MLRS 2023 Tutorial: Part I

https://www.mlrs.ai/

Introduction to Causal Machine Learning

Amit Sharma

Principal Researcher

Microsoft Research

www.amitsharma.in

Twitter: @amt_shrma

Two session series on Causal ML

- **Session 1:** Intro to causal machine learning
  - Estimating causal effect, explaining outcomes, and out-of-distribution generalization
- **Session 2:** Causal machine learning in practice
  - PyWhy/DoWhy and the promise of large language models

Session goals: Intro to causal machine learning

- What is the difference between **causal** and **predictive** machine learning (ML)?
  - **Side-goal:** Learn about causality fundamentals

- When is causal ML **useful**?

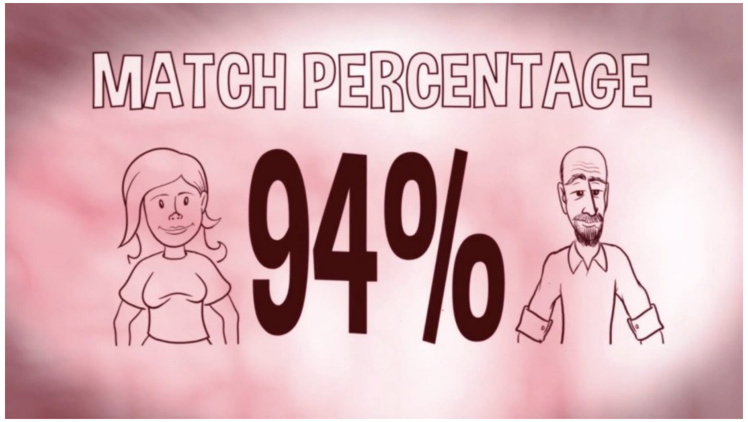
- What can you achieve with causality + ML?
  - **Looking forward:** Take better decisions
  - **Looking backward:** Explain the reasons for observed outcomes
  - **Improving ML models:** Better generalization of ML models

Outline

1. **What is causal ML and why do we need it?**
    1. What can you achieve using causal ML?
    2. When is it practically useful compared to predictive ML?
2. **Fundamental concepts in causality**
    1. Interventions and counterfactuals
    2. Causal graph
3. **Three main applications**
    1. Choosing the “best” decision for a target outcome
    2. Attributing causes for a target outcome
    3. Building predictive models that generalize out-of-distribution

Section 1: What is causal ML and why do we need it?

When we think of machine learning, we often think of **predictions: What does the data say?**



But there’s an important class of problems about **decisions: what action should I take?**

Which customers should we provide discounts to improve sales?

Which treatment will have the best improvement for a patient?

What is the best way to share an important public safety message?

Would this government regulation lead to a decrease in air pollution?

Sometimes, these problems overlap…

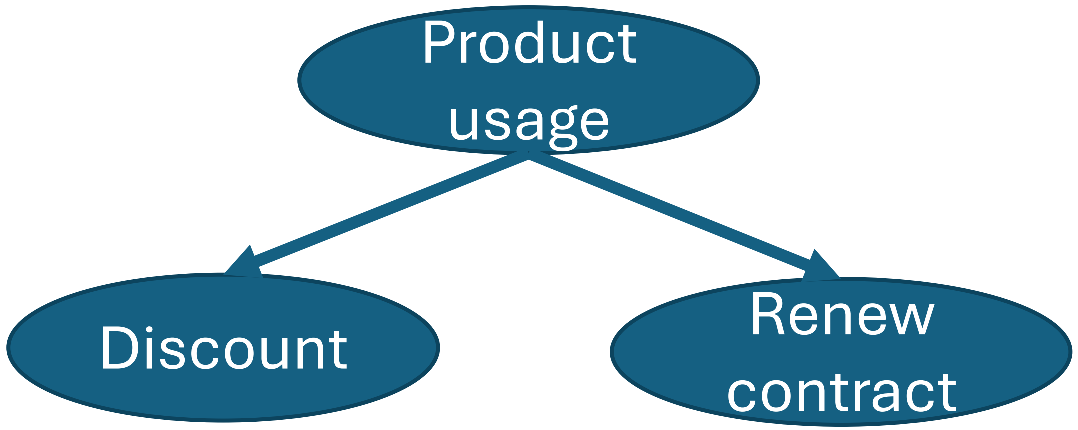
- Accurate prediction also means accurate decision-making.

- **Prediction task:** Does the X-ray image indicate a tumor?
- **Decision task:** Should we give tumor treatment or not?

But sometimes, they do not

- **Prediction task:** Predict the customers most likely to churn out.
- **Decision task:** Who to provide discounts to?
  - Discounts may not work on people likely to churn out (low activity)
  - May be unnecessary for people with high activity.
  - **Only need to find the people in the middle,** who are undecided.

Reason: Correlation versus causation



- Today's product usage can predict tomorrow's probability of churn (not renewing contract).
- But does not tell us anything about effect of discount.
  - Effect could even be zero!

And often, decision-making requires solving a new kind of problem: **effect estimation**

- **Effect estimation:** What is the effect of an action on the outcome?

What is the best way to share an important public safety message?

Q: What is the effect of sharing medium on response rate for the safety message?

Would this government regulation lead to a decrease in air pollution?

Q: What is the effect of the regulation on air pollution?

In effect estimation, the most important task is **how to avoid being fooled by correlations**

What is the best way to share an important public safety message?

Observed data: The response rate of text messages is the highest.

Selection bias: Dataset contains mostly young people.

Would this government regulation lead to a decrease in air pollution?

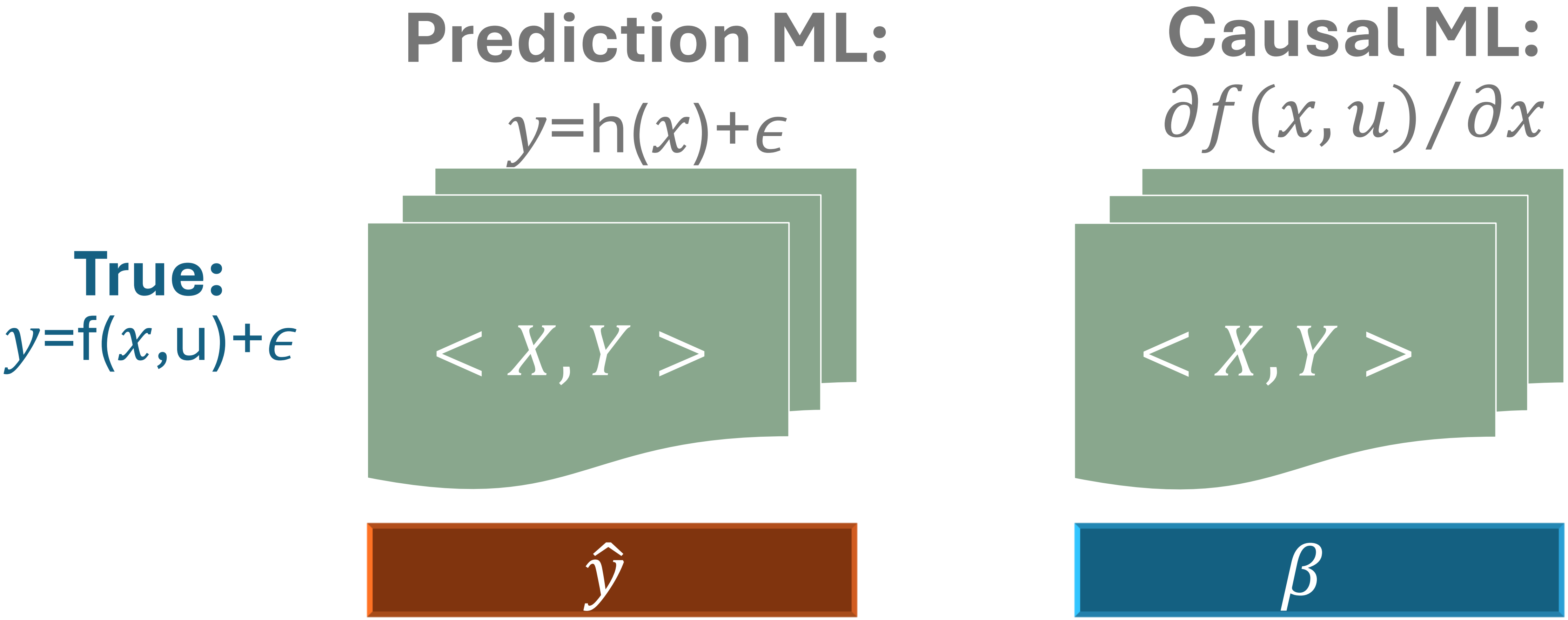
Observed data: In other states, pollution decreased after the regulation.

Confounding bias: Other states differ on the kind of industries they have.

So, how to solve these problems in a systematic way?

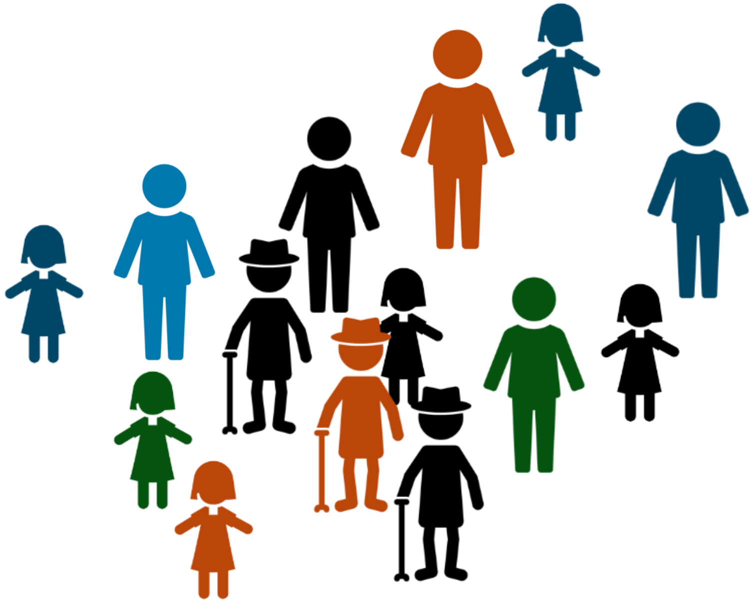
Incorporate techniques for **learning causality** in ML models.

Causal ML is about inferring the **best actions** (and the effects of actions in general)



Hofman, **Sharma**, and Watts (2017). Prediction and Explanation in Social Systems. *Science*, 355.6324

Three key applications of causal ML: **Better decision-making** (what to do next?)

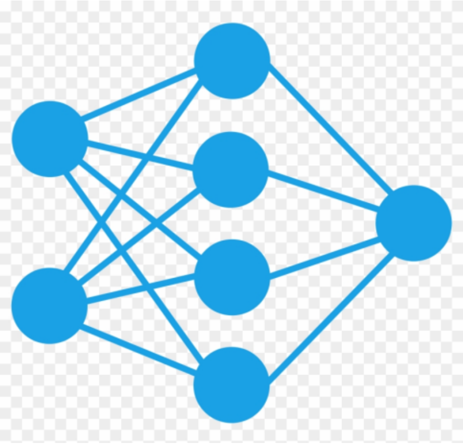


People who do not cycle have high cholesterol

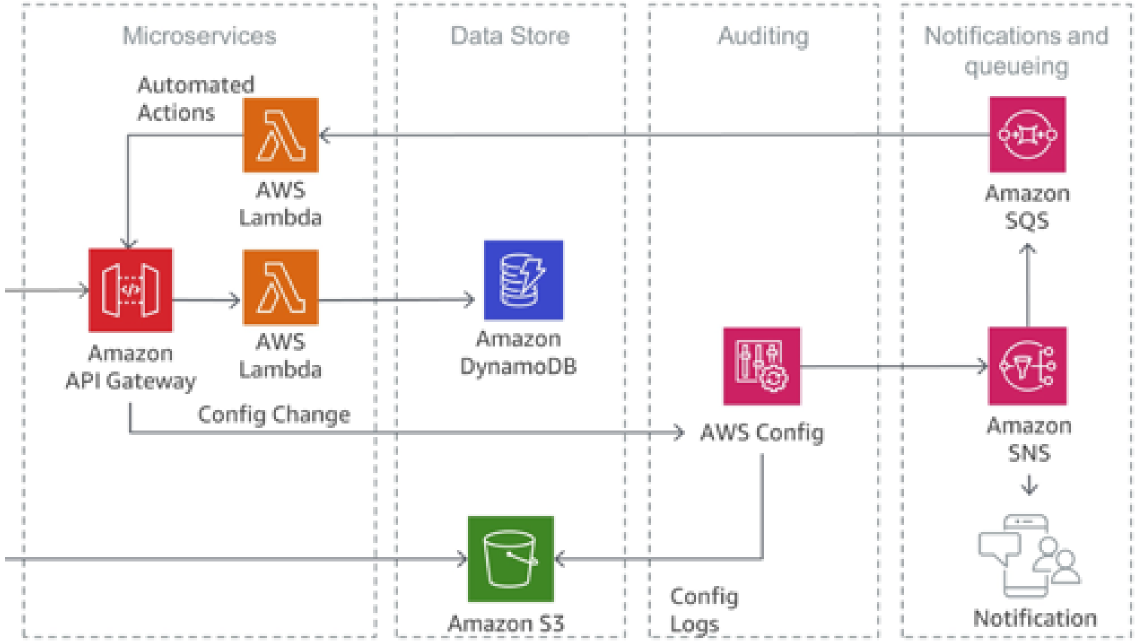
People who cycle regularly have low cholesterol

Decision: To improve cholesterol levels of the population, should the city government invest in programs for encouraging cycling (e.g., giving free bikes)?

Three key applications of causal ML:
Root cause attribution (why did this happen?)



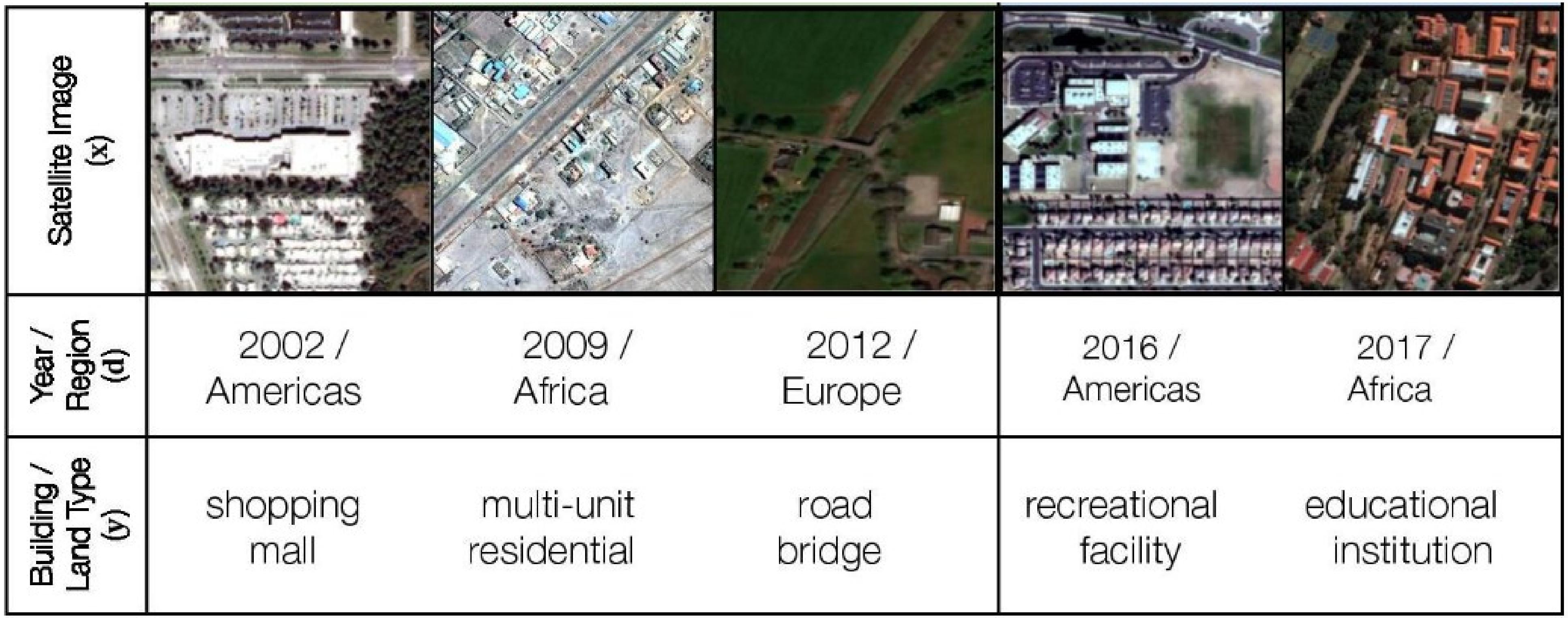
Predicted Class
Class: 1
Class: 0
Class: 0
Class: 1



Attribution: Why did the classifier predict Class:1 for the first image?

