Final Q1

Kenny Mai

12/8/2020

Question 1: Propensity Scores (6 points).

All worlds should have 5 covariates, one binary treatment variable, and potential outcomes for one variable. The covariates should have a dependence structure. The response surface for World A can be linear and should satisfy all other assumptions as well. At least one of the response surfaces (that is, E[Y(0) | X] or E[Y(1) | X] in World B should be non-linear (with R2 less than .75) but the other assumptions should hold. World C can use the same response surface as World B but should violate one of the other assumptions. You should estimate a causal effect in each world both with linear regression and either propensity score matching or IPTW. You must present and discuss balance diagnostics and overlap plots for each of the three worlds.

1) Description of a Hypothetical Real Life Scenario

In New York City(NYC), the Department of Education(DOE) funds a free Saturday and summer academic program called DREAM-SHSI that prepares enrolled seventh graders in the NYC public school system to take the Specialized High Schools Admissions Test (SHSAT) in the eighth grade. Students must meet the following criteria to be eligible to apply for the program: be a NYC resident, be enrolled in seventh grade in a DOE public or charter school, and score a minimum of 3.2 on their grade six English Language Arts NY State test and minimum of 4.0 on their grade six Mathematics NY State test. Students must also either meet certain income requirements or live in specific districts specified by the DOE. The sample data to be generated will be of NYC eighth graders who take the SHSAT. In real life, this number is about 27500 students, of which about 2300 were in the DREAM program. For simplicity, students who are accepted into the program but do not commit to the program will not be included.

The response variable will be SHSAT composite scores. The five covariates will be: gender, ethnicity, birthplace, Grade 6 ELA State Test score, and Grade 6 Math State Test score. The research question is: does being accepted and committing to the DREAM program have an effect on a students SHSAT score?

2) Data Generating Process

1000 observations will be generated, with 100 eighth graders in the treatment group and 900 eighth graders in the control group.

Students will have 5 covariates: Gender: Schools in NYC have slightly more boys than girls, about 51/49 split in favor of boys. 0 - Female 1 - Male

Ethnicity: For simplicity, data will be simulated with 5 ethnic categories, ordered by achievement gap. 1 - Black 2 - Hispanic 3 - Other/Unknown/Unspecified 4 - White, not of Hispanic Origin 5 - Asian

Birthplace: For simplicity, data will be simulated as binary, either born in the U.S., or not. 0 - Not born in the U.S. 1 - Born in the U.S.

G6ELA: Grade 6 ELA NY State test scale score. Range: 500 - 650, median at 600

G6MATH: Grade 6 Mathematics NY State test scale score. Range: 500 - 650, median at 600

Students will have the potential outcomes of scaled SHSAT composite scores ranging theoretically between 200 and 800. The expected treatment effect should be 50, and the residual standard deviation of the composite scores should be 30. In World A, potential outcomes will assume a linear model for E[Y(0)|X] and E[Y(1)|X]. Two datasets will be generated, fullA (containing all covariates and both potential outcomes) and obsA (containing all covariates but only the observed outcome). In World B, the response surface E[Y(1)|X] will be non-linear (in this case, square root). Two datasets will be generated, fullB and obsB, similar to World A. World C will have the same response surface as World B, but the ignorability assumption will be violated. Two datasets will be generated, fullC and obsC, similar to Worlds A and B. This dataset will have an additional covariate that was missing from Worlds A and B, tutoring.

Tutoring: Binary indicator for whether or not the student receives tutoring outside of the DREAM program. 0 - Does not receive outside tutoring 1 - Does receive outside tutoring

3) Assumptions Required

To yield a valid causal estimate for the estimand of the average treatment on the treated using propensity scores, the following assumptions must be satisfied:

Ignoribility: All confounders are assumed to have been measured. The way that a propensity score is generated is that it is a single number representing all the covariates in order to draw similarities between different observations. If this assumption holds, matching observations with similar covariate values should yield an unbiased treatment effect estimate. For the data to be generated, this would mean that we assume that the covariates of gender, ethnicity, birthplace, pretest ELA score, and pretest Math score are the only confounders on the outcome, SHSAT scaled composite score.

SUTVA: Stable unit treatment values must be assumed. There should be no effect across observations, meaning that the treatment of a single observation should not effect the outcome of another in the data.

