Introduction to `DeclareDesign`

Prepared for EGAP Learning Days VI, Malawi, February 2017

Macartan Humphreys and Tara Slough

February 2017

What is `DeclareDesign`?

- Framework, R package for formally characterizing research designs
  - Problem: Degree of detail about research design varies greatly in published work, pre-analysis plans etc.
  - Motivation: Create framework capable of characterizing all research designs (any/all methods)
- Tools for analysis of designs
  - Big question: What could we have learned from a design?
  - Ancillary benefits: Create "dummy" datasets with which you can practice different analyses

What could we have learned from a design?

- A research design at the front-end consists of:
  - Design choices by researcher/nature/dataset maker (depends on method)
  - Set of beliefs/assumptions about how the world works
- We can learn about how a design functions through simulation
  - Logic of Monte Carlo analysis
  - Draw data, analyze, save estimates, repeat many times
  - Diagnoses based on estimates

Six Components of a Research Design

1. **Population**: Set of units about which inferences are sought and their characteristics
2. **Potential outcomes**: Outcomes each unit might exhibit depending on how causal process changes the world
3. **Sampling strategy**: Strategy used to select units to include in study
4. **Assignment**: Manner in which units are assigned to reveal one potential outcome or another
5. **Estimand**: Quantities that we want to learn about in the world, in terms of potential outcomes
6. **Estimator**: Procedure for generating estimates of the quantities we want to learn about

Population

- Start from theory:
  - Where should the theory apply?
  - Where shouldn't it apply? (scope conditions)
- What is the population that we want to make an inference about?
- Often practical limitations about the population that we can study
  - Regardless of method employed
  - But stay idealistic for the moment

Sampling Strategy

- How are we selecting units to analyze?
  - How do we choose the case (context)?
  - What does this selection method mean about inference to the population above?
- Types of samples:
  - Population ("big data," very specific populations)
  - Convenience (lab experiments, some surveys etc.)
  - Some sort of random sample (some surveys etc.)
- If we can't make a population inference from a sample, do we:
  - Redefine population?
  - Only worry about sample diagnostics?
- One source of concerns about external validity
  - Unclear (to me) that this is an "experiments" issue...

Potential Outcomes

- Codifies our asssumptions about the relationships between different treatment conditions, baseline covariates, and outcomes
  - Should be rooted in theory
- Functional form of relationships must be specified
  - Should be informed by theory, but often theories are not specific on this point
  - Less difficult in case of binary treatments
  - Another source of concerns about external validity (model-based inference)
- Some difficulties of experimental analysis should be viewed as potential outcomes!
  - (Non)-compliance
  - Spillovers
  - Attrition

Assignment

How is treatment/independent variable of interest assigned?

- Experiments:
  - Input the randomizations.
  - Defaults allow for simple, complete, blocked, clustered, and blocked and clustered randomization, among others
- Quasi-experiments/natural experiments:
  - Treatment assignment requires more assumptions about the assignment process
- Other observational work:
  - Stronger assumptions about the assignment of treatment, assignment could be modeled on covariates

Estimand, Estimator

- What do we want to know?
  - We will talk about the ATE, various marginal effects, conditional marginal effects
  - Other effects of interest: ITT, LATE, CATE
  - Estimands not specified frequently (enough) in existing literature
  - Snarky comment: Stars don't mean much if we don't know what the coefficient/estimate is measuring
- Estimator:
  - Too often we utilize estimators without identifying estimand
  - Many estimators consistent with each estimand – though some work better than others...

Putting it All Together

- Beyond framework for research design, `DeclareDesign` is an R package:
  - Relies on *simulation* of data under assumptions built into research design
  - Idea: simulate many datasets, estimate estimands, assess the properties of these many estimates
  - We will see an example tomorrow
- Big question: What can I learn given my research design?

Mapping from `DeclareDesign` to Research Design Form

There are a few additional components to the research design form, but most map quite clearly into the `DeclareDesign` framework.

