

# Causality and its applications in Machine Learning

**Amit Sharma**

Microsoft Research India

[www.amitsharma.in](http://www.amitsharma.in)

[@amt\\_shrma](#)

# How I got interested in causality

- 2012: Building a recommender system at LinkedIn
  - “Groups you may want to join”
- Use historical user activity as training data
  - Training is till month  $t$ , test is after month  $t$
- Obtain accuracy numbers.
  - Suppose Algorithm A is better than B.
- Now deploy the recommendation algorithms A and B in a randomized experiment.
  - My manager: “You will be lucky if you find A performs better than B”
  - Why?

What is the **impact** of a recommender system on sales?

Can a recommender system **change** user's preferences?

## Recommender System

How to **explain** why an item was recommended to a user?

What would happen if we **changed** the recommender algorithm?

What is the **impact** of a recommender system on sales?

Can a recommender system **change** user's preferences?

These are all questions about action and effect.  
Need **causality** to answer them.

How to **explain** why an item was recommended to a user?

What would happen if we **changed** the recommender algorithm?

- I. What is causality?
- II. How can we reason about causality mathematically?  
From Bayesian Networks to Causal Bayesian Networks (causal DAGs)
- III. Can we learn a causal DAG?
- IV. Application 1: Estimating the effect of actions
- V. Application 2: Building more generalizable prediction models
- VI. Open questions

## I. What is causality?

## II. How can we reason about causality mathematically?

From Bayesian Networks to Causal Bayesian Networks (causal DAGs)

## III. Can we learn a causal DAG?

## IV. Application 1: Estimating the effect of actions

## V. Application 2: Building more generalizable prediction models

## VI. Open questions

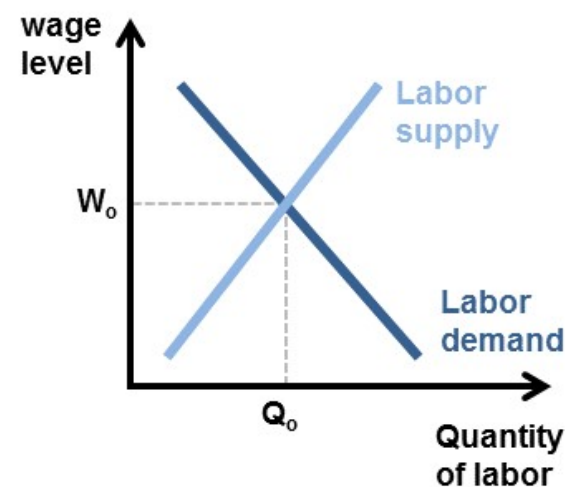
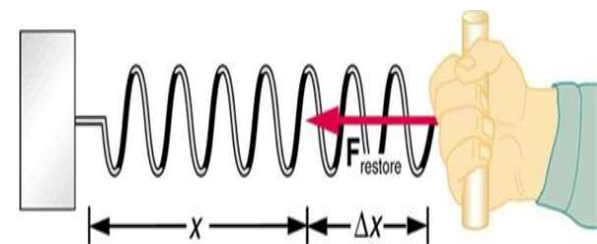
# The Three Layer Causal Hierarchy

Pearl, Theoretical Impediments to Machine Learning with Seven Sparks from the Causal Revolution, arXiv:1801.04016v1. 11 Jan 2018

Level	Typical Activity	Typical Question	Examples
1. <b>Association</b> $P(y   x)$	<i>Seeing</i>	What is? How would seeing $X$ change my belief in $Y$ ?	What does a symptom tell me about a disease? What does a survey tell us about the election results?
2. <b>Intervention</b> $P(y   do(x), z)$	<i>Doing, Intervening</i>	What if? What if I do $X$ ?	What if I take aspirin, will my headache be cured? What if we ban cigarettes?
3. <b>Counterfactuals</b> $P(y_x   x', y')$	<i>Imagining, Retrospection</i>	Why? Was it $X$ that caused $Y$ ? What if I had acted differently?	Was it the aspirin that stopped my headache? What I had not been smoking the past 2 years?

# Intervention and counterfactuals

- Questions of interventions and counterfactuals common in biomedical and social sciences
- Such questions form the basis of almost all scientific inquiry
  - Medicine: drug trials, effect of a drug
  - Social sciences: effect of a certain policy
  - Genetics: effect of genes on disease
- **So what is causality?**
- **What does it mean to *cause* something?**





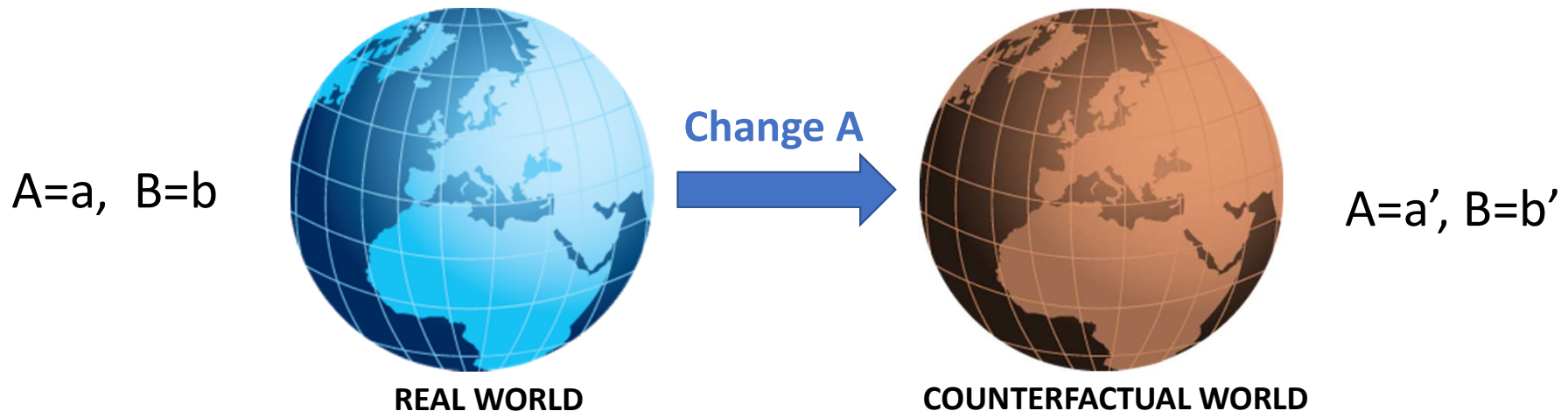
# Defining causality

A longstanding philosophical debate.

“Interventionist” definition of causality:

*A causes B if and only if*

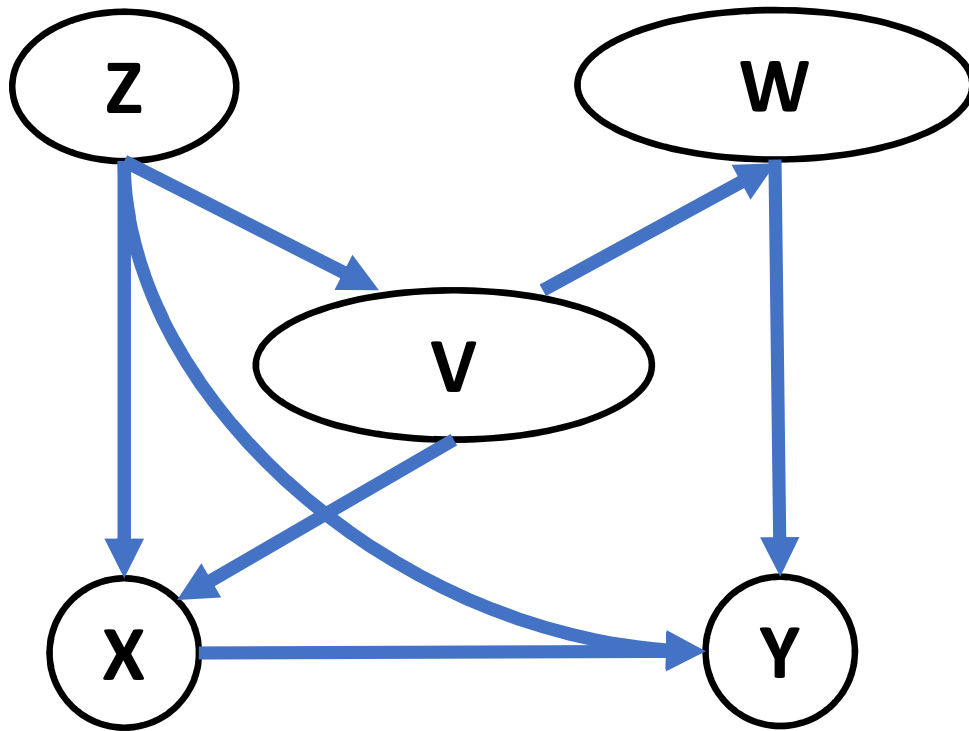
*B would change if an appropriate manipulation on A were to be done.*



[Causation and Manipulability \(Stanford Encyclopedia of Philosophy\)](#)

- I. What is causality?
- II. How can we reason about causality mathematically?**  
From Bayesian Networks to Causal Bayesian Networks (causal DAGs)
- III. Can we learn a causal DAG?
- IV. Application 1: Estimating the effect of actions
- V. Application 2: Building more generalizable prediction models
- VI. Open questions

## Recap: Bayesian Networks

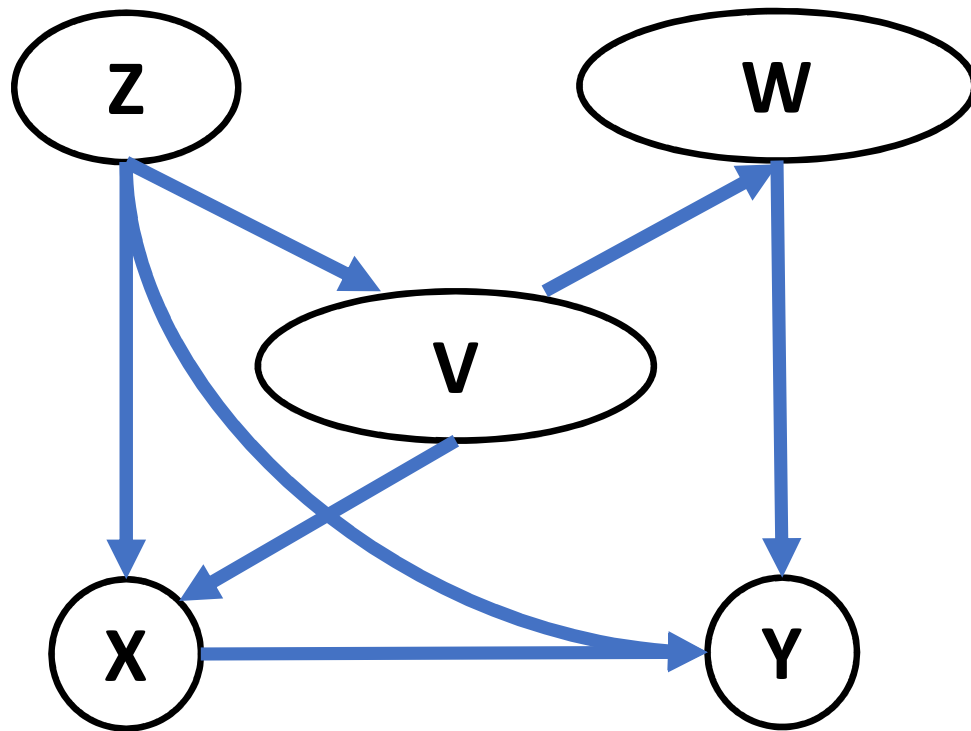


$$\begin{aligned} P(V, W, X, Y, Z) \\ = P(Y|X, W, Z) P(X|V, Z) \\ P(W|V) P(V|Z) P(Z) \end{aligned}$$

Provides a factorization of the joint probability.

