0211_Shuxi Chen_Problem Set 2

2025-02-11

Packages Used:

```
pacman::p_load(tidyverse,gt,foreign,knitr,webshot2,
               reshape2,viridis,ggthemes,stargazer,texreg,
               sandwich,modelsummary,Rmisc,neatStats,gmodels, dplyr, tidyr)
```

Experimental data

Load and Inspect the Data

```
df_exp <- read.dta("nsw_exper.dta")
```

check NA

```
table(is.na(df_exp$column_name))
```

```
## < table of extent 0 >
```

```

treated <- df_exp$re78[df_exp$nsw == 1]
control <- df_exp$re78[df_exp$nsw == 0]

# mean
mean_T <- mean(treated, na.rm = TRUE)
mean_C <- mean(control, na.rm = TRUE)
ate <- mean_T - mean_C
print(paste("ATE: ", ate))
```

Using the experimental data, obtain an unbiased estimate of the effect of NSW on 1978 earnings and its standard error.

```
## [1] "ATE:  1794.34308292048"
```

```
# standard error
seDiffMeans <- function(y, tx){
 y1 = y[tx == 1]
 y0 = y[tx == 0]
 n1 = length(y1)
 n0 = length(y0)

 sqrt(((var(y1) / n1 + var(y0) / n0)))
}

se_ate <- seDiffMeans(df_exp$re78, df_exp$nsw)
print(paste("S.E.: ", se_ate))
```

```
## [1] "S.E.:  670.996728049429"
```

```r
reg <- lm(re78 ~ . - re75 - u75 - u78, data = df_exp)

# HC2 robust SE
hc2_vcov <- vcovHC(reg, type = "HC2")
robust_se <- sqrt(diag(hc2_vcov))

# coefficient
coefficient <- summary(reg)$coef[, "Estimate"]

result_table <- data.frame(
  Coefficient = coefficient,
  HC2_SE = robust_se,
  row.names = rownames(summary(reg)$coef)
)

result_table
```

Estimate this effect again using a linear regression that controls for age, education, race, ethnicity, marital status, employment in 1974 and earnings in 1974

```
##               Coefficient       HC2_SE
## (Intercept)   144.7116671 2869.8421657
## nsw          1720.7544585  677.9793381
## age            52.9567847   40.1987165
## educ          414.9402501  164.2376269
## black       -2165.7902648 1021.4320063
## hisp          255.4063180 1412.0002640
## married       -66.0806800  840.0627653
## re74            0.1303191    0.1201542
## u74           528.3037613 1094.0567552
```

Compare these two estimates and comment The regression estimate (1720.75) is lower than the Naive difference in means (1794.34). This is because, although randomization is theoretically balanced, there may still be some imbalances in the covariates, and regression can capture them, disentangling their effect on outcome. While there's gap, the small difference proves the success of the randomization.

However, standard error is larger in the regression estimate (677.98 > 670.99), meaning that controling for those covariates didn't reduce variance very much, which implies that the baseline differences between the two groups was clear enough (at least from these covariates), further proving the Naive estimate's reliability.

Non-experimental data

Compare these two estimates and comment Load and Inspect the Data

```r
df_psid <- read.dta("nsw_psid_withtreated.dta")
```

check NA

```r
table(is.na(df_psid$column_name))
```

```
## < table of extent 0 >
```

Calculate the (naive) ATE of employment program on trainee's by the same two methods you used before (controlling for the same covariates) Naive

```r
treated <- df_psid$re78[df_psid$nsw == 1]
control <- df_psid$re78[df_psid$nsw == 0]

# mean
mean_T <- mean(treated, na.rm = TRUE)
mean_C <- mean(control, na.rm = TRUE)
ate <- mean_T - mean_C
print(paste("ATE: ", ate))
```

```
## [1] "ATE:  -15204.7755516708"
```

```r
# se
seDiffMeans <- function(y, tx){
 y1 = y[tx == 1]
 y0 = y[tx == 0]
 n1 = length(y1)
 n0 = length(y0)

 sqrt(((var(y1) / n1 + var(y0) / n0)))
}

se_ate <- seDiffMeans(df_psid$re78, df_psid$nsw)
print(paste("S.E.: ", se_ate))
```

```
## [1] "S.E.:  657.076472591643"
```

