Causal Effect Estimation with Hidden Confounders using Instruments and Proxies

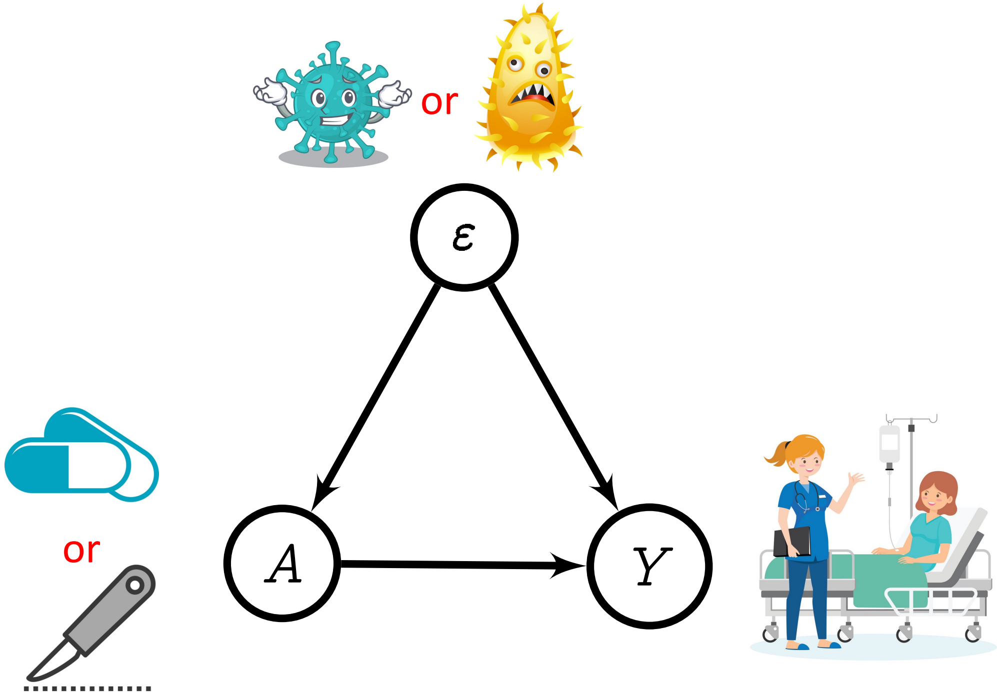
Arthur Gretton

Gatsby Computational Neuroscience Unit
Google Deepmind

ELLIS Robust Machine Learning Workshop, 2023

Introduction: observation vs intervention

Conditioning from observation: $\mathbb{E}[Y|A=a] = \sum_x \mathbb{E}[Y|a,\varepsilon]p(\varepsilon|a)$

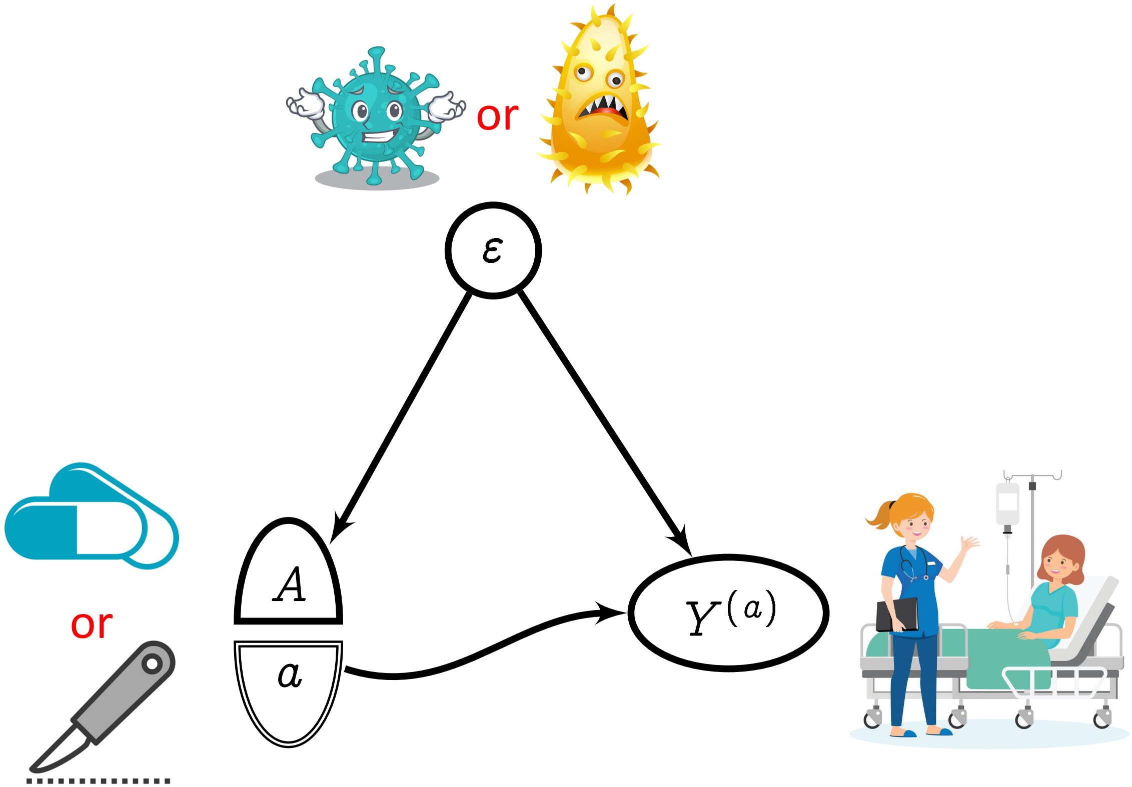


From our *observations* of historical hospital data:

- $P(Y = \text{cured}|A = \text{pills}) = 0.80$
- $P(Y = \text{cured}|A = \text{surgery}) = 0.72$

Introduction: observation vs intervention

Average causal effect (intervention): $\mathbb{E}[Y^{(a)}] = \sum_{\varepsilon} \mathbb{E}[Y|a,\varepsilon]p(\varepsilon)$

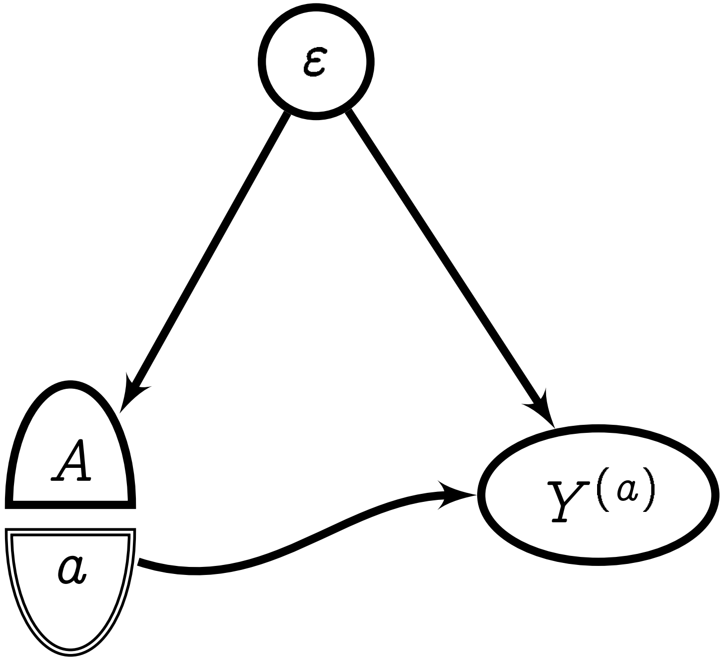


From our *intervention* (making all patients take a treatment):

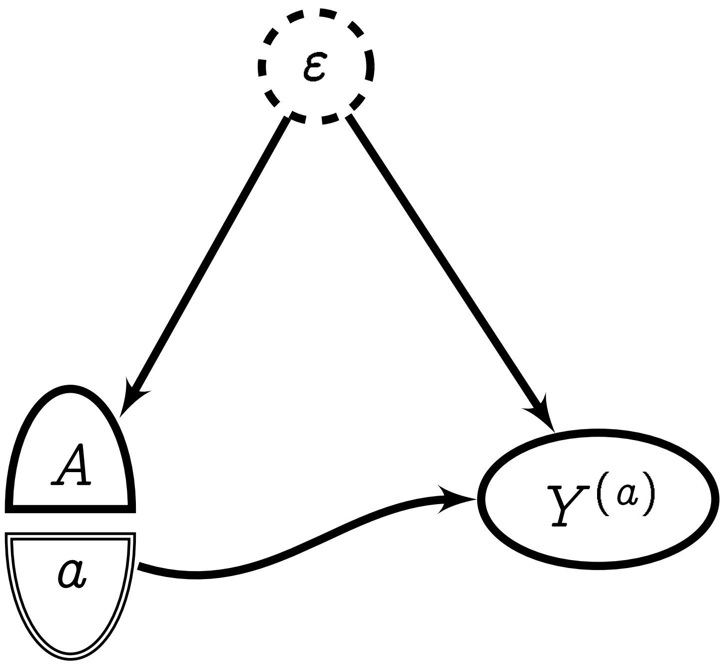
- $P(Y^{(\text{pills})} = \text{cured}) = 0.64$
- $P(Y^{(\text{surgery})} = \text{cured}) = 0.75$

Richardson, Robins (2013), Single World Intervention Graphs (SWIGs): A Unification of the Counterfactual and Graphical Approaches to Causality

Questions we will solve



Questions we will solve



Outline

Causal effect estimation, robust to hidden covariates:

- Instrumental variables
- Proxy variables

What's new? What is it good for?