Each of the nodes' conditional probability can be estimated independently.

# Causal Bayesian Networks



Each directed edge corresponds to a **causal** relationship.

W causes Y.

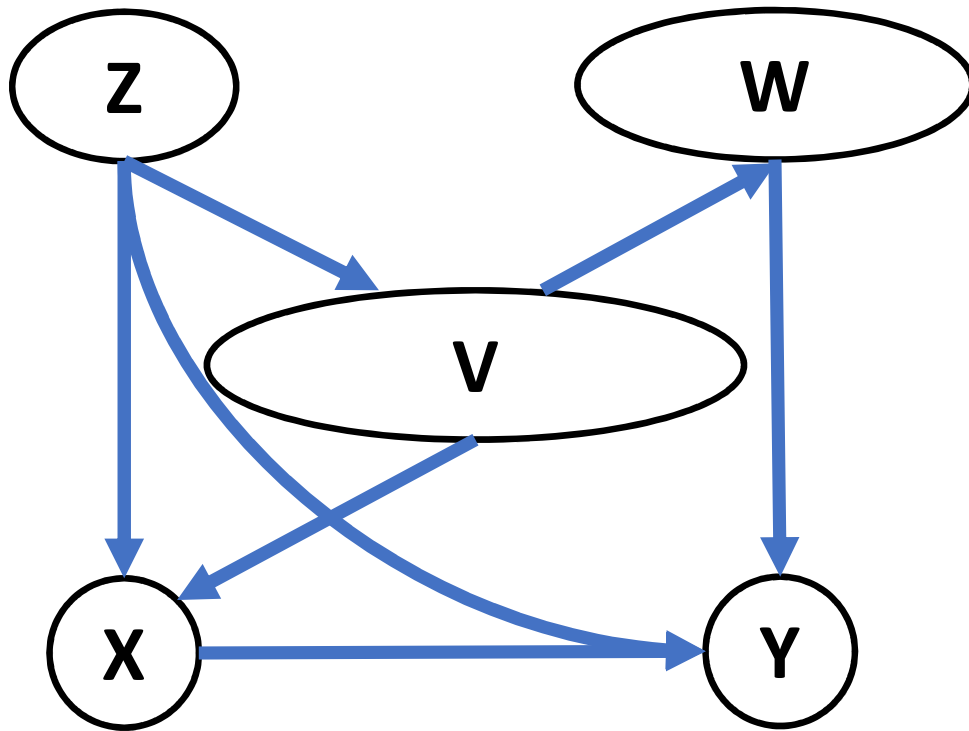
Z causes V, X, and Y.

Provides an explicit **data-generation process**.

Changing Y should not affect W.

Changing Z will affect V.

Important: Assumptions are the edges that are *missing*



**Assumption 1:** W does affect outcome Y.

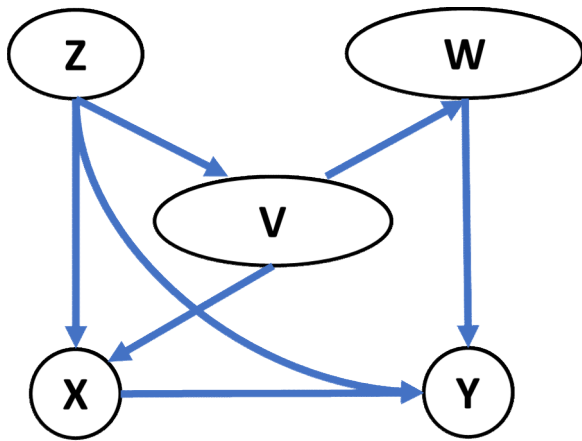
**Assumption 2:** Z does not affect W.

**Assumption 3:** W does not affect V.

**Assumption 4:** X does not affect W.

*..and so on.*

# Structural Equations: An equivalent representation for causal Bayesian networks



$$P(.) = P(Y|X, W, Z) P(X|V, Z) \\ P(W|V) P(V|Z) P(Z)$$

$$y := f(x, w, z, \epsilon_y) \\ x := g(v, z, \epsilon_x) \\ w := h(v, \epsilon_w) \\ v := i(z, \epsilon_v)$$

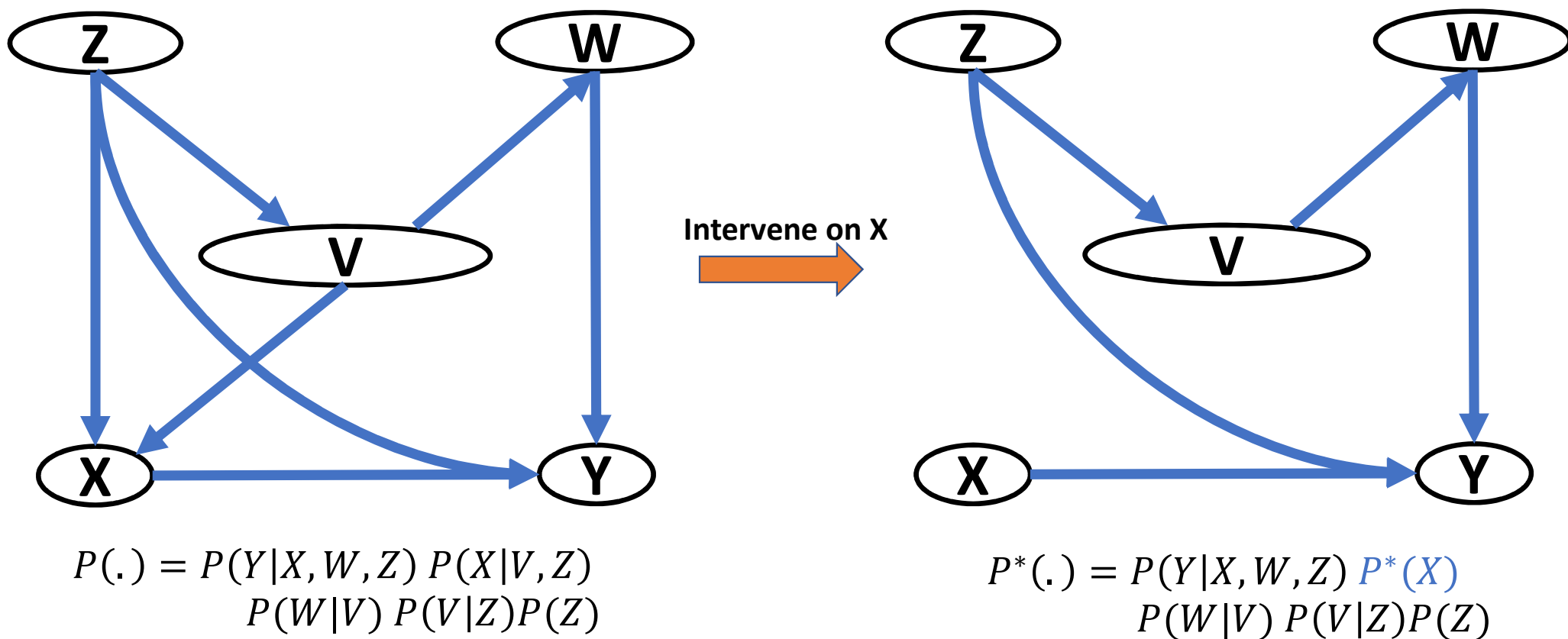
**Independence of cause and mechanism (ICM) assumption**

$f, g, h$  and  $i$  are independent functions.

All  $\epsilon$  error terms are mutually independent.

# Now we are ready to define an intervention

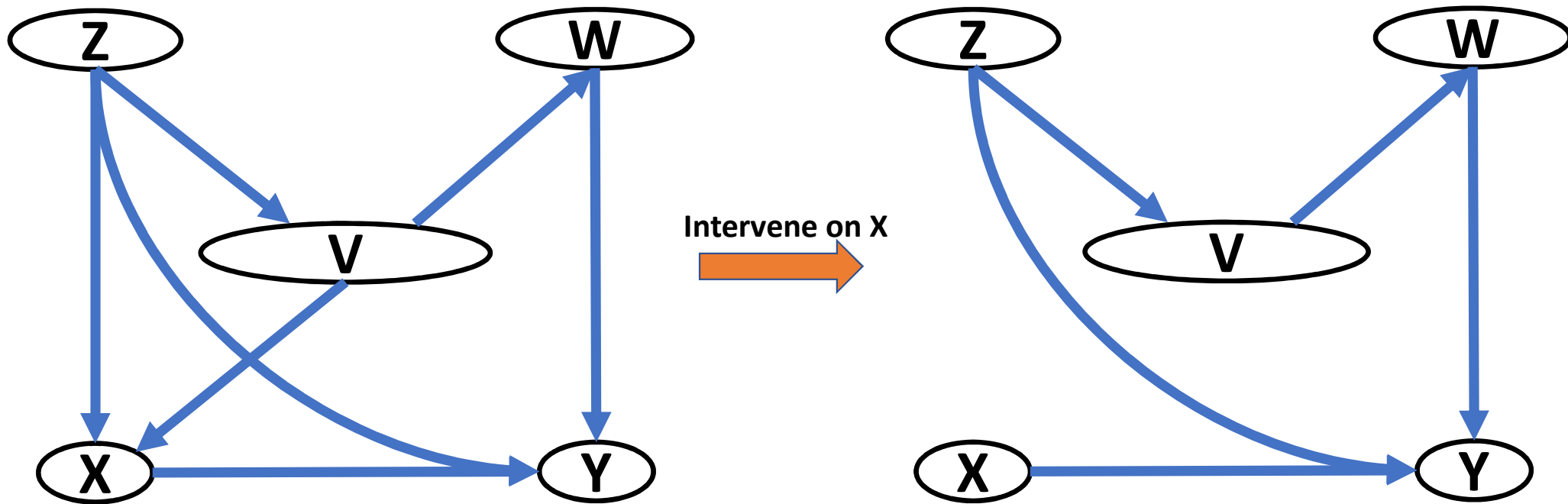
An intervention on a variable changes its generating process.  
Removes its existing relationship with its parents.



# Now we are ready to define an intervention

An intervention on a variable changes its generating process.

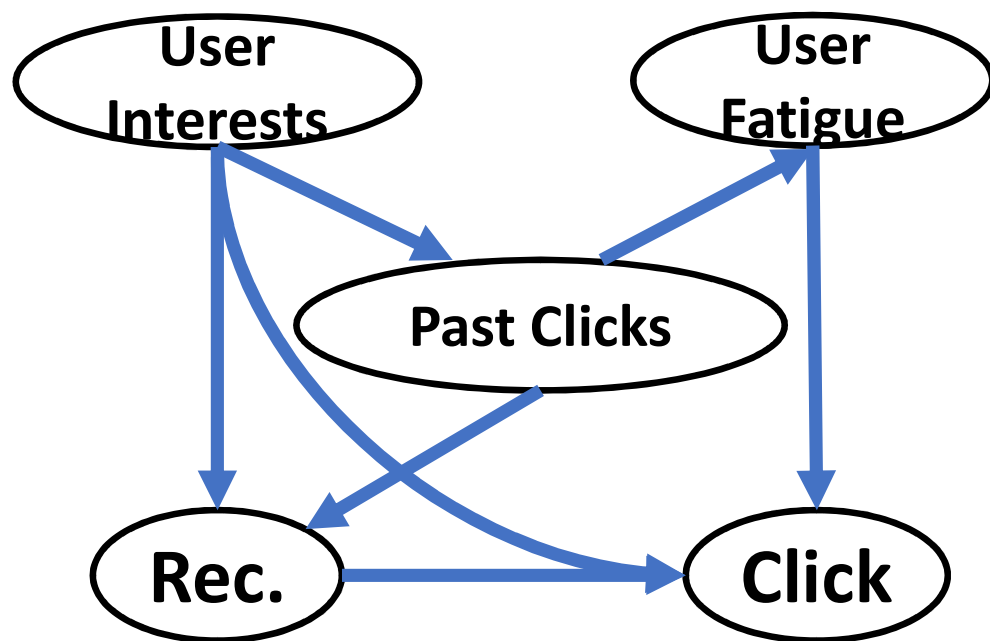
Removes its existing relationship with its parents.



