

# Causality

## *Correlation vs Causation*

Understanding Machine Learning

# Data-Generating Process

story behind the data as important as the data itself

Why are the statistical dependencies as observed?

→ data-generating process mostly governed by causal dependencies

but language of algebra symmetric

no way to tell that a storm causes barometer to go down and not the other way around → need for asymmetric mathematical language

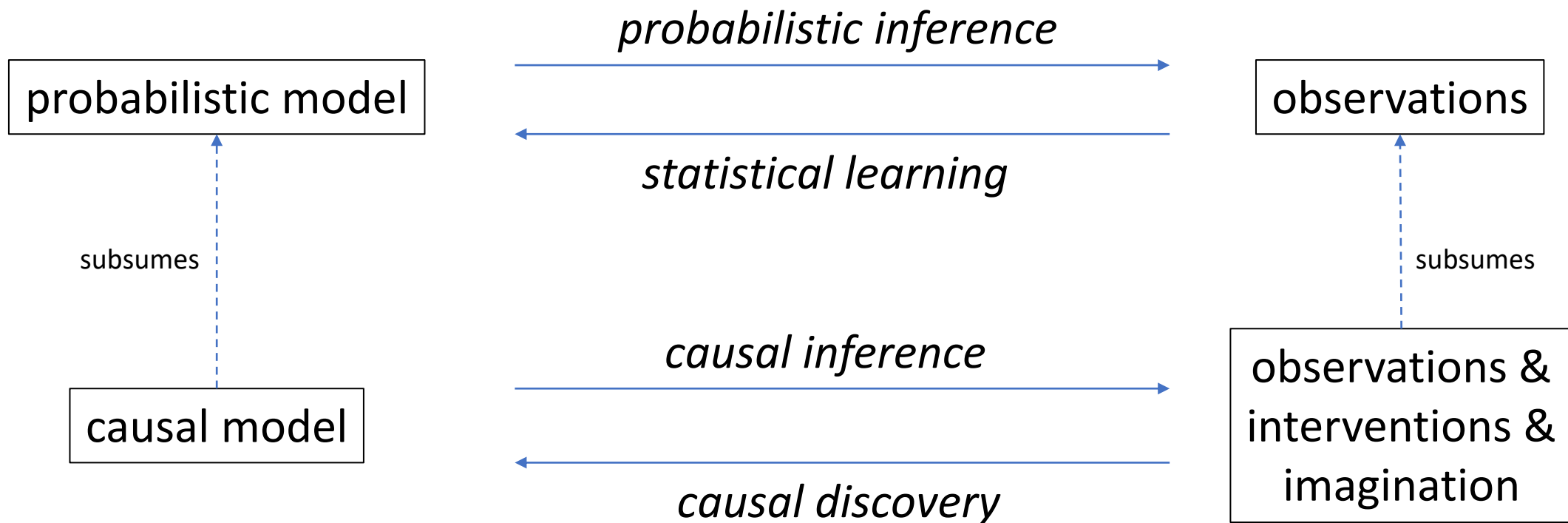
# Levels of Causal Modeling

physical or deterministic models:  
described by differential equations  
*needs somebody to come up with*

causal models:  
described by probabilities with causal structure  
*learned from data with causal assumptions*

statistical models (including ML):  
described by probabilities  
*can be learned from data*

# Probabilistic and Causal Models



from *Elements of Causal Inference* (Peters, Janzing, Schölkopf)

# Causality and ML

causation as addition to ML: synonym for understanding

- no ML method can understand causes and effects from data (statistical dependencies) alone → need for causal assumptions (causal model)
- examples: transportability and robustness of models (generalization), handling of missing data, causal explainability

(see [Pearl](#) and Schölkopf [1](#) [2](#) [3](#))

but also: ML as help for causal discovery and inference

- counterfactuals (potential outcomes) need matching of individual cases
- adjusting for many confounders (e.g., propensity scores)

# Graphical Models

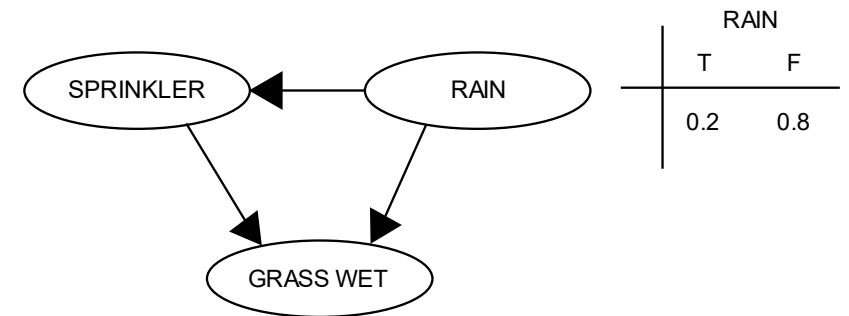
# Probabilistic Graphical Model

Directed Acyclic Graph (DAG): random variables (nodes) and their (conditional) dependencies (edges)

Bayesian network: DAG with updates on new evidence according to Bayes' rule (belief propagation)

- message from parent to child: update using conditional probabilities
- message from child to parent: update by multiplication with likelihood ratio

RAIN	SPRINKLER	
	T	F
F	0.4	0.6
T	0.01	0.99



SPRINKLER RAIN		GRASS WET	
		T	F
F	F	0.0	1.0
F	T	0.8	0.2
T	F	0.9	0.1
T	T	0.99	0.01

from wikipedia

# Markov Property

Each variable in a DAG is independent of its non-descendants conditional on its parents.

chain rule:

$$A \rightarrow B \rightarrow C: \quad P(A, B, C) = P(A)P(B|A)P(C|B)$$

→ compact representation of conditional probability table



# Conditional Independencies

DAG junctions:

- chains:  $A \rightarrow B \rightarrow C$  ( $A$  and  $C$  conditionally independent given  $B$ )
- forks:  $A \leftarrow B \rightarrow C$  ( $A$  and  $C$  conditionally independent given  $B$ )
- colliders:  $A \rightarrow B \leftarrow C$  ( $A$  and  $C$  conditionally dependent given  $B$ )

d-separation (d means directional): conditional independencies in data corresponding to chains, forks, and colliders in a graph

→ allows for model testing

# Toward Causal Diagrams

load graph with causal assumptions (model)

→ arrows give direction of cause-effect relationships

causal Markov property: Each variable in a causal graph is probabilistically independent of its non-effects conditional on its direct causes.