Attribution: For a given the microservice system, why did the latency increase?

Three key applications of causal ML:
Out-of-distribution generalization



Koh et al., WILDS, ICML 2021

To summarize,

Causal ML:
Machine learning + causality

A necessary ingredient for general-purpose AI

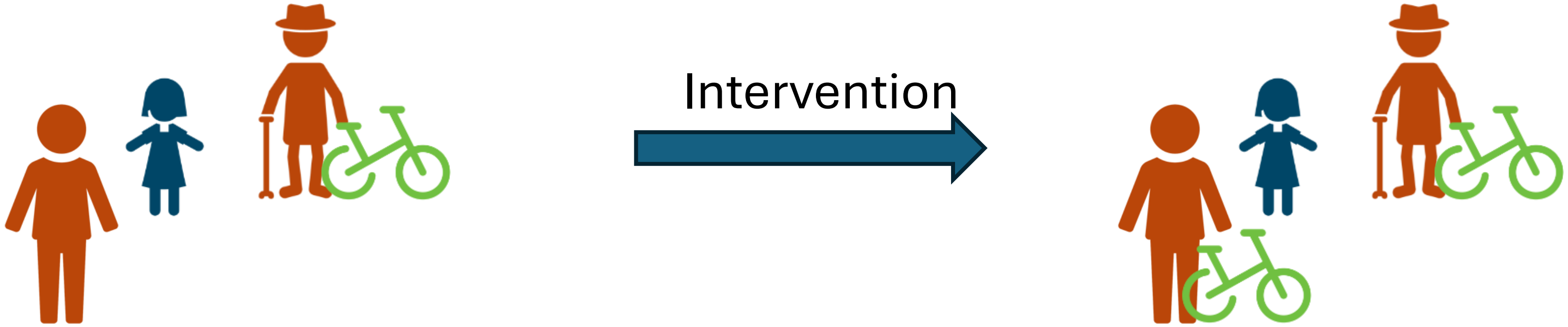
Section 2: Fundamental concepts in causality

(intervention, counterfactual, & causal graph)

Intervention: A formal definition for taking an action

Intervention: An active action taken that changes the distribution of a variable *T*.

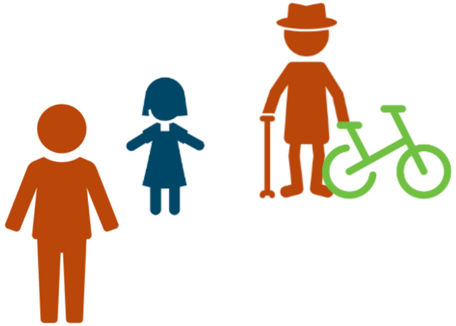
- Different from *observing* two different values of T.



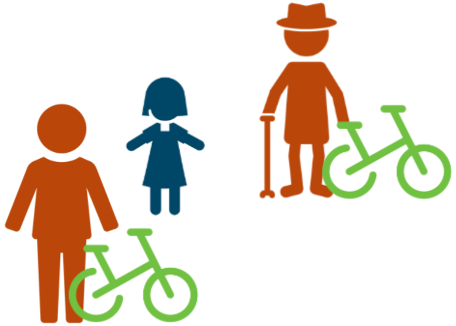
Observed data
Cycling (T)={0,0,1}

Interventional data
Cycling (T)={1,0,1}

Mathematically represented using the **do-operator**



Intervention



Observed data

Cycling (T)={0,0,1}

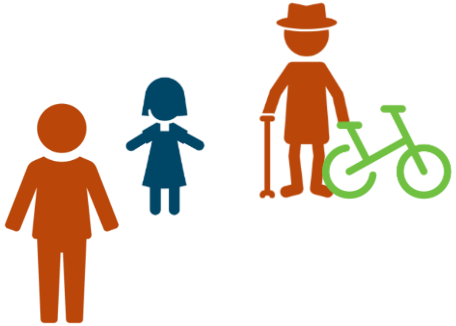
Interventional data

Cycling (T)={1,0,1}

$$P(Health|Cycling, Age, Color)$$

$$P^*(Health|Cycling, Age, Color)$$
$$=P(Health|do(Cycling), Age, Color)$$

Important: A do-intervention affects only the desired variable, **keeping everything else fixed**



Not an Intervention

Observed data 1

Cycling (T)={0,0,1}

$$P(Health|Cycling, Age, Color)$$

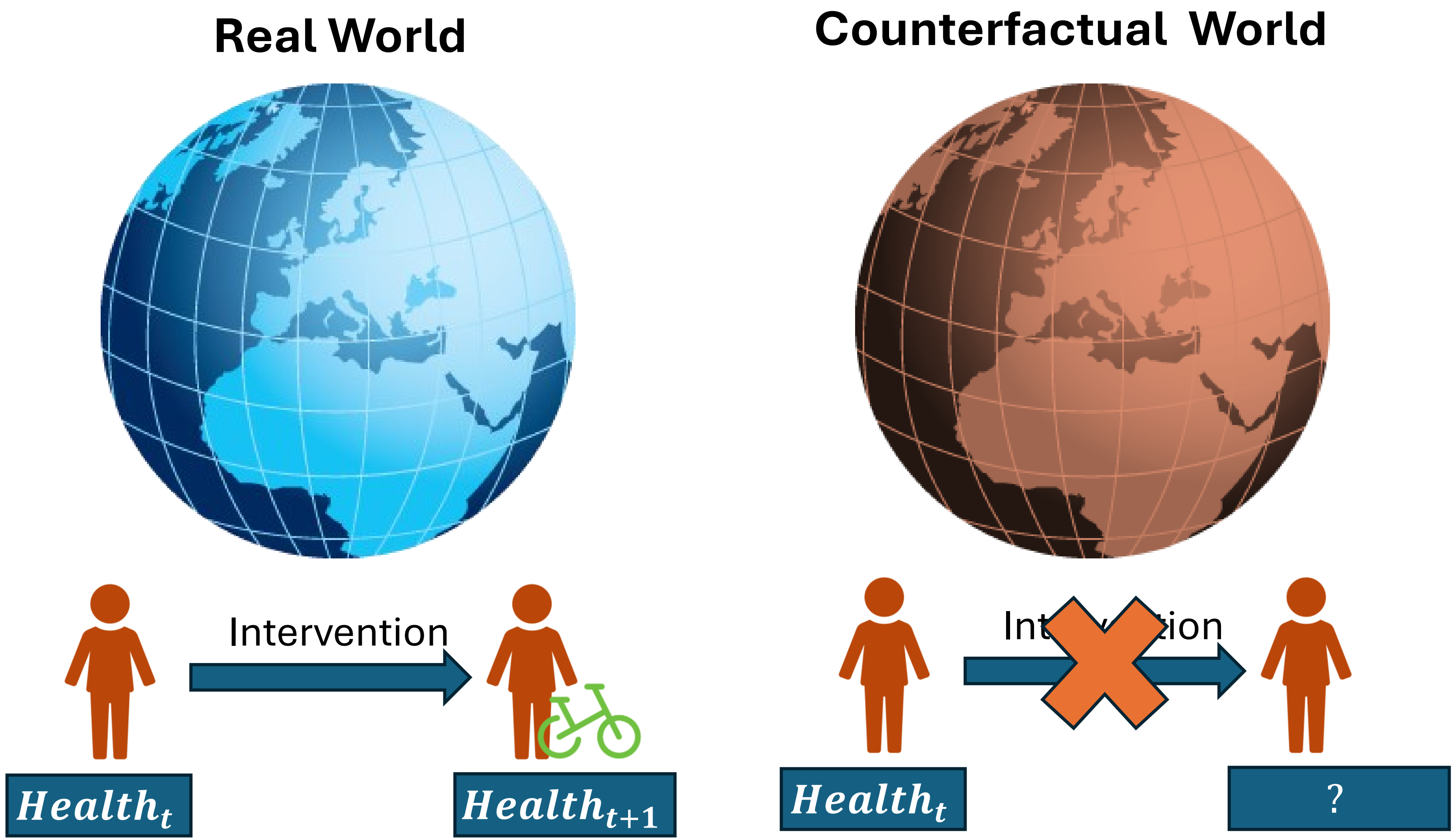
Observed data 2

Cycling (T)={1,0,1}

$$P^*(Health|Cycling, Age, Color)$$

$$!= \cancel{P(Health|do(Cycling), Age, Color)}$$

Second important concept: **counterfactual**
(What would have happened *if*)



Counterfactual: Complicated to express formally, but intuitive to grasp

$$\boldsymbol{P\left(Health_{t+1,0} \middle| Health^*_{t+1}, Health^*_{t}, Cycle_{t+1} = 1, do(Cycle_{t+1}) = 0\right)}$$

Given that person started cycling and improved their health, ***what would have happened to their health if they did not start cycling, but everything else remained the same***?

Now we are ready to define the causal effect of a variable

Definition: X causes Y iff

changing X leads to a change in Y,

keeping everything else constant.

The **causal effect** is the magnitude by which Y is changed by a unit change in X.

$$P(Y|do(X=1)) - P(Y|do(X=0))$$

Real World **Counterfactual World**

-

As we will see, the key problem is that one of the terms is **never observed in data**

$$P(Y|do(X=1)) - P(Y|do(X=0))$$

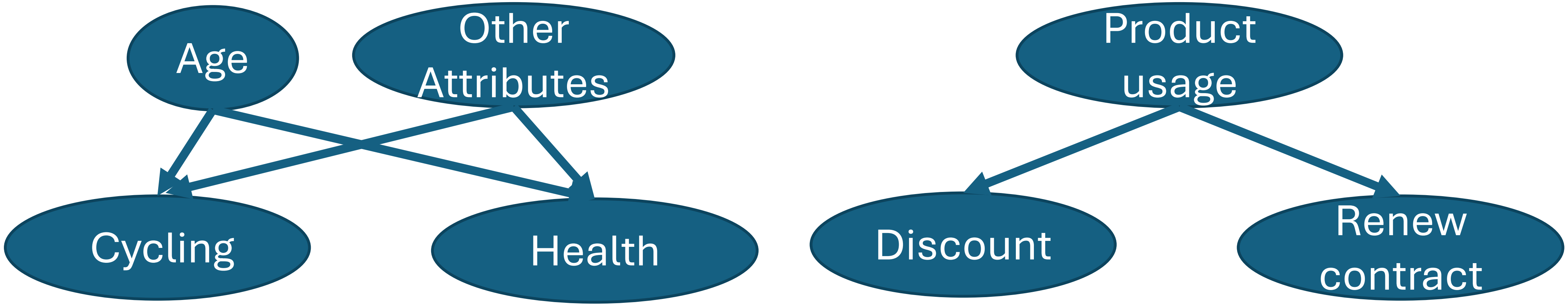
$$= P^{obs}(Y|X=1) - P(Y|do(X=0))$$

Predictive ML solves the problem by assuming

$$P(Y|do(X=0)) = P^{obs}(Y|X=0)$$

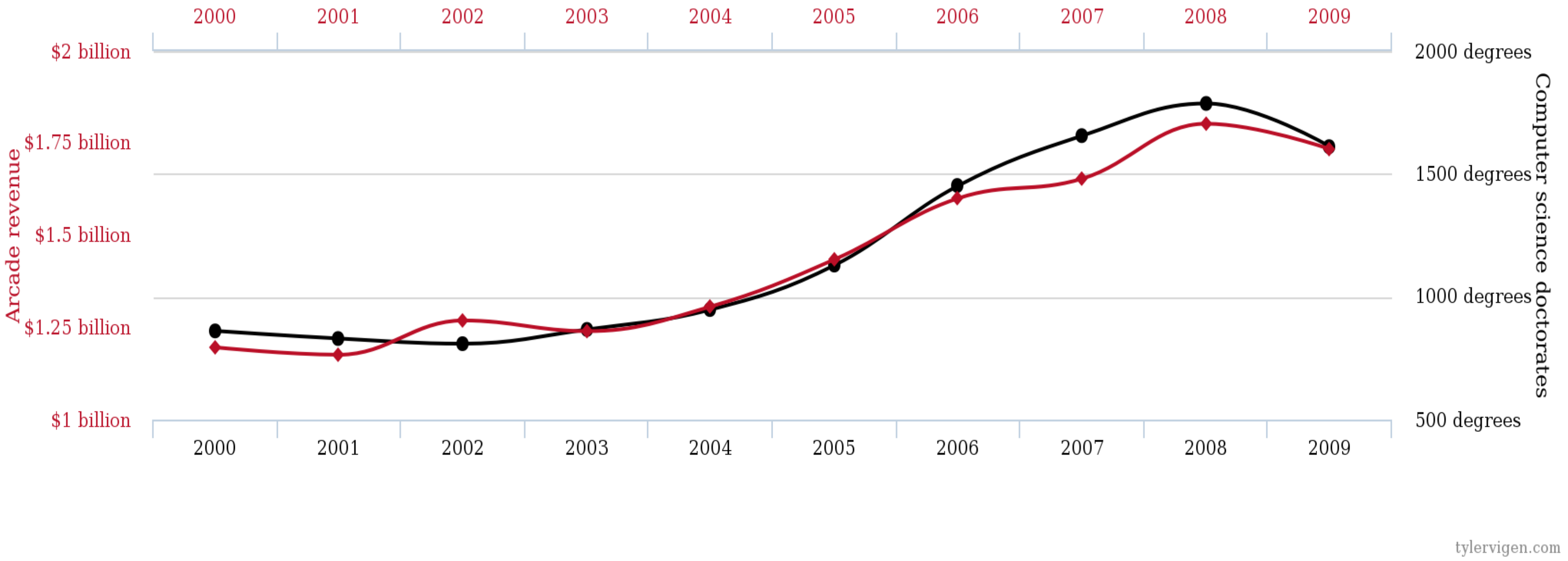
The goal of causal ML is to find a better approximation.

Final concept: **Causal graph** to encode assumptions that help estimate the unseen



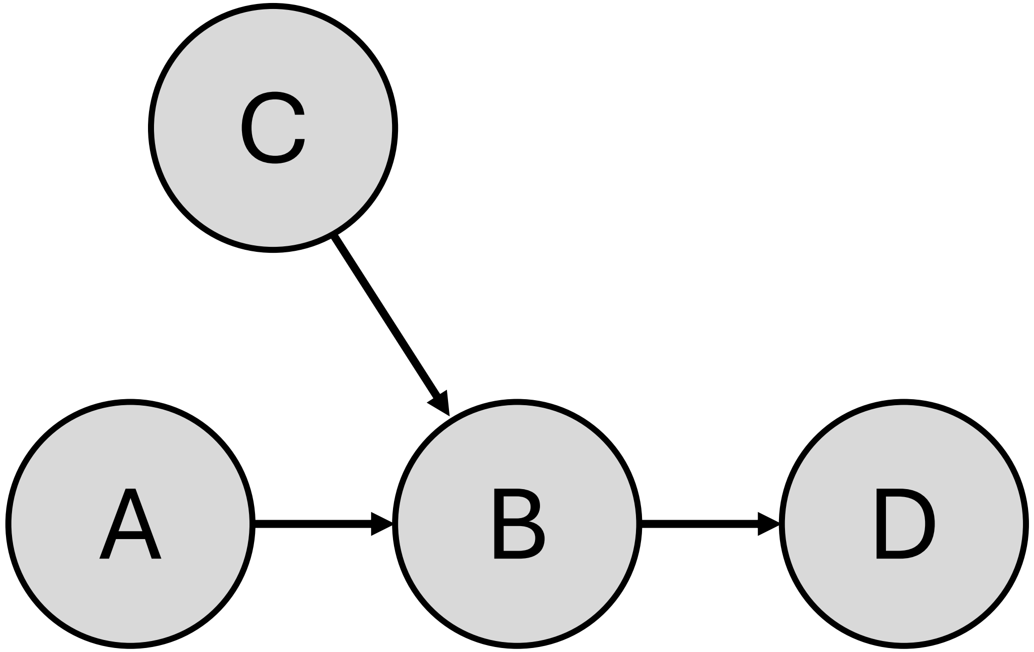
- A good graph exposes the key assumptions about how different variables affect each other
  - $A \rightarrow B$ or $B \rightarrow A$?