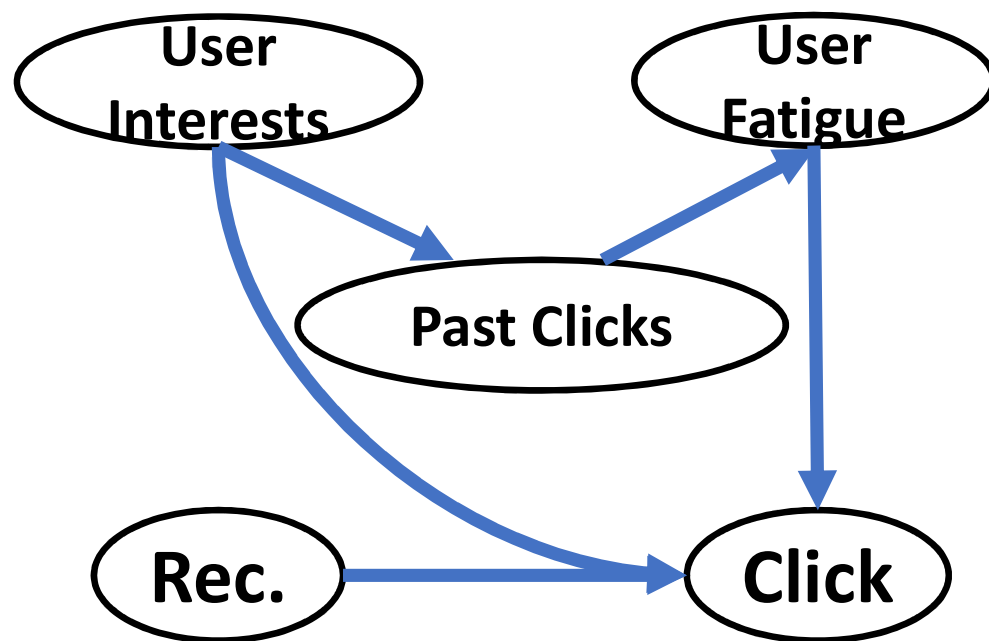
**Causal Effect:**  $P(Y|\text{do}(X = x)) = P^*(Y|X = x) \neq P(Y|X = x)$



Example: How much does showing recommendations impact product discovery for users?



$P(\text{Click}|\text{Rec.})$  is confounded by the effect of User Interests.

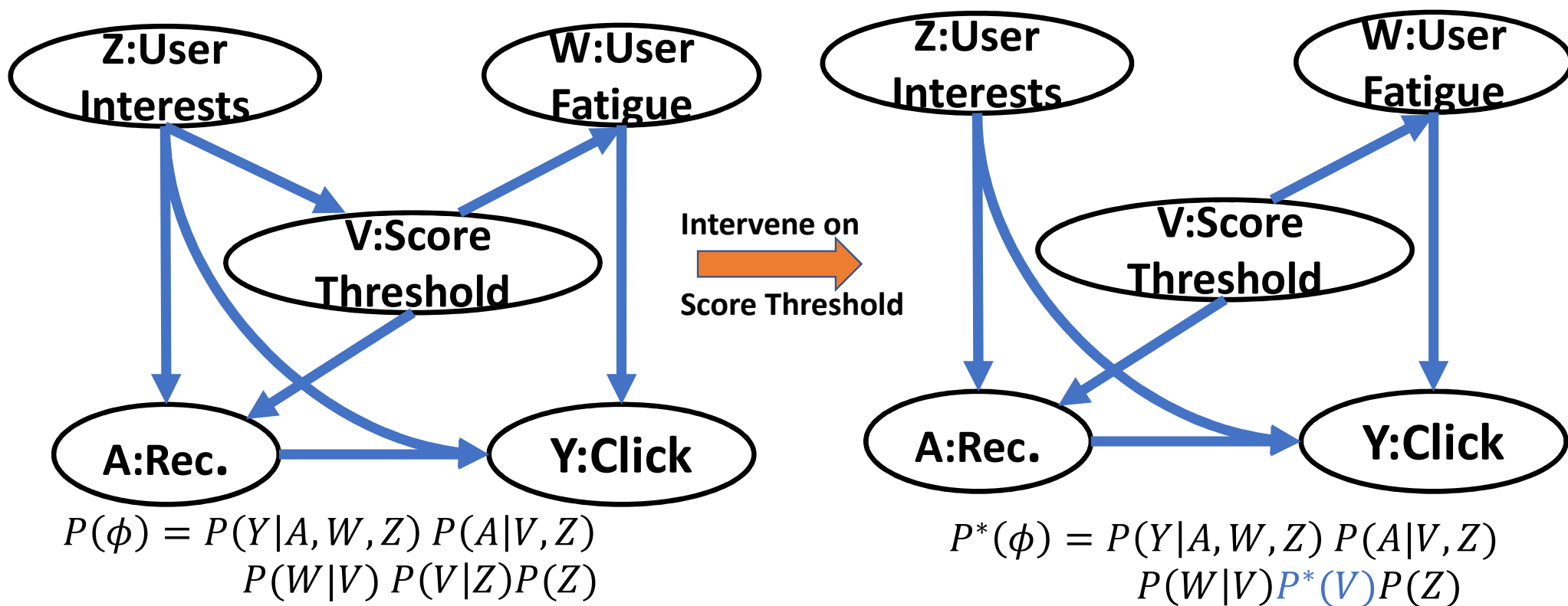


Instead, estimate  $P(\text{Click}|\text{do}(\text{Rec.})) = P^*(\text{Click}|\text{Rec.})$  where Rec. are generated independently of User Interests.

# Power of causal Bayesian networks: Can estimate the result of interventions

Suppose  $V$  is a system parameter. Often is changed by systems team.

Build predictor for different interventions on  $V$ .



$$P(\phi) = \frac{P(Y|X, W, Z) P(X|V, Z)}{P(W|V) P(V|Z) P(Z)}$$

$$P^*(\phi) = \frac{P(Y|A, W, Z) P(A|V, Z)}{P(W|V) P^*(V) P(Z)}$$

$$\begin{aligned} E_{P^*}[Y] &= E_{\phi \sim P^*(\Phi)}[y] \\ &= \int_{\phi} y P^*(\phi) \\ &= \int_{\phi} y \frac{P(\phi)}{P(\phi)} P^*(\phi) = \int_{\phi} y \frac{P^*(\phi)}{P(\phi)} P(\phi) \\ &= \int_{\phi} y \left[ \frac{P^*(V)}{P(V|Z)} \right] P(\phi) \approx \frac{1}{n} \sum_{i=1:n} y \frac{P^*(V)}{P(V|Z)} \end{aligned}$$

Can predict OOD outcome  
Y without any training data  
from the new distribution!

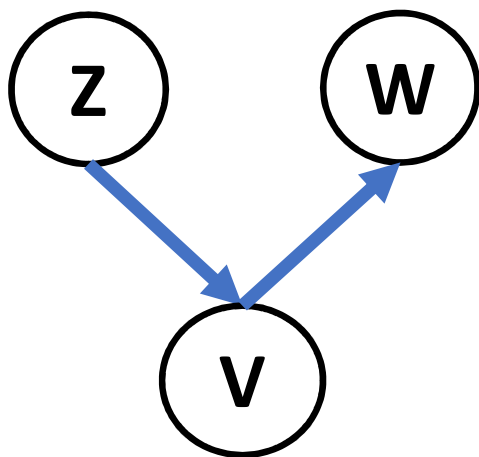
**Importance Sampling.** Adjusts to the new distribution. Estimate  $P(V|Z)$  using observed data.  
 $P^*(V)$  could be a constant ( $V=0.5$ ), a random variable  $\sim U(0,1)$  or any other arbitrary function.

## Can also define **counterfactuals** using causal Bayesian networks

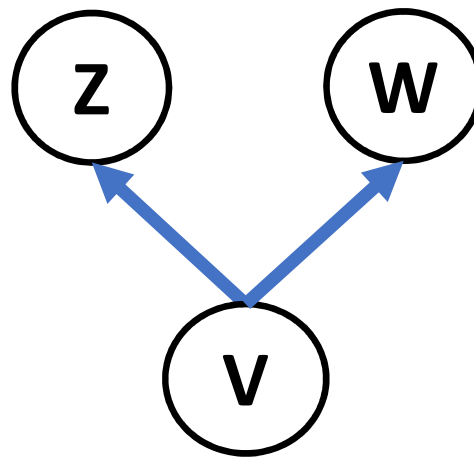
- *E.g., We observe that a user was shown a recommendation and clicked on the product. What would have happened if he had not been shown the recommendation?*
- Interventions are about the effect of future actions.
- Counterfactuals are about hypothetical effects.
  - Advanced topic. Will skip.

- I. What is causality?
- II. How can we reason about causality mathematically?  
From Bayesian Networks to Causal Bayesian Networks (causal DAGs)
- III. Can we learn a causal DAG?**
- IV. Application 1: Estimating the effect of actions
- V. Application 2: Building more generalizable prediction models
- VI. Open questions

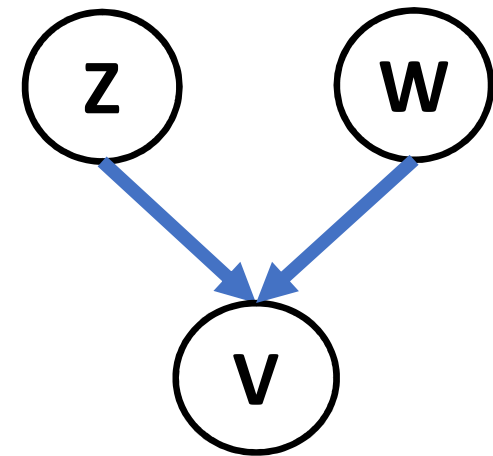
Like Bayesian networks, causal DAGs also imply conditional independencies



Chain  
 $Z \not\perp W$   
 $Z \perp W \mid V$



Fork  
 $Z \not\perp W$   
 $Z \perp W \mid V$



Collider  
 $Z \perp W$   
 $Z \not\perp W \mid V$

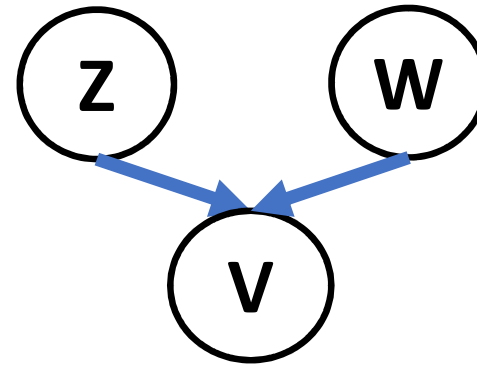
Two variables are **d-separated** if they are independent of each other.

Two variables are **conditionally d-separated** if they are independent conditional on some other variables.

