A class of algorithms for general instrumental variable models

https://arxiv.org/abs/2006.06366

joint work with
Matt Kusner & Ricardo Silva

Niki Kilbertus
HELMHOLTZAI

looking for PhD students
and PostDocs at
Helmholtz AI and TU Munich!

Motivation

Let’s start with a classic

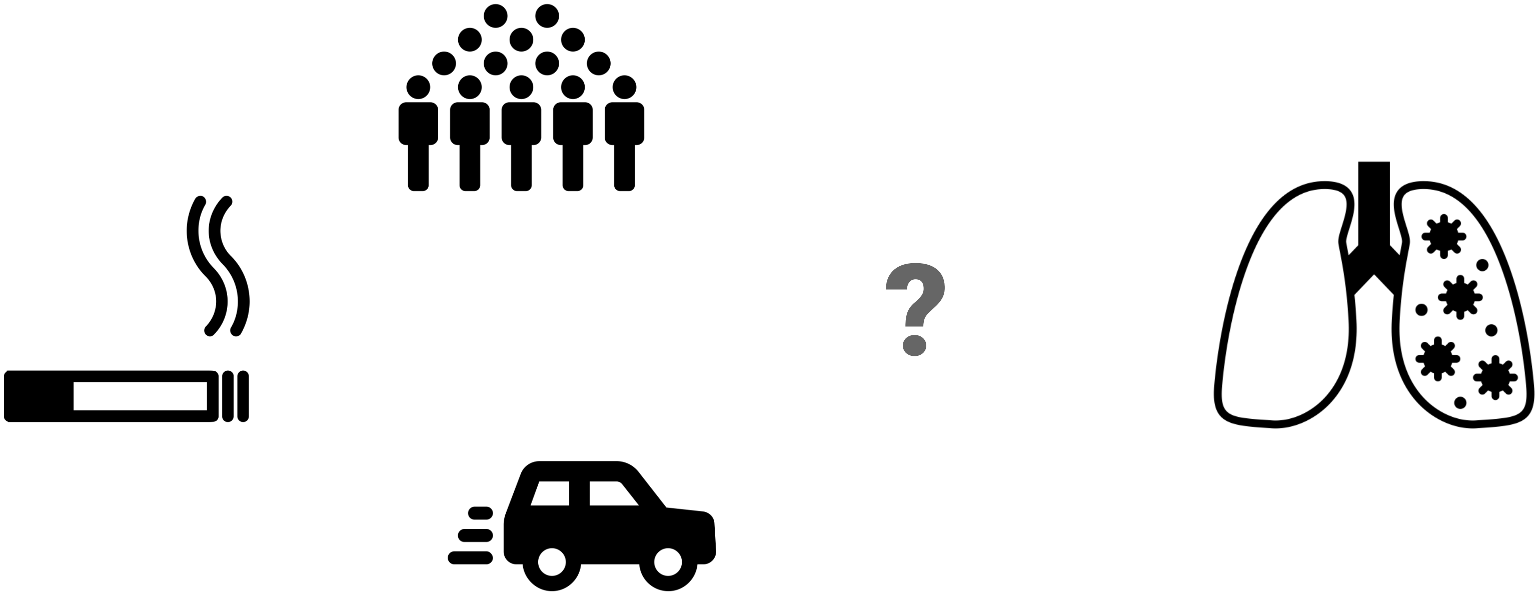
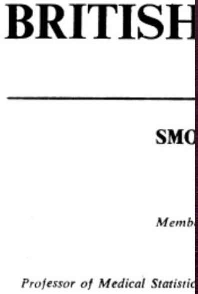


Image credit (from Noun Project):
Andrew Nielsen, mungang kim, Adrien Coquet

There was “a lot of correlation”

