Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam

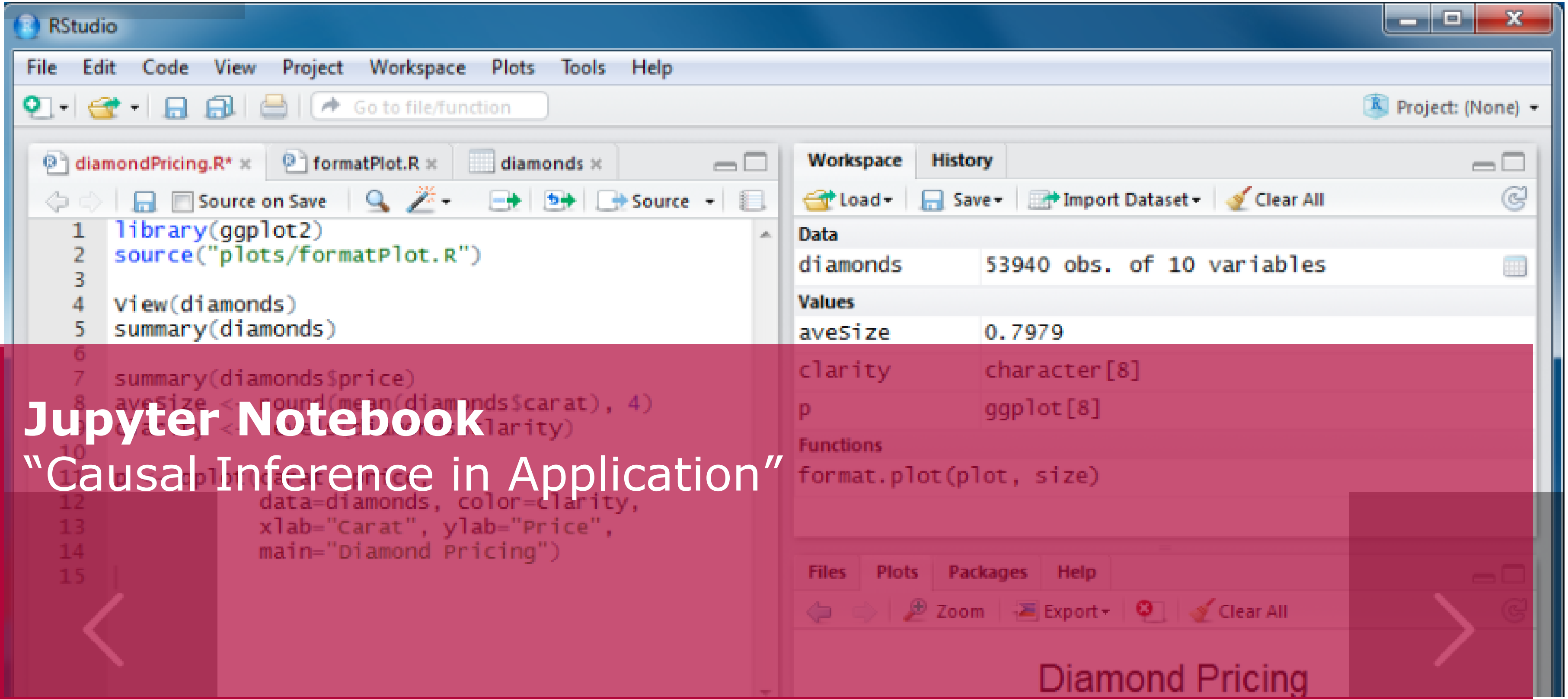
Causal Inference – Theory and Applications

Dr. Matthias Uflacker, Johannes Huegle, Christopher Schmidt

April 24, 2018

Agenda
April 24, 2018

- **Jupyter Notebook „Causal Inference in Application“**
- **Recap Causal Inference in a Nutshell**
- **Introduction to Structural Causal Models**
  1. Preliminaries
  2. Structural Causal Models
  3. (Local) Markov Condition
  4. Factorization
  5. Global Markov Condition
  6. Functional Model and Markov conditions
  7. Faithfulness
  8. Constraint-based Causal Inference
  9. Markov Equivalence Class
  10. Summary
  11. Excursion: Maximal Ancestral Graphs



Jupyter Notebook

“Causal Inference in Application”

Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam

Jupyter Notebook

Causal Inference in Application

Causal Inference - Theory and Applications

In our lecture Causal Inference - Theory and Applications, we look at the mathematical concepts that build the basis of causal inference.

