Applying DNN to Causal Inference

Key messages:

- All the contents contained in this technical blog are based on public paper and websites.
- This blog is written for self-learning purpose and document archiving.

Answers to common questions:

• We have done experiments and get the data. Why do we need to make causal inference based on the data? Moreover, why do we need to train a model to estimate the outcome?
Answer: There are two common reasons to make causal inference based on the data acquired from a experiment; that is, we regard the experiment as an observational study. Examples can be found in Section 1. The reasons are

1. There exists self selection bias in the treatment group.
2. After the experiment, we found the treatment and control groups have quite different user distributions.

The reason why we need to train a model to estimate the outcome is to make inference about the treatment effect. The naive idea is to train the following model using the linear regression or ML algorithms like tree-based regressors:

$$Y = \theta T + g(\boldsymbol{X}) + U$$

where Y is the outcome, T is the treatment (for binary treatment, $T = 0$ or 1), $g(\boldsymbol{X})$ is a function of all covariates $\boldsymbol{X}$, and U is the error. Then based on the estimate $\hat{\theta}$ (for parametric model) or partial dependence of T (for non-parametric model like tree-based models), we can analyze treatment effect. However, $\boldsymbol{X}$ is not always independent on T. If T depends on $\boldsymbol{X}$, vice versa, the inference will be biased. Multicollinearity is one simple example of such dependence. In practice, the dependence is complicated and not easy to remove. That's why we need to estimate the treatment effect using Inverse Probability Weights (https://en.wikipedia.org/wiki/Inverse_probability_weighting, Bang & Robins [2]) or the double machine learning framework Chernozhukov et al. [3], where the propensity score model $P(T = 1|\boldsymbol{X})$ is also necessary to be trained.

• What is difference between policy Optimization and personalization?
Answer: The policy refers to a system of rules to assign treatments to individuals. For example, the economic stimulus is given to Americans with income below a certain threshold ($75,000). This rule is quite simple and only one variable (income) is needed. How to determine the threshold? First, an objective function, sum of objective value for each individual, needs to be determined such as the total social welfare for this example. The policy optimization is the process of exploring the rules that maximize the objective function. The policy optimization is for a large group of people. The optimized policy might not be best for a specific individual but it should be best for the group of individuals.

The personalization focus on a specific individual: the individual should be assigned the treatment if that makes the objective metric of that individual better.

1 Why Is Causal Inference based on Observational Data Necessary? –Answer causal questions when doing experiments are impossible or too expensive

Suppose there is only one binary treatment (0: not treated, 1: treated). First, some common denotations are defined as follows:

- $Y_i(0)$: potential outcome for the i-th individual under ***no*** treatment
- $Y_i(1)$: potential outcome for the i-th individual under treatment
- $Y(0)$: population-level potential outcome under ***no*** treatment
- $Y(1)$: population-level potential outcome under treatment
- Y: observed outcome
- T: observed treatment
- y_i: observed outcome of the i-th individual
- t_i: observed treatment of the i-th individual

An individual who is treated is almost impossible to go back to the original status and be untreated. That's why 'potential' is used in the definitions of some denotations. For example, potential outcome for the i-th individual under ***no*** treatment, $Y_i(0)$, means the outcome of the i-th individual if it is not treated no matter whether $t_i = 0$ or $t_i = 1$.

In most cases, average treatment effect (ATE) is the essential parameter of interest we care about. ATE is defined by

$$\text{ATE} = \mathbb{E}[Y(1)] - \mathbb{E}[Y(0)] = \frac{1}{n}\sum_{i=1}^{n}(Y_i(1) - Y_i(0))$$

where n denotes the number of observations.

The association is defined by

$$\text{association} = \mathbb{E}[Y|T=1] - \mathbb{E}[Y|T=0] = \frac{\sum_{i=1}^{n} y_i 1\{t_i = 1\}}{\sum_{i=1}^{n} 1\{t_i = 1\}} - \frac{\sum_{i=1}^{n} y_i 1\{t_i = 0\}}{\sum_{i=1}^{n} 1\{t_i = 0\}}$$

where $\mathbb{E}[Y|T=1]$ denotes the expected value of the observed outcome of the treated individuals and $\mathbb{E}[Y|T=0]$ denotes the expected value of the observed outcome of the untreated individuals.

