

# SOC-GA 2332 Intro to Stats Lab 13

Risa Gelles-Watnick

11/26/2025

## Part 0: Logistics

- **Problem set 4** due on Dec. 5th, 11:59pm
  - Treat `numsibling` as fixed within household even though it doesn't make a ton of sense
- **Replication Part II** due Dec. 15th, 5:00pm (no extensions)
  - Grades for Part I coming shortly

## Replication project tips

- Knit consistently and frequently as you write code
  - If you do hit a knitting error:
    - \* Start commenting things out until you can get it to run and then work backwards
    - \* Delete everything from your environment (broom) and run your code from the beginning
- If you're hitting space issues on your computer, feel free to use a smaller sample (ex. 1%)
- Papers should be formatted as clean, final projects
  - Code chunks not showing
  - Text formatted correctly and consistently
  - Figures and tables numbered & labelled, with all text clearly readable
  - Utilize the [template on GitHub](#) or [RMarkdown cheatsheets](#)
- Read instructions completely and carefully
- Comment up your code (both for me and for your future self!)
- Get started early so you can take advantage of your resources (office hours, class, etc.)

## Part 1: Exploring data sample

- Today we are going to review a specific type of weighting, Inverse Probability of Treatment Weighting (IPTW). We will use the Early Childhood Longitudinal Study dataset.
- We will examine the effect of going to a Catholic school (`catholic = 1`), as opposed to a public school (`catholic = 0`), on students' standardized math score (`c5r2mtsc_std`). The pre-treatment covariates are:
  - `race_white`: Is the student white (1) or not (0)?
  - `p5hmage`: Mother's age
  - `w3income`: Family income
  - `p5numpla`: Number of places the student has lived for at least 4 months
  - `w3momed_hsb`: Is the mother's education level high-school or below (1) or some college or more (0)?

```
## import data
ecls <- read.csv("data/ecls.csv")

## covariates variable name vector
ecls_cov <- c('race_white', 'p5himage', 'w3income', 'p5numpla', 'w3momed_hsb')

## remove observations with NAs
ecls_nomiss <- ecls %>%
  select(c5r2mtsc_std, catholic, all_of(ecls_cov)) %>%
  na.omit()
```

## Explore sample

- First let's see if there is any apparent difference in outcome by school type without controlling for any covariates

```
## check difference in mean outcomes by school type
ecls %>%
  group_by(catholic) %>%
  summarise(n_students = n(),
            mean_math = mean(c5r2mtsc_std),
            std_error_math = sd(c5r2mtsc_std) / sqrt(n_students))
```

```
## # A tibble: 2 x 4
##   catholic n_students mean_math std_error_math
##   <int>      <int>      <dbl>      <dbl>
## 1         0       9568     -0.0306      0.0104
## 2         1       1510      0.194      0.0224
```

```
## two Sample t-test: (H0: mean math scores do not differ by school types)
with(ecls, t.test(c5r2mtsc_std ~ catholic))
```

```
##
## Welch Two Sample t-test
##
## data: c5r2mtsc_std by catholic
## t = -9.1069, df = 2214.5, p-value < 0.00000000000000022
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
## -0.2727988 -0.1761292
## sample estimates:
## mean in group 0 mean in group 1
## -0.03059583 0.19386817
```

