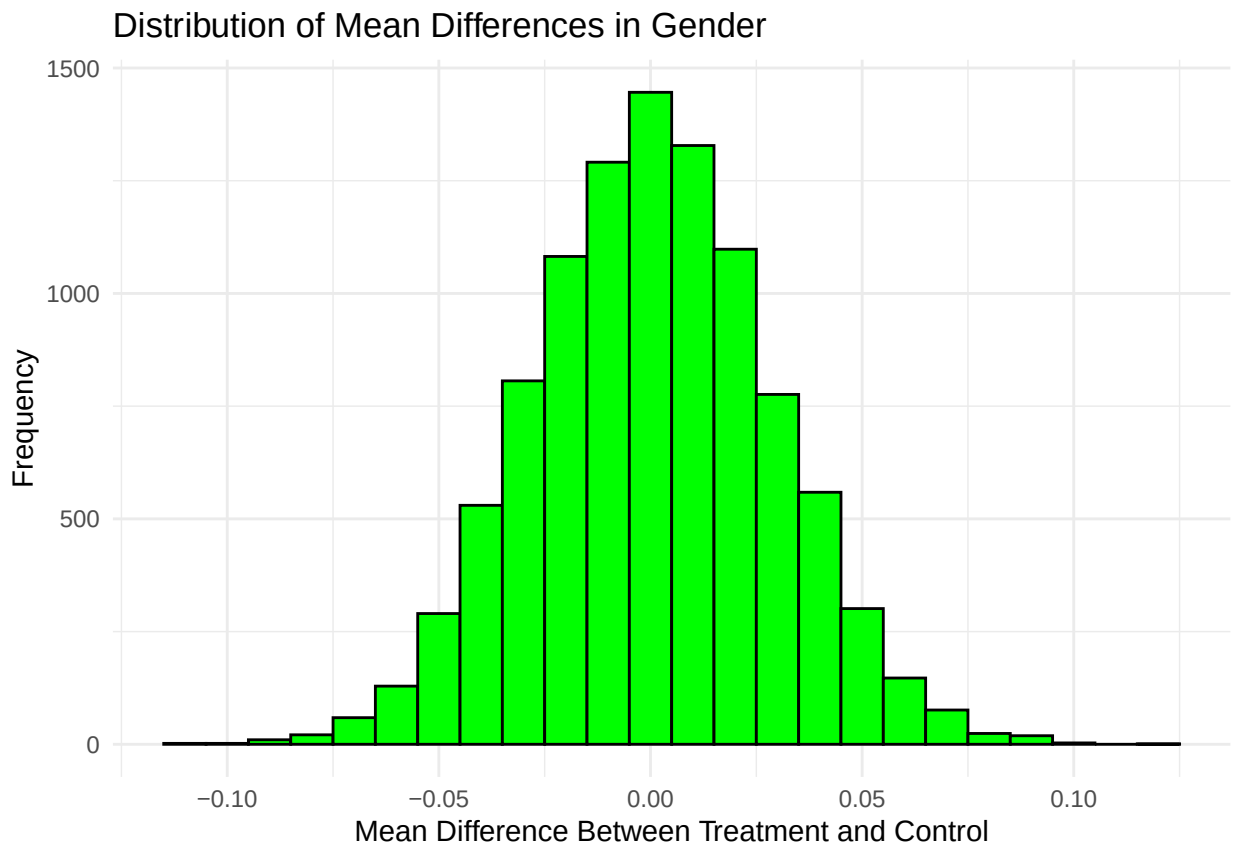
```
# Set seed
set.seed(42)

# Generate a vector that randomly assigns each unit to treatment/control
length(ypsps$interviewid)
```

```
## [1] 1254
```

```
n <- 1254

# generate vector
sim <- data.frame(treat = (sample(c(0, 1),
                                  size = n,
                                  replace = TRUE)))

# Choose a baseline covariate (use dplyr for this)
sim$gender <- ypsps$student_Gen

# Visualize the distribution by treatment/control (ggplot)

sim %>%
  ggplot(aes(x = gender, fill = treat)) +
  geom_histogram(binwidth = 1) +
  facet_grid(cols = vars(treat))
```




```r
# Simulate this 10,000 times (monte carlo simulation - see R Refresher for a hint)

simulate_diff <- function(data) {
  data_sample <- data %>%
    mutate(treat = sample(c(0, 1), n(), replace = TRUE)) %>% # Randomly reassign treatment/control
    group_by(treat) %>%
    summarise(mean_gender = mean(student_Gen, na.rm = TRUE)) %>%
    ungroup()

  diff <- diff(data_sample$mean_gender)
  return(diff)
}

set.seed(123) # Ensure reproducibility
simulations <- replicate(10000, simulate_diff(ypsps), simplify = TRUE)

# Visualize the distribution of the mean differences
ggplot(data.frame(MeanDifferences = simulations), aes(x = MeanDifferences)) +
  geom_histogram(binwidth = 0.01, fill = "green", color = "black") +
  labs(title = "Distribution of Mean Differences in Gender",
       x = "Mean Difference Between Treatment and Control",
       y = "Frequency") +
  theme_minimal()
```



Questions

1. **What do you see across your simulations? Why does independence of treatment assignment and baseline covariates not guarantee balance of treatment assignment and baseline covariates?**

Your Answer: Across 10,000 simulations, we see that the mean difference in the baseline variable (student gender) between treatment and control groups follows a normal distrubution. This means that the independence assumption between treatment and voting is correct. However, we can see that in some of the simulations, there is an imbalance in the baseline variable between treatment and control. This is because independence between treatment and baseline characteristics does not guarantee balance of baseline characteristics between treatment groups. This could be due to a few circumstances: skew of baseline characteristics within the population (i.e. perhaps only 5% of students are women - the small number of women could be unequally allocated by random chance). Random assignment could still assign a smaller number of similar participants to the same group. However, as the sample size increases, the balance of baseline characteristics between treatment and control groups should improve.

Propensity Score Matching

One Model

Select covariates that you think best represent the "true" model predicting whether a student chooses to attend college, and estimate a propensity score model to calculate the Average Treatment Effect on the

