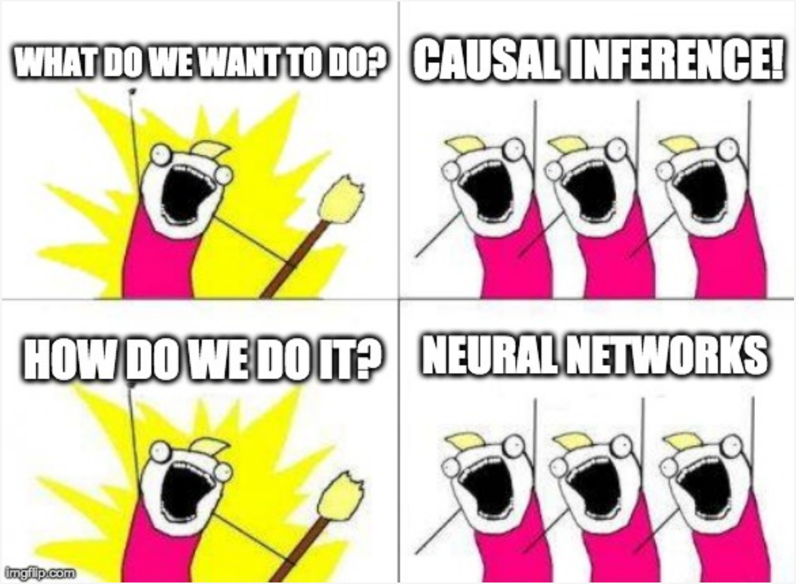
Adapting Neural Networks for the Estimation of Treatment Effects

Claudia Shi, David Blei, Victor Veitch

Columbia University

Causal Inference with Neural Nets using Observational Data



What is a Causal Question?

Questions about prediction:

- Will I have a headache tomorrow, given that I take this pill?
- What is the rate of drowning death, conditional on the ice cream sales is high?

Questions involve intervention:

- If I take this pill, will I have a headache tomorrow?
- Given that we increase the ice cream sales, what will the rate of drowning death be?

What is a Causal Question?

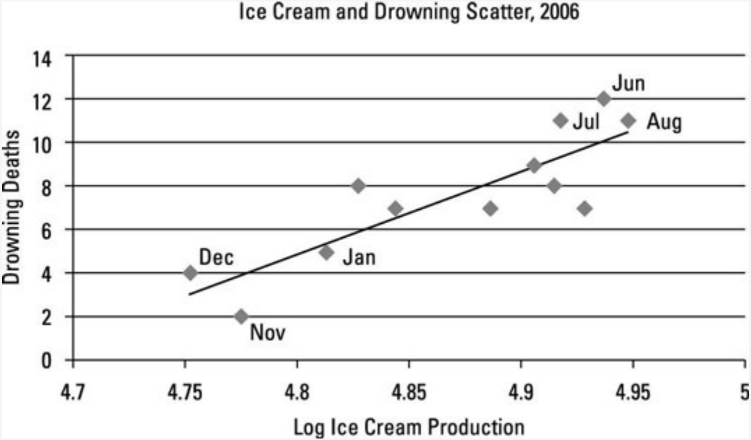
Questions about prediction:

- Will I have a headache tomorrow, given that I take this pill?
- What is the rate of drowning death, conditional on the ice cream sales is high?

Questions involve intervention:

- If I take this pill, will I have a headache tomorrow?
- Given that we increase the ice cream sales, what will the rate of drowning death be?

Observational Data: Confounding is a Problem



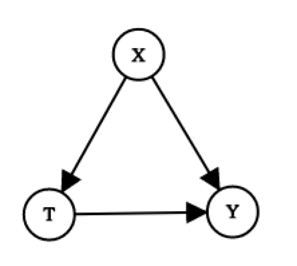
RCT Data: Not Accessible

Causal Inference with Observational Data

Example

- treatment T (patient gets a drug)
- outcome Y (whether they recover)
- covariates X (illness severity, socioeconomic status)

What is expected effect of *intervening* by assigning the drug?

