

# Inference

EC 425/525, Lab 8

Edward Rubin

24 May 2019

# Prologue

# Schedule

## Last time

Helpful tips and tricks in R

## Today

Inference (in R)

# Inference

# Inference

## Motivation

So far, we've focused on carefully **obtaining causal estimates** of the effect of some treatment  $D_i$  on our outcome  $Y_i$ .

Our discussion of research designs and their requirements/assumptions has centered on **avoiding selection and securing unbiased and/or consistent estimates** for  $\tau$ .

In other words, we've concentrated on **point estimates**.

What about **inference**?

# Inference

## Shminference

**Q** Why care about inference?

**A** I'll give you two reasons.

1. We often want to **test theories/hypotheses**. Point estimates (*i.e.*,  $\hat{\beta}$ ) can't do this alone. Inference finishes the job.
1. Other times, we want to **measure the effect** of a treatment. Inference helps us think about the **precision** of our estimates.

*Note:* Similar reasoning can apply to bounding forecasting/predictions.

If you want answers, then you need to do inference correctly.

# Inference

## What's so complicated?

Angrist and Pischke told us that "correcting" our standard errors for heteroskedasticity may increase the standard errors up to 25%.

What else are we worried about?

# Inference

## What we're worried about

- **Transformations of estimators**, i.e.,  $\text{Var}\left[f\left(\hat{\beta}\right)\right] \neq f\left(\text{Var}\left[\hat{\beta}\right]\right)$
- **Dependence/correlation in our disturbance**, i.e.,  $\text{Cov}\left(\varepsilon_i, \varepsilon_j\right) \neq 0$ 
  - Autocorrelation  $\varepsilon_t = \rho\varepsilon_{t-1} + \varepsilon_t$
  - Correlated shocks within groups  $\varepsilon_i = \varepsilon_{g(i)} + \varepsilon_i$
- **Finite-sample properties** vs. asymptotic properties
- **Power** and **minimal detectable effects**
- **Multiple-hypothesis testing** and ***p-hacking***

*In other words: We've got a lot to worry/think about.*



# Clustering

# Clustering

## Setup

Many studies—observational and experimental—have a treatment that is assigned to all/most individuals within a group.

- Classrooms/schools
- Households
- Villages/counties/states

Furthermore, we might imagine individuals within the same group may have correlated disturbances. For  $i$  and  $j$  in group  $g$

$$\text{Cov}(\varepsilon_i, \varepsilon_j) = E[\varepsilon_i \varepsilon_j] = \rho_\varepsilon \sigma_\varepsilon^2$$

where  $\rho_\varepsilon$  gives the within-group correlation of disturbances—what MHE calls the **intraclass correlation coefficient**.

# Clustering

## Setup

In other words, we have a regression

$$y_i = \beta_0 + \beta_1 x_{g(i)} + \varepsilon_i$$

where individual  $i$  is in group  $g$ , and  $\mathbf{X}_{g(i)}$  only varies across groups.

For within-group correlation, we can use an additive random-effects model

$$\varepsilon_i = \nu_{g(i)} + \eta_i$$

meaning group members all receive a common shock  $\nu_{g(i)}$ , and individuals receive independent shocks  $\eta_i$ .

*Note* We assume  $\eta_i$  is independent of  $\eta_j$  ( $i \neq j$ ) and  $\nu_g$  ( $\forall g$ ).

# Clustering

## Additive random effects

Based upon this model we've set up

$$\varepsilon_i = \nu_{g(i)} + \eta_i$$

the covariance between individuals  $i$  and  $j$  in group  $g$  is

$$\begin{aligned}\text{Cov}(\varepsilon_i, \varepsilon_j) &= E[\varepsilon_i \varepsilon_j] = E[(\nu_g + \eta_i)(\nu_g + \eta_j)] = E[\nu_g^2] = \sigma_\nu^2 \\ &= \rho_\varepsilon \sigma_\varepsilon^2 \\ &= \rho_\varepsilon (\sigma_\nu^2 + \sigma_\eta^2)\end{aligned}$$

Thus, we can write the intraclass correlation coefficient as

$$\rho_\varepsilon = \frac{\sigma_\nu^2}{\sigma_\varepsilon^2} = \frac{\sigma_\nu^2}{\sigma_\nu^2 + \sigma_\eta^2}$$

# Clustering

## What is $\rho_\varepsilon$ ?

Let's review what we know.

$$\varepsilon_i = \nu_{g(i)} + \eta_i \quad \text{and} \quad \rho_\varepsilon = \frac{\sigma_\nu^2}{\sigma_\varepsilon^2} = \frac{\sigma_\nu^2}{\sigma_\nu^2 + \sigma_\eta^2}$$

One way to think about  $\rho_\varepsilon$  is as the **share of the variance of the disturbance  $\varepsilon_i$  accounted for by the shared disturbance  $\nu_{g(i)}$** .

