# 1

We can compute the ATET in two ways: (1) by regression and (2) by non-regression. Doing the regression method, we can run the following code:

```
# Imports data and runs regression to find the ATET

nsw_dw <- read_dta("http://www.nber.org/~rdehejia/data/nsw_dw.dta")
summary(treatment <- lm_robust(re78~treat+age+education+black+hispanic+married+
    nodegree+re74,data=nsw_dw))
```

Table 1:

| | *Dependent variable:* |
|---|---|
| | re78 |
| treat | 1,693.116*** |
| | (636.608) |
| age | 56.145 |
| | (45.190) |
| education | 401.960* |
| | (226.631) |
| black | −2,187.164* |
| | (1,165.744) |
| hispanic | 176.173 |
| | (1,547.636) |
| married | −64.252 |
| | (857.860) |
| nodegree | −20.196 |
| | (995.018) |
| re74 | 0.102* |
| | (0.058) |
| Constant | 694.617 |
| | (3,363.693) |
| Observations | 445 |
| $R^2$ | 0.055 |
| Adjusted $R^2$ | 0.037 |
| Residual Std. Error | 6,507.140 (df = 436) |
| F Statistic | 3.141*** (df = 8; 436) |
| *Note:* | *p<0.1; **p<0.05; ***p<0.01 |

This means, on average, if an individual participates in the NSW program, their income would increase by $1,693.12, ceteris paribus. Doing the non-regression method, we can run:

```
# Filters treated
treated <- nsw_dw[nsw_dw$treat==1,]
# Filters untreated
untreated <- nsw_dw[nsw_dw$treat==0,]
# Subtracts the mean of treated in 1978 by the mean of the treated in 1978
ATET <- mean(treated$re78)-mean(untreated$re78)
```

We obtain $1,794.34. This means, on average, if an individual participates in the NSW program, their income would increase by $1,794.34, ceteris paribus. From the two results, we expect to find the NSW

program will increase between $1,693.12 and $1,794.34 depending on whether exogenous covariates were included in the regression.