- Treatment A, covariates X, etc can be multivariate, complicated...
- ...by using kernel or adaptive neural net feature representations

Model assumption: linear functions of features

All learned functions will take the form:

$$\gamma(x) = \gamma^\top \varphi(x) = \langle \gamma, \varphi(x) \rangle_{\mathcal{H}}$$

Model assumption: linear functions of features

All learned functions will take the form:

$$\gamma(x) = \gamma^\top \varphi(x) = \langle \gamma, \varphi(x) \rangle_{\mathcal{H}}$$

Option 1: Finite dictionaries of learned neural net features $\varphi_\theta(x)$ (linear final layer γ)

Xu, G., A Neural mean embedding approach for back-door and front-door adjustment. (ICLR 23)

Xu, Chen, Srinivasan, de Freitas, Doucet, G. Learning Deep Features in Instrumental Variable Regression. (ICLR 21)

Option 2: Infinite dictionaries of fixed kernel features:

$$\langle \varphi(x_i), \varphi(x) \rangle_{\mathcal{H}} = k(x_i, x)$$

Kernel is feature dot product.

Singh, Xu, G. Kernel Methods for Causal Functions: Dose, Heterogeneous, and Incremental Response Curves. (Biometrika, in revision)

Singh, Sahani, G. Kernel Instrumental Variable Regression. (NeurIPS 19)

Model fitting: ridge regression

Learn $\gamma_0(x) := \mathbb{E}[Y|X = x]$ from features $\varphi(x_i)$ with outcomes y_i:

$$\hat{\gamma} \quad = \quad \arg\min_{\gamma \in \mathcal{H}} \left(\sum_{i=1}^{n} \left(y_i - \langle \gamma, \varphi(x_i) \rangle_{\mathcal{H}} \right)^2 + \lambda \|\gamma\|_{\mathcal{H}}^2 \right).$$

Model fitting: ridge regression

Learn $\gamma_0(x) := \mathbb{E}[Y|X=x]$ from features $\varphi(x_i)$ with outcomes y_i:

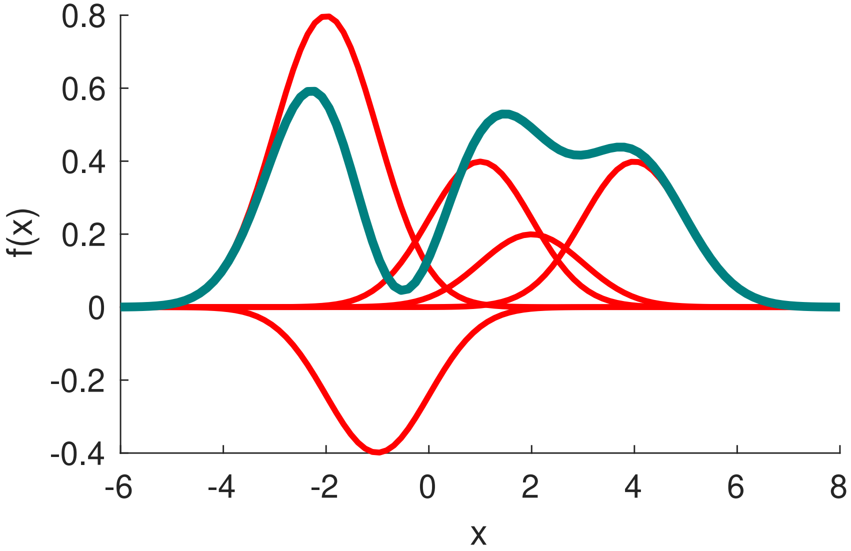
$$\hat{\gamma} \quad = \quad \arg\min_{\gamma\in\mathcal{H}} \left(\sum_{i=1}^{n} (y_i - \langle \gamma, \varphi(x_i) \rangle_{\mathcal{H}})^2 + \lambda \|\gamma\|_{\mathcal{H}}^2 \right).$$

Neural net solution at x:

$$\hat{\gamma}(x) = C_{YX}(C_{XX}+\lambda)^{-1}\varphi(x)$$

$$C_{YX} = \frac{1}{n}\sum_{i=1}^{n} [y_i \, \varphi(x_i)^\top]$$

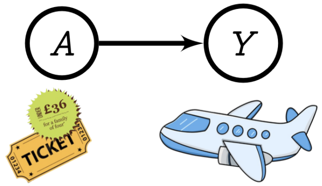
$$C_{XX} = \frac{1}{n}\sum_{i=1}^{n} [\varphi(x_i) \, \varphi(x_i)^\top]$$



Instrumental variable regression

Illustration: ticket prices for air travel

Ticket price A, seats sold Y.

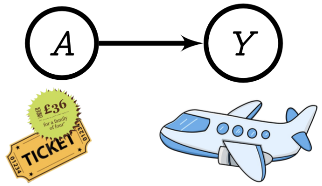


What is the effect on seats sold $Y^{(a)}$ of intervening on price a?

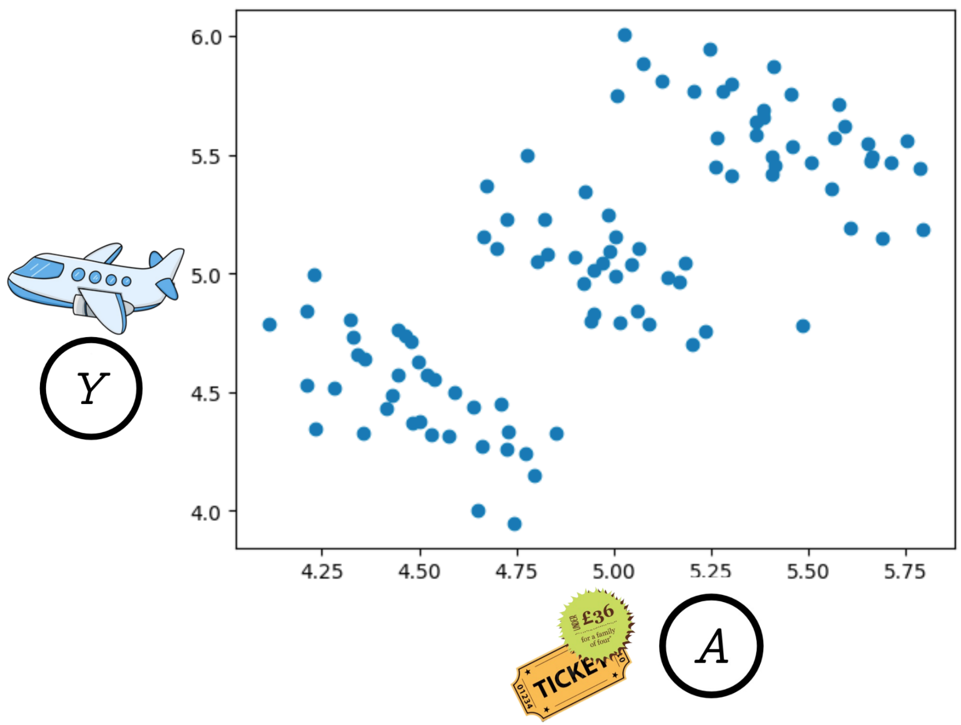
Simplification of example from Hartford, Lewis, Leyton-Brown, Taddy (2017): Deep IV: A Flexible Approach for Counterfactual Prediction.

Illustration: ticket prices for air travel

Ticket price A, seats sold Y.



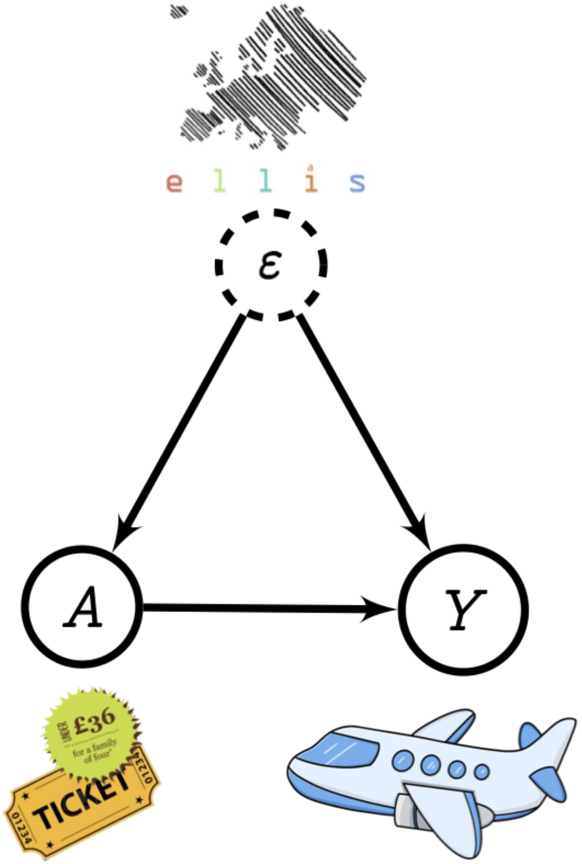
What is the effect on seats sold $Y^{(a)}$ of intervening on price a?



Simplification of example from Hartford, Lewis, Leyton-Brown, Taddy (2017): Deep IV: A Flexible Approach for Counterfactual Prediction.

Illustration: ticket prices for air travel

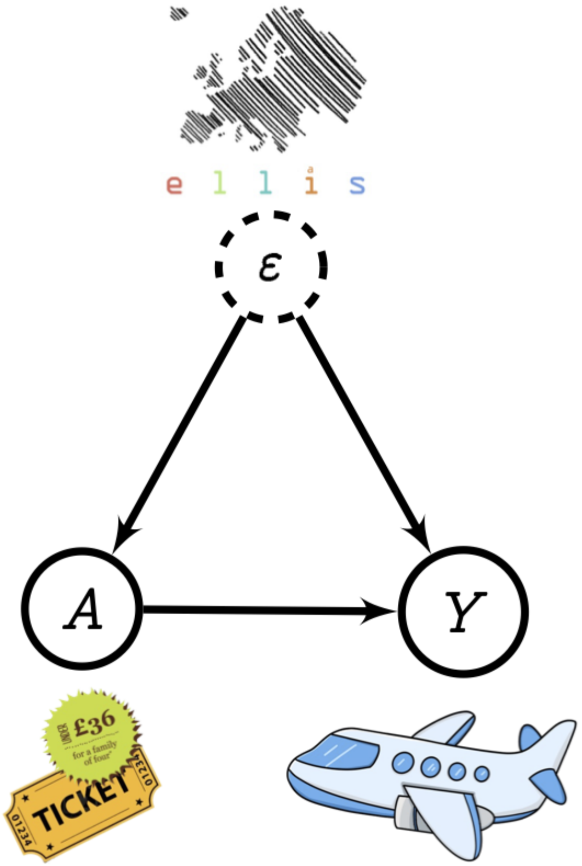
Unobserved variable ε =desire for travel, affects *both* price (via airline algorithms) *and* seats sold.



