Estimando estimandos con estimadores

EGAP Learning days, Ciudad de Mexico

09-08-2023

Puntos clave

Recapitulación

Estimandos, estimadores y promedios

Puntos clave

Puntos clave para la estimación I

- Un efecto causal, τ_i, es una comparación de resultados potenciales no observados para cada unidad i, por ejemplo: $\tau_i = Y_i(T_i = 1) - Y_i(T_i = 0)$ or $\tau_i = \frac{Y_i(T_i=1)}{Y_i(T_i=0)}$.
- Para aprender sobre τ_i, podemos tratar a τ_i como un **estimando** o una cantidad objetivo a ser estimada, o como una cantidad objetivo sobre la cual se formulan hipótesis.
- Hay muchas personas que se enfoncan en el **efecto promedio del tratamiento** (average treatment effect, ATE), $\bar{\tau} = \sum_{i=1}^{n} \tau_i$, en parte, porque permite una **estimación** fácil.

Puntos clave para la estimación II

La clave para la estimación en la inferencia causal es elegir un estimando que permite aprender sobre alguna pregunta teórica o de políticas públicas. Para esto, el ATE es una opción, pero otros estimandos comunes también incluyen el ITT, LATE/CACE, ATT o ATE para algún subgrupo (o incluso una diferencia de un efecto causal entre grupos).

- Un **estimador** es una fórmula para hacer una estimación sobre el valor de un estimando. Por ejemplo, la diferencia de medias observadas para m unidades tratadas es un estimador de $\bar{\tau}$: $\hat{\bar{\tau}} = \frac{\sum_{i=1}^{n}(T_i Y_i)}{m} - \frac{\sum_{i=1}^{n}((1-T_i)Y_i)}{(n-m)}$.

Puntos clave para la estimación III

- El **error estándar** de un estimador en un experimento aleatorio resume cómo varían las estimaciones si se repitiera el experimento.
- Usamos el **error estándar** para producir **intervalos de confianza** y **valores p** para que podamos comenzar con un estimador y terminemos con una prueba de hipótesis.
- Diferentes aleatorizaciones producirán diferentes valores del mismo estimador que busca estimar el mismo estimando. Un **error estándar** resume esta variabilidad en un estimador.
- Un **intervalo de confianza** del $100(1-\alpha)\%$ es una colección de hipótesis que no se pueden rechazar a un nivel α. Es común reportar intervalos de confianza que contienen hipótesis sobre los valores de nuestro estimando y usar nuestro estimador como una estadística de prueba.

Puntos clave sobre la estimación IV

- Los estimadores deben:
  - evitar errores sistemáticos al estimar el estimando (ser insesgados);
  - varíar poco en las estimaciones de un experimento a otro. (ser precisos o eficientes) y
  - quizá idealmente converger al estimando a medida que se utiliza más información (ser consistentes).

Recapitulación

Recapitulación: Efectos causales

Recapitulación: La inferencia causal se puede resumir en una comparación de resultados potenciales fijos no observados.

Por ejemplo:

- El resultado potencial, o posible, de la unidad i cuando se asigna al tratamiento, $T_i = 1$ es $Y_i(T_i = 1)$.
- El resultado potencial, o posible, de la unidad i cuando se asigna al control, $T_i = 0$ es $Y_i(T_i = 0)$

La asignación al tratamiento, T_i, tiene un efecto causal para la unidad i al que llamamos τ_i, si $Y_i(T_i = 1) - Y_i(T_i = 0) \neq 0$ o $Y_i(T_i = 1) \neq Y_i(T_i = 0)$.

Estimandos, estimadores y promedios

¿Cómo podemos aprender sobre los efectos causales utilizando los datos observados?

1. Recordemos que podemos **probar hipótesis** sobre los dos resultados potenciales $\{Y_i(T_i = 1), Y_i(T_i = 0)\}$.
2. Podemos **definir estimandos** en términos de $\{Y_i(T_i = 1), Y_i(T_i = 0)\}$ o τ_i, **desarrollar estimadores** para esos estimandos, y luego calcular los valores y los errores estándar para esos estimadores.