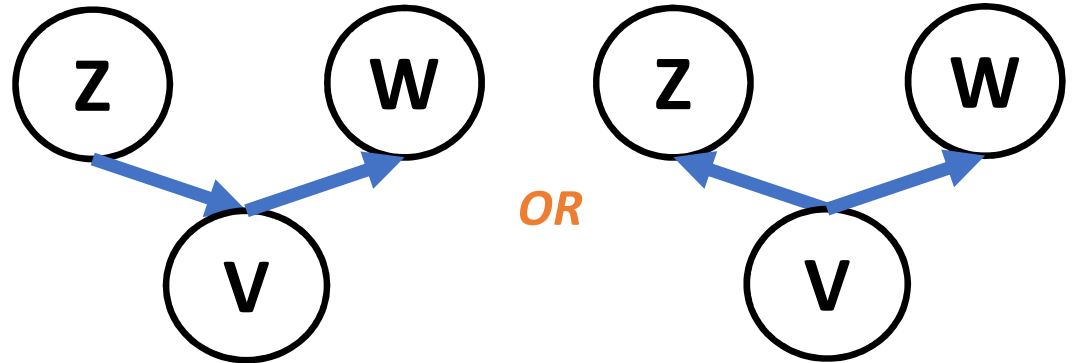
The conditional dependencies can be used to learn graph structure from data

Given a dataset  $(V, W, Z)$

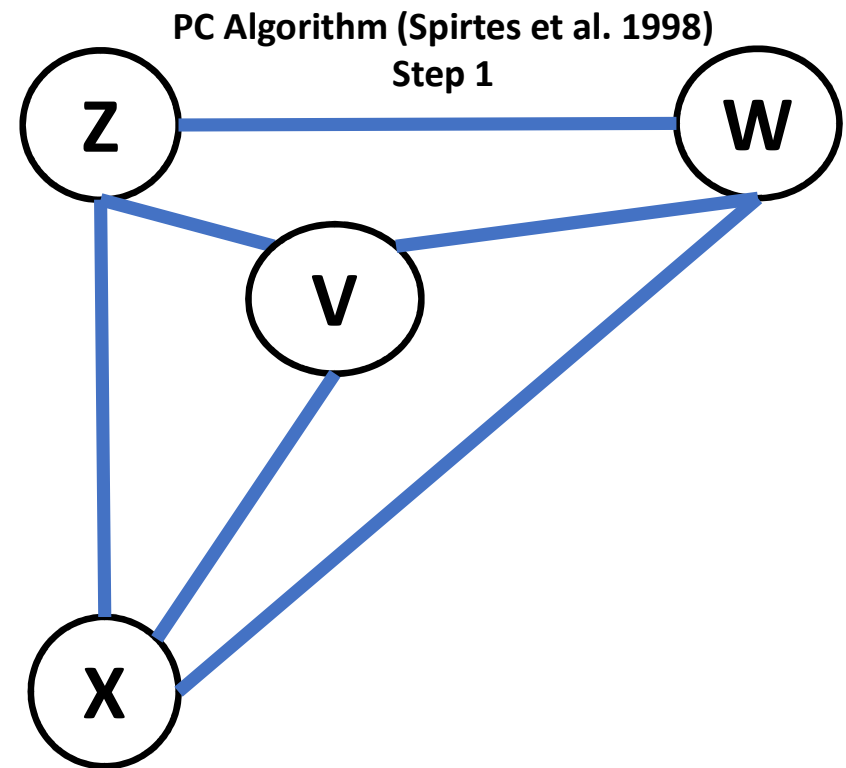
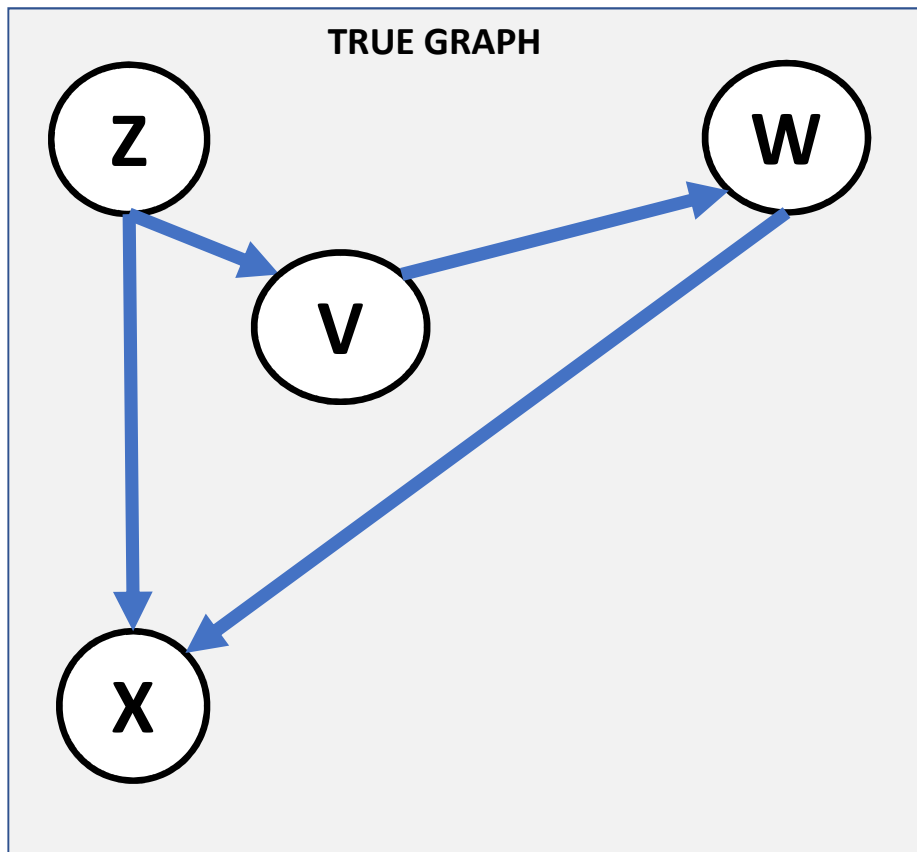
If  $Z \perp\!\!\!\perp W$  and  $Z \not\perp\!\!\!\perp W|V$



If  $Z \not\perp\!\!\!\perp W$  and  $Z \perp\!\!\!\perp W|V$



Causal DAGs can be learnt only upto Markov-equivalent graphs

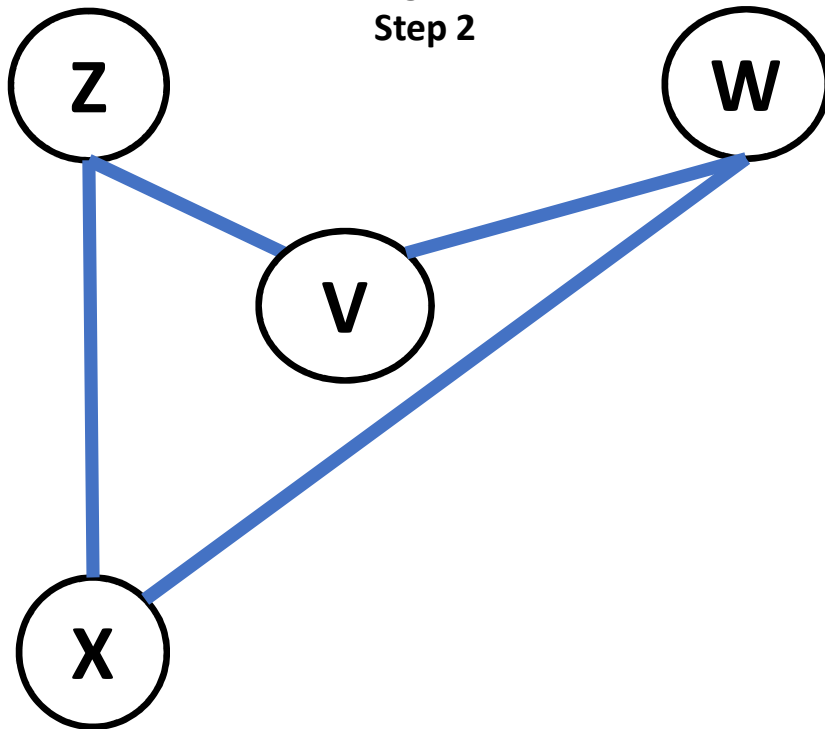


$Z \perp W | V \Rightarrow Z-W$  arrow can be removed.  
 $X \perp V | Z, W \Rightarrow X-V$  arrow can be removed.



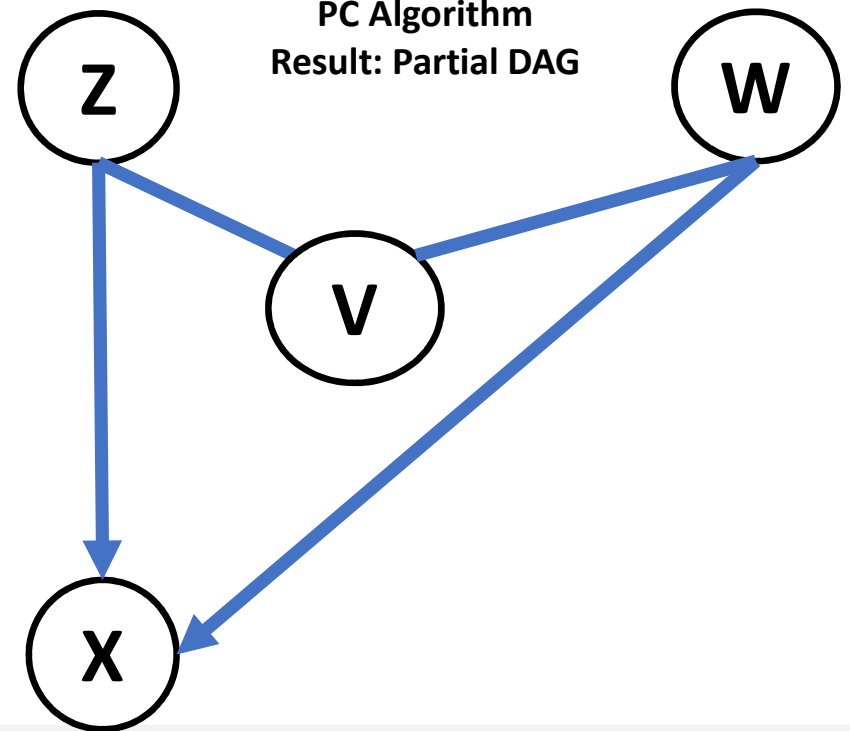
# Causal DAGs can be learnt only upto Markov-equivalent graphs

PC Algorithm  
Step 2



$Z \not\perp W | X \Rightarrow$  orient edges as  $Z \rightarrow X \leftarrow W$

PC Algorithm  
Result: Partial DAG



**Partial DAG.** Impossible to learn the direction of Z-V-W because both forks and chains imply the same observational distribution.

