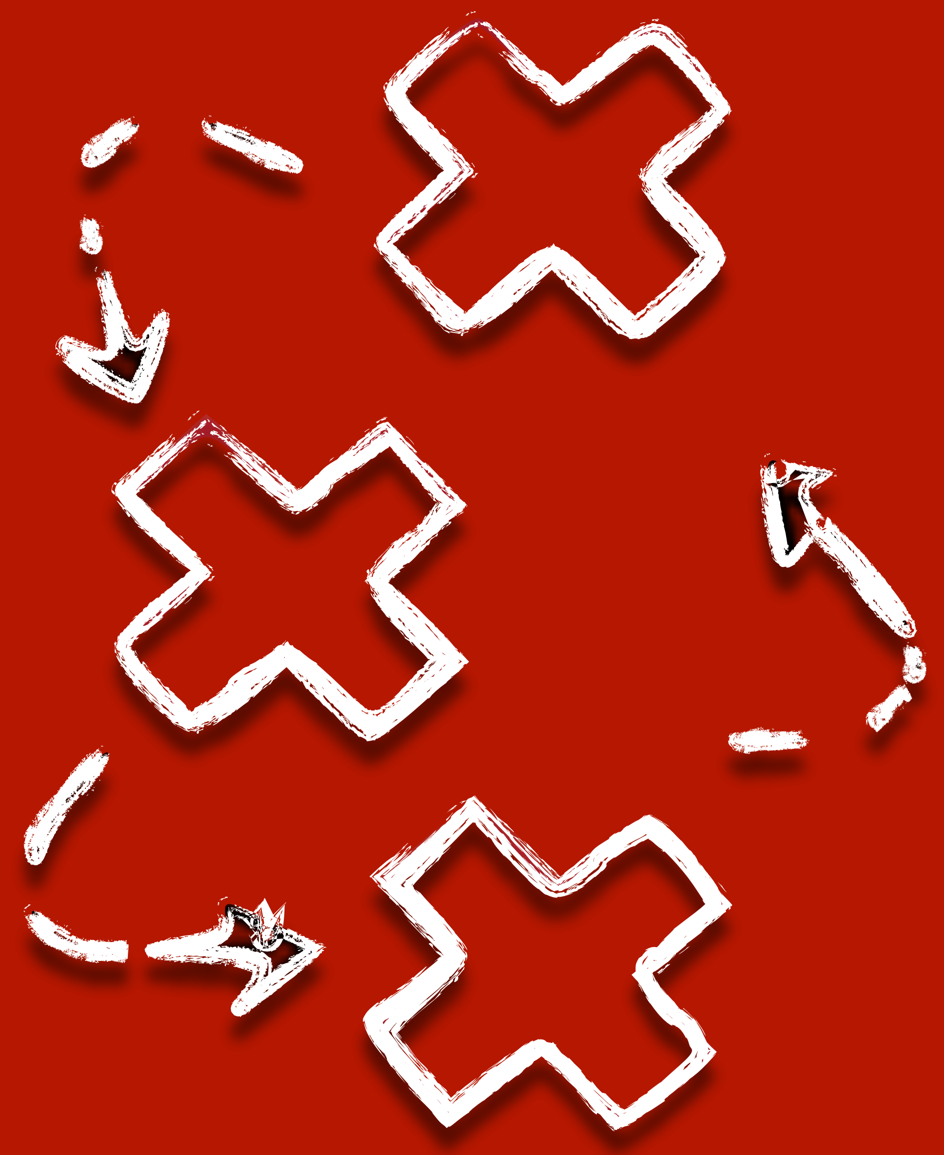


Constraint-based causal discovery

dr. Sara Magliacane (University of Amsterdam, MIT-IBM Watson AI Lab)

SIKS

This class

- Introduction to causal discovery
  - Common assumptions: causal sufficiency, acyclicity, faithfulness
- Constraint-based causal discovery on observational data (causal sufficiency)
  - SGS, PC

- Learning from multiple contexts or interventional data
  - Invariant Causal Prediction
  - Joint Causal Inference

Inspired by https://stat.ethz.ch/lectures/ss21/causality.php

Does chocolate cause Nobel prizes?

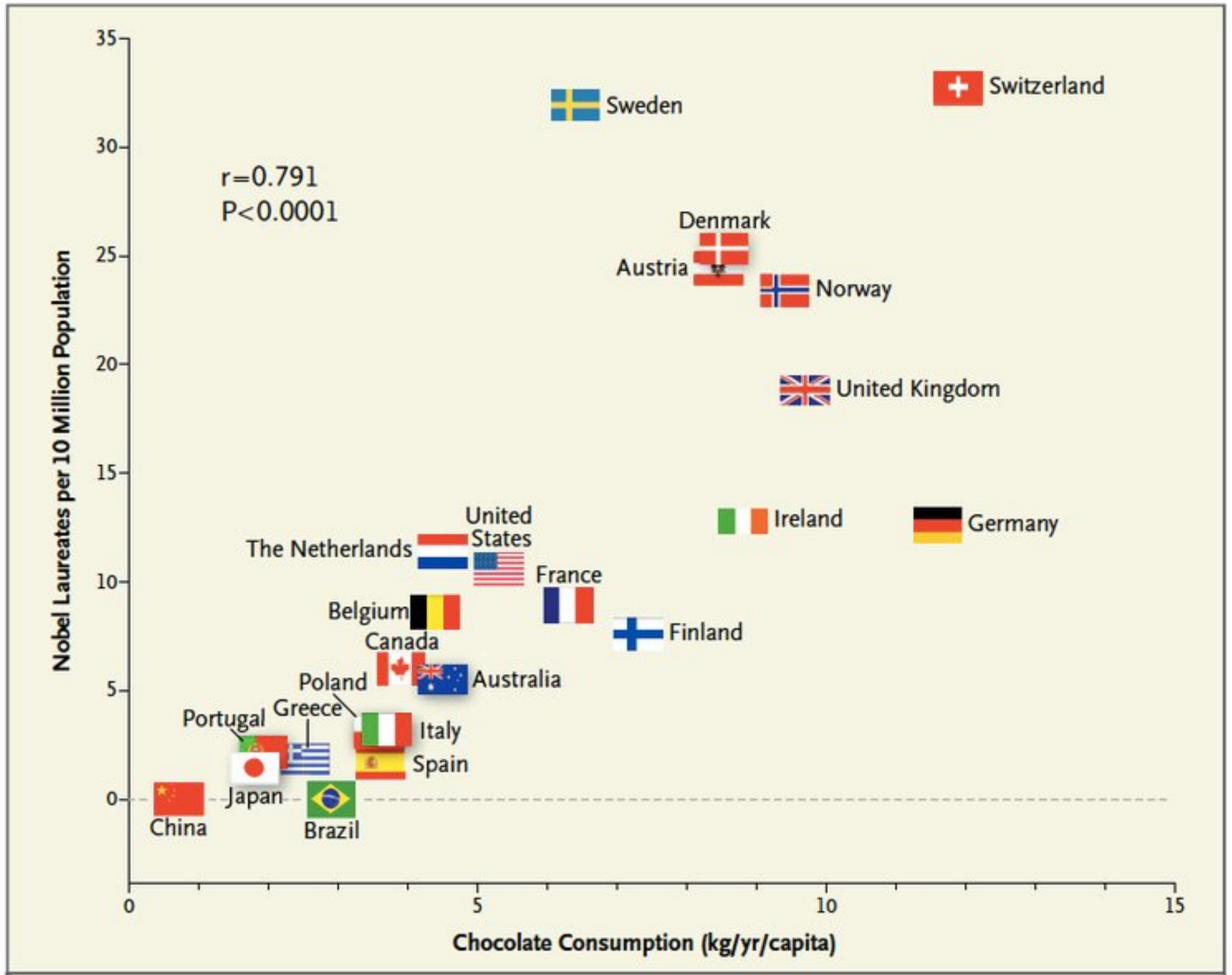


Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.

[Messerli, 2012] https://www.nejm.org/doi/full/10.1056/NEJMon1211064

What is the causal graph here?

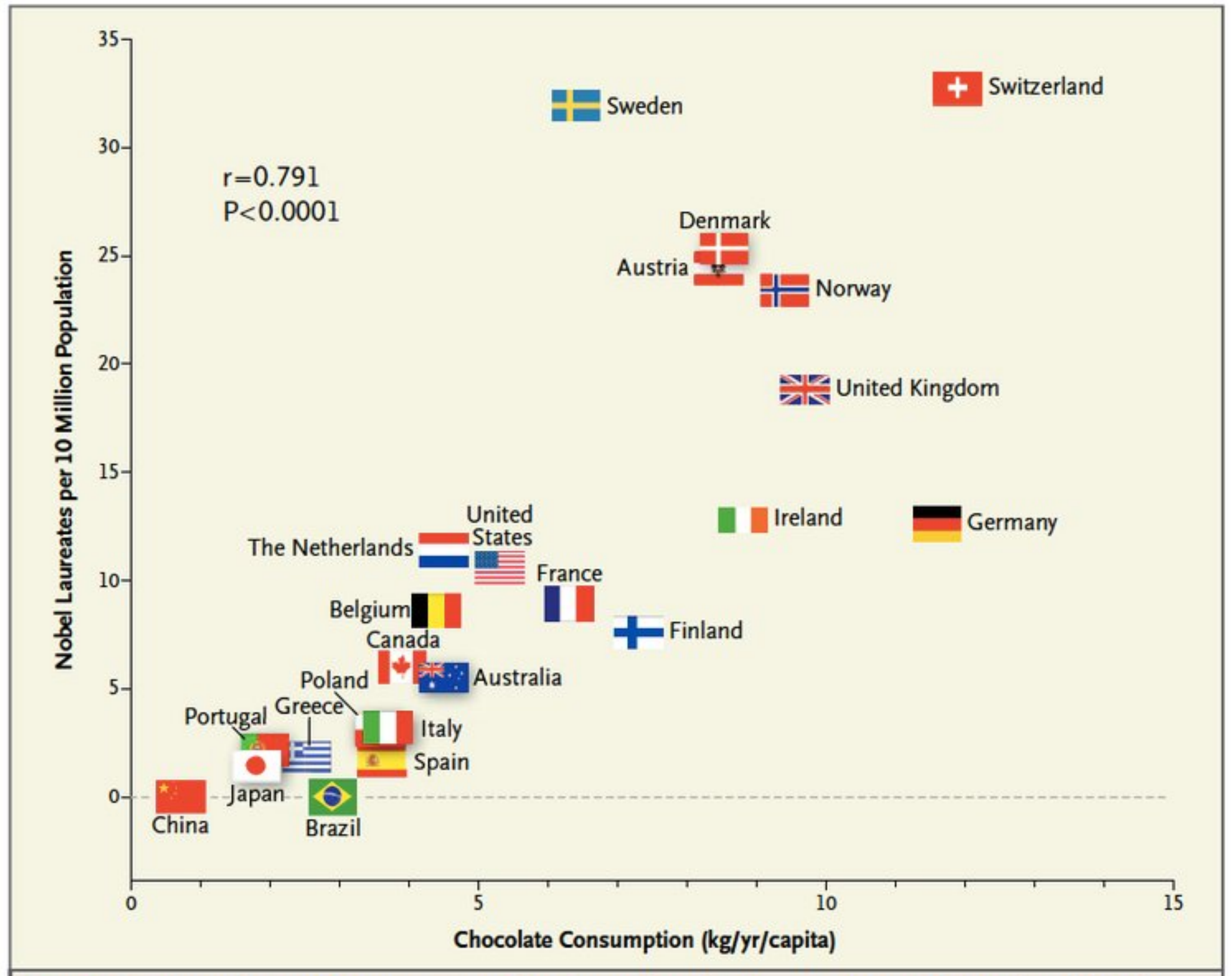


Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.

Reichenbach's principle of common cause: A correlation between X and Y implies that

- X causes Y, or
- Y causes X, or
- There exists a common cause between X and Y *(a confounder)*

(or any combination of the above)

((we ignore selection bias*))

What is the causal graph here?

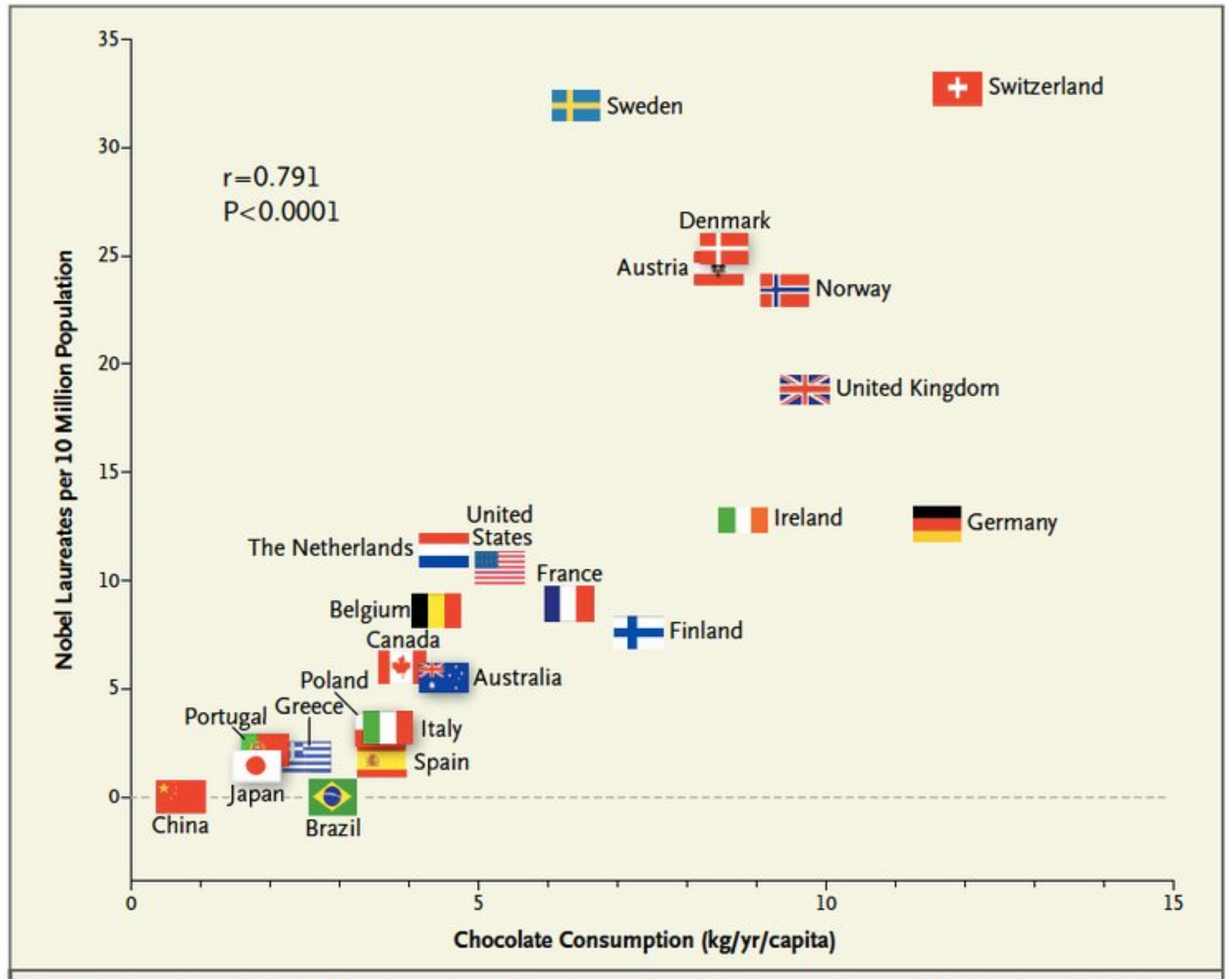
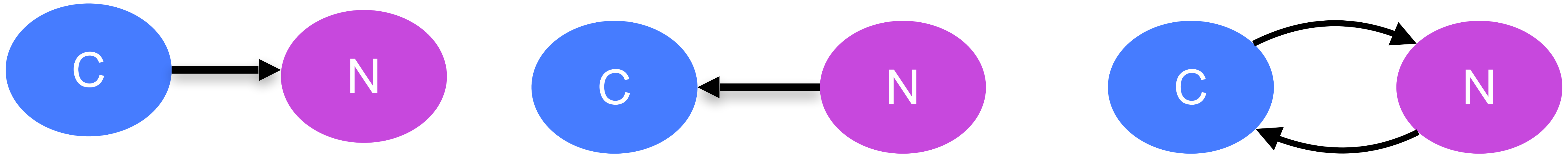


Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.



Reichenbach's principle of common cause:

A correlation between X and Y implies that X causes Y, Y causes X, or there exists a common cause between X and Y (or any combination)

What is the causal graph here?

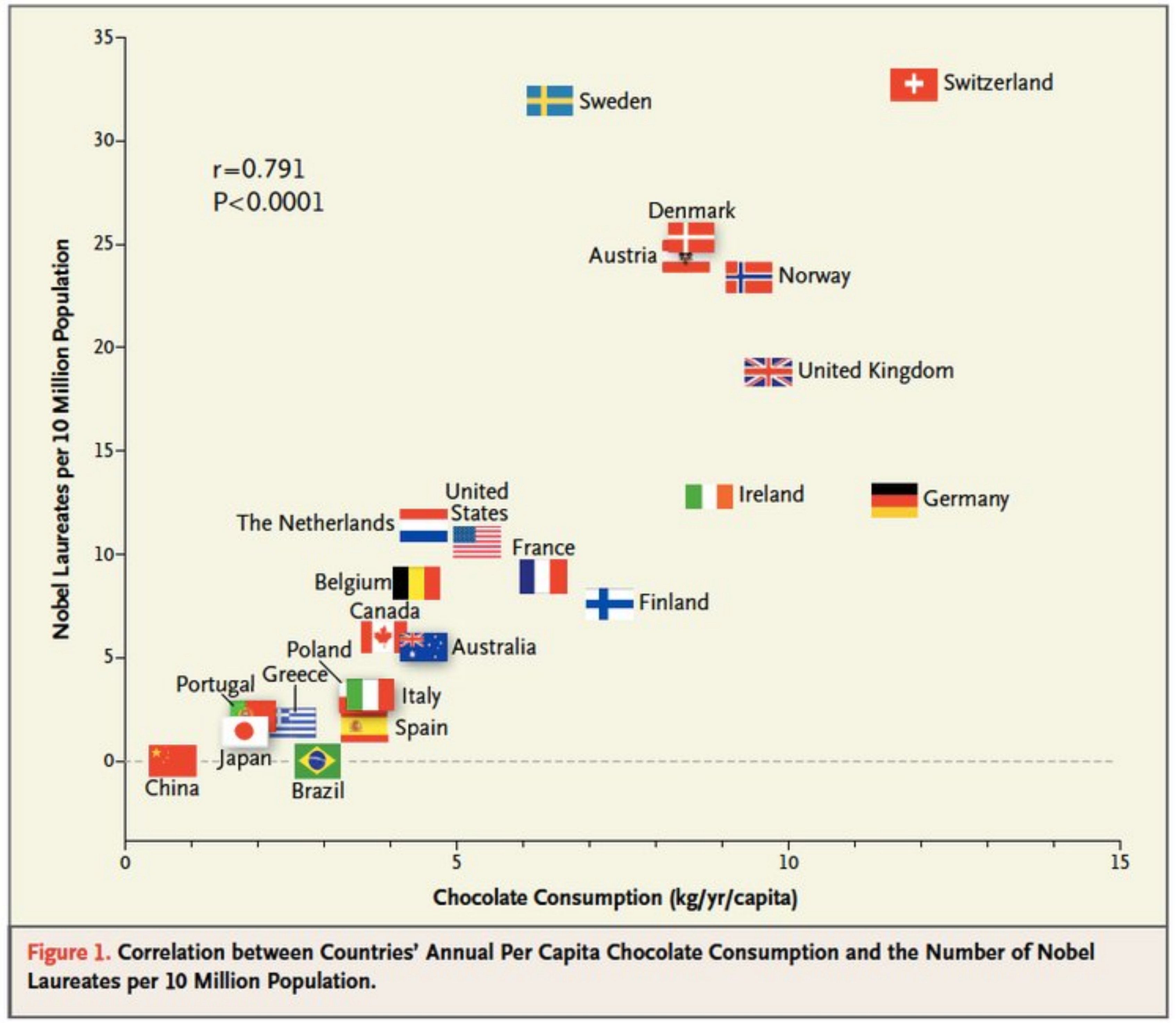
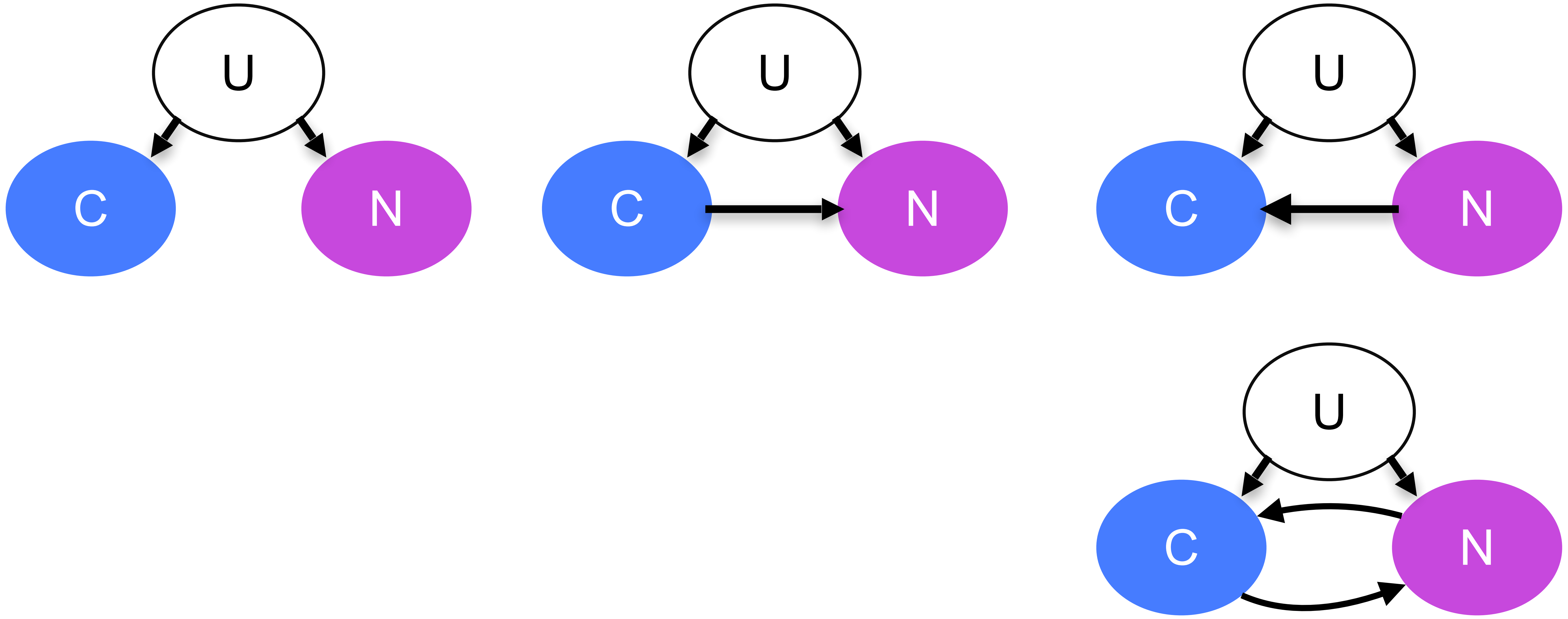


Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.



Reichenbach's principle of common cause:

A correlation between X and Y implies that X causes Y, Y causes X, or there exists a common cause between X and Y (or any combination)

Latent confounding

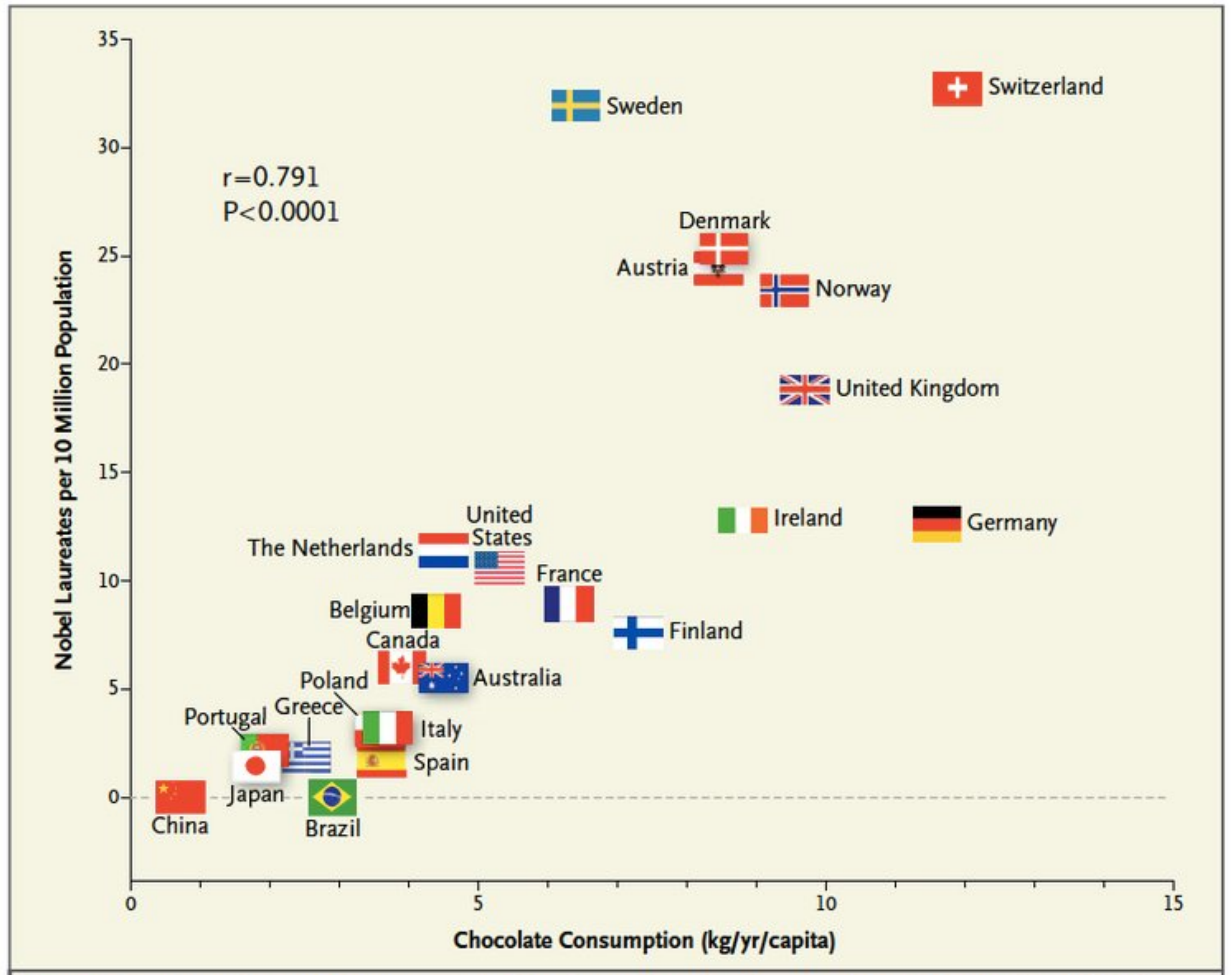
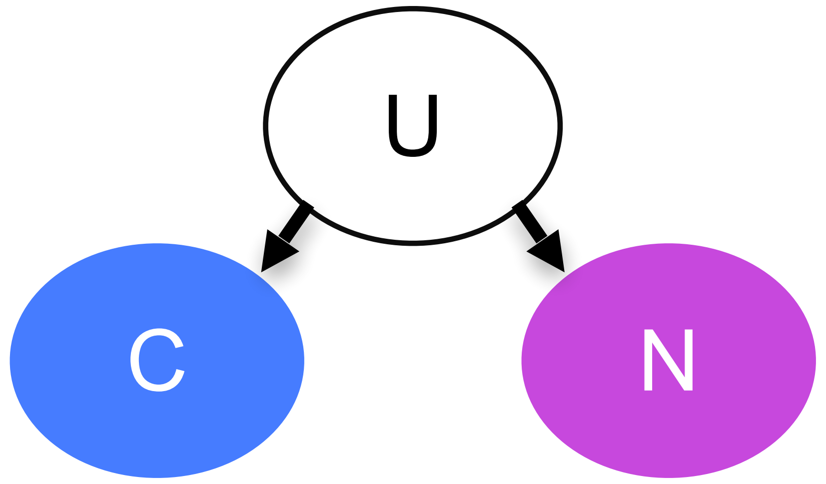


Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.



Latent confounding

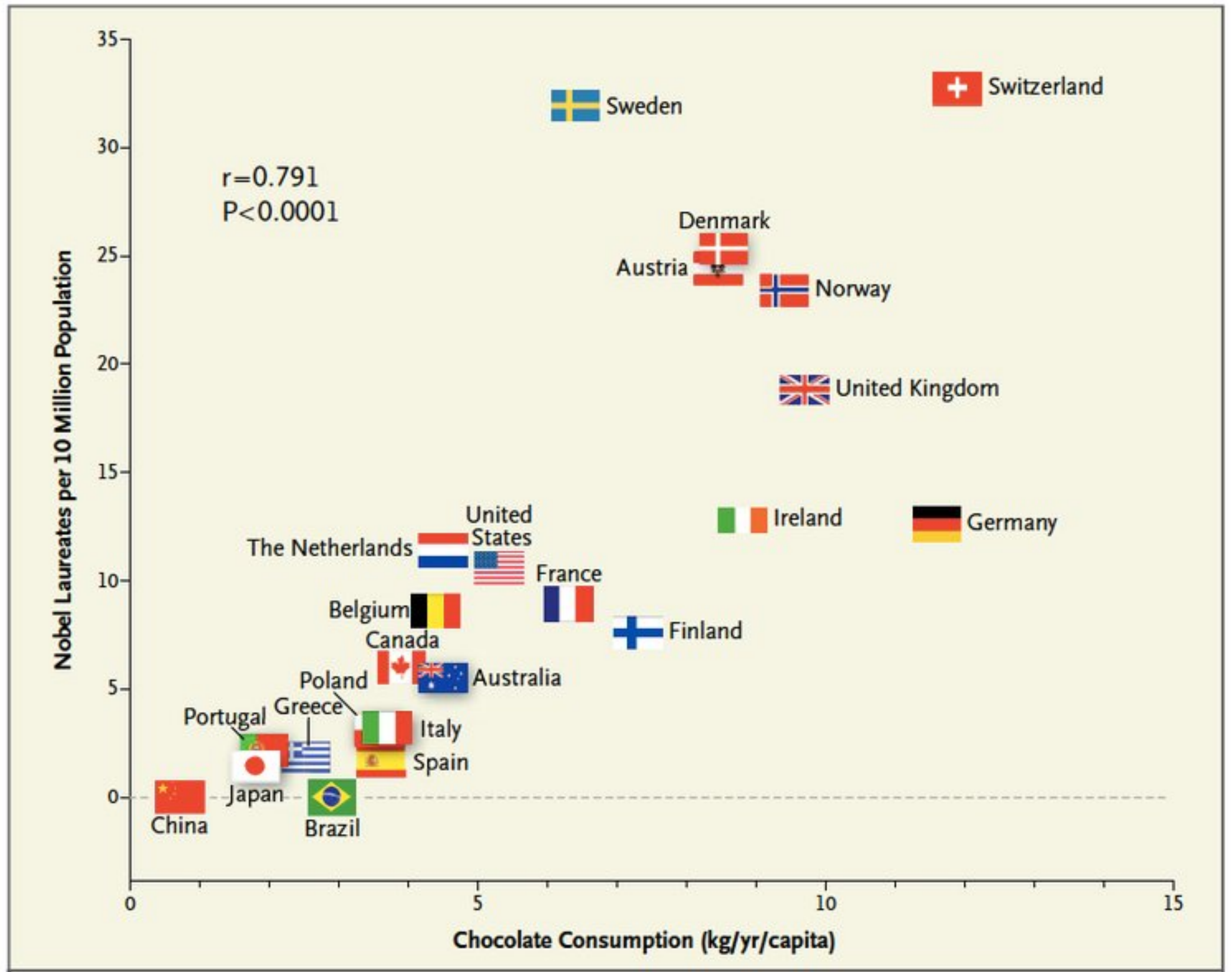
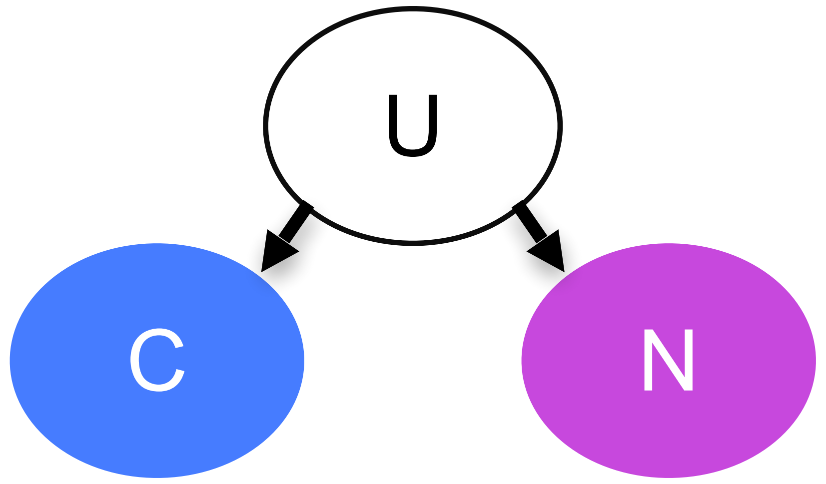
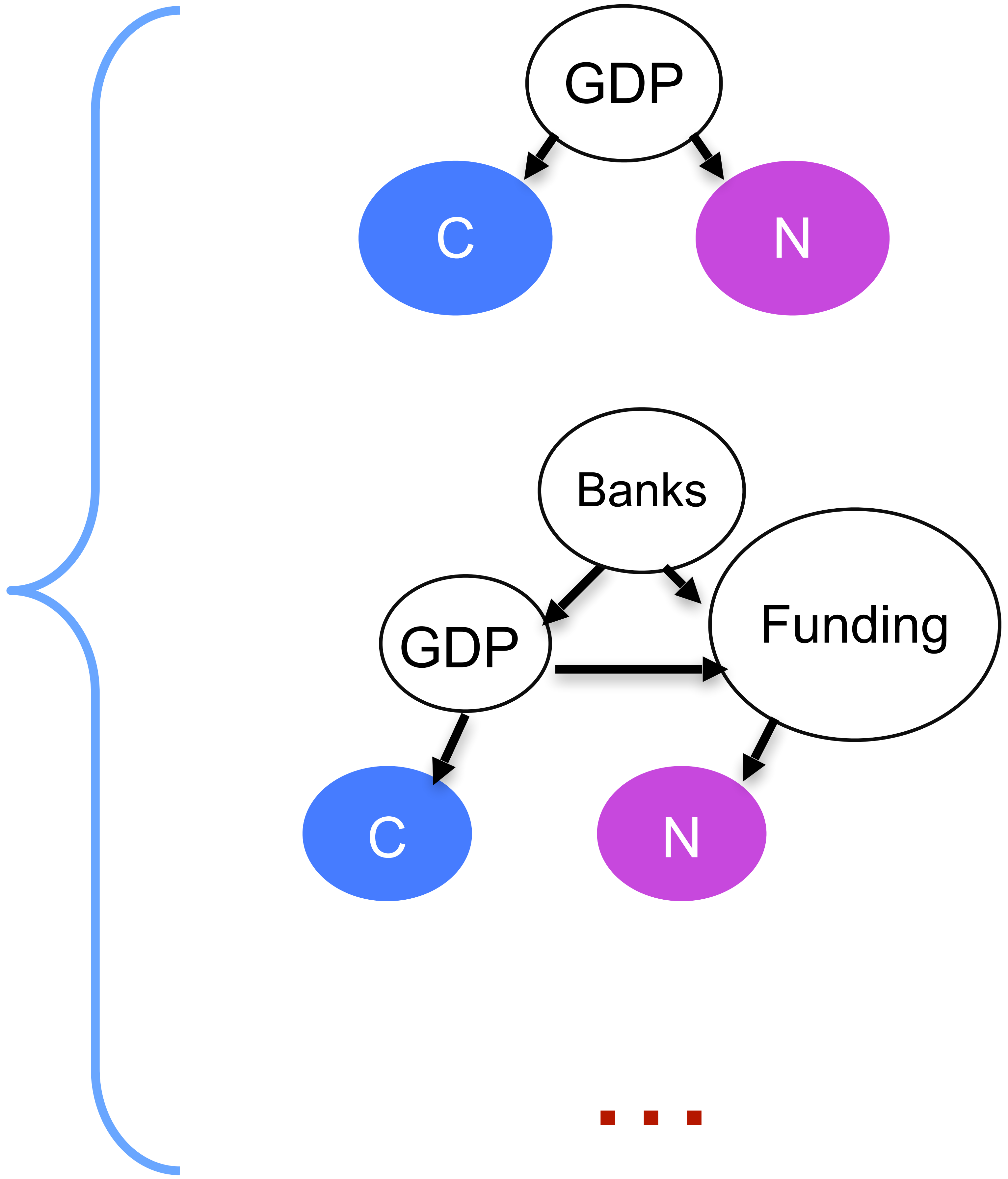


Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.





Statistically indistinguishable from only these data

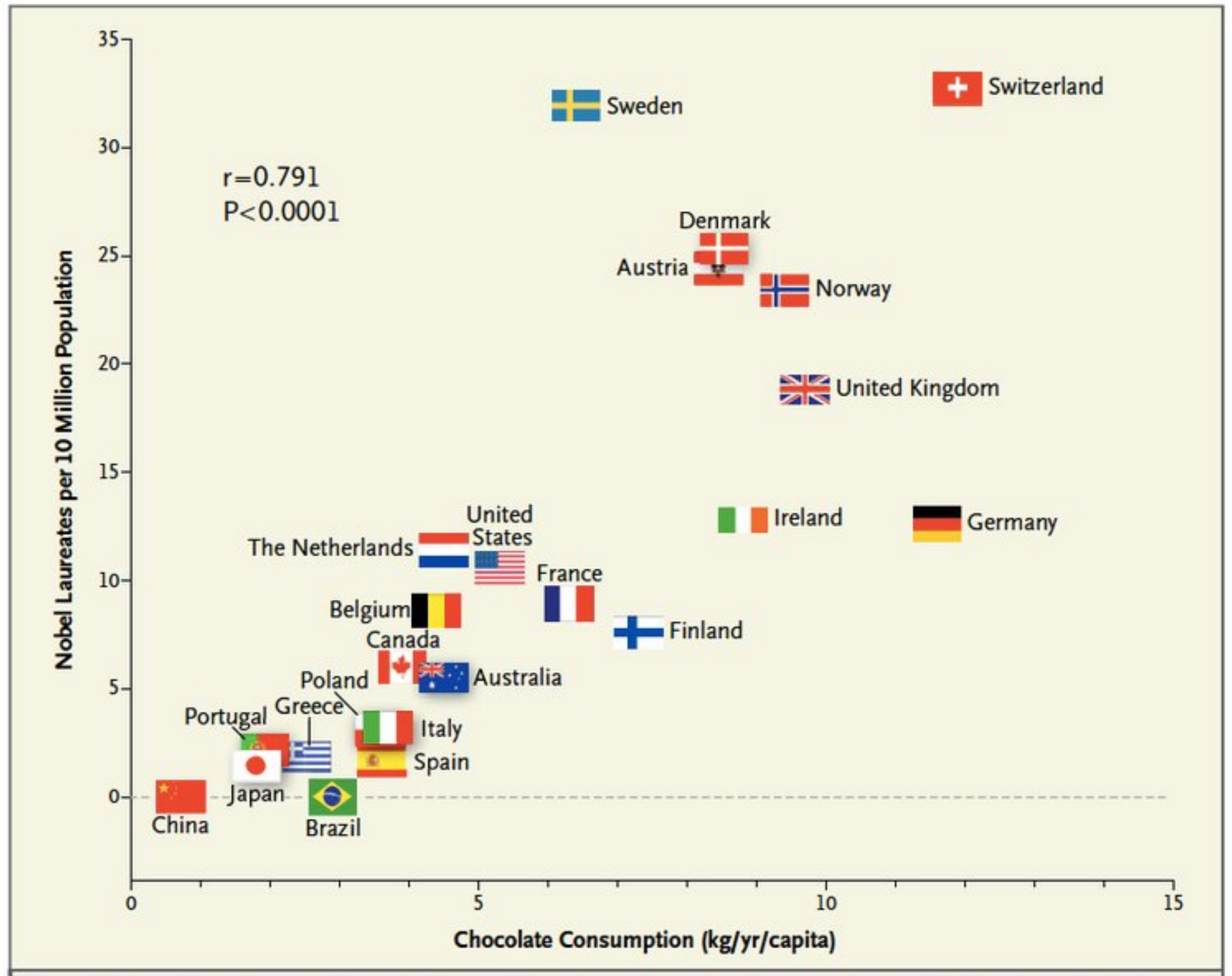
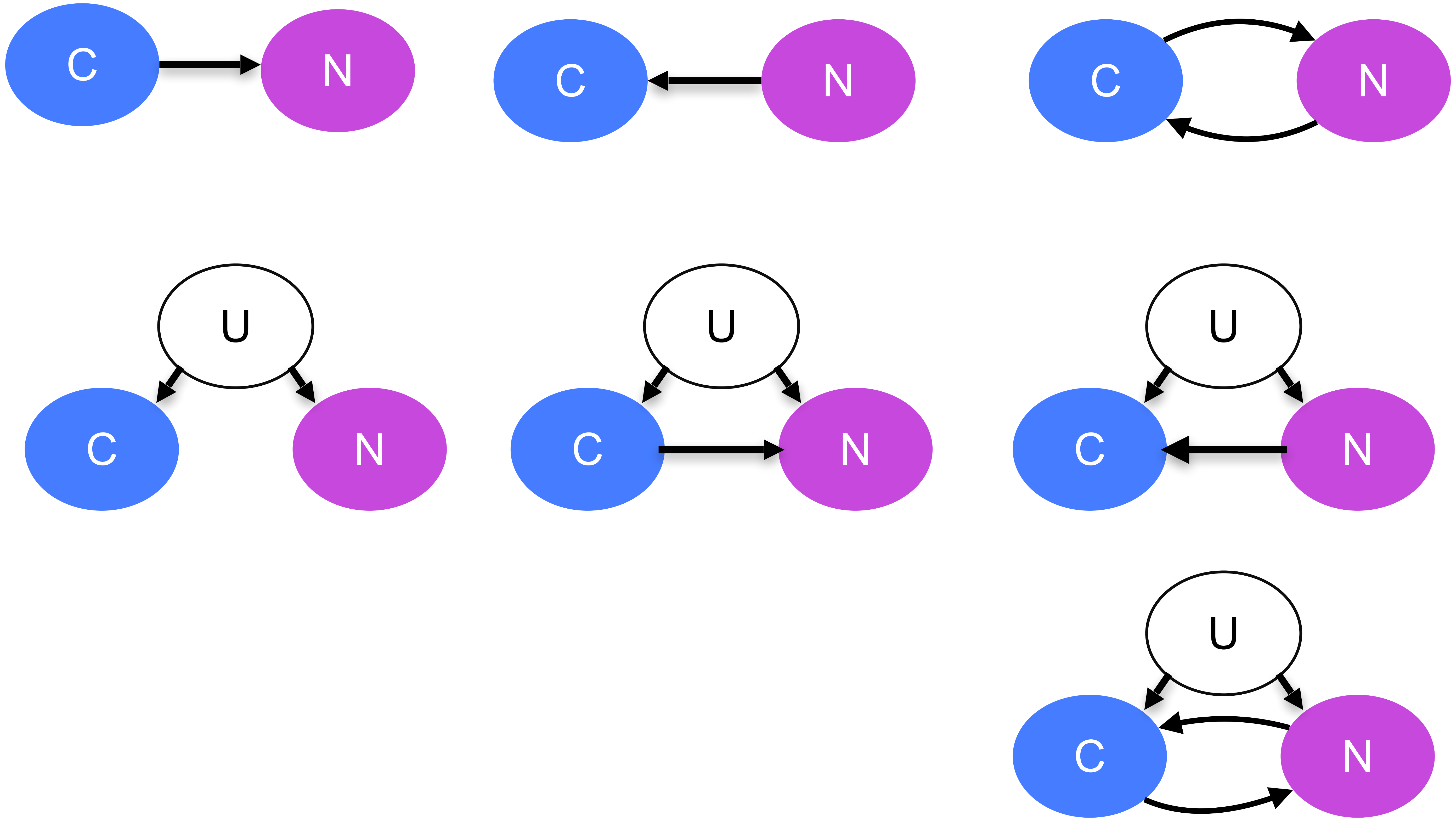


Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.