causal models (represented by graphs) allow to answer interventional and counterfactual (fictional) queries

often even without conducting experiments

# Causal Inference

# The Ladder of Causation

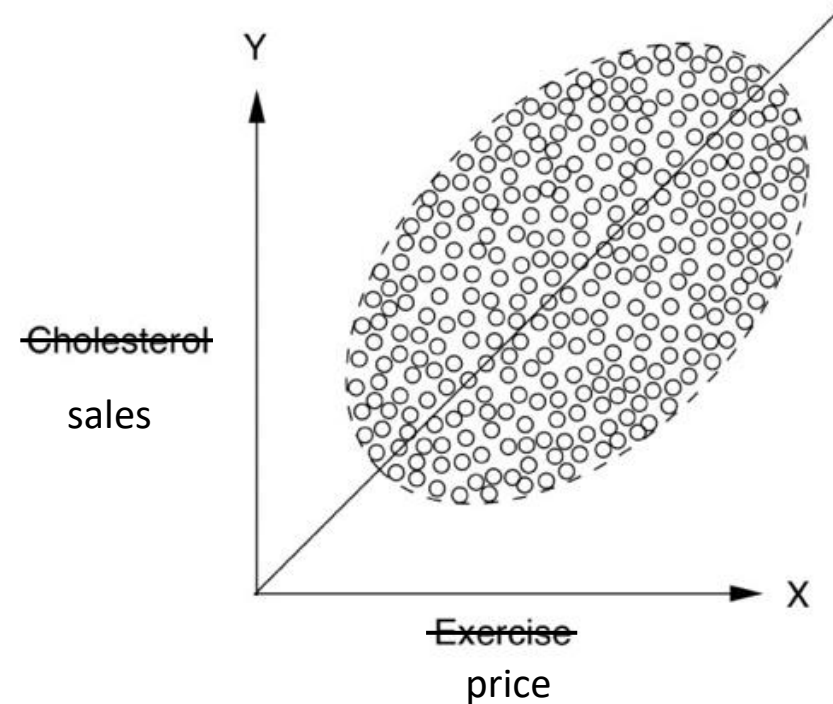
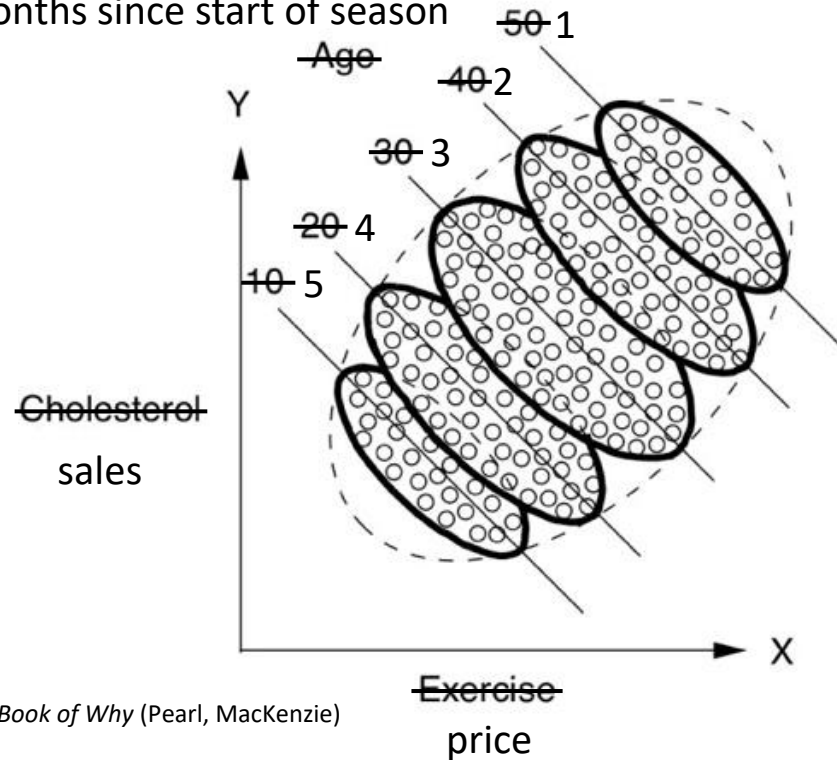
III	<i>imagining</i>	<b>counterfactuals</b>	What if I had done ...? Why?
II	<i>doing</i>	<b>intervention</b>	What if I do ...? How?
I	<i>seeing</i>	<b>association</b>	What if I see ...? → Realm of ML

from *The Book of Why* (Pearl, MacKenzie)

# Simpson's Paradox

example: monthly sales of a fashion article in different shops during winter season  
Lower price yields higher sales for each month individually, but lower sales overall?

months since start of season



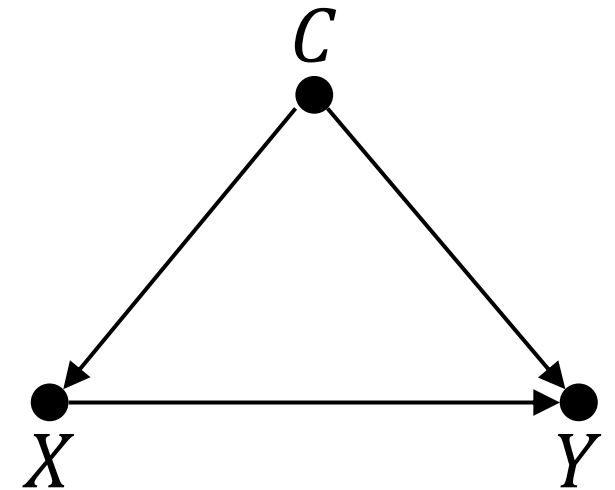
from *The Book of Why* (Pearl, MacKenzie)

# Confounding

causal effect of  $X$  on  $Y$  confounded by common cause  $C$ :

$$X \leftarrow C \rightarrow Y$$

→ spurious correlation between  $X$  and  $Y$   
(overlying potential true causal effect of  $X$  on  $Y$ )



common cause principle:

every correlation either due to a direct causal effect linking the correlated entities or brought about by a third factor (confounder)

# Example: Pricing in Retail

price setting for a product can be considered a demand shaping method

the idea is a causal effect: lower price leads to higher demand

→ by estimating this causal effect one can find an optimal price according to a given policy (like maximizing profit)

problem: confounders influencing both pricing in past (observed) data and demand, e.g., lowering prices on weekends only for grocery or lowering prices toward end of a season for fashion (most sales at beginning of season)