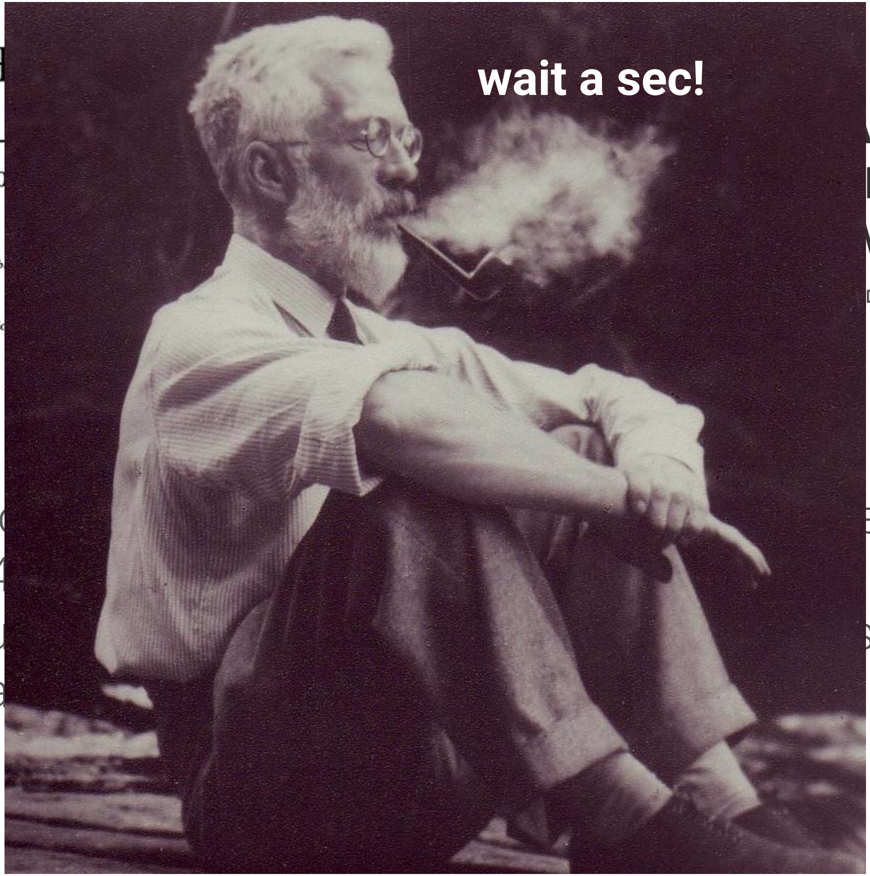


BRITISH

SMO

Memb

Professor of Medical Statistic

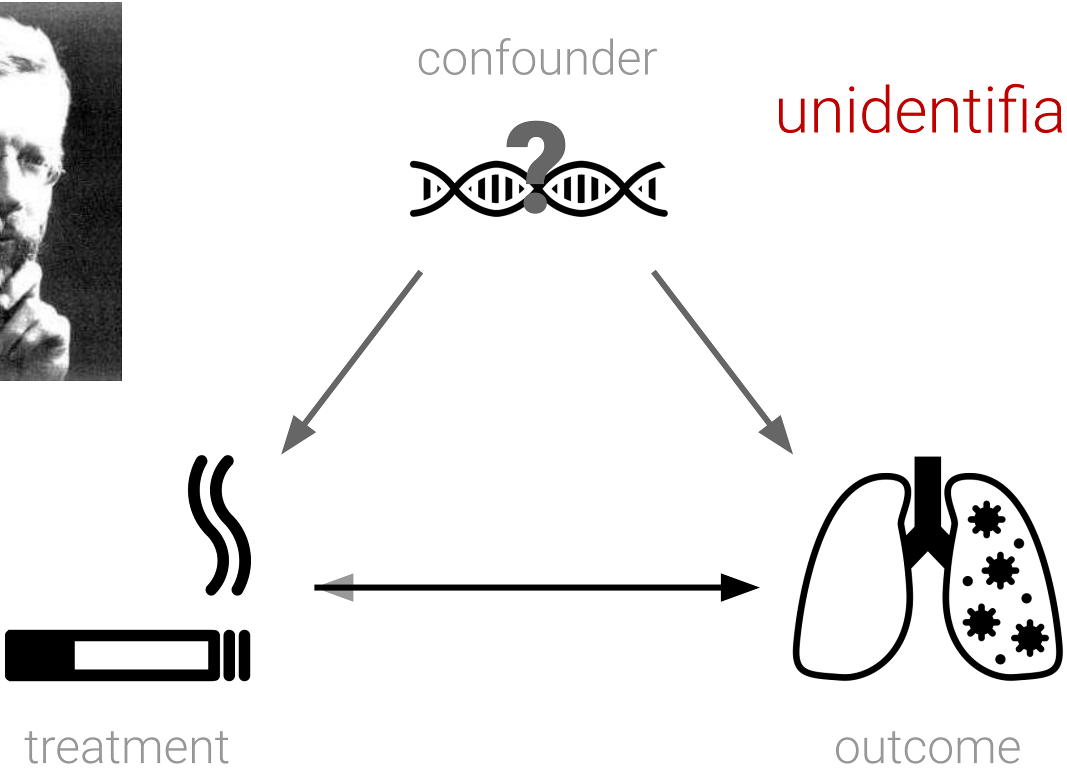


TIONSHIP BETWEEN HUMAN SMOKING
ND DEATH RATES
V-UP STUDY OF 187,766 MEN

D.; Daniel Horn, Ph.D.

- 3 ers, 56 were heavy smokers
- 14 . 23.9% other cancer patients
- su st (all 36 who died of lung ca
- m

Unobserved confounding



confounder

treatment

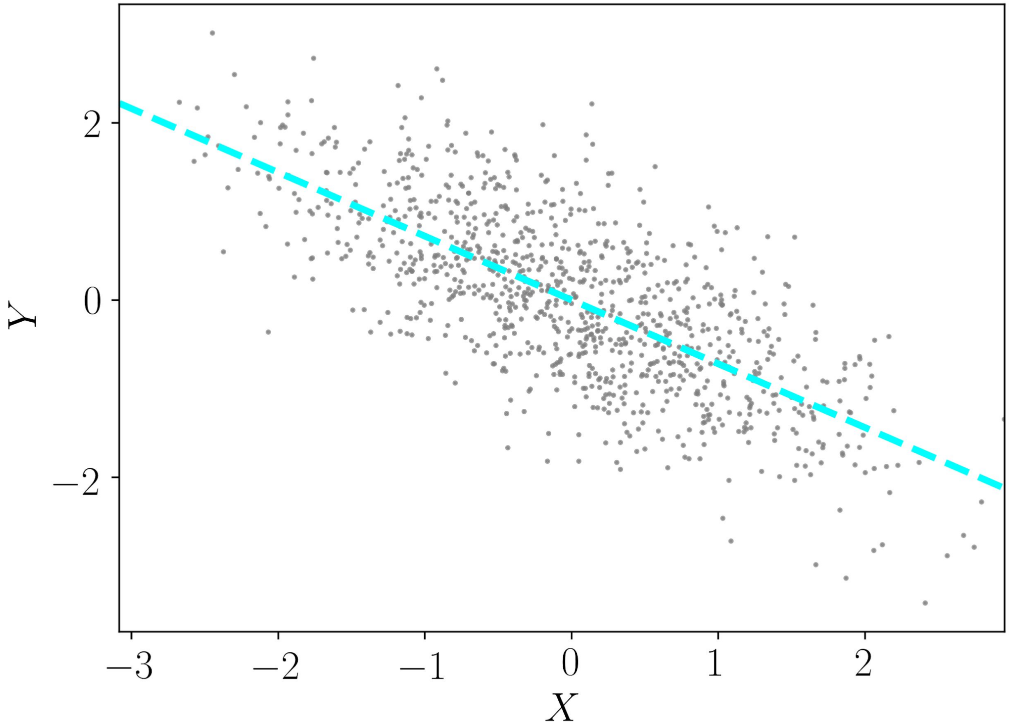
outcome

unidentifiability

Image credit (from Noun Project):
priyanka, Andrew Nielsen, mungang kim

Introduction

Naive ML approach: standard regression



$$X \in \mathbb{R},\ Y \in \mathbb{R}$$

$$\boxed{Y = f(X) + e_Y}$$

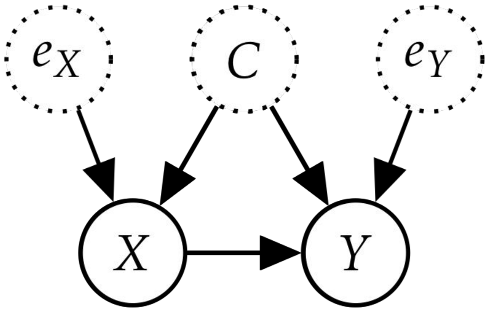
$$\boxed{\mathbb{E}[e_Y \mid X] = 0}$$

$$\mathbb{E}[Y - f(X) \mid X] = 0$$

$$\Rightarrow \mathbb{E}[Y \mid X] = f(X)$$

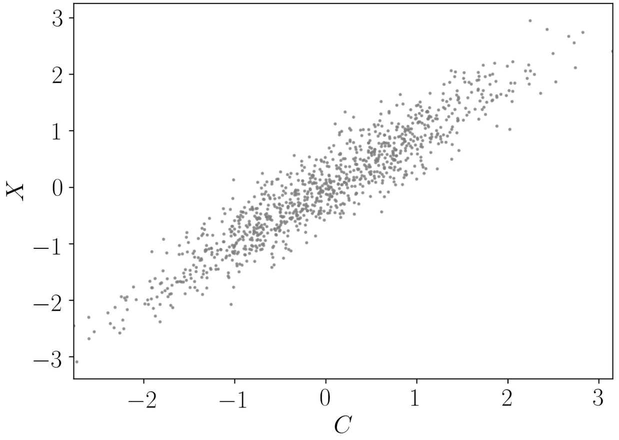
linear least squares: $f = \arg\min_{\hat{f}} \sum_i \left(\hat{f}(x_i) - y_i\right)^2$