1. **Population**:
   - Units (10)
2. **Potential outcomes**:
   - X (8)
   - Y (9)
   - Measurement strategy (how are X and Y measured/quantified? (13))
   - Hypotheses (7)
   - Heterogeneity (14)
   - Effect Size (16)

Mapping, continued

3. **Sampling strategy**:
   - Sample (12)
   - (to some extent: Threats to external validity (22))
4. **Assignment**:
   - Random Assignment (11)
5. **Estimand**
   - Analysis strategy (19)
6. **Estimator**
   - Analysis strategy (19)

Mapping of `DeclareDesign` Output:

The output of a design diagnosis provides:

- Power Calculation (18)–plus *much* more
- Threats to Internal Validity (21)
- Threats to External Validity (22)

The Guts: Declare Design

```
population <- declare_population()
sampling <- declare_sampling()
assignment <- declare_assignment()
potential_outcomes <- declare_potential_outcomes()
estimand <- declare_estimand()
estimator <- declare_estimator(estimand = estimand)

my_design <- declare_design(
          population          = population,
          sampling            = sampling,
          potential_outcomes  = potential_outcomes,
          assignment          = assignment,
          estimator           = estimator)
```

Audience for `DeclareDesign`

Three-ish audiences:

1. **Ninjas:** Advanced R users that (might) specify user-input functions for any design
2. **Advanced:** Use built-in functionalities to characterize a wide range of designs
3. **Novices:** Use template functions (some here, many forthcoming) to characterize and examine a variety of designs with a few simple arguments.

Using DeclareDesign

```
install.packages("devtools") # run once only!
library(devtools)
install_github("DeclareDesign/DeclareDesign") # run once o
library(devtools)



source("k_arm_template.R")
```

Two templates in this file:

- Generalized m-arm
- 2×2 factorial

See .pdf for detail about all arguments to these functions.

How Can we Use `DeclareDesign` to Learn Designs?

- Suppose you want to extend Chong, De la O, Karlan, Wantchekon (2014) to a different context.
- Three treatment arms
  1. Pure control (no flyer)
  2. Placebo (flyer about the federal transfer)
  3. Treatment (flyer about federal transfer with results of corruption audit)
- DV: Turnout
- 600 municipalities are candidates for evaluation
- You can only afford to implement treatment and do data collection in 450 municipalities
- Hypothesized treatment effect comes from Chong et al. (2014) findings

Design 1: No Pretreatment Covariate

- Assume the following Potential Outcomes Function:

$$\text{Turnout}_i = 60 - 1.5 \times \text{Treatment}_i + 0.5 \times \text{Placebo}_i + \epsilon_i$$

- We can enter this entire design with the following code:

```
three_arm_des_1 <- k_arm_template(
      N             = 600,           # 600 munis in pop.
      n             = 450,           # 450 in sample
      k             = 3,             # 3 arms, 150/arm
      mu_Y0         = 60,            # 60% turnout in ctrl
      ATEs          = c(-1.5, 0.5),  # Treatment effects
      noise_scale = 8)               # SD of error term
```

- Estimands are ATEs, Estimator is OLS

Design 2: Lagged Turnout as Pretreatment Covariate

Assume the following Potential Outcomes Function:

$$\text{Turnout}_i = 28.5 - 1.5 \times \text{Treatment}_i + 0.5 \times \text{Placebo}_i + 0.5 \times \text{Turnout}_{t-1} + \epsilon_i$$

Declare design without covariate adjustment:

```
three_arm_des_2 <- k_arm_template(
 N                 = 600,           # 600 in population
 n                 = 450,           # 450 in sample
 k                 = 3,             # 3 arms, 150/arm
 mu_Y0             = 28.5,          # basline in ctrl
 ATEs              = c(-1.5, 0.5), # Treatment effects
 noise_scale       = 4,             # SD of error term
 coef_X            = 0.5,           # Coef. on turnout, t-1
 location_scale_X = c(65, 8),       # Mean, SD of turnout, t
 cov_adjustment    = FALSE)         # No covariate adjustmen
```

Design 3: Lagged Turnout as Pretreatment Covariate

Assume the following Potential Outcomes Function:

$$\text{Turnout}_i = 28.5 - 1.5 \times \text{Treatment}_i + 0.5 \times \text{Placebo}_i + 0.5 \times \text{Turnout}_{t-1} + \epsilon_i$$