Treated (ATT). Plot the balance of the top 10 (or fewer if you select fewer covariates). Report the balance of the p-scores across both the treatment and control groups, and using a threshold of standardized mean difference of p-score $\leq .1$, report the number of covariates that meet that balance threshold.

```
##
## Call:
## glm(formula = college ~ parent_EducHH + parent_HHInc + student_Govt4All +
##     student_SchClub + student_GPA + student_SchOfficer + student_FPlans +
##     student_LifeWish + parent_TrGovt + student_OccClub + student_Newspaper +
##     student_Gen, family = binomial(link = "logit"), data = ypsps)
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)         0.39553    0.63449   0.623 0.533038
## parent_EducHH       0.47138    0.05661   8.327  < 2e-16 ***
## parent_HHInc        0.16237    0.03356   4.838 1.31e-06 ***
## student_Govt4All   -0.03644    0.10546  -0.346 0.729677
## student_SchClub     0.23475    0.07418   3.165 0.001552 **
## student_GPA        -0.75077    0.11369  -6.604 4.00e-11 ***
## student_SchOfficer -0.27408    0.08303  -3.301 0.000964 ***
## student_FPlans     -0.18718    0.07352  -2.546 0.010903 *
## student_LifeWish    0.03368    0.07287   0.462 0.643968
## parent_TrGovt       0.13036    0.10810   1.206 0.227844
## student_OccClub    -0.13822    0.06112  -2.262 0.023726 *
## student_Newspaper  -0.07438    0.05003  -1.487 0.137079
## student_Gen         0.76430    0.14732   5.188 2.12e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1638.3  on 1253  degrees of freedom
## Residual deviance: 1297.0  on 1241  degrees of freedom
## AIC: 1323
##
## Number of Fisher Scoring iterations: 4
```

```
##
## Call:
## matchit(formula = college ~ parent_EducHH + parent_HHInc + +student_SchClub +
##     student_GPA + student_SchOfficer + student_FPlans + parent_TrGovt +
##     student_OccClub + student_Newspaper + student_Gen, data = ypsps,
##     method = "exact", estimand = "ATT")
##
## Summary of Balance for Matched Data:
##                    Means Treated Means Control Std. Mean Diff. Var. Ratio
## parent_EducHH             2.6667        2.6667              -0     0.9810
## parent_HHInc              7.0000        7.0000               0     0.9810
## student_SchClub           1.0000        1.0000               0     0.2452
## student_GPA               2.7333        2.7333               0     0.9810
## student_SchOfficer        2.4667        2.4667               0     0.9810
## student_FPlans            1.4000        1.4000               0     0.9810
## parent_TrGovt             2.0667        2.0667               0     0.9810
## student_OccClub           1.4000        1.4000               0     0.9810
```

```
## student_Newspaper         1.8667        1.8667              -0     0.9810
## student_Gen               0.8667        0.8667               0          .
##                     eCDF Mean eCDF Max Std. Pair Dist.
## parent_EducHH               0        0               0
## parent_HHInc                0        0               0
## student_SchClub             0        0               0
## student_GPA                 0        0               0
## student_SchOfficer          0        0               0
## student_FPlans              0        0               0
## parent_TrGovt               0        0               0
## student_OccClub             0        0               0
## student_Newspaper           0        0               0
## student_Gen                 0        0               0
##
## Sample Sizes:
##               Control Treated
## All            451.       803
## Matched (ESS)   11.84      15
## Matched         13.        15
## Unmatched      438.       788
## Discarded        0.         0


##
## Call:
## lm(formula = student_ppnscal ~ college + parent_EducHH + parent_HHInc +
##     student_Govt4All + student_SchClub + student_GPA + student_SchOfficer +
##     student_FPlans + student_LifeWish + parent_TrGovt + student_OccClub +
##     student_Newspaper + student_Gen, data = match_exact_att_data,
##     weights = weights)
##
## Weighted Residuals:
##     Min      1Q  Median      3Q     Max
## -1.6682 -0.7494 -0.2713  0.8142  2.0688
##
## Coefficients: (1 not defined because of singularities)
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -24.275423  15.899691  -1.527   0.1476
## college              0.655153   0.577093   1.135   0.2741
## parent_EducHH       -0.705505   0.532562  -1.325   0.2051
## parent_HHInc         1.319605   0.911150   1.448   0.1681
## student_Govt4All     0.281997   0.632455   0.446   0.6621
## student_SchClub            NA         NA      NA       NA
## student_GPA          0.349933   1.301982   0.269   0.7918
## student_SchOfficer  -0.216597   0.479466  -0.452   0.6579
## student_FPlans      -0.195278   0.495146  -0.394   0.6988
## student_LifeWish    -0.004675   0.317383  -0.015   0.9884
## parent_TrGovt        9.235568   4.762123   1.939   0.0715 .
## student_OccClub      0.604820   0.889627   0.680   0.5070
## student_Newspaper   -0.449752   0.383232  -1.174   0.2589
## student_Gen         -1.833155   1.436179  -1.276   0.2212
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.459 on 15 degrees of freedom
```

```
## Multiple R-squared:  0.5557, Adjusted R-squared:  0.2003
## F-statistic: 1.564 on 12 and 15 DF,  p-value: 0.2047
```

```
## [1] 0.6551527
```

```
## Balance Measures
##                       Type Diff.Un     M.Threshold.Un
## parent_EducHH      Contin.  0.7880 Not Balanced, >0.1
## parent_HHInc       Contin.  0.6445 Not Balanced, >0.1
## student_Govt4All   Contin. -0.0641     Balanced, <0.1
## student_SchClub    Contin.  0.3182 Not Balanced, >0.1
## student_GPA        Contin. -0.5332 Not Balanced, >0.1
## student_SchOfficer Contin. -0.2956 Not Balanced, >0.1
## student_FPlans     Contin. -0.2966 Not Balanced, >0.1
## student_LifeWish   Contin. -0.1688 Not Balanced, >0.1
## parent_TrGovt      Contin.  0.0804     Balanced, <0.1
## student_OccClub    Contin. -0.1678 Not Balanced, >0.1
## student_Newspaper  Contin. -0.2209 Not Balanced, >0.1
## student_Gen         Binary  0.1156 Not Balanced, >0.1
##
## Balance tally for mean differences
##                     count
## Balanced, <0.1          2
## Not Balanced, >0.1     10
##
## Variable with the greatest mean difference
##       Variable Diff.Un     M.Threshold.Un
##  parent_EducHH   0.788 Not Balanced, >0.1
##
## Sample sizes
##     Control Treated
## All     451     803
```

Simulations

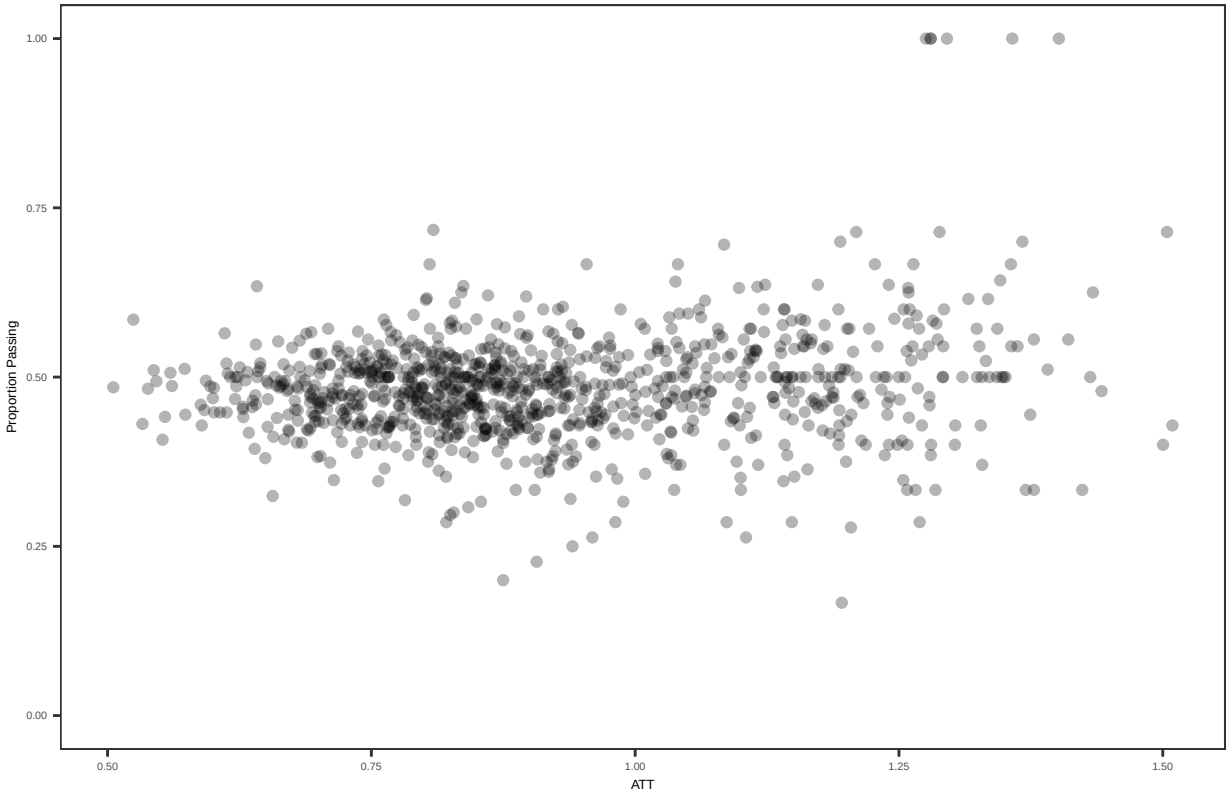
Henderson/Chatfield argue that an improperly specified propensity score model can actually *increase* the bias of the estimate. To demonstrate this, they simulate 800,000 different propensity score models by choosing different permutations of covariates. To investigate their claim, do the following:

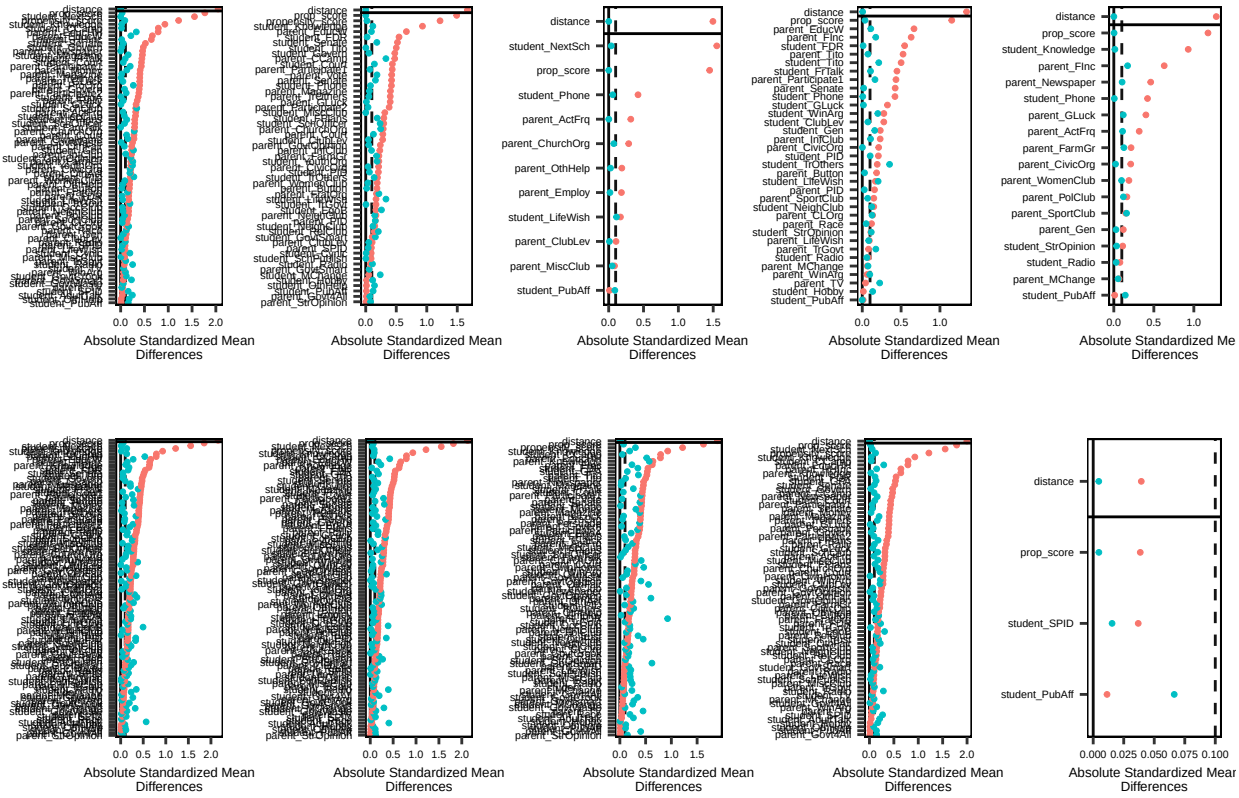
- Using as many simulations as is feasible (at least 10,000 should be ok, more is better!), randomly select the number of and the choice of covariates for the propensity score model.
- For each run, store the ATT, the proportion of covariates that meet the standardized mean difference $\leq .1$ threshold, and the mean percent improvement in the standardized mean difference. You may also wish to store the entire models in a list and extract the relevant attributes as necessary.
- Plot all of the ATTs against all of the balanced covariate proportions. You may randomly sample or use other techniques like transparency if you run into overplotting problems. Alternatively, you may use plots other than scatterplots, so long as you explore the relationship between ATT and the proportion of covariates that meet the balance threshold.
- Finally choose 10 random models and plot their covariate balance plots (you may want to use a library like gridExtra to arrange these)

Note: There are lots of post-treatment covariates in this dataset (about 50!)! You need to be careful not to include these in the pre-treatment balancing. Many of you are probably used to selecting or dropping columns manually, or positionally. However, you may not always have a convenient arrangement of columns, nor is it fun to type out 50 different column names. Instead see if you can use dplyr 1.0.0 functions to programatically drop post-treatment variables (here is a useful tutorial).

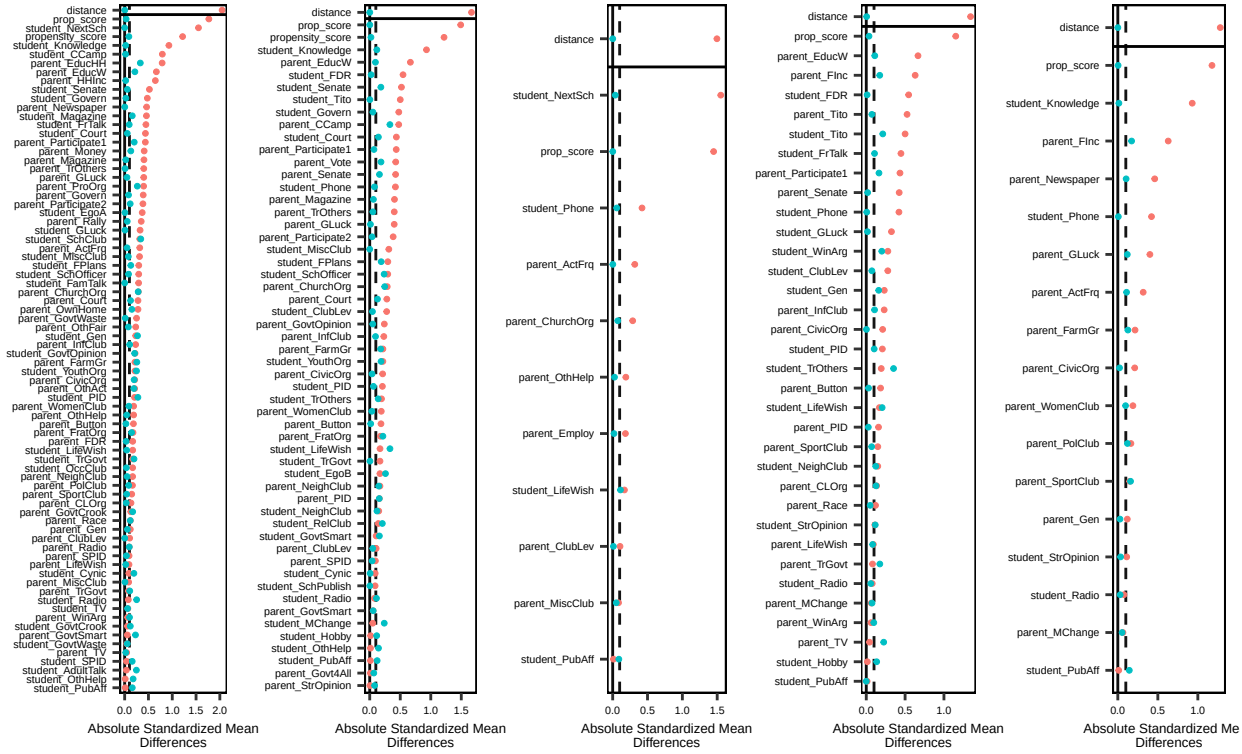
```
## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge
```

