Modelación en Ciencia de datos

Matching

Prof. Manuel Lecuanda

3.1. Propensity Score Matching

Objetivo

Aplicar la técnica cuasiexperimental de *matching* mediante el cálculo del *propensity score* y analizar e interpretar sus resultados en la estimación del efecto de tratamiento sobre una variable resultado.

En particular se estima el efecto en el rendimiento de los estudiantes de haber acudido a una escuela privada (católica) versus una escuela pública, utilizando la estimación de *propensity score* a partir de las covariables disponibles en la ECLS.

La variable de resultado es el rendimiento académico del alumno. El grupo tratado serán los alumnos que acudieron a una escuela católica, el grupo de control los de la escuela pública.

Referencia:

ECLS: Early Childhood Longitudinal Study [United States], (ICPSR 4075)

Instrucciones

1. Obtenga la diferencia entre medias para la variable de resultado para el grupo de tratamiento y control.
2. Evalúe las diferencia entre medias para las covariables entre grupos de tratamiento y control previas al *matching*.
3. Estime el *propensity score* como la probabilidad de recibir tratamiento dado un conjunto de covariables previas al tratamiento.
4. Analizar la región de *commun support*
5. Utilice el procedimiento de *matching* mediante el criterio de la vecindad más cercana
6. Evalúe el equilibrio de covariables después del *matching*
7. Estime los efectos del tratamiento sobre la variable de resultado.

Lectura de datos

Iniciamos por activar las librerías:

```
library(haven)
library(stargazer)
library(magrittr)
library(tidyverse)
library(knitr)
library(kableExtra)
library(MatchIt)
library(dplyr)
library(ggplot2)
```

y ahora leemos los datos que puede revisar con `View(ecls)`:

```
archivo<-"/Users/manuellecuanda/Desktop/Pantalla/Taller/ecls.c
ecls <- read.csv(archivo)
```

Lectura de datos

La descripción de la base de datos aparece en: `ecls-codebook.txt`

	childid	catholic	race	race_white	race_black	race_hispanic	race_asian	p5numpla	p5hmage
1	0001002C	0	WHITE, NON-HISPANIC	1	0	0	0	1	47
2	0001004C	0	WHITE, NON-HISPANIC	1	0	0	0	1	41
3	0001005C	0	WHITE, NON-HISPANIC	1	0	0	0	*NA*	*NA*
4	0001010C	0	WHITE, NON-HISPANIC	1	0	0	0	1	43
5	0001011C	1	WHITE, NON-HISPANIC	1	0	0	0	1	38
6	0001012C	0	WHITE, NON-HISPANIC	1	0	0	0	1	47
7	0002003C	0	WHITE, NON-HISPANIC	1	0	0	0	1	30
8	0002004C	0	WHITE, NON-HISPANIC	1	0	0	0	1	41
9	0002005C	0	HISPANIC, RACE SPECIFIED	0	0	1	0	1	38
10	0002006C	0	WHITE, NON-HISPANIC	1	0	0	0	1	28
11	0002008C	0	WHITE, NON-HISPANIC	1	0	0	0	1	31
12	0002010C	0	WHITE, NON-HISPANIC	1	0	0	0	1	38
13	0002011C	0	WHITE, NON-HISPANIC	1	0	0	0	1	26
14	0002012C	0	MORE THAN ONE RACE, NON HISPANIC	0	0	0	0	2	36
15	0002018C	0	WHITE, NON-HISPANIC	1	0	0	0	1	38
16	0002019C	0	WHITE, NON-HISPANIC	1	0	0	0	1	36
17	0002022C	0	WHITE, NON-HISPANIC	1	0	0	0	1	27

Showing 1 to 16 of 11,078 entries, 22 total columns

1. Diferencia en medias para el resultado

La variable de resultado es la puntuación en matemáticas estandarizada `c5r2mtsc_std`.

```
ecls %>%
  group_by(catholic) %>%
  summarise(n_students = n(),
            mean_math = mean(c5r2mtsc_std),
            std_error = sd(c5r2mtsc_std)/sqrt(n_students)) %>%
  kable()
```

catholic	n_students	mean_math	std_error
0	9568	-0.0305958	0.0103854
1	1510	0.1938682	0.0223528

El puntaje promedio de matemáticas de los estudiantes de escuelas católicas es más del 20% de una desviación estándar más alta que la de los estudiantes de escuelas públicas.

