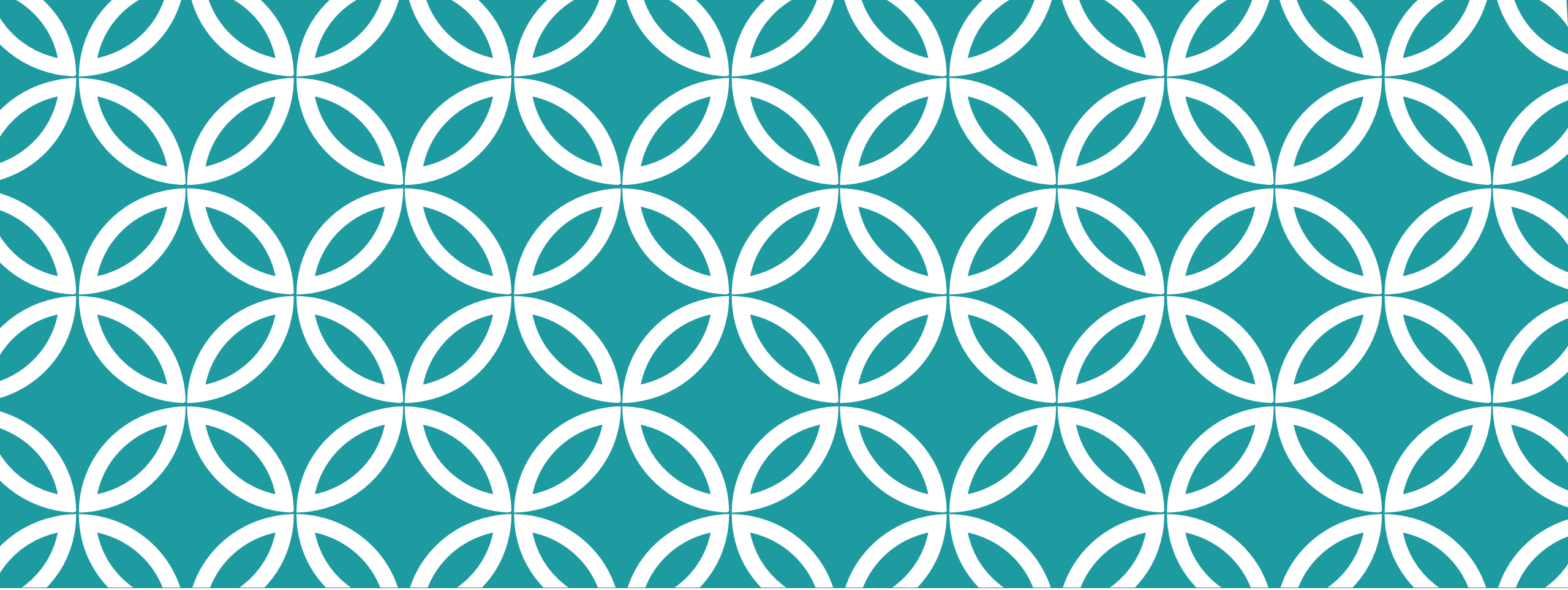


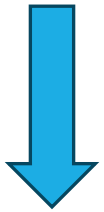
CAUSAL DISCOVERY

- Taxonomy of causal discovery methods
- Peter-Clark
- Fast Causal Inference

SCOPE

Conditional independence facts can be easily recovered from DAGs: can we recover a DAG from data?

Is conditional independence enough to establish a causal DAG? Under which assumptions?

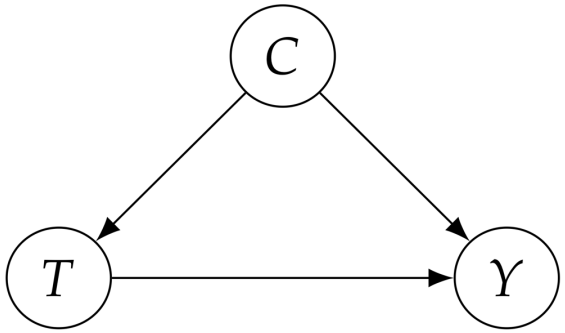


Causal Discovery's primary aim is to recover a causal DAG (or its equivalence class) from data!

TWO MAIN ASSUMPTIONS

Causal sufficiency:

A set of observed variables V is **Causally Sufficient** if every common cause of any two variables in V is also in V. In other words, there are no unobserved confounders (latent variables) that influence more than one variable in the set.



Example:

$V = \{T, Y, C\}$ is causal sufficient
$V = \{T, Y\}$ is not causal sufficient

TWO MAIN ASSUMPTIONS

Causal sufficiency:

A set of observed variables V is **Causally Sufficient** if every common cause of any two variables in V is also in V. In other words, there are no unobserved confounders (latent variables) that influence more than one variable in the set.

Intuition

A set of variables is **Causally Sufficient** if the **Markov Blanket** of every variable is 'fully observable'.

- If you can block all influence on a variable X using only its observed neighbors, you have sufficiency.
- If X continues to 'talk' to Y despite conditioning on all potential observed parents/children, there is a ghost in the machine **(a latent confounder)** and your set is not causally sufficient.

