Causal Inference:
Principled Search Algorithms

The Fundamental Problem

- "Correlation is not causation!"
  - If X and Y are associated, then it could be:
    - $X \rightarrow Y$
    - $X \leftarrow Y$
    - $X \leftarrow L \rightarrow Y$ [where L is unobserved]
    - Or any combination of these

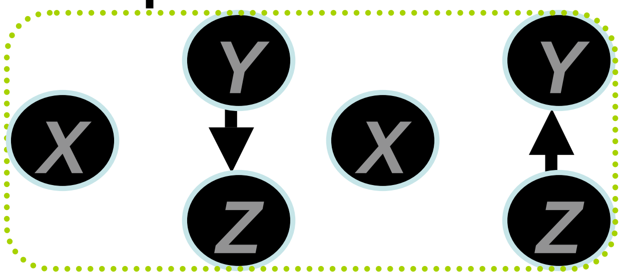
The Fundamental Problem

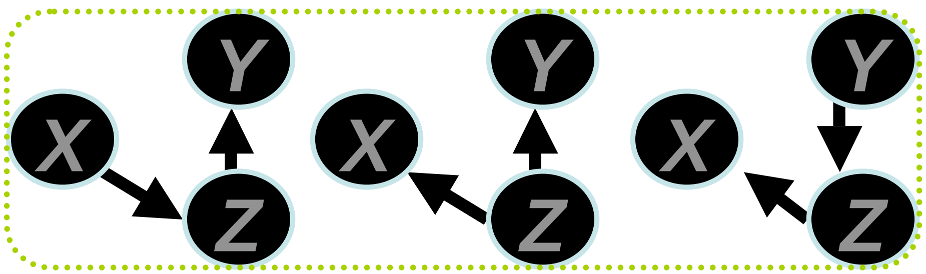
- Fundamental problem of causal search:
  - Multiple causal structures can produce the same associations and independencies
    - Even though the set of associations for any causal structure is determinate
  - Causation ↔ Association map is
    many ↔ one

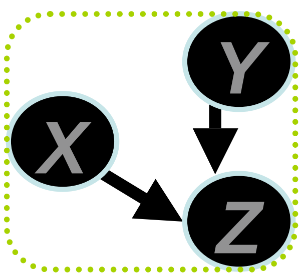
Markov Equivalence

- Formally, we say that:
  - Two causal graphs are <u>Observationally Markov Equivalent</u> (OME) if they imply the same set of conditional independence relations among the observed variables
    - By the Markov and Faithfulness assumptions
- We define the *OME class of graph G* to be the set of graphs OME to G

Markov Equivalence

- Examples:
  - 

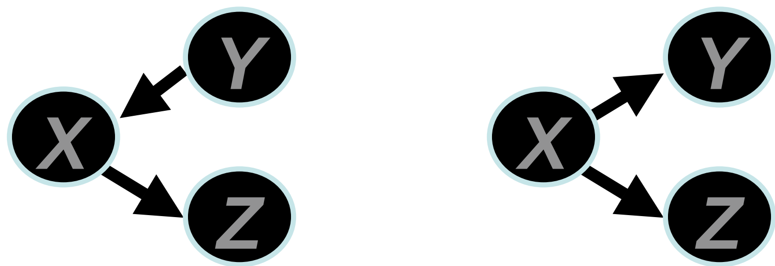
$\Rightarrow X \perp\!\!\!\perp \{Y, Z\}$
  - 

$\Rightarrow X \perp\!\!\!\perp Y \mid Z$
  - 

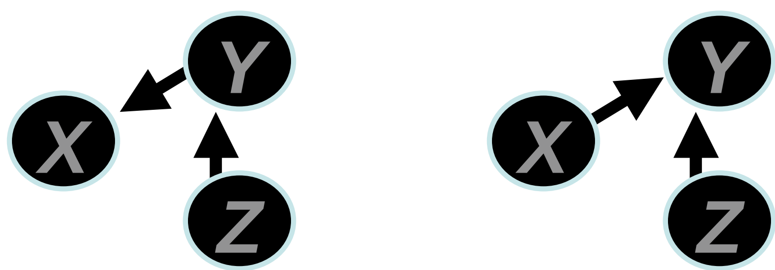
$\Rightarrow X \perp\!\!\!\perp Y$

Markov Equivalence

- Two more examples:
  - Are these graphs Markov equivalent?