# Example: Temporal Confounding

forecasting of times series can rely on endogenous (past values of the time series to forecast in the future) or exogenous (other variables) information

- endogenous information means auto-correlation
- auto-correlation often spurious (temporal confounding, e.g., common causes at consecutive times), i.e., no direct causal effects

most state-of-the-art forecasting methods rely mainly on endogenous information, i.e., temporal confounding

but learning of direct causal effects from exogenous information offers several advantages, like explainability, long-term forecasting, predictability of rare events

one simple possible solution: no use of target auto-correlation in (quasi-causal) ML model, but subsequent residual correction



# Interventions

aim:

prediction of (average) causal effect of (yet untried) actions (interventions)

two ways:

- intervene via randomized controlled trials (RCT): set action randomly and check effect → physically disassociate cause from confounders
- causal model: emulating interventions by smart calculations → predict effects of potential interventions from observational studies only, i.e., without experiment

# Solution for Average Causal Effects

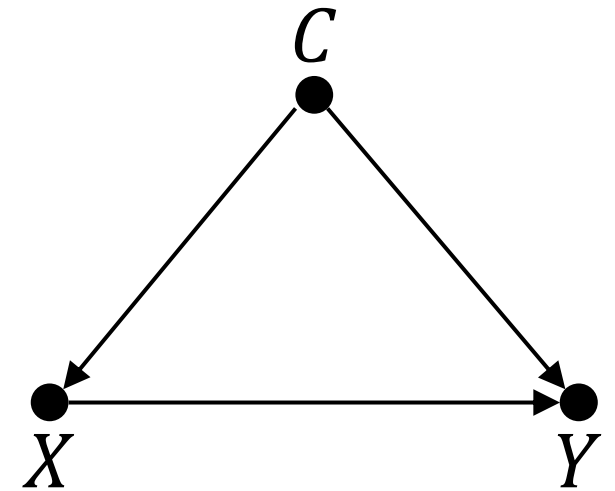
causal effect of  $X$  on  $Y$  confounded by common cause  $C$

solution for computation of average causal effect:

adjust for  $C$  (if there are measurements of  $C$ ), i.e.,  
stratification of data in terms of  $C$

$$\sum_c P(Y|X, C = c) P(C = c)$$

fashion example: adjust for months-in-season groups



# Colliders and Mediators

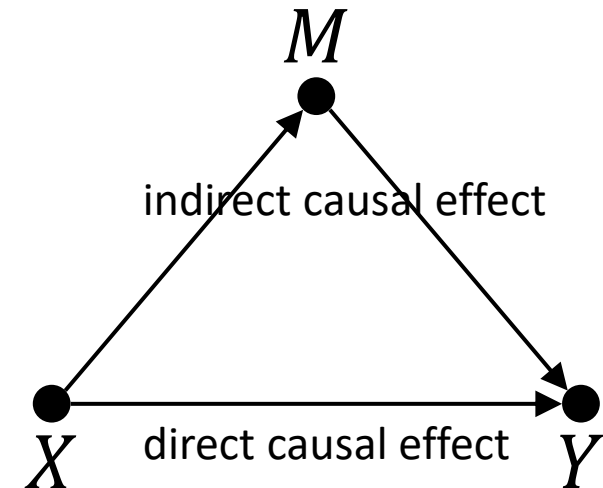
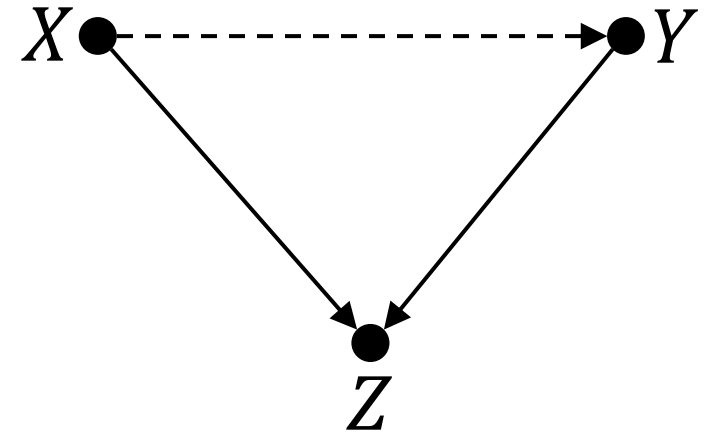
unwanted effects of adjusting for third variable on causal effect between  $X$  and  $Y$ :

collider  $Z$

- correlation without causation (exception to common cause principle)
- selection bias

mediator  $M$  (mechanism of effects)

→ need for counterfactual description



# Back-Door Criterion

to stop information flow from  $X$  to  $Y$  (d-separation):

- $X \rightarrow Z \rightarrow Y \quad \rightarrow$  adjust for  $Z$
- $X \leftarrow Z \rightarrow Y \quad \rightarrow$  adjust for  $Z$
- $X \rightarrow Z \leftarrow Y \quad \rightarrow$  do **not** adjust for  $Z$

back-door path: any path from  $X$  to  $Y$  starting with an arrow pointing into  $X$

de-confounding  $X$  and  $Y$  means blocking all back-door paths  
(which need to be identified before)

# *do*-Calculus

effect of intervention represented by  $P(Y|do(X))$

aim: calculate  $P(Y|do(X))$  in terms of data such as  $P(Y|X, A, B, \dots)$

no *do*-operator remaining  $\rightarrow$  can be calculated from observational data

$do(X)$  means removing all arrows of a causal diagram going into  $X$

definition of confounding:  $P(Y|X) \neq P(Y|do(X))$

back-door adjustment:  $P(Y|do(X)) = \sum_c P(Y|X, C = c) P(C = c)$

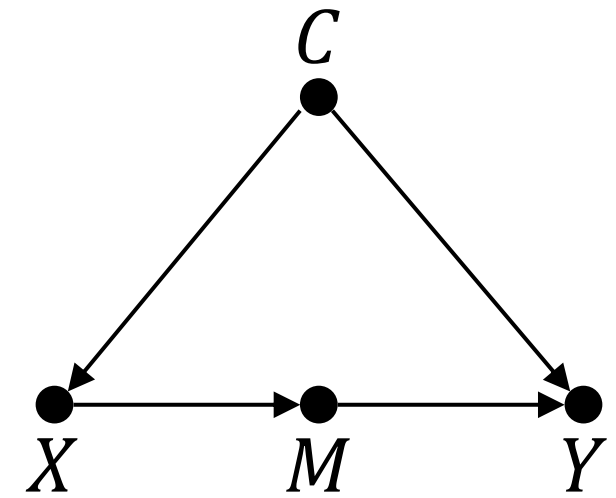
# Front-Door Criterion

unobservable confounder  $\rightarrow$  no back-door adjustment

front-door adjustment by means of observable mediator  $M$ :

$$P(Y|do(X)) = \sum_m P(M = m|X) \sum_x P(Y|X = x, M = m) P(X = x)$$

adjust for two variables:  $M$  and  $X$



derived from simple axioms of do-calculus:

- observation of  $W$  irrelevant to  $Y$ :  $P(Y|do(X), Z, W) = P(Y|do(X), Z)$
- $Z$  blocking all back-door paths:  $P(Y|do(X), Z) = P(Y|X, Z)$
- no causal paths from  $X$  to  $Y$ :  $P(Y|do(X)) = P(Y)$

# Individual Causal Effects

# Counterfactuals

so far: average causal effects over (sub-)populations by means of interventions (RCTs, *do*-calculus)

individual causal effects, acting on individual units  $u$ , as counterfactuals:

*What happened?*      and      *What could have happened?*

only one realization per individual: cannot simply measure difference between respective potential outcomes (all but one purely hypothetical)

counterfactual notation (had  $X$  been  $x$ ):  $Y_{X=x}(u)$



# Estimation of Individual Causal Effects

impossible to correctly estimate individual causal effects by means of statistical interpolation techniques (impute missing data of potential outcomes)

→ need for additional causal assumptions (a causal model)

two different approaches:

1. structural causal models (Pearl): guided by causal graph and response functions
2. Rubin causal model (potential outcomes): guided by structural assumptions (e.g., the need for adjusting for all confounders)

# Structural Causal Models (SCM)

also known as structural equation models (SEM)

modeler needs to specify which (causal graph) and how (deterministically, e.g., linearly) variables interact

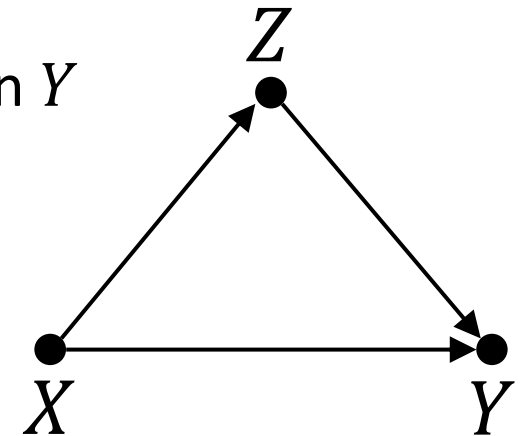
→ average and individual causal effects can be inferred from the model

example: direct and indirect (via mediator  $Z$ ) causal effects of  $X$  on  $Y$

$$Y(u) = f_Y(X(u), Z(u), U_Y(u))$$

$$Z(u) = f_Z(X(u), U_Z(u))$$

$U$ : effects from unobserved  
(exogenous) variables



# Counterfactual Queries in SCM

receipt for counterfactual queries on individual  $u$ :

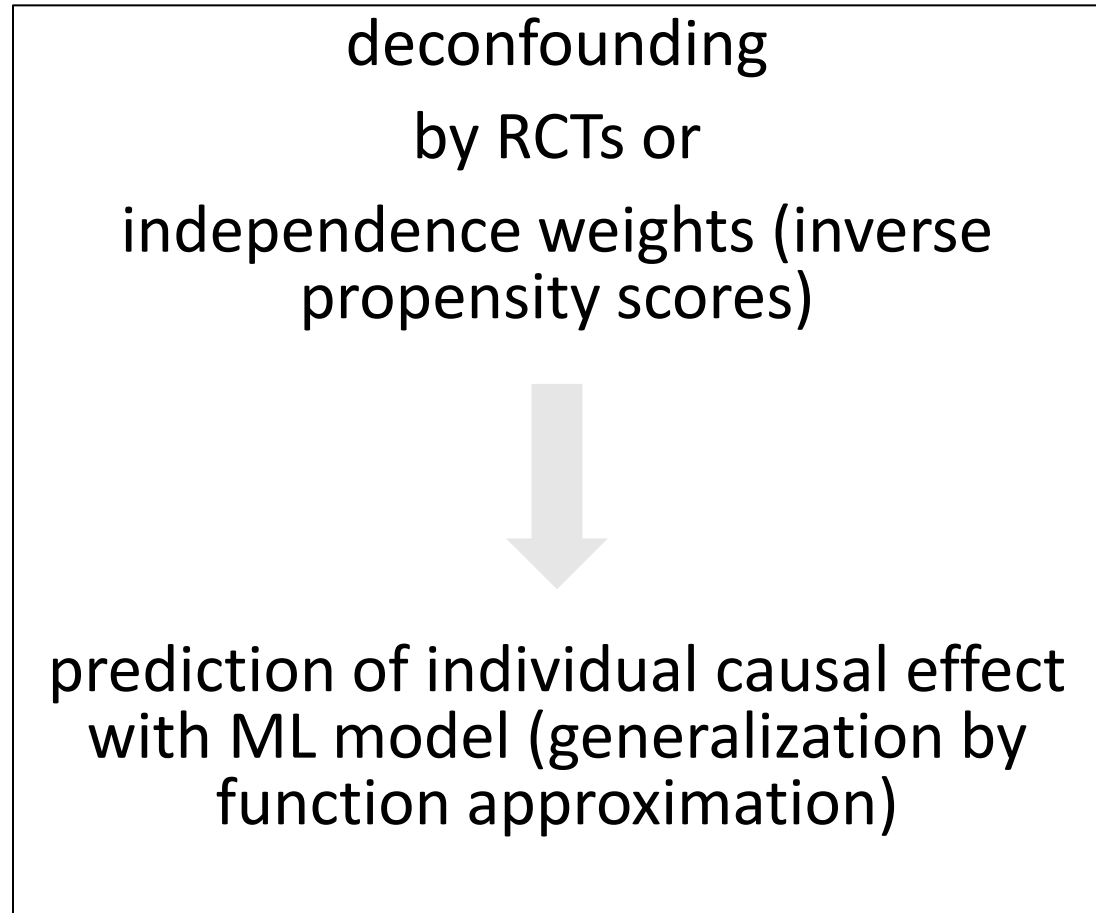
1. abduction: use data to estimate individual  $U(u)$  (potentially ML)
2. action: *do*-operation to change model according to counterfactual query (erasing arrows)
3. prediction: use modified model and updated information on  $U(u)$  to estimate individual causal effect

connection to generative models:

unobserved variables  $U$  in SCMs correspond to latent noise variables in generative models

