Hôpitaux Universitaires de Strasbourg

Résultats d'analyses statistiques pour

Julien GODET
MCU-PH
Pôle de Santé Publique
julien.godet@chru-strasbourg.fr
julien.godet@unistra.fr
August 11, 2021

Contents

Objective

Compare different methods to estimate ATE

- synthetic data with binary treatment, binary outcome and binomial/uniform/normal covariates
- known ψ (true-Psi) $E(Y^1 - Y^0)$
- sampling from the population to create observational data (removing counterfactuals)
- different sampling sizes from the fixed population (n = 5,000; 1,000; 500; 200; 100 and 50)
- estimations using TMLE, IPTW, misspecified IPTW model, g-formula for the different samples

☐ TODO : add IPTW with stabilized weights !!

Results

Sample Size (n)	true-Psi	TMLE	IPTW	IPTW (unm. confounding)	g-formula
5000	0.2203	0.216 [0.187 - 0.244]	0.223 [0.194 - 0.253]	0.257 [0.195 - 0.252]	0.218 [0.190 - 0.244]
1000	0.2203	0.224 [0.158 - 0.290]	0.236 [0.170 - 0.302]	0.271 [0.174 - 0.299]	0.236 [0.168 - 0.293]
500	0.2203	0.196 [0.110 - 0.281]	0.218 [0.119 - 0.316]	0.255 [0.123 - 0.312]	0.198 [0.104 - 0.290]
200	0.2203	0.191 [0.041 - 0.340]	0.217 [0.066 - 0.368]	0.259 [0.073 - 0.361]	0.188 [0.038 - 0.311]
100	0.2203	0.244 [0.071 - 0.418]	0.222 [0.011 - 0.433]	0.253 [0.018 - 0.427]	0.224 [0.040 - 0.382]
50	0.2203	0.241 [-0.011 - 0.494]	0.177 [-0.145 - 0.499]	0.278 [-0.125 - 0.479]	0.140 [-0.165 - 0.403]

unm. confounding : unmeasured confounding / misspecified PS model

Code

```
# JuG ------------------------------------------------------------------------
# Wed Aug 11 17:58:01 2021 -----------------------------

library(tidyverse)
library(SuperLearner)
set.seed(7)

# using data generating code from Miguel Angel Luque Fernandez' tutorial
#modified from Kat Hoffman @ https://github.com/hoffmakl/causal-club
generate_data <- function(n){
  w1 <- rbinom(n, size=1, prob=0.5)
  w2 <- rbinom(n, size=1, prob=0.65)
  w3 <- round(runif(n, min=0, max=4), digits=3)
  w4 <- round(runif(n, min=0, max=5), digits=3)
  w5 <- round(runif(n, min=0, max=5), digits=3)
  w6 <- round(rnorm(n, mean = 2, sd = .5), digits=3)
  w7 <- round(rnorm(n, mean = 0, sd=.3), digits=3)
  w8 <- rbinom(n, size=1, prob=0.1)

  A  <- rbinom(n, size=1, prob= plogis(-2+ 0.2*w2 + 0.15*w3 + 0.2*w4 + 0.15*w2*w4 + .3 * w8))
  # counterfactual
  Y_1 <- rbinom(n, size=1, prob= plogis(-1.5 + 1 -0.1*w1 + 0.3*w2 + 0.25*w3 + 0.2*w4 + 0.15*w2*w4 + .3 * w8))
  Y_0 <- rbinom(n, size=1, prob= plogis(-1.5 + 0 -0.1*w1 + 0.3*w2 + 0.25*w3 + 0.2*w4 + 0.15*w2*w4 + .3 * w8))
  # Observed outcome
  Y <- Y_1*A + Y_0*(1 - A)
  # return data.frame
  tibble(w1, w2, w3, w4, w5, w6, w7, w8, A, Y, Y_1, Y_0)
}

# observations N
n <- 100000

# full data set, including Y0 and Y0
dat_full <- generate_data(n)

# calculate the true psi if we saw both outcomes
true_psi <- mean(dat_full$Y_1 - dat_full$Y_0)
true_psi
#NOTE: keep this upper part as it is - change nsamp size below!

# make a data set with observed data only
nsamp <- 50
dat_obs <- dat_full %>%
  dplyr::select(-Y_1,-Y_0) %>%
  sample_n(nsamp)
```

```r
table(dat_obs$A)
table(dat_obs$Y)
# set SL libraries
lib <- c('SL.speedglm', # faster glm
         'SL.glmnet', # lasso
         'SL.ranger', # random forest
         'SL.earth')  #

Y <- dat_obs$Y
A <- dat_obs$A
X_A <- dat_obs %>% dplyr::select(-Y, -A)

# Compare with TMLE package ------------------------------------------------------------

tmle_fit <- tmle::tmle(Y, A, X_A,
                       gbound = .0001, # trying
                       Q.SL.library = lib, g.SL.library = lib)
tmle_fit$epsilon # mine are different :(
tmle_fit$estimates$ATE


# g-formula----------------------------------------------------------------


library(RISCA)
#Marginal effects of the treatment (ATE)
dat_obs2 <- as.data.frame(dat_obs) #NOTE does not work with tibble!

glm.multi <- glm(Y ~ ., data=dat_obs2, family = binomial)

gc.ate <- gc.logistic(glm.obj=glm.multi, data=dat_obs2, group="A", effect="ATE",
                      var.method="simulations", iterations=1000)

#-------------------------------------------------------------------------------
#-------------------------------------------------------------------------------

# IPTW
library(survey)
library(tableone)
library(sandwich) #for robust variance estimation

psModel <- glm(A~., data= dat_obs %>% dplyr::select(-Y), family = "binomial")
# require(MASS)
# stepAIC(psModel)
# psModel <- glm(formula = A ~  w1+w2+w4+w8, family = "binomial",
#                data = dat_obs)
#psModel <- glm(A~w1 + w2 + w3 + w4, data= dat_obs, family = "binomial")

## value of propensity score for each subject
ps <-predict(psModel, type = "response")

#create weights
weight<-ifelse(dat_obs$A==1,1/(ps),1/(1-ps))

#apply weights to data
weighteddata<-svydesign(ids = ~ 1, data =dat_obs, weights = ~ weight)

#weighted table 1
weightedtable <-svyCreateTableOne(vars = paste("w", 1:8, sep=""), strata = "A",
                                  data = weighteddata, test = FALSE)
## Show table with SMD
print(weightedtable, smd = TRUE)

#get causal risk difference
glm.obj<-glm(Y~A,weights=weight,family=binomial(link="log"), data=dat_obs)




summary(weight)
#truncate weights at 5

truncweight<-replace(weight,weight>5,5)
#get causal risk difference
glm.obj<-glm(Y~A,weights=truncweight,family=quasibinomial(link="identity"), data=dat_obs)
#summary(glm.obj)
betaiptw<-coef(glm.obj)

SE<-sqrt(diag(vcovHC(glm.obj, type="HC0")))
```

```r
causalrd<-(betaiptw[2])
lcl<-(betaiptw[2]-1.96*SE[2])
ucl<-(betaiptw[2]+1.96*SE[2])
c(lcl,causalrd,ucl)



# misspecified model ------------------------------------------------------------------

mpsModel <- glm(A~w1+w2+w3+w7, data= dat_obs %>% dplyr::select(-Y), family = "binomial")
# require(MASS)
# stepAIC(psModel)
# psModel <- glm(formula = A ~  w1+w2+w4+w8, family = "binomial",
#                data = dat_obs)
#psModel <- glm(A~w1 + w2 + w3 + w4, data= dat_obs, family = "binomial")

## value of propensity score for each subject
mps <-predict(mpsModel, type = "response")

#create weights
mweight<-ifelse(dat_obs$A==1,1/(mps),1/(1-mps))

#apply weights to data
mweighteddata<-svydesign(ids = ~ 1, data =dat_obs, weights = ~ mweight)

#weighted table 1
mweightedtable <-svyCreateTableOne(vars = paste("w", 1:8, sep=""), strata = "A",
                                   data = mweighteddata, test = FALSE)
## Show table with SMD
print(mweightedtable, smd = TRUE)

#get causal risk difference
glm.obj<-glm(Y~A,weights=mweight,family=binomial(link="log"), data=dat_obs)
summary(mweight)

#truncate weights at 5
mtruncweight<-replace(mweight,mweight>5,5)
#get causal risk difference
mglm.obj<-glm(Y~A,weights=mtruncweight,family=quasibinomial(link="identity"), data=dat_obs)
#summary(glm.obj)
mbetaiptw<-coef(mglm.obj)

SE<-sqrt(diag(vcovHC(mglm.obj, type="HC0")))

mcausalrd<-(mbetaiptw[2])
mlcl<-(betaiptw[2]-1.96*SE[2])
mucl<-(betaiptw[2]+1.96*SE[2])
c(mlcl,mcausalrd,mucl)

# output ------------------------------------------------------------------

# dput(c(paste(nsamp),
#        paste(format(round(true_psi,4), nsmall=4)),
# paste(format(round(tmle_fit$estimates$ATE$psi, 3), nsmall=3), " [",
#       format(round(tmle_fit$estimates$ATE$CI[1], 3),nsmall=3), " - ",
#       format(round(tmle_fit$estimates$ATE$CI[2], 3),nsmall=3), "]", sep=""),
# paste(format(round(causalrd, 3), nsmall=3), " [", format(round(lcl, 3),nsmall=3), " - ",
#       format(round(ucl, 3),nsmall=3), "]", sep=""),
# paste(format(round(mcausalrd, 3), nsmall=3), " [", format(round(mlcl, 3),nsmall=3), " - ",
#       format(round(mucl, 3),nsmall=3), "]", sep=""),
# paste(format(round(gc.ate$delta[[1]], 3), nsmall=3), " [",
#       format(round(gc.ate$delta[[3]], 3),nsmall=3), " - ",
#       format(round(gc.ate$delta[[4]], 3),nsmall=3), "]",
#       sep="")))
```