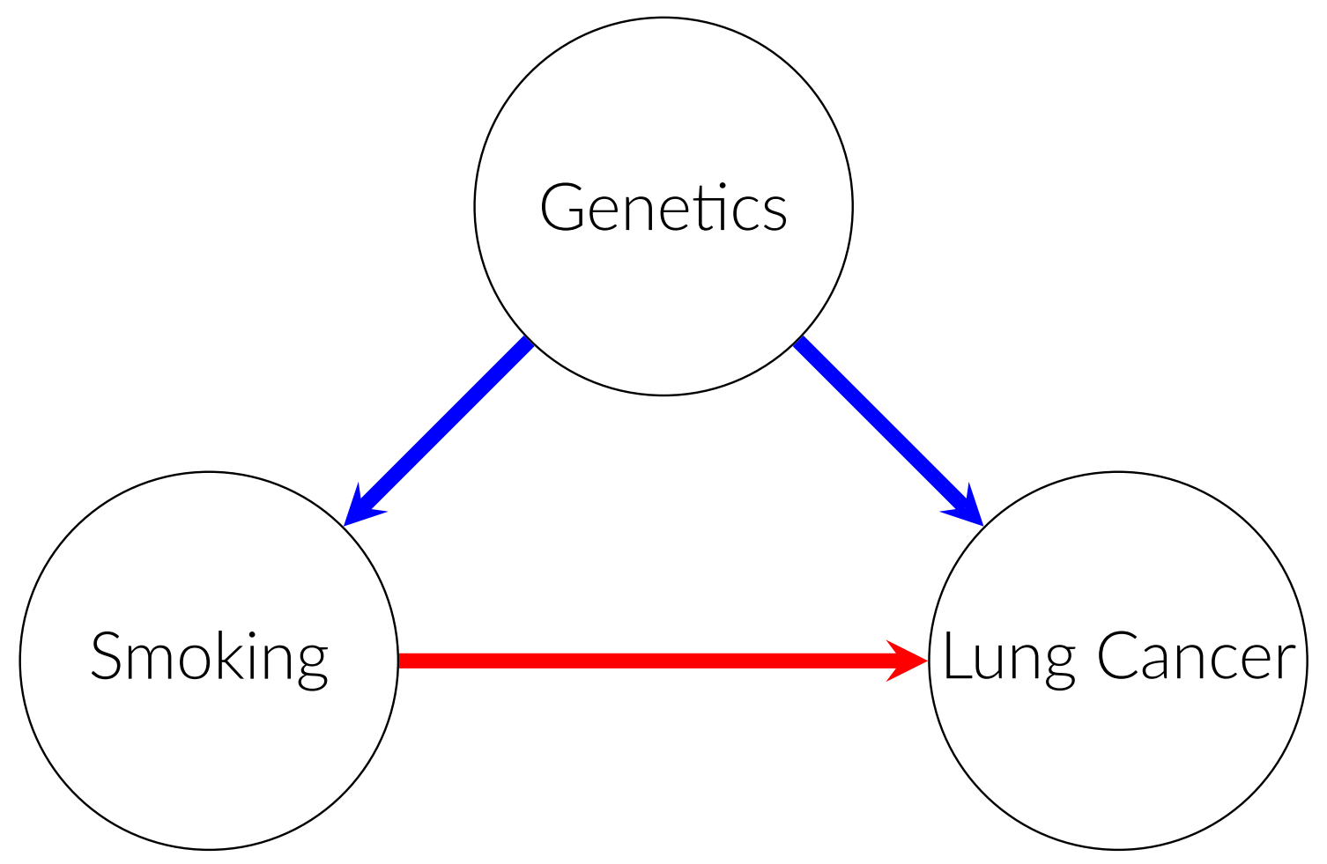
One Proxy Proximal Causal Learning With An $\varepsilon - \delta$ Private Algorithm

Chase Mathis [1] Alexander Volfovsky [1] Robin Evans [2]

[1]Duke University [2]University of Oxford

Introduction to Causal Inference

Causal inference aims to mathematize how one can make causal statements from data. Causality is often drawn from a well done randomized controlled trial (RCT). Sometimes RCTs are unfeasible and we must work with only observational data. With certain causal assumptions, we can estimate the causal effect with observational information alone. These assumptions are depicted in a graph like the one below.