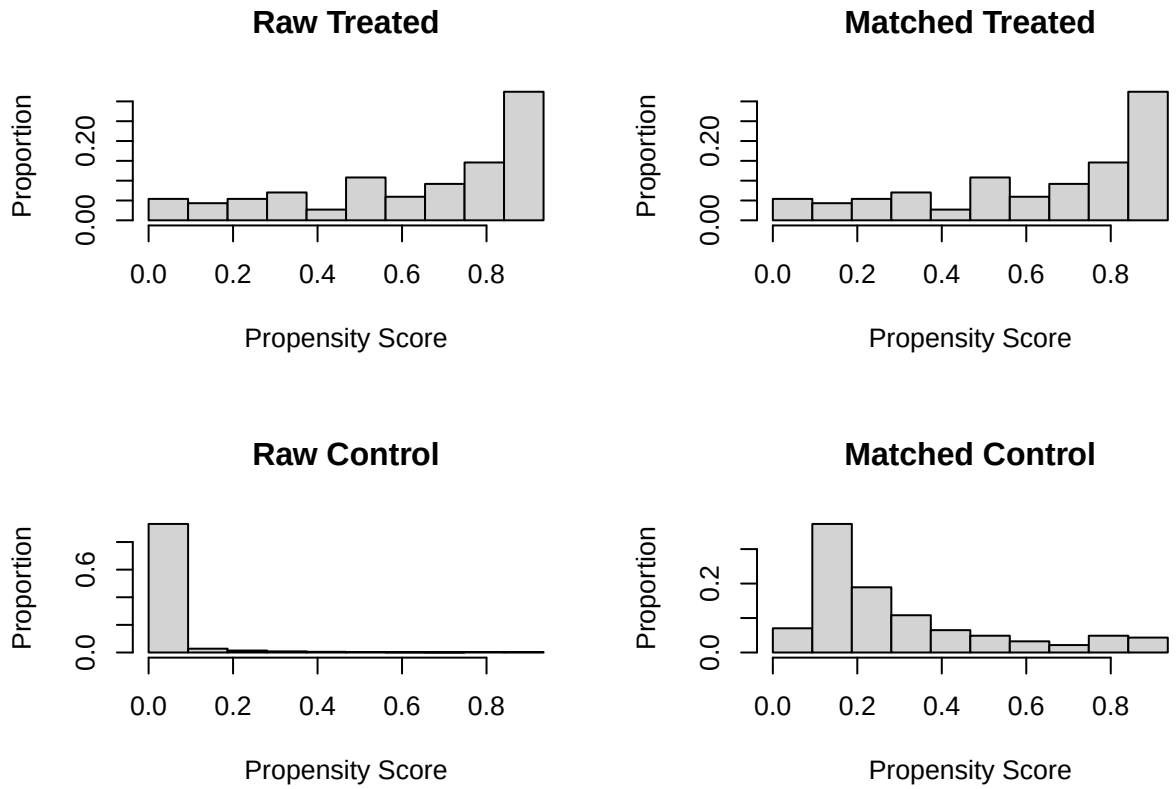
## 2

In order to estimate the propensity scores for the nearest neighbor without replacement, we can run:

```
m.out <- matchit(treat ~ age + agesq + agecube + education + educsq + black +
    hispanic + married +
nodegree + re74 + re75 + u74 + u75 + interaction1, data = psid_data,
method = "nearest", replacement = FALSE, ratio = 1)
```

To view the distribution of propensity scores for our treated and controls, we can run:

```
plot(m.out, type="hist")
```




