THE UNIVERSITY of EDINBURGH

# Methods for Causal Inference
# Lecture 13: Do-Calculus

Ava Khamseh

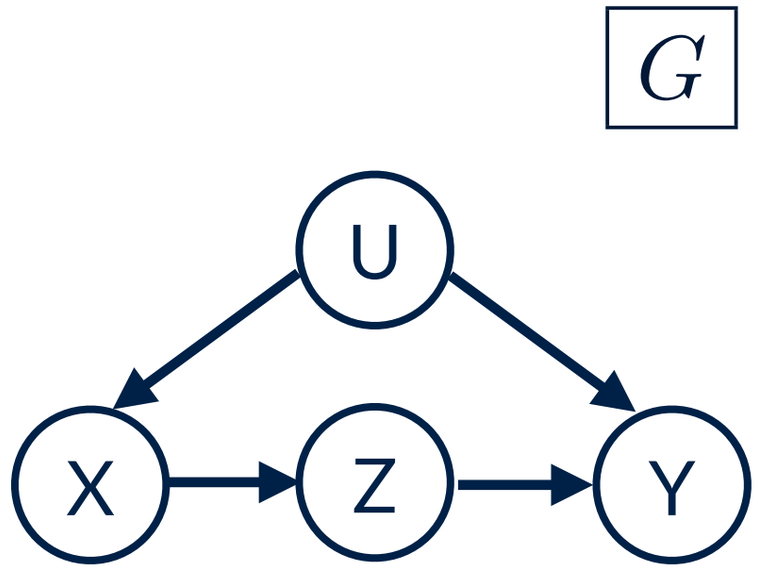
School of Informatics
2023-2024

# Do-Calculus

Not all causal quantities are identifiable
(this depends on the structure of the graph)

Here, we generalise the rules of front/back-door criteria: **do-calculus**

Let X, Y, Z be arbitrary disjoint sets of nodes in a DAG G.



# Do-Calculus

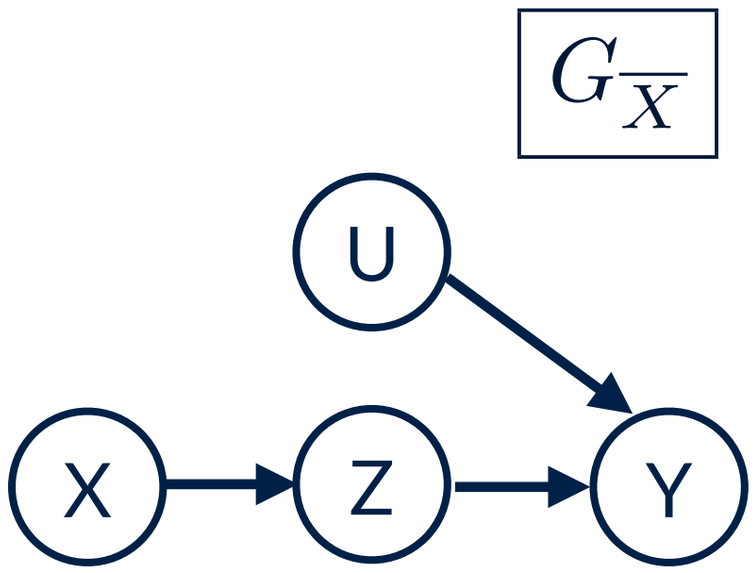
Not all causal quantities are identifiable
(this depends on the structure of the graph)

Here, we generalise the rules of front/back-door criteria: **do-calculus**

Let X, Y, Z be arbitrary disjoint sets of nodes in a DAG G.

Notation

$G_{\overline{X}}$ : The graph obtained by deleting all arrows pointing to nodes in X



# Do-Calculus

Not all causal quantities are identifiable
(this depends on the structure of the graph)

Here, we generalise the rules of front/back-door criteria: **do-calculus**

Let X, Y, Z be arbitrary disjoint sets of nodes in a DAG G.

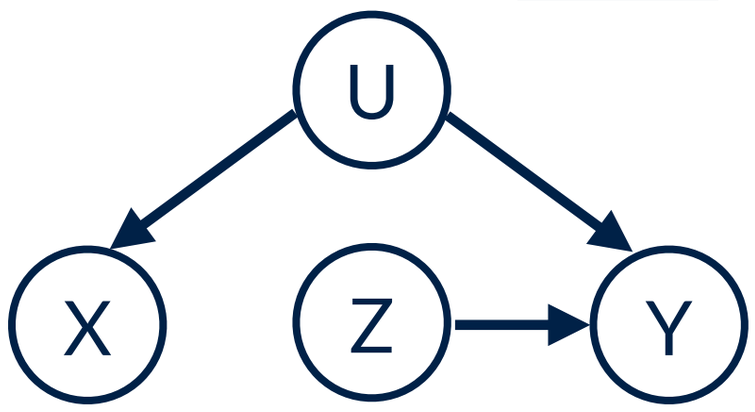
Notation

$G_{\overline{X}}$ : The graph obtained by deleting all arrows pointing to nodes in X

$G_{\underline{X}}$ : The graph obtained by deleting all arrow emerging from nodes X

Note for example: $G_{\underline{X}} = G_{\overline{Z}}$

$G_{\underline{X}}$



# Do-Calculus

Not all causal quantities are identifiable
(this depends on the structure of the graph)

Here, we generalise the rules of front/back-door criteria: **do-calculus**

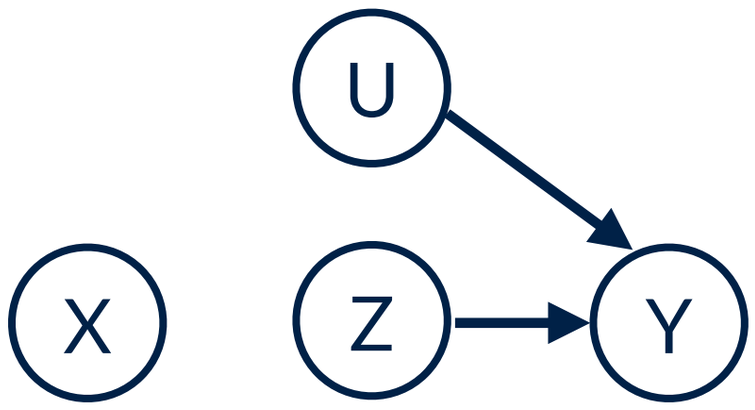
Let X, Y, Z be arbitrary disjoint sets of nodes in a DAG G.

Notation

$G_{\overline{X}}$ : The graph obtained by deleting all arrows pointing to nodes in X

$G_{\underline{X}}$ : The graph obtained by deleting all arrow emerging from nodes X

More examples: $G_{\overline{X}\,\overline{Z}}$



# Do-Calculus

Not all causal quantities are identifiable
(this depends on the structure of the graph)

Here, we generalise the rules of front/back-door criteria: **do-calculus**

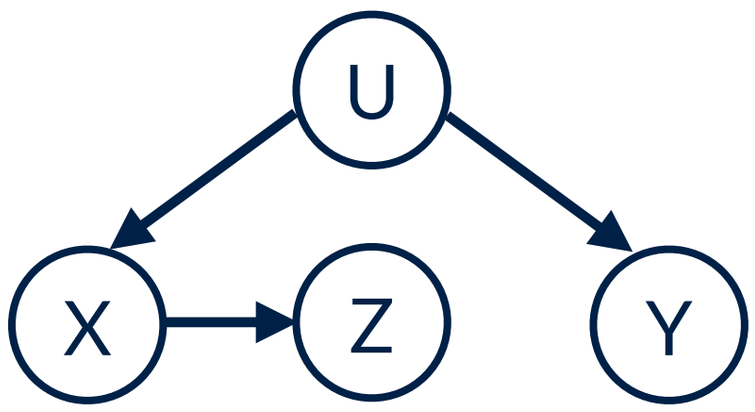
Let X, Y, Z be arbitrary disjoint sets of nodes in a DAG G.

Notation

$G_{\overline{X}}$ : The graph obtained by deleting all arrows pointing to nodes in X

$G_{\underline{X}}$ : The graph obtained by deleting all arrow emerging from nodes X

More examples: $G_{\underline{Z}}$



# Do-Calculus

Not all causal quantities are identifiable
(this depends on the structure of the graph)

Here, we generalise the rules of front/back-door criteria: **do-calculus**

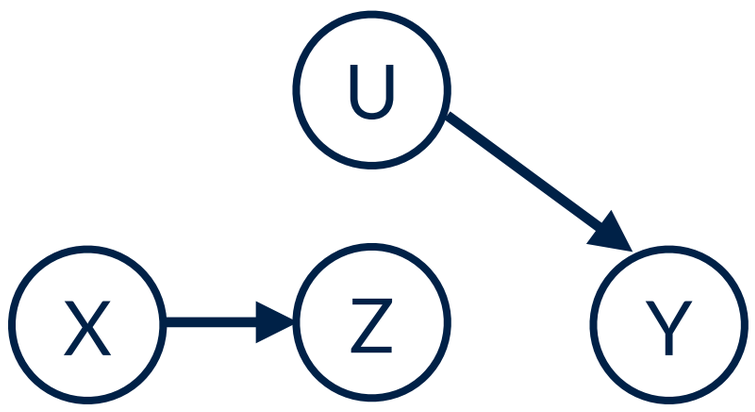
Let X, Y, Z be arbitrary disjoint sets of nodes in a DAG G.