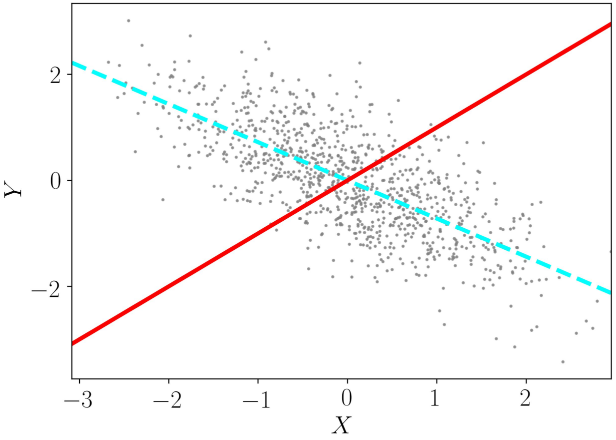
Naive ML approach: standard regression



$$X = \alpha \cdot C + e_X$$

$$Y = \boxed{X} + \boxed{\beta \cdot C + e_Y}$$

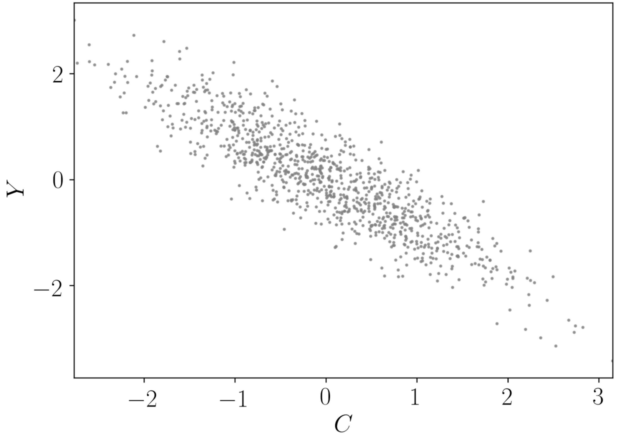




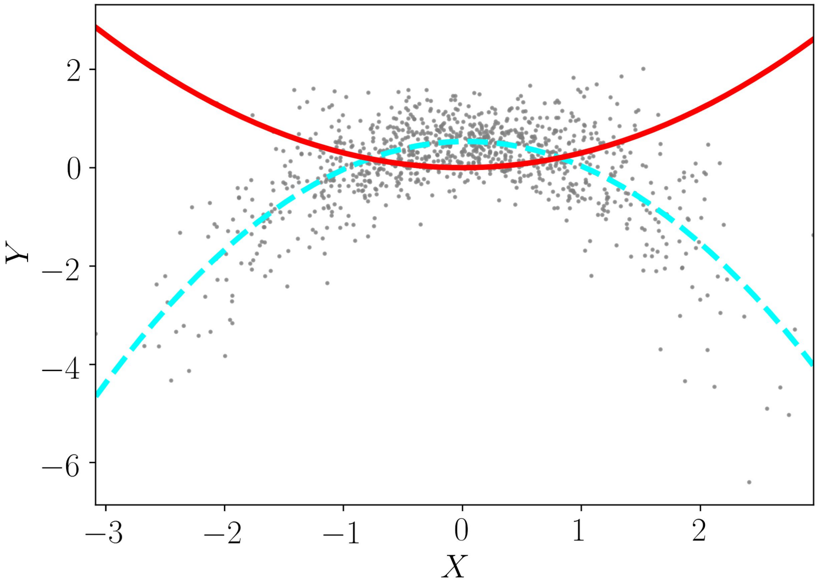
$$Y = \boxed{f(X)} + \boxed{e_Y}$$

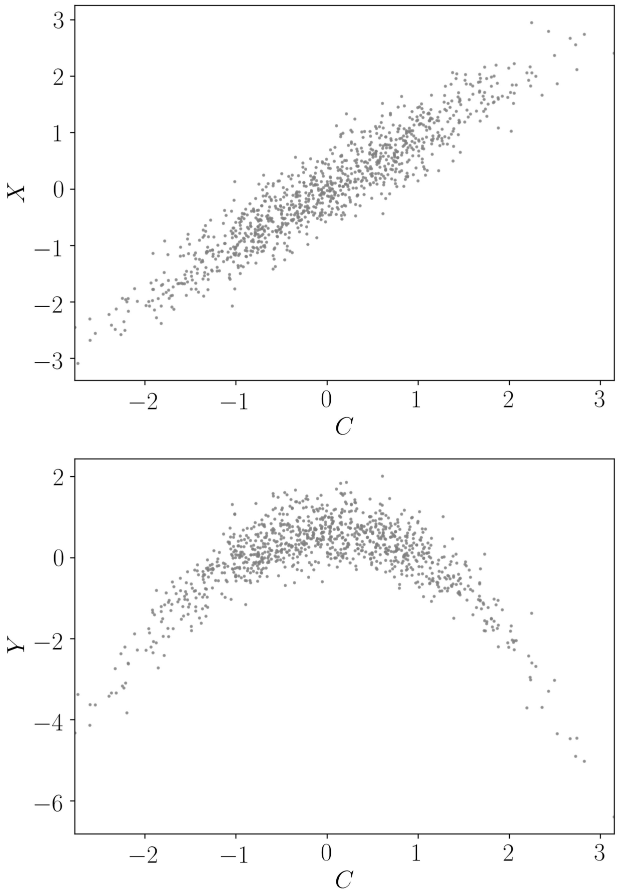
$$\cancel{\mathbb{E}[e_Y | X] = 0}$$

$$\boxed{\mathbb{E}[Y | do(X)]}$$

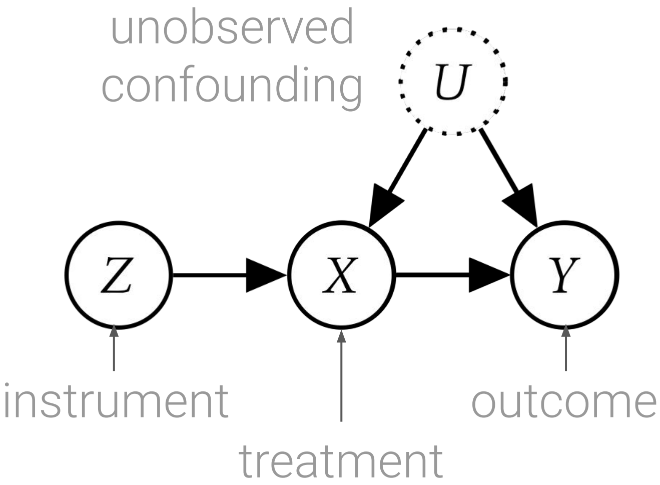


Losing hope...