Overlap: Treatment and control groups must have sufficient overlap. If this assumption doesn't hold, inference must be restricted to only areas of overlap, or rely on another model to infer on areas outside overlap.

Balance: The predictors should be similar across treatment and control groups. If imbalanced, the analysis would have to rely on the correctness of the model used.

4) R Code for Data Generating

```r
library(arm)
library(ggplot2)
```

```r
# Generate treatment assignment and covariates.
set.seed(0)
# 100 treated, 900 control.
treatment   <- c(rep(1,100),rep(0,900))
# NYC schools are roughly 51/49 split between boys and girls.
gender      <- rbinom(1000,1,0.51)
# Probability vector for ethnicity comes from NYC demographics in 2019.
ethnic      <- c(sample(c(1:6),1000,replace=T,prob=c(0.007,0.162,0.255,0.406,0.151,0.02)))
# Probability of being foreign born, if not white.
birthplace  <- ifelse(ethnic!=5,rbinom(sum(ethnic!=4),1,0.87),1)
# Grade 6 English Language Arts score, foreign born students assumed overall lower score
g6ela       <- round(ifelse(birthplace==1,rnorm(sum(birthplace==1),600,15),rnorm(sum(birthplace==0),500
g6ela       <- ifelse(g6ela>650,650,g6ela)
# Grade 6 Mathematics score
g6math      <- round(rnorm(1000,600,15))
g6math      <- ifelse(g6math>650,650,g6math)
# Probability the student is taking tutoring outside of the treatment
tutor       <- ifelse(ethnic>=4,rbinom(sum(ethnic>=4),1,0.5),rbinom(sum(ethnic<4),1,0.1))

# Store treatment assignment and covariates
```

```
covtrt      <- data.frame(treatment=treatment,gender=gender,ethnic=ethnic,birthplace=birthplace,g6ela=g
head(covtrt)
```

```
##   treatment gender ethnic birthplace g6ela g6math
## 1         1      0      4          1   631    606
## 2         1      1      3          0   457    598
## 3         1      1      2          0   391    584
## 4         1      0      4          1   603    604
## 5         1      0      5          1   617    614
## 6         1      1      4          1   622    634
```

```
# World A, both linear models
set.seed(0)
# Girls outperform boys at all levels in middle school about 6.3%.
# Effects of ethnicity are based on achievement gaps in proficiency rates across NY State.
y_0a  <- round(-35*gender + 40*ethnic + 30*birthplace + 0.3*g6ela + 0.3*g6math + rnorm(1000, mean = 0, 
# Adding treatment effect.
y_1a  <- round(y_0a + rnorm(1000, mean = 50, sd = 1))
# Making sure no scores are above the highest possible score.
y_1a[y_1a > 800] = 800

# Quick sanity check
summary(y_0a)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   250.0   469.0   512.5   510.1   553.0   678.0
```

```
summary(y_1a)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   301.0   519.0   563.0   560.1   603.0   727.0
```

```
mean(y_1a - y_0a)
```

```
## [1] 49.974
```

```
# Create World A datasets
fullA   <- data.frame(y_0a=y_0a,y_1a=y_1a,treatment=treatment,gender=gender,ethnic=ethnic,birthplace=bi
y_a     <- ifelse(treatment==1,y_1a,y_0a)
obsA    <- data.frame(y_a=y_a,treatment=treatment,gender=gender,ethnic=ethnic,birthplace=birthplace,g6el
```

```
# World B, both non-linear models
set.seed(1)
# Girls outperform boys at all levels in middle school about 6.3%.
# Effects of ethnicity are based on achievement gaps in proficiency rates across NY State.
y_0b  <- round(-35*gender + 40*ethnic + 30*birthplace + 2*sqrt(g6ela) + 2*sqrt(g6math) + rnorm(1000, mea
# Adding treatment effect.
y_1b  <- round(y_0b + rnorm(1000, mean = 50, sd = 1))
# Making sure no scores are above the highest possible score.
y_1b[y_1b > 800] = 800

# Quick sanity check
summary(y_0b)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   320.0   463.8   504.0   501.1   538.0   656.0
```

```
summary(y_1b)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##     370     513     554     551     588     706
```

```
mean(y_1b - y_0b)
```

```
## [1] 49.981
```

```
# Create World A datasets
fullB   <- data.frame(y_0b=y_0b,y_1b=y_1b,treatment=treatment,gender=gender,ethnic=ethnic,birthplace=bi
y_b     <- ifelse(treatment==1,y_1b,y_0b)
obsB    <- data.frame(y_b=y_b,treatment=treatment,gender=gender,ethnic=ethnic,birthplace=birthplace,g6e
```

```
# World C, both non-linear models, but with another confounding covariate
set.seed(2)
# Girls outperform boys at all levels in middle school about 6.3%.
# Effects of ethnicity are based on achievement gaps in proficiency rates across NY State.
y_0c  <- round(-35*gender + 40*ethnic + 30*birthplace + 2*sqrt(g6ela) + 2*sqrt(g6math) + rnorm(1000, mea
# Adding treatment effect.
y_1c  <- round(y_0c + rnorm(1000, mean = 50, sd = 1))
# Making sure no scores are above the highest possible score.
y_1c[y_1c > 800] = 800

# Quick sanity check
summary(y_0c)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   340.0   473.0   521.0   521.1   569.0   709.0
```

```
summary(y_1c)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   391.0   523.0   571.0   571.1   619.2   757.0
```

```
mean(y_1c - y_0c)
```

```
## [1] 50.013
```

```
# Create World A datasets
fullC   <- data.frame(y_0c=y_0c,y_1c=y_1c,treatment=treatment,gender=gender,ethnic=ethnic,birthplace=bi
y_c     <- ifelse(treatment==1,y_1c,y_0c)
obsC    <- data.frame(y_c=y_c,treatment=treatment,gender=gender,ethnic=ethnic,birthplace=birthplace,g6e
```