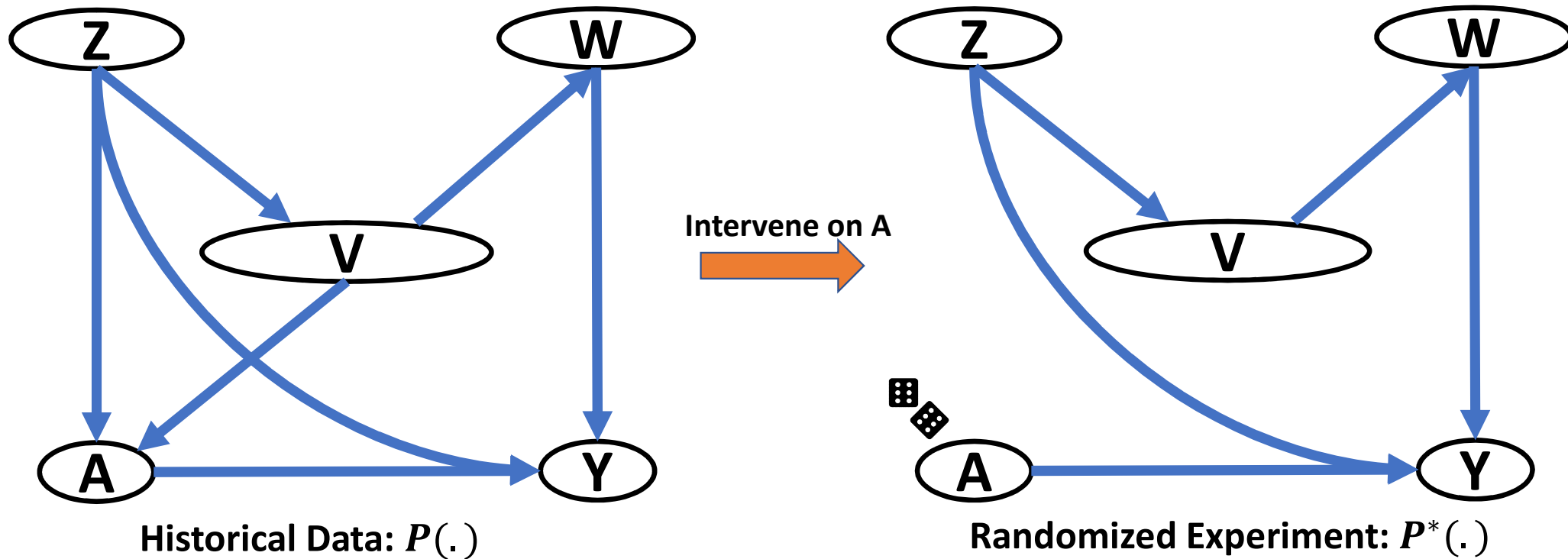
Even when all variables are observed, the true graph cannot be learnt from a dataset

- Some exceptions under parametric assumptions (e.g., non-gaussian noise). But no general result.
- When some variables are unobserved, it is a harder problem.
- Active area of research. Recent work proposes a differentiable constraint to search for DAGs.
  - “No-tears” algorithm [NeurIPS 2018].
  - Connections to reinforcement learning and bandits

- I. What is causality?
- II. How can we reason about causality mathematically?  
From Bayesian Networks to Causal Bayesian Networks (causal DAGs)
- III. Can we learn a causal DAG?
- IV. Application 1: Estimating the effect of actions**
- V. Application 2: Building more generalizable prediction models
- VI. Open questions

# Estimating causal effect

- **Input:** Causal DAG, Action variable  $A$ , outcome variable  $Y$
- **Output:**  $P(Y|do(A))$ , usually  $E[Y|do(A)]$



**Causal Effect:**  $E_P[Y|do(A)] = E_{P^*}(Y|A = a)$

$$E_P[Y|do(A = 1)] - E_P[Y|do(A = 0)] = E_{P^*}(Y|A = 1) - E_{P^*}(Y|A = 0)$$

But what if randomized  $P^*$  distribution is not available?

**Identification** problem:

*Can we  $P(Y|do(A))$  purely as a probability expression computable over  $P$ ?*

Understanding **do-intervention** that leads to  $P^*$ :

The intervention makes  $A$  d-separated from  $Y$  after removing the  $A \rightarrow Y$  edge.

**Q:** Can we find how  $A$  can be d-separated from  $Y$  in the original distribution  $P$ ?

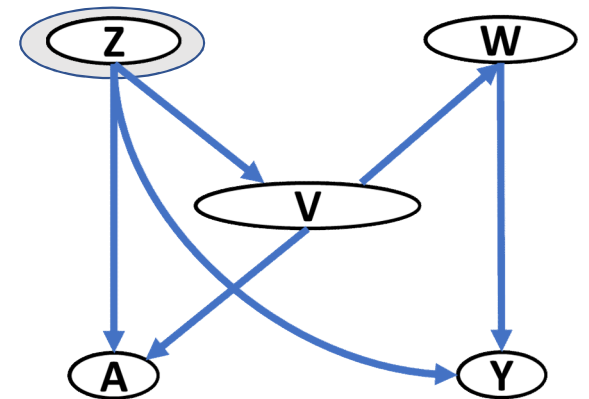
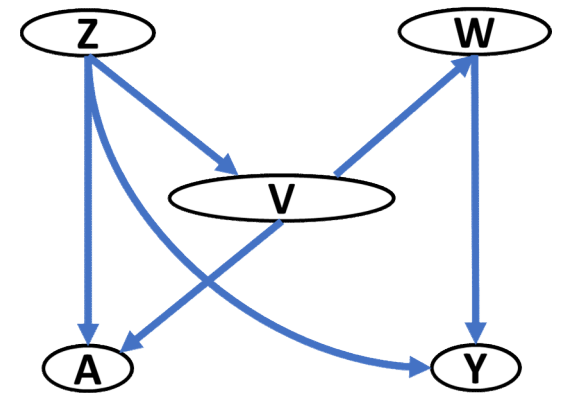
# The backdoor criterion

1. Remove any edges going out from A.
2. Find the set of variables such that A is conditionally d-separated from Y.
3. Condition on them.

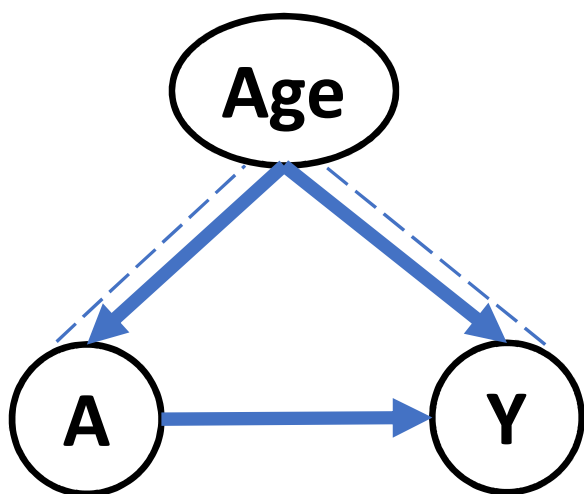
**Backdoor set:**  $\{Z, W\}$  or  $\{Z, V\}$

$P(Y|do(A))$

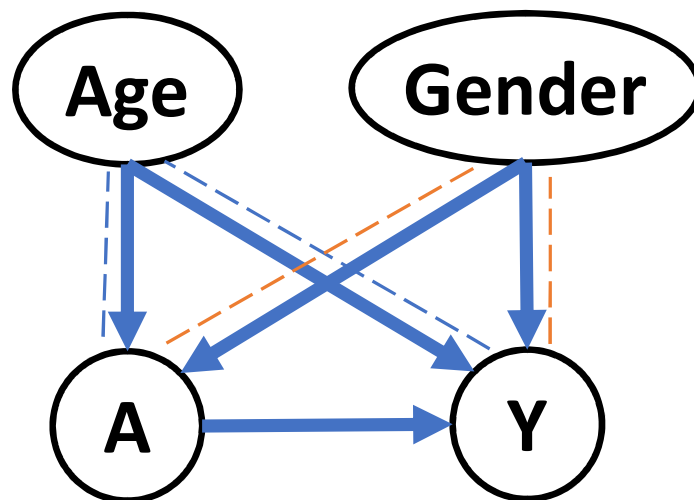
$$= \sum_{Z, W} P(Y|A, Z = z, W = w)P(Z = z, W = w)$$



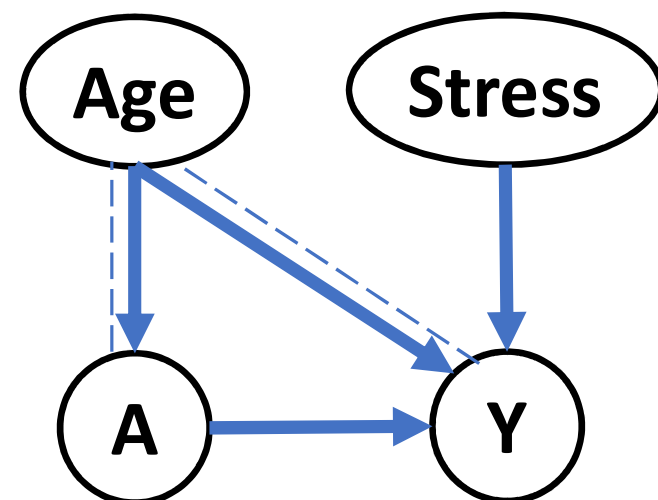
Find the backdoor set!



$$B = \{Age\}$$

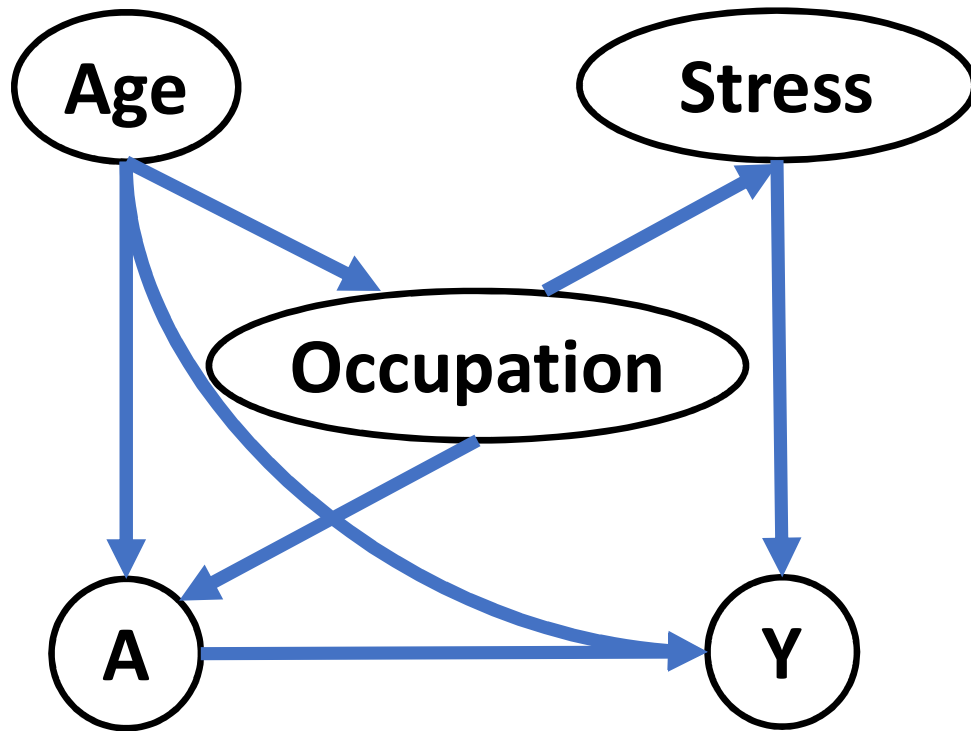


$$B = \{Age, Gender\}$$

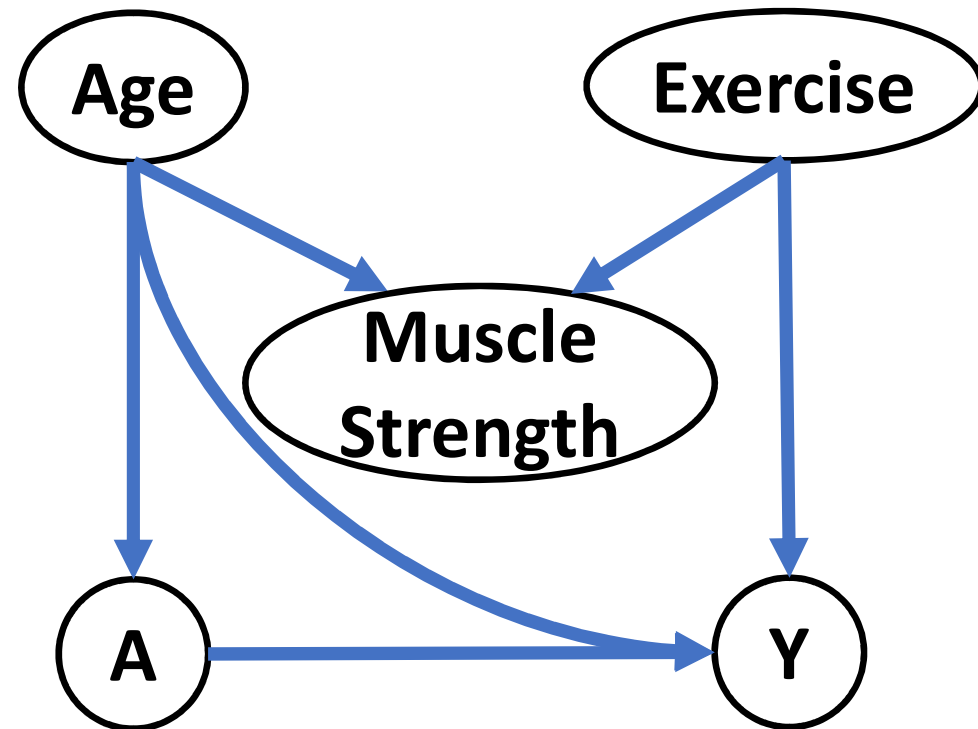


$$B = \{Age\}$$

Find the backdoor set!