→ both use reparametrization trick: randomness as exogenous model input (rather than intrinsic component)

# Potential Outcomes with ML

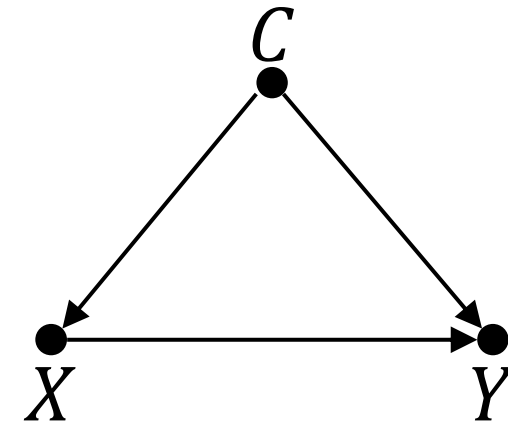


mimicking RCT: propensity score  
(probability of  $u$  being subject to  
action) matching (importance  
weighting)

(see also doubly-robust methods)

can be seen as reverse engineering  
of artificial control group: find  
(averaged) similar individuals in  
data as potential outcome partners

# Independence Weights with ML



scenario:  $X$  binary (for simplicity), several confounders  $C$

aim: prediction of individual causal effect using observational data only

- i. ML model to predict past action policy  $P(X|C)$  (beware: need to include all  $C$ )
- ii. inverse propensity score weighting of each unit (to adjust for multiple confounders):

$$P(Y|do(X)) = \sum_z P(Y|X, Z = z) P(Z = z) = \sum_z \frac{P(Y, X, Z = z)}{P(X|Z = z)}$$

} alternative:  
A/B test

- iii. train ML model on deconfounded data to predict  $Y$  with  $X$  and  $C$  as features
- iv. individual causal effect as difference between predictions for setting feature  $x = 1$  and  $x = 0$ , respectively (what-if scenario)

# Causal Discovery

# Causal Discovery from Observations

conditional independence testing can reveal aspects of causal graphs by means of observed statistical dependencies

subject to causal Markov property and causal faithfulness assumption:  
conditional independence only between d-separated variables in a DAG

hard for finite data sets without additional assumptions

→ need for approximative methods

# Causal Discovery Methods

causal Bayesian networks: estimating relationships between all variables without the need to specify interactions before (like in SEMs)

restriction of function classes (e.g., smoothness): breaking cause-effect symmetry (e.g., by means of additive noise model)

meta-learning of causal structure based on speed of adaption to modified distributions ([Bengio](#))



# Graph Neural Networks (GNN)

# Example: Application in Particle Physics

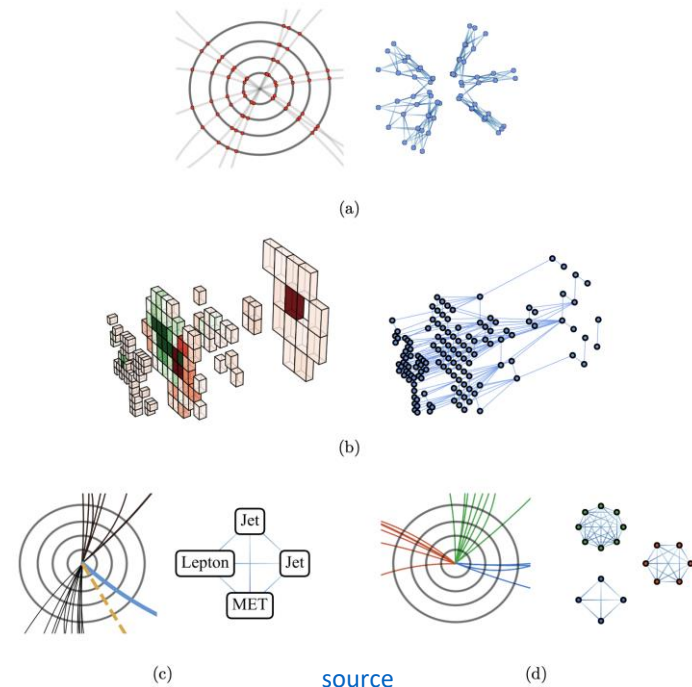
ML used in particle physics experiments for quite some time, e.g., for signal classification in detector measurements

but usual ML methods often require unnatural data representations

many problems graph-like  $\rightarrow$  GNN

many applications ([review](#)), e.g.,

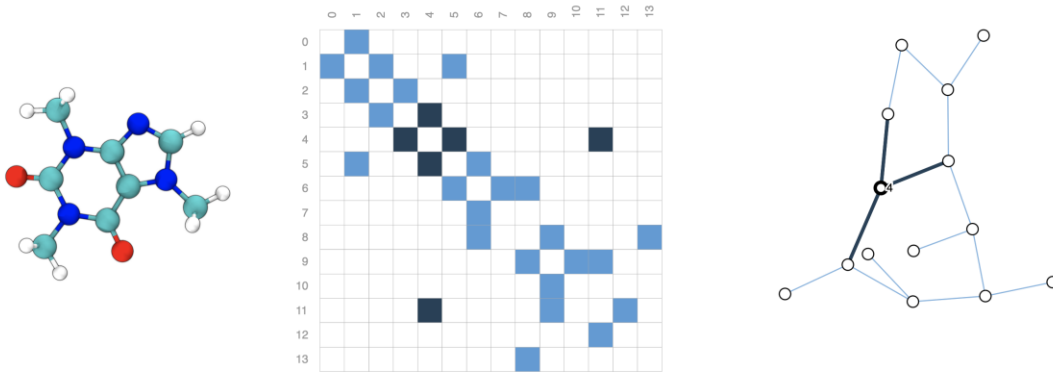
- jet reconstruction and classification
- [classification of astrophysical neutrinos](#)



[source](#)