Statistically indistinguishable from only these data

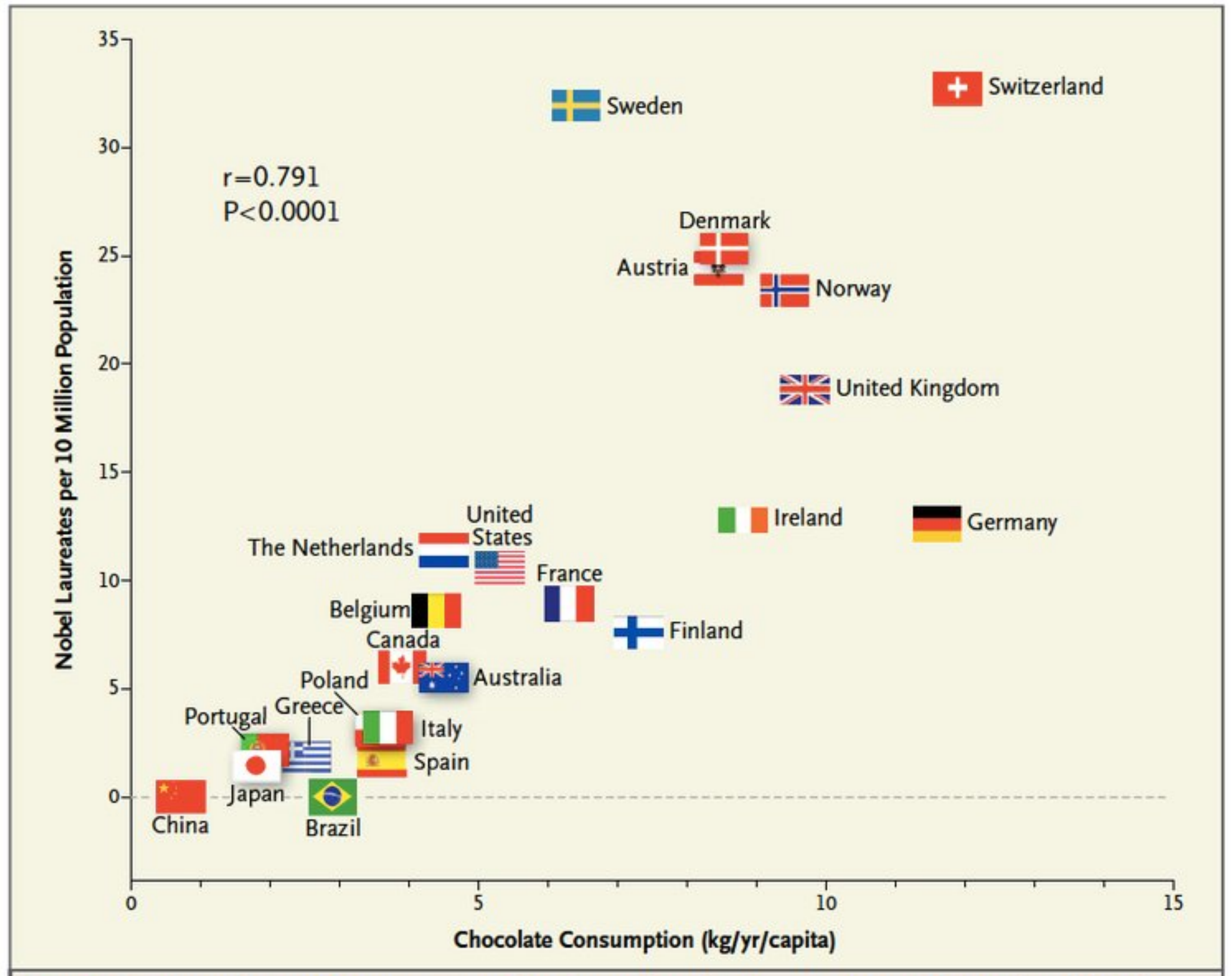
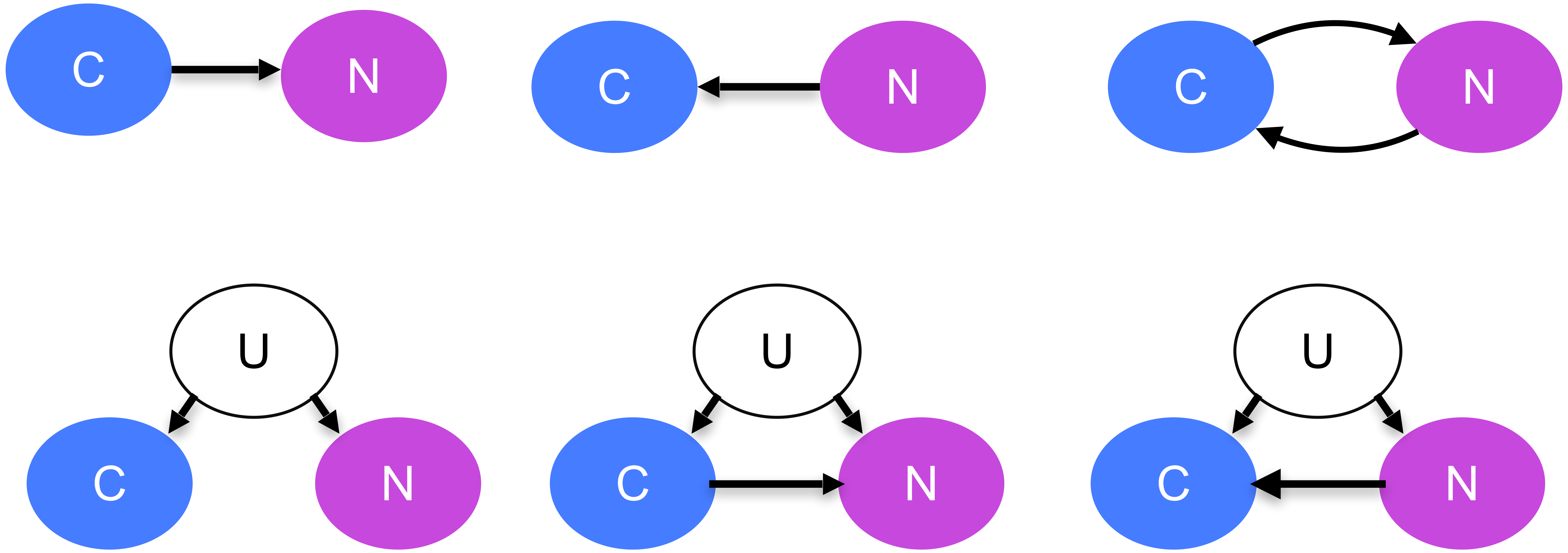
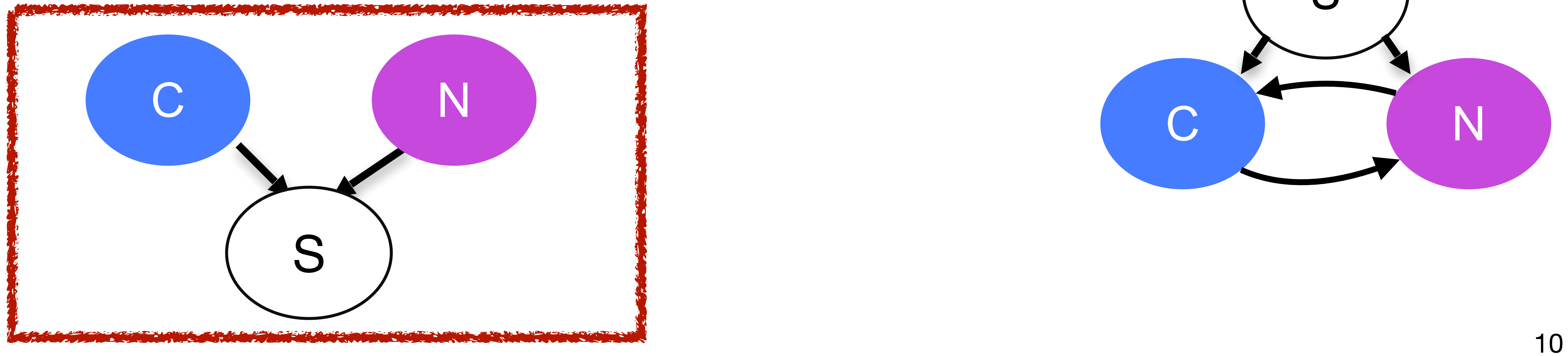


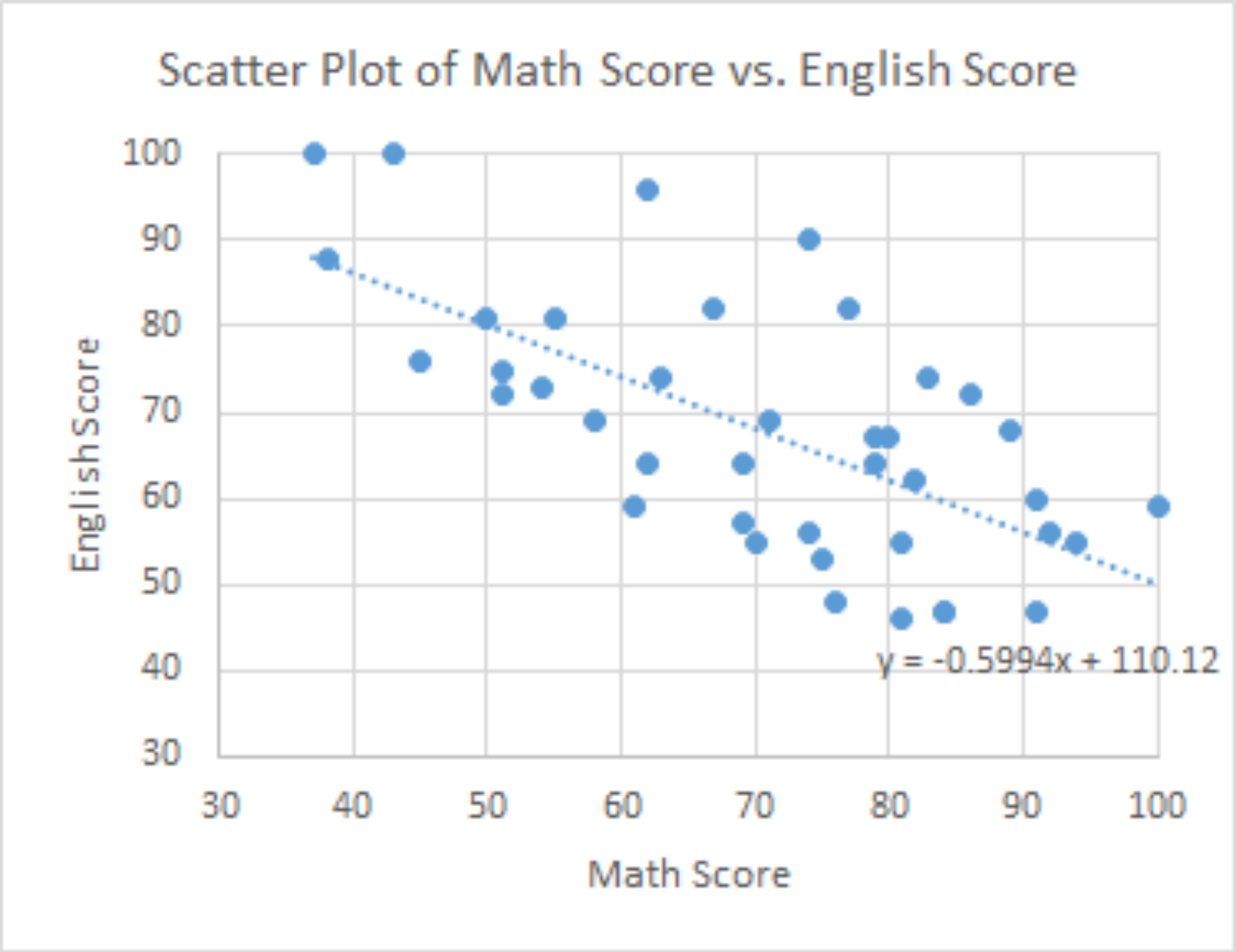
Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.



Or even selection bias



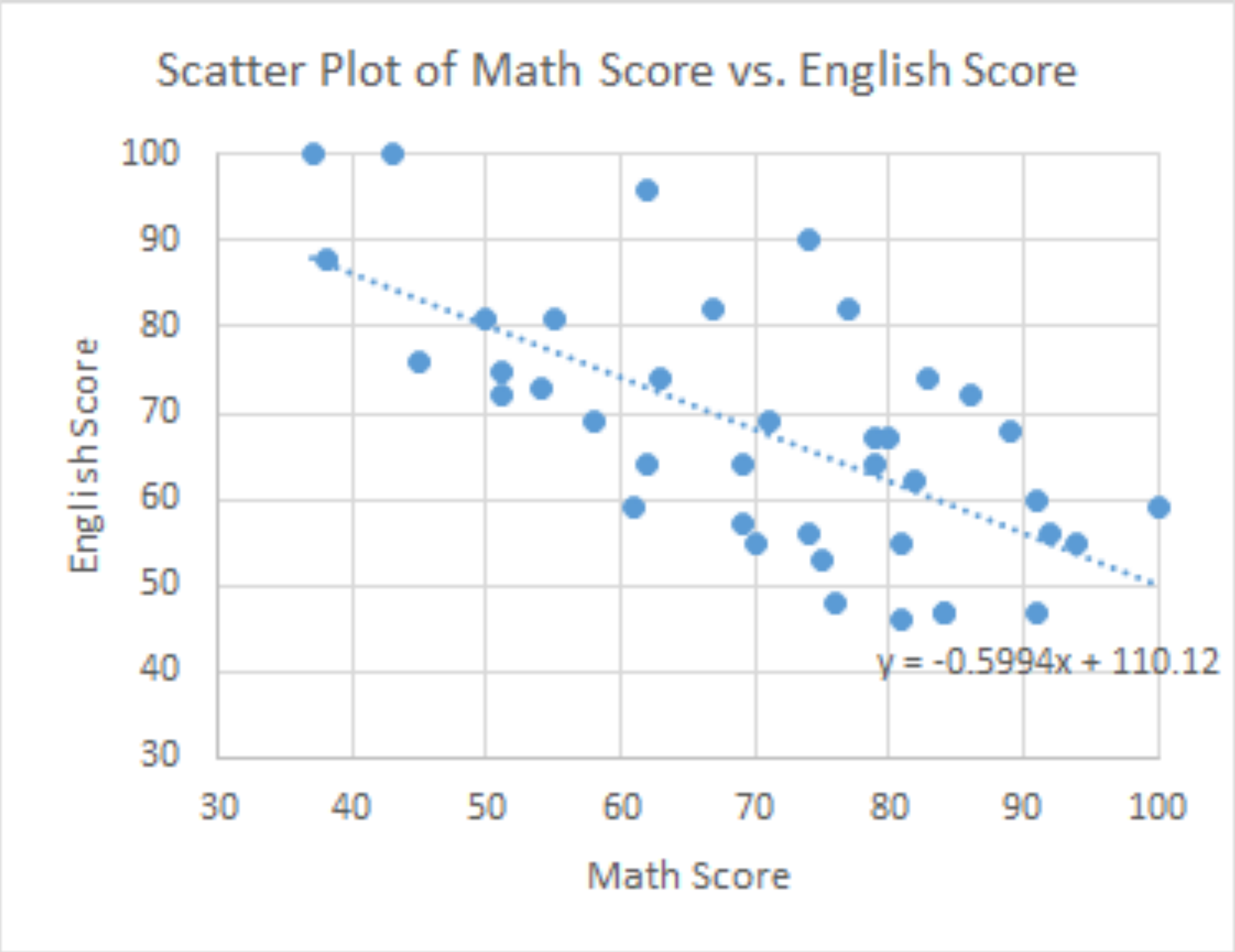
An example of selection bias

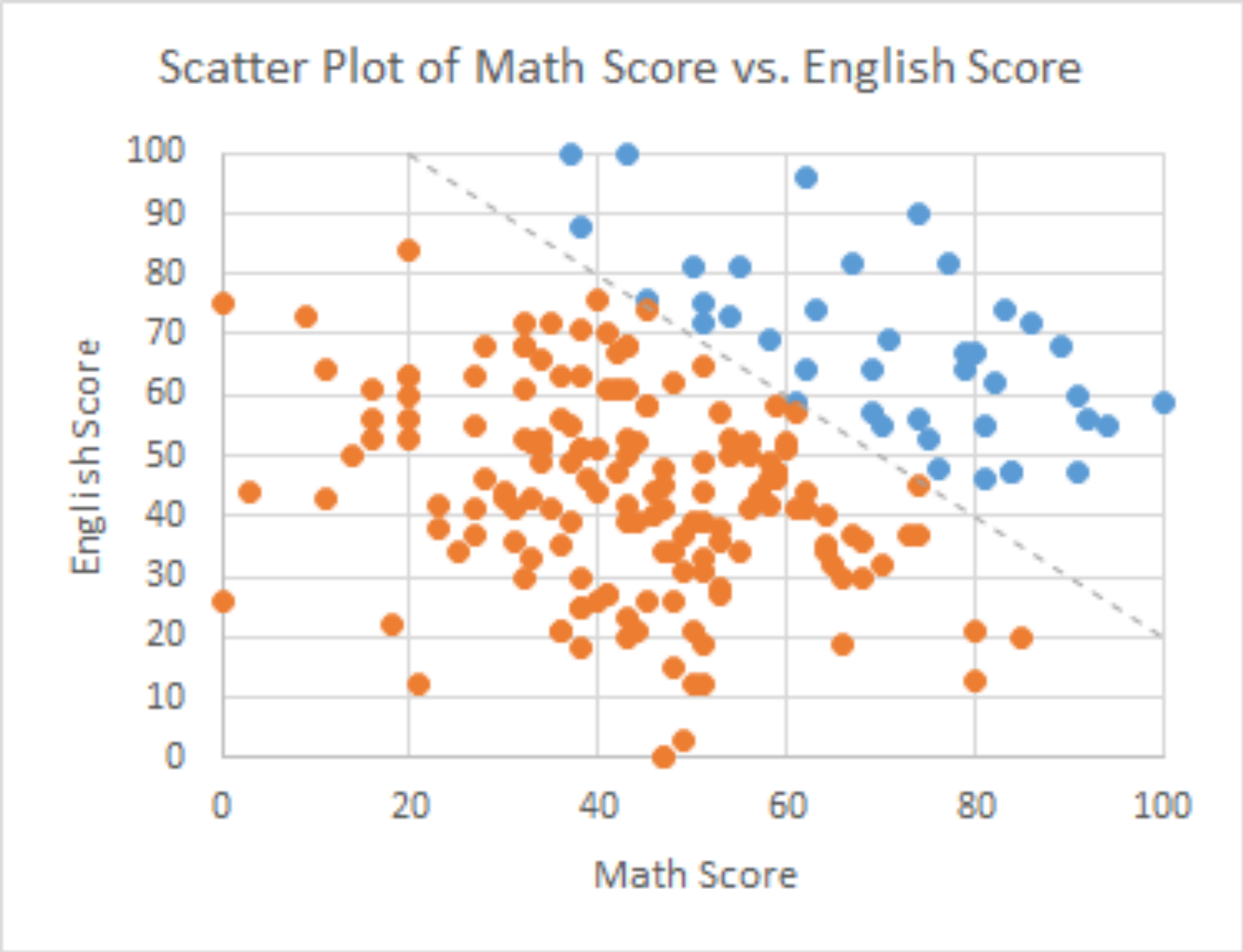


College admissions: English score is negatively correlated with Math Score

https://medium.com/analytics-vidhya/simpsons-paradox-when-you-derive-a-wrong-insight-from-your-analysis-ee488b346427

An example of selection bias

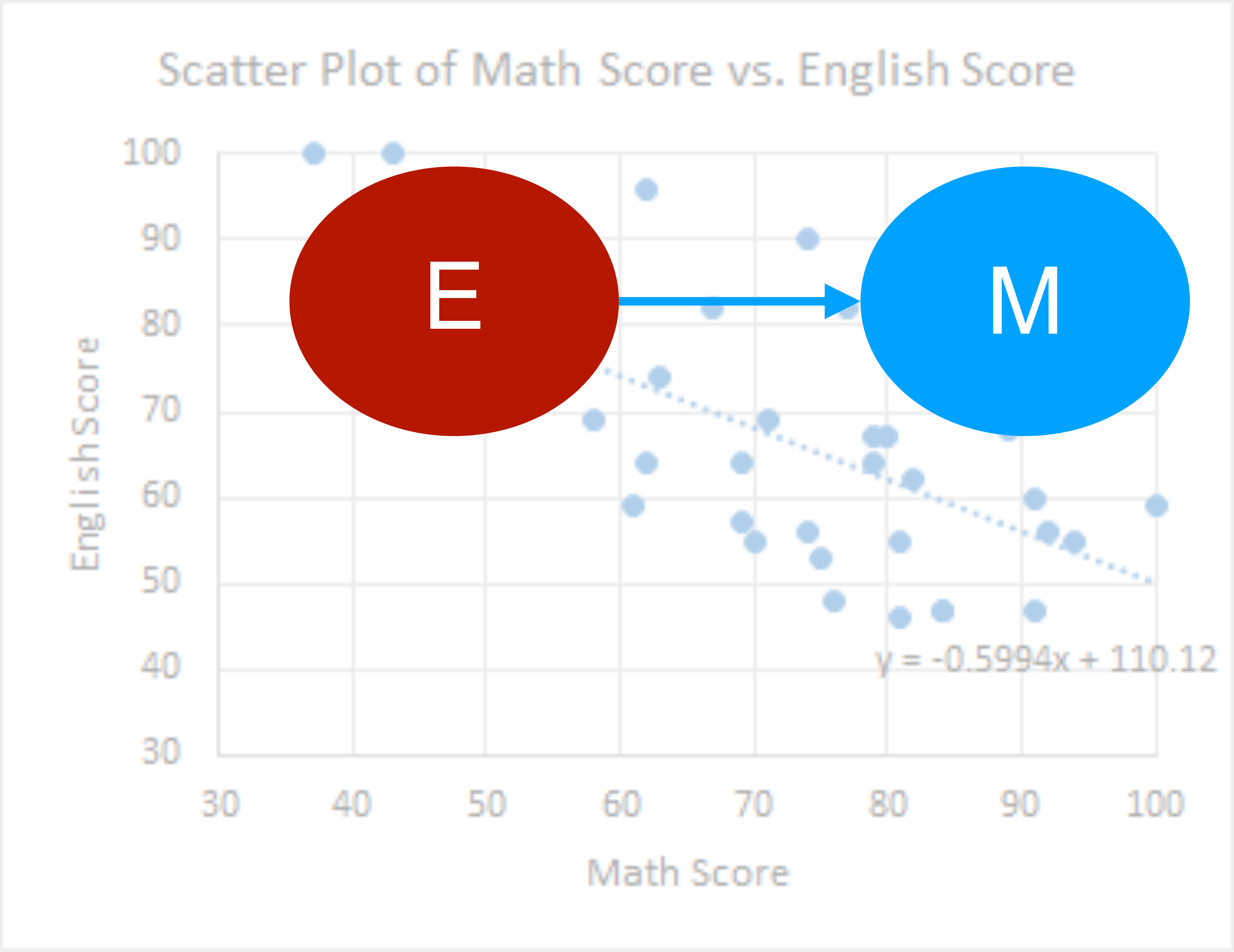


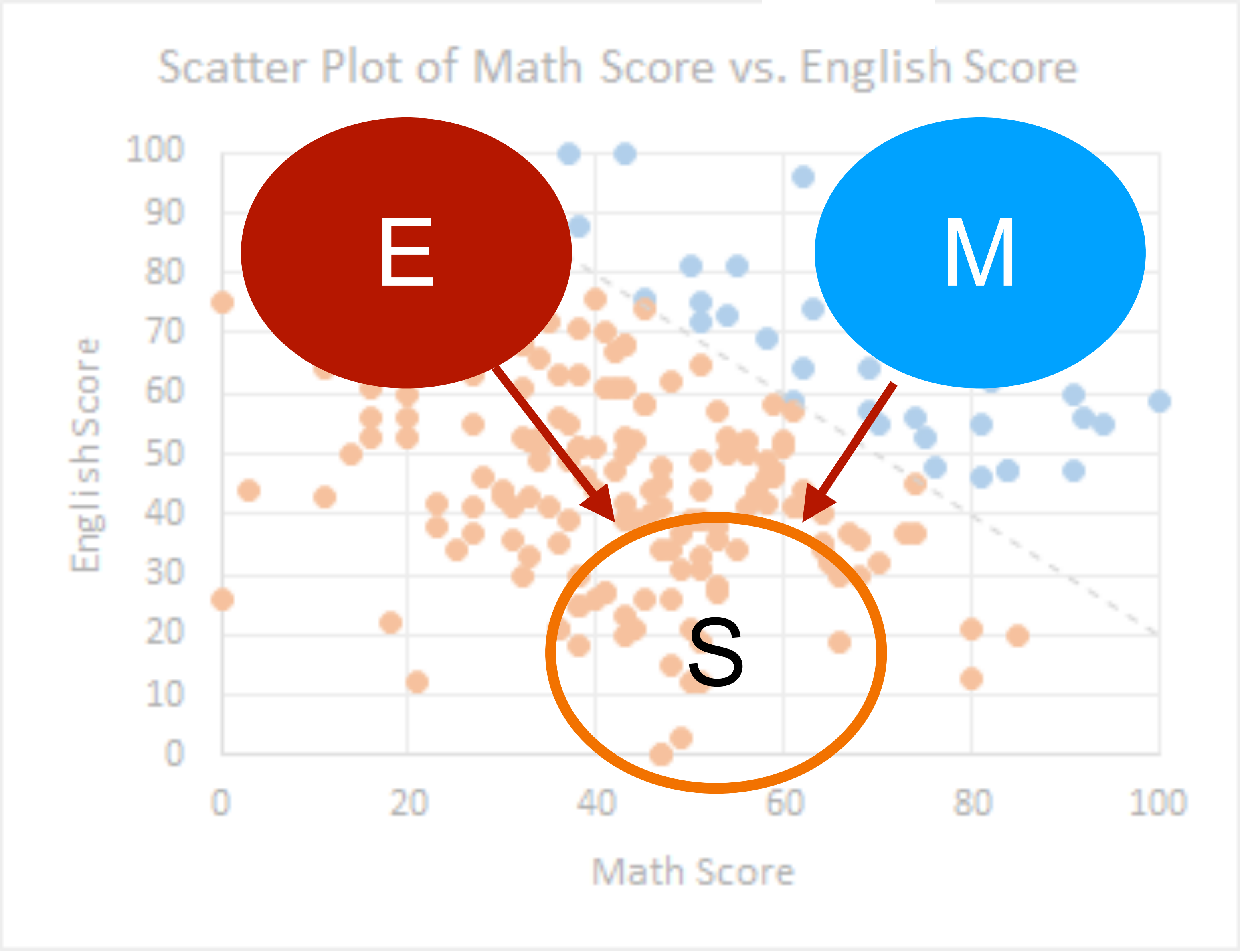


Selection rule: English score + Math Score > 120

https://medium.com/analytics-vidhya/simpsons-paradox-when-you-derive-a-wrong-insight-from-your-analysis-ee488b346427

An example of selection bias





Selection rule: English score + Math Score > 120

https://medium.com/analytics-vidhya/simpsons-paradox-when-you-derive-a-wrong-insight-from-your-analysis-ee488b346427

Common assumptions

- If p is **Markov and faithful** to G, then for any disjoint $\mathbf{A}, \mathbf{B}, \mathbf{C} \subseteq \mathbf{V}$:

$$\mathbf{A} \perp_G \mathbf{B} \,|\, \mathbf{C} \iff X_{\mathbf{A}} \perp\!\!\!\perp_p X_{\mathbf{B}} \,|\, X_{\mathbf{C}}$$

Common assumptions

- If p is **Markov and faithful** to G, then for any disjoint $\mathbf{A}, \mathbf{B}, \mathbf{C} \subseteq \mathbf{V}$:

$$\mathbf{A} \perp_G \mathbf{B} \,|\, \mathbf{C} \iff X_{\mathbf{A}} \perp\!\!\!\perp_p X_{\mathbf{B}} \,|\, X_{\mathbf{C}}$$

- **Causal sufficiency** - no **latent** confounders (common causes), no selection bias

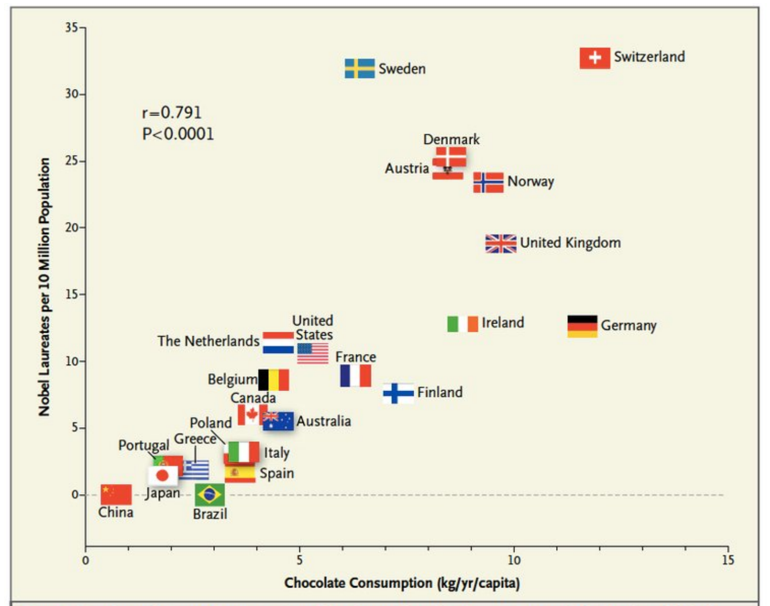
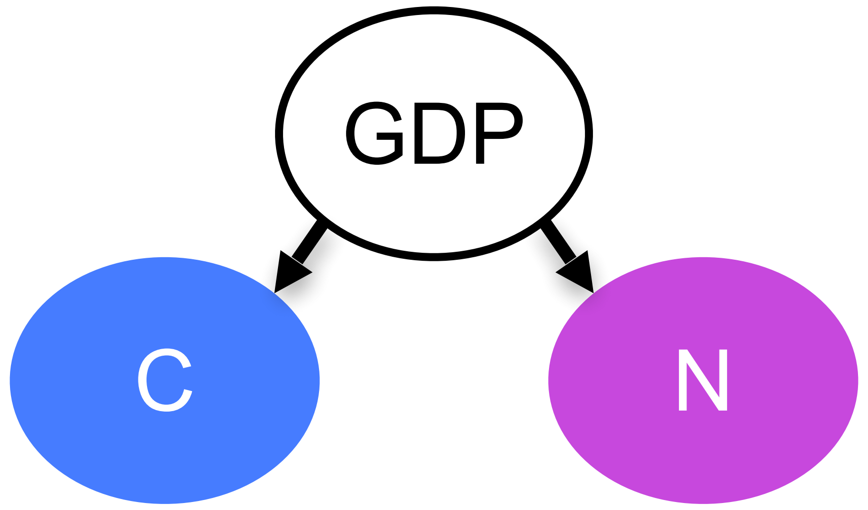


Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.



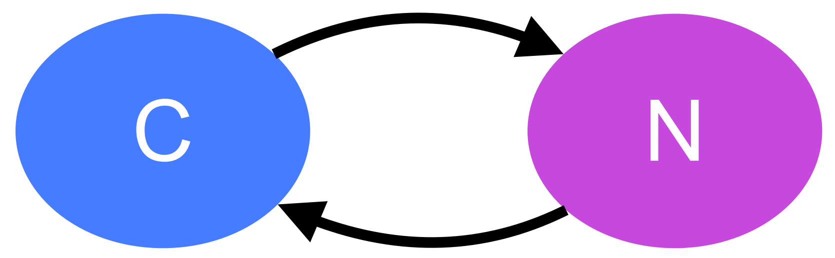
Common assumptions

- If p is **Markov and faithful** to G, then for any disjoint $\mathbf{A}, \mathbf{B}, \mathbf{C} \subseteq \mathbf{V}$:

$$\mathbf{A} \perp_G \mathbf{B} \,|\, \mathbf{C} \iff X_\mathbf{A} \perp\!\!\!\perp_p X_\mathbf{B} \,|\, X_\mathbf{C}$$

- **Causal sufficiency** - no latent confounders (common causes), no selection bias

- **Acyclicity** - the underlying graph is acyclic



  - Cycles + causal insufficiency: sigma separation, Joint Causal Inference

Causal discovery (causal structure learning)

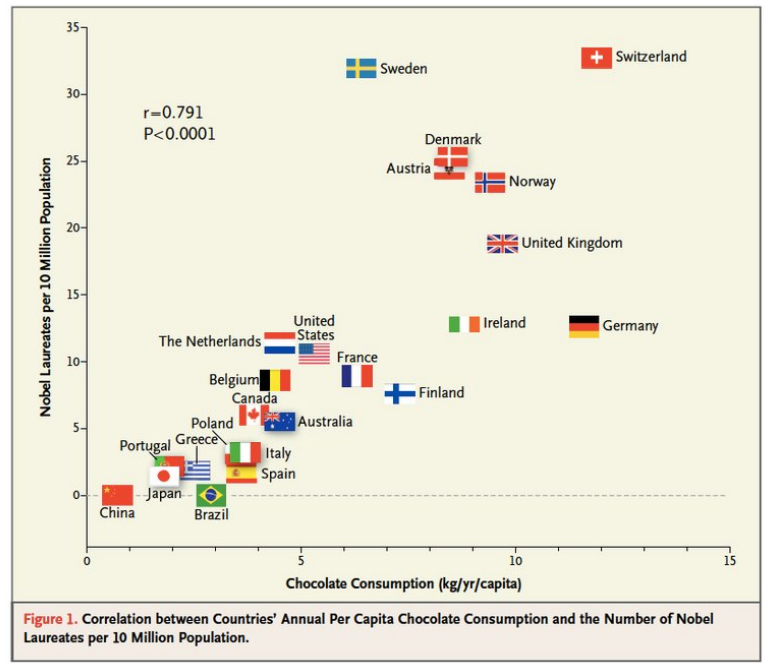


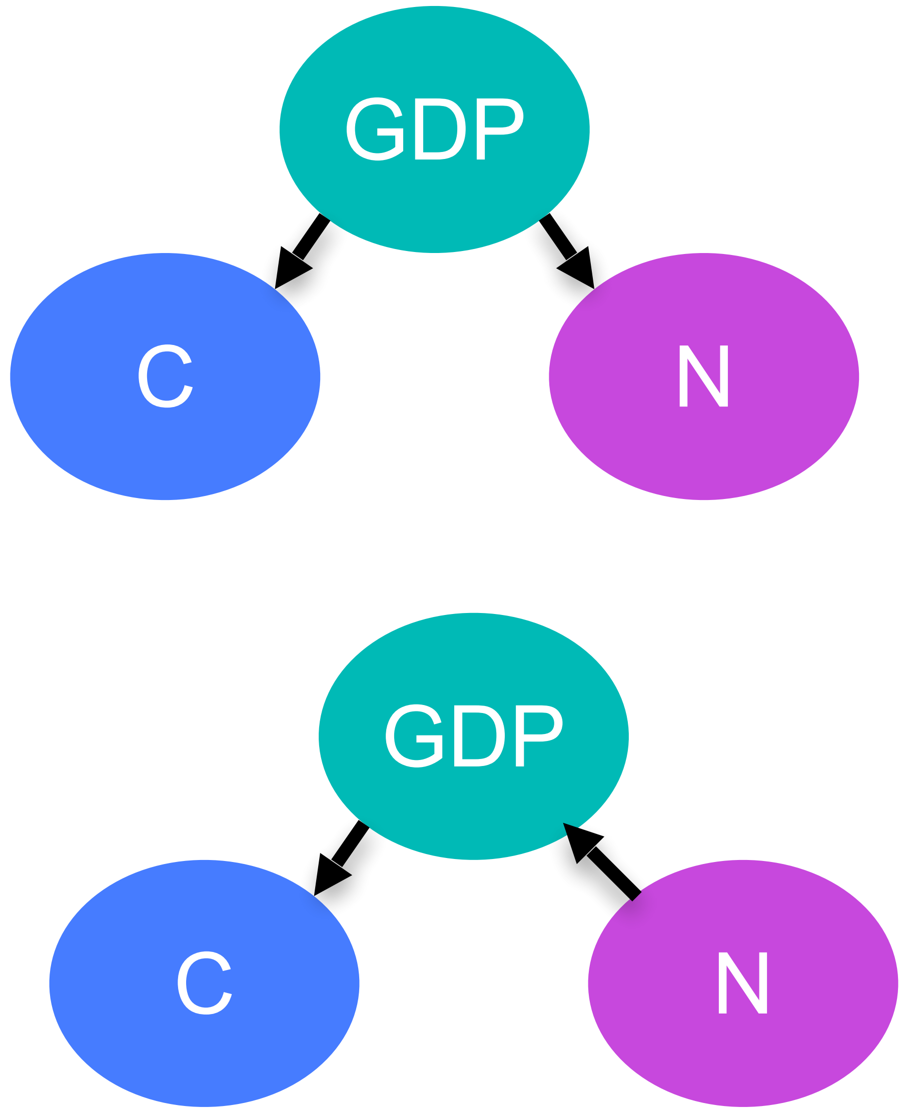
Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.

C	N	GDP
4.5	5	33k
12	30	86k
10	20	46k
….	…	…

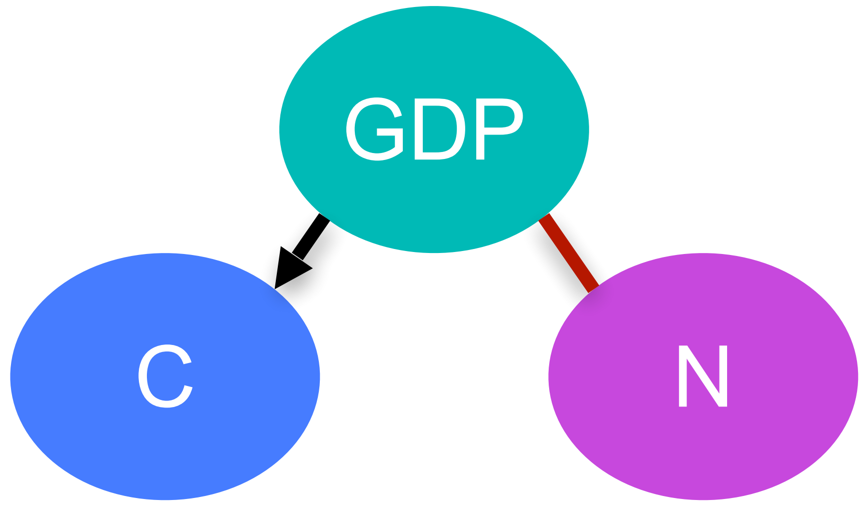
Observational data

$$C \nrightarrow GDP$$

[Optional] Background knowledge



Sets of graphs that fit the data and background knowledge



Summary graph

Causal discovery simplified overview

Constraint-based causal discovery

- Conditional independence tests
- Observational data
- Output: MEC
- SGS, PC, FCI

Score-based causal discovery

- Penalised likelihood
- Observational data
- Output: MEC
- GES, MMHC

Restricted models

- Nonlinear additive noise, Linear Non-Gaussianity
- Observational data
- Output: DAG
- RESIT, LINGAM

Interventional causal discovery / causal invariance

- Observational and Interventional data
- Output: parents of Y, I-MEC
- ICP, GIES, JCI

Causal discovery - this class

Constraint-based causal discovery

- Conditional independence tests
- Observational data
- Output: MEC
- SGS, PC

Score-based causal discovery

- Penalised likelihood
- Observational data
- Output: MEC
- GES, MMHC

Restricted models

- Nonlinear additive noise, Linear Non-Gaussianity
- Observational data
- Output: DAG
- RESIT, LINGAM

Interventional causal discovery / causal invariance

- Observational and Interventional data
- Output: parents of Y, I-MEC
- ICP, JCI

Recap: Global Markov Property & faithfulness

- If (G, p) is a Bayesian network with a DAG $G = (\mathbf{V}, \mathbf{E})$, i.e. **p factorizes according to G**, then for any disjoint $\mathbf{A}, \mathbf{B}, \mathbf{C} \subseteq \mathbf{V}$:

$$\mathbf{A} \perp_d \mathbf{B} \,|\, \mathbf{C} \implies X_{\mathbf{A}} \perp\!\!\!\perp X_{\mathbf{B}} \,|\, X_{\mathbf{C}}$$

If p has a density (e.g. no deterministic relations) [Lauritzen 1996]

- **d-separations** that can be read purely from a graph imply **conditional independences** in the random variables and data generated by the graph

Recap: Global Markov Property & faithfulness

- If (G, p) is a Bayesian network with a DAG $G = (\mathbf{V}, \mathbf{E})$, i.e. **p factorizes according to G**, then for any disjoint $\mathbf{A}, \mathbf{B}, \mathbf{C} \subseteq \mathbf{V}$:

$$\mathbf{A} \perp_d \mathbf{B} \,|\, \mathbf{C} \implies X_{\mathbf{A}} \perp\!\!\!\perp X_{\mathbf{B}} \,|\, X_{\mathbf{C}}$$

If p has a density (e.g. no deterministic relations) [Lauritzen 1996]

- **d-separations** that can be read purely from a graph imply **conditional independences** in the random variables and data generated by the graph

- The reverse implication is not true in general, but if it is, we say that **p is faithful to G:**

$$X_{\mathbf{A}} \perp\!\!\!\perp X_{\mathbf{B}} \,|\, X_{\mathbf{C}} \implies \mathbf{A} \perp_d \mathbf{B} \,|\, \mathbf{C}$$

Faithfulness violation example - cancelling paths

$$
\begin{cases}
X_1 = \epsilon_1 \\
X_2 = 3X_1 + \epsilon_2 \\
X_3 = X_2 - 3X_1 + \epsilon_3 \\
\epsilon_1, \epsilon_2, \epsilon_3 \sim N(0,1) \\
\epsilon_1 \perp\!\!\!\perp \epsilon_2, \epsilon_1 \perp\!\!\!\perp \epsilon_3, \epsilon_2 \perp\!\!\!\perp \epsilon_3
\end{cases}
$$

$$
\begin{tikzcd}
1 \arrow[r, "3"] \arrow[dr, "-3"'] & 2 \arrow[d, "1"] \\
& 3
\end{tikzcd}
$$

Faithfulness violation example - cancelling paths

$$
\begin{cases}
X_1 = \epsilon_1 \\
X_2 = 3X_1 + \epsilon_2 \\
X_3 = X_2 - 3X_1 + \epsilon_3 \\
\epsilon_1, \epsilon_2, \epsilon_3 \sim N(0,1) \\
\epsilon_1 \perp\!\!\!\perp \epsilon_2, \epsilon_1 \perp\!\!\!\perp \epsilon_3, \epsilon_2 \perp\!\!\!\perp \epsilon_3
\end{cases}
$$

$$
\begin{tikzcd}
1 \arrow[r, "3"] \arrow[dr, "-3"'] & 2 \arrow[d, "1"] \\
 & 3
\end{tikzcd}
$$

$$
\begin{cases}
X_1 = \epsilon_1 \\
X_2 = 3\epsilon_1 + \epsilon_2 \\
X_3 = 3\epsilon_1 + \epsilon_2 - 3\epsilon_1 + \epsilon_3 \\
\epsilon_1, \epsilon_2, \epsilon_3 \sim N(0,1) \\
\epsilon_1 \perp\!\!\!\perp \epsilon_2, \epsilon_1 \perp\!\!\!\perp \epsilon_3, \epsilon_2 \perp\!\!\!\perp \epsilon_3
\end{cases}
$$

Faithfulness violation example - cancelling paths

$$
\begin{cases}
X_1 = \epsilon_1 \\
X_2 = 3X_1 + \epsilon_2 \\
X_3 = X_2 - 3X_1 + \epsilon_3 \\
\epsilon_1, \epsilon_2, \epsilon_3 \sim N(0,1) \\
\epsilon_1 \perp\!\!\!\perp \epsilon_2, \epsilon_1 \perp\!\!\!\perp \epsilon_3, \epsilon_2 \perp\!\!\!\perp \epsilon_3
\end{cases}
$$

$$
\begin{tikzcd}
1 \arrow[r, "3"] \arrow[dr, "-3"'] & 2 \arrow[d, "1"] \\
 & 3
\end{tikzcd}
$$

$$
\begin{cases}
X_1 = \epsilon_1 \\
X_2 = 3\epsilon_1 + \epsilon_2 \\
X_3 = \cancel{3\epsilon_1} + \epsilon_2 - \cancel{3\epsilon_1} + \epsilon_3 \\
\epsilon_1, \epsilon_2, \epsilon_3 \sim N(0,1) \\
\epsilon_1 \perp\!\!\!\perp \epsilon_2, \epsilon_1 \perp\!\!\!\perp \epsilon_3, \epsilon_2 \perp\!\!\!\perp \epsilon_3
\end{cases}
$$

Faithfulness violation example - cancelling paths

$$\begin{cases} X_1 = \epsilon_1 \\ X_2 = 3X_1 + \epsilon_2 \\ X_3 = X_2 - 3X_1 + \epsilon_3 \\ \epsilon_1, \epsilon_2, \epsilon_3 \sim N(0,1) \\ \epsilon_1 \perp\!\!\!\perp \epsilon_2, \epsilon_1 \perp\!\!\!\perp \epsilon_3, \epsilon_2 \perp\!\!\!\perp \epsilon_3 \end{cases}$$

$$\begin{tikzcd} 1 \arrow[r, "3"] \arrow[dr, "-3"'] & 2 \arrow[d, "1"] \\ & 3 \end{tikzcd}$$

$$\begin{cases} X_1 = \epsilon_1 \\ X_2 = 3\epsilon_1 + \epsilon_2 \\ X_3 = \epsilon_2 + \epsilon_3 \\ \epsilon_1, \epsilon_2, \epsilon_3 \sim N(0,1) \\ \epsilon_1 \perp\!\!\!\perp \epsilon_2, \epsilon_1 \perp\!\!\!\perp \epsilon_3, \epsilon_2 \perp\!\!\!\perp \epsilon_3 \end{cases}$$

Faithfulness violation example - cancelling paths

$$
\begin{cases}
X_1 = \epsilon_1 \\
X_2 = 3X_1 + \epsilon_2 \\
X_3 = X_2 - 3X_1 + \epsilon_3 \\
\epsilon_1, \epsilon_2, \epsilon_3 \sim N(0,1) \\
\epsilon_1 \perp\!\!\!\perp \epsilon_2, \epsilon_1 \perp\!\!\!\perp \epsilon_3, \epsilon_2 \perp\!\!\!\perp \epsilon_3
\end{cases}
$$

$$
\begin{tikzcd}
1 \arrow[r, "3"] \arrow[dr, "-3"'] & 2 \arrow[d, "1"] \\
& 3
\end{tikzcd}
$$

$$
\begin{cases}
X_1 = \epsilon_1 \\
X_2 = 3\epsilon_1 + \epsilon_2 \\
X_3 = \epsilon_2 + \epsilon_3 \\
\epsilon_1, \epsilon_2, \epsilon_3 \sim N(0,1) \\
\epsilon_1 \perp\!\!\!\perp \epsilon_2, \epsilon_1 \perp\!\!\!\perp \epsilon_3, , \epsilon_2 \perp\!\!\!\perp \epsilon_3
\end{cases}
$$

$X_1 \not\perp\!\!\!\perp X_2$

Faithfulness violation example - cancelling paths

$$
\begin{cases}
X_1 = \epsilon_1 \\
X_2 = 3X_1 + \epsilon_2 \\
X_3 = X_2 - 3X_1 + \epsilon_3 \\
\epsilon_1, \epsilon_2, \epsilon_3 \sim N(0,1) \\
\epsilon_1 \perp\!\!\!\perp \epsilon_2, \epsilon_1 \perp\!\!\!\perp \epsilon_3, \epsilon_2 \perp\!\!\!\perp \epsilon_3
\end{cases}
$$

$$
\begin{tikzcd}
1 \arrow[r, "3"] \arrow[dr, "-3"'] & 2 \arrow[d, "1"] \\
 & 3
\end{tikzcd}
$$