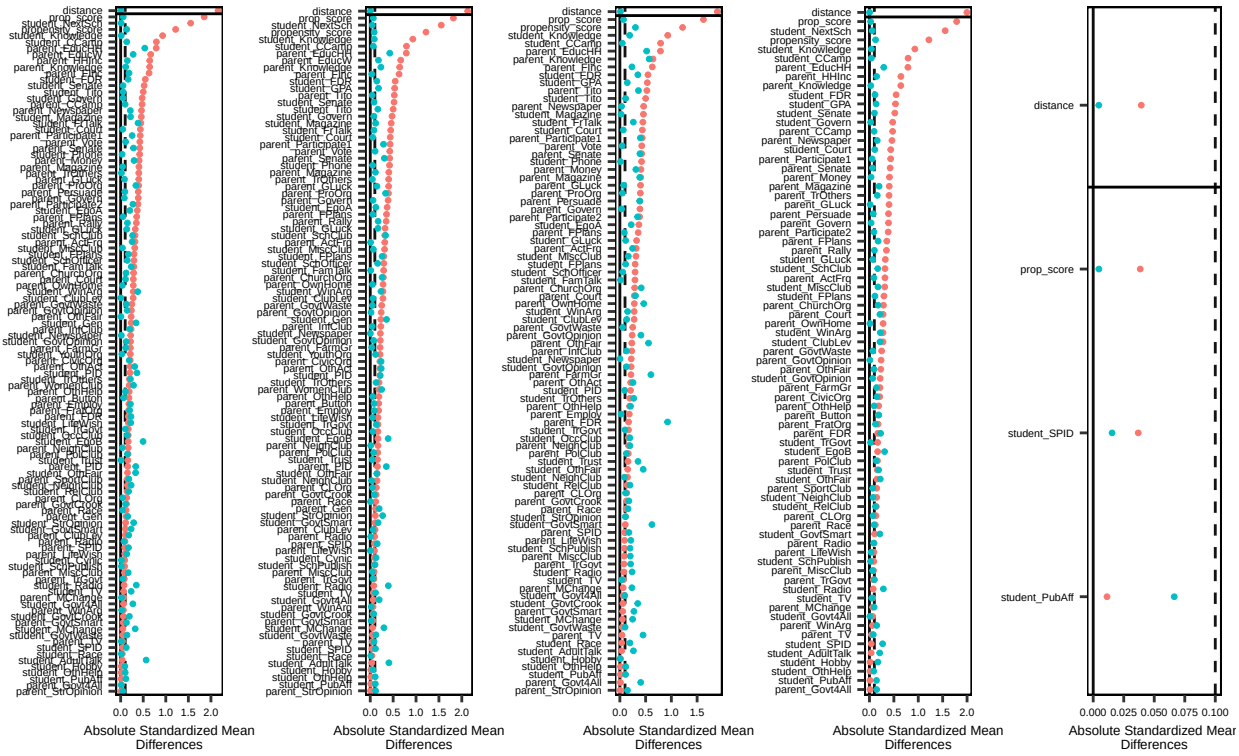
ATT v. Proportion for Simulations




distance
prop_score
student_NextSch
student_Phone
parent_ActFrq
parent_ChurchOrg
parent_OthHelp
parent_Employ
student_LifeWish
parent_ClubLev
parent_MiscClub
student_PubAff
parent_FInc
parent_Newspaper
student_Knowledge
parent_GLuck
parent_FarmGr
parent_CivicOrg
parent_WomenClub
parent_PolClub
parent_SportClub
parent_Gen
student_StrOpinion
student_Radio
parent_MChange
0.0 0.5 1.0 1.5 2.0
Absolute Standardized Mean Differences

distance
prop_score
student_SPID
student_PubAff
0.0 0.5 1.0 1.5 2.0
0.000 0.025 0.050 0.075 0.100
Absolute Standardized Mean Differences


distance
prop_score
student_NextSch
propensity_score
student_Knowledge
student_CCamp
parent_EducHH
parent_EducW
parent_HHinc
student_Senate
student_Govern
parent_Newspaper
student_Magazine
student_FrTalk
student_Court
parent_Participate1
parent_Money
parent_Magazine
parent_TrOthers
parent_GLuck
parent_ProOrg
parent_Govern
parent_Particigate2
student_EgoA
parent_Rally
student_GLuck
student_SchClub
parent_ActFrq
student_MiscClub
student_FPlans
student_SchOfficer
student_FamTalk
parent_ChurchOrg
parent_Court
parent_OwnHome
parent_GovtWaste
parent_OthFair
student_Gen
parent_InfClub
student_GovtOpinion
parent_FarmGr
student_YouthOrg
parent_CivicOrg
parent_OthAct
student_PID
parent_WomenClub
parent_OthHelp
parent_Button
parent_FratOrg
parent_FDR
student_LifeWish
student_TrGovt
student_OccClub
parent_NeighClub
parent_PolClub
parent_SportClub
parent_CLOrg
parent_GovtCrook
parent_Race
parent_Gen
parent_ClubLev
parent_Radio
parent_SPID
parent_LifeWish
student_Cynic
parent_MiscClub
parent_TrGovt
student_Radio
student_TV
parent_WinArg
student_GovtCrook
parent_GovtSmart
student_GovtWaste
parent_TV
student_SPID
student_AdultTalk
student_OthHelp
student_PubAff
0.0 0.5 1.0 1.5 2.0
Absolute Standardized Mean Differences
distance
prop_score
propensity_score
student_Knowledge
parent_EducW
student_FDR
student_Senate
student_Tito
student_Govern
parent_CCamp
student_Court
parent_Participate1
parent_Vote
parent_Senate
student_Phone
parent_Magazine
parent_TrOthers
parent_GLuck
parent_Participate2
student_MiscClub
student_FPlans
student_SchOfficer
parent_ChurchOrg
parent_Court
student_ClubLev
parent_GovtOpinion
parent_InfClub
parent_FarmGr
student_YouthOrg
parent_CivicOrg
student_PID
student_TrOthers
parent_WomenClub
parent_Button
parent_FratOrg
student_LifeWish
student_TrGovt
student_EgoB
parent_NeighClub
parent_PID
student_NeighClub
student_RelClub
student_GovtSmart
parent_ClubLev
parent_SPID
student_Cynic
student_SchPublish
student_Radio
parent_GovtSmart
student_MChange
student_Hobby
student_OthHelp
student_PubAff
parent_Govt4All
parent_StrOpinion
0.0 0.5 1.0 1.5
Absolute Standardized Mean Differences
distance
student_NextSch
prop_score
student_Phone
parent_ActFrq
parent_ChurchOrg
parent_OthHelp
parent_Employ
student_LifeWish
parent_ClubLev
parent_MiscClub
student_PubAff
0.0 0.5 1.0 1.5
Absolute Standardized Mean Differences
distance
prop_score
parent_EducW
parent_FInc
student_FDR
parent_Tito
student_Tito
student_FrTalk
parent_Participate1
parent_Senate
student_Phone
student_GLuck
student_WinArg
student_ClubLev
student_Gen
parent_InfClub
parent_CivicOrg
student_PID
student_TrOthers
parent_Button
student_LifeWish
parent_PID
parent_SportClub
student_NeighClub
parent_CLOrg
parent_Race
student_StrOpinion
parent_LifeWish
parent_TrGovt
student_Radio
parent_MChange
parent_WinArg
parent_TV
student_Hobby
student_PubAff
0.0 0.5 1.0
Absolute Standardized Mean Differences
distance
prop_score
student_Knowledge
parent_FInc
parent_Newspaper
student_Phone
parent_GLuck
parent_ActFrq
parent_FarmGr
parent_CivicOrg
parent_WomenClub
parent_PolClub
parent_SportClub
parent_Gen
student_StrOpinion
student_Radio
parent_MChange
student_PubAff
0.0 0.5 1.0
Absolute Standardized Me Differences