Un estimando y un estimador común: el efecto promedio del tratamiento y la diferencia de medias

Supongamos que estamos interesados en el ATE, o $\bar{\tau}_i = \sum_{i=1}^{n} \tau_i$. ¿Cuál sería un buen estimador?

Dos candidatos:

1. La diferencia de medias: $\hat{\bar{\tau}} = \frac{\sum_{i=1}^{n}(T_i Y_i)}{m} - \frac{\sum_{i=1}^{n}((1-T_i)Y_i)}{n-m}$.
2. Una diferencia de medias después de recodificar el valor máximo de las observaciones Y_i (una especie de media "truncada" (winsorized), con lo que se busca evitar que los valores extremos tengan demasiada influencia sobre nuestro estimador; se usa para aumentar la *precisión*).

¿Cómo saber cuál estimador es mejor para un diseño de investigación en particular?

¡Simulemos!

Paso 1 de la simulación: generar datos con un ATE conocido

Tengan en cuenta que necesitamos *conocer* los resultados potenciales y la asignación al tratamiento para saber si el estimador propuesto funciona bien.

Z	y0	y1
0	0	10
0	0	30
0	0	200
0	1	91
1	1	11
1	3	23
0	4	34
0	5	45
1	190	280
1	200	220

Paso 1 de la simulación: generar datos con un ATE conocido

Z	y0	y1
0	0	10
0	0	30
0	0	200
0	1	91
1	1	11
1	3	23
0	4	34
0	5	45
1	190	280
1	200	220

```
El ATE real es 54
```

IMPORTANTE: En la vida real sólo podemos observar una realización de los resultados potenciales. Recuerden que cada unidad tiene su propio efecto bajo el tratamiento.

Primero: generar datos artificiales

La tabla de la diapositiva anterior fue generada en R con:

```
# Tenemos 10 unidades
N <- 10
#  y0 es la resultado potencial bajo el control
y0 <- c(0, 0, 0, 1, 1, 3, 4, 5, 190, 200)
# Para cada unidad el efecto del tratamiento es intrínseco
tau <- c(10, 30, 200, 90, 10, 20, 30, 40, 90, 20)
## y1 es la resultado potencial bajo el tratamiento
y1 <- y0 + tau
# Dos bloques: a y b
block <- c("a", "a", "a", "a", "a", "a", "b", "b", "b", "b")
# Z  es la asignación al tratamiento
# ( en l código usamos Z en vez de  T)
Z <- c(0, 0, 0, 0, 1, 1, 0, 0, 1, 1)
# Y es la resultado potencial observado
Y <- Z * y1 + (1 - Z) * y0
# Los datos
dat <- data.frame(Z = Z, y0 = y0, y1 = y1, tau = tau, b = block, Y = Y)
```

DeclareDesign

En DeclareDesign se pueden representar diseños de investigación en unos pocos pasos:

```
# # Seleccionar los resultados potenciales bajo control y tratamiento
small_dat <- dat[, c("y0", "y1")]

# El primer paso en DeclareDesign es declarar la población
pop <- declare_population(small_dat)

# 5 unidades asignadas al tratamiento; DD hace asignación simple
trt_assign <- declare_assignment(
  Z = conduct_ra(N = 10, m = 5),
  legacy = FALSE
)
# El valor observado de Y es y1 si Z=1 y y0 si Z=0
pot_out <- declare_potential_outcomes(Y ~ Z * y1 + (1 - Z) * y0)

# Especificar variable de resultado y asignación al tratamiento
reveal <- declare_reveal(Y, Z)

# El objeto de diseño de investigación básico
# incluye cuatro objetos
base_design <- pop + trt_assign + pot_out + reveal
```

DeclareDesign: creación de datos artificiales

DeclareDesign renombra `y0` and `y1` como `Y_Z_0` y `Y_Z_1` por defecto:

```
## Una simulación es una asignación aleatoria al tratamiento
set.seed(12345)
sim_dat1 <- draw_data(base_design)

# Datos simulados (sólo las primeras 6 lineas)
sim_dat1
```

```
    y0  y1 Z Y_Z_0 Y_Z_1   Y
1    0  10 1     0    10  10
2    0  30 1     0    30  30
3    0 200 0     0   200   0
4    1  91 1     1    91  91
5    1  11 0     1    11   1
6    3  23 1     3    23  23
7    4  34 0     4    34   4
8    5  45 1     5    45  45
9  190 280 0   190   280 190
10 200 220 0   200   220 200
```