Many people take it for granted that association always equals to ATE. However, that is not the truth. Only for randomized control trials without confounding variables can we have association = ATE! If confounding variables exist, block randomization must be used, or we will have issues that are similar to the famous Simpson's paradox.

The following situations make the experiments impossible and causal inference based on observational data is needed to estimate ATE and other parameters of interest:

- There exists self selection bias.

For example, now we have a new version of an APP and send relevant notifications to selected individuals that can be seen as the treated group. We are interested in the effect of the new

version. The issue is that not everyone in the treatment group will download the new version at last. Then selection bias arises. In the treatment group, those who download the APP are different from the ones who do not download. To avoid the selection bias, we can compare the control group to the entire treatment group regardless of the APP download. Then the estimated treatment effect will be diluted because some ones in the treatment group are not actually treated. We can estimate the treatment effect by seeing the experiment as an observational study.

• It is impossible to randomly assign treatment.
For example, we want to investigate whether smoking affects the review score of a Rides-on-demand service driver. Now the treatment is smoking and it is obviously impossible to assign the treatment to an individual. If we directly compute the association and regard it to be the ATE, the results are biased: smoking itself is a proxy of many covariates such as if the car is smelly, if the car is clean and so on. Then the treatment effect needs to be estimated by the observations.

• There are a lot of covariates such that the block randomization is impossible or too time-consuming

• We randomly assign the treatment and do experiments but finally we find the treatment and control groups do not have similar user distribution. For example, there exist some false understanding of the covariates that determine whether two individuals are similar. If so, we can consider the experiment to be an observational study.

2 Applying DNN to Causal Inference

The DNNs overcome several technical difficulties: there might exist a huge amount of potential covariates that might be discrete or continuous, or mixed; there might exist unknown non-linearity relationships between those covariates, the outcome, and the treatment assignment. DNN is applied to causal inference in a two-step semi-parametric framework: the first step is the non-parametric step and the second one is the parametric step. In the first step, two DNNs are used to estimate the unknown functions in the propensity score model and outcome model. In the second step, parameter of interests are estimated through the corresponding influence functions.

2.1 Estimating Unknown Functions in Propensity Score & Outcome Models

There are two main models in the causal inference: propensity score model and outcome model. Two approximate DNNs are constructed to estimate these two models, respectively. In what follows, suppose we focus on the observational data with one binary treatment and one continuous outcome. Moreover, no omitted variable bias exists.

2.1.1 Propensity Score Model

The propensity score model describes the treatment assigning probability conditional on some covariates (or features):

$$p(\boldsymbol{x}^{(1)}) = \Pr(T = 1|\boldsymbol{X}^{(1)} = \boldsymbol{x}^{(1)})$$

where $p(\boldsymbol{x}^{(1)})$ denotes the propensity score, T denotes the binary treatment, and $\boldsymbol{x}^{(1)}$ is a vector of the observed covariates . The superscript (1) denotes the covariates of Set 1. The $p(\cdot)$ is one unknown function to be estimated by a DNN.

The DNN for the propensity score model, a binary classifier, estimates the binary treatment using covariates of Set 1. It trains the covariates observations in the input dataset. The input layer of that DNN corresponds to $\boldsymbol{X}^{(1)}$ and the output layer corresponds to the observed treatment variable. The activation functions for the hidden layers are all rectified linear unit (ReLU) functions and the activation function for the output layer is a sigmoid function. The loss function is the negative log-likelihood function.