We can take a look at our covariate balance by running

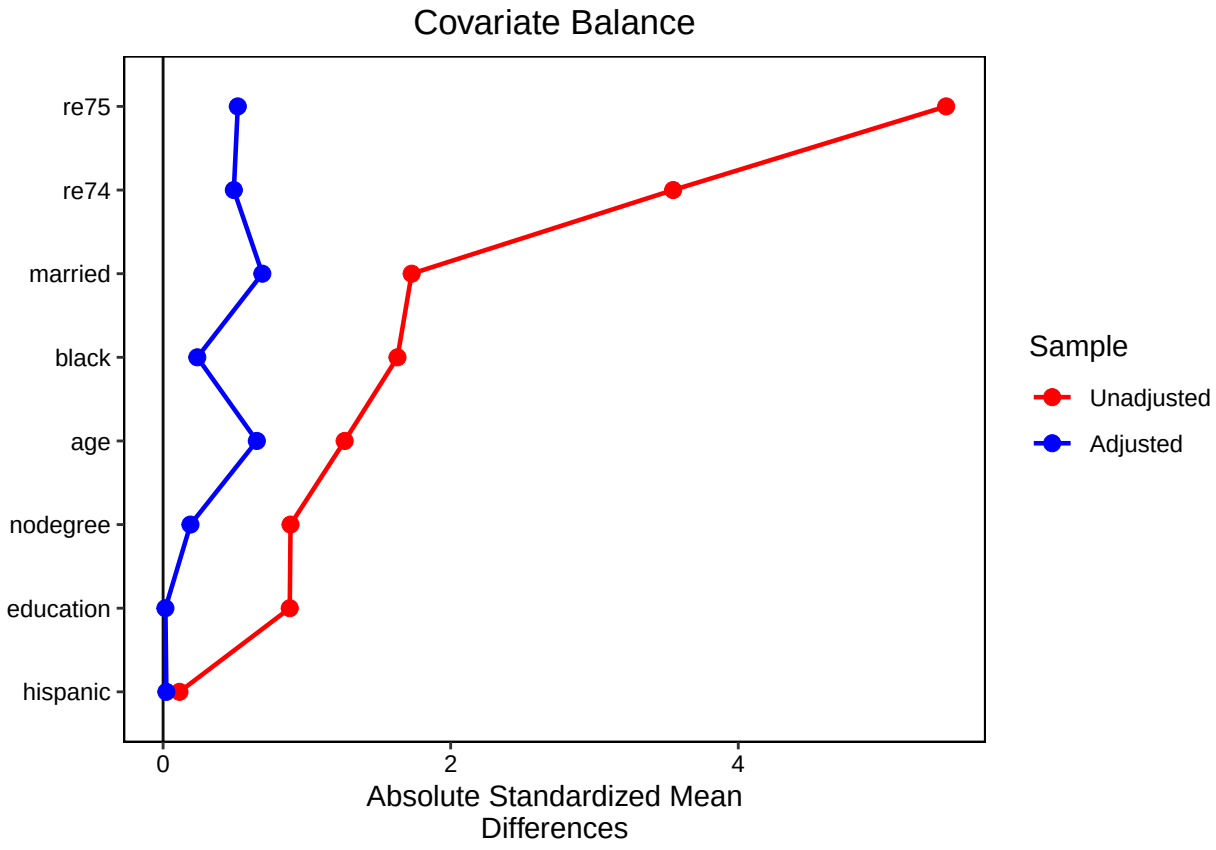
```
summary(m.out)
```

Which gives us the following

```
##
## Call:
## matchit(formula = treat ~ age + agesq + agecube + education + educsq + black + hispanic +
##     married + nodegree + re74 + re75 + u74 + u75 + interaction1, data = psid_data, method = "near
##     ratio = 1, replacement = FALSE)
##
## Summary of Balance for All Data:
##           Means Treated Means Control Std. Mean Diff. Var. Ratio eCDF Mean
## distance         0.6364        0.0270          2.1674     8.0268    0.4816
## age             25.8162       34.8506         -1.2627     0.4696    0.2317
```

```
## education       10.3459       12.1169         -0.8808     0.4255    0.1091
## black            0.8432        0.2506          1.6301          .    0.5926
## hispanic         0.0595        0.0325          0.1139          .    0.0269
## married          0.1892        0.8663         -1.7287          .    0.6771
## nodegree         0.7081        0.3052          0.8862          .    0.4029
## re74          2095.5737    19428.7458         -3.5471     0.1329    0.4684
## re75          1532.0553    19063.3377         -5.4458     0.0561    0.4695
##           eCDF Max
## distance    0.8817
## age         0.3771
## education   0.4029
## black       0.5926
## hispanic    0.0269
## married     0.6771
## nodegree    0.4029
## re74        0.7292
## re75        0.7736
##
## Summary of Balance for Matched Data:
##           Means Treated Means Control Std. Mean Diff. Var. Ratio eCDF Mean
## distance         0.6364        0.2934          1.2200     1.4702    0.0432
## age             25.8162       30.4811         -0.6520     0.4149    0.1196
## education       10.3459       10.3784         -0.0161     0.4745    0.0407
## black            0.8432        0.7568          0.2379          .    0.0865
## hispanic         0.0595        0.0649         -0.0229          .    0.0054
## married          0.1892        0.4595         -0.6901          .    0.2703
## nodegree         0.7081        0.6216          0.1902          .    0.0865
## re74          2095.5737     4499.8428         -0.4920     1.1020    0.0722
## re75          1532.0553     3204.3968         -0.5195     0.7389    0.0605
##           eCDF Max Std. Pair Dist.
## distance    0.5568          1.2200
## age         0.1784          1.3561
## education   0.0919          1.3281
## black       0.0865          0.9515
## hispanic    0.0054          0.5257
## married     0.2703          1.0213
## nodegree    0.0865          0.9036
## re74        0.4162          0.8667
## re75        0.2973          0.9044
##
## Percent Balance Improvement:
##           Std. Mean Diff. Var. Ratio eCDF Mean eCDF Max
## distance             43.7       81.5      91.0     36.9
## age                  48.4      -16.4      48.4     52.7
## education            98.2       12.8      62.7     77.2
## black                85.4          .      85.4     85.4
## hispanic             79.9          .      79.9     79.9
## married              60.1          .      60.1     60.1
## nodegree             78.5          .      78.5     78.5
## re74                 86.1       95.2      84.6     42.9
## re75                 90.5       89.5      87.1     61.6
##
## Sample Sizes:
##           Control Treated
## All          2490     185
## Matched       185     185
## Unmatched    2305       0
## Discarded       0       0
```

Additionally, we can view our covariate balance graphically.