Instrumental variables



(a) Z influences X $\quad Z \not\perp\!\!\!\perp X$

(b) Z is independent of U $\quad Z \perp\!\!\!\perp U$

(c) Z only influences Y via X $\quad Z \perp\!\!\!\perp Y \,|\, \{X, U\}$

assume: $Y = f(X) + e_Y \quad \textbf{with} \quad \mathbb{E}[e_Y] = 0$

$$\boxed{\mathbb{E}[Y\,|\,z]} = \mathbb{E}[f(X) + e_Y\,|\,z] = \mathbb{E}[f(X)\,|\,z] = \int \boxed{f(x)}\,\boxed{p(x\,|\,z)}\,dx$$

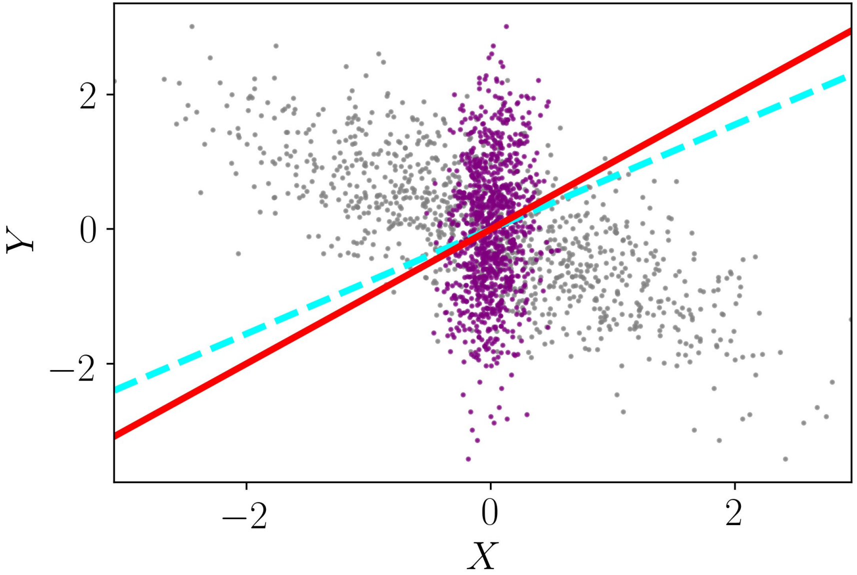
identifiable

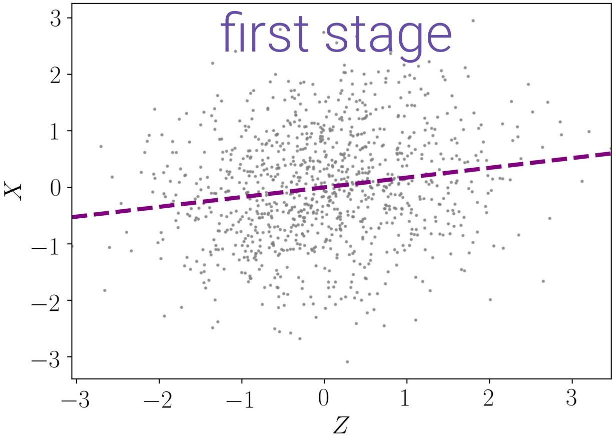
unique under
mild conditions

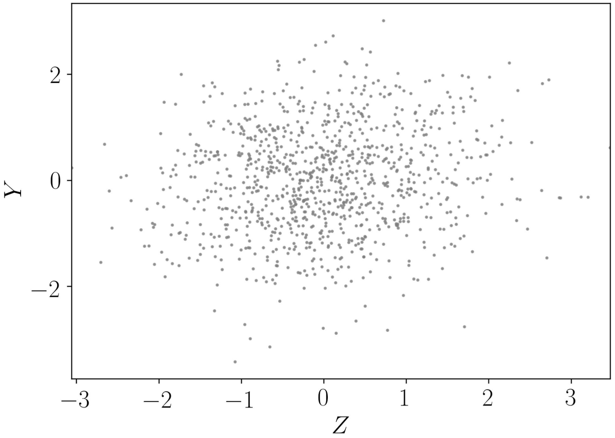
identifiable

Two stage least squares (2SLS) -- linear case

second stage

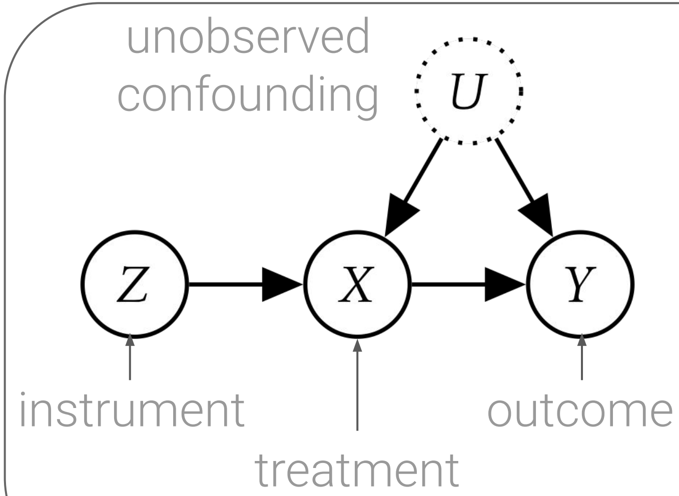






Problem formulation

General problem formulation



Assumptions

(a) Z influences X $Z \not\perp\!\!\!\perp X$

(b) Z is independent of U $Z \perp\!\!\!\perp U$

(c) Z only influences Y via X $Z \perp\!\!\!\perp Y \mid \{X, U\}$

$$X = g(Z, U) \qquad Y = f(X, U)$$

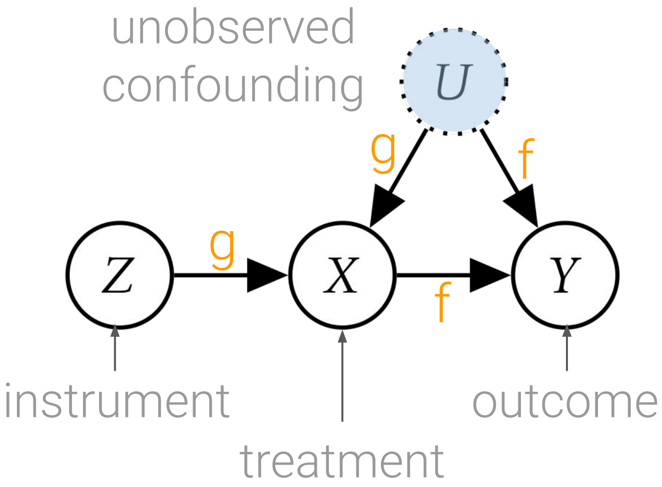
non-linear, non-additive

Goal

For any x* compute lower and upper bounds on the causal effect

$$\mathbb{E}[Y \mid do(x^\star)]$$

General problem formulation as optimization



optimize over “all” distributions

$$X = g(Z, U) \qquad Y = f(X, U)$$

optimize over “all” functions

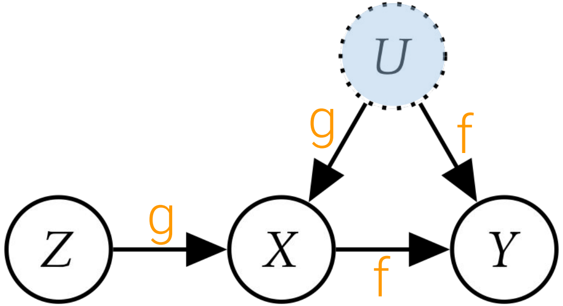
Goal

among all possible {*g*, *f*} and distributions over *U*
that reproduce the observed densities {p(x | z), p(y | z)},
estimate the min and max expected outcomes under intervention

Operationalizing this optimization

- without any restrictions on functions and distributions:
  effect is not identifiable and average treatment effect bounds are vacuous
  [Pearl, 1995; Bonet, 2001; Gunsilius 2018]
- mild assumptions suffice for meaningful bounds:
  f and *g* have a finite number of discontinuities [Gunsilius, 2019]
- rest of the talk:
  operationalize the optimization

find convenient representation of U from which we can sample

