Selection on the Observables

July 3, 2023

In this problem set, we will use *selection on the observables* to identify the effect of legislative elections on the use of mass repression in authoritarian regimes.

Before you start, please download the dataset from Moodle and import it into `RStudio`. If you would like to submit this problem set, please complete the questions at the end.

1 Variables

For this exercise, we will use the following variables retrieved from a variety of cross-national datasets on civil liberties (Freedom House, see https://freedomhouse.org/report/freedom-world) and socioeconomic development.

The key **outcome** variable of interest is `fh_CL` – a rating of civil liberties given by the Freedom House for all post-Cold War authoritarian regimes.

The **treatment** of current interest is `leg_elec` – a binary variable to indicate whether the authoritarian regime allows election or not.

We will consider the following confounders:

- `lg_fh_CL` – the outcome variable lagged by one year.
- `lg_epr_gdpcapl` – lagged GDP per capita (logged).
- `lg_grow` – lagged economic growth rate.
- `ross_population` – population size.
- `epr_ethfrac` – the index of ethnolinguistic fractionalization.
- `arc_turn` – a binary variable to indicate whether the country experienced leadership turnover in the prior year.

1.1 Load Packages

Other than the packages we have used before, we will need the following packages, we will need `Matching` and `ebal` to carry out the matching estimation.

```
library(ggplot2)
library(stargazer)
library(tidyverse)
library(Matching)
```

```
Warning: package 'Matching' was built under R version 4.3.1
```

```
library(ebal)
```

1.2 Read Data

Here we can use four functions to take a peak at the dataset.

```
names(dta_sel) # list all columns (variables)
ls(dta_sel) # list all columns (variables) alphabetically
summary(dta_sel) # show summary statistics
head(dta_sel) # show first 6 rows
```

1.3 Disable Scientific Notation (Optional)

We will also need to use `options` at the very beginning to disable print out our results in scientific notation.

```
options(scipen=999)
```

2 OLS

```
mod_ols_1 <- lm(fh_CL ~ leg_elec +
  lg_epr_gdpcapl + lg_grow +
  ross_population + epr_ethfrac + arc_turn, data=dta_sel)
mod_ols_2 <- lm(fh_CL ~ lg_fh_CL +
  leg_elec + lg_epr_gdpcapl + lg_grow +
  ross_population + epr_ethfrac + arc_turn, data=dta_sel)
```

```r
stargazer(list(mod_ols_1, mod_ols_2),
          omit.stat = c("f", "rsq", "ser"),
          covariate.labels = c("Lagged civil liberties",
                               "Legislative election (=1)",
                               "GDP per capita",
                               "Economic growth",
                               "Population",
                               "Ethnic diversity",
                               "Leadership turnover"),
          omit = c("as.factor"),
          type = "text",
          digits = 3,
          no.space = T,
          intercept.bottom = TRUE,
          star.cutoffs = c(0.05, 0.01, 0.001))
```

```
=============================================================
                              Dependent variable:
                          ---------------------------
                                      fh_CL
                                (1)             (2)
-------------------------------------------------------------
Lagged civil liberties                          0.895***
                                                (0.013)
Legislative election (=1)     -0.713***         -0.072*
                               (0.075)          (0.035)
GDP per capita                -0.103***          0.0004
                               (0.030)          (0.013)
Economic growth                -0.009            0.003
                               (0.028)          (0.012)
Population                    0.203***           0.019
                               (0.024)          (0.011)
Ethnic diversity              -1.272***          -0.116
                               (0.129)          (0.060)
Leadership turnover            -0.013            -0.004
                               (0.144)          (0.064)
Constant                      3.095***           0.324
                               (0.406)          (0.185)
-------------------------------------------------------------
```

```
Observations                    1,205          1,205
Adjusted R2                     0.202          0.842
=================================================================
Note:                         *p<0.05; **p<0.01; ***p<0.001
```

3 Matching

The strategy of selection on the observables (SOO) is very similar to multiple regression analysis, as it rests upon the assumption of **conditional ignorability** – that is, we assume that conditional on some **pre-treatment observable covariates**, whether or not an observation will receive the treatment can be considered **as if** random. This is a very strong assumption indeed.[1]

To implement SOO is to carry out **matching**. We will discuss the distinction between SOO and multiple regression in class, but here we provide the intuition that SOO usually will give us **ATT**, as we are trying to use matched observations to approximate the potential outcomes of treated units. Below we will carry out the estimation step by step.