regression

```r
reg <- lm(re78 ~ . - re75 - u75 - u78, data = df_psid)

hc2_vcov <- vcovHC(reg, type = "HC2")
robust_se <- sqrt(diag(hc2_vcov))

coefficient <- summary(reg)$coef[, "Estimate"]

result_table <- data.frame(
  Coefficient = coefficient,
  HC2_SE = robust_se,
  row.names = rownames(summary(reg)$coef)
)

result_table
```

```
##                 Coefficient       HC2_SE
## (Intercept)   254.4302378 1.503747e+03
## nsw         -1459.6133145 9.327112e+02
## age           -86.1133076 2.262536e+01
## educ          661.8764728 8.649220e+01
## black        -834.6460026 4.717086e+02
## hisp         1148.7570713 1.316119e+03
## married      1452.6353658 5.312366e+02
## re74            0.7715412 3.238084e-02
## u74          2363.4393529 1.082312e+03
```

Briefly but concretely describe what are you estimating? Do these methods recover the experimental results? Unlike the experimental sample, the PSID controls were not randomly assigned, meaning they likely differ systematically from the treated group, leading to selection bias. This is proved by the completely opposite result, where the intervention appears to lower earnings instead of improving them. Here naive ATE shows a huge negative effect (-15204.78), and while regression ATE is closer to the experimental result after controlling for observed covariates, it's still negative and thus significantly different from the true effect. So, neither of them recovers the experimental results, as the baseline differences between the two groups are not fully accounted for.

```r
library(Matching)
```

Using the non-experimental dataset, check covariate balance in the unmatched dataset for all covariates. Your output should be in the form of a balance table. Make sure to present statistical tests of the similarity of means and similarity of distributions.

```
## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

## ##
## ##  Matching (Version 4.10-15, Build Date: 2024-10-14)
## ##  See https://www.jsekhon.com for additional documentation.
## ##  Please cite software as:
## ##   Jasjeet S. Sekhon. 2011. ``Multivariate and Propensity Score Matching
## ##   Software with Automated Balance Optimization: The Matching package for R.''
## ##   Journal of Statistical Software, 42(7): 1-52.
## ##
```

```r
df_psid$nsw <- as.numeric(df_psid$nsw)
covariates <-  setdiff(colnames(df_psid), c("nsw", "re78"))

balance_table <- data.frame(Variable = character(),
                            Treated_Mean = numeric(),
                            Control_Mean = numeric(),
                            TTest_p = numeric(),
                            KS_p = numeric(),
                            Variance_Ratio = numeric(),
                            stringsAsFactors = FALSE)

for (cov in covariates) {
  treated <- df_psid %>% filter(nsw == 1) %>% pull(!!sym(cov))
  control <- df_psid %>% filter(nsw == 0) %>% pull(!!sym(cov))

  mean_T <- mean(treated, na.rm = TRUE)
  mean_C <- mean(control, na.rm = TRUE)

  treated_se <- sd(treated, na.rm = TRUE)
  control_se <- sd(control, na.rm = TRUE)

  # t-test
  t_test <- t.test(treated, control, var.equal = FALSE)
  # KS test
  ks_test <- ks.test(treated, control)

  # variance ratio
  variance_ratio <- (treated_se^2) / (control_se^2)

  balance_table <- rbind(balance_table,
                         data.frame(Variable = cov,
                                    Treated_Mean = mean_T,
                                    Control_Mean = mean_C,
                                    tTest_p = t_test$p.value,
                                    KS_p = ks_test$p.value,
                                    Variance_Ratio = variance_ratio))
}
```

```
## Warning in ks.test.default(treated, control): p-value will be approximate in
## the presence of ties
## Warning in ks.test.default(treated, control): p-value will be approximate in
## the presence of ties
## Warning in ks.test.default(treated, control): p-value will be approximate in
## the presence of ties
## Warning in ks.test.default(treated, control): p-value will be approximate in
## the presence of ties
## Warning in ks.test.default(treated, control): p-value will be approximate in
## the presence of ties
## Warning in ks.test.default(treated, control): p-value will be approximate in
## the presence of ties
## Warning in ks.test.default(treated, control): p-value will be approximate in
## the presence of ties
## Warning in ks.test.default(treated, control): p-value will be approximate in
## the presence of ties
## Warning in ks.test.default(treated, control): p-value will be approximate in
## the presence of ties
## Warning in ks.test.default(treated, control): p-value will be approximate in
## the presence of ties
```

```
print(balance_table)
```

```
##    Variable Treated_Mean Control_Mean       tTest_p         KS_p Variance_Ratio
## 1       age 2.581622e+01 3.485060e+01  7.515784e-40 1.061827e-21     0.46963194
## 2      educ 1.034595e+01 1.211687e+01  1.945334e-23 1.052333e-24     0.42548606
## 3     black 8.432432e-01 2.506024e-01  5.432535e-55 5.841354e-53     0.70739349
## 4      hisp 5.945946e-02 3.253012e-02  1.317327e-01 9.996365e-01     1.78589042
## 5   married 1.891892e-01 8.662651e-01  3.375817e-58 5.378921e-69     1.33075963
## 6      re74 2.095574e+03 1.942875e+04 4.582953e-143 5.723212e-80     0.13285024
## 7      re75 1.532056e+03 1.906334e+04 2.467430e-251 6.035188e-90     0.05605655
## 8       u74 7.081081e-01 8.634538e-02  3.790498e-44 2.996767e-58     2.63317599
## 9       u75 6.000000e-01 1.000000e-01  3.306655e-30 8.073418e-38     2.68008265
## 10      u78 2.432432e-01 1.148594e-01  9.700091e-05 6.849519e-03     1.81969088
```