2.1.2 Outcome Model

The outcome model describes how the outcome depends on the treatment and some covariates of Set 2. The covariates of Set 1 and covariates of Set 2 might be different. The outcome model is given as follows:

$$\mathbb{E}[Y|\boldsymbol{X}^{(2)} = \boldsymbol{x}^{(2)}, T = t] = G(\alpha(\boldsymbol{x}^{(2)}) + \beta(\boldsymbol{x}^{(2)})t)$$

where Y is the continuous outcome, t is the treatment, and $G(\cdot)$ is a known function. For example, $G(u)$ might be the identity function, i.e., $G(u) = u$. The superscript (2) denotes covariates Set 2. In what follows, $G(\cdot)$ is set to identity function. The $\alpha(\cdot)$ and $\beta(\cdot)$ are unknown functions to be estimated by a DNN.

The DNN for the outcome model estimates the outcome using covariates of Set 2 and the treatment variable. It trains the covariates observations in the input dataset. The input layer of that DNN corresponds to $\boldsymbol{X}^{(2)}$. After the hidden layers, there is a parameter layer with two neurons that correspond to $\alpha(\boldsymbol{x}^{(2)})$ and $\beta(\boldsymbol{x}^{(2)})$. The activation functions for this parameter layer are linear functions. Subsequently, t is included and $G(\alpha(\boldsymbol{x}^{(2)}) + \beta(\boldsymbol{x}^{(2)})t)$ is calculated as the result of the final output layer. When $G(\cdot)$ is set to the identity function, $\alpha(\boldsymbol{x}^{(2)}) + \beta(\boldsymbol{x}^{(2)})t$ is outputted as the prediction.

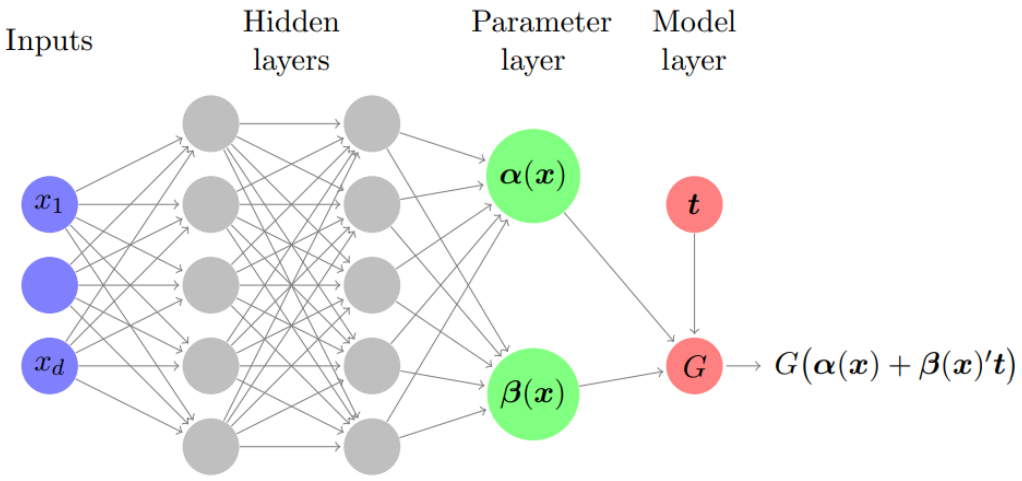

Figure 1: DNN for the outcome model

2.2 Estimating Parameters of Interest

A parameter of interest can be estimated through the following structural model:

$$\theta_0 = \mathbb{E}[H(Y, \boldsymbol{X}, T, \alpha(\boldsymbol{X}^{(2)}), \beta(\boldsymbol{X}^{(2)}); t^*)]$$

where θ_0 is a certain parameter of interest, $H(\cdot)$ is a known function that corresponds to θ_0, and $\boldsymbol{X}$ is the union of $\boldsymbol{X}^{(1)}$ and $\boldsymbol{X}^{(2)}$, and t^* is a fixed treatment value of interest. To have the expected value, a naive approach is to plug the DNN-based estimates $\hat{\alpha}(\boldsymbol{x}^{(2)})$ and $\hat{\beta}(\boldsymbol{x}^{(2)})$ into the structual model of θ_0. This estimator is known as the plug-in estimator and is given by

