A guided tour in targeted learning territory

David Benkeser, Antoine Chambaz, Nima Hejazi

10/11/2017

Contents

1 Introduction **1**

2 A simulation study **1**
2.1 Reproducible experiment as a law. 1
2.2 The parameter of interest, first pass. 3
2.3 The parameter of interest, second pass. 4

1 Introduction

This is a very first draft of our article. The current *tentative* title is "A guided tour in targeted learning territory".

Explain our objectives and how we will meet them.

Use sectioning a lot to ease cross-referencing.

```
set.seed(54321) ## because reproducibility matters...
suppressMessages(library(R.utils)) ## make sure it is installed
expit <- plogis
logit <- qlogis
```

2 A simulation study

blabla

2.1 Reproducible experiment as a law. We are interested in a reproducible experiment. The generic summary of how one realization of the experiment unfolds, our observation, is called O. We view O as a random variable drawn from what we call the law P_0 of the experiment. The law P_0 is viewed as an element of what we call the model. Denoted by $\mathcal{M}$, the model is the collection of *all* laws from which O can be drawn and that meet some constraints. The constraints translate the knowledge we have about the experiment. The more we know about the experiment, the smaller is $\mathcal{M}$. In all our examples, model $\mathcal{M}$ will put very few restrictions on the candidate laws.

Consider the following chunk of code:

```
drawFromExperiment <- function(n, full = FALSE) {
  ## preliminary
  n <- Arguments$getInteger(n, c(1, Inf))
  full <- Arguments$getLogical(full)
  ## ## 'gbar' and 'Qbar' factors
  gbar <- function(W) {
    expit(-0.3 + 2 * W - 1.5 * W^2)
  }
```

```
  Qbar <- function(AW) {
    A <- AW[, 1]
    W <- AW[, 2]
    A * cos(2 * pi * W) + (1 - A) * sin(2 * pi * W^2)
  }
  ## sampling
  ## ## context
  W <- runif(n)
  ## ## counterfactual rewards
  zeroW <- cbind(A = 0, W)
  oneW <- cbind(A = 1, W)
  Yzero <- rnorm(n, mean = Qbar(zeroW), sd = 1)
  Yone <- rnorm(n, mean = Qbar(oneW), sd = 1)
  ## ## action undertaken
  A <- rbinom(n, size = 1, prob = gbar(W))
  ## ## actual rewards
  Y <- A * Yone + (1 - A) * Yzero
  ## ## observation
  if (full) {
    obs <- cbind(W = W, Yzero = Yzero, Yone = Yone, A = A, Y = Y)
  } else {
    obs <- cbind(W = W, A = A, Y = Y)
  }
  attr(obs, "gbar") <- gbar
  attr(obs, "Qbar") <- Qbar
  attr(obs, "QW") <- dunif
  ##
  return(obs)
}
```

We can interpret `drawFromExperiment` as a law P_0 since we can use the function to sample observations from a common law. It is even a little more than that, because we can tweak the experiment, by setting its `full` argument to `TRUE`, in order to get what appear as intermediary (counterfactual) variables in the regular experiment. The next chunk of code runs the (regular) experiment five times independently:

```
five.obs <- drawFromExperiment(5)
five.obs
```

```
##              W A          Y
## [1,] 0.4290078 1 -0.4410530
## [2,] 0.4984304 0  0.1905250
## [3,] 0.1766923 0 -0.1325567
## [4,] 0.2743935 1 -1.6415356
## [5,] 0.2165102 0  0.6829717
## attr(,"gbar")
## function (W)
## {
##     expit(-0.3 + 2 * W - 1.5 * W^2)
## }
## <bytecode: 0x41435a0>
## <environment: 0x3ca0fa0>
## attr(,"Qbar")
## function (AW)
## {
##     A <- AW[, 1]
```

```
##     W <- AW[, 2]
##     A * cos(2 * pi * W) + (1 - A) * sin(2 * pi * W^2)
## }
## <bytecode: 0x37e9560>
## <environment: 0x3ca0fa0>
## attr(,"QW")
## function (x, min = 0, max = 1, log = FALSE)
## .Call(C_dunif, x, min, max, log)
## <bytecode: 0x3c35ff0>
## <environment: namespace:stats>
```