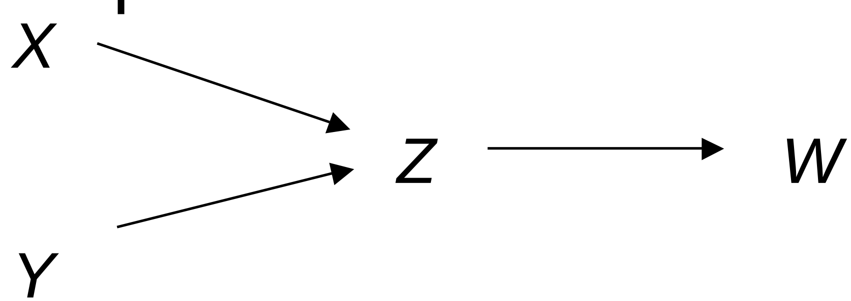


  - Are these two graphs?



Shared Structure

- Every member of an OME class has the same “skeleton” (adjacencies)
- They can also share directionality
  - Example:



Every graph in the OME class of this graph contains the path Z → W

Formal Problem of Search

- Given some data *D*, determine the OME class that predicts the observed associations and independencies
  - And then extract the shared structure

- Or, find the graphs that could have produced the observed data

General Features of Search

- Huge model and parameter spaces
  - Even when we (necessarily) use prior information about the family of probability distributions.
  - Relevant statistics must be rapidly computed
- But substantive knowledge about the domain may restrict the space of alternative models
  - Time order of variables
  - Required cause/effect relationships
  - Existence or non-existence of latent variables
- Depending on algorithm, search over graphs or directly over OME classes

Three Forms of Search

- Bayesian / score-based
  - Find the graph(s) with highest P(graph | data)
- Constraint-based
  - Find the graph(s) that predict exactly the observed associations and independencies
- Combined
  - Get “close” with constraint-based, and then find the best graph using score-based

Bayesian / Score-Based

- Informally:
  - Give each model an initial score using “prior beliefs”
  - Update each score based on the likelihood of the data if the model were true
  - Output the highest-scoring model
- Formally:
  - Specify $P(M, v)$ for all models M and possible parameter values v of M
  - For any data D, $P(D \mid M, v)$ can easily be calculated
  - $P(M \mid D) \propto \sum_v P(D \mid M, v) P(M, v)$

Bayesian / Score-Based

- In practice, this strategy is completely computationally intractable
  - There are too many graphs to check them all
- So, we use a greedy search strategy
  - Start with an initial graph and compare its posterior probability with that of each one step (or two step) modification by edge addition, deletion or reversal
  - Iterate until no improvement in posterior

Bayesian / Score-Based

- Problem #1: Local maxima
  - Sometimes, greedy searches get stuck

- Solution:
  - Greedy search on the OME *classes*, rather than on the graphs themselves (Meek)
    - Has a proof of correctness and convergence
    - But it gets to the right answer *slowly*

Bayesian / Score-Based

- Problem #2: Unobserved variables
  - Huge number of graphs
  - Huge number of different parameterizations
  - No fast, general way to compute likelihoods from latent variable models

- Partial solution:
  - Focus on a small, “plausible” set of models for which we can compute scores

Constraint-Based

- Find patterns of association in the data
  - Using a variety of statistical techniques
- Then “build” the OME class of graphs that predict data with this pattern of statistics
  - Note that we might have to introduce a latent variable to explain the pattern of statistics
- Output some description of the OME class
  - Or if there is no OME class that predicts the exact pattern, output known structure

Constraint-Based Output

- Constraint-based search algorithms typically output a *Pattern*, which encodes an OME class
  - Their output may (and typically will) have undirected edges.
  - Any set of edge directions that does not introduce new colliders is permissible