$B = \{Age, Stress\}$   
 $B = \{Age, Occupation\}$



$B = \{Age, Exercise\}$    
 $B \neq \{Age, MuscleStrength\}$



# Backdoor is necessary, but not a sufficient criterion for identification

- There can be other ways to derive the do-expression (e.g., frontdoor criterion)
- Fortunately, there exists an algorithm that is both necessary and sufficient for an arbitrary causal DAG.
  - If it returns a probability expression, it is a valid identification.
  - If it fails to return an expression, then no valid non-parametric identification exists.
- Called **ID algorithm** [Shpitser and Pearl, 2006].
- Implemented in software libraries like DoWhy.

# How to estimate the backdoor expression

$$P(Y|do(A)) = \sum_{z,w} P(Y|A, Z = z, W = w)P(Z = z, W = w)$$

$$E(Y|do(A)) = \sum_{z,w} E(Y|A, Z = z, W = w)P(Z = z, W = w)$$

- Conditional probability expression
  - Can apply any conditional expectation estimator (e.g., regression)
  - But certain methods have better bias-variance tradeoff
- Simple estimator for a binary action: [Matching](#)

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

**Q. What is the effect of cycling on health outcomes?**



Control



Action(Cycling)

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

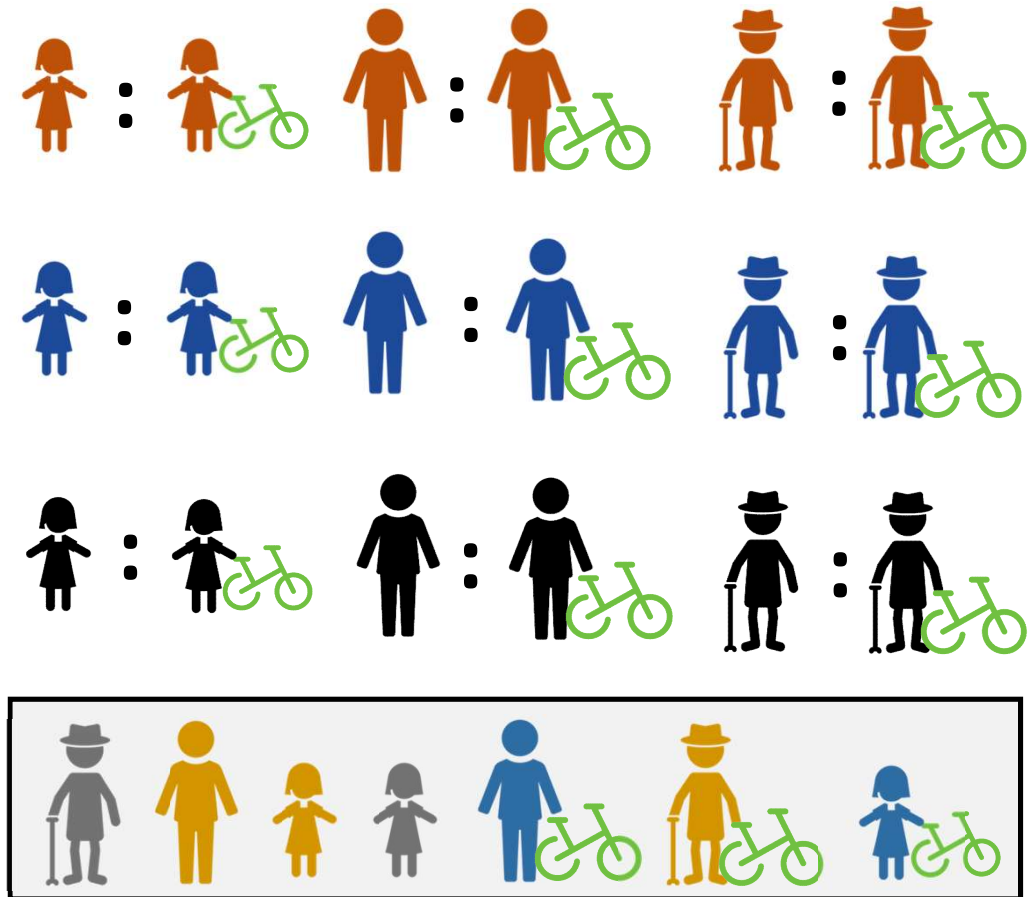
Identify pairs of treated ( $j$ ) and untreated individuals ( $k$ ) who are similar or identical to each other.

$$\text{Match} := \text{Distance}(W_j, W_k) < \epsilon$$

- Paired individuals have almost the same confounders.

Causal Effect =

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



# Challenges of building a good matching 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.

Advanced methods try to navigate the [bias-variance tradeoff](#).

**Important:** There is no way to validate the estimate since there is no ground-truth causal effect available!

We never observe the ground-truth effect since action is either taken or not taken

Person	A	$Y_{A=1}$	$Y_{A=0}$
P1	1	0.4	0.3
P2	0	0.8	0.6
P3	1	0.3	0.2
P4	0	0.3	0.1
P5	1	0.5	0.5
P6	0	0.6	0.5
P7	0	0.3	0.1

Causal effect:

$$E[Y|do(A = 1)] - E[Y|do(A = 0)] \\ = E[Y_{A=1} - Y_{A=0}]$$

**Fundamental problem of causal inference:** For any person, observe only one: either  $Y_{A=1}$  or  $Y_{A=0}$

# How to evaluate a causal effect estimate?

**Gold Standard:** Run a randomized experiment varying the action.

E.g., in recommender systems

Causal analysis can help identify the recommendation algorithms with highest expected causal effect.

Run an experiment using only those recommendations.

**Other ways:**


Sanity checks

Placebo Action. Replace action by a random variable=>effect goes to 0

Sensitivity analysis (what if the graph was missing a confounder)

Active area of research!

## Summary: Causal effect inference has 4 steps

 1. **Modeling:** Learn or build the causal graph.

 2. **Identification:** Given the graph, derive  $P(Y|do(A))$  in terms of observed data distribution only.

 3. **Estimation:** Estimate the derived expression using a suitable method.

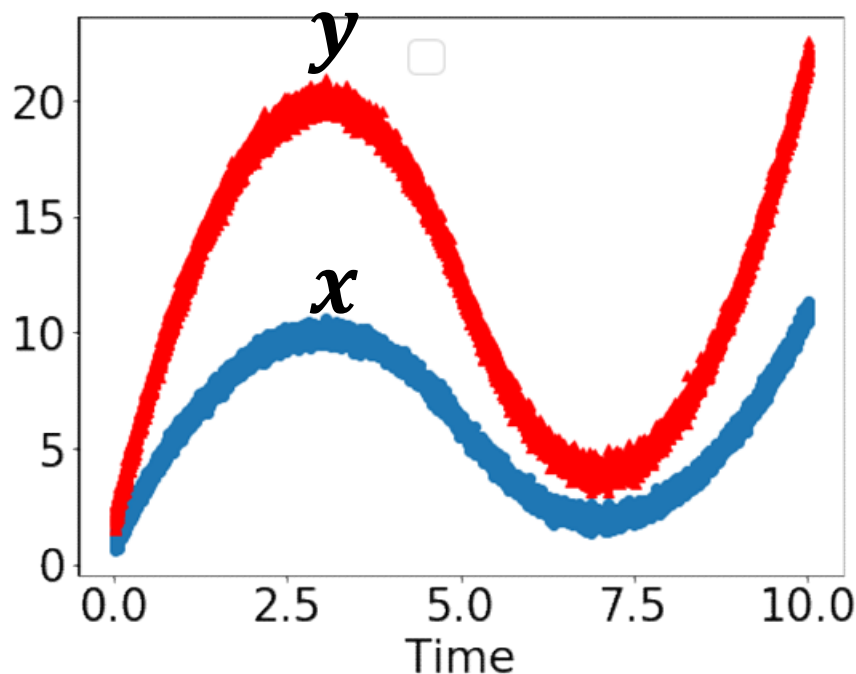
 4. **Validation/Refutation:** Check whether the estimate is correct or not.



Example using DoWhy library: Search volume for two queries,  $x$  and  $y$ .

Does  $x$ 's search volume have any effect on  $y$ ?

<https://github.com/microsoft/dowhy>



$x$	$y$	$w$
8.329680	16.546904	2.546634
2.083811	4.096492	-3.995819
6.138014	12.041800	0.041479
8.874336	17.833621	2.988497
5.282355	10.481077	-0.860686

You can **try out this example** on Github:

[https://github.com/microsoft/dowhy/blob/master/docs/source/example\\_notebooks/dowhy\\_confounder\\_example.ipynb](https://github.com/microsoft/dowhy/blob/master/docs/source/example_notebooks/dowhy_confounder_example.ipynb)

1

```
model= CausalModel(
    data=df,
    treatment=data_dict["treatment_name"],
    outcome=data_dict["outcome_name"],
    common_causes=data_dict["common_causes_names"],
    instruments=data_dict["instrument_names"])
```

2

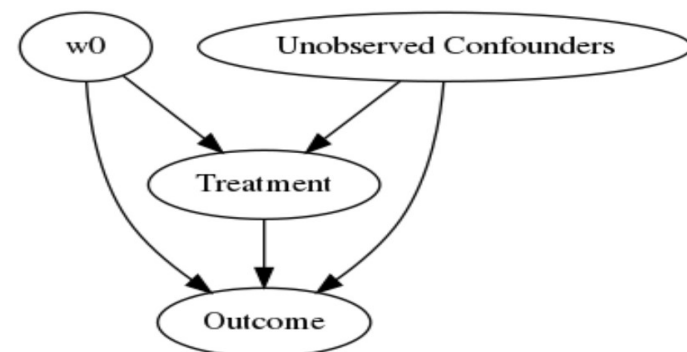
```
identified_estimand = model.identify_effect()
```

3

```
estimate = model.estimate_effect(identified_estimand,
    method_name="backdoor.linear_regression")
```

4

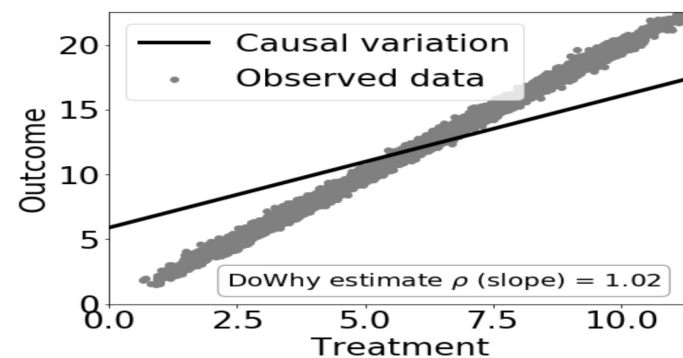
```
res_placebo=model.refute_estimate(identified_estimand, estimate,
    method_name="placebo_treatment_refuter", placebo_type="permute")
```



Estimand name: backdoor

Estimand expression:

$$\frac{d}{d[\text{Treatment}]} \text{Expectation}(\text{Outcome} | w_0)$$



Refute: Use a Placebo Treatment

Estimated effect:(1.0154712956668286,)