- *Question:* How would we interpret this t-test?
- Now let's explore the relationship between school type and other covariates.
- *Question:* What is endogenous sampling? When selection into the treatment group is not random (i.e. varies by some other factor)
- To check if we might be endogenous sampling, we first examine the difference in means by treatment status for covariates.

```
## check difference in means for pre-treatment covariates by school types
## summarise group means for covariates
ecls %>%
  group_by(catholic) %>%
```

```

select(one_of(ecls_cov)) %>%
summarise_all(funs(mean(., na.rm = T)))

## # A tibble: 2 x 6
##   catholic race_white p5hmage w3income p5numpla w3momed_hsb
##   <int>      <dbl>   <dbl>   <dbl>   <dbl>      <dbl>
## 1      0      0.556    37.6   54889.    1.13      0.464
## 2      1      0.725    39.6   82074.    1.09      0.227

## Two sample t-test for every covariate
## lapply: a build-in loop that apply the t-test function along the name vector
lapply(ecls_cov, function(v){
  t.test(ecls[, v] ~ ecls[, 'catholic'])
})

## [[1]]
##
## Welch Two Sample t-test
##
## data:  ecls[, v] by ecls[, "catholic"]
## t = -13.453, df = 2143.3, p-value < 0.00000000000000022
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -0.1936817 -0.1444003
## sample estimates:
## mean in group 0 mean in group 1
##    0.5561246      0.7251656
##
##
## [[2]]
##
## Welch Two Sample t-test
##
## data:  ecls[, v] by ecls[, "catholic"]
## t = -12.665, df = 2186.9, p-value < 0.00000000000000022
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -2.326071 -1.702317
## sample estimates:
## mean in group 0 mean in group 1
##    37.56097     39.57516
##
##
## [[3]]
##
## Welch Two Sample t-test
##
## data:  ecls[, v] by ecls[, "catholic"]
## t = -20.25, df = 1825.1, p-value < 0.00000000000000022
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -29818.10 -24552.18
## sample estimates:
## mean in group 0 mean in group 1
##    54889.16     82074.30

```

```
##
##
## [[4]]
##
## Welch Two Sample t-test
##
## data:  ecl[, v] by ecl[, "catholic"]
## t = 4.2458, df = 2233.7, p-value = 0.00002267
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  0.02150833 0.05842896
## sample estimates:
## mean in group 0 mean in group 1
##      1.132669      1.092701
##
##
## [[5]]
##
## Welch Two Sample t-test
##
## data:  ecl[, v] by ecl[, "catholic"]
## t = 18.855, df = 2107.3, p-value < 0.00000000000000022
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  0.2122471 0.2615226
## sample estimates:
## mean in group 0 mean in group 1
##      0.4640918      0.2272069
```

- As we can see, the difference in mean for covariates are statistically significant.

## Part 2: Propensity Score

- *Question:* What is Inverse Probability of Treatment Weighting (IPTW)? It's weighting the sample by the inverse probability of being selected into the treatment group, given a person's other characteristics.
- *Question:* Why might we use IPTW? Individuals who were assigned to the treatment group even though they were much more likely to be assigned to the control group are a rare, and valuable counterfactual, so we want to count them more.
- To conduct IPTW, we usually use propensity scores.
- Propensity scores are a one-number summary of all the different covariates' values for each individual. In this case, we'll use them to represent the probability of being treated given a set of pre-treatment covariates.
- In R, we can estimate propensity score given the covariates by fitting a logistic regression with the treatment status as the outcome and covariates as predictors. So basically we're trying to predict what each observation's probability of being treated is.
  - We still leverage the strong ignorability assumption and correct specification assumption to derive an unbiased estimate of the true propensity score.
    - \* *Question:* What are these two assumptions? Strong ignorability: no unobserved confounders (i.e., no variables are affecting treatment assignment that we can't include in the propensity score model). Correct specification: we specify the correct formula for our propensity score model.

- Now let's try to fit propensity scores to our data. In reality this is usually a much longer and more iterative process, but for this demonstration I've simplified it into fitting a simple logistic regression. There are also ML algorithms that will find the best propensity score formula for you (ex. using SuperLearner through the `WeightIt` package).