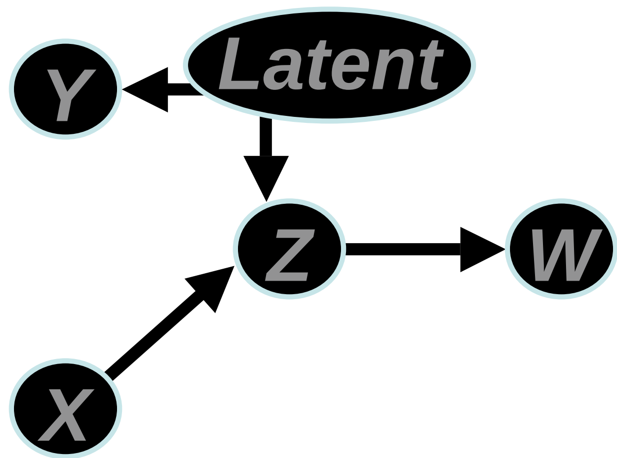
Constraint-Based Output

- Searches that allow for latent variables can also have edges of the form X o$\rightarrow$ Y
- This indicates one of three possibilities:
  - $X \rightarrow Y$
  - There is at least one unobserved common cause of X and Y
  - Both of these

Constraint-Based Example

- Let's walk through a full, realistic example

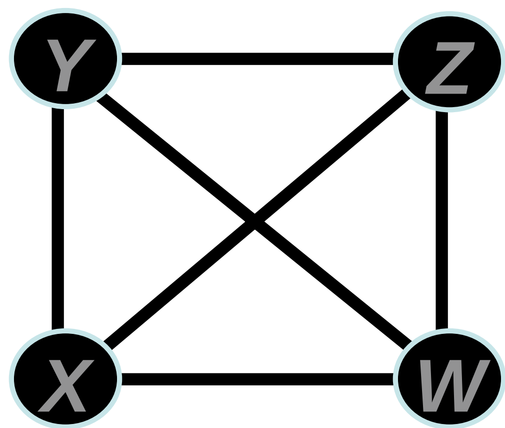
- Suppose the true structure is:



- We measure only *W*, *X*, *Y*, *Z*

Constraint-Based Example

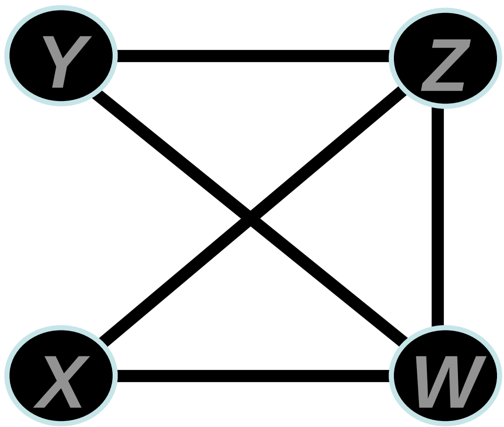
- Step 1: Form the complete graph using undirected edges



Constraint-Based Example

- Step 2: For each pair of variables, remove the edge between them if they're unconditionally independent

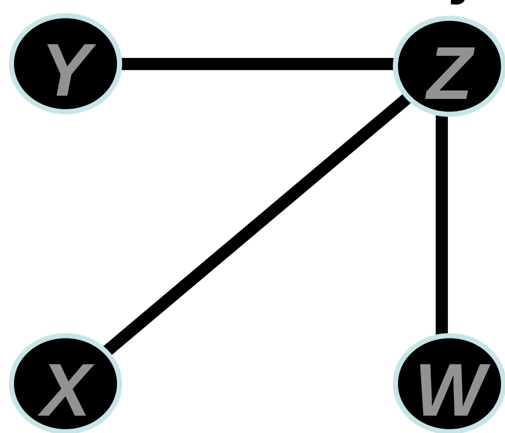
$$X \perp\!\!\!\perp Y \quad \Rightarrow$$