# Graph Data

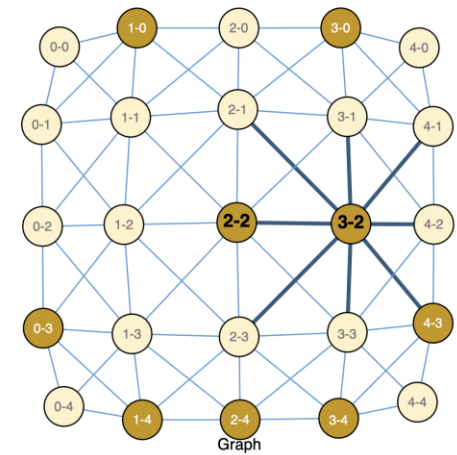
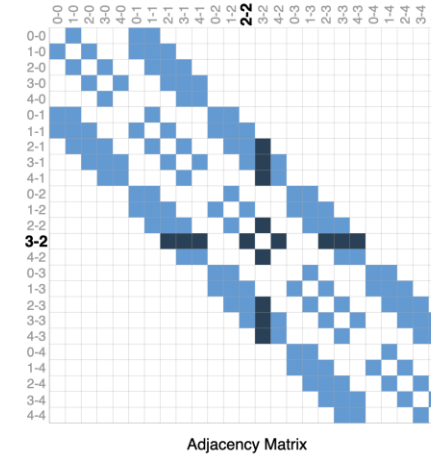
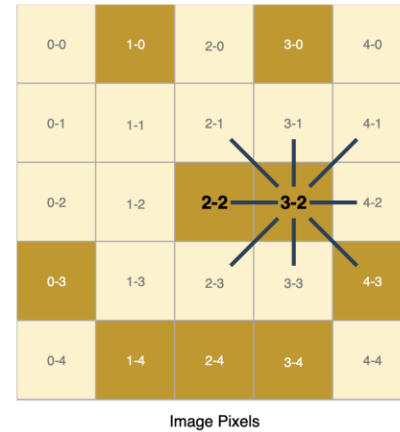
graphs: molecules, delivery routes, or social networks, but also images or text



(Left) 3d representation of the Caffeine molecule (Center) Adjacency matrix of the bonds in the molecule (Right) Graph representation of the molecule.



(Left) Image of a scene from the play "Othello". (Center) Adjacency matrix of the interaction between characters in the play. (Right) Graph representation of these interactions.



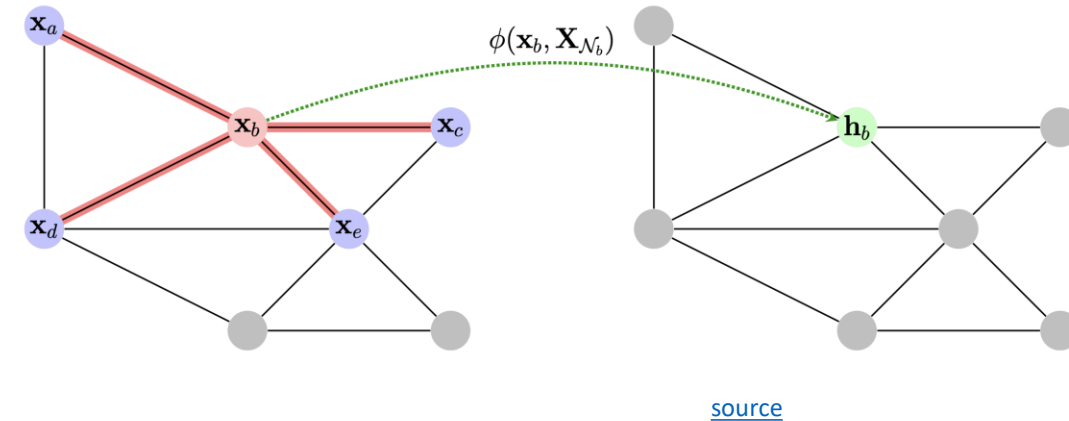
nodes  $\times$  nodes adjacency matrix

all taken from [this](#) nice introduction to GNNs

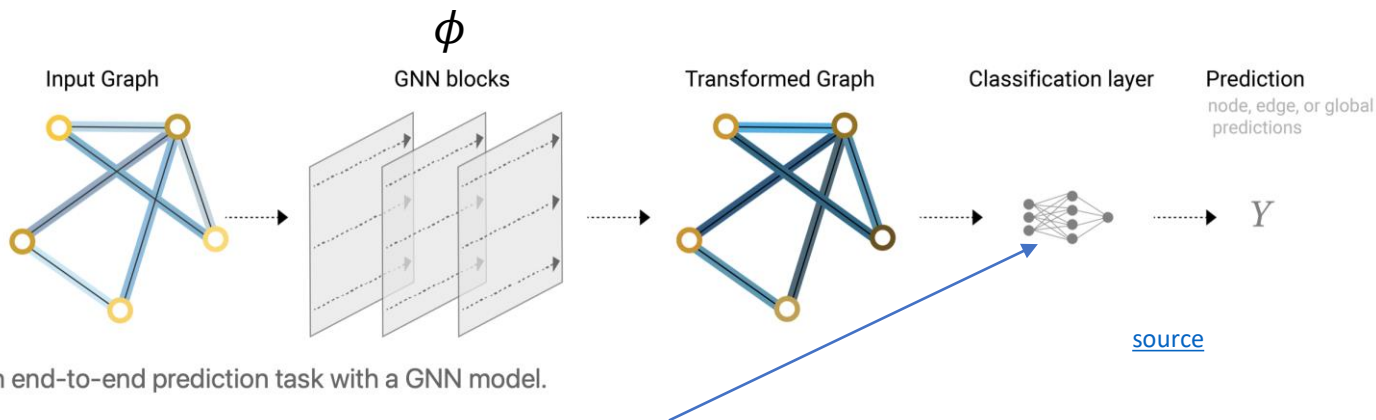
# Neural Networks on Graph Data

graph can be specified with adjacency matrix as well as node (vertex), edge (link), and global attributes/features → learned as representation (graph encoder, embeddings)

often, transductive setting: just one partially labelled graph rather than training and test graphs/samples (as in usual inductive ML setting)



GNNs: neural networks operating on graph data



global pooling layer: aggregation of node/edge representations (can also be node/edge predictions via adjacency instead of global one)

