

Lab 8: Field Experiments I

Due: Monday, March 11, 11:59 PM

Name: Your name here

Mac ID: The first half of your Mac email address

1 Non-compliance

The [Banerjee et al \(2021\)](#) reading focuses on the effect of a “big push” program to overcome poverty traps. In this program, households in the treatment group are invited to join the program, but they are free to choose not to participate.

This reflects a common problem in field experiments, participants may not experience the condition intended by researchers. This could be for logistics problems or because people simply reject treatments.

The consequence is that knowing which condition units were assigned to is not sufficient to characterize potential outcomes anymore. For example, in a two-arm trial, we can now have four different response profiles. The first two columns indicate the condition individuals are assigned to, the rows indicate the condition that participants actually experience:

Table 1: Potential outcome profiles under non-compliance

Assigned to control	Assigned to treatment	Type
Receive control	Receive control	Never-taker

Assigned to control	Assigned to treatment	Type
Receive control	Receive treatment	Complier
Receive treatment	Receive control	Defier
Receive treatment	Receive treatment	Always-taker

In words:

- Never- and always-takers experience the same condition regardless of treatment assignment
- Compliers go along with what the researcher intends
- Defiers go with the opposite of what the researcher intends

Since they are potential outcomes, compliance types cannot be observed directly. If someone rejects treatment, you do not know if they are defiers or never-takers.

To simplify potential outcomes, researchers usually assume **one-sided non-compliance**, which implies that units can only deviate from assigned conditions in treatment, but not in control. This would make sense for a field experiment similar to the TUP program, people can choose not to participate in the program if invited, but can't force their way into the program when not invited (at least in theory).

This means that the only compliance problem to worry about is units rejecting treatment, and whether they are never-takers or defiers becomes irrelevant.

The following code design simulates a field experiment with one-sided non-compliance. See [Section 18.6](#) of the textbook for details.

```
compliance_rate = 0.8

# M
```

```

comp_model = declare_model(
  N = 500,
  U = rnorm(N),
  # indicate whether respondent would reject treatment if assigned to
  comply = complete_rs(
    N = N,
    prob = compliance_rate
  ),
  # potential outcomes with respect to D
  potential_outcomes(
    Y ~ 0.25 * D + U,
    conditions = list(D = c(0, 1))
  ),
  # potential outcomes with respect to Z
  potential_outcomes(
    D ~ case_when(
      Z == 1 & comply == 1 ~ 1,
      Z == 1 & comply == 0 ~ 0,
      Z == 0 ~ 0
    ),
    conditions = list(Z = c(0, 1))
  )
)

```

Here we use Z to characterize treatment *assignment* and D to characterize treatment *reception*. The `compliance_rate` indicates the proportion of units in our sample that would accept treatment if assigned to.

Our inquiry now specifies two quantities of interest. On top of the usual average treatment effect (ATE), we can also calculate the **average treatment effect on the treated (ATT)**.

```
# I
comp_inquiry = declare_inquiry(
  ATE = mean(Y_D_1 - Y_D_0),
  ATT = mean(Y_D_1[comply == 1] - Y_D_0[comply == 1])
)
```

The data strategy first assigns Z and then realizes D and Y accordingly.

```
comp_assign = declare_assignment(
  Z = conduct_ra(N = N)
)

comp_measure = declare_measurement(
  D = reveal_outcomes(D ~ Z),
  Y = reveal_outcomes(Y ~ D)
)
```

Since we have two inquiries, we also have two answer strategies. The first is known as the **intention to treat (ITT)** estimator, which is just the difference in means based on treatment assignment. You usually calculate this when you do not observe compliance status but suspect that participants may reject treatment.

The second is a **local average treatment effect (LATE)**, which tries to factor in both assigned and received treatment to adjust the ITT by the compliance rate. The estimation procedure for this is known as *two-stage least squares* or *instrumental variable*. The details are beyond the scope of this lab.

```
# A
itt = declare_estimator(
  Y ~ Z,
  inquiry = c("ATE", "ATT"),
  label = "ITT"
)

late = declare_estimator(
  Y ~ D | Z,
  .method = iv_robust,
  inquiry = c("ATE", "ATT"),
  label = "LATE"
)
```

Now, we can put the design together:

```
comp_design = comp_model + comp_inquiry +
  comp_assign + comp_measure +
  itt + late
```

i Task 1

Diagnose the `comp_design` and evaluate each answer strategy in terms of bias, RMSE, and power. Which estimator would you recommend using? Why?

What happens to the three diagnosands when you decrease the `compliance_rate`? Why?

What happens to the diagnosands when we have a `compliance_rate` equal to 1?

2 Stepped-wedge design

[Pennycook et al \(2021\)](#) use a stepped-wedge experiment. The following code declares such a design with two conditions and an arbitrary number of time periods or waves.

The model is a little more involved since each wave is a different level and we observe each units as many times as we have levels.

```
# M
t = 3 # number of periods

swd_model = declare_model(
  units = add_level(
    N = 100,
    U_unit = rnorm(N)
  ),
  periods = add_level(
    N = t,
    time = 1:max(periods),
    U_time = rnorm(N),
    nest = FALSE
  ),
  unit_period = cross_levels(
    by = join_using(units, periods),
    U = rnorm(N),
    potential_outcomes(
      Y ~ scale(U_unit + U_time + time + U) + 0.35 * Z
    )
  )
)
```

```
)
```

The inquiry is the usual average treatment effect, but this quantity is not defined at the last time period since by then the control group is empty.

```
# I
swd_inquiry = declare_inquiry(
  ATE = mean(Y_Z_1 - Y_Z_0),
  subset = time < max(time)
)
```

The answer strategy assigns in which period a unit receives treatment, then the unit stays treated for the rest of the study.

```
# D
swd_assign = declare_assignment(
  wave = cluster_ra(clusters = units,
                    conditions = 1:max( periods)),
  Z = if_else(time >= wave, 1, 0)
)

swd_measure = declare_measurement(
  Y = reveal_outcomes(Y ~ Z)
)
```

Our estimator needs to account for how outcome Y varies over time and across units in ways that are unrelated to the experiment and that we observe the same units multiple times. This is best done with a regression model with *two-way fixed-effects* and *clustered standard errors* (again, further details are beyond this lab).

```
# A
swd_estimator = declare_estimator(
  Y ~ Z,
  fixed_effects = ~ periods + units,
  clusters = units,
  subset = time < max(time),
  inquiry = "ATE",
  label = "TWFE")
```

Then we put the design together:

```
swd = swd_model + swd_inquiry + swd_assign + swd_measure + swd_estimator
```

Task 2

Diagnose the `swd` design with the current number of time periods `t` and two more alternative values larger than 3 but less or equal than 10. Looking at bias, RMSE, and power, Which number of time periods would you recommend? Why? Remember that collecting data is costly, so you do not want to have more time periods than necessary.

3 Answers

Remember to show the code and output you use to inform your answers!

3.1 Task 1

Work on your answers here.

3.2 Task 2

Work on your answers here.