- Desire for travel:
  $\varepsilon \sim \mathcal{N}(\mu, 0.1)$
  $\mu \sim \mathcal{U}\left\{-\frac{1}{2}, 0, \frac{1}{2}\right\}$

Illustration: ticket prices for air travel

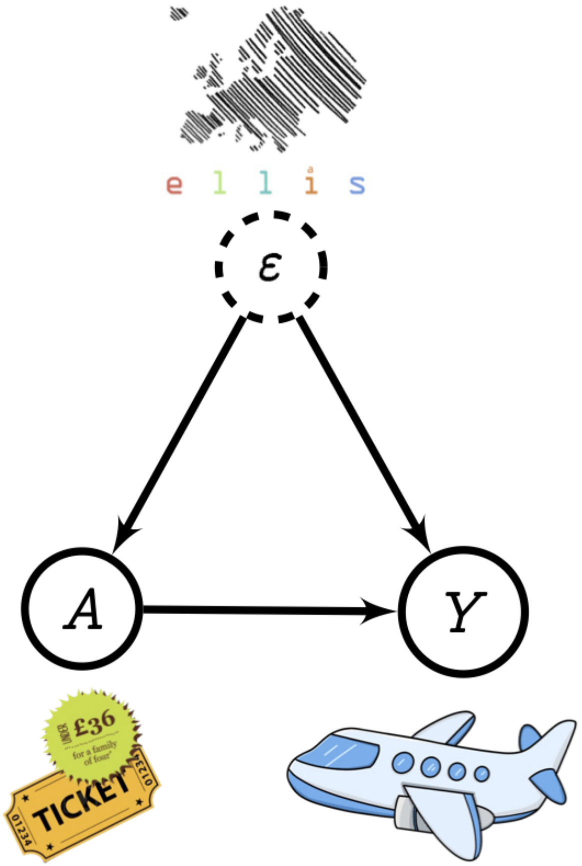
Unobserved variable ε =desire for travel, affects *both* price (via airline algorithms) *and* seats sold.



- **Desire for travel:**
  $\varepsilon \sim \mathcal{N}(\mu, 0.1)$
  $\mu \sim \mathcal{U}\left\{-\frac{1}{2}, 0, \frac{1}{2}\right\}$
- Price:
  $A = \varepsilon + Z,$
  $Z \sim \mathcal{N}(5, 0.04)$

Illustration: ticket prices for air travel

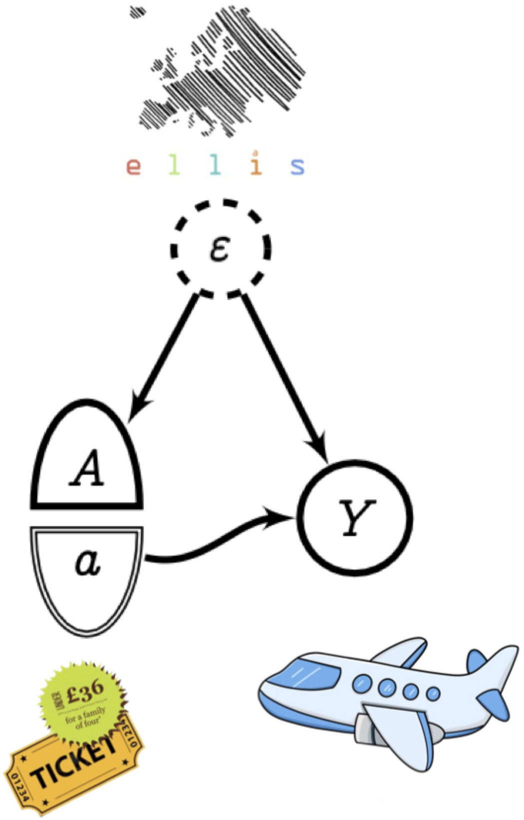
Unobserved variable ε =desire for travel, affects *both* price (via airline algorithms) *and* seats sold.



- Desire for travel:
  $\varepsilon \sim \mathcal{N}(\mu, 0.1)$
  $\mu \sim \mathcal{U}\left\{-\frac{1}{2}, 0, \frac{1}{2}\right\}$
- Price:
  $A = \varepsilon + Z,$
  $Z \sim \mathcal{N}(5, 0.04)$
- Seats sold:
  $Y = 10 - A + 2\varepsilon$

Illustration: ticket prices for air travel

Unobserved variable ε =desire for travel, affects *both* price (via airline algorithms) *and* seats sold.



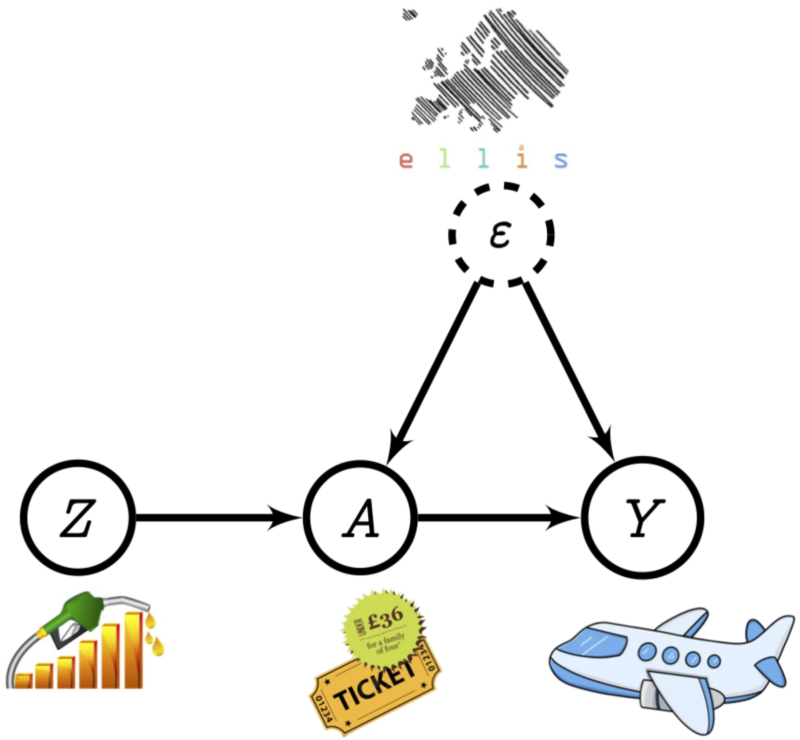
- Desire for travel:
  $\varepsilon \sim \mathcal{N}(\mu, 0.1)$
  $\mu \sim \mathcal{U}\left\{-\frac{1}{2}, 0, \frac{1}{2}\right\}$
- Price:
  $A = \varepsilon + Z,$
  $Z \sim \mathcal{N}(5, 0.04)$
- Seats sold:
  $Y = 10 - A + 2\varepsilon$

Average treatment effect:

$$\mathrm{ATE}(a) = \mathbb{E}[Y^{(a)}] = \int (10 - a + 2\varepsilon)\, dp(\varepsilon) = 10 - a$$

Illustration: ticket prices for air travel

Unobserved variable ε =desire for travel, affects *both* price (via airline algorithms) *and* seats sold.



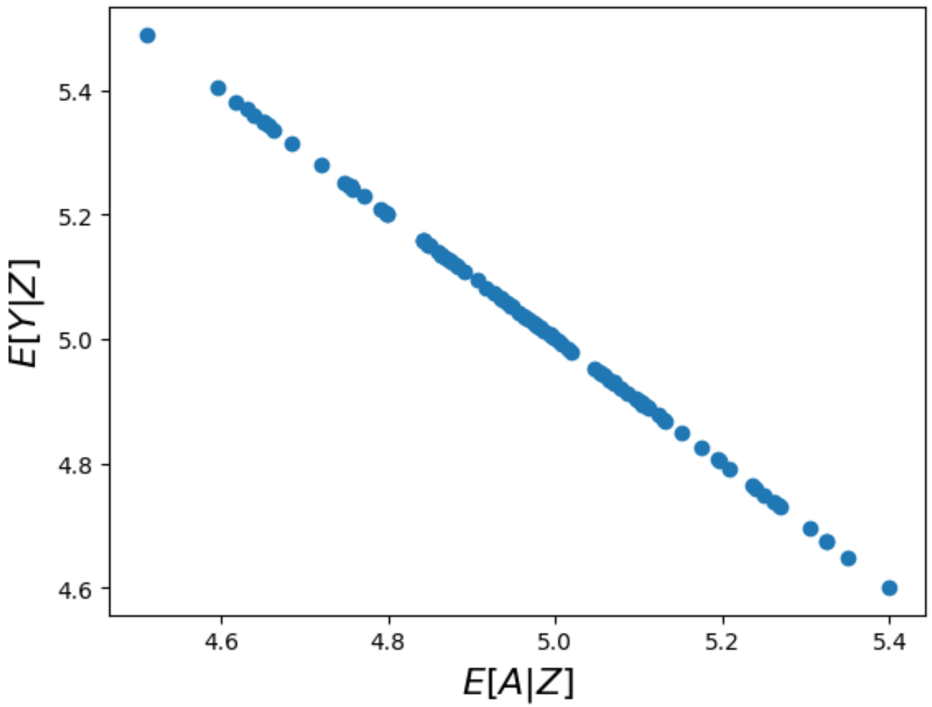
- Desire for travel:
  $\varepsilon \sim \mathcal{N}(\mu, 0.1)$
  $\mu \sim \mathcal{U}\left\{-\frac{1}{2}, 0, \frac{1}{2}\right\}$
- Price:
  $A = \varepsilon + Z,$
  $Z \sim \mathcal{N}(5, 0.04)$
- Seats sold:
  $Y = 10 - A + 2\varepsilon$

Z is an instrument (cost of fuel). Condition on Z,

$$\mathbb{E}[Y|Z] = 10 - \mathbb{E}[A|Z] + 2\underbrace{\mathbb{E}[\varepsilon|Z]}_{=0}$$

Illustration: ticket prices for air travel

Unobserved variable ε =desire for travel, affects *both* price (via airline algorithms) *and* seats sold.