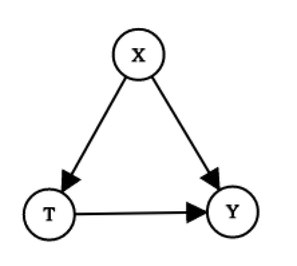


Causal Inference with Observational Data

Example

- treatment T (patient gets a drug)
- outcome Y (whether they recover)
- covariates X (illness severity, socioeconomic status)

What is expected e□ect of *intervening* by assigning the drug?



Adjustment

The average treatment effect is:

$$y = \mathbb{E}[Y \mid \mathrm{do}(T=1)] - \mathbb{E}[Y \mid \mathrm{do}(T=0)]$$

$$y \neq \mathbb{E}[Y \mid T=1] - \mathbb{E}[Y \mid T=0]$$

Theorem (No unobserved confounding)

If covariates X "block all backdoor paths" then

$$\begin{aligned} y &= \mathbb{E}[Y \mid \mathrm{do}(T=1)] - \mathbb{E}[Y \mid \mathrm{do}(T=0)] \\ &= \mathbb{E}[\mathbb{E}[Y \mid T=1, X] - \mathbb{E}[Y \mid T=0, X]] \end{aligned}$$

Adjustment

The average treatment effect is:

$$y = \mathbb{E}[Y \mid \mathrm{do}(T = 1)] - \mathbb{E}[Y \mid \mathrm{do}(T = 0)]$$

$$y \neq \mathbb{E}[Y \mid T = 1] - \mathbb{E}[Y \mid T = 0]$$

Theorem (No unobserved confounding)

If covariates X "block all backdoor paths" then

$$\begin{aligned} y &= \mathbb{E}[Y \mid \mathrm{do}(T = 1)] - \mathbb{E}[Y \mid \mathrm{do}(T = 0)] \\ &= \mathbb{E}[\mathbb{E}[Y \mid T = 1, X] - \mathbb{E}[Y \mid T = 0, X]] \end{aligned}$$

Estimation

Average Treatment E□ect:

$$y = \mathbb{E}[\mathbb{E}[Y \mid T = 1, X] - \mathbb{E}[Y \mid T = 0, X]]$$

Back Door Adjustment

Let $Q(t, x) = \mathbb{E}[Y \mid t, x]$, here is an estimator:

$$\hat{y}^Q = \frac{1}{n} \underset{i}{\text{å}} \left[\hat{Q}(1, x_i) - \hat{Q}(0, x_i) \right]$$

Estimation

Average Treatment E□ect:

$$y = \mathbb{E}[\mathbb{E}[Y \mid T = 1, X] - \mathbb{E}[Y \mid T = 0, X]]$$

Back Door Adjustment

Let $Q(t, x) = \mathbb{E}[Y \mid t, x]$, here is an estimator:

$$\hat{y}^{Q} = \frac{1}{n} \underset{i}{\text{å}} \left[\hat{Q}(1, x_i) - \hat{Q}(0, x_i) \right]$$

Alternatively:

Average Treatment E□ect:

$$y = \mathbb{E}[\mathbb{E}[Y \mid T = 1, X] - \mathbb{E}[Y \mid T = 0, X]]$$

Inverse Probability of Treatment Weighted Estimator (IPTW)

Let $g(x) = \mathrm{P}(T = 1 \mid x)$, here is another estimator

$$\hat{y}^{g} = \frac{1}{n} \underset{i}{\text{å}} \left(\frac{t_i}{\hat{g}(x_i)} - \frac{1 - t_i}{1 - \hat{g}(x_i)} \right) y_i$$

Models

expected outcome: $Q(t,x) = \mathbb{E}[Y \mid t,x]$
propensity score: $g(x) = \mathrm{P}(T = 1 \mid x)$
ATE: $y = \mathbb{E}[\mathbb{E}[Y \mid T = 1, X] - \mathbb{E}[Y \mid T = 0, X]]$

Semi-parametric e□cient

- More complicated $\hat{y}$ use both $\hat{Q}$ and $\hat{g}$
- Nice asymptotic properties: low bias / e□cient

Example: A Semi-parametric E cient Estimator

Augmented IPTW

$$\hat{y} = \hat{Q}(1,x) - \hat{Q}(0,x) + \left(\frac{t}{\hat{g}(x)} - \frac{1-t}{1-\hat{g}(x)}\right)\left\{y - \hat{Q}(t,x)\right\}$$

This talk

We want to use neural networks to model Q and g.