Based on your table, which of the observed covariates seem to be the most important factors in selection into the program? Earnings in 1974 and 1975 are the most important predictors of selection into the NSW program. Their extremely low p values are telling us that treated and control groups are entirely different. This is double confirmed by the variance ratios, which are far both from 1. Unemployment status is another critical factor, with the treated group having a much higher rate of unemployment compared to the controls. This difference is proved by the low p values and high variance ratios.

In terms of demographic characteristics, the table shows that those selected into the program are predominantly Black, unmarried, younger, and have lower levels of education.

Comparing propensity scores

```
# exp
df_exp$nsw <- as.numeric(df_exp$nsw)
ps_exp <- glm(nsw ~ . - re78, data = df_exp, family = binomial(link = logit))
exp_pscore <- predict(ps_exp, type = "response")
summary(exp_pscore)
```

Estimate propensity scores using logistic regression for both the experimental and non-experimental data.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.1468  0.3387  0.4250  0.4157  0.4924  0.6286
```

```
# psid
df_psid$nsw <- as.numeric(df_psid$nsw)
ps_psid <- glm(nsw ~ . - re78, data = df_psid, family = binomial(link = logit))
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

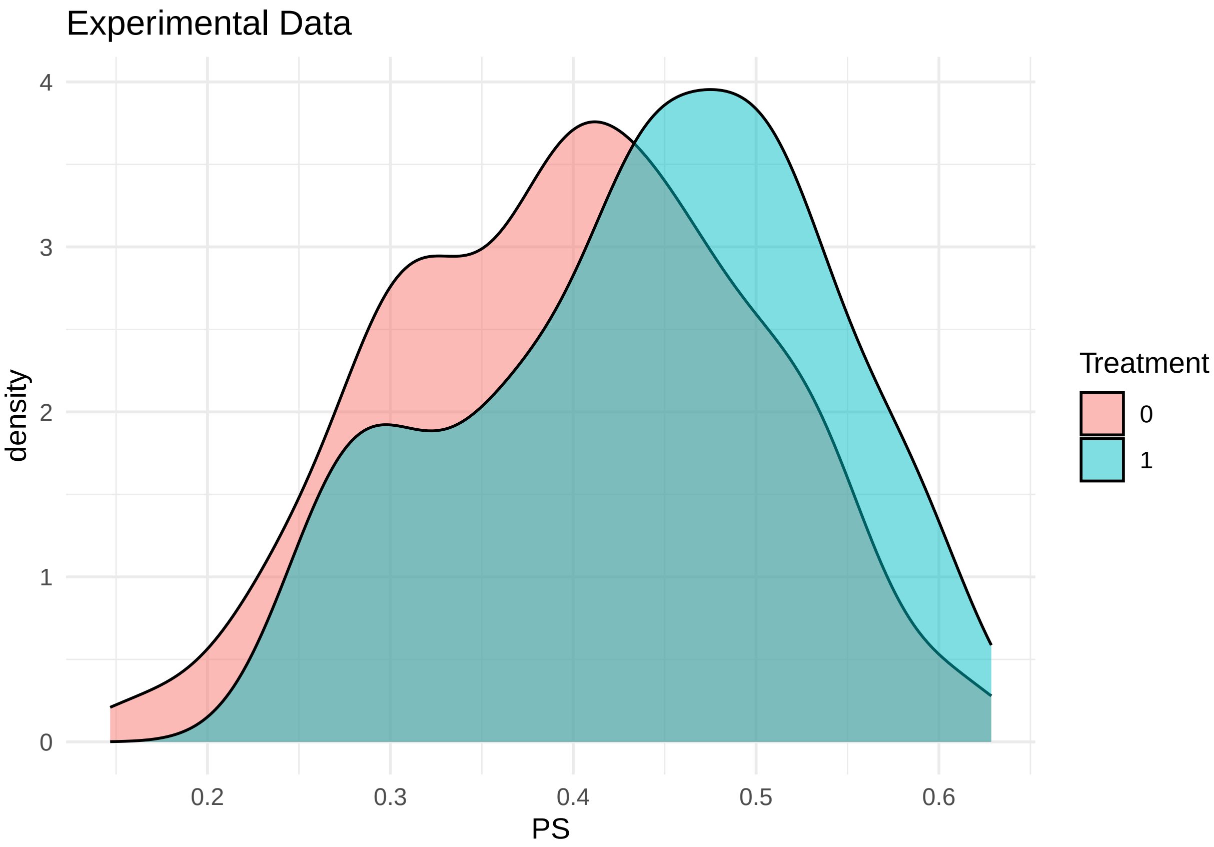
```
psid_pscore <- predict(ps_psid, type = "response")
summary(psid_pscore)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 0.0000000 0.0000147 0.0003610 0.0691589 0.0105103 0.9896958
```