Utilizando DeclareDesign: definiendo estimandos y estimadores

El siguiente código no produce ningun resultado. Sólo define las funciones, los estimadores y un estimando.

```
## El estimando
estimandATE <- declare_inquiry(ATE = mean(Y_Z_1 - Y_Z_0))

## El primer estimador es la diferencia de medias
diff_means <- declare_estimator(Y ~ Z,
  inquiry = estimandATE,
  .method = lm_robust, se_type = "classical", label = "Diff-Means/OLS"
)
```

DeclareDesign: definiendo estimandos y estimadores

```r
## El segundo estimador es la diferencia de medias recodificada (truncada)
diff_means_topcoded_fn <- function(data) {
  data$rankY <- rank(data$Y)
  ## Reemplace el valor del máximo de Y por el segundo valor más alto de Y
  data$newY <- with(
    data,
    ifelse(rankY == max(rankY), Y[rankY == (max(rankY) - 1)], Y)
  )
  obj <- lm_robust(newY ~ Z, data = data, se_type = "classical")
  res <- tidy(obj) %>% filter(term == "Z")
  return(res)
}
diff_means_topcoded <- declare_estimator(
  handler = label_estimator(diff_means_topcoded_fn),
  inquiry = estimandATE, label = "Top-coded Diff Means"
)
```

DeclareDesign: definiendo estimandos y estimadores

Extra: Que hace `rank`?

```
sim_dat1$Y
```

```
 [1]  10  30   0  91   1  23   4  45 190 200
```

```
rank(sim_dat1$Y, ties.method = "average")
```

```
 [1]  4  6  1  8  2  5  3  7  9 10
```

DeclareDesign: definiendo estimandos y estimadores

A continuación presentamos cómo funcionan los estimadores en Declare Design utilizando datos simulados.

```r
## Demuestra que el estimando funciona:
estimandATE(sim_dat1)
```

```
  inquiry estimand
1     ATE       54
```

```r
## Demuestra que los estimadores estiman

## Estimador1(diferencia de medias)
diff_means(sim_dat1)[-c(1, 2, 10, 11)]
```

```
  estimate std.error statistic p.value conf.low conf.high df
1    -39.2     49.41   -0.7934  0.4505   -153.1     74.74  8
```

```r
## Estimator 2 (diferencia de medias truncada)
diff_means_topcoded(sim_dat1)[-c(1, 2, 10, 11)]
```

```
  estimate std.error statistic p.value conf.low conf.high df
1    -37.2     48.21   -0.7716  0.4625   -148.4     73.98  8
```

Simulemos una aleatorización

Recordemos cuál es el ATE real:

```
trueATE <- with(sim_dat1, mean(y1 - y0))
with(sim_dat1, mean(Y_Z_1 - Y_Z_0))
```

```
[1] 54
```

Estos son los estimados de un experimento (una simulación de los datos) (recuerda que `Z` fue hecho arriba).

```
## Extra: Dos formas de calcular el
# estimador de las diferencia de medias
est_diff_means_1 <- with(sim_dat1, mean(Y[Z == 1]) - mean(Y[Z == 0]))
est_diff_means_2 <- coef(lm_robust(Y ~ Z,
  data = sim_dat1,
  se = "classical"
))[["Z"]]
c(est_diff_means_1, est_diff_means_2)
```

```
[1] -39.2 -39.2
```

Simulemos una aleatorización

```r
## dos formas de calcular la diferencia de medias acotada
sim_dat1$rankY <- rank(sim_dat1$Y)
sim_dat1$Y_tc <- with(sim_dat1, ifelse(rankY == max(rankY),
  Y[rankY == (max(rankY) - 1)], Y
))

est_topcoded_2 <- coef(lm_robust(Y_tc ~ Z,
  data = sim_dat1,
  se = "classical"
))[["Z"]]
c(est_topcoded_2)
```

```
[1] -37.2
```

¿Cómo se comportan nuestros estimadores para este diseño en particular?