How should we adapt the design and training of these networks so that $\hat{y}$ is a good estimate of y?

Setup

How should we adapt the design and training of these networks so that $\hat{y}$ is a good estimate of y?

Adaptations

1. A neural network architecture—the Dragonnet—based on the sufficiency of the propensity score for causal estimation.
2. A regularization procedure—targeted regularization—based on non-parametric estimation theory.

Setup

How should we adapt the design and training of these networks so that $\hat{y}$ is a good estimate of y?

Adaptations

1. A neural network architecture—the Dragonnet—based on the sufficiency of the propensity score for causal estimation.
2. A regularization procedure—targeted regularization—based on non-parametric estimation theory.

Dragonnet

Propensity Score Suffices

Highlight

If the average treatment effect y is identifiable from observational data by adjusting for X, then adjusting for the propensity score also suffices.

Theorem (Rosenbaum and Rubin 1983)

If $y = \mathbb{E}[\mathbb{E}[Y \mid T=1, X] - \mathbb{E}[Y \mid T=0, X]]$, *then*

$$y = \mathbb{E}[\mathbb{E}[Y \mid T=1, g(X)] - \mathbb{E}[Y \mid T=0, g(X)]]$$

$\Longrightarrow$ estimate $\hat{Q}(t, x)$ using only parts of X relevant for T

Propensity Score Su□ces

Highlight

If the average treatment e□ect y is identifiable from observational data by adjusting for X, then adjusting for the propensity score also su□ces.

Theorem (Rosenbaum and Rubin 1983)

If $y = \mathbb{E}[\mathbb{E}[Y \mid T = 1, X] - \mathbb{E}[Y \mid T = 0, X]]$, *then*

$$y = \mathbb{E}[\mathbb{E}[Y \mid T = 1, g(X)] - \mathbb{E}[Y \mid T = 0, g(X)]]$$

$\Longrightarrow$ estimate $\hat{Q}(t, x)$ using only parts of X relevant for T

One Natural Approach: Nednet

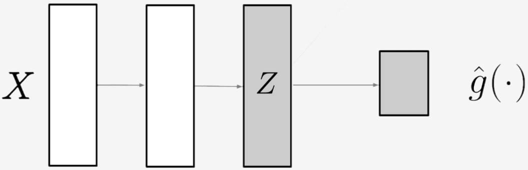
Goal

Estimate $\hat{Q}(t, x)$ using only parts of X relevant for T

One Natural Approach: Nednet

Goal

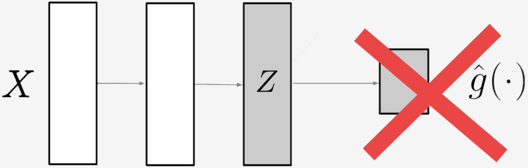
Estimate $\hat{Q}(t, x)$ using only parts of X relevant for T



One Natural Approach: Nednet

Goal

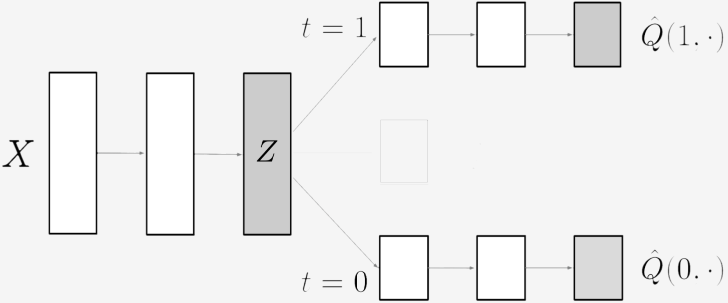
Estimate $\hat{Q}(t, x)$ using only parts of X relevant for T