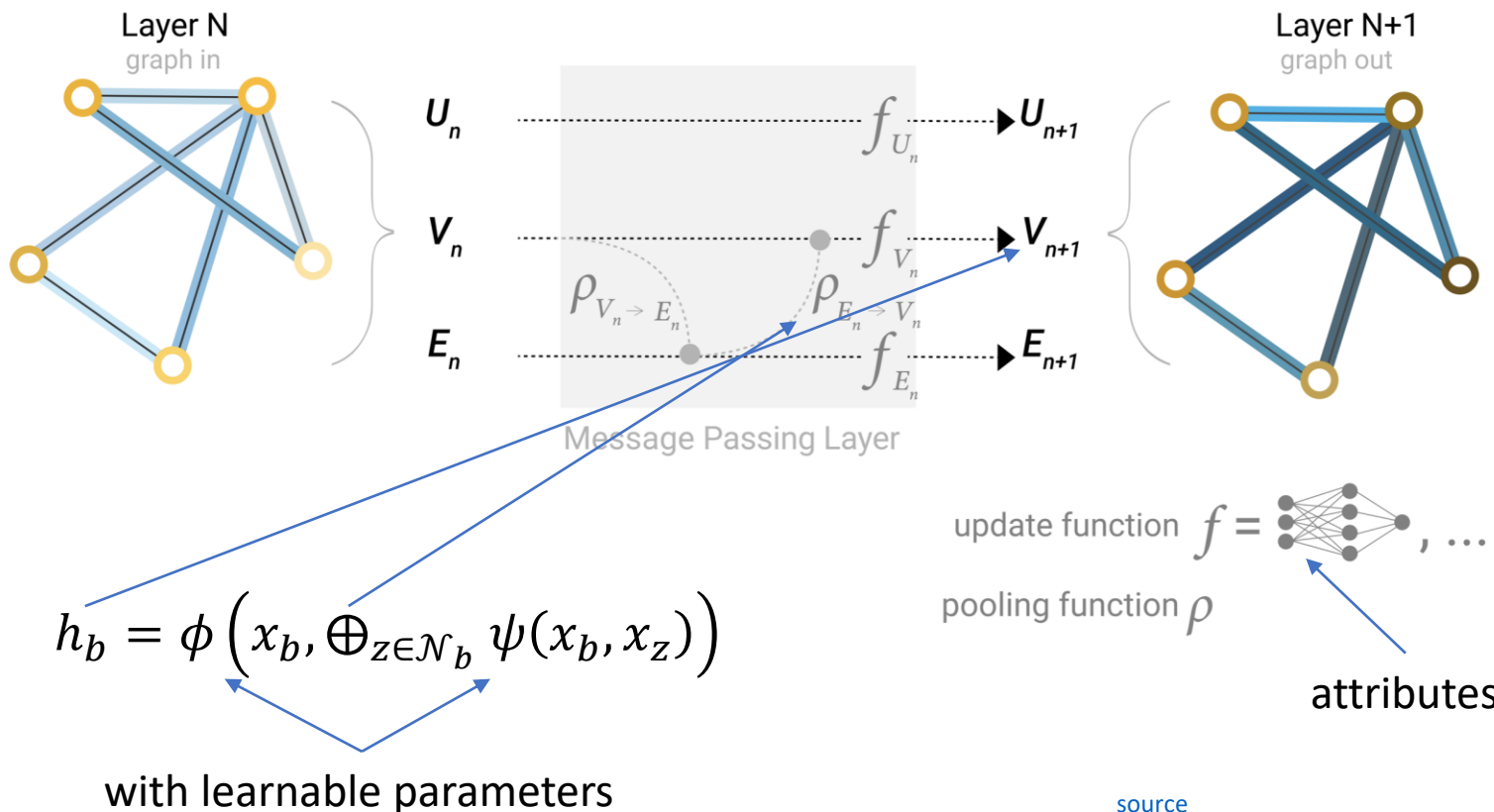
different tasks:

- graph-level: predict property of entire graph (e.g., molecule binding to receptor or labeling of image)
- node-level: predict the identity of nodes (e.g., image segmentation)
- edge-level: e.g., image scene understanding

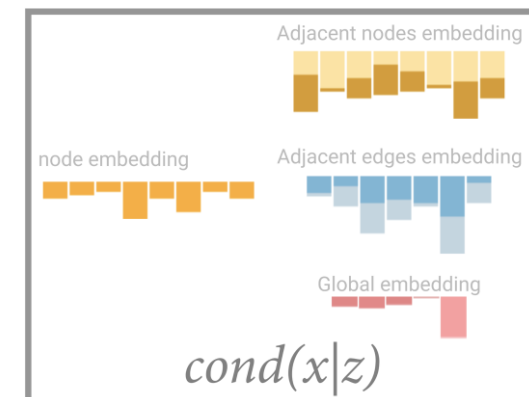
potentially also generative (decoder): e.g., generating routes for traveling salesman problem

# GNN Layers and Message Passing

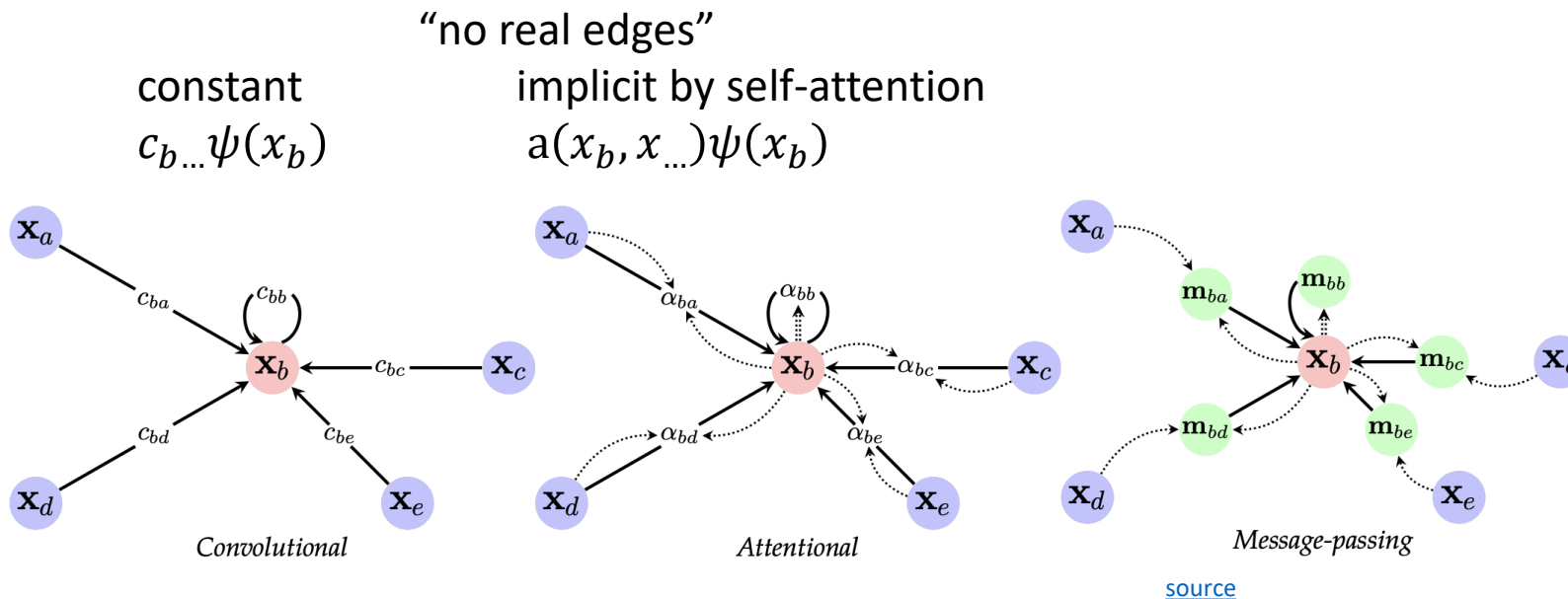
updating node, edge, and global-context representations by message passing



