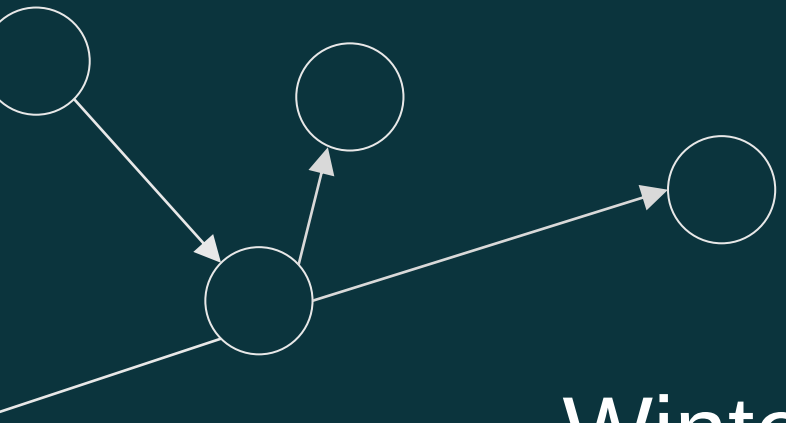


TECHNISCHE UNIVERSITÄT DARMSTADT

AIML Lab

Winter Semester 2025/26 Lecture

Causality for AI & ML

"Causal Discovery"

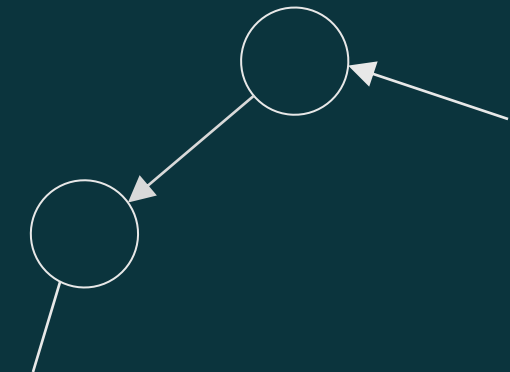
Prof. Dr. Kristian Kersting

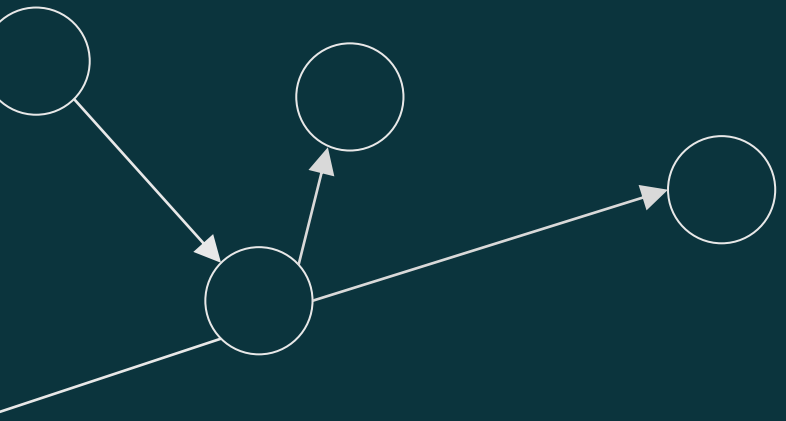
Moritz Willig

Tim Woydt

Florian Busch — Today's speaker

Matej Zečević

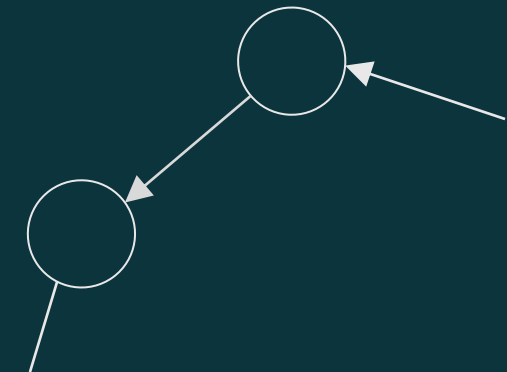




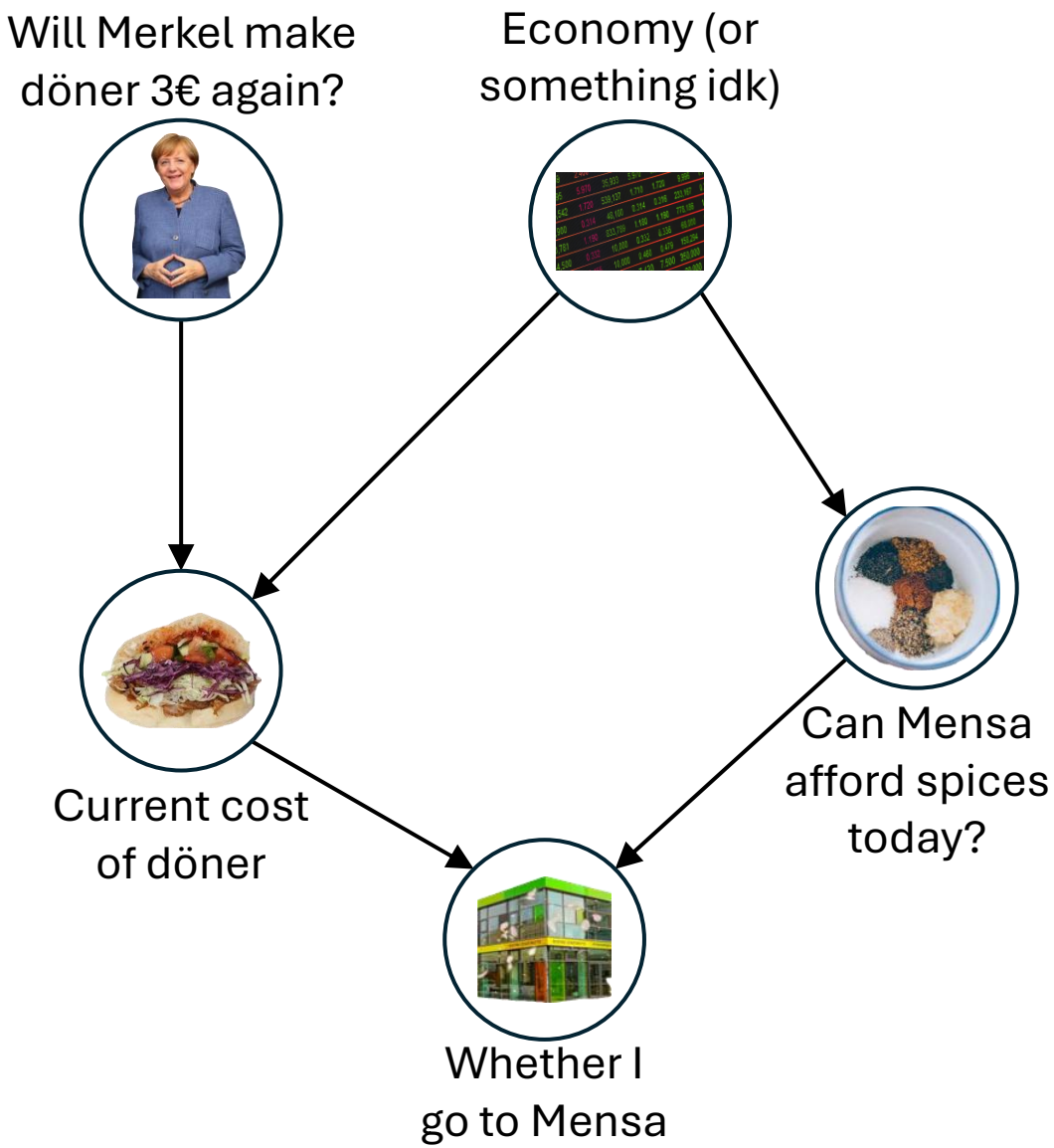
Section

1

Causal Discovery Basics



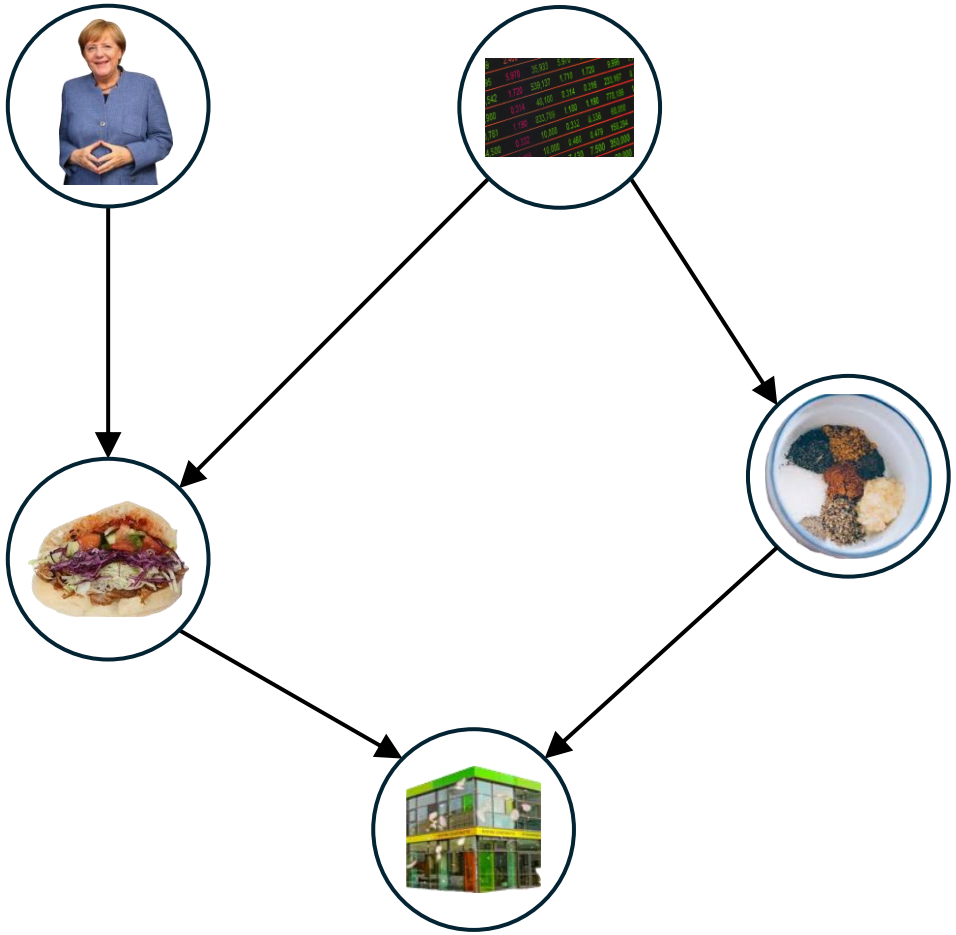
Causal Graphs Can Be Very Useful!



- Answer probabilistic queries, e.g., *What is the probability that I go to Mensa today, given that Merkel made Döner 3€ again?*
- Answer causal queries, e.g., *If we intervene to provide lots of spice to Mensa, does the cost of döner change?*
- Determine causal effects (do calculus)
- …

Thanks to my HiWi Florian Guldan for providing this example
Not to be taken seriously ;)

How to Understand a Causal Graph



- Variables are determined by their parents
- Variables provide information on many other variables (all variables, to which they are not d-separated)
- But a variable only has **an effect** on all descendants (→ interventions)

➔We can answer many causal queries, maybe even including interventions and counterfactuals

But Often the Causal Graph Is Unknown

Experts can provide edges

- This is often done
- But it can be expensive and unrealistic

Just ask an LLM

- Can work out, but generally not reliable
- *(a bit more on that later)*

But there is a lot of data

- Can we discover causal relations from data? → **Causal Discovery**

Causal Discovery (CD)

[1] Zanga, Alessio, Elif Ozkirimli, and Fabio Stella. "A survey on causal discovery: theory and practice." *International Journal of Approximate Reasoning* 151 (2022): 101-129.

Discover the true causal graph from tabular data

Causal Discovery

Let $\boldsymbol{G}$ be the set of graphs defined over the variables $\boldsymbol{V}$ of a dataset $\boldsymbol{D}$ and $G^* \in \boldsymbol{G}$ be the $\textit{true but unknown graph}$ from which $\boldsymbol{D}$ has been generated. The **causal discovery** problem consists in recovering the *true* graph G^* from the given dataset $\boldsymbol{D}$. [1]

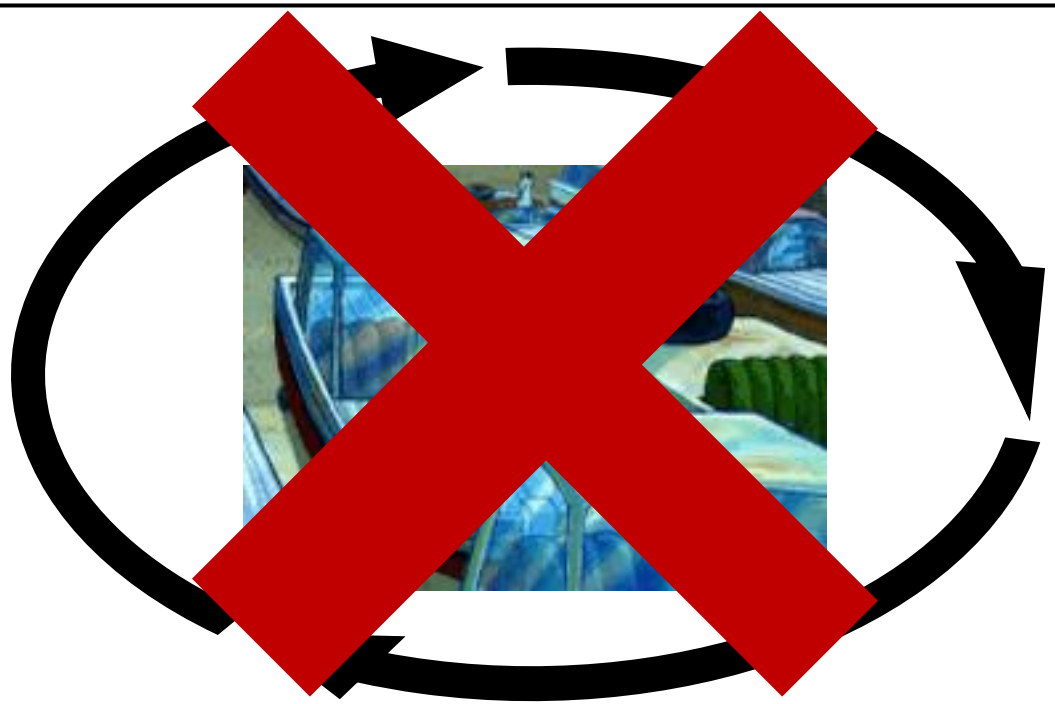
→Variables are known and there is data for all variables