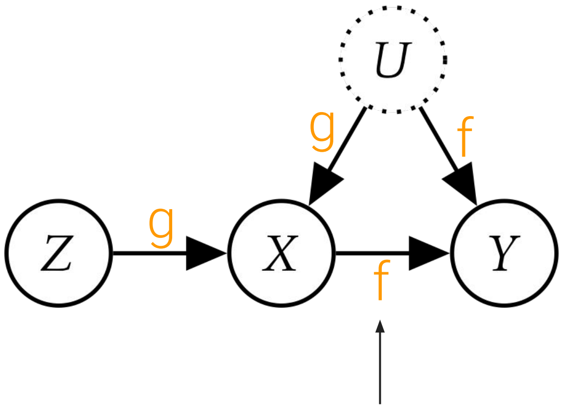


choose convenient function spaces

approximate constraints of preserving p(x | z) and p(y | z)

Our practical approach

Response functions I [Balke & Pearl, 1994]



ultimately, we care about this functional relation

each value of *U* fixes a functional relation *f*: *X* → *Y*

collect the set of all possible resulting functions

label these functions by and index summarizing all states of *U* that lead to the same function

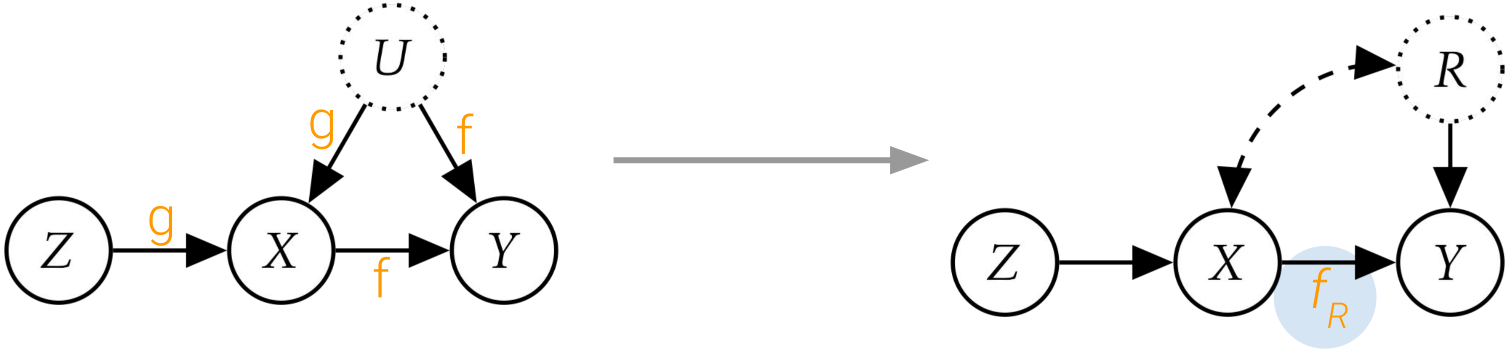
$$Y = f(X, U) = \lambda_1 X + \lambda_2 X U_1 + U_2$$

$$f(x, u) = \lambda_1 x + \lambda_2 x \quad \text{for} \quad u_1 = 1, u_2 = 0$$

$$f_r(x) = (\lambda_1 + \lambda_2)x \quad \text{where} \quad r \text{ is an alias for } (1, 0)$$

→ Instead of a potentially multivariate distribution over confounders *U* directly, we can think of a distribution *R* over functions *f*: *X* → *Y*

Response functions II



choose convenient function spaces

find convenient representation of U from which we can sample

find convenient representation of distributions over response functions

Parameterizing response functions

We choose a simple parameterization

$$f_r(x) := f_{\theta_r}(x) \quad \text{for} \quad \theta \in \Theta \subset \mathbb{R}^K$$

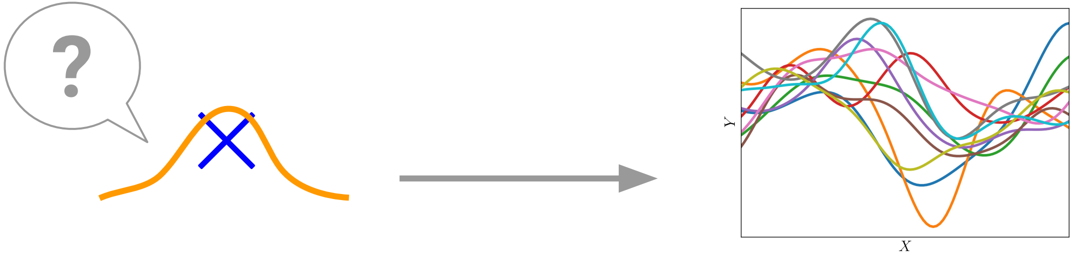
For simplicity, work with linear combination of (non-linear) basis functions:

$$f_\theta(x) = \sum_{k=1}^{K} \theta_k \psi_k(x) \quad \text{for basis functions} \quad \{\psi_k : \mathbb{R} \to \mathbb{R}\}_{k=1}^{K}$$

polynomials

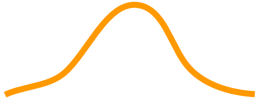
neural networks

Gaussian process samples



$$\theta \qquad\qquad f_\theta : X \to Y$$

Parameterizing the distribution over θ



$p_{\mathcal{M}}(\theta)$ implies a causal model

Goal

optimize over distributions $p_{\mathcal{M}}(\theta)$ such that

$$\int p_{\mathcal{M}}(x, y | z, \theta) p_{\mathcal{M}}(\theta)\, d\theta \quad \text{matches (estimated) marginals} \quad p(x|z), p(y|z)$$