$$\hat{\theta}_0 = \frac{1}{n}\sum_{i=1}^{n} H(y_i, \boldsymbol{x}_i, t_i, \hat{\alpha}(\boldsymbol{x}_i^{(2)}), \hat{\beta}(\boldsymbol{x}_i^{(2)}); t^*)$$

where n is the sample size and the subscript i stands for the i-th observation. For example, if T is binary ($T \in \{0, 1\}$), θ_0 is the estimator of average treatment effect (ATE) when $H(Y, \boldsymbol{X}, T, \alpha(\boldsymbol{X}^{(2)}), \beta(\boldsymbol{X}^{(2)}); t^*) = \beta(\boldsymbol{X}^{(2)})$. Then the plug-in estimator of ATE is given by

$$\hat{\theta}_0 = \frac{1}{n}\sum_{i=1}^{n} \hat{\beta}(\boldsymbol{x}_i^{(2)})$$

This estimator might be highly biased because the covariates might have both effects on the outcome and treatment assignment. Using an influence function based estimator is one way to solve this problem. Farrell et al. [4], [5] proposed the following influence function:

$$\begin{aligned}
&\psi(y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*) \\
=&H(y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}); t^*) + \nabla \boldsymbol{H}(\boldsymbol{x})\Lambda(\boldsymbol{x})^{-1}(1, t)'(Y - G(\alpha(\boldsymbol{x}^{(2)}) + \beta(\boldsymbol{x}^{(2)})t))
\end{aligned}$$

where $\nabla \boldsymbol{H}(\boldsymbol{x}) = (\frac{\partial H}{\partial \alpha}, \frac{\partial H}{\partial \beta})$, and $\Lambda(\boldsymbol{x}) = \mathbb{E}[(1, T)'(1, T)|\boldsymbol{X} = \boldsymbol{x}]$. This estimator is doubly robust.

For inference problem with one binary treatment and the identity function $\dot{G()}$, the inverse of $\Lambda(\boldsymbol{x})$ can be computed directly and the influence function above can be simplified as follows:

$$\begin{aligned}
&\psi(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*) \\
=&H(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}); t^*) \\
&+ \frac{\dot{H}_\alpha(\boldsymbol{x})(\lambda_2(\boldsymbol{x}) - \lambda_1(\boldsymbol{x})t) + \dot{H}_\beta(\boldsymbol{x})(\lambda_0(\boldsymbol{x})t - \lambda_1(\boldsymbol{x}))}{\lambda_2(\boldsymbol{x})\lambda_0(\boldsymbol{x}) - \lambda_1(\boldsymbol{x})^2}(Y - \alpha(\boldsymbol{x}^{(2)}) - \beta(\boldsymbol{x}^{(2)})t)
\end{aligned}$$

where $\lambda_j(\boldsymbol{x}) = \mathbb{E}[T^j|\boldsymbol{X} = \boldsymbol{x}]$, $j = 0, 1, 2$, $\dot{H}_\alpha(\boldsymbol{x}) = \frac{\partial H}{\partial \alpha}$, and $\dot{H}_\beta(\boldsymbol{x}) = \frac{\partial H}{\partial \beta}$. With the binary treatment ($T \in \{0, 1\}$) we have $\lambda_0(\boldsymbol{x}) = 1$ and $\lambda_1(\boldsymbol{x}) = \lambda_2(\boldsymbol{x}) = p(\boldsymbol{x})$.

Therefore, the estimator for the parameter of interest is given by

$$\hat{\theta}_0 = \frac{\sum_{i=1}^{n} \psi(y_i, \boldsymbol{x}_i, t_i, \hat{\alpha}(\boldsymbol{x}_i^{(2)}), \hat{\beta}(\boldsymbol{x}_i^{(2)}), \Lambda(\boldsymbol{x}_i), t^*)}{n}$$

The standard error of $\hat{\theta}_0$, which is root-n consistent, is

$$\sigma(\hat{\theta}_0) = \frac{\sqrt{\sum_{i=1}^{n} \left(\psi(y_i, \boldsymbol{x}_i, t_i, \hat{\alpha}(\boldsymbol{x}_i^{(2)}), \hat{\beta}(\boldsymbol{x}_i^{(2)}), \Lambda(\boldsymbol{x}_i), t^*) - \hat{\theta}_0\right)^2}}{n}$$

Next, the calculation of parameters of interest are introduced.