Full Presentation

Figure 1. A Causal Directed Acyclic Graph (DAG). Is this red arrow real?

A common causal example is that of smoking and lung cancer. Perhaps people who get lung cancer and people who smoke share a genetic factor. Then there would be a positive correlation between smoking and lung cancer without there being any causal influence.

One method to get rid of confounding is to "control" for the confounder. When running a regression, this means including genetic information in your regression.

Proximal Causal Learning

We wish that we have access to the true confounder, but oftentimes we can only get noisy measurements. Some reasons may include:

- Privacy Issues People who take surveys do not want to divulge identifying information.
- Too Expensive It may be too expensive to collect the true gold-standard data.
- Technologically Impossible It is impossible to get the data we suspect is a confounder.

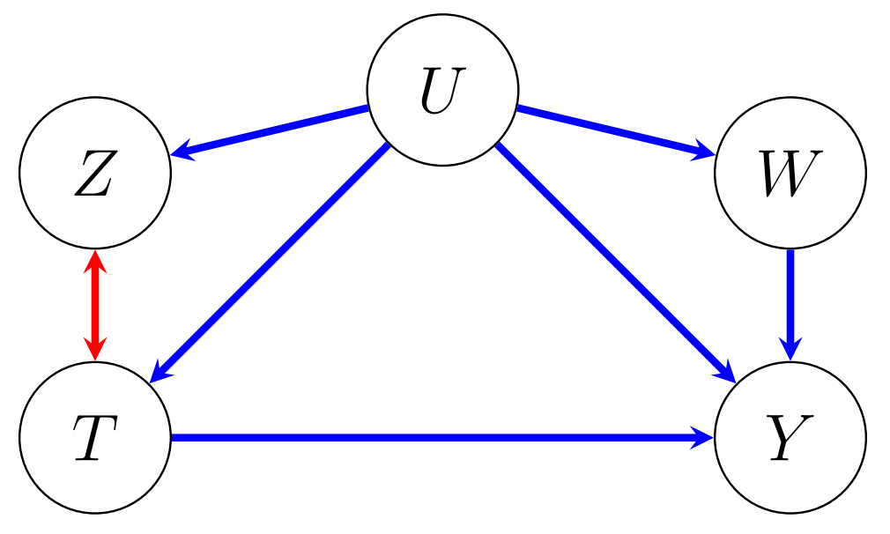


Figure 2. Allowed Causal DAG for Proximal Causal Learning

Recent work has shown that with two conditionally independent proxies, we can *"bypass"* using U to find the direct effect of T on Y [8].

Proximal Two Stage Least Squares (P2SLS)

Tchetgen Tchetgen dubs his method with proxies "Proximal Two Stage Least Squares (P2SLS)" after that of the more common Two Stage Least Squared estimator [1].

I provide a proof of the linear case as it illustrates some issues we deal with and our motivations.

Proof:

Assume $\mathbb{E}[Y \mid U, T, Z] = \beta_T * T + \beta_U * U + \beta_0$ and that $\mathbb{E}[W \mid U, T, Z] = \alpha_U * U + \alpha_0$ (this assumption is violated in our second problem).

Then, via the tower rule of expectation:

$$\mathbb{E}[Y \mid T, Z] = \beta_T * T + \beta_U * \mathbb{E}[U \mid T, Z] \quad \mathbb{E}[W \mid T, Z] = \alpha_U * \mathbb{E}[U \mid T, Z] + \alpha_0 \tag{1}$$

Noting that $\mathbb{E}[U \mid T, Z] \propto \mathbb{E}[W \mid T, Z]$, we estimate $\hat{W} = \mathbb{E}[W \mid T, Z]$ through linear regression and use $\hat{W}$ as our proxy control variable [5].