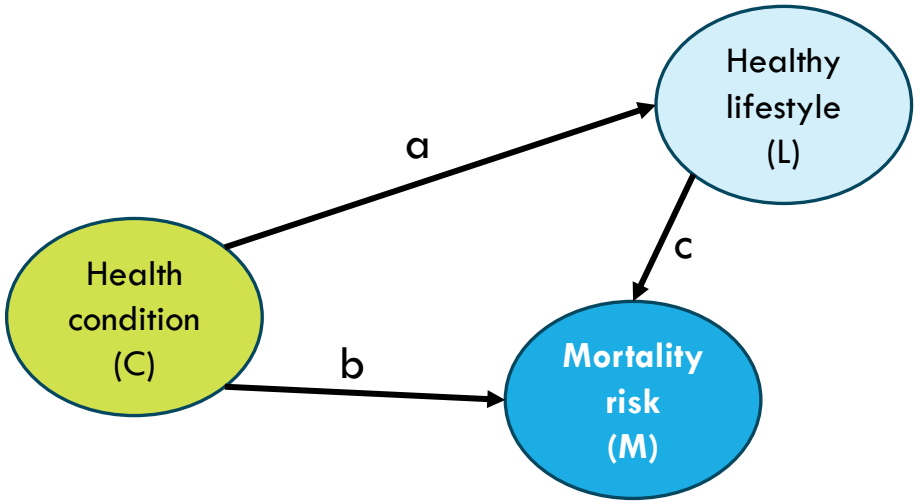
TWO MAIN ASSUMPTIONS

Faithfulness:

Given a graph over a set of observed variables V, and a joint distribution $\mathbb{P}(V)$, the **Faithfulness Condition** is satisfied iff the only conditional independencies in $\mathbb{P}$ are the ones entailed by d-separation.

conditional independence $\Longrightarrow$ d-separation

Example of faithfulness violation:



Suppose that they are Gaussian variable linearly related as:

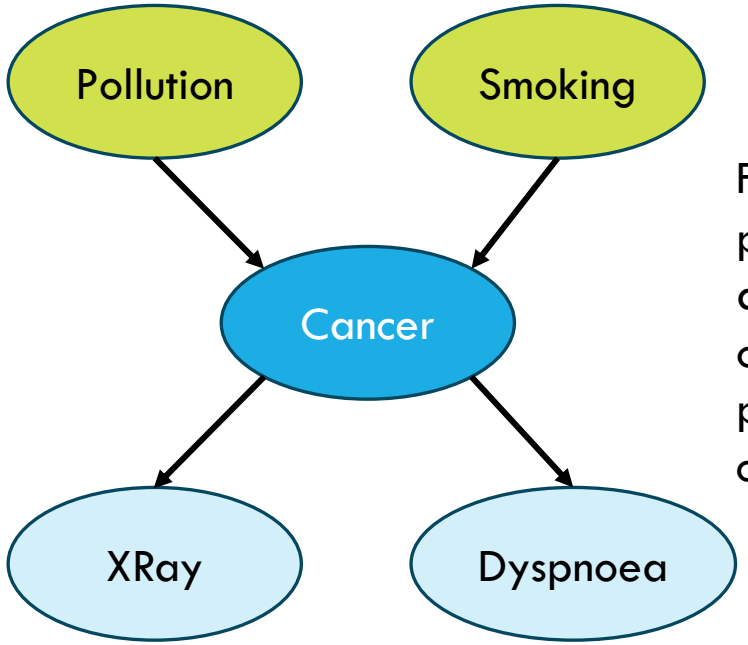
$$\begin{cases} L = a \times C \\ M = b \times C + c \times L \end{cases}$$

$\Longrightarrow M = (b+ac)\, C$

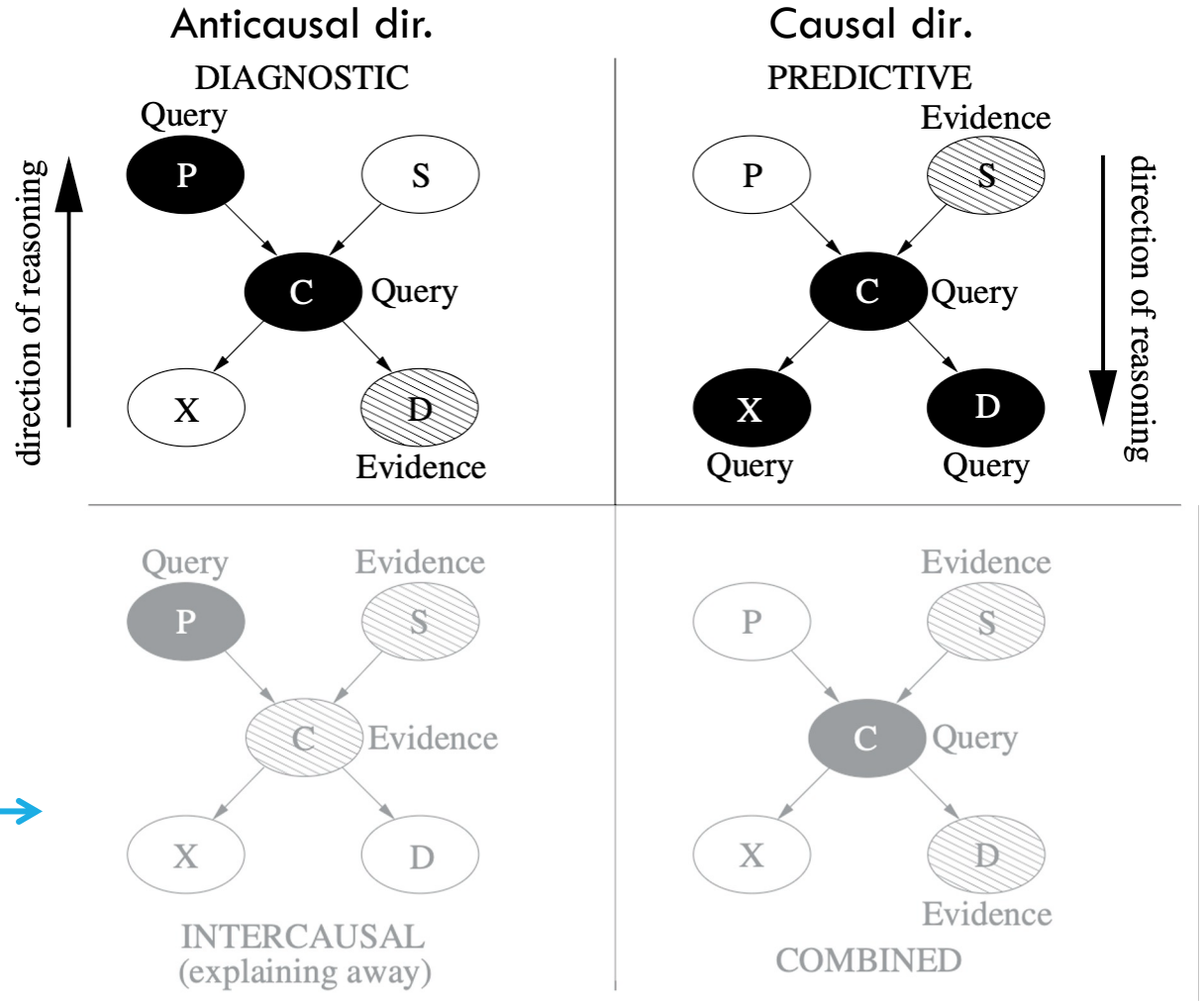
$\Longrightarrow$ if $b+ac=0$, the mortality risk is independent of the health condition!

