Day 2, Lecture 1

Targeted Minimum Loss-based Estimation (TMLE)

Overview of today

Before lunch (9 – 12):

- Targeted Minimum Loss-based Estimation (TMLE).
- The targeting step: updating/modifying initial nuisance parameter estimators.
- The ATE as a concrete example; the ATT as a different example.
- Valid inference still requires strong initial learners.

* TMLE as a two-step procedure with involving an initial estimation step followed by a targeting step.
* Implementation of the targeting step.
* The link between the theoretical decomposition from yesterday, and TMLE as a practical estimation method.

After lunch (13 – 15): Super learning.

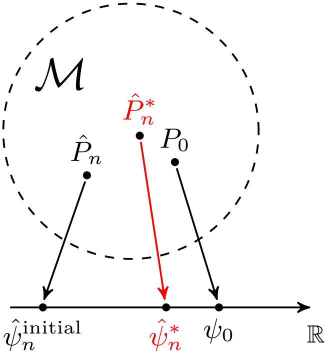
Targeted learning

1. Data is a random variable O with a probability P_0
2. P_0 belongs to a statistical model $\mathcal{M}$
3. Our target is a parameter $\Psi : \mathcal{M} \to \mathbb{R}$
4. Construct estimator $\hat{P}_n$ for (relevant part of) P_0 and estimate the target parameter by $\hat{\psi}_n = \Psi(\hat{P}_n)$
5. Quantify uncertainty for the estimator $\hat{\psi}_n = \Psi(\hat{P}_n)$

Estimation paradigm

1. P_0 is assumed to belong to a nonparametric model $\mathcal{M}$
2. Construction of $\sqrt{n}$-consistent and asymptotically linear estimation of $\psi_0 = \Psi(P_0)$ based the efficient influence function.

Targeted Minimum Loss-based Estimation (TMLE)



Tools from semiparametric efficiency theory and empirical process theory tell us how to construct an optimal estimator for a **given target parameter** $\Psi : \mathcal{M} \to \mathbb{R}$

- asymptotic linearity/normality
- asymptotic efficiency

Targeted Minimum Loss-based Estimation (TMLE)

TMLE is a two-step procedure:

Step 1 Construct initial estimator $\hat{P}_n$ for P.

Step 2 Update the estimator $\hat{P}_n \mapsto \hat{P}_n^*$ such that $\hat{P}_n^*$ solves the efficient influence curve equation, i.e.,

$$\mathbb{P}_n \phi^*(\hat{P}_n^*) = \frac{1}{n} \sum_{i=1}^{n} \phi^*(\hat{P}_n^*)(O_i) \approx 0.$$

Step 1 = "initial estimation step"

Step 2 = "targeting step"

Targeted Minimum Loss-based Estimation (TMLE)

$$\Psi(\hat{P}_n) - \Psi(P_0) = \mathbb{P}_n \phi^*(P_0) + o_P(n^{-1/2})$$
$$+ R(\hat{P}_n, P_0)$$
$$- \mathbb{P}_n \phi^*(\hat{P}_n)$$

- The role of the targeting step (Step 2):
  - Gain double robustness in consistency.
  - Easier to achieve asymptotic lineariy (amounts to getting rid of second-order remainder).
- The role of the initial estimation step (Step 1):
  - This should be done well enough to get rid of the second-order remainder.

Targeted Minimum Loss-based Estimation (TMLE)

$$\boxed{f(A, X) = \mathbb{E}_P[Y \mid A, X]}$$

A **loss function** $\mathscr{L}(f)(O)$ measuring the distance between an estimator f and the observed outcome Y, e.g., the negative log-likelihood:

$$\mathscr{L}(\hat{f}_n)(Y_i, A_i, X_i) = -(Y_i \log(\hat{f}_n(A_i, X_i)) + (1 - Y_i) \log(1 - \hat{f}_n(A_i, X_i))).$$

- The estimator $\hat{f}_n$ closest to the true f_0 minimizes the risk:

$$\mathbb{E}_{P_0}[\mathscr{L}(\hat{f}_n)(Y_i, A_i, X_i)].$$

- Loss-based super learning: Minimizing the cross-validated empirical risk with respect to the loss function $\mathscr{L}$ over the statistical model.

Targeted Minimum Loss-based Estimation (TMLE)

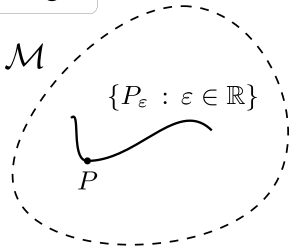
This is all about constructing a good estimator for the conditional expectation f;

- does not necessarily yield a good estimator for the particular feature of interest, the target parameter.

This is Step 1.

Targeted Minimum Loss-based Estimation (TMLE)

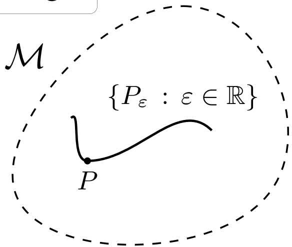
Step 2: We can minimize along a loss function in a certain way *that results in a good estimator for the target*.



Loss function $\mathscr{L}(f)(O)$ + clever choice of a **parametric submodel** $\{P_\varepsilon : \varepsilon \in \mathbb{R}\} \subset \mathcal{M}$.

Targeted Minimum Loss-based Estimation (TMLE)

Step 2: We can minimize along a loss function in a certain way *that results in a good estimator for the target*.