We can find the treatment effects of our model by running

```
```{r}
m_data <- match.data(m_out)

z_out <- zelig(re78 ~ treat + age + agesq + agecube + education +
educsq + married + nodegree +
black + hispanic + re74 + re75 + interaction1,
model = "ls", data = m_data)

x_out <- setx(z_out, treat = 0)
x1_out <- setx(z_out, treat = 1)

s_out <- sim(z_out, x = x_out, x1 = x1_out)

summary(s_out)
```
```

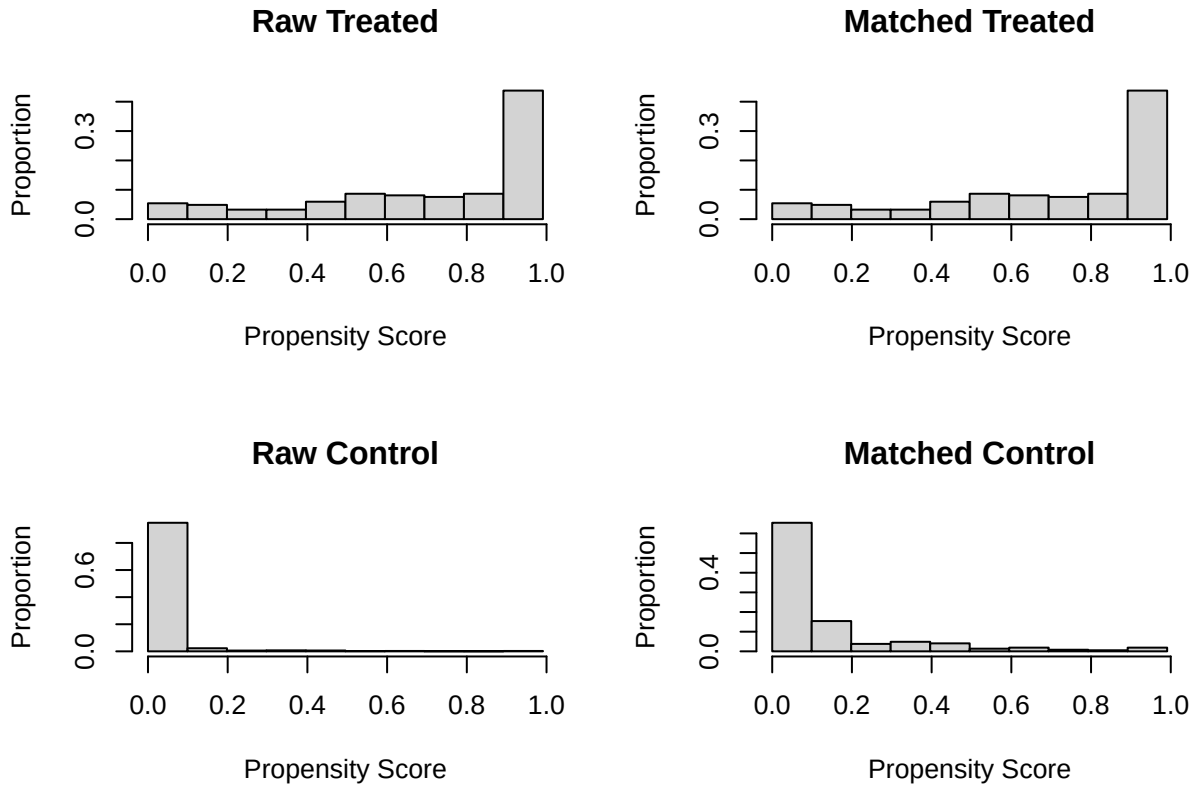
Where we get an average treatment effect on the treated as: $1,240.86