New effect:(-0.001212143363314766,)

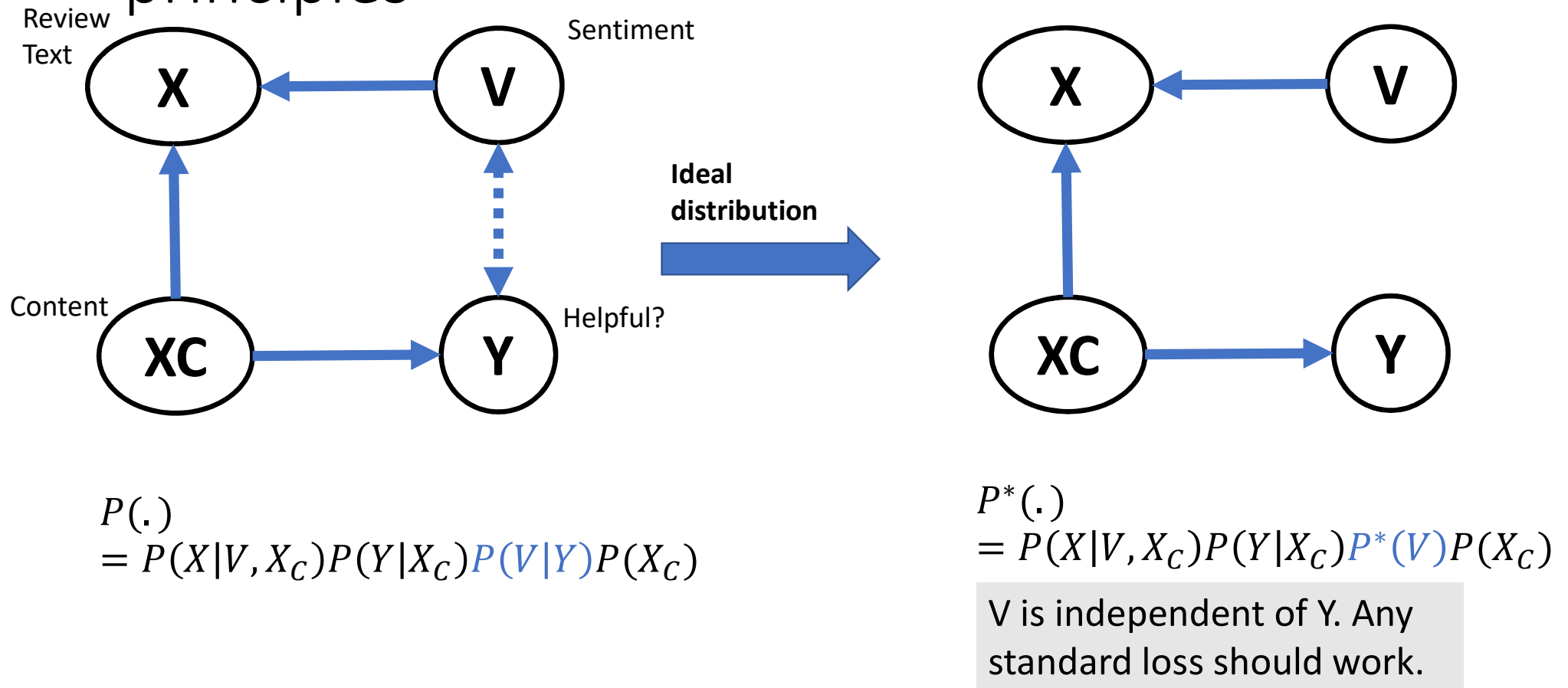
- I. What is causality?
- II. How can we reason about causality mathematically?  
From Bayesian Networks to Causal Bayesian Networks (causal DAGs)
- III. Can we learn a causal DAG?
- IV. Application 1: Estimating the effect of actions
- V. Application 2: Building more generalizable prediction models**
- VI. Open questions

# How can causal DAGs help in prediction?

- Causal DAGs represent the **data-generating process** that leads to the training dataset.
- Interventions on DAGs provide a **formal** way of expressing plausible distribution shifts.
- So using the same techniques, we can build predictors that are robust to those interventions.

**IMPORTANT:** Causal DAGs can help in formulating the right loss function or indicate which data to collect. But for training the model, we use standard ML techniques.

# Building a generalizable predictor from first principles



# How to translate training dataset to ideal distribution?

$$P(.) \\ = P(X|V, X_C)P(Y|X_C)P(V|Y)P(X_C)$$

$$P^*(.) \\ = P(X|V, X_C)P(Y|X_C)P^*(V)P(X_C)$$

$$\begin{aligned} E_{P^*}[l(y, \hat{y})] &= E_{\phi \sim P^*(\Phi)}[l(y, \hat{y})] \\ &= \int_{\phi} l(y, \hat{y}) P^*(\phi) \\ &= \int_{\phi} l(y, \hat{y}) \frac{P(\phi)}{P(\phi)} P^*(\phi) = \int_{\phi} l(y, \hat{y}) \frac{P^*(\phi)}{P(\phi)} P(\phi) \\ &= \int_{\phi} l(y, \hat{y}) \left[ \frac{P^*(V)}{P(V|Y)} \right] P(\phi) \approx \frac{1}{n} \sum_{i=1:n} l(y, \hat{y}) \frac{P^*(V)}{P(V|Y)} = \frac{1}{n} \sum_{i=1:n} l(y, \hat{y}) \frac{1}{P(V|Y)} \end{aligned}$$

**Insight:** Minimize weighted loss to obtain a predictor that does not depend on spurious feature V

Sometimes weighting can have high variance.  
 So, use constraints from  $P^*$  on  $X_C$

**Going from  $P$  to  $P^*$**

$$Loss1: \frac{1}{n} \sum_{i=1:n} l(y, \hat{y}) \frac{1}{P(V_i|y)}$$

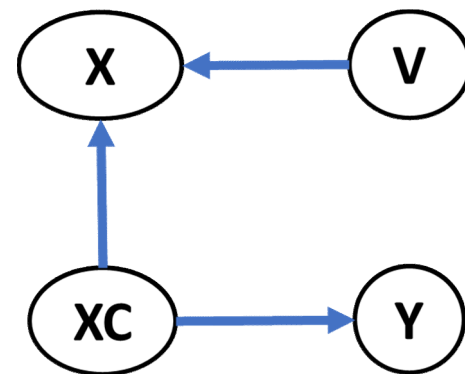
**Going from  $P$  to  $P^*$ :  $X_C \perp V$**

Add a regularizer.

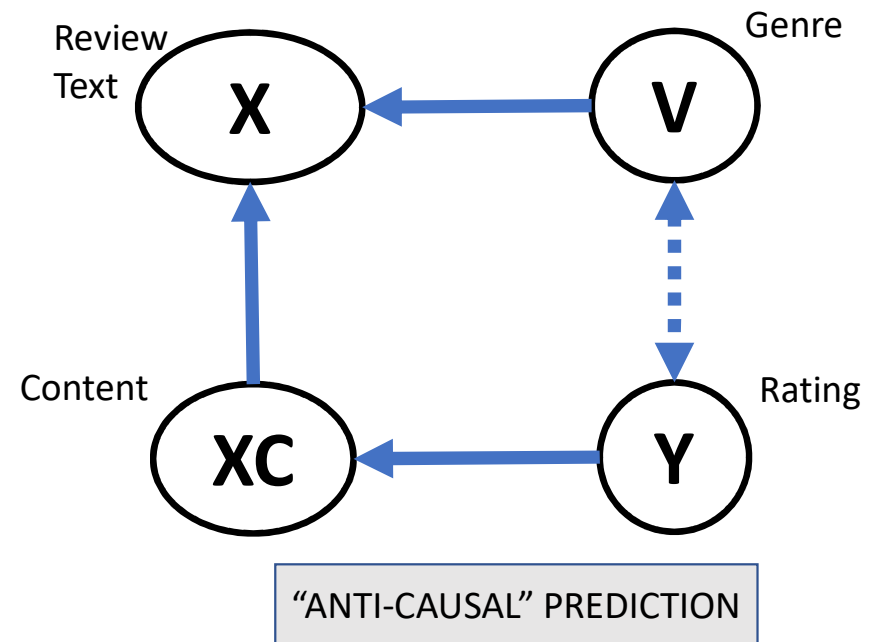
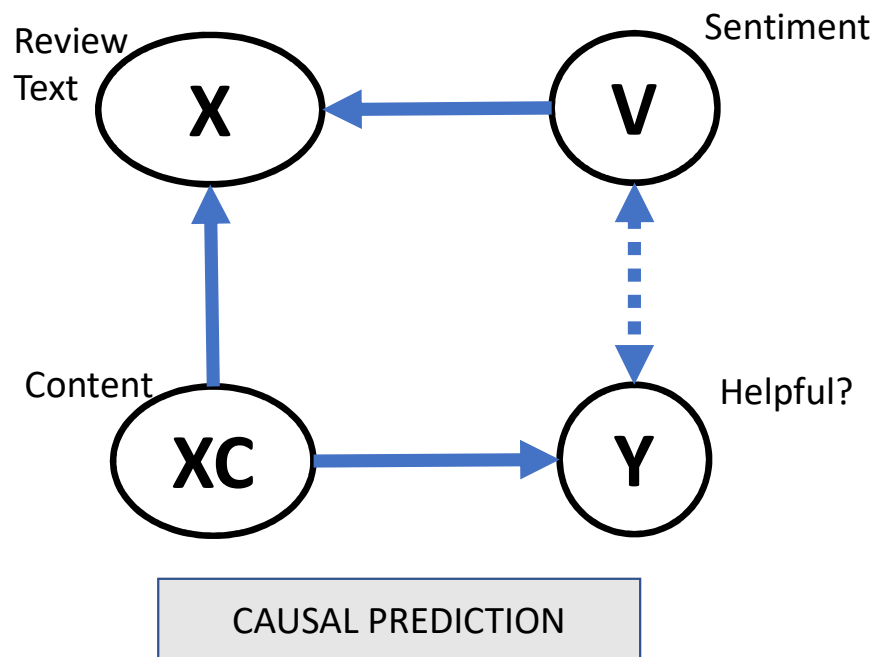
$$Loss2: \frac{1}{n} \sum_{i=1:n} l(y, \hat{y}) + Ind(\psi(x), V)$$

**Can also combine both:**

$$Loss: \frac{1}{n} \sum_{i=1:n} l(y_i, h(\psi(x))) \frac{1}{P(V_i|Y_i)} + \lambda Ind(\psi(x), V)$$

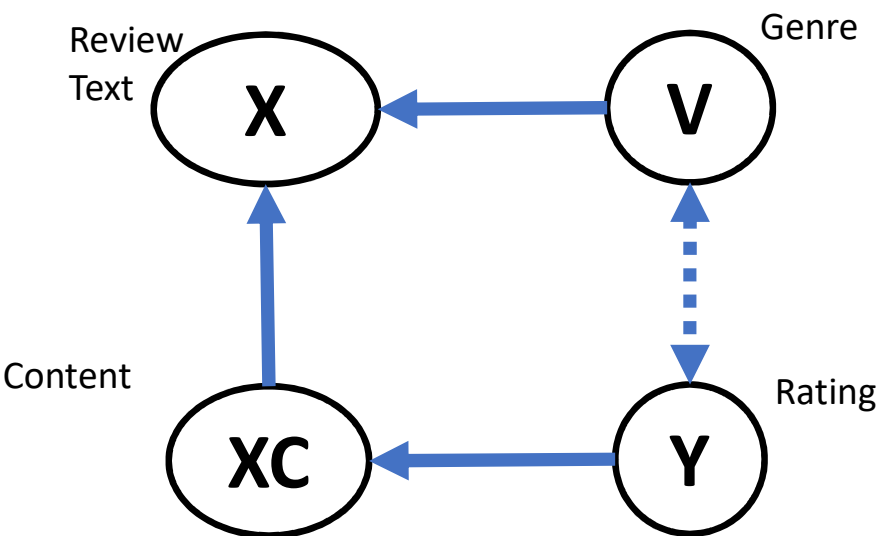


So far, we discussed prediction from parents.  
But we can also predict from children.