- Desire for travel:
  $\varepsilon \sim \mathcal{N}(\mu, 0.1)$
  $\mu \sim \mathcal{U}\left\{-\frac{1}{2}, 0, \frac{1}{2}\right\}$
- Price:
  $A = \varepsilon + Z,$
  $Z \sim \mathcal{N}(5, 0.04)$
- Seats sold:
  $Y = 10 - A + 2\varepsilon$

Z is an instrument (cost of fuel). Condition on Z,

$$\mathbb{E}[Y|Z] = 10 - \mathbb{E}[A|Z] + 2\underbrace{\mathbb{E}[\varepsilon|Z]}_{=0}$$

Regressing from $\mathbb{E}[A|Z]$ to $\mathbb{E}[Y|Z]$ recovers ATE!

Instrumental variable regression

The Sveriges Riksbank Prize in Economic Sciences in Memory of Alfred Nobel 2021

© Nobel Prize Outreach. Photo: Paul Kennedy

David Card

Prize share: 1/2

© Nobel Prize Outreach. Photo: Risdon Photography

Joshua D. Angrist

Prize share: 1/4

© Nobel Prize Outreach. Photo: Paul Kennedy

Guido W. Imbens

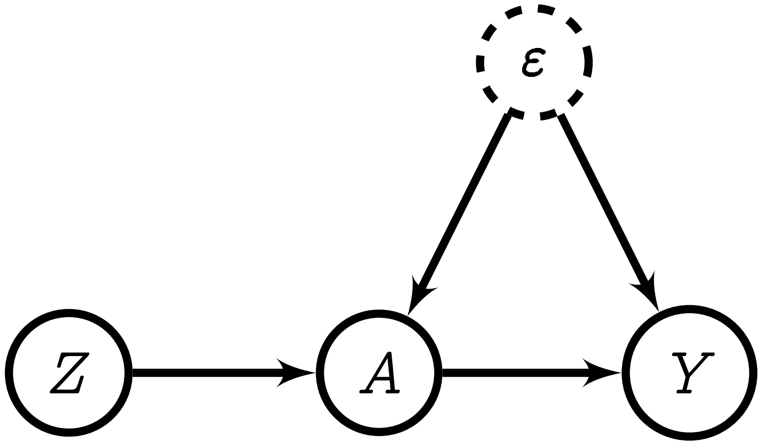
Prize share: 1/4

The Sveriges Riksbank Prize in Economic Sciences in Memory of Alfred Nobel 2021 was divided, one half awarded to David Card "for his empirical contributions to labour economics", the other half jointly to Joshua D. Angrist and Guido W. Imbens "for their methodological contributions to the analysis of causal relationships"

Instrumental variable regression with NN features

Definitions:

- ε: unobserved confounder.
- A: treatment
- Y: outcome
- Z: instrument



Assumptions

$$\mathbb{E}[\varepsilon] = 0 \qquad \mathbb{E}[\varepsilon|Z] = 0$$

$$Z \not\perp\!\!\!\perp A$$

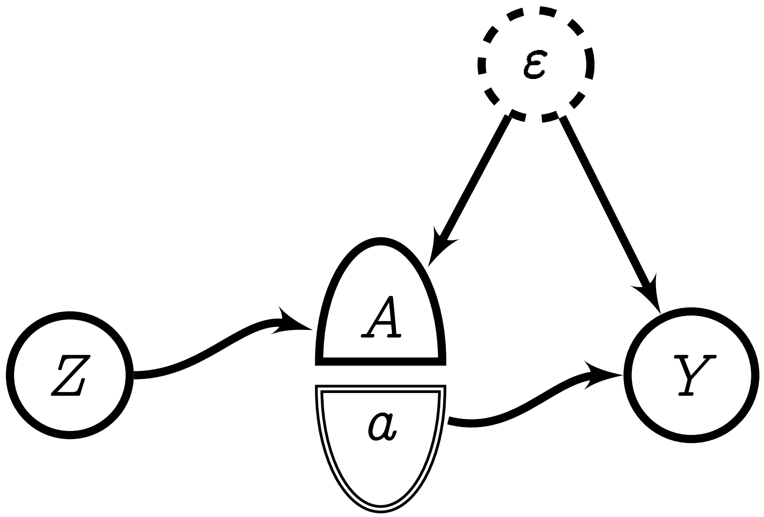
$$(Y \perp\!\!\!\perp Z|A)_{G_{\bar{A}}}$$

$$Y = \gamma^\top \phi_\theta(A) + \varepsilon$$

Instrumental variable regression with NN features

Definitions:

- ε: unobserved confounder.
- A: treatment
- Y: outcome
- Z: instrument



Assumptions

$$\mathbb{E}[\varepsilon] = 0 \qquad \mathbb{E}[\varepsilon|Z] = 0$$

$$Z \not\perp\!\!\!\perp A$$

$$(Y \perp\!\!\!\perp Z|A)_{G_{\bar{A}}}$$

$$Y = \gamma^\top \phi_\theta(A) + \varepsilon$$

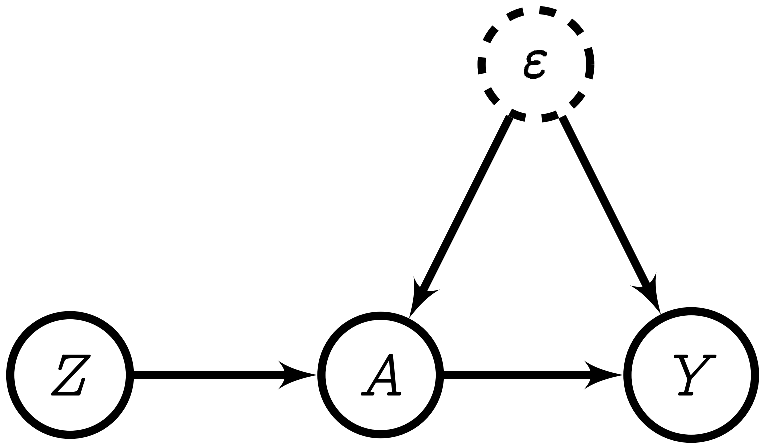
Average treatment effect:

$$\mathrm{ATE}(a) = \int \mathbb{E}(Y|\varepsilon, a)dp(\varepsilon) = \gamma^\top \phi_\theta(a)$$

Instrumental variable regression with NN features

Definitions:

- ε: unobserved confounder.
- A: treatment
- Y: outcome
- Z: instrument



Assumptions

$$\mathbb{E}[\varepsilon] = 0 \qquad \mathbb{E}[\varepsilon|Z] = 0$$

$$Z \not\perp\!\!\!\perp A$$

$$(Y \perp\!\!\!\perp Z|A)_{G_{\bar{A}}}$$

$$Y = \gamma^\top \phi_\theta(A) + \varepsilon$$

Average treatment effect:

$$\mathrm{ATE}(a) = \int \mathbb{E}(Y|\varepsilon, a)\,dp(\varepsilon) = \gamma^\top \phi_\theta(a)$$

IV regression: Condition both sides on Z,

$$\mathbb{E}[Y|Z] = \gamma^\top \mathbb{E}[\phi_\theta(A)|Z] + \underbrace{\mathbb{E}[\varepsilon|Z]}_{=0}$$

Two-stage least squares for IV regression

Kernel features (NeurIPS 2019):

arXiv.org > cs > arXiv:1906.00232

Computer Science > Machine Learning

[Submitted on 1 Jun 2019 (v1), last revised 15 Jul 2020 (this version, v6)]

Kernel Instrumental Variable Regression

Rahul Singh, Maneesh Sahani, Arthur Gretton

NN features (ICLR 2021):

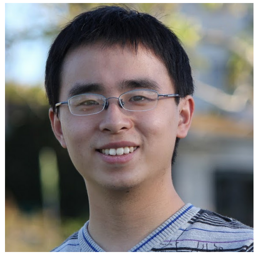
arXiv > cs > arXiv:2010.07154

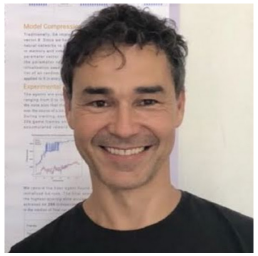
Computer Science > Machine Learning

[Submitted on 14 Oct 2020 (v1), last revised 1 Nov 2020 (this version, v3)]

Learning Deep Features in Instrumental Variable Regression

Liyuan Xu, Yutian Chen, Siddarth Srinivasan, Nando de Freitas, Arnaud Doucet, Arthur Gretton





Code for NN and kernel IV methods:

`https://github.com/liyuan9988/DeepFeatureIV/`

Two-stage least squares for IV regression

Kernel features (NeurIPS 2019):

arXiv.org > cs > arXiv:1906.00232

Computer Science > Machine Learning

[Submitted on 1 Jun 2019 (v1), last revised 15 Jul 2020 (this version, v6)]

Kernel Instrumental Variable Regression

Rahul Singh, Maneesh Sahani, Arthur Gretton

NN features (ICLR 2021):

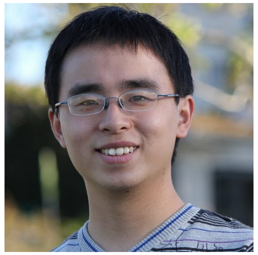
arXiv > cs > arXiv:2010.07154

Computer Science > Machine Learning

[Submitted on 14 Oct 2020 (v1), last revised 1 Nov 2020 (this version, v3)]

Learning Deep Features in Instrumental Variable Regression

Liyuan Xu, Yutian Chen, Siddarth Srinivasan, Nando de Freitas, Arnaud Doucet, Arthur Gretton



Code for NN and kernel IV methods:

`https://github.com/liyuan9988/DeepFeatureIV/`

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 \|\gamma\|^2$$

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2$$

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}||\phi_\theta(A) - F\phi_\zeta(Z)||^2 + \lambda_1 ||F||^2_{HS}$$

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2$$

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}||\phi_\theta(A) - F\phi_\zeta(Z)||^2 + \lambda_1 ||F||^2_{HS}$$

Challenge: how to learn θ?