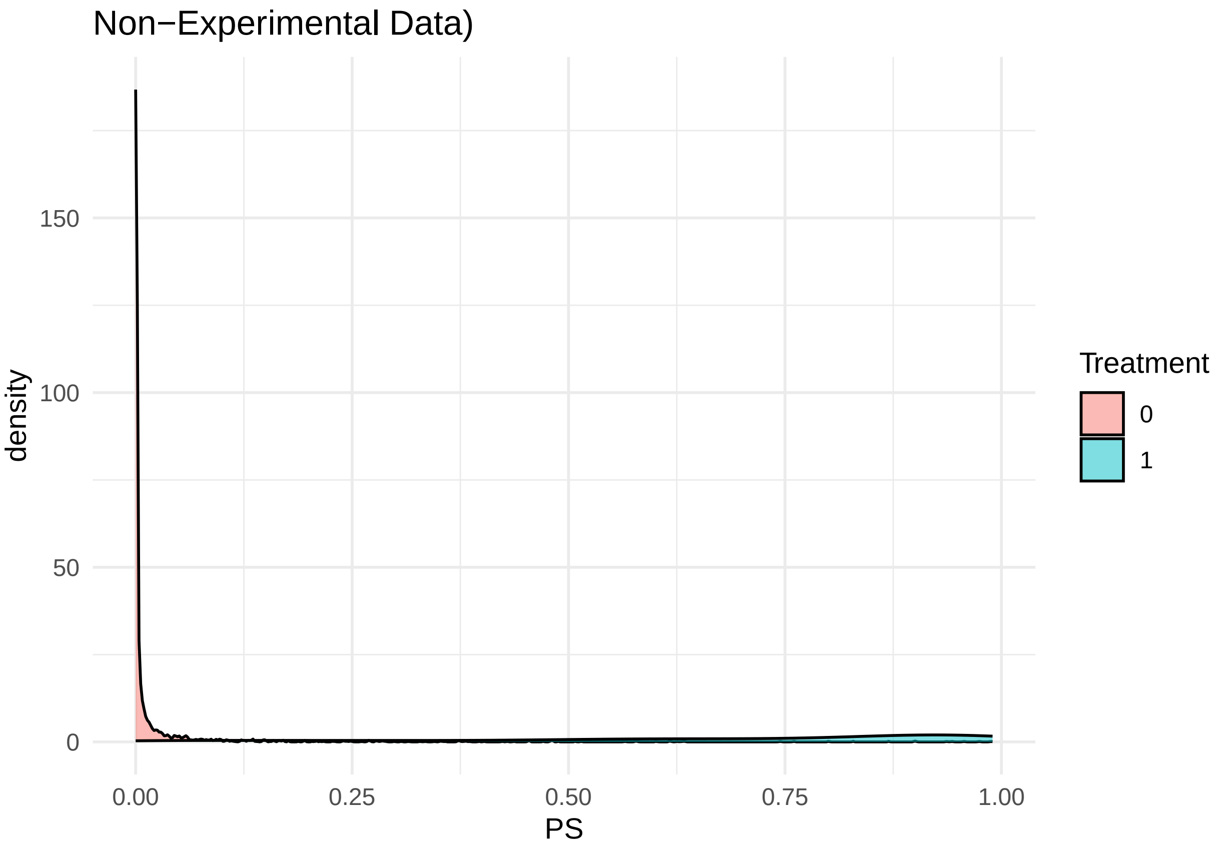
Report the distributions of propensity scores for treated and control groups. Comment on the overlap for both data sets. How do they differ and why? In the experimental data, there is a good amount of overlap between the treated and control groups. While they are not perfectly matched, their distributions are reasonably similar. In contrast, the PSID data shows almost complete separation, where the treated group has much higher propensity scores, while the control group is close to 0. This means that the randomization is safe and sound, whereas selection bias is witnessed in the PSID dataset (due to earnings, unemployment history, race…)

```
df_exp_plot <- df_exp
df_exp_plot$pscore <- exp_pscore
df_psid_plot <- df_psid
df_psid_plot$pscore <- psid_pscore

ggplot(df_exp_plot, aes(x = pscore, fill = as.factor(nsw))) +
  geom_density(alpha = 0.5) +
  labs(title = "Experimental Data",
       x = "PS", fill = "Treatment") +
  theme_minimal()
```



```
ggplot(df_psid_plot, aes(x = pscore, fill = as.factor(nsw))) +
  geom_density(alpha = 0.5) +
  labs(title = "Non-Experimental Data)",
       x = "PS", fill = "Treatment") +
  theme_minimal()
```