Constraint-Based Example

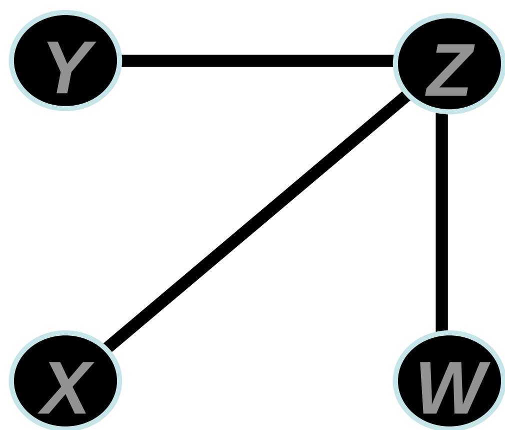
- Step 3: For each adjacent pair, remove the edge if they're independent conditional on some variable adjacent to one of them

$\{X, Y\} \perp\!\!\!\perp W \mid Z \Rightarrow$



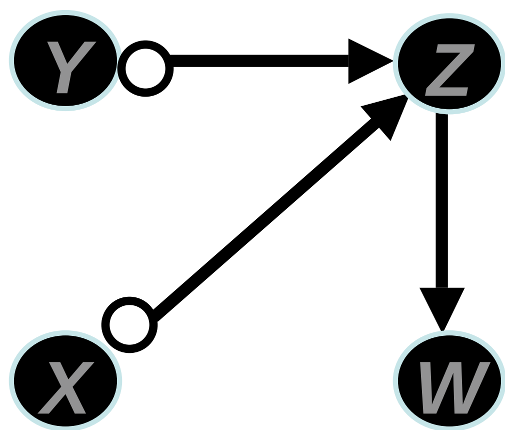
Constraint-Based Example

- Step 4: Continue removing edges, checking independence conditional on 2 (or 3, or 4, or…) variables



Constraint-Based Example

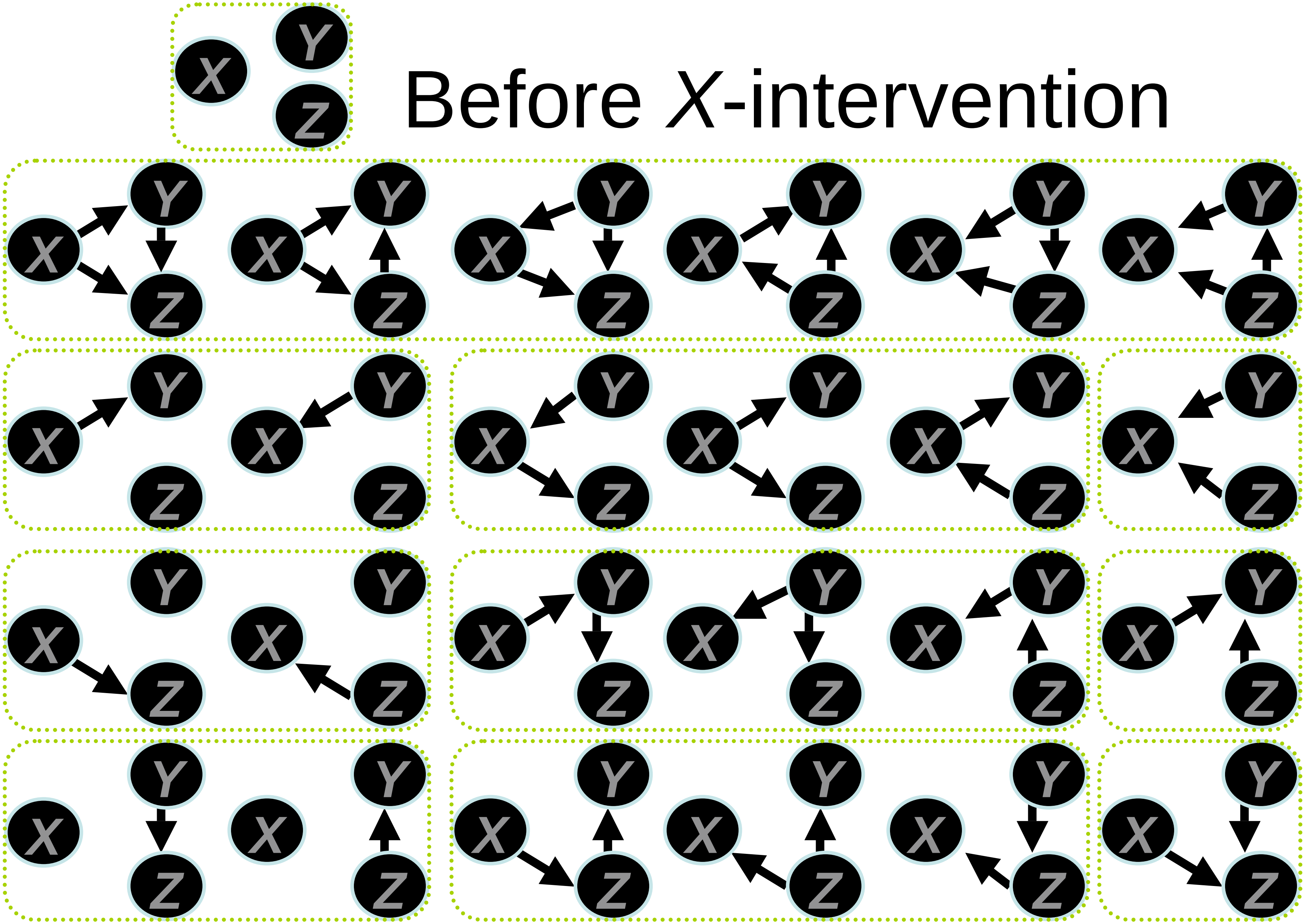
- Step 5: Orientation
  - For *X* – *Z* – *Y*, if $X \perp\!\!\!\perp Y$ without conditioning on *Z*, then make *Z* a collider
  - If *Z* is a collider, orient *Z* – *W* away from *Z*