3.1 Step 1: Verify the treatment assignment is not random

```r
pre_balance <- lm(leg_elec ~ lg_fh_CL + lg_epr_gdpcapl + lg_grow + ross_population + epr
summary(pre_balance)
```

```
Call:
lm(formula = leg_elec ~ lg_fh_CL + lg_epr_gdpcapl + lg_grow +
    ross_population + epr_ethfrac + arc_turn, data = dta_sel)

Residuals:
     Min       1Q   Median       3Q      Max
-1.09085 -0.08459  0.18796  0.23287  0.50595

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     1.180074   0.150291   7.852 9.04e-15 ***
lg_fh_CL       -0.098054   0.010311  -9.510  < 2e-16 ***
lg_epr_gdpcapl  0.006644   0.011116   0.598   0.5502
```

[1] Another assumption for SOO is common support, which says for a given pretreatment covariate the probability of a unit receiving the treatment is always larger than zero.

```
lg_grow          -0.026138   0.010285  -2.541   0.0112 *
ross_population   0.004259   0.009003   0.473   0.6362
epr_ethfrac       0.054761   0.049534   1.106   0.2692
arc_turn         -0.078664   0.053224  -1.478   0.1397
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4009 on 1198 degrees of freedom
Multiple R-squared:  0.0928,    Adjusted R-squared:  0.08826
F-statistic: 20.42 on 6 and 1198 DF,  p-value: < 2.2e-16
```

3.2 Step 2: Study pre-matching balance between the treatment and control groups

```
vars <- c("Lagged civil liberties", "GDP per capita", "Economic growth", "Population", "
mb <- MatchBalance(leg_elec ~ lg_fh_CL + lg_epr_gdpcapl + lg_grow + ross_population + ep
```

```
***** (V1) lg_fh_CL *****
before matching:
mean treatment........ 4.9699
mean control.......... 5.8145
std mean diff......... -71.153

mean raw eQQ diff..... 0.84727
med  raw eQQ diff..... 1
max  raw eQQ diff..... 2

mean eCDF diff........ 0.14078
med  eCDF diff........ 0.12098
max  eCDF diff........ 0.35844

var ratio (Tr/Co)..... 1.216
T-test p-value........ < 2.22e-16
KS Bootstrap p-value.. < 2.22e-16
KS Naive p-value...... 0
KS Statistic.......... 0.35844


***** (V2) lg_epr_gdpcapl *****
```

```
before matching:
mean treatment........ 0.88014
mean control.......... 0.85946
std mean diff......... 1.9006

mean raw eQQ diff..... 0.37904
med  raw eQQ diff..... 0.34256
max  raw eQQ diff..... 1.4798

mean eCDF diff........ 0.078207
med  eCDF diff........ 0.059609
max  eCDF diff........ 0.18839

var ratio (Tr/Co)..... 0.53
T-test p-value........ 0.83118
KS Bootstrap p-value.. < 2.22e-16
KS Naive p-value...... 5.7347e-07
KS Statistic.......... 0.18839


***** (V3) lg_grow *****
before matching:
mean treatment........ -0.032755
mean control.......... 0.16198
std mean diff......... -20.007

mean raw eQQ diff..... 0.25623
med  raw eQQ diff..... 0.010861
max  raw eQQ diff..... 20.877

mean eCDF diff........ 0.024412
med  eCDF diff........ 0.025513
max  eCDF diff........ 0.062972

var ratio (Tr/Co)..... 0.41097
T-test p-value........ 0.045389
KS Bootstrap p-value.. 0.356
KS Naive p-value...... 0.36916
KS Statistic.......... 0.062972
```

```
***** (V4) ross_population *****
before matching:
mean treatment........ 16.246
mean control.......... 16.453
std mean diff......... -16.666

mean raw eQQ diff..... 0.33291
med  raw eQQ diff..... 0.20777
max  raw eQQ diff..... 2.6603

mean eCDF diff........ 0.031963
med  eCDF diff........ 0.025181
max  eCDF diff........ 0.10878

var ratio (Tr/Co)..... 0.58675
T-test p-value........ 0.051421
KS Bootstrap p-value.. 0.018
KS Naive p-value...... 0.013173
KS Statistic.......... 0.10878


***** (V5) epr_ethfrac *****
before matching:
mean treatment........ 0.50722
mean control.......... 0.44025
std mean diff......... 25.086

mean raw eQQ diff..... 0.079443
med  raw eQQ diff..... 0.056465
max  raw eQQ diff..... 0.24752

mean eCDF diff........ 0.076172
med  eCDF diff........ 0.064536
max  eCDF diff........ 0.22002

var ratio (Tr/Co)..... 0.81065
T-test p-value........ 0.00083962
KS Bootstrap p-value.. < 2.22e-16
KS Naive p-value...... 2.3818e-09
KS Statistic.......... 0.22002
```

```
***** (V6) arc_turn *****
before matching:
mean treatment........ 0.044086
mean control.......... 0.069091
std mean diff......... -12.174

mean raw eQQ diff..... 0.025455
med  raw eQQ diff..... 0
max  raw eQQ diff..... 1

mean eCDF diff........ 0.012502
med  eCDF diff........ 0.012502
max  eCDF diff........ 0.025005

var ratio (Tr/Co)..... 0.65355
T-test p-value........ 0.13598


Before Matching Minimum p.value: < 2.22e-16
Variable Name(s): lg_fh_CL lg_epr_gdpcapl epr_ethfrac  Number(s): 1 2 5
```

```
btest <- baltest.collect(mb, var.names=vars, after=F)
round(btest[, c("mean.Tr","mean.Co","T pval")], 3)
```

```
                      mean.Tr mean.Co T pval
Lagged civil liberties   4.970   5.815  0.000
GDP per capita           0.880   0.859  0.831
Economic growth         -0.033   0.162  0.045
Population              16.246  16.453  0.051
EFL                      0.507   0.440  0.001
Leadership turnover      0.044   0.069  0.136
```