Distance matching

Choose some covariates on which to match, and then do so using a package of your choice (e.g., Matching). Briefly justify your choice of covariates. Be sure to carefully check the options available to you in the matching function. For now, find only one match for each treated unit, use the Mahalanobis distance metric to select matches, and do not use exact matching. Mahalanobis distance should not be used on categorical variables. So here I just chose all numerical covariates as they are all imbalanced from the balance tests we've seen earlier.

```
match_out <- Match(Y = df_psid$re78,
                   Tr = df_psid$nsw,
                   X = df_psid[, c("age", "educ", "re74", "re75")],
                   M = 1,
                   Weight = 2,
                   estimand = "ATT")
```

```

att <- match_out$est
se <- match_out$se

cat("ATT:", att, "\n")
```

Apply the matching estimator to estimate the average effect of the employment program on trainee earnings i.e., the ATT. Report your estimate and standard error, as well as balance statistics for the matched data.

```
## ATT: 311.0455
```

```r
cat("S.E.:", se, "\n")
```

```
## S.E.: 1182.173
```

```r
# check covariate balance for matched data
MatchBalance(nsw ~ age + educ + re74 + re75, data = df_psid, match.out = match_out)
```

```
##
## ***** (V1) age *****
##                        Before Matching        After Matching
## mean treatment........     25.816             25.816
## mean control..........     34.851             26.213
## std mean diff.........    -126.27            -5.5401
##
## mean raw eQQ diff.....     9.0432            0.62562
## med  raw eQQ diff.....          8                  1
## max  raw eQQ diff.....         17                  2
##
## mean eCDF diff........    0.23165           0.018958
## med  eCDF diff........    0.25299          0.0098522
## max  eCDF diff........    0.37714            0.12315
##
## var ratio (Tr/Co).....    0.46963             1.0576
## T-test p-value........ < 2.22e-16         0.00076291
## KS Bootstrap p-value.. < 2.22e-16              0.062
## KS Naive p-value...... < 2.22e-16           0.092018
## KS Statistic..........    0.37714            0.12315
##
##
## ***** (V2) educ *****
##                        Before Matching        After Matching
## mean treatment........     10.346             10.346
## mean control..........     12.117             10.465
## std mean diff.........    -88.077            -5.9145
##
## mean raw eQQ diff.....     1.8595            0.16256
## med  raw eQQ diff.....          2                  0
## max  raw eQQ diff.....          5                  2
##
## mean eCDF diff........     0.1091           0.012505
## med  eCDF diff........    0.01944          0.0098522
## max  eCDF diff........    0.40289           0.029557
##
## var ratio (Tr/Co).....    0.42549             1.2115
## T-test p-value........ < 2.22e-16         0.00012147
## KS Bootstrap p-value.. < 2.22e-16              0.964
## KS Naive p-value...... < 2.22e-16            0.99999
## KS Statistic..........    0.40289           0.029557
##
##
## ***** (V3) re74 *****
##                        Before Matching        After Matching
## mean treatment........     2095.6             2095.6
## mean control..........      19429               2976
## std mean diff.........    -354.71            -18.017
##
## mean raw eQQ diff.....      17663             867.75
## med  raw eQQ diff.....      18417                  0
## max  raw eQQ diff.....     102109             3231.2
##
## mean eCDF diff........    0.46806           0.073673
## med  eCDF diff........    0.54766           0.022167
## max  eCDF diff........    0.72924            0.23153
##
## var ratio (Tr/Co).....    0.13285            0.94292
## T-test p-value........ < 2.22e-16         3.3521e-11
## KS Bootstrap p-value.. < 2.22e-16         < 2.22e-16
## KS Naive p-value...... < 2.22e-16         3.7595e-05
## KS Statistic..........    0.72924            0.23153
##
##
## ***** (V4) re75 *****
##                        Before Matching        After Matching
## mean treatment........     1532.1             1532.1
## mean control..........      19063             2233.7
## std mean diff.........    -544.58            -21.795
##
## mean raw eQQ diff.....      17978              640.4
## med  raw eQQ diff.....      17903                  0
## max  raw eQQ diff.....     131511             5293.3
##
## mean eCDF diff........    0.46947           0.070399
## med  eCDF diff........    0.53317           0.068966
## max  eCDF diff........    0.77362            0.15271
##
## var ratio (Tr/Co).....   0.056057             0.7071
## T-test p-value........ < 2.22e-16         5.1789e-07
## KS Bootstrap p-value.. < 2.22e-16               0.01
## KS Naive p-value...... < 2.22e-16           0.017583
## KS Statistic..........    0.77362            0.15271
##
##
## Before Matching Minimum p.value: < 2.22e-16
## Variable Name(s): age educ re74 re75  Number(s): 1 2 3 4
##
## After Matching Minimum p.value: < 2.22e-16
## Variable Name(s): re74  Number(s): 3
```