2.2.1 Full-population Average Effect Parameters

The full-population average effect parameters are estimated as follows:

• $\mu_0 = \mathbb{E}[Y(0)]$:
The influence function, denoted by $\psi_{\mu_0}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)$, is defined by setting $H(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}); t^*) = \alpha(\boldsymbol{x}^{(2)})$ in the original influence function. μ_0 is estimated by

$$\mu_0 = \frac{\sum_{i=1}^n \psi_{\mu_0}(y_i, \boldsymbol{x}_i, t_i, \hat{\alpha}(\boldsymbol{x}_i^{(2)}), \hat{\beta}(\boldsymbol{x}_i^{(2)}), \Lambda(\boldsymbol{x}_i), t^*)}{n}$$

• $\mu_1 = \mathbb{E}[Y(1)]$:
The influence function, denoted by $\psi_{\mu_1}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)$, is defined by setting $H(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}); t^*) = \alpha(\boldsymbol{x}^{(2)}) + \beta(\boldsymbol{x}^{(2)})$ in the original influence function. μ_1 is estimated by

$$\mu_1 = \frac{\sum_{i=1}^n \psi_{\mu_1}(y_i, \boldsymbol{x}_i, t_i, \hat{\alpha}(\boldsymbol{x}_i^{(2)}), \hat{\beta}(\boldsymbol{x}_i^{(2)}), \Lambda(\boldsymbol{x}_i), t^*)}{n}$$

• average treatment effect (ATE), $\text{ATE} = \mathbb{E}[Y(1) - Y(0)]$:
The influence function, denoted by $\psi_{\text{ATE}}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)$, is defined by setting $H(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}); t^*) = \beta(\boldsymbol{x}^{(2)})$ in the original influence function. ATE is estimated by

$$\text{ATE} = \frac{\sum_{i=1}^n \psi_{\text{ATE}}(y_i, \boldsymbol{x}_i, t_i, \hat{\alpha}(\boldsymbol{x}_i^{(2)}), \hat{\beta}(\boldsymbol{x}_i^{(2)}), \Lambda(\boldsymbol{x}_i), t^*)}{n}$$

2.2.2 Subpopulation Average Effect Parameters

The subpopulation average effect parameters are estimated as follows:

• $\rho_{t_1,t_2} = \mathbb{E}[Y(t_1)|T = t_2]$ for $t_1, t_2 \in \{0, 1\}$:
For a single ρ_{t_1,t_2}, the influence function should be calculated by

$$\begin{aligned}
&\psi_{t_1,t_2}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*) \\
=&\frac{\Pr(t = t_2|\boldsymbol{X}^{(1)} = \boldsymbol{x}^{(1)})}{\Pr(t = t_2)} \frac{1\{t = t_1\}(Y - \mathbb{E}[Y|\boldsymbol{X}^{(2)} = \boldsymbol{x}^{(2)}, t = t_1])}{\Pr(t = t_1|\boldsymbol{X}^{(1)} = \boldsymbol{x}^{(1)})} \\
&+ \frac{1\{t = t_2\}\mathbb{E}[Y|\boldsymbol{X}^{(2)} = \boldsymbol{x}^{(2)}, t = t_1]}{\Pr(t = t_2)}
\end{aligned}$$

where $\Pr(t = t_i|\boldsymbol{X}^{(1)} = \boldsymbol{x}^{(1)})$ is acquired from the DNN for the propensity model and $\mathbb{E}[Y|\boldsymbol{X}^{(2)} = \boldsymbol{x}^{(2)}, t = t_1]$ is acquired from the DNN for the outcome model.

