Actual matching for principal turnover

Ham and Miratrix

2023-07-25

The following is code to actually conduct the matching in order to obtain point estimates of the naive DiD and matching estimators. Used only for illustrative purposes.

Load data

```
dat = read_csv( here::here("../data/cleaned_data.csv" ),
                show_col_types = FALSE )
```

```
## New names:
## * `` -> `...1`
```

```
c_vars = c( "ssize_1000" , "savg_frpl0" , "savg_hisp0" , "savg_black0" ,
            "prop_new" , "principal_yrs" , "principal_transition")

names(dat)
```

```
##  [1] "...1"                 "year"                 "school_id"
##  [4] "year0"                "savg_math0"           "ssize_1000"
##  [7] "savg_frpl0"           "savg_hisp0"           "savg_black0"
## [10] "savg_math1"           "savg_math2"           "savg_math3"
## [13] "savg_math4"           "savg_math5"           "treat"
## [16] "prop_new"             "principal_yrs"        "principal_transition"
## [19] "district_id"          "savg_read"            "savg_math"
```

```
# Our years are number of lags, so 5 is the furthest in the past year.
pre_years = paste0( "savg_math", 5:0 )
pre_years
```

```
## [1] "savg_math5" "savg_math4" "savg_math3" "savg_math2" "savg_math1"
## [6] "savg_math0"
```

```
# This is the outcome after treatment
tx_year = "savg_math"
```

Drop all 0s in the outcomes

```
head(dat)
```

```
## # A tibble: 6 x 21
##    ...1  year school_id year0 savg_math0 ssize_1000 savg_frpl0 savg_hisp0
##   <dbl> <dbl>     <dbl> <dbl>      <dbl>      <dbl>      <dbl>      <dbl>
## 1     1  2002      1000  2001      0.858       3.39      0.289    0.0236
## 2     2  2002      1001  2001     -0.287       2.65      0.653    0.00755
## 3     3  2002      1002  2001     -0.732       1.65      0.867    0.0485
## 4     4  2002      1003  2001     -0.783       2.26      0.858    0.0841
## 5     5  2002      1004  2001     -0.627       5.59      0.735    0.0340
## 6     6  2002      1005  2001      1.06        4.55      0.174    0.0132
## # i 13 more variables: savg_black0 <dbl>, savg_math1 <dbl>, savg_math2 <dbl>,
## #   savg_math3 <dbl>, savg_math4 <dbl>, savg_math5 <dbl>, treat <dbl>,
## #   prop_new <dbl>, principal_yrs <dbl>, principal_transition <dbl>,
## #   district_id <dbl>, savg_read <dbl>, savg_math <dbl>
```

```
maths = which( str_detect( names(dat), "savg_math",  ) )
for ( m in maths ) {
    zeros = dat[[m]] == 0
    dat[zeros,m] = NA
}
nrow(dat)
```

```
## [1] 24628
```

```
dat = na.omit( dat )
nrow(dat)
```

```
## [1] 19651
```

Naive DiD Estimates

```
trt = filter( dat, treat == 1 )
ctrl = filter( dat, treat == 0 )
final_df = as.data.frame(dat)

(mean(trt$savg_math) - mean(ctrl$savg_math) ) -
    (mean(trt$savg_math0) - mean(ctrl$savg_math0) )
```

```
## [1] -0.02303342
```

DiD Estimates while matching on X

```
library(MatchIt)
rownames(final_df) = 1:nrow(final_df)
matching = matchit(treat ~ ssize_1000 + savg_frpl0 + savg_hisp0 +
                       savg_black0 + prop_new + principal_yrs + principal_transition,
                   data = final_df)

matched_controls = final_df[as.numeric(matching$match.matrix), ]

(mean(trt$savg_math) - mean(matched_controls$savg_math) ) - (mean(trt$savg_math0) - mean(matched_contro
```

```
## [1] -0.02869214
```

DiD Estimates while matching on additionally pre-treatment outcome

```
matching = matchit(treat ~ ssize_1000 + savg_frpl0 + savg_hisp0 + savg_black0 +
                       prop_new + principal_yrs + principal_transition +
                       savg_math0 + savg_math1 + savg_math2 + savg_math3 +
                       savg_math4 + savg_math5, data = final_df)

matched_controls = final_df[as.numeric(matching$match.matrix), ]

(mean(trt$savg_math) - mean(matched_controls$savg_math) ) -
    (mean(trt$savg_math0) - mean(matched_controls$savg_math0) )
```

```
## [1] -0.03306963
```