Handout 5 Difference in Diffrence

Chunyu Qu

11/21/2021

Causal Inference: Difference in Difference

In this handout we check the fundamentals of difference in difference method, with illustrating several different cases.

Two by Two DiD - Minimum Wage

1. Introduction and Data

1.1. Introduction This famous and milestone work in minimum wage has long been taken as an example of arguing the method of difference in difference. The controversial part of this study illustrate the points that we need to consider carefully well. We will use this as the example of 2×2 DiD design, considering the parallel trend and other endogeneoous issue.

The context is simple: NJ experienced a revision of minimum wage law on April, 1992, lifting up the rate from 4.25 USD/h to 5.05 USD/h while PA keeping still with its minimum wage rate. They showed that fast food restaurants in NJ increased employment by 13%, which viloates the common expecatation. To eliminate the endogenous issue, they compare the employment growth at stores in NJ that were initially paying much higher above the minimum wage line with the ones that at lower-wage stores. By illustrating that the employment rate change at those stores unaffected by the revision are identical in NJ and PA they verify the paralleled trend assumption.

1.2. Assumptions and Estimate Here is the 2×2 DiD. To evaluate the impact of a program or treatment on an outcome Y over a population of individuals, we set up two groups indexed by treatment status T = 0, 1 where 0 indicates individuals who do not receive treatment (control), and 1 indicates individuals who do receive treatment (Treatment). Assume that we observe individuals in two time periods, t = 0, 1 where 0 indicates a time period before the treatment group receives treatment (pre-treatment), and 1 indicates a time period after the treatment group receives treatment, (post-treatment).
Denote:
1. $\bar{Y_o}^T$ and $\bar{Y_1}^T$ be be the sample averages of the outcome for the treatment group before and after treatment.
2. $\bar{Y_o}^C$ and $\bar{Y_1}^C$ be be the sample averages of the outcome for the control group before and after treatment.
A classic specification of such an DiD is as

$$Y_i = \alpha + \beta T_i + \gamma t_i + \delta (T_i \times t_i) + \epsilon_i$$

where i = 1, 2, ... n denote the observations. Here,

- β is the treatment group specific effect (to account for average permanent differences between treatment and control)

- γ is the time trend common to control and treatment groups

- δ is the true effect of treatment

To make sure that the estimators are unbiased and consistent, besides the problem is correctly specified, we need the following key assumptions
1. $E[\epsilon] = 0$
2. Parallel Trend Assumption: $cov(\epsilon_i, T_i) = cov(\epsilon_i, t_i) = cov(\epsilon_i, T_i \times t_i) = 0$

Then based on our setup, we have that

- $E[\bar{Y_o}^T] = \alpha + \beta$

- $E[\bar{Y_o}^C] = \alpha$

- $E[\bar{Y_1}^T] = \alpha + \beta + \gamma + \delta$

- $E[\bar{Y_1}^C] = \alpha + \gamma$

To estimate the average difference in outcome before and after treatment in the treatment group alone, let $\hat{delta}_1 = \bar{Y_1}^T - \bar{Y_0}^T$
Then

$$E[\hat{\delta_1}] = E[\bar{Y_1}^T] - E[\bar{Y_0}^T] = [\alpha + \beta + \gamma + \delta] - [\alpha + \beta] = \gamma + \delta$$

Similarly for control before and after, we have

$$E[\hat{\delta_2}] = E[\bar{Y_1}^C] - E[\bar{Y_0}^C] = [\alpha + \gamma +] - \alpha = \gamma$$

Then the DiD estimator $\hat{\delta_{DD}} = \hat{delta}_1 - \hat{delta}_2 = \gamma + \delta - \gamma = \delta$

1.3. Data Please check the link on my website on the original data public.dat at the MHE Data Archive and there are some R reproduction files provides by Ropponen (2011).
Here is the description of the variables:

- y_ft_employment_before: Full time equivalent employment before treatment [Outcome]
- y_ft_employment_after: Full time equivalent employment after treatment [Outcome]
- d_nj: 1 if New Jersey; 0 if Pennsylvania (treatment variable) [Treatment]
- x_co_owned: If owned by company = 1
- x_southern_nj: If in southern NJ = 1
- x_central_nj: If if in central NJ = 1
- x_northeast_philadelphia: If in Pennsylvania, northeast suburbs of Philadelphia = 1
- x_easton_philadelphia: If in Pennsylvania, Easton = 1
- x_st_wage_before: Starting wage ($/hr) before treatment
- x_st_wage_after: Starting wage ($/hr) after treatment
- x_burgerking: If Burgerking = 1
- x_kfc: If KFC = 1
- x_roys: If Roys = 1
- x_wendys: If Wendys = 1
- x_closed_permanently: Closed permanently after treatment

```r
# Directly import data from shared google folder into R
data <- readr::read_csv("https://docs.google.com/uc?id=10h_5og14wbNHU-lapQaS1W6SBdzI7W6Z&export=downloa
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   x_co_owned = col_double(),
##   x_southern_nj = col_double(),
##   x_central_nj = col_double(),
##   x_northeast_philadelphia = col_double(),
##   x_easton_philadelphia = col_double(),
##   x_st_wage_before = col_double(),
##   x_st_wage_after = col_double(),
##   x_hrs_open_weekday_before = col_double(),
##   x_hrs_open_weekday_after = col_double(),
##   y_ft_employment_before = col_double(),
##   y_ft_employment_after = col_double(),
##   d_nj = col_double(),
##   d_pa = col_double(),
##   x_burgerking = col_double(),
##   x_kfc = col_double(),
##   x_roys = col_double(),
##   x_wendys = col_double(),
##   x_closed_permanently = col_double()
## )
```

```r
write.csv(data,"minwage.csv")
# Or download and import: data <- readr::read_csv("data-difference-in-differences.csv")
head(data)
```

```
## # A tibble: 6 x 18
##   x_co_owned x_southern_nj x_central_nj x_northeast_philadel~ x_easton_philadel~
##        <dbl>         <dbl>        <dbl>                 <dbl>              <dbl>
## 1          0             0            0                     1                  0
## 2          0             0            0                     1                  0
## 3          1             0            0                     1                  0
## 4          1             0            0                     1                  0
## 5          1             0            0                     1                  0
## 6          1             0            0                     1                  0
## # ... with 13 more variables: x_st_wage_before <dbl>, x_st_wage_after <dbl>,
## #   x_hrs_open_weekday_before <dbl>, x_hrs_open_weekday_after <dbl>,
## #   y_ft_employment_before <dbl>, y_ft_employment_after <dbl>, d_nj <dbl>,
## #   d_pa <dbl>, x_burgerking <dbl>, x_kfc <dbl>, x_roys <dbl>, x_wendys <dbl>,
## #   x_closed_permanently <dbl>
```

```r
# install.packages("psych")
library(psych)
```

```
## Warning: package 'psych' was built under R version 4.1.2
```

```r
describe(data)
```

```
##                            vars   n  mean   sd median trimmed  mad  min   max
## x_co_owned                    1 410  0.34 0.48   0.00    0.30 0.00 0.00  1.00
## x_southern_nj                 2 410  0.23 0.42   0.00    0.16 0.00 0.00  1.00
## x_central_nj                  3 410  0.15 0.36   0.00    0.07 0.00 0.00  1.00
## x_northeast_philadelphia      4 410  0.09 0.28   0.00    0.00 0.00 0.00  1.00
## x_easton_philadelphia         5 410  0.10 0.31   0.00    0.01 0.00 0.00  1.00
## x_st_wage_before              6 390  4.62 0.35   4.50    4.58 0.37 4.25  5.75
## x_st_wage_after               7 389  5.00 0.25   5.05    5.03 0.00 4.25  6.25
## x_hrs_open_weekday_before     8 410 14.44 2.81  15.50   14.48 2.22 7.00 24.00
## x_hrs_open_weekday_after      9 399 14.47 2.75  15.00   14.43 2.97 8.00 24.00
## y_ft_employment_before       10 398 21.00 9.75  19.50   19.94 7.41 5.00 85.00
## y_ft_employment_after        11 396 21.05 9.09  20.50   20.51 8.90 0.00 60.50
## d_nj                         12 410  0.81 0.39   1.00    0.88 0.00 0.00  1.00
## d_pa                         13 410  0.19 0.39   0.00    0.12 0.00 0.00  1.00
## x_burgerking                 14 410  0.42 0.49   0.00    0.40 0.00 0.00  1.00
## x_kfc                        15 410  0.20 0.40   0.00    0.12 0.00 0.00  1.00
## x_roys                       16 410  0.24 0.43   0.00    0.18 0.00 0.00  1.00
## x_wendys                     17 410  0.15 0.35   0.00    0.06 0.00 0.00  1.00
## x_closed_permanently         18 410  0.01 0.12   0.00    0.00 0.00 0.00  1.00
##                           range  skew kurtosis   se
## x_co_owned                  1.0  0.65    -1.58 0.02
## x_southern_nj               1.0  1.30    -0.31 0.02
## x_central_nj                1.0  1.91     1.67 0.02
## x_northeast_philadelphia    1.0  2.90     6.44 0.01
## x_easton_philadelphia       1.0  2.57     4.61 0.02
## x_st_wage_before            1.5  0.66    -0.35 0.02
## x_st_wage_after             2.0 -1.13     4.43 0.01
## x_hrs_open_weekday_before  17.0  0.08     0.06 0.14
## x_hrs_open_weekday_after   16.0  0.29     0.42 0.14
## y_ft_employment_before     80.0  1.75     6.34 0.49
## y_ft_employment_after      60.5  0.72     1.46 0.46
## d_nj                        1.0 -1.55     0.41 0.02
## d_pa                        1.0  1.55     0.41 0.02
## x_burgerking                1.0  0.34    -1.89 0.02
## x_kfc                       1.0  1.53     0.35 0.02
## x_roys                      1.0  1.20    -0.55 0.02
## x_wendys                    1.0  1.99     1.98 0.02
## x_closed_permanently        1.0  8.05    63.02 0.01
```