Nuestras estimaciones varían según las aleatorizaciones. ¿Varían también nuestros dos estimadores de la misma manera?

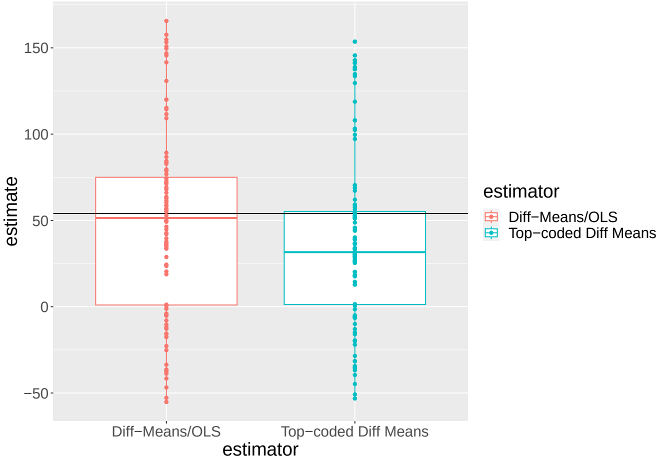
```r
## Combinar en un objeto diseño DeclareDesign
## Este tiene el diseño base, el estimando y luego nuestros dos estimadores
diff_means <- declare_estimator(Y ~ Z,
  inquiry = estimandATE,
  .method = lm_robust, se_type = "classical", label = "Diff-Means/OLS"
)

design_plus_ests <- base_design + estimandATE + diff_means +
  diff_means_topcoded
## Correr 100 simulaciones (reasignaciones del tratamiento) y
## utilizar los dos estimadores (diff_means y diff_means_topcoded)
diagnosis1 <- diagnose_design(design_plus_ests,
  bootstrap_sims = 0, sims = 100
)
sims1 <- get_simulations(diagnosis1)
head(sims1[, -c(1:6)])
```

```
  estimate std.error statistic p.value conf.low conf.high df outcome
1     82.8     63.82    1.2974  0.2306   -64.37     230.0  8       Y
2     44.0     57.45    0.7658  0.4688   -91.86     179.9  7    newY
3     89.2     62.07    1.4371  0.1886   -53.93     232.3  8       Y
4     51.7     56.08    0.9219  0.3873   -80.91     184.3  7    newY
```

¿Cómo se comportan nuestros estimadores para este diseño en particular?

Nuestras estimaciones varían según las aleatorizaciones. ¿Varían también nuestros dos estimadores de la misma manera?

¿Cómo interpretar esta gráfica?



¿Cuál estimador se acerca más al valor real?

Un criterio para elegir entre los estimadores es elegir el estimador que siempre esté más **cerca del valor real**, independientemente de la aleatorización específica.

Un estimador "insesgado" es aquel para el que **el promedio de las estimaciones en los diseños repetidos** es igual al valor real (o $E_R(\hat{\bar{\tau}}) = \bar{\tau}$).

Una cantidad para medir "la cercanía" al valor real es el **error cuadrático medio de la raíz** (RMSE, por sus siglas en inglés), que registra las distancias cuadráticas entre la verdad y las estimaciones individuales.

¿Cuál estimador se acerca más al valor real?

¿Cuál estimador es mejor? (Uno está más cerca del valor real en promedio (RMSE) y es más preciso. El otro no tiene un error sistemático: es insesgado).

```
  Inquiry              Estimator Mean Estimand Mean Estimate   Bias SD Estimate
1     ATE         Diff-Means/OLS         54.00         48.49  -5.51       53.70 5
2     ATE Top-coded Diff Means         54.00         36.56 -17.44       49.09 5
```

Estimadores sesgados e insesgados

Resumen:

- Siempre podemos *decidir* sobre los estimandos y estimadores
- Un buen estimador debe funcionar bien independientemente de la aleatorización particular que se esté considerando de un diseño dado. El que *funcione bien* puede significar que sea "insesgado" y/o un "error cuadrático medio bajo" (o "consistente", lo que quiere decir que a medida que el tamaño de la muestra aumenta el estimador se acerca más al valor real).
- Las simulaciones nos permiten saber qué tan bien trabaja un estimador para un estudio dado.