Questions

1. **How many simulations resulted in models with a higher proportion of balanced covariates? Do you have any concerns about this?** Your Answer: Compared to my model, where only 2/12 covariates were balanced, essentially all the runs had a higher proportion of balance.

2. **Analyze the distribution of the ATTs. Do you have any concerns about this distribution?** Your Answer: ATT ranged from 0.6 to 1.5, with the largest concentration between 0.7 and 1. Notably, as ATT increases the proportion of balanced covarariates becomes wider. There's a slight positive association between ATT and proportion of balanced covariates.

3. **Do your 10 randomly chosen covariate balance plots produce similar numbers on the same covariates? Is it a concern if they do not?** Your Answer: The mean differences for specific covariates are generally similar across balance plots. Given that the sample remains the same and the simulation is just changing the number of covariates included, it would be a concern if the balance changed across balance plots.

   NOTE: I was not able to run 10,000 simulations within a reasonable amount of time. I ended up running 1,000 simulations, so the results would likely be better if I had been able to run more sims.

Matching Algorithm of Your Choice

Simulate Alternative Model

Henderson/Chatfield propose using genetic matching to learn the best weights for Mahalanobis distance matching. Choose a matching algorithm other than the propensity score (you may use genetic matching if you wish, but it is also fine to use the greedy or optimal algorithms we covered in lab instead). Repeat the same steps as specified in Section 4.2 and answer the following questions:

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: Fewer control units than treated units; not all treated units will get
## a match.
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: Fewer control units than treated units; not all treated units will get a match.
## Fewer control units than treated units; not all treated units will get a match.
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: Fewer control units than treated units; not all treated units will get a match.
## Fewer control units than treated units; not all treated units will get a match.
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: Fewer control units than treated units; not all treated units will get a match.
## Fewer control units than treated units; not all treated units will get a match.
## Fewer control units than treated units; not all treated units will get a match.
## Fewer control units than treated units; not all treated units will get a match.
## Fewer control units than treated units; not all treated units will get a match.
## Fewer control units than treated units; not all treated units will get a match.
## Fewer control units than treated units; not all treated units will get a match.
```

