Econometrics III - PS 2

Lionel Chambon

2024-09-23

Exercise 1

Background

108 schools were selected. The two first grades were either formed based on ability (tracking group) or randomly (non-tracking group).

Question 1

```
tracking_data <- read_dta("tracking.dta")
summary(tracking_data)
```

```
##     schoolid         tracking        bottomhalf     scoreendfirstgrade
##  Min.   : 430.0   Min.   :0.0000   Min.   :0.0000   Min.   :-1.4296
##  1st Qu.: 685.5   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:-0.7930
##  Median : 789.0   Median :1.0000   Median :0.0000   Median :-0.1976
##  Mean   : 775.0   Mean   :0.5764   Mean   :0.4878   Mean   : 0.0000
##  3rd Qu.: 938.0   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.: 0.6359
##  Max.   :1020.0   Max.   :1.0000   Max.   :1.0000   Max.   : 3.2585
```

```
head(tracking_data)
```

```
## # A tibble: 6 x 4
##   schoolid tracking bottomhalf scoreendfirstgrade
##      <dbl>    <dbl>      <dbl>              <dbl>
## 1      430        1          1             -1.11
## 2      430        1          1             -1.40
## 3      430        1          1             -0.348
## 4      430        1          1             -0.705
## 5      430        1          1             -0.859
## 6      430        1          1             -0.361
```

First, we aggregate the data onto the school level.

```
school_level_data <- tracking_data %>%
  group_by(schoolid) %>%
  summarize(
    mean_grades = mean(scoreendfirstgrade, na.rm = TRUE),
```

```r
    treated = mean(tracking, na.rm = TRUE),
    below_median = mean(bottomhalf, na.rm = TRUE),
    n_students = n()
  )

head(school_level_data)
```

```
## # A tibble: 6 x 5
##   schoolid mean_grades treated below_median n_students
##      <dbl>       <dbl>   <dbl>        <dbl>      <int>
## 1      430     -0.184        1        0.509         53
## 2      432     -0.178        1        0.52          50
## 3      436     -0.0682       1        0.489         45
## 4      443     -0.0245       0        0.5           48
## 5      451     -0.758        0        0.35          40
## 6      452     -0.204        1        0.551         49
```

Now, I choose to run a linear regression to estimate the treatment effect.

```r
reg_q1 <- lm(mean_grades ~ treated, data = school_level_data)
summary(reg_q1)
```

```
##
## Call:
## lm(formula = mean_grades ~ treated, data = school_level_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.99292 -0.27882 -0.04172  0.25364  1.14125
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.06586    0.06047  -1.089    0.279
## treated      0.13391    0.08113   1.650    0.102
##
## Residual standard error: 0.419 on 106 degrees of freedom
## Multiple R-squared:  0.02506,    Adjusted R-squared:  0.01586
## F-statistic: 2.724 on 1 and 106 DF,  p-value: 0.1018
```

```r
reg_q1_c <- lm(mean_grades ~ treated + below_median, data = school_level_data)
summary(reg_q1_c)
```

```
##
## Call:
## lm(formula = mean_grades ~ treated + below_median, data = school_level_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.99568 -0.28079 -0.03937  0.25434  1.14011
##
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.08759    0.50788  -0.172    0.863
## treated       0.13299    0.08431   1.577    0.118
## below_median  0.04574    1.06154   0.043    0.966
##
## Residual standard error: 0.421 on 105 degrees of freedom
## Multiple R-squared:  0.02507,    Adjusted R-squared:  0.006503
## F-statistic:  1.35 on 2 and 105 DF,  p-value: 0.2637
```

This first analysis tells us that on average, students in tracked schools are predicted to have higher grades of ~0.13 (standardized) points. We can also see that students in the bottom half are expected to have a higher grade on average, but this effect is not statistically significant. The positive effect of tracking is about the same even when not controlling for the *bottomhalf* variable.