CAUSAL DATA STRUCTURING AND REASONING

Let us suppose the following (simplified) Lung-cancer problem[1], for which a causal DAG has been established.



Full representation of the probability distribution allows to explain the type of reasoning we can perform through conditioning.



1. Kevin B. Korb, Ann E. Nicholson. Bayesian AI.

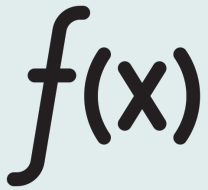
CAUSAL DISCOVERY METHODS – A TAXONOMY

Constraint

- Uses statistical independence tests to prune edges from a fully connected graph.
- Examples: PC, FCI,…

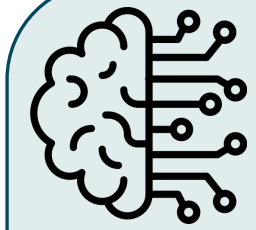
Score-based

- Searches the space of possible DAGs and optimizes a goodness-of-fit score (like BIC).
- Examples: GES, Hill climbing,…



Functional-based

- Assumes a specific functional model to describe the causal flow. Exploits noise properties to establish/orient edges.
- Examples: LiNGAM, ANM,…



Continuous

- DL/ML approach framing discovery as a continuous optimization problem (NNs + gradient descent).
- Examples: NOTEARS, Dag-GNN,…

CAUSAL DISCOVERY METHODS – A TAXONOMY

Constraint

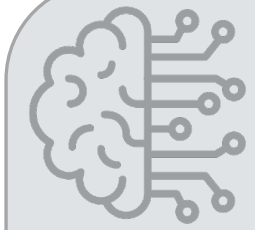
- Uses statistical independence tests to prune edges from a fully connected graph.
- Examples: **PC, FCI,…**

Score-based

- Searches the space of possible DAGs and optimizes a goodness-of-fit score (like BIC).
- Examples: GES, Hill climbing,…

Functional-based

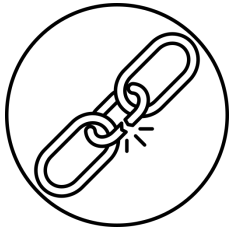
- Assumes a specific functional model to describe the causal flow. Exploits noise properties to establish/orient edges.
- Examples: LiNGAM, ANM,…



Continuous

- DL/ML approach framing discovery as a continuous optimization problem (NNs + gradient descent).
- Examples: NOTEARS, Dag-GNN,…

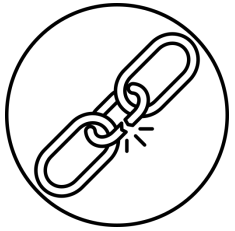
CAUSAL DISCOVERY METHODS – CONSTRAINT-BASED



We focus on Constraint-based methods:

- ✓ **Interpretability:** Every edge removed is backed by a statistical test.
- ✓ **Reliability:** Well-understood theoretical limits in small samples/provably identifiable (under certain assumptions).
- ✓ **Latent Handling:** FCI is one of the few methods that detects unobserved confounders.

CAUSAL DISCOVERY METHODS – CONSTRAINT-BASED



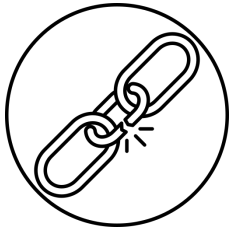
Constraint-based methods rely on (conditional) independence tests to recover the DAG:

- Markov condition
- Faithfulness condition

} To smoothly go from independence to d-separation to DAG

Causal sufficiency is also required most of the time, to ensure completeness → all possible conditional independence facts can be actually tested!

CAUSAL DISCOVERY METHODS – CONSTRAINT-BASED



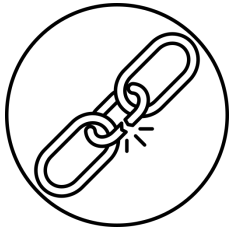
Intuition on conditional independence testing – the linear Gaussian case

If X_2, X_3 are jointly normally distributed, then: $X_2 \perp\!\!\!\perp X_3 \Leftrightarrow \mathrm{Corr}(X_2, X_3) = 0$

Zero correlation can be tested e.g. with the Fisher's z test:

1. Calculate sample correlation (statistic): $r_{X_1,X_2} = \frac{\widehat{\mathrm{Cov}}(X_1, X_2)}{\hat{\sigma}_{X_1}\hat{\sigma}_{X_2}} = \frac{\sum_i (X_{1_i} - \bar{X}_1)(X_{2_i} - \bar{X}_2)}{\sqrt{\sum_i (X_{1_i} - \bar{X}_1)^2}\sqrt{\sum_i (X_{2_i} - \bar{X}_2)^2}}$
2. Under H0 (zero correlation): $z := \frac{1}{2}\ln\left(\frac{1 + r_{X_1,X_2}}{1 - r_{X_1,X_2}}\right) \sim \mathcal{N}\left(0, \frac{1}{N-3}\right)$ # samples
3. Given the statistic and its null distribution, we can find the p-value.