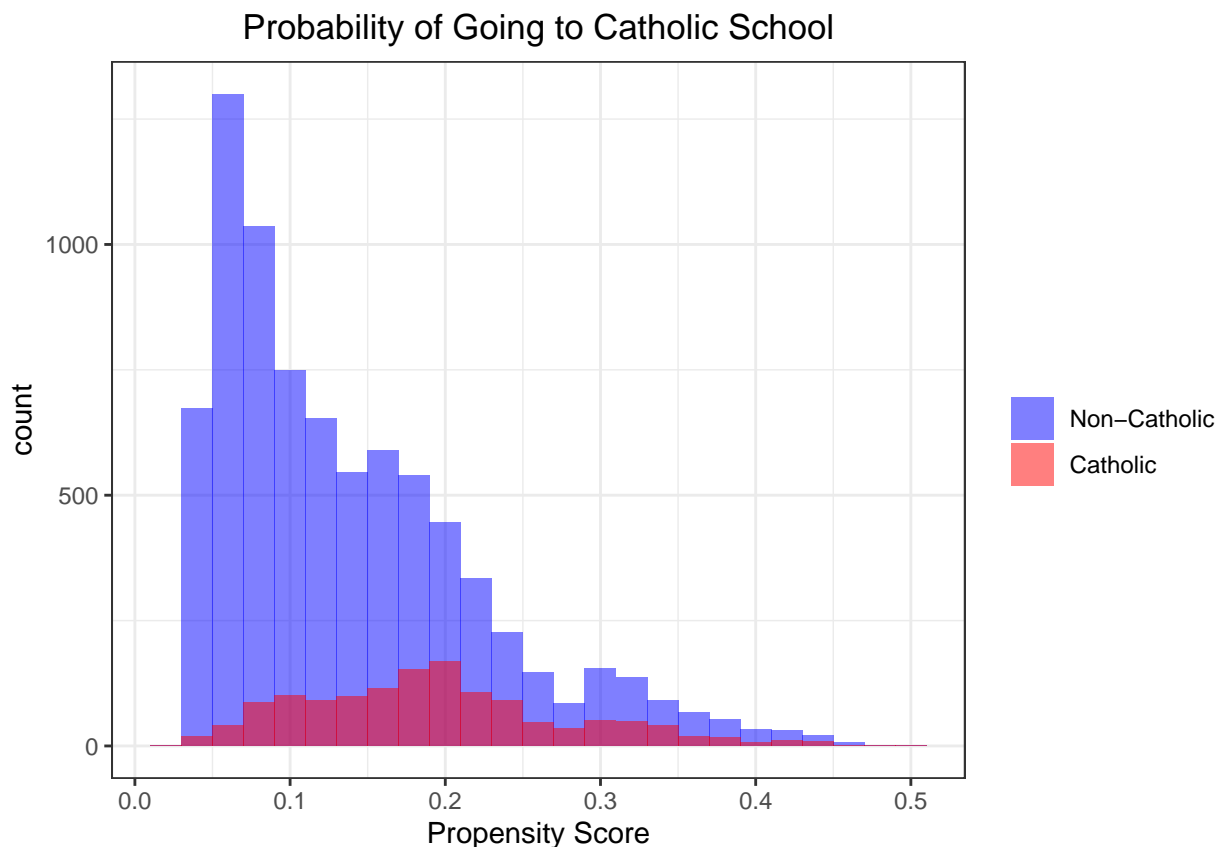
```
## rescale income
ecls <- ecls %>% mutate(w3income_1k = w3income/1000)

## fit a logistic regress to generate propensity score using covariates
m_ps <- glm(catholic ~ race_white + w3income_1k + p5hmage + p5numpla + w3momed_hsb,
            family = binomial(), data = ecls)
summary(m_ps)

##
## Call:
## glm(formula = catholic ~ race_white + w3income_1k + p5hmage +
##      p5numpla + w3momed_hsb, family = binomial(), data = ecls)
##
## Coefficients:
##              Estimate Std. Error z value      Pr(>|z|)
## (Intercept) -3.2125519   0.2379826 -13.499 < 0.0000000000000002 ***
## race_white   0.3145014   0.0700895   4.487   0.00000721848 ***
## w3income_1k  0.0073038   0.0006495  11.245 < 0.0000000000000002 ***
## p5hmage      0.0292168   0.0050771   5.755   0.00000000869 ***
## p5numpla     -0.1439392   0.0912255  -1.578     0.115
## w3momed_hsb -0.6935868   0.0743207  -9.332 < 0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 7701.3  on 9266  degrees of freedom
## Residual deviance: 7168.8  on 9261  degrees of freedom
## (1811 observations deleted due to missingness)
## AIC: 7180.8
##
## Number of Fisher Scoring iterations: 5

## use above model to generate propensity
## (the probability of being treated given a set of pre-treatment covariates)
prs_df <- data.frame(pr_score = predict(m_ps, type = "response"),
                    catholic = m_ps$model$catholic)

## check the region of common support
## in every unit in the treatment, is there a control unit
prs_df %>%
  ggplot(aes(x = pr_score, fill = as.factor(catholic))) +
  geom_histogram(binwidth = 0.02, alpha = 0.5, position="identity") +
  ggtitle("Probability of Going to Catholic School") +
  xlab("Propensity Score") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_fill_manual(name=" ",
                   values = c("blue", "red"),
                   labels = c("Non-Catholic", "Catholic"))
```