Notation

$G_{\overline{X}}$ : The graph obtained by deleting all arrows pointing to nodes in X

$G_{\underline{X}}$ : The graph obtained by deleting all arrow emerging from nodes X

More examples: $G_{\overline{X}\underline{Z}}$



# Do-Calculus Rules

Let X, Y, Z, W be arbitrary disjoint sets of nodes in a DAG G

**Rule 1** (insertion/deletion of observations):

$$p(Y|do(X=x),Z,W) = p(Y|do(X=x),W) \text{ if } (Y \perp\!\!\!\perp Z)|X,W \text{ in } G_{\overline{X}}$$

i.e. if Y and Z are d-separated by X, W in a graph where incoming edges in X have been removed.

# Do-Calculus Rules

Let X, Y, Z, W be arbitrary disjoint sets of nodes in a DAG G

**Rule 1** (insertion/deletion of observations):

$$p(Y|do(X=x),Z,W) = p(Y|do(X=x),W) \text{ if } (Y \perp\!\!\!\perp Z)|X,W \text{ in } G_{\overline{X}}$$

i.e. if Y and Z are d-separated by X, W in a graph where incoming edges in X have been removed.

In the special case where $X = \emptyset$ the above states:

$$p(Y|Z,W) = p(Y|W) \text{ if } (Y \perp\!\!\!\perp Z)|W$$

Which is simply d-separation. So the above is the
**generalisation of d-separation in the presence of an intervention do(X=x)**

# Do-Calculus Rules

Let X, Y, Z, W be arbitrary disjoint sets of nodes in a DAG G

**Rule 1** (insertion/deletion of observations):

$$p(Y|do(X=x),Z,W) = p(Y|do(X=x),W) \text{ if } (Y \perp\!\!\!\perp Z)|X,W \text{ in } G_{\overline{X}}$$

**Rule 2** (Action/observation exchange):

$$p(Y|do(X=x),do(Z=z),W) = p(Y|do(X=x),z,W) \text{ if } (Y \perp\!\!\!\perp Z)|X,W \text{ in } G_{\overline{X}\underline{Z}}$$

i.e. if Y and Z are d-separated by X, W in a graph where incoming edges in X and outgoing edges from Z have been removed.

This rules provides a condition for an external intervention do(Z=z) to have the same effect on Y as the passive observation Z=z.

# Do-Calculus Rules

Let X, Y, Z, W be arbitrary disjoint sets of nodes in a DAG G

**Rule 1** (insertion/deletion of observations):

$$p(Y|do(X=x),Z,W) = p(Y|do(X=x),W) \text{ if } (Y \perp\!\!\!\perp Z)|X,W \text{ in } G_{\overline{X}}$$

**Rule 2** (Action/observation exchange):

$$p(Y|do(X=x),do(Z=z),W) = p(Y|do(X=x),z,W) \text{ if } (Y \perp\!\!\!\perp Z)|X,W \text{ in } G_{\overline{X}\underline{Z}}$$

In the special case were $X = \emptyset$ the above states:

$$p(Y|do(Z=z),W) = p(Y|z,W) \text{ if } (Y \perp\!\!\!\perp Z)|W \text{ in } G_{\underline{Z}}$$

**Which is the generalisation of backdoor criterion (adjustment formula).**

# Do-Calculus Rules

Let X, Y, Z, W be arbitrary disjoint sets of nodes in a DAG G

**Rule 1** (insertion/deletion of observations):

$$p(Y|do(X=x),Z,W) = p(Y|do(X=x),W) \text{ if } (Y \perp\!\!\!\perp Z)|X,W \text{ in } G_{\overline{X}}$$

**Rule 2** (Action/observation exchange):

$$p(Y|do(X=x),do(Z=z),W) = p(Y|do(X=x),z,W) \text{ if } (Y \perp\!\!\!\perp Z)|X,W \text{ in } G_{\overline{X}\underline{Z}}$$

**Rule 3** (Insertion/deletion of actions):

$$p(Y|do(X=x),do(Z=z),W) = p(Y|do(X=x),W) \text{ if } (Y \perp\!\!\!\perp Z)|X,W \text{ in } G_{\overline{X}\overline{Z(W)}}$$

where Z(W) is the set of Z-nodes that are not ancestors of any W-node in $G_{\overline{X}}$

# Do-Calculus Rules

Let X, Y, Z, W be arbitrary disjoint sets of nodes in a DAG G

**Rule 1** (insertion/deletion of observations):

$$p(Y|do(X=x),Z,W) = p(Y|do(X=x),W) \text{ if } (Y \perp\!\!\!\perp Z)|X,W \text{ in } G_{\overline{X}}$$

**Rule 2** (Action/observation exchange):

$$p(Y|do(X=x),do(Z=z),W) = p(Y|do(X=x),z,W) \text{ if } (Y \perp\!\!\!\perp Z)|X,W \text{ in } G_{\overline{X}\underline{Z}}$$

**Rule 3** (Insertion/deletion of actions):

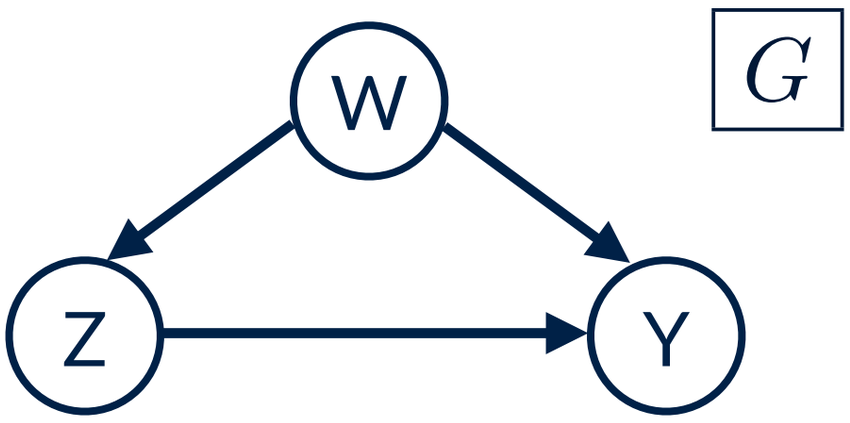
$$p(Y|do(X=x),do(Z=z),W) = p(Y|do(X=x),W) \text{ if } (Y \perp\!\!\!\perp Z)|X,W \text{ in } G_{\overline{X}\overline{Z(W)}}$$

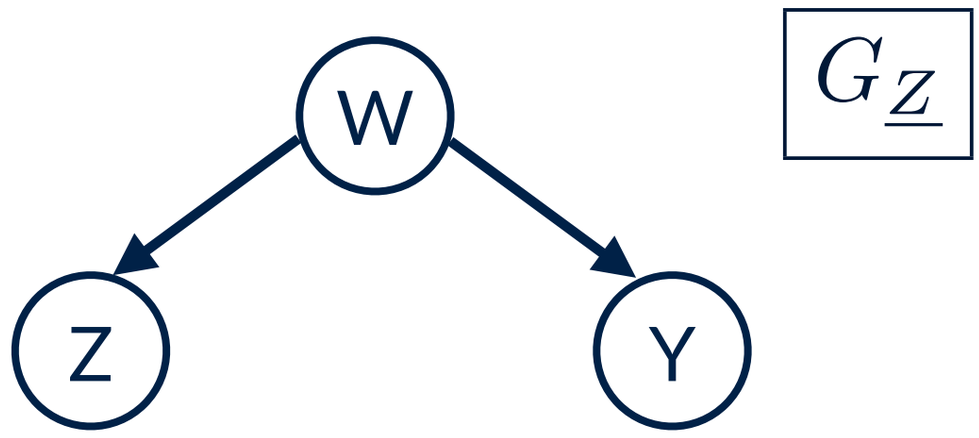
Provides conditions for introducing/deleting an external intervention without affecting the conditional probability of Y.