each message passing layer corresponds to one hop in graph  
→ need for lots of stacking to reach full graph  
→ add global representation  $U$  (context vector) connected to all nodes and edges



# Different Types of GNN Layers



convolutional special case of  
attentional GNN layers ([GAT](#))

both convolutional and  
attentional special cases of  
message passing GNN layers

GNNs very general deep learning architecture: most other architectures  
special cases of GNNs with additional geometric structure (i.e., different  
inductive biases)

e.g., transformers in NLP operating over complete graph of words

# The Bigger Picture: Symmetries

idea of [Geometric Deep Learning](#) (borrowed from geometry):

derive different inductive biases and network architectures from symmetries and invariances

CNN from translation invariance of convolutions

GNNs from permutation invariance of graph nodes:

need for permutation-equivariant GNN layers with permutation-invariant aggregations (e.g., element-wise sum)

# Link to Causality

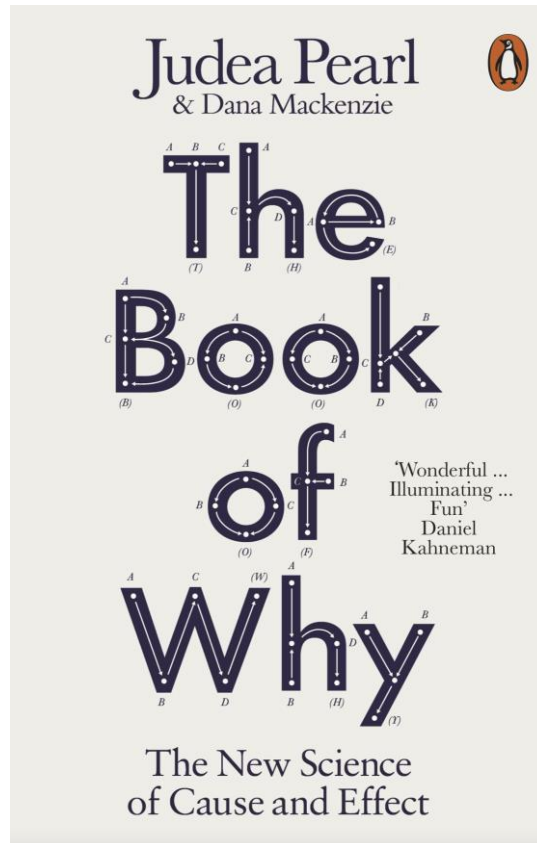
[neural causal models](#): connect SCMs and neural networks (gradient descent)

one step further: connection between [SCMs and GNNs](#)

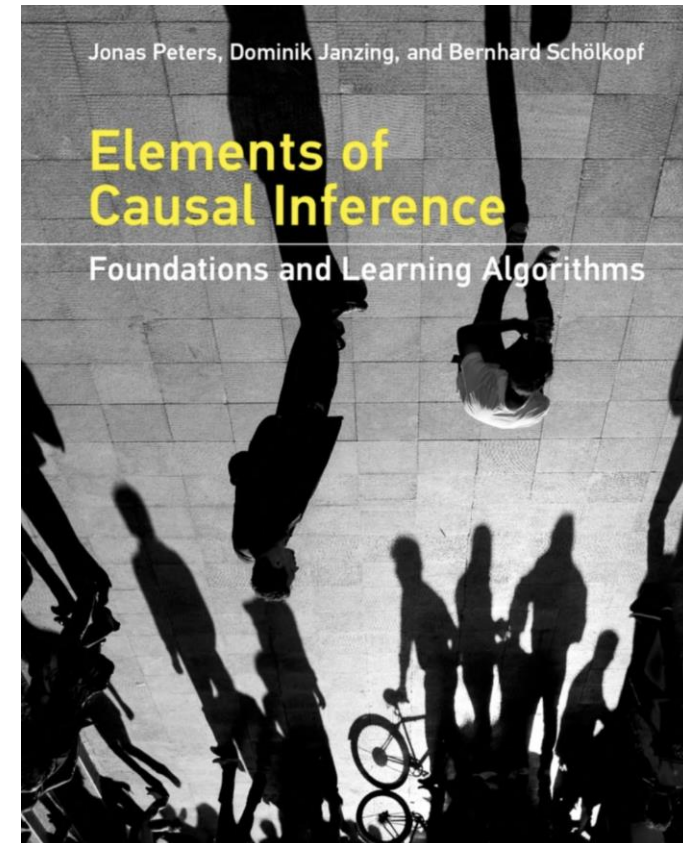


# Literature

gentle but genuine introduction,  
including most ideas:



... and with a little bit more maths 😊:



# Fictional Thinking

fictional thinking (imagination of non-existent things) drastically improved human's communication and collaboration abilities (see *Sapiens: A Brief History of Humankind* from Harari)

counterfactual causal reasoning as base of human's fictional capabilities

enrich ML with causality:

- not only important for long-term goal of human-level AI
- also enabling better human-AI interaction (common causal language)