1.4. Visualization: Figure 1 and Figure 2 Now let's replicate Figure 1

```r
# Visuallization
#FIGURE 1
  x_st_wage_before_nj <-
  data$x_st_wage_before[data$d_nj == 1]
  x_st_wage_before_pa <-

    data$x_st_wage_before[data$d_pa == 1]

# Make a stacked bar plot - Plotly
# Set histogram bins
  xbins <- list(start=4.20, end=5.60, size=0.1)

library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

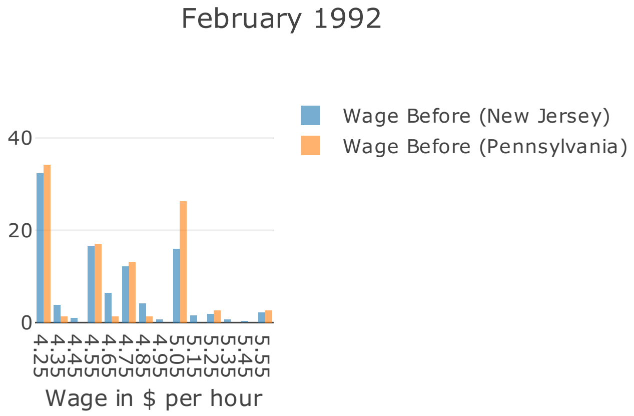
```
# install.packages("plotly")
library(plotly)
```

```
## Warning: package 'plotly' was built under R version 4.1.2

## Loading required package: ggplot2

##
## Attaching package: 'ggplot2'

## The following objects are masked from 'package:psych':
##
##     %+%, alpha

##
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':
##
##     last_plot

## The following object is masked from 'package:stats':
##
##     filter

## The following object is masked from 'package:graphics':
##
##     layout
```

```
# Plotly histogram
p <- plot_ly(alpha = 0.6) %>%
    add_histogram(x = x_st_wage_before_nj,
                  xbins = xbins,
                  histnorm = "percent",
                  name = "Wage Before (New Jersey)") %>%
    add_histogram(x = x_st_wage_before_pa,
                  xbins = xbins,
                  histnorm = "percent",
                  name = "Wage Before (Pennsylvania)") %>%
    layout(barmode = "group", title = "February 1992",
           xaxis = list(tickvals=seq(4.25, 5.55, 0.1),
                        title = "Wage in $ per hour"),
           yaxis = list(range = c(0, 50)),
                      margin = list(b = 100,
                          l = 80,
                          r = 80,
                          t = 80,
                          pad = 0,
                          autoexpand = TRUE))
p
```

```
## Warning: Ignoring 17 observations
```

```
## Warning: Ignoring 3 observations
```

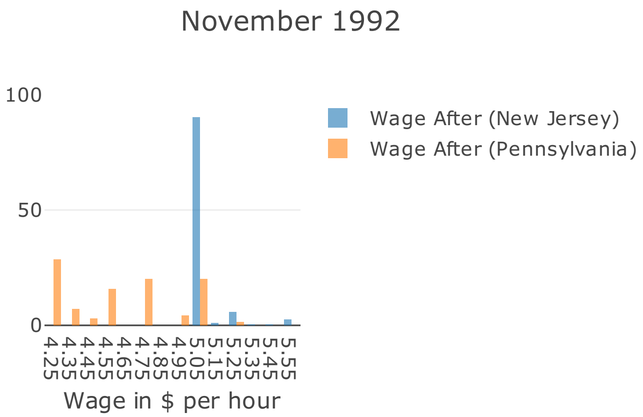


And here is Figure 2

You can also embed plots, for example:

```
## Warning: Ignoring 13 observations
```

```
## Warning: Ignoring 8 observations
```



2. Results

2.1.Replicate of Table 3

Table 3 presents mean comparisons of our outcome variable - measure at t0, t1 and the change between t0 and t1 - that we can reproduce.

