

# Why use an Average Treatment Effect? How to do statistical inference for average treatment effects?

*Jake Bowers*<sup>1, 2</sup>

EGAP Learning Days 4: Chile

<sup>1</sup> *Political Science & Statistics & NCSA @ University of Illinois*

<sup>2</sup> *White House Social and Behavioral Sciences Team*

*jwbowers@illinois.edu — <http://jakebowers.org>*

# Why randomize?

To facilitate interpretable statements about comparisons (i.e. to remove confounds, to decide on intervention).

To facilitate interpretable statements about information (i.e. to justify hypothesis tests and estimators.)



## Recall the ATE

Like hypotheses and imputation, the ATE can help with the fundamental problem of causal inference.



*I don't know the truth, but I can provide a good guess of the average causal effect.*

$i$	$z_i$	$y_i$	$y_{i1}$	$y_{i0}$
A	0	16	?	16
B	1	22	22	?
C	0	7	?	7
D	1	14	14	?
			$\overline{y_{i1}}$	$\overline{y_{i0}}$

$$\begin{aligned}\widehat{ATE} &= \overline{y_{i1}}|z_i=1 - \overline{y_{i0}}|z_i=0 \\ &= \frac{22+14}{2} - \frac{16+7}{2} = 6.5\end{aligned}$$

$E(\widehat{ATE}) = ATE$  means that  $\widehat{ATE}$  is unbiased for ATE.

```
## Bias refers to a relationship between the repeated operation of a procedure and a truth. So we have to invent a truth.
dat$y0<-dat$rpre ## create true potential outcomes to control
trueATE<-0.2 ## posit a true average treatment effect
dat$y1<-dat$y0+trueATE+rnorm(nrow(dat),mean=0,sd=sd(dat$y0)) ## create potential outcomes to treatment
dat$obsy<-with(dat, z*y1+(1-z)*y0 ) ## what we observe
trueATE<-with(dat, mean(y1)-mean(y0))
estATE<-coef(lm(obsy~z,dat))["z"] ## same as a mean difference on obsy
## Define two functions: (1) calc est ATE and (2) re-assign treatment
makeNewObsyAndEst<-function(thez){
  newobsy<-with(dat, thez*y1+(1-thez)*y0 )
  lmATE<-coef(lm(newobsy~thez))["thez"]
  return(c(lmATE=lmATE))
}
makeNewZ<-function(thez,theb){
  unsplit(lapply(split(thez,theb),sample),theb)
}
## Does the pair of functions do what we want them to do?
replicate(5,makeNewObsyAndEst(makeNewZ(dat$z,dat$s)))
```

lmATE	lmATE	lmATE	lmATE	lmATE
6.246452	1.432284	-1.547790	1.841729	5.240106

```

nsims <- 1000
set.seed(20150313)

## For many of the possible ways to run the experiment, calculate this mean difference
## dist.sample.est<-replicate(nsims,makeNewObsyAndEst(makeNewObsyAndEst(makeNewZ(dat$z, dat$s)
## on your unix-based machine (mac or linux):
require(parallel)
ncores <- detectCores()
dist.sample.est <- simplify2array(mclapply(1:nsims, function(i) {
  makeNewObsyAndEst(makeNewZ(dat$z, dat$s))
}, mc.cores = ncores))

c(EestATE = mean(dist.sample.est), ATE = trueATE, estATE = estATE)

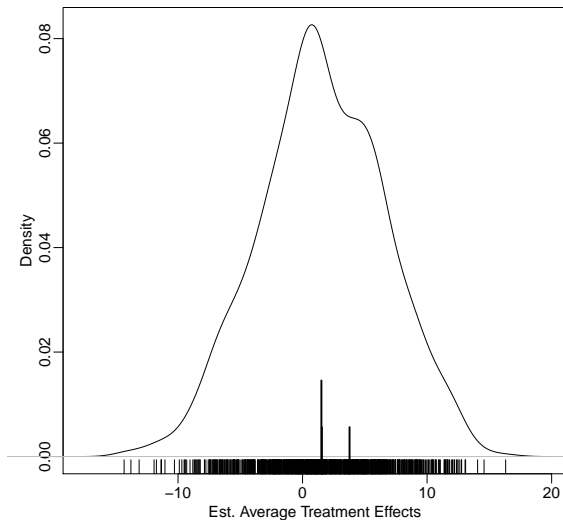
EestATE      ATE estATE.z
1.551555 1.513148 3.773564

## And recall that we have simulation error on the order of 1/sqrt(nsims)
SEsims <- sqrt(var(dist.sample.est)/nsims)
SEsims

[1] 0.1586654

```

What does it mean to say we have an unbiased estimator?