## Part 3: Inverse probability weighting

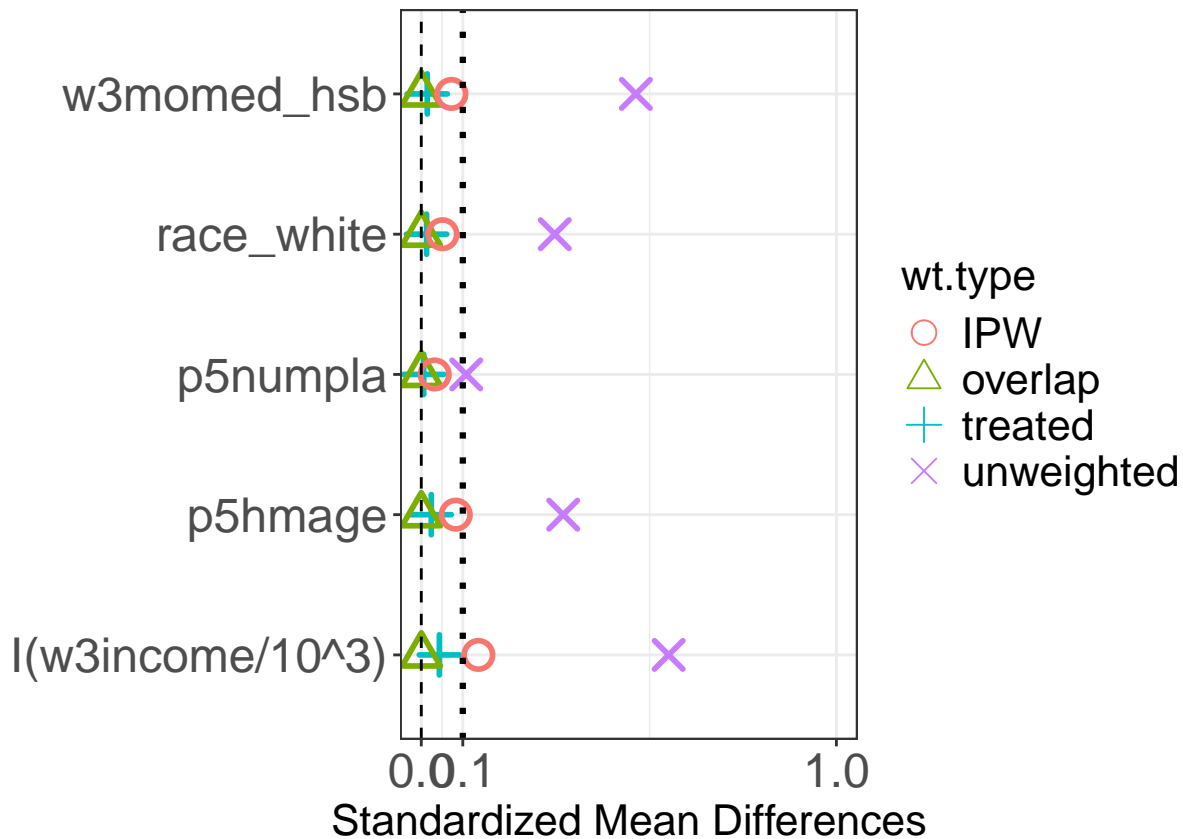
- Use propensity scores to weight units based on their probabilities of being treated.
  - $e(x)$  is the propensity score
  - $w_1$  is weights if the unit is treated
  - $w_0$  is weights if the unit is untreated
- There are several options that measure different quantities of interest
  - IPW measures ATE.  $weight(w_1, w_0) = (\frac{1}{e(x)}, \frac{1}{1-e(x)})$ 
    - \* The core of IPW is to create a weighted treatment group and a weighted control group whose covariates' distribution resembles the distribution of the whole sample
  - **treated** measures ATT.  $weight(w_1, w_0) = (1, \frac{e(x)}{1-e(x)})$ 
    - \* The core of **treated** is to create only a weighted control group whose covariates' distribution resembles the distribution of the unweighted treatment group
  - **overlap** measures ATO (Average Treatment on Overlap).  $weight(w_1, w_0) = (1 - e(x), e(x))$ 
    - \* The core of **overlap** is to give more weights to the observations near the center of the propensity, or the units under “equipoise”
- You may check [this site](#) for more detailed explanations

```
## matching algorithm
ps.formula <- catholic ~ race_white + p5hmage +
  I(w3income / 10^3) + p5numpla + w3momed_hsb

bal.ipw <- SumStat(ps.formula = ps.formula, zname = "catholic",
weight = c("treated", "overlap", "IPW"), data = ecl_s_nomiss)

# check balance by weight type
```

```
plot(bal.ipw)
```