Failure case: What can happen **without causal assumptions**?



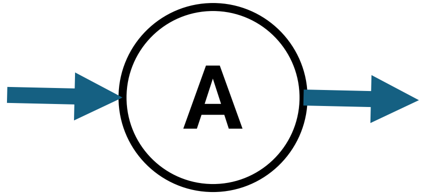
http://www.tylervigen.com/spurious-correlations

Interpreting a causal graph: **d-separation**

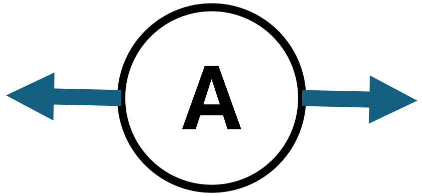


- Edges encode mechanisms
  - *direct causes*
- Graph implies conditional statistical independences
  - E.g., $A \perp\!\!\!\perp C,\ D \perp\!\!\!\perp \mathrm{A} \mid \mathrm{B}, \ldots$
  - Identified by *d-separation* rules

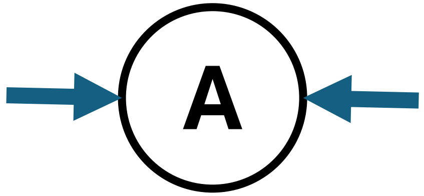
Interpreting a causal graph: d-separation



If conditioned on X



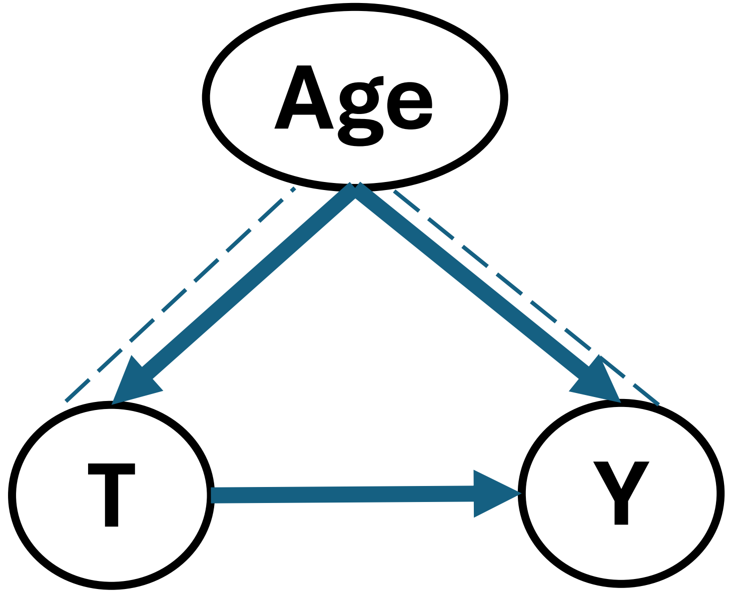
If conditioned on X



If not conditioned on X

Three kinds of node-edges

Path is "blocked" → path is d-separated



$\boldsymbol{X} = \{Age\}$

Insight: The assumptions are not the edges you create, but the edges you omit

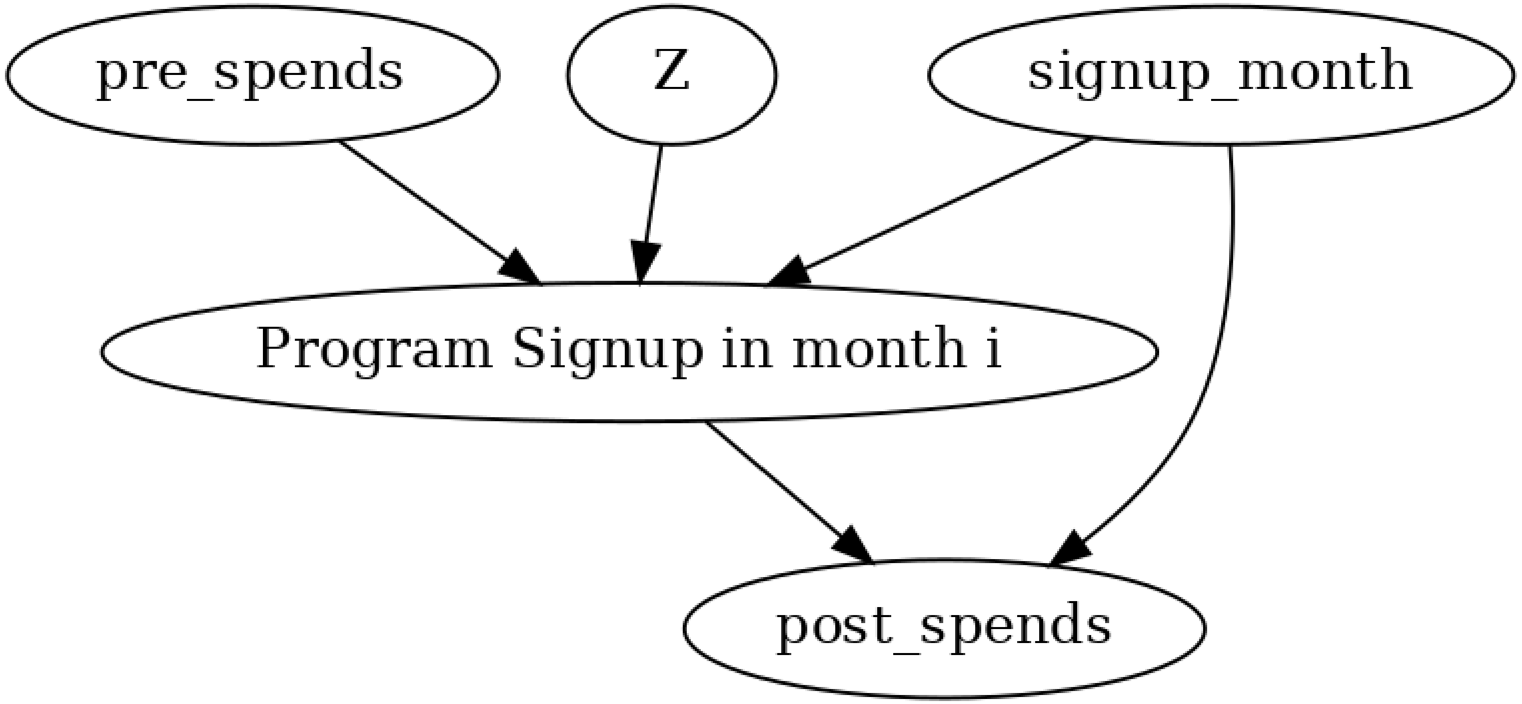
- Assumptions are encoded by *missing edges*, and *direction* of edges

- Relationships represent stable and independent mechanisms

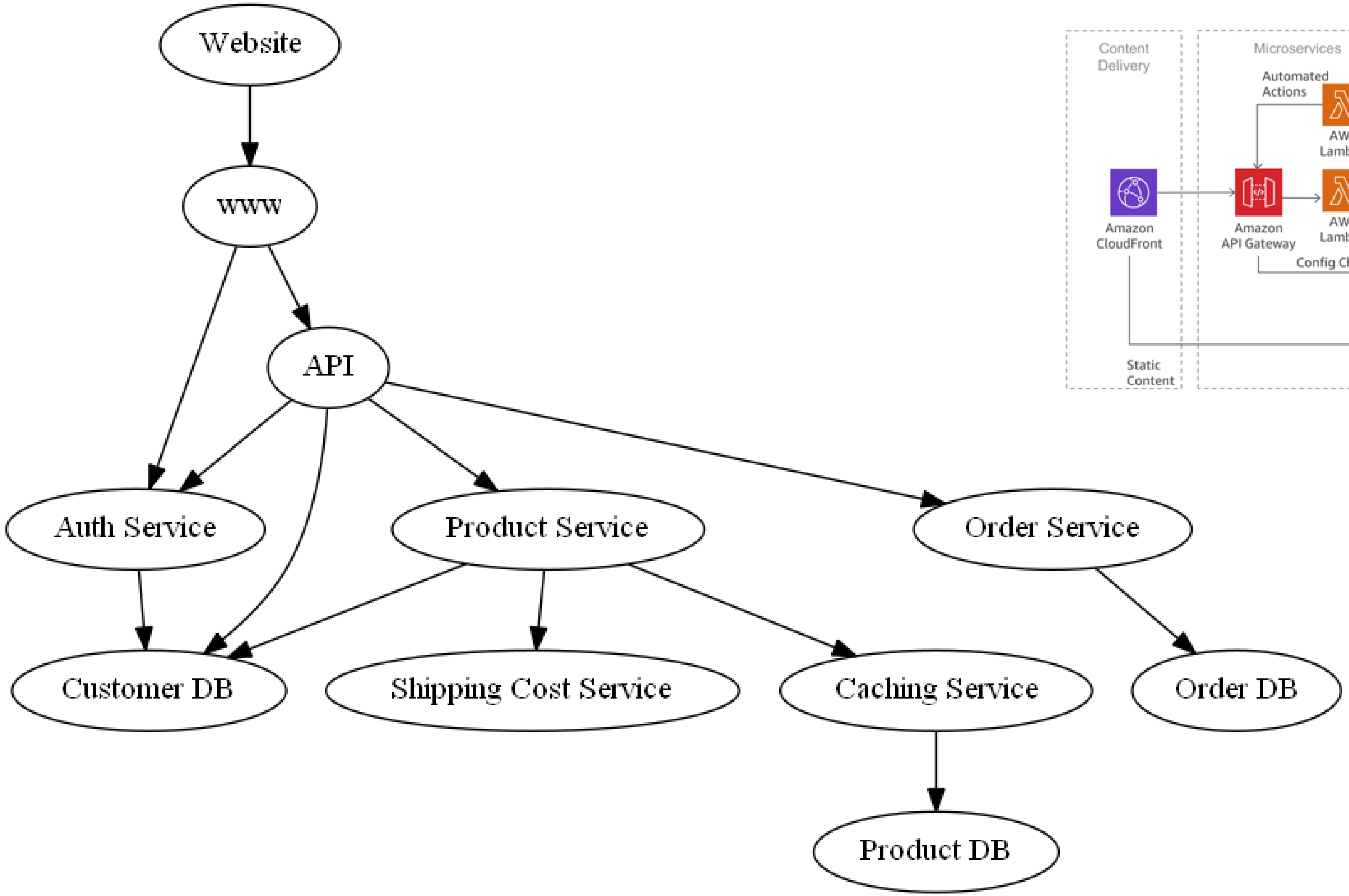
- It is **not always possible** to learn a graph from observational data

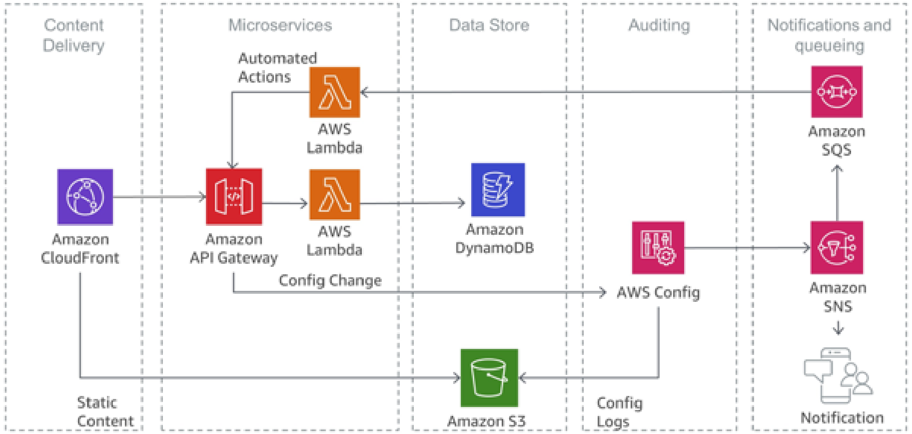
How to obtain a causal graph? Use **domain knowledge** (Example 1)

- Estimating the effect of customer rewards program



How to obtain a causal graph?
Use **domain knowledge** (Example 2)





Section 3: Applications of causal ML

(decision-making, root cause attribution, out-of-distribution generalization)

Decision-making: Given a target outcome, which action maximizes the outcome value?

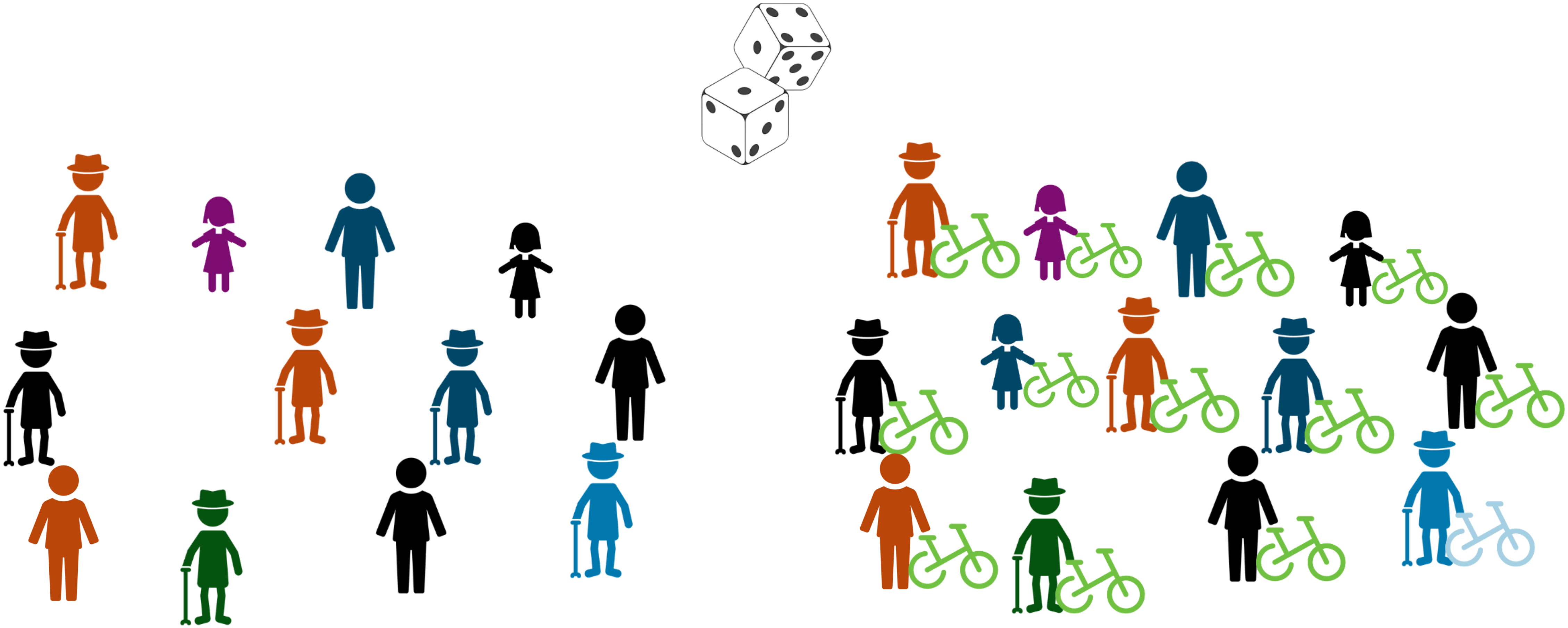
Frame as causal effect estimation problem

$$P(Y|do(A = a_1)), P(Y|do(A = a_2)), \ldots$$

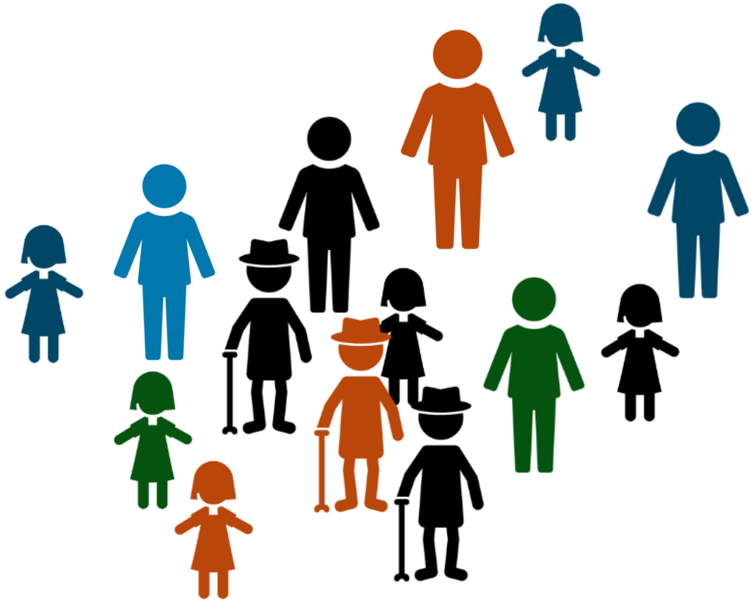
Rank the different causal effects

Choose the action with **highest causal effect on outcome**.