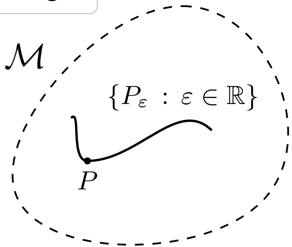


Loss function $\mathscr{L}(f)(O)$ + clever choice of a **parametric submodel** $\{P_\varepsilon : \varepsilon \in \mathbb{R}\} \subset \mathcal{M}$.

$\Rightarrow$ minimize the loss along the submodel, given the estimator $\hat{f}_n$ from Step 1.

Targeted Minimum Loss-based Estimation (TMLE)

Step 2: We can minimize along a loss function in a certain way *that results in a good estimator for the target*.



Loss function $\mathscr{L}(f)(O)$ + clever choice of a **parametric submodel** $\{P_\varepsilon : \varepsilon \in \mathbb{R}\} \subset \mathcal{M}$.

- $\Rightarrow$ minimize the loss along the submodel, given the estimator $\hat{f}_n$ from Step 1.
- $\Rightarrow$ update $\hat{f}_n$ along the path defined by P_ε: moving by $\hat{\varepsilon}_n$ that minimizes the loss.

The targeting step (Step 2)

Construction of the targeting step for a given target parameter $\Psi : \mathcal{M} \to \mathbb{R}$ with efficient influence function $\phi^*(P)$ requires:

(i) A parametric submodel $\{P_\varepsilon : \varepsilon \in \mathbb{R}\} \subset \mathcal{M}$

(ii) A loss function $(O, P) \mapsto \mathscr{L}(P)(O)$

such that: (1) $P_{\varepsilon=0} = P$, and, (2) $\left.\dfrac{d}{d\varepsilon}\right|_{\varepsilon=0} \mathscr{L}(P_\varepsilon)(O) = \phi^*(P)(O)$

The targeting step (Step 2)

Construction of the targeting step for a given target parameter $\Psi : \mathcal{M} \to \mathbb{R}$ with efficient influence function $\phi^*(P)$ requires:

(i) A parametric submodel $\{P_\varepsilon : \varepsilon \in \mathbb{R}\} \subset \mathcal{M}$

(ii) A loss function $(O, P) \mapsto \mathscr{L}(P)(O)$

such that: (1) $P_{\varepsilon=0} = P$, and, (2) $\left.\frac{d}{d\varepsilon}\right|_{\varepsilon=0} \mathscr{L}(P_\varepsilon)(O) = \phi^*(P)(O)$

- Initial estimator $\hat{P}_n^0$
- Minimizer $\hat{\varepsilon}_{n,0}$ of $\varepsilon \mapsto \mathbb{P}_n \mathscr{L}(\hat{P}^0_{n,\varepsilon})$
- Update: $\hat{P}_n^1 := \hat{P}^0_{\hat{\varepsilon}_{n,0}}$

Then: $$\mathbb{P}_n \left.\frac{d}{d\varepsilon}\right|_{\varepsilon=\hat{\varepsilon}_{n,0}} \mathscr{L}(\hat{P}^0_{n,\varepsilon})(O) = 0$$

The targeting step (Step 2)

Construction of the targeting step for a given target parameter $\Psi : \mathcal{M} \to \mathbb{R}$ with efficient influence function $\phi^*(P)$ requires:

(i) A parametric submodel $\{P_\varepsilon : \varepsilon \in \mathbb{R}\} \subset \mathcal{M}$

(ii) A loss function $(O, P) \mapsto \mathscr{L}(P)(O)$

such that: (1) $P_{\varepsilon=0} = P$, and, (2) $\left.\frac{d}{d\varepsilon}\right|_{\varepsilon=0} \mathscr{L}(P_\varepsilon)(O) = \phi^*(P)(O)$

- Updated estimator $\hat{P}_n^1$
- Minimizer $\hat{\varepsilon}_{n,1}$ of $\varepsilon \mapsto \mathbb{P}_n \mathscr{L}(\hat{P}^1_{n,\varepsilon})$
- Update: $\hat{P}_n^2 := \hat{P}^1_{\hat{\varepsilon}_{n,1}}$

Then: $$\mathbb{P}_n \left.\frac{d}{d\varepsilon}\right|_{\varepsilon=\hat{\varepsilon}_{n,1}} \mathscr{L}(\hat{P}^1_{n,\varepsilon})(O) = 0$$

The targeting step (Step 2)

Construction of the targeting step for a given target parameter $\Psi : \mathcal{M} \to \mathbb{R}$ with efficient influence function $\phi^*(P)$ requires:

(i) A parametric submodel $\{P_\varepsilon : \varepsilon \in \mathbb{R}\} \subset \mathcal{M}$

(ii) A loss function $(O, P) \mapsto \mathscr{L}(P)(O)$

such that: (1) $P_{\varepsilon=0} = P$, and, (2) $\left.\dfrac{d}{d\varepsilon}\right|_{\varepsilon=0} \mathscr{L}(P_\varepsilon)(O) = \phi^*(P)(O)$

- kth updated estimator $\hat{P}_n^k$
- Minimizer $\hat{\varepsilon}_{n,k}$ of $\varepsilon \mapsto \mathbb{P}_n \mathscr{L}(\hat{P}^k_{n,\varepsilon})$
- Update: $\hat{P}_n^{k+1} := \hat{P}^k_{\hat{\varepsilon}_{n,k}}$

Then: $\mathbb{P}_n \left.\dfrac{d}{d\varepsilon}\right|_{\varepsilon=\hat{\varepsilon}_{n,k}} \mathscr{L}(\hat{P}^k_{n,\varepsilon})(O) = 0$

