Running a replication and extension from an existing design and data

Macartan

September 3, 2018

This is a very short intro to design declaration. There are many more resources at declaredesign.org and in our paper https://paper.declaredesign.org/paper.pdf. The first section goes over design declaration and the next ones illustrate implementation and replication.

This note is written using Rmarkdown in Rstudio so you can see all the code used to do everything here.

1 The design phase

1.1 Declaration

In the beginning someone declares a design and saves it.

Here is a short way to declare a design using the design library.

```
library(DesignLibrary)
great_design <- simple_two_arm_designer(N = 1000, ate = .2)
save(great_design, file = "mydesign.Rdata")
```

For any design you generate using the library you can extract the generating code using:

```
get_design_code(great_design)
```

You can also declare a design "by hand" — explicitly describing all the steps.

e.g.

```
# M -- Model: Speculation on variables and relations between them
population <- declare_population(N = 100, u = rnorm(N))
potential_outcomes <- declare_potential_outcomes(Y_Z_0 = 0,
                                                 Y_Z_1 = .2 + u)

# I -- Inquiry: A query defined in terms of potential outcomes
estimand <- declare_estimand(ATE = mean(Y_Z_1 - Y_Z_0))

# D -- Data Strategy: Researcher interventions on the world
assignment <- declare_assignment(m = 50)
reveal_Y   <- declare_reveal(Y,Z)

# A -- Answer Strategy: Conclusions to be drawn from data
estimator  <- declare_estimator(Y ~ Z, estimand = estimand)

# Design: Putting it all together
design <- population + potential_outcomes + estimand + assignment + reveal_Y + estimator
```

1.2 Diagnosis

Once declared, a design can be "diagnosed":

```
diagnosis <- diagnose_design(design)
kable(reshape_diagnosis(diagnosis)[, -(1:5)])  # Print results (anything odd?)
```

Bias	RMSE	Power	Coverage	Mean Estimate	SD Estimate	Mean Se	Type S Rate	Mean Estimand
0.00	0.10	0.24	0.99	0.19	0.14	0.14	0.00	0.19
(0.00)	(0.00)	(0.02)	(0.01)	(0.01)	(0.00)	(0.00)	(0.00)	(0.00)

2 The implementation phase

Months later. Someone gathers data and saves it. Here we just create data using the design object.

```
great_data <- draw_data(great_design)
write.csv(great_data, file = "mydata.csv")
```

They publish their results:

```
estimates <-   get_estimates_data(great_design, great_data)
kable(estimates[,-(9:11)]) # Print results
```

estimator_label	term	estimate	std.error	statistic	p.value	conf.low	conf.high
estimator	Z	0.0682838	0.0626339	1.090206	0.2758863	-0.0546259	0.1911936

3 Replication

Years later. A new researcher downloads the design, downloads the data, and uses the design to replicate the analysis thus:

3.1 Get design and data

```
rm(list = ls())
load("mydesign.Rdata")                      # Ideally from url
downloaded_data <- read.csv("mydata.csv")   # Ideally from url
```

3.2 Do replication

```
replication <-   get_estimates_data(great_design, downloaded_data)[,-(9:11)]

kable(replication[,-(9:11)])
```

estimator_label	term	estimate	std.error	statistic	p.value	conf.low	conf.high
estimator	Z	0.0682838	0.0626339	1.090206	0.2758863	-0.0546259	0.1911936

3.3 Go further

They might even go further by e.g. adding new data or by modifying the design:

```
new_variable <- rnorm(nrow(downloaded_data))
new_data     <- data.frame(downloaded_data, new_variable)

new_estimation_strategy <- declare_estimator(Y~Z + new_variable,
                                             label = "Robustness", model = lm)
new_design  <- great_design + new_estimation_strategy

extension   <-   get_estimates_data(new_design, new_data)
kable(extension[,-(9:11)])  # Print results
```

estimator_label	term	estimate	std.error	statistic	p.value	conf.low	conf.high
estimator	Z	0.0682838	0.0626339	1.090206	0.2758863	-0.0546259	0.1911936
Robustness	Z	0.0694065	0.0626025	1.108687	0.2678327	-0.0534412	0.1922543