Randomized “A/B” test: A simple solution if **you can intervene** and create new data



But what to do if we **cannot intervene**?



People who do not cycle
have high cholesterol

People who cycle regularly
have low cholesterol

Simple Matching: Match data points with the same confounders and then compare their outcomes

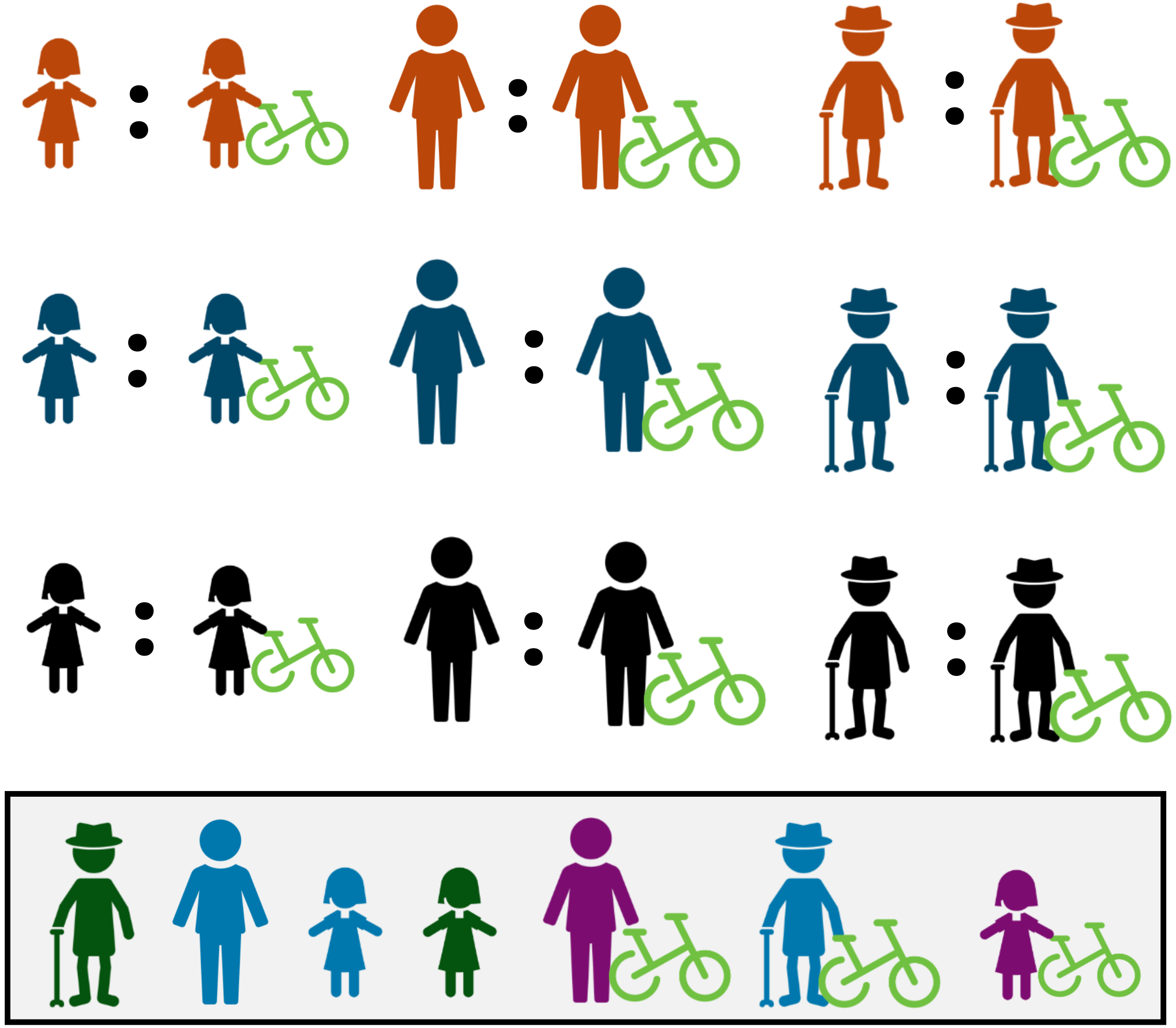
Identify pairs of treated (j) and untreated individuals (k) who are similar or identical to each other.

Match := $Distance\left(W_j, W_k\right) < \epsilon$

- Paired individuals have almost the same confounders.

  Causal Effect =

  $$\sum_{(j,k)\in Match}(y_j - y_k)$$



Challenges of building a good estimator

- **Variance:** If we have a stringent matching criterion, we may obtain very few matches and the estimate will be unreliable.

- **Bias:** If we relax the matching criterion, we obtain many more matches but now the estimate does not capture the target estimand.

- **Uneven treatment assignment:** If very few people have treatment, leads to both high bias and variance.

Need better methods to navigate the bias-variance tradeoff.

The intuition leads to a popular principle for estimating causal effect: **backdoor criterion**

Backdoor formula

$$p(Y|do(T)) = \sum_{Z} p(Y|T,Z)p(Z)$$

Where Z must be a valid adjustment set:

- The set of all parents of T
- Features identified via *backdoor criterion*
- Features identified via "towards necessity" criterion

Intuitions:

- The union of all features is *not* necessarily a valid adjustment set
- Why not always use parents? Sometimes parent features are unobserved

Voila! Effect inference problem reduced to estimating **conditional expectation**

For common identification strategies using adjustment sets,

$$E[Y|do(T = t), W = w] = E[Y|T = t, W = w]$$

assuming W is a valid adjustment set.

- For binary treatment,

Causal Effect = $E[Y|T = 1, W = w] - E[Y|T = 0, W = w]$

Goal: Estimating conditional probability Y|T=t when all confounders W are kept constant.

Machine learning methods can help find a better match for each data point

Synthetic Control: If a good match does not exist for a data point, can we create it synthetically?

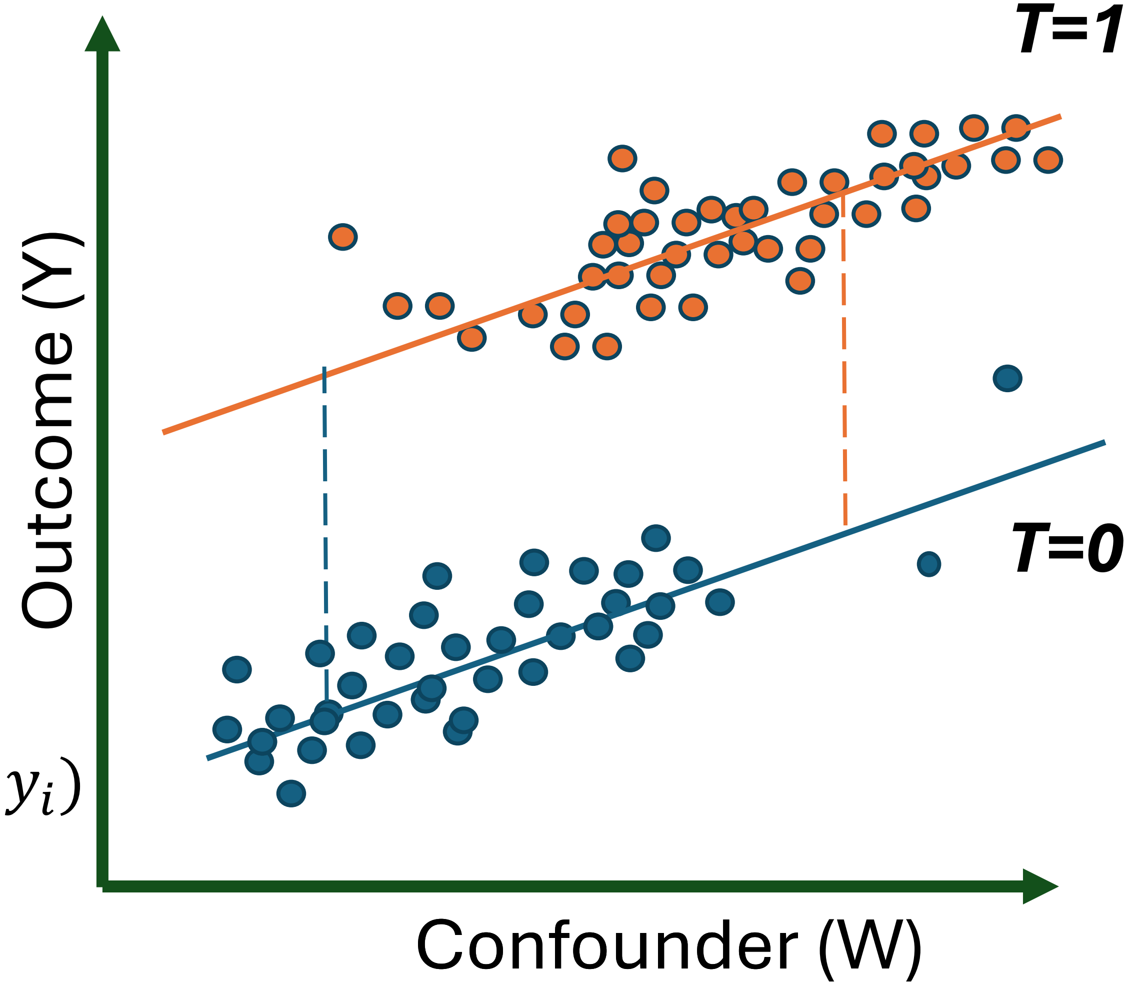
Learn $y = f_{t=0}(w)$,

$y = f_{t=1}(w)$

Assuming f approximates the true relationship between Y and W,

Causal Effect =

$$\sum_i t_i(y_i - f_{t=0}(w_i)) + (1 - t_i)(f_{t=1}(w_i) - y_i)$$



A better solution: use **debiased ML** estimator

The standard predictor, $y = f(t, w) + \epsilon$

may not provide the right estimate for $\frac{\partial y}{\partial t}$.

Debiased-ML [Chernozhukov et al. 2016]**:**

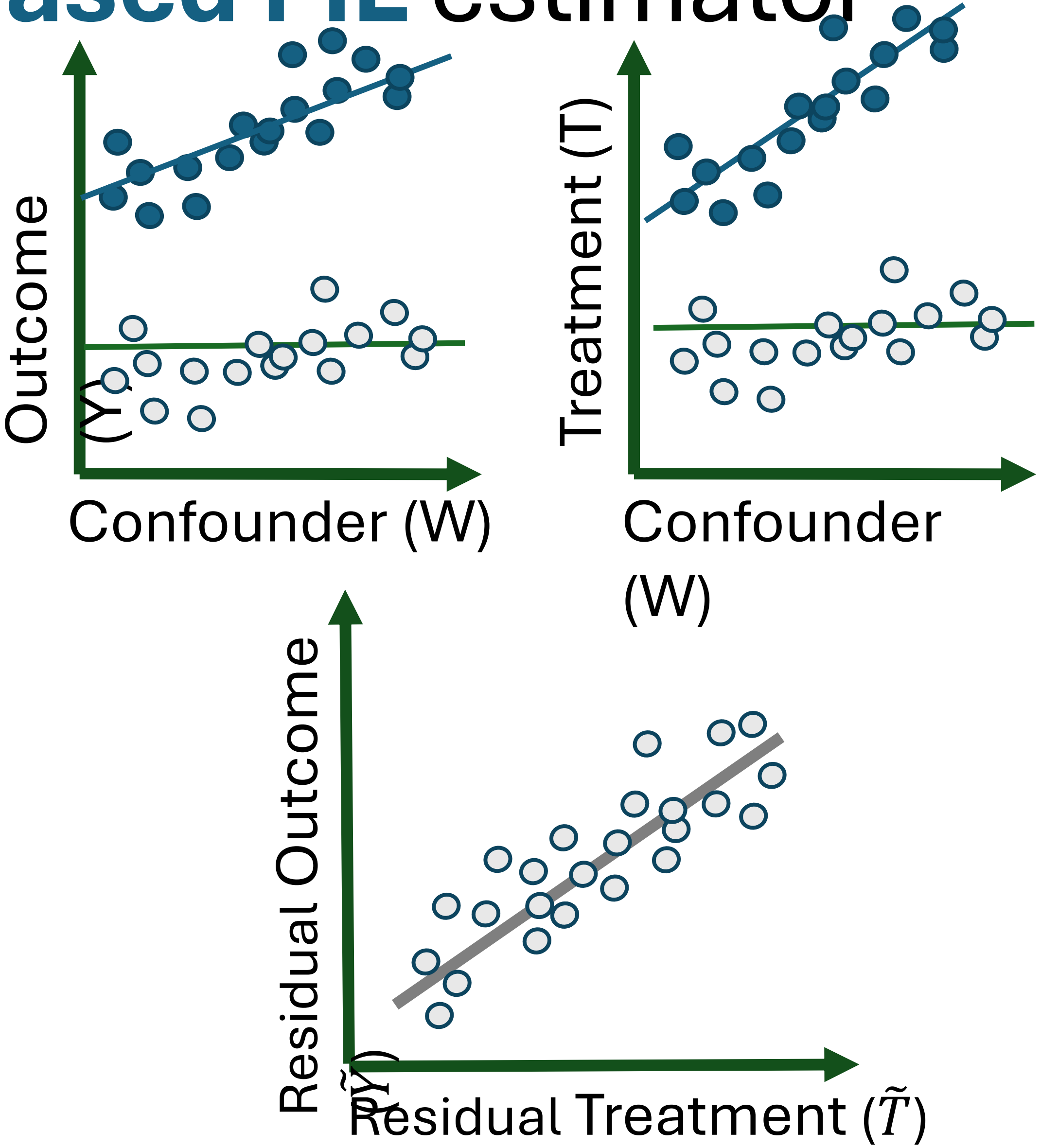
- Stage 1: Break down conditional estimation into two prediction sub-tasks.

$$\hat{y} = g(w) + \tilde{y}$$
$$\hat{t} = h(w) + \tilde{t}$$

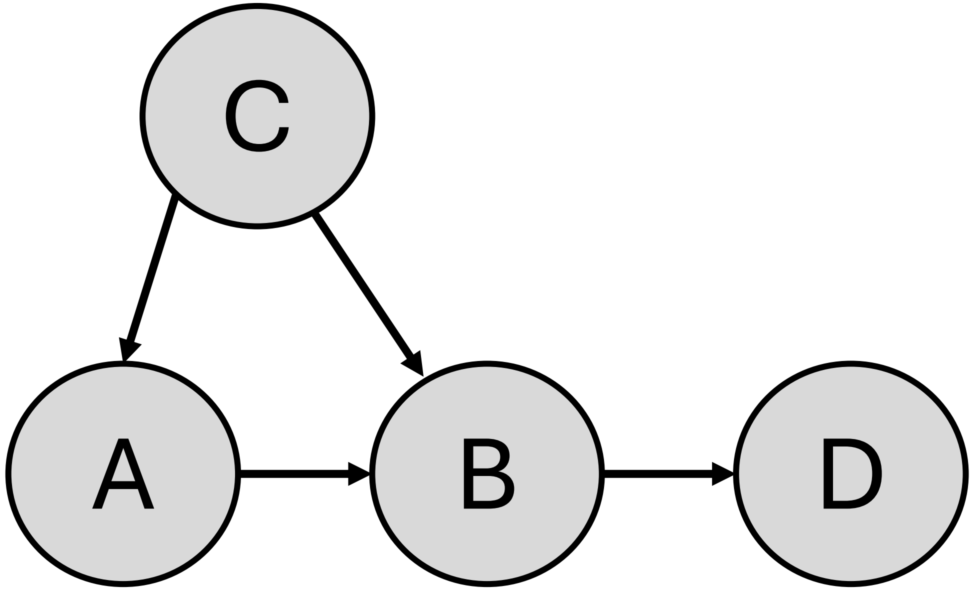
$\tilde{y}$ and $\tilde{t}$ refer to the unconfounded variation in Y and T respectively after conditioning on w.

