Contents:

§1 Policy Learning

1. Policy 的定义

2. 无限制时的 optimal policy

§2 Policy Evaluation

1. Policy Learning 的流程

2. Policy evaluation

§3 Empirical-Welfare Maximization

1. regret 的定义

2. Empirical-welfare maximization

3. Empirical-welfare maximization 的 bound

14 Policy learning

So far, we have focused on methods for estimating treatments effects. In many application areas, however, the fundamental goal of performing a causal analysis is not to estimate treatment effects, but rather to guide decision making: We want to understand treatment effects so that we can effectively prescribe treatment and allocate limited resources. In other words, for a set of new units $X_{n+1}, \cdots$, we want to assign it to either placebo or treatment $\pi(X_i) \in \{0, 1\}$ such that the average outcome is optimal.

The problem of learning optimal treatment assignment policies is closely related tobut subtly different fromthe problem of estimating treatment heterogeneity. On one hand, policy learning appears easier: All we care about is assigning people to treatment or to control, and we dont care about accurately estimating treatment effects beyond that. On the other hand, when learning policies, we need to account for considerations that were not present when simply estimating treatment effects: Any policy we actually want to use must be simple enough we can actually deploy it, cannot discriminate on protected

Policy 的定义：一个 function $\pi: R^p \mapsto \{0,1\}$ s.t. we assign treatment when $\pi(X_i)=1$

$\Rightarrow Y(\pi(X)) = \pi(X)\cdot Y(1) + [1-\pi(X)]\cdot Y(0)$

e.g. ① $\pi(X) = 1\{\sum_{i=1}^p a_i X_i > 1\}$ ② binary decision tree

characteristics, should not rely on game-able features, etc. In other words, when we do policy learning, we always have sorts of constraints.

---

1. Policy

Definition 14.1. *Suppose the features $X \in \mathbf{R}^d$, a policy $\pi(\cdot)$ is a function $\mathbf{R}^d \to \{0,1\}$ such that an individual with feature $X = x$ is treated if and only if $\pi(x) = 1$. We define the **Value** of the policy π as*

$$V(\pi) = \mathbf{E}Y_i(\pi(X_i)). \quad \text{(对 } X_i \text{ 取期望)} \tag{4}$$

Notice that

$$Y_i(\pi(X_i)) = \begin{cases} Y_i(1) \text{ if } \pi(X_i) = 1, \\ Y_i(0) \text{ if } \pi(X_i) = 0. \end{cases}$$

Therefore,

$$Y_i(\pi(X_i)) = Y_i(1)\pi(X_i) + Y_i(0)(1-\pi(X_i)).$$

In such case, for any $\pi(\cdot)$,

$$V(\pi) = \mathbf{E}Y_i(0) + \mathbf{E}\left(Y_i(1) - Y_i(0)\right) \times \pi(X_i).$$

$= E[\pi(X_i)\cdot \tau(X_i)]$, $\tau(X_i)$: CATE $= E[Y_i(1) - Y_i(0) \mid X_i]$

The problem we are interested in is to find

$$\widehat{\pi} = \arg_{\pi\in\Pi} V(\pi). \tag{5}$$

---

2. 无限制时的 optimal policy

If there is no constraints, finding the optimal policy is simple, which is demonstrated as follows:

Theorem 14.1. *Suppose we do not have any constraints, then the optimal policy is given by*

$$\widehat{\pi}(x) = \mathbf{1}_{\tau(x)\geq 0}, \text{ where } \tau(x) = \mathbf{E}(Y_i(1) - Y_i(0)) | X_i = x. \tag{6}$$

CATE > 0 时施加 treatment

Proof Notice that

$$V(\pi) = \mathbf{E}Y_i(0) + \mathbf{E}[\pi(X_i) \times \mathbf{E}\left(Y_i(1) - Y_i(0)\right) | X_i]$$

$$= \mathbf{E}Y_i(0) + \mathbf{E}[\pi(X_i) \times \tau(X_i)] \leq \mathbf{E}Y_i(0) + \mathbf{E}\tau(X_i) \times \mathbf{1}_{\tau(X_i)\geq 0},$$