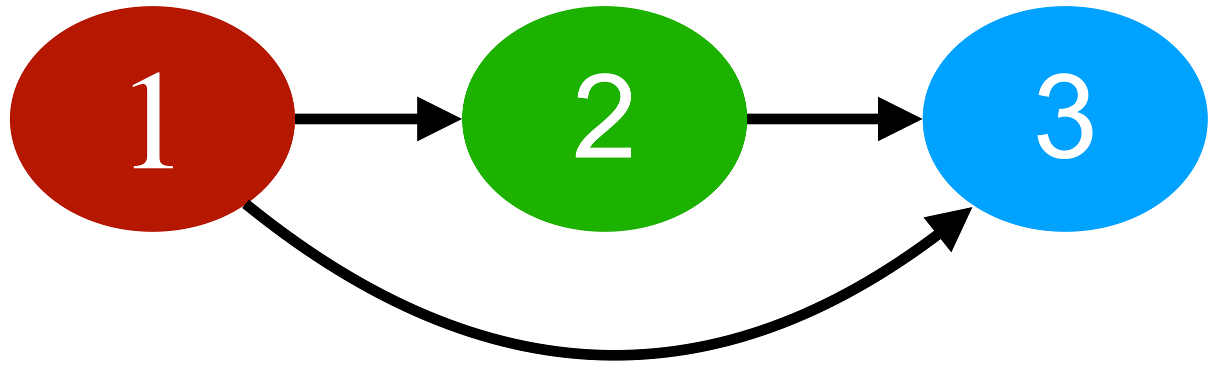
$$
\begin{cases}
X_1 = \epsilon_1 \\
X_2 = 3\epsilon_1 + \epsilon_2 \\
X_3 = \epsilon_2 + \epsilon_3 \\
\epsilon_1, \epsilon_2, \epsilon_3 \sim N(0,1) \\
\epsilon_1 \perp\!\!\!\perp \epsilon_2, \epsilon_1 \perp\!\!\!\perp \epsilon_3, , \epsilon_2 \perp\!\!\!\perp \epsilon_3
\end{cases}
$$

$X_1 \not\!\perp\!\!\!\perp X_2$ $X_2 \not\!\perp\!\!\!\perp X_3$ $X_1 \perp\!\!\!\perp X_3$

Faithfulness violation example - cancelling paths

$$\begin{cases} X_1 = \epsilon_1 \\ X_2 = 3X_1 + \epsilon_2 \\ X_3 = X_2 - 3X_1 + \epsilon_3 \\ \epsilon_1, \epsilon_2, \epsilon_3 \sim N(0,1) \\ \epsilon_1 \perp\!\!\!\perp \epsilon_2, \epsilon_1 \perp\!\!\!\perp \epsilon_3, \epsilon_2 \perp\!\!\!\perp \epsilon_3 \end{cases}$$

$X_1 \not\!\perp\!\!\!\perp X_2 \quad X_2 \not\!\perp\!\!\!\perp X_3 \quad X_1 \perp\!\!\!\perp X_3$



$X_1 \not\!\perp_d X_2 \quad X_2 \not\!\perp_d X_3 \quad X_1 \not\!\perp_d X_3$

Faithfulness violation: conditional independence does not imply d-separation

Perfect maps

- If p is Markov and faithful to G, we say that G **is a perfect map of** p**.**
  Then, for any disjoint $\mathbf{A}, \mathbf{B}, \mathbf{C} \subseteq \mathbf{V}$:

$$\mathbf{A} \perp_d \mathbf{B} \,|\, \mathbf{C} \iff X_{\mathbf{A}} \perp\!\!\!\perp X_{\mathbf{B}} \,|\, X_{\mathbf{C}}$$

- This correspondence is the basis of learning causal graphs from data

- **In a nutshell:** we perform a set of conditional independence tests on the data and use them to constrain the possible graphs using d-separation

Perfect maps - existence

- Not every distribution p has a perfect map

$$
\begin{cases}
X_1 = \epsilon_1 \\
X_2 = X_1 & \boxed{\text{Deterministic function}} \\
X_3 = X_2 + \epsilon_3 \\
\epsilon_1, \epsilon_3 \sim N(0,1) \\
\epsilon_1 \perp\!\!\!\perp \epsilon_3
\end{cases}
$$

Perfect maps - existence

- Not every distribution p has a perfect map

$$\begin{cases} X_1 = \epsilon_1 \\ X_2 = X_1 \\ X_3 = X_2 + \epsilon_3 \\ \epsilon_1, \epsilon_3 \sim N(0,1) \\ \epsilon_1 \perp\!\!\!\perp \epsilon_3 \end{cases}$$

Deterministic function

$$\begin{cases} X_1 = \epsilon_1 \\ X_2 = \epsilon_1 \\ X_3 = \epsilon_1 + \epsilon_3 \\ \epsilon_1, \epsilon_3 \sim N(0,1) \\ \epsilon_1 \perp\!\!\!\perp \epsilon_3 \end{cases}$$

$X_1 \not\perp\!\!\!\perp X_2$

$X_2 \not\perp\!\!\!\perp X_3$

$X_1 \not\perp\!\!\!\perp X_3$

$X_1 \perp\!\!\!\perp X_3 \mid X_2$

Perfect maps - existence

- Not every distribution p has a perfect map

$$\begin{cases} X_1 = \epsilon_1 \\ X_2 = X_1 \\ X_3 = X_2 + \epsilon_3 \\ \epsilon_1, \epsilon_3 \sim N(0,1) \\ \epsilon_1 \perp\!\!\!\perp \epsilon_3 \end{cases}$$

Deterministic function

$$\begin{cases} X_1 = \epsilon_1 \\ X_2 = \epsilon_1 \\ X_3 = \epsilon_1 + \epsilon_3 \\ \epsilon_1, \epsilon_3 \sim N(0,1) \\ \epsilon_1 \perp\!\!\!\perp \epsilon_3 \end{cases}$$

$X_1 \not\perp\!\!\!\perp X_2$

$X_2 \not\perp\!\!\!\perp X_3$

$X_1 \not\perp\!\!\!\perp X_3$

$X_1 \perp\!\!\!\perp X_3 \mid X_2$

$p(\epsilon_1 \mid \epsilon_1, \epsilon_1 + \epsilon_3) = p(\epsilon_1 \mid \epsilon_1)$

$X_2 \perp\!\!\!\perp X_3 \mid X_1$

Perfect maps - existence

- Not every distribution p has a perfect map

$$1 \rightarrow 2 \rightarrow 3$$

$$\begin{cases} X_1 = \epsilon_1 \\ X_2 = X_1 \\ X_3 = X_2 + \epsilon_3 \\ \epsilon_1, \epsilon_3 \sim N(0,1) \\ \epsilon_1 \perp\!\!\!\perp \epsilon_3 \end{cases}$$

Deterministic function

$$\begin{cases} X_1 = \epsilon_1 \\ X_2 = \epsilon_1 \\ X_3 = \epsilon_1 + \epsilon_3 \\ \epsilon_1, \epsilon_3 \sim N(0,1) \\ \epsilon_1 \perp\!\!\!\perp \epsilon_3 \end{cases}$$

$X_1 \not\perp\!\!\!\perp X_2$

$X_2 \not\perp\!\!\!\perp X_3$

$X_1 \not\perp\!\!\!\perp X_3$

$X_1 \perp\!\!\!\perp X_3 \mid X_2$

$X_2 \perp_d X_3 \mid X_1 ?$

$X_2 \perp\!\!\!\perp X_3 \mid X_1$

Perfect maps - existence

- Not every distribution p has a perfect map

$$1 \to 2 \to 3$$

$$\begin{cases} X_1 = \epsilon_1 \\ X_2 = X_1 \\ X_3 = X_2 + \epsilon_3 \\ \epsilon_1, \epsilon_3 \sim N(0,1) \\ \epsilon_1 \perp\!\!\!\perp \epsilon_3 \end{cases}$$

Deterministic function

$$\begin{cases} X_1 = \epsilon_1 \\ X_2 = \epsilon_1 \\ X_3 = \epsilon_1 + \epsilon_3 \\ \epsilon_1, \epsilon_3 \sim N(0,1) \\ \epsilon_1 \perp\!\!\!\perp \epsilon_3 \end{cases}$$

$X_1 \not\!\perp\!\!\!\perp X_2$

$X_2 \not\!\perp\!\!\!\perp X_3$

$X_1 \not\!\perp\!\!\!\perp X_3$

$X_1 \perp\!\!\!\perp X_3 \mid X_2$

$X_2 \not\!\perp\!\!\!\perp_d X_3 \mid X_1$ $X_2 \perp\!\!\!\perp X_3 \mid X_1$

Perfect maps - existence

- Not every distribution p has a perfect map

There exists no Bayesian network that can represent these conditional in/dependences as d-separations perfectly!

$$1 \to 2 \to 3$$

$$\begin{cases} X_1 = \epsilon_1 \\ X_2 = X_1 \\ X_3 = X_2 + \epsilon_3 \\ \epsilon_1, \epsilon_3 \sim N(0,1) \\ \epsilon_1 \perp\!\!\!\perp \epsilon_3 \end{cases}$$

Deterministic function

$$\begin{cases} X_1 = \epsilon_1 \\ X_2 = \epsilon_1 \\ X_3 = \epsilon_1 + \epsilon_3 \\ \epsilon_1, \epsilon_3 \sim N(0,1) \\ \epsilon_1 \perp\!\!\!\perp \epsilon_3 \end{cases}$$

$X_1 \not\!\perp\!\!\!\perp X_2$

$X_2 \not\!\perp\!\!\!\perp X_3$

$X_1 \not\!\perp\!\!\!\perp X_3$

$X_1 \perp\!\!\!\perp X_3 \mid X_2$

$X_2 \not\!\perp_d X_3 \mid X_1$

$X_2 \perp\!\!\!\perp X_3 \mid X_1$

Markov equivalence class (MEC)

- If p is Markov and faithful to G, we say that G **is a perfect map of** p**.** Then, for any disjoint $\mathbf{A}, \mathbf{B}, \mathbf{C} \subseteq \mathbf{V}$:

$$\mathbf{A} \perp_d \mathbf{B} \,|\, \mathbf{C} \iff X_{\mathbf{A}} \perp\!\!\!\perp X_{\mathbf{B}} \,|\, X_{\mathbf{C}}$$

- In general there are multiple DAGs that can describe the same d-separations (and independences)

- We call these DAGs **Markov equivalent** and we **cannot distinguish them from observational data alone** (or without further assumptions)

Question: Markov equivalence class

$X \not\perp\!\!\!\perp Y \quad X \perp\!\!\!\perp Y \mid Z$

$X - Z - Y$

Which graphs on X, Y, Z are perfect maps of these conditional independences?

Hint: we can start by orienting this undirected graph

Question: Markov equivalence class

$X \not\perp\!\!\!\perp Y \quad X \perp\!\!\!\perp Y \mid Z$

Which graphs on X, Y, Z are perfect maps of these conditional independences?

$X - Z - Y$

$X \to Z \to Y$

$X \not\perp_d Y$ ✓

$X \perp_d Y \mid Z$ ✓

Question: Markov equivalence class

$X \not\perp\!\!\!\perp Y \quad X \perp\!\!\!\perp Y \mid Z$

Which graphs on X, Y, Z are perfect maps of these conditional independences?

$X - Z - Y$

$X \rightarrow Z \rightarrow Y$

$X \not\perp_d Y$ ✓

$X \perp_d Y \mid Z$ ✓

$X \leftarrow Z \leftarrow Y$

$X \not\perp_d Y$ ✓

$X \perp_d Y \mid Z)$ ✓

Question: Markov equivalence class

Which graphs on X, Y, Z are perfect maps of these conditional independences?

$X \not\perp\!\!\!\perp Y \quad X \perp\!\!\!\perp Y \mid Z$

$X - Z - Y$

$X \rightarrow Z \rightarrow Y$

$X \not\perp_d Y$ ✓

$X \perp_d Y \mid Z$ ✓

$X \leftarrow Z \rightarrow Y$

$X \not\perp_d Y$ ✓

$X \perp_d Y \mid Z$ ✓

$X \leftarrow Z \leftarrow Y$

$X \not\perp_d Y$ ✓

$X \perp_d Y \mid Z)$ ✓

Question: Markov equivalence class

Which graphs on X, Y, Z are perfect maps of these conditional independences?

$X \not\perp\!\!\!\perp Y \qquad X \perp\!\!\!\perp Y \mid Z$

$X - Z - Y$

$X \rightarrow Z \rightarrow Y \qquad X \not\perp_d Y$ ✓ $\quad X \perp_d Y \mid Z$ ✓

$X \leftarrow Z \rightarrow Y \qquad X \not\perp_d Y$ ✓ $\quad X \perp_d Y \mid Z$ ✓

$X \leftarrow Z \leftarrow Y \qquad X \not\perp_d Y$ ✓ $\quad X \perp_d Y \mid Z)$ ✓

$X \rightarrow Z \leftarrow Y \qquad X \perp_d Y$ × $\quad X \not\perp_d Y \mid Z$ ×

Question: Markov equivalence class

$X \not\perp\!\!\!\perp Y \quad X \perp\!\!\!\perp Y \mid Z$

Which graphs on X, Y, Z are perfect maps of these conditional independences?

$X \rightarrow Z \rightarrow Y$

$X \not\perp_d Y$ ✓
$X \perp_d Y \mid Z$ ✓

$X \leftarrow Z \rightarrow Y$

$X \not\perp_d Y$ ✓
$X \perp_d Y \mid Z$ ✓

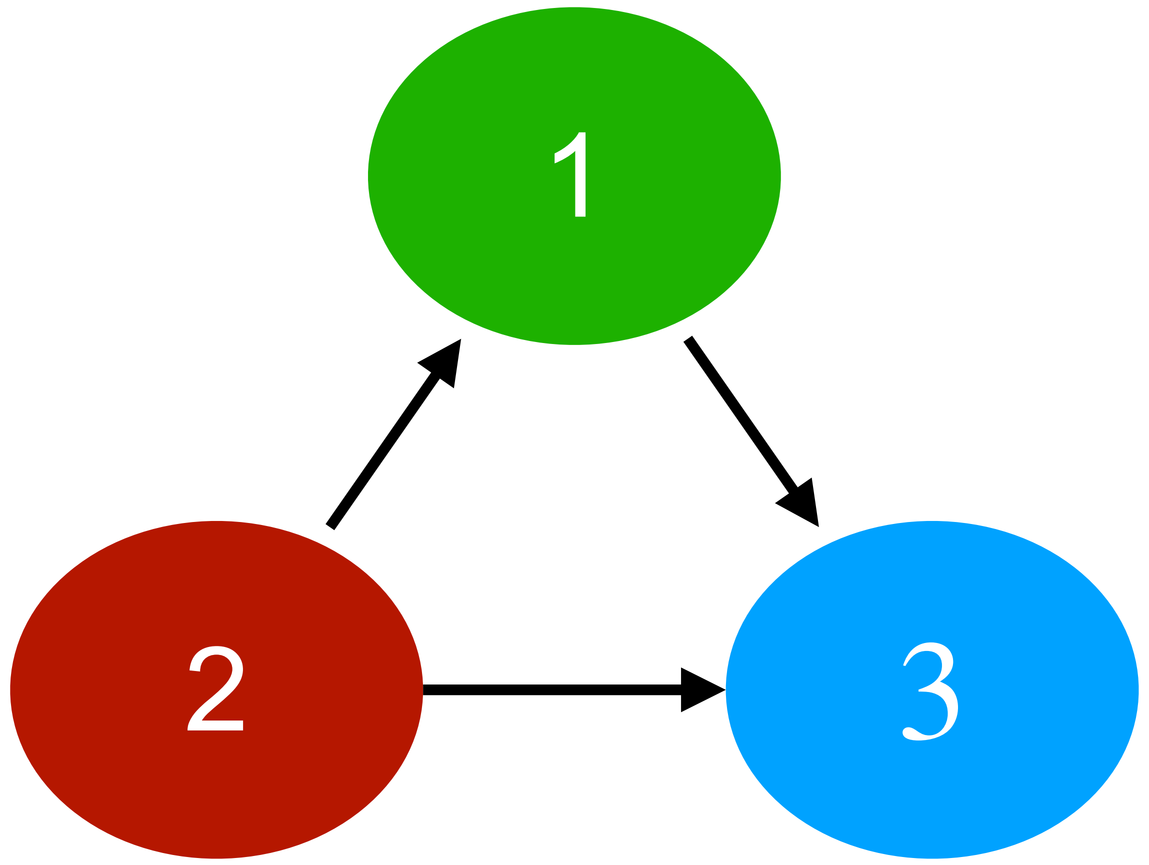
$X \leftarrow Z \leftarrow Y$

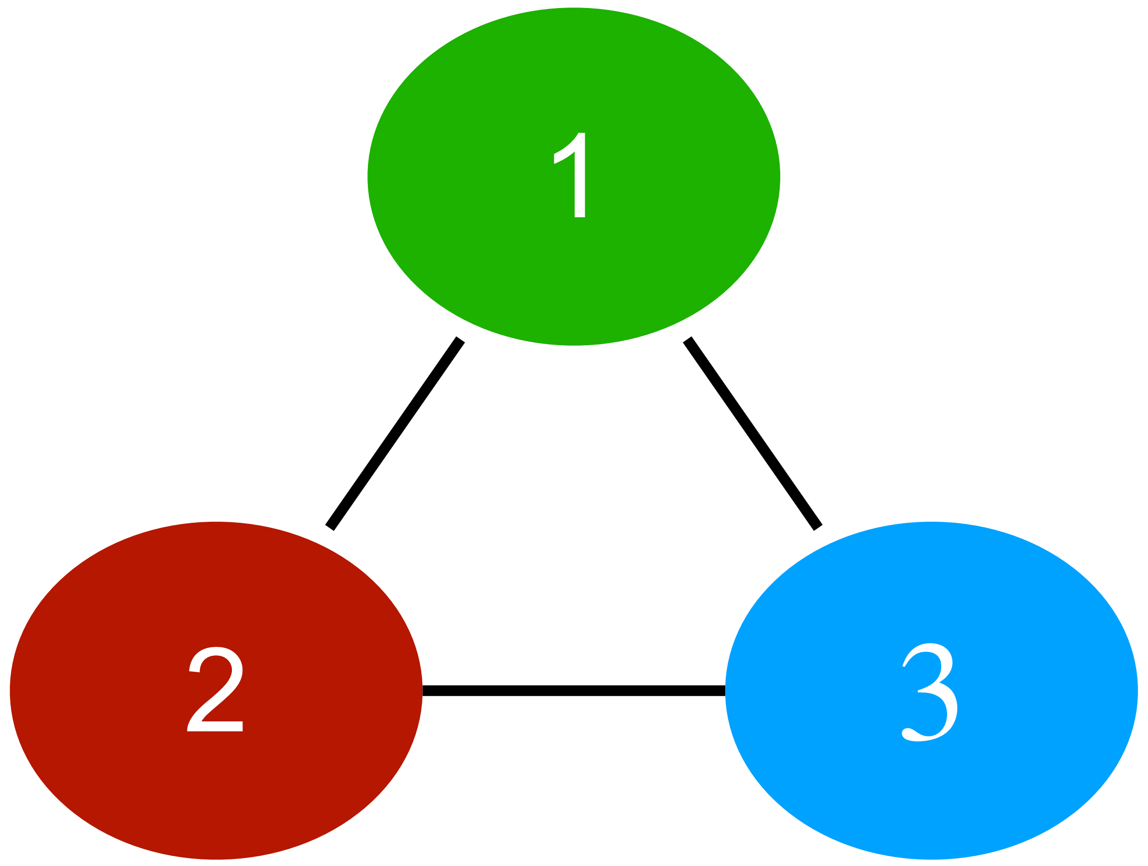
$X \not\perp_d Y$ ✓
$X \perp_d Y \mid Z)$ ✓

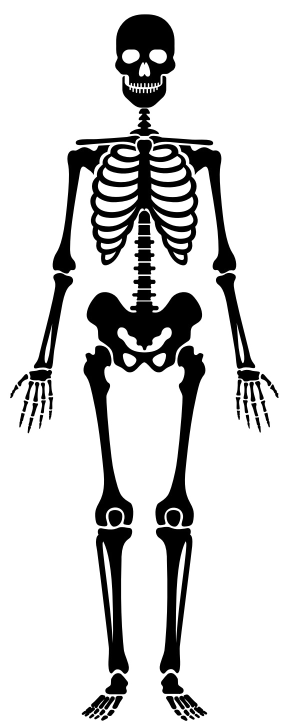
These three graphs represent a Markov Equivalence Class

Graph terminology: skeletons

- The skeleton of a DAG $G = (\mathbf{V}, \mathbf{E})$ is the undirected graph $U = (\mathbf{V}, \mathbf{E}')$ that has **an undirected edge** $(i, j) \in \mathbf{E}'$ for every **directed edge** $i \rightarrow j \in \mathbf{E}$ (and no other edges)



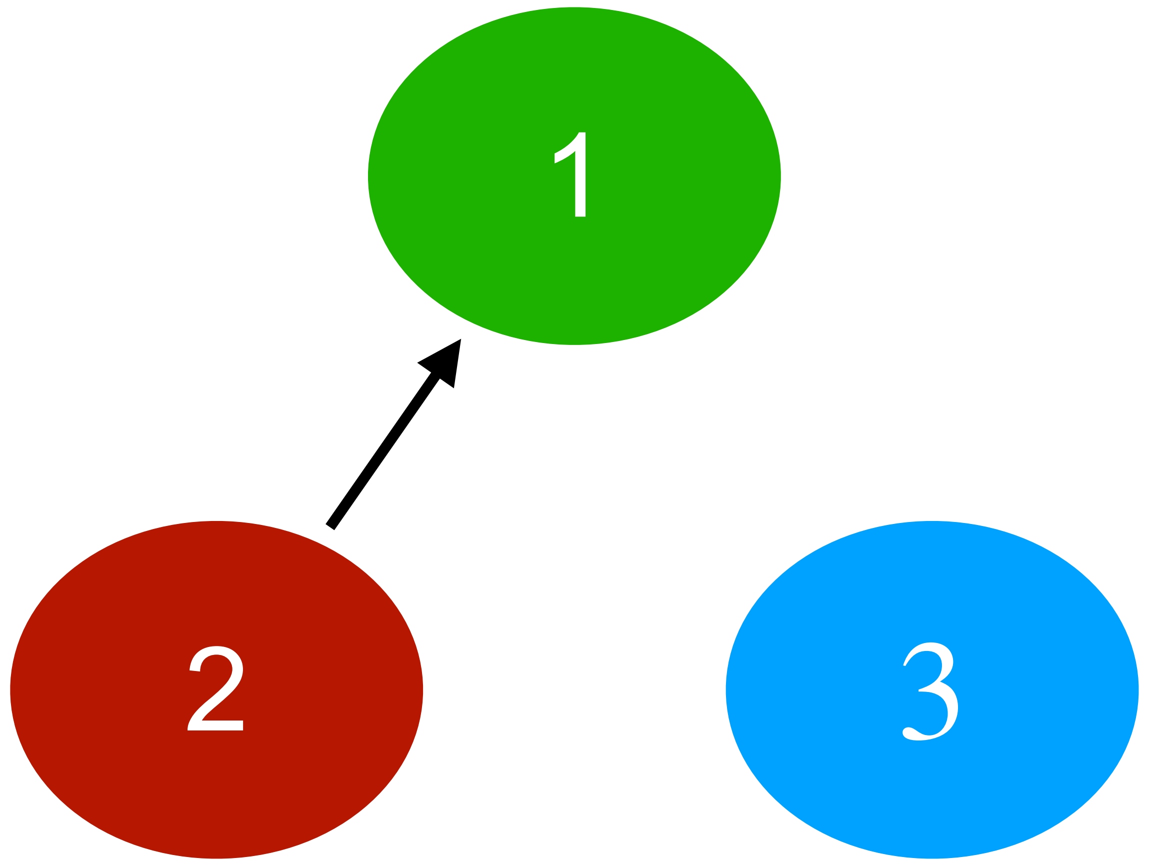




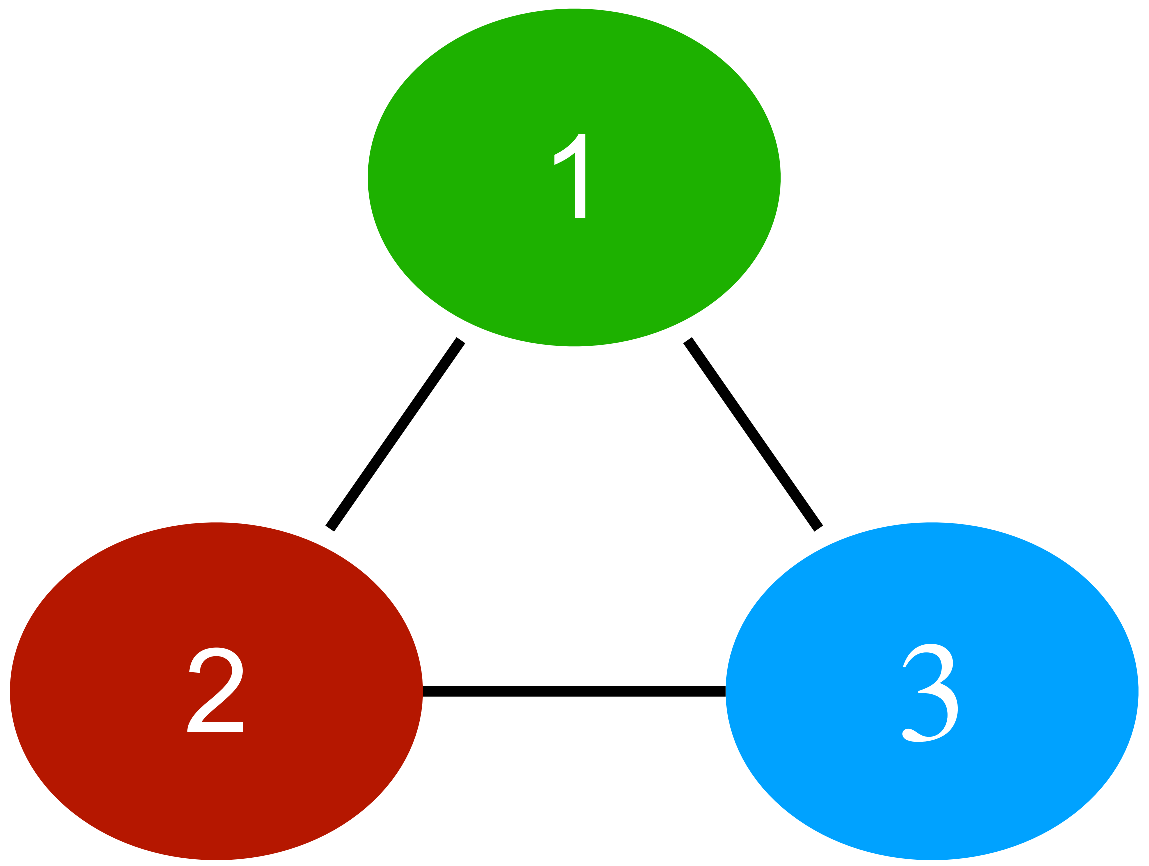
In other words a graph that is the same, but without the arrowheads

Skeleton exercise

- The skeleton of a DAG $G = (\mathbf{V}, \mathbf{E})$ is the undirected graph $U = (\mathbf{V}, \mathbf{E}')$ that has **an undirected edge** $(i, j) \in \mathbf{E}'$ for every **directed edge** $i \rightarrow j \in \mathbf{E}$ and no other edges



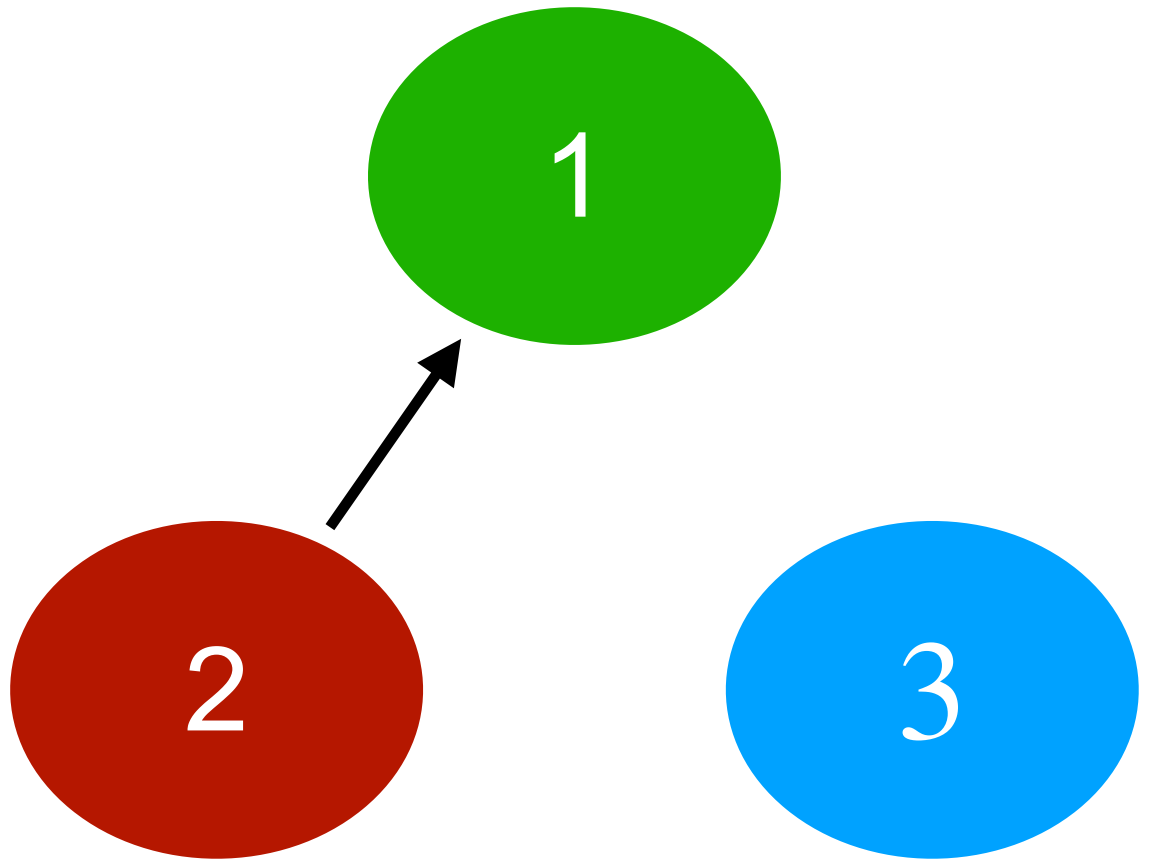
G



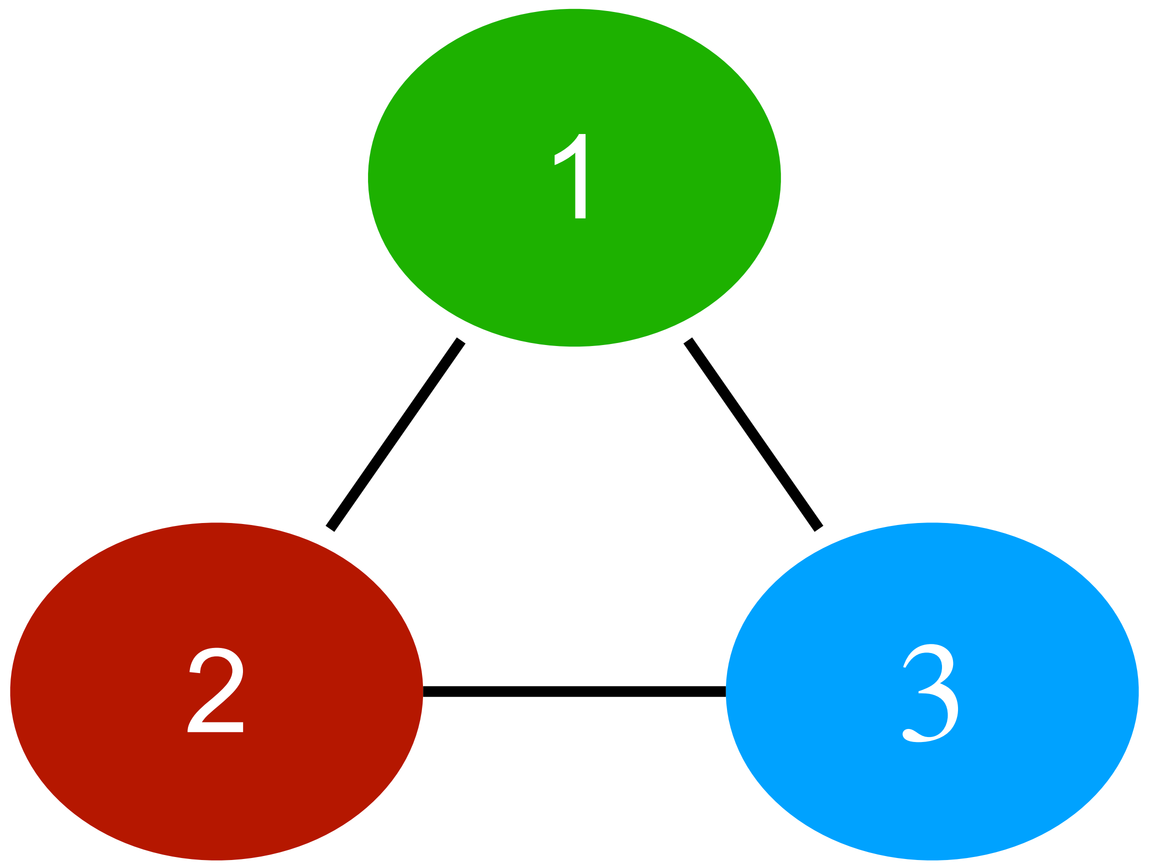
Is this the skeleton of G?

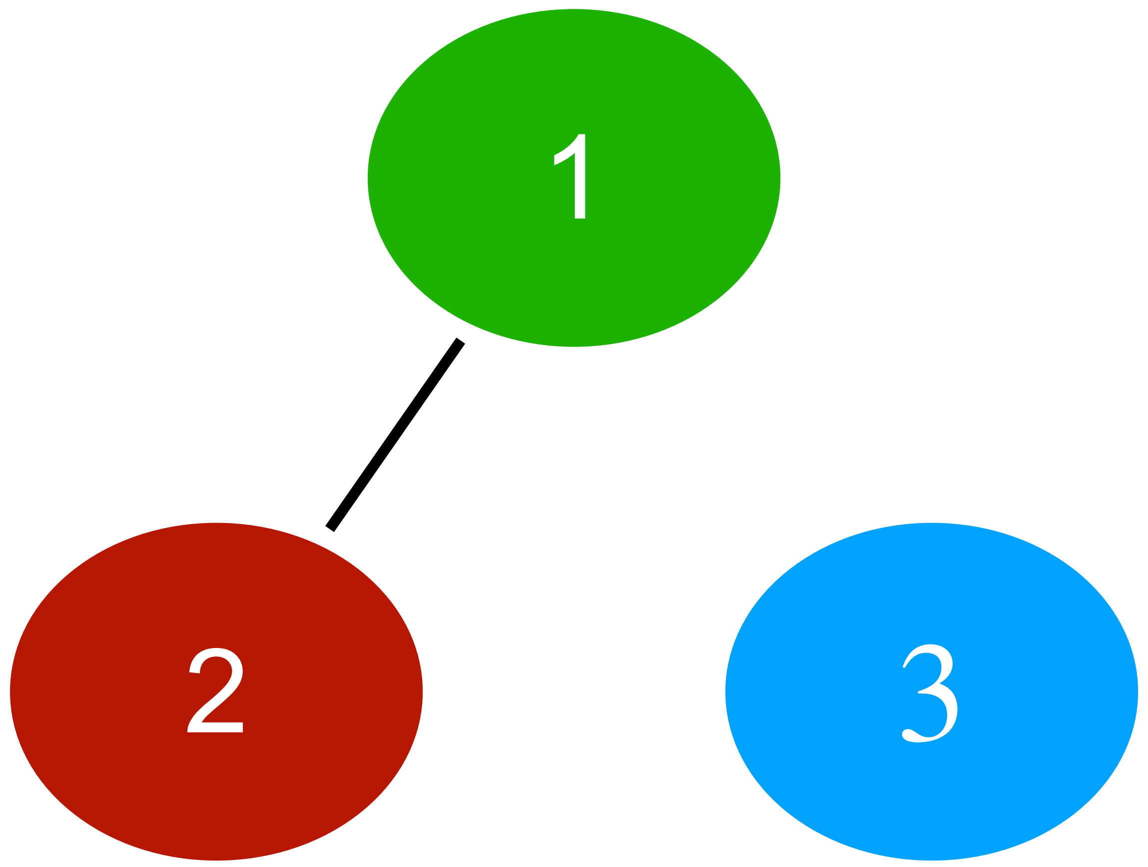
Skeleton exercise

- The skeleton of a DAG $G = (\mathbf{V}, \mathbf{E})$ is the undirected graph $U = (\mathbf{V}, \mathbf{E}')$ that has **an undirected edge** $(i, j) \in \mathbf{E}'$ for every **directed edge** $i \rightarrow j \in \mathbf{E}$ and no other edges



G

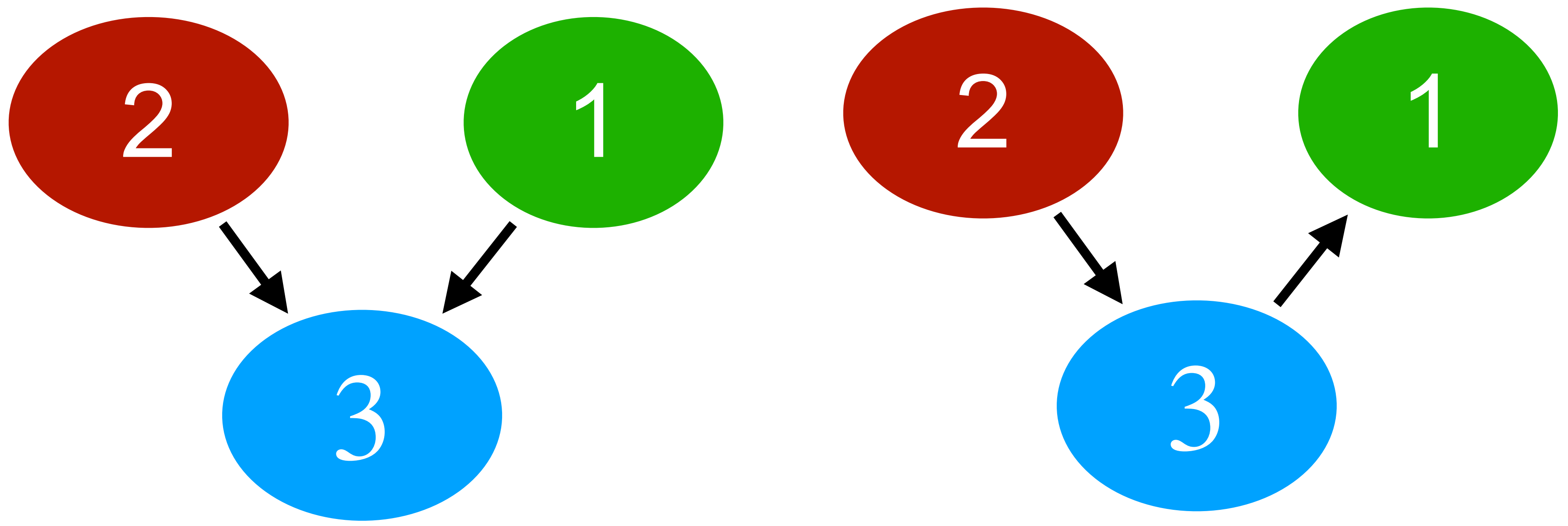




Is this the skeleton of G?

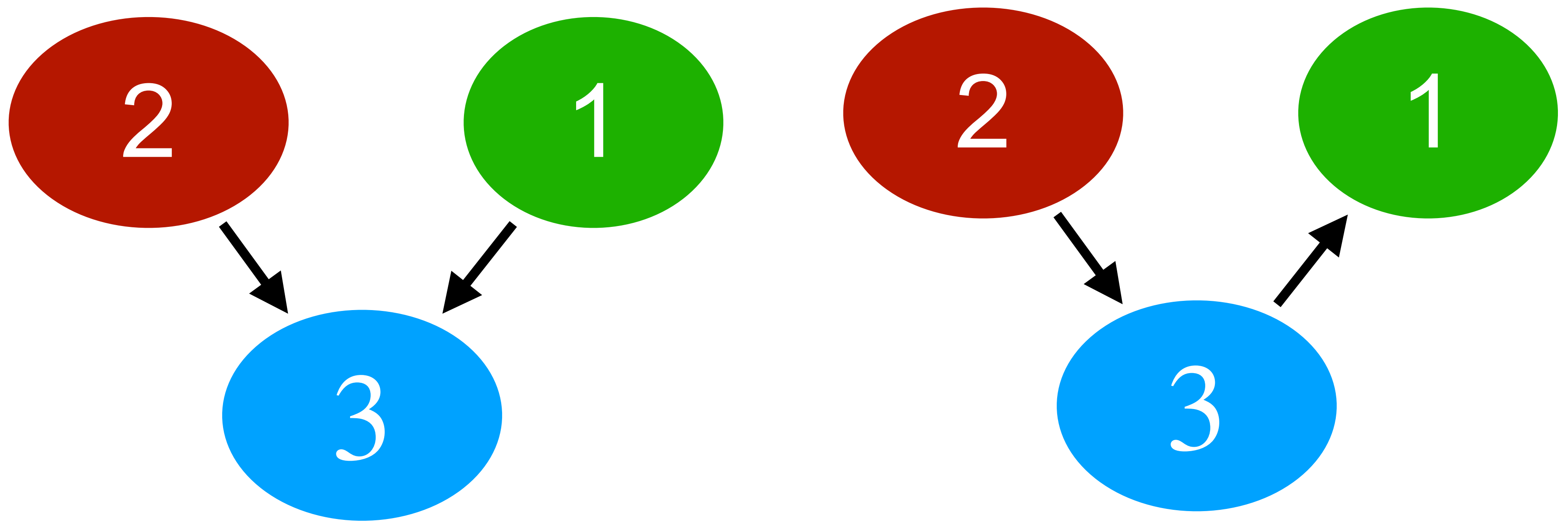
More graph terminology: adjacent

- Nodes i and j in a DAG G are **adjacent/neighbours** if $i \rightarrow j$ or $j \rightarrow i$ in G



More graph terminology: adjacent

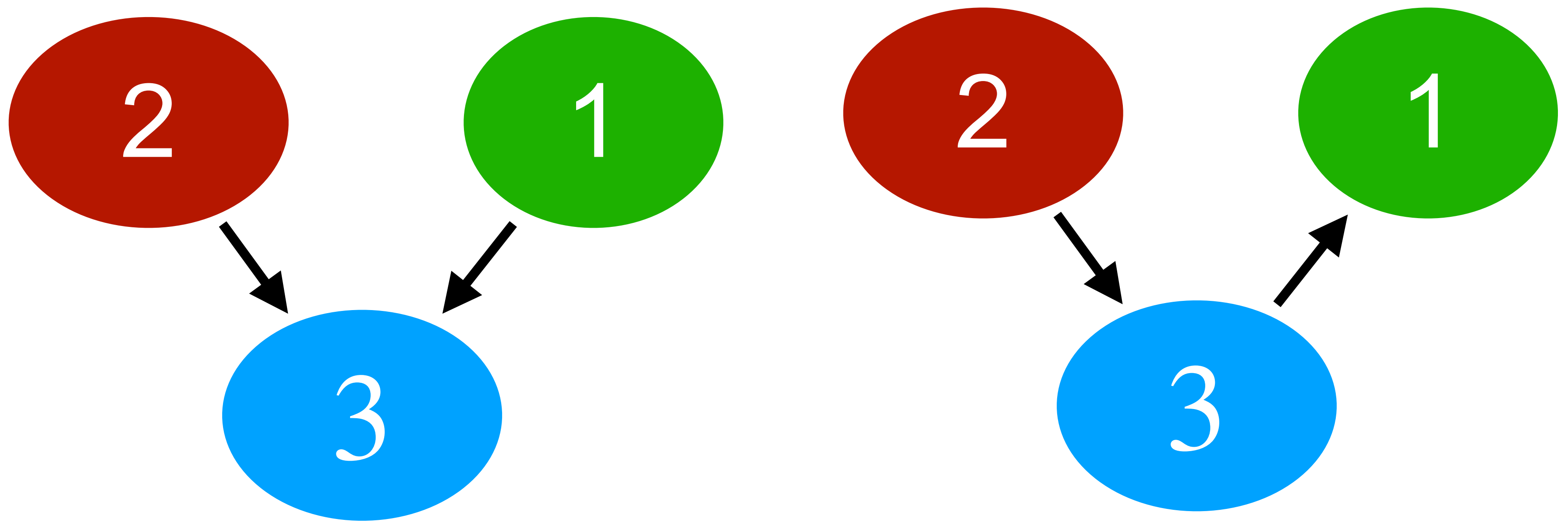
- Nodes i and j in a DAG G are **adjacent/neighbours** if $i \to j$ or $j \to i$ in G
  - i.e., they are connected by an undirected edge in the skeleton of G
  - We denote adjacency with $i - j$, while $i \not- j$ means non-adjacent



More graph terminology: adjacent

- Nodes i and j in a DAG G are **adjacent/neighbours** if $i \to j$ or $j \to i$ in G
  - i.e., they are connected by an undirected edge in the skeleton of G
  - We denote adjacency with $i - j$, while $i \not- j$ means non-adjacent

Is 2 adjacent to 3?

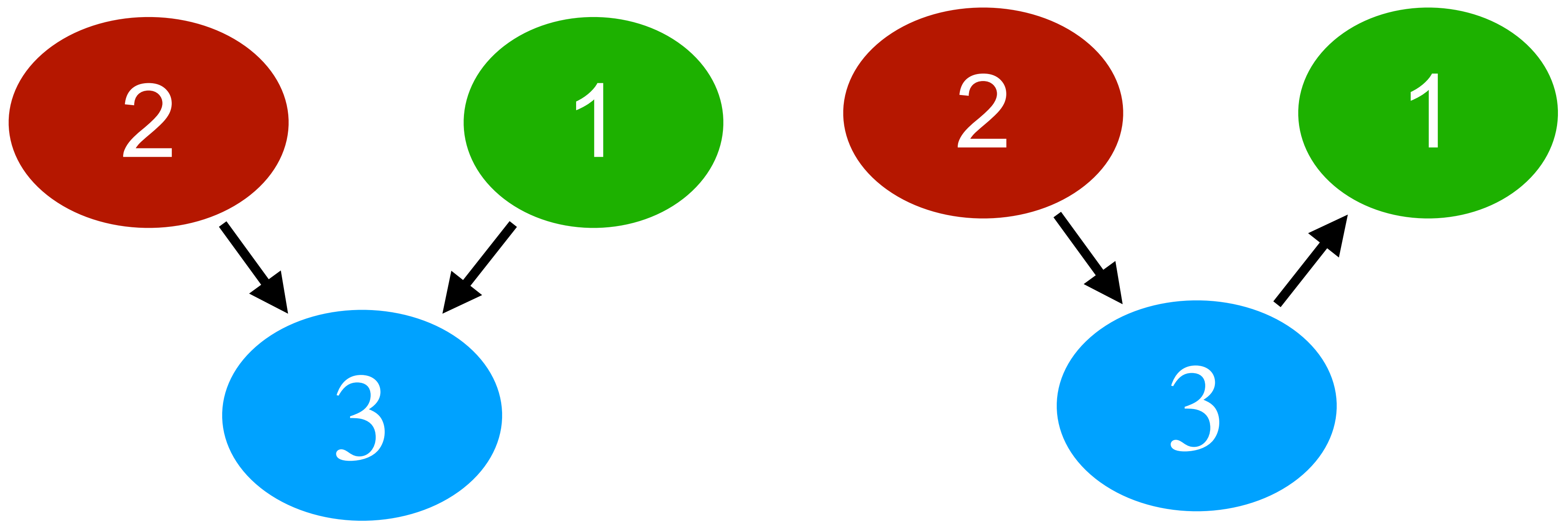


More graph terminology: adjacent

- Nodes i and j in a DAG G are **adjacent/neighbours** if $i \to j$ or $j \to i$ in G
  - I.e., they are connected by an undirected edge in the skeleton of G
  - We denote adjacency with $i - j$, while $i \not - j$ means non-adjacent

Is 2 adjacent to 3?