# Confidence Interval Ingredients

$$CI(ATE) = \widehat{ATE} \pm z_{\alpha/2} SE(\widehat{ATE})$$

where,  $z_{\alpha/2}$  for  $\alpha = .05$  is 1.96.

# Standard Errors for the Estimated ATE

What is a standard error in the context of a randomized experiment?

Here, pretending that the randomization was simple and not blocked.

```
## See the Dunning / Freedman, Pisani, Purves derivation
y0 <- dat$y0
y1 <- dat$y1
Z <- dat$z
Y <- Z * y1 + (1 - Z) * y0
V <- var(cbind(y0, y1))
varc <- V[1, 1]
vart <- V[2, 2]
covtc <- V[1, 2]
N <- length(y0)
n <- sum(Z)
m <- N - n
varestATE <- ((N - n)/(N - 1)) * (vart/n) + ((N - m)/(N - 1)) * (varc/m) + (2/(N - 1)) * covtc
## And the *feasible* version (where we do not observe the potential outcomes)
varYc <- var(Y[Z == 0])
varYt <- var(Y[Z == 1])
fvarestATE <- (N/(N - 1)) * ((varYt/n) + (varYc/m))
```

```
lm1 <- lm(Y ~ Z)
iidSElm1 <- sqrt(diag(vcov(lm1)))[["Z"]]

c(trueSE = sqrt(varestATE), feasible = sqrt(fvarestATE), iid = iidSElm1, simSE = sd(dif
  trueSE feasible      iid      simSE
4.745100 5.383047 5.326679 5.017441
```

# Different Confidence Intervals

```
theiidci <- confint(lm1, level = 0.95, parm = "Z")
feasCI <- estATE + c(1, -1) * qnorm(0.05/2) * sqrt(fvarestATE)
bestCI <- estATE + c(1, -1) * qnorm(0.05/2) * sqrt(varestATE)

rbind(feasCI, theiidci, bestCI)
```

	2.5 %	97.5 %
feasCI	-6.777015	14.32414
Z	-6.948484	14.49561
bestCI	-5.526661	13.07379

## Which is better?

A good test casts doubt on the truth rarely.

A good confidence interval contains the truth at least  $100\alpha$  % of the time. (Because a confidence interval is a collection of hypotheses against which we have little information to argue. A confidence interval is collection of unsurprising hypotheses.)

# Checking Coverage

```
makeFeasibleSE <- function(y, z) {  
  varYc <- var(y[z == 0])  
  varYt <- var(y[z == 1])  
  N <- length(y)  
  stopifnot(N == length(z)) ## a test of the code  
  fvarestATE <- (N/(N - 1)) * ((varYt/n) + (varYc/m))  
  return(fvarestATE)  
}  
  
makeCIs <- function(y, thez) {  
  lm1 <- lm(y ~ thez)  
  estATE <- coef(lm1)["thez"]  
  theiidci <- confint(lm1, level = 0.95, parm = "thez")  
  fvarestATE <- makeFeasibleSE(y = y, z = thez)  
  thefeasci <- estATE + c(1, -1) * qnorm(0.05/2) * sqrt(fvarestATE)  
  truthinIIDci <- 0 >= min(theiidci) & 0 <= max(theiidci)  
  truthinFeasci <- 0 >= min(thefeasci) & 0 <= max(thefeasci)  
  return(c(truthinIIDci = truthinIIDci, truthinFeasci = truthinFeasci))  
}
```

```
makeCIs(y = Y, thez = sample(Z))
```

truthinIIDci	truthinFeasci
TRUE	TRUE

# Checking Coverage

```
set.seed(20160509)
nsims <- 10000
coverageCheck <- simplify2array(mclapply(1:nsims, function(i) {
  makeCIs(y = Y, thez = sample(Z))
}, mc.cores = ncores))
## coverageCheck<-replicate(10000, makeCIs(y=Y,thez=sample(Z))) ##makeNewZ(Z,Y))
apply(coverageCheck, 1, mean)

truthinIIDci truthinFeasci
0.9528      0.9498
```

# Review

What is unbiasedness? Why do we care? How would we assess bias?

What is a confidence interval? How would we assess coverage?