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2$$

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}||\phi_\theta(A) - F\phi_\zeta(Z)||^2 + \lambda_1 ||F||^2_{HS}$$

Challenge: how to learn θ?

From Stage 2 regression?

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2$$

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}||\phi_\theta(A) - F\phi_\zeta(Z)||^2 + \lambda_1 ||F||_{HS}^2$$

Challenge: how to learn θ?

From Stage 2 regression?

...which requires $\mathbb{E}[\phi_\theta(A)|Z]$ from Stage 1 regression

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2$$

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}||\phi_\theta(A) - F\phi_\zeta(Z)||^2 + \lambda_1 ||F||^2_{HS}$$

Challenge: how to learn θ?

From Stage 2 regression?

...which requires $\mathbb{E}[\phi_\theta(A)|Z]$ from Stage 1 regression

...which requires $\phi_\theta(A)$... which requires θ...

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[\left(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z]\right)^2\right] + \lambda_2 \|\gamma\|^2$$

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}\|\phi_\theta(A) - F\phi_\zeta(Z)\|^2 + \lambda_1 \|F\|^2_{HS}$$

Challenge: how to learn θ?

From Stage 2 regression?

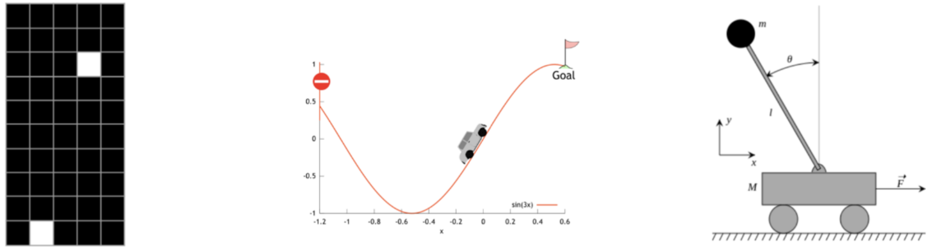
...which requires $\mathbb{E}[\phi_\theta(A)|Z]$ from Stage 1 regression

...which requires $\phi_\theta(A)$... which requires θ...

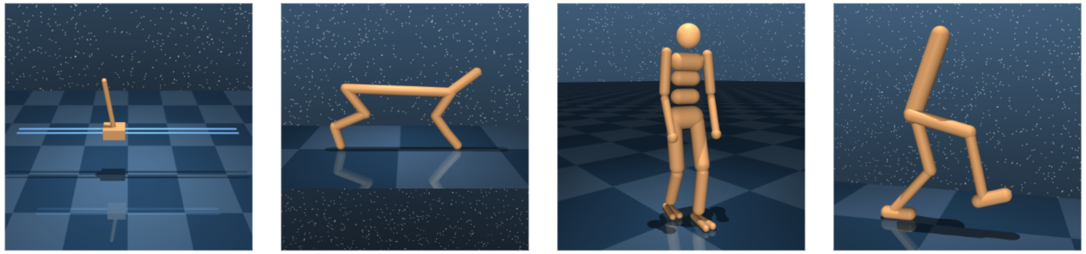
Use the linear final layers! (i.e. γ and F)

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

Neural IV in reinforcement learning



(a) Catch (b) Mountain Car (c) Cartpole



(a) Cartpole Swingup (b) Cheetah Run (c) Humanoid Run (d) Walker Walk

Policy evaluation: want Q-value:

$$Q^{\pi}(s, a) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R_t \middle| S_0 = s, A_0 = a\right]$$

for policy $\pi(A|S = s)$.

Osband et al (2019). Behaviour suite for reinforcement learning.https://github.com/deepmind/bsuite
Tassa et al. (2020). dm_control:Software and tasks for continuous control.
https://github.com/deepmind/dm_control

Application of IV: reinforcement learning

Q value is a minimizer of Bellman loss

$$\mathcal{L}_{\text{Bellman}} = \mathbb{E}_{SAR}\left[\left(R + \gamma[\mathbb{E}\left[Q^{\pi}(S', A')|S, A\right] - Q^{\pi}(S, A)\right)^2\right].$$

Corresponds to "IV-like" problem

$$\mathcal{L}_{\text{Bellman}} = \mathbb{E}_{YZ}\left[(Y - \mathbb{E}[f(X)|Z])^2\right]$$

with

$$\begin{aligned} Y &= R, \\ X &= (S', A', S, A) \\ Z &= (S, A), \\ f_0(X) &= Q^{\pi}(s, a) - \gamma Q^{\pi}(s', a') \end{aligned}$$

RL experiments and data:

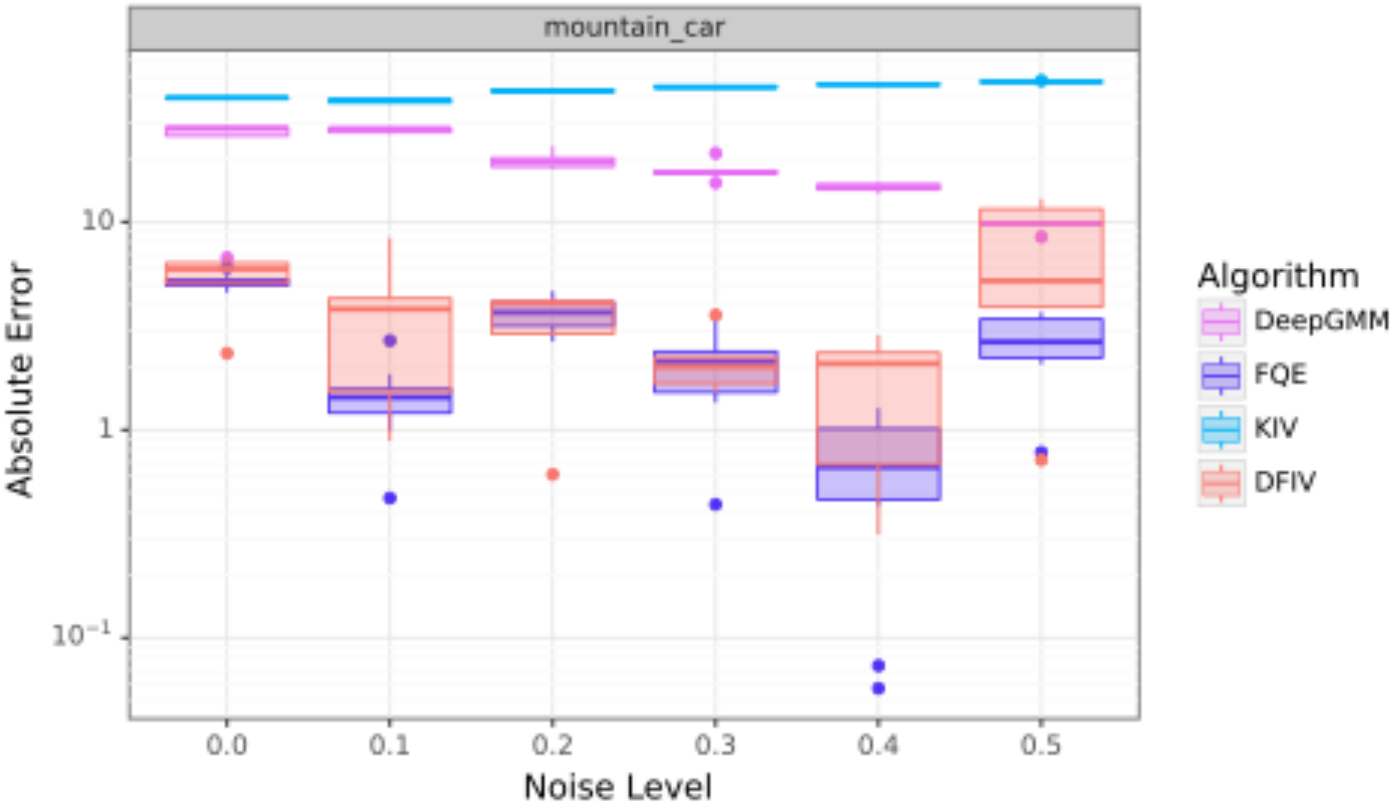
https://github.com/liyuan9988/IVOPEwithACME

Bradtke and Barto (1996). Linear least-squares algorithms for temporal difference learning.

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021)

Chen, Xu, Gulcehre, Le Paine, G, De Freitas, Doucet (2022). On Instrumental Variable Regression for Deep Offline Policy Evaluation.

Results on mountain car problem



Good performance compared with FQE.

Warning: IV assumption can fail when regression underfits. See papers for details.

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021)

Chen, Xu, Gulcehre, Le Paine, G, De Freitas, Doucet (2022). On Instrumental Variable Regression for Deep Offline Policy Evaluation.