The targeting step (Step 2)

Construction of the targeting step for a given target parameter $\Psi : \mathcal{M} \to \mathbb{R}$ with efficient influence function $\phi^*(P)$ requires:

(i) A parametric submodel $\{P_\varepsilon : \varepsilon \in \mathbb{R}\} \subset \mathcal{M}$

(ii) A loss function $(O, P) \mapsto \mathscr{L}(P)(O)$

such that: (1) $P_{\varepsilon=0} = P$, and, (2) $\left.\frac{d}{d\varepsilon}\right|_{\varepsilon=0} \mathscr{L}(P_\varepsilon)(O) = \phi^*(P)(O)$

- kth updated estimator $\hat{P}_n^k$
- Minimizer $\hat{\varepsilon}_{n,k}$ of $\varepsilon \mapsto \mathbb{P}_n \mathscr{L}(\hat{P}_{n,\varepsilon}^k)$
- Update: $\hat{P}_n^{k+1} := \hat{P}^k_{\hat{\varepsilon}_{n,k}}$

Then: $\mathbb{P}_n \left.\frac{d}{d\varepsilon}\right|_{\varepsilon=\hat{\varepsilon}_{n,k}} \mathscr{L}(\hat{P}_{n,\varepsilon}^k)(O) = 0$, so when $\hat{\varepsilon}_{n,k} \approx 0$:

$$\mathbb{P}_n \left.\frac{d}{d\varepsilon}\right|_{\varepsilon=0} \mathscr{L}(\hat{P}_{n,\varepsilon}^k) = 0.$$

The targeting step (Step 2)

Construction of the targeting step for a given target parameter $\Psi : \mathcal{M} \to \mathbb{R}$ with efficient influence function $\phi^*(P)$ requires:

(i) A parametric submodel $\{P_\varepsilon : \varepsilon \in \mathbb{R}\} \subset \mathcal{M}$

(ii) A loss function $(O, P) \mapsto \mathscr{L}(P)(O)$

such that: (1) $P_{\varepsilon=0} = P$, and, (2) $\frac{d}{d\varepsilon}\Big|_{\varepsilon=0} \mathscr{L}(P_\varepsilon)(O) = \phi^*(P)(O)$

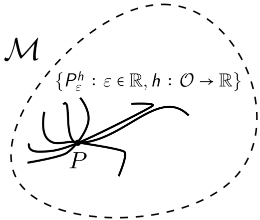
- kth updated estimator $\hat{P}_n^k$
- Minimizer $\hat{\varepsilon}_{n,k}$ of $\varepsilon \mapsto \mathbb{P}_n \mathscr{L}(\hat{P}_{n,\varepsilon}^k)$
- Update: $\hat{P}_n^{k+1} := \hat{P}_{\hat{\varepsilon}_{n,k}}^k$

Then: $\mathbb{P}_n \frac{d}{d\varepsilon}\Big|_{\varepsilon=\hat{\varepsilon}_{n,k}} \mathscr{L}(\hat{P}_{n,\varepsilon}^k)(O) = 0$, so when $\hat{\varepsilon}_{n,k} \approx 0$:

$$\mathbb{P}_n \frac{d}{d\varepsilon}\Big|_{\varepsilon=0} \mathscr{L}(\hat{P}_{n,\varepsilon}^k) = 0.$$

The targeting step (Step 2) *ASIDE*

What happens?



Parametric submodels $\{P_\varepsilon : \varepsilon \in \mathbb{R}\} \subset \mathcal{M}$ are also what we use to:

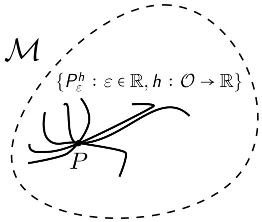
- define pathwise differentiability:[1]

$$\frac{d}{d\varepsilon}\bigg|_{\varepsilon=0} \Psi(P_\varepsilon) = \int \phi(P)(o)b(o)dP(o), \tag{1}$$

- derive a nonparametric lower bound on the variance.

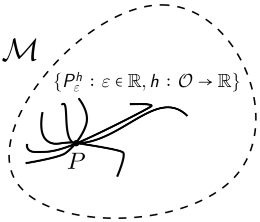
[1](1) should hold across any smooth submodel $\{P_\varepsilon : \varepsilon \in \mathbb{R}\} \subset \mathcal{M}$.

The targeting step (Step 2) *ASIDE*



- Index submodel by its score function: $\{P_\varepsilon^h : \varepsilon \in \mathbb{R}, h : \mathcal{O} \to \mathbb{R}\}$.
- Easier to estimate Ψ in the smaller model $\{P_\varepsilon^h : \varepsilon \in \mathbb{R}\}$ than in $\mathcal{M}$.
- The supremum over Cramér-Rao bounds over all submodels $\{P_\varepsilon^h : \varepsilon \in \mathbb{R}\}$ for estimating $\varepsilon \mapsto \Psi(P_\varepsilon^h)$ at $\varepsilon = 0$ provides lower bound on the variance for estimating Ψ in $\mathcal{M}$:

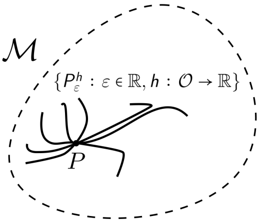
The targeting step (Step 2) *ASIDE*



- Index submodel by its score function: $\{P_\varepsilon^h : \varepsilon \in \mathbb{R}, h : \mathcal{O} \to \mathbb{R}\}$.
- Easier to estimate Ψ in the smaller model $\{P_\varepsilon^h : \varepsilon \in \mathbb{R}\}$ than in $\mathcal{M}$.
- The supremum over Cramér-Rao bounds over all submodels $\{P_\varepsilon^h : \varepsilon \in \mathbb{R}\}$ for estimating $\varepsilon \mapsto \Psi(P_\varepsilon^h)$ at $\varepsilon = 0$ provides lower bound on the variance for estimating Ψ in $\mathcal{M}$:

$$\frac{\left(\frac{d}{d\varepsilon}\big|_{\varepsilon=0}\Psi(P_\varepsilon^h)\right)^2}{Ph^2} \tag{$*$}$$

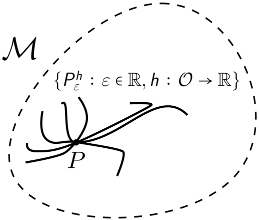
The targeting step (Step 2) *ASIDE*



- Index submodel by its score function: $\{P_\varepsilon^h : \varepsilon \in \mathbb{R}, h : \mathcal{O} \to \mathbb{R}\}$.
- Easier to estimate Ψ in the smaller model $\{P_\varepsilon^h : \varepsilon \in \mathbb{R}\}$ than in $\mathcal{M}$.
- The supremum over Cramér-Rao bounds over all submodels $\{P_\varepsilon^h : \varepsilon \in \mathbb{R}\}$ for estimating $\varepsilon \mapsto \Psi(P_\varepsilon^h)$ at $\varepsilon = 0$ provides lower bound on the variance for estimating Ψ in $\mathcal{M}$:

$$\frac{\left(\frac{d}{d\varepsilon}\big|_{\varepsilon=0}\Psi(P_\varepsilon^h)\right)^2}{Ph^2} = \frac{(P\phi^*(P)h)^2}{Ph^2} \qquad (*)$$

The targeting step (Step 2) *ASIDE*



- Index submodel by its score function: $\{P_\varepsilon^h : \varepsilon \in \mathbb{R}, h : \mathcal{O} \to \mathbb{R}\}$.
- Easier to estimate Ψ in the smaller model $\{P_\varepsilon^h : \varepsilon \in \mathbb{R}\}$ than in $\mathcal{M}$.
- The supremum over Cramér-Rao bounds over all submodels $\{P_\varepsilon^h : \varepsilon \in \mathbb{R}\}$ for estimating $\varepsilon \mapsto \Psi(P_\varepsilon^h)$ at $\varepsilon = 0$ provides lower bound on the variance for estimating Ψ in $\mathcal{M}$:

$$\frac{\left(\frac{d}{d\varepsilon}\big|_{\varepsilon=0}\Psi(P_\varepsilon^h)\right)^2}{Ph^2} = \frac{(P\phi^*(P)h)^2}{Ph^2} \leq P\{\phi^*(P)\}^2 \qquad (*)$$

The targeting step (Step 2) *ASIDE*

The submodel which attains the supremum of the Cramér-Rao bounds over all parametric submodels is called the least favorable submodel;

- It is the submodel for which the score is equal to the efficient influence function $\phi^*(P)$.
- The minimum loss-based estimator at $\varepsilon = 0$ along this submodel is asymptotically equivalent to the efficient estimator in the large nonparametric model.

The targeting step (Step 2) *ASIDE*

The submodel which attains the supremum of the Cramér-Rao bounds over all parametric submodels is called the least favorable submodel;

- It is the submodel for which the score is equal to the efficient influence function $\phi^*(P)$.
- The minimum loss-based estimator at $\varepsilon = 0$ along this submodel is asymptotically equivalent to the efficient estimator in the large nonparametric model.

The TMLE step uses the least favorable submodel as a fluctuation model

- given a current estimator $\hat{P}_n^k$ the updated estimator is found by fluctuating along the least favorable submodel;
- the nonparametric efficiency bound is reached when no further fluctuation is needed ($\varepsilon \approx 0$);

The targeting step (Step 2) *ASIDE*

The submodel which attains the supremum of the Cramér-Rao bounds over all parametric submodels is called the least favorable submodel;

- It is the submodel for which the score is equal to the efficient influence function $\phi^*(P)$.
- The minimum loss-based estimator at $\varepsilon = 0$ along this submodel is asymptotically equivalent to the efficient estimator in the large nonparametric model.

The TMLE step uses the least favorable submodel as a fluctuation model

- given a current estimator $\hat{P}_n^k$ the updated estimator is found by fluctuating along the least favorable submodel;
- the nonparametric efficiency bound is reached when no further fluctuation is needed ($\varepsilon \approx 0$);
- the estimator solves the efficient influence curve equation.

The targeting step (Step 2)

Conditions (asymptotic linearity and efficiency)

(C1) Solve the efficient influence curve equation: $\mathbb{P}_n \phi^*(\hat{P}_n) = o_P(n^{-1/2})$

(C2) Remainder $R(\hat{P}_n, P_0) = o_P(n^{-1/2})$

(C3) Donsker class conditions for $\{\phi^*(P) : P \in \mathcal{M}\}$

Then: $\Psi(\hat{P}_n) \overset{as}{\sim} N(\Psi(P_0), P_0\phi^*(P_0)^2/n)$

- The targeting step ensures that (C1) holds.
- Assume that (C2) and (C3) hold.