CAUSAL DISCOVERY METHODS – CONSTRAINT-BASED

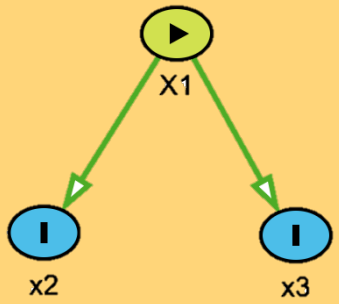


Intuition on conditional independence testing – the linear Gaussian case

Conditional independence ↔ partial correlation: can be tested by using the residuals!

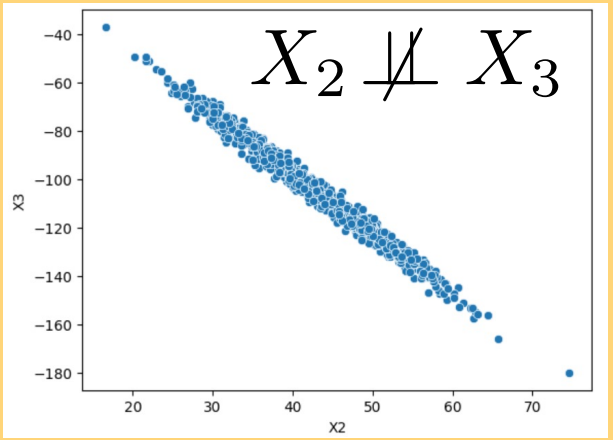
1. Regress X_2 on X_1 and X_3 on X_1: $\widehat{X}_2(X_1);\ \widehat{X}_3(X_1)$
2. Calculate the residuals (→ noise): $\mathrm{res}_{X_2} = X_2 - \widehat{X}_2(X_1);\ \mathrm{res}_{X_3} = X_3 - \widehat{X}_3(X_1)$
3. If $X_2 \perp\!\!\!\perp X_3 | X_1$ then the residuals are zero-correlated, which can be tested as before with Fisher-z!

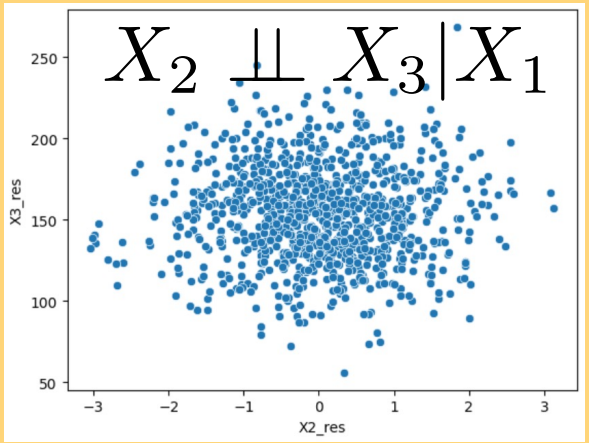
$X_1 = \mathcal{N}(50, 10^2)$



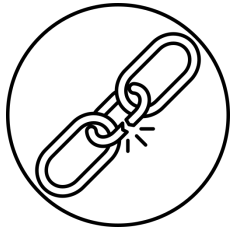
$X_2 = 2 + 0.8X_1 + \mathcal{N}(0,1)$

$X_3 = -3 - 2X_1 + \mathcal{N}(0,1)$





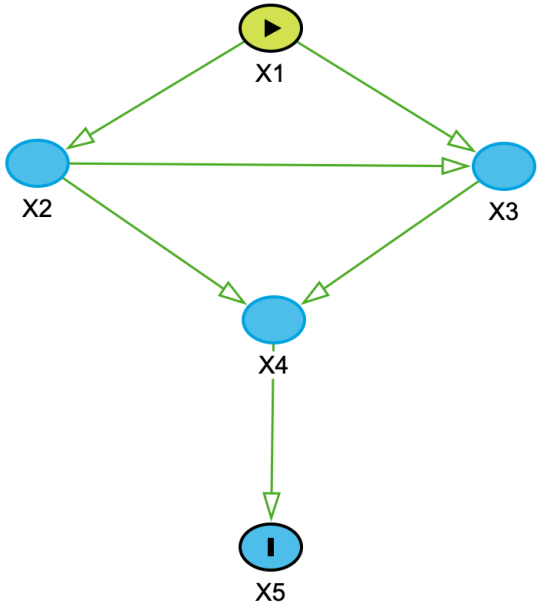
CAUSAL DISCOVERY METHODS – CONSTRAINT-BASED



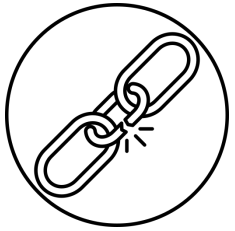
Are we always able to correctly recover the true DAG by testing for conditional independencies?

X1	X2	X3	X4	X5
2.099343	2.586254	-0.661093	8.039057	6.757995
1.972347	2.520444	-2.360024	10.356054	7.349395
2.129538	3.306731	-1.705014	8.162636	5.965670
2.304606	1.239023	-4.266158	7.896162	5.646977
1.953169	1.251667	-3.309183	9.865793	5.454375
1.953173	2.414299	-2.072563	7.364751	4.962531
2.315843	2.145090	-3.541617	8.782157	5.930440
2.153487	3.390991	-1.359504	6.208884	6.161564
1.906105	2.045028	-2.959345	7.353365	6.020301
2.108512	1.641952	-2.892420	9.086643	4.780281

$$X_1 \perp\!\!\!\perp X_4 | X_2, X_3$$
$$X_1 \perp\!\!\!\perp X_5 | X_4$$
$$X_1 \perp\!\!\!\perp X_5 | X_2, X_3$$
$$X_2 \perp\!\!\!\perp X_5 | X_4$$
$$X_3 \perp\!\!\!\perp X_5 | X_4$$



CAUSAL DISCOVERY METHODS – CONSTRAINT-BASED



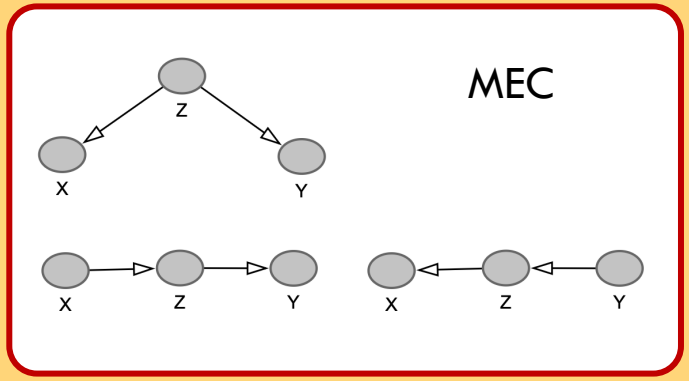
Markov equivalence class (MEC):

A set of DAGs that have the same skeleton and the same v-structures.