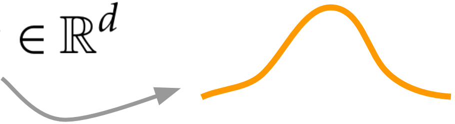
ideally

low variance Monte-Carlo gradient estimation

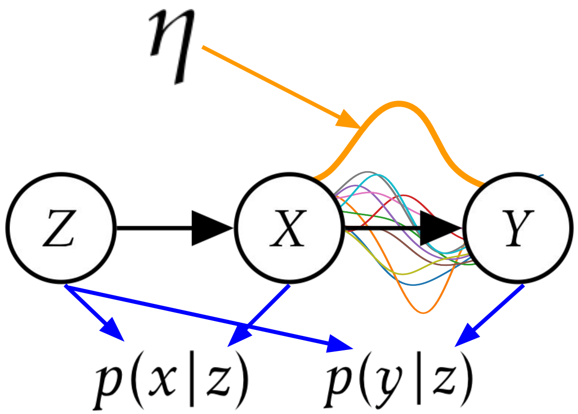
differentiable sampling

again, assume parametric form of $p_{\mathcal{M}}(\theta)$

$$p_{\eta}(\theta) \quad \text{with} \quad \eta \in \mathbb{R}^d$$



Objective function



objective

$$\min_{\eta}/\max_{\eta}\ \mathbb{E}[Y\,|\,do(x^{\star})] = \min_{\eta}/\max_{\eta} \int f_{\theta}(x^{\star})p_{\eta}(\theta)\,d\theta$$

How can we ensure the constraints: our model must match the observed data.

Match p(x | z) and enforcing Z ⊥ U

identified from data
manually *fix* it

factor $p_\eta(x,\theta|z) = p(x|z)\,p_\eta(\theta|x,z)$

$$p_\eta(\theta|x,z) := c_\eta\Big(F(x|z), F_\eta(\theta_1), \dots, F_\eta(\theta_K)\Big)\prod_{k=1}^{K} p_\eta(\theta_k)$$

copula density

univariate CDFs

Gaussian marginal densities

$$p_\eta(\theta_k) = \mathcal{N}(\theta_k; \mu_k, \sigma_k^2)$$

for a multivariate Gaussian copula, the optimization parameters are

$$\eta := \{\mu_1, \ln(\sigma_1^2), \dots, \mu_K, \ln(\sigma_K^2), L\} \in \mathbb{R}^{K(K+1)/2+2K}$$

Match p(y | z)

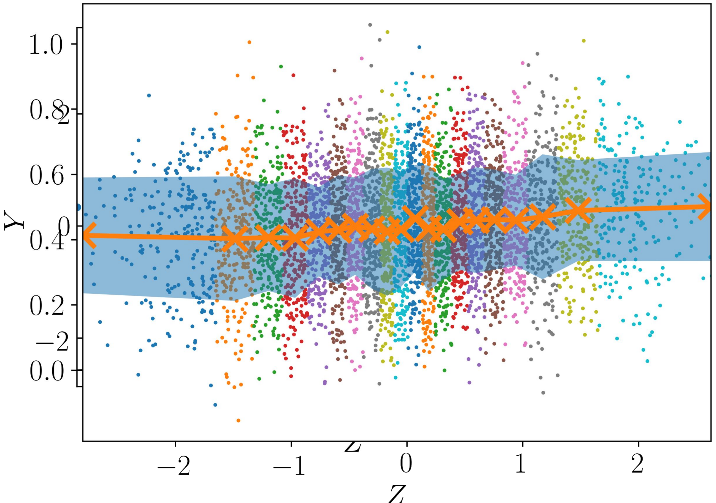
exact constraint in the continuous outcome setting

data $$\Pr(Y \le y \,|\, Z = z) = \int \mathbf{1}(f_\theta(x) \le y)\, p_\eta(x, \theta \,|\, z)\, dx\, d\theta$$ our model

- infinite number of constraints
- integral over non-continuous indicator

choose discrete finite grid in z and assign points to bins

$$z^{(m)} := F_Z^{-1}\left(\frac{m}{M+1}\right) \quad \text{for } m \in [M]$$



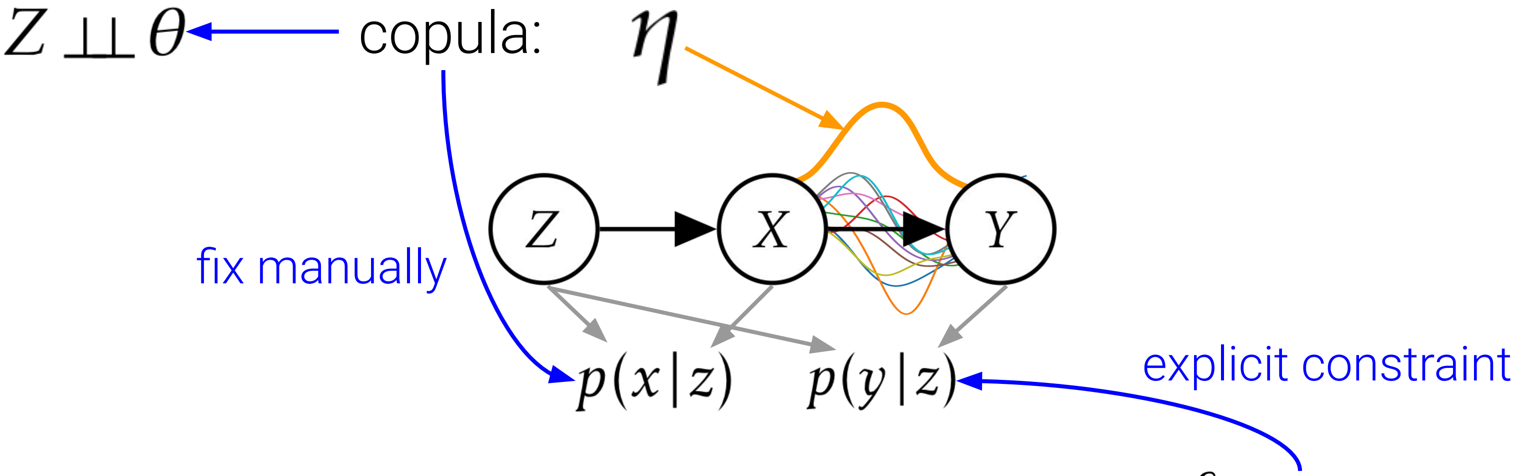
for a dictionary of basis functions $\{\phi_l\}_{l=1}^{L}$