We see that the result does not change by much in magnitude if we use the unmodified data:

```
reg_q1_b <- lm(scoreendfirstgrade ~ tracking, data = tracking_data)
summary(reg_q1)
```

```
##
## Call:
## lm(formula = mean_grades ~ treated, data = school_level_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.99292 -0.27882 -0.04172  0.25364  1.14125
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.06586    0.06047  -1.089    0.279
## treated      0.13391    0.08113   1.650    0.102
##
## Residual standard error: 0.419 on 106 degrees of freedom
## Multiple R-squared:  0.02506,    Adjusted R-squared:  0.01586
## F-statistic: 2.724 on 1 and 106 DF,  p-value: 0.1018
```

```
reg_q1_c_b <- lm(scoreendfirstgrade ~ tracking + bottomhalf, data = tracking_data)
summary(reg_q1_c)
```

```
##
## Call:
## lm(formula = mean_grades ~ treated + below_median, data = school_level_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.99568 -0.28079 -0.03937  0.25434  1.14011
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.08759    0.50788  -0.172    0.863
## treated       0.13299    0.08431   1.577    0.118
## below_median  0.04574    1.06154   0.043    0.966
##
```

```
## Residual standard error: 0.421 on 105 degrees of freedom
## Multiple R-squared:  0.02507,    Adjusted R-squared:  0.006503
## F-statistic:  1.35 on 2 and 105 DF,  p-value: 0.2637
```

Question 2

A randomized inference test investigates $H_0 : \beta_1 = 0$. To do so, we would like to "shuffle" the treatment randomly across observations. Then, we recompute a null distribution of coefficients and then determine whether or not to reject H_0. First, we need to aggregate data by each school:

```r
observed_coef <- coef(reg_q1)["treated"]

perm <- 1000

permuted_coef <- numeric(perm)

for (i in 1:perm) {
  shuffled <- sample(school_level_data$treated)

  permuted_reg <- lm(mean_grades ~ shuffled, data = school_level_data)

  permuted_coef[i] <- coef(permuted_reg)["shuffled"]
}

conf_interval <- quantile(permuted_coef, probs = c(0.05, 0.95))

if (observed_coef < conf_interval[1] | observed_coef > conf_interval[2]) {
  print("We can reject the null hypothesis")
} else {
  print("We fail to reject the null hypothesis.")
}
```

```
## [1] "We can reject the null hypothesis"
```

We narrowly reject H_0

Question 3

One idea could be to investigate this using an interaction term with the *bottomhalf* variable:

```r
reg_q1_c_i <- lm(mean_grades ~ treated + below_median + (treated*below_median), data = school_level_data

summary(reg_q1_c_i)
```

```
##
## Call:
## lm(formula = mean_grades ~ treated + below_median + (treated *
##     below_median), data = school_level_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -0.87096 -0.28598 -0.02775  0.26325  1.10635
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -0.7290     0.6464  -1.128   0.2620
## treated                 1.7990     1.0539   1.707   0.0908 .
## below_median            1.3960     1.3550   1.030   0.3052
## treated:below_median   -3.4191     2.1561  -1.586   0.1158
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.418 on 104 degrees of freedom
## Multiple R-squared:  0.04809,    Adjusted R-squared:  0.02063
## F-statistic: 1.751 on 3 and 104 DF,  p-value: 0.1611
```

The coefficient of the interaction term reveals that the effect is reduced for below-median performing students, although the coefficient is not significant at $\alpha = 10\%$.

Question 2

Question 1

Question 2

```r
n=1000
v = rnorm(n, 0,1)
y_0 = rnorm(n, 0,1)
y_1 = 0.5*y_0 + 0.5*v + 0.2

sim_data <- data.frame(y_0 = y_0, y_1 = y_1)

head(sim_data)
```

```
##          y_0        y_1
## 1 -2.0773330 -1.3752359
## 2  0.5852878  1.3569103
## 3 -1.0546213 -0.2889577
## 4 -1.6780070 -0.8959711
## 5  0.2857104 -0.3260942
## 6 -2.2887414 -0.9287316
```

Question 3

I would expect the treatment effect to be heteregenous. If we compute the treatment effect as the difference between y_0 and y_1, we have $0.5 * y_0 - 0.5 * v - 0.2$. y_0 and V are both independent random normal variables, so the realizations will vary across units.

Question 4

```r
sim_data = sim_data %>%
  mutate(te = y_1 - y_0)

ate = sim_data %>%
  summarise(mean(te))

print(ate)
```

```
##    mean(te)
## 1 0.1960529
```

Question 5

```r
cor = cor(y_0, y_1)
var = var(sim_data$te)

print(cor)
```

```
## [1] 0.7083891
```

```r
print(var)
```

```
## [1] 0.478341
```