```
## Warning: glm.fit: algorithm did not converge
```

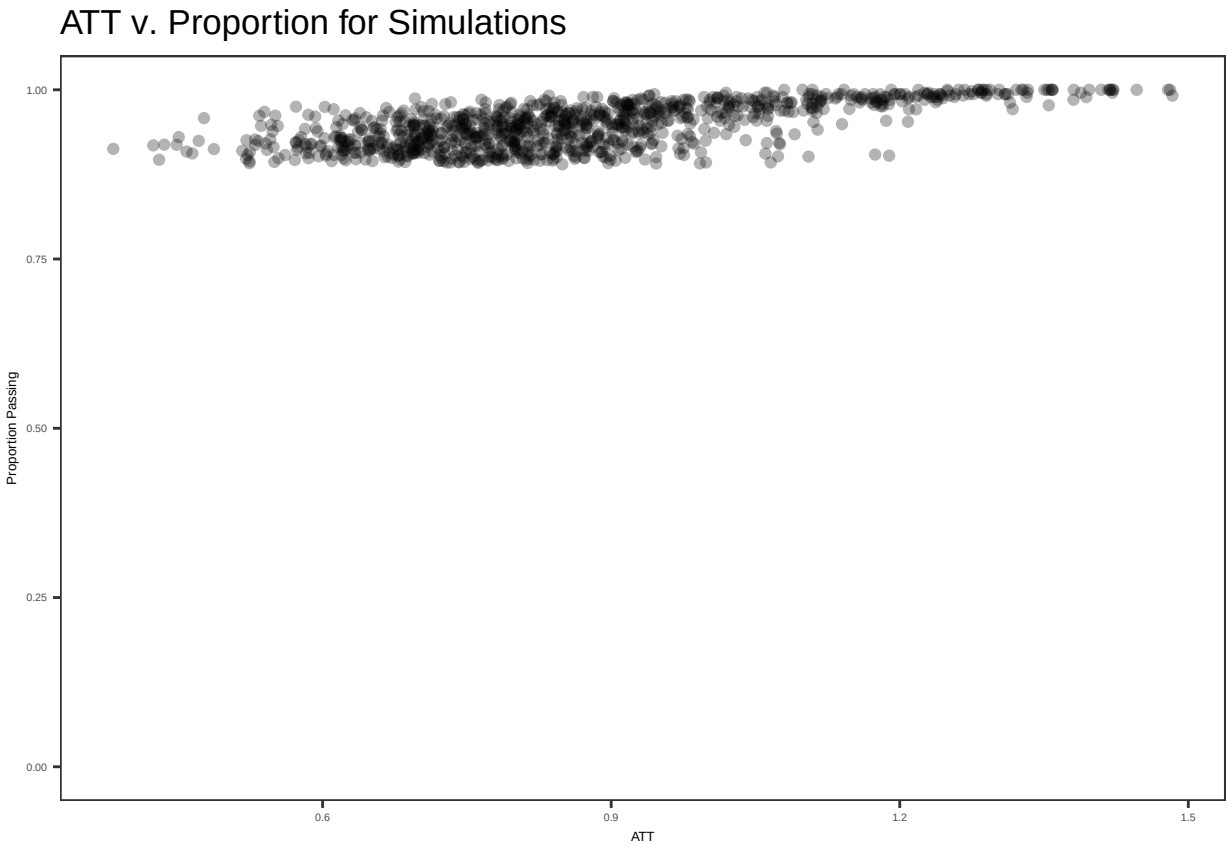
```
## Warning: Fewer control units than treated units; not all treated units will get a match.
## Fewer control units than treated units; not all treated units will get a match.
```

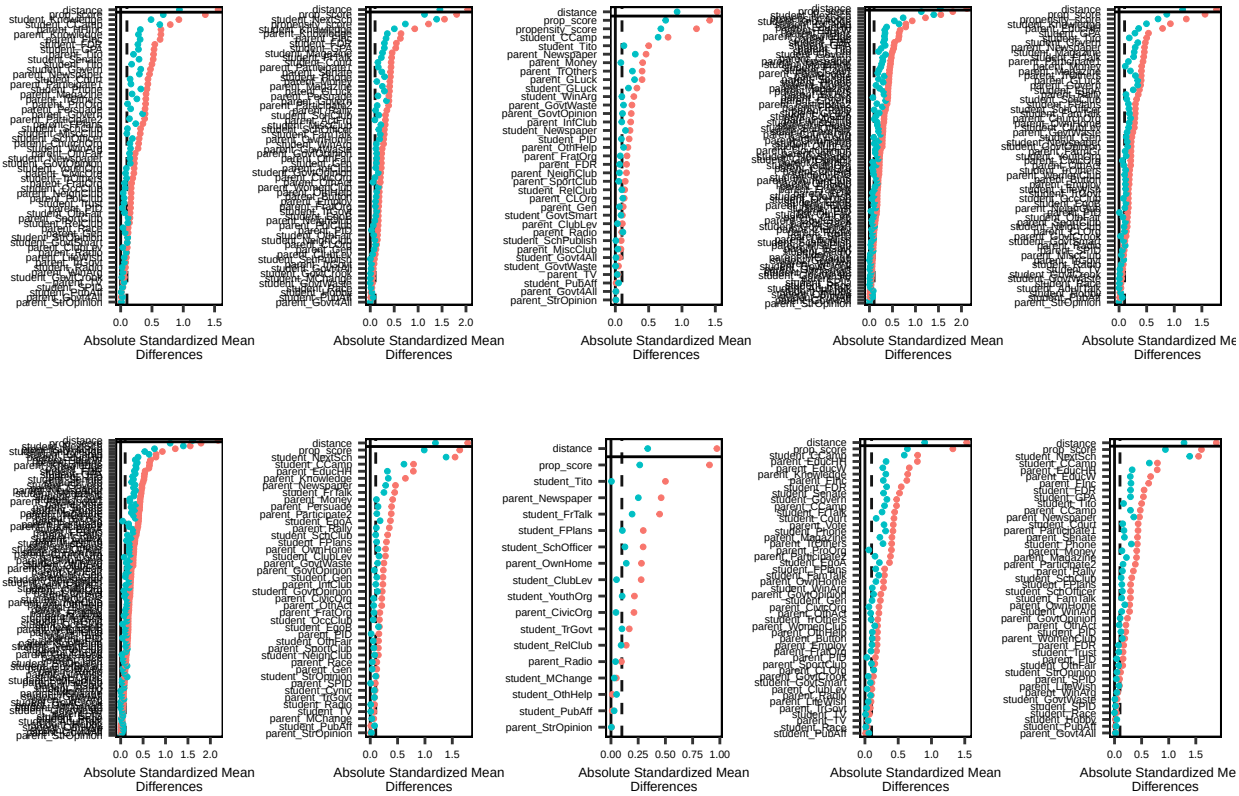
```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: Fewer control units than treated units; not all treated units will get
## a match.
```

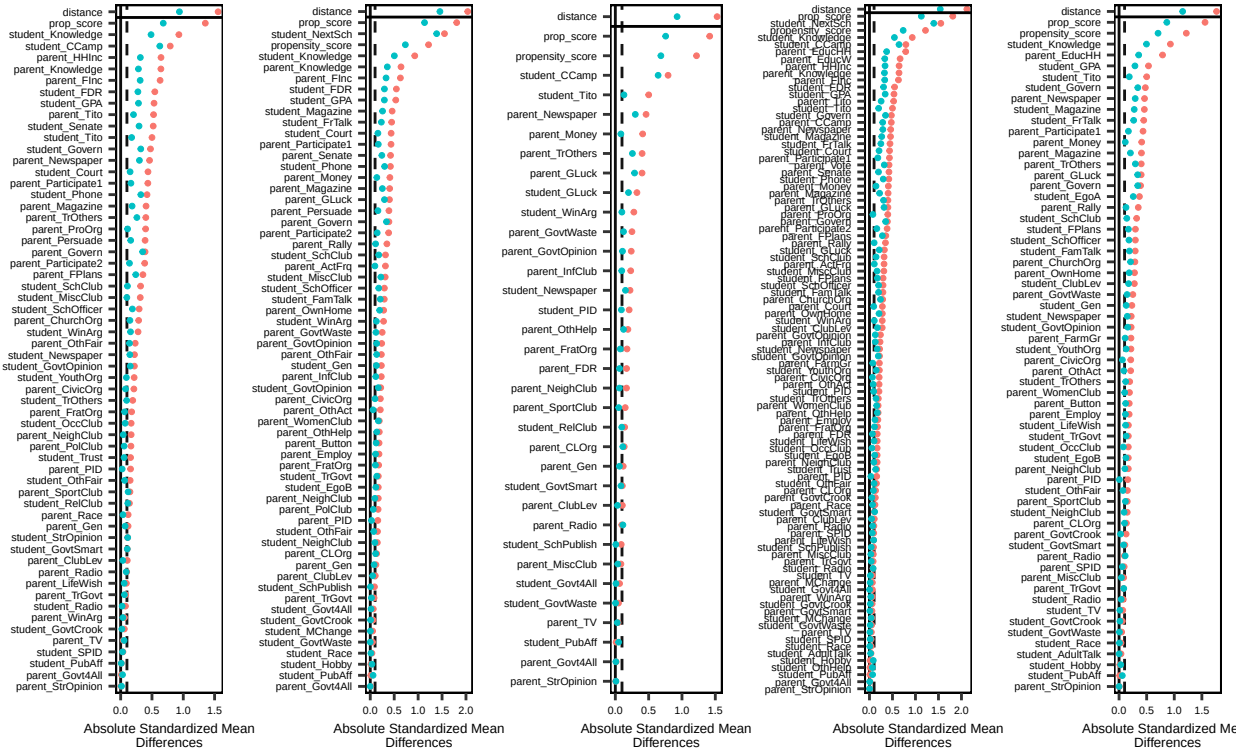
```
## Warning: glm.fit: algorithm did not converge
```

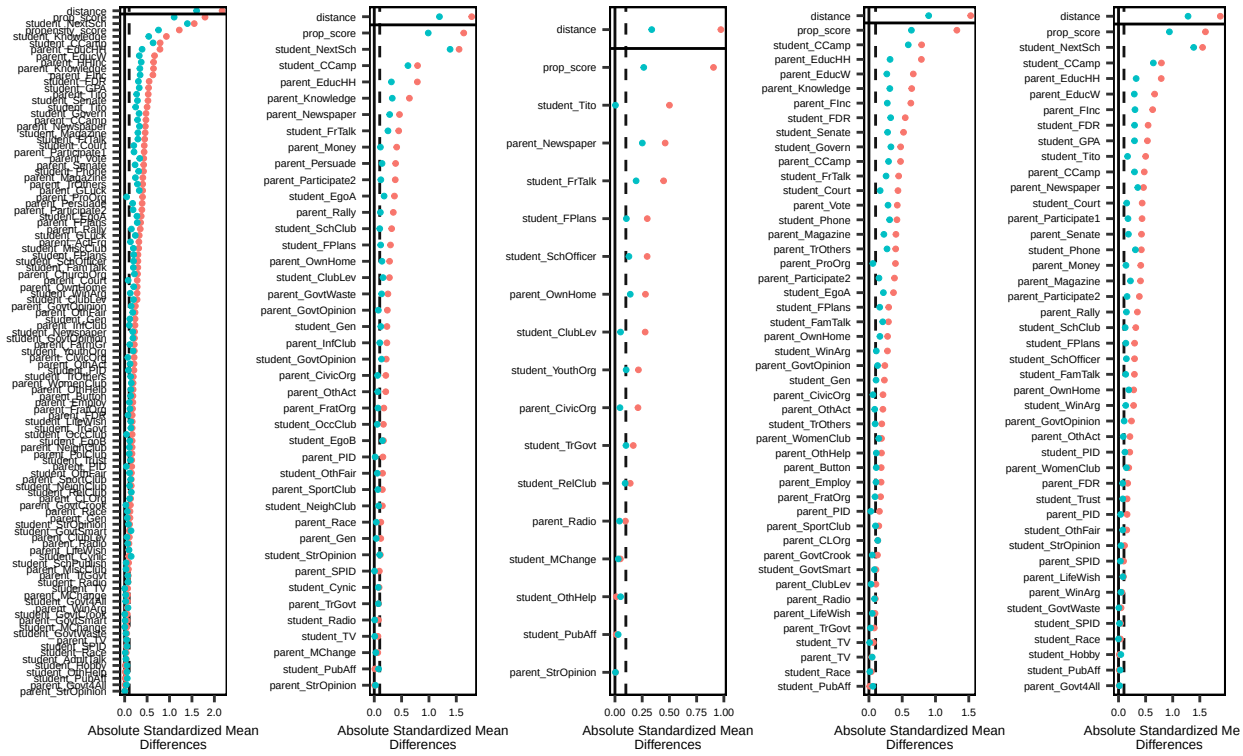
```
## Warning: glm.fit: algorithm did not converge