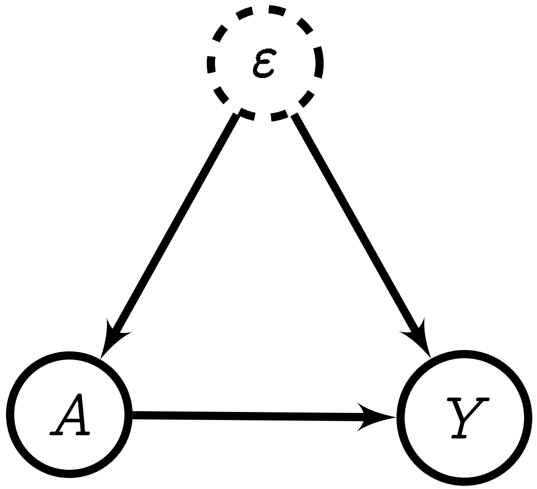
Proxy/Negative Control Methods

The proxy correction

Unobserved ε with (possibly) complex nonlinear effects on A, Y

The definitions are:

- ε: unobserved confounder.
- A: treatment
- Y: outcome



If ε were observed (which it isn't),

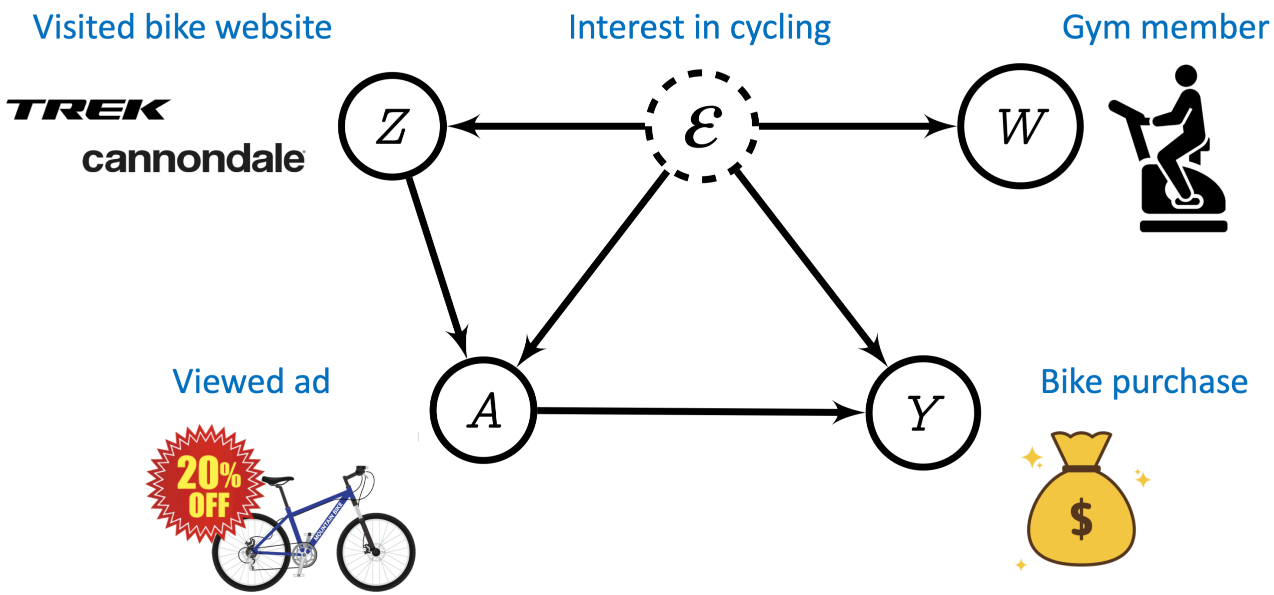
$$\mathbb{E}[Y^{(a)}] = \int \mathbb{E}[Y|\varepsilon, a]dp(\varepsilon)$$

The proxy correction

Unobserved ε with (possibly) complex nonlinear effects on A, Y

The definitions are:

- ε: unobserved confounder.
- A: treatment
- Y: outcome
- Z: treatment proxy
- W outcome proxy



Miao, Geng, Tchetgen Tchetgen (2018): Identifying causal effects with proxy variables of an unmeasured confounder.

Tennenholtz, Mannor, Shalit (2020), OPE in Partially Observed Environments.

Uehara, Sekhari, Lee, Kallus, Sun (2022) Provably Efficient Reinforcement Learning in Partially Observable Dynamical Systems.

Unobserved confounders: proxy methods

Kernel features (ICML 2021):

arXiv.org > cs > arXiv:2105.04544

Computer Science > Machine Learning

[Submitted on 10 May 2021 (v1), last revised 9 Oct 2021 (this version, v4)]

Proximal Causal Learning with Kernels: Two-Stage Estimation and Moment Restriction

Afsaneh Mastouri, Yuchen Zhu, Limor Gultchin, Anna Korba, Ricardo Silva, Matt J. Kusner, Arthur Gretton, Krikamol Muandet







NN features (NeurIPS 2021):

arXiv.org > cs > arXiv:2106.03907

Computer Science > Machine Learning

[Submitted on 7 Jun 2021 (v1), last revised 7 Dec 2021 (this version, v2)]

Deep Proxy Causal Learning and its Application to Confounded Bandit Policy Evaluation

Liyuan Xu, Heishiro Kanagawa, Arthur Gretton


Code for NN and kernel proxy methods:

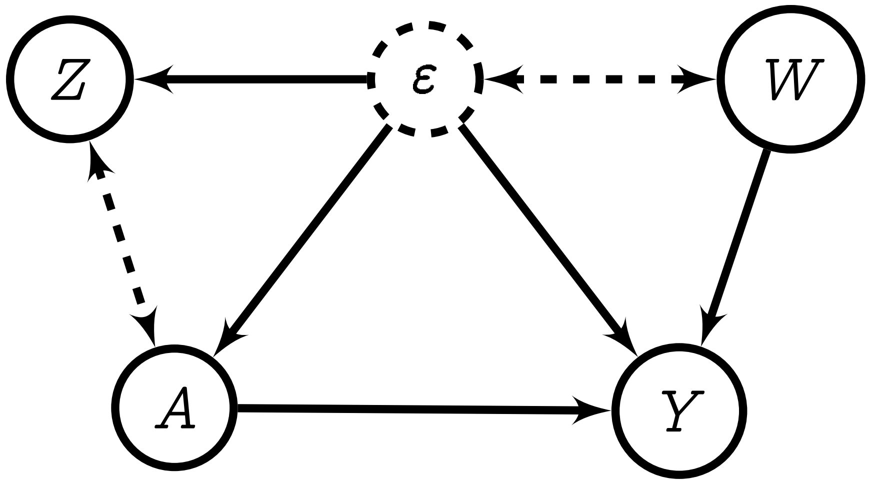
https://github.com/liyuan9988/DeepFeatureProxyVariable/

The proxy correction

Unobserved ε with (possibly) complex nonlinear effects on A, Y

The definitions are:

- ε: unobserved confounder.
- A: treatment
- Y: outcome
- Z: treatment proxy
- W outcome proxy



Structural assumption:

$$W \perp\!\!\!\perp (Z, A) | \varepsilon$$

$$Y \perp\!\!\!\perp Z | (A, \varepsilon)$$

$\implies$ Can recover $E(Y^{(a)})$ from observational data!

Main theorem

If ε were observed, we would write (average treatment effect)

$$p(y|do(a)) = \int_u p(y|a,\varepsilon)p(\varepsilon)d\varepsilon.$$

....but we do not observe ε.

Main theorem

If ε were observed, we would write (average treatment effect)

$$p(y|do(a)) = \int_u p(y|a,\varepsilon)p(\varepsilon)d\varepsilon.$$

....but we do not observe ε.

Main theorem: Assume we solved:

$$p(y|a,z) = \int h_y(w,a)p(w|a,z)dw$$

Both $p(y|a,z)$ and $p(w|a,z)$ are in terms of observed quantities.

Main theorem

If ε were observed, we would write (average treatment effect)

$$p(y|do(a)) = \int_u p(y|a,\varepsilon)p(\varepsilon)d\varepsilon.$$

....but we do not observe ε.

Main theorem: Assume we solved:

$$p(y|a,z) = \int h_y(w,a)p(w|a,z)dw$$

Both $p(y|a,z)$ and $p(w|a,z)$ are in terms of observed quantities.

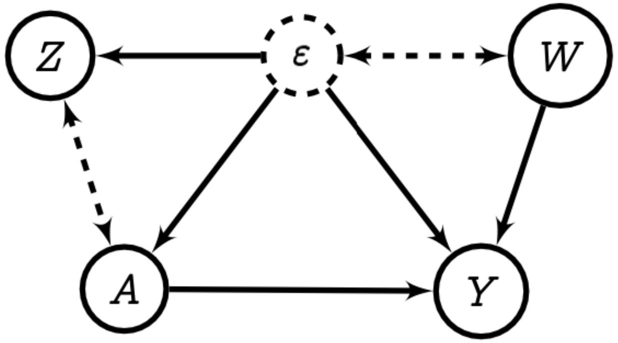
Average treatment effect via $p(w)$:

$$p(y|do(a)) = \int h_y(a,w)p(w)dw$$

Proof (1)

Because $W \perp\!\!\!\perp (Z, A) | \varepsilon$, we have

$$p(w|a,z) = \int p(w|\varepsilon)p(\varepsilon|a,z)d\varepsilon$$



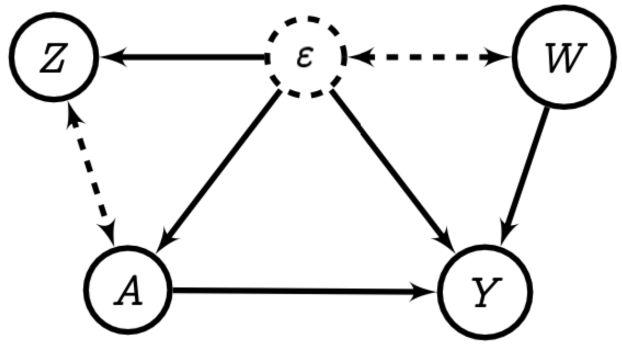
Proof (1)

Because $W \perp\!\!\!\perp (Z, A)|\varepsilon$, we have

$$p(w|a,z) = \int p(w|\varepsilon)p(\varepsilon|a,z)d\varepsilon$$

Because $Y \perp\!\!\!\perp Z|(A,\varepsilon)$ we have

$$p(y|a,z) = \int p(y|a,\varepsilon)p(\varepsilon|a,z)d\varepsilon$$



Proof (3)