Causal Inference in Application

We now look how these concepts are applied on observational data to derive causal relationships and how to use the do-operator to receive an estimation of the causal effect. In order to give you an overview on therelated procedure, this notebook gives a step by step approach in the context of a simple cooling house example.

Table of Contents

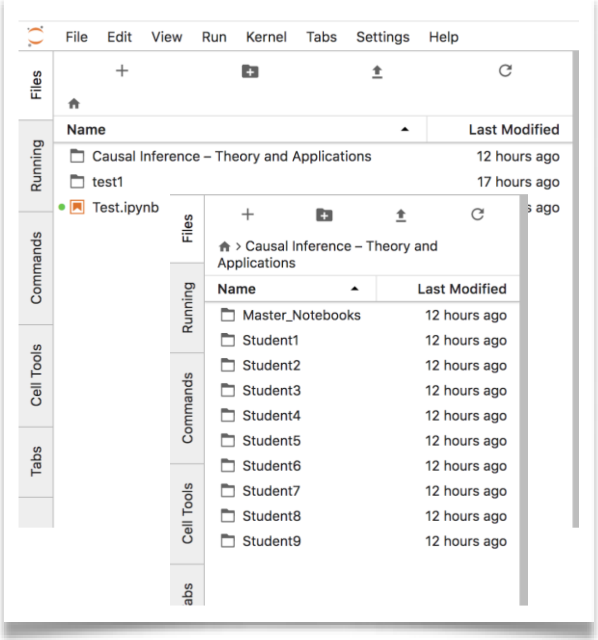
Jupyter Notebook
Access Information

System

Link will be provided via email once we have the list of participants!