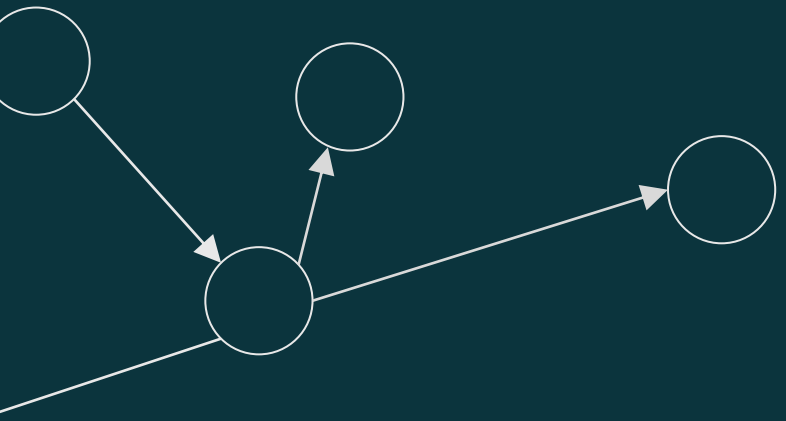
(a later lecture: causal representation learning, where the variables must be inferred from higher dimensional and / or entangled data representations)

Acyclicity

- Often, causal discovery is not restricted to acyclic graphs (DAGs)
- However, most approaches and applications focus on DAGs

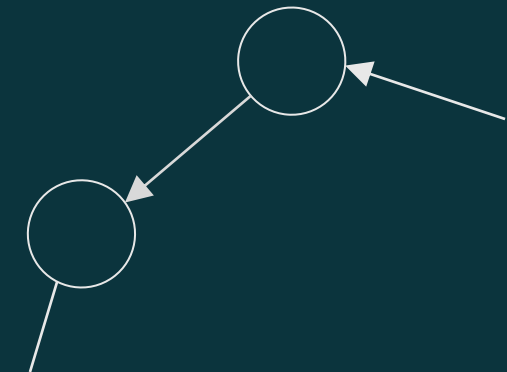
Until the very end of this lecture, we assume acyclicity for all problems





Section 2

Constraint-Based Causal Discovery



From d-separation to Causal Discovery

Reminder: d-separation

The graph structure induces independencies

Variables are not independent

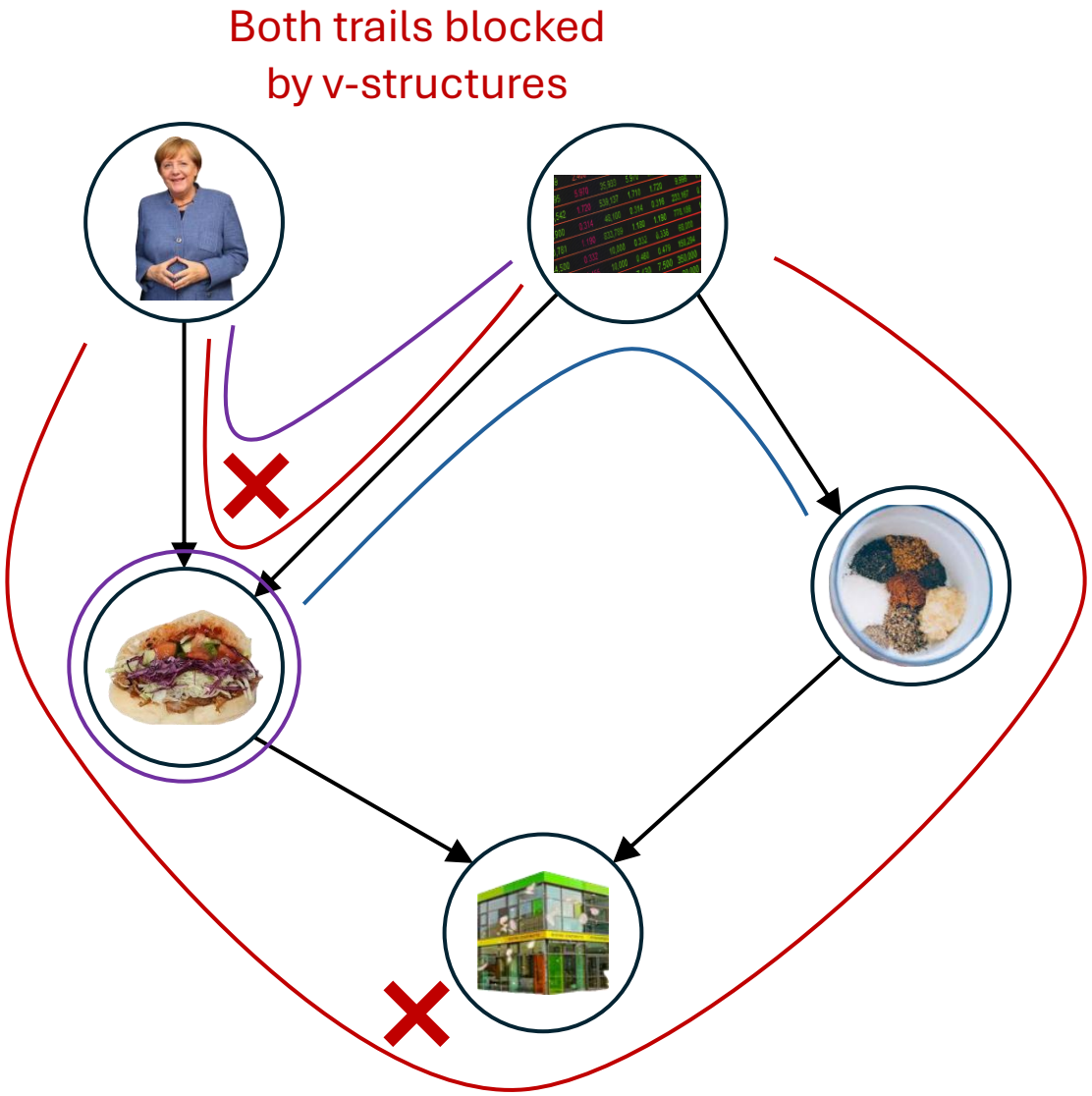
→there must be an active trail

Variables are independent

→there is no active trail

Variables are independent but become dependent after conditioning

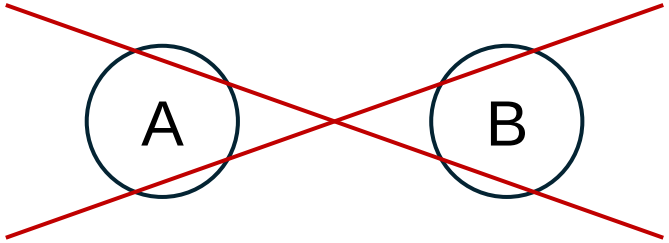
→ some v-structure involved



From d-separation to Causal Discovery

Does this mean we can easily detect the entire causal graph?

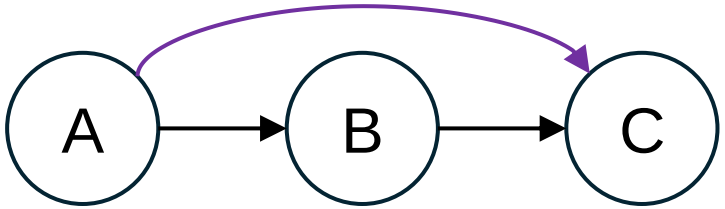
- For example: consider A and B are **not** independent
- No, independency alone is not enough to identify all directions

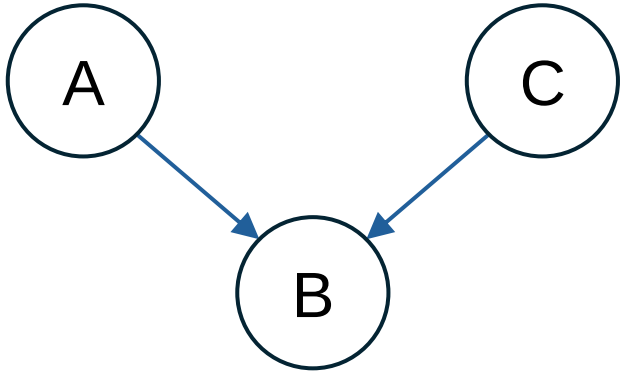


Two possible graphs

Then can we at least predict some edge directions?

- Yes, because of v-structures + acyclicity
- More details in a few slides





Core Assumptions (reminder from previous lectures)

P_M is called Markovian to G_M if all independencies implied by the graph also hold true in the distribution:

Markov Condition

$$(X \perp\!\!\!\perp Y|Z)_G \Rightarrow (X \perp\!\!\!\perp Y|Z)_P$$

G_M is called Markovian to P_M if all independencies implied by the distribution also hold true in the graph:

Faithfulness

$$(X \perp\!\!\!\perp Y|Z)_G \Leftarrow (X \perp\!\!\!\perp Y|Z)_P$$

➔ Together: The independencies of the probability distribution and the causal graph are identical (P-Map)

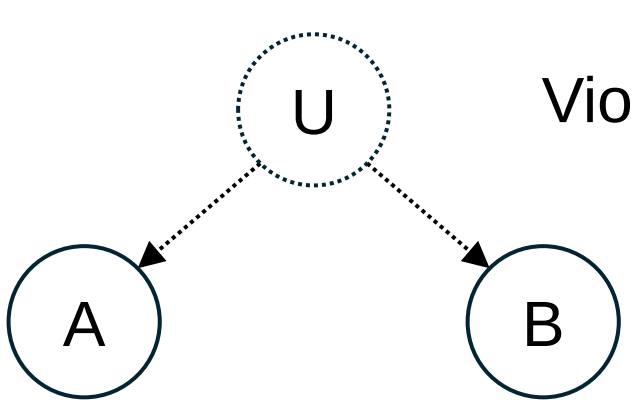
Peter-Clark Algorithm (PC) [2]

[2] Spirtes, Peter, Clark N. Glymour, and Richard Scheines. *Causation, prediction, and search*. MIT press, 2000.

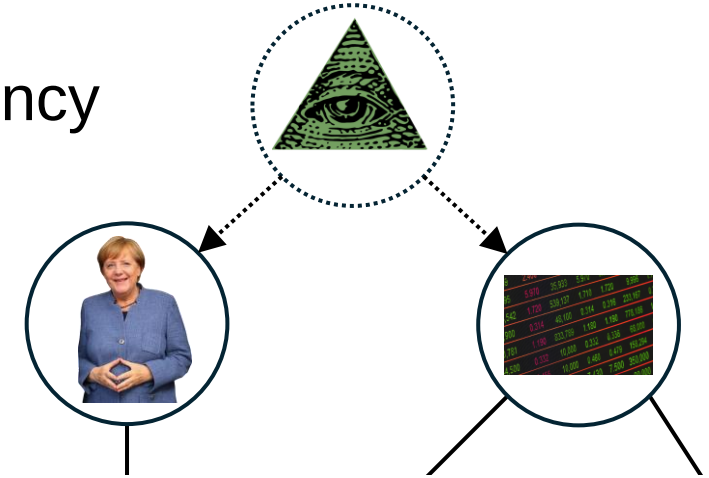
Assumptions: acyclicity, Markov, faithfulness, sufficiency (new)

Causal Sufficiency

A set V of variables is **causally sufficient** for a population if and only if in the population every common cause of any two or more variables in V is in V, or has the same value for all units in the population. [2]



Violations of Sufficiency



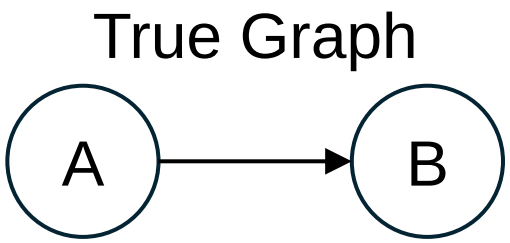
Intuition: If sufficiency does not hold, variables that should be independent in the graph (Markov condition) are correlated. This can not be fixed by drawing a directed edge between those, as either direction would be false.

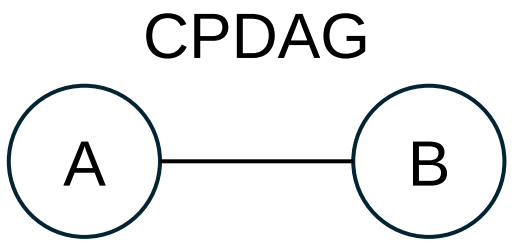
Peter-Clark Algorithm (PC)

Input: Dataset $\boldsymbol{D}^{m \times n}$ (m: number of samples, n: number of variables)

Output: CPDAG (Completed Partially Directed Acyclic Graph)

- A CPDAG represents the Markov Equivalence Class for the true graph
- Some edges can remain undirected
- An undirected edge between A and B means that either A causes B (A → B) or that B causes A (B → A); NOT that they could be confounded
- For example:





Peter-Clark Algorithm (PC)

Phase I: Skeleton discovery (all edges are undirected)

1. Start with fully connected graph
2. Remove all pairwise or conditionally independent variable edges

Phase II: Direct edges wherever possible

3. Direct edges using v-structures

- For all unshielded triplets A—B—C, i.e., A and C are not connected, direct it as A→B←C if B was not part of the conditioning set establishing independence between A and C before (phase I, step 2)

4. Meek rules

- Direct edges that can only be directed in one direction (more details soon)

Food for thought: Why does this work?