Is 2 adjacent to 1?



More graph terminology: v-structures/immoralities

- A triple of nodes (i, j, k) in a DAG G is a **v-structure (unshielded collider)** if

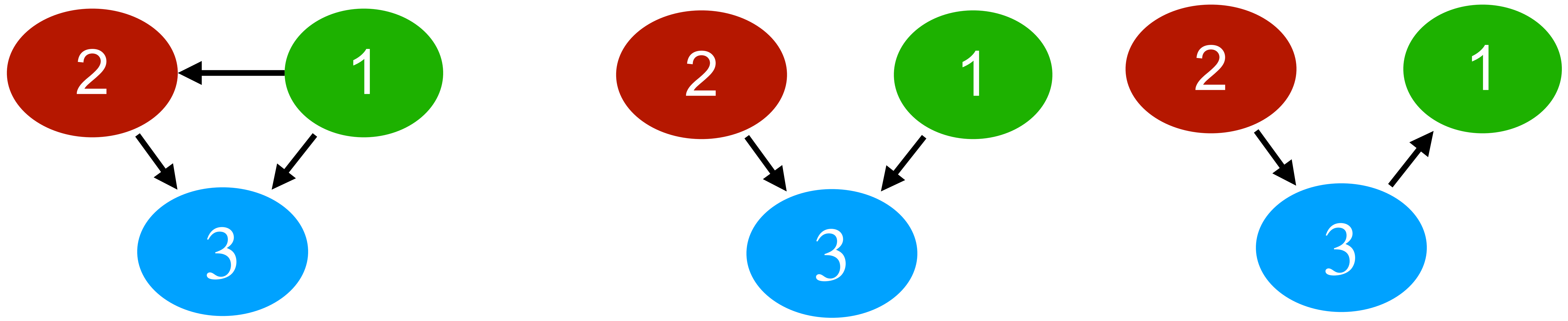
  $i \rightarrow j \leftarrow k$ in G and i **is not adjacent to** k

An edge from i to k would be called a shield

It's also called an immorality, because the two parents i and j who have a common child k are not "married" (adjacent)

More graph terminology: v-structures/immoralities

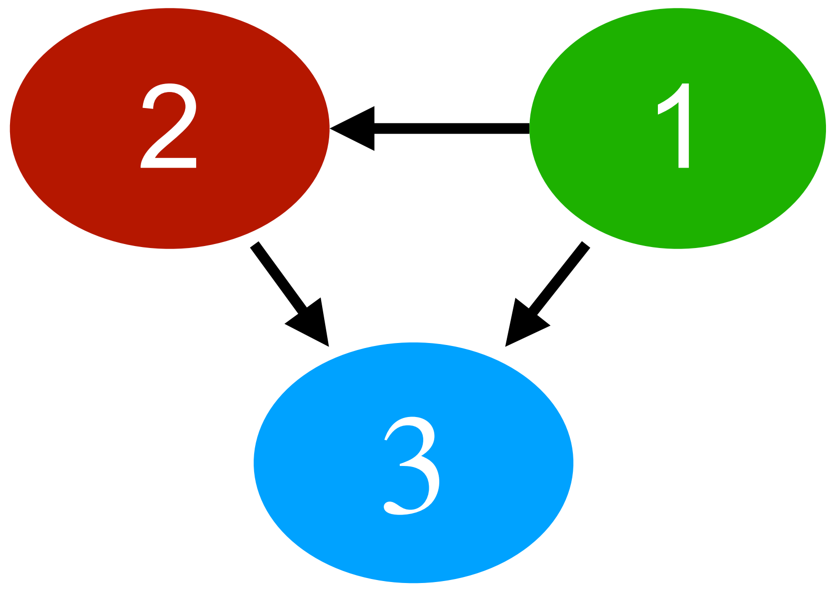
- A triple of nodes (i, j, k) in a DAG G is a **v-structure (unshielded collider)** if $i \rightarrow j \leftarrow k$ in G and i **is not adjacent to** k

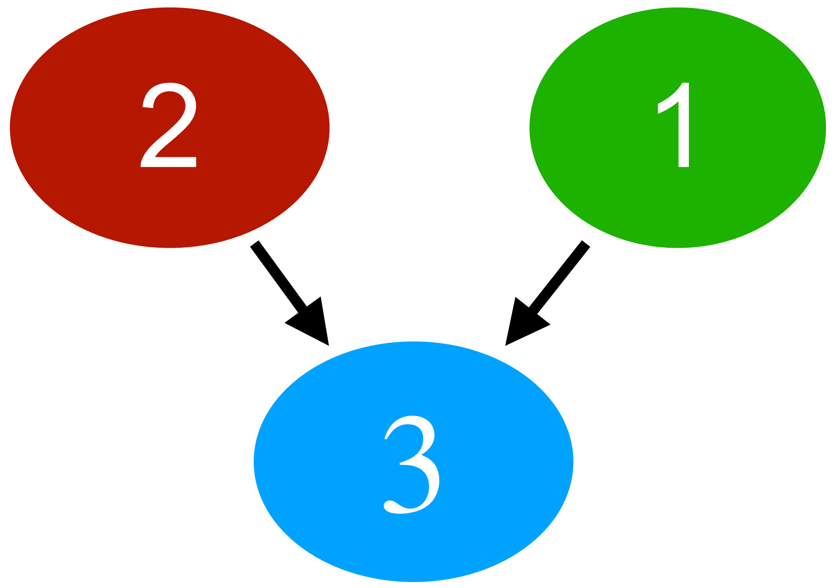


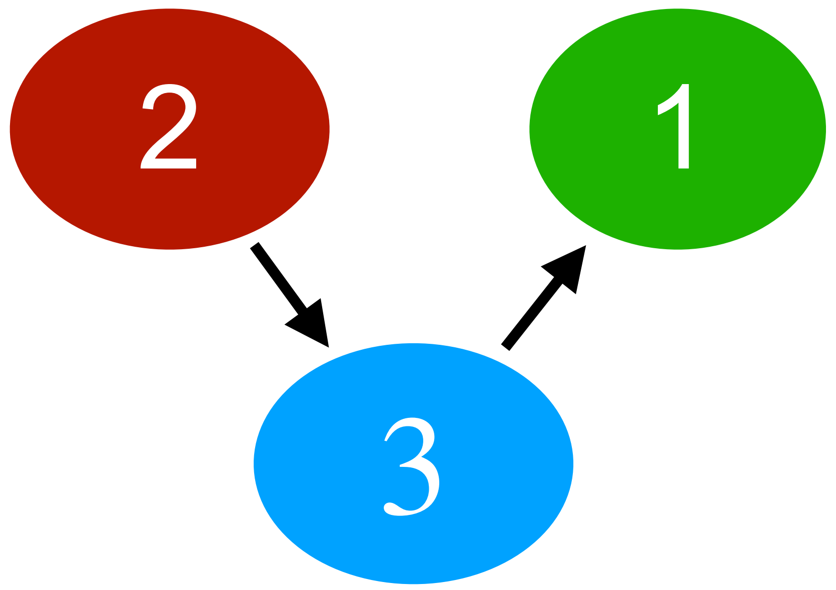
More graph terminology: v-structures/immoralities

- A triple of nodes (i, j, k) in a DAG G is a **v-structure (unshielded collider)** if $i \rightarrow j \leftarrow k$ in G and i **is not adjacent to** k

In which graphs is 3 a collider on the path between 2 and 1?



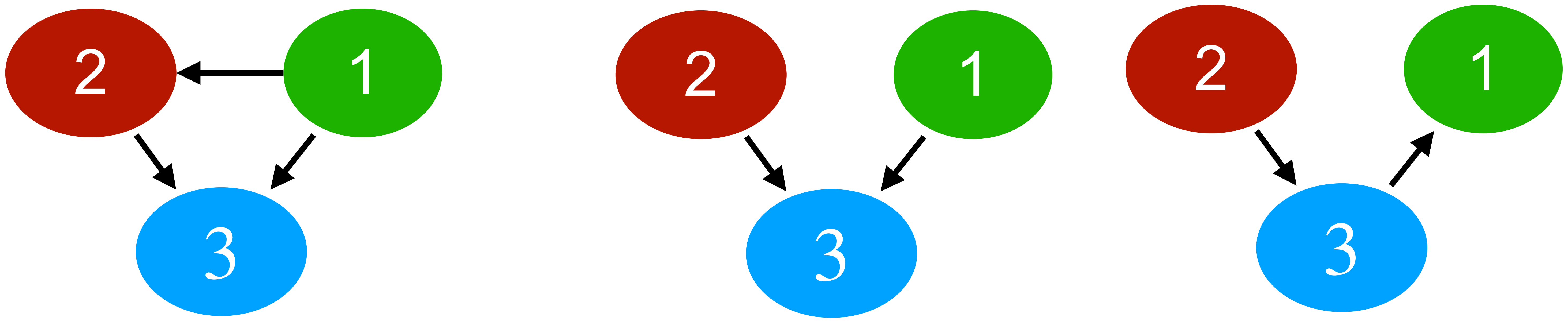




More graph terminology: v-structures/immoralities

- A triple of nodes (i, j, k) in a DAG G is a **v-structure (unshielded collider)** if $i \rightarrow j \leftarrow k$ in G and i **is not adjacent to** k

In which graphs is 3 part of a v-structure (2,3,1)?

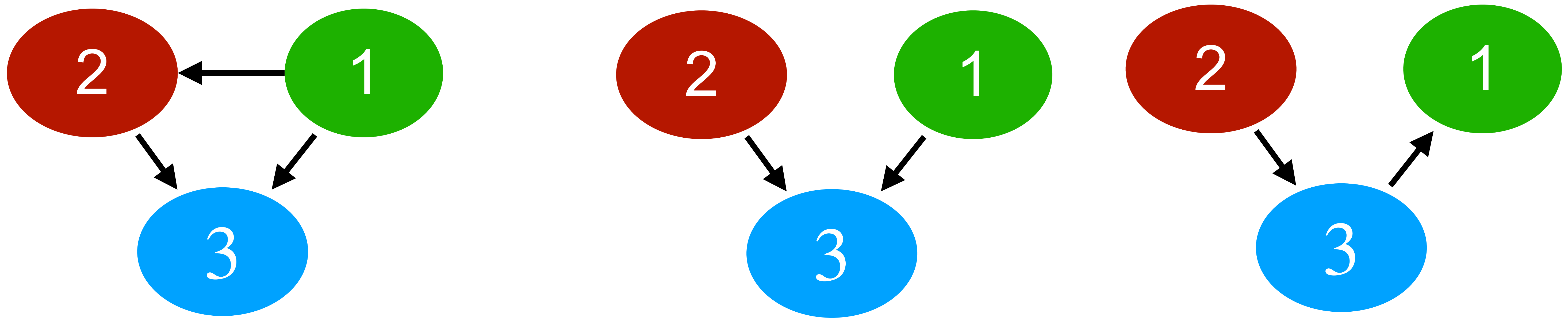


More graph terminology: unshielded triple

- A triple of nodes (i, j, k) in a DAG G is a **v-structure (unshielded collider)** if $i \to j \leftarrow k$ in G and i **is not adjacent to** k in G

- A triple of nodes (i, j, k) in a DAG G is a **an unshielded triple** if $i - j, j - k$ and i **is not adjacent to k** in G

In which graphs is (2,3,1) an unshielded triple?



Markov equivalence class and CPDAGs

essential graphs Summary graphs

- (Verma and Pearl 1990) show that all DAGs in a Markov equivalence class have the **same skeleton** and the **same v-structures**

Markov equivalence class and CPDAGs

- (Verma and Pearl 1990) show that all DAGs in a Markov equivalence class have the **same skeleton** and the **same v-structures**

- We can represent the skeleton and the orientations (edge marks) all DAGs in a Markov equivalence class (MEC) have in common with a **Complete Partially Directed Acyclic Graph (CPDAG):**

Markov equivalence class and CPDAGs

- (Verma and Pearl 1990) show that all DAGs in a Markov equivalence class have the **same skeleton** and the **same v-structures**

- We can represent the skeleton and the orientations (edge marks) all DAGs in a Markov equivalence class (MEC) have in common with a **Complete Partially Directed Acyclic Graph (CPDAG):**
    - We have a **directed** edge $i \to j$ if **all DAGs** in the MEC have $i \to j$

Markov equivalence class and CPDAGs

- (Verma and Pearl 1990) show that all DAGs in a Markov equivalence class have the **same skeleton** and the **same v-structures**

- We can represent the skeleton and the orientations (edge marks) all DAGs in a Markov equivalence class (MEC) have in common with a **Complete Partially Directed Acyclic Graph (CPDAG):**
  - We have a **directed** edge $i \to j$ if **all DAGs** in the MEC have $i \to j$
  - We have an **undirected** edge $i - j$ if **some DAGs** in the MEC have $i \to j$ and others have $j \to i$

Markov equivalence class and CPDAGs

- **Complete Partially Directed Acyclic Graph (CPDAG):** Summary graphs
  - We have a directed edge $i \to j$ if all DAGs in the MEC have $i \to j$
  - We have an undirected edge $i - j$ if some DAGs have $i \to j$ and others $j \to i$

$X \to Z \to Y$ $\quad X \not\perp_d Y$ ✓ $\quad X \perp_d Y \mid Z$ ✓

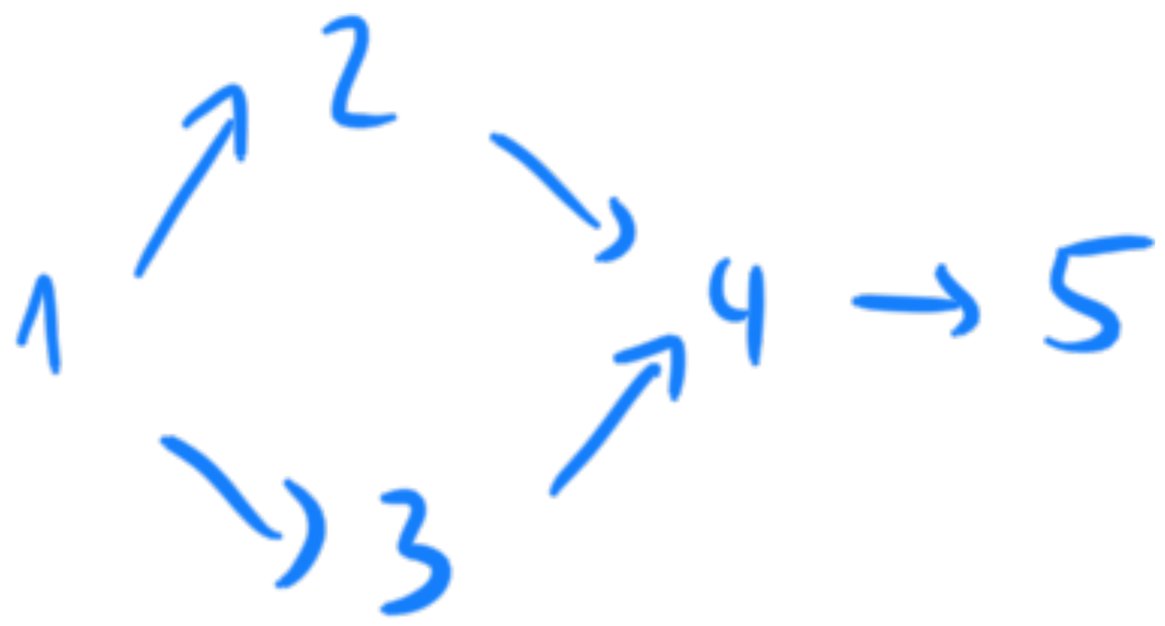
$X \leftarrow Z \to Y$ $\quad X \not\perp_d Y$ ✓ $\quad X \perp_d Y \mid Z$ ✓

$X \leftarrow Z \leftarrow Y$ $\quad X \not\perp_d Y$ ✓ $\quad X \perp_d Y \mid Z)$ ✓

$X - Z - Y$

CPDAG question

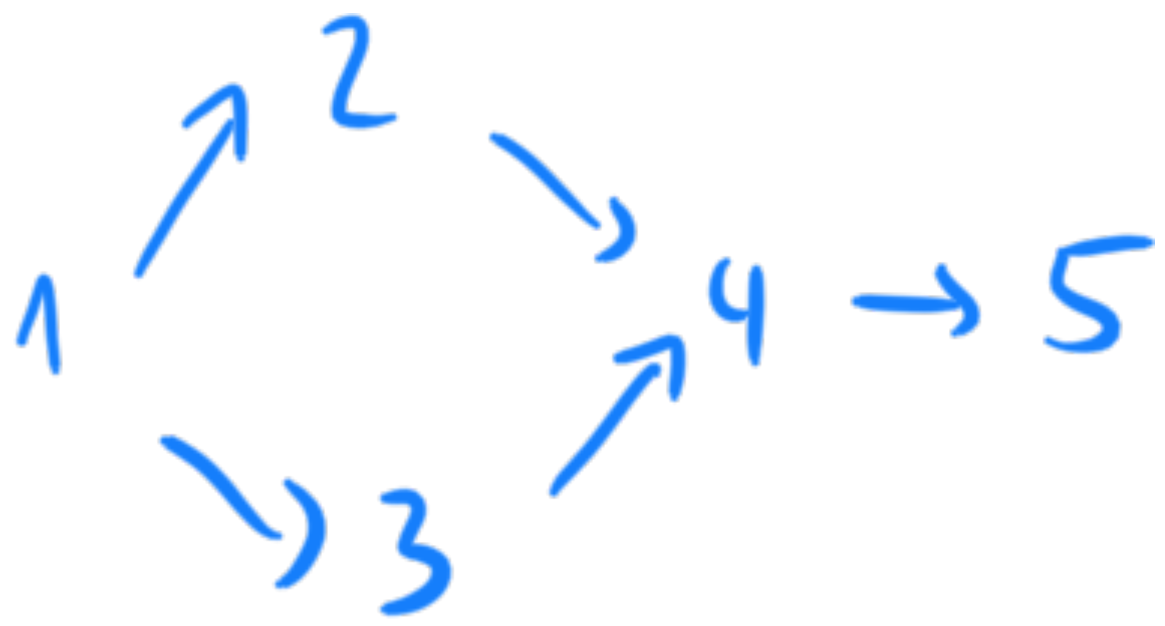
- (Verma and Pearl 1990) show that all DAGs in a Markov equivalence class have the **same skeleton** and the **same v-structures**

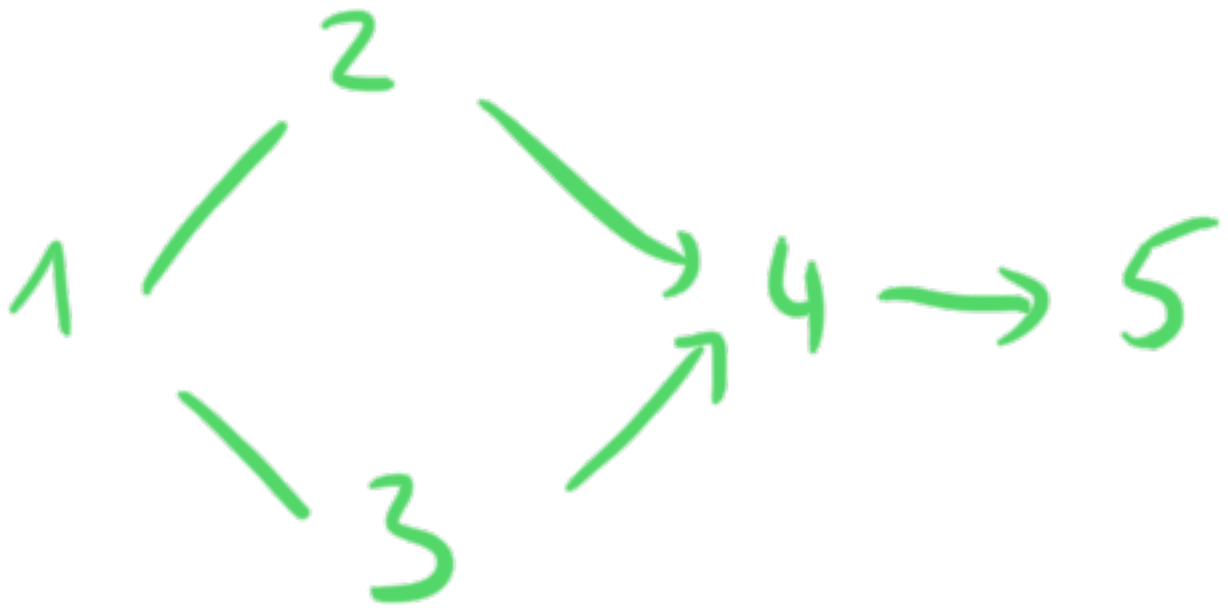


If the true graph is known we can compute the CPDAG representing the MEC

CPDAG question

- (Verma and Pearl 1990) show that all DAGs in a Markov equivalence class have the **same skeleton** and the **same v-structures**





If the true graph is known we can compute the CPDAG representing the MEC

What if it isn't known?!

Constraint-based causal discovery

- **Idea:** we perform conditional independence tests on **observational** data and use them to constrain the possible graphs using d-separation

$X \not\perp\!\!\!\perp Y \quad X \perp\!\!\!\perp Y \mid Z$

$X - Z - Y$

Constraint-based causal discovery

- **Idea:** we perform conditional independence tests on **observational** data and use them to constrain the possible graphs using d-separation
- In general, we can narrow down the possible graphs only up to their **Markov equivalence class (MEC)**

Constraint-based causal discovery

- **Idea:** we perform conditional independence tests on **observational** data and use them to constrain the possible graphs using d-separation
- In general, we can narrow down the possible graphs only up to their **Markov equivalence class (MEC)**
- The output of the algorithms we will see (e.g. SGS, PC) is a **CPDAG**, a mixed graph in which directed edges represent causal relations on which all DAGs in the MEC agree - these relations are **identifiable**

SGS algorithm (Spirtes, Glymour, Scheines)

- Assuming p is Markov and faithful to an unknown graph G

- We can estimate a CPDAG from samples of p in three steps:
  1. Determine the **skeleton**
  2. Determine the **v-structures**
  3. Direct as many remaining edges as possible

SGS algorithm (Spirtes, Glymour, Scheines)

- Assuming p is Markov and faithful to an unknown graph G

- We can estimate a CPDAG from samples of p in three steps:
  1. Determine the **skeleton**
  2. Determine the **v-structures** *(given the tests in the previous phase)*
  3. Direct as many remaining edges as possible

- **Note**: the directed parts of the CPDAG will agree with G, but some parts might stay undirected

Step 1: Skeleton learning

- Given $G = (\mathbf{V}, \mathbf{E})$, nodes $i, j \in \mathbf{V}, i \neq j$ then:
  - If **i is adjacent to j,** they cannot be d-separated by any subset of remaining nodes (and vice-versa)

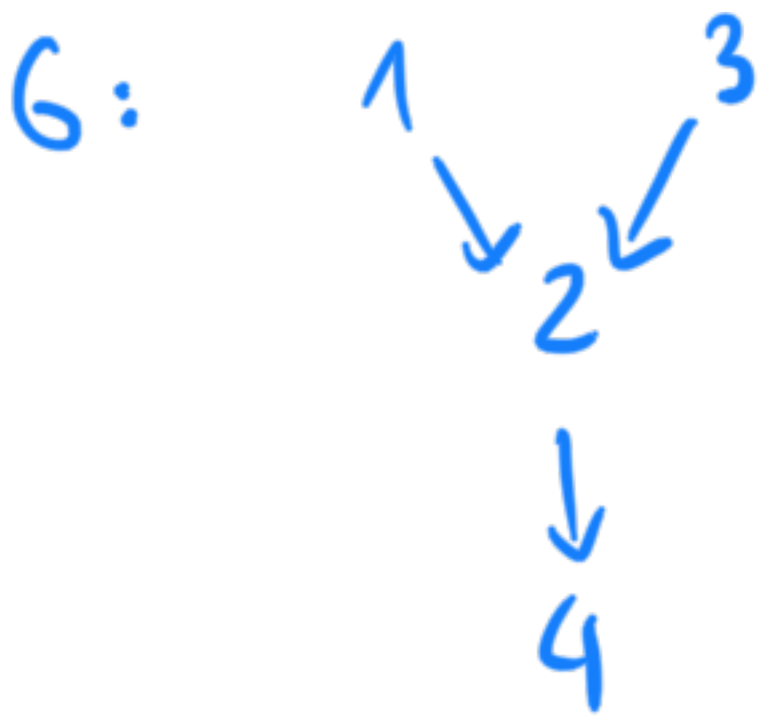
Step 1: Skeleton learning

- Given $G = (\mathbf{V}, \mathbf{E})$, nodes $i, j \in \mathbf{V}, i \neq j$ then:
  - If **i is adjacent to j,** they cannot be d-separated by any subset of remaining nodes (and vice-versa)

1. Start with **completely connected undirected graph U**
2. For each pair $i, j \in \mathbf{V}, i \neq j$, and for any subset $\mathbf{S} \subseteq \mathbf{V} \backslash \{i, j\}$
   - Check **if $X_i \perp\!\!\!\perp X_j \mid X_{\mathbf{S}}$** for any $\mathbf{S}$ in data
     - If this is true, by faithfulness $i \perp_G j \mid \mathbf{S}$, so we can **remove $i - j$ in U**

Step 1: Skeleton learning - example

True causal graph

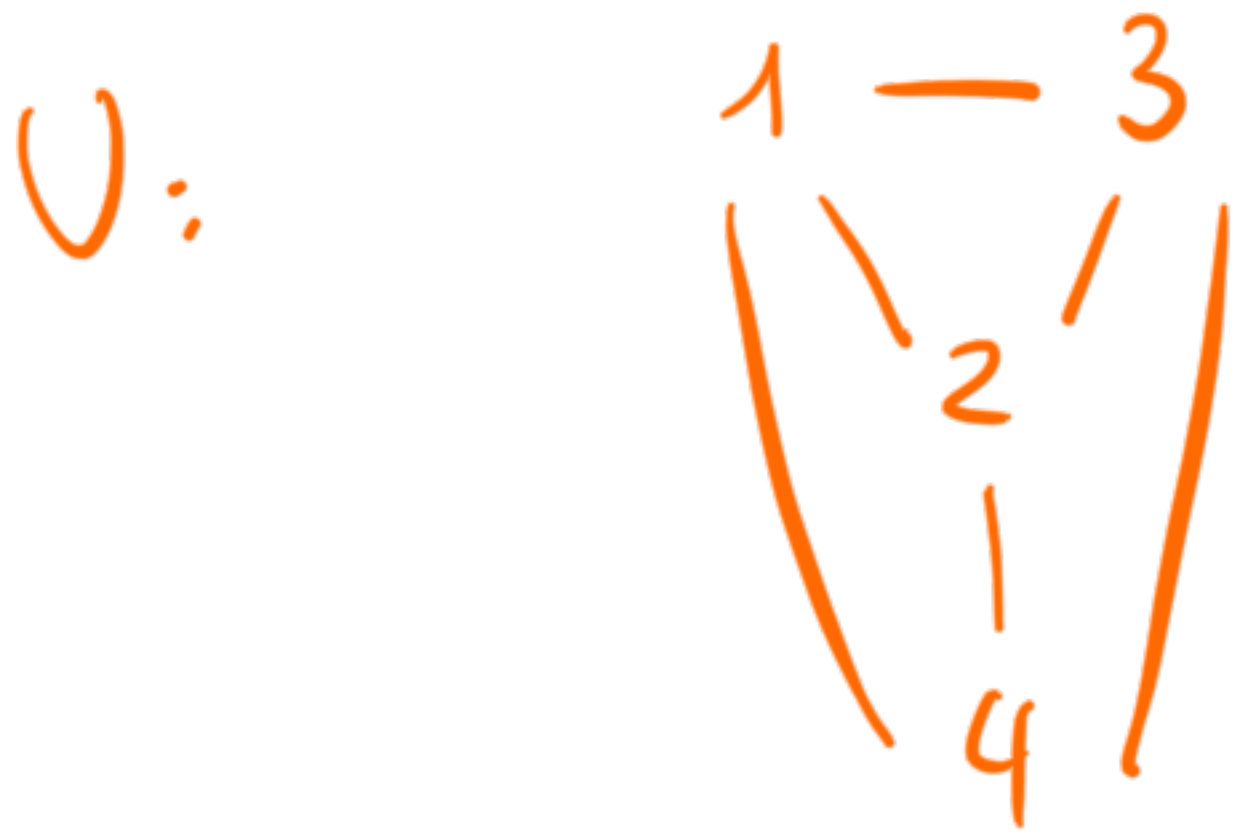


$P: P(X_1) \cdot P(X_3) \cdot P(X_2 \mid X_1, X_3) P(X_4 \mid X_2)$

Data

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0

Algorithm output



For this example, we will pretend we do not know the true graph, but only the CIs that we can derive from data

Step 1: Skeleton learning - example

True causal graph

G:

P: $P(X_1) \cdot P(X_3) \cdot P(X_2 \mid X_1, X_3)\, P(X_4 \mid X_2)$

Data

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0

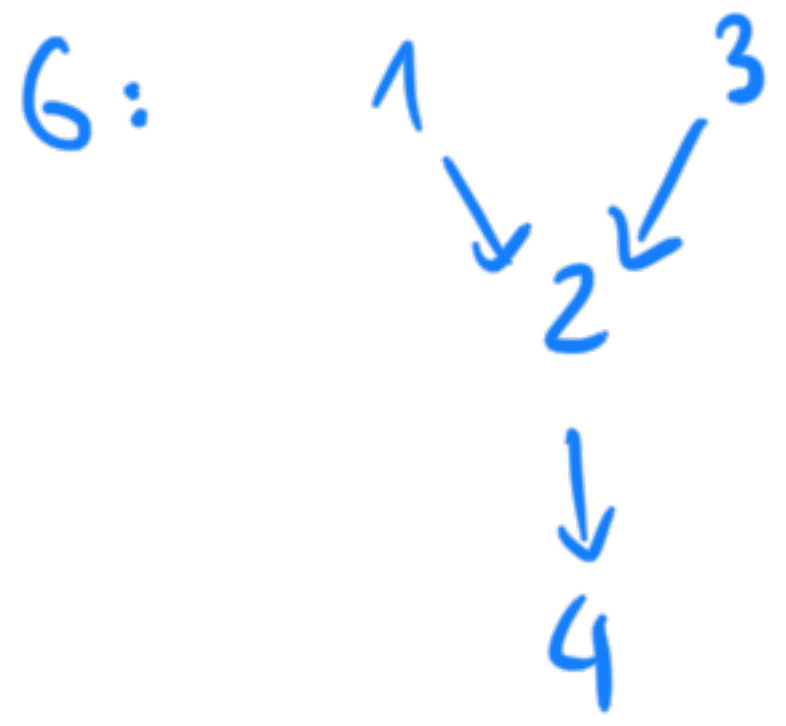
$X_1 \perp\!\!\!\perp X_3$

Algorithm output

U:

Step 1: Skeleton learning - example

True causal graph



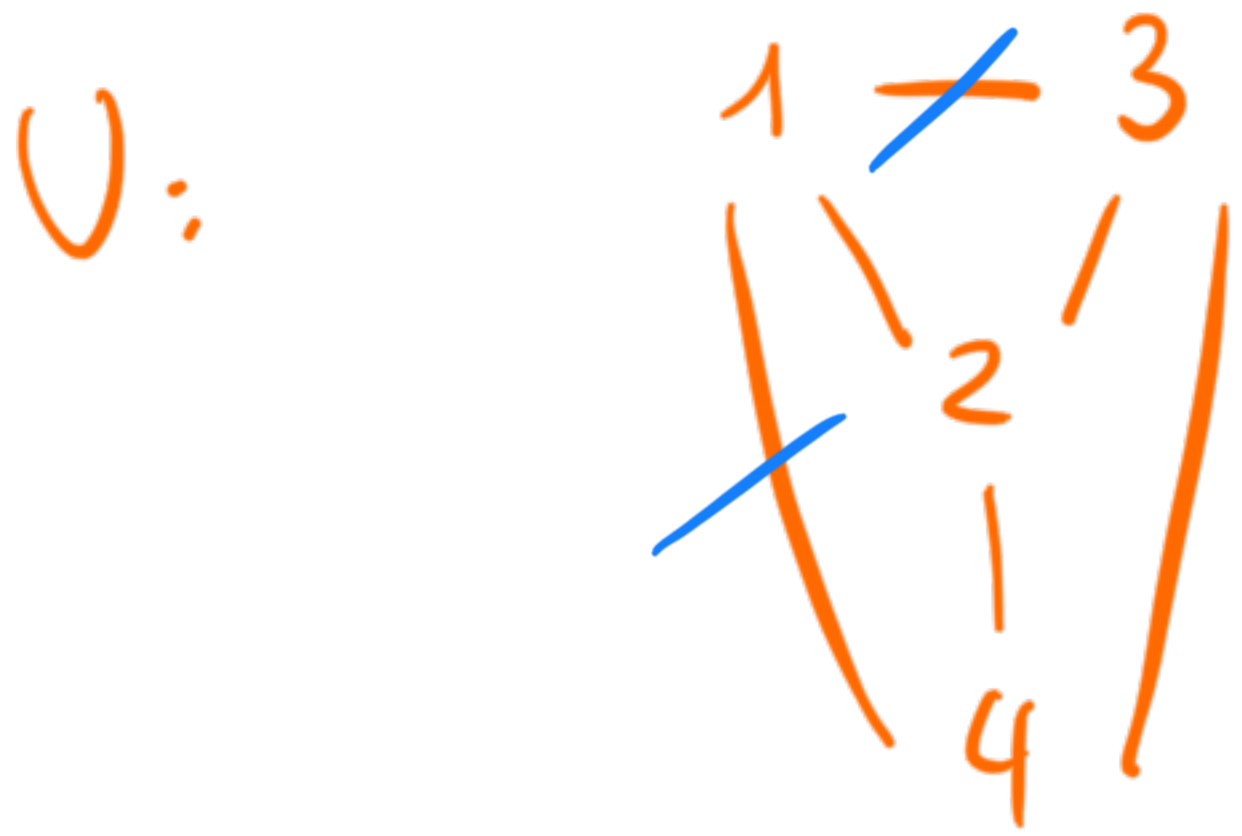
P: $P(X_1) \cdot P(X_3) \cdot P(X_2 | X_1, X_3) P(X_4 | X_2)$

Data

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0

$X_1 \perp\!\!\!\perp X_4 | X_2$

Algorithm output



Step 1: Skeleton learning - example

True causal graph

Data

Algorithm output

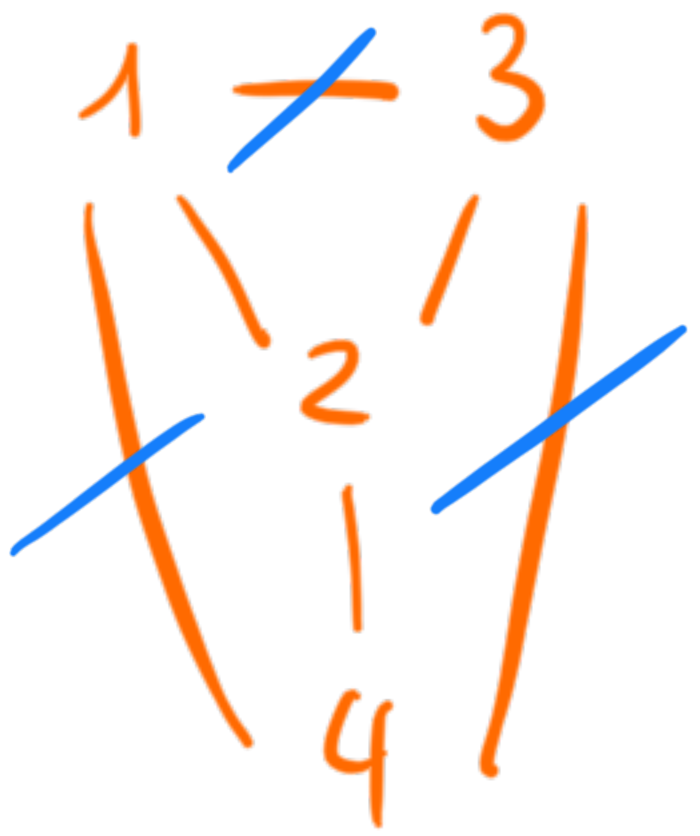
G:

$$P: \; P(X_1) \cdot P(X_3) \cdot P(X_2 \mid X_1, X_3)\, P(X_4 \mid X_2)$$

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0

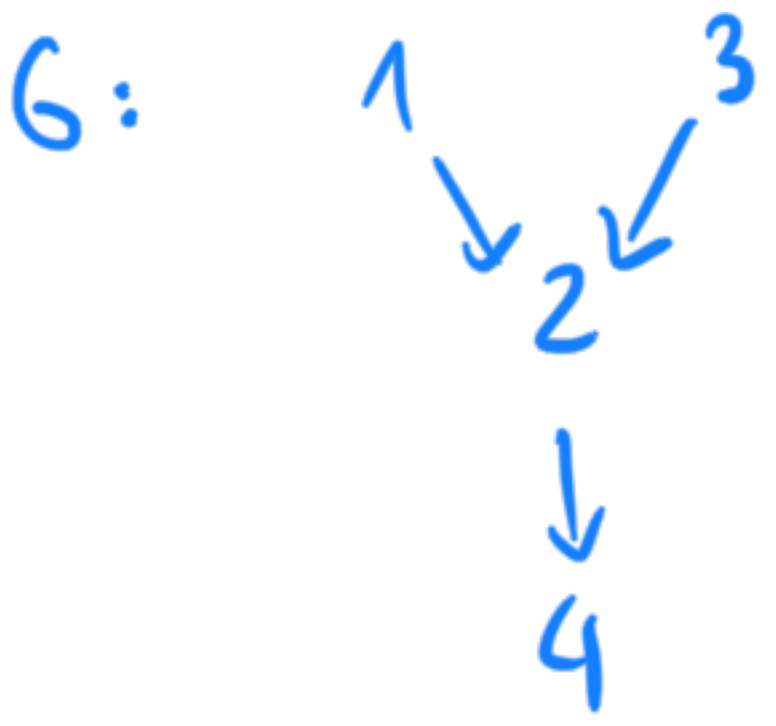
$$X_3 \perp\!\!\!\perp X_4 \mid X_2$$

U:



Step 1: Skeleton learning - example

True causal graph



P: $P(X_1) \cdot P(X_3) \cdot P(X_2 \mid X_1, X_3) P(X_4 \mid X_2)$

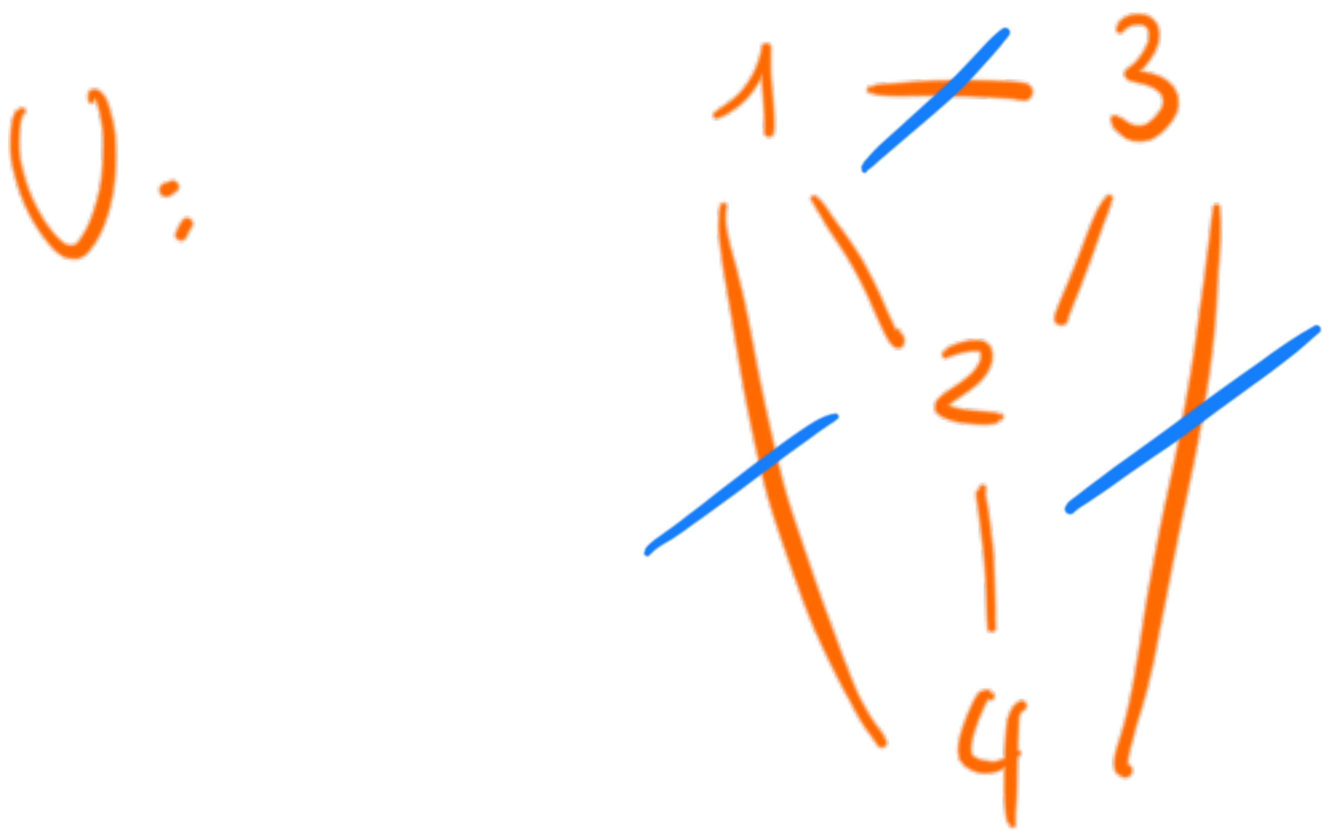
Data

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0

$X_1 \perp\!\!\!\perp X_4 \mid X_2, X_3$

$X_3 \perp\!\!\!\perp X_4 \mid X_2, X_1$

Algorithm output



Nothing changes, we can stop

Step 1: Skeleton learning - example

True causal graph

G:

P: $P(X_1) \cdot P(X_3) \cdot P(X_2 \mid X_1, X_3) P(X_4 \mid X_2)$

Data

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0

Algorithm output

U:

Step 1 finds the correct skeleton

SGS algorithm (Spirtes, Glymour, Scheines)

- Assuming p is Markov and faithful to an unknown graph G

- We can estimate a CPDAG from samples of p in three steps:
  1. ~~Determine the **skeleton**~~
  2. Determine the **v-structures** *(given the tests in the previous phase)*
  3. Direct as many remaining edges as possible

- **Note**: the directed parts of the CPDAG will agree with G, but some parts might stay undirected

Step 2: Determine v-structures

- A triple of nodes (i, j, k) in a DAG G is a **an unshielded triple** if $i - j, j - k$ and i **is not adjacent to k**, i.e. $i \not- k$, in G

Step 2: Determine v-structures

- A triple of nodes (i, j, k) in a DAG G is a **an unshielded triple** if $i - j, j - k$ and i **is not adjacent to k**, i.e. $i \not- k$, in G

1. Start from the skeleton U from previous step
2. For each unshielded triple (i, j, k) in U, i.e. $i - j, j - k$ **and** $i \not- k$ **in** U
   - For all $\mathbf{S} \subseteq \mathbf{V} \setminus \{i, j, k\}$ check if $X_i \not\perp\!\!\!\perp X_k \mid X_j \cup X_{\mathbf{S}}$ in data
     - If this is true, $i \rightarrow j \leftarrow k$ **is a v-structure**

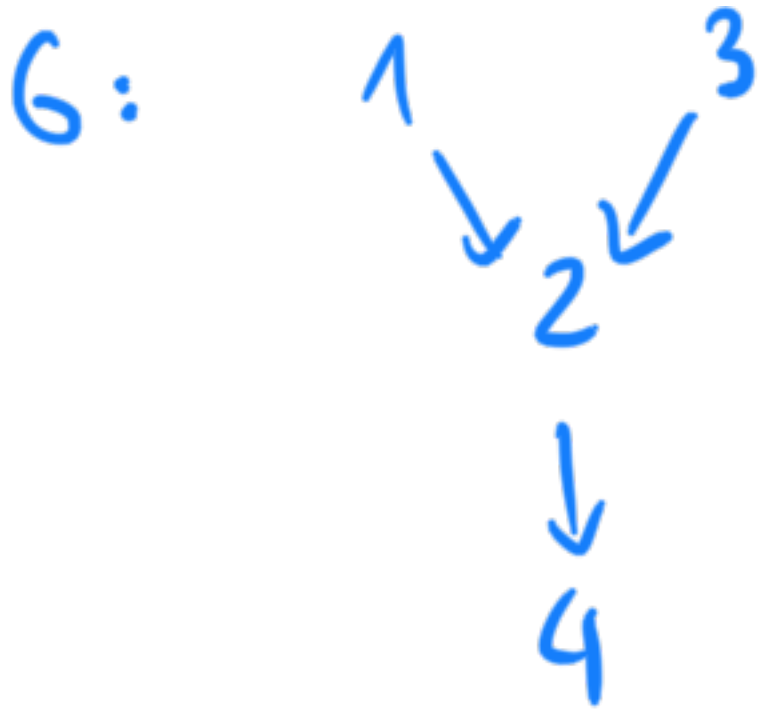
Step 2: Determine v-structures

- A triple of nodes (i, j, k) in a DAG G is a **an unshielded triple** if $i - j, j - k$ and i **is not adjacent to k**, i.e. $i \not- k$ in G