- Stage 2: A final regression of $\tilde{y}$ on $\tilde{t}$ gives the causal effect.

$$\tilde{y} \sim \beta \tilde{t} + \epsilon$$



Second key application for causal ML:
Root cause attribution



Causal effect can be identified with the just the graph structure.

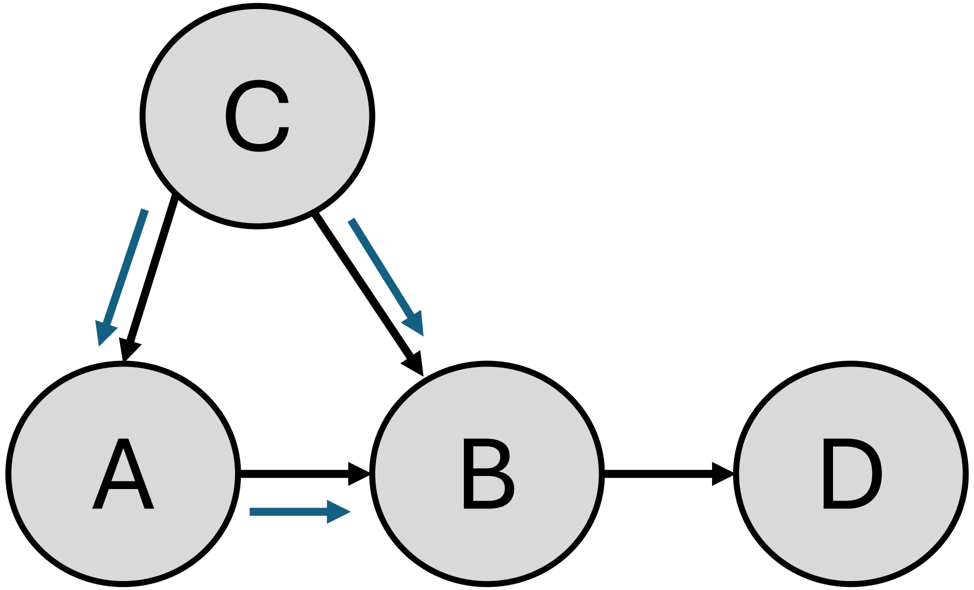
- Effect of A on B: $P(B|do(A)) = \sum_c P(B|A,C)\,P(C)$

For attribution, need to estimate the **counterfactual.**

Q: Given that $B = b$ and $C = c$, **how would B change if C was changed?**

$$P(B_{c'}\,|B = b, C = c, A = a, do(C = c'))$$

Second key application for causal ML: **Root cause attribution**



Q: Given that $B = b$ and $C = c$, **how would B change if C was changed?**

$$P\left(B_{c'} \middle| B = b, C = c, A = a, do(C = c')\right)$$

If only do-intervention,

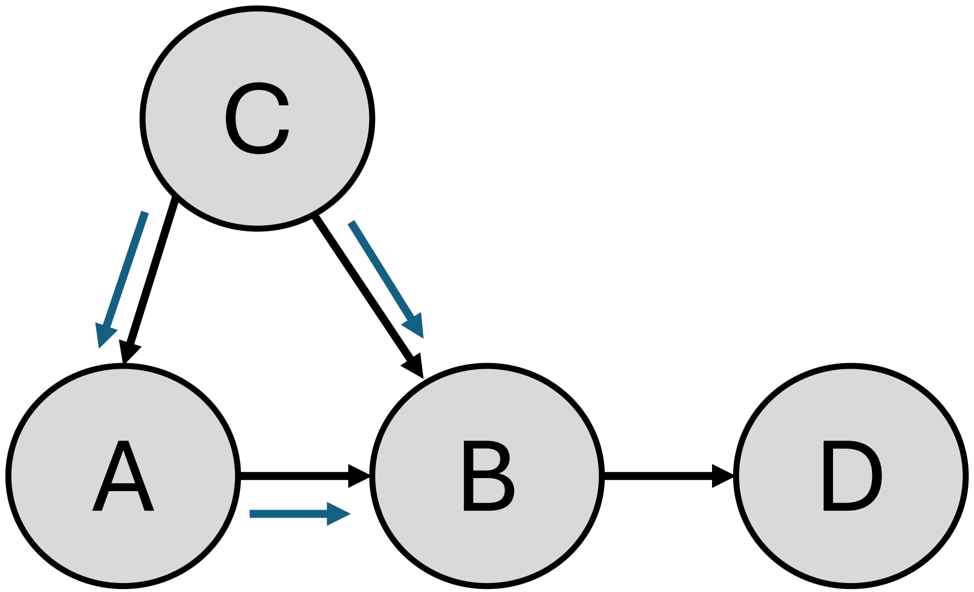
$P(B|do(C = c')) = P(B|C)$

But we also know that B=b, C=c, A=a in observed data.

That constrains the value of B'.

=> We need to know the **functional relationships** between A,B,C too.

The **counterfactual generation** algorithm



SCM: $A = g(C) + \epsilon_1;\ B = f(A, C) + \epsilon_2$

1. Abduction: Infer values of $\boldsymbol{\epsilon}$ using observed data.

$$\epsilon_1^* = a - g(c); \epsilon_2^* = b - f(a, c)$$

2. Action: Set C=c'.

3. Prediction: Now propagate the change downstream through graph.

$$a' = g(c') + \epsilon_1^*$$
$$b' = f(a', c') + \epsilon_2^*$$

Root cause attribution: Ranking over counterfactuals

- Using counterfactuals, we can now simulate the effect of different causes for an outcome.

  $P(Y_{X1}), P(Y_{X2}), P(Y_{X3}), \ldots$

  For attribution, we can rank the counterfactual effect of each cause.

  Can also average wrt. the values of all other causes
  (e.g., using Shapley value)

  Challenge: functional form is often unknown.
  Practical usecase in attributing outcomes of a ML model.

Example: Consider a classification ML model over face images

Meryem Öztürk

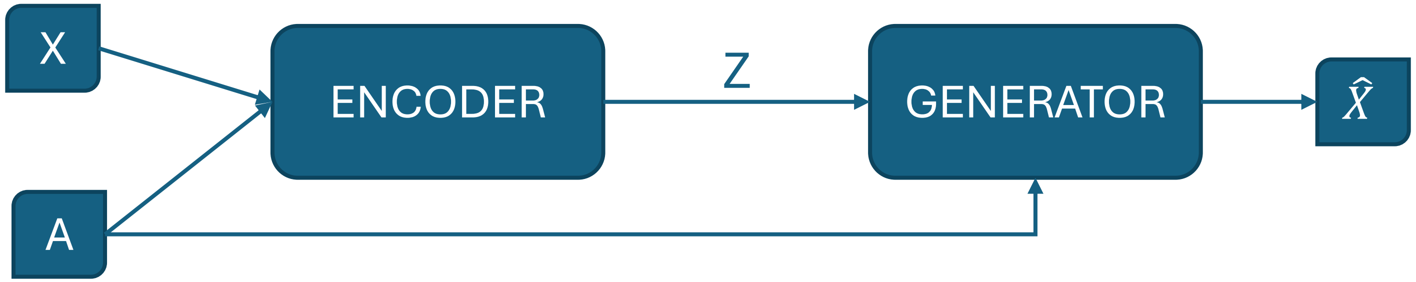
- Let’s say there is a trained binary classification model using the image.
- It outputs class=1 for image 1. **Why?**
  - We may look at important features through ML explanation models like LIME, SHAP etc.
  - But those do not tell us how the model will behave if we change the input.

How to generate a counterfactual for a ML model

Tabular data: Simply change the input name.

Image data: Need an encoder-generator architecture.

$z = E(x, a)$; Counterfactual$_{(A=a')}$ = $G(z, a')$



Train using Adversarially Learnt Inference [Dumoulin et al. 2016]

$$\min_{G,E}\max_{D} V(G,E,D) = E[\log(D(x,E(x,a),a)] + E[1 - D(G(z,a),z,a]$$

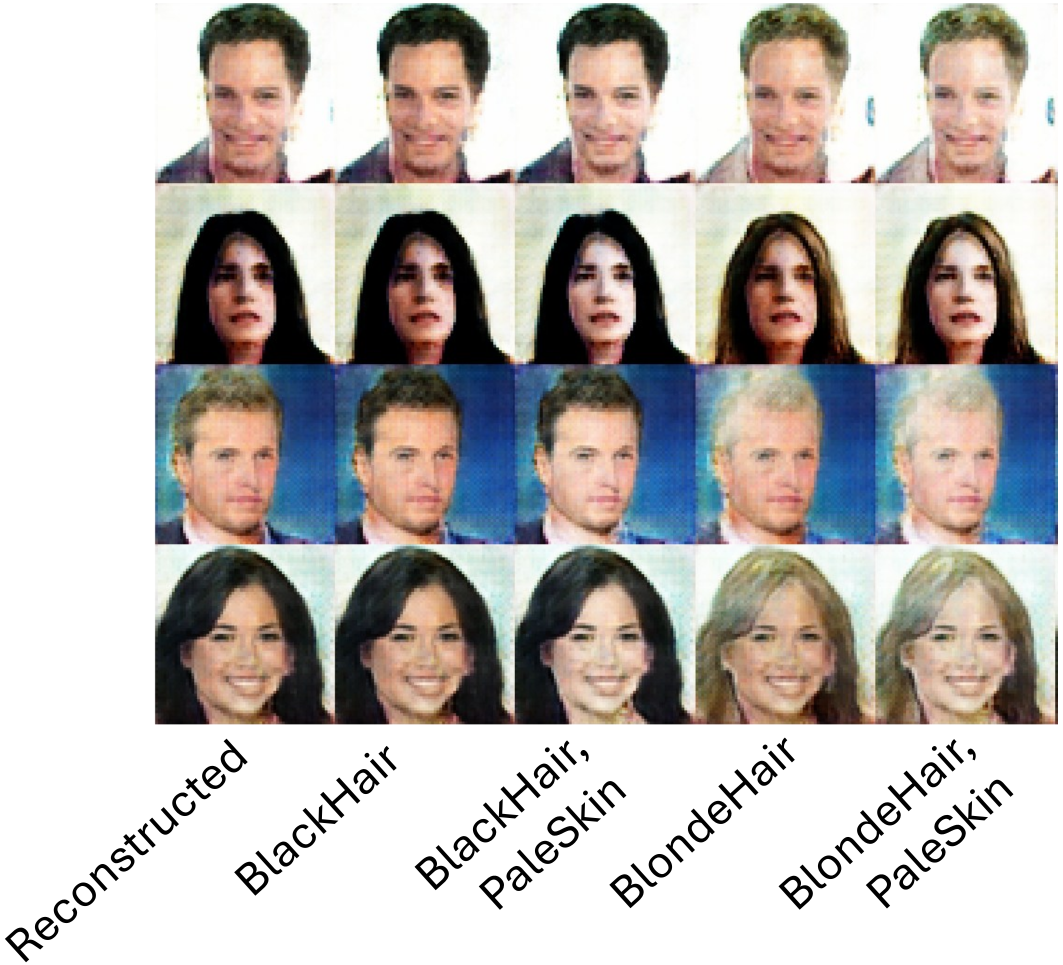
Evaluating a ML classifier on a Face dataset

CelebA dataset (Photos with 40 attributes like hair color, skin color, etc.)

Generate counterfactuals for each image.

Given a CNN classifier for one of the attributes ("attractiveness"), trained using standard loss minimization.

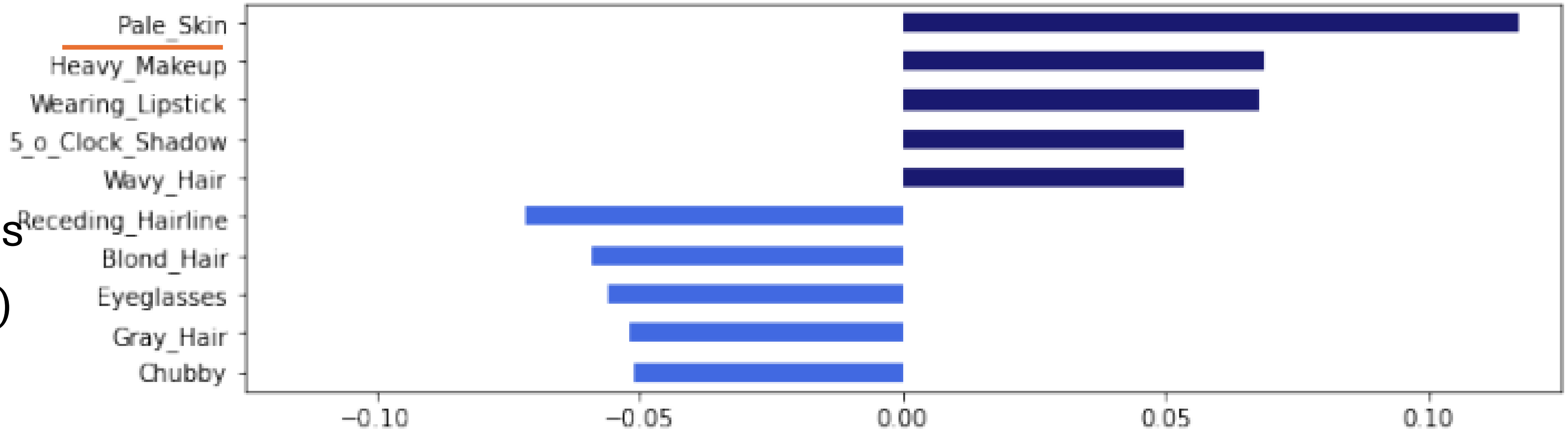
- **Explain:** Which attributes are considered important for prediction?
- **Fairness:** Is it fair wrt. certain attributes (*pale skin* attribute)?