$$\mathbb{E}[\phi_l(Y) \,|\, z^{(m)}] = \int \phi_l\big(f_\theta(x)\big)\, p_\eta(x, \theta \,|\, z^{(m)})\, dx\, d\theta$$

data our model

$\phi_1(Y) := \mathbb{E}[Y],\ \phi_2(Y) := \mathbb{V}[Y]$

Intermediate overview



$Z \perp\!\!\!\perp \theta$

copula: η

fix manually

$p(x\,|\,z)$ $p(y\,|\,z)$

explicit constraint

$$\mathbb{E}[\phi_l(Y)\,|\,z^{(m)}] = \int \phi_l\big(f_\theta(x)\big)\,p_\eta(x,\theta\,|\,z^{(m)})\,dx\,d\theta$$

objective

$$\min_\eta/\max_\eta \mathbb{E}[Y\,|\,do(x^\star)] = \min_\eta/\max_\eta \int f_\theta(x^\star)\,p_\eta(\theta)\,d\theta$$

The final optimization problem

can sample from these in a differentiable fashion (w.r.t. η)

objective: $$o_{x^\star}(\eta) := \int f_\theta(x^\star)\boxed{p_\eta(\theta)}\,d\theta$$

$$\boxed{\text{constraint LHS:} \quad \mathrm{LHS}_{m,l} := \mathbb{E}[\phi_l(Y)\,|\,z^{(m)}]}$$

precompute once up front from data

constraint RHS: $$\mathrm{RHS}_{m,l}(\eta) := \int \phi_l(f_\theta(x))\boxed{p_\eta(x,\theta\,|\,z^{(m)})}\,dx\,d\theta$$

opt. problem: $$\min_\eta / \max_\eta o_{x^\star}(\eta) \quad \text{s.t.} \quad \boxed{\mathrm{LHS}_{m,l}} = \mathrm{RHS}_{m,l}(\eta) \text{ for all } m \in [M], l \in [L]$$

only satisfy this approximately

use augmented Lagrangian with stochastic gradient descent

- for each $z^{(m)}$ sample batch of θ
- take average to estimate objective and constraint term RHS
- use auto-differentiation to get gradient and take gradient step

Empirical results

Choices of response functions

$$f_\theta(x) = \sum_{k=1}^{K} \theta_k \psi_k(x) \quad \text{for basis functions} \quad \{\psi_k : \mathbb{R} \to \mathbb{R}\}_{k=1}^{K}$$

Polynomials

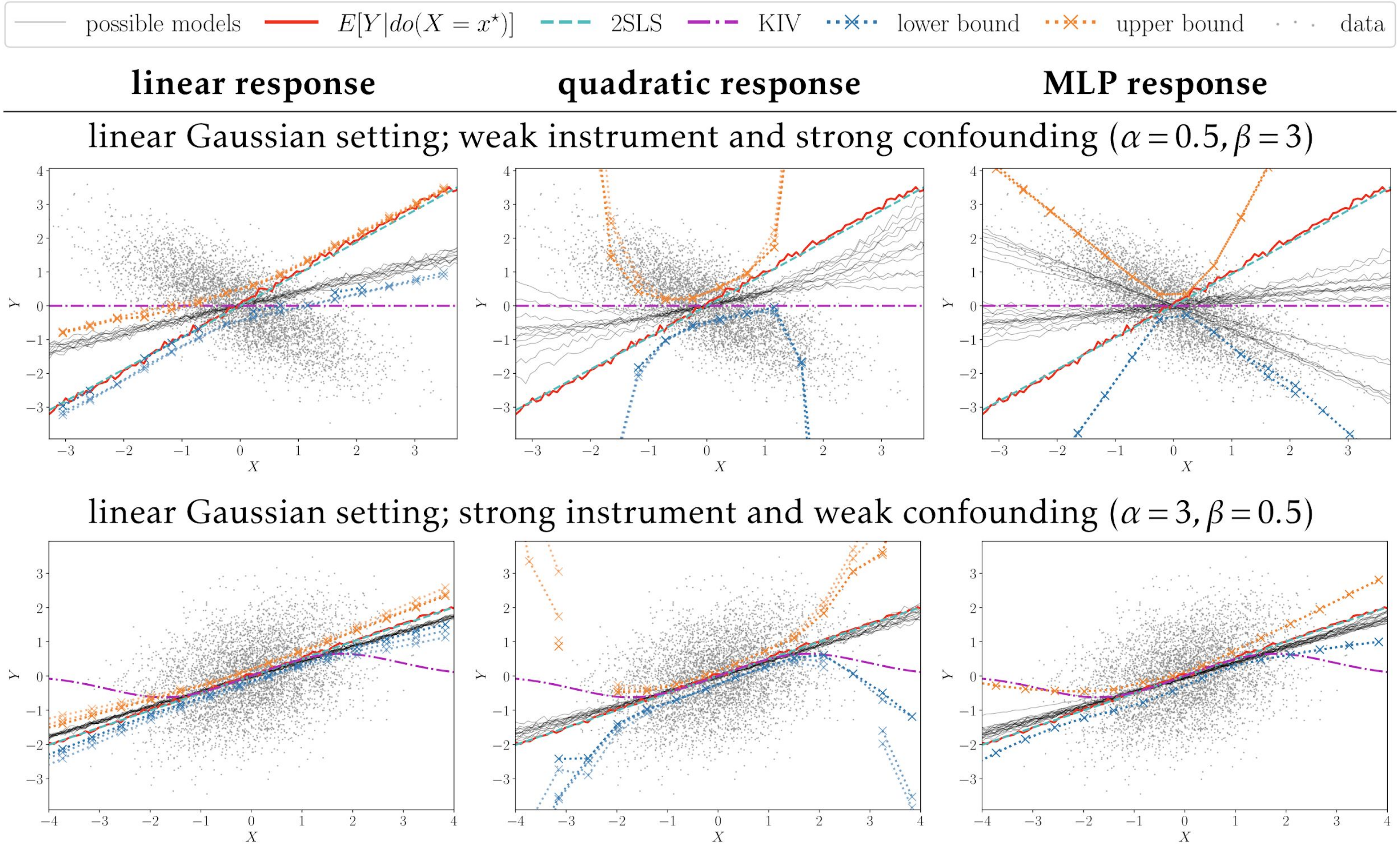
$\psi_k(x) = x^{k-1}$ for $k \in [K]$

Neural network

Train a small fully connected network on observed data $X \to Y$ and take activations of last hidden layer as basis functions.