PC – 0. Input Data

				
3 hours	**High**	**Bad**	**Low**	**No**
0 hours	**None**	**Good**	**High**	**Yes**
2 hours	**Low**	**Good**	**High**	**Yes**
3 hours	**High**	**Bad**	**Low**	**No**
1 hour	**Low**	**Bad**	**High**	**No**
⋮	⋮	⋮	⋮	⋮

Duration of gaming session (hours)

Caffeine consumption [None, Low, High]

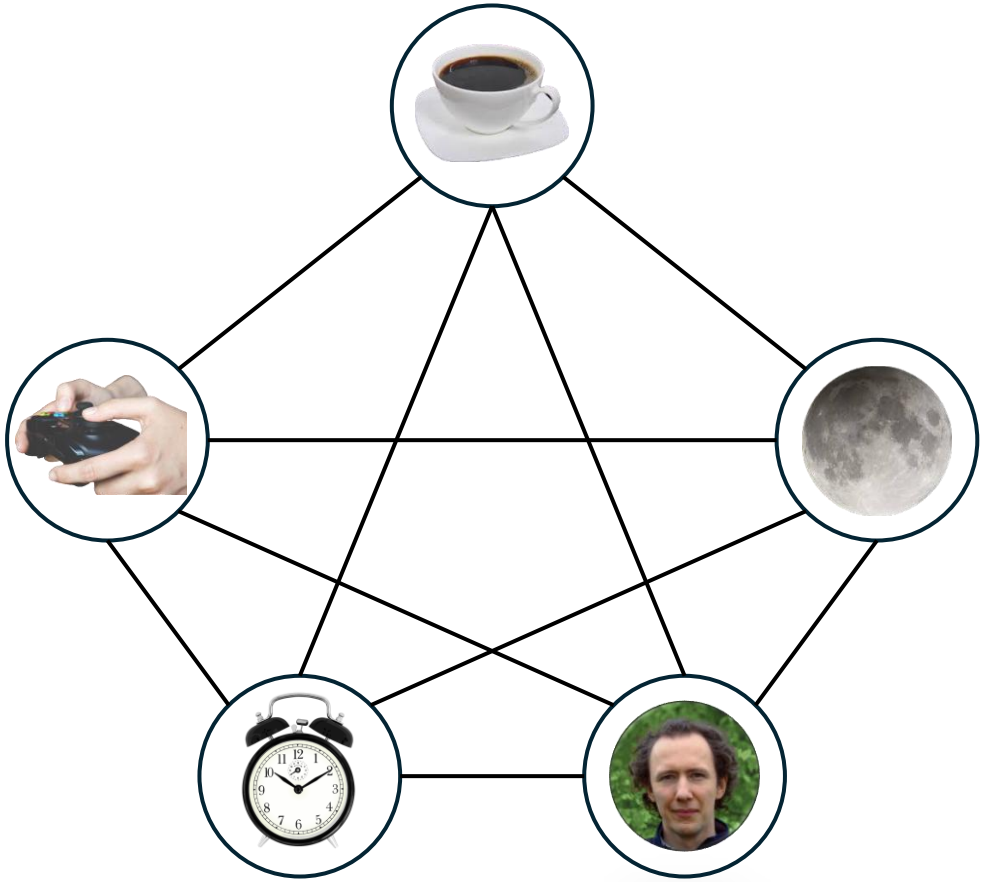
Sleep quality [Bad, Good, Low]

Motivation to get up [High, Low]

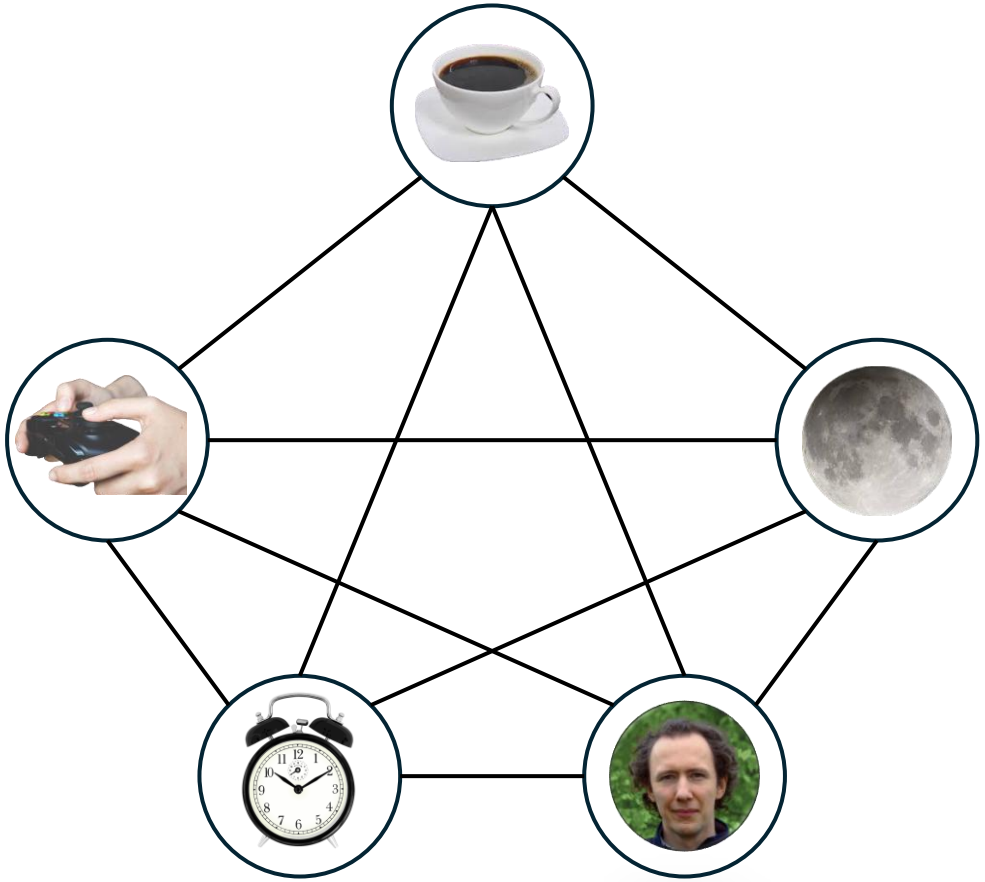
Presence at lecture [Yes, No]

Side note: Use the appropriate independency tests when dealing with discrete vs continuous vs mixed variables

PC - 1. Start with Fully Connected Graph with Undirected Edges



PC - 2. Remove All Pairwise or Conditionally Independent Edges

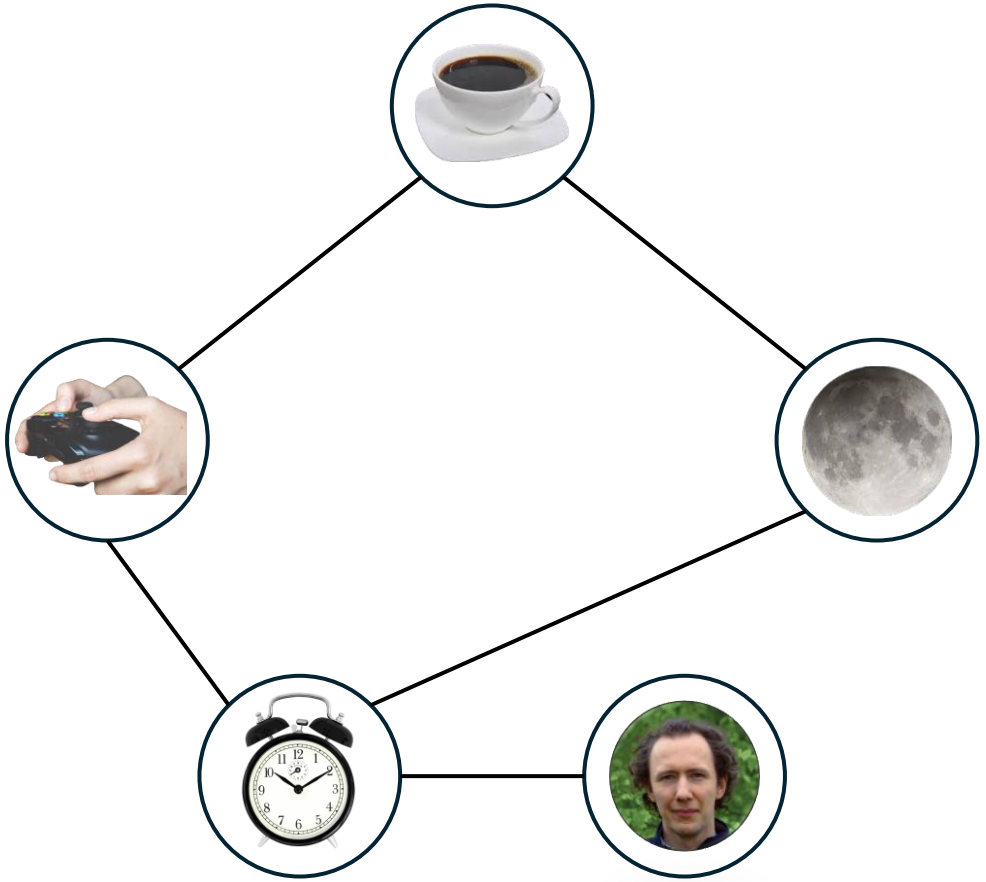


No pairwise independencies

Condition on 1 variable:

Condition on 2 variables:

PC - 2. Remove All Pairwise or Conditionally Independent Edges



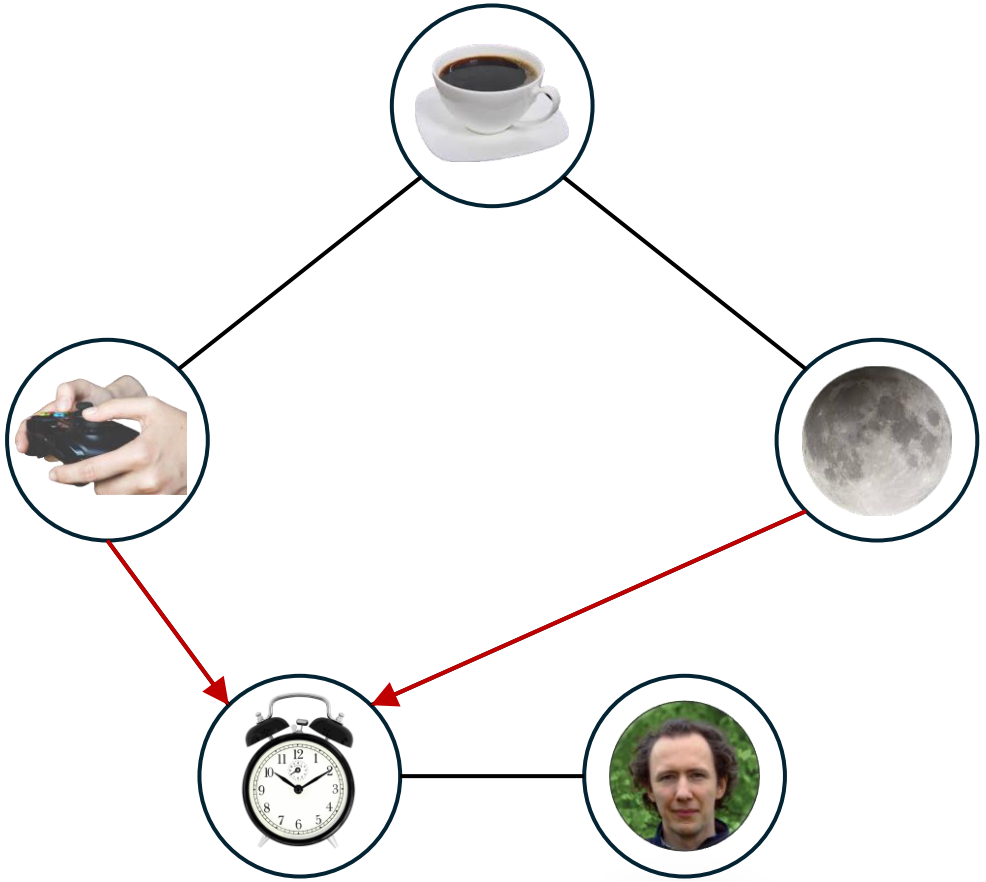
No pairwise independencies

Condition on 1 variable:

Condition on 2 variables:

No further independencies

PC – 3. Direct v-structures



Iterate over all triplets

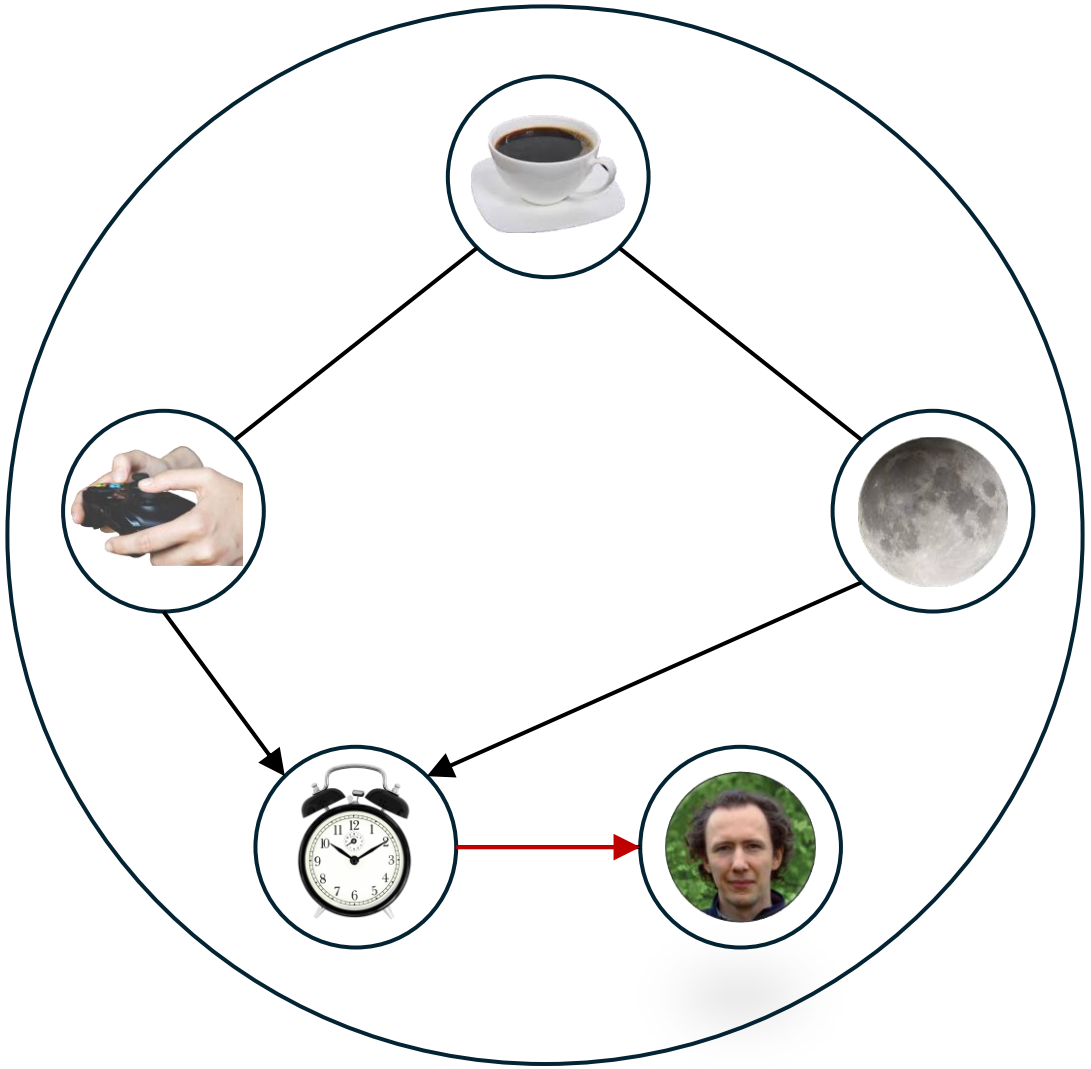
Analyze independencies, for example:

- Consider the triplet 🤳 - ⏰ - 🌕
- This edge was removed by conditioning on ☕
- ⏰ was not in the conditioning set

→v-structure

We detect no further v-structures on the other triplets

PC – 4. Meek Rules



Iterate over Meek Rules until convergence

- If  →  would be the true edge direction, we should have discovered the v-structure  →  → 
- Therefore, direct in the other direction

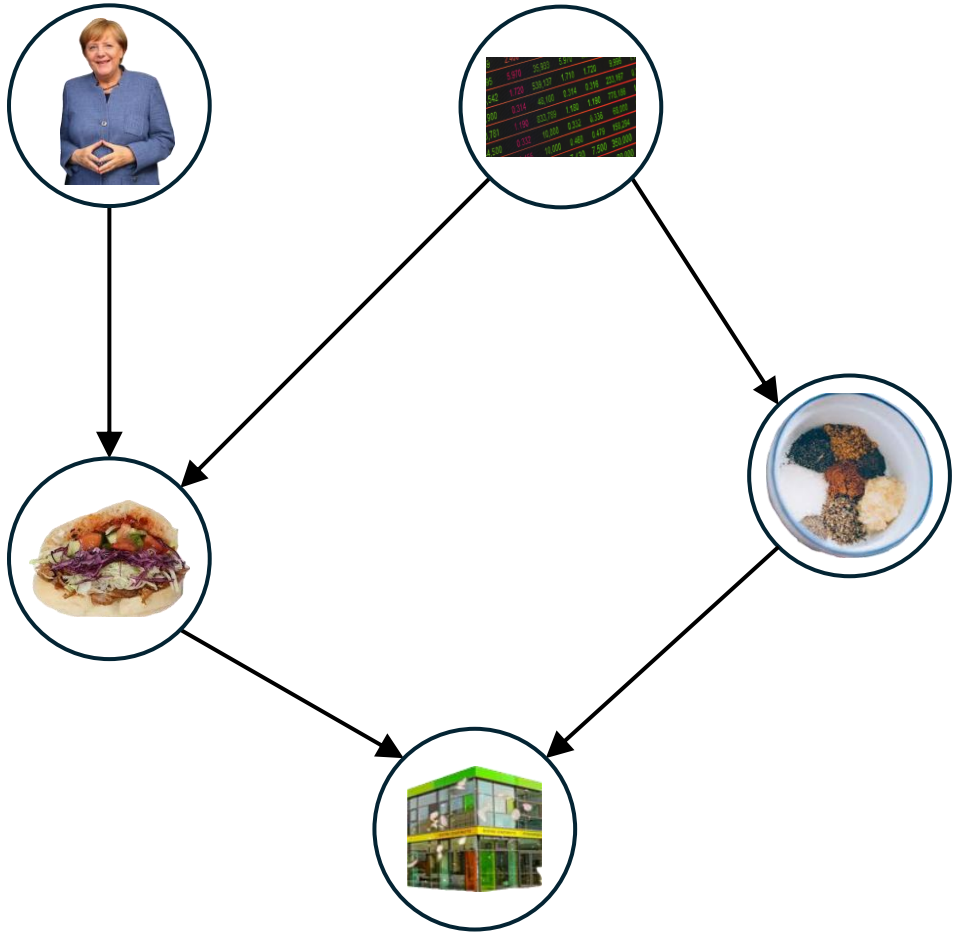
This is the final CPDAG

PC – Exercise

1. Independency and d-separation

Write down all pairwise and conditional independencies for the displayed Mensa graph. You can ignore all independencies that do not have a minimal conditioning set.

Minimal conditioning set: If $X \perp\!\!\!\perp Y$ but also $X \perp\!\!\!\perp Y \mid Z$, you don't have to write down $X \perp\!\!\!\perp Y \mid Z$, as the minimal conditioning set for X and Y is the empty set.



PC – Exercise

2. PC Algorithm

Imagine you want to apply the PC algorithm discovery on the Mensa problem. First, you apply pairwise and conditional independency tests on the data and record the results. Assume that this gave you exactly the independencies that you determined in the exercise from the previous slide.

Making use of these independency statements, apply the full PC algorithm. How does the final CPDAG look like?

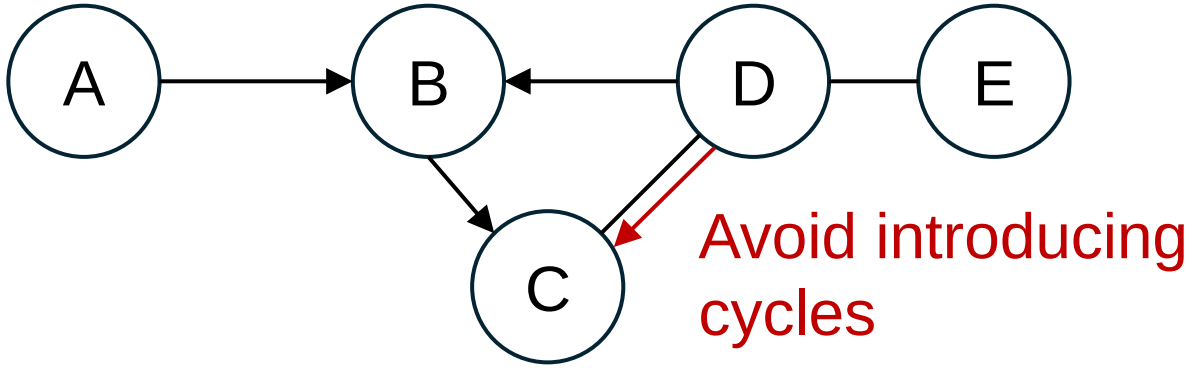




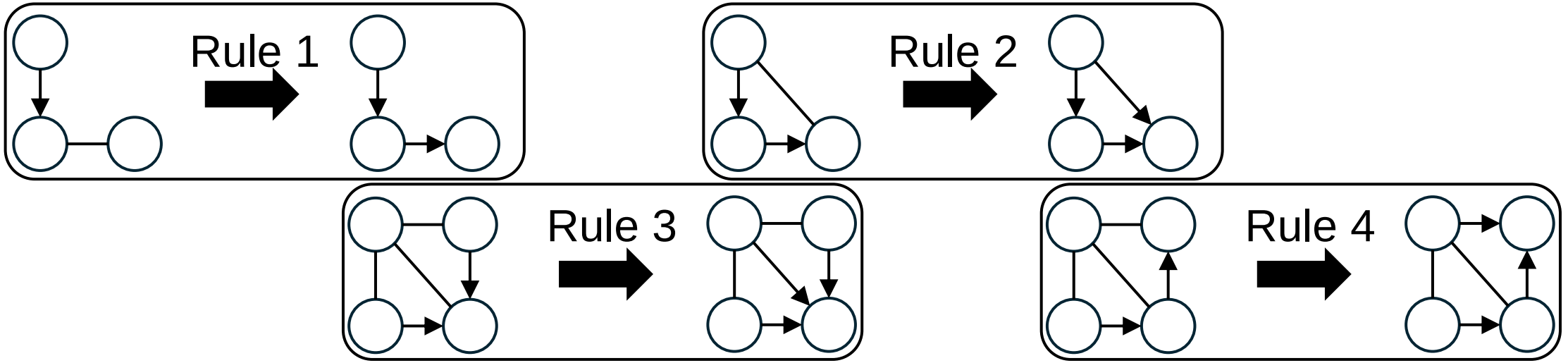
Meek Rules [3] More Generally

[3] Meek, Christopher. "Causal inference and causal explanation with background knowledge." *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*. 1995.

- Assuming acyclicity and that all v-structures have been identified
- Meek rules: 4 rules, directing edges based on these assumptions
- Another example for Meek rules:



- All rules:



Let's Relax Causal Sufficiency

Causal Sufficiency

A set V of variables is **causally sufficient** for a population if and only if in the population every common cause of any two or more variables in V is in V, or has the same value for all units in the population. [2]

In other words: there should be no unobserved confounding

Food for thought: Why does PC require an assumption on common causes but not on unobserved children or simple unobserved mediator variables?

However, even causal sufficiency can be dropped, for example: **FCI**

FCI algorithm (Fast Causal Inference algorithm) [4]

[4] Spirtes, Peter, Christopher Meek, and Thomas Richardson. "Causal inference in the presence of latent variables and selection bias." *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*. 1995.

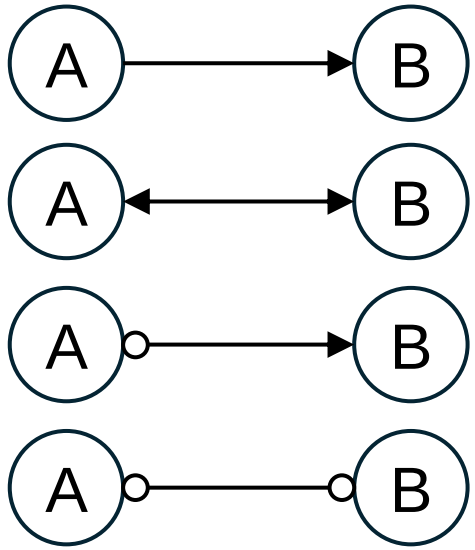
Assumptions: acyclicity, Markov, faithfulness

Input: Dataset $\boldsymbol{D}^{m \times n}$ (m: number of samples, n: number of variables)

Output: PAG (Partial Ancestral Graph)

PAG: four different edge types

- A is a cause of B:
- There is a latent common cause of A and B:
- B is not an ancestor of A:
- No set d-separates A and B:



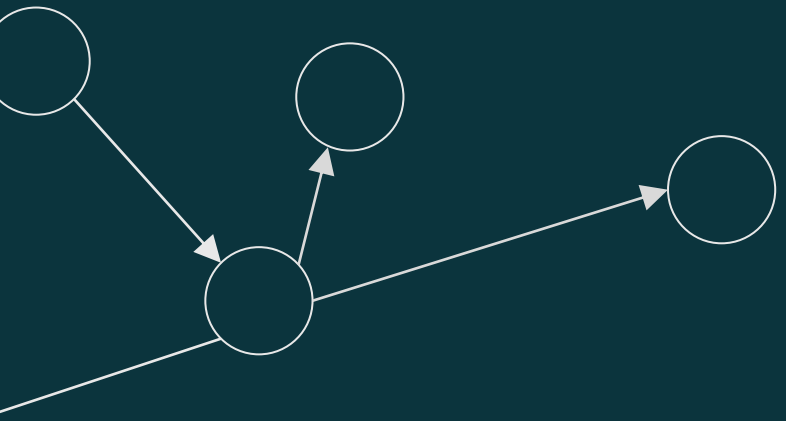
FCI implements a set of 10 different rules (general idea same as PC)

Why PC over FCI Then?

- If sufficiency is met, PC is simply better: the CPDAG is more informative than a PAG and the algorithm is more efficient as it does not have to account for the possibility of hidden confounder
- Causal discovery algorithms can fail (e.g., independency tests); always use all the information you have to solve the simplest problem possible
  - (Same in other areas: if you know that you want to learn a perfectly linear function, do not use a neural network to do so)
- Generally, many causal discovery algorithms exist for different scenarios and with different assumptions: **pick what is the best fit for your problem**

Constraint-Based Causal Discovery

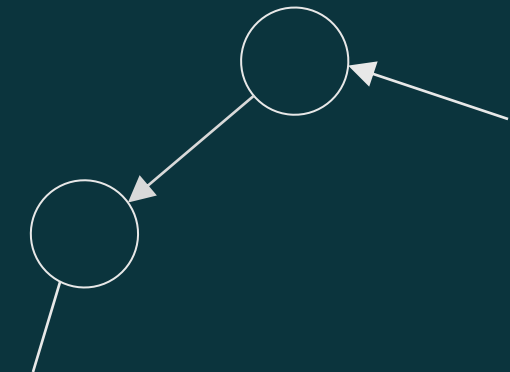
Constraint-based causal discovery algorithms infer the Markov Equivalence Class of a causal graph by systematically using statistical conditional independence tests to prune impossible edges and orient remaining ones based on the resulting constraints.