• average treatment effect for the treated (ATT), $\text{ATT} = \rho_{1,1} - \rho_{0,1}$:
For ATT, the influence function should be replaced by

$$\begin{aligned}
&\psi_{\text{ATT}}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*) \\
=&\psi_{1,1}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*) - \psi_{0,1}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)
\end{aligned}$$

• $\Delta_X = \rho_{1,1} - \rho_{1,0}$:
For Δ_X, the influence function should be replaced by

$$\begin{aligned}
&\psi_{\Delta_X}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*) \\
=&\psi_{1,1}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*) - \psi_{1,0}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)
\end{aligned}$$

- $\Delta_\mu = \rho_{1,0} - \rho_{0,0}$:

For Δ_μ, the influence function should be replaced by

$$\begin{aligned}&\psi_{\Delta_\mu}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)\\ =&\psi_{1,0}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*) - \psi_{0,0}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)\end{aligned}$$

- $\Delta = \rho_{1,1} - \rho_{0,0}$:

For Δ, the influence function should be replaced by

$$\begin{aligned}&\psi_{\Delta}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)\\ =&\psi_{1,1}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*) - \psi_{0,0}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)\end{aligned}$$

2.2.3 Estimating Utility of a Policy

First, a policy is defined to a rule that assigns a given set of information, represented by covariates $\boldsymbol{X}$, to treatment status; that is, a policy is a known map function that is defined by

$$s(\boldsymbol{X}) : \boldsymbol{X} \to \{0, 1\}$$

The expected utility of a policy, denoted by $\pi(s)$, is given by

$$\pi(s) = \mathbb{E}[s(\boldsymbol{X})Y(1) + (1 - s(\boldsymbol{X}))Y(0)]$$

To estimate $\pi(s)$, the influence function should be replaced by

$$\begin{aligned}&\psi_{\pi(s)}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)\\ =&s(\boldsymbol{x})\psi_{\mu_1}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*) + (1 - s(\boldsymbol{x}))\psi_{\mu_0}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)\end{aligned}$$

where $\psi_{\mu_j}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)$ denotes the influence function for μ_j ($j = 0$ or 1). The difference in expected profits of two policies is defined by

$$\pi(s_1, s_0) = \mathbb{E}[(s_1(\boldsymbol{X}) - s_0(\boldsymbol{X}))(Y(1) - Y(0))]$$

To estimate $\pi(s_1, s_0)$, the influence function should be replaced by

$$\begin{aligned}&\psi_{\pi(s_1,s_0)}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)\\ =&(s_1(\boldsymbol{x}) - s_0(\boldsymbol{x}))\psi_{\text{ATE}}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)\end{aligned}$$

where $\psi_{\text{ATE}}(Y, \boldsymbol{x}, t, \alpha(\boldsymbol{x}^{(2)}), \beta(\boldsymbol{x}^{(2)}), \Lambda(\boldsymbol{x}), t^*)$ denotes the influence function for estimating ATE.

2.3 Instrumental Variable

The instrumental variable is necessary when the residuals of the propensity model or the outcome model is correlated with the corresponding covariates (Angrist & Imbens [1]).

3 Resources

How to Use Machine Learning to Accelerate AB Testing, https://medium.com/teconomics-blog/using-ml-to-resolve-experiments-faster-bd8053ff602e

Double ML website, https://docs.doubleml.org/stable/index.html, https://github.com/DoubleML/doubleml-for-py

References

Angrist, J., & Imbens, G. (1995). *Identification and estimation of local average treatment effects.* National Bureau of Economic Research Cambridge, Mass., USA.

Bang, H., & Robins, J. M. (2005). Doubly robust estimation in missing data and causal inference models. *Biometrics*, *61*(4), 962–973.

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., & Robins, J. (2018). *Double/debiased machine learning for treatment and structural parameters.* Oxford University Press Oxford, UK.

Farrell, M. H., Liang, T., & Misra, S. (2020). Deep learning for individual heterogeneity. *arXiv preprint arXiv:2010.14694*.

Farrell, M. H., Liang, T., & Misra, S. (2021). Deep neural networks for estimation and inference. *Econometrica*, *89*(1), 181–213.