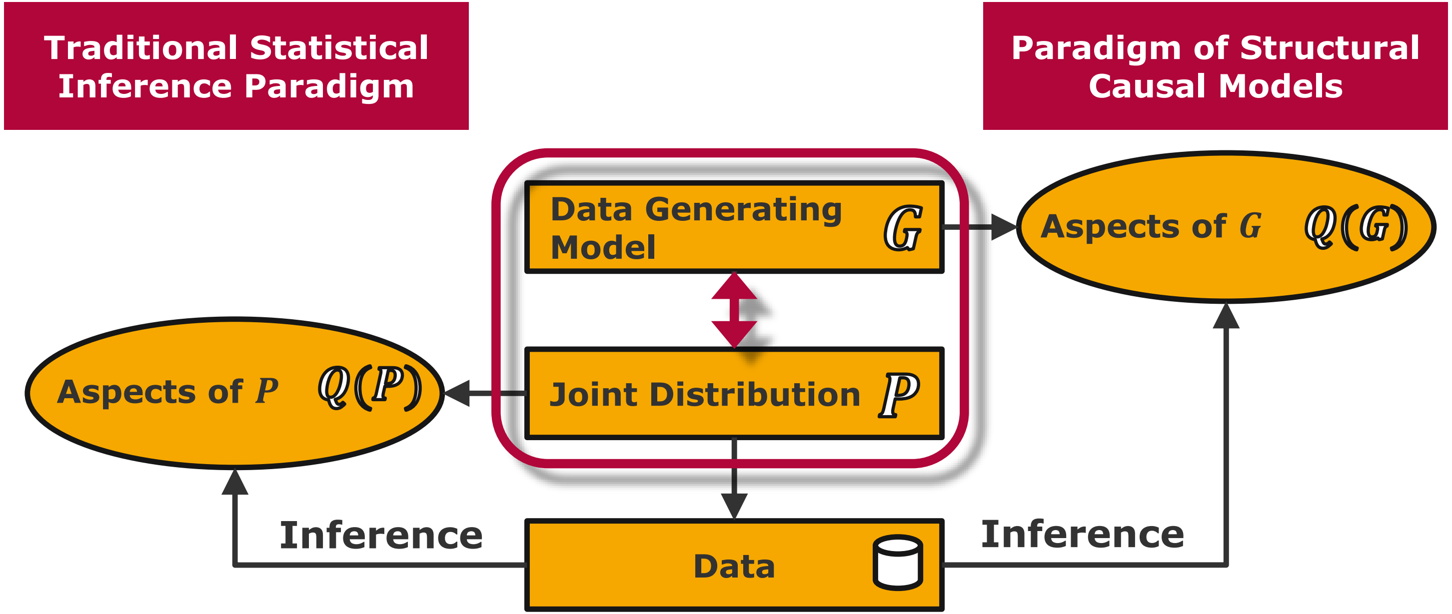
Procedure

1. Login via LDAP (standard HPI credentials)
2. Use folder Causal Inference – Theory and Applications
3. We provide a Master Notebook
   Please use as a read only resource
   Copy relevant information into your local workspace
4. Your local workspace either in your home directory or as a separate folder in our courses' folder
5. Let us know if you require new packages



Causal Inference in a Nutshell

Causal Inference in a Nutshell
Recap: The Concept



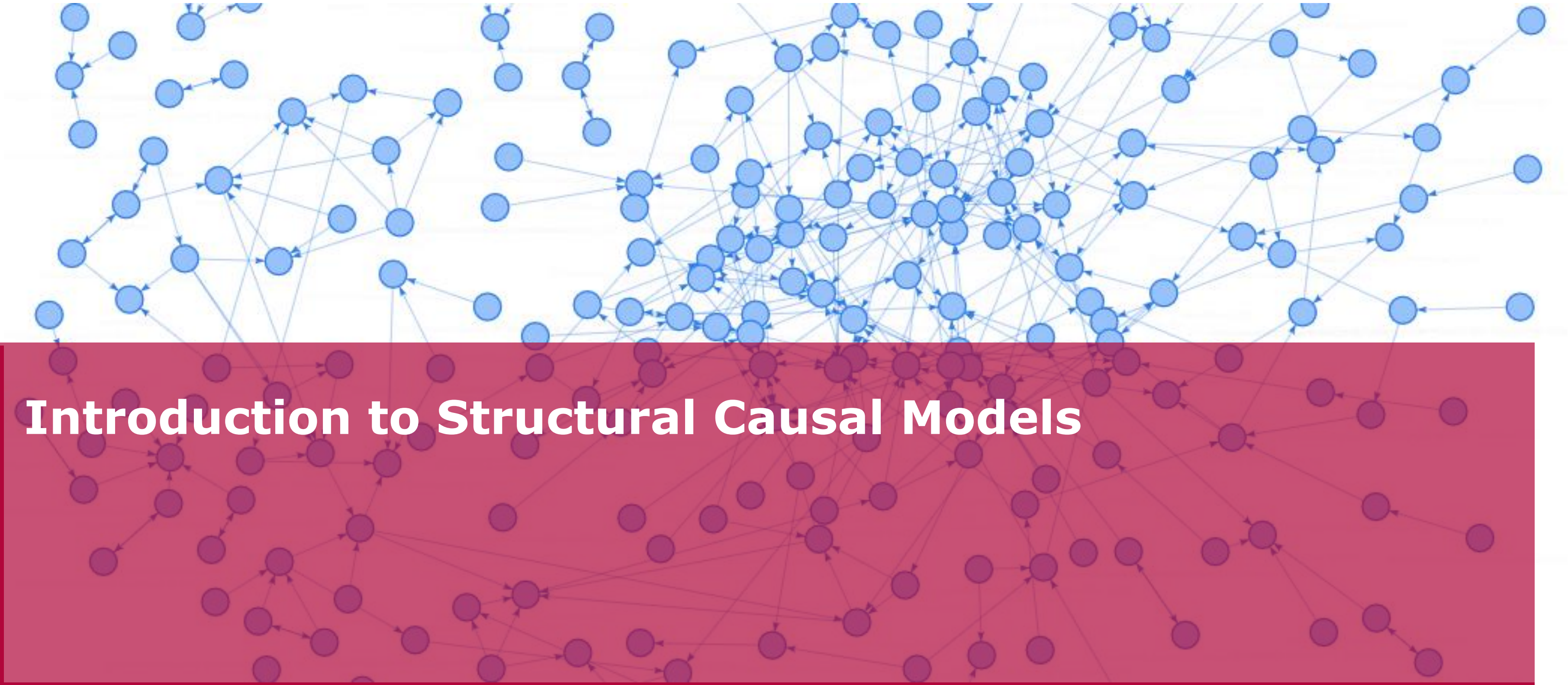
E.g., what is the sailors' probability of recovery when **we see** a treatment with lemons?