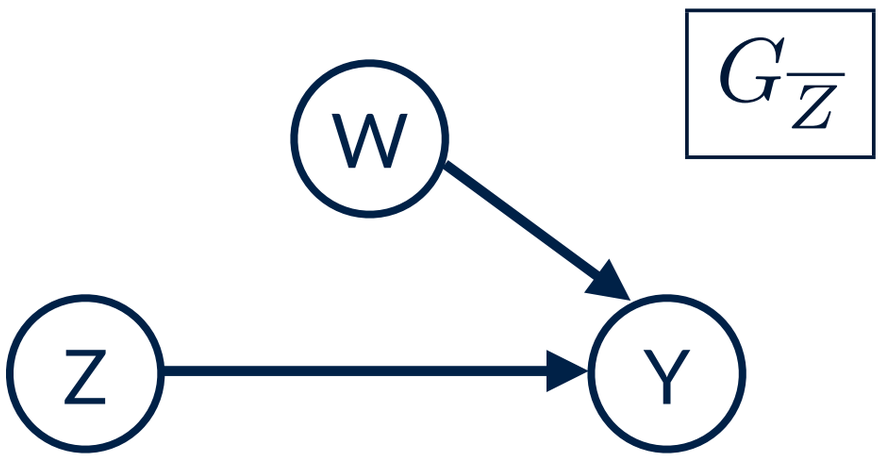


Proof in Pearl 1995a

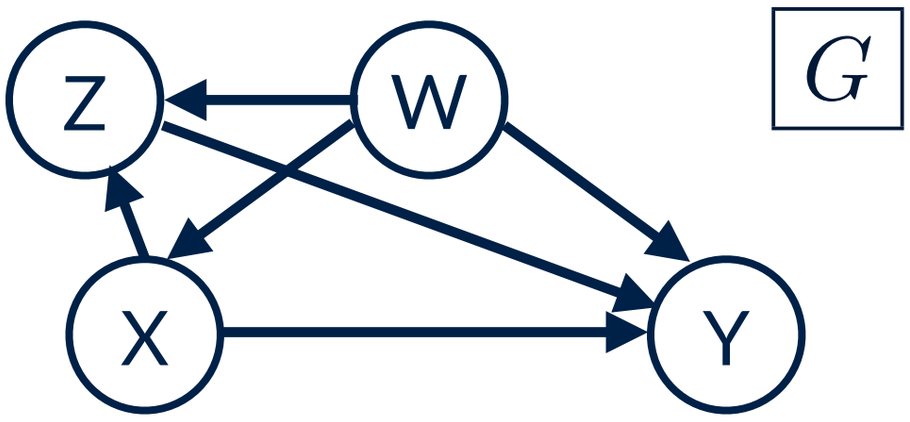
# Graph examples

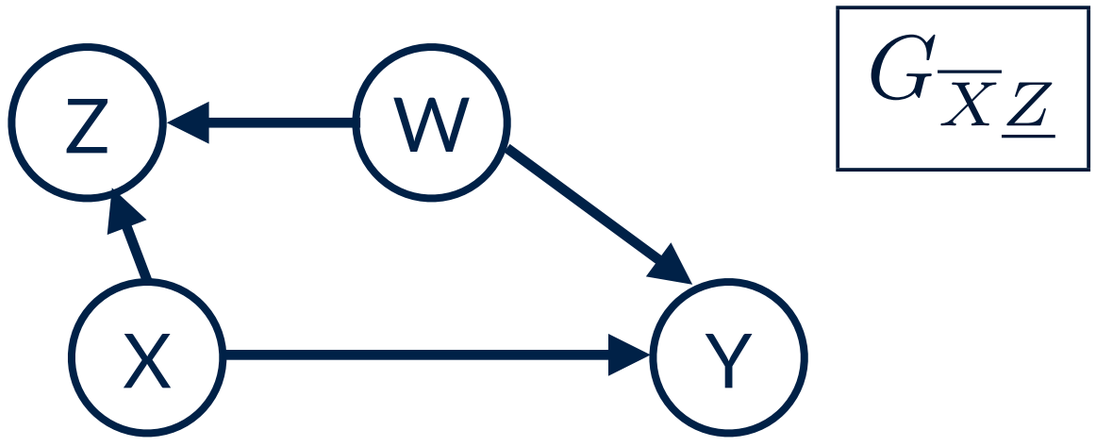


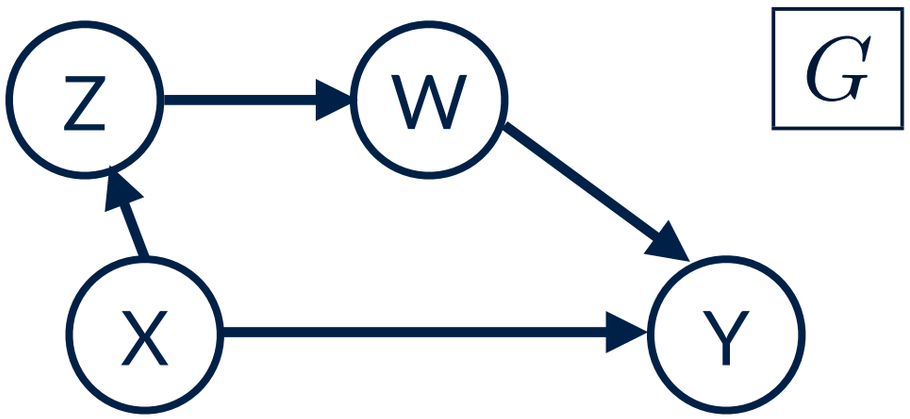



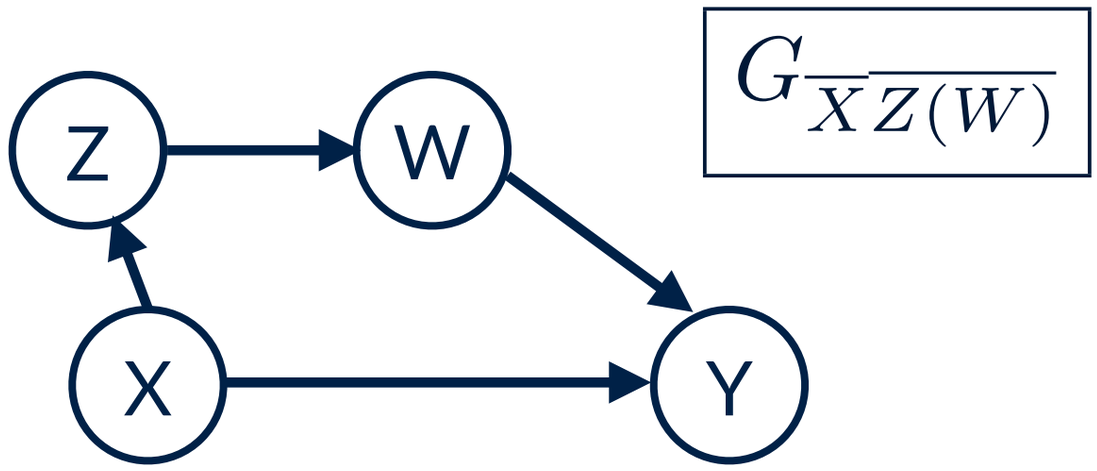
# Graph examples



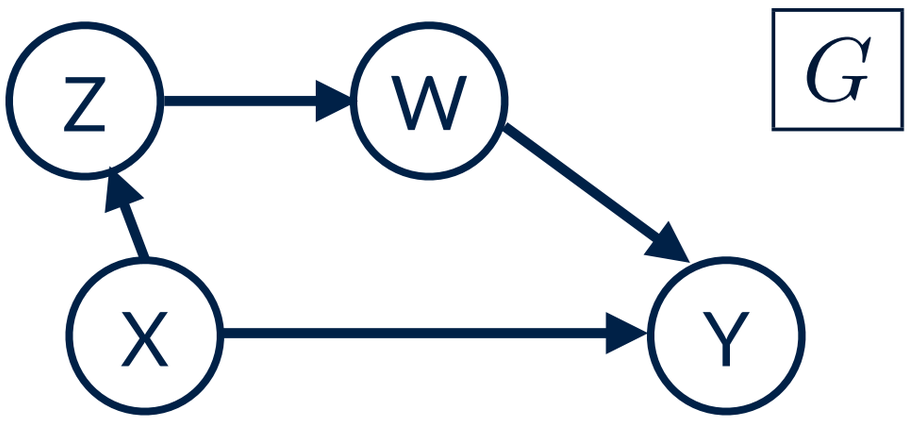



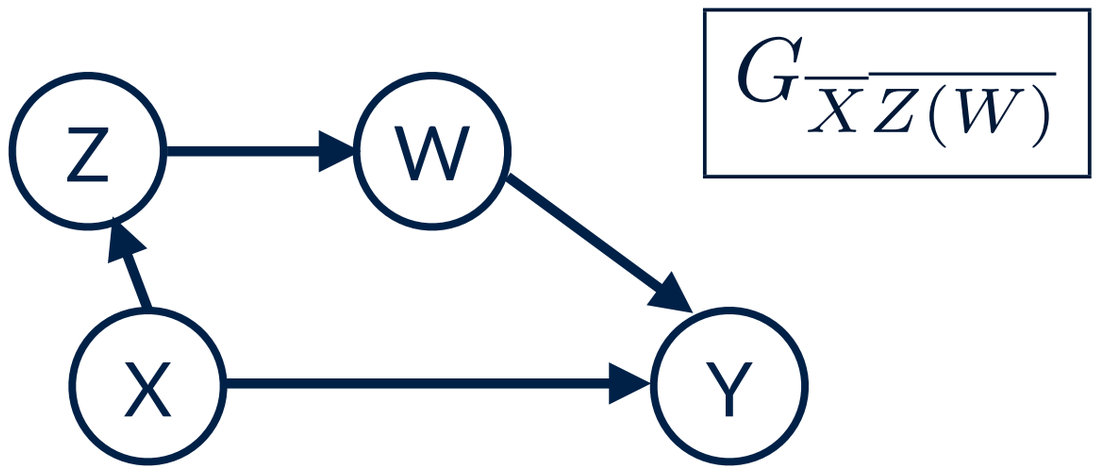
# Graph examples

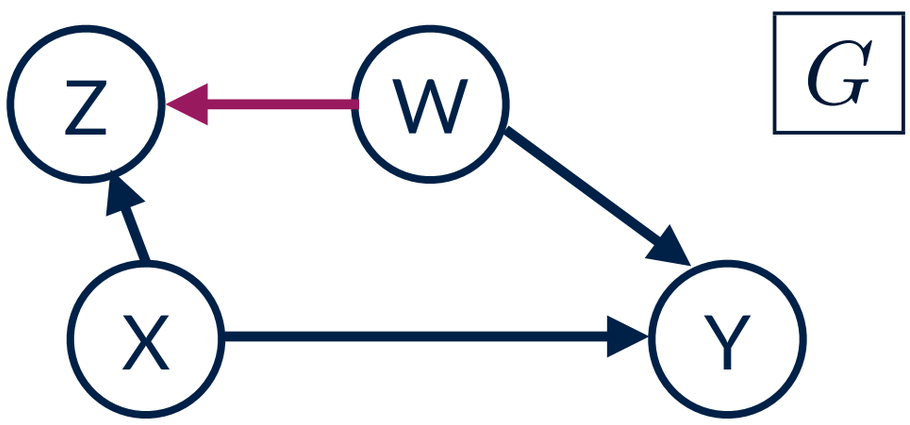


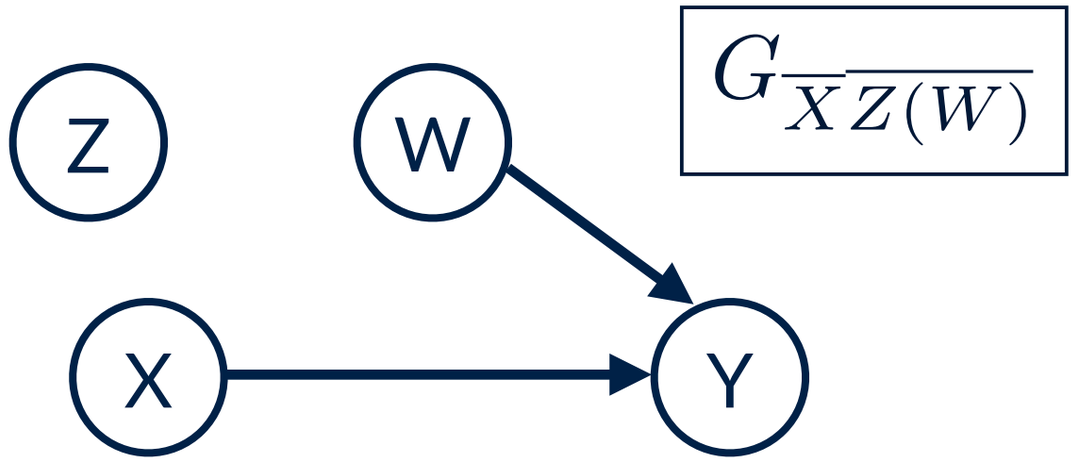



# Graph examples




where Z(W) is the set of Z-nodes that are not ancestors of any W-node in $G_{\overline{X}}$

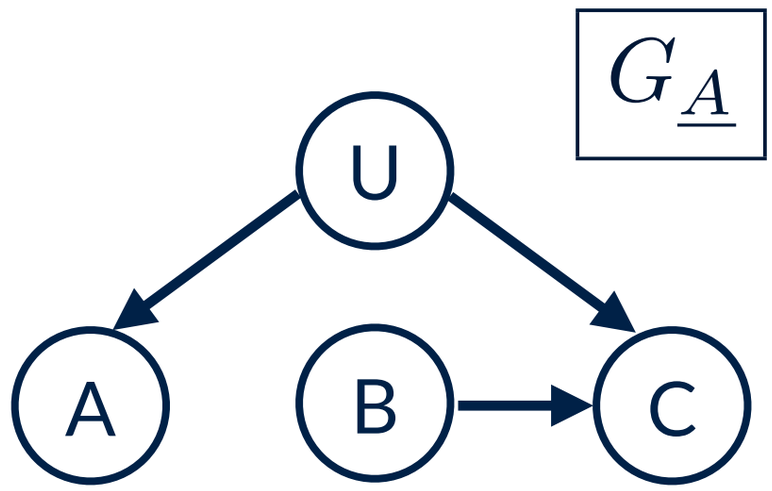
# Derivation of front-door criterion using do-calculus

Task 1: Compute $p(B|do(A=a))$

We need to write this in a format without the 'do'. Rule 2 is useful here.
We use Rule 2, special case:

$$p(B|do(A=a)) = p(B|a) \text{ if } (B \perp\!\!\!\perp A) \text{ in } G_{\underline{A}}$$

And the condition is satisfied because the path $A \leftarrow U \rightarrow C \leftarrow B$
is blocked by C, so B and A are d-separated in this graph.



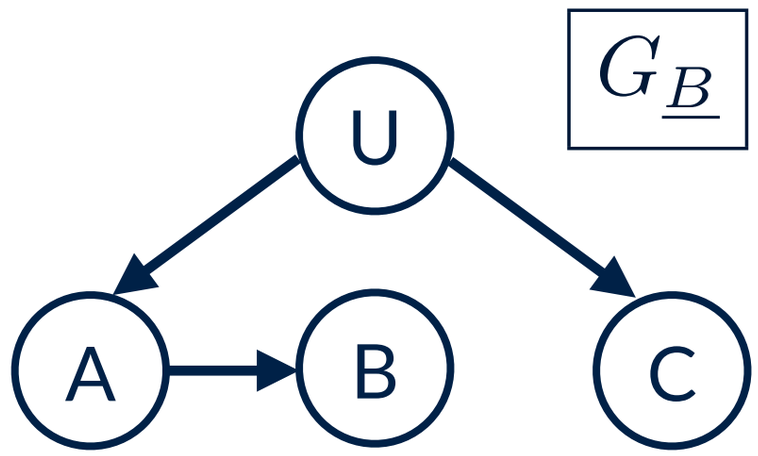



Pearl's 2009 book, page 87

# Derivation of front-door criterion using do-calculus

Task 2: Compute $p(C|do(B = b))$

We cannot apply rule 2 to replace $do(B = b)$ with $b$ because $G_{\underline{B}}$ contains a back-door path from B to C: $B \leftarrow A \leftarrow U \rightarrow C$




Pearl's 2009 book, page 87

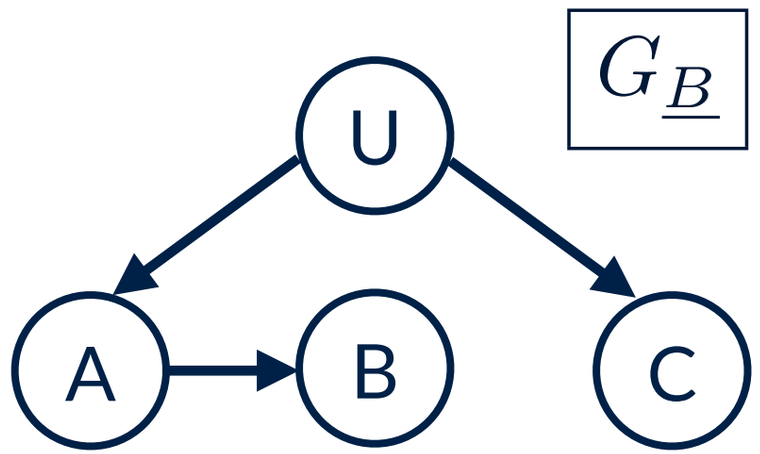