We can use the efficient influence function to compute an estimator for the standard error of the TMLE estimator:

$$\hat{\sigma}_n = \sqrt{\frac{\mathbb{P}_n\{\phi^*(\hat{P}_n)\}^2}{n}}$$

Targeting the average treatment effect (ATE)

Targeting the average treatment effect (ATE)

EXAMPLE: Average treatment effect (ATE)

Observed data $O = (X, A, Y) \in \mathbb{R}^d \times \{0,1\} \times \{0,1\} = \mathcal{O}$

* $X \in \mathbb{R}^d$ are covariates
* $A \in \{0,1\}$ is a binary exposure variable (treatment decision)
* $Y \in \{0,1\}$ is a binary outcome variable

$O \sim P_0$ where P_0 assumed to belong to nonparametric model $\mathcal{M}$.

We are interested in estimating the ATE:

$$\Psi(P) = \mathbb{E}_P\Big[\mathbb{E}_P[Y \mid A = 1, X] - \mathbb{E}_P[Y \mid A = 0, X]\Big].$$

Targeting the average treatment effect (ATE)

EXAMPLE: Average treatment effect (ATE)

For the ATE, as we have seen, we can also write the target parameter $\Psi : \mathcal{M} \to \mathbb{R}$ as

$$\Psi(P) = \tilde{\Psi}(f, \mu_X) = \int_{\mathbb{R}^d} \big(f(1,x) - f(0,x)\big) d\mu_X(x) \qquad (*)$$

where

$$f(a,x) = \mathbb{E}[\, Y \mid A = a, X = x]$$

and μ_X is the marginal distribution of X.

I.e., $\hat{\psi}_n = \tilde{\Psi}(\hat{f}_n, \hat{\mu}_n)$.

Targeting the average treatment effect (ATE)

EXAMPLE: Average treatment effect (ATE)

Step 1 Construct initial estimators $\hat{f}_n, \hat{\pi}_n$ for f, π.

Step 2 Update the estimator $\hat{f}_n \mapsto \hat{f}_n^*$ for f such that $\hat{f}_n^*$ for the fixed $\hat{\pi}_n$ solves the efficient influence curve equation.

For the ATE, Step 2 is simply just an additional logistic regression step.

Targeting the average treatment effect (ATE)

EXAMPLE: Average treatment effect (ATE)

We need:

0. The efficient influence function:

$$\tilde{\phi}^*(f,\pi)(O) = \left(\frac{A}{\pi(A \mid X)} - \frac{1-A}{\pi(A \mid X)}\right)\big(Y - f(A,X)\big)$$

$$+ \; f(1,X) - f(0,X) - \tilde{\Psi}(f)$$

Targeting the average treatment effect (ATE)

EXAMPLE: Average treatment effect (ATE)

We need:

0. The efficient influence function:

$$\tilde{\phi}^*(f,\pi)(O) = \underbrace{\left(\frac{A}{\pi(A \mid X)} - \frac{1-A}{\pi(A \mid X)}\right)\big(Y - f(A,X)\big)}_{=\tilde{\phi}^*_f(f,\pi)(O)} + f(1,X) - f(0,X) - \tilde{\Psi}(f)$$

Targeting the average treatment effect (ATE)

EXAMPLE: Average treatment effect (ATE)

We need:

0. The efficient influence function:

$$\tilde{\phi}^*(f,\pi)(O) = \underbrace{\left(\frac{A}{\pi(A \mid X)} - \frac{1-A}{\pi(A \mid X)}\right)\big(Y - f(A,X)\big)}_{=\tilde{\phi}^*_f(f,\pi)(O)}$$
$$+ \; f(1,X) - f(0,X) - \tilde{\Psi}(f)$$

Further, we need:

(i) A parametric submodel $\{f_\varepsilon : \varepsilon \in \mathbb{R}\} \subset \mathcal{M}$

(ii) A loss function $(O,f) \mapsto \mathscr{L}(f)(O)$

such that

$$(1) \quad f_{\varepsilon=0} = f \qquad (2) \quad \frac{d}{d\varepsilon}\bigg|_{\varepsilon=0} \mathscr{L}(f_\varepsilon)(O) = \tilde{\phi}^*_f(f,\pi)(O)$$

Targeting the average treatment effect (ATE)

(i) Log-likelihood loss function:

$$\boxed{\operatorname{logit}(p) = \operatorname{expit}^{-1}(p) = \log\left(\frac{p}{1-p}\right)}$$

$$\mathscr{L}(f)(O) = -\big(Y\log(f(A,X)) + (1-Y)\log(1-f(A,X))\big)$$

(ii) Logistic regression model:

$$f_\varepsilon(A,X) = \operatorname{expit}\big(\operatorname{logit}(f(A,X)) + \varepsilon H(A,X)\big)$$

with the so-called "clever covariate": $H(A,X) := \dfrac{2A-1}{\pi(A \mid X)}$.

Targeting the average treatment effect (ATE)

(i) Log-likelihood loss function:

$$\mathrm{logit}(p) = \mathrm{expit}^{-1}(p) = \log\left(\frac{p}{1-p}\right)$$

$$\mathscr{L}(f)(O) = -\Big(Y\log(f(A,X)) + (1-Y)\log(1-f(A,X))\Big)$$

(ii) Logistic regression model:

$$f_\varepsilon(A,X) = \mathrm{expit}\Big(\mathrm{logit}(f(A,X)) + \varepsilon H(A,X)\Big)$$

with the so-called "clever covariate": $H(A,X) := \dfrac{2A-1}{\pi(A \mid X)}$.