## Warning: Fewer control units than treated units; not all treated units will get
## a match.

## Warning: glm.fit: algorithm did not converge

## Warning: Fewer control units than treated units; not all treated units will get a match.
## Fewer control units than treated units; not all treated units will get a match.
## Fewer control units than treated units; not all treated units will get a match.
## Fewer control units than treated units; not all treated units will get a match.
## Fewer control units than treated units; not all treated units will get a match.
```



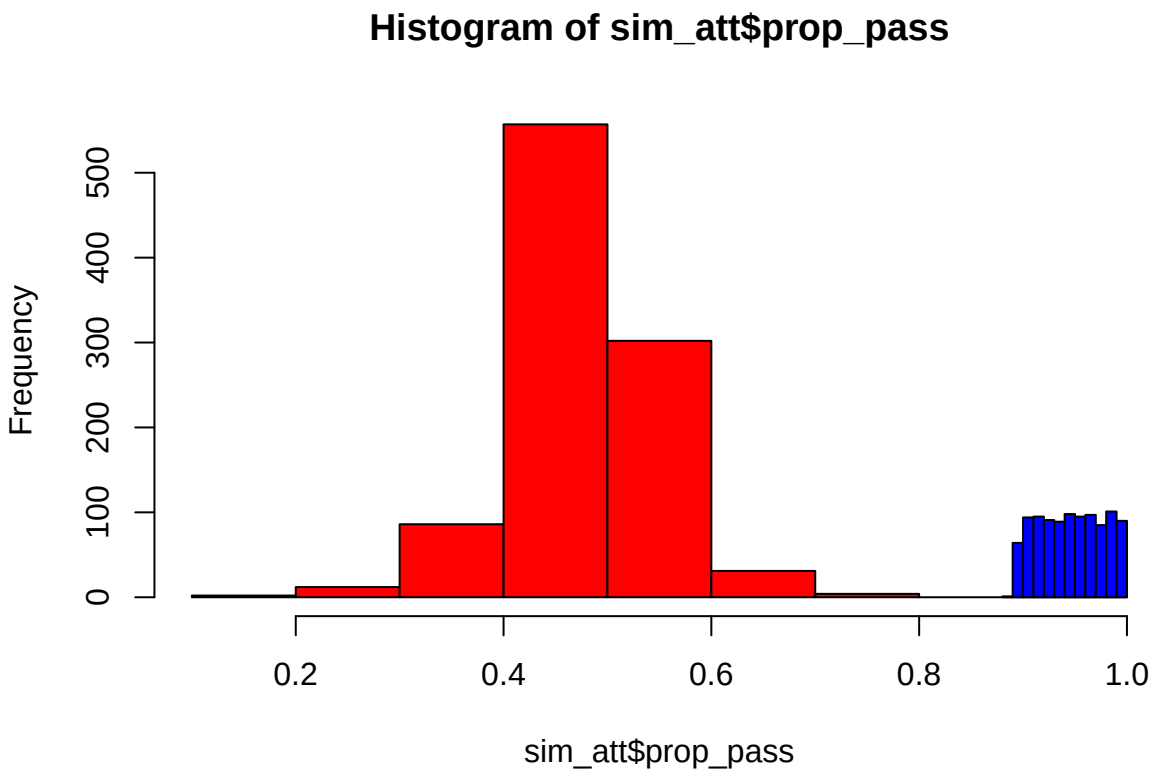

distance
prop_score
propensity_score
student_CCamp
student_Tito
parent_Newspaper
parent_Money
parent_TrOthers
parent_GLuck
student_GLuck
student_WinArg
parent_GovtWaste
parent_GovtOpinion
parent_InfClub
student_Newspaper
student_PID
parent_OthHelp
parent_FratOrg
parent_FDR
parent_NeighClub
parent_SportClub
student_RelClub
parent_CLOrg
parent_Gen
student_GovtSmart
parent_ClubLev
parent_Radio
student_SchPublish
parent_MiscClub
student_Govt4All
student_GovtWaste
parent_TV
student_PubAff
parent_Govt4All
parent_StrOpinion
0.0 0.5 1.0 1.5
0.0 0.5 1.0 1.5 2.0
Absolute Standardized Mean Differences

distance
prop_score
student_Tito
parent_Newspaper
student_FrTalk
student_FPlans
student_SchOfficer
parent_OwnHome
student_ClubLev
student_YouthOrg
parent_CivicOrg
student_TrGovt
student_RelClub
parent_Radio
student_MChange
student_OthHelp
student_PubAff
parent_StrOpinion
0.00 0.25 0.50 0.75 1.00
Absolute Standardized Mean Differences


distance
prop_score
student_Knowledge
student_CCamp
parent_HHInc
parent_Knowledge
parent_FInc
student_FDR
student_GPA
parent_Tito
student_Senate
student_Tito
student_Govern
parent_Newspaper
student_Court
parent_Participate1
student_Phone
parent_Magazine
parent_TrOthers
parent_ProOrg
parent_Persuade
parent_Govern
parent_Participate2
parent_FPlans
student_SchClub
student_MiscClub
student_SchOfficer
parent_ChurchOrg
student_WinArg
parent_OthFair
student_Newspaper
student_GovtOpinion
student_YouthOrg
parent_CivicOrg
student_TrOthers
parent_FratOrg
student_OccClub
parent_NeighClub
parent_PolClub
student_Trust
parent_PID
student_OthFair
parent_SportClub
student_RelClub
parent_Race
parent_Gen
student_StrOpinion
student_GovtSmart
parent_ClubLev
parent_Radio
parent_LifeWish
parent_TrGovt
student_Radio
parent_WinArg
student_GovtCrook
parent_TV
student_SPID
student_PubAff
parent_Govt4All
parent_StrOpinion
0.0 0.5 1.0 1.5
Absolute Standardized Mean Differences
distance
prop_score
student_NextSch
propensity_score
student_Knowledge
parent_Knowledge
parent_FInc
student_FDR
student_GPA
student_Magazine
student_FrTalk
student_Court
parent_Participate1
parent_Senate
student_Phone
parent_Money
parent_Magazine
parent_GLuck
parent_Govern
parent_Participate2
parent_Rally
student_SchClub
student_MiscClub
student_SchOfficer
student_FamTalk
parent_OwnHome
student_WinArg
parent_GovtWaste
parent_GovtOpinion
parent_OthFair
student_Gen
parent_InfClub
student_GovtOpinion
parent_CivicOrg
parent_OthAct
parent_WomenClub
parent_OthHelp
parent_Button
parent_Employ
parent_FratOrg
student_TrGovt
student_EgoB
parent_NeighClub
parent_PolClub
parent_PID
student_OthFair
student_NeighClub
parent_CLOrg
parent_Gen
parent_ClubLev
student_SchPublish
parent_TrGovt
student_Govt4All
student_GovtCrook
student_MChange
student_GovtWaste
student_Race
student_Hobby
student_PubAff
parent_Govt4All
0.0 0.5 1.0 1.5 2.0
Absolute Standardized Mean Differences
distance
prop_score
propensity_score
student_CCamp
student_Tito
parent_Newspaper
parent_Money
parent_TrOthers
parent_GLuck
student_GLuck
student_WinArg
parent_GovtWaste
parent_GovtOpinion
parent_InfClub
student_Newspaper
student_PID
parent_OthHelp
parent_FratOrg
parent_FDR
parent_NeighClub
parent_SportClub
student_RelClub
parent_CLOrg
parent_Gen
student_GovtSmart
parent_ClubLev
parent_Radio
student_SchPublish
parent_MiscClub
student_Govt4All
student_GovtWaste
parent_TV
student_PubAff
parent_Govt4All
parent_StrOpinion
0.0 0.5 1.0 1.5
Absolute Standardized Mean Differences
distance
prop_score
student_NextSch
propensity_score
student_Knowledge
parent_Knowledge
parent_FInc
student_FDR
student_GPA
parent_Tito
student_Govern
parent_Newspaper
student_Magazine
student_FrTalk
student_Court
parent_Participate1
parent_Senate
student_Phone
parent_Money
parent_Magazine
parent_TrOthers
parent_GLuck
parent_ProOrg
parent_Govern
parent_Participate2
parent_Rally
student_SchClub
student_MiscClub
student_FPlans
parent_ChurchOrg
student_FamTalk
parent_Court
parent_OwnHome
student_WinArg
student_ClubLev
parent_GovtOpinion
parent_InfClub
student_Newspaper
student_GovtOpinion
parent_FarmGr
student_YouthOrg
parent_CivicOrg
parent_OthAct
student_PID
student_TrOthers
parent_WomenClub
parent_OthHelp
parent_Employ
parent_FratOrg
parent_FDR
student_LifeWish
student_OccClub
student_EgoB
parent_NeighClub
student_Trust
parent_PID
student_OthFair
parent_GovtCrook
parent_Race
student_GovtSmart
parent_ClubLev
parent_Radio
parent_SPID
parent_LifeWish
student_SchPublish
parent_MiscClub
parent_TrGovt
student_Radio
parent_MChange
student_Govt4All
parent_WinArg
student_GovtCrook
parent_GovtSmart
student_MChange
student_GovtWaste
parent_TV
student_SPID
student_Race
student_AdultTalk
student_Hobby
student_OthHelp
student_PubAff
parent_Govt4All
parent_StrOpinion
0.0 0.5 1.0 1.5 2.0
Absolute Standardized Mean Differences
distance
prop_score
propensity_score
student_Knowledge
parent_EducHH
student_GPA
student_Tito
student_Govern
parent_Newspaper
student_Magazine
student_FrTalk
parent_Participate1
parent_Money
parent_Magazine
parent_TrOthers
parent_GLuck
parent_Govern
student_EgoA
parent_Rally
student_SchClub
student_FPlans
student_SchOfficer
student_FamTalk
parent_ChurchOrg
parent_OwnHome
student_ClubLev
parent_GovtWaste
student_Gen
student_Newspaper
student_GovtOpinion
parent_FarmGr
student_YouthOrg
parent_CivicOrg
parent_OthAct
student_TrOthers
parent_WomenClub
parent_Button
parent_Employ
student_LifeWish
student_TrGovt
student_OccClub
student_EgoB
parent_NeighClub
parent_PID
student_OthFair
parent_SportClub
student_NeighClub
parent_CLOrg
parent_GovtCrook
student_GovtSmart
parent_Radio
parent_SPID
parent_MiscClub
parent_TrGovt
student_Radio
student_TV
student_GovtCrook
student_GovtWaste
student_Race
student_AdultTalk
student_Hobby
student_PubAff
parent_StrOpinion
0.0 0.5 1.0 1.5
Absolute Standardized Me Differences