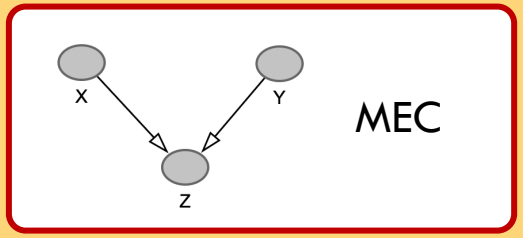
Intuition

Constraint-based methods face **Observational Equivalence**: multiple DAGs can satisfy the same set of independence statements.

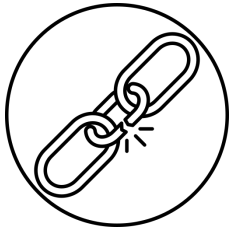
We have already seen that chains and forks share the same independence statements.



This is not the case for v-structures!



CAUSAL DISCOVERY METHODS – CONSTRAINT-BASED



Markov equivalence class (MEC):

A set of DAGs that have the same skeleton and the same v-structures.

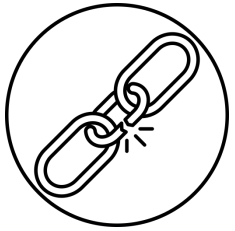
A MEC can be represented through a **CPDAG** (Complete Partially Directed Graph):

→ **Directed Edges:** These are invariant across the entire MEC. We are certain about this causal direction (often found via v-structures). → in every member of the MEC.

— **Undirected Edges:** These represent reversible relations. The data is consistent with arrows in either direction. → in some members of the MEC; ← in some others.

No edge: not adjacent in any member of the MEC.

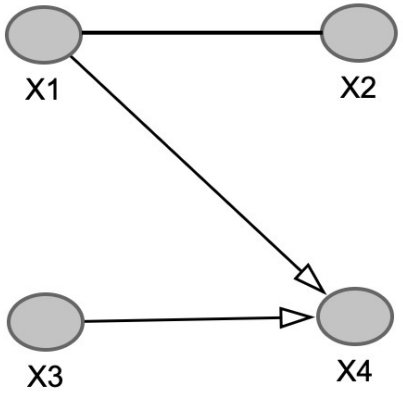
CAUSAL DISCOVERY METHODS – CONSTRAINT-BASED

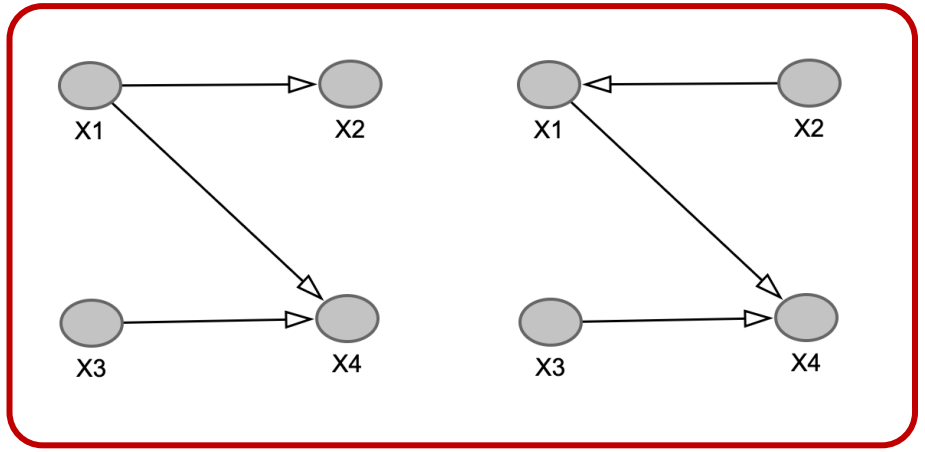


Markov equivalence class (MEC):

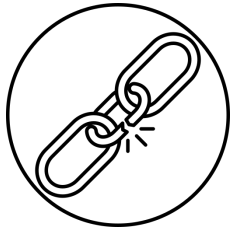
A set of DAGs that have the same skeleton and the same v-structures.

Example: how many DAGs in this MEC?





THE PC ALGORITHM



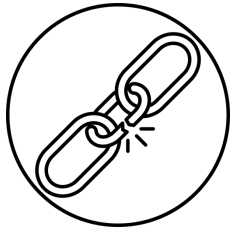
The PC algorithm, named after Peter Sprites and Clark Glymour[1], is a classical and still widely used constraint-based causal discovery method. Its output is a CPDAG. It assumes: Markov condition, Faithfulness and causal sufficiency.

Step 1 – Skeleton discovery

Start – complete undirected graph
Test for (conditional) independence with an increased cardinality of the conditioning set to remove edges.

1. P.Spirtes, C.N.Glymour. Causation, prediction, and search. MIT press, 2000.

THE PC ALGORITHM

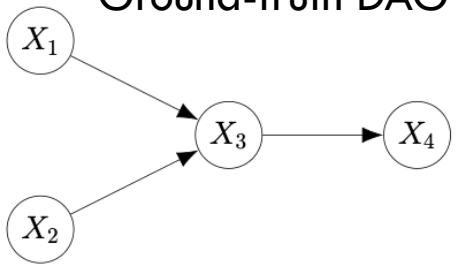


The PC algorithm, named after Peter Sprites and Clark Glymour[1], is a classical and still widely used constraint-based causal discovery method. Its output is a CPDAG. It assumes: Markov condition, Faithfulness and causal sufficiency.