1. Diferencia en medias para el resultado

Esta diferencia en medias es estadísticamente significativa

```
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 < 2.2e-16
## alternative hypothesis: true difference in means 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
```

2. Diferencia en medias para las covariables

Se han seleccionado cinco covariables de la base de datos como variables explicativas, que se definen como el conjunto de covariables: `ecls_cov`

```
ecls_cov <- c('race_white', 'p5hmage', 'w3income',
              'p5numpla', 'w3momed_hsb')
ecls %>%
  group_by(catholic) %>%
  select(one_of(ecls_cov)) %>%
  summarise_all(funs(mean(., na.rm = T))) %>%
kable()
```

catholic	race_white	p5hmage	w3income	p5numpla	w3momed_hsb
0	0.5561246	37.56097	54889.16	1.132669	0.4640918
1	0.7251656	39.57516	82074.30	1.092701	0.2272069

2. Diferencia en medias para las covariables

Se pueden comprobar si estas diferencias por grupo son significativas mediante la prueba de hipótesis t apropiada.

Para no repetir el procedimiento varias veces, se puede utilizar la función `lapply` que permite aplicar la misma instrucción a una lista, por ejemplo, para nuestro conjunto de covariables:

```
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 < 2.2e-16
## alternative hypothesis: true difference in means 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. Diferencia en medias para las covariables

```
## [[2]]
##
##  Welch Two Sample t-test
##
## data:  ecls[, v] by ecls[, "catholic"]
## t = -12.665, df = 2186.9, p-value < 2.2e-16
## alternative hypothesis: true difference in means 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 < 2.2e-16
## alternative hypothesis: true difference in means 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
##
```

2. Diferencia en medias para las covariables

```
## [[4]]
##
##  Welch Two Sample t-test
##
## data:  ecls[, v] by ecls[, "catholic"]
## t = 4.2458, df = 2233.7, p-value = 2.267e-05
## alternative hypothesis: true difference in means 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:  ecls[, v] by ecls[, "catholic"]
## t = 18.855, df = 2107.3, p-value < 2.2e-16
## alternative hypothesis: true difference in means 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
```

3. Estimación del *propensity score*

Estimaremos el *propensity score* mediante un modelo logit, donde la variable dependiente es la dicotómica del tratamiento y como independientes las covariables seleccionadas.

Utilizamos la función de un modelo lineal generalizado para la estimación del modelo logit: `glm`. La variable del ingreso se convierte a una varible en miles.

```
ecls <- ecls %>% mutate(w3income_1k = w3income / 1000)
m_ps <- glm(catholic ~ race_white + w3income_1k +
              p5hmage + p5numpla + w3momed_hsb,
            family = binomial(), data = ecls)
summary(m_ps)
```

3. Estimación del *propensity score*

```
##
## Call:
## glm(formula = catholic ~ race_white + w3income_1k + p5hmage +
##     p5numpla + w3momed_hsb, family = binomial(), data = ecls)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1883  -0.6140  -0.4508  -0.3336   2.5659
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.2125519  0.2379826 -13.499  < 2e-16 ***
## race_white   0.3145014  0.0700895   4.487 7.22e-06 ***
## w3income_1k  0.0073038  0.0006495  11.245  < 2e-16 ***
## p5hmage      0.0292168  0.0050771   5.755 8.69e-09 ***
## p5numpla    -0.1439392  0.0912255  -1.578    0.115
## w3momed_hsb -0.6935868  0.0743207  -9.332  < 2e-16 ***
## ---
## 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
```

3. Estimación del *propensity score*

Usando este modelo, ahora calculamos el *propensity score* para cada estudiante, es decir, la probabilidad predicha para cada estudiante de ser tratado dadas las estimaciones del modelo logit.