$= E[\pi(X_i)\cdot\tau(X_i)\cdot 1_{\tau(X_i)\geq 0}] + E[\pi(X_i)\cdot\tau(X_i)\cdot 1_{\tau(X_i)<0}]$, $\pi(X_i)<1$, $\tau(X_i)<0$

which reaches maximum when $\pi(x) = \mathbf{1}_{\tau(x)\geq 0}$. □

This approach may be reasonable in some applications, but may result in policies that are hard to interpret or may not respect other practical constraints that are called for in the application. The focus of this chapter will be on developing methods for learning policies that do respect such constraints.

注：现实中（为了 policy 的易懂性等因素）会存在 constraints，如：

$\Pi = \{1\{\sum_{i=1}^p a_i X_i \geq 0\} \mid a_i \in R, \forall i = 1, \cdots, p\}$

$\Pi = \{2\text{-layer decision tree}\}$

Example 7 In Kitagawa and Tetenov (2018), the authors considered linear treatment rules of the form

$$\pi(x) = \mathbf{1}_{prior\ earining \times \alpha_1 + education \times \alpha_2 > c}.$$

15 Policy evaluation

1. PL的流程

The workflow of policy learning has 3 steps:

1. Collect data with random or quasi-random treatment assignments W_i to learn a (optimal) policy $\widehat{\pi}$.

2. In a second (optional) phase, we may want to evaluate the quality of the learned policy, i.e., estimate $V(\widehat{\pi})$. In this stage, we always resort to a out-of-the-sample test set which are independent of training set.

3. Finally, once were done learning, we enter the last phase where we may choose to deploy the learned policy, i.e., to set $W_i = \widehat{\pi}(X_i)$ for future observations.

If we know the value $V(\cdot)$, evaluating the value of a policy is simple. Unfortunately, in most of the time, we do not know the value function. In such case, with a (suppose estimated) given policy $\widehat{\pi}$, and a new out-of-the-sample test set, we want to estimate $V(\widehat{\pi})$. Notably, because the test set and training sets are independent of each other, this task is equivalent to using the test set to estimate $V(\pi)$ for an arbitrary fixed policy π.

2. Policy evaluation

Theorem 15.1. *Suppose we have n observations $(Y_i, W_i, X_i) \in \mathbf{R} \times \{0,1\} \times \mathbf{R}^d$, where the unconfoundness condition holds true. Suppose the propensity score $e(x) = \mathbf{E}W_i|X_i = x$ is known, then the Inverse-propensity weighting estimator*

$$\widehat{V}_{IPW}(\pi) = \frac{1}{n}\sum_{i=1}^{n} \frac{\mathbf{1}_{W_i=\pi(X_i)} Y_i}{Prob(W_i = \pi(X_i)|X_i)}, \tag{7}$$

is an unbiased estimator for $V(\pi)$.

$$= \frac{1}{n}\left(\sum_{\pi(X_i)=1} \frac{1_{W_i=1} Y_i}{e(X_i)} + \sum_{\pi(X_i)=0} \frac{1_{W_i=0} Y_i}{1-e(X_i)}\right)$$

Notice that

$$Prob(W_i = \pi(X_i)|X_i) = \begin{cases} Prob(W_i = 1|X_i) = e(X_i) \text{ if } \pi(X_i) = 1, \\ Prob(W_i = 0|X_i) = 1 - e(X_i) \text{ if } \pi(X_i) = 0. \end{cases}$$

Besides,

$$\begin{aligned} \mathbf{1}_{W_i=\pi(X_i)} Y_i &= \mathbf{1}_{W_i=\pi(X_i)} \times (Y_i(1) \times \mathbf{1}_{W_i=1} + Y_i(0) \times \mathbf{1}_{W_i=0}) \\ &= Y_i(1) \times \mathbf{1}_{W_i=\pi(X_i)=1} + Y_i(0) \times \mathbf{1}_{W_i=\pi(X_i)=0}. \end{aligned}$$