3.3 Step 3: Carry out (bias-adjusted) matching

```
matchout <- Match(Y=dta_sel[,2], Tr=dta_sel[,1], X=dta_sel[,3:8], M=5, exact=rep(FALSE,
```

```
Warning in Match(Y = dta_sel[, 2], Tr = dta_sel[, 1], X = dta_sel[, 3:8], :
length of exact != ncol(X). Ignoring exact option
```

```r
summary(matchout)
```

```
Estimate...  -0.15154
AI SE......  0.049676
T-stat.....  -3.0506
p.val......  0.002284

Original number of observations..............  1205
Original number of treated obs...............  930
Matched number of observations...............  930
Matched number of observations  (unweighted).  4650
```

3.4 Step 4: Examine post-matching balance

```r
vars <- c("Lagged civil liberties", "GDP per capita", "Economic growth", "Population",
mb.out <- MatchBalance(match.out=matchout,
                       leg_elec ~ lg_fh_CL + lg_epr_gdpcapl + lg_grow + ross_population
```

```
***** (V1) lg_fh_CL *****
                       Before Matching       After Matching
mean treatment........     4.9699             4.9699
mean control.........      5.8145             5.1127
std mean diff.........    -71.153           -12.029

mean raw eQQ diff.....    0.84727            0.1428
med  raw eQQ diff.....          1                 0
max  raw eQQ diff.....          2                 1

mean eCDF diff........    0.14078          0.023799
med  eCDF diff........    0.12098          0.017419
max  eCDF diff........    0.35844          0.072043

var ratio (Tr/Co).....      1.216            1.0583
T-test p-value........ < 2.22e-16         1.573e-12
KS Bootstrap p-value.. < 2.22e-16        < 2.22e-16
KS Naive p-value...... < 2.22e-16        6.6007e-11
KS Statistic..........    0.35844          0.072043
```

```
***** (V2) lg_epr_gdpcapl *****
                        Before Matching       After Matching
mean treatment........    0.88014           0.88014
mean control..........    0.85946           0.70007
std mean diff.........     1.9006            16.549

mean raw eQQ diff.....    0.37904           0.23804
med  raw eQQ diff.....    0.34256           0.22201
max  raw eQQ diff.....     1.4798           0.90376

mean eCDF diff........   0.078207          0.059422
med  eCDF diff........   0.059609          0.066022
max  eCDF diff........    0.18839           0.13204

var ratio (Tr/Co).....       0.53           0.94118
T-test p-value........    0.83118        < 2.22e-16
KS Bootstrap p-value.. < 2.22e-16        < 2.22e-16
KS Naive p-value...... 5.7347e-07        < 2.22e-16
KS Statistic..........    0.18839           0.13204


***** (V3) lg_grow *****
                        Before Matching       After Matching
mean treatment........  -0.032755         -0.032755
mean control..........    0.16198        -0.00096036
std mean diff.........    -20.007           -3.2665

mean raw eQQ diff.....    0.25623          0.084109
med  raw eQQ diff.....   0.010861          0.013848
max  raw eQQ diff.....     20.877            20.877

mean eCDF diff........   0.024412          0.047133
med  eCDF diff........   0.025513          0.036237
max  eCDF diff........   0.062972            0.1357

var ratio (Tr/Co).....    0.41097             10.3
T-test p-value........   0.045389          0.27732