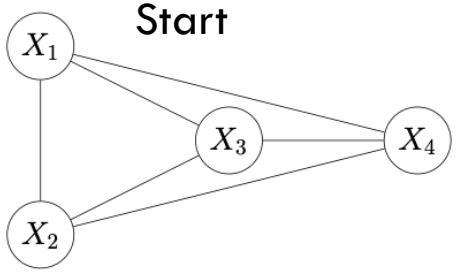
Step 1 – Skeleton discovery

Example

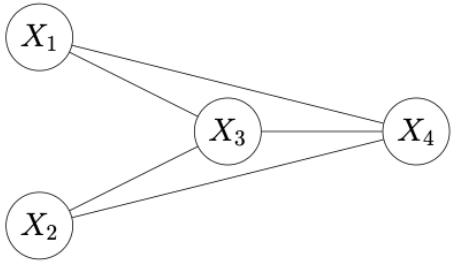
Ground-truth DAG



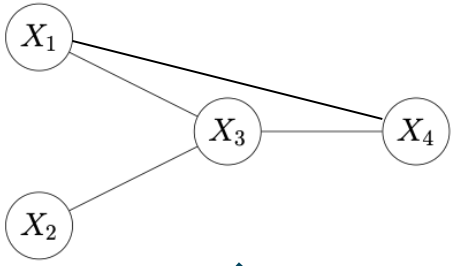
Start



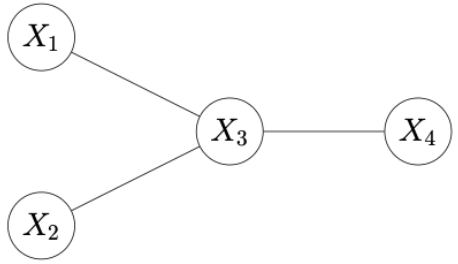
$X_1 \perp\!\!\!\perp X_2$



$X_2 \perp\!\!\!\perp X_4 | X_3$

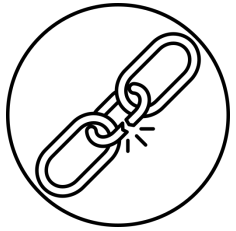


$X_1 \perp\!\!\!\perp X_4 | X_3$



1. P.Spirtes, C.N.Glymour. Causation, prediction, and search. MIT press, 2000.

THE PC ALGORITHM



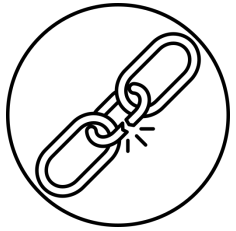
The PC algorithm, named after Peter Sprites and Clark Glymour[1], is a classical and still widely used constraint-based causal discovery method. Its output is a CPDAG. It assumes: Markov condition, Faithfulness and causal sufficiency.

Step 2 – v-structure discovery

Orient as a v-structure each triple of nodes (X, Z, Y) such that X, Z and Y, Z are adjacent in the discovered skeleton, but X, Y are independent (without conditioning).

1. P.Spirtes, C.N.Glymour. Causation, prediction, and search. MIT press, 2000.

THE PC ALGORITHM

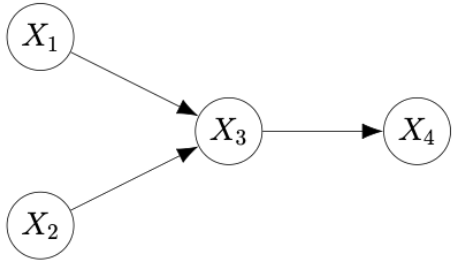


The PC algorithm, named after Peter Sprites and Clark Glymour[1], is a classical and still widely used constraint-based causal discovery method. Its output is a CPDAG. It assumes: Markov condition, Faithfulness and causal sufficiency.

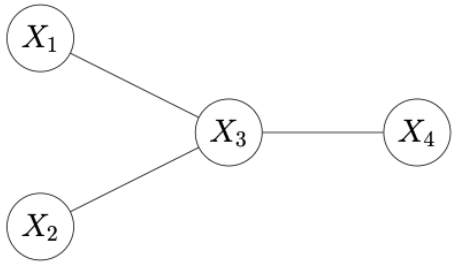
Step 2 – v-structure discovery

Example

Ground-truth DAG

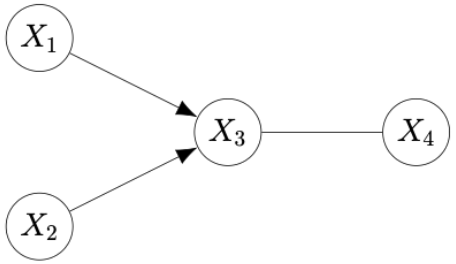


Skeleton after step 1



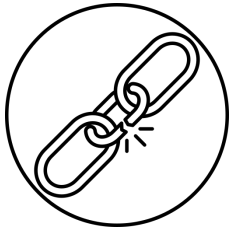
$X_1 \perp\!\!\!\perp X_2$

A v-structure



1. P.Spirtes, C.N.Glymour. Causation, prediction, and search. MIT press, 2000.

THE PC ALGORITHM



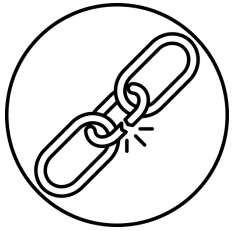
The PC algorithm, named after Peter Sprites and Clark Glymour[1], is a classical and still widely used constraint-based causal discovery method. Its output is a CPDAG. It assumes: Markov condition, Faithfulness and causal sufficiency.