To show this, we verify that (i)–(ii) fulfill

$$(1) \quad f_{\varepsilon=0} = f \qquad (2) \quad \frac{d}{d\varepsilon}\bigg|_{\varepsilon=0} \mathscr{L}(f_\varepsilon)(O) = \tilde{\phi}^*_f(f,\pi)(O)$$

Targeting the average treatment effect (ATE)

(i) Log-likelihood loss function:

$$\mathrm{logit}(p) = \mathrm{expit}^{-1}(p) = \log\left(\frac{p}{1-p}\right)$$

$$\mathscr{L}(f)(O) = -\big(Y\log(f(A,X)) + (1-Y)\log(1-f(A,X))\big)$$

(ii) Logistic regression model:

$$f_\varepsilon(A,X) = \mathrm{expit}\big(\mathrm{logit}(f(A,X)) + \varepsilon H(A,X)\big)$$

with the so-called "clever covariate": $H(A,X) := \dfrac{2A-1}{\pi(A\,|\,X)}$.

SMALL EXERCISE: To show this, we verify that (i)–(ii) fulfill

$$(1)\quad f_{\varepsilon=0} = f \qquad (2)\quad \frac{d}{d\varepsilon}\bigg|_{\varepsilon=0} \mathscr{L}(f_\varepsilon)(O) = \tilde{\phi}^*_f(f,\pi)(O)$$

Targeting the average treatment effect (ATE)

- Initial estimators $\hat{f}_n$, $\hat{\pi}_n$.
- Estimate clever covariate by:

$$\hat{H}_n(A, X) = \frac{2A - 1}{\hat{\pi}_n(A \mid X)}.$$

- The minimizer $\hat{\varepsilon}_n$ of $\varepsilon \mapsto \mathbb{P}_n \mathscr{L}(\hat{f}_{n,\varepsilon})$ equals the maximum likelihood estimator for ε in the fixed-intercept logistic regression:

$$\operatorname{logit} \mathbb{E}[Y \mid A, X] = \operatorname{logit}(\hat{f}_n(A, X)) + \varepsilon \hat{H}_n(A, X)$$

- Update: $\hat{f}^*_n := \hat{f}_{n,\hat{\varepsilon}_n}$.

Then: $\quad \mathbb{P}_n \frac{d}{d\varepsilon}\Big|_{\varepsilon=\hat{\varepsilon}_n} \mathscr{L}(\hat{f}_{n,\varepsilon}) = 0, \quad$ i.e.,

$$\mathbb{P}_n \tilde{\phi}^*_f(\hat{f}_{n,\hat{\varepsilon}_n}, \hat{\pi}_n) = \mathbb{P}_n \tilde{\phi}^*_f(\hat{f}^*_n, \hat{\pi}_n) = 0.$$

Targeting the average treatment effect (ATE)

$$\tilde{\phi}^*(f,\pi)(O) = \underbrace{\left(\frac{A}{\pi(A \mid X)} - \frac{1-A}{\pi(A \mid X)}\right)\big(Y - f(A,X)\big)}_{=\tilde{\phi}^*_f(f,\pi)(O)}$$
$$+ \underbrace{f(1,X) - f(0,X) - \tilde{\Psi}(f)}_{=\tilde{\phi}^*_{\mu_X}(f)(O)}$$

Per construction we already have: $\mathbb{P}_n\phi^*_\mu(\hat{f}^*_n) = 0,$

since: $\tilde{\Psi}(\hat{f}^*_n) = \frac{1}{n}\sum_{i=1}^n \big(\hat{f}^*_n(1,X_i) - \hat{f}^*_n(0,X_i)\big) = \mathbb{P}_n\big(\hat{f}^*_n(1,\cdot) - \hat{f}^*_n(0,\cdot)\big).$

The targeting step thus yields:

$$\mathbb{P}_n\tilde{\phi}^*(\hat{f}^*_n,\hat{\pi}_n) = \mathbb{P}_n\tilde{\phi}^*_f(\hat{f}^*_n,\hat{\pi}_n) + \mathbb{P}_n\phi^*_\mu(\hat{f}^*_n) = 0.$$

Targeting the average treatment effect (ATE)

Doing the targeting in practice using the simulated dataset:

```
set.seed(5)
n <- 500
X <- runif(n, -2, 2)
A <- rbinom(n, 1, prob=plogis(-0.25 + 1.2*X))
Y <- rbinom(n, 1, prob=plogis(-0.9 + 1.9*X^2 + 0.5*A))
(sim.data <- data.table(id=1:n,X=X,A=A,Y=Y))
```

```
      id          X A Y
  1:   1 -1.1991422 0 1
  2:   2  0.7408744 1 1
  3:   3  1.6675031 1 1
  4:   4 -0.8624022 0 1
  5:   5 -1.5813995 0 1
 ---
496: 496 -0.3978523 1 0
497: 497 -1.5069379 0 1
498: 498  1.8340120 1 1
499: 499  0.6349484 1 0
500: 500 -0.5214807 0 1
```

Targeting the average treatment effect (ATE)

Initial estimation:

```
#-- treatment distribution;
glm.A <- glm(A~X, data=sim.data, family=binomial)
pi.1 <- predict(glm.A, type="response")

#-- outcome distribution (misspecified);
glm.Y <- glm(Y~A+X, data=sim.data, family=binomial)
sim.data[, f:=predict(glm.Y, type="response")]
sim.data[, f.A1:=predict(glm.Y, type="response",
             newdata=copy(sim.data)[, A:=1])]
sim.data[, f.A0:=predict(glm.Y, type="response",
             newdata=copy(sim.data)[, A:=0])]

#-- initial estimate of the ATE;
fit.ate.initial <- sim.data[, mean(f.A1 - f.A0)]
```