Given the solution h_y to:

$$p(y|a,z) = \int h_y(w,a)p(w|a,z)dw$$

(well defined under identifiability conditions for Fredholm equation of first kind)

Proof (3)

Given the solution h_y to:

$$p(y|a,z) = \int h_y(w,a)p(w|a,z)dw$$

(well defined under identifiability conditions for Fredholm equation of first kind)

From last slide

$$\int p(y|a,\varepsilon)p(\varepsilon|a,z)d\varepsilon = \int h_y(w,a)\int p(w|\varepsilon)p(\varepsilon|a,z)d\varepsilon dw$$

Proof (3)

Given the solution h_y to:

$$p(y|a,z) = \int h_y(w,a)p(w|a,z)dw$$

(well defined under identifiability conditions for Fredholm equation of first kind)

From last slide

$$\int p(y|a,\varepsilon)p(\varepsilon|a,z)d\varepsilon = \int h_y(w,a)\int p(w|\varepsilon)p(\varepsilon|a,z)d\varepsilon dw$$

This implies:

$$p(y|a,\varepsilon) = \int h_y(w,a)p(w|\varepsilon)dw$$

under identifiability condition

$$\mathbb{E}[f(\varepsilon)|A=a, Z=z] = 0,\ \forall(z,a) \iff f(\varepsilon) = 0,\ \mathbb{P}_{\varepsilon|A=a}\text{ a.s.} \quad (\triangle)$$

Proof (4)

From last slide,

$$p(y|a, \varepsilon) = \int h_y(w, a) p(w|\varepsilon) dw$$

Thus

$$p(y|do(a)) = \int_u p(y|a, \varepsilon) p(\varepsilon) du$$

Proof (4)

From last slide,

$$p(y|a,\varepsilon) = \int h_y(w,a)p(w|\varepsilon)dw$$

Thus

$$\begin{aligned} p(y|do(a)) &= \int_u p(y|a,\varepsilon)p(\varepsilon)du \\ &= \int_u \left[\int h_y(w,a)p(w|\varepsilon)dw \right] p(\varepsilon)d\varepsilon \end{aligned}$$

Proof (4)

From last slide,

$$p(y|a, \varepsilon) = \int h_y(w, a) p(w|\varepsilon) dw$$

Thus

$$\begin{aligned} p(y|do(a)) &= \int_u p(y|a, \varepsilon) p(\varepsilon) du \\ &= \int_u \left[\int h_y(w, a) p(w|\varepsilon) dw \right] p(\varepsilon) d\varepsilon \\ &= \int h_y(w, a) p(w) dw \end{aligned}$$

Feature implementation

Stage 2: minimize

$$h_{\lambda_2} = \arg\min_{h \in \mathcal{H}} \mathbb{E}_{y,a,z} \left(y - h^\top \left(\mu_{W|a,z} \otimes \phi(a) \right) \right)^2 + \lambda_2 \|h\|^2_{\mathcal{H}}$$

which is conditional feature mean implementation of

$$p(y|a,z) = \int h_y(w,a) p(w|a,z) dw$$

Deaner (2021).
Mastouri, Zhu, Gultchin, Korba, Silva, Kusner, G., Muandet (2021).
Xu, Kanagawa, G. (2021).

Feature implementation

Stage 2: minimize

$$h_{\lambda_2} = \arg\min_{h \in \mathcal{H}} \mathbb{E}_{y,a,z} \left(y - h^\top \left(\mu_{W|a,z} \otimes \phi(a) \right) \right)^2 + \lambda_2 \|h\|^2_{\mathcal{H}}$$

which is conditional feature mean implementation of

$$p(y|a,z) = \int h_y(w,a) p(w|a,z) dw$$

Stage 1: ridge regression

$$F_{\lambda_1} = \arg\min_{F \in HS} \mathbb{E}_{w,a,z} \left\| \phi(w) - F^\top [\phi(a) \otimes \phi(z)] \right\|^2_{\mathcal{H}_\mathcal{W}} + \lambda_1 \|F\|^2_{HS}$$

which gives us

$$\mu_{W|a,z} = F_{\lambda_1}{}^\top [\phi(a) \otimes \phi(z)]$$

Deaner (2021).
Mastouri, Zhu, Gultchin, Korba, Silva, Kusner, G., Muandet (2021).
Xu, Kanagawa, G. (2021).

Feature implementation

Stage 2: minimize

$$h_{\lambda_2} = \arg\min_{h \in \mathcal{H}} \mathbb{E}_{y,a,z} \left(y - h^\top \left(\mu_{W|a,z} \otimes \phi(a) \right) \right)^2 + \lambda_2 \|h\|_{\mathcal{H}}^2$$

which is conditional feature mean implementation of

$$p(y|a,z) = \int h_y(w,a) p(w|a,z) dw$$

Stage 1: ridge regression

$$F_{\lambda_1} = \arg\min_{F \in HS} \mathbb{E}_{w,a,z} \left\| \phi(w) - F^\top [\phi(a) \otimes \phi(z)] \right\|_{\mathcal{H}_\mathcal{W}}^2 + \lambda_1 \|F\|_{HS}^2$$

which gives us

$$\mu_{W|a,z} = F_{\lambda_1}{}^\top [\phi(a) \otimes \phi(z)]$$

Average treatment effect estimate:

$$\mathbb{E}_y(y|do(a)) = h_{\lambda_2}{}^\top \left(\phi(a) \otimes \mu_W \right),$$

where $\mu_W = \mathbb{E}_W \phi(W)$

Deaner (2021).

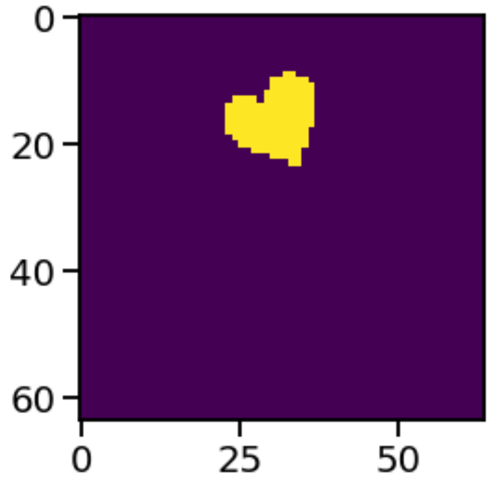
Mastouri, Zhu, Gultchin, Korba, Silva, Kusner, G., Muandet (2021).

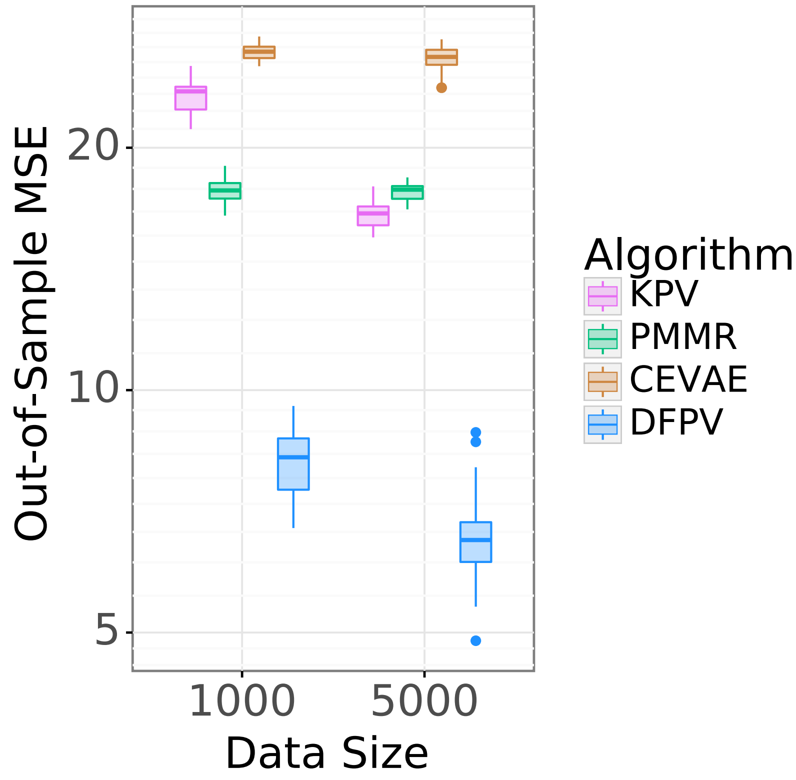
Xu, Kanagawa, G. (2021).

Synthetic experiment, adaptive neural net features

dSprite example:

- $\varepsilon = \{\texttt{scale}, \texttt{rotation}, \texttt{posX}, \texttt{posY}\}$
- Treatment A is the image generated (with Gaussian noise)
- Outcome Y is quadratic function of A with multiplicative confounding by $\texttt{posY}$.
- $Z = \{\texttt{scale}, \texttt{rotation}, \texttt{posX}\}$,
  W = noisy image sharing $\texttt{posY}$
- Comparison with CEVAE (Louzios et al. 2017)



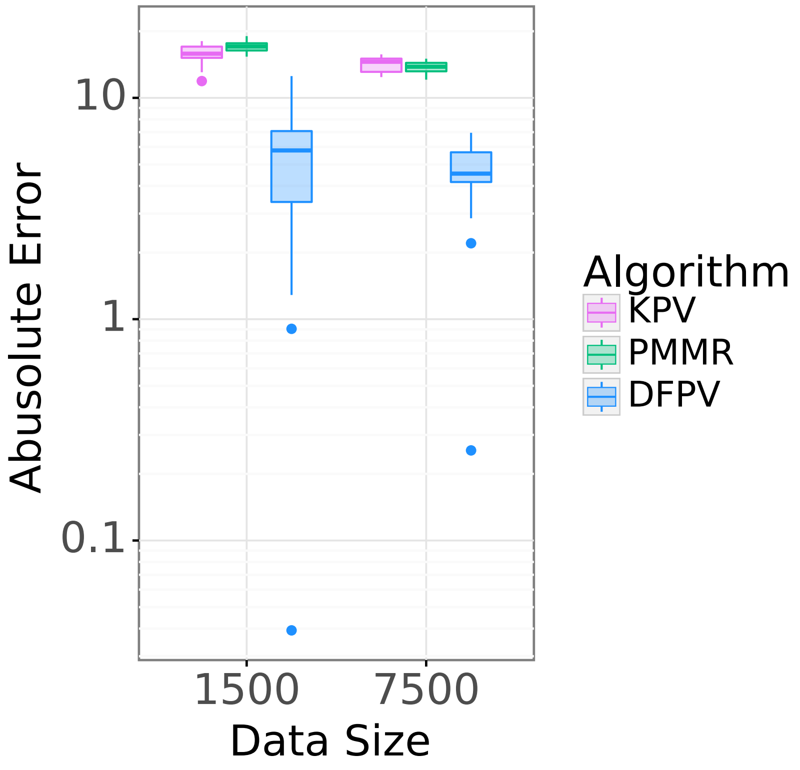


Louizos, Shalit, Mooij, Sontag, Zemel, Welling, Causal Effect Inference with Deep Latent-Variable Models (2017)

Confounded offline policy evaluation

Synthetic dataset, demand prediction for flight purchase.