Explain:

Feature importance scores

$$f\left(X_{A_i=a'}\right) - f(X)$$



Fairness:

Bias wrt. different features

$$\mathrm{P}\left(f\left(X_{A_i=a'}\right) = 1, f(X) = 0\right) - \mathrm{P}\left(f\left(X_{A_i=a'}\right) = 0, f(X) = 1\right)$$

	$p(a_r \neq a_c)$	$p(0 \rightarrow 1)$	bias
Horizontal_Flip	0.071	0.496	0.000
Brightness	0.137	0.619	0.033
black_h	0.067	0.838	0.045
black_h, pale	0.177	0.996	**0.175**
blond_h	0.090	0.115	**- 0.069**
blond_h, pale	0.095	0.773	**0.052**
brown_h	0.057	0.877	0.043
brown_h, pale	0.165	0.999	**0.165**
bangs	0.084	0.845	**0.058**

Third key application for causal ML: Out-of-distribution generalization

Domain generalization

Multiple domains: Assume access to data from multiple distributions

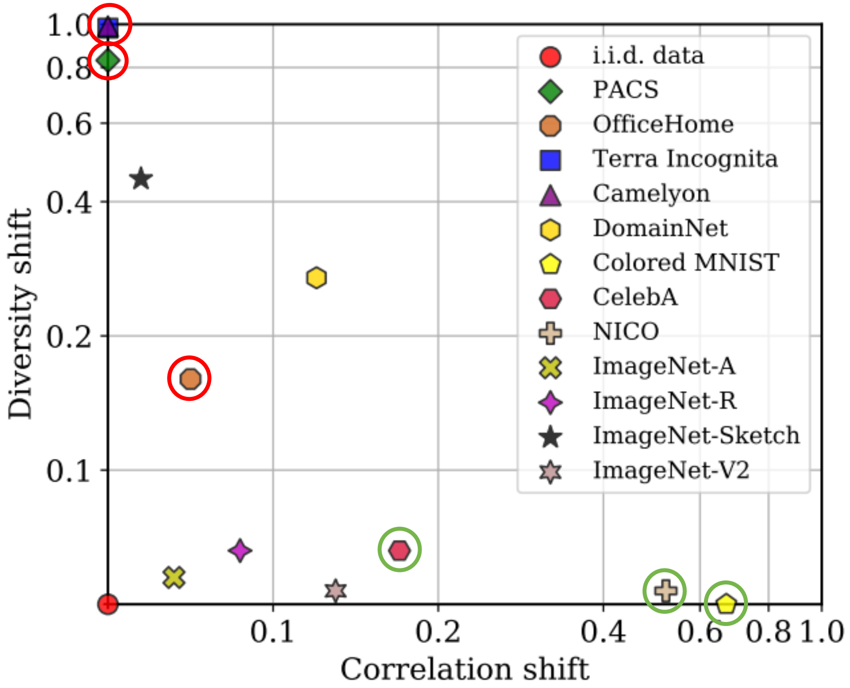
- Learn invariant patterns across the different sources
  - Invariant Risk Minimization (Arjovsky et al., 2019)
  - (Krueger et al. 2020, Ganin et al. 2016, Gulrajani & Lopez-Paz 2021, Nam et al. 2021)

Group generalization

Single domain: Assume access to group attributes for each input

- Equalize accuracy across groups/maximize worst-group accuracy
  - Group-DRO (Sagawa et al., 2020), (Ahmed et al. 2021)

Sobering state of SoTA algorithms

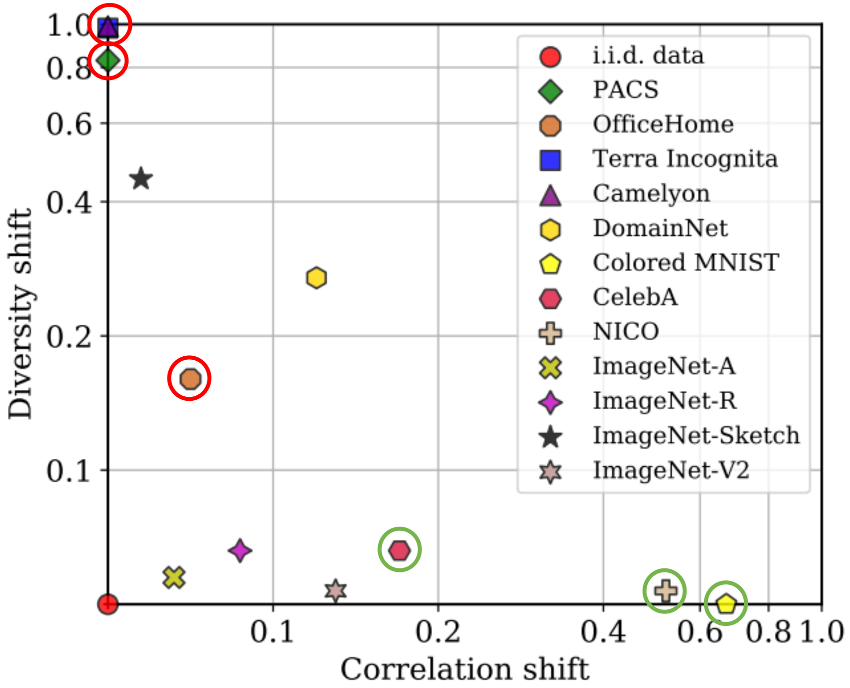


Ye et al., OoD-Bench, CVPR 2022

Sobering state of SoTA algorithms

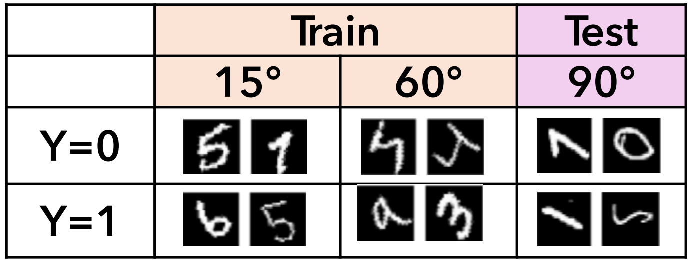
	Train		Test
	15°	60°	90°
Y=0			
Y=1			

Rotated MNIST

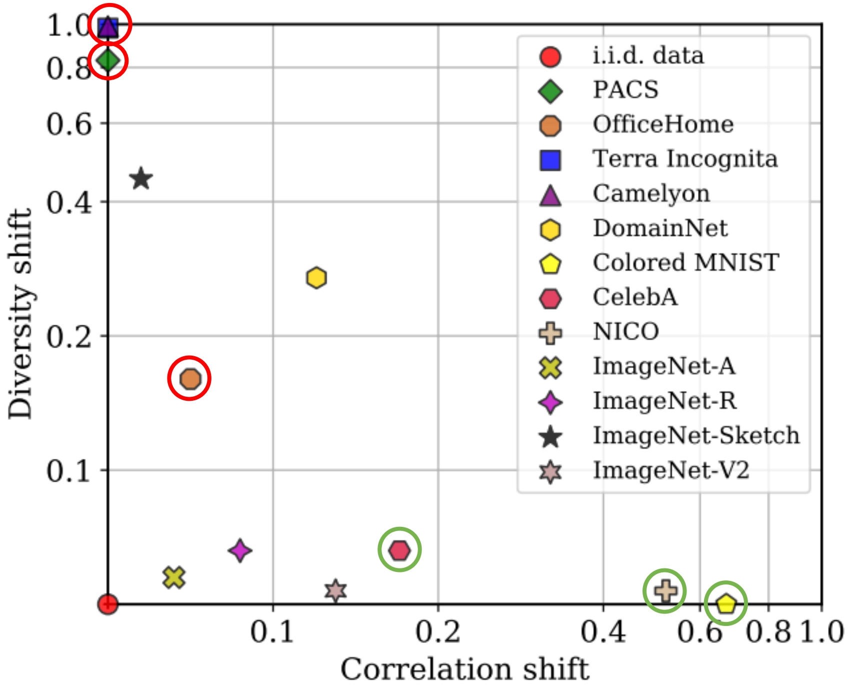


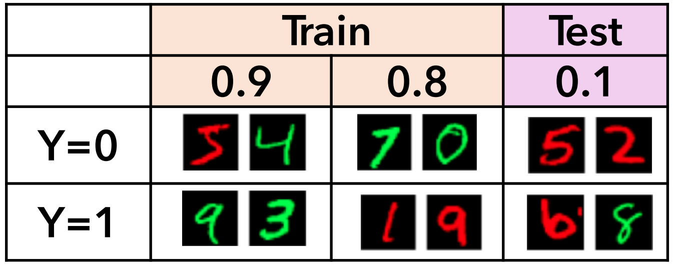
Ye et al., OoD-Bench, CVPR 2022

Sobering state of SoTA algorithms



Rotated MNIST

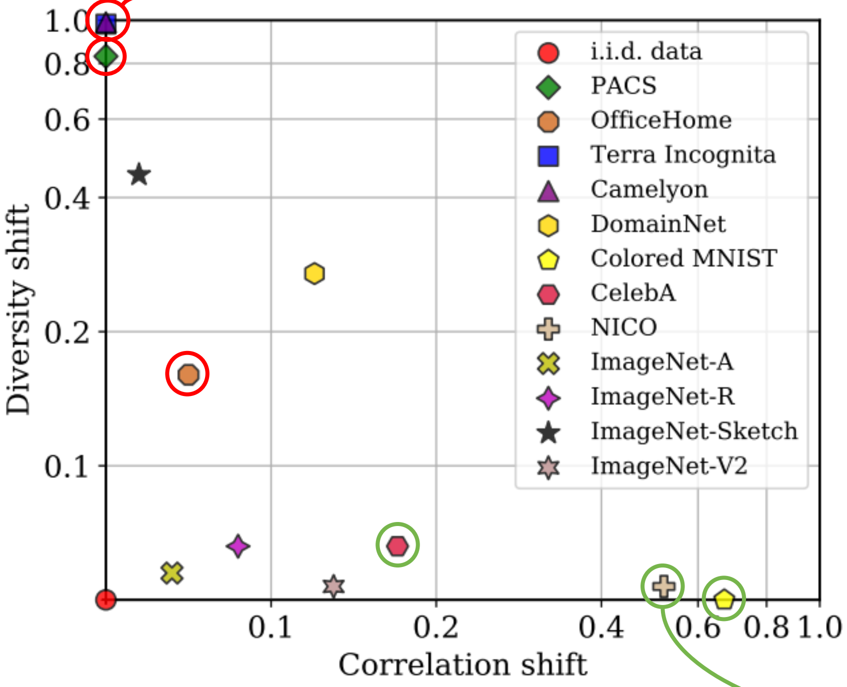




Colored MNIST

Ye et al., OoD-Bench, CVPR 2022

Sobering state of SoTA algorithms

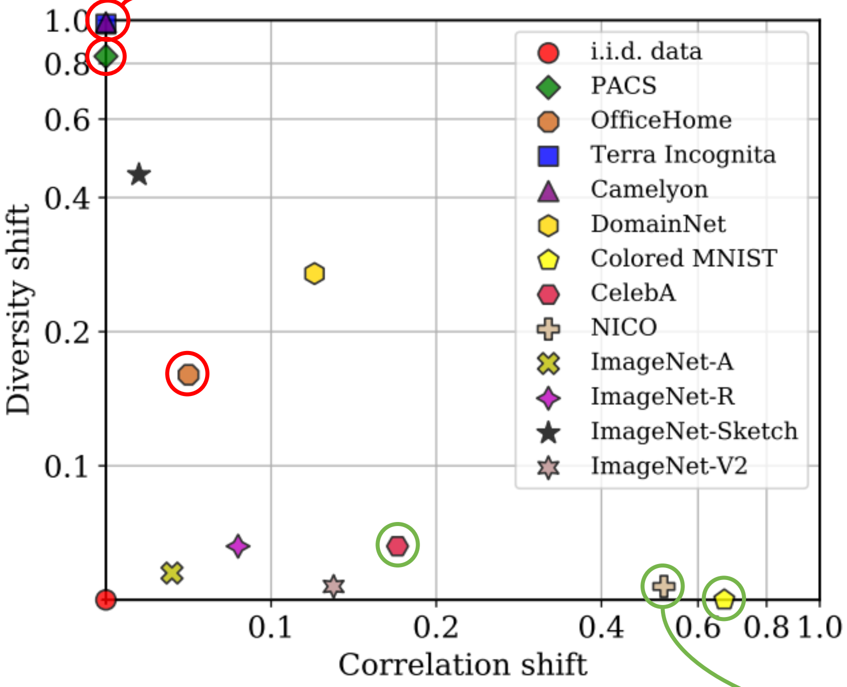


Algorithm	PACS	OfficeHome	TerraInc	Camelyon	Ranking score
MMD [42]	$81.7 \pm 0.2^{\uparrow}$	$63.8 \pm 0.1^{\uparrow}$	$38.3 \pm 0.4^{\downarrow}$	$94.9 \pm 0.4^{\uparrow}$	+2
ERM [69]	81.5 ± 0.0	63.3 ± 0.2	42.6 ± 0.9	94.7 ± 0.1	0
VREx [38]	$81.8 \pm 0.1^{\uparrow}$	63.5 ± 0.1	$40.7 \pm 0.7^{\downarrow}$	$94.1 \pm 0.3^{\downarrow}$	-1
GroupDRO [63]	$80.4 \pm 0.3^{\downarrow}$	63.2 ± 0.2	$36.8 \pm 1.1^{\downarrow}$	$95.2 \pm 0.2^{\uparrow}$	-1

Algorithm	Colored MNIST	CelebA	NICO	Prev score	Ranking score
VREx [38]	$56.3 \pm 1.9^{\uparrow}$	87.3 ± 0.2	71.0 ± 1.3	-1	+1
GroupDRO [63]	$32.5 \pm 0.2^{\uparrow}$	87.5 ± 1.1	71.8 ± 0.8	-1	+1
ERM [69]	29.9 ± 0.9	87.2 ± 0.6	71.4 ± 1.3	0	0
MMD [42]	$50.7 \pm 0.1^{\uparrow}$	$86.0 \pm 0.5^{\downarrow}$	$68.3 \pm 1.0^{\downarrow}$	+2	-1

Ye et al., OoD-Bench, CVPR 2022

Sobering state of SoTA algorithms



Algorithm	**PACS**	**OfficeHome**	**TerraInc**	**Camelyon**	**Ranking score**
MMD [42]	$81.7 \pm 0.2^{\uparrow}$	$63.8 \pm 0.1^{\uparrow}$	$38.3 \pm 0.4^{\downarrow}$	$94.9 \pm 0.4^{\uparrow}$	+2
ERM [69]	81.5 ± 0.0	63.3 ± 0.2	42.6 ± 0.9	94.7 ± 0.1	0
VREx [38]	$81.8 \pm 0.1^{\uparrow}$	63.5 ± 0.1	$40.7 \pm 0.7^{\downarrow}$	$94.1 \pm 0.3^{\downarrow}$	-1
GroupDRO [63]	$80.4 \pm 0.3^{\downarrow}$	63.2 ± 0.2	$36.8 \pm 1.1^{\downarrow}$	$95.2 \pm 0.2^{\uparrow}$	-1

No method can surpass ERM on all kinds of shifts!

Algorithm	**Colored MNIST**	**CelebA**	**NICO**	**Prev score**	**Ranking score**
VREx [38]	$56.3 \pm 1.9^{\uparrow}$	87.3 ± 0.2	71.0 ± 1.3	-1	+1
GroupDRO [63]	$32.5 \pm 0.2^{\uparrow}$	87.5 ± 1.1	71.8 ± 0.8	-1	+1
ERM [69]	29.9 ± 0.9	87.2 ± 0.6	71.4 ± 1.3	0	0
MMD [42]	$50.7 \pm 0.1^{\uparrow}$	$86.0 \pm 0.5^{\downarrow}$	$68.3 \pm 1.0^{\downarrow}$	+2	-1

Ye et al., OoD-Bench, CVPR 2022

Sobering state of SoTA algorithms

IID

[Correlation Shift]

Spurious correlation
b/w category and lighting

[Diversity Shift]

Unseen data shift
unseen azimuth values

Best methods are not consistent over different datasets and shifts

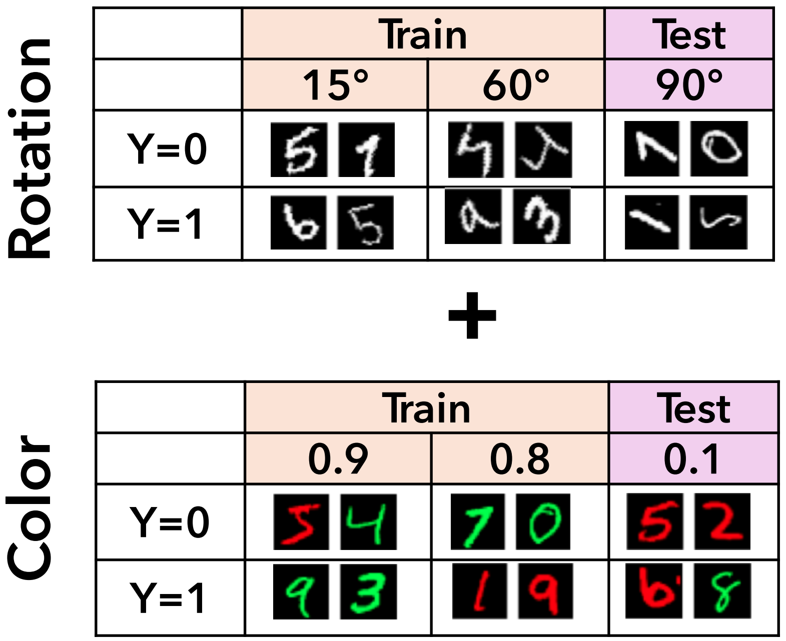
Wiles et al., ICLR 2022

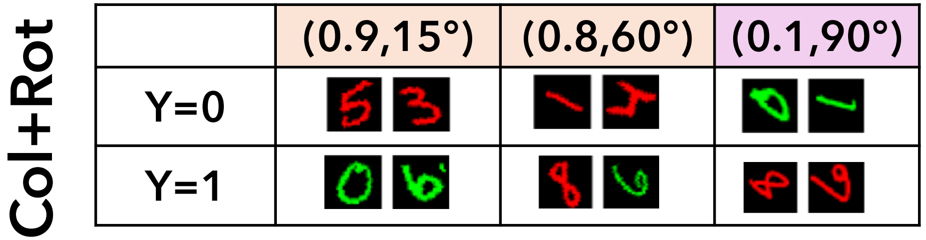
What if different distribution shifts co-exist?