Question 6

```r
iterations = 800

results = data.frame(
  ATE = numeric(iterations),
  Variance = numeric(iterations),
  Confidence_Inclusion = integer(iterations)
)

for (i in 1:iterations) {

  random_sort = runif(n)

  sim_data_sorted = sim_data %>%
    mutate(random_sort)

  sim_data_sorted = sim_data_sorted %>%
    arrange(random_sort)

  sim_data_sorted = sim_data_sorted %>%
    mutate(D = ifelse(row_number() <= 500, 1, 0)) %>%
    mutate(Y = (1 - D) * y_0 + D * y_1)
```

```
  reg_q2 = feols(Y ~ D, se = "hetero", data = sim_data_sorted)

  ate_hat = coef(reg_q2)["D"]
  var_hat = se(reg_q2)["D"]^2

  # if (ate == coef(reg_q2)) {
  #  print(TRUE)
  # } else {
  #   print(FALSE)
  # }

  # if (var == var_hat) {
  #   print(TRUE)
  # } else {
  #   print(FALSE)
  # }

  ci_lower = ate_hat - 1.96 * sqrt(var_hat)
  ci_upper = ate_hat + 1.96 * sqrt(var_hat)

  indicator = as.numeric(ate >= ci_lower & ate <= ci_upper)

  results[i, ] = c(ate_hat, var_hat, indicator)

}
```

I used this code that is in comment mode above to check whether ate and var_hat were equal to the previous result. I commented it to avoid having a long list as output on the document. In short, the values are never equal.

```
mean_ate_hat = mean(results$ATE)

result_ate <- glue("The mean of the estimates is {mean_ate_hat}, the true value is {ate}.")
print(result_ate)
```

```
## The mean of the estimates is 0.197218458934481, the true value is 0.196052938278703.
```

```
var_ate_hat <- var(results$ATE)
mean_var_hat <- mean(var_ate_hat)
result_var <- glue("The mean of the variace is {mean_var_hat}, the true value is {var}.")
print(result_var)
```

```
## The mean of the variace is 0.00224048996988565, the true value is 0.478340972002665.
```

```
coverage_probability = mean(results$Confidence_Inclusion)
print(coverage_probability)
```

```
## [1] 0.97125
```

Interpretation here.

Question 7

```r
iterations = 800

results_2 = data.frame(
  ATE = numeric(iterations),
  Variance = numeric(iterations),
  Confidence_Inclusion = integer(iterations)
)

for (i in 1:iterations) {

  random_sort_2 = runif(n)

  sim_data_sorted_2 = sim_data %>%
    mutate(random_sort_2)

  sim_data_sorted_2 = sim_data_sorted_2 %>%
    arrange(random_sort_2)

  sim_data_sorted_2 = sim_data_sorted_2 %>%
    mutate(D_2 = ifelse(row_number() <= 500, 1, 0)) %>%
    mutate(Y_2 = (1 - D_2) * y_0 + D_2 * y_1)

  reg_q2_2 = feols(Y_2 ~ D_2, se = "hetero", data = sim_data_sorted_2)

  ate_hat_2 = coef(reg_q2_2)["D_2"]
  var_hat_2 = se(reg_q2_2)["D_2"]^2

  # if (ate == coef(reg_q2_2)) {
  #  print(TRUE)
  # } else {
  #   print(FALSE)
  # }

  # if (var == var(var_hat_2) {
  #   print(TRUE)
  # } else {
  #   print(FALSE)
  # }

  ci_lower_2 = ate_hat_2 - 1.96 * sqrt(var_hat_2)
  ci_upper_2 = ate_hat_2 + 1.96 * sqrt(var_hat_2)

  indicator_2 = as.numeric(0.2 >= ci_lower_2 & ate <= ci_upper_2)

  results_2[i, ] = c(ate_hat_2, var_hat, indicator_2)

}
```

Finally:

```r
mean_ate_hat_2 = mean(results_2$ATE)

result_ate_2 <- glue("The mean of the estimates is {mean_ate_hat}, the true value is {ate}.")
print(result_ate_2)
```

```
## The mean of the estimates is 0.197218458934481, the true value is 0.196052938278703.
```

```r
var_ate_hat_2 <- var(results_2$ATE)
mean_var_hat_2 <- mean(var_ate_hat_2)
result_var_2 <- glue("The mean of the variace is {mean_var_hat}, the true value is {var}.")
print(result_var_2)
```

```
## The mean of the variace is 0.00224048996988565, the true value is 0.478340972002665.
```

```r
coverage_probability_2 = mean(results_2$Confidence_Inclusion)
print(coverage_probability_2)
```

```
## [1] 0.97375
```

Interpretation here.