As  $\nu_{g(i)}$  accounts for more and more of the variation in  $\varepsilon_i$ ,  $\rho_\varepsilon \rightarrow 1$ .

# Clustering

So...

Q Why do we care about  $\rho_\varepsilon$ ?

A It tells us by how wrong our standard errors can be if we treat all observations as independent.

Let  $\text{Var}_o(\hat{\beta}_1)$  denote the conventional variance formula for OLS estimator.<sup>†</sup>

Let  $\text{Var}(\hat{\beta}_1)$  denote the actual variance of  $\hat{\beta}_1$ .

<sup>†</sup> which treats all disturbances as independent (and identically distributed).

# Clustering

So....

With (1) nonstochastic regressors fixed by group and (2) groups of size  $n$

$$\frac{\text{Var}(\hat{\beta}_1)}{\text{Var}_o(\hat{\beta}_1)} = 1 + (n-1)\rho_\varepsilon \quad \Rightarrow \quad \frac{\text{S.E.}(\hat{\beta}_1)}{\text{S.E.}_o(\hat{\beta}_1)} = \sqrt{1 + (n-1)\rho_\varepsilon}$$

The term  $\sqrt{1 + (n-1)\rho_\varepsilon}$  is called the **Moulton factor**<sup>†</sup>.

The **Moulton factor** tells us by what factor standard errors will be wrong if we ignore within-group correlation (conditional on assumptions 1 and 2).

Q What happens if  $\rho = 1$ ? What if you duplicated your dataset?

Q What happens as  $n$  increases?

<sup>†</sup> After Moulton (1986).

# Clustering

## The Moulton factor

The Moulton factor

$$\frac{\text{S.E.}(\hat{\beta}_1)}{\text{S.E.}_o(\hat{\beta}_1)} = \sqrt{1 + (n - 1)\rho_\varepsilon}$$

shows even when  $\rho_\varepsilon$  is small, we can have vary large standard error issues.

*Ex* An experiment on 400 schools, each with 1,000 students.

If  $\rho_\varepsilon = 0.01$ , the Moulton factor is  $\sqrt{1 + (1,000 - 1) \times 0.01} \approx 3.32$ .



# Clustering

## Test statistics

$$\text{Recall } t_{\text{stat}} = \frac{\hat{\beta}_1}{\text{S.E.}(\hat{\beta}_1)}.$$

$$\therefore \frac{t_o}{t} = \frac{\hat{\beta}_1 / \text{S.E.}_o(\hat{\beta}_1)}{\hat{\beta}_1 / \text{S.E.}(\hat{\beta}_1)} = \frac{\text{S.E.}(\hat{\beta}_1)}{\text{S.E.}_o(\hat{\beta}_1)} = \text{the Moulton factor.}$$

*Ex* Thus, in our example of 400 schools with 1,000 students, ignoring within-school correlation of  $\rho_\varepsilon = 0.01$  would lead us test statistics that are more than 3 times as large as they should be.

This is why economics seminars have standard-error police.

# Clustering

## Relaxing assumptions

If we allow regressors to vary by individual and groups to differ in size ( $n_g$ ),

$$\frac{\text{Var}(\hat{\beta}_1)}{\text{Var}_o(\hat{\beta}_1)} = 1 + \left[ \frac{\text{Var}(n_g)}{\bar{n}} + \bar{n} - 1 \right] \rho_x \rho_\varepsilon$$

where  $\rho_x$  denotes the intraclass (within-group) correlation of  $x_i$ .<sup>†</sup>

**Important** The Moulton factor for this general model depends upon the amount of within-group correlation in  $x_i$  and  $\varepsilon_i$ .

The special case is also important, as treatment is often fixed at some level.

<sup>†</sup> See *MHE* for mathematical definitions and the derivation.

# Clustering

## The answer

**Q** So what do we do now?

**A** We've got options (as usual)

1. Parametrically model the random effects
2. Cluster-robust standard error (estimator)
3. Aggregate up to the group (or a similar method)
4. Block (group-based) bootstrap
5. GLS/MLE modeling  $y_i$  and  $\varepsilon_i$