Re-estimate the ATT using exact matching on education, race, ethnicity and married. Report your estimate, its standard error, and produce a balance table as before. In general, do your results differ from previous results? The results differ significantly from the previous estimates. Mahalanobis Matching shows a positive effect, whereas Exact Matching flips the ATT negative. While exact matching perfectly balances the matched covariates, it doesn't necessarily improve balance on other key variables, like re74 and re75, which remain imbalanced. The larger SE also hints at smaller effective sample sizes, as exact matching discards units that don't have exact counterparts. So while it eliminates bias on the matched covariates, it may come at the cost of increased variance, leaving some selection bias unresolved.

```
covariates <- c("age", "re74", "re75")
exact_vars <- c("educ", "black", "hisp", "married")

df_psid$nsw <- as.numeric(df_psid$nsw)
df_psid$educ <- as.numeric(as.factor(df_psid$educ))
df_psid$black <- as.numeric(as.factor(df_psid$black))
df_psid$hisp <- as.numeric(as.factor(df_psid$hisp))
df_psid$married <- as.numeric(as.factor(df_psid$married))

match_out <- Match(Y = df_psid$re78,
                   Tr = df_psid$nsw,
                   X = df_psid[, c(exact_vars, covariates)],
                   M = 1,
                   exact = c(TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, FALSE),
                   estimand = "ATT")

att <- match_out$est
se <- match_out$se

# Print results
cat("\nATT:", att, "\n")
```

```
##
## ATT: -447.2495
```

```
cat("S.E.:", se, "\n")
```

```
## S.E.: 1239.851
```

```
MatchBalance(nsw ~ age + educ + black + hisp + married + re74 + re75, data = df_psid, match.out = match_
```

```
##
## ***** (V1) age *****
##                        Before Matching        After Matching
## mean treatment........     25.816             25.882
## mean control..........     34.851              26.17
## std mean diff.........    -126.27              -3.99
##
## mean raw eQQ diff.....     9.0432            0.89785
## med  raw eQQ diff.....          8                  1
## max  raw eQQ diff.....         17                  7
##
## mean eCDF diff........    0.23165            0.02621
## med  eCDF diff........    0.25299           0.016129
## max  eCDF diff........    0.37714            0.13978
##
## var ratio (Tr/Co).....    0.46963            0.96777
## T-test p-value........ < 2.22e-16            0.47863
## KS Bootstrap p-value.. < 2.22e-16               0.02
## KS Naive p-value...... < 2.22e-16           0.052798
## KS Statistic..........    0.37714            0.13978
##
##
## ***** (V2) educ *****
##                        Before Matching        After Matching
## mean treatment........     10.346             10.371
## mean control..........     12.118             10.371
## std mean diff.........    -88.137                  0
##
## mean raw eQQ diff.....     1.8541                  0
## med  raw eQQ diff.....          2                  0
## max  raw eQQ diff.....          5                  0
##
## mean eCDF diff........     0.1091                  0
## med  eCDF diff........    0.01944                  0
## max  eCDF diff........    0.40289                  0
##
## var ratio (Tr/Co).....    0.42674                  1
## T-test p-value........ < 2.22e-16                  1
## KS Bootstrap p-value.. < 2.22e-16                  1
## KS Naive p-value...... < 2.22e-16                  1
## KS Statistic..........    0.40289          3.296e-17
##
##
## ***** (V3) black *****
##                        Before Matching        After Matching
## mean treatment........     1.8432             1.8764
## mean control..........     