KS Bootstrap p-value..      0.362        < 2.22e-16
KS Naive p-value......    0.36916        < 2.22e-16
```

```
KS Statistic.........   0.062972           0.1357


***** (V4) ross_population *****
                       Before Matching       After Matching
mean treatment........     16.246             16.246
mean control.........     16.453             16.115
std mean diff.........    -16.666             10.484

mean raw eQQ diff.....    0.33291           0.22652
med  raw eQQ diff.....    0.20777           0.15916
max  raw eQQ diff.....     2.6603            1.8476

mean eCDF diff........   0.031963          0.043499
med  eCDF diff........   0.025181          0.042796
max  eCDF diff........    0.10878           0.12258

var ratio (Tr/Co).....    0.58675             1.4376
T-test p-value........   0.051421        3.0581e-08
KS Bootstrap p-value..      0.012        < 2.22e-16
KS Naive p-value......   0.013173        < 2.22e-16
KS Statistic.........    0.10878           0.12258


***** (V5) epr_ethfrac *****
                       Before Matching       After Matching
mean treatment........    0.50722           0.50722
mean control.........    0.44025           0.52214
std mean diff.........     25.086           -5.5854

mean raw eQQ diff.....   0.079443          0.025859
med  raw eQQ diff.....   0.056465          0.026435
max  raw eQQ diff.....    0.24752          0.099856

mean eCDF diff........   0.076172           0.02817
med  eCDF diff........   0.064536          0.026452
max  eCDF diff........    0.22002          0.068387

var ratio (Tr/Co).....    0.81065             1.0633
T-test p-value........ 0.00083962        0.00084218
KS Bootstrap p-value.. < 2.22e-16        < 2.22e-16
```

```
KS Naive p-value..... 2.3818e-09        7.1843e-10
KS Statistic.........    0.22002          0.068387


***** (V6) arc_turn *****
                       Before Matching       After Matching
mean treatment........   0.044086          0.044086
mean control..........   0.069091          0.044301
std mean diff.........    -12.174           -0.1047

mean raw eQQ diff.....   0.025455        0.00021505
med  raw eQQ diff.....          0                 0
max  raw eQQ diff.....          1                 1

mean eCDF diff........   0.012502        0.00010753
med  eCDF diff........   0.012502        0.00010753
max  eCDF diff........   0.025005        0.00021505

var ratio (Tr/Co).....    0.65355           0.99537
T-test p-value........    0.13598           0.65479


Before Matching Minimum p.value: < 2.22e-16
Variable Name(s): lg_fh_CL lg_epr_gdpcapl epr_ethfrac  Number(s): 1 2 5

After Matching Minimum p.value: < 2.22e-16
Variable Name(s): lg_fh_CL lg_epr_gdpcapl lg_grow ross_population epr_ethfrac  Number(s):
```

```r
btest_after <- baltest.collect(mb.out, var.names=vars, after=T)
round(btest_after[,c("mean.Tr","mean.Co","T pval")], 3)
```

```
                     mean.Tr mean.Co T pval
Lagged civil liberties   4.970   5.113  0.000
GDP per capita           0.880   0.700  0.000
Economic growth         -0.033  -0.001  0.277
Population              16.246  16.115  0.000
EFL                      0.507   0.522  0.001
Leadership turnover      0.044   0.044  0.655
```