- Estimate ATT, ATO, and ATE:

```
## average treatment effect among the treated population
att <- PSweight(ps.formula = ps.formula, yname = "c5r2mtsc_std", data = eclis_nomiss, family = "gaussian",
weight = "treated")

## average treatment effect among the overlap population
ato <- PSweight(ps.formula = ps.formula, yname = "c5r2mtsc_std", data = eclis_nomiss, family = "gaussian",
weight = "overlap")

## average treatment effect using IPW
ate <- PSweight(ps.formula = ps.formula, yname = "c5r2mtsc_std", data = eclis_nomiss, family = "gaussian",
weight = "IPW")

## check results
summary(att)
```

```
##
## Closed-form inference:
##
## Original group value: 0, 1
## Treatment group value: 1
##
## Contrast:
##           0 1
## Contrast 1 -1 1
```

```
##
##           Estimate Std.Error      lwr      upr    Pr(>|z|)
## Contrast 1 -0.12161    0.02595 -0.17247 -0.070751 0.000002781 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(ate)
```

```
##
## Closed-form inference:
##
## Original group value:  0, 1
##
## Contrast:
##           0 1
## Contrast 1 -1 1
##
##           Estimate Std.Error      lwr      upr Pr(>|z|)
## Contrast 1 -0.0087096  0.0290430 -0.0656329 0.048214  0.7643
```