	Train			Test	
Satellite Image (x)					
Year / Region (d)	2002 / Americas	2009 / Africa	2012 / Europe	2016 / Americas	2017 / Africa
Building / Land Type (y)	shopping mall	multi-unit residential	road bridge	recreational facility	educational institution

Koh et al., WILDS, ICML 2021

What if different distribution shifts co-exist?





Accuracy decreases further for all algorithms.

Algorithm	Color	Rotation	Col+Rot
ERM	30.9 ± 1.6	61.9 ± 0.5	25.2 ± 1.3
IRM	50.0 ± 0.1	61.2 ± 0.3	39.6 ± 6.7
MMD	29.7 ± 1.8	62.2 ± 0.5	24.1 ± 0.6
C-MMD	29.4 ± 0.2	62.3 ± 0.4	32.2 ± 7.0

I. Causal reasoning can explain this failure

[single shift] Explain results from causal perspective

- Different distribution shifts arise due to differences in data-generating process (DGP)
  - Leading to different independence constraints
- No single independence constraint can work for all shifts

II. Causal reasoning can provide a better algorithm

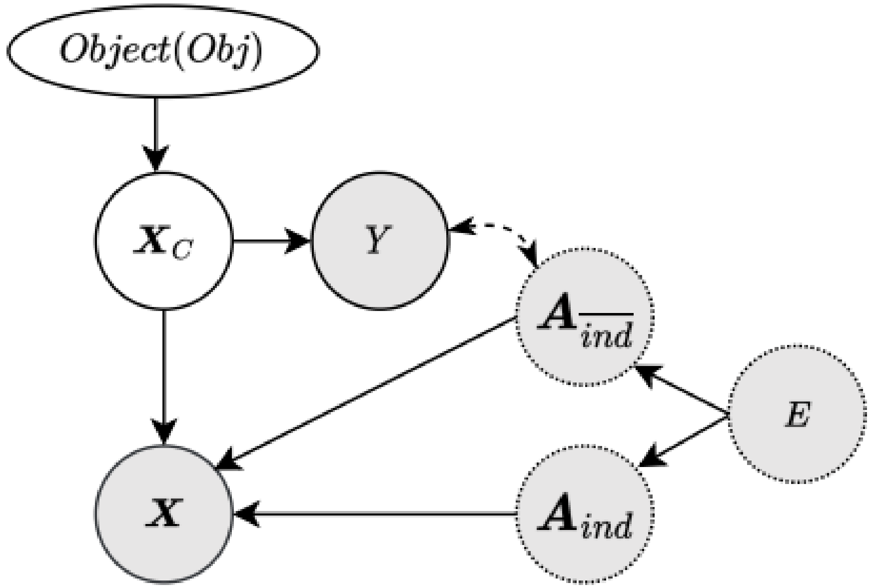
[single shift] Explain results from causal perspective

- Different distribution shifts arise due to differences in data-generating process (DGP)
  - Leading to different independence constraints
- No single independence constraint can work for all shifts

[multi-shift] Can we develop an algorithm that generalizes to individual as well as multi-attribute shifts?

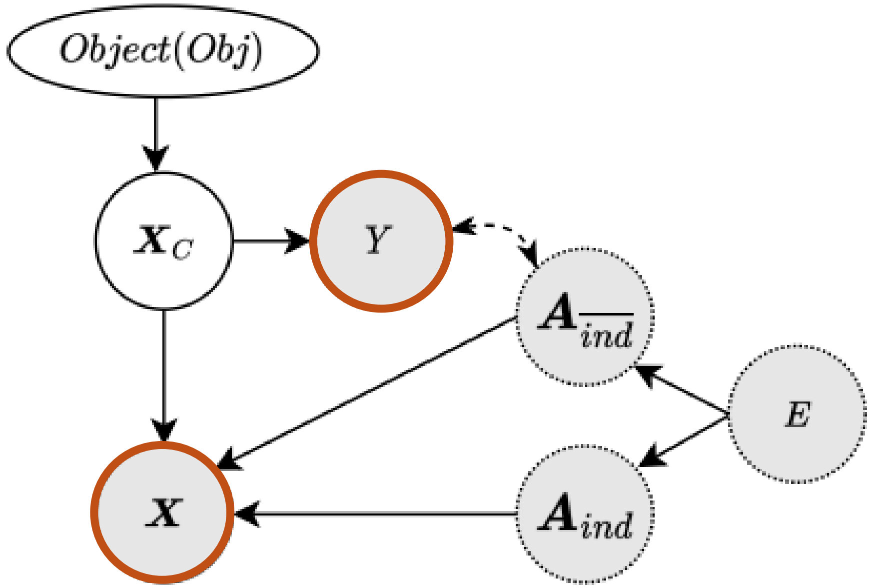
- We propose *Causally Adaptive Constraint Minimization (CACM)* to model the causal relationships in DGP

Representation of shifts using causal graph



Causal DAG to specify multi-attribute shifts

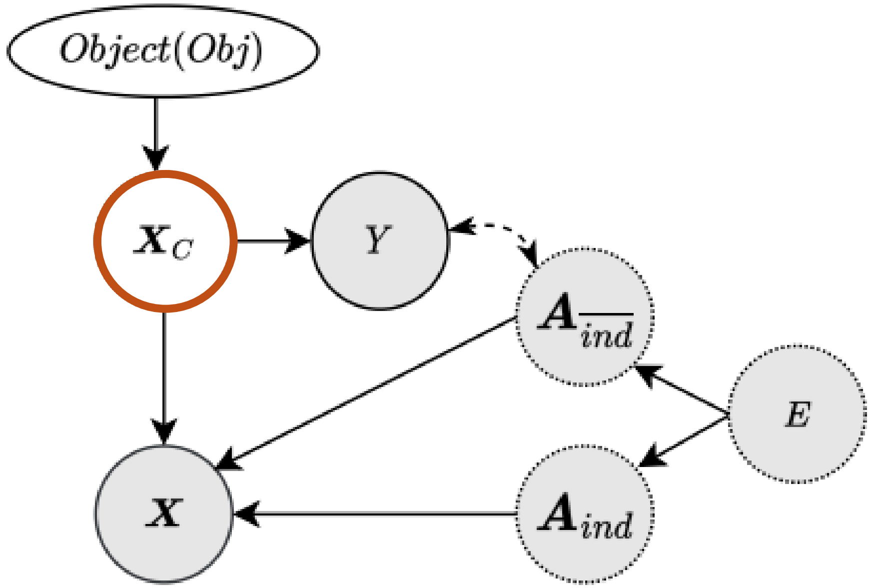
Representation of shifts using causal graph



Observed variables $\boldsymbol{X}, Y$

Causal DAG to specify multi-attribute shifts

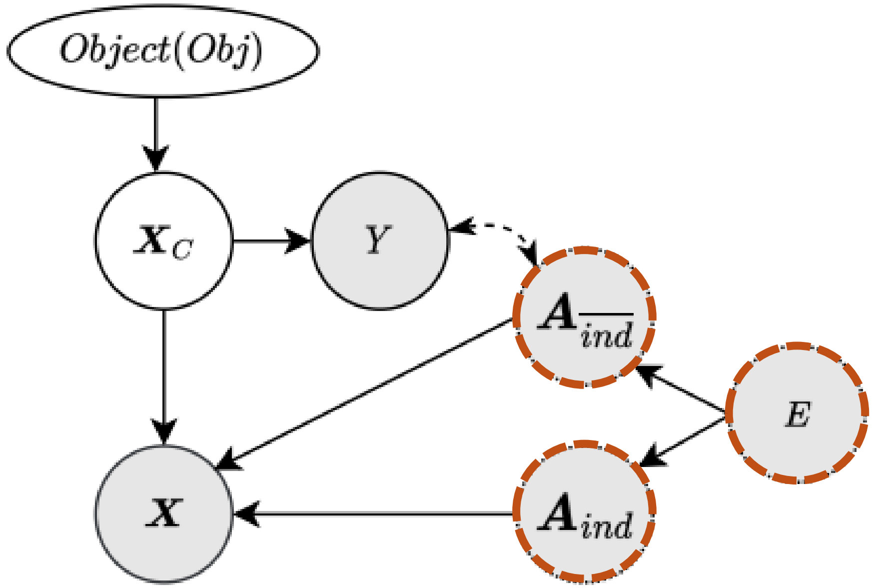
Representation of shifts using causal graph



Observed variables $\boldsymbol{X}, Y$
Causal features $\boldsymbol{X_c}$

Causal DAG to specify multi-attribute shifts

Representation of shifts using causal graph



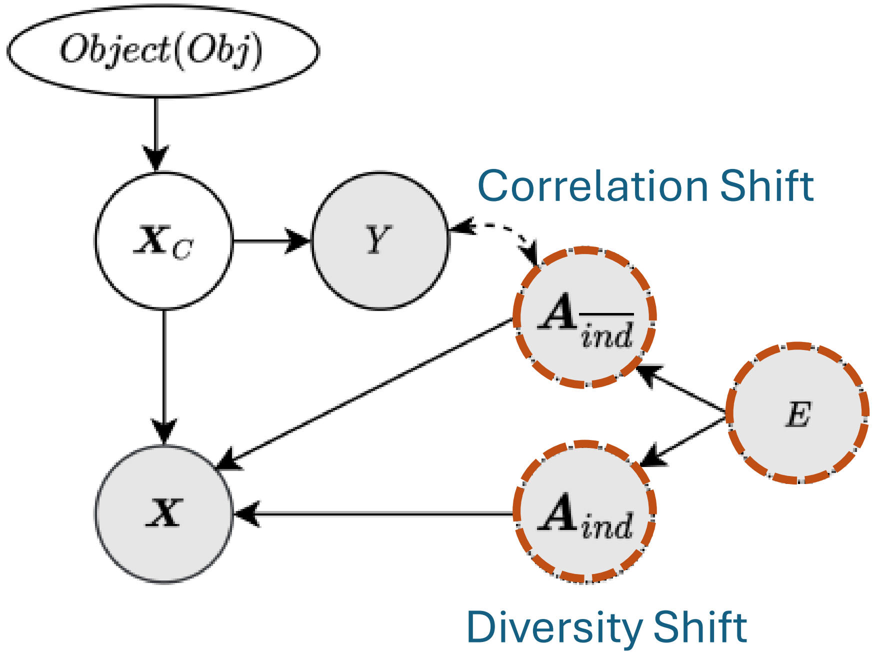
Causal DAG to specify multi-attribute shifts

Observed variables $\boldsymbol{X}, Y$

Causal features $\boldsymbol{X}_c$

Attributes $\boldsymbol{A}_{ind}, \boldsymbol{A}_{\overline{ind}}, E$ st $\boldsymbol{A}_{ind} \cup \boldsymbol{A}_{\overline{ind}} \cup \{E\} = \boldsymbol{A}$

Representation of shifts using causal graph



Causal DAG to specify multi-attribute shifts

Observed variables $\boldsymbol{X}, Y$

Causal features $\boldsymbol{X_c}$

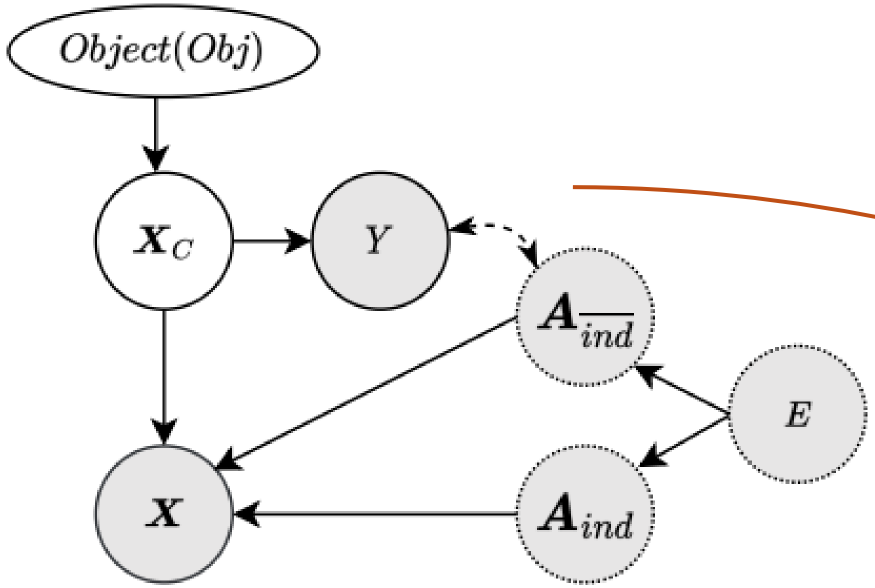
Attributes $\boldsymbol{A_{ind}}, \boldsymbol{A_{\overline{ind}}}, E$ st $\boldsymbol{A_{ind}} \cup \boldsymbol{A_{\overline{ind}}} \cup \{E\} = \boldsymbol{A}$

independent of label

correlated with label

domain attribute

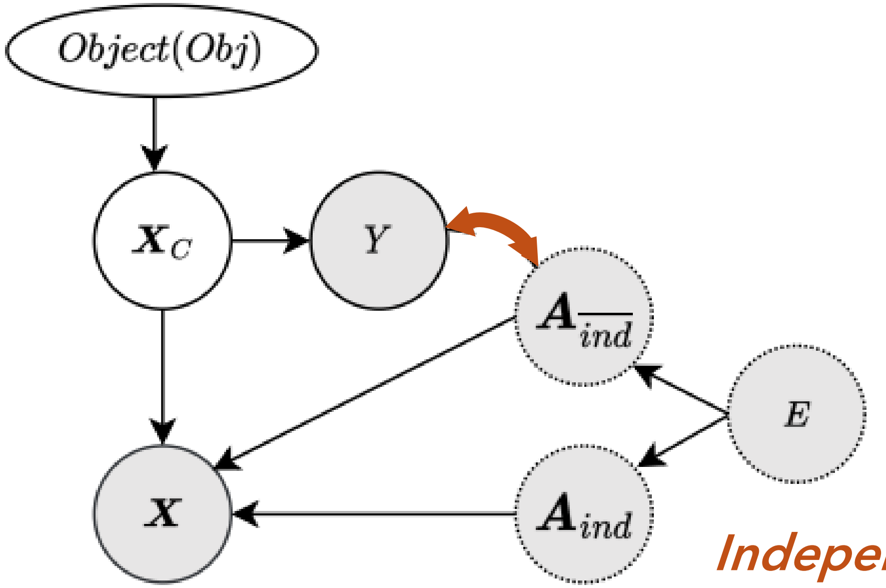
Representation of shifts using causal graph



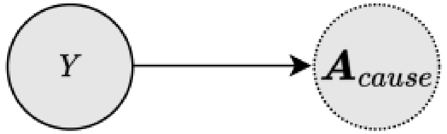
Causal DAG to specify multi-attribute shifts