4 Matching by Propensity Score

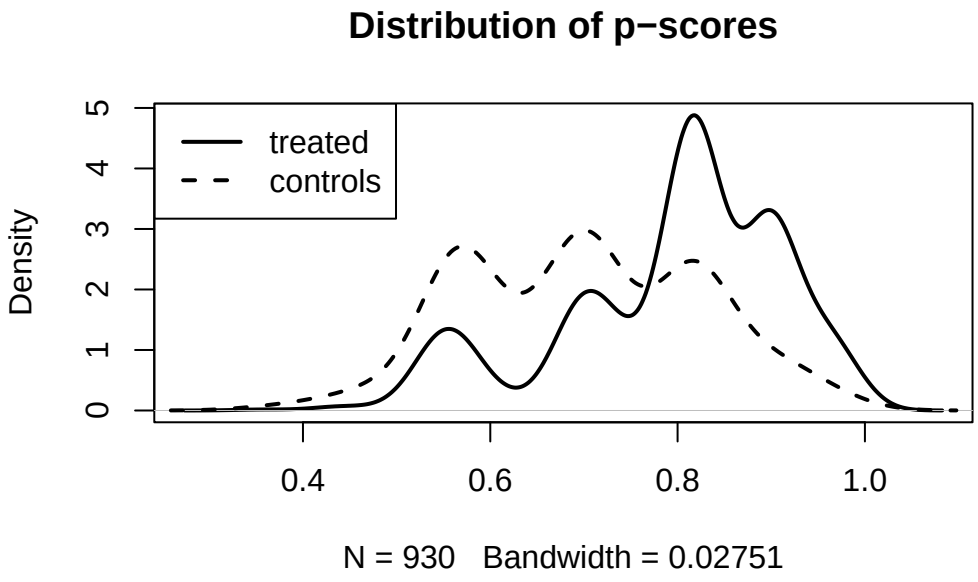
Rather than using the values of individual confounders or covariates to carry out the matching estimation, we can also use **propensity score** to match treated and control units. To do so, we will first need to estimate the probability that a unit will receive the treatment, using the covariates we have just specified.

4.1 Step 1: Estimating the propensity score

```
pi.out <- glm(leg_elec ~ lg_fh_CL + lg_epr_gdpcapl + lg_grow + ross_population + epr_etl
              data=dta_sel, family=binomial(link="probit"))
#pi.out$fit
```

4.2 Step 1b: Compare the density plot of propensity scores for treatment and control groups

```
plot(density(pi.out$fit[dta_sel$leg_elec==1]), lwd=2,
  main="Distribution of p-scores")
lines(density(pi.out$fit[dta_sel$leg_elec==0]), lwd=2, lty=2)
legend("topleft", legend=c("treated","controls"), lty=c(1,2), lwd=2)
```



4.3 Step 2: Matching

```
matchout.pi <- Match(Y=dta_sel$fh_CL, Tr=dta_sel$leg_elec, X=pi.out$fit,
                     M=5, exact=FALSE, estimand="ATT", BiasAdjust=T)
summary(matchout.pi)
```

```
Estimate...  -0.11356
AI SE......  0.05748
T-stat.....  -1.9757
p.val......  0.048192

Original number of observations..............  1205
Original number of treated obs...............  930
Matched number of observations...............  930
Matched number of observations  (unweighted).  4791

Number of obs dropped by 'exact' or 'caliper'  0
```

4.4 Step 3: Examine post-matching balance

```
vars <- c("Lagged civil liberties", "GDP per capita", "Economic growth", "Population",
mb.out.pi <- MatchBalance(match.out=matchout.pi,
                          leg_elec ~ lg_fh_CL + lg_epr_gdpcapl + lg_grow + ross_populat
btest_after <- baltest.collect(mb.out.pi, var.names=vars, after=T)
round(btest_after[,c("mean.Tr","mean.Co","T pval")], 3)
```