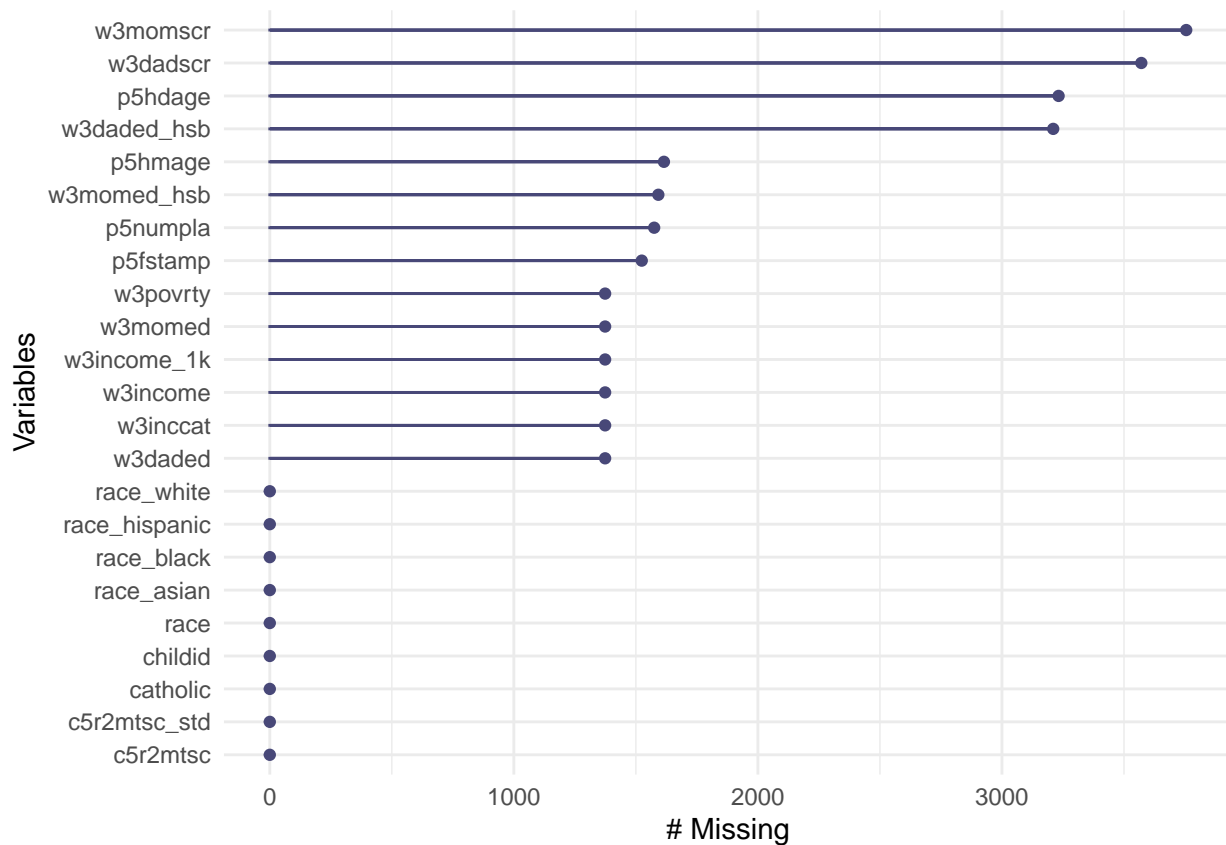
- *Question:* Why is ATE and ATT different? Looking at different groups. ATE changes both treatment and control group to make covariates equivalent whereas ATT makes control group covariates mimic treatment group without changing the treatment group.