# Derivation of front-door criterion using do-calculus

Task 2: Compute $p(C|do(B=b))$

We cannot apply rule 2 to replace $do(B=b)$ with $b$ because $G_{\underline{B}}$ contains a back-door path from B to C: $B \leftarrow A \leftarrow U \rightarrow C$

BUT, we can use block this path by measuring A. So marginalising gives:

$$p(C|do(B=b)) = \sum_A p(A, C|do(B=b)) = \sum_A p(C|A, do(B=b))p(A|do(B=b))$$






Pearl's 2009 book, page 87

# Derivation of front-door criterion using do-calculus

Task 2: Compute $p(C|do(B=b))$

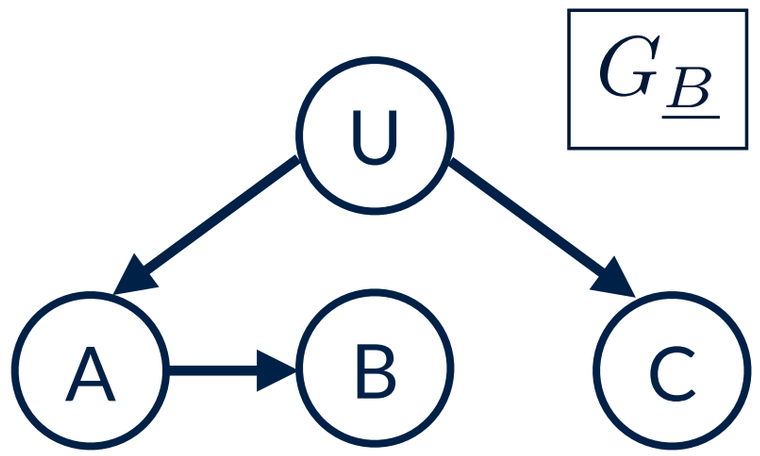
We cannot apply rule 2 to replace $do(B=b)$ with $b$ because $G_{\underline{B}}$ contains a back-door path from B to C: $B \leftarrow A \leftarrow U \rightarrow C$

BUT, we can use block this path by measuring A. So marginalising gives:

$$p(C|do(B=b)) = \sum_A p(A, C|do(B=b)) = \sum_A p(C|A, do(B=b)) \boxed{p(A|do(B=b))}$$

$$p(A|do(B=b)) = p(A) \qquad (A \perp\!\!\!\perp B) \text{ in } G_{\overline{B}}$$

Immediate via do-operation/graph manipulation (with B being a descendent of A in G), or, Rule 3: Due to d-separation of A and B (conditional on nothing) in graph $\boxed{G_{\overline{B}}}$






Pearl's 2009 book, page 87

# Derivation of front-door criterion using do-calculus

Task 2: Compute $p(C|do(B=b))$

We cannot apply rule 2 to replace $do(B=b)$ with $b$ because $G_{\underline{B}}$ contains a back-door path from B to C: $B \leftarrow A \leftarrow U \rightarrow C$

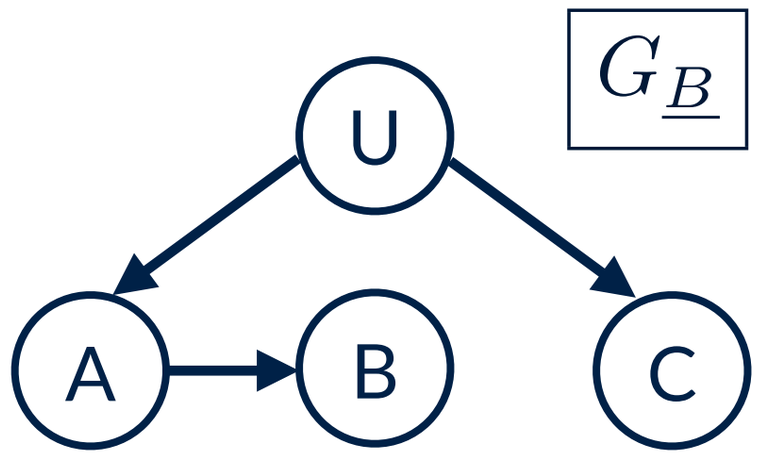
BUT, we can use block this path by measuring A. So marginalising gives:

$$p(C|do(B=b)) = \sum_A p(A,C|do(B=b)) = \sum_A p(C|A,do(B=b))p(A|do(B=b))$$

$$p(C|A,do(B=b)) = p(C|A,b) \quad (C \perp\!\!\!\perp B|A) \text{ in } G_{\underline{B}}$$

Which uses Rule 2, with C and B d-separated given A.
Therefore,

$$p(C|do(B=b)) = \sum_A p(C|A,b)p(A)$$






Pearl's 2009 book, page 87

# Derivation of front-door criterion using do-calculus

Task 3: Compute $p(C|do(A=a))$ . Marginalising over B gives:

$$p(C|do(A=a)) = \sum_B p(C|B, do(A=a)) p(B|do(A=a))$$

Second term already done. First term, no rule can be applied to eliminate do(A).



Pearl's 2009 book, page 87

# Derivation of front-door criterion using do-calculus

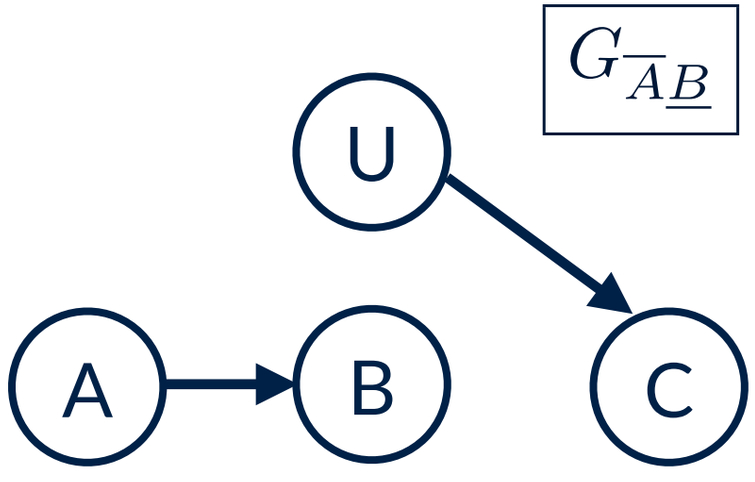
Task 3: Compute $p(C|do(A=a))$ . Marginalising over B gives:

$$p(C|do(A=a)) = \sum_B p(C|B, do(A=a))p(B|do(A=a))$$

Second term already done. First term, no rule can be applied to eliminate do(A). Instead, use Rule 2 to add do(B):

$$p(C|B, do(A=a)) = p(C|do(B=b), do(A=a))$$

since, $(C \perp\!\!\!\perp B|A)$ in $G_{\overline{A}\underline{B}}$






Pearl's 2009 book, page 87

# Derivation of front-door criterion using do-calculus

Task 3: Compute $p(C|do(A=a))$ . Marginalising over B gives:

$$p(C|do(A=a)) = \sum_B p(C|B, do(A=a))p(B|do(A=a))$$

Second term already done. First term, no rule can be applied to eliminate do(A). Instead, use Rule 2 to add do(B):

$$p(C|B, do(A=a)) = p(C|do(B=b), do(A=a))$$

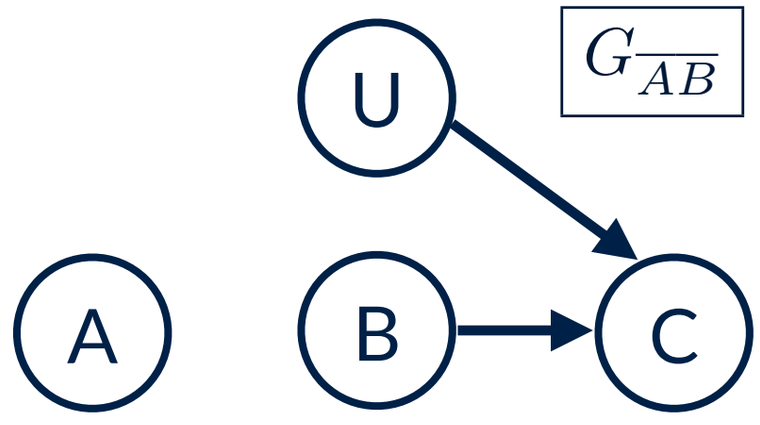
since, $(C \perp\!\!\!\perp B|A)$ in $G_{\overline{A}\underline{B}}$

Then, we use Rule 3, to delete do(A):

$$p(C|B, do(A=a)) = p(C|do(B=b))$$

since, $(C \perp\!\!\!\perp A|B)$ in $G_{\overline{AB}}$

which again, we have competed before.




Pearl's 2009 book, page 87

# Derivation of front-door criterion using do-calculus

Task 3: Compute $p(C|do(A=a))$ . Marginalising over B gives:

$$p(C|do(A=a)) = \sum_B p(C|B, do(A=a))p(B|do(A=a))$$

Putting all terms together:

$$p(C|do(A=a)) = \sum_B p(B|a) \sum_{A'} p(C|A', B)p(A')$$

Front-door criterion!



Pearl's 2009 book, page 87

# A statement about estimation

# Recall: The Backdoor Criterion

**Backdoor Criterion:** Given an ordered pair of variables (T,Y) in a DAG G, a set of variables X satisfies the backdoor criterion relative to (T,Y) if:

(i) no node in X is a descendent of T

(ii) X block every path between T and Y that contains an arrow into T

If X satisfies the backdoor criterion then the causal effect of T on Y is given by:

$$p(Y = y|do(T = t)) = \sum_x p(Y = y|T = t, X = x)p(X = x)$$

In other words, condition on a set of nodes X such that:

(i) We block all spurious paths between T and Y

(ii) We leave all direct paths from T to Y unperturbed

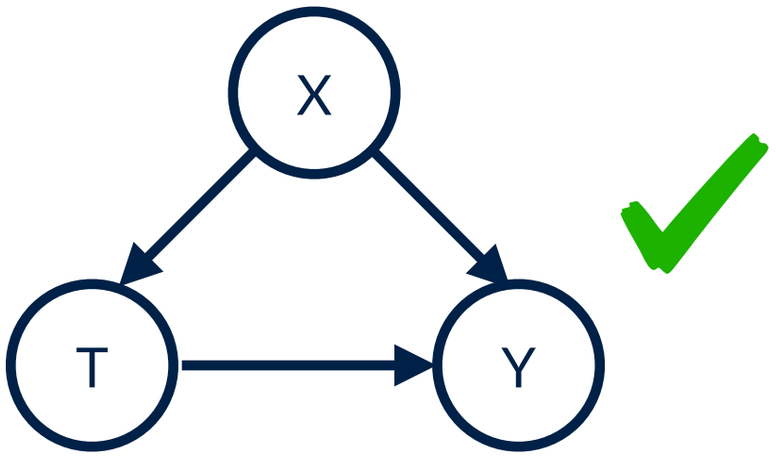
(iii) We create no new spurious paths (do not unblock any new paths)

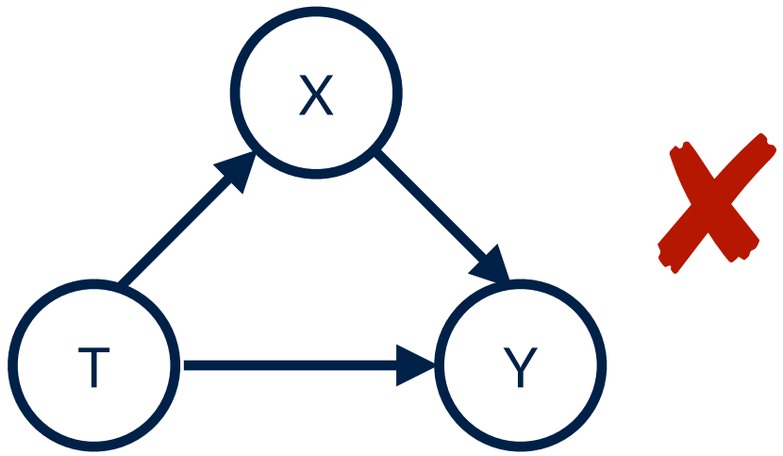
Any set X that satisfies the backdoor criterion (hence can be used in the adjustment formula) is called an **Adjustment Set**

Pearl, Causal Inference in Statistics (2016)

# Pearl: To adjust or not to adjust

Pearls algorithmic approach (*do*-calculus) tells us to adjust or not.



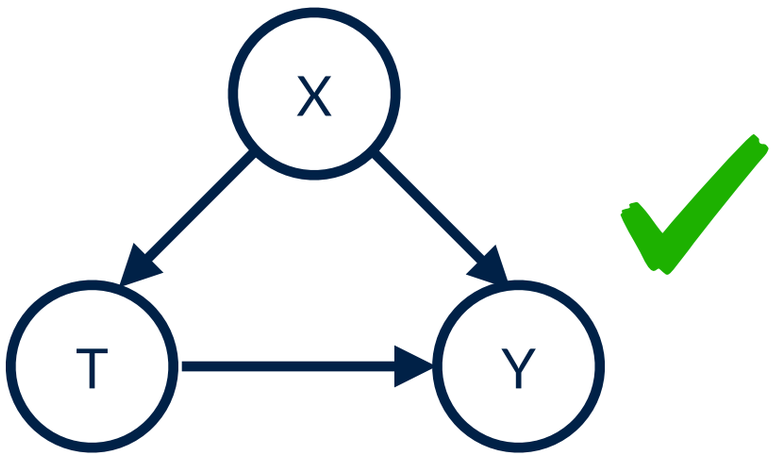



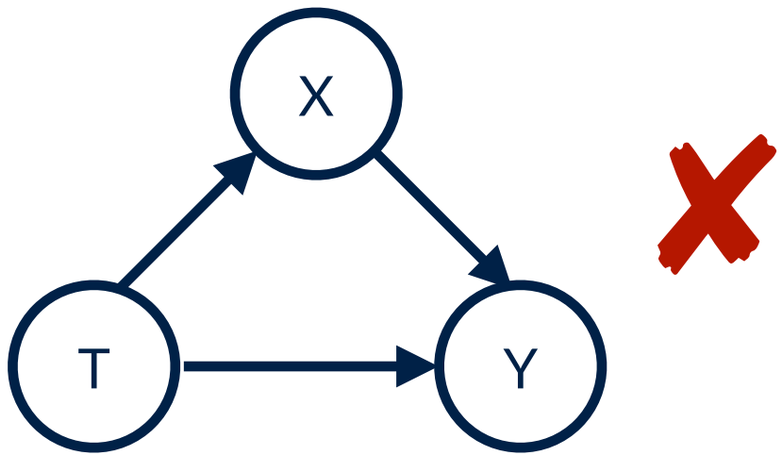



Pearl, Causal Inference in Statistics (2016)

# Pearl: To adjust or not to adjust

Pearls algorithmic approach (*do*-calculus) tells us to adjust or not.




**The Causal Effect Rule:** Given a graph G in which a set of variables PA are designated as the parents of T, the causal effect of T on Y is given by:

$$p(Y = y|do(T = t)) = \sum_x p(Y = y|T = t, PA = X)p(PA = X)$$

**Conclusion:** The set of **parents of T** is always an adjustment set for the causal effect of T on Y, i.e., to identify



Pearl, Causal Inference in Statistics (2016)

# Optimal adjustment sets

**Question:** Suppose we identify multiple adjustment sets, which do we choose?

**Idea:** We aim to estimate a causal effect, e.g., $p(Y = y|do(X) = x)$ but we do so from *observational data*. Thus, there will be some error due to finite data and the *smaller* this error, the *better* our estimate of the causal effect.

# Optimal adjustment sets

**Question:** Suppose we identify multiple adjustment sets, which do we choose?

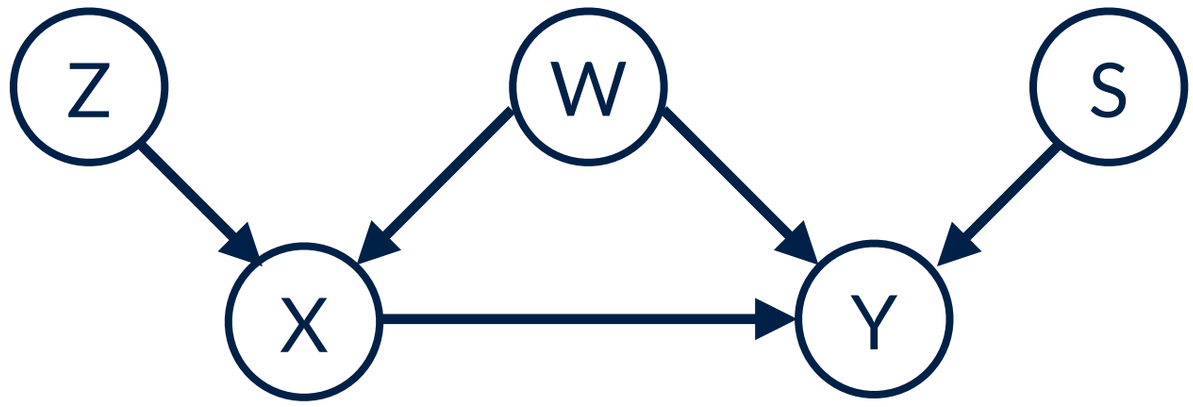
**Idea:** We aim to estimate a causal effect, e.g., $p(Y = y|do(X) = x)$ but we do so from *observational data*. Thus, there will be some error due to finite data and the *smaller* this error, the *better* our estimate of the causal effect.

**Initial guess:** The more variables we condition on, the harder it is to estimate a conditional probability or conditional expectation value …

… so the *smallest* adjustment set should be the *optimal* adjustment set!

Adjustment sets:
{W}, {W, Z}, {W, S}, and {W, S, Z}

… so it should be {W}?



# Simulation

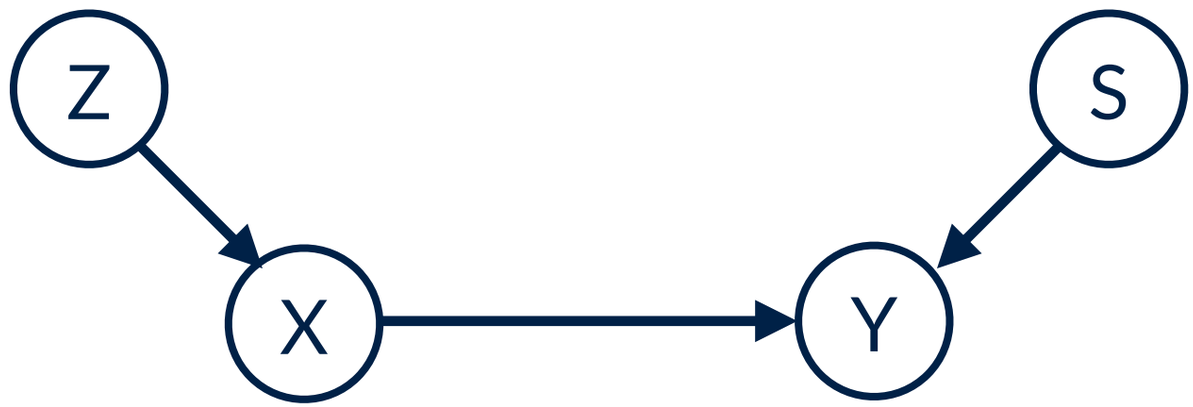
W is a confounder, so always needs to be adjusted for

For simplicity, we simulate a linear model from the causal graph below, and consider different noise on source ($\tau^2$), and noise on target ($\sigma^2$)

$$\begin{cases} X \sim \mathcal{N}(0, \tau^2) \\ Y = X + \epsilon, \qquad \text{with } \epsilon \sim \mathcal{N}(0, \sigma^2) \end{cases}$$

**Idea:** Lowering/Increasing noise on X corresponds to conditioning on Z

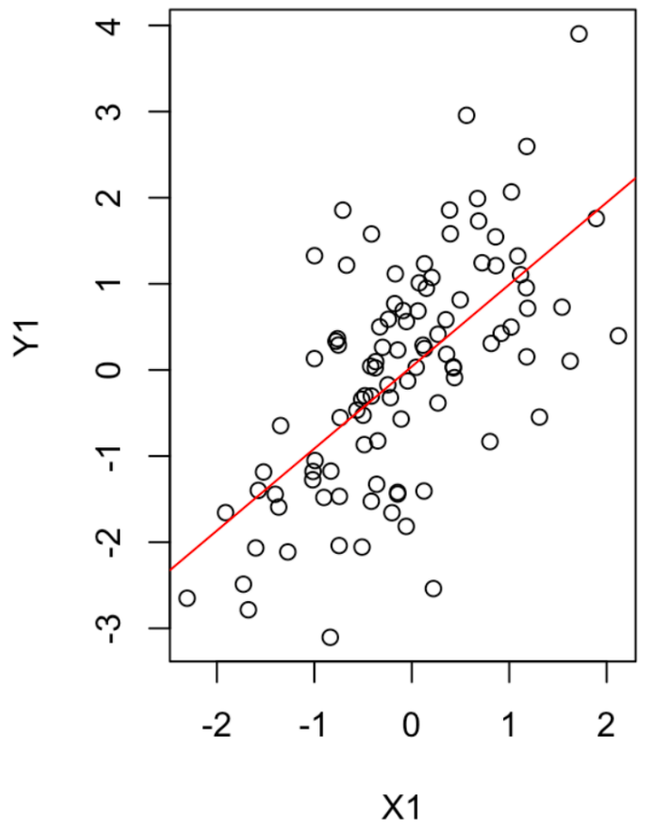
1. Conditioning on Z reduces variance in X
2. Conditioning on S reduces variance in Y



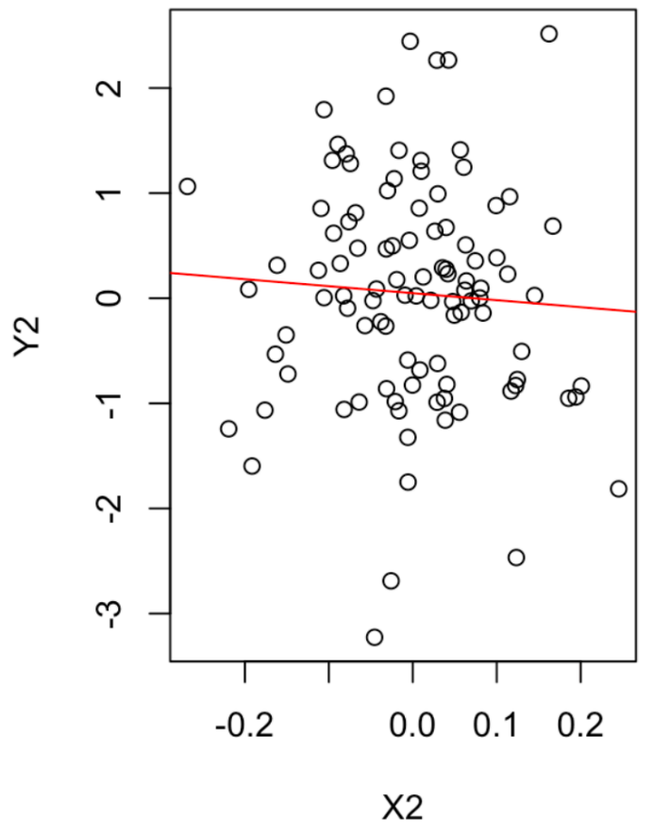
# **Simulation:** Lower noise on source X (**condition on Z**)

For simplicity, we simulate a linear model from the causal graph below, and consider different noise on source ($\tau^2$), and noise on target ($\sigma^2$)

$$\begin{cases} X \sim \mathcal{N}(0, \tau^2) \\ Y = X + \epsilon, \qquad \text{with } \epsilon \sim \mathcal{N}(0, \sigma^2) \end{cases}$$




```
##               Estimate Std. Error
## (Intercept)       0.04       0.10
## X1                0.95       0.12
```

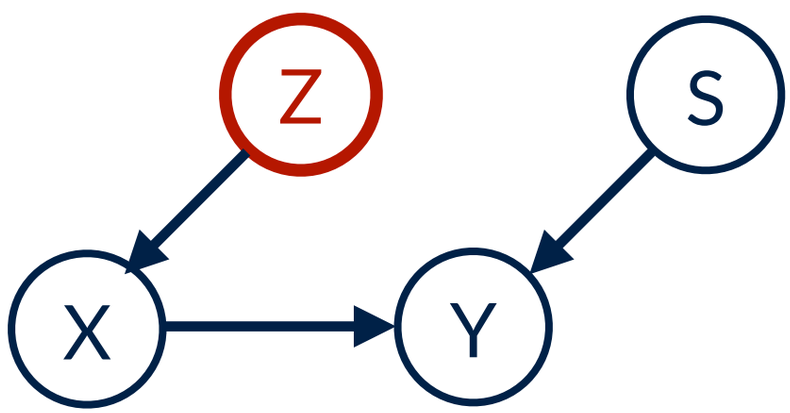



```
##               Estimate Std. Error
## (Intercept)      0.048       0.11
## X2              -0.670       1.10
```

Parameters (n=100):

$\tau_1 = 1$ vs $\tau_2 = 0.1$

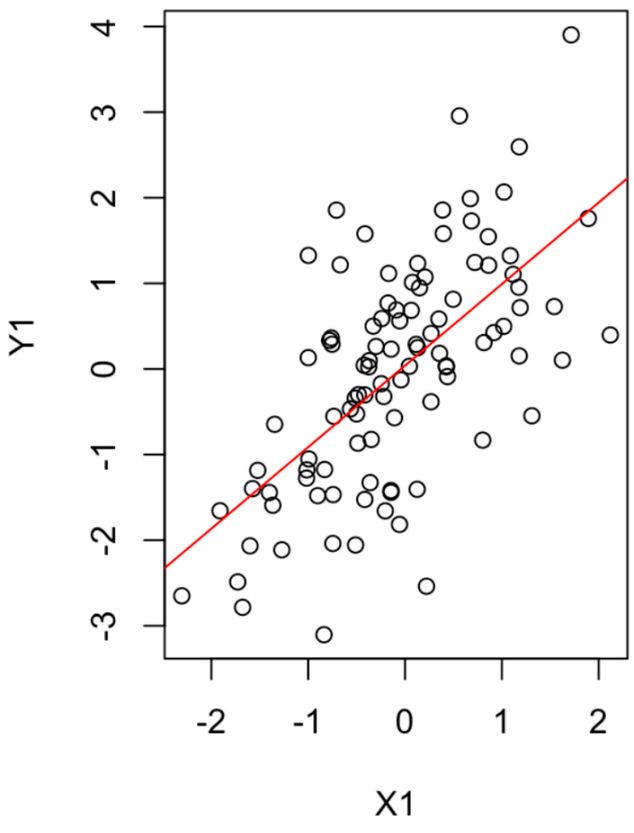
$\sigma_1 = 1$ vs $\sigma_2 = 1$

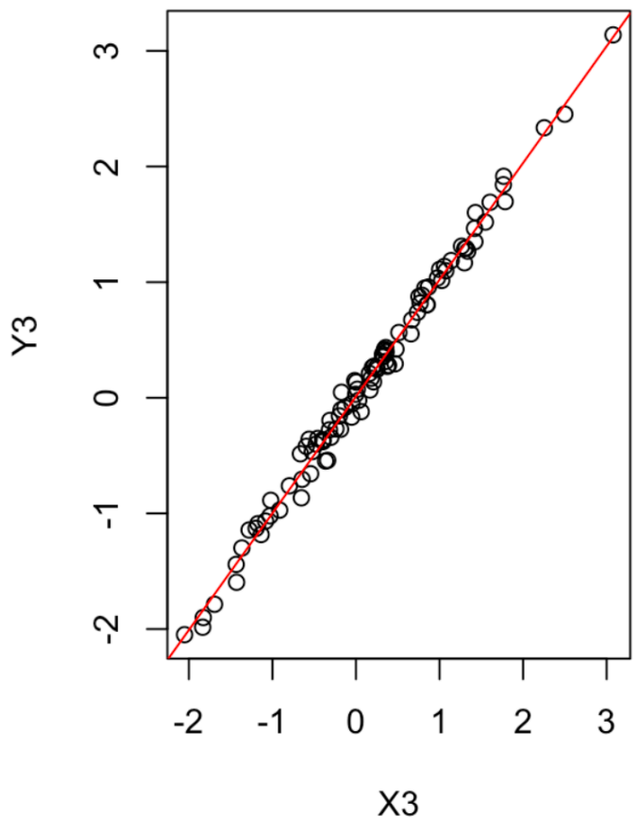


# **Simulation:** Lower noise on target Y (**condition on S**)

For simplicity, we simulate a linear model from the causal graph below, and consider different noise on source ($\tau^2$), and noise on target ($\sigma^2$)

$$\begin{cases} X \sim \mathcal{N}(0, \tau^2) \\ Y = X + \epsilon, \qquad \text{with } \epsilon \sim \mathcal{N}(0, \sigma^2) \end{cases}$$



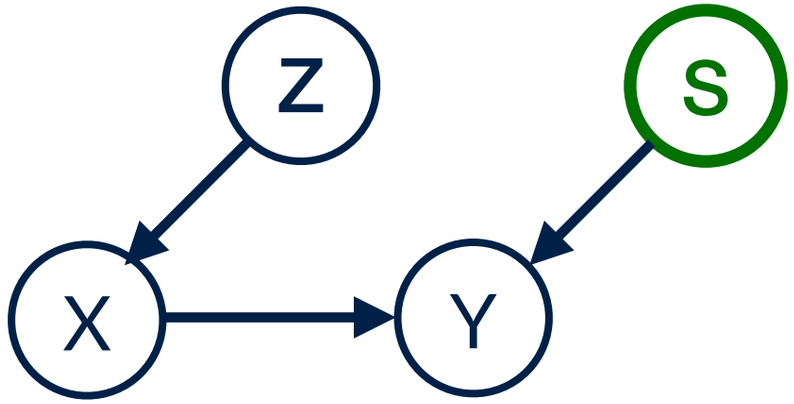

```
##               Estimate Std. Error
## (Intercept)       0.04       0.10
## X1                0.95       0.12
```

```
##               Estimate Std. Error
## (Intercept)      0.011     0.0095
## X3               1.000     0.0096
```

Parameters (n=100):

$\tau_1 = 1$ vs $\tau_2 = 1$

$\sigma_1 = 1$ vs $\sigma_2 = 0.1$



# Optimal adjustment sets

**Question:** Suppose we identify multiple adjustment sets, which do we choose?

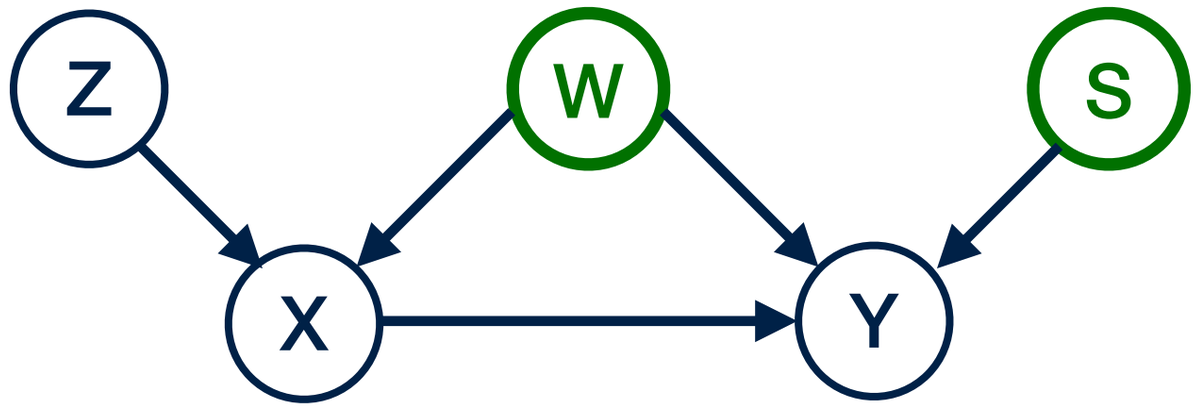
**Idea:** We aim to estimate a causal effect, e.g., $p(Y = y|do(X) = x)$ but we do so from *observational data*. Thus, there will be some error due to finite data and the *smaller* this error, the *better* our estimate of the causal effect.

**Initial guess:** The more variables we condition on, the harder it is to estimate a conditional probability or conditional expectation value …

… so the *smallest* adjustment set should be the *optimal* adjustment set!

Adjustment sets:
{W}, {W, Z}, {W, S}, and {W, S, Z}

**Optimal set is {W,S}!!!**



# Optimal adjustment sets

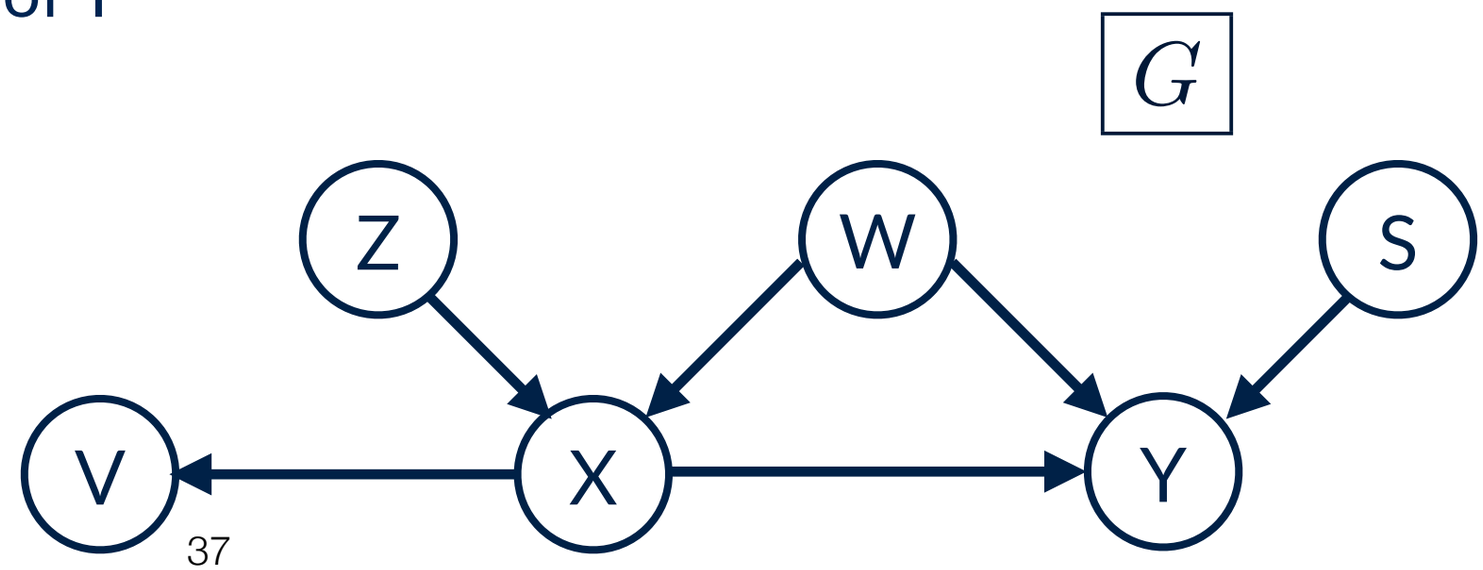
**Theorem** (Rotnitzky and Smucler, 2020)
The most efficient adjustment set to use for the effect of X on Y is

$$\mathrm{pa}_G\big(\mathrm{cn}_G(X \to Y)\big) \setminus \big(\mathrm{cn}_G(X \to Y) \cup \{X\}\big)$$

where $\mathrm{cn}_G(X \to Y)$ are all the nodes on a causal (i.e. directed) path from X to Y, but excluding X itself. (So parents of this set ***not*** on the causal path.)

**Example**
Here $\mathrm{cn}_G(X \to Y)$ consists only of Y
The parents of Y are X, W, and S

Thus, by the above the optimal adjustment set is {W,S}



# Optimal adjustment sets

**Theorem** (Rotnitzky and Smucler, 2020)
The most efficient adjustment set to use for the effect of X on Y is

$$\mathrm{pa}_G\big(\mathrm{cn}_G(X \to Y)\big) \setminus \big(\mathrm{cn}_G(X \to Y) \cup \{X\}\big)$$

where $\mathrm{cn}_G(X \to Y)$ are all the nodes on a causal (i.e. directed) path from X to Y, but excluding X itself. (So parents of this set ***not*** on the causal path.)

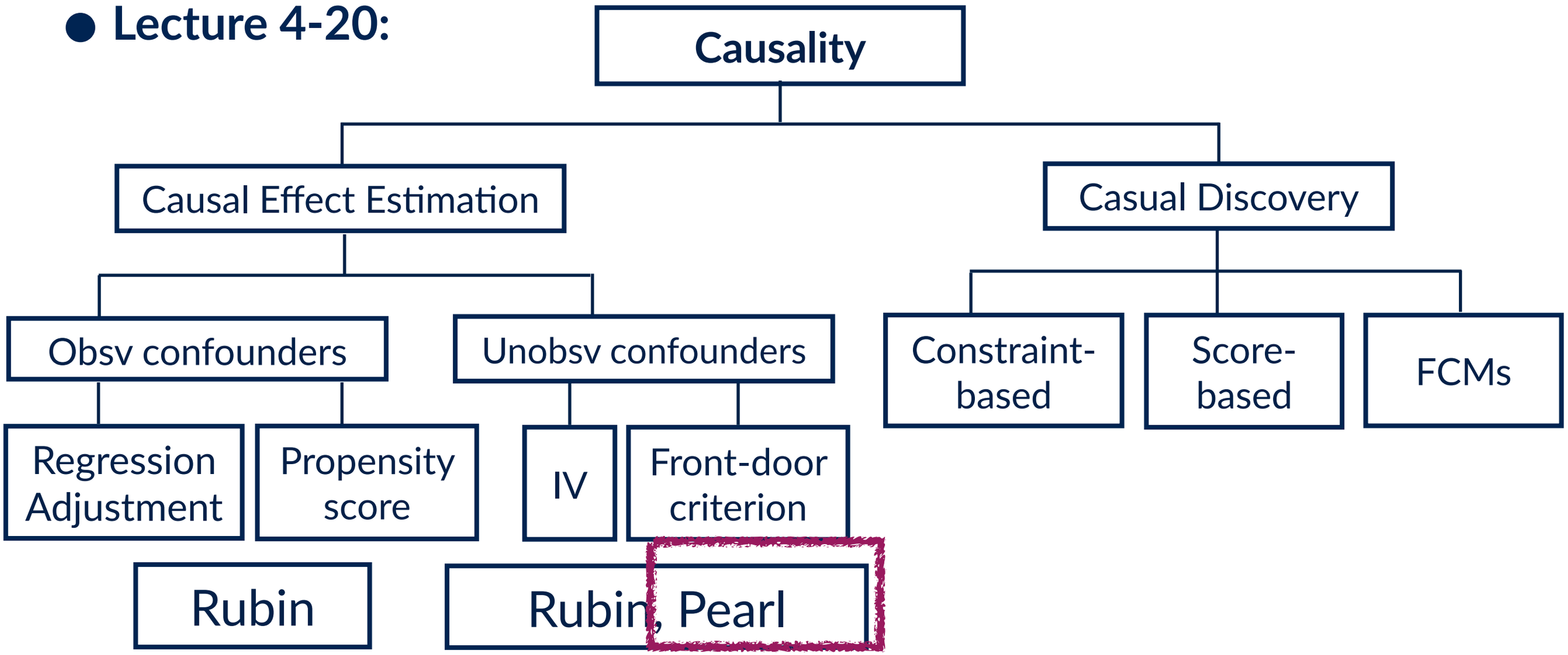
**Remarks**

1. Optimal set adjusts for some unnecessary variables (here, S) since these are not actually confounders
2. Optimal set does **not control for instruments** (here, Z)

The key quantity to keep as small as possible for optimality is $\dfrac{\text{variance in } Y}{\text{variance in } X}$

# Overview of the course

- **Lecture 1**: Introduction & Motivation, why do we care about causality? Why deriving causality from observational data is non-trivial.
- **Lecture 2:** Recap of probability theory, variables, events, conditional probabilities, independence, law of total probability, Bayes' rule
- **Lecture 3**: Recap of regression, multiple regression, graphs, SCM
- **Lecture 4-20:**



THE UNIVERSITY of EDINBURGH

# Methods for Causal Inference
# Lecture 13: Do-Calculus

Ava Khamseh

School of Informatics
2023-2024