My contributions

1. Demonstrate a new Bayesian bootstrapping method that performs better than the naive regression often employed under an ordinal confounder.
2. Illustrate the importance of cross-fitting for the proximal two stage least squares.
3. Introduce an ε, δ differential privacy application of proximal causal learning.

Ordinal Confounders

Ordinal variables are ordered, but cannot be interpreted as having constant scale. For instance, education level is ordinal: College Degree > High School, but perhaps the difference between Graduate Degree and College Degree is less than the difference between College and High School Degree.

In various experiments we may not care about the individual binned data, but instead a hidden process that generates the bin. We may think education level is continuous: no same two people are equally knowledgeable. It is difficult, if not impossible, to measure this directly, so we try to approximate it by using the degree.

Graphically, let O be the binned ordinal data and U the hidden confounder we really want to measure. Then, a diagram of our assumptions can be seen below:

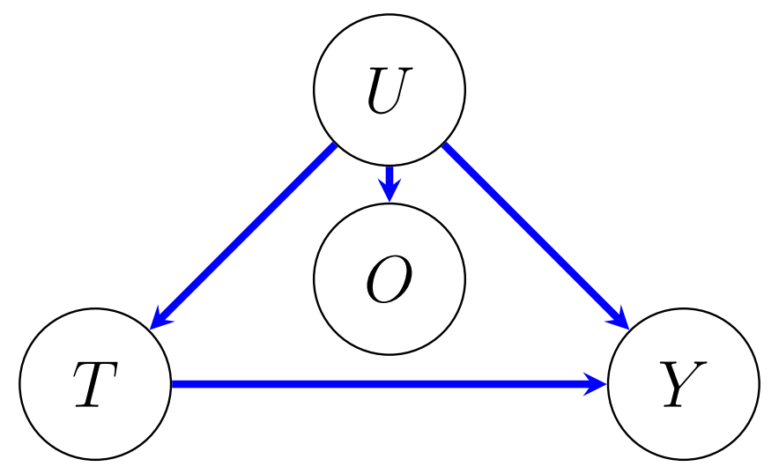


Figure 3. Assume ordinal proxy.

Main Result: Bayesian Bootstrapping Proxies

If U is normal, then we can use a probit regression to simulate values of U conditional on either T, or Y, or both using its posterior distribution. The posterior is a constrained normal distribution [see [4] for more].

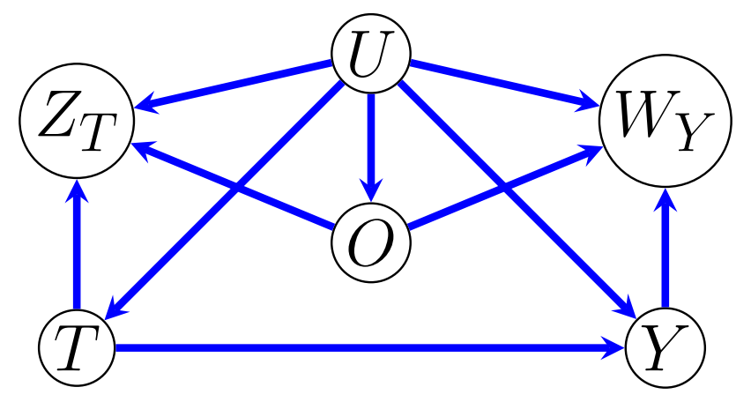


Figure 4. Bayesian Bootstrap Causal DAG. Here, Z_T is the simulations from T and W_Y is the simulations from Y.

Naturally T generates Z_T and because we are using O as our ordinal data O helps in generating Z_T. The remarkable thing is that empirically, $Z_T \not\perp\!\!\!\perp U \mid (T, O)$. We are getting *new* information about U that is not in O. (The same goes for W_Y.