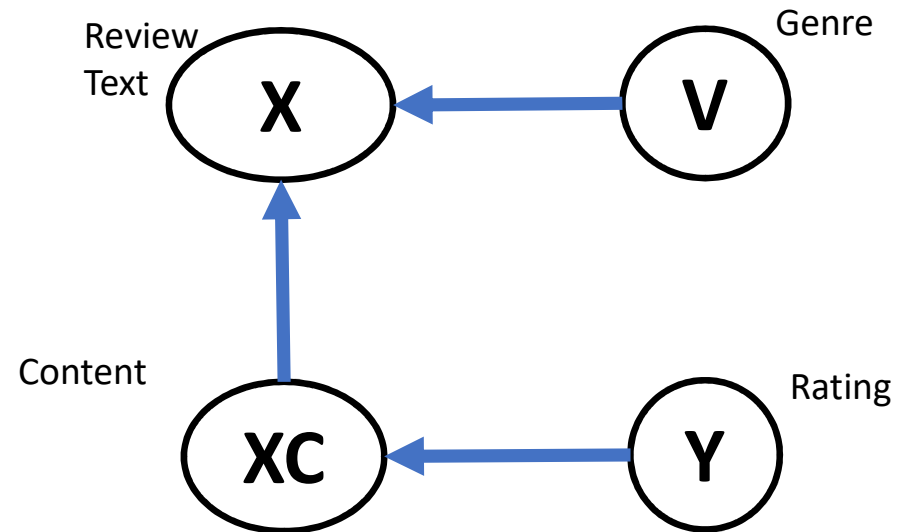


The same constraints won't work. Need to analyze afresh.





$$P(.) \\ = P(X|V, X_C)P(X_C|Y)P(V|Y)P(Y)$$

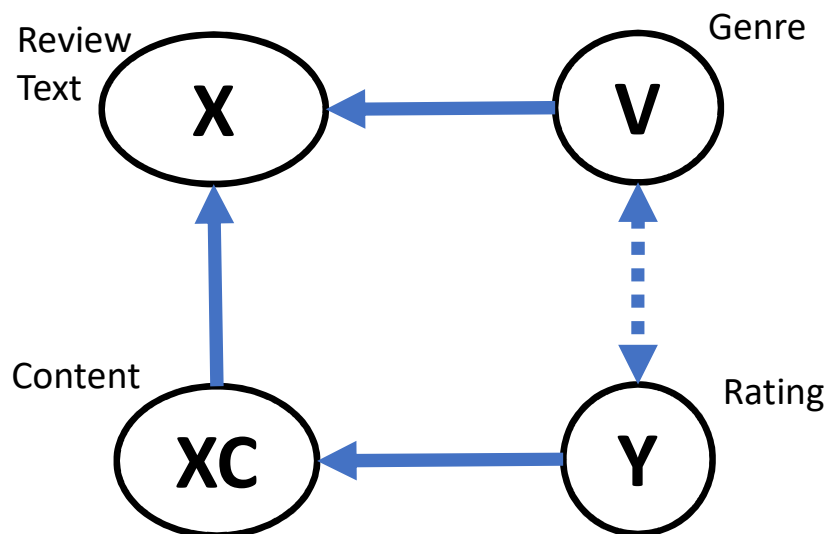


$$P^*(.) \\ = P(X|V, X_C)P(X_C|Y)P^*(V)P(Y)$$

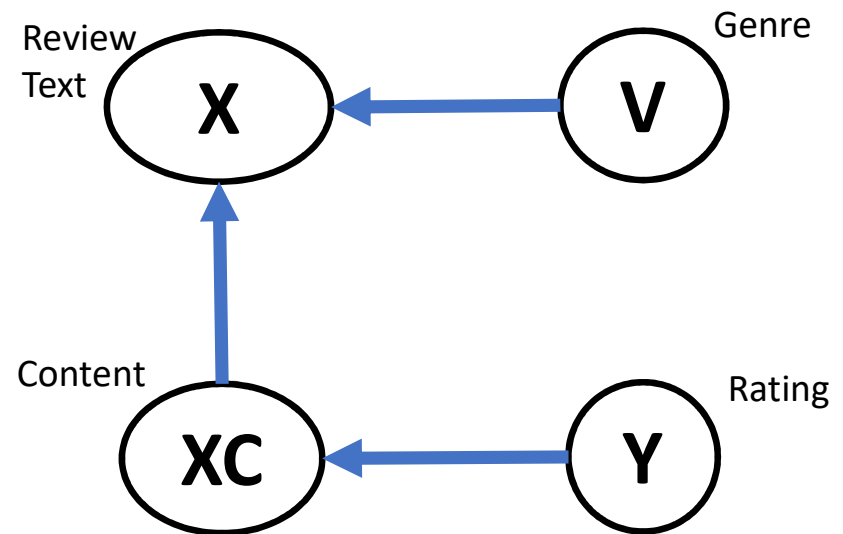
Weighting for the loss remains the same.

But Loss2 is not the correct constraint since  $X_C \not\perp V$  under P.

$$\text{✗ } \text{Loss2: } \frac{1}{n} \sum_{i=1:n} l(y, \hat{y}) + \text{Ind}(\psi(x), V)$$



$$P(.) \\ = P(X|V, X_C)P(X_C|Y)P(V|Y)P(Y)$$



$$P^*(.) \\ = P(X|V, X_C)P(X_C|Y)P^*(V)P(Y)$$

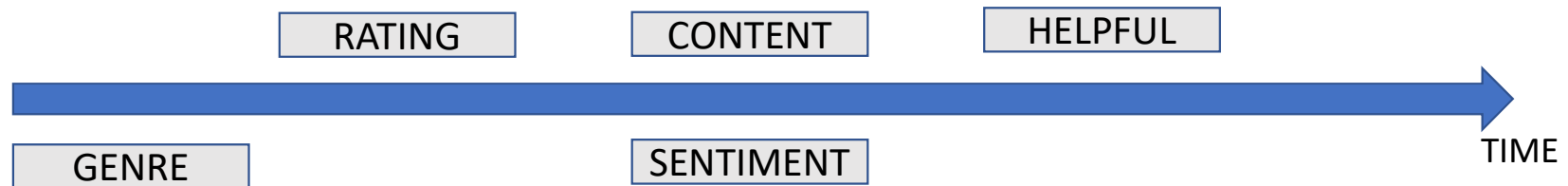
### Correct Loss (conditional independence)

$$Loss2: \frac{1}{n} \sum_{i=1:n} l(y_i, \hat{y}_i) + CondInd(\psi(x), V, Y)$$

### Correct Loss (combine with weighting):

$$Loss: \frac{1}{n} \sum_{i=1:n} l(y_i, h(\psi(x))) \frac{1}{P(V_i|Y_i)} + \lambda Ind(\psi(x), V)$$

But wait, why does the regularization constraints depend on the graph?



**Task:** Predict helpfulness (causal direction)

- When content is written, there is no helpfulness label.
- So, content and sentiment are independently generated based on the rating and genre.

$$\Rightarrow X_C \perp V$$

**Task:** Predict rating (anti-causal direction)

- When content is written, it depends on the rating.
- So content, sentiment and genre-related words are generated based on the rating.

$$\Rightarrow X_C \perp V | Y$$

Without causal analysis, regularizer will either *over-constraint* the loss or *under-constraint* it.

But wait, how do we know that the graph is correct?

- Can try multiple graphs and derive constraints from each
  - If constraints agree, that's a very good sign.
  - If they don't, try to justify which graph captures the data-generating process.
- In any case, can use cross-validation if labelled data is available from a new distribution.

# Causal representation learning

- An extension of these ideas
- Often using data from [multiple distributions](#)
- How to generate the right constraints?
- Connections to domain adaptation, domain generalization, and OOD generalization literature.
  - Causal Representation learning. Scholkopf et al. 2021.
  - On causal and anti-causal learning. Schokopf et al. 2012.
  - Counterfactual invariance to spurious tests. Veitch et al. 2021
  - Causally Motivated Shortcut Removal Using Auxiliary Labels. Makar et al. 2022.

- I. What is causality?
- II. How can we reason about causality mathematically?  
From Bayesian Networks to Causal Bayesian Networks (causal DAGs)
- III. Can we learn a causal DAG?
- IV. Application 1: Estimating the effect of actions
- V. Application 2: Building more generalizable prediction models
- VI. Open questions**

# Conclusion and open questions:

Causality is a natural concept to model, but still early days for causality in machine learning

**1:** Causal discovery using single training distribution is impossible.

**How to do causal discovery with multiple train distributions?**

**2:** Causal analysis still depends on the correct graph.

**What are other ways of formally encoding domain knowledge?**

**3:** Effect inference does not have ground-truth available.

**What are good ways of evaluating causal effect estimates?**

**4:** Causal analysis can provide regularizers for prediction.

**Would causally motivated predictors have a significant impact in real-world applications?**

# Questions?

Amit Sharma



# Building a predictor that generalizes to a new distribution

## Markov Blanket:

Parents of  $Y$

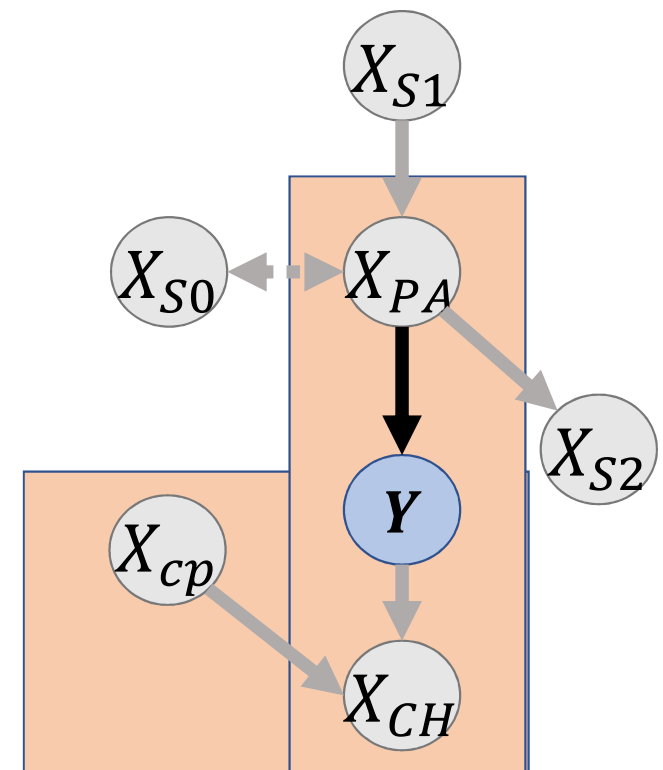
Children of  $Y$

Parents of children of  $Y$

*Conditioned on the Markov Blanket,  $Y$  is independent of all other variables in the causal DAG.*

So should we always use the Markov Blanket for predictive models that will be robust to distribution shifts?

***Depends.***

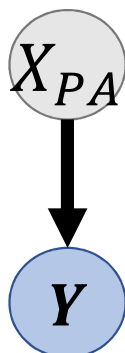


If you have unlabelled data from the new distribution, then it is okay to use children of  $Y$ .

If not, then only use the parents.

# Two kinds of prediction

## CAUSAL



*Markov factorization*

$$P(Y, X_{PA}) = P(Y|X_{PA})P(X_{PA})$$

Thus, learnt function  $P(Y|X_{PA})$  is independent of  $P(X_{PA})$ .

Unlabelled data  $P^*(X_{PA})$  will not

## ANTI-CAUSAL