Gaussian process

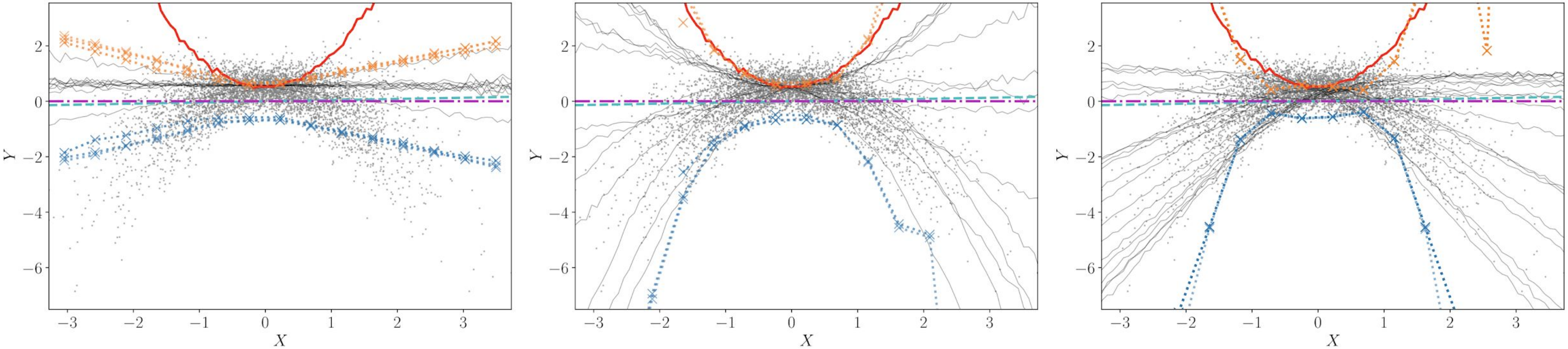
Train GPs on subsets of observed data X→Y and take random samples from the GP as basis functions.



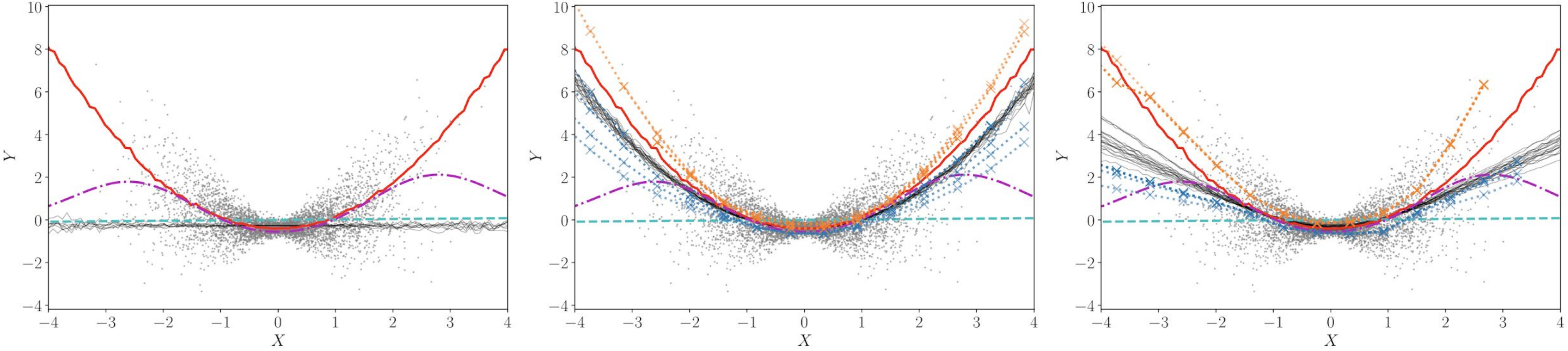
linear Gaussian setting; weak instrument and strong confounding ($\alpha = 0.5, \beta = 3$)

linear Gaussian setting; strong instrument and weak confounding ($\alpha = 3, \beta = 0.5$)

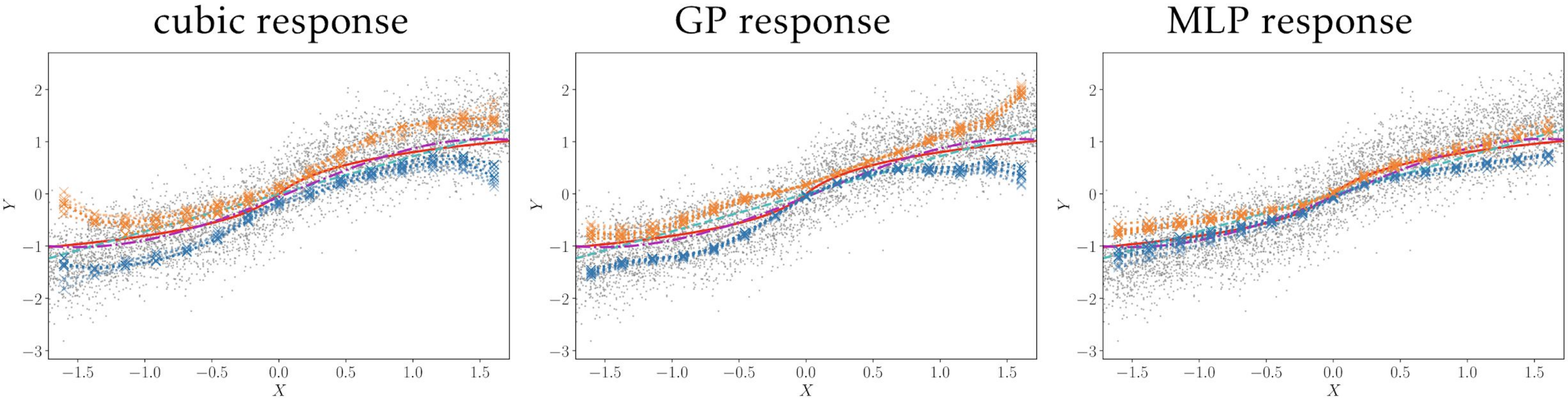
non-additive, non-linear setting; weak instrument and strong confounding ($\alpha = 0.5, \beta = 3$)



non-additive, non-linear setting; strong instrument and weak confounding ($\alpha = 3, \beta = 0.5$)



Sigmoidal cause-effect design



more details and experiments (also in the small data regime) in the paper
https://arxiv.org/abs/2006.06366

Thank you