One Natural Approach: Nednet

Goal

Estimate $\hat{Q}(t, x)$ using only parts of X relevant for T



One Natural Approach: Nednet

Goal

Estimate $\hat{Q}(t, x)$ using only parts of X relevant for T

Better: Dragonnet

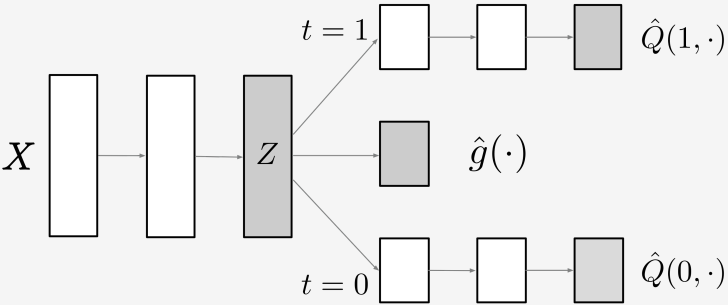
Goal: Estimate $\hat{Q}(t,x)$ using only parts of X relevant for T

- Downstream estimator: $\hat{y}^{Q} = \frac{1}{n}\sum_{i}\left[\hat{Q}(1,x_i) - \hat{Q}(0,x_i)\right]$

Better: Dragonnet

Goal: Estimate $\hat{Q}(t,x)$ using only parts of X relevant for T

- Downstream estimator: $\hat{y}^{Q} = \frac{1}{n}\sum_i \left[\hat{Q}(1,x_i) - \hat{Q}(0,x_i)\right]$



Dragonnet

Is the End-to-end model better?

Is the End-to-end model better?

Table 4: Dragonnet produces more accurate estimates than NEDnet, a multi-stage alternative. Table entries are mean absolute error over all datasets.

IHDP	$\hat{\psi}^{Q}$	$\hat{\psi}^{TMLE}$
Dragonnet	0.12 ± 0.00	0.12 ± 0.00
NEDnet	0.15 ± 0.01	0.12 ± 0.00

ACIC	$\hat{\psi}^{Q}$	$\hat{\psi}^{TMLE}$
Dragonnet	0.55	1.97
NEDnet	1.49	2.80

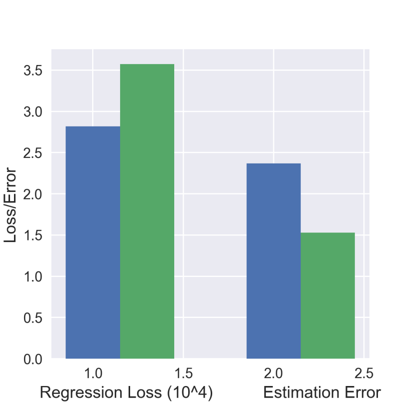
Does Dragonnet Actually Use Propensity Score Sufficiency?

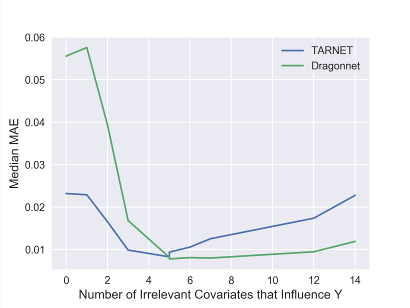
- TARNET [1] = Dragonnet without treatment head
- $\hat{y}^{Q} = \frac{1}{n} \sum_i \left[\hat{Q}(1, x_i) - \hat{Q}(0, x_i) \right]$

[1] https://arxiv.org/abs/1606.03976

Does Dragonnet Actually Use Propensity Score Sufficiency?