5) Methods and Estimand

Two methods will be used to find the estimand, the average treatment on the treated (ATT): Linear regression and propensity score matching.

Linear Regression: - Fit linear model onto the covariates and call the estimate for the treatment effect.

Propensity Score Matching: - Select covariates to be included in the model. - Use logistic regression to obtained propensity scores. - Restructure the data using k-1 with replacement weights - Match with replacement using one-to-one nearest neighbor matching on propensity scores. Using the arm package, call the "cnts" variable in the output to create a weight variable. - Apply the IPTW to re-weight the control group to look closer to the treatment group. - Check overlap and balance, and rebalance if needed. Overlap checking ensures sufficient common support between treatment and control group, so that a match can be made between control and treatment such that they can be treated as counterfactuals of each other. Balancing is necessary to ensure no variables are more biased than others. - Estimate average treatment effect on the weighted and balanced dataset.

```
# Method 1: Linear regression with all covariates
lm.a <- lm(y_a ~ ., data = obsA)
summary(lm.a)
```

```
##
## Call:
## lm(formula = y_a ~ ., data = obsA)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -93.683 -20.644  -1.155  20.742 102.296
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  12.70044   39.22457   0.324 0.746167
## treatment    51.56690    3.15060  16.367  < 2e-16 ***
## gender      -32.59085    1.89578 -17.191  < 2e-16 ***
## ethnic       39.86731    0.93780  42.511  < 2e-16 ***
## birthplace   18.60776    4.29071   4.337 1.59e-05 ***
## g6ela         0.36053    0.02625  13.734  < 2e-16 ***
## g6math        0.23422    0.06115   3.830 0.000136 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29.85 on 993 degrees of freedom
## Multiple R-squared:  0.7682, Adjusted R-squared:  0.7668
## F-statistic: 548.6 on 6 and 993 DF,  p-value: < 2.2e-16
```

```
lm.b <- lm(y_b ~ ., data = obsB)
summary(lm.b)
```

```
##
## Call:
## lm(formula = y_b ~ ., data = obsB)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -87.08 -20.14  -0.67  20.52 114.05
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 304.618260  40.777279   7.470 1.75e-13 ***
## treatment    54.043834   3.275316  16.500  < 2e-16 ***
## gender      -33.815422   1.970825 -17.158  < 2e-16 ***
## ethnic       38.762565   0.974928  39.759  < 2e-16 ***
## birthplace   26.803922   4.460557   6.009 2.62e-09 ***
## g6ela         0.087906   0.027290   3.221  0.00132 **
## g6math       -0.004728   0.063569  -0.074  0.94073
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 31.03 on 993 degrees of freedom
## Multiple R-squared:  0.715,  Adjusted R-squared:  0.7133
## F-statistic: 415.2 on 6 and 993 DF,  p-value: < 2.2e-16
```

```
lm.c <- lm(y_c ~ .-tutor, data = obsC)
summary(lm.c)
```