Therefore,

$$\mathbf{E}\frac{\mathbf{1}_{W_i=\pi(X_i)}Y_i}{Prob(W_i=\pi(X_i)|X_i)}|X_i = \frac{1}{Prob(W_i=\pi(X_i)|X_i)} \times \left(\mathbf{E}Y_i(1) \times \mathbf{1}_{W_i=\pi(X_i)=1}|X_i + \mathbf{E}Y_i(0) \times \mathbf{1}_{W_i=\pi(X_i)=0}|X_i\right)$$

$$= \frac{1}{Prob(W_i=\pi(X_i)|X_i)} \times (\pi(X_i)\mathbf{E}Y_i(1)|X_i \times e(X_i) + (1-\pi(X_i))\mathbf{E}Y_i(0)|X_i \times (1-e(X_i)))$$

$$= \begin{cases} \pi(X_i) \times \mathbf{E}Y_i(1)|X_i \text{ if } \pi(X_i)=1, \\ (1-\pi(X_i))\mathbf{E}Y_i(0)|X_i \text{ if } \pi(X_i)=0, \end{cases} = \pi(X_i) \times \mathbf{E}Y_i(1)|X_i + (1-\pi(X_i))\mathbf{E}Y_i(0)|X_i,$$

which proves the result.

If we want to incorporate a parametric model, we can consider a double-robust type estimator

$$\widehat{V}_{AIPW}(\pi) = \frac{1}{n}\sum_{i=1}^{n}\widehat{\mu}(X_i,\pi(X_i)) + \frac{\mathbf{1}_{W_i=\pi(X_i)}}{Prob(W_i=\pi(X_i)|X_i)}(Y_i - \widehat{\mu}(X_i,\pi(X_i))).$$

16 Empirical-welfare maximization

1. regret

We now return to the task of learning a policy, i.e., using experimental or quasi- experimental data to choose a good treatment assignment rule $\widehat{\pi}$. Throughout, we assume that the policymaker is constrained to choose a policy π belonging to some class Π of acceptable policies. For example, one may consider $\pi(x) = \mathbf{1}_{a^\top x \geq c}$ for some vector a and threshold c, or some fixed-depth decision trees.

In such setting, we define the optimal policy as

$$\pi^* = \arg\max V(\pi) \text{ such that } \pi \in \Pi.$$

We define the regret

$$R(\pi) = \sup_{\zeta\in\Pi} V(\zeta) - V(\pi).$$

Our goal is to learn a policy with guaranteed worst-case bounds on the regret. We refer this task as a learning (rather than estimation) task because the performance of π is only assessed in terms of its regret. No requirements will be made on π converging to the optimal policy π^* in terms of its functional form (and in fact no assumption is made that there is a unique optimal policy).

2. Empirical-welfare maximization

If the optimal policy π^* maximizes the true value function, then it is natural to attempt learn $\widehat{\pi}$ by maximizing an estimated value function

$$\widehat{\pi} = \arg\max\{\widehat{V}(\pi) : \pi \in \Pi\}.$$

This approach is called empirical-welfare maximization by Kitagawa and Tetenov(2018). A plausible

estimator is the IPW estimator $\widehat{V}_{IPW}(\pi)$, in such case we define

$$\widehat{\pi}_{IPW} = \arg\max\{\widehat{V}_{IPW}(\pi) : \pi \in \Pi\}.$$

3. Empirical-welfare maximization 的 bound

To bound the regret,

$$\begin{aligned} R(\widehat{\pi}) = V(\pi^*) - V(\widehat{\pi}) &= V(\pi^*) - \widehat{V}(\pi^*) + \widehat{V}(\pi^*) - V(\widehat{\pi}) \\ &\leq |V(\pi^*) - \widehat{V}(\pi^*)| + |V(\widehat{\pi}) - \widehat{V}(\pi^*)| \leq 2 \sup_{\pi \in \Pi} |V(\pi) - \widehat{V}(\pi)|. \end{aligned}$$

In other words, bounding the estimator of the value function is core for the learning of a policy.

References

Ding, Peng (2023). *A First Course in Causal Inference.*