Targeting the average treatment effect (ATE)

Targeting step:

```
#-- tmle;
sim.data[, clever.covariate:=((A==1)/pi.1 - (A==0)/(1-pi.1))]
eps <- coef(glm(Y ~ offset(qlogis(f))+clever.covariate-1,
         data=sim.data, family=binomial()))
```

eps = -0.0157708436790858

Targeting the average treatment effect (ATE)

Targeting step:

```
#-- tmle;
sim.data[, clever.covariate:=((A==1)/pi.1 - (A==0)/(1-pi.1))]
eps <- coef(glm(Y ~ offset(qlogis(f))+clever.covariate-1,
         data=sim.data, family=binomial()))
```

eps = -0.0157708436790858

```
#-- tmle update;
sim.data[, f.A1.tmle:=plogis(qlogis(f.A1) + eps/pi.1)]
sim.data[, f.A0.tmle:=plogis(qlogis(f.A0) - eps/(1-pi.1))]
```

i.e., `f.A1.tmle` is the estimate of $f(1,X) = \mathbb{E}[Y \mid A = 1, X]$, obtained via the submodel:

$$\hat{f}_n^*(1,X) = \hat{f}_{n,\hat{\varepsilon}_n}(1,X) = \operatorname{expit}\Big(\operatorname{logit}(\hat{f}_n(1,X)) + \hat{\varepsilon}_n \hat{H}_n(1,X)\Big),$$

and likewise with `f.A0.tmle`.

Targeting the average treatment effect (ATE)

```
      id          X A Y      f.A1      f.A0 f.A1.tmle f.A0.tmle
  1:   1 -1.1991422 0 1 0.7655621 0.6713853 0.7488795 0.6755825
  2:   2  0.7408744 1 1 0.7396070 0.6399080 0.7349584 0.6504368
  3:   3  1.6675031 1 1 0.7265721 0.6244167 0.7228545 0.6481588
  4:   4 -0.8624022 0 1 0.7611886 0.6660214 0.7488197 0.6705960
  5:   5 -1.5813995 0 1 0.7704590 0.6774205 0.7463439 0.6813231
 ---
496: 496 -0.3978523 1 0 0.7550638 0.6585507 0.7464799 0.6639337
497: 497 -1.5069379 0 1 0.7695108 0.6762494 0.7471142 0.6802008
498: 498  1.8340120 1 1 0.7241872 0.6216047 0.7205492 0.6495635
499: 499  0.6349484 1 0 0.7410712 0.6416611 0.7362345 0.6513868
500: 500 -0.5214807 0 1 0.7567041 0.6605467 0.7472996 0.6656728
```

Targeting the average treatment effect (ATE)

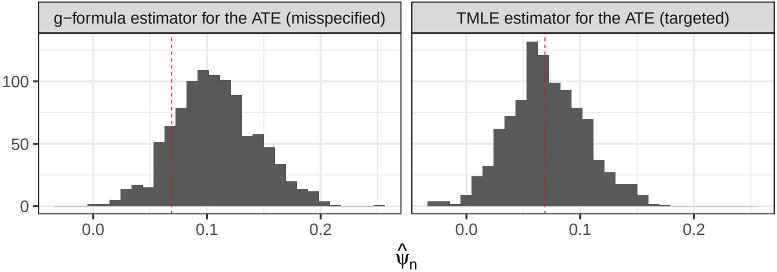
```
      id          X A Y      f.A1      f.A0 f.A1.tmle f.A0.tmle
  1:   1 -1.1991422 0 1 0.7655621 0.6713853 0.7488795 0.6755825
  2:   2  0.7408744 1 1 0.7396070 0.6399080 0.7349584 0.6504368
  3:   3  1.6675031 1 1 0.7265721 0.6244167 0.7228545 0.6481588
  4:   4 -0.8624022 0 1 0.7611886 0.6660214 0.7488197 0.6705960
  5:   5 -1.5813995 0 1 0.7704590 0.6774205 0.7463439 0.6813231
 ---
496: 496 -0.3978523 1 0 0.7550638 0.6585507 0.7464799 0.6639337
497: 497 -1.5069379 0 1 0.7695108 0.6762494 0.7471142 0.6802008
498: 498  1.8340120 1 1 0.7241872 0.6216047 0.7205492 0.6495635
499: 499  0.6349484 1 0 0.7410712 0.6416611 0.7362345 0.6513868
500: 500 -0.5214807 0 1 0.7567041 0.6605467 0.7472996 0.6656728
```

```r
fit.ate.tmle <- sim.data[, mean(f.A1.tmle - f.A0.tmle)]
```

```
initial ate est = 0.0975
tmle ate est    = 0.0768
```

Targeting the average treatment effect (ATE)

With 500 repeated simulations:



Practical 1: Implementing the targeting step

Practical Part 1 Implementing the targeting step.

Practical Part 2 Computing the variances of the ATE, the $\log \mathrm{RR}$ and the $\log \mathrm{OR}$.

Practical Part 3 Large-sample properties (simulation study).

The exercise is described in detail in: **day2-practical1.pdf**.

[More comments on the following slides].

Comments for practical

We focused on the ATE as an example of a causal parameter.

But note that other simple causal parameters can be constructed from $\mathbb{E}_P[Y^1]$ and $\mathbb{E}_P[Y^0]$.