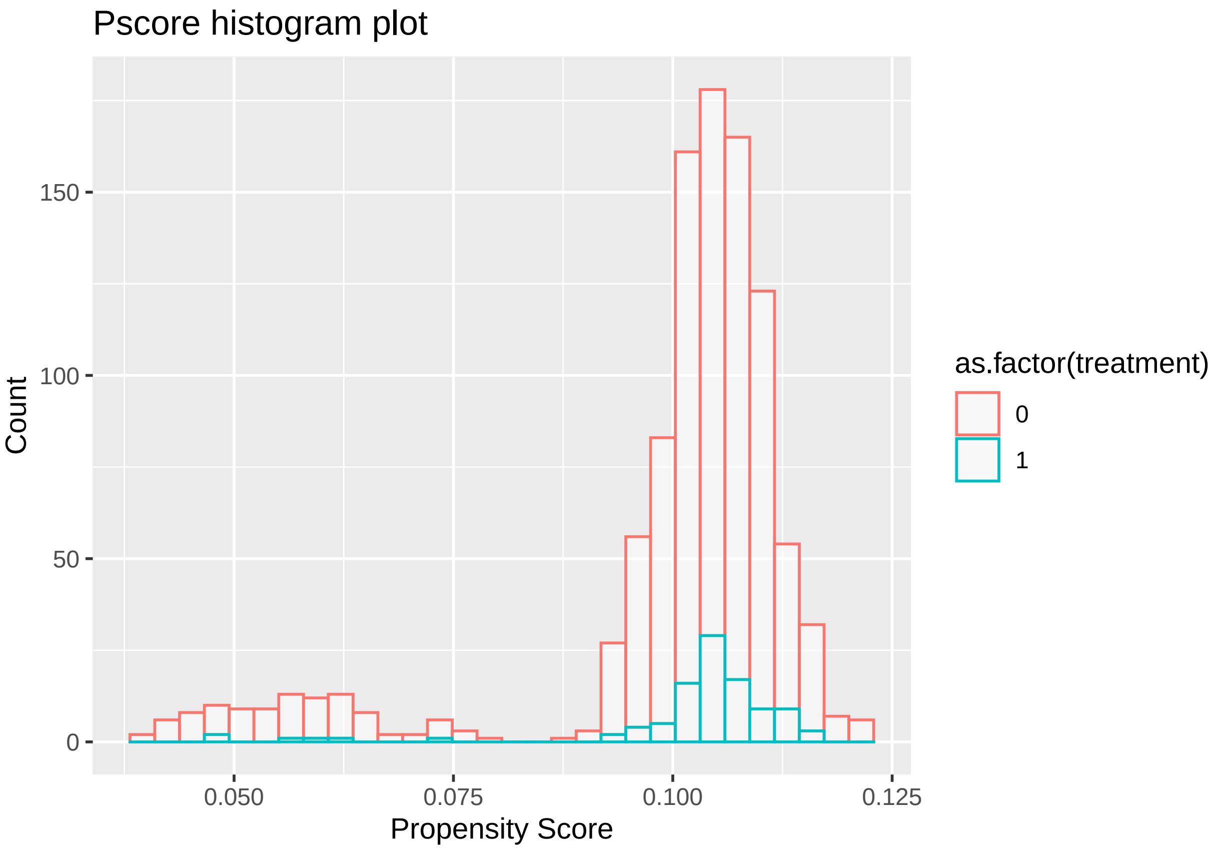
```
##
## Call:
## lm(formula = y_c ~ . - tutor, data = obsC)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -105.080  -25.297   -1.242   24.822  126.584
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 299.60159   48.06330   6.233 6.73e-10 ***
## treatment    48.18369    3.86054  12.481  < 2e-16 ***
## gender      -36.28641    2.32297 -15.621  < 2e-16 ***
## ethnic       47.52151    1.14913  41.354  < 2e-16 ***
## birthplace   26.26569    5.25756   4.996 6.92e-07 ***
## g6ela         0.06097    0.03217   1.895   0.0583 .
## g6math        0.01413    0.07493   0.189   0.8505
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 36.57 on 993 degrees of freedom
## Multiple R-squared:  0.7038, Adjusted R-squared:  0.702
## F-statistic: 393.3 on 6 and 993 DF,  p-value: < 2.2e-16
```

```
# Method 2: Propensity Score Matching
# Logistic regression and k-1 with replacement matching
glm.1           <- glm(treatment ~ ., data = covtrt, family = binomial(link = "logit"))
pscores         <- data.frame(pscore = predict(glm.1, type = "response"), treatment = glm.1$model$treat
match           <- matching(z = covtrt$treatment, score = pscores$pscore, replace = T)
weights         <- data.frame(treatment = covtrt$treatment, weight = NA)
weights$weight  <- ifelse(weights$treatment==1,1,match$cnts)
# Quick sanity check
table(weights)
```

```
##          weight
## treatment   0   1   2   4
##         0 808  86   5   1
##         1   0 100   0   0
```