Section

3

Score-Based Causal Discovery



Score-Based Causal Discovery

Find the causal graph with the best score using a scoring criterion $S(G, \boldsymbol{D})$ [GES]

$$G^* = \text{argmax}_{G \in \mathbb{G}} S(G, \boldsymbol{D})$$

- Brute force all graphs? Super exponential growth (infeasible)
- More efficient simple strategy:

  greedily selecting the next improvement

- Risk of getting stuck in **local optima**

#Variables	#DAGS
1	1
2	3
3	25
4	543
5	29281
6	3781503
7	1138779265
8	783702329343
9	1213442454842881
10	4175098976430598143
11	3160345939641891760742 5
12	5219396513438294050205040 63
13	1867660074443203518666481 6926721
14	5143942814104439833494179071 9839535103
15	2377252655534103549921802182863767192535 05

Remember lecture 1?

Bayesian Information Criterion (BIC)

$$\mathrm{BIC}(\mathrm{G}, \boldsymbol{X}) = k \ln(n) - 2 \ln\big(p(\boldsymbol{X}|G)\big)$$

Goal: minimize BIC

- k: parameters, i.e., number of edges (**lower** is better)
- n: sample size, i.e., number of data points (not a parameter)
- $p(\boldsymbol{X}|G)$: likelihood, i.e., the probability of observing the data $\boldsymbol{X}$ given the graph G (**higher** is better)

BIC is **decomposable** (can be computed independently for all m nodes):

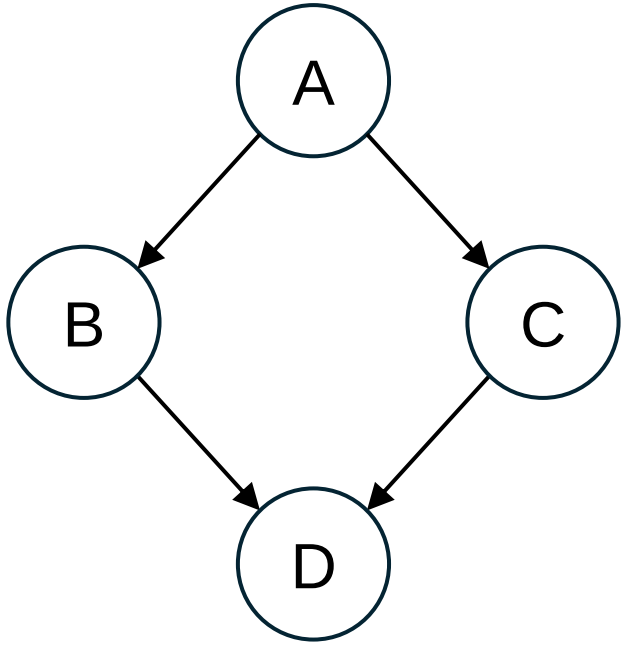
$$\mathrm{BIC}(\mathrm{G}, \boldsymbol{X}) = k \ln(n) - 2 \ln\Big(\prod_{i=1}^{m} p(\boldsymbol{x}_i|G)\Big) = k \ln(n) - 2 \sum_{i=1}^{m} \ln(p(\boldsymbol{x}_i|G))$$

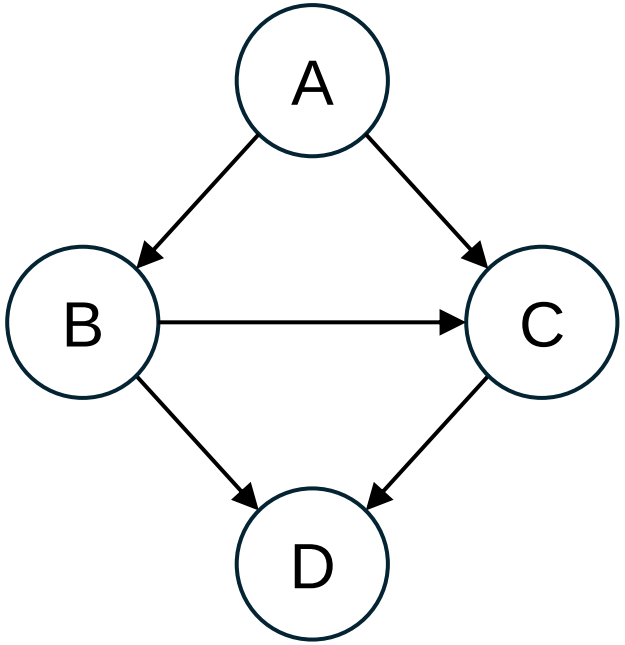
Greedy Equivalence Search (GES) [5]

[5] Chickering, David Maxwell. "Optimal structure identification with greedy search." *Journal of machine learning research* 3.Nov (2002): 507-554.

Covered Edge [X]

An edge $X_i \to X_j$ is **covered** in G if $\mathbf{Pa}(X_j) = \mathbf{Pa}(X_i) \cup X_i$.





Idea: A covered edge can be reversed without changing the MEC

Greedy Equivalence Search (GES)

Chickering Sequence

Let G, H be DAGs such that $I(H) \subseteq I(G)$, then a Chickering sequence from G to H is a sequence of DAGs $G_0, \dots, G_M$ such that $G_0 = G$, $G_M = H$ and each G_m in the sequence is obtained from G_{m-1} either by an edge addition or a covered edge reversal.

Intuition: H has a subset of G's independencies ($I(H) \subseteq I(G)$). Adding an edge can only "remove" an independence, so $I(H) \subseteq I(G)$ is still true. On the other hand, redirecting covered edges ensure that the independencies do not change, also keeping $I(H) \subseteq I(G)$ intact.

There provably exists such a sequence if $I(H) \subseteq I(G)$.

Greedy Equivalence Search (GES)

Assumptions: acyclicity, Markov, faithfulness, sufficiency

Input: Dataset $\boldsymbol{D}^{m \times n}$ (m: number of samples, n: number of variables)

Output: CPDAG (Completed Partially Directed Acyclic Graph)

Idea: follow a 2-phase procedure:

I. Forward phase: “Grow” the graph by adding edges that improve the score following a Chickering sequence

II. Backwards phase: Remove covered edges if that improves the score

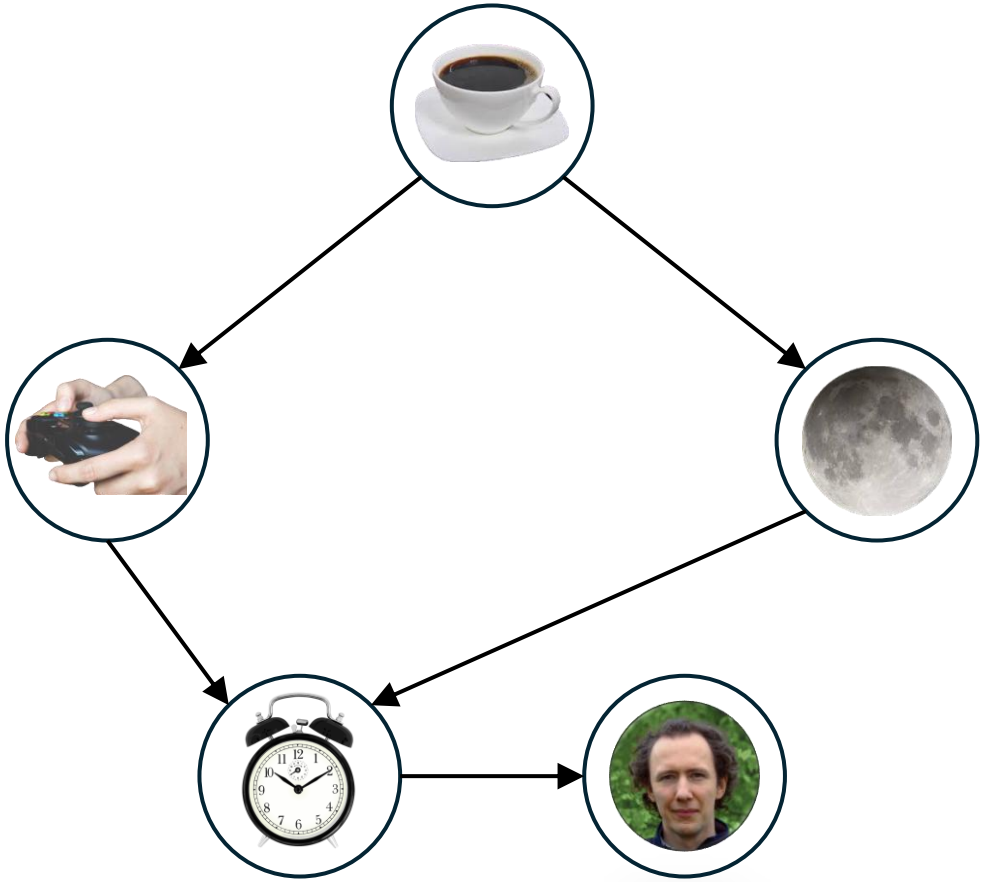
Greedy Equivalence Search (GES)

1. Start with graph without any edges (all independencies)
2. Forward phase: iterate over the following steps until no more edge is added
- Compute score for all edge additions
- Select edge with best score
- If that score improves the current score, add the edge, else stop
3. Backward phase: iterate over the following steps until no more edge is removed
- Compute score for all edge deletions
- Select edge with best score
- If that score improve the current score, delete the edge, else stop

Food for thought: Try to understand for yourself why this approach, where edges are added in single steps, still should discover all v-structures correctly.

GES – 1. Initialization

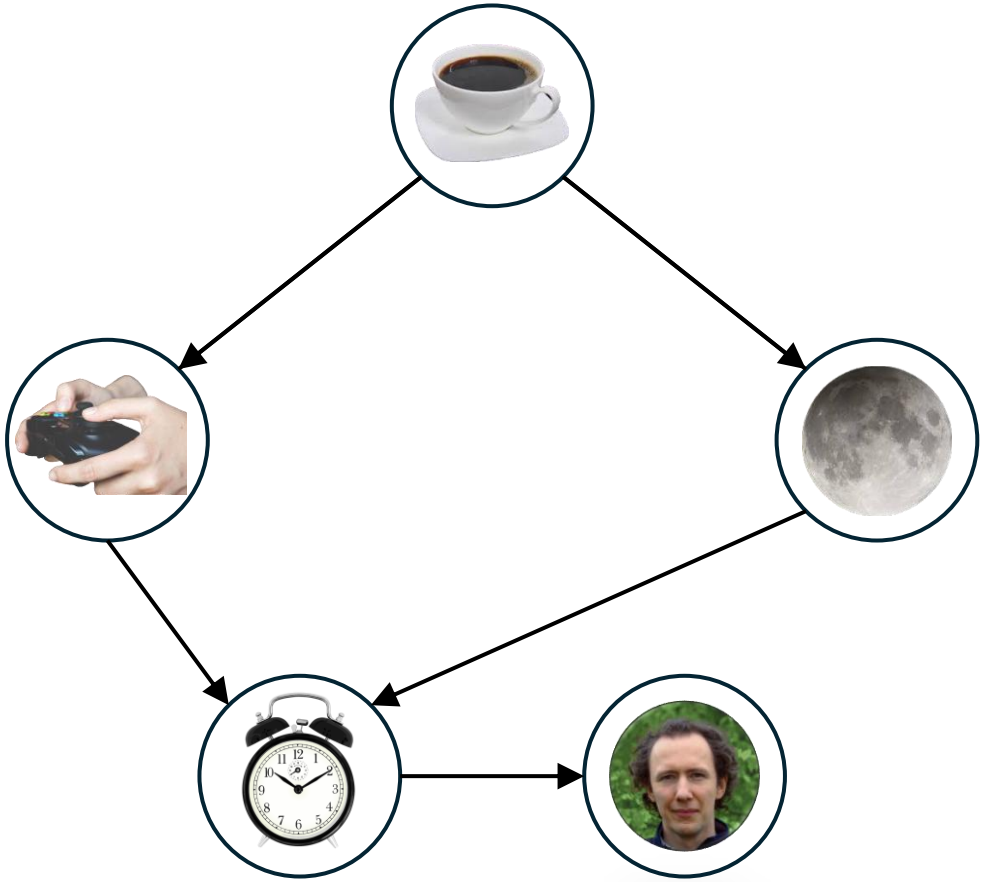
GES – 2. Forward Phase



Repeat:

- Add edge with highest score
- If edge is not covered, undirect it again
- (In each step, a possible direction is chosen to compute the BIC)
- If graph has not changed since last iteration, exit loop