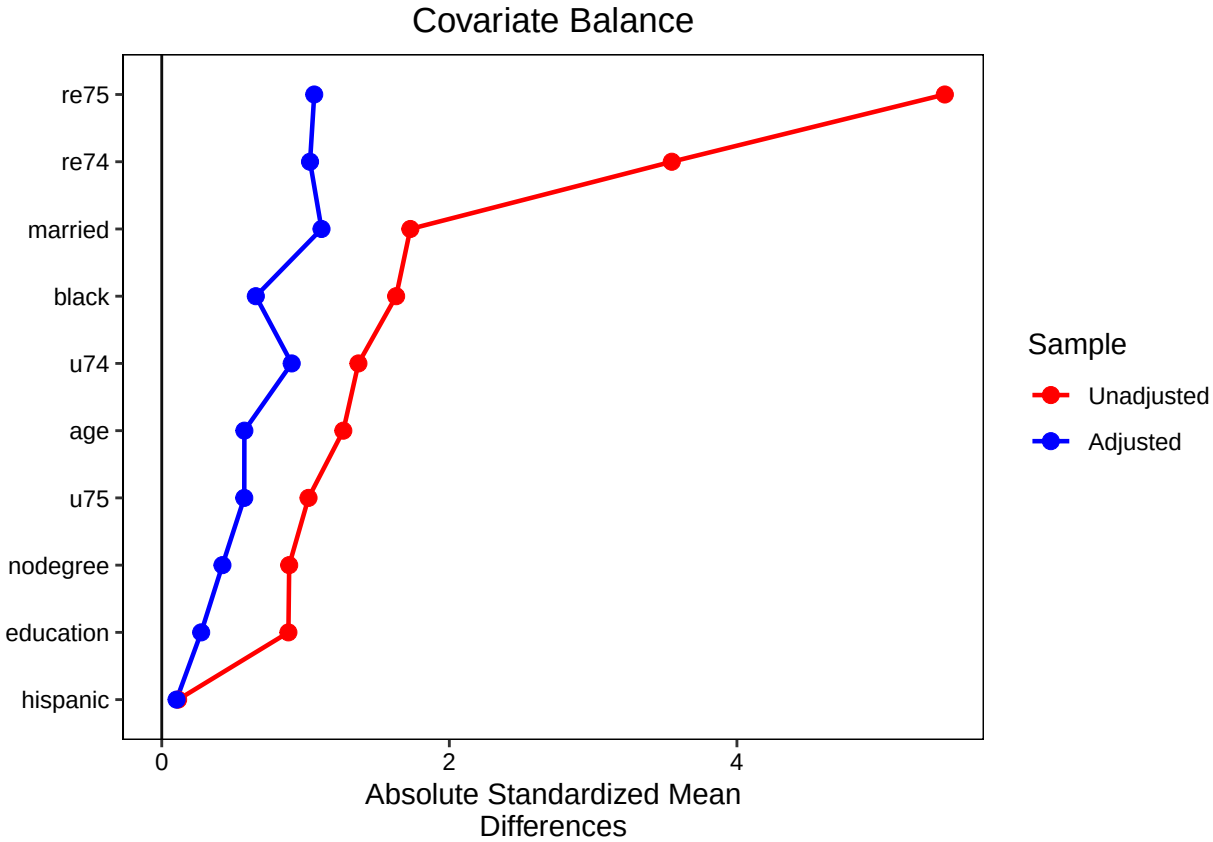
## 3

In order to find the nearest second neighbor, we can just change the ratio to 2:

```
m.out <- matchit(treat ~ age + agesq + agecube + education + educsq+ black +
    hispanic + married +
nodegree + re74 + re75 + u74 + u75 + interaction1, data = psid_data,
method = "nearest", replacement = FALSE, ratio = 2)
```