Keep in mind: for unshielded triples (i, j, k) we check if X_i, X_k are always dependent **given any conditioning set containing X_j**

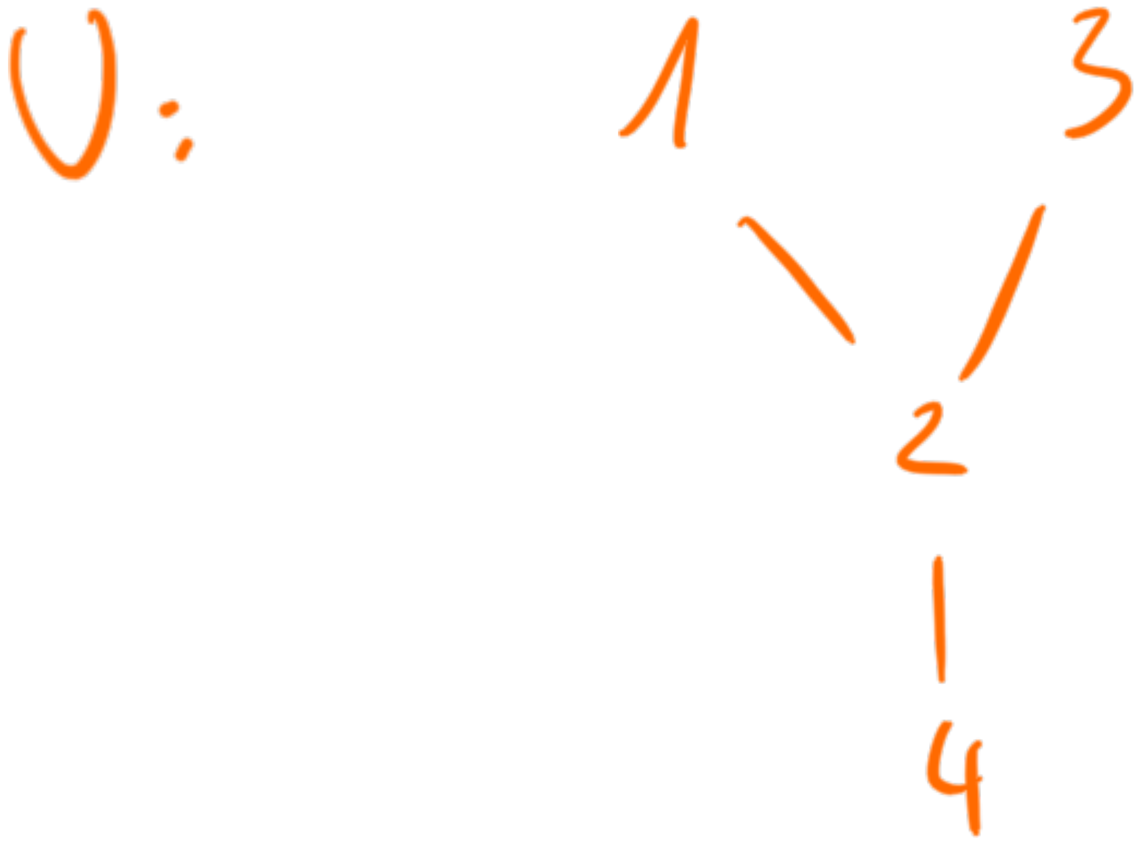
1. Start from the skeleton
2. For each unshielded triple (i, j, k) in U, i.e. $i - j, j - k$ **and** $i \not- k$ **in** U
   - For all $\mathbf{S} \subseteq \mathbf{V} \setminus \{i, j, k\}$ check if $X_i \not\perp\!\!\!\perp X_k \mid X_j \cup X_{\mathbf{S}}$ in data
     - If this is true, $i \rightarrow j \leftarrow k$ **is a v-structure**

Step 2: Determine v-structures - example



P: $P(X_1) \cdot P(X_3) \cdot P(X_2 \mid X_1, X_3) \, P(X_4 \mid X_2)$

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0



Step 2: Determine v-structures - example

G:

P: $P(X_1) \cdot P(X_3) \cdot P(X_2 \mid X_1, X_3)\, P(X_4 \mid X_2)$

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0

U:

$1 - 2 - 4$: $X_1 \perp\!\!\!\perp X_4 \mid X_2$

$X_1 \perp\!\!\!\perp X_4 \mid X_2, X_3$

We can reuse the CIs from phase 1

$X_1 \perp\!\!\!\perp X_3$

$X_1 \perp\!\!\!\perp X_4 \mid X_2$

$X_3 \perp\!\!\!\perp X_4 \mid X_2$

$X_1 \perp\!\!\!\perp X_4 \mid X_2, X_3$

$X_3 \perp\!\!\!\perp X_4 \mid X_2, X_1$

Step 2: Determine v-structures - example

G:

U:

P: $P(X_1)\cdot P(X_3)\cdot P(X_2 \mid X_1, X_3)\, P(X_4 \mid X_2)$

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0

We can reuse the CIs from phase 1

$X_1 \perp\!\!\!\perp X_3$

$X_1 \perp\!\!\!\perp X_4 \mid X_2$

$X_3 \perp\!\!\!\perp X_4 \mid X_2$

$X_1 \perp\!\!\!\perp X_4 \mid X_2, X_3$

$X_3 \perp\!\!\!\perp X_4 \mid X_2, X_1$

3 − 2 − 4

$X_3 \perp\!\!\!\perp X_4 \mid X_2$

$X_3 \perp\!\!\!\perp X_4 \mid X_2, X_1$

X

Step 2: Determine v-structures - example

G:

$P: \; P(X_1)\cdot P(X_3)\cdot P(X_2 \mid X_1, X_3)\, P(X_4 \mid X_2)$

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0

U:

We can reuse the CIs from phase 1

$X_1 \perp\!\!\!\perp X_3$

$X_1 \perp\!\!\!\perp X_4 \mid X_2$

$X_3 \perp\!\!\!\perp X_4 \mid X_2$

$X_1 \perp\!\!\!\perp X_4 \mid X_2, X_3$

$X_3 \perp\!\!\!\perp X_4 \mid X_2, X_1$

$1 - 2 - 3$

$X_1 \not\perp\!\!\!\perp X_3 \mid X_2$ ✓

$X_1 \not\perp\!\!\!\perp X_3 \mid X_2, X_4$ ✓

$X_1 \perp\!\!\!\perp X_3$ has $\mathbf{Z} = \emptyset$, so it does not contain X_2 and we ignore it

Step 2: Determine v-structures - example

G:

P: $P(X_1) \cdot P(X_3) \cdot P(X_2 \mid X_1, X_3) P(X_4 \mid X_2)$

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0

U:

We can reuse the CIs from phase 1

$X_1 \perp\!\!\!\perp X_3$

$X_1 \perp\!\!\!\perp X_4 \mid X_2$

$X_3 \perp\!\!\!\perp X_4 \mid X_2$

$X_1 \perp\!\!\!\perp X_4 \mid X_2, X_3$

$X_3 \perp\!\!\!\perp X_4 \mid X_2, X_1$

1 − 2 − 3

$X_1 \not\!\perp\!\!\!\perp X_3 \mid X_2$ ✓

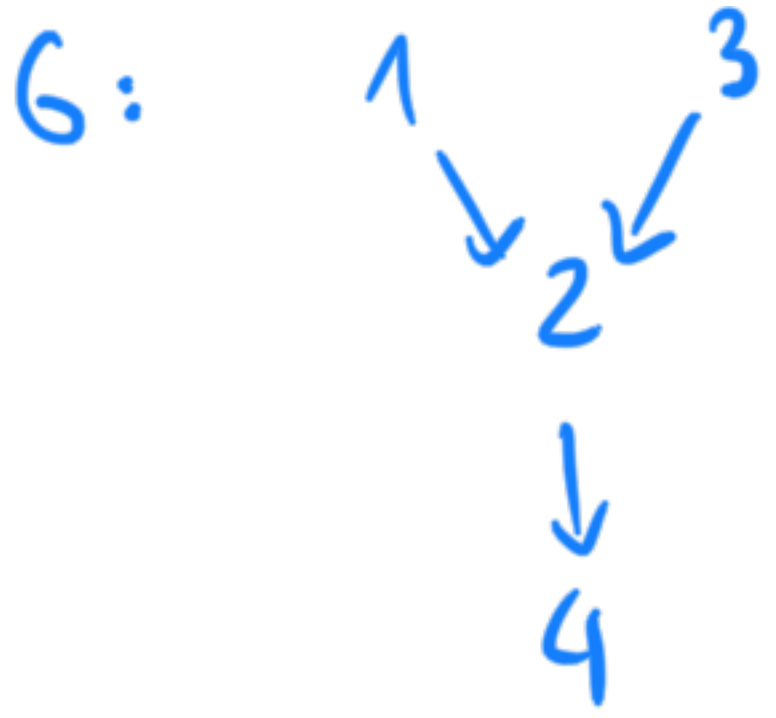
$X_1 \not\!\perp\!\!\!\perp X_3 \mid X_2, X_4$ ✓

SGS algorithm (Spirtes, Glymour, Scheines)

- Assuming P is Markov and faithful to an unknown graph G
- We can estimate a CPDAG from samples of P in three steps:
  1. ~~Determine the **skeleton**~~
  2. ~~Determine the **v-structures** *(given the tests in the previous phase)*~~
  3. Direct as many remaining edges as possible
- **Note**: the directed parts of the CPDAG will agree with G, but some parts might stay undirected

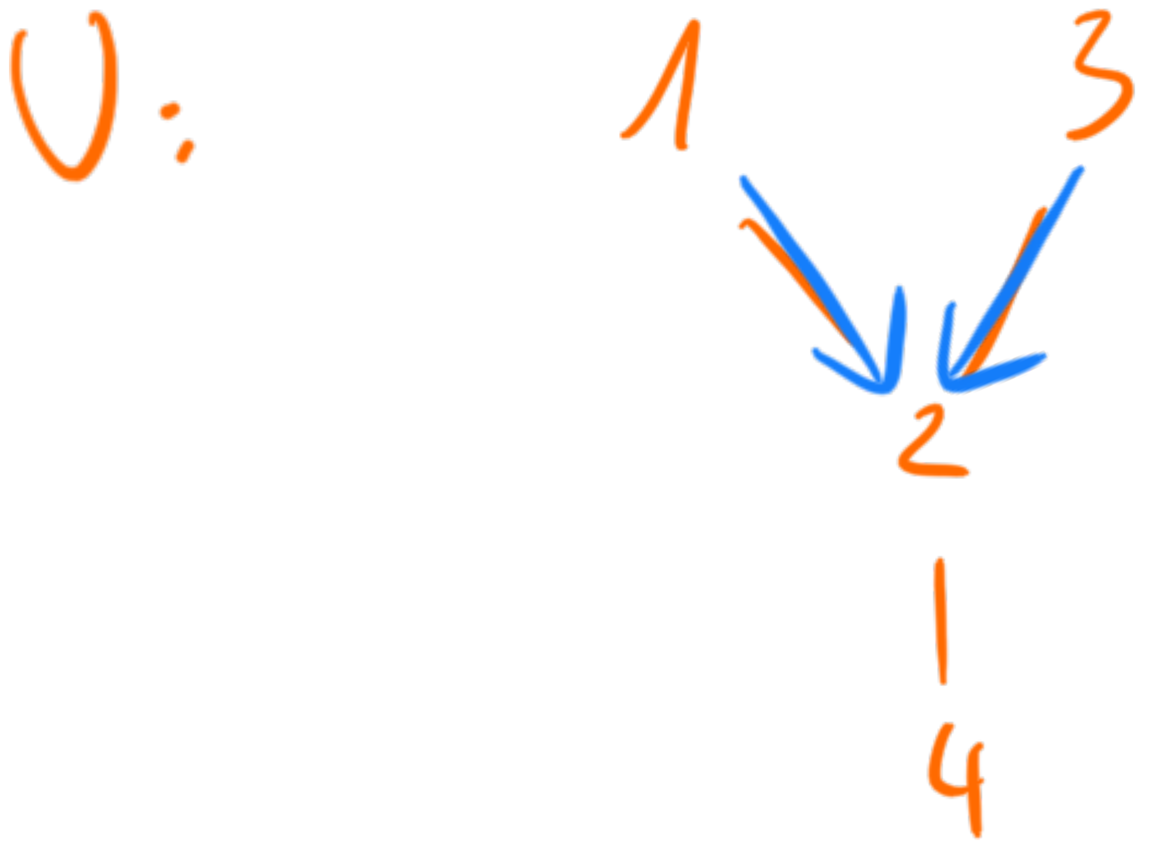
Step 3: Direct as many edges as possible - example

- Cannot create cycles or new v-structures
- Some of the edges can be oriented to disallow these situations to happen



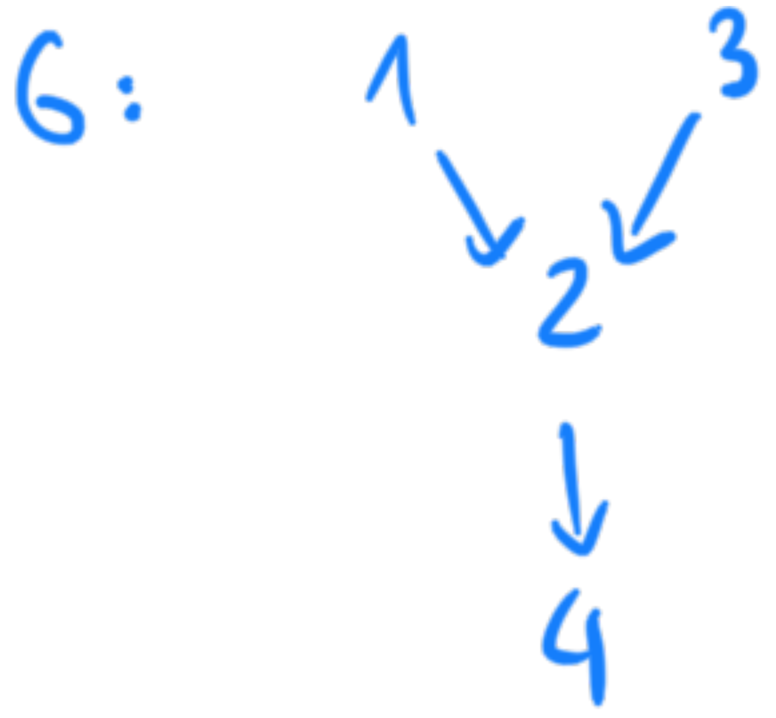
P: $P(X_1) \cdot P(X_3) \cdot P(X_2 | X_1, X_3) P(X_4 | X_2)$

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0



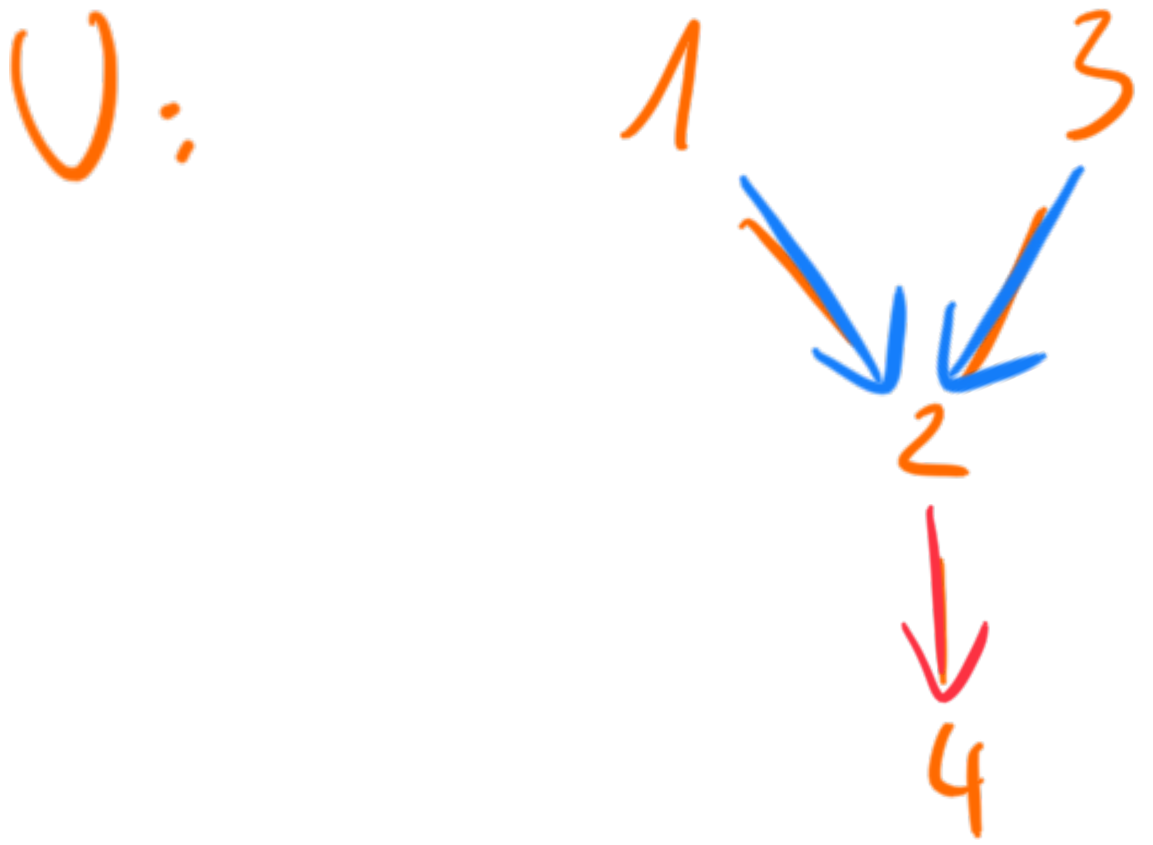
Step 3: Direct as many edges as possible - example

- Cannot create cycles or new v-structures
- Some of the edges can be oriented to disallow these situations to happen



$$P: \quad P(X_1) \cdot P(X_3) \cdot P(X_2 \mid X_1, X_3)\, P(X_4 \mid X_2)$$

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0



Step 3: Direct as many edges as possible - example

- Cannot create cycles or new v-structures
- Some of the edges can be oriented to disallow these situations to happen

G:

P: $P(X_1) \cdot P(X_3) \cdot P(X_2 | X_1, X_3) P(X_4 | X_2)$

X_1	X_2	X_3	X_4
1	1	1	0
1	0	0	1
0	1	0	1
1	0	1	0
1	0	0	0

CPDAG:

Same as G (special case)

Step 3: Meek's rules (1995)

Sound and complete rules for additional orientations (also with added background knowledge)

otherwise v-structure

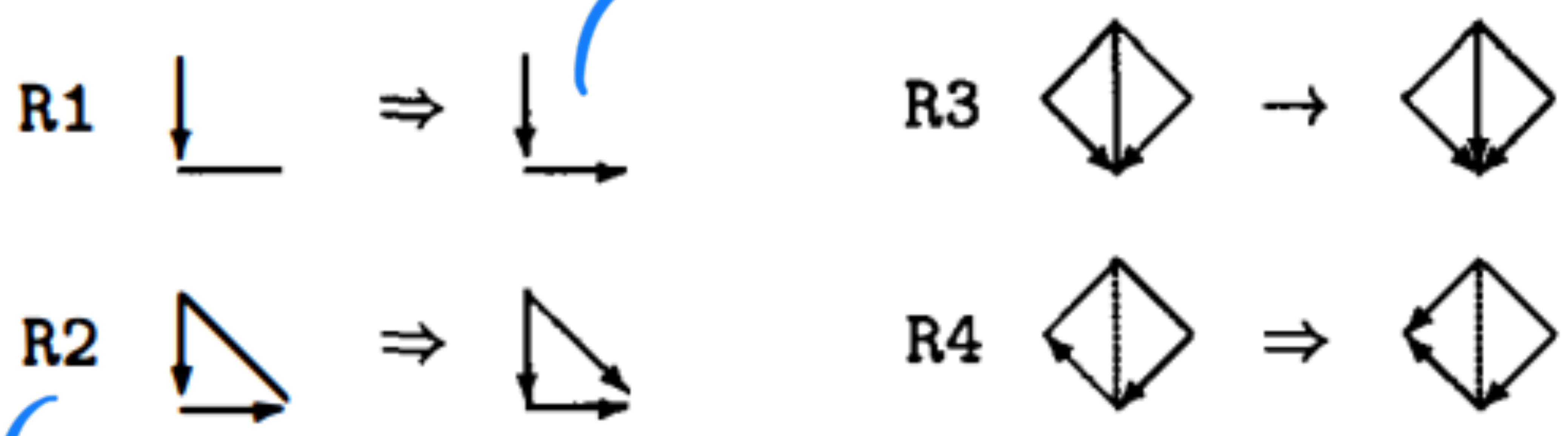


Figure 1: Orientation rules for patterns

otherwise cycle

https://arxiv.org/pdf/1302.4972.pdf

SGS algorithm (Spirtes, Glymour, Scheines)

- Assuming p is Markov and faithful to an unknown graph G
- We can estimate a CPDAG from samples of p in three steps:
  1. Determine the **skeleton**
  2. Determine the **v-structures** *(given the tests in the previous phase)*
  3. Direct as many remaining edges as possible
- **Computationally inefficient:** potentially $O(2^p)$ tests
  - Even if we reuse the test results from skeleton phase in other phases

PC algorithm (Peter Spirtes, Clark Glymour)

- Assuming p is Markov and faithful to an unknown graph G

- We can estimate a CPDAG from samples of p in three steps:
  1. Determine the **skeleton in an optimised way**
  2. Determine the **v-structures** *(given the tests in the previous phase)*
  3. Direct as many remaining edges as possible

PC algorithm skeleton learning

- If i **is not adjacent to** j, then they can be d-separated by $\mathrm{Pa}(i)$ or $\mathrm{Pa}(j)$
- Determine the **skeleton in an optimised way**
  - Since we do not know the parents we will use the nodes that are adjacent, $\mathrm{Adj}(i)$ or $\mathrm{Adj}(j)$ in U at a given iteration (*superset*)

PC algorithm skeleton learning

- If **i is not adjacent to j,** then they can be d-separated by $\mathrm{Pa}(i)$ or $\mathrm{Pa}(j)$
- Determine the **skeleton in an optimised way**
  - Since we do not know the parents we will use the nodes that are adjacent, $\mathrm{Adj}(i)$ or $\mathrm{Adj}(j)$ in U at a given iteration (*superset*)

1. Start with **completely connected undirected graph U**

PC algorithm skeleton learning

- If i **is not adjacent to** j**,** then they can be d-separated by $\mathrm{Pa}(i)$ or $\mathrm{Pa}(j)$
- Determine the **skeleton in an optimised way**
  - Since we do not know the parents we will use the nodes that are adjacent, $\mathrm{Adj}(i)$ or $\mathrm{Adj}(j)$ in U at a given iteration (*superset*)

1. Start with **completely connected undirected graph** U
2. For $k = 0,1,2\ldots, p-2$
   - If $i - j$ in U and there exists a set $\mathbf{S} \subseteq \mathrm{Adj}(i) \backslash \{j\}$ **of size at least k**

PC algorithm skeleton learning

- If i **is not adjacent to** j, then they can be d-separated by $\mathrm{Pa}(i)$ or $\mathrm{Pa}(j)$
- Determine the **skeleton in an optimised way**
  - Since we do not know the parents we will use the nodes that are adjacent, $\mathrm{Adj}(i)$ or $\mathrm{Adj}(j)$ in U at a given iteration (*superset*)

1. Start with **completely connected undirected graph** U
2. For $k = 0,1,2\ldots,p-2$
   - If $i - j$ in U and there exists a set $\mathbf{S} \subseteq \mathrm{Adj}(i)\backslash\{j\}$ **of size at least k**
     - If holds $X_i \perp\!\!\!\perp X_j \mid \mathbf{S}$, then **remove** $i - j$ **in** U

PC algorithm skeleton learning

- If **i is not adjacent to j,** then they can be d-separated by $\mathrm{Pa}(i)$ or $\mathrm{Pa}(j)$
- Determine the **skeleton in an optimised way**
  - Since we do not know the parents we will use the nodes that are adjacent, $\mathrm{Adj}(i)$ or $\mathrm{Adj}(j)$ in U at a given iteration (*superset*)

We stop when there are no more untested sets of size k in the adjacencies of any variable

1. Start with **completely connect**
2. For $k = 0,1,2\ldots,p-2$
   - If $i - j$ in U and there exists a set $\mathbf{S} \subseteq \mathrm{Adj}(i)\backslash\{j\}$ **of size at least k**
     - If holds $X_i \perp\!\!\!\perp X_j \mid \mathbf{S}$, then **remove $i - j$ in U**

SGS vs PC

- **SGS:** we can estimate a CPDAG from samples of p in three steps:
  1. Determine the **skeleton**
  2. Determine the **v-structures**
  3. Direct as many remaining edges as possible

- **PC:** we can estimate a CPDAG from samples of p in three steps:
  1. Determine the **skeleton in an optimised way**
  2. Determine the **v-structures**
  3. Direct as many remaining edges as possible

PC algorithm - when does it fail?

- If the conditional independence tests give the wrong result
  - Too few samples
  - A very weak dependence
  - Wrong parametric assumption (e.g. partial correlation on nonlinear data)

If you're curious, you can see here some variants (like conservative PC) that try to circumvent the problem by doing more tests https://rdrr.io/cran/pcalg/man/pc.html

PC algorithm - when does it fail?

- If the conditional independence tests give the wrong result
  - Too few samples
  - A very weak dependence
  - Wrong parametric assumption (e.g. partial correlation on nonlinear data)

- If there are unmeasured confounders or selection bias
  - For example Chocolate - Nobel prizes

$C - N$ CPDAG $\begin{cases} C \to N \\ C \leftarrow N \end{cases}$

PC algorithm - when does it fail?

- If the conditional independence tests give the wrong result
  - Too few samples
  - A very weak dependence
  - Wrong parametric assumption (e.g. partial correlation on nonlinear data)

- If there are unmeasured confounders or selection bias

- We use advanced constraint-based algorithms - Fast Causal Inference (FCI)
  - Chapter 6 in [SGS] Causation Prediction and Search
    - https://www.researchgate.net/publication/242448131_Causation_Prediction_and_Search

Break?

Optional SGS exercise:
https://drive.google.com/file/d/14IR7BSH2N7ZasRXIO5nXrF3Xn49hnmRn/view?usp=sharing

Causal discovery - this class

Constraint-based causal discovery

- Conditional independence tests

- Observational data
- Output: MEC

- SGS, PC

Score-based causal discovery

- Penalised likelihood

- Observational data
- Output: MEC

- GES, MMHC

Restricted models

- Nonlinear additive noise, Linear Non-Gaussianity

- Observational data
- Output: DAG

- RESIT, LINGAM

Interventional causal discovery / causal invariance

- Observational and Interventional data
- Output: parents of Y, I-MEC

- ICP, JCI

Learning from interventional data - intuition

Until now we have only used **observational data.**

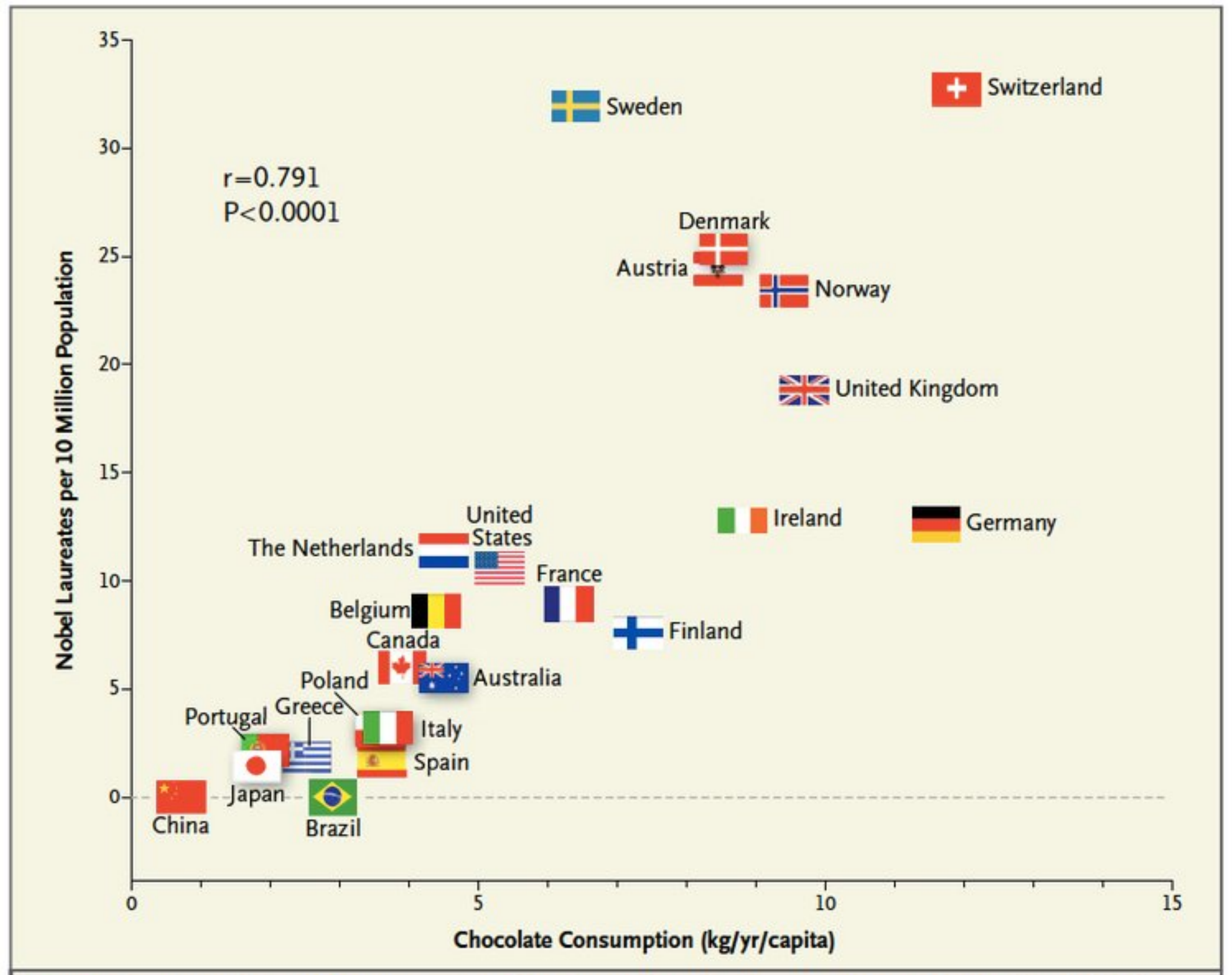
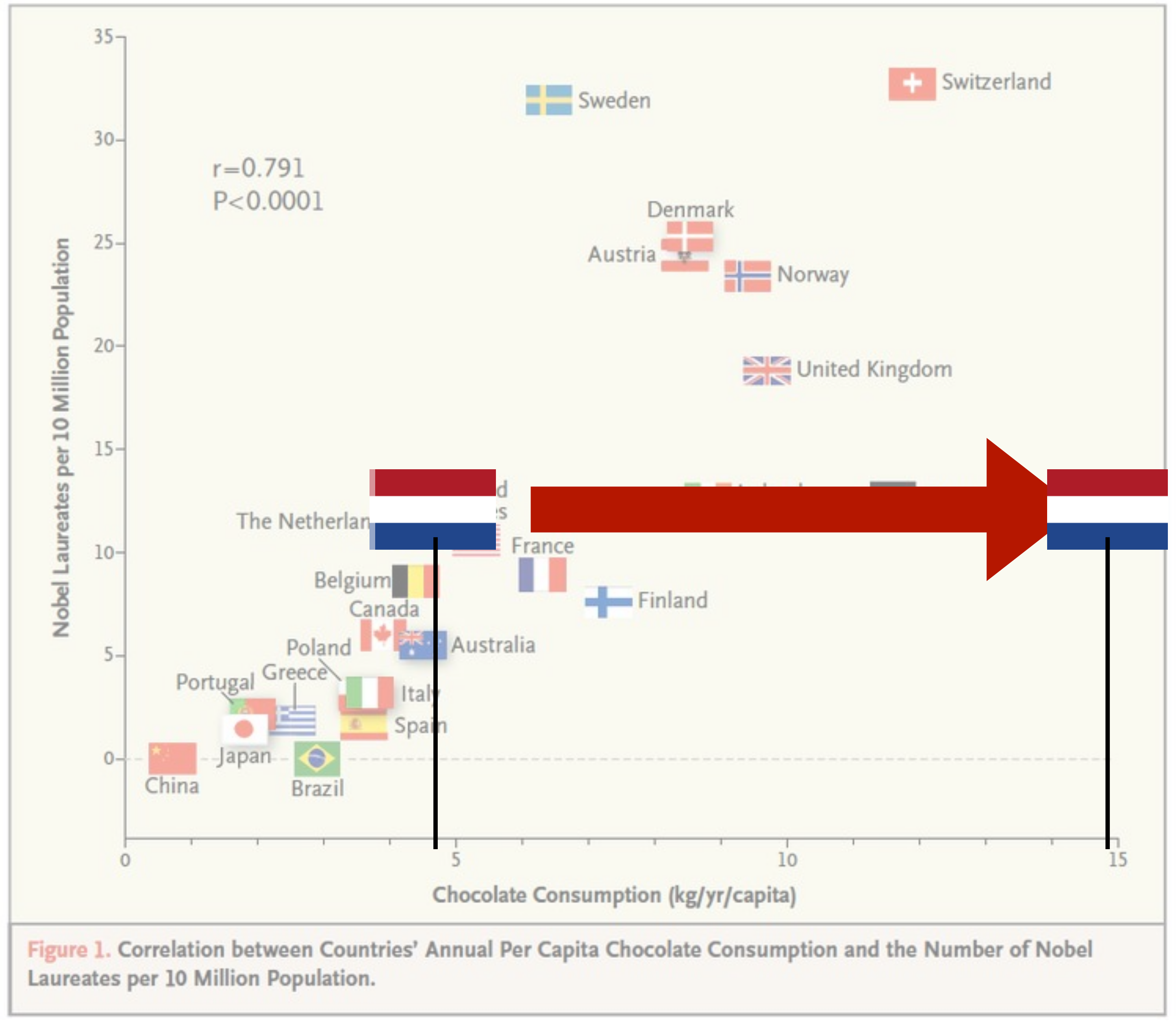


Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.

[Messerli, 2012] https://www.nejm.org/doi/full/10.1056/NEJMon1211064

Learning from interventional data - intuition

Hypothetical world: we perform the experiment and see these results:



NL eats more chocolate => nothing changes

[Messerli, 2012] https://www.nejm.org/doi/full/10.1056/NEJMon1211064

Learning from interventional data - intuition

Hypothetical world: we perform the experiment and see these results:

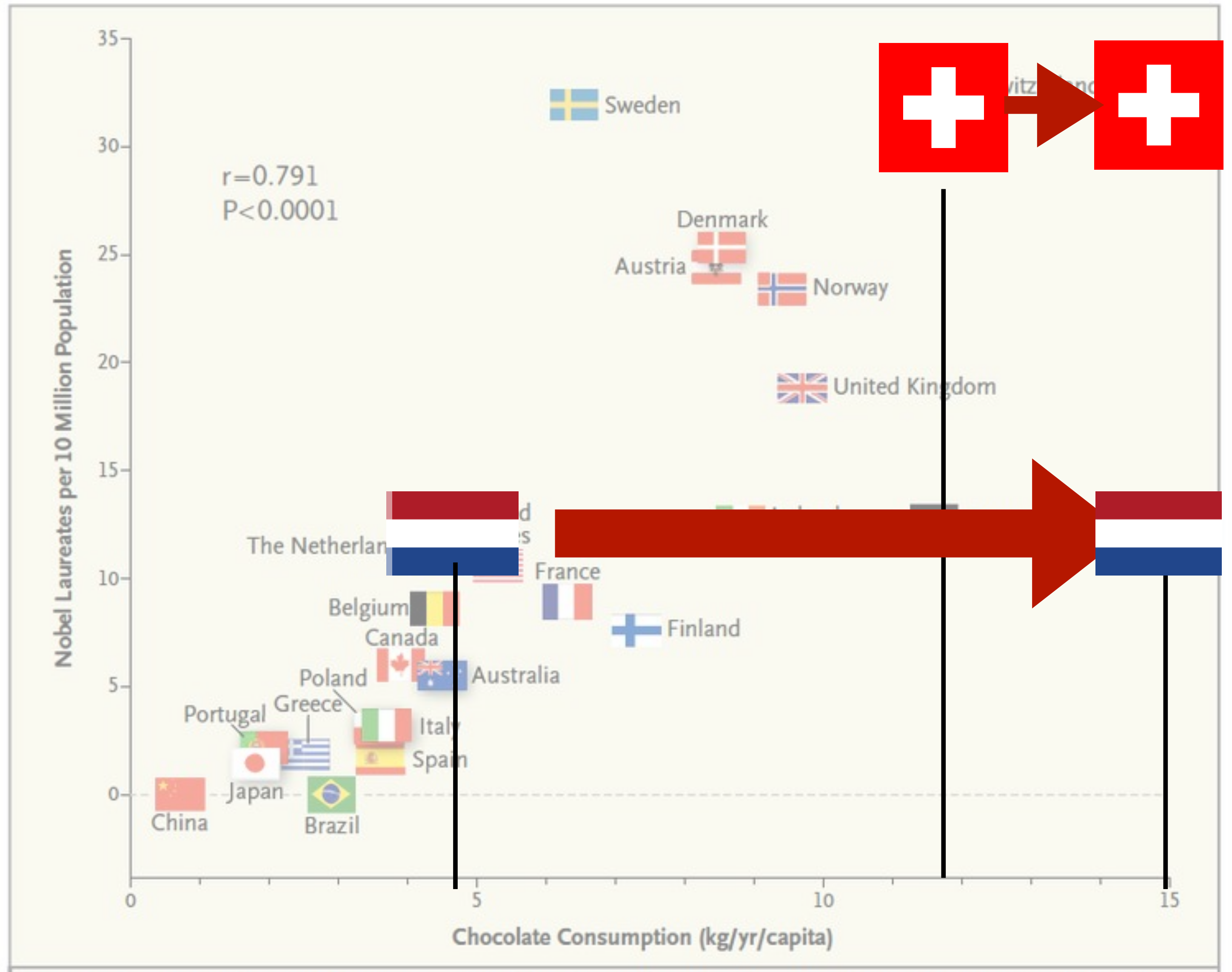


Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.

NL eats more chocolate => nothing changes

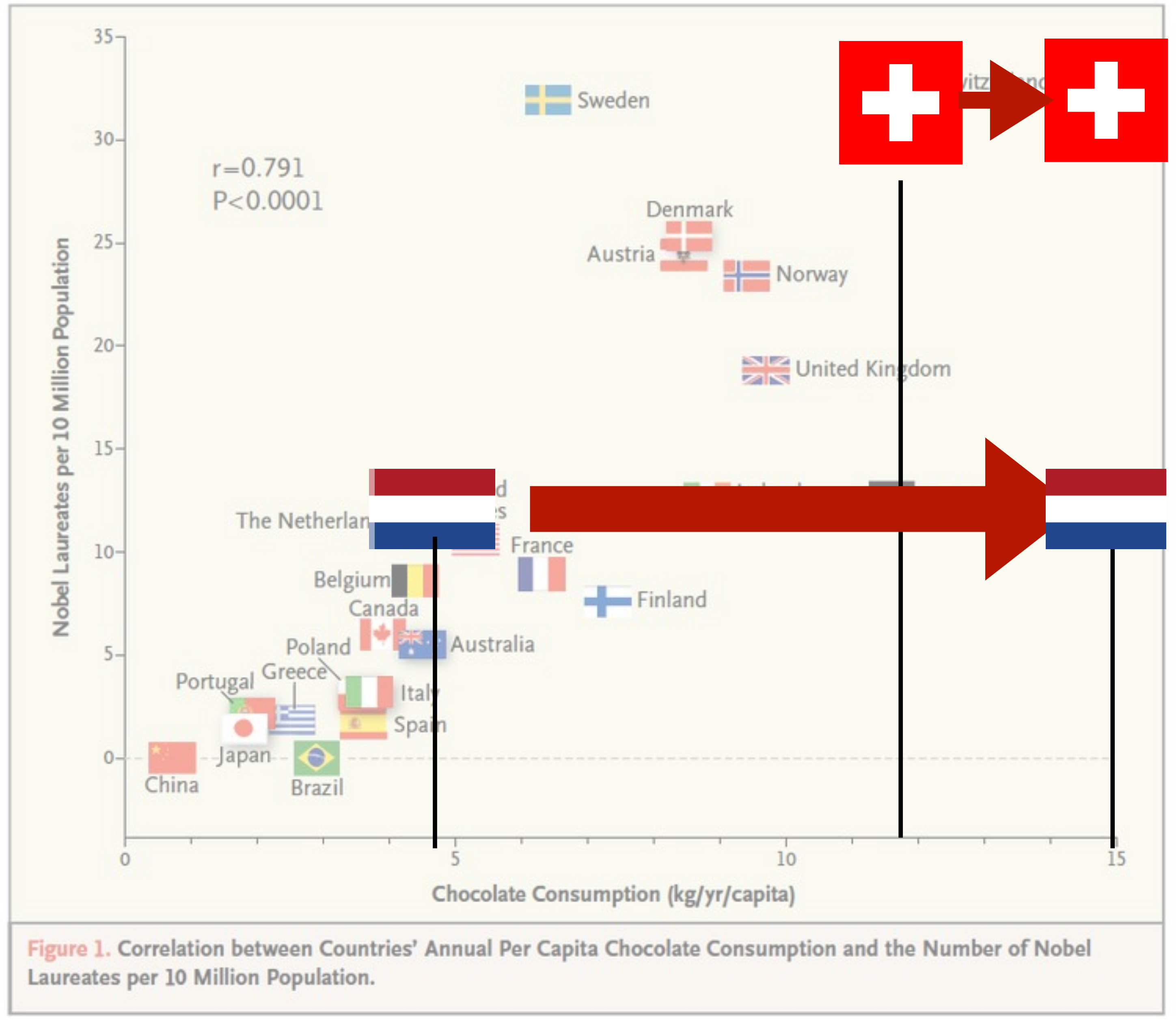
… and similarly for other countries (and other values)

Chocolate does not cause Nobel prizes

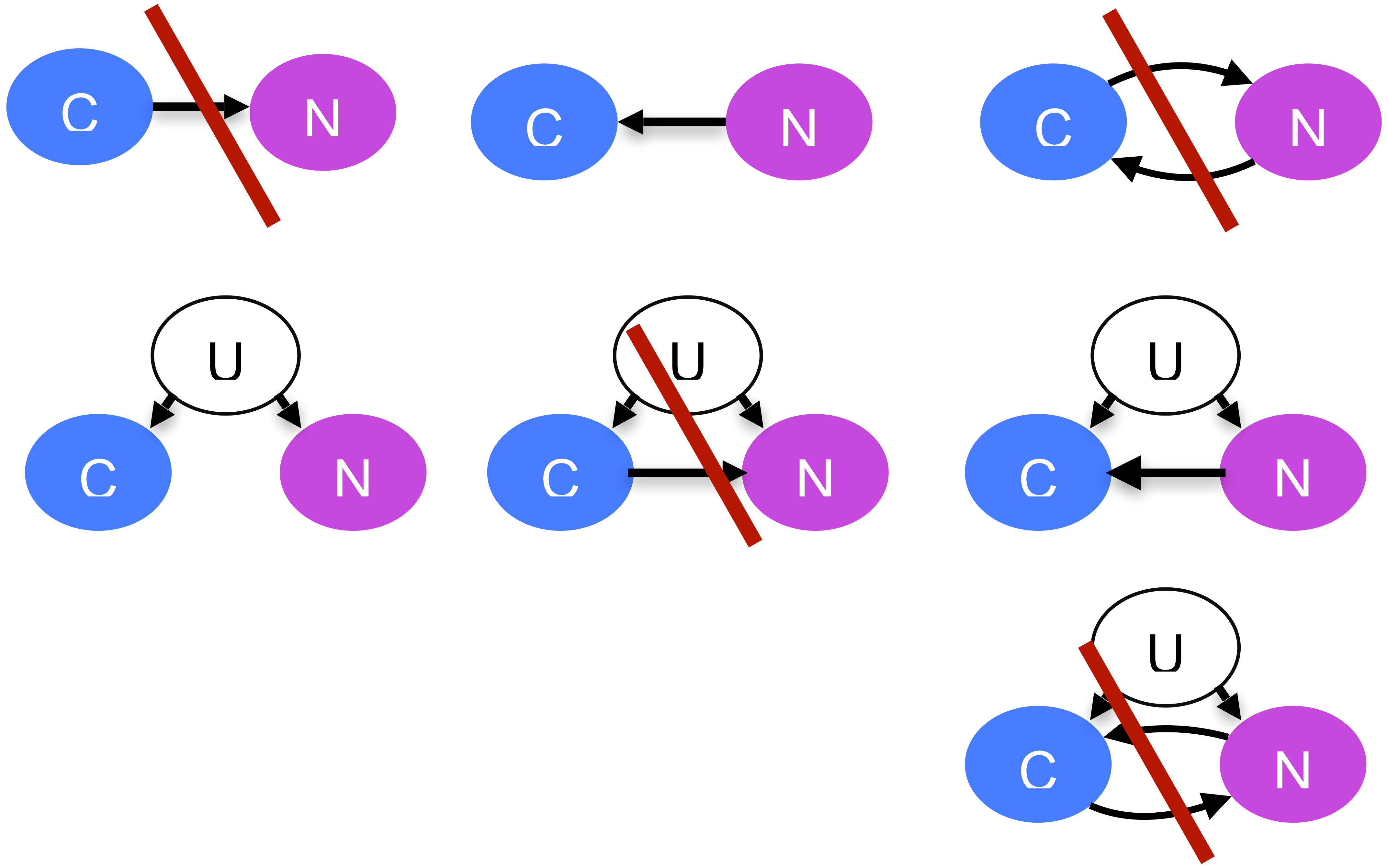
[Messerli, 2012] https://www.nejm.org/doi/full/10.1056/NEJMon1211064

Learning from interventional data - intuition

Hypothetical world: we perform the experiment and see these results:



Chocolate does not cause Nobel prizes



[Messerli, 2012] https://www.nejm.org/doi/full/10.1056/NEJMon1211064

Learning from interventional data - intuition

Hypothetical world: we perform another experiment and see these results:

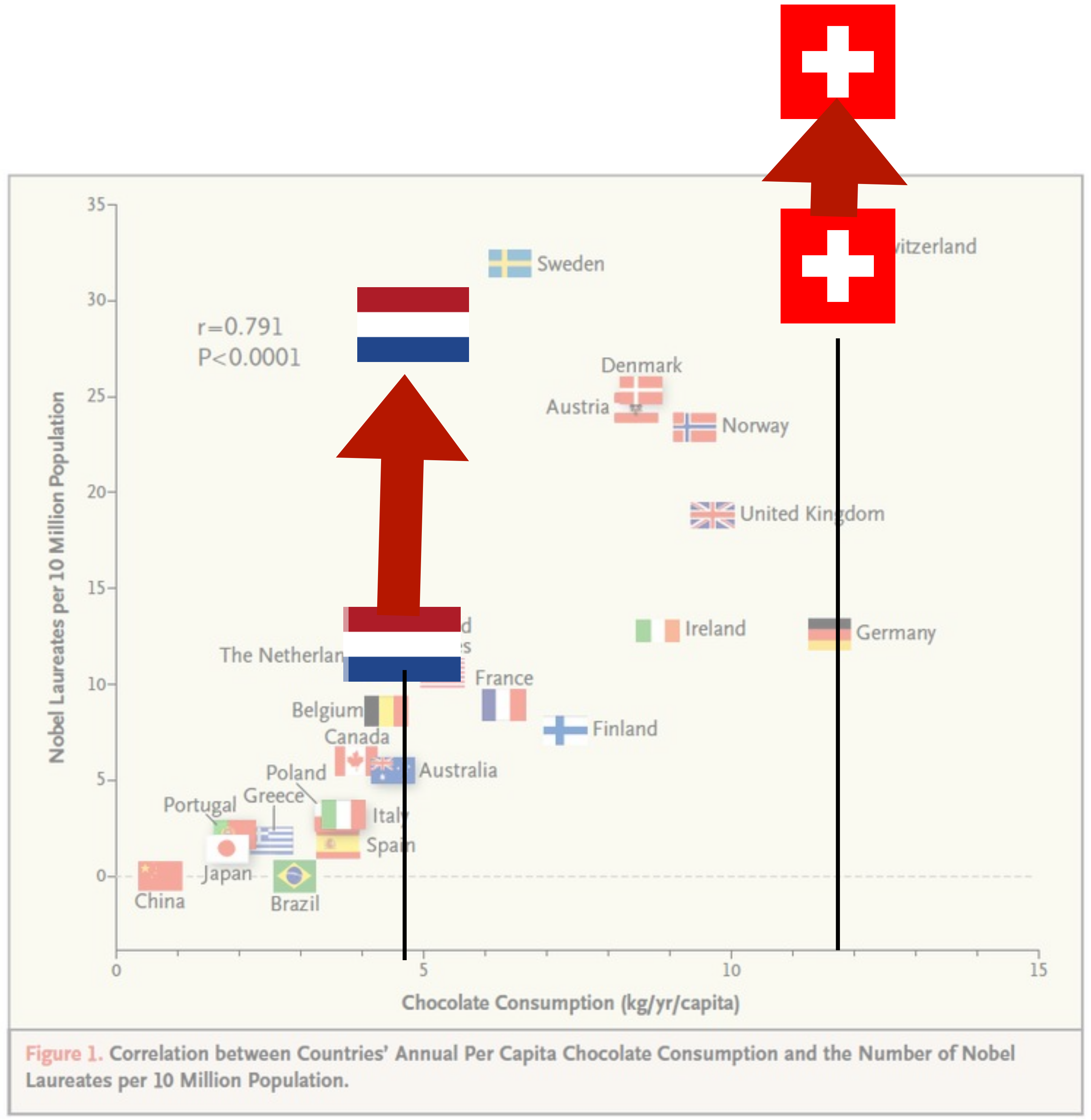
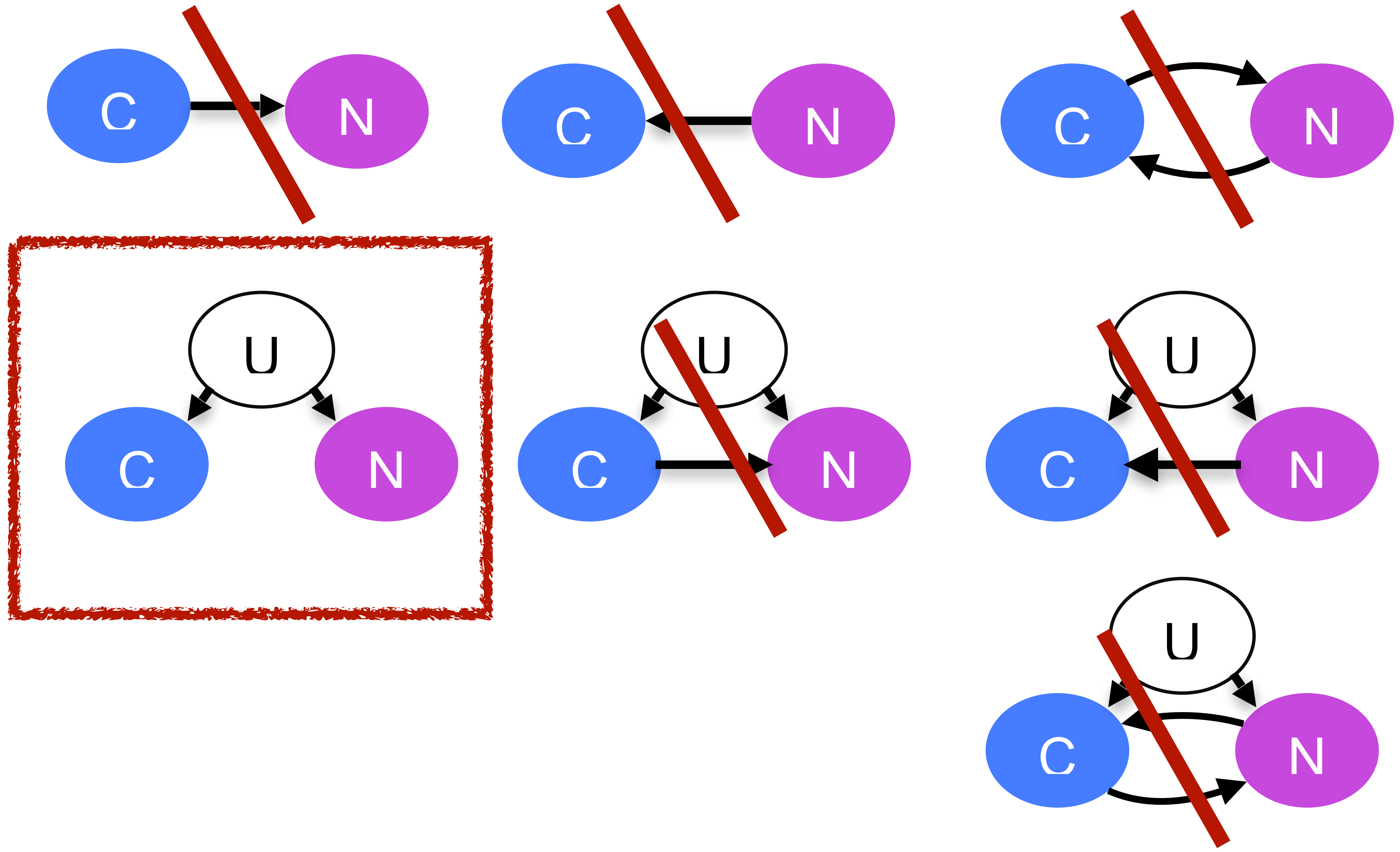


Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.

Nobel does not cause chocolate

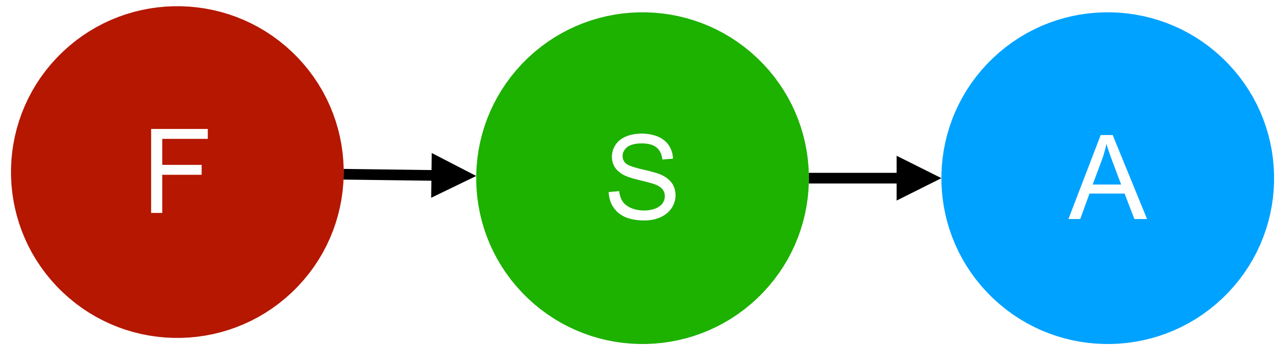


[Messerli, 2012] https://www.nejm.org/doi/full/10.1056/NEJMon1211064

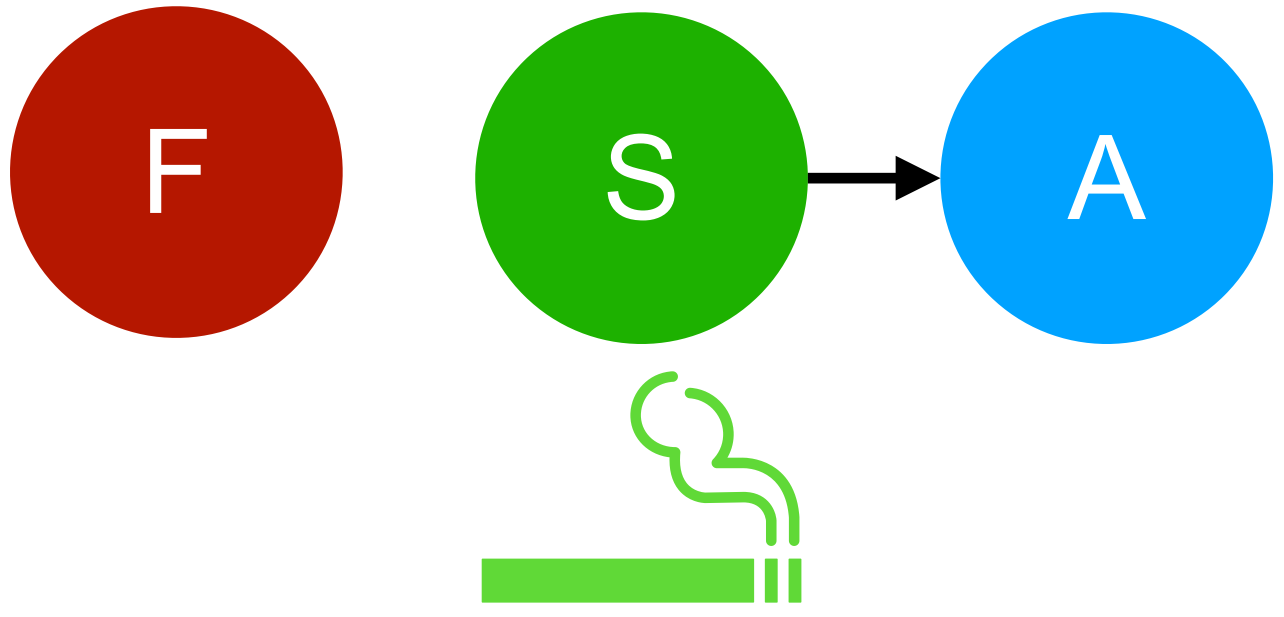
Single-node interventions identify parents and children

- The skeleton (and v-structures) can be identified from observational data
- Intervening on a node i identifies its parents and children:
  - For all j **adjacent** i **in** G**:**
    - If j is not adjacent i in $G_{do(i)}$ then $j \in \mathrm{Pa}(i)$. Otherwise $j \in \mathrm{Ch}(i)$

G



$G_{\mathrm{do}(S=1)}$



$\mathrm{do}(S=1)$

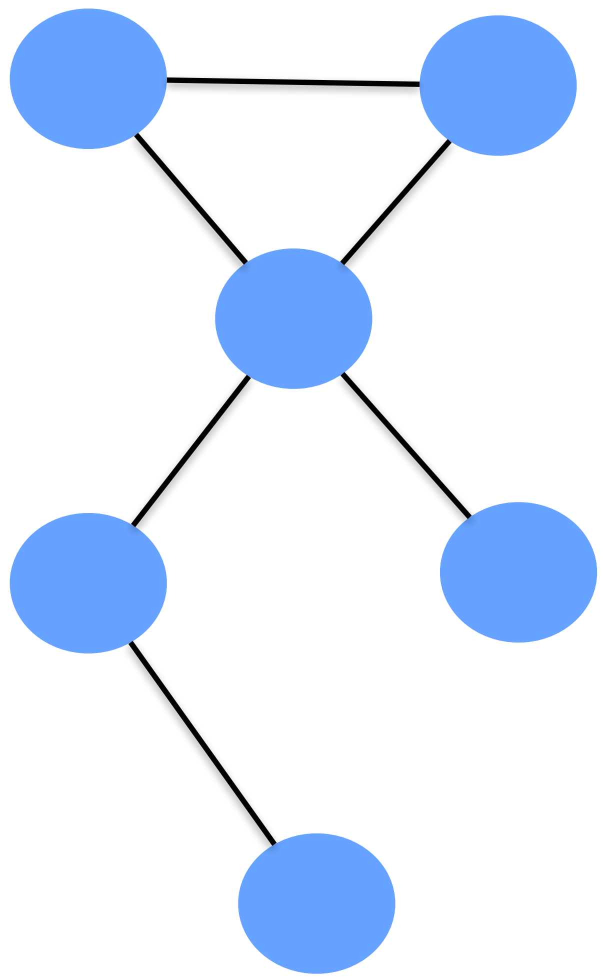
Single-node interventions identify parents and children

- The skeleton (and v-structures) can be identified from observational data
- Intervening on a node i identifies its parents and children:
  - For all j **adjacent** i **in** G**:**
    - If j is not adjacent i in $G_{do(i)}$ then $j \in \mathrm{Pa}(i)$. Otherwise $j \in \mathrm{Ch}(i)$

- Worst case: need $p - 1$ interventions to fully identify the graph (for $p > 2$)
- **Intervention on multiple nodes:** worst case need $O(log_2 p)$ experiments to fully orient the graph **[Hyttinen et al 2013].**

Side note: Intervention design/Experiment selection

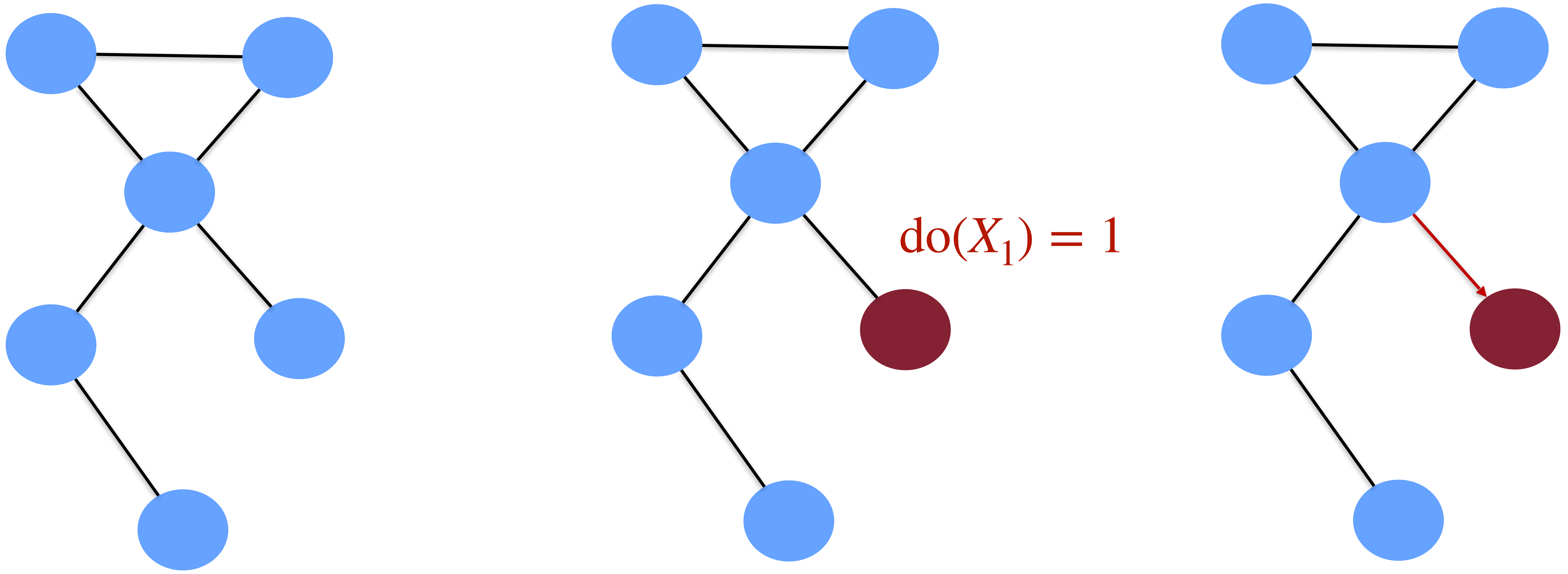
Design a set of interventions, so that we can **accurately** reconstruct **as much as possible** the causal graph **with the least samples,** also when **noisy**



https://papers.nips.cc/paper/2019/hash/5ee5605917626676f6a285fa4c10f7b0-Abstract.html https://arxiv.org/abs/2011.00641

Side note: Intervention design/Experiment selection

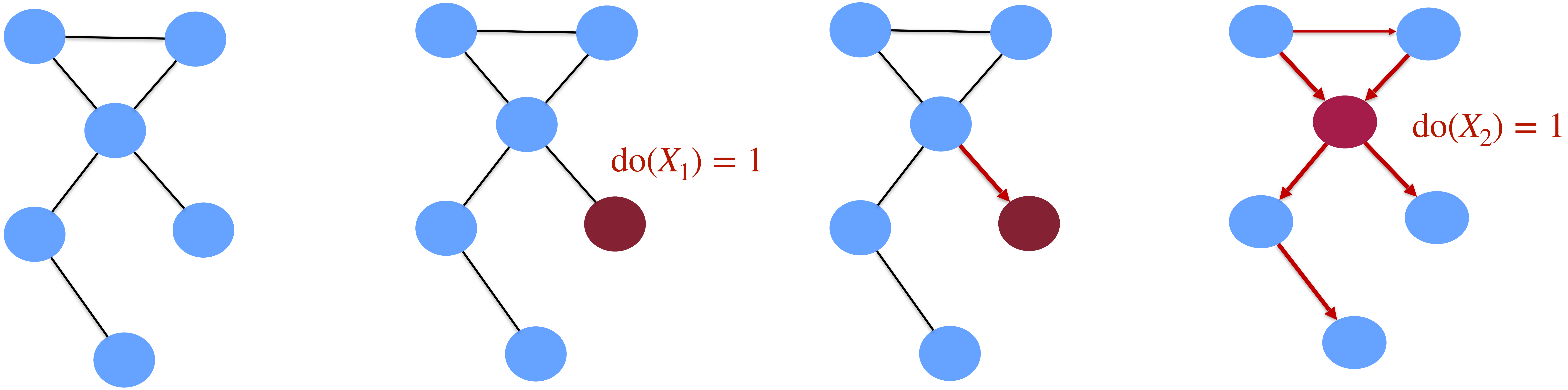
Design a set of interventions, so that we can **accurately** reconstruct **as much as possible** the causal graph **with the least samples,** also when **noisy**



https://papers.nips.cc/paper/2019/hash/5ee5605917626676f6a285fa4c10f7b0-Abstract.html https://arxiv.org/abs/2011.00641

Side note: Intervention design/Experiment selection

Design a set of interventions, so that we can **accurately** reconstruct **as much as possible** the causal graph **with the least samples,** also when **noisy**



https://papers.nips.cc/paper/2019/hash/5ee5605917626676f6a285fa4c10f7b0-Abstract.html https://arxiv.org/abs/2011.00641

Learning from multiple contexts

- Now we cannot decide which intervention to perform (**intervention design**)
  - In intervention design, we have **known intervention targets**, e.g. $\mathrm{do}(S = 1)$

Learning from multiple contexts

- Now we cannot decide which intervention to perform (**intervention design**)
  - In intervention design, we have **known intervention targets**, e.g. $\mathrm{do}(S = 1)$
- Instead, somebody gives us a **set of data from multiple contexts**
  - Possibly **unknown intervention targets**

Learning from multiple contexts

- Now we cannot decide which intervention to perform (**intervention design**)
  - In intervention design, we have **known intervention targets**, e.g. $\mathrm{do}(S = 1)$
- Instead, somebody gives us a **set of data from multiple contexts**
  - Possibly **unknown intervention targets**
  - Possibly **soft interventions** instead of **perfect interventions**

Perfect vs soft interventions

- Recap: we introduced an operator that can represent a **hypothetical intervention** on the whole population, i.e. a perturbation of the system:

$$\mathrm{do}(X_i = x_i) \text{ which changes } P(X_i \,|\, X_{\mathrm{Pa}(i)}) \to \mathbf{1}(X_i = x_i)$$

- This is called a **perfect** (or surgical) **intervention**

- There are also other types of intervention, e.g. **soft interventions which change $P(X_i \,|\, X_{\mathrm{Pa}(i)}) \to P'(X_i \,|\, X_{\mathrm{Pa}(i)})$,** which might not change the graph

Example 3.2 in Elements of Causal Inference

$$\begin{cases} X \leftarrow \epsilon_x \\ Y \leftarrow 4 \cdot X + \epsilon_Y \end{cases}$$

$$\epsilon_X, \epsilon_Y \sim \mathcal{N}(0,1)$$

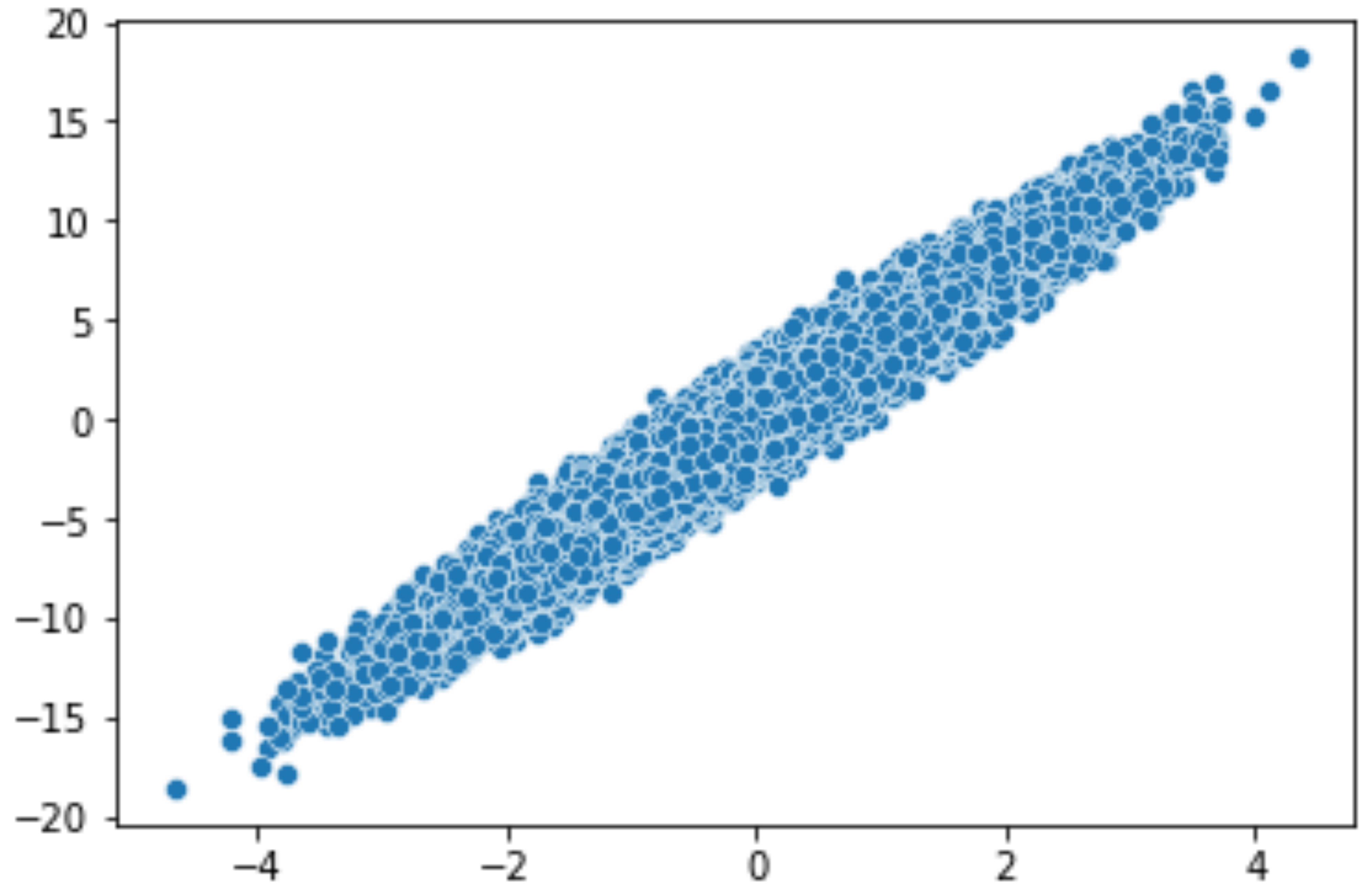
$$P(X) = \mathcal{N}(0,1)$$

$$P(Y) = \mathcal{N}(0,17)$$

```
# We need a lot of samples to plot the conditional distribution:
n_samples=100000
```

```
x = randn(n_samples)
y = 4 * x + randn(n_samples)
# plot P(X,Y)
sns.scatterplot(x=x,y=y)
```

<AxesSubplot:>



Example 3.2 in Elements of Causal Inference

$$\begin{cases} X \leftarrow \epsilon_X \\ Y \leftarrow \cancel{4 \cdot X + \epsilon_Y} \quad Y \leftarrow 4 \end{cases}$$

$$\epsilon_X, \epsilon_Y \sim \mathcal{N}(0,1)$$

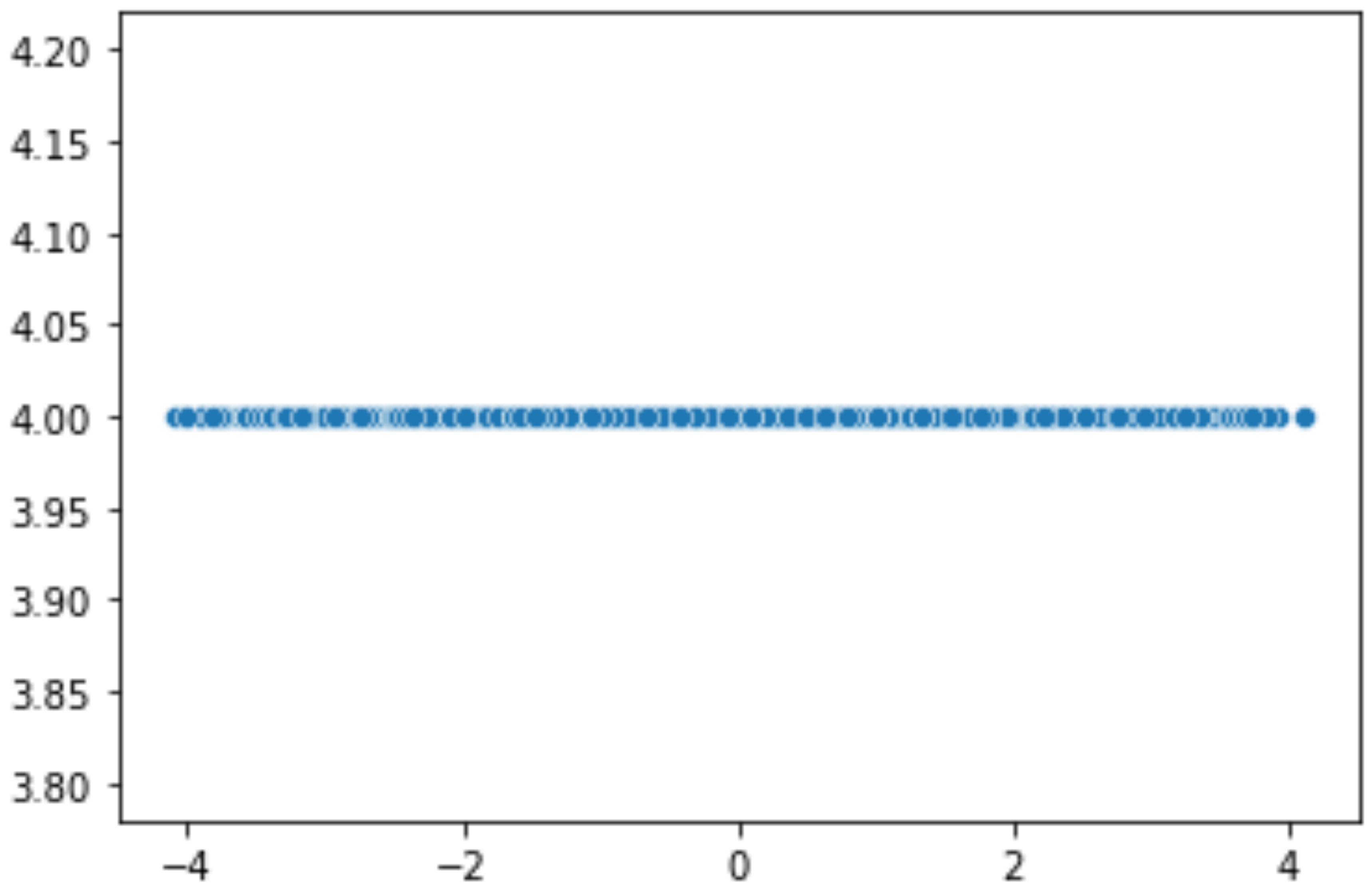
$$P(X \mid do(Y=4)) = N(0,1)$$

$$P(Y \mid do(Y=4)) = \begin{cases} 1 & y = 4 \\ 0 & y \neq 4 \end{cases}$$

```
x_do_y = randn(n_samples)
y_do_y = 4

# plot P(X,Y |do(X=2))
sns.scatterplot(x=x_do_y,y=y_do_y)
```

<AxesSubplot:>



Soft interventions, shift interventions

$$\begin{cases} X \leftarrow \epsilon_x \\ Y \leftarrow 4 \cdot X + \epsilon_Y \end{cases}$$

$$\epsilon_X, \epsilon_Y \sim \mathcal{N}(0,1)$$

$$P(X) = \mathcal{N}(0,1)$$

$$P(Y) = \mathcal{N}(0,17)$$

$$\begin{cases} X \leftarrow \epsilon_x \\ Y \leftarrow \cancel{4 \cdot X} + \epsilon_Y \end{cases}$$

$$Y \leftarrow 2 \cdot X + \varepsilon_Y$$

$$\begin{cases} X \leftarrow \epsilon_x \\ Y \leftarrow 4 \cdot X + \epsilon_Y + \varepsilon' \end{cases}$$

$$Pa_{G^{do(Y)}}(Y) \subseteq Pa_G(Y)$$

Causal parents as example of robust prediction

$$\begin{cases} X_1 = \varepsilon_1 \\ Y = X_1 + \varepsilon_Y \\ X_2 = Y + \varepsilon_{X_2} \end{cases}$$

$\varepsilon_1, \varepsilon_Y \sim N(0,1)$, $\varepsilon_{X_2} \sim N(0, 0.01)$

$X_1 \to Y \to X_2$

Causal parents as example of robust prediction

$$\begin{cases} X_1 = \varepsilon_1 \\ Y = X_1 + \varepsilon_Y \\ X_2 = Y + \varepsilon_{X_2} \end{cases}$$

$X_1 \rightarrow Y \rightarrow X_2$

$\varepsilon_1, \varepsilon_Y \sim N(0,1)$, $\varepsilon_{X_2} \sim N(0, 0.01)$

M1: $Y \sim X_1$

M2: $Y \sim X_2$

M2 has smaller error

Causal parents as example of robust prediction

$$\begin{cases} X_1 = \varepsilon_1 \\ Y = X_1 + \varepsilon_Y \\ X_2 = Y + \varepsilon_{X_2} \end{cases}$$

$\varepsilon_1, \varepsilon_Y \sim N(0,1)$, $\varepsilon_{X_2} \sim N(0, 0.01)$

M1: $Y \sim X_1$ M2: $Y \sim X_2$

$X_1 \to Y \to X_2$

$X_1 \to Y \quad X_2$ $do(X_2)$

M2 has smaller error

but it fails in $do(X_2)$

Causal parents as example of robust prediction

$$\begin{cases} X_1 = \varepsilon_1 \\ Y = X_1 + \varepsilon_Y \\ X_2 = Y + \varepsilon_{X_2} \end{cases}$$

$X_1 \to Y \to X_2$

$\varepsilon_1, \varepsilon_Y \sim N(0,1)$, $\varepsilon_{X_2} \sim N(0, 0.01)$

Using causal parents of Y helps with DISTRIBUTION SHIFTS

M1: $Y \sim X_1$ M2: $Y \sim X_2$

Causal discovery simplified overview

Constraint-based causal discovery

- Observational data
- Output: MEC

- SGS, PC

Score-based causal discovery

- Output: MEC

- GES, MMHC

Restricted models

- Nonlinear additive

- Observational data
- Output: DAG

- RESIT, LINGAM

Interventional causal discovery / causal invariance

- Output: parents of Y, I-MEC

- ICP, JCI

Causal inference using invariant prediction: identification and confidence intervals

Jonas Peters, Peter Bühlmann, Nicolai Meinshausen

https://arxiv.org/pdf/1501.01332.pdf

Invariant Causal Prediction (ICP) [Peters et al 2016]

- Given a target variable Y and features $(X_1, \ldots, X_p)$, we want to **find the causal parents of Y, i.e. $\mathrm{Pa}(Y)$**

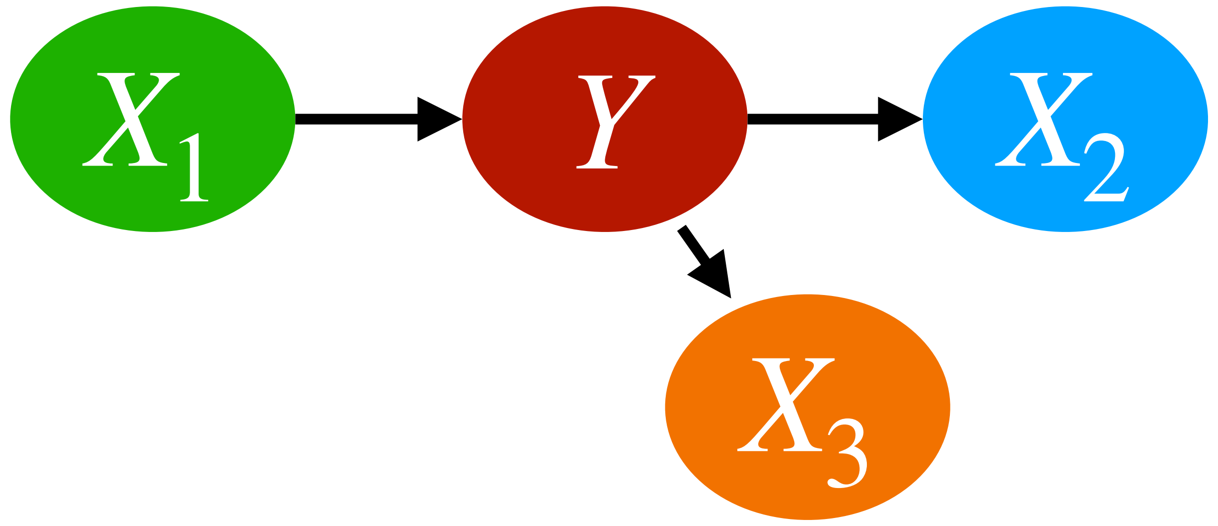
Invariant Causal Prediction (ICP) [Peters et al 2016]

- Given a target variable Y and features $(X_1, \ldots, X_p)$, we want to **find the causal parents of Y, i.e. $\mathrm{Pa}(Y)$**
- We assume we have access to **a set of different environments $\mathscr{E}$** (e.g. interventional or observational data), s.t. for $e \in \mathscr{E}$, $(X_1^e, \ldots, X_p^e, Y^e) \sim P^e$

Multiple environments

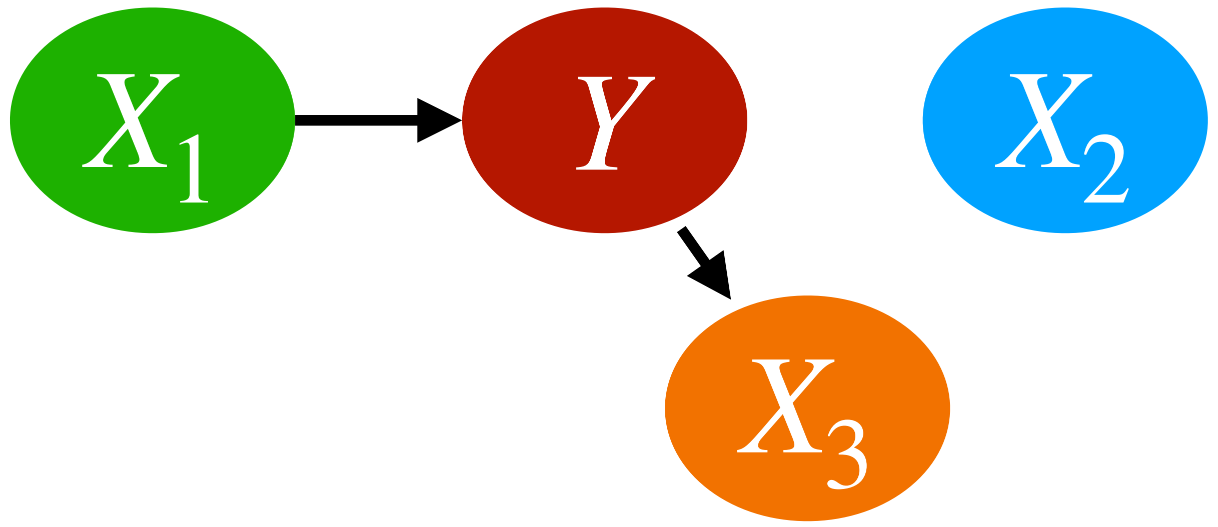
$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$$E = 0$$



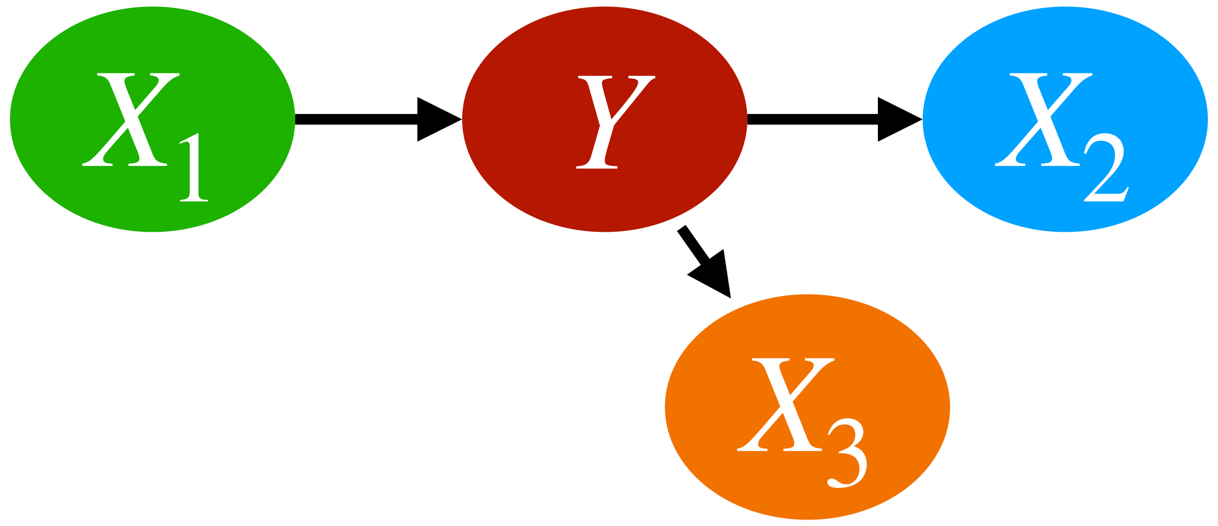
$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 1 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$$E = 1$$



$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 10Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$$E = 2$$



Invariant Causal Prediction (ICP) [Peters et al 2016]

- Given a target variable Y and features $(X_1, \ldots, X_p)$, we want to **find the causal parents of Y, i.e. $\mathrm{Pa}(Y)$**
- We assume we have access to **a set of different environments $\mathscr{E}$** (e.g. interventional or observational data), s.t. for $e \in \mathscr{E}$, $(X_1^e, \ldots, X_p^e, Y^e) \sim P^e$
- We further assume that in **none of the environments Y is intervened upon**
  - We can then show that $e, f \in \mathscr{E} : P^e(Y^e \,|\, \mathrm{Pa}(Y^e)) = P^f(Y^f \,|\, \mathrm{Pa}(Y^f))$
- We represent the environment index with E

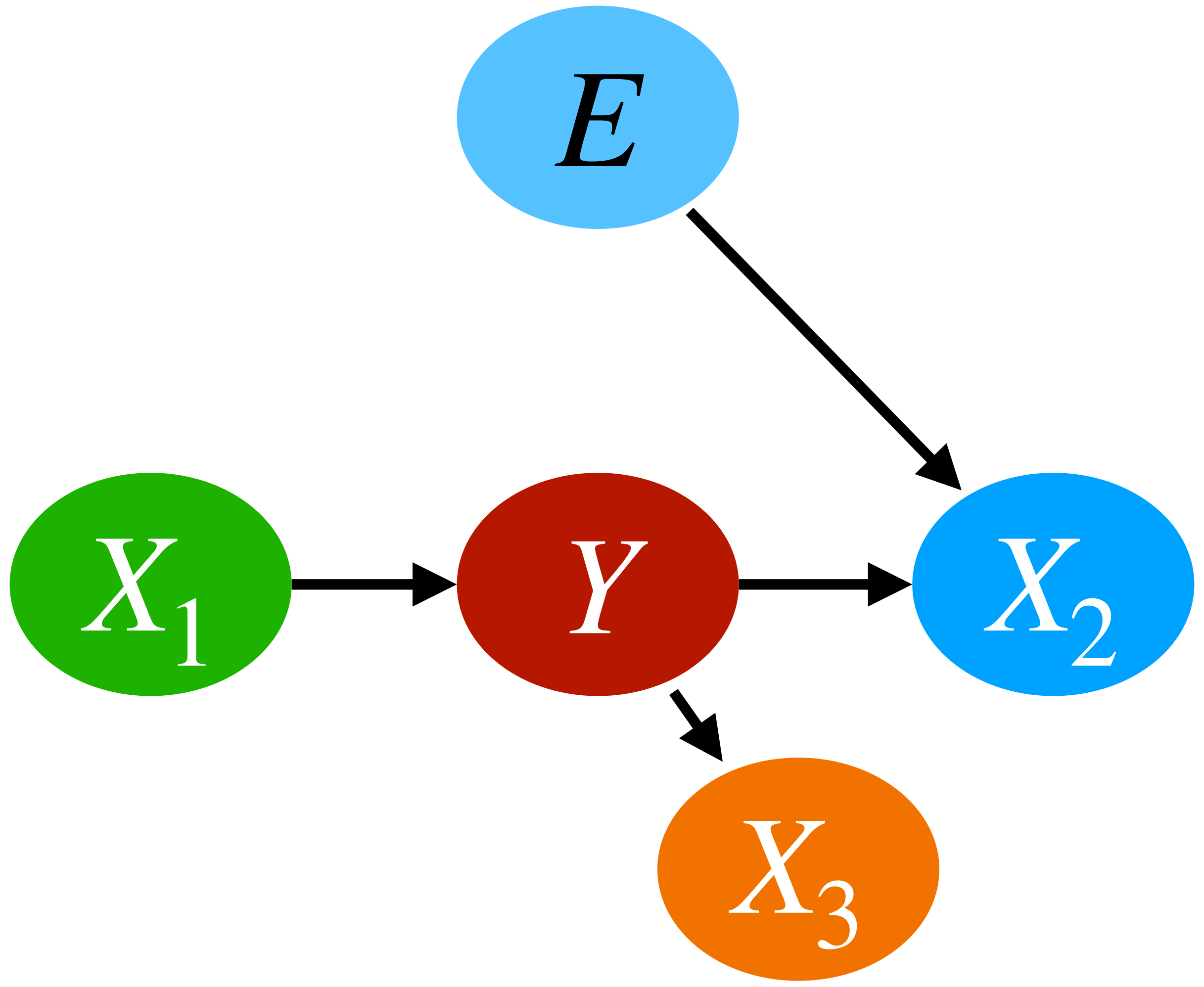
Multiple environments in a single SCM

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 0$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 1 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 1$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 10Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 2$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = \begin{cases} -2Y + \epsilon_2 & \text{if } E = 0 \\ 1 & \text{if } E = 1 \\ 10Y + \epsilon_2 & \text{if } E = 2 \end{cases} \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$