**Most common:** Cluster-robust standard errors

**Runner up:** Block bootstrap

**Second runner up:** Group-level analysis

# Clustering

## Cluster-robust standard errors

Liang and Zeger (1986) extend White's heteroskedasticity-robust covariance matrix to allow for both clustering and heteroskedasticity.<sup>†</sup>

$$\hat{\Omega}_{cl} = (\mathbf{X}'\mathbf{X})^{-1} \left( \sum_g \mathbf{X}'_g \hat{\Psi}_g \mathbf{X}_g \right) (\mathbf{X}'\mathbf{X})^{-1}$$
$$\hat{\Psi}_g = a e_g e'_g = a \begin{bmatrix} e_{1g}^2 & e_{1g}e_{2g} & \cdots & e_{1g}e_{n_gg} \\ e_{1g}e_{2g} & e_{2g}^2 & e_{2g} \cdots & e_{2g}e_{n_gg} \\ \vdots & \vdots & \ddots & \vdots \\ e_{1g}e_{n_gg} & e_{2g}e_{n_gg} & \cdots & e_{n_gg}^2 \end{bmatrix}$$

such that  $e_g$  are the OLS residuals for group  $g$ ,  $e_{ig}$  is the residual for individual  $i$  in group  $g$ , and  $a$  is a degrees-of-freedom adjustment.

<sup>†</sup> When people say *clustering*, they typically mean *correlated disturbances within a group*.

# Clustering

## Cluster-robust standard errors

So now you know what `lm_robust()`, `iv_robust()`, *etc.* are doing when you specify a variable for clustering (e.g., `clusters = var`).

Time for a simulation.

# Cluster simulation

# Cluster simulation

## The DGP

Let's opt for a simple-ish example.<sup>†</sup>

$$y_{ig} = 1 + 2x_g + \varepsilon_{ig}$$
$$\varepsilon_{ig} = \nu_g + \eta_i$$

where the  $\eta_i \perp \eta_j$ ,  $\eta_i \perp \nu_g$ , and  $\nu_g \perp \nu_h$ .

Let's assume  $\eta_i \sim N(0, 1)$  and  $\nu_g \sim N(0, 1)$ . And  $x_g \sim N(0, 1)$ .

Plus  $n = 100$  with 10 groups.

<sup>†</sup> So we have more room for problem sets.

First we need to write the data generating process for one iteration.

```
# The DGP
sim_dgp ← function(n = 100, n_grps = 10,  $\sigma_v$  = 1,  $\sigma_\eta$  = 1) {
  # Create the right number of observations
  sample_df ← expand.grid(i = 1:n, g = 1:n_grps) %>% as_tibble()
  # Create a unique ID (from 1 to number of observations)
  sample_df %<>% mutate(id = 1:(n * n_grps))
  # Sample  $v$  at the group level
  # NOTE: Ungroup after grouping
  sample_df %<>% group_by(g) %>%
    mutate(v = rnorm(1, sd =  $\sigma_v$ )) %>% ungroup()
  # Sample  $\eta$  at the individual level
  sample_df %<>% mutate( $\eta$  = rnorm(n * n_grps, sd =  $\sigma_\eta$ ))
  # Sample  $x_g$  from  $N(0,1)$ 
  sample_df %<>% group_by(g) %>% mutate(x = rnorm(1)) %>% ungroup()
  # Calculate  $y$ 
  sample_df %<>% mutate(y = 1 + 2 * x + v +  $\eta$ )
  # Return
  return(sample_df)
}
```



Now we analyze.

```
# Analyze 'data'
sim_analyze ← function(data) {
  # Conventional SEs
  se_ols ← lm_robust(y ~ x, data = data, se_type = "classical") %>%
    tidy() %>% extract2(2,3)
  # Cluster-robust SEs
  se_cl ← lm_robust(y ~ x, data = data, clusters = g) %>%
    tidy() %>% extract2(2,3)
  # Return a data frame of results
  data.frame(
    se = c(se_ols, se_cl),
    type = c("conventional", "clustered")
  )
}
```

Now put them together with another function.

```
# Join sim_dgp and sim_analyze
sim_iter <- function(n = 100, n_grps = 10,  $\sigma_v$  = 1,  $\sigma_\eta$  = 1) {
  # Run the analysis in sim_analyze on the output of sim_dgp
  sim_dgp(n = 100, n_grps = 10,  $\sigma_v$  = 1,  $\sigma_\eta$  = 1) %>% sim_analyze()
}
```

And we run the simulation.

```
# Load and set up 'furrr'
p_load(furrr)
plan(multiprocess, workers = 8)
# Set a seed
set.seed(1234)
# Run the simulation 1,000 times
sim_df ← future_map_dfr(
  # Repeat sample size 100 for 1000 times
  rep(100, 1000),
  # Our function
  sim_iter,
  # Let furrr know we want to set a seed
  .options = future_options(seed = T)
)
```



# Table of contents

## Inference

1. Motivation
2. Clustering
3. Moulton factors
  - Example
  - Test statistics
4. Answers
5. Cluster-robust S.E.s
6. Simulation