1.2506             1.8764
## std mean diff.........     162.56                  0
##
## mean raw eQQ diff.....    0.58919                  0
## med  raw eQQ diff.....          1                  0
## max  raw eQQ diff.....          1                  0
##
## mean eCDF diff........    0.29632                  0
## med  eCDF diff........    0.29632                  0
## max  eCDF diff........    0.59264                  0
##
## var ratio (Tr/Co).....    0.70739                  1
## T-test p-value........ < 2.22e-16                  1
##
##
## ***** (V4) hisp *****
##                        Before Matching        After Matching
## mean treatment........     1.0595             1.0225
## mean control..........     1.0325             1.0225
## std mean diff.........     11.357                  0
##
## mean raw eQQ diff.....   0.027027                  0
## med  raw eQQ diff.....          0                  0
## max  raw eQQ diff.....          1                  0
##
## mean eCDF diff........   0.013465                  0
## med  eCDF diff........   0.013465                  0
## max  eCDF diff........   0.026929                  0
##
## var ratio (Tr/Co).....     1.7859                  1
## T-test p-value........    0.13173                  1
##
```

```
##
## ***** (V5) married *****
##                        Before Matching        After Matching
## mean treatment........     1.1892            1.1966
## mean control..........     1.8663            1.1966
## std mean diff.........    -172.41                 0
##
## mean raw eQQ diff.....    0.67568                 0
## med  raw eQQ diff.....          1                 0
## max  raw eQQ diff.....          1                 0
##
## mean eCDF diff........    0.33854                 0
## med  eCDF diff........    0.33854                 0
## max  eCDF diff........    0.67708                 0
##
## var ratio (Tr/Co).....     1.3308                 1
## T-test p-value........ < 2.22e-16                 1
##
##
## ***** (V6) re74 *****
##                        Before Matching        After Matching
## mean treatment........     2095.6            2053.9
## mean control..........      19429            4334.9
## std mean diff.........    -354.71           -46.735
##
## mean raw eQQ diff.....      17663              2364
## med  raw eQQ diff.....      18417            2088.6
## max  raw eQQ diff.....     102109            5877.8
##
## mean eCDF diff........    0.46806           0.17045
## med  eCDF diff........    0.54766           0.11828
## max  eCDF diff........    0.72924           0.43548
##
## var ratio (Tr/Co).....    0.13285            0.9469
## T-test p-value........ < 2.22e-16         3.078e-12
## KS Bootstrap p-value.. < 2.22e-16        < 2.22e-16
## KS Naive p-value...... < 2.22e-16        9.5861e-16
## KS Statistic..........    0.72924           0.43548
##
##
## ***** (V7) re75 *****
##                        Before Matching        After Matching
## mean treatment........     1532.1            1502.9
## mean control..........      19063            3844.7
## std mean diff.........    -544.58           -72.933
##
## mean raw eQQ diff.....      17978            2373.9
## med  raw eQQ diff.....      17903            2685.5
## max  raw eQQ diff.....     131511            8527.8
##
## mean eCDF diff........    0.46947           0.20609
## med  eCDF diff........    0.53317           0.24194
## max  eCDF diff........    0.77362           0.34409
##
```

```
## var ratio (Tr/Co).....   0.056057            0.45974
## T-test p-value........ < 2.22e-16         9.4245e-12
## KS Bootstrap p-value.. < 2.22e-16         < 2.22e-16
## KS Naive p-value...... < 2.22e-16         5.4602e-10
## KS Statistic.........    0.77362            0.34409
##
##
## Before Matching Minimum p.value: < 2.22e-16
## Variable Name(s): age educ black married re74 re75  Number(s): 1 2 3 5 6 7
##
## After Matching Minimum p.value: < 2.22e-16
## Variable Name(s): re74 re75  Number(s): 6 7
```