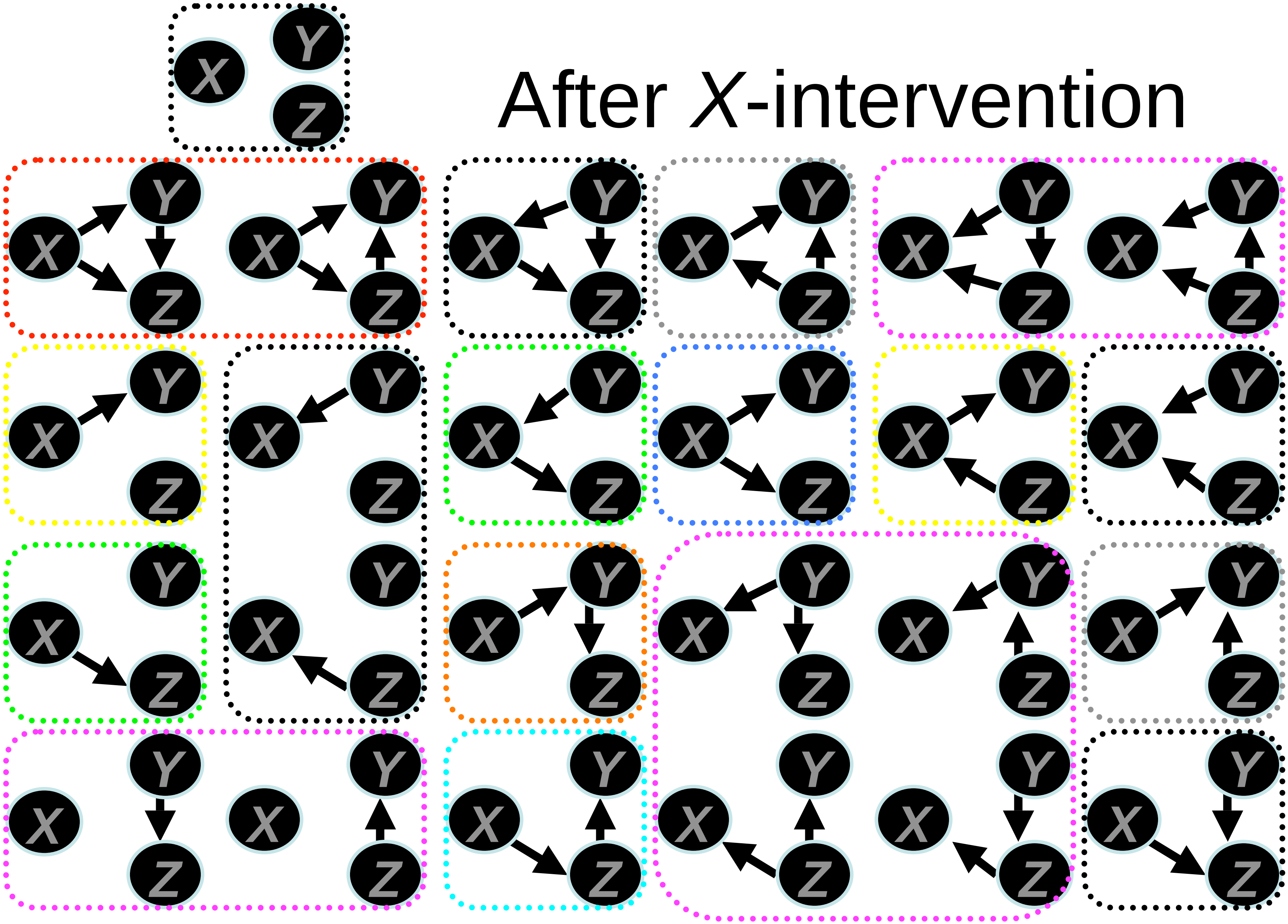


Interventions to the Rescue?

- Interventions helped us solve an earlier equivalence class problem
  - Randomization meant that:
    Treatment-Effect association $\Rightarrow T \rightarrow E$

- Interventions alter equivalence classes, but don't make them all into singletons
  - The fundamental problem of search remains


Before X-intervention

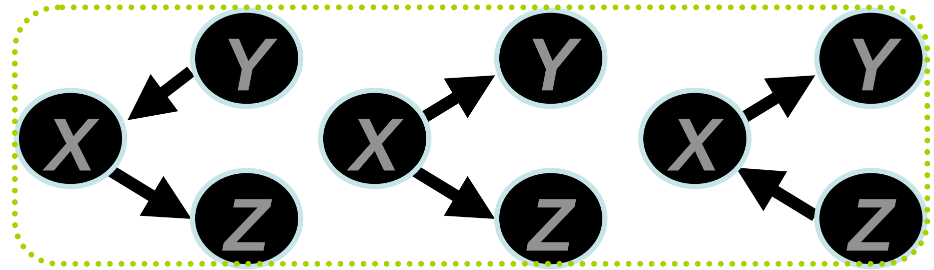


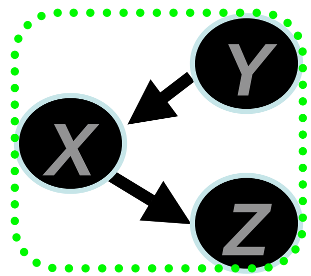
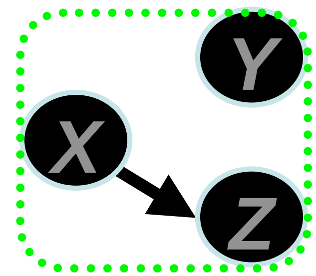
After X-intervention

Search with Interventions

- Search with interventions is the same as search with observations, except
  - We adjust the graphs in the search space to account for the intervention
- For multiple experiments, we search for graphs in every output equivalence class
  - More complicated than this in the real world due to sampling variation

Example

- Observation
  - $Y \perp\!\!\!\perp Z \mid X$ $\Rightarrow$ 

- Intervention on *X*
  - $Y \perp\!\!\!\perp Z$ $\Rightarrow$  & 

- Only possible graph: