Natural experiments and LATE

Tutorial 4

Stanislav Avdeev

Goal for today's tutorial

1. Discuss LATE, 2SLS, and IV
2. Discuss the connections between unconditional means and the regression coefficients

Natural experiments

- Suppose we want to estimate a simple OLS model

$$Y_i = \alpha + \delta D_i + U_i$$

  where Y_i is the outcome variable, D_i is getting a treatment
- But what happens if not everyone gets a treatment even when they are assigned to it

$$D_i = \gamma + \beta Z_i + V_i$$

  where Z_i is being assigned to get a treatment
- How can we estimate the effect of D_i on Y_i taking into account Z_i?

Natural experiments

- A **natural experiment** can take many forms, but the basic idea is that something experiment-like occurs without the researcher's control
  - In other words, there is a form of **exogenous variation** in the real world
  - or at least conditionally exogenous
- And we can use that exogenous variation to identify our effect of interest
- However, not everyone will **comply** with a treatment they are assigned
  - what can we do in this case?

Better LATE than never

- We can use the IV estimator
- IV allows variation in the treatment **that is driven by the instrument**
- This also means that we can only see the effect **among people for whom the instrument drives their treatment**
  - if a treatment improves **your** outcome by 2, but **my** outcome by only 1, and the instrument has a **big** effect on whether you get treatment, but only a **little** effect on me, then our IV estimate will be a lot closer to 2 than to 1
- This is **LATE** - Local Average Treatment Effect
  - our estimate is **local** to people who are affected by the instrument
  - and even **more local** to those affected more heavily than others
- This means that the IV estimate won't be representative of **everyone's** effect
  - or even of the people who actually **were treated**
- But we might have to live with that to be able to use the cleaner identification

LATE: simulation

- Let's apply one of the common uses of IV - when you have a randomized experiment
  - in normal circumstances, if we have an experiment and assign people a treatment Z, we just compare Y across values of Z

```
set.seed(7)
df ← tibble(Z = sample(c(0, 1), 1000, replace = T),
            D = Z,
            Y = 5*D + rnorm(1000))
```

```
## # A tibble: 6 × 3
##       Z     D      Y
##   <dbl> <dbl>  <dbl>
## 1     1     1  4.79
## 2     0     0 -0.588
## 3     0     0 -0.685
## 4     1     1  6.00
## 5     0     0 -0.773
## 6     1     1  3.01
```

LATE: simulation

```
#The true effect is 5
m ← lm(Y ~ Z, df)
```

	Model 1
(Intercept)	−0.033
	(0.045)
Z	5.008***
	(0.063)
Num.Obs.	1000
+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001	

- We could identify the unbiased effect of Z on Y

LATE: simulation

- But what happens if you run a randomized experiment and assign people to a treatment Z, but not everyone does what you say?
  - some "treated" people don't get a treatment, and some "untreated" people do get it
- When this happens, we can't just compare Y across Z
  - but Z is still a **valid** instrument

```
set.seed(7)
df ← tibble(Z = sample(c(0, 1), 1000, replace=T),
            # some people do not comply - 20% do the opposite
            D = ifelse(runif(1000) < 0.8, Z, 1 - Z),
            Y = 5*D + rnorm(1000))
```

```
## # A tibble: 6 × 3
##       Z     D      Y
##   <dbl> <dbl>  <dbl>
## 1     1     1  6.25
## 2     0     1  4.23
## 3     0     0  0.216
## 4     1     0 -0.364
## 5     0     0 -0.821
## 6     1     0  0.582
```

LATE: simulation

```
#The true effect is 5
m ← lm(Y ~ Z, df)
```

	Model 1
(Intercept)	1.031***
	(0.101)
Z	3.009***
	(0.141)
Num.Obs.	1000
+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001	

- The effect is biased due to **non-compliance**
  - but what is this effect?

LATE: Intent-to-treat effect

- When you run an experiment, you can certainly **assign** people to treatment but you can't **make them do it**
  - that means we have **non-compliance**
- Especially a problem if it's non-random, since that brings endogeneity back
  - what can we do about that?
- If we have some non-compliance and just ignore the problem, we end up with an **intent-to-treat** (ITT) effect
  - basically, it all still works, except the effect we get is not **the effect of treatment**, it's **the effect of being assigned to treatment**, which is different
  - this can still be handy, especially if treatment might be assigned the same way in the future
- This will in general **underestimate** the effect of the treatment itself, since we include people in the "treated" category who were not actually treated, and people in the "untreated" category who were, so the two groups get closer together
  - so we get a smaller effect

LATE: Intent-to-treat effect

- If we can observe whether people actually received treatment (separate from us assigning it to them), we can use two-stage least squares (2SLS) to adjust the ITT so that we get the effect of actual treatment instead
  - in other words, we can use LATE
- Basically, LATE takes the effect of assignment and scales it up by how much the treatment assignment increases the treatment rate

$$LATE = \frac{\mathbb{E}(Y|Z=1) - \mathbb{E}(Y|Z=0)}{Pr(D=1|Z=1) - Pr(D=1|Z=0)}$$

  - keep in mind this is more representative of the effect **among those who respond really strongly to treatment**
  - and does not work if people **intentionally do the opposite of what you say**
- If it is more complex - you have control variables, etc., you can't just do the scaling, and have to perform two-stage least squares - IV
  - In the `fixest` package we have `feols()` which can do the IV estimation

LATE: simulation

```
# The true effect is 5
# Intent-to-treat
itt ← lm(Y ~ Z, df)

# Use IV to adjust for compliance
late ← feols(Y ~ 1 | D ~ Z, df)
```

	Model 1	Model 2
(Intercept)	1.031***	−0.034
	(0.101)	(0.065)
Z	3.009***	
	(0.141)	
fit_D		5.103***
		(0.110)
Num.Obs.	1000	1000
+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001		

LATE: simulation

- Let us do the scaling by hand, since this is a simple case (no covariates)

```
# The true effect is 5
df_late ← df %>%
  group_by(Z) %>%
  summarize(D = mean(D),
            Y = mean(Y))
df_late
```

```
## # A tibble: 2 × 3
##       Z     D     Y
##   <dbl> <dbl> <dbl>
## 1     0 0.209  1.03
## 2     1 0.798  4.04
```

```
assignment_effect ← df_late$Y[2] - df_late$Y[1]
treatment_increase ← df_late$D[2] - df_late$D[1]
late ← assignment_effect/treatment_increase
```

```
## [1] 3.0094296 0.5897719 5.1027007
```

LATE: simulation

- Estimate the same outcomes using regressions

```
# The true effect is 5
assignment_effect_reg  ← lm(Y ~ Z, df)
treatment_increase_reg ← lm(D ~ Z, df)
late_reg ← feols(Y ~ 1 | D ~ Z, df)
```

	Model 1	Model 2	Model 3
(Intercept)	1.031***	0.209***	−0.034
	(0.101)	(0.018)	(0.065)
Z	3.009***	0.590***	
	(0.141)	(0.026)	
fit_D			5.103***
			(0.110)
Num.Obs.	1000	1000	1000
+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001			

References

Books

- Huntington-Klein, N. The Effect: An Introduction to Research Design and Causality, Chapter 19.2.2: Instrumental Variables and Treatment Effects

Slides

- Huntington-Klein, N. Causality Inference Course, Lecture 7: Front Doors and Lecture 15: Instrumental Variables