GES – 3. Backward Phase



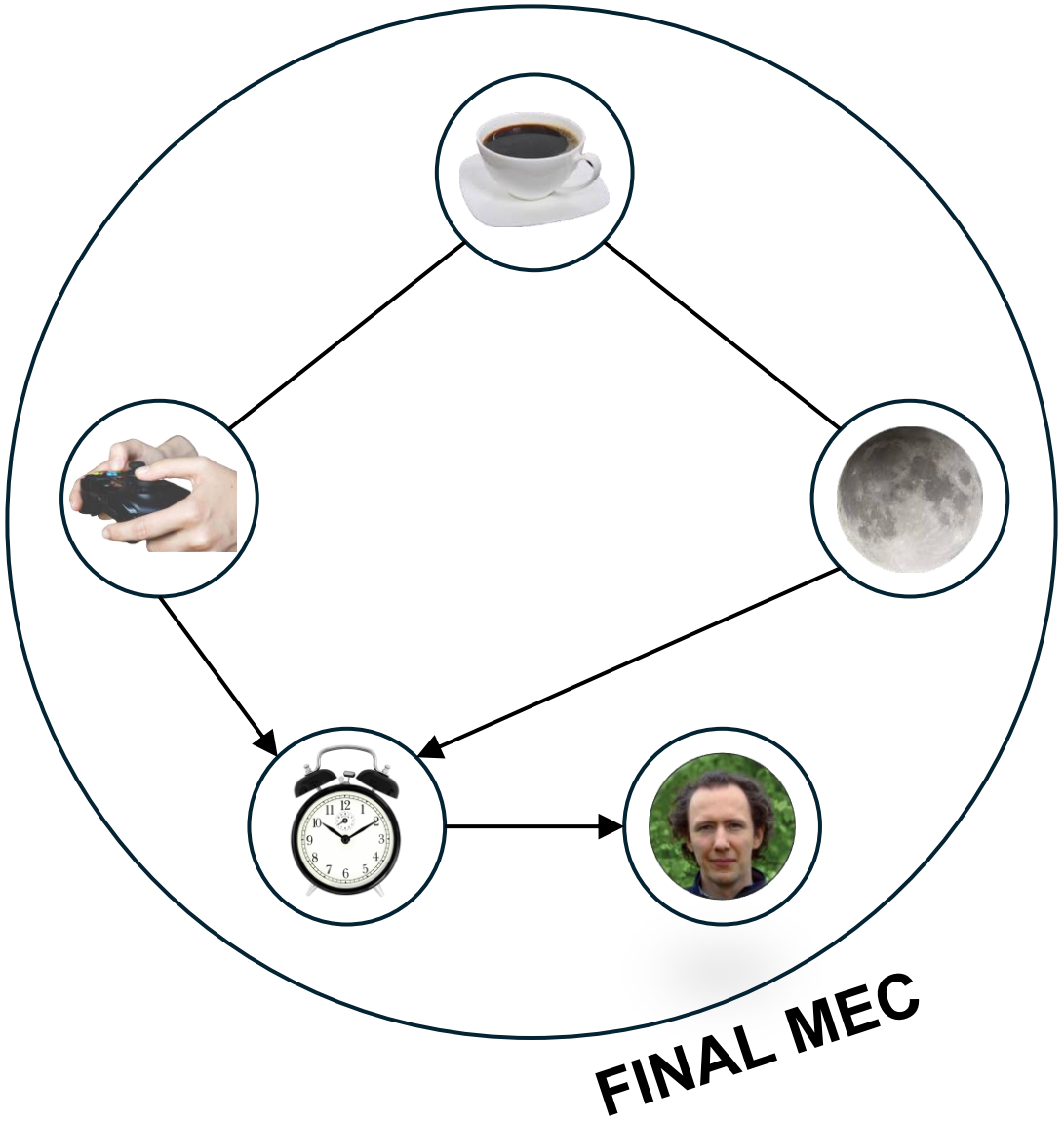
Repeat:

- Check if any edge removal would increase the score
- If true: remove edge
- If graph has not changed since last iteration, exit loop

No edge to remove in this example

Final step: Undirect all covered edges, as their direction might not be correct → MEC

GES – 3. Backward Phase



Repeat:

- Check if any edge removal would increase the score
- If true: remove edge
- If graph has not changed since last iteration, exit loop

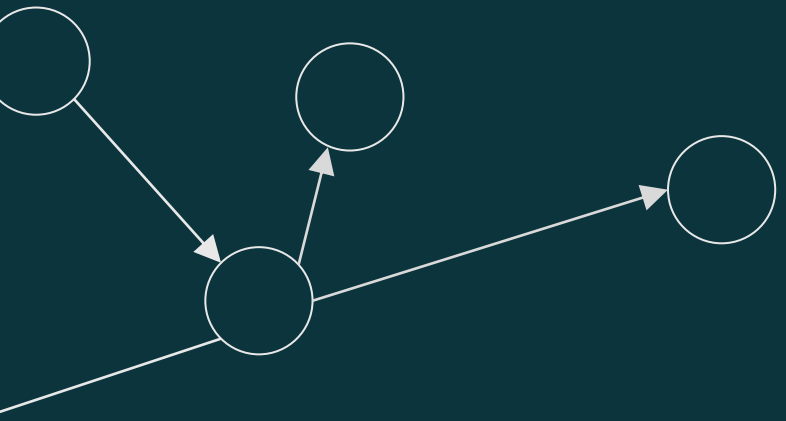
No edge to remove in this example

Final step: Undirect all covered edges, as their direction might not be correct → MEC

Summary

[6] Ramsey, Joseph, et al. "A million variables and more: the fast greedy equivalence search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images." International journal of data science and analytics 3.2 (2017): 121-129.
[7] Claassen, Tom, and Ioan G. Bucur. "Greedy equivalence search in the presence of latent confounders." Uncertainty in Artificial Intelligence. Pmlr, 2022.

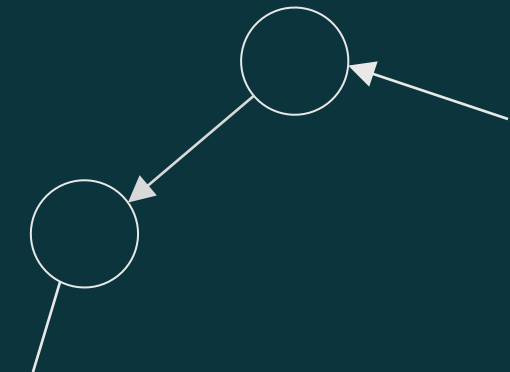
- Score-based methods optimize a score, independencies are considered only implicitly
- GES is a famous score-based method that, if all assumptions are satisfied, even discovers the true MEC
- Generally, greedy constrain-based methods might suffer from landing in local optima (e.g., Hill-Climb Search), but most state-of-the-art methods implement strategies to avoid this (at least in theory)
- Multiple variants of GES exists that, for example, improve speed (FGES [6]) or remove the sufficiency assumption (GPS [7])



Section

4

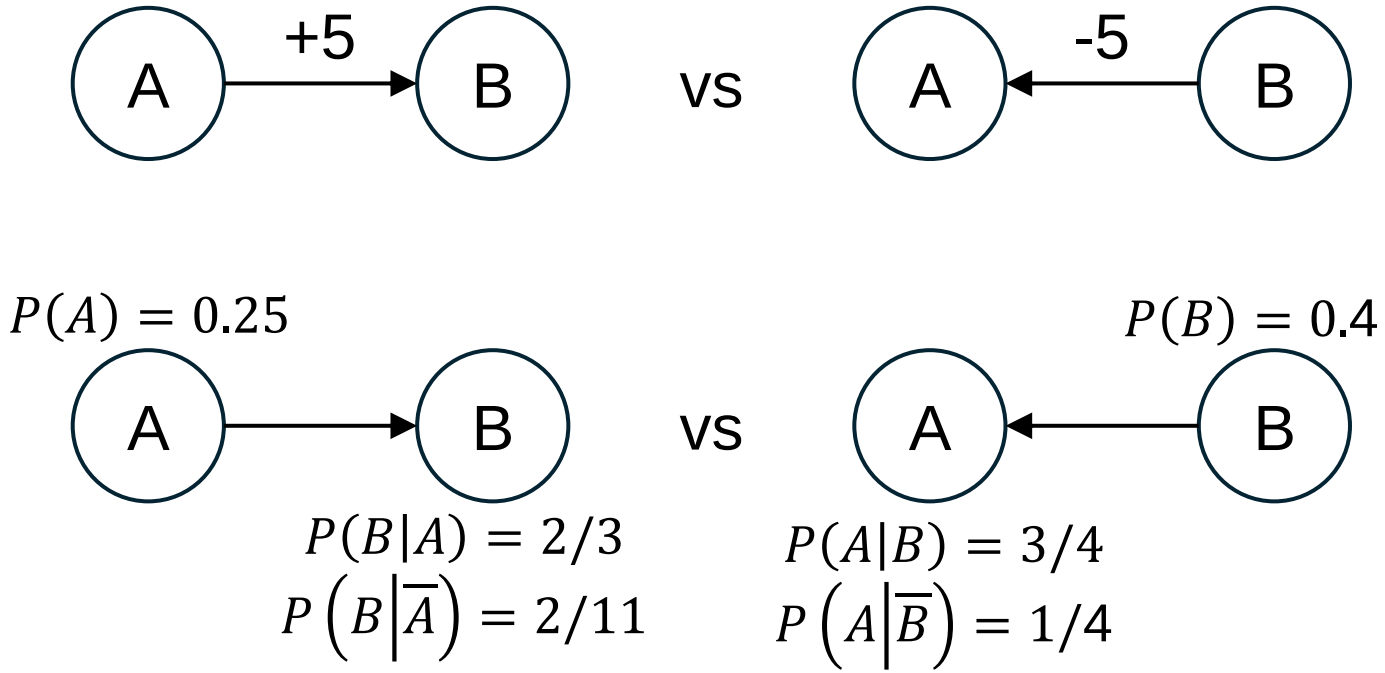
Additive Noise Models



Different Types of Assumptions

- Previous approaches can only learn the MEC
- We know that it is generally impossible to go beyond that
- Examples

	A	$\overline{A}$
B	0.3	0.1
$\overline{B}$	0.15	0.45

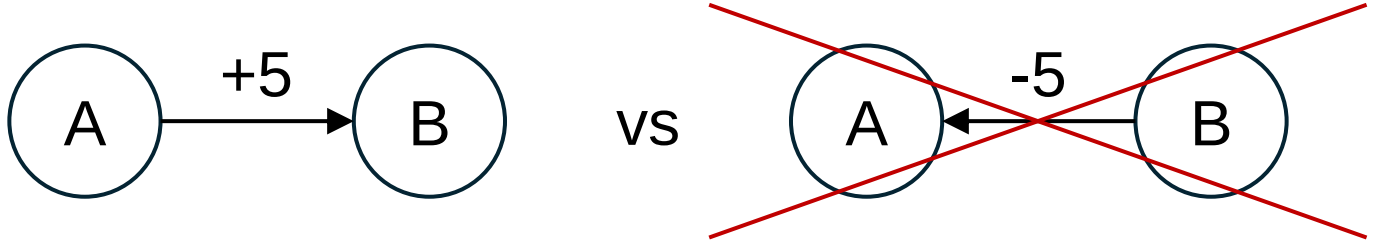


- From data alone, these are **indistinguishable**

Functional Models

- But what if we can include knowledge (assumptions) on the function space?

- For example: *All functions must add a positive number*



That works!!!

But it is generally a very restrictive assumption…*

*(but if you know it is true for your application, go for it, then it works perfectly!)

Additive Noise Models (ANMs)

[8] Peters, Jonas, Dominik Janzing, and Bernhard Schölkopf. Elements of causal inference: foundations and learning algorithms. The MIT press, 2017.

- But there is something much more “subtle” and less restrictive that is a given on almost all real-world datasets with continuous variables

- Assuming that noise is purely additive enables new CD approaches

Additive Noise Models (ANMs)

An ANM is an SCM where the structural assignments are of the form

$$X_j := f_j(\mathbf{PA}_j) + N_j, \qquad j = 1, \dots, d,$$

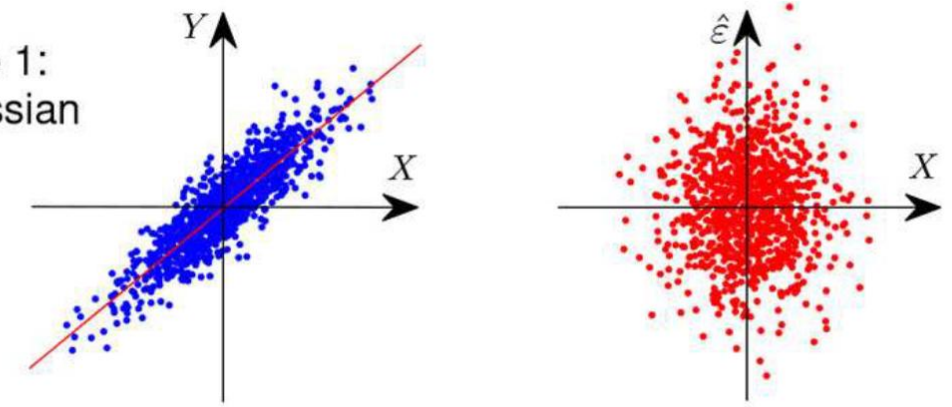
that is, if the noise is additive. [8]

Identifying Direction from Noise (Example)

Regression of Y given X: $Y = bX + \varepsilon$

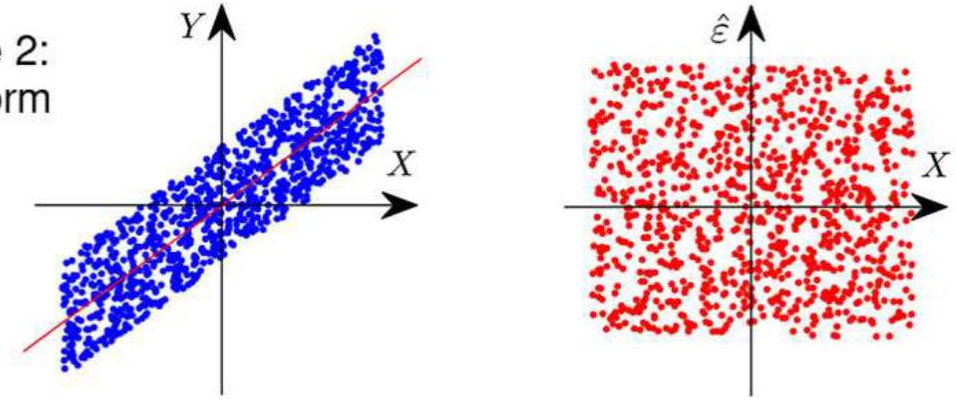
Regression of X given Y: $X = b_Y Y + \varepsilon_Y$

Case 1:
Gaussian



?