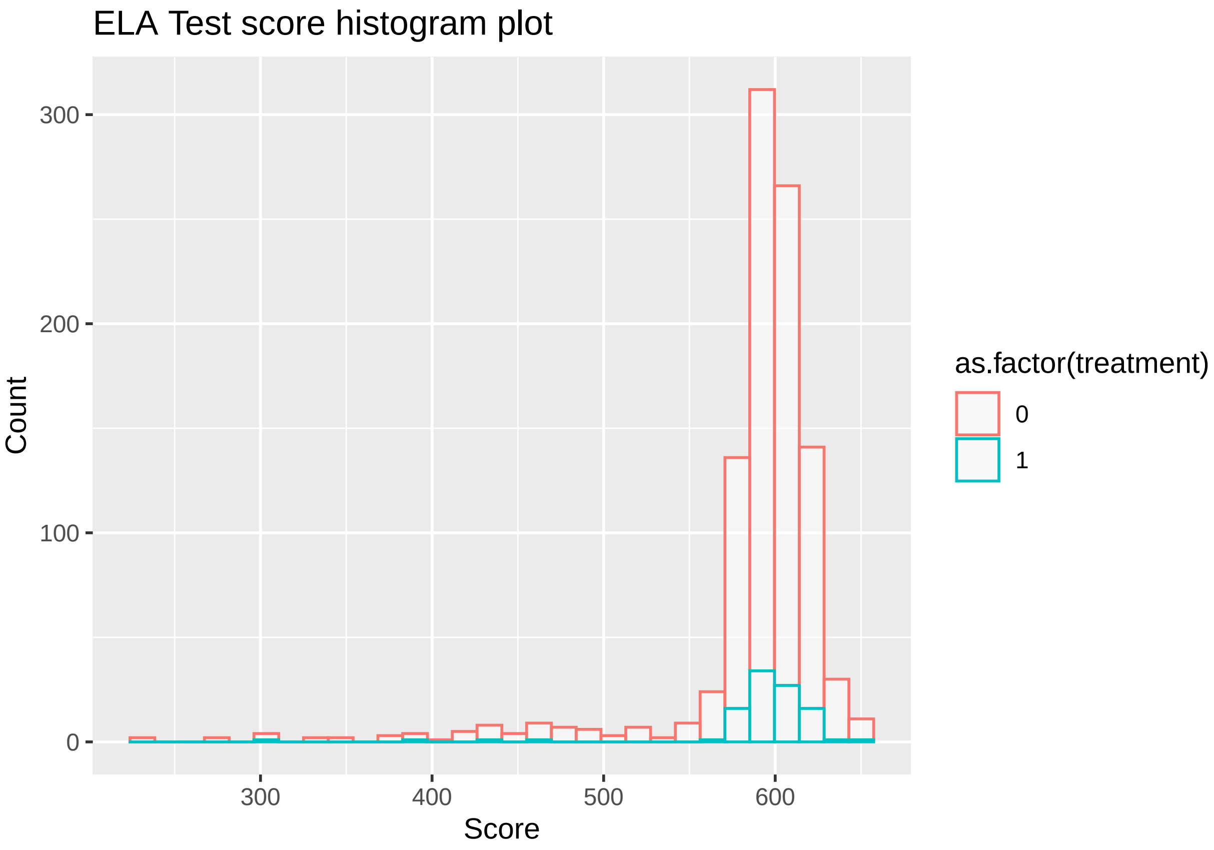
```
# Checking overlap
# Propensity scores
ggplot(pscores,aes(x=pscore,colour=as.factor(treatment))) +
 geom_histogram(fill="white",alpha=0.5) +
  labs(title="Pscore histogram plot",x="Propensity Score", y = "Count")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



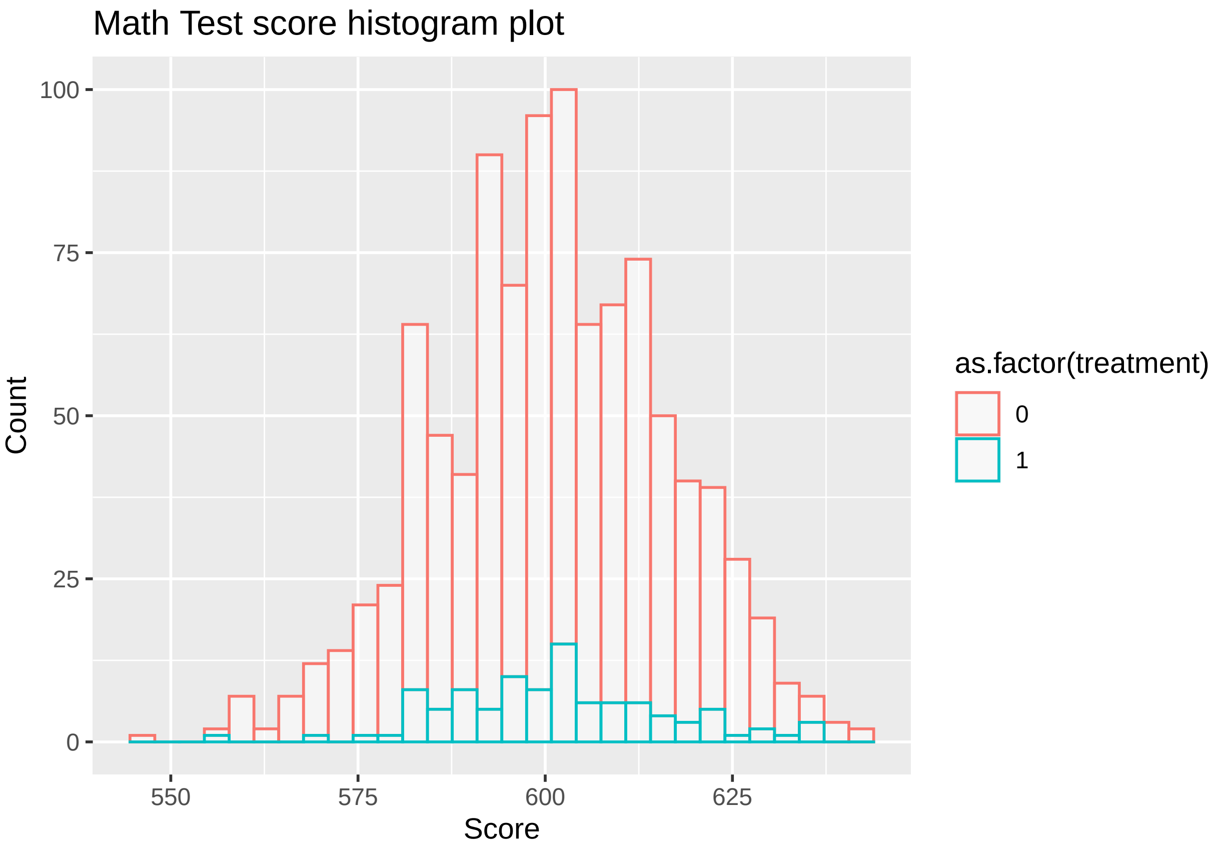
```
# Grade 6 ELA
ggplot(covtrt,aes(x=g6ela,colour=as.factor(treatment))) +
 geom_histogram(fill="white",alpha=0.5) +
  labs(title="ELA Test score histogram plot",x="Score", y = "Count")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
# Grade 6 MATH
ggplot(covtrt,aes(x=g6math,colour=as.factor(treatment))) +
 geom_histogram(fill="white",alpha=0.5) +
  labs(title="Math Test score histogram plot",x="Score", y = "Count")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Overlap looks good; no areas of concern can be seen.

```
# Check balance
# Create function
checkbalance <- function(data, covariates, weights){
  treated <- data[data$treatment == 1, ]
  control <- data[data$treatment == 0, ]

  treatedwts<- weights[weights$treatment == 1, "weight"]
  controlwts<- weights[weights$treatment == 0, "weight"]

  # Make binary indicator on whether or not the covariate is binary.
  bivar <- ifelse(sapply(covariates, function(x) length(unique(data[, x]))) == 2, 1, 0)
  weightedsd <- function(x, w) sqrt(sum(w*(x - weighted.mean(x, w))^2)/sum(w))

  # Find means
  mean1 <- sapply(covariates, function(x) mean(treated[, x]))