The `attributes` of the object `obs` are visible because, in this section, we act as oracles, *i.e.*, we know completely the nature of the experiment. From a probabilistic point of view, the attributes `gbar`, `Qbar` and `QW` are infinite-dimensional features of P_0. There is more to P_0 than $\bar{g}_0$ (`gbar`), $\bar{Q}_0$ (`Qbar`), formally defined by

$$\bar{g}_0(W) \equiv P_0(A = 1|W), \quad \bar{Q}_0(A, W) \equiv E_{P_0}(Y|A, W), \tag{1}$$

and the marginal distribution $Q_{0,W}$ of W under P_0 (`QW`), for instance the conditional distribution (not expectation) of Y given (A, W), but $\bar{g}_0$, $\bar{Q}_0$ and $Q_{0,W}$ will play a prominent role in our story.

2.2 The parameter of interest, first pass. It happens that we especially care for a finite-dimensional feature of P_0 that we denote by ψ_0. Its definition involves the aforementioned infinite-dimensional features:

$$\begin{aligned}\psi_0 &\equiv E_{P_0}\left(\bar{Q}_0(1, W) - \bar{Q}_0(0, W)\right) \\ &= \int \left(\bar{Q}_0(1, w) - \bar{Q}_0(0, w)\right) dQ_{0,W}(w).\end{aligned}$$

Acting as oracles, we can compute explicitely the numerical value of ψ_0.

Our interest in ψ_0 is of causal nature. Taking a closer look at `drawFromExperiment` reveals indeed that the random making of an observation O drawn from P_0 can be summarized by the following causal graph and nonparametric system of structural equations:

```
## plot the causal diagram
```

and, for some deterministic functions f_w, f_a, f_y and independent sources of randomness U_w, U_a, U_y,

1. sample the context where the rest of the experiment will take place, $W = f_w(U_w)$;
2. sample the two counterfactual rewards of the two actions that can be undertaken, $Y_0 = f_y(0, W, U_y)$ and $Y_1 = f_y(1, W, U_y)$;
3. sample which action is carried out in the given context, $A = f_a(W, U_a)$;
4. define the corresponding reward, $Y = AY_1 + (1 - A)Y_0$;
5. summarize the course of the experiment with the observation $O = (W, A, Y)$, thus concealing Y_0 and Y_1.

The above description of the experiment `drawFormExperiment` is useful to ram home what it means to run the "full" experiment by setting argument `full` to `TRUE` in a call to `drawFormExperiment`. Doing so triggers a modification of the nature of the experiment, enforcing that the counterfactual rewards Y_0 and Y_1 be part of the summary of the experiment eventually. In light of the above enumeration, $\mathbb{O} \equiv (W, Y_0, Y_1, A, Y)$ is output, as opposed to its summary measure O. This defines another experiment and its law, that we denote $\mathbb{P}_0$.

It is well known (do we give the proof or refer to other articles?) that

$$\psi_0 = E_{\mathbb{P}_0}\left(Y_1 - Y_0\right).$$

Thus, ψ_0 compares (additively) the averages of the two counterfactual rewards. In other words, ψ_0 quantifies the difference in average of the reward one would get in a world where one would always enforce action $a = 1$ with the reward one would get in a world where one would always enforce action $a = 0$. This said, it is worth emphasizing that ψ_0 is a well-defined parameter beyond its causal interpretation.

To conclude this subsection, we draw advantage from the possibility to sample full observations from `drawFromExperiment` by setting its argument `full` to `TRUE` in order to numerically approximate ψ_0. By the law of large numbers, the following chunk of code approximates ψ_0:

```
B <- 1e6
full.obs <- drawFromExperiment(B, full = TRUE)
psi.hat <- mean(full.obs[, "Yone"] - full.obs[, "Yzero"])
psi.hat
```

```
## [1] -0.170182
```

In fact, the central limit theorem and Slutsky's lemma allow us to build a confidence interval with asymptotic level 95% for ψ_0:

```
sd.hat <- sd(full.obs[, "Yone"] - full.obs[, "Yzero"])
alpha <- 0.05
psi.CI <- psi.hat + c(-1, 1) * qnorm(1 - alpha / 2) * sd.hat / sqrt(B)
psi.CI
```

```
## [1] -0.1738905 -0.1664735
```

2.3 The parameter of interest, second pass.