Propensity score matching and weighting

```r
match_psid <- Match(
  Y = df_psid$re78,
  Tr = df_psid$nsw,
  X = log(psid_pscore / (1 - psid_pscore)),
  M = 1,
  estimand = "ATT"
)

att <- match_psid$est
se <- match_psid$se
cat("ATT:", att, "\n")
```

Now let's use the propensity scores we calculated before to match on the estimated propensity scores and obtain an estimator of the average treatment effect on the treated for the NSW program.

```
## ATT: 1143.529
```

```r
cat("S.E.:", se, "\n")
```

```
## S.E.: 1667.342
```

Finally, use weighting on the propensity score to estimate the average effect of the treatment on the treated for the NSW program. Do your results accord with your previous findings? PSM ATT is lower than the experimental result, though still positive, and IPW ATT is even smaller. Both estimates indicate that the NSW program increased earnings, but the effect is much weaker than what the experimental data suggests. The larger standard errors suggests that there may be some potential unobserved differences between the treated and control groups, something that was already implied at by the poor overlap in propensity score distributions. The higher SE for PSM makes sense since it drops unmatched units, reducing statistical power and increasing variability.

```r
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
psid_pscore <- predict(ps_psid, type = "response")

# trim extreme ps to avoid infinity
psid_pscore <- pmax(pmin(psid_pscore, 0.99), 0.01)

# IPW
df_psid$ipw <- df_psid$nsw + (1 - df_psid$nsw) * (psid_pscore / (1 - psid_pscore))
ipw_model <- lm(re78 ~ nsw, data = df_psid, weights = ipw)

att <- coef(ipw_model)["nsw"]
se <- sqrt(diag(vcovHC(ipw_model, type = "HC2"))["nsw"])
cat("ATT:", att, "\n")
```

```
## ATT: 966.7503
```

```
cat("S.E.:", se, "\n")
```

```
## S.E.: 1290.887
```

```
weighted_balance <- lm(nsw ~ . - re78, data = df_psid, weights = ipw)
summary(weighted_balance)
```

```
##
## Call:
## lm(formula = nsw ~ . - re78, data = df_psid, weights = ipw)
##
## Weighted Residuals:
##      Min       1Q   Median       3Q      Max
## -1.99604 -0.03678 -0.01769  0.00238  0.86897
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.451e-01  1.118e-01   7.559 5.54e-14 ***
## age         -5.977e-03  1.330e-03  -4.495 7.24e-06 ***
## educ        -3.929e-02  4.073e-03  -9.645  < 2e-16 ***
## black        1.255e-01  2.903e-02   4.324 1.59e-05 ***
## hisp        -2.477e-02  3.613e-02  -0.685   0.4931
## married      4.668e-02  2.542e-02   1.837   0.0664 .
## re74         5.323e-06  2.401e-06   2.217   0.0267 *
## re75        -1.594e-05  2.487e-06  -6.410 1.71e-10 ***
## u74          3.179e-01  3.359e-02   9.463  < 2e-16 ***
## u75         -2.308e-01  2.730e-02  -8.453  < 2e-16 ***
## u78         -4.877e-02  2.152e-02  -2.267   0.0235 *
## ipw         -1.310e-02  4.399e-04 -29.779  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1681 on 2663 degrees of freedom
## Multiple R-squared:  0.3432, Adjusted R-squared:  0.3405
## F-statistic: 126.5 on 11 and 2663 DF,  p-value: < 2.2e-16
```

Reflection

(No answer required, just think about it!) Under what assumptions is the ATT you estimated identified? Does matching make any identification assumption more plausible? If we want to identify ATT, we can achieve this through SOO, which is built upon conditional ignorability and common support assumptions. For the former, it means that that controlling for pre-treatment covariates can simulate an environment where treatment assignment is as good as random; for the latter, it requires that treated and control groups share similar characteristics so that each treated unit has a comparable control counterpart.

Matching makes this assumption more plausible by preprocessing the data to improve balance between treated and control groups. While it helps approximate a randomized experiment, it doesn't magically create randomization, as there will always be unobserved covariates that we can't control for.