```r
# Visualization for distributions of percent improvement
 hgA <- hist(sim_att$prop_pass, plot = FALSE) # Save first histogram data
 hgB <- hist(sim_att2$prop_pass, plot = FALSE) # Save 2nd histogram data

 plot(hgA, col = 'red') # Plot 1st histogram using a transparent color
 plot(hgB, col = 'blue', add = TRUE) # Add 2nd histogram using different color
```



Questions

1. **Does your alternative matching method have more runs with higher proportions of balanced covariates?** Your Answer: Yes - the optimal matching method has a much higher proportion of balanced covariates - between 90-100 percent.

2. **Use a visualization to examine the change in the distribution of the percent improvement in balance in propensity score matching vs. the distribution of the percent improvement in balance in your new method. Which did better? Analyze the results in 1-2 sentences.** Your Answer: We see that the proportion of balanced covariates in the optimal matching model is much higher than propensity score matching. The optimal matching model performs about twice as well as propensity score matching, suggesting that propensity score matching is not a good fit for the data.

   NOTE: I was not able to run 10,000 simulations within a reasonable amount of time. I ended up running 1,000 simulations, so the results would likely be better if I had been able to run more sims.

Optional: Looking ahead to the discussion questions, you may choose to model the propensity score using an algorithm other than logistic regression and perform these simulations again, if you wish to explore the second discussion question further.

Discussion Questions

1. **Why might it be a good idea to do matching even if we have a randomized or as-if-random design?** Your Answer: Within-group heterogeneity would be a good reason to use matching.

If the outcomes of interest may differ across specific sub-populations, or there's an interest in exploring the ATT for a subgroup, matching would help isolate the treatment effect. Further, regardless of the outcome of interest, randomization does not ensure balance between treatment and control groups. Matching can help alleviate differences in the makeup of treatment/control groups that could be related to the outcome of interest.

2. **The standard way of estimating the propensity score is using a logistic regression to estimate probability of treatment. Given what we know about the curse of dimensionality, do you think there might be advantages to using other machine learning algorithms (decision trees, bagging/boosting forests, ensembles, etc.) to estimate propensity scores instead?** Your Answer: Assuming that computational resources are not a problem, it makes sense that ML methods could improve propensity score matching. However, we saw last semester that sometimes logistic regression did not perform meaningfully better than ML methods. While there is a risk of overfitting with logistic regression, there are ML models that suffer the same issues. However, BART, lasso, or ridge regression could be good fits as they minimize covariates that don't meaningfully contribute to the outcome of interest.