- TARNET [1] = Dragonnet without treatment head
- $\hat{y}^{Q} = \frac{1}{n} \sum_i \left[\hat{Q}(1, x_i) - \hat{Q}(0, x_i) \right]$





[1] https://arxiv.org/abs/1606.03976

Targeted Regularization

Recall:

expected outcome: $Q(t, x) = \mathbb{E}[Y \mid t, x]$
propensity score: $g(x) = \mathrm{P}(T = 1 \mid x)$
Average treatment effect: $y = \mathbb{E}[\mathbb{E}[Y \mid T = 1, X] - \mathbb{E}[Y \mid T = 0, X]]$

Semi-parametric efficient

- More complicated $\hat{y}$ use both $\hat{Q}$ and $\hat{g}$
- Nice asymptotic properties: low bias / efficient

Recall:

expected outcome: $Q(t, x) = \mathbb{E}[Y \mid t, x]$
propensity score: $g(x) = \mathrm{P}(T = 1 \mid x)$
Average treatment e□ect: $y = \mathbb{E}[\mathbb{E}[Y \mid T = 1, X] - \mathbb{E}[Y \mid T = 0, X]]$

Semi-parametric e□cient

- More complicated $\hat{y}$ use both $\hat{Q}$ and $\hat{g}$
- Nice asymptotic properties: low bias / e□cient

Targeted Regularization

- targeted regularization is a modification to the objective function used for neural network training.
- based on non-parametric estimation theory.
- yields a fitted model that, with a suitable downstream estimator, guarantees desirable asymptotic properties.

Targeted Regularization

- targeted regularization is a modification to the objective function used for neural network training.
- based on non-parametric estimation theory.
- yields a fitted model that, with a suitable downstream estimator, guarantees desirable asymptotic properties.

Targeted Regularization

- targeted regularization is a modification to the objective function used for neural network training.
- based on non-parametric estimation theory.
- yields a fitted model that, with a suitable downstream estimator, guarantees desirable asymptotic properties.

Targeted Regularization

- targeted regularization is a modification to the objective function used for neural network training.
- based on non-parametric estimation theory.
- yields a fitted model that, with a suitable downstream estimator, guarantees desirable asymptotic properties.

Non-parametric Estimation

Asymptotics

If $(\hat{Q}, \hat{g}, \hat{y})$ satisify a certain equation, then

- robustness in the double machine-learning sense---$\hat{y}$ converges to y at a fast rate even if $\hat{Q}$ and $\hat{g}$ converge slowly
- e□ciency---asymptotically, $\hat{y}$ has the lowest variance of any consistent estimator of y

Non-parametric Estimation

Asymptotics

If $(\hat{Q}, \hat{g}, \hat{y})$ satisify a certain equation, then

- robustness in the double machine-learning sense---$\hat{y}$ converges to y at a fast rate even if $\hat{Q}$ and $\hat{g}$ converge slowly
- e□ciency---asymptotically, $\hat{y}$ has the lowest variance of any consistent estimator of y

Non-parametric Estimating Equation

Asymptotics hold if

1. $\hat{Q}$ and $\hat{g}$ are consistent
2. $(\hat{Q}, \hat{g}, \hat{\psi})$ satisfy non-parametric estimating equation,

$$0 = \frac{1}{n} \sum_i \varphi(y_i, t_i, x_i; \hat{Q}, \hat{g}, \hat{\psi}),$$

where

$$\begin{aligned} \varphi(y, t, x; Q, g, \psi) &= Q(1, x) - Q(0, x) \\ &\quad + \left(\frac{t}{g(x)} - \frac{1-t}{1-g(x)} \right) \{ y - Q(t, x) \} - \psi \end{aligned}$$

Strategies

A-IPTW

1. Obvious: fit $\hat{Q}$ and $\hat{g}$, then choose $\hat{y}$ so non-parametric estimating equation is satisfied
2. $$\hat{y} = \hat{Q}(1,x) - \hat{Q}(0,x) + \left(\frac{t}{\hat{g}(x)} - \frac{1-t}{1-\hat{g}(x)}\right)\left\{y - \hat{Q}(t,x)\right\}$$
3. Bad in practice; $\hat{y}$ has $1/\hat{g}$ terms $\implies$ finite sample = :(