```r
library(knitr)
# Table 3: Column 1-3, Row 1 (from left to right)

# 1st row: MEANs and SEs across subgroups
  results <- data %>% group_by(d_nj) %>% # group_by the treatment variable
            dplyr::select(d_nj, y_ft_employment_before) %>% # only keep variabel of interest
            group_by(N = n(), add = TRUE) %>% # count number of rows
            summarize_all(funs(mean, var, na_sum = sum(is.na(.))), na.rm = TRUE) %>% # aggregate/summar
            mutate(n = N - na_sum) %>%
            mutate(se = sqrt(var/n))
```

```
## Warning: The `add` argument of `group_by()` is deprecated as of dplyr 1.0.0.
## Please use the `.add` argument instead.
## Warning: `funs()` was deprecated in dplyr 0.8.0.
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with `tibble::lst()`:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
```

```
# Add row with differences
  results <- bind_rows(results, results[2,]-results[1,])
  results$group<- c("Control (Pennsylvania)", "Treatment (New Jersey)", "Difference")
  kable(results, digits=2)
```

d_nj	N	mean	var	na_sum	n	se	group
0	410	23.33	140.57	2	408	0.59	Control (Pennsylvania)
1	410	20.44	82.92	10	400	0.46	Treatment (New Jersey)
1	0	-2.89	-57.65	8	-8	-0.13	Difference

```
# Calculate SE for difference: SE = SQR(VAR/N + VAR/N)
  diff_se <- sqrt(results$var[1]/results$n[1] + results$var[2]/results$n[2])
  diff_se
```

```
## [1] 0.7428639
```

calculate the values for row 2 and row 3, i.e., y_ft_employment_after. Example code below.

```
# 2nd row: MEANs, SEs etc.
   results <- data %>% group_by(d_nj) %>% # group_by the treatment variable
            dplyr::select(d_nj, y_ft_employment_after) %>%
            group_by(N = n(), add = TRUE) %>% # count number of rows
            summarize_all(funs(mean, var, na_sum = sum(is.na(.))), na.rm = TRUE) %>% # aggregate/summar
            mutate(n = N - na_sum) %>%
            mutate(se = sqrt(var/n))
  results <- bind_rows(results, results[2,]-results[1,])
  results$group<- c("Control (Pennsylvania)", "Treatment (New Jersey)", "Difference")
  kable(results, digits=2)
```

d_nj	N	mean	var	na_sum	n	se	group
0	410	21.17	68.50	2	408	0.41	Control (Pennsylvania)
1	410	21.03	86.36	12	398	0.47	Treatment (New Jersey)
1	0	-0.14	17.86	10	-10	0.06	Difference

2.2.Replicate of Table 4

In Table 4 (p. 780) Card & Krueger control for some covariates. Check out the table notes of the table. Remember to always provide the number of observations in such tables for any model that you provide. Here it's rather intransparent.

```r
# Setup the data for table 4
data2 <- dplyr::select(data,
                         y_ft_employment_after,
                         y_ft_employment_before,
                         d_nj,
                         x_burgerking,
                         x_kfc,
                         x_roys,
                         x_co_owned,
                         x_st_wage_before,
                         x_st_wage_after,
                         x_closed_permanently,
                         x_southern_nj,
                         x_central_nj,
                         x_northeast_philadelphia,
                         x_easton_philadelphia) %>%
           mutate(x_st_wage_after = case_when(x_closed_permanently == 1 ~ NA_character_, # these stores
                                              TRUE ~ as.character(x_st_wage_after)),
                  x_st_wage_after = as.numeric(x_st_wage_after)) %>%
                  na.omit()
```

```r
# Model (i)/Column 1 (See exercise)

# Model (ii)/Column 2: Controls Chain/Ownership
  fit2 <- lm((y_ft_employment_after-y_ft_employment_before) ~
               d_nj + x_burgerking + x_kfc + x_roys + x_co_owned,
             data = data2)
  summary(fit2)
```

```
##
## Call:
## lm(formula = (y_ft_employment_after - y_ft_employment_before) ~
##     d_nj + x_burgerking + x_kfc + x_roys + x_co_owned, data = data2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -40.050  -3.685   0.584   4.077  27.169
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.2067     1.6082  -1.372   0.1709
## d_nj           2.2815     1.1970   1.906   0.0575 .
## x_burgerking   0.7566     1.4911   0.507   0.6122
## x_kfc          0.9912     1.6750   0.592   0.5544
## x_roys        -1.3280     1.6811  -0.790   0.4301
## x_co_owned     0.3729     1.0988   0.339   0.7345
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 8.721 on 345 degrees of freedom
## Multiple R-squared:  0.02038,    Adjusted R-squared:  0.006185
## F-statistic: 1.436 on 5 and 345 DF,  p-value: 0.2108
```