Different $Y - \boldsymbol{A}_{\overline{ind}}$ relationships

Representation of shifts using causal graph



Independent

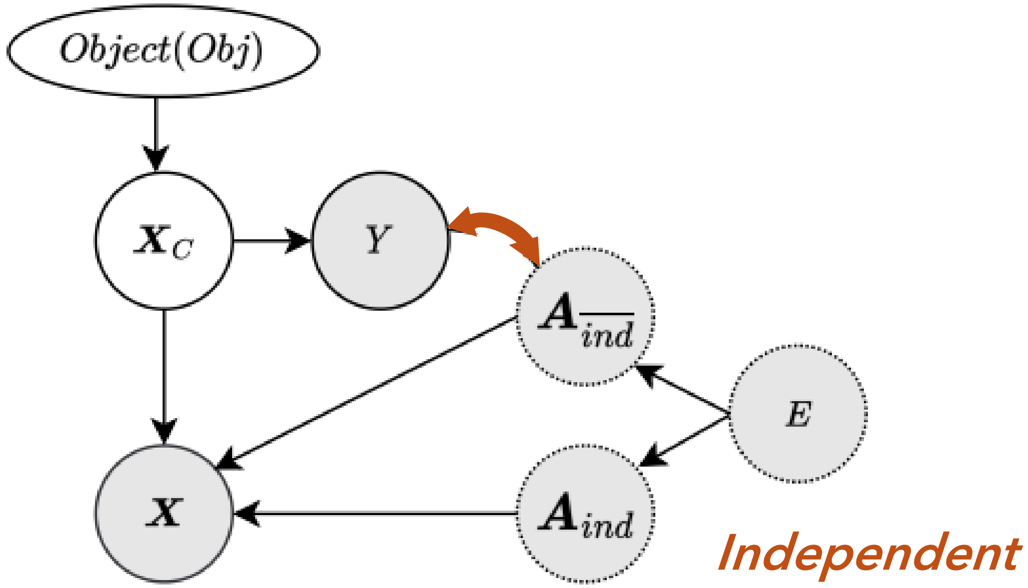


Causal

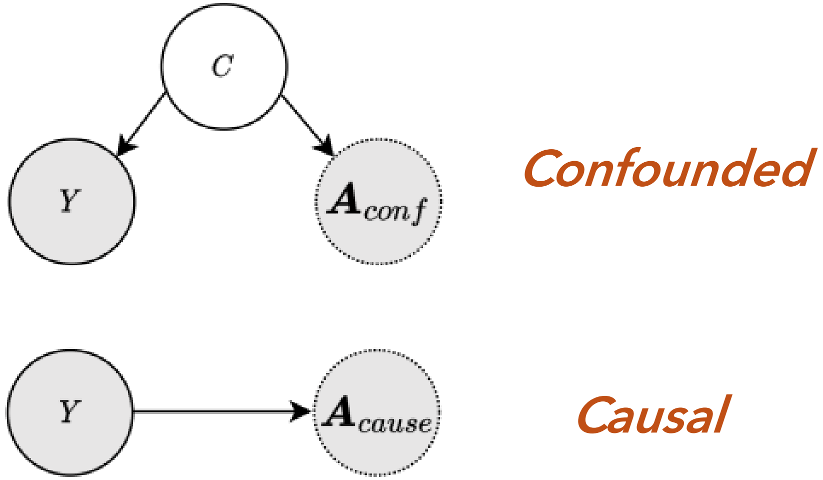
Causal DAG to specify multi-attribute shifts

Different $Y - \boldsymbol{A}_{\overline{ind}}$ relationships

Representation of shifts using causal graph

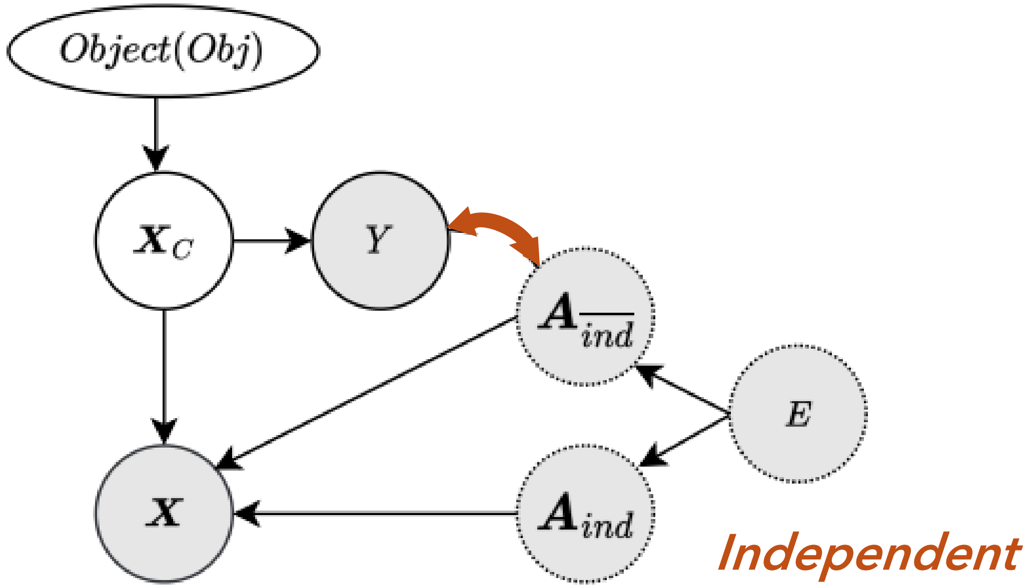


Causal DAG to specify multi-attribute shifts

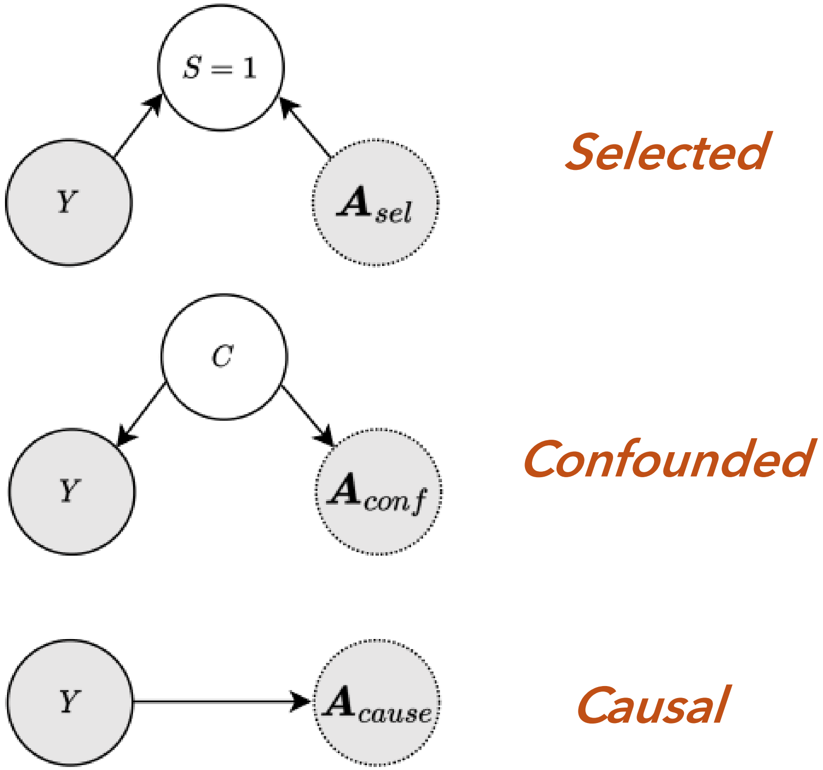


Different $Y - A_{\overline{ind}}$ relationships

Representation of shifts using causal graph



Causal DAG to specify multi-attribute shifts



Different $Y - A_{\overline{ind}}$ relationships

Back to the MNIST example

Rotation

	Train		Test
	15°	60°	90°
Y=0			
Y=1			

A_{ind}

+

Color

	Train		Test
	0.9	0.8	0.1
Y=0			
Y=1			

A_{cause}

$(A_{\overline{ind}})$

Col+Rot

	(0.9,15°)	(0.8,60°)	(0.1,90°)
Y=0			
Y=1			

Causal + Independent

$A_{cause} \cup A_{ind}$

Generalization to multi-attribute shifts

Algorithm	Color	Rotation	Col+Rot
ERM	30.9 ± 1.6	61.9 ± 0.5	25.2 ± 1.3
IRM	50.0 ± 0.1	61.2 ± 0.3	39.6 ± 6.7
MMD	29.7 ± 1.8	62.2 ± 0.5	24.1 ± 0.6
C-MMD	29.4 ± 0.2	62.3 ± 0.4	32.2 ± 7.0
CACM	**70.4 ± 0.5**	**62.4 ± 0.4**	**54.1 ± 0.3**

CACM outperforms on individual as well as combination of shifts

The *CACM* Approach

Identifying the correct regularizer under multi-attribute shifts

The *CACM* Approach

Identifying the correct regularizer under multi-attribute shifts

I. Derive correct independence constraints for $\boldsymbol{X_c}$ based on causal graph

II. Apply the constraints as regularizer to standard ERM loss.

Step I: Deriving independence constraints

Predictor $g(\boldsymbol{x}) = g_1(\phi(\boldsymbol{x}))$
Representation ϕ should follow same conditional independence constraints as $\boldsymbol{X_c}$

Mahajan et al., ICML 2021; Veitch et al., NeurIPS 2021; Makar et al., AISTATS 2022

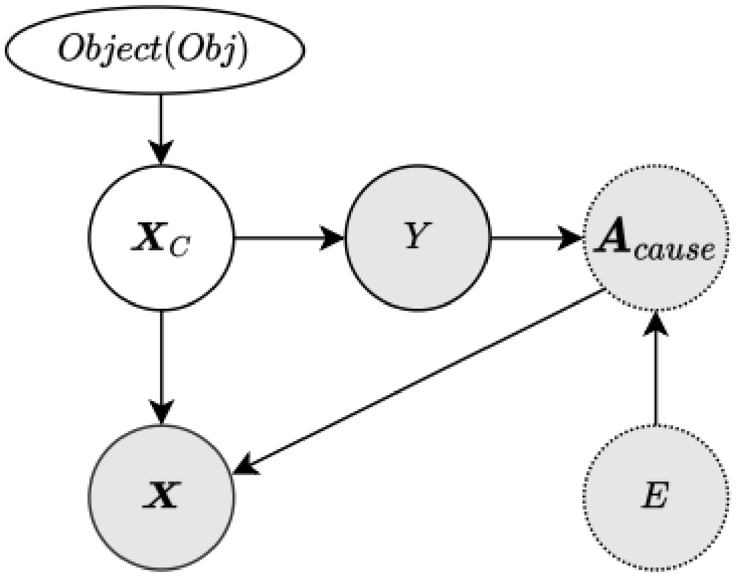
Step I: Deriving independence constraints

Predictor $g(\boldsymbol{x}) = g_1(\phi(\boldsymbol{x}))$

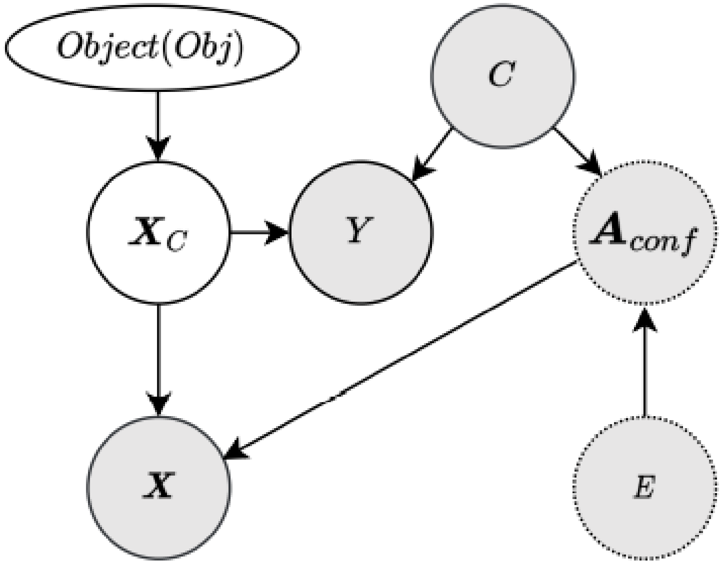
Representation ϕ should follow same conditional independence constraints as $\boldsymbol{X_c}$

Proposition 3.1. Given a dataset $(\boldsymbol{x}_i, \boldsymbol{a}_i, y_i)_{i=1}^{n}$ and a causal DAG over $\langle\ \boldsymbol{X_c}, \boldsymbol{X}, \boldsymbol{A}, Y\ \rangle$ such that $\boldsymbol{X_c}$ is the only variable (or set of variables) that causes Y and is not independent of $\boldsymbol{X}$, then the conditional independence constraints satisfied by $\boldsymbol{X_c}$ are necessary for a risk-invariant predictor.

Step I: Deriving independence constraints



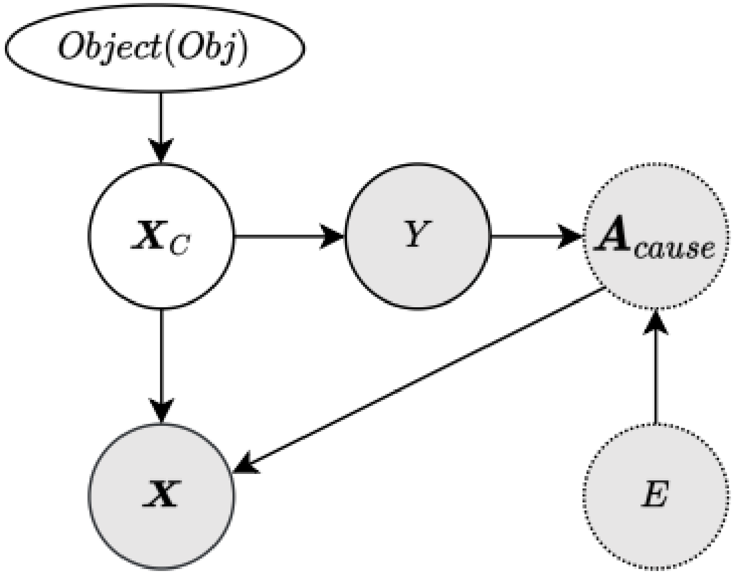
Causal



Confounded

Different $Y - \boldsymbol{A}_{\overline{ind}}$ relationships lead to different constraints

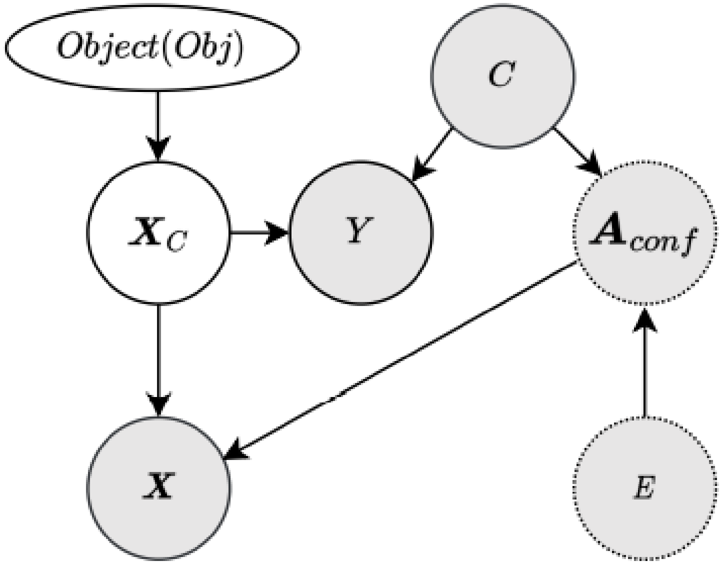
Step I: Deriving independence constraints



Causal

$X_c \perp\!\!\!\perp A_{cause} \mid Y, E$ ✓

$X_c \perp\!\!\!\perp A_{cause} \mid E$ ✗



Confounded

$X_c \perp\!\!\!\perp A_{conf} \mid Y, E$ ✗

$X_c \perp\!\!\!\perp A_{conf} \mid E$ ✓

Step II: Applying regularization penalty

Constraint: $\boldsymbol{X_c} \perp\!\!\!\perp \boldsymbol{A_{cause}} \mid Y, E$ [***Causal*** shift]

$$RegPenalty_{\boldsymbol{A}_{cause}} = \sum_{|E|} \sum_{y \in Y} \sum_{i=1}^{|\boldsymbol{A_{cause}}|} \sum_{j>i} \mathrm{MMD}\left(P\big(g_1(\phi(\boldsymbol{x}))\big|a_{i,cause}, y\big), P\big(g_1(\phi(\boldsymbol{x}))\big|a_{j,cause}, y\big)\right)$$

$$\boldsymbol{g_1}, \boldsymbol{\phi} = \mathrm{argmin}_{g_1,\phi}\ \mathrm{L}\big(g_1(\phi(\boldsymbol{x})), y\big) + \lambda^*\big(RegPenalty_{\boldsymbol{A}_{cause}}\big)$$

Summary of Session 1

- Causal ML is important whenever we have decision-making or attribution tasks, or want generalizability of predictive model beyond the training distribution.
- Causal graph is the most important assumption.
  - “No causes in, no causes out” – Judea Pearl
- The goal is to develop methods that use the least amount of assumptions.
  - E.g., debiased ML for effect estimation
  - Simple, high-level causal graph for images