Strategies

A-IPTW

1. Obvious: fit $\hat{Q}$ and $\hat{g}$, then choose $\hat{y}$ so non-parametric estimating equation is satisfied
2. $$\hat{y} = \hat{Q}(1,x) - \hat{Q}(0,x) + \left(\frac{t}{\hat{g}(x)} - \frac{1-t}{1-\hat{g}(x)}\right)\left\{y - \hat{Q}(t,x)\right\}$$
3. Bad in practice; $\hat{y}$ has $1/\hat{g}$ terms $\implies$ finite sample = :(

Strategies

A-IPTW

1. Obvious: fit $\hat{Q}$ and $\hat{g}$, then choose $\hat{y}$ so non-parametric estimating equation is satisfied
2. $$\hat{y} = \hat{Q}(1,x) - \hat{Q}(0,x) + \left(\frac{t}{\hat{g}(x)} - \frac{1-t}{1-\hat{g}(x)}\right)\left\{y - \hat{Q}(t,x)\right\}$$
3. Bad in practice; $\hat{y}$ has $1/\hat{g}$ terms $\implies$ finite sample = :(

Strategies

A-IPTW

1. Obvious: fit $\hat{Q}$ and $\hat{g}$, then choose $\hat{y}$ so non-parametric estimating equation is satisfied
2. $$\hat{y} = \hat{Q}(1,x) - \hat{Q}(0,x) + \left(\frac{t}{\hat{g}(x)} - \frac{1-t}{1-\hat{g}(x)}\right)\left\{y - \hat{Q}(t,x)\right\}$$
3. Bad in practice; $\hat{y}$ has $1/\hat{g}$ terms $\implies$ finite sample = :(

Alternative

- Choose $\hat{y}^Q = \frac{1}{n}\sum_i \left[\hat{Q}(1,x_i) - \hat{Q}(0,x_i)\right]$—no bad $1/\hat{g}$ terms
- Fit $\hat{Q}$ and $\hat{g}$ so non-parametric estimating equation is satisfied

Strategies

A-IPTW

1. Obvious: fit $\hat{Q}$ and $\hat{g}$, then choose $\hat{y}$ so non-parametric estimating equation is satisfied
2. $$\hat{y} = \hat{Q}(1,x) - \hat{Q}(0,x) + \left(\frac{t}{\hat{g}(x)} - \frac{1-t}{1-\hat{g}(x)}\right)\left\{y - \hat{Q}(t,x)\right\}$$
3. Bad in practice; $\hat{y}$ has $1/\hat{g}$ terms $\implies$ finite sample = :(

Alternative

- Choose $\hat{y}^{Q} = \frac{1}{n}\text{å}_i\left[\hat{Q}(1,x_i) - \hat{Q}(0,x_i)\right]$—no bad $1/\hat{g}$ terms
- Fit $\hat{Q}$ and $\hat{g}$ so non-parametric estimating equation is satisfied

Targeted Regularization

Introduce extra parameter e and regularization term $g(y,t,x;q,e)$

$$\tilde{Q}(t_i,x_i;q,e) = Q^{\mathrm{nn}}(t_i,x_i;q) + e\left[\frac{t_i}{g^{\mathrm{nn}}(x_i;q)} - \frac{1-t_i}{1-g^{\mathrm{nn}}(x_i;q)}\right]$$

$$g(y_i,t_i,x_i;q,e) = (y_i - \tilde{Q}(t_i,x_i;q,e))^2$$

Then train as

$$\hat{q},\hat{e} = \underset{q,e}{\mathrm{argmin}}\Big[\underbrace{\hat{R}(q;\boldsymbol{X})}_{\text{usual objective}} + \underbrace{a\frac{1}{n}\mathring{a}_i\, g(y_i,t_i,x_i;q,e)}_{\text{targeted regularization}}\Big]$$