Step 3 – orientation propagation

$X \to Z - Y$, and X, Y are not adjacent $\Longrightarrow X \to Z \to Y$

1. P.Spirtes, C.N.Glymour. Causation, prediction, and search. MIT press, 2000.

THE PC ALGORITHM

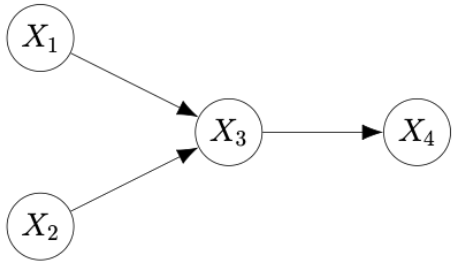


The PC algorithm, named after Peter Sprites and Clark Glymour[1], is a classical and still widely used constraint-based causal discovery method. Its output is a CPDAG. It assumes: Markov condition, Faithfulness and causal sufficiency.

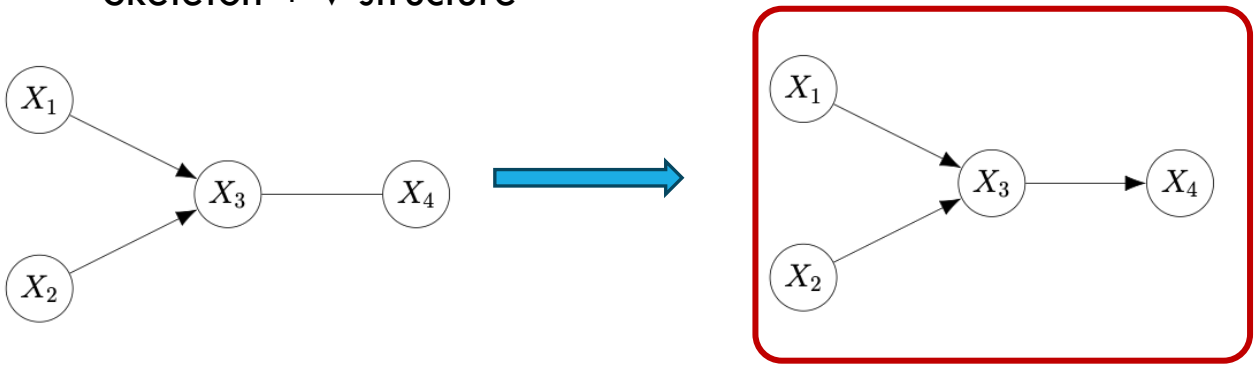
Step 3 – orientation propagation

Example

Ground-truth DAG

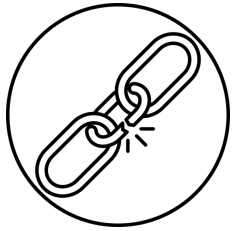


Skeleton + v-structure



1. P.Spirtes, C.N.Glymour. Causation, prediction, and search. MIT press, 2000.

THE FCI ALGORITHM



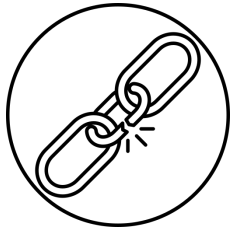
The FCI[1] algorithm (Fast Causal Inference) extends the PC algorithm allowing for the presence of latent variables, i.e. variables which are not observed but play a role in the generating process of the observed variables. It assumes: Markov condition, Faithfulness, BUT NOT causal sufficiency.

Note: Suppose we are observing samples from the observable variables in the set $\boldsymbol{O}$, which have been actually generated by the interplay of variables in $\boldsymbol{V} = \{\boldsymbol{O}, \boldsymbol{L}\}$, $\boldsymbol{L}$ being latent/unobserved. We could marginalize out all latent variables. BUT, DAGs are not closed under marginalization so in the presence of latent variables, the causal process over measured $\boldsymbol{O}$ variables is not necessarily a DAG!

How to visualize the conditional independencies among the observed variables $\boldsymbol{O}$ only for DAGs with *observed* and *latent variables*? How can we represent (independence) equivalence classes over $\boldsymbol{O}$?

1. P.Spirtes, C.N.Glymour. Causation, prediction, and search. MIT press, 2000.

THE FCI ALGORITHM



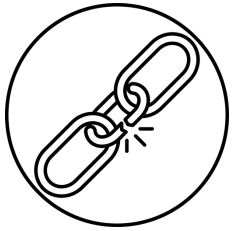
The FCI[1] algorithm (Fast Causal Inference) extends the PC algorithm allowing for the presence of latent variables, i.e. variables which are not observed but play a role in the generating process of the observed variables. It assumes: Markov condition, Faithfulness, BUT NOT causal sufficiency.

Partial Ancestral Graph (PAG):

Edge	Meaning
Directed $X \to Y$	X is an *ancestor* of Y, and there *may* further be unobserved confounding between the two
Bidirected $X \longleftrightarrow Y$	*unobs confounding* between X and Y, but *no* causal ancestral relationship in either direction
Possibly bidirected edge $X \circ\!\!\to Y$	either $X \to Y$ or $X \longleftrightarrow Y$
Undetermined edge $X \circ\!\!-\!\!\circ Y$	no info about the relationship between X and Y. Either $X \to Y$, $X \leftarrow Y$ or $X \leftrightarrow Y$.

1. P.Spirtes, C.N.Glymour. Causation, prediction, and search. MIT press, 2000.

THE FCI ALGORITHM



The FCI[1] algorithm (Fast Causal Inference) extends the PC algorithm allowing for the presence of latent variables, i.e. variables which are not observed but play a role in the generating process of the observed variables. It assumes: Markov condition, Faithfulness, BUT NOT causal sufficiency.