Issues and our Proposed Solutions

Problem 1: The stability of the proximal two stage least squares estimator

Proximal two stage least squares lets $\mathbb{E}[W \mid T, Z]$ be our proximal confounder [8]. For simplicity, assume linearity:

$$\mathbb{E}[W \mid T, Z] = \alpha_T * T + \alpha_Z * Z \quad \text{and if } \alpha_T \gg \alpha_Z, \text{ then}$$
$$\mathbb{E}[Y \mid T, \mathbb{E}[W \mid T, Z]] \approx T + \alpha_T * T$$

This creates very unstable estimates for the parameter of interest.

Solution: Cross-fitting Cross-fitting de-correlates $\mathbb{E}[W \mid T, Z]$ with T so that even if $\alpha_T \gg \alpha_Z$, the estimates are ok. While it was popularized in double machine learning, we find that it holds another application here [2].

Problem 2: W_Y is a descendant of Y. This creates a biased estimate.

As seen in Figure 4, $Y \rightarrow W_Y$. Ultimately, this leads to a biased estimate.

Solution: Create a dummy proxy variable We wish to get rid of the direct effect of Y on W_Y. One way to do this is to create a noisy proxy of Y call it $Y_0 \equiv Y + \varepsilon : \varepsilon \sim N(0, 1)$. Then we can estimate the direct effect and remove it through the backdoor criterion [6].

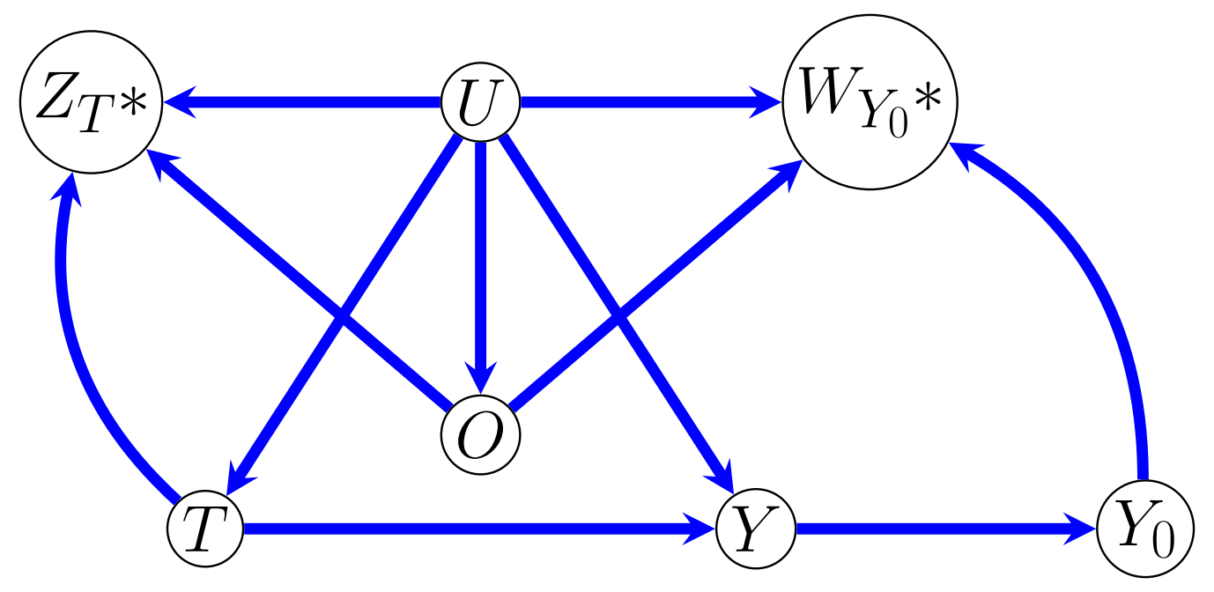


Figure 5. Dummy variable graph

Introducing an $\varepsilon - \delta$ Differential Privacy Algorithm

User privacy is becoming increasingly important as personal data is fuelling algorithms we use every day. One framework for privacy is ε, δ differential privacy.