Case 2:
Uniform

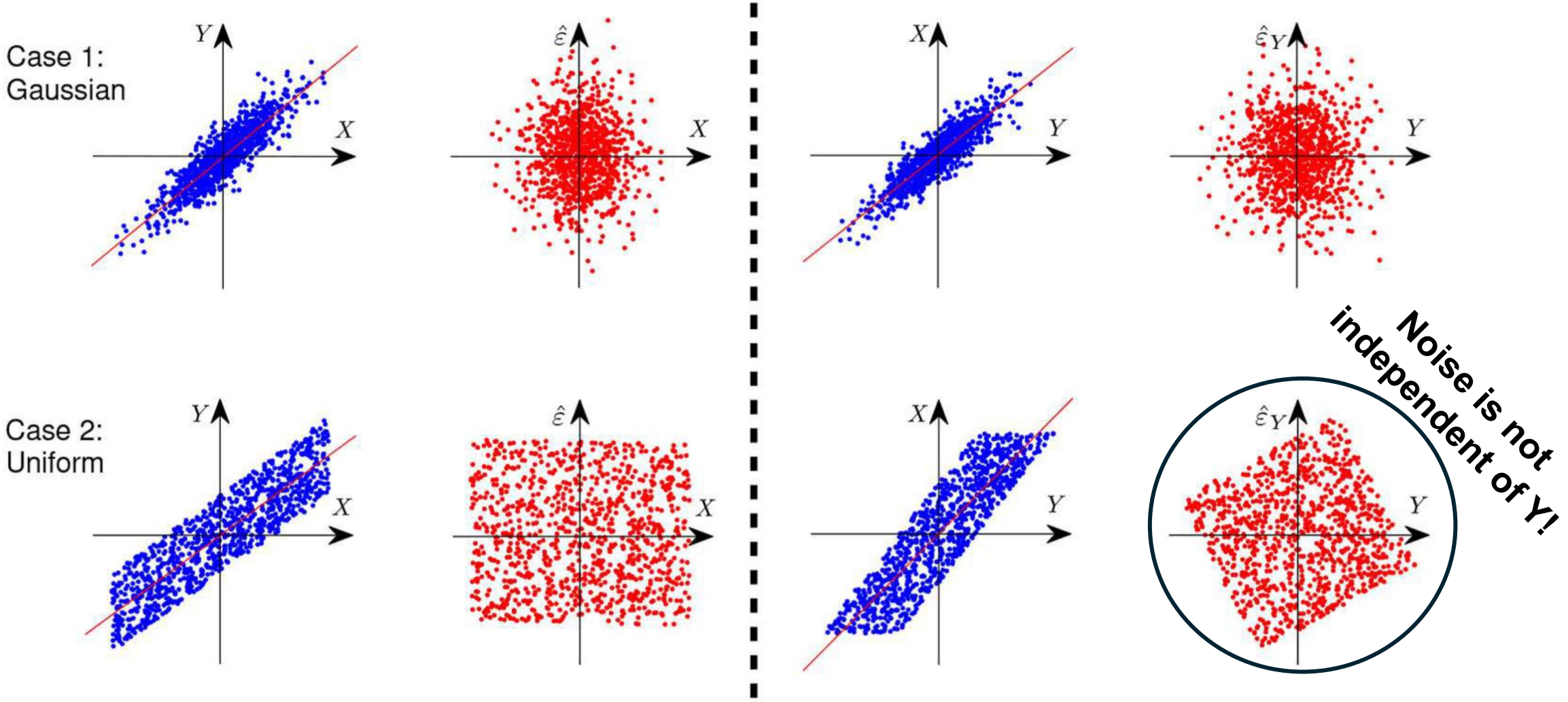


?

Identifying Direction from Noise (Example)

Regression of Y given X: $Y = bX + \varepsilon$

Regression of X given Y: $X = b_Y Y + \varepsilon_Y$



Basic Idea in This Scenario

Discovering the edge direction $X_i \to X_j$ vs $X_j \to X_i$

1. Fit function/model for both directions
2. Save residuals
3. Test independency between residual and covariate
4. Choose direction where independency holds

Noise must not be Gaussian (or generally, not on more than one node)

LiNGAM (Linear Non-Gaussian Acyclic Mode) [9]

[9] Shimizu, Shohei, et al. "A linear non-Gaussian acyclic model for causal discovery." Journal of Machine Learning Research 7.10 (2006).

Assumptions: Underlying model is *Linear*, with *Non-Gaussian* Noise and *Acyclic*(*)

Input: Dataset $\boldsymbol{D}^{m \times n}$ (m: number of samples, n: number of variables)

Output: DAG (Directed Acyclic Graph)

(*) Specifically, variable X_i must be determined as follows ($k(\cdot)$: causal order):

$$X_i = \sum_{k(j)<k(i)} b_{ij} x_j + e_i + c_i,$$

where the noise terms e_i must be non-Gaussian distributed with non-zero variances and independent of each other.

Idea: Disentangle noise using ICA (independent component analysis) by leveraging that non-Gaussian noise will look “increasingly Gaussian” the further down it propagates through the graph. (also see: Central Limit Theorem)

LiNGAM (Linear Non-Gaussian Acyclic Mode)

In matrix form: $\boldsymbol{X} = \boldsymbol{BX} + \boldsymbol{e}$ ➔ write as $\boldsymbol{X} = \boldsymbol{Ae}$, where $\boldsymbol{A} = (\boldsymbol{I} - \boldsymbol{B})^{-1}$

Why does this work? Example:

$$X_1 = N_1$$
$$X_2 = aX_1 + N_2$$
$$X_3 = bX_1 + cX_2 + N_3$$

$$\boldsymbol{X} = \begin{pmatrix} 0 & 0 & 0 \\ a & 0 & 0 \\ b & c & 0 \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} + \begin{pmatrix} N_1 \\ N_2 \\ N_3 \end{pmatrix}$$

Substitute variables

$$\begin{aligned} X_3 &= bX_1 + cX_2 + N_3 \\ &= bX_1 + c(aX_1 + N_2) + N_3 \\ &= b(N_1) + c(a(N_1) + N_2) + N_3 \\ &= (b + ca)N_1 + cN_2 + N_3 \end{aligned}$$

$$\boldsymbol{X} = \begin{pmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ (b + ac) & c & 1 \end{pmatrix} \begin{pmatrix} N_1 \\ N_2 \\ N_3 \end{pmatrix} = \boldsymbol{A} \begin{pmatrix} N_1 \\ N_2 \\ N_3 \end{pmatrix}$$

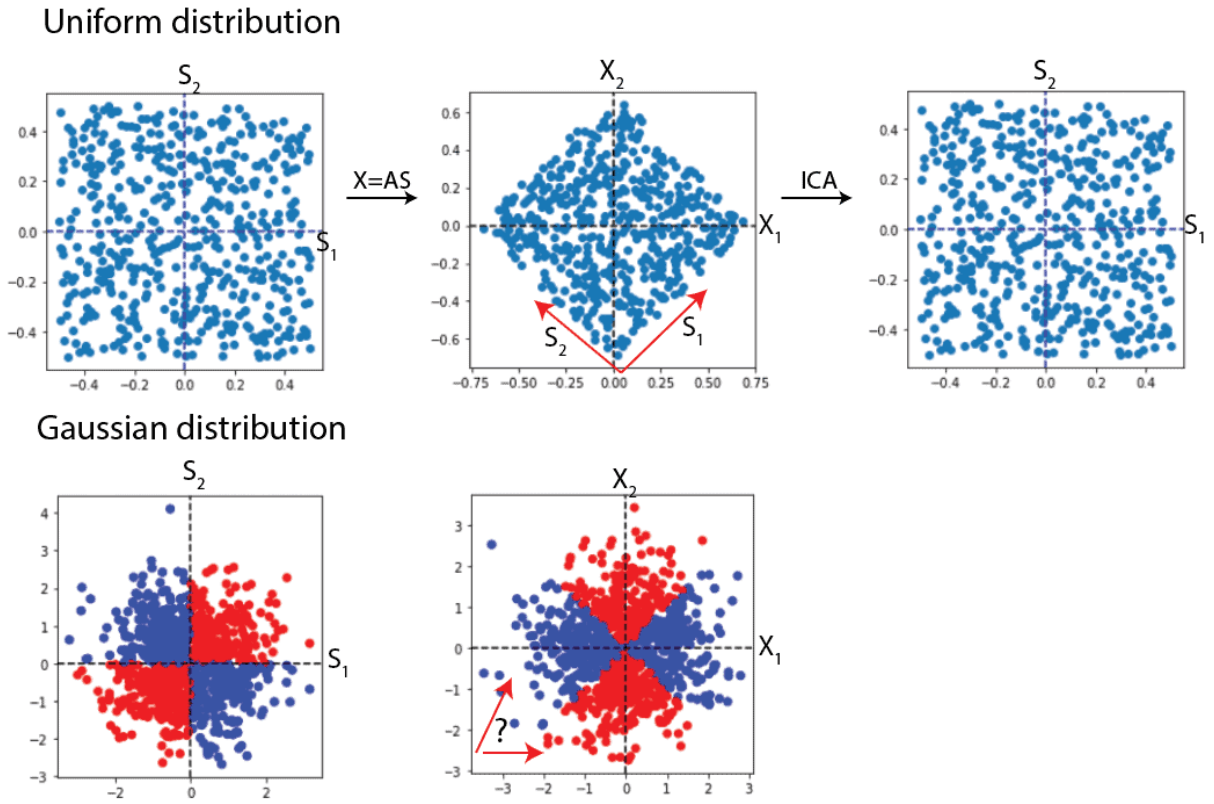
$\boldsymbol{X}$ is written only w.r.t. the noise

LiNGAM (Linear Non-Gaussian Acyclic Mode)

We have: $\boldsymbol{X = Ae}$

→ICA-problem (independent component analysis)

“unmix” a linear mixture of independent components



https://www.baeldung.com/cs/independent-component-analysis

Central Limit Theorem: mixtures of random variables become “more gaussian” than the components

➔Minimize Gaussianity of Marginals (see why can’t have gaussian noise?)

LiNGAM (Linear Non-Gaussian Acyclic Mode)

With $\boldsymbol{A}$, let us compute $\boldsymbol{W} = \boldsymbol{A}^{-1}$ and we want to get $\boldsymbol{B}$

There are two more steps:

1. ICA has “permutation ambiguity”: rows are not in the correct order, i.e., the correspondence between noise and variables doesn’t match

   → Find a row-wise permutation of $\boldsymbol{W}$ such that no zeros lie on the diagonal

   → Scale to normalize diagonal to 1 and then compute $\boldsymbol{B} = \boldsymbol{I} - \boldsymbol{W}$

2. But the causal graph might not be respected yet (acyclicity)

   → Find a permutation that is strictly lower triangular (i.e., acyclic)

In practice, both steps often do not give the exact desired results (non-zero diagonal; lower triangular matrix) due to noise, and an optimization problem is solved to find the optimal solution

RESIT (REgression with Subsequent Independence Test) [10]

Assumptions: acyclicity, non-linear functions, additive + independent noise, sufficiency

Input: Dataset $\boldsymbol{D}^{m \times n}$ (m: number of samples, n: number of variables)

Output: DAG (Directed Acyclic Graph)

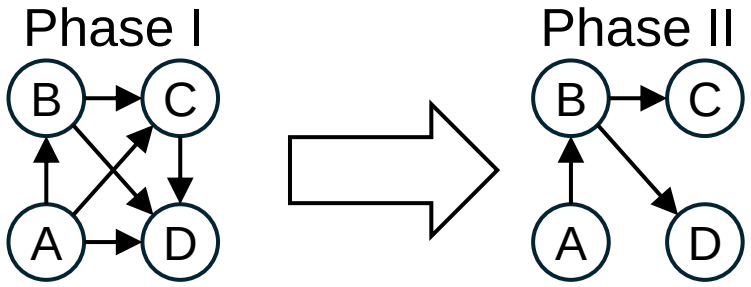
Noise can be Gaussian, but functions must not be linear

Two Phases:

I. Determine topological (causal) order (each node is a cause of every successor)

II. Remove unnecessary edges to get the final DAG

Example:



[10] Peters, Jonas, et al. "Causal discovery with continuous additive noise models." The Journal of Machine Learning Research 15.1 (2014): 2009-2053.

RESIT (REgression with Subsequent Independence Test)

I. Determine topological order

1. Regress each variable on all variables
2. Choose the variable with minimal dependence between residuals (noise) and predictors (other variables) as last variable in causal order (sink node)
3. Repeat steps with all but that variable or stop if all variable are accounted for

II. Remove unnecessary edges

1. Iterate over each variable
2. Iterate over all parents, regressing the variable on all but the current parent
3. If residuals are independent from that parent, remove edge from that parent

Section 5

Other Approaches

NOTEARS [11], DAGMA [12]

[11] Zheng, Xun, et al. "Dags with no tears: Continuous optimization for structure learning." Advances in neural information processing systems 31 (2018).
[12] Bello, Kevin, Bryon Aragam, and Pradeep Ravikumar. "Dagma: Learning dags via m-matrices and a log-determinant acyclicity characterization." Advances in Neural Information Processing Systems 35 (2022): 8226-8239.

There is another maths-heavy approach? Can you handle that?

No overreactions or emotional outcries?

$\Rightarrow$ NOTEARS

NOTEARS: Convert combinatorial optimization problem into continuous program, allowing for gradient descent

$$\min_{G} \; score(G) \quad \text{s.t.} \quad G \in \text{DAGs} \iff \min_{W \in \mathbb{R}^{d \times d}} \; score(W) \quad \text{s.t.} \quad h(W) = 0$$

DAGMA: newer, more optimized version; underlying idea is the same

Different Sources of Knowledge Apart From Data

Expert knowledge

- Specify known edges, forbidden edges, information about causal order,…
- Different algorithms can incorporate different types of extra knowledge

Powerful and automatable source of extra (meta) knowledge: **LLMs**

- Potentially very useful, depending on the dataset and the variables
- Different strategies exist (and are developed currently)

Purely LLM-based causal discovery

- Querying the LLM for pairs of variables [13] or the entire graph [14]
- More sophisticated strategies, e.g., breadth-first [15]

[13] Zečević, Matej, et al. "Causal Parrots: Large Language Models May Talk Causality But Are Not Causal." Transactions on Machine Learning Research.
[14] Kiciman, Emre, et al. "Causal reasoning and large language models: Opening a new frontier for causality." Transactions on Machine Learning Research (2023).
[15] Jiralerspong, Thomas, et al. "Efficient causal graph discovery using large language models." arXiv preprint arXiv:2402.01207 (2024).