We would get the following propensity score distributions and covariate balance tables:




```
## Call
##  matchit(formula = treat ~ age + agesq + agecube + education +
##      educsq + black + hispanic + married + nodegree + u74 + u75 +
##      re74 + re74sq + re75 + re75sq + interaction1,
##      data = psid_data, method = "nearest", distance = "logit",
##      ratio = 2, replacement = TRUE)
##
```

```
## Balance Measures
##                  Type         M.0.Un        M.1.Un Diff.Un        M.0.Adj
## distance     Distance         0.0221        0.7032  2.3075         0.1411
## age           Contin.        34.8506       25.8162 -1.2627        29.9270
## agesq         Contin.      1323.5301      717.3946 -1.4055       966.1595
## agecube       Contin.     54102.2771    21554.6595 -1.5525     33509.3108
## education     Contin.        12.1169       10.3459 -0.8808        10.8973
## educsq        Contin.       156.3161      111.0595 -1.1515       124.8486
## black          Binary         0.2506        0.8432  0.5926         0.6054
## hispanic       Binary         0.0325        0.0595  0.0269         0.0838
## married        Binary         0.8663        0.1892 -0.6771         0.6243
## nodegree       Binary         0.3052        0.7081  0.4029         0.5162
## u74            Binary         0.0863        0.7081  0.6218         0.2973
## u75            Binary         0.1000        0.6000  0.5000         0.3189
## re74          Contin.     19428.7458     2095.5737 -3.5471      7134.7230
## re74sq        Contin. 557148332.5722 28141411.6013 -4.6362 113933126.0256
## re75          Contin.     19063.3377     1532.0553 -5.4458      4946.0516
## re75sq        Contin. 548213776.7900 12654750.3741 -9.5578  46744924.8194
## interaction1  Contin.    248073.3675    22898.7265 -3.9233     79251.6218
## interaction2   Binary         0.0036        0.0324  0.0288         0.0162
##                     M.1.Adj Diff.Adj
## distance             0.7032   1.9042
## age                 25.8162  -0.5745
## agesq              717.3946  -0.5768
## agecube          21554.6595  -0.5702
## education           10.3459  -0.2742
## educsq             111.0595  -0.3508
## black                0.8432   0.2378
## hispanic             0.0595  -0.0243
## married              0.1892  -0.4351
## nodegree             0.7081   0.1919
## u74                  0.7081   0.4108
## u75                  0.6000   0.2811
## re74              2095.5737  -1.0312
## re74sq        28141411.6013  -0.7519
## re75              1532.0553  -1.0605
## re75sq        12654750.3741  -0.6084
## interaction1     22898.7265  -0.9819
## interaction2         0.0324   0.0162
##
## Sample sizes
##           Control Treated
## All          2490     185
## Matched       370     185
## Unmatched    2120       0
```

With an average treatment effect on the treated as $1,360.43.

## 4

R does not have a package to perform the kernel matching method yet. Instead, I will use inverse probability weighting, which is similar to kernel matching. We can run

```
‘‘‘{r}
N <- nrow(nsw_dw_cpscontrol)
psid_data <- psid_data %>%
mutate(d1 = treat/pscore,
d0 = (1-treat)/(1-pscore))
```

```
s1 <- sum(psid_data$d1)
s0 <- sum(psid_data$d0)


psid_data <- psid_data %>%
mutate(y1 = treat * re78/pscore,
y0 = (1-treat) * re78/(1-pscore),
ht = y1 - y0)

#- Manual with normalized weights
psid_data <- psid_data %>%
mutate(y1 = (treat*re78/pscore)/(s1/N),
y0 = ((1-treat)*re78/(1-pscore))/(s0/N),
norm = y1 - y0)

psid_data %>%
pull(ht) %>%
mean()

psid_data %>%
pull(norm) %>%
mean()

#-- trimming propensity score
psid_data <- psid_data %>%
dplyr::select(-d1, -d0, -y1, -y0, -ht, -norm) %>%
filter(!(pscore >= 0.9)) %>%
filter(!(pscore <= 0.1))

N <- nrow(psid_data)

#- Manual with non-normalized weights using trimmed data
psid_data <- psid_data %>%
mutate(d1 = treat/pscore,
d0 = (1-treat)/(1-pscore))

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

psid_data <- psid_data %>%
mutate(y1 = treat * re78/pscore,
y0 = (1-treat) * re78/(1-pscore),
ht = y1 - y0)

#- Manual with normalized weights with trimmed data
psid_data <- psid_data %>%
mutate(y1 = (treat*re78/pscore)/(s1/N),
y0 = ((1-treat)*re78/(1-pscore))/(s0/N),
norm = y1 - y0)

psid_data %>%
pull(ht) %>%
mean()

psid_data %>%
pull(norm) %>%
mean()
```

This gets us an estimated treatment effect on the treated of $401.07 for non-normalized weights and $1,681.21 for normalized weights.

## 5

As I used nearest neighbor to second nearest neighbor to inverse proportional weighting, I was able to get closer and closer to the ATET that was estimated in the first part of the assignment. I used the original covariates and the model of the original paper.