```
prs_df<-data.frame(pr_score=predict(m_ps,type ="response"),
                   catholic = m_ps$model$catholic)
head(prs_df)
```

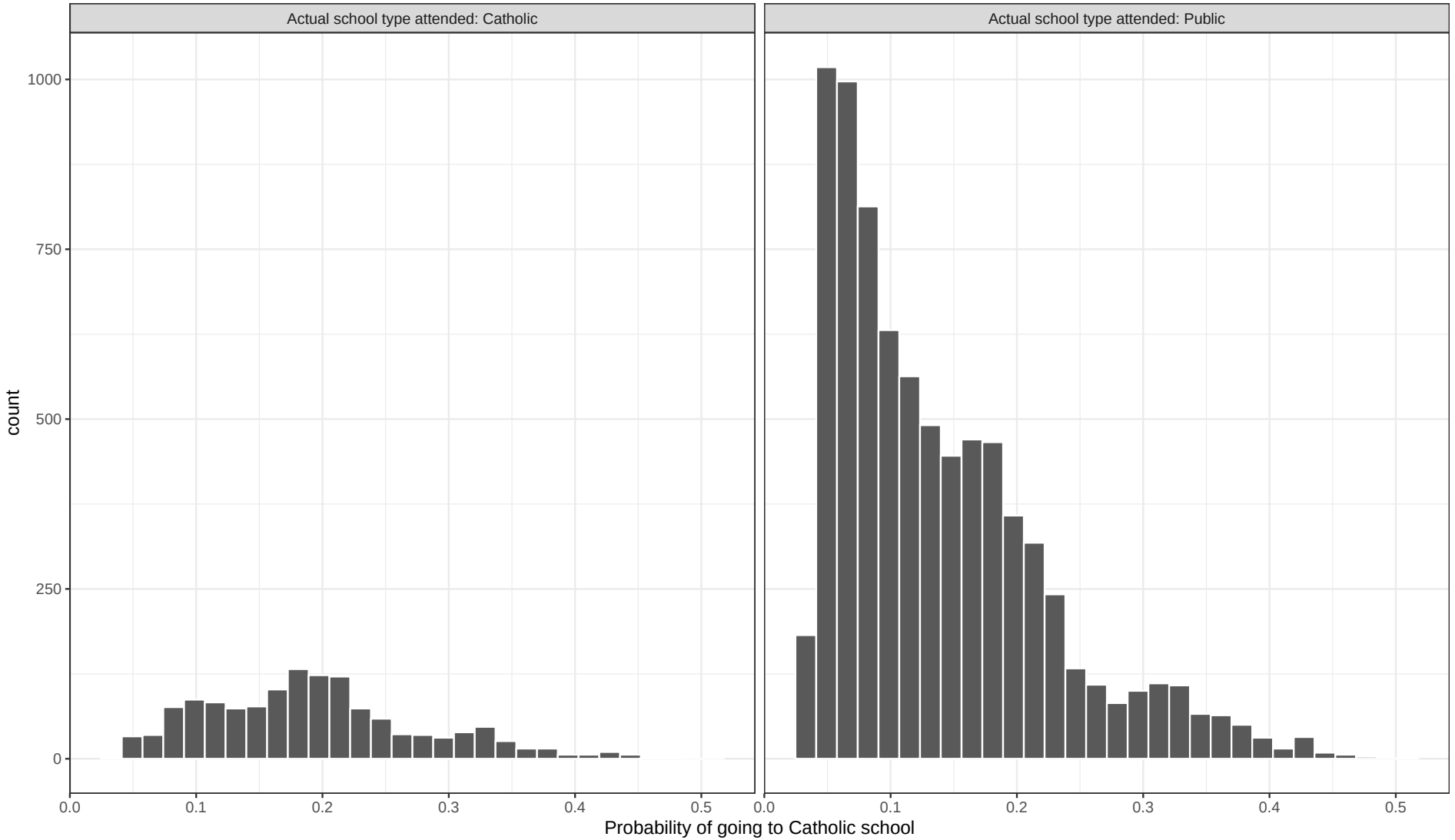
```
##    pr_score catholic
## 1 0.2292928        0
## 2 0.1801360        0
## 4 0.2092957        0
## 5 0.2154022        1
## 6 0.3604931        0
## 7 0.1080608        0
```

4. Analizar la región de *commun support*

Luego de estimar el *propensity score*, graficamos su histograma para el grupo de tratamiento y control:

```
labs <- paste("Actual school type attended:",
              c("Catholic", "Public"))
prs_df %>%
  mutate(catholic=ifelse(catholic==1,labs[1],labs[2])) %>%
  ggplot(aes(x = pr_score)) +
  geom_histogram(color = "white") +
  facet_wrap(~catholic) +
  xlab("Probability of going to Catholic school") +
  theme_bw()
```

4. Analizar la región de *commun support*



5. *Matching* con el criterio de la vecindad más cercana

Para realizarlo, es necesario omitir las observaciones incompletas:

```
ecls_nomiss <- ecls %>%
  select(c5r2mtsc_std, catholic, one_of(ecls_cov)) %>%
  na.omit()
```

Y utilizamos las funciones propias del paquete `MatchIt`:

```
mod_match <- matchit(catholic ~ race_white + w3income
                     + p5hmage + p5numpla + w3momed_hsb,
                     method = "nearest", data = ecls_nomiss)
```

Se puede generar un nuevo *dataframe* con el resultado:

```
dta_m <- match.data(mod_match)
dim(dta_m)
```

5. *Matching* con el criterio de la vecindad más cercana

```
head(dta_m)
```

```
##   c5r2mtsc_std catholic race_white p5hmage w3income p5numpla w3momed_hsb
## 1    0.9817533        0          1      47  62500.5        1            0
## 2    0.5943775        0          1      41  45000.5        1            0
## 4    0.4906106        0          1      43  62500.5        1            0
## 5    1.4512779        1          1      38  87500.5        1            0
## 6    2.5956991        0          1      47 150000.5        1            0
## 8    0.3851966        0          1      41  62500.5        1            0
##    distance weights subclass
## 1 0.2292928       1      912
## 2 0.1801360       1       56
## 4 0.2092957       1     1344
## 5 0.2154022       1      671
## 6 0.3604931       1      881
## 8 0.1997897       1      128
```

5. *Matching* con el criterio de la vecindad más cercana

```
summary(mod_match)
```

```
Call:
matchit(formula = catholic ~ race_white + w3income + p5hmage +
    p5numpla + w3momed_hsb, data = ecls_nomiss, method = "nearest")

Summary of Balance for All Data:
             Means Treated Means Control Std. Mean Diff. Var. Ratio eCDF Mean eCDF Max
distance             0.1927        0.1379          0.6486     1.0007    0.2086   0.3109
race_white           0.7411        0.5914          0.3418          .    0.1497   0.1497
w3income          82568.9357    55485.0210          0.5777     1.1373    0.1565   0.3062
p5hmage              39.5932       37.5658          0.3874     0.6383    0.0408   0.1893
p5numpla              1.0917        1.1298         -0.1242     0.6132    0.0076   0.0277
w3momed_hsb           0.2234        0.4609         -0.5703          .    0.2375   0.2375