## Exercise

Instead of removing observations with NAs, use one of the missing data strategies we covered in previous labs (see Lab 7). Then, run a regression to see whether school type affects math score. Do you get the same results as when you use IPTW? Why or why not?

```
# checking missingness
gg_miss_var(ecls)
```





```
# running multiple imputation
a.out <- eclis %>%
  select(catholic, c5r2mtsc_std, race_white, p5hmage, w3income, p5numpla, w3momed_hsb) %>% # selecting
  amelia(., # telling amelia to use every variable in our dataset for imputation
    m = 5 # number of times to impute/number of datasets to create
  )
```

```
## -- Imputation 1 --
##
## 1 2 3 4 5
##
## -- Imputation 2 --
##
## 1 2 3 4 5
##
## -- Imputation 3 --
##
## 1 2 3 4
##
## -- Imputation 4 --
##
## 1 2 3 4
##
## -- Imputation 5 --
##
## 1 2 3 4
```

```

# setting seed since multiple imputation involves randomness
set.seed(3636)

# running our regression model with each data set
mult_imp <- with(
  a.out,
  lm(c5r2mtsc_std ~ catholic + race_white + p5hmage +
      I(w3income / 10^3) + p5numpla + w3momed_hsb)
)

# combining results
mar_mi <- mi.combine(mult_imp, conf.int = TRUE)

# showing results
mar_mi

## # A tibble: 7 x 10
##   term      estimate std.error statistic  p.value    df      r miss.info conf.low
##   <chr>      <dbl>    <dbl>    <dbl>    <dbl>   <dbl> <dbl>    <dbl>    <dbl>
## 1 (Inter~ -0.515    0.0701     -7.35 2.00e+ 0 9.01e1 0.267    0.228   -0.376
## 2 cathol~ -0.0797    0.0262     -3.04 2.00e+ 0 2.78e3 0.0394    0.0386  -0.0282
## 3 race_w~  0.352     0.0185     19.0 2.37e-79 1.12e4 0.0192    0.0190   0.389
## 4 p5hmage  0.00781    0.00160      4.89 7.51e- 6 6.22e1 0.340    0.277    0.0110
## 5 I(w3in~  0.00493    0.000230    21.5 1.64e-73 5.17e2 0.0965    0.0915   0.00538
## 6 p5nump~ -0.0889     0.0237     -3.75 2.00e+ 0 2.57e2 0.142    0.131   -0.0423
## 7 w3mome~ -0.352     0.0224    -15.7 2.00e+ 0 5.30e1 0.379    0.301   -0.307
## # i 1 more variable: conf.high <dbl>

```

We don't get the exact same results, but it's in the same direction and not statistically significant. We don't get exactly the same results because we haven't done away with the base issue, which is that some people are more likely to select into the treatment group than others.

```

## check difference in means for pre-treatment covariates by school types
## summarise group means for covariates
a.out[[1]]$imp1 %>%
  as.tibble() %>%
  group_by(catholic) %>%
  select(one_of(ecls_cov)) %>%
  summarise_all(funs(mean(., na.rm = T)))

```

```

## # A tibble: 2 x 6
##   catholic race_white p5hmage w3income p5numpla w3momed_hsb
##   <int>      <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1      0      0.556    37.4    53632.    1.14     0.471
## 2      1      0.725    39.5    81478.    1.09     0.231

```

```

## Two sample t-test for every covariate
## lapply: a build-in loop that apply the t-test function along the name vector
lapply(ecls_cov, function(v){
  t.test(ecls[, v] ~ ecls[, 'catholic'])
})

```

```

## [[1]]
##
## Welch Two Sample t-test
##

```

```

## data:  ecls[, v] by ecls[, "catholic"]
## t = -13.453, df = 2143.3, p-value < 0.00000000000000022
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -0.1936817 -0.1444003
## sample estimates:
## mean in group 0 mean in group 1
##      0.5561246      0.7251656
##
##
## [[2]]
##
## Welch Two Sample t-test
##
## data:  ecls[, v] by ecls[, "catholic"]
## t = -12.665, df = 2186.9, p-value < 0.00000000000000022
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -2.326071 -1.702317
## sample estimates:
## mean in group 0 mean in group 1
##      37.56097      39.57516
##
##
## [[3]]
##
## Welch Two Sample t-test
##
## data:  ecls[, v] by ecls[, "catholic"]
## t = -20.25, df = 1825.1, p-value < 0.00000000000000022
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -29818.10 -24552.18
## sample estimates:
## mean in group 0 mean in group 1
##      54889.16      82074.30
##
##
## [[4]]
##
## Welch Two Sample t-test
##
## data:  ecls[, v] by ecls[, "catholic"]
## t = 4.2458, df = 2233.7, p-value = 0.00002267
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##   0.02150833 0.05842896
## sample estimates:
## mean in group 0 mean in group 1
##      1.132669      1.092701
##
##
## [[5]]
##

```

```
## Welch Two Sample t-test
##
## data:  ecl[, v] by ecl[, "catholic"]
## t = 18.855, df = 2107.3, p-value < 0.00000000000000022
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  0.2122471 0.2615226
## sample estimates:
## mean in group 0 mean in group 1
##      0.4640918      0.2272069
```