Day 1, Practical 2

Helene Charlotte Wiese Rytgaard

June 5, 2023

In this practical we will continue to work with the simulated data from Practical 1. The point of this part is to explore validity of TMLE (and likewise EE) inference for ATE estimation. More specifically, we aim to:

1. Utilize simulation studies to investigate and assess the assumptions underlying inference based on the efficient influence curve.
2. Assess the similarities and potential finite-sample differences between efficient influence curve based estimators in simulated settings with and without positivity violations.

Particularly, inference for efficient influence curve equation based estimators (with standard errors estimated from the empirical variance of the efficient influence function) is only valid when both the propensity score π and the outcome regression f are consistently estimated, and at a fast enough rate.

Repetition: The asymptotic properties of the TMLE estimator and the estimating equation (EE) estimator are the same, because they are both (although in different ways) constructed such as to solve the efficient influence curve equation.

1 Simulate data

We will work with the simulation function defined in the first practical.

Task 1: Use the simulation function from the first practicals from day 1 (Task 1) to draw a random dataset with sample size $n = 1000$.
Recall that the true value of the average treatment effect (ATE) can be approximated as follows:

```
set.seed(12)
message(paste0("EY1 = ", E.Y1 <- sim.fun(1e6, a=1)))
message(paste0("EY0 = ", E.Y0 <- sim.fun(1e6, a=0)))
message(paste0("ATE = ", ATE <- E.Y1 - E.Y0))
```

```
EY1 = 0.749921
EY0 = 0.68208
ATE = 0.0678409999999999
```

2 Implement the estimating equation estimator and its variance

Task 2: Implement the estimating equation estimator, as outlined below:

1. Fit the models below for the outcome regression f and the propensity score π. Use `fit.f` to predict the conditional expectations $\mathbb{E}_P[Y \mid A, X]$ and $\mathbb{E}_P[Y \mid A = a, X]$. Add these as columns to the dataset. Use `fit.pi` to estimate the propensity score $\pi(a \mid X) = P(A = a \mid X)$. Add this as a column to the simulated dataset.

```
fit.f <- glm(Y~A+X1+X2+X3, family=binomial, data=sim.data)
fit.pi <- glm(A~X1+X2+X3, family=binomial, data=sim.data)
```

2. Implement $\hat{\psi}_n^{\text{ee}}$ based on Equation (1):

$$\begin{aligned}\hat{\psi}_n^{\text{ee}} &= \tilde{\Psi}_{\text{ee}}(\hat{f}_n, \hat{\pi}_n, \hat{P}_n) \\ &= \frac{1}{n}\left\{\left(\frac{A_i}{\hat{\pi}_n(1 \mid X_i)} - \frac{1 - A_i}{\hat{\pi}_n(0 \mid X_i)}\right)\big(Y_i - \hat{f}_n(A_i, X_i)\big) + \hat{f}_n(1, X_i) - \hat{f}_n(0, X_i)\right\}. \qquad (1)\end{aligned}$$

Comments. As we have seen, this estimator uses the representation for the target parameter:

$$\tilde{\Psi}_{\text{ee}}(f, \pi, p) = \mathbb{E}_P\left[\left(\frac{A}{\pi(A \mid X)} - \frac{1 - A}{\pi(A \mid X)}\right)\big(Y - f(A, X)\big) + f(1, X) - f(0, X)\right],$$

involving really an average over all but the last terms of the efficient influence curve:

$$\begin{aligned}\phi^*(P)(O) &= \tilde{\phi}^*(f, \pi)(O) \\ &= \left(\frac{A}{\pi(A \mid X)} - \frac{1 - A}{\pi(A \mid X)}\right)\big(Y - f(A, X)\big) + f(1, X) - f(0, X) - \Psi(P).\end{aligned}$$

Particularly, $\hat{\psi}_n^{\text{ee}}$ solves by construction the efficient influence equation:

$$\begin{aligned}&\mathbb{P}_n\tilde{\phi}^*(\hat{f}_n, \hat{\pi}_n) = \frac{1}{n}\sum_{i=1}^{n} \tilde{\phi}^*(\hat{f}_n, \hat{\pi}_n) \\ &= \frac{1}{n}\left\{\left(\frac{A_i}{\hat{\pi}_n(1 \mid X_i)} - \frac{1 - A_i}{\hat{\pi}_n(0 \mid X_i)}\right)\big(Y_i - \hat{f}_n(A_i, X_i)\big) + \hat{f}_n(1, X_i) - \hat{f}_n(0, X_i)\right\} - \hat{\psi}_n^{\text{ee}} \\ &= 0.\end{aligned}$$

You can check this in your implementation, but you may also just note that it is trivial.

3. Implement the variance estimator based on Equation (2) below, and compute corresponding confidence intervals. Discuss if inference based on (2) is valid.

$$\hat{\sigma}_n^2 = \mathbb{P}_n\{\tilde{\phi}^*(\hat{f}_n, \hat{\pi}_n)\}^2/n = \frac{1}{n^2}\sum_{i=1}^{n}\{\tilde{\phi}^*(\hat{f}_n, \hat{\pi}_n)(O)\}^2 \qquad (2)$$

Comments. Recall the following decomposition in analyzing the large-sample properties of an estimator:

$$\hat{\psi}_n^{\text{ee}} - \Psi(P_0) = \mathbb{P}_n\phi^*(P_0) + o_P(n^{-1/2}) + R(\hat{P}_n, P_0) - \underbrace{\mathbb{P}_n\phi^*(\hat{P}_n)}_{=0};$$

when $R(\hat{P}_n, P_0) = o_P(n^{-1/2})$, then $\Psi(\hat{P}_n) \overset{as}{\sim} N(\Psi(P_0), P_0\phi^*(P_0)^2/n)$, and the variance of the estimator can be estimated by Equation (2). Whether $R(\hat{P}_n, P_0) = o_P(n^{-1/2})$ depends on the performance of estimation of $f(a,x) = \mathbb{E}[Y \mid A = a, X = x]$ and $\pi(a \mid x) = P(A = a \mid X = x)$. When this hold, confidence intervals can be computed as $\hat{\psi}_n^{\text{ee}} \pm 1.96\,\text{SE}(\hat{\psi}_n^{\text{ee}})$.

3 Compare with the TMLE estimator

Task 3. Load the `tmle` package and use the `tmle()` function to get the TMLE estimate and variance using the same models as in **Task 2** using the code below. Check that you get about the same, and comment.

```
library(tmle)
tmle.fit <- tmle(Y=sim.data$Y, A=sim.data$A,
         cbind(X1=sim.data$X1,
               X2=sim.data$X2,X3=sim.data$X3),
         gform=A~X1+X2+X3, ## treatment model
         Qform=Y~A+X1+X2+X3, ## outcome model
         family="binomial",
         cvQinit=FALSE)
#-- get the ATE estimate:
tmle.fit$estimates$ATE$psi
#-- get the variance estimate:
tmle.fit$estimates$ATE$var
```

4 Look at results of simulation studies

Task 4. You can access simulation results from **Task 9** (Practical 1) as follows by downloading the file:

data/sim-data-output/save-est-sim-setting-1.rds

from github. Load it to R as below (changing the path):

```
library(here)
estimator.list <- readRDS(paste0(here(), "/data/sim-data-output/",
                 "save-est-sim-setting-1",
                 ".rds"))
```

Task 5. The code below shows how vectors of estimates and estimated variances saved across the simulation repetitions are extracted from the object above. These results are for correctly specified models. Compute the bias, variance, mean squared error and coverage for the TMLE estimator and the estimating equation (EE) estimators. Comment on the results.

```
fit.tmle <- unlist(estimator.list$fit.tmle2)
fit.ee <- unlist(estimator.list$fit.ee2)
```

```
var.tmle <- unlist(estimator.list$fit.tmle2.var)
var.ee <- unlist(estimator.list$fit.ee2.var)
```

Task 6. The code below shows how vectors of estimates and estimated variances saved across the simulation repetitions are extracted from the object above, for the situation when misspecified models were used. Compute the bias, variance, mean squared error and coverage of confidence intervals for the TMLE estimator and the estimating equation (EE) estimators. Comment on the results.

```
fit.miss.tmle <- unlist(estimator.list$fit.tmle)
fit.miss.ee <- unlist(estimator.list$fit.ee)
var.miss.tmle <- unlist(estimator.list$fit.tmle.var)
var.miss.ee <- unlist(estimator.list$fit.ee.var)
```

Task 7. You can access simulation results from a simulation study with positivity issues as in **Task 11** of Practical 1 by downloading the file:

data/sim-data-output/save-est-sim-setting-2.rds

from github. Load it to R as below (changing the path), and then repeat **Task 5** and **Task 6** above. Comment on the results.

```
library(here)
estimator.list <- readRDS(paste0(here(), "/data/sim-data-output/",
                  "save-est-sim-setting-2",
                  ".rds"))
```

NB: Note that for the data setting analyzed in **Task 7**, TMLE results are produced both with and without weight truncation. Those *with* weight truncation are accessed as follows:

```
#-- correctly specified:
fit.wt.tmle <- unlist(estimator.list$fit.wt.tmle2)
var.wt.tmle <- unlist(estimator.list$fit.wt.tmle2.var)
#-- misspecified:
fit.wt.miss.tmle <- unlist(estimator.list$fit.wt.tmle)
var.wt.miss.tmle <- unlist(estimator.list$fit.wt.tmle.var)
```