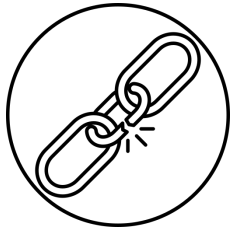
Note: PAGs are the equivalence classes of MAGs (Maximal Ancestral Graphs)

A MAG

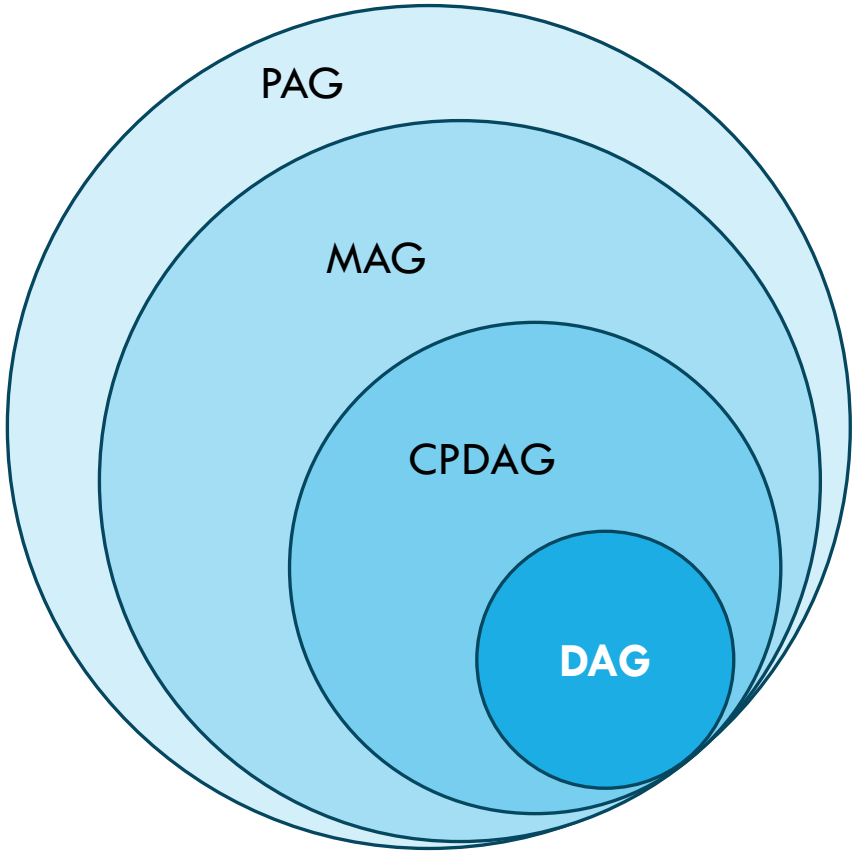
- may contain two kinds of edges:
  - directed edges ($X \rightarrow Y$)
  - bi-directed edges ($X \leftrightarrow Y$), which means *no directed* path from X to Y
- has no directed cycles
- is maximal: no edge can be added without changing the independence model

1. P.Spirtes, C.N.Glymour. Causation, prediction, and search. MIT press, 2000.

THE FCI ALGORITHM

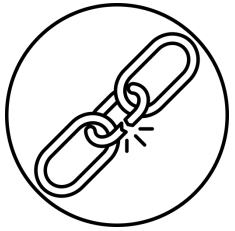


The FCI[1] algorithm (Fast Causal Inference) extends the PC algorithm allowing for the presence of latent variables, i.e. variables which are not observed but play a role in the generating process of the observed variables. It assumes: Markov condition, Faithfulness, BUT NOT causal sufficiency.



1. P.Spirtes, C.N.Glymour. Causation, prediction, and search. MIT press, 2000.

THE FCI ALGORITHM



The FCI[1] algorithm (Fast Causal Inference) extends the PC algorithm allowing for the presence of latent variables, i.e. variables which are not observed but play a role in the generating process of the observed variables. It assumes: Markov condition, Faithfulness, BUT NOT causal sufficiency.

FCI works similarly to PC (same steps), but:

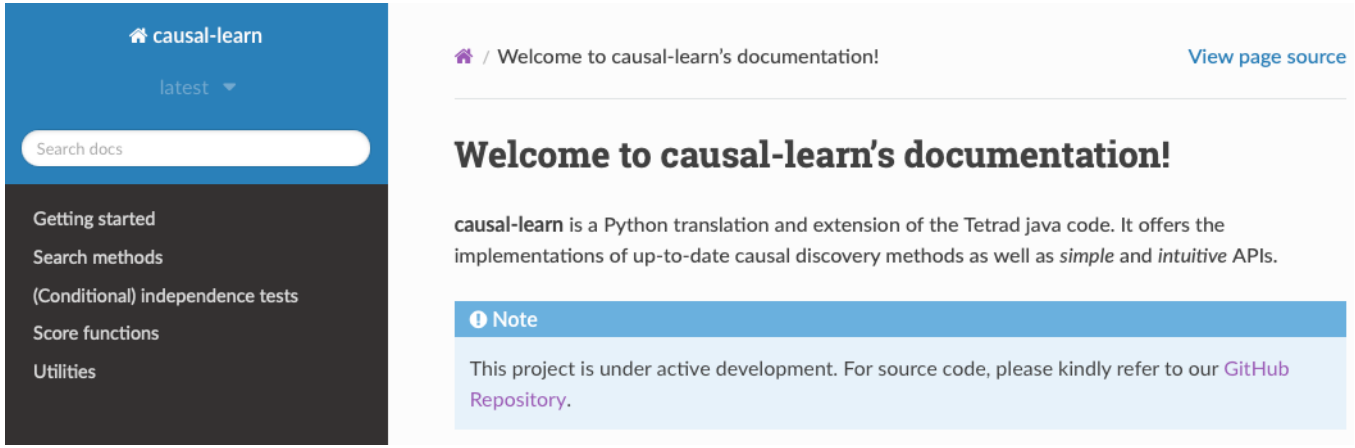
1. Instead of reasoning in terms of independence, it reasons in terms of paths and d-separation (ancestors/descendants rather than parents/children).

2. The rules for edge orientation are more theoretically and technically elaborate, but the principle remains similar.

1. P.Spirtes, C.N.Glymour. Causation, prediction, and search. MIT press, 2000.

IMPLEMENTATION AND PACKAGES

- Several implementations of PC, FCI, and other causal discovery methods are available, both in R and Python
  → we will use the Python package causal-learn (intuitive and easy to use).



- Another widely used tool is Tetrad, which can also provide a graphical user interface (GUI).