Payo□

Define an estimator $\hat{y}^{\text{treg}}$,

$$\hat{y}^{\text{treg}} = \frac{1}{n} \sum_i \hat{Q}^{\text{treg}}(1, x_i) - \hat{Q}^{\text{treg}}(0, x_i), \quad \text{where}$$

$$\hat{Q}^{\text{treg}} = \tilde{Q}(\cdot, \cdot; \hat{q}, \hat{e}).$$

Payoff

Define an estimator $\hat{y}^{\text{treg}}$,

$$\hat{y}^{\text{treg}} = \frac{1}{n} \sum_{i} \hat{Q}^{\text{treg}}(1, x_i) - \hat{Q}^{\text{treg}}(0, x_i), \quad \text{where}$$

$$\hat{Q}^{\text{treg}} = \tilde{Q}(\cdot, \cdot; \hat{q}, \hat{e}).$$

The point is:

$$0 = \partial_e \left(\hat{R}(q; \boldsymbol{X}) + a \frac{1}{n} \sum_{i} g(y_i, t_i, x_i; q, e) \right) \Big|_{\hat{e}} = a \frac{1}{n} \sum_{i} j\,(y_i, t_i, x_i; \hat{Q}^{\text{treg}}, \hat{g}, \hat{y}^{\text{treg}}).$$

Payo□

Define an estimator $\hat{y}^{\text{treg}}$,

$$\hat{y}^{\text{treg}} = \frac{1}{n}\,\text{å}_i\, \hat{Q}^{\text{treg}}(1, x_i) - \hat{Q}^{\text{treg}}(0, x_i), \quad \text{where}$$
$$\hat{Q}^{\text{treg}} = \tilde{Q}(\cdot, \cdot; \hat{q}, \hat{e}).$$

The point is:

$$0 = \P_e\left(\hat{R}(q; \boldsymbol{X}) + a\frac{1}{n}\,\text{å}_i\, g(y_i, t_i, x_i; q, e)\right)\Big|_{\hat{e}} = a\frac{1}{n}\,\text{å}_i\, j\,(y_i, t_i, x_i; \hat{Q}^{\text{treg}}, \hat{g}, \hat{y}^{\text{treg}}).$$

That is, minimizing the targeted regularization term forces $(\hat{Q}^{\text{treg}}, \hat{g}, \hat{y}^{\text{treg}})$ to satisfy the non-parametric estimating equation.

Experiment

Infant Health Development Program Benchmark (IHDP)

Method	Δ_{in}	Δ_{out}	Δ_{all}
BNN [JSS16]	$0.37 \pm .03$	$0.42 \pm .03$	—
TARNET [SJS16]	$0.26 \pm .01$	$0.28 \pm .01$	—
CFR Wass[SJS16]	$0.25 \pm .01$	$0.27 \pm .01$	—
CEVAEs [Lou+17]	$0.34 \pm .01$	$0.46 \pm .02$	—
GANITE [YJS18]	$0.43 \pm .05$	$0.49 \pm .05$	—
baseline (TARNET)	$0.16 \pm .01$	$0.21 \pm .01$	$0.13 \pm .00$
baseline + t-reg	$0.15 \pm .01$	$0.20 \pm .01$	$0.12 \pm .00$
Dragonnet	$0.14 \pm .01$	$0.21 \pm .01$	$0.12 \pm .00$
Dragonnet + t-reg	$0.14 \pm .01$	$0.20 \pm .01$	$0.11 \pm .00$

IBM Causal Inference Benchmarking Framework (ACIC)

Table 2: Dragonnet and targeted regularization improve estimation on average on ACIC 2018. Table entries are mean absolute error over all datasets.

Method	Δ_{all}
baseline (TARNET)	1.45
baseline + t-reg	1.40
Dragonnet	0.55
Dragonnet + t-reg	0.35

Table 3: Dragonnet and targeted regularization improve over the baseline about half the time, but improvement is substantial when it does happen. Error values are mean absolute error on ACIC 2018.

ψ^Q	$\%_{improve}$	$\uparrow_{avg}$	$\downarrow_{avg}$
baseline:	0%	0	0
+ t-reg	42%	0.30	0.11
+ dragon	63%	1.42	0.01
+ dragon & t-reg	46%	2.37	0.01

Summary

Summary

- Dragonnet: a neural network architecture based on the sufficiency of the propensity score for causal estimation.
- targeted regularization: a regularization procedure based on non-parametric estimation theory.
- They both work!

Thank You!

- Adapting Neural Networks for the Estimation of Treatment Effects.
  arxiv:1906.02120