  mean0 <- sapply(covariates, function(x) mean(control[, x]))

  mean1.m <- sapply(covariates, function(x) weighted.mean(treated[, x], treatedwts))
  mean0.m <- sapply(covariates, function(x) weighted.mean(control[, x], controlwts))

  # Find differences
  diff <- ifelse(bivar == 1, mean1-mean0, (mean1-mean0)/sapply(covariates, function(x) sd(treated[, x])
  diff.m <- ifelse(bivar == 1, mean1.m-mean0.m, (mean1.m-mean0.m)/sapply(covariates, function(x) sd(trea

  # Find ratios
  ratio <- ifelse(bivar == 0, sapply(covariates, function(x) sd(control[, x])) / sapply(covariates, fun
  ratio.m <- ifelse(bivar == 0, sapply(covariates, function(x) weightedsd(control[, x], controlwts)) /

  # Construct final data frame output
  data.frame(mean1, mean0, mean1.m,mean0.m, diff, diff.m, ratio, ratio.m)
}
# Define covariates
covnames <- c("gender", "ethnic","birthplace", "g6ela", "g6math")
# Use function and round the numbers.
balance <- round(checkbalance(obsA, covnames, weights), 2)
balance
```

```
##             mean1  mean0 mean1.m mean0.m  diff diff.m ratio ratio.m
## gender       0.52   0.53    0.52    0.50 -0.01   0.02    NA      NA
## ethnic       3.67   3.64    3.67    3.75  0.03  -0.09  1.11    1.02
## birthplace   0.94   0.89    0.94    0.94  0.05   0.00    NA      NA
## g6ela      591.67 587.41  591.67  591.35  0.10   0.01  1.14    0.90
## g6math     601.47 600.75  601.47  600.36  0.05   0.07  1.04    0.87
```