Algorithms take in inputs (datasets) and generate outputs. Consider two datasets that differ with one entry (one row): X, X', such that $X \Delta X' = 1$. Then a random algorithm $\mathcal{A}$ that outputs$\mathcal{A}(X)$ is ε, δ differentially private if

$$\frac{\Pr\left(\mathcal{A}(X) \in S\right)}{\Pr\left(\mathcal{A}(X') \in S\right)} \leq \exp(\varepsilon) + \delta \quad \forall S \in \text{Range}(\mathcal{A})$$

Essentially, an algorithm is private if one cannot guess with high probability whether someone's data is used in the algorithm or not.

Consider a dataset (T, U, Y) and we wish to keep (U, Y) private, consider the ε, δ proximal causal learning algorithm that estimates an ATE.

1. Use the Laplacian mechanism to add noise to U and Y [3].
2. Use proximal learning algorithms and the two proxies of U to estimate the ATE.

Bayesian Bootstrapping Simulation Results

We sample edge coefficients $\in [-5, 5] \subset \mathbb{R}$ and dispersion parameters uniformly at random $\in [1, 10] \subset \mathbb{R}$. We make sure to sample fraction edge-coefficients so that we do not fall into the trap of varsortability [7].

Statistic	Cross-fit	P2SLS	Naive	Full Bootstrap		Known W		Oracle
				Fig 4	Fig 5	Bootstrap	Naive	
Mean	Yes	1.52	**1.14**	**0.69**	9.13	0.35	347.58	0.03
	No	39.76	**1.14**	**0.68**	43.56	0.10	0.79	0.03
Median	Yes	0.001	**0.12**	**0.05**	0.39	0.002	0.001	0.002
	No	0.001	**0.12**	**0.05**	0.39	0.001	0.001	0.002

Table 1. **Simulation Results:** Squared Error across many initial setups (sims = 60,000; n = 100).

- Our full bootstrap method performs better than the naive regression adjustment.
- Residualizing away the $Y \rightarrow W_Y$ effect is the worst estimator.
- Cross-fitting is *essential* to reduce the prevalence of high estimates in P2SLS.
- Even with cross-fitting, P2SLS estimates highly erroneous values.

Questions and Future Work

- How does this method work asymptotically?
- Since Y is a parent of W_Y, the estimate is biased. Why is the residual result much worse?
- How does this method work with other statistical families?
- Can we create an ε, δ algorithm with hidden treatments?

References

[1] Joshua D. Angrist and Guido W. Imbens.
Two-stage least squares estimation of average causal effects in models with variable treatment intensity.
90(430):431–442.

[2] Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins.
Double/debiased machine learning for treatment and causal parameters.

[3] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith.
Calibrating noise to sensitivity in private data analysis.
In Shai Halevi and Tal Rabin, editors, *Theory of Cryptography*, pages 265–284. Springer.

[4] Peter D. Hoff.
Latent variable methods for ordinal data.
In Peter D. Hoff, editor, *A First Course in Bayesian Statistical Methods*, pages 209–223. Springer.

[5] Jiewen Liu, Chan Park, Kendrick Li, and Eric J. Tchetgen Tchetgen.
Regression-based proximal causal inference.

[6] Judea Pearl.
[bayesian analysis in expert systems]: Comment: Graphical models, causality and intervention.
8(3):266–269.
Publisher: Institute of Mathematical Statistics.

[7] Alexander Reisach, Christof Seiler, and Sebastian Weichwald.
Beware of the simulated DAG! causal discovery benchmarks may be easy to game.
In *Advances in Neural Information Processing Systems*, volume 34, pages 27772–27784. Curran Associates, Inc.

[8] Eric J. Tchetgen Tchetgen, Andrew Ying, Yifan Cui, Xu Shi, and Wang Miao.
An introduction to proximal causal learning.