Declare design **with** covariate adjustment:

```
three_arm_des_3 <- k_arm_template(
 N                 = 600,           # 600 in population
 n                 = 450,           # 450 in sample
 k                 = 3,             # 3 arms, 150/arm
 mu_Y0             = 28.5,          # basline in ctrl
 ATEs              = c(-1.5, 0.5), # Treatment effects
 noise_scale       = 4,             # SD of error term
 coef_X            = 0.5,           # Coef. on turnout, t-1
 location_scale_X = c(65, 8),      # Mean, SD of turnout, t
 cov_adjustment    = TRUE)          # No covariate adjustmen
```

Use the Design: Draw Data

- `draw_data()` generates a single dataset with the characteristics built into the design
- Subset of data for `three_arm_des_1`

```
mock_data <- draw_data(design = three_arm_des_1)
```

Y_Z_c	Y_Z_t1	Y_Z_t2	Z	Y
58.600	57.100	59.100	treatment2	59.100
57.622	56.122	58.122	treatment1	56.122
50.486	48.986	50.986	treatment2	50.986
60.226	58.726	60.726	treatment1	58.726
53.745	52.245	54.245	control	53.745
57.474	55.974	57.974	treatment1	55.974

Use the Design: Implement the Analysis

- `get_estimates()` executes the estimator in a sample "dummy" dataset

```
ests <- get_estimates(
                estimator = three_arm_des_1$estimator,
                data = mock_data)
```

	Treatment1	Treatment2
est	-1.085	0.279
se	0.907	0.868
p	0.232	0.748
ci_lower	-2.867	-1.427
ci_upper	0.698	1.984
df	447.000	447.000

- Measure of signal (est) and noise (se)

Intuition Behind Simulation

- Re-generate the dataset
- Re-estimate the estimates
- Record estimates

```
ests2 <- get_estimates(
           estimator = three_arm_des_1$estimator,
           data = draw_data(three_arm_des_1))
```

	Treatment1	Treatment2
est	-1.660	0.441
se	0.815	0.838
p	0.042	0.599
ci_lower	-3.261	-1.206
ci_upper	-0.059	2.087
df	447.000	447.000

Use the Design: Diagnose Design via Simulation

- A design can be diagnosed on the basis of simulations
- Diagnosands are statistical properties of the design

```
diag <- diagnose_design(three_arm_des_1,
          population_draws = 200, sample_draws = 1,
          assignment_draws = 1)
```

Diagnosand	Goal	Treatment_1	Treatment_2
mean(estimand)	-	-1.500	0.500
mean(estimate)	(Estimand)	-1.622	0.555
sd(estimate)	0	0.893	0.931
bias	0	-0.122	0.055
RMSE	0	0.899	0.930
coverage	0.95	0.950	0.925
power	1	0.395	0.105
type S rate	0	0.040	0.290

Design and Diagnosis: A feedback looop

- Ideally, we can learn about a design by iteratively designing and diagnosing

$$\text{Design} \leftrightarrow \text{Diagnosis}$$

- Our comparison of three different designs for the Chong et al. (2014) studies adopts this logic
- Let's compare the properties of the designs on treatment 1 (corruption + fund allocation information)

```
diag2 <- diagnose_design(three_arm_des_2)
diag3 <- diagnose_design(three_arm_des_3)
```

Comparing designs

Three variants:

1. Orignal (as above)
2. Different POs $\rightarrow$ Including lagged turnout
3. Different POs + Estimator with covariate adjustment

Diagnosand	Goal	Original	Diff_POs	Diff_POs_Est
mean(estimand)	-	-1.500	-1.500	-1.500
mean(estimate)	(Estimand)	-1.622	-1.517	-1.515
sd(estimate)	0	0.893	0.681	0.484
bias	0	-0.122	-0.017	-0.015
RMSE	0	0.899	0.680	0.483
coverage	0.95	0.950	0.935	0.940
power	1	0.395	0.635	0.875
type S rate	0	0.040	0.015	0.005

Take Aways

1. Framework to think about complete research designs
    - Move toward qustion "what could I learn given my design"
    - Way to conceptualize differences between different approaches
2. Tools for examining research designs *ex-ante*
    - Creating mock datasets
    - Diagnosing designs