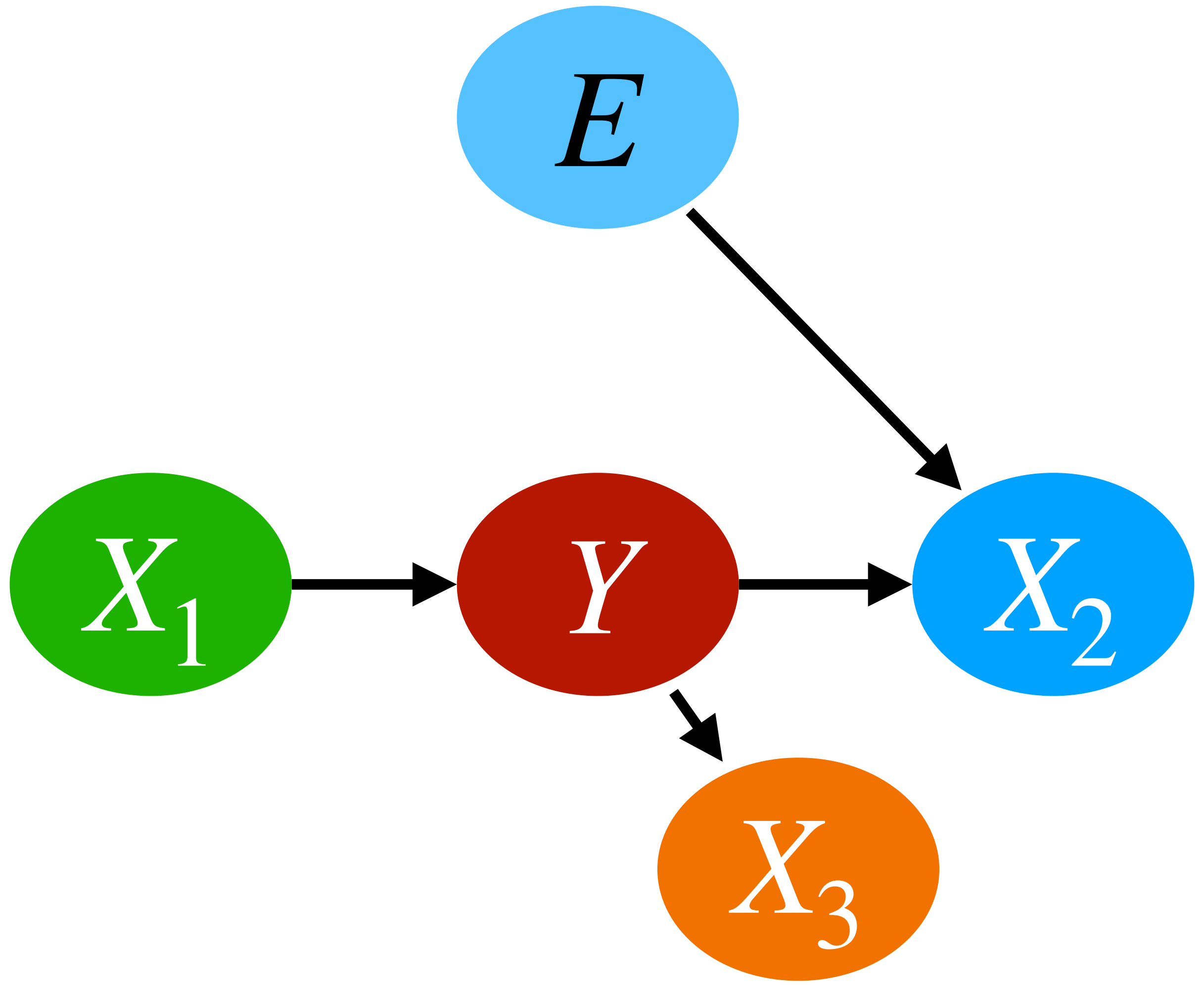
Multiple environments in a single SCM

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 0$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 1 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 1$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 10Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 2$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = \begin{cases} -2Y + \epsilon_2 & \text{if } E = 0 \\ 1 & \text{if } E = 1 \\ 10Y + \epsilon_2 & \text{if } E = 2 \end{cases} \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$



$$P^e(Y^e \,|\, \mathrm{Pa}(Y^e)) = P^f(Y^f \,|\, \mathrm{Pa}(Y^f))$$

$$E \perp\!\!\!\perp Y \,|\, \mathrm{Pa}(Y) \iff E \perp_d Y \,|\, \mathrm{Pa}(Y)$$

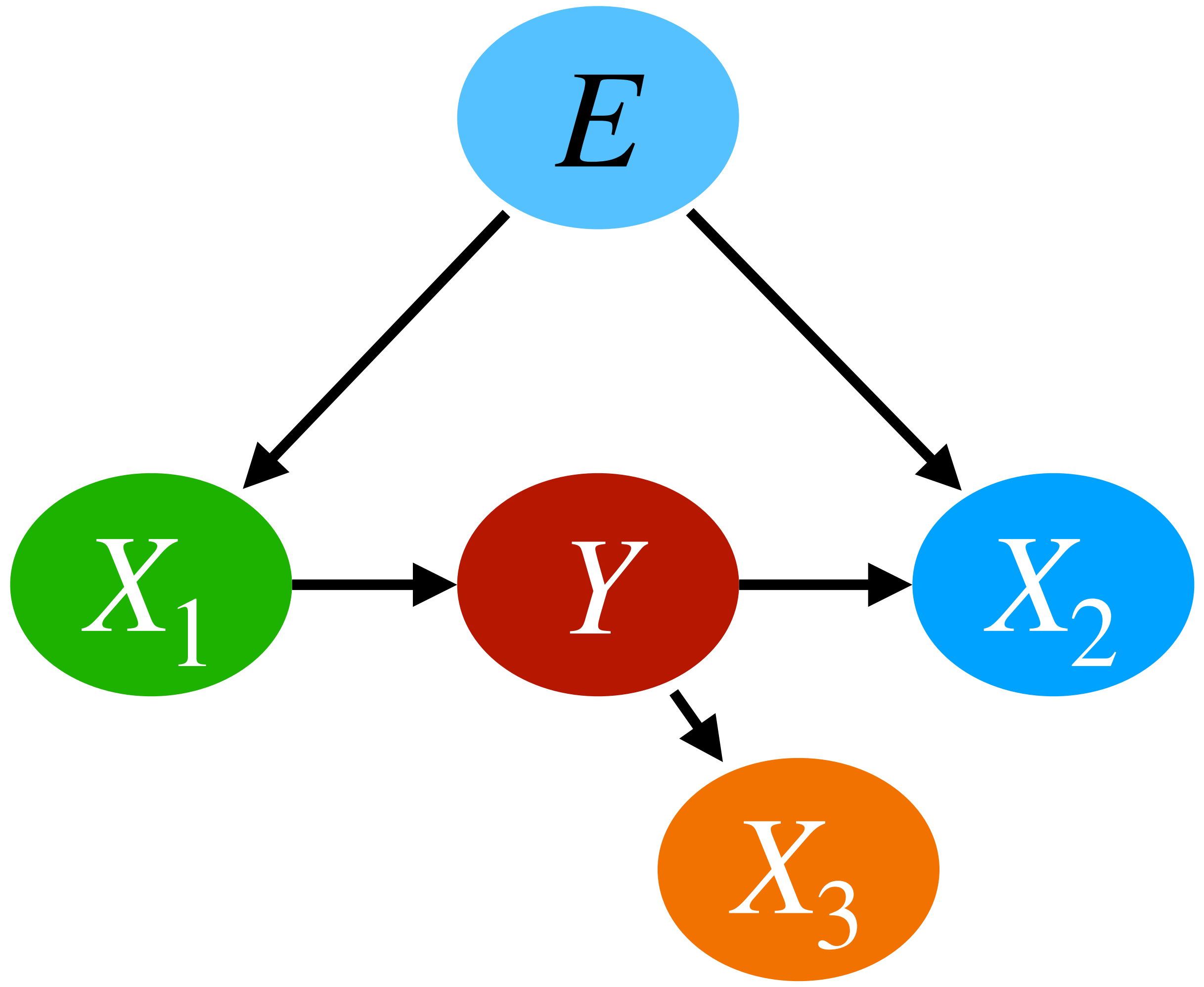
$$E \perp_d Y \,|\, X_1$$

Multiple environments in a single SCM

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 0$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 100 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 1 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 1$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 10Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 2$$



$$\mathbf{S} = \mathrm{Pa}(Y) \implies E \perp_d Y \mid \mathbf{S}$$

$$E \perp_d Y \mid \mathbf{S} \implies \mathbf{S} = \mathrm{Pa}(Y)?$$

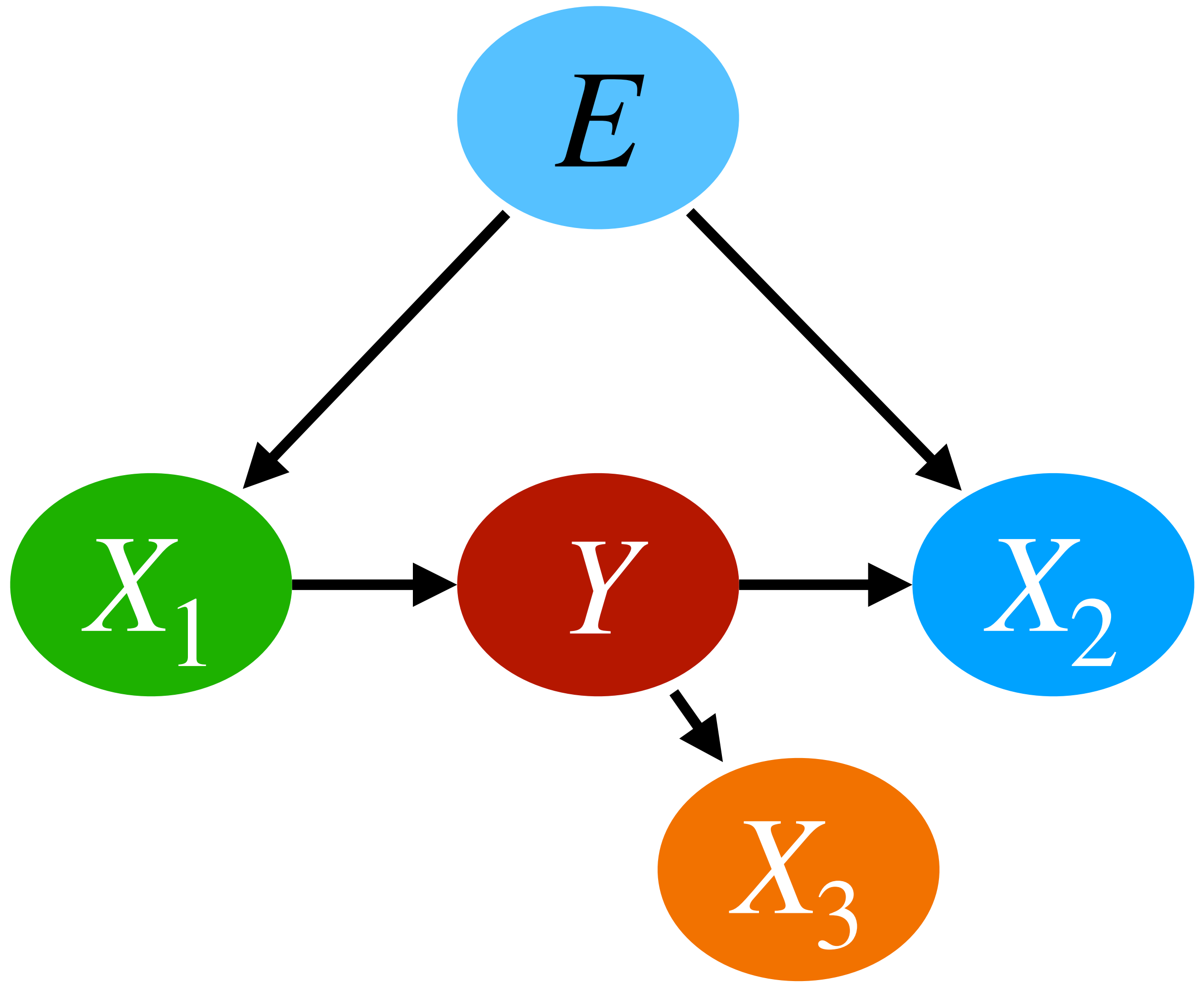
$$E \perp_d Y \mid X_1$$

Multiple environments in a single SCM

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 0$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 100 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 1 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 1$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 10Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 2$$



$$\mathbf{S} = \mathrm{Pa}(Y) \implies E \perp_d Y | \mathbf{S}$$

$$E \perp_d Y | \mathbf{S} \not\Longrightarrow \mathbf{S} = \mathrm{Pa}(Y)$$

$$E \perp_d Y | X_1$$

Other subset of nodes also satisfy this relationship

$$E \perp_d Y | \{X_1, X_3\}$$

Invariant Causal Prediction (ICP)

- We assume we have access to **a set of different environments** $\mathscr{E}$ (e.g. interventional or observational data), s.t. for $e \in \mathscr{E}$, $(X_1^e, \ldots, X_p^e, Y^e) \sim P^e$
- We further assume that in **none of the environments** Y **is intervened upon**

Invariant Causal Prediction (ICP)

- We assume we have access to **a set of different environments** $\mathscr{E}$ (e.g. interventional or observational data), s.t. for $e \in \mathscr{E}$, $(X_1^e, \ldots, X_p^e, Y^e) \sim P^e$
- We further assume that in **none of the environments** Y **is intervened upon**
- We represent the environment index with E
- If there are **no latent confounders,** one can prove that:

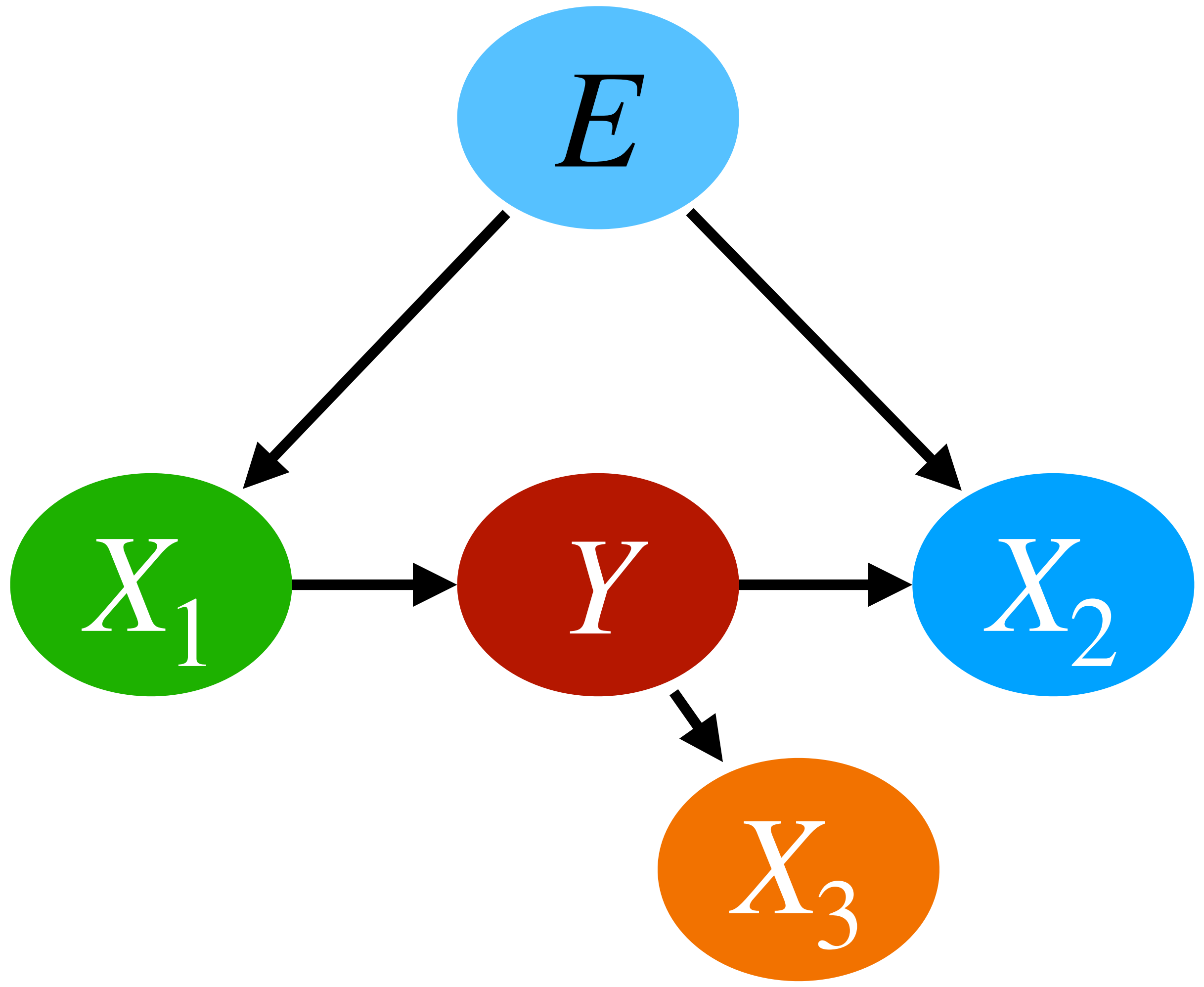
$$\bigcap_{\mathbf{S}\subseteq\{1,\ldots,p\} \text{ s.t. } E \perp\!\!\!\perp Y | \mathbf{S}} \mathbf{S} \subseteq \mathrm{Pa}(Y)$$

Invariant Causal Prediction example

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 0$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 100 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 1 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 1$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 10Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \quad E = 2$$



$$\bigcap_{\mathbf{S} \subseteq \{1,\ldots,p\} \text{ s.t. } E \perp\!\!\!\perp Y | \mathbf{S}} \mathbf{S} \subseteq \mathrm{Pa}(Y)$$

$E \perp_d Y | X_1$

$E \perp_d Y | \{X_1, X_3\}$

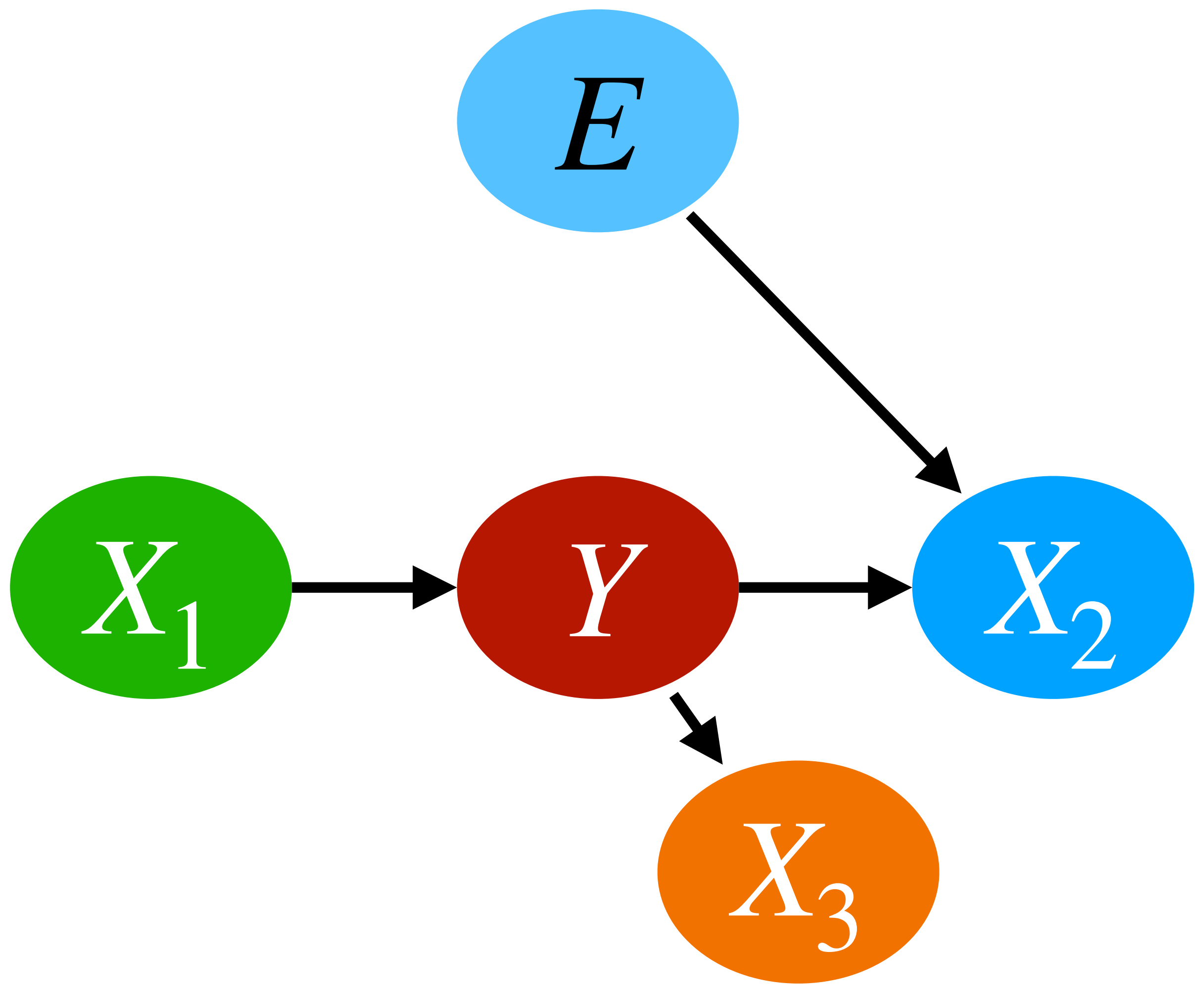
$$\{X_1\} \cap \{X_1, X_3\} = \{X_1\} = \mathrm{Pa}(Y)$$

Invariant Causal Prediction example 2

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \qquad E = 0$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 1 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \qquad E = 1$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 10Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases} \qquad E = 2$$



$$\bigcap_{\mathbf{S} \subseteq \{1,\ldots,p\} \text{ s.t. } E \perp\!\!\!\perp Y | \mathbf{S}} \mathbf{S} \subseteq \mathrm{Pa}(Y)$$

$E \perp_d Y | X_1$

$E \perp_d Y | \{X_1, X_3\}$

$E \perp_d Y$

$$\{X_1\} \cap \{X_1, X_3\} \cap \varnothing = \varnothing \subseteq \mathrm{Pa}(Y)$$

ICP finds SUBSETS of parents

ICP improves with more environments

$E \rightarrow X_1 \rightarrow X_2 \rightarrow Y$

$$\left.\begin{array}{l} E \perp\!\!\!\perp Y \mid X_1 \\ E \perp\!\!\!\perp Y \mid X_2 \\ E \perp\!\!\!\perp Y \mid X_1, X_2 \end{array}\right\} \cap \quad \phi$$

ICP on these environments finds empty set

+ new enviroment e3

$$X_2 = g^{new}(X_1, \varepsilon_2)$$

ICP improves with more environments

$E \to X_1 \to X_2 \to Y$

$$\left.\begin{array}{l} E \perp\!\!\!\perp Y \mid X_1 \\ E \perp\!\!\!\perp Y \mid X_2 \\ E \perp\!\!\!\perp Y \mid X_1, X_2 \end{array}\right\} \cap \quad \phi$$

ICP on these environments finds empty set

+ new enviroment e3

$X_2 = g^{new}(X_1, \varepsilon_2)$

$E \to X_1 \to X_2 \to Y$, $E \to X_2$

$$\left.\begin{array}{l} E \perp\!\!\!\perp Y \mid X_2 \\ E \perp\!\!\!\perp Y \mid X_2, X_1 \end{array}\right. \cap \quad X_2 \in Pa(Y)$$

ICP on these environments (including new) finds X_2

ICP improves with more environments

$$E \to X_1 \to X_2 \to Y$$

$$\left.\begin{matrix} E \perp\!\!\!\perp Y \mid X_1 \\ E \perp\!\!\!\perp Y \mid X_2 \\ E \perp\!\!\!\perp Y \mid X_1, X_2 \end{matrix}\right\} \cap \quad \phi$$

ICP on these environments finds empty set

+ new enviroment e3

$$X_2 = f^{new}(X_1, \varepsilon_2)$$

$$E \to X_1 \to X_2 \to Y, \quad E \to X_2$$

ICP on these environments (including new) finds X_2

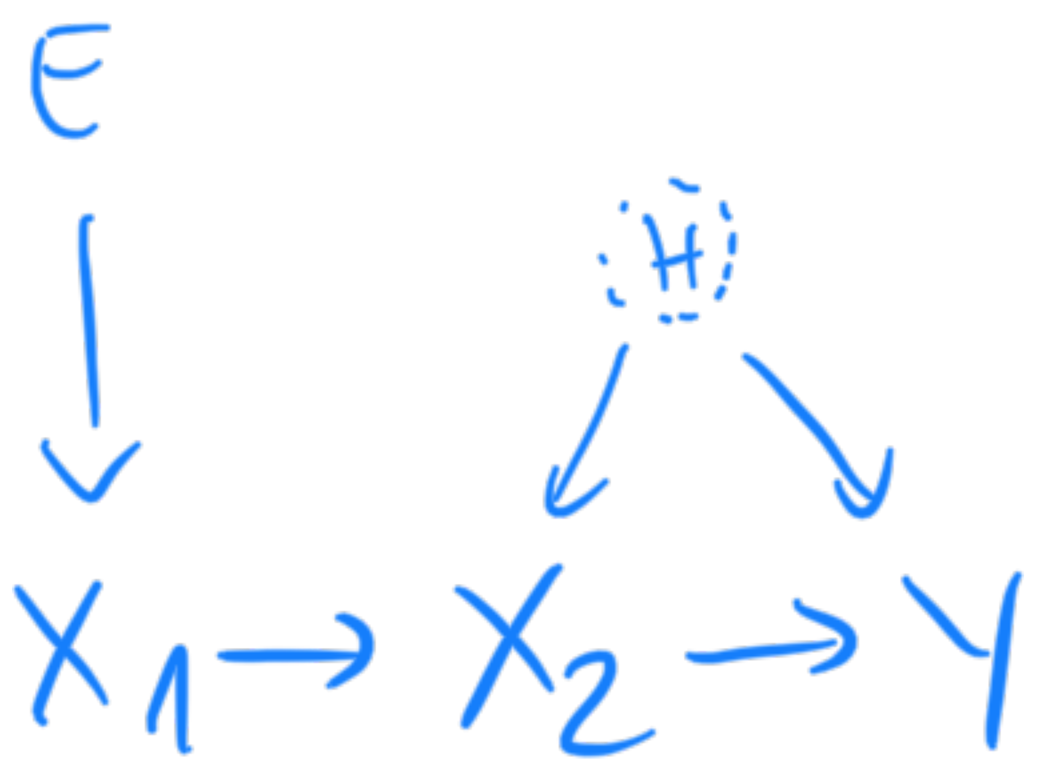
$$\left.\begin{matrix} E \perp\!\!\!\perp Y \mid X_2 \\ E \perp\!\!\!\perp Y \mid X_2, X_1 \end{matrix}\right. \cap \quad X_2 \in Pa(Y)$$

If all variables are caused by E (so we see enough environments), then we find ALL parents

Invariant Causal Prediction - latent confounders

- If there are latent confounders, one can prove that:

$$\bigcap_{\mathbf{S}\subseteq\{1,\ldots,p\} \text{ s.t. } Y \perp\!\!\!\perp E \mid \mathbf{S}} \mathbf{S} \subseteq \mathrm{Anc}(Y)$$



$Y \not\perp\!\!\!\perp E \mid X_2$

$Y \perp\!\!\!\perp E \mid X_1$

$Y \perp\!\!\!\perp E \mid X_1, X_2$

$\Rightarrow X_1 \in \mathrm{Anc}(Y)$

Learning from multiple contexts

- Now we cannot decide which intervention to perform (**intervention design**)
  - We then also have **known intervention targets**, e.g. $\mathrm{do}(S = 1)$
- Instead, somebody gives us a **set of data from multiple contexts**
  - Possibly **unknown intervention targets**
  - Possibly **soft interventions** instead of **perfect interventions**

ICP finds subsets of parents, what about finding (an equivalence class of) the causal graph?

Joint Causal Inference from Multiple Contexts

Joris M. Mooij, Sara Magliacane, Tom Claassen

- We represent different distributions (including interventional) as an **unknown joint causal graph** *(possibly cyclic or with latent confounders)*

- We **add context variables** so we can **disentangle** changes in distribution across the datasets

https://arxiv.org/abs/1611.10351

Joint Causal Inference intuition

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

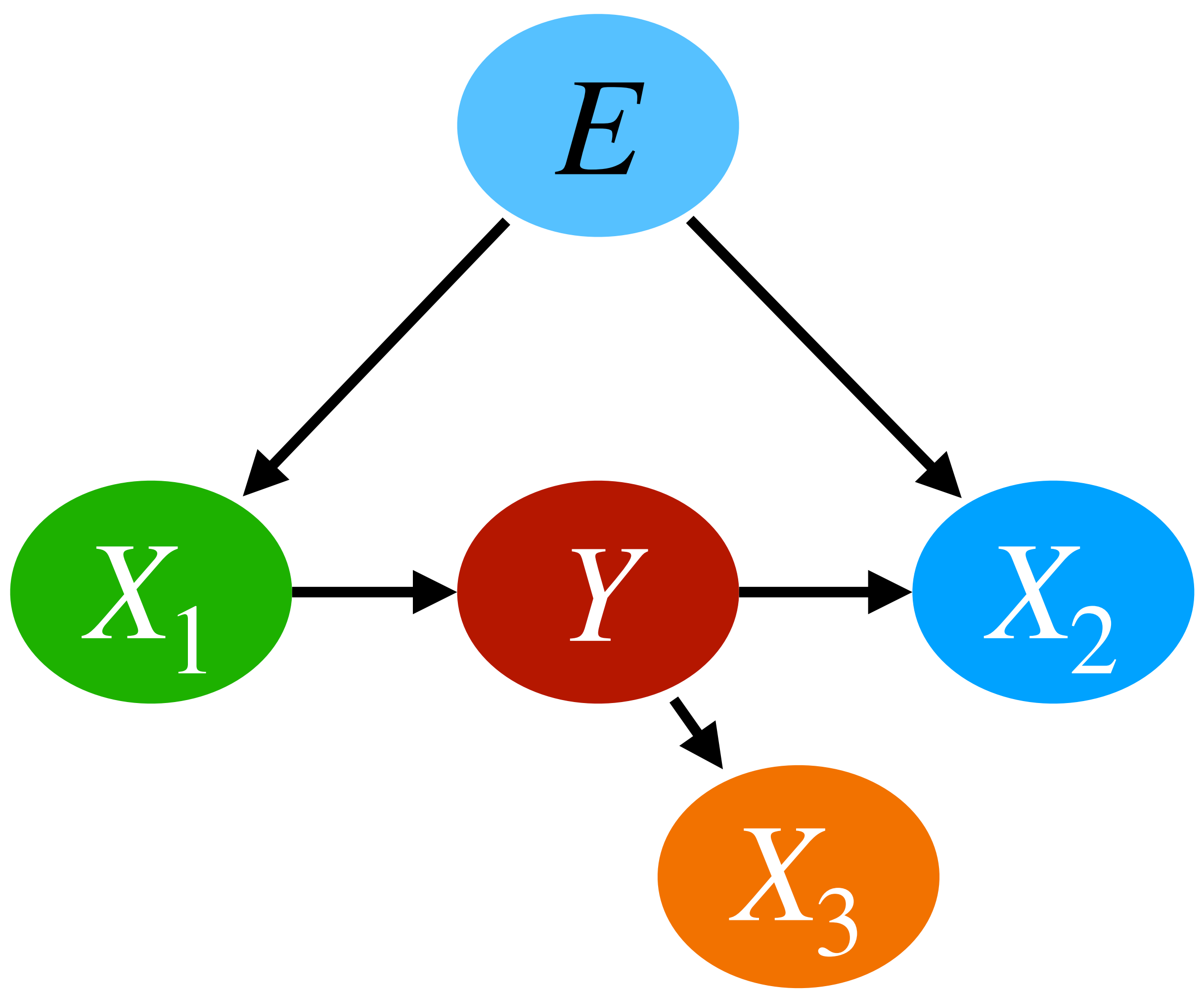
$E = 0$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 100 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$E = 1$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 10Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$E = 2$



Joint Causal Inference intuition

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$E = 0$

$C_1 = 0$
$C_2 = 0$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 100 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

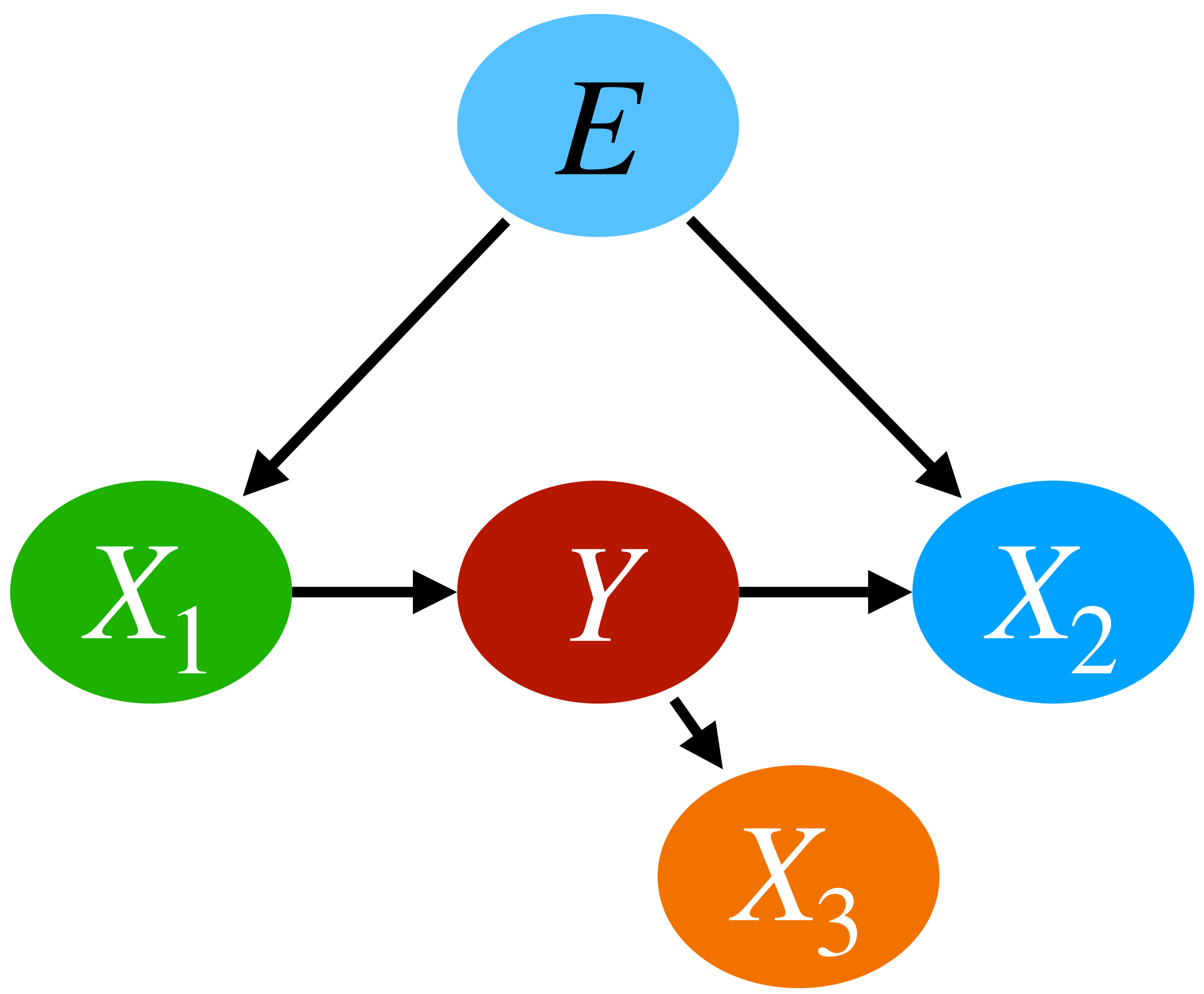
$E = 1$

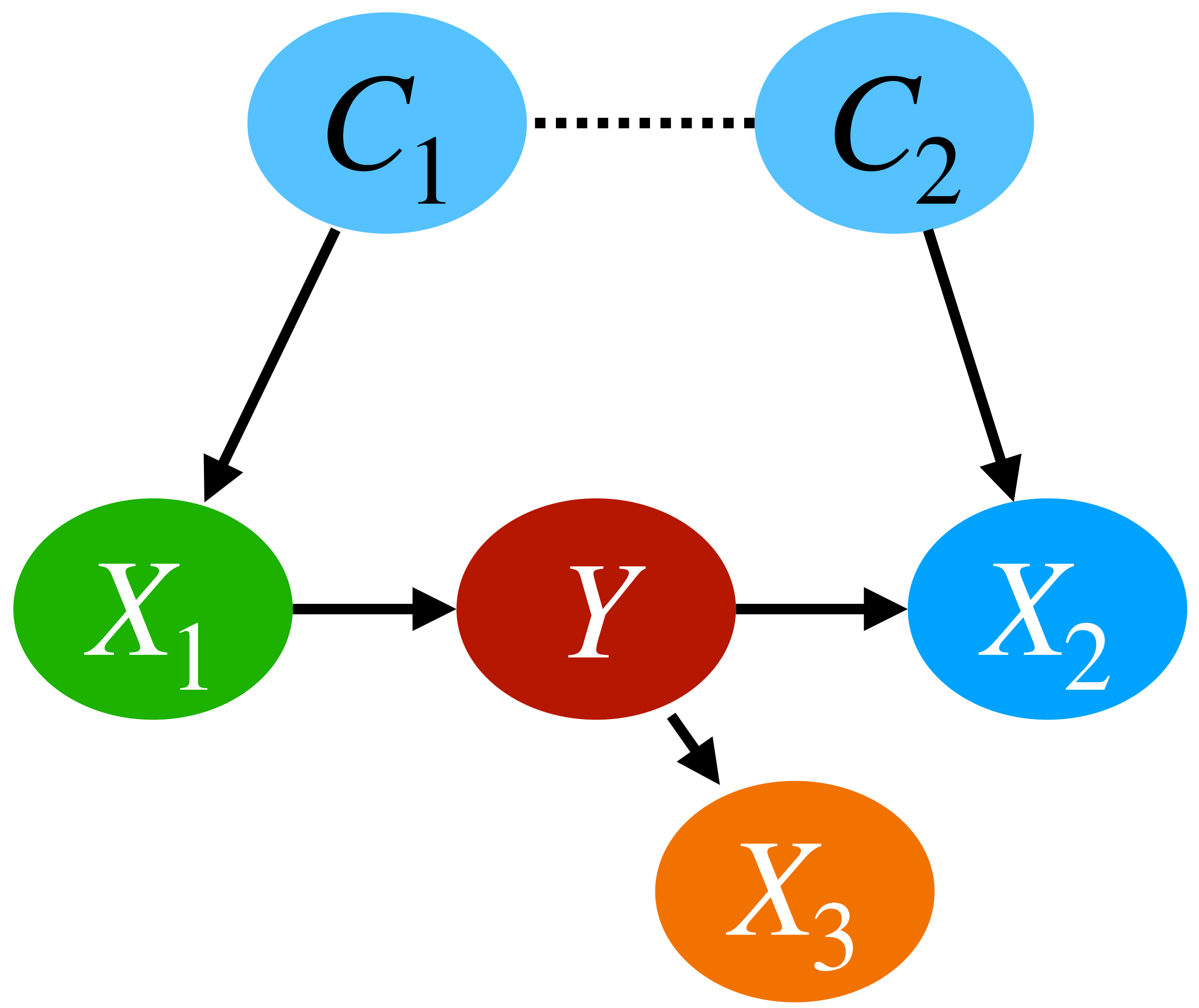
$C_1 = 1$
$C_2 = 0$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 10Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$E = 2$

$C_1 = 0$
$C_2 = 1$





Adding context variables C_1 and C_2 helps disentangle the changes in each environment

Joint Causal Inference intuition

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$E = 0$

$C_1 = 0$
$C_2 = 0$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 100 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$E = 1$

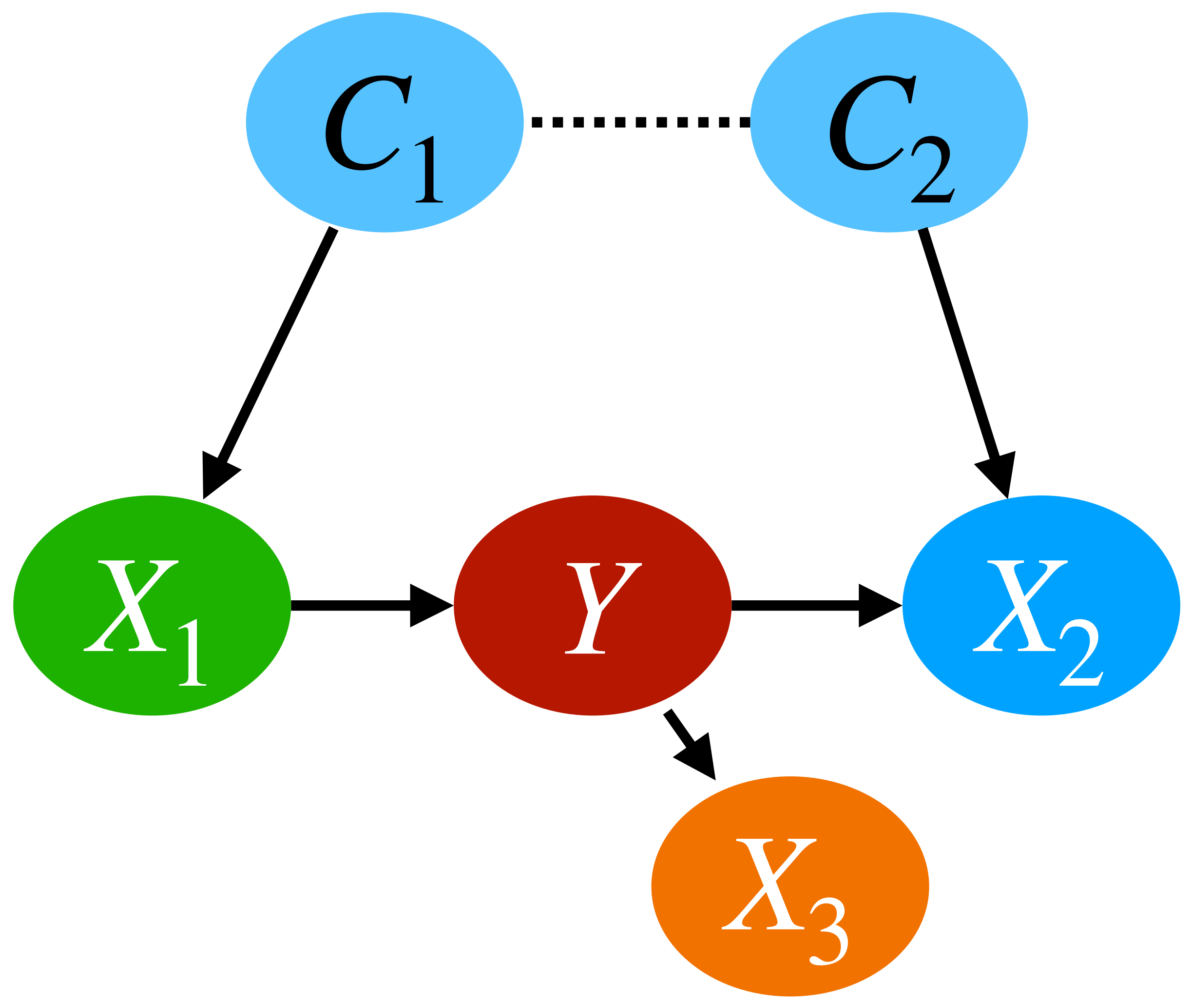
$C_1 = 1$
$C_2 = 0$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 10Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$E = 2$

$C_1 = 0$
$C_2 = 1$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = \begin{cases} 10 + \epsilon_1 \text{ if } C_1 = 0 \\ 100 + \epsilon_1 \text{ if } C_1 = 1 \end{cases} \\ Y = 3X_1 + \epsilon_Y \\ X_2 = \begin{cases} -2Y + \epsilon_2 \text{ if } C_2 = 0 \\ 10Y + \epsilon_2 \text{ if } C_2 = 1 \end{cases} \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$



Joint Causal Inference intuition

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$E = 0$

$C_1 = 10$
$C_2 = -2$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 100 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

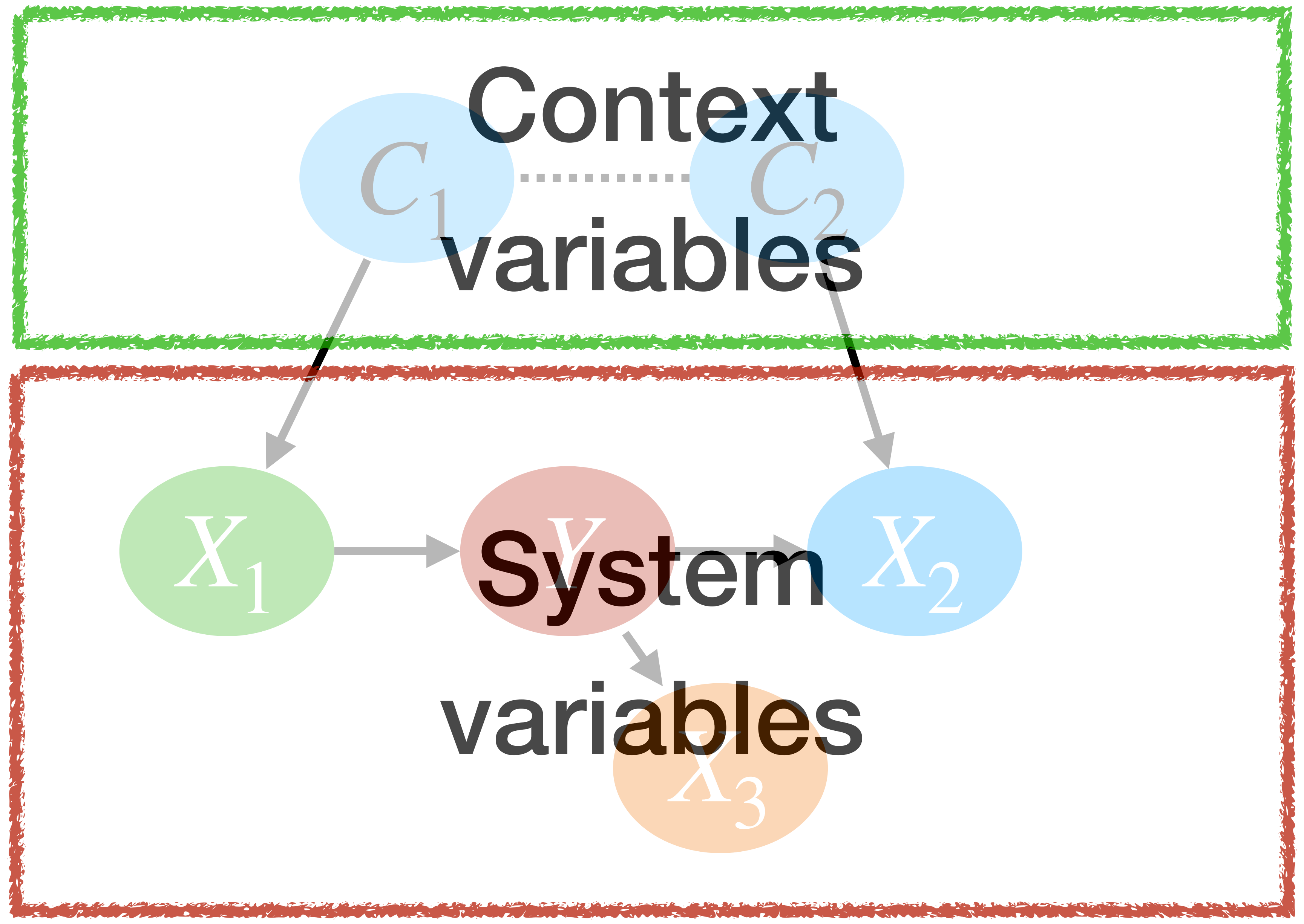
$E = 1$

$C_1 = 100$
$C_2 = -2$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 10Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$E = 2$

$C_1 = 10$
$C_2 = 10$



The context variables C_1 and C_2 can be also descriptive of the intervention in each environment