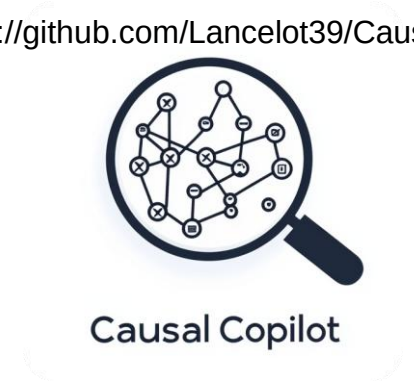
Different Sources of Knowledge Apart From Data

LLM + causal discovery hybrid approaches

- Predictor combining LLM prediction and causal discovery results [16]
- PC algorithm on LLM independency results instead of data [17]
- Assigning types [18] or tags [19] and directing undirected edges based on common information

Using an LLM to guide the entire causal discovery process [20]

https://github.com/Lancelot39/Causal-Copilot



[16] Clivio, Oscar, et al. "Learning to Defer for Causal Discovery with Imperfect Experts." CoRR (2025).
[17] Cohrs, Kai-Hendrik, et al. "Large Language Models for Constrained-Based Causal Discovery." CoRR (2024).
[18] Brouillard, Philippe, et al. "Typing assumptions improve identification in causal discovery." Conference on Causal Learning and Reasoning. PMLR, 2022.
[19] Busch, Florian Peter, et al. "Tagged for Direction: Pinning Down Causal Edge Directions with Precision." arXiv preprint arXiv:2506.19459 (2025).
[20] Wang, Xinyue, et al. "Causal-copilot: An autonomous causal analysis agent." arXiv preprint arXiv:2504.13263 (2025).

Causal Discovery for Time Series Data

Remember lecture 3 on time series?

Different approaches for causal discovery on time series exist, often as extensions of previous approaches to the time series setting:

- Extension of PC: PCMCI [21]
- Extension of Lingam: VarLiNGAM [22]
- Extension of NOTEARS: DYNOTEARS [23]

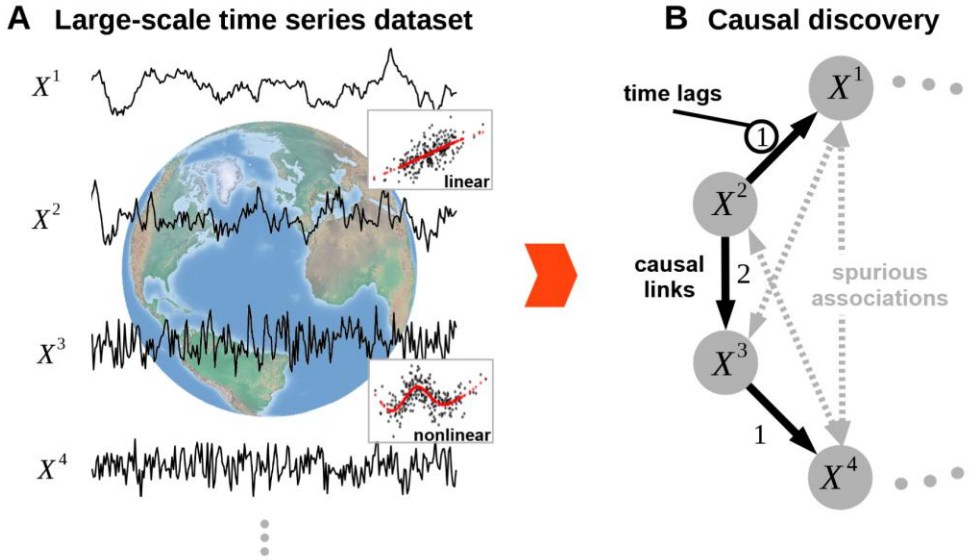


Figure taken from Runge et al. [21]

[21] Runge, Jakob, et al. "Detecting and quantifying causal associations in large nonlinear time series datasets." Science advances 5.11 (2019): eaau4996.
[22] Hyvärinen, Aapo, et al. "Estimation of a structural vector autoregression model using non-Gaussianity." Journal of Machine Learning Research 11.5 (2010).
[23] Pamfil, Roxana, et al. "Dynotears: Structure learning from time-series data." International Conference on Artificial Intelligence and Statistics. Pmlr, 2020.

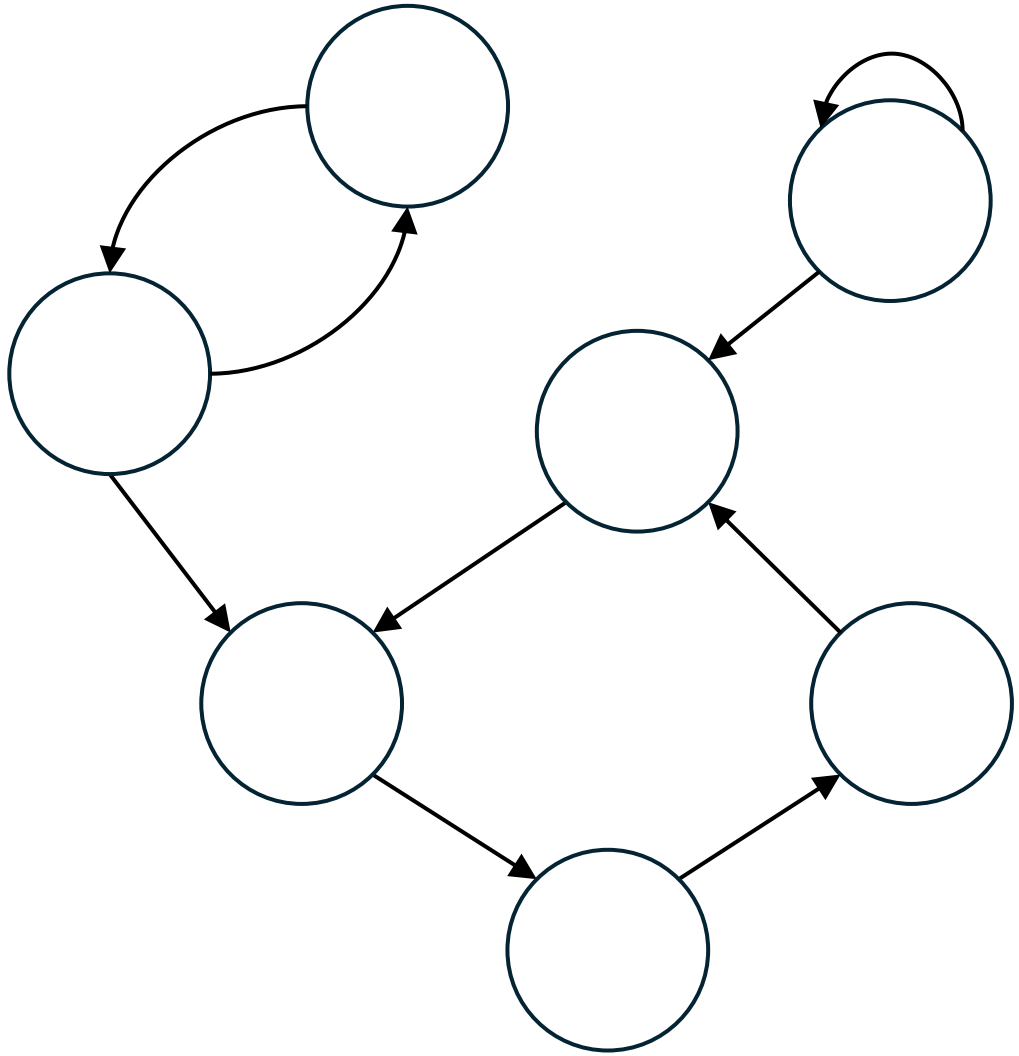
Dropping the Acyclicity Assumption

Much harder problem in general

Two general directions:

1. Times series setting → can apply causal discovery methods that take time into account

2. Assume data has reached an equilibrium → can resolve ambiguities but is a strong assumption



Theory and Practice

[24] Reisach, Alexander, Christof Seiler, and Sebastian Weichwald. "Beware of the simulated dag! causal discovery benchmarks may be easy to game." Advances in Neural Information Processing Systems 34 (2021): 27772-27784.

- Data quality is very important; an algorithm can only learn what is represented by the data
- For example, independency methods might fail

- One might make prediction for the wrong reasons, even (or maybe specifically) in simulated settings [24]… **more on this next lecture!**

Exercise

CD with known causal order

Imagine you are in a real-world setting, where you have a dataset and want to apply causal discovery. In this specific case, assume you know when each variable has been measured and that there is a clear temporal **order** according to which all variables have been recorded (e.g., A has always been recorded before B). Also assume causal sufficiency to hold.

Now, argue why (or why not) an independency based causal discovery method should always be able to return a specific DAG instead of only a Markov equivalence class here.

Section 6

Summary and Outlook

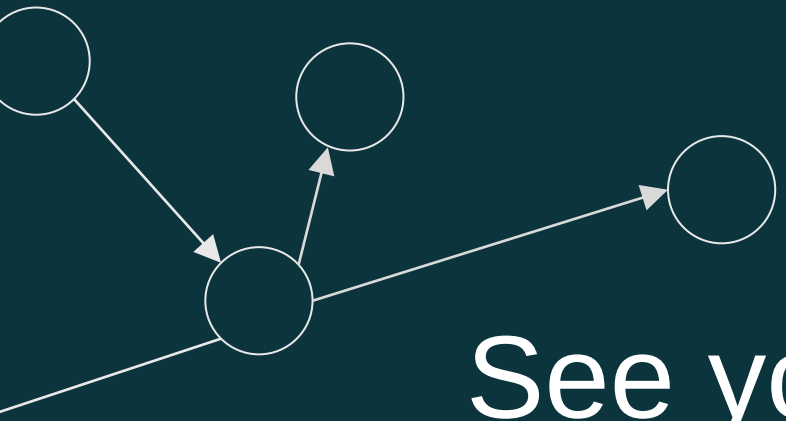
Summary

- Causal Discovery: Identifying the causal graph for a set of variables

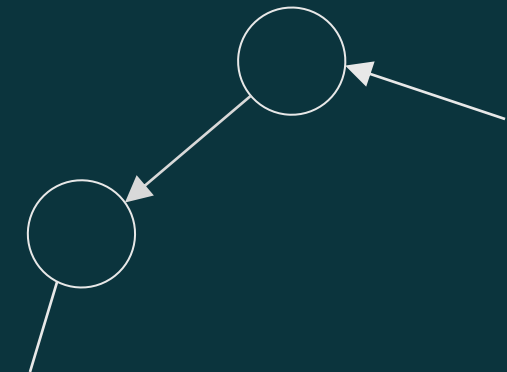
- Three major categories of approaches
  - Constraint-Based Causal Discovery (independency tests)
  - Score-Based Causal Discovery (score optimization)
  - Functional Models (assumption on functions, e.g., Additive Noise Models)

- Other sources of knowledge can also be leveraged (e.g., experts, LLMs)

- Many specialized Causal Discovery methods for various scenarios exist



See you next week!



Bibliography

[1] Zanga, Alessio, Elif Ozkirimli, and Fabio Stella. "A survey on causal discovery: theory and practice." *International Journal of Approximate Reasoning* 151 (2022): 101-129.

[2] Spirtes, Peter, Clark N. Glymour, and Richard Scheines. *Causation, prediction, and search*. MIT press, 2000.

[3] Meek, Christopher. "Causal inference and causal explanation with background knowledge." *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*. 1995.

[4] Spirtes, Peter, Christopher Meek, and Thomas Richardson. "Causal inference in the presence of latent variables and selection bias." *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*. 1995.

[5] Chickering, David Maxwell. "Optimal structure identification with greedy search." Journal of machine learning research 3.Nov (2002): 507-554.

[6] Ramsey, Joseph, et al. "A million variables and more: the fast greedy equivalence search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images." International journal of data science and analytics 3.2 (2017): 121-129.

[7] Claassen, Tom, and Ioan G. Bucur. "Greedy equivalence search in the presence of latent confounders." Uncertainty in Artificial Intelligence. Pmlr, 2022.

[8] Peters, Jonas, Dominik Janzing, and Bernhard Schölkopf. Elements of causal inference: foundations and learning algorithms. The MIT press, 2017.

Bibliography

[9] Shimizu, Shohei, et al. "A linear non-Gaussian acyclic model for causal discovery." Journal of Machine Learning Research 7.10 (2006).

[10] Peters, Jonas, et al. "Causal discovery with continuous additive noise models." The Journal of Machine Learning Research 15.1 (2014): 2009-2053.

[11] Zheng, Xun, et al. "Dags with no tears: Continuous optimization for structure learning." Advances in neural information processing systems 31 (2018).

[12] Bello, Kevin, Bryon Aragam, and Pradeep Ravikumar. "Dagma: Learning dags via m-matrices and a log-determinant acyclicity characterization." Advances in Neural Information Processing Systems 35 (2022): 8226-8239.

[13] Zečević, Matej, et al. "Causal Parrots: Large Language Models May Talk Causality But Are Not Causal." Transactions on Machine Learning Research.

[14] Kiciman, Emre, et al. "Causal reasoning and large language models: Opening a new frontier for causality." Transactions on Machine Learning Research (2023).

[15] Jiralerspong, Thomas, et al. "Efficient causal graph discovery using large language models." arXiv preprint arXiv:2402.01207 (2024).

[16] Clivio, Oscar, et al. "Learning to Defer for Causal Discovery with Imperfect Experts." CoRR (2025).

Bibliography

[17] Cohrs, Kai-Hendrik, et al. "Large Language Models for Constrained-Based Causal Discovery." CoRR (2024).

[18] Brouillard, Philippe, et al. "Typing assumptions improve identification in causal discovery." Conference on Causal Learning and Reasoning. PMLR, 2022.

[19] Busch, Florian Peter, et al. "Tagged for Direction: Pinning Down Causal Edge Directions with Precision." arXiv preprint arXiv:2506.19459 (2025).

[20] Wang, Xinyue, et al. "Causal-copilot: An autonomous causal analysis agent." arXiv preprint arXiv:2504.13263 (2025).

[21] Runge, Jakob, et al. "Detecting and quantifying causal associations in large nonlinear time series datasets." Science advances 5.11 (2019): eaau4996.

[22] Hyvärinen, Aapo, et al. "Estimation of a structural vector autoregression model using non-Gaussianity." Journal of Machine Learning Research 11.5 (2010).

[23] Pamfil, Roxana, et al. "Dynotears: Structure learning from time-series data." International Conference on Artificial Intelligence and Statistics. Pmlr, 2020.

[24] Reisach, Alexander, Christof Seiler, and Sebastian Weichwald. "Beware of the simulated dag! causal discovery benchmarks may be easy to game." Advances in Neural Information Processing Systems 34 (2021): 27772-27784.