Summary of Balance for Matched Data:
             Means Treated Means Control Std. Mean Diff. Var. Ratio eCDF Mean eCDF Max Std. Pair Dist.
distance             0.1927        0.1927          0.0000     1.0000    0.0000   0.0030          0.0002
race_white           0.7411        0.7470         -0.0135          .    0.0059   0.0059          0.0811
w3income          82568.9357    81403.9926          0.0248     1.0114    0.0059   0.0118          0.0536
p5hmage              39.5932       39.5503          0.0082     1.0036    0.0016   0.0059          0.1416
p5numpla              1.0917        1.0762          0.0507     1.0627    0.0040   0.0163          0.1184
w3momed_hsb           0.2234        0.2152          0.0195          .    0.0081   0.0081          0.0728
```

5. *Matching* con el criterio de la vecindad más cercana

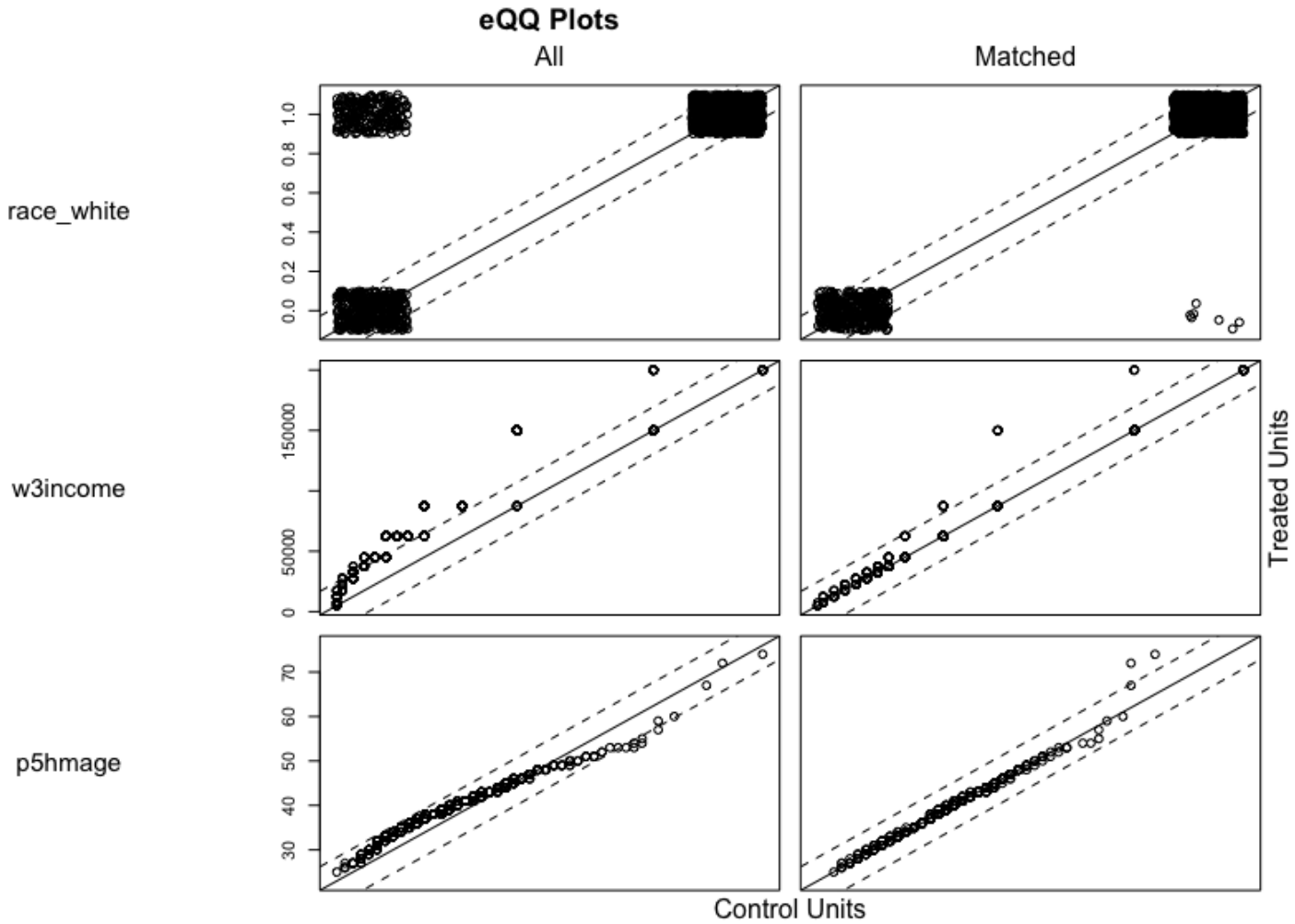
```
summary(mod_match)
```

```
Percent Balance Improvement:
            Std. Mean Diff. Var. Ratio eCDF Mean eCDF Max
distance             100.0       94.7     100.0     99.0
race_white            96.0          .      96.0     96.0
w3income              95.7       91.2      96.2     96.1
p5hmage               97.9       99.2      96.1     96.9
p5numpla              59.2       87.6      47.5     41.2
w3momed_hsb           96.6          .      96.6     96.6