Means are close to 0, ratios are close to 1, balance seems good.

```
# Estimate ATT
reconstA <- cbind(obsA, pscores = pscores$pscore, weight = weights$weight)
reconstB <- cbind(obsB, pscores = pscores$pscore, weight = weights$weight)
reconstC <- cbind(obsC, pscores = pscores$pscore, weight = weights$weight)
estA <- lm(y_a ~ treatment + weight, data = reconstA)
estB <- lm(y_b ~ treatment + weight, data = reconstB)
estC <- lm(y_c ~ treatment + weight, data = reconstC)
summary(estA)
```

```
##
## Call:
## lm(formula = y_a ~ treatment + weight, data = reconstA)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -258.463  -40.463    1.537   42.573  169.537
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  508.463      2.080 244.503  < 2e-16 ***
## treatment     47.712      8.032   5.940 3.93e-09 ***
## weight         9.145      5.648   1.619    0.106
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 59.48 on 997 degrees of freedom
## Multiple R-squared:  0.07595,    Adjusted R-squared:  0.0741
## F-statistic: 40.98 on 2 and 997 DF,  p-value: < 2.2e-16
```

```
summary(estB)
```

```
##
## Call:
## lm(formula = y_b ~ treatment + weight, data = reconstB)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max 
## -179.713  -37.713    3.287   37.287  156.287 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  499.713      1.935 258.208  < 2e-16 ***
## treatment     52.473      7.475   7.020  4.1e-12 ***
## weight         5.514      5.256   1.049    0.294    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 55.35 on 997 degrees of freedom
## Multiple R-squared:  0.08933,    Adjusted R-squared:  0.0875 
## F-statistic:  48.9 on 2 and 997 DF,  p-value: < 2.2e-16
```

```
summary(estC)
```

```
## 
## Call:
## lm(formula = y_c ~ treatment + weight, data = reconstC)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -180.11  -49.11    0.42   48.88  188.89 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  520.112      2.280 228.148  < 2e-16 ***
## treatment     45.453      8.805   5.162 2.95e-07 ***
## weight         7.015      6.192   1.133    0.258    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 65.2 on 997 degrees of freedom
## Multiple R-squared:  0.05484,    Adjusted R-squared:  0.05294 
## F-statistic: 28.92 on 2 and 997 DF,  p-value: 6.171e-13
```