Joint Causal Inference from Multiple Contexts

Joris M. Mooij, Sara Magliacane, Tom Claassen

- We represent different distributions (including interventional) as an **unknown** **joint causal graph** (possibly cyclic or with latent confounders)

- We **add context variables** so we can **disentangle** changes in distribution across the datasets

	X1	X2	X3
Normal	0,1	2	0
Normal	0,2	3	0

	X1	X2	X3
Gene A	3,1	2	1
Gene A	3,2	3	1
			1
			1

	X1	X2	X3
Gene B	0.2	1	0
Gene B	0.3	1	1
Gene B	0.3	2	1
Gene B	0.4	1	1

C1	C2	X1	X2	X3
0	0	0,1	2	0
0	0	0,2	3	0
0	0	1,1	2	1
0	0	0,1	3	0
1	0	3,1	2	1
1	0	3,2	3	1
1	0	4	1	1
1	0	3,2	3	1
0	1	0,2	1	0
0	1	0,3	1	1
0	1	0,3	2	1
0	1	0,4	1	1

Joint Causal Inference from Multiple Contexts

Joris M. Mooij, Sara Magliacane, Tom Claassen

- We **add context variables** so we can **disentangle** changes in distribution across the datasets (and optionally background knowledge, e.g. context variables are uncaused)
- We can reuse **any standard method for observational data** that fits any chosen assumptions

C1	C2	X1	X2	X3
0	0	0,1	2	0
0	0	0,2	3	0
0	0	1,1	2	1
0	0	0,1	3	0
1	0	3,1	2	1
1	0	3,2	3	1
1	0	4	1	1
1	0	3,2	3	1
0	1	0,2	1	0
0	1	0,3	1	1
0	1	0,3	2	1
0	1	0,4	1	1

Joint Causal Inference from Multiple Contexts

Joris M. Mooij, Sara Magliacane, Tom Claassen

- We **add context variables** so we can **disentangle** changes in distribution across the datasets (and optionally background knowledge, e.g. context variables are uncaused)
- We can reuse **any standard method for observational data** that fits any chosen assumptions

C1	C2	X1	X2	X3
0	0	0,1	2	0
0	0	0,2	3	0
0	0	1,1	2	1
0	0	0,1	3	0
1	0	3,1	2	1
1	0	3,2	3	1
1	0	4	1	1
1	0	3,2	3	1
0	1	0,2	1	0
0	1	0,3	1	1
0	1	0,3	2	1
0	1	0,4	1	1

$X_2 \not\perp\!\!\!\perp C_2$

$X_1 \perp\!\!\!\perp C_2 \mid C_1$

$X_2 \perp\!\!\!\perp C_1 \mid X_3$

$\ldots$

Joint Causal Inference from Multiple Contexts

Joris M. Mooij, Sara Magliacane, Tom Claassen

- We **add context variables** so we can **disentangle** changes in distribution across the datasets (and optionally background knowledge, e.g. context variables are uncaused)

- We can reuse **any standard method for observational data** that fits any chosen assumptions

C1	C2	X1	X2	X3
0	0	0,1	2	0
0	0	0,2	3	0
0	0	1,1	2	1
0	0	0,1	3	0
1	0	3,1	2	1
1	0	3,2	3	1
1	0	4	1	1
1	0	3,2	3	1
0	1	0,2	1	0
0	1	0,3	1	1
0	1	0,3	2	1
0	1	0,4	1	1

$X_2 \not\perp\!\!\!\perp C_2$

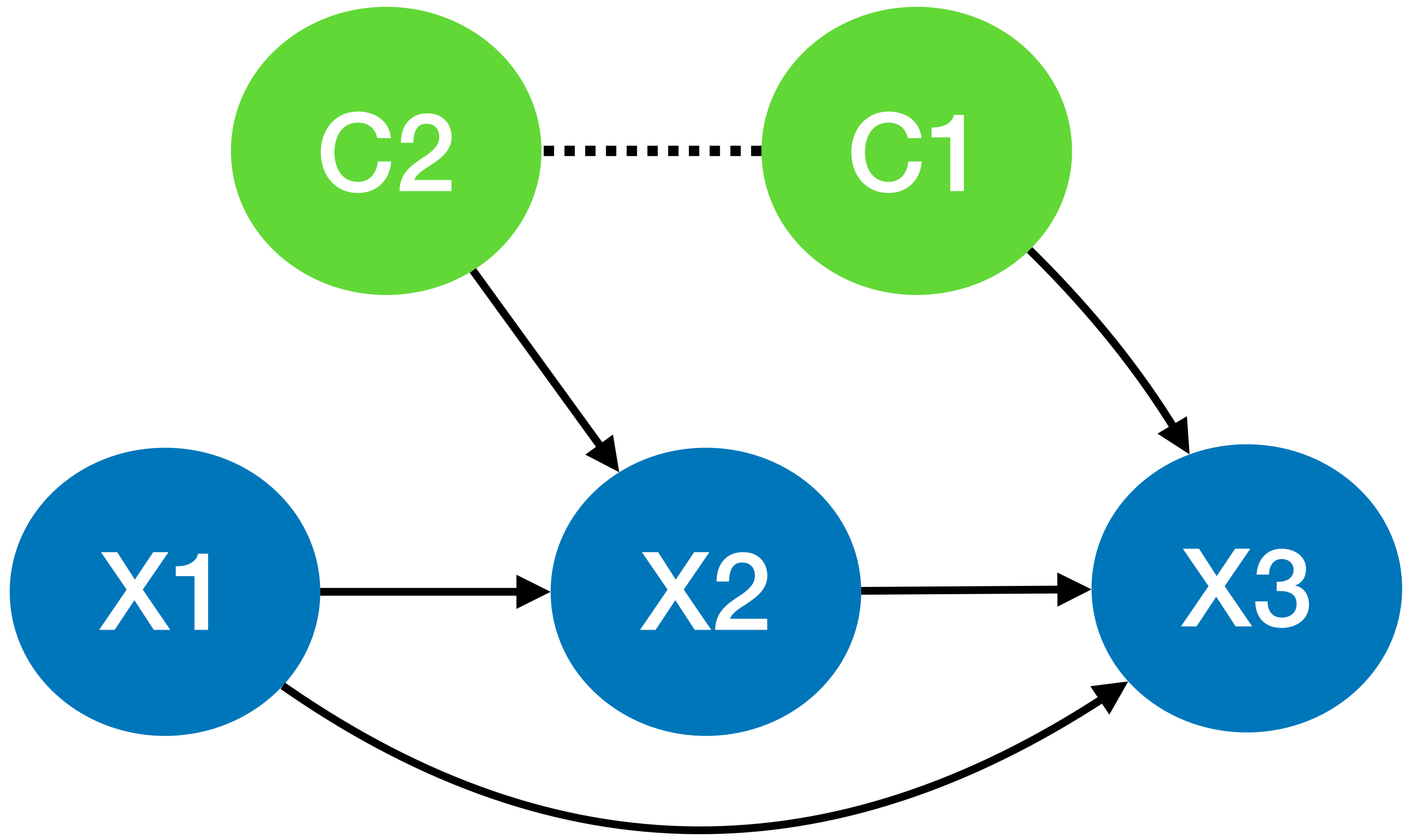
$X_1 \perp\!\!\!\perp C_2 \mid C_1$

$X_2 \perp\!\!\!\perp C_1 \mid X_3$

$\dots$

PC-JCI

FCI-JCI

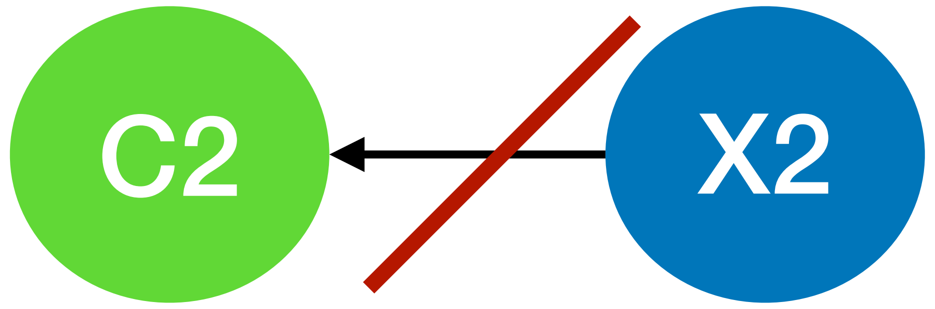


Joint Causal Inference from Multiple Contexts

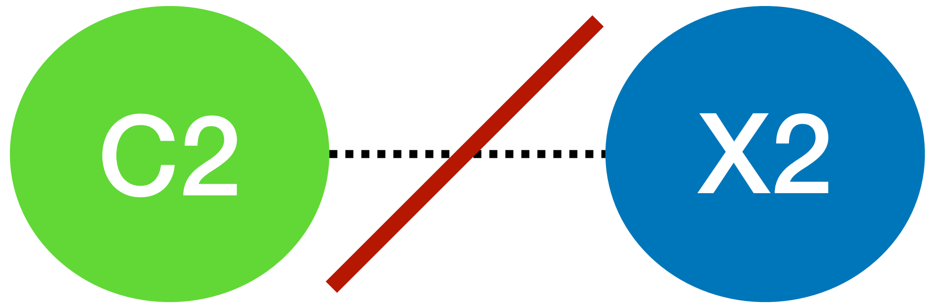
Joris M. Mooij, Sara Magliacane, Tom Claassen

- Additional optional background knowledge based on assumptions:

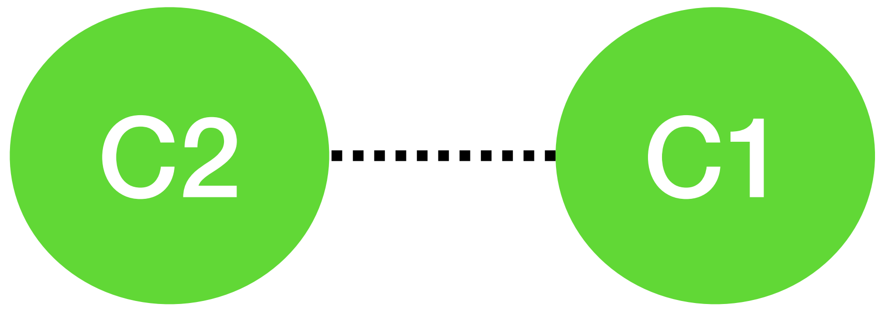
1. No system variable causes any context variable.



2. No context variable is confounded with a system variable.



3. The context variables do not cause each other and they are assumed to be confounded.



https://arxiv.org/abs/1611.10351

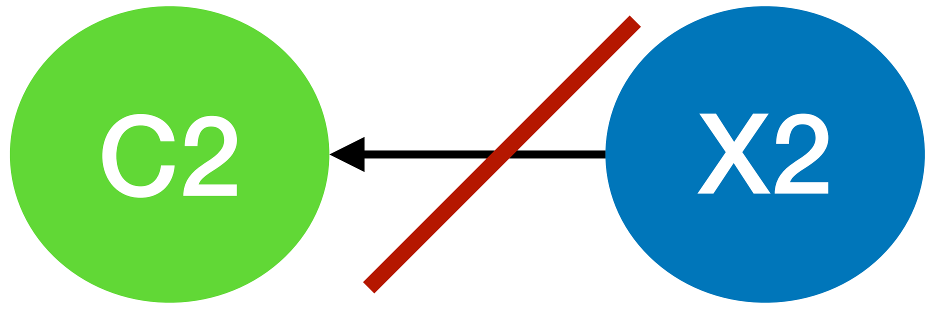
Joint Causal Inference from Multiple Contexts

Joris M. Mooij, Sara Magliacane, Tom Claassen

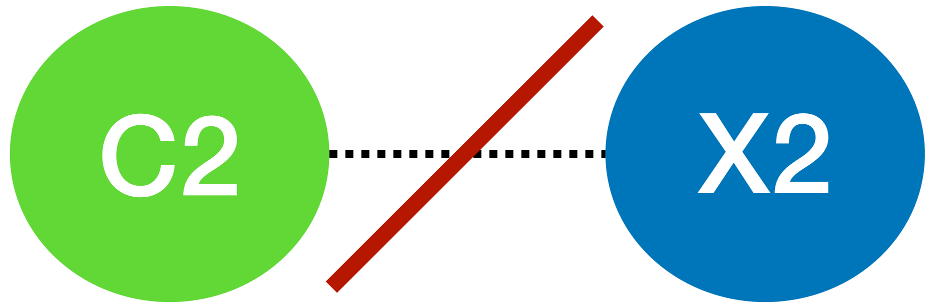
In this talk we assume that there are no latent confounders (except some dependence between the context variables)

- Additional (mptions:

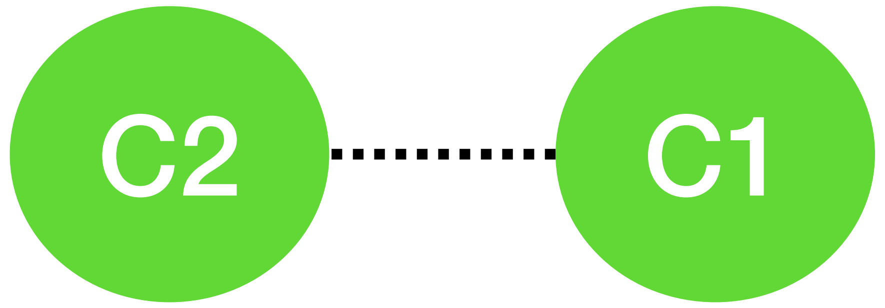
1. No system variable causes any context variable.



2. ~~No context variable is confounded with a system variable.~~

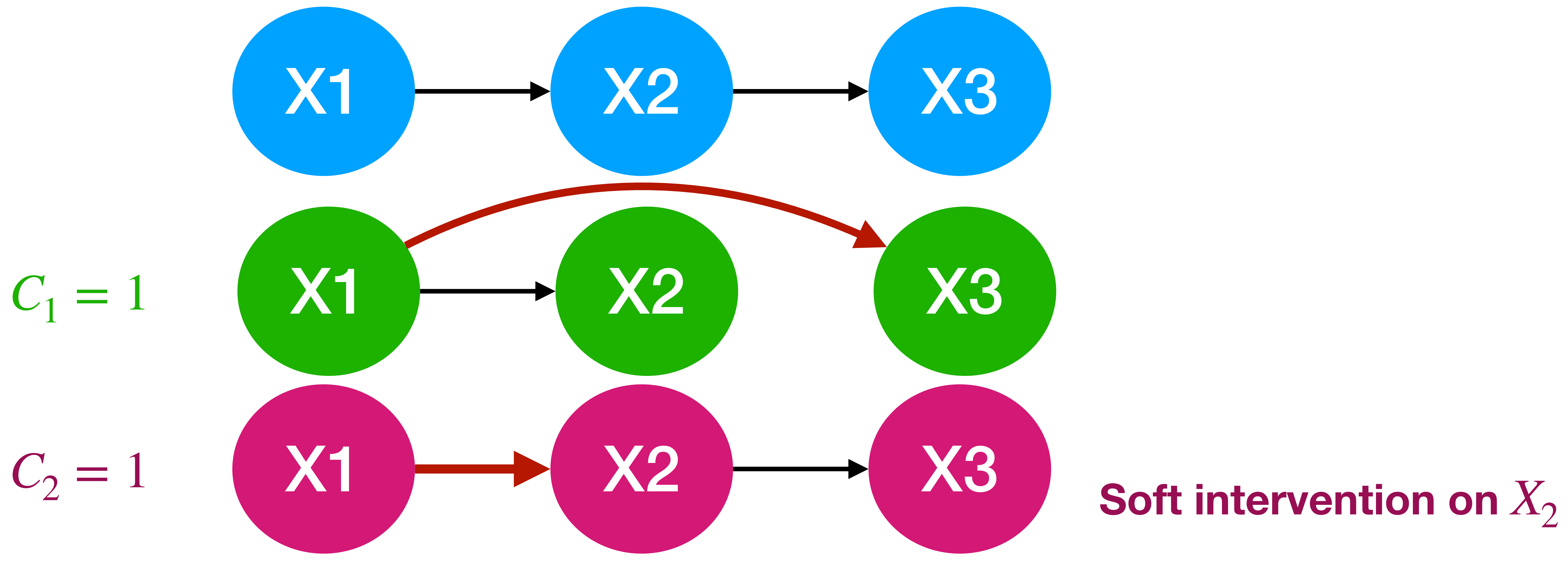


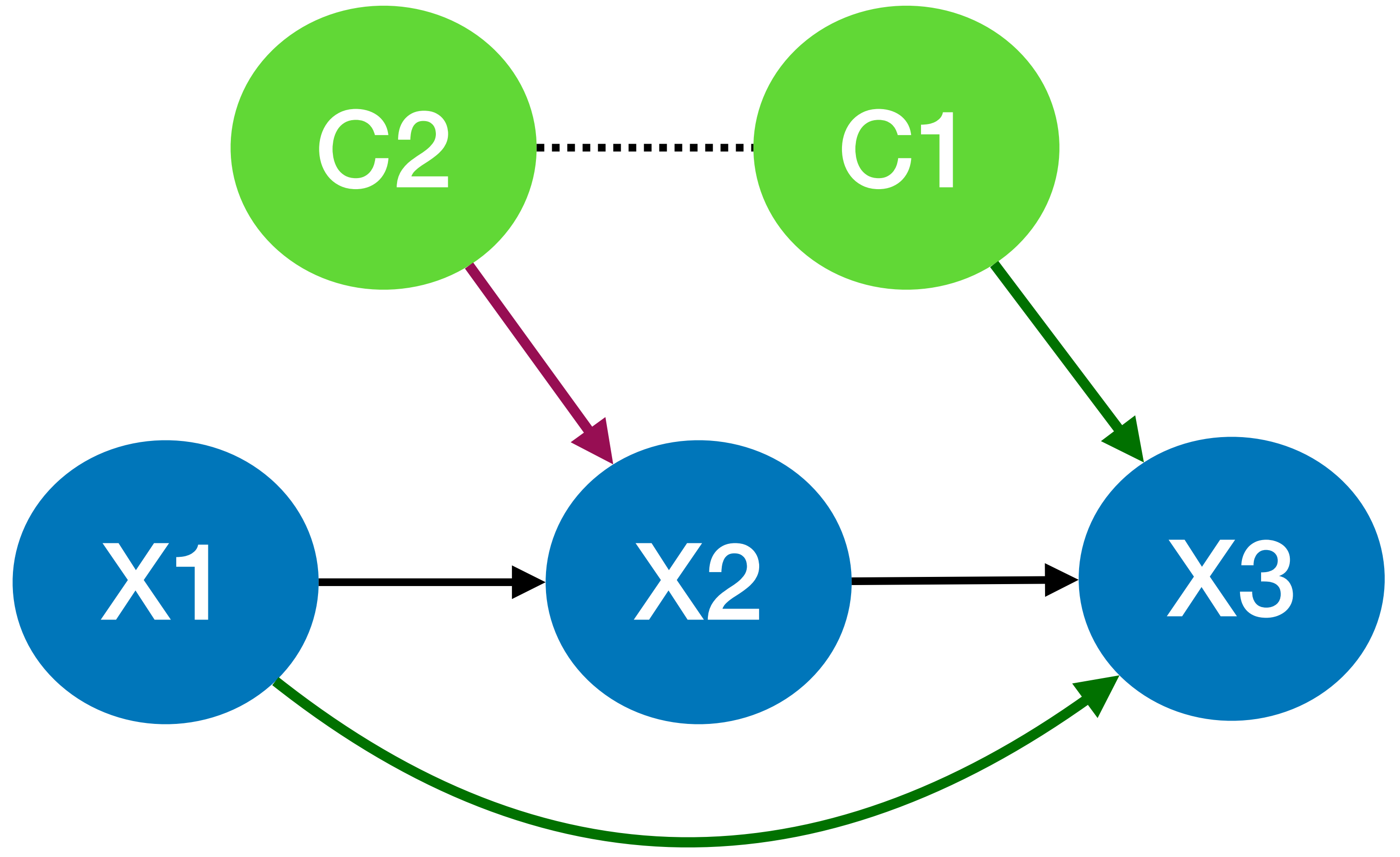
3. The context variables do not cause each other and they are assumed to be confounded.



https://arxiv.org/abs/1611.10351

Joint Causal Inference example - setting

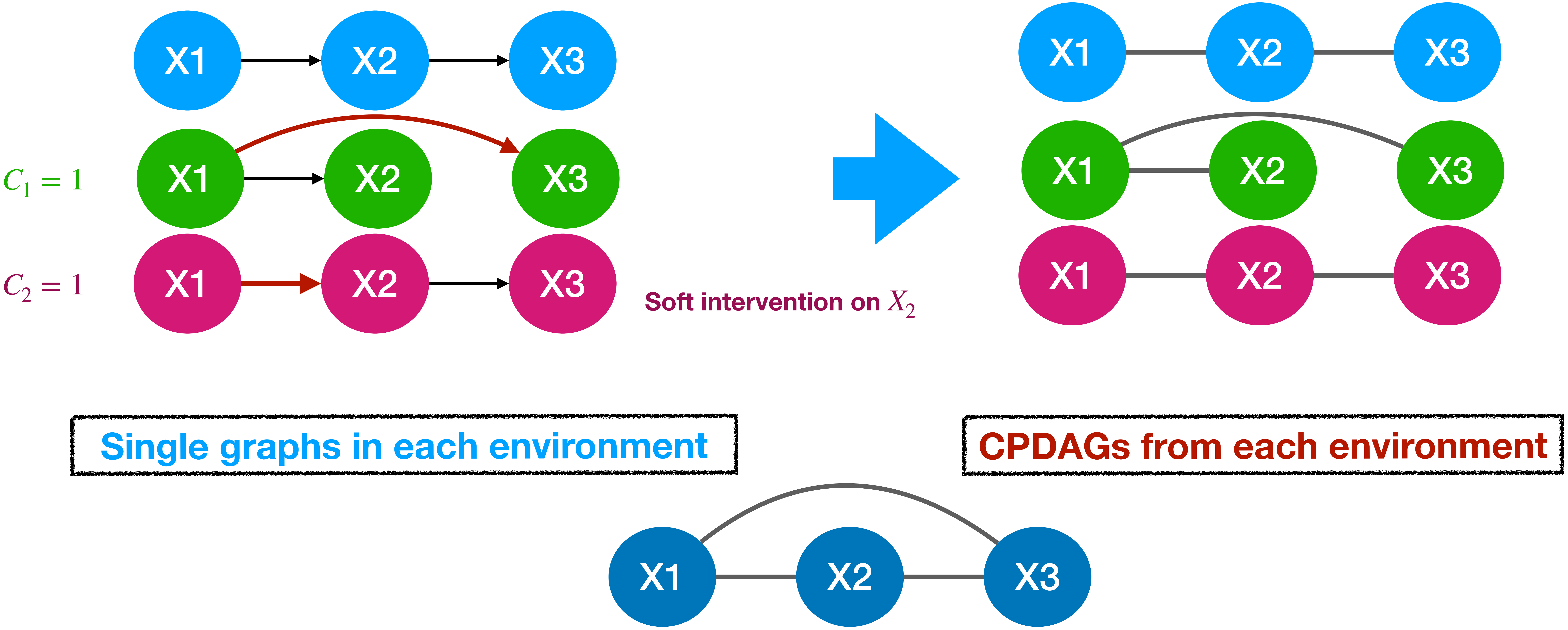




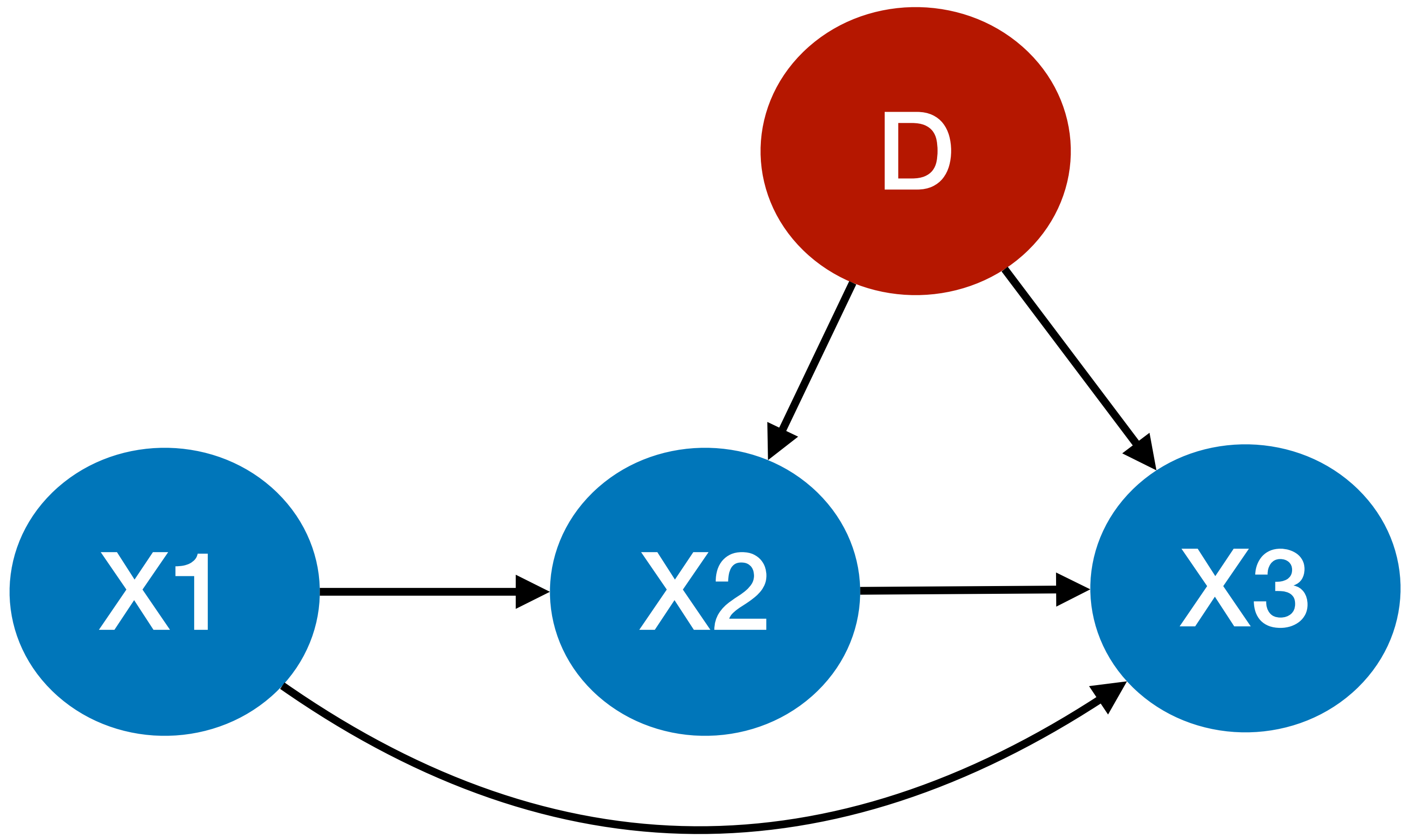
Single graphs in each environment

The joint graph is the union of the single graphs + edges from context variables for the causal mechanisms that change

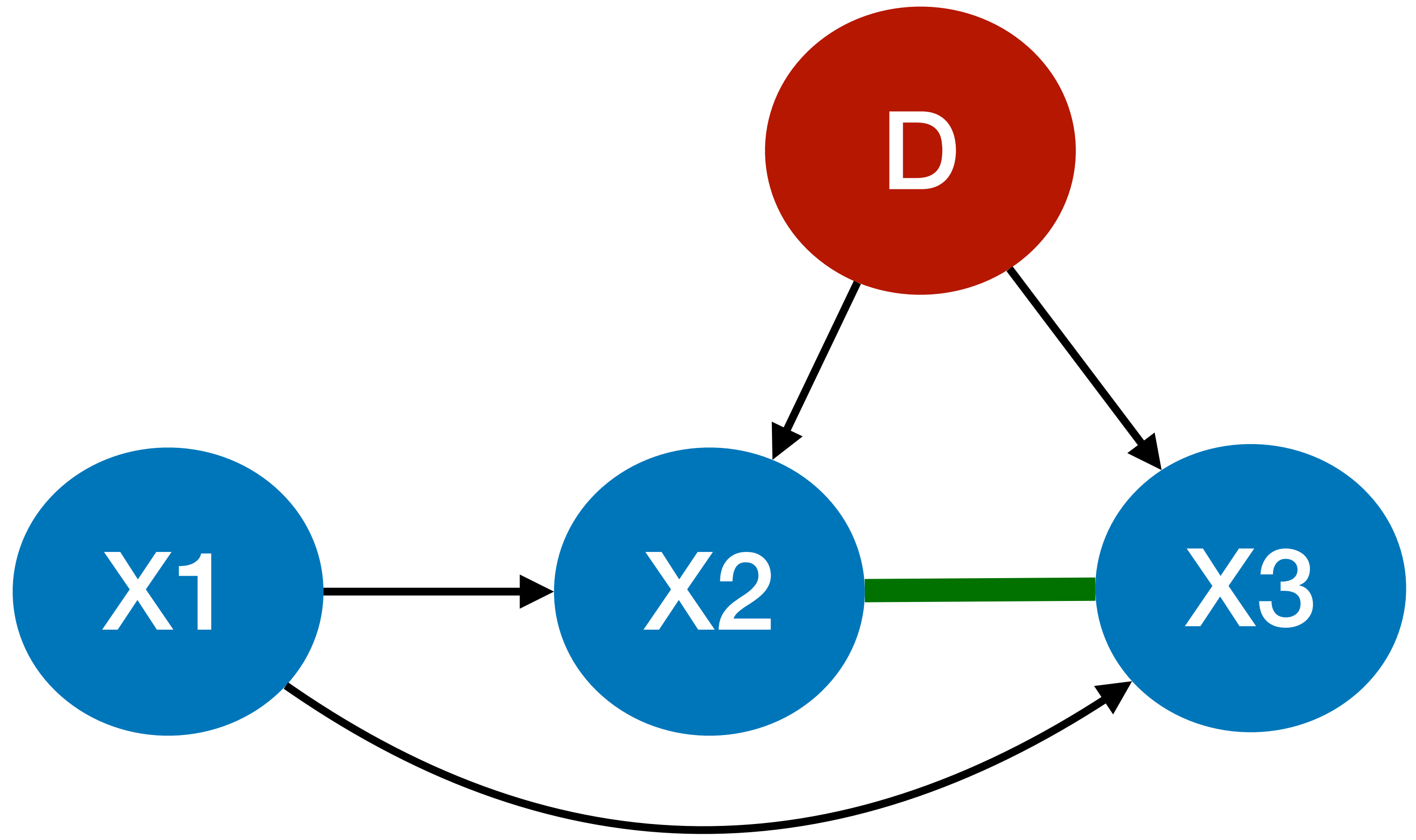
Learning graphs separately in each environment with PC



Joint causal inference + PC with a single domain variable

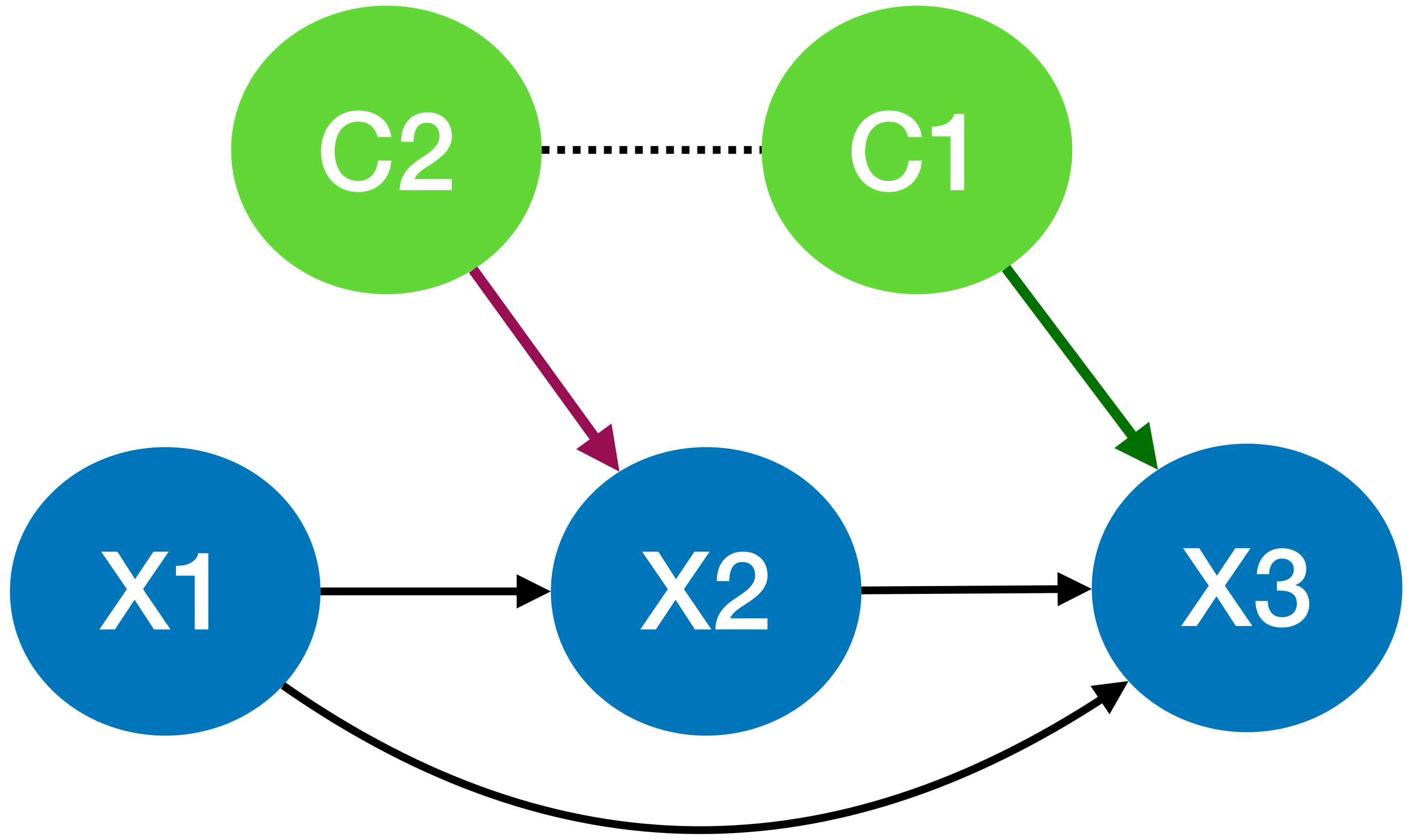


True underlying joint graph with domain variable

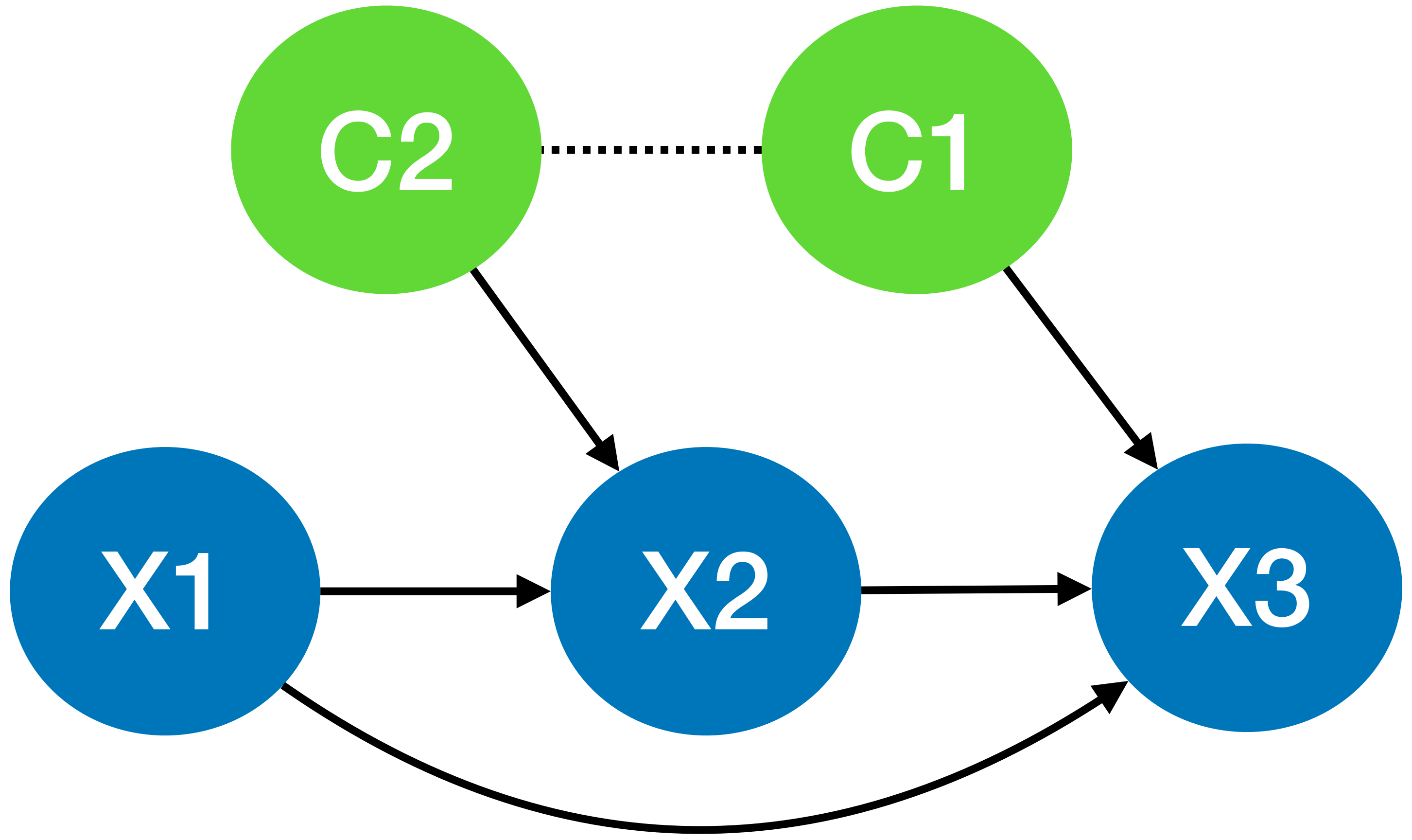


CPDAG with Joint Causal Inference and PC with a single domain variable

Joint causal inference + PC with multiple context variables



True underlying joint graph with multiple context variable

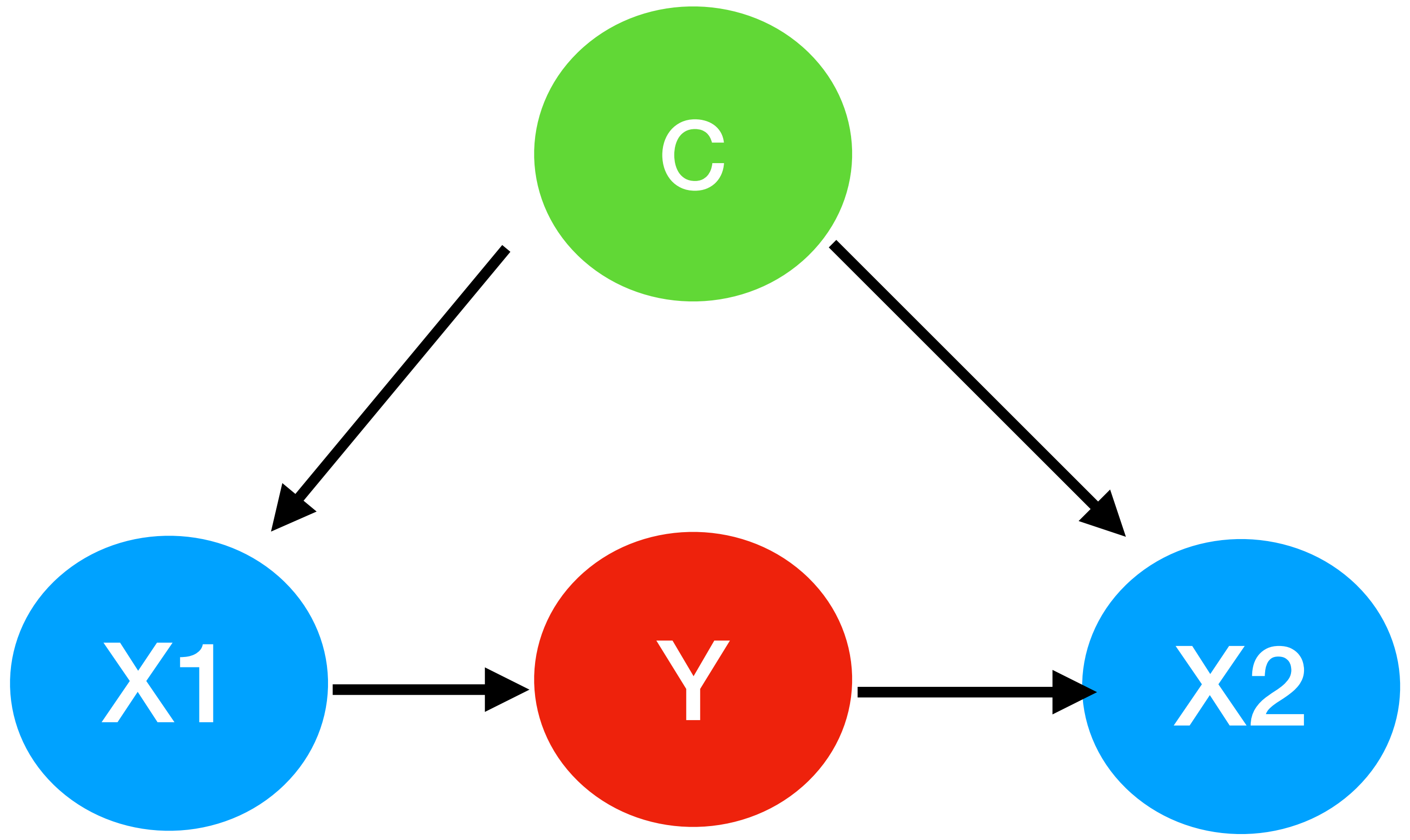


CPDAG with Joint Causal Inference and PC with multiple context variables

Application - Domain adaptation

C	X1	X2	Y
0	0,1	2	0
0	0,2	3	0
0	1,1	2	1
0	0,1	3	0
1	3,1	2	?
1	3,2	3	?
1	4	2	?
1	3,2	3	?

Source domain (C = 0); Target domain (C = 1)



We can represent P(X1, X2,Y, C) with an **(unknown)** causal graph

- We can represent the different datasets jointly with Joint Causal Inference
- We can use it to reason about **features that offer a robust prediction of Y**

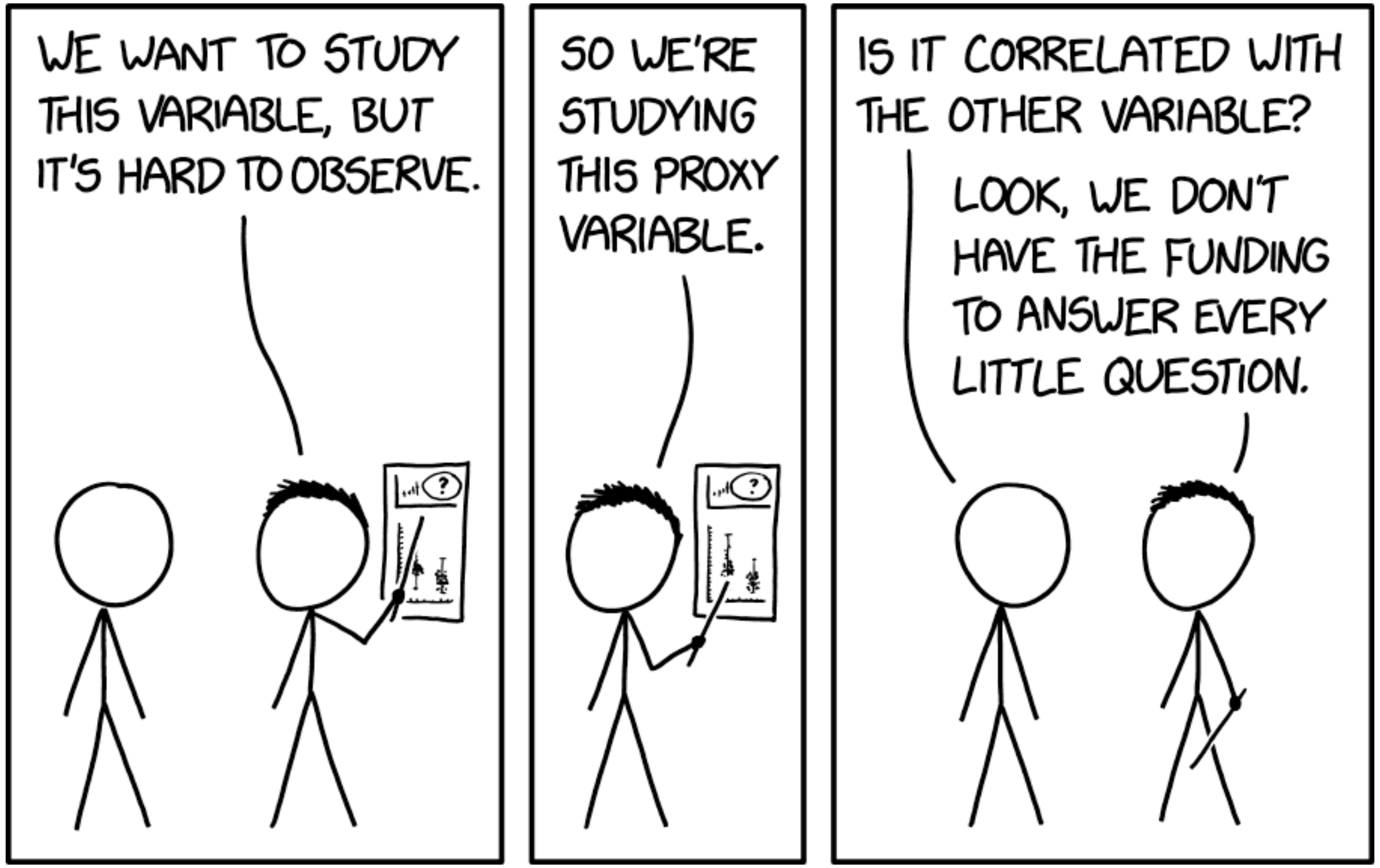
This class

- Introduction to causal discovery
  - Common assumptions: causal sufficiency, acyclicity, faithfulness
- Constraint-based causal discovery on observational data (causal sufficiency)
  - SGS, PC

- Learning from multiple contexts or interventional data
  - Invariant Causal Prediction
  - Joint Causal Inference

Inspired by https://stat.ethz.ch/lectures/ss21/causality.php

Questions?



https://xkcd.com/2652/