Like:

$$\Psi_{\mathrm{RR}}(P) = \frac{\mathbb{E}_P[Y^1]}{\mathbb{E}_P[Y^0]},$$

or,

$$\Psi_{\mathrm{OR}}(P) = \frac{\mathbb{E}_P[Y^1]/(1-\mathbb{E}_P[Y^1])}{\mathbb{E}_P[Y^0]/(1-\mathbb{E}_P[Y^0])},$$

Comments for practical

For the targeting step, we can choose to target $\Psi_1(P) = \mathbb{E}_P[Y^1]$ and $\Psi_0(P) = \mathbb{E}_P[Y^0]$ *separately*.

Comments for practical

For the targeting step, we can choose to target $\Psi_1(P) = \mathbb{E}_P[Y^1]$ and $\Psi_0(P) = \mathbb{E}_P[Y^0]$ *separately*.

The efficient influence function for the treatment-specific mean $\Psi_a(P) = \mathbb{E}_P[Y^a]$:

$$\tilde{\phi}_a^*(f, \pi)(O) = \underbrace{\frac{1\{A = a\}}{\pi(a \mid X)}}_{\text{clever covar.}} \big(Y - f(A, X)\big) + f(a, X) - \Psi_a(P)$$

Comments for practical

For the targeting step, we can choose to target $\Psi_1(P) = \mathbb{E}_P[Y^1]$ and $\Psi_0(P) = \mathbb{E}_P[Y^0]$ *separately*.

The efficient influence function for the treatment-specific mean $\Psi_a(P) = \mathbb{E}_P[Y^a]$:

$$\tilde{\phi}^*_a(f, \pi)(O) = \underbrace{\frac{1\{A = a\}}{\pi(a \mid X)}}_{\text{clever covar.}} \big(Y - f(A, X)\big) + f(a, X) - \Psi_a(P)$$

If we target $\Psi_1(P)$ and $\Psi_0(P)$ separately, we obtain two sets of updated estimators $\hat{f}_n \mapsto \hat{f}^*_{n,1}$ and $\hat{f}_n \mapsto \hat{f}^*_{n,0}$

- one to construct a targeted estimator $\hat{\psi}^*_{1,n}$ for $\Psi_1(P)$;
- and the other to construct a targeted estimator $\hat{\psi}^*_{0,n}$ for $\Psi_0(P)$.

Comments for practical

We can then compute an estimate for the ATE as

$$\hat{\psi}_n^* = \hat{\psi}_{n,1}^* - \hat{\psi}_{n,0}^*,$$

and we can estimate the variance of this estimator by

$$\mathbb{P}_n\{\tilde{\phi}_1^*(\hat{f}_{n,1}^*, \hat{\pi}_n) - \tilde{\phi}_0^*(\hat{f}_{n,0}^*, \hat{\pi}_n)\}^2;$$

since efficient influence function for the ATE is

$$\tilde{\phi}^*(f, \pi) = \tilde{\phi}_1^*(f, \pi) - \tilde{\phi}_0^*(f, \pi).$$

Comments for practical

Similarly we can construct estimators for the RR and the OR by simple plug-in:

$$\hat{\psi}^*_{\mathrm{RR},n} = \frac{\hat{\psi}^*_{1,n}}{\hat{\psi}^*_{0,n}},$$

and,

$$\hat{\psi}^*_{\mathrm{OR},n} = \frac{\hat{\psi}^*_{1,n}/(1-\hat{\psi}^*_{1,n})}{\hat{\psi}^*_{0,n}/(1-\hat{\psi}^*_{0,n})}.$$

Comments for practical

Similarly we can construct estimators for the RR and the OR by simple plug-in:

$$\hat{\psi}^*_{\mathrm{RR},n} = \frac{\hat{\psi}^*_{1,n}}{\hat{\psi}^*_{0,n}},$$

and,

$$\hat{\psi}^*_{\mathrm{OR},n} = \frac{\hat{\psi}^*_{1,n}/(1-\hat{\psi}^*_{1,n})}{\hat{\psi}^*_{0,n}/(1-\hat{\psi}^*_{0,n})}.$$

We can use the delta method to derive the efficient influence functions of $\Psi_{\mathrm{RR}}(P)$ and $\Psi_{\mathrm{OR}}(P)$.

Comments for practical

Let $\phi^*(P)$ be the efficient influence function for a parameter $\Psi(P)$. Say that interest is in $h(\Psi(P))$ for a function h.

The delta method yields that:

If the first derivative $h'(\psi) = \frac{d}{d\psi}h(\psi)$ of h exists and is non-zero, then the efficient influence function of $h(\Psi(P))$ is:

$$\phi_h^*(P) = h'(\Psi(P))\phi^*(P).$$

Comments for practical

So, once we have TMLE (targeted) estimators for $\Psi_1(P) = \mathbb{E}[Y^1]$ and $\Psi_0(P) = \mathbb{E}[Y^0]$:

- We can construct estimators for the ATE, the RR and the OR.
- We can compute the variance of the ATE estimator, the $\log \mathrm{RR}$ estimator and the $\log \mathrm{OR}$ estimator.

Practical 1: Implementing the targeting step

Practical Part 1 Implementing the targeting step.

Practical Part 2 Computing the variances of the ATE, the $\log \mathrm{RR}$ and the $\log \mathrm{OR}$.

Practical Part 3 Large-sample properties (simulation study).

The exercise is described in detail in: **day2-practical1.pdf**.