- Treatment A is ticket price.
- Policy $A \sim \pi(Z)$ depends on fuel price.



Conclusions

Neural net and kernel solutions:

- ...for instrumental variable regression
- ...for proxy methods
- ...with treatment A, covariates X, V, proxies (W, Z) multivariate, "complicated"
- Convergence guarantees for kernels and NN

Code available for all methods

Research support

Work supported by:

The Gatsby Charitable Foundation

Deepmind

Questions?


THE KERNEL
BREWERY LONDON
BALTIC PORTER
7.3% ABV
THE KERNEL
EXPORT STOUT

IV regression using neural net features

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 \|\gamma\|^2$$

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2$$

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}||\phi_\theta(A) - F\phi_\zeta(Z)||^2 + \lambda_1 ||F||^2_{HS}$$

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 \|\gamma\|^2$$

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}\|\phi_\theta(A) - F\phi_\zeta(Z)\|^2 + \lambda_1 \|F\|_{HS}^2$$

Challenge: how to learn θ?

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[(Y-\gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right]+\lambda_2\|\gamma\|^2$$

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z]\approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}\|\phi_\theta(A)-F\phi_\zeta(Z)\|^2+\lambda_1\|F\|_{HS}^2$$

Challenge: how to learn θ?

From Stage 2 regression?

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2$$

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}||\phi_\theta(A) - F\phi_\zeta(Z)||^2 + \lambda_1 ||F||^2_{HS}$$

Challenge: how to learn θ?

From Stage 2 regression?

...which requires $\mathbb{E}[\phi_\theta(A)|Z]$ from Stage 1 regression

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2$$

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}||\phi_\theta(A) - F\phi_\zeta(Z)||^2 + \lambda_1 ||F||^2_{HS}$$

Challenge: how to learn θ?

From Stage 2 regression?

...which requires $\mathbb{E}[\phi_\theta(A)|Z]$ from Stage 1 regression

...which requires $\phi_\theta(A)$... which requires θ...

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2$$

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}||\phi_\theta(A) - F\phi_\zeta(Z)||^2 + \lambda_1 ||F||^2_{HS}$$

Challenge: how to learn θ?

From Stage 2 regression?

...which requires $\mathbb{E}[\phi_\theta(A)|Z]$ from Stage 1 regression

...which requires $\phi_\theta(A)$... which requires θ...

Use the linear final layers! (i.e. γ and F)

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_{\theta}(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}\left[||\phi_{\theta}(A) - F\phi_\zeta(Z)||^2\right] + \lambda_1||F||^2_{HS}$$

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}\left[||\phi_\theta(A) - F\phi_\zeta(Z)||^2\right] + \lambda_1||F||_{HS}^2$$

$\hat{F}_{\theta,\zeta}$ in closed form wrt ϕ_θ, ϕ_ζ:

$$\hat{F}_{\theta,\zeta} = C_{AZ}(C_{ZZ} + \lambda_1 I)^{-1} \qquad \begin{aligned} C_{AZ} &= \mathbb{E}[\phi_\theta(A)\phi_\zeta^\top(Z)] \\ C_{ZZ} &= \mathbb{E}[\phi_\zeta(Z)\phi_\zeta^\top(Z)] \end{aligned}$$

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

IV using neural net features

Stage 1 regression: learn NN features $\phi_\zeta(Z)$ and linear layer F:

$$\mathbb{E}[\phi_\theta(A)|Z] \approx F\phi_\zeta(Z)$$

with RR loss

$$\mathbb{E}\left[||\phi_\theta(A) - F\phi_\zeta(Z)||^2\right] + \lambda_1||F||^2_{HS}$$

$\hat{F}_{\theta,\zeta}$ in closed form wrt ϕ_θ, ϕ_ζ:

$$\hat{F}_{\theta,\zeta} = C_{AZ}(C_{ZZ} + \lambda_1 I)^{-1} \qquad C_{AZ} = \mathbb{E}[\phi_\theta(A)\phi_\zeta^\top(Z)]$$

$$C_{ZZ} = \mathbb{E}[\phi_\zeta(Z)\phi_\zeta^\top(Z)]$$

Plug $\hat{F}_{\theta,\zeta}$ into S1 loss, take gradient steps for ζ (...but not θ...)

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021) Learning Deep Features in Instrumental Variable Regresion

Stage 2: IV regression

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\mathcal{L}_2(\gamma, \theta) = \mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2$$

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021)

Stage 2: IV regression

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\begin{aligned}\mathcal{L}_2(\gamma,\theta) &= \mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2 \\ &= \mathbb{E}_{YZ}[(Y - \gamma^\top \underbrace{\hat{F}_{\theta,\zeta}\phi_\zeta(Z))}_{\text{Stage 1}}{}^2] + \lambda_2 ||\gamma||^2\end{aligned}$$

Stage 2: IV regression

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$
\begin{aligned}
\mathcal{L}_2(\gamma, \theta) &= \mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2 \\
&= \mathbb{E}_{YZ}[(Y - \gamma^\top \hat{F}_{\theta,\zeta}\phi_\zeta(Z))^2] + \lambda_2 ||\gamma||^2
\end{aligned}
$$

$\hat{\gamma}_\theta$ in closed form wrt ϕ_θ:

$$
\hat{\gamma}_\theta := \widetilde{C}_{YA|Z}(\widetilde{C}_{AA|Z} + \lambda_2 I)^{-1} \qquad \widetilde{C}_{YA|Z} = \mathbb{E}\left[Y\,[\hat{F}_{\theta,\zeta}\phi_\zeta(Z)]^\top\right]
$$

$$
\widetilde{C}_{AA|Z} = \mathbb{E}\left[[\hat{F}_{\theta,\zeta}\phi_\zeta(Z)]\,[\hat{F}_{\theta,\zeta}\phi_\zeta(Z)]^\top\right]
$$

Stage 2: IV regression

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\begin{aligned}\mathcal{L}_2(\gamma, \theta) &= \mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2 \\ &= \mathbb{E}_{YZ}[(Y - \gamma^\top \hat{F}_{\theta,\zeta}\phi_\zeta(Z))^2] + \lambda_2 ||\gamma||^2\end{aligned}$$

$\hat{\gamma}_\theta$ in closed form wrt ϕ_θ:

$$\hat{\gamma}_\theta := \widetilde{C}_{YA|Z}(\widetilde{C}_{AA|Z} + \lambda_2 I)^{-1} \qquad \widetilde{C}_{YA|Z} = \mathbb{E}\left[Y\ [\hat{F}_{\theta,\zeta}\phi_\zeta(Z)]^\top\right]$$

$$\widetilde{C}_{AA|Z} = \mathbb{E}\left[[\hat{F}_{\theta,\zeta}\phi_\zeta(Z)]\ [\hat{F}_{\theta,\zeta}\phi_\zeta(Z)]^\top\right]$$

From linear final layers in Stages 1,2:

Learn $\phi_\theta(A)$ by plugging $\hat{\gamma}_\theta$ into S2 loss, taking gradient steps for θ

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021)

Stage 2: IV regression

Stage 2 regression (IV): learn NN features $\phi_\theta(A)$ and linear layer γ to obtain Y with RR loss:

$$\begin{aligned}\mathcal{L}_2(\gamma,\theta) &= \mathbb{E}_{YZ}\left[(Y - \gamma^\top \mathbb{E}[\phi_\theta(A)|Z])^2\right] + \lambda_2 ||\gamma||^2 \\ &= \mathbb{E}_{YZ}[(Y - \gamma^\top \hat{F}_{\theta,\zeta}\phi_\zeta(Z))^2] + \lambda_2||\gamma||^2\end{aligned}$$

$\hat{\gamma}_\theta$ in closed form wrt ϕ_θ:

$$\hat{\gamma}_\theta := \widetilde{C}_{YA|Z}(\widetilde{C}_{AA|Z} + \lambda_2 I)^{-1} \qquad \widetilde{C}_{YA|Z} = \mathbb{E}\left[Y\ [\hat{F}_{\theta,\zeta}\phi_\zeta(Z)]^\top\right]$$

$$\widetilde{C}_{AA|Z} = \mathbb{E}\left[[\hat{F}_{\theta,\zeta}\phi_\zeta(Z)]\ [\hat{F}_{\theta,\zeta}\phi_\zeta(Z)]^\top\right]$$

From linear final layers in Stages 1,2:

Learn $\phi_\theta(A)$ by plugging $\hat{\gamma}_\theta$ into S2 loss, taking gradient steps for θ

....but ζ changes with θ

...so alternate first and second stages until convergence.

Xu, Chen, Srinivasan, De Freitas, Doucet, G. (2021)