Sample Sizes:
          Control Treated
All          7915    1352
Matched      1352    1352
Unmatched    6563       0
Discarded       0       0
```

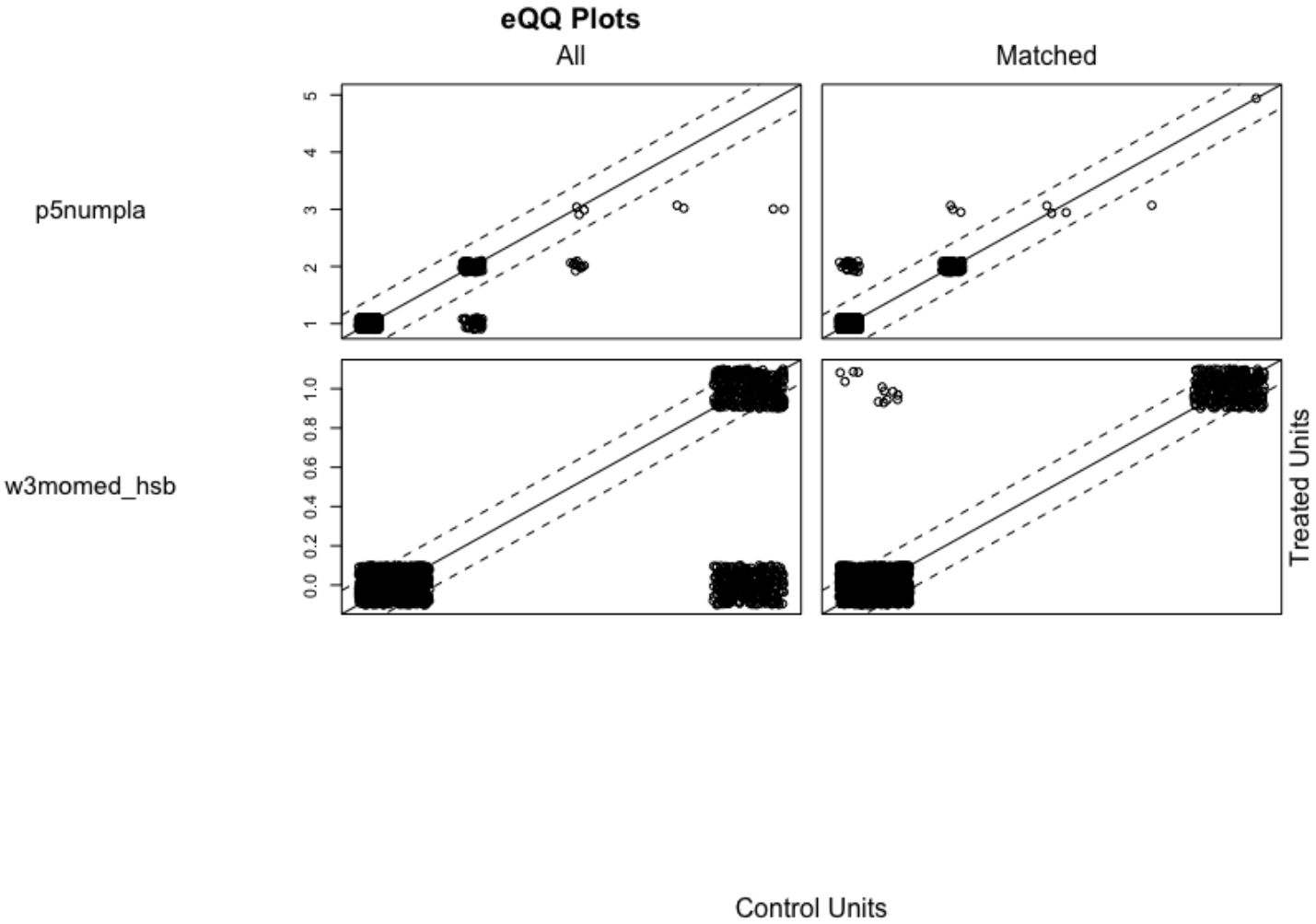
5. *Matching* con el criterio de la vecindad más cercana

```
plot(mod_match)
```



5. *Matching* con el criterio de la vecindad más cercana

```
plot(mod_match)
```



6. Equilibrio de covariables después del *matching*

```
dta_m %>%
  group_by(catholic) %>%
  select(one_of(ecls_cov)) %>%
  summarise_all(funs(mean)) %>%
  kable()
```

catholic	race_white	p5hmage	w3income	p5numpla	w3momed_hsb
0	0.7470414	39.5503	81403.99	1.076183	0.2152367
1	0.7411243	39.5932	82568.94	1.091716	0.2233728

Como antes, se puede hacer la prueba estadística apropiada, no debe rechazarse ninguna de las hipótesis nulas ahora:

```
lapply(ecls_cov,
    function(v) { t.test(dta_m[, v] ~ dta_m$catholic)})
```

6. Equilibrio de covariables después del *matching*

```
[[1]]

        Welch Two Sample t-test

data:  dta_m[, v] by dta_m$catholic
t = 0.35243, df = 2701.8, p-value = 0.7245
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.02700440  0.03883872
sample estimates:
mean in group 0 mean in group 1
      0.7470414       0.7411243


[[2]]

        Welch Two Sample t-test

data:  dta_m[, v] by dta_m$catholic
t = -0.21331, df = 2702, p-value = 0.8311
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.4372485  0.3514496
sample estimates:
mean in group 0 mean in group 1
        39.5503         39.5932
```

6. Equilibrio de covariables después del *matching*

```
[[3]]

        Welch Two Sample t-test

data:  dta_m[, v] by dta_m$catholic
t = -0.64787, df = 2701.9, p-value = 0.5171
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -4690.731  2360.845
sample estimates:
mean in group 0 mean in group 1
       81403.99        82568.94


[[4]]

        Welch Two Sample t-test

data:  dta_m[, v] by dta_m$catholic
t = -1.339, df = 2699.5, p-value = 0.1807
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.038278301  0.007213213
sample estimates:
mean in group 0 mean in group 1
       1.076183        1.091716
```

6. Equilibrio de covariables después del *matching*

```
[[5]]

        Welch Two Sample t-test

data:  dta_m[, v] by dta_m$catholic
t = -0.51108, df = 2701.5, p-value = 0.6093
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.03935185  0.02307966
sample estimates:
mean in group 0 mean in group 1
      0.2152367       0.2233728
```

7. Estimación del efecto de tratamiento

Para la estimación del efecto del tratamiento simplemente se puede realizar una prueba de comparación de medias en el grupo posterior al *matching*:

```
with(dta_m, t.test(c5r2mtsc_std ~ catholic))
```

```
##
##  Welch Two Sample t-test
##
## data:  c5r2mtsc_std by catholic
## t = 4.2645, df = 2676.3, p-value = 2.073e-05
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.08086619 0.21853174
## sample estimates:
## mean in group 0 mean in group 1
##       0.3593668       0.2096679
```

7. Estimación del efecto de tratamiento

O bien, mediante la estimación de la recta de regresión por MCO con o sin covariantes:

```
lm_treat1 <- lm(c5r2mtsc_std ~ catholic, data = dta_m)
summary(lm_treat1)
```

```
##
## Call:
## lm(formula = c5r2mtsc_std ~ catholic, data = dta_m)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5110 -0.5670  0.0574  0.6087  2.8456
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.35937    0.02482  14.478  < 2e-16 ***
## catholic    -0.14970    0.03510  -4.264 2.07e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9127 on 2702 degrees of freedom
## Multiple R-squared:  0.006686,   Adjusted R-squared:  0.006318
## F-statistic: 18.19 on 1 and 2702 DF,  p-value: 2.072e-05
```

7. Estimación del efecto de tratamiento

```
lm_treat2 <- lm(c5r2mtsc_std ~ catholic + race_white +
    p5hmage+I(w3income/10^3)+p5numpla+w3momed_hsb,data=dta_m)
summary(lm_treat2)
```

```
##
## Call:
## lm(formula = c5r2mtsc_std ~ catholic + race_white + p5hmage +
##     I(w3income/10^3) + p5numpla + w3momed_hsb, data = dta_m)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5663 -0.5439  0.0439  0.5990  2.7409
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -0.4822479  0.1482211  -3.254 0.001154 **
## catholic         -0.1485468  0.0331477  -4.481 7.73e-06 ***
## race_white        0.3097157  0.0385640   8.031 1.43e-15 ***
## p5hmage           0.0119232  0.0032307   3.691 0.000228 ***
## I(w3income/10^3)  0.0031910  0.0003752   8.504  < 2e-16 ***
## p5numpla         -0.0409054  0.0552001  -0.741 0.458734
## w3momed_hsb      -0.3580381  0.0413037  -8.668  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8613 on 2697 degrees of freedom
## Multiple R-squared:  0.1169, Adjusted R-squared:  0.115
```