6) Results

The ATT and corresponding standard error are shown in the printed data frame:

```
result <- data.frame(World = rep(c("A", "B", "C"), each = 3),
                     Method = rep(c("Difference in Means", "Linear", "Logit and K1"), 3),
                     TreatmentEffect = NA,
                     StandardDeviation = NA)
# unmatched estimates
result$TreatmentEffect[1] = diff(tapply(obsA$y_a, obsA$treatment, mean))
result$TreatmentEffect[4] = diff(tapply(obsB$y_b, obsB$treatment, mean))
result$TreatmentEffect[7] = diff(tapply(obsC$y_c, obsC$treatment, mean))
# linear estimates and standard error
result$TreatmentEffect[2] = coef(lm.a)["treatment"]
result$TreatmentEffect[5] = coef(lm.b)["treatment"]
result$TreatmentEffect[8] = coef(lm.c)["treatment"]
result$StandardDeviation[2] = summary(lm.a)$coefficients["treatment","Std. Error"]
result$StandardDeviation[5] = summary(lm.b)$coefficients["treatment","Std. Error"]
```

```r
result$StandardDeviation[8] = summary(lm.c)$coefficients["treatment","Std. Error"]
# logit regression and k1 weights
result$TreatmentEffect[3] = coef(estA)["treatment"]
result$TreatmentEffect[6] = coef(estB)["treatment"]
result$TreatmentEffect[9] = coef(estC)["treatment"]
result$StandardDeviation[3] = summary(estA)$coefficients["treatment","Std. Error"]
result$StandardDeviation[6] = summary(estB)$coefficients["treatment","Std. Error"]
result$StandardDeviation[9] = summary(estC)$coefficients["treatment","Std. Error"]
print(result)
```

```
##   World              Method TreatmentEffect StandardDeviation
## 1     A Difference in Means        55.84111                NA
## 2     A              Linear        51.56690          3.150599
## 3     A        Logit and K1        47.71196          8.032126
## 4     B Difference in Means        57.37444                NA
## 5     B              Linear        54.04383          3.275316
## 6     B        Logit and K1        52.47309          7.474931
## 7     C Difference in Means        51.68889                NA
## 8     C              Linear        48.18369          3.860545
## 9     C        Logit and K1        45.45331          8.805123
```

In World A, the outcomes are simulated from a linear model, so the estimated treatment effect is the closest to the true treatment effect out of the three worlds. In World B, the outcomes are not simnulated from a linear model, and linear regression is a less ideal choice than it would be for World A. In World C, the estimate is also not as close to the true treatment effect, as the model is missing a key covariate, violating the assumption of ignorability.

7) Discuss the Bias

The method of linear regression is only reliable if the relationship between the outcome and all the covariates are linear. Otherwise, using linear regression to estimate a causal effect will result in extrapolating a relationship over areas where there is no data. This will happen even if the data are linear, but the bias increases dramatically if the relationship is non-linear. Propensity scores, however, reduce bias by reconstructing the samples in such a way that the effects of bias due to the covariates are balanced across the control and treatment groups via matching. In this case, there is sufficient overlap between the groups that the matching is reliable without needing to restrict the space. As the case in World C, regardless of whichever method is used, the validity of ignorability is important. Missing a covariate, or several, in the model can produce estimates of ATT that are much farther from the true effect.

8) Conclusion

The usage of linear regression to estimate causal effects is extremely limited, and relies on a variety of factors to make the estimate more accurate. In the real world, these factors rarely line up and even more rarely have easy relationships with all the covariates. Intersectionality and confounding variables are the norm, and as such, linear regression should be used with extreme caution.

Robust methods like propensity scores attempt to account for these infinite confounding relationships by using matching to pair similar observations as reliable counterfactuals of each other, presenting the opportunity to reconstruct a theoretical dataset that is capable of estimating causal effects.