$$Q(P) = P(recovery|lemons)$$

E.g., what is the sailors' probability of recovery if **we do** treat them with lemons?

$$Q(G) = P(recovery|do(lemons))$$

Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam



Introduction to Structural Causal Models

Introduction to Causal Graphical Models
Content

1. Preliminaries
Notation

- A, B events
- X, Y, Z random variables
- x value of random variable

- Pr probability measure
- P_X probability distribution of X
- p density
- p_X or $p(X)$ density of P_X
- $p(x)$ density of P_X evaluated at the point x

- $X \perp Y$ independence of X and Y
- $X \perp Y \mid Z$ conditional independence of X and Y given Z

- Two events A and B are called *independent* if

$$\Pr(A \cap B) = \Pr(A) \cdot \Pr(B)\,,$$

  or - rewritten in *conditional probabilities - if*

$$\Pr(\mathrm{A}) = \frac{A \cap B}{B} = \Pr(A|B)\,,$$

$$\Pr(\mathrm{B}) = \frac{A \cap B}{A} = \Pr(B|A).$$

- $A_1, \dots, A_n$ are called *(mutually) independent* if for every subset $S \subset \{1, \dots, n\}$ we have

$$\Pr\left(\bigcap_{i \in S} A_i\right) = \prod_{i \in S} \Pr(A_i)\,.$$

- **Note:**
  for $n \geq 3$, pairwise independence $\Pr(A_i \cap A_j) = \Pr(A_i) \cdot \Pr(A_j)$ for all i, j does not imply (mutual) independence.

1. Preliminaries
Independence of Random Variables

- Two real-valued random variables X and Y are called *independent*,
$$X \perp Y,$$
if for every $\mathrm{x}, y \in \mathbb{R}$, the events $\{X \leq x\}$ and $\{Y \leq y\}$ are independent,
Or, in terms of densities: for all x, y,
$$p(x, y) = p(x)p(y).$$

- **Note:**
If $X \perp Y$, then $\mathrm{E[XY] = E[X]E[Y]}$, and $cov(X, Y) = E[XY] - E[X]E[Y] = 0$.
The converse is not true: If, $cov(X, Y) = 0$, then $X \perp Y$.

No correlation does not imply independence

However, we have, for large $\mathcal{F}$: $\left(\forall f, g \in \mathcal{F}: cov\big(f(X), g(Y)\big) = 0\right)$, then $X \perp Y$.

- Two real-valued random variables X and Y are called *conditionally independent* given Z,

  $$X \perp Y \mid Z \text{ or } (X \perp Y \mid Z)_P$$

  if

  $$p(x, y|z) = p(x|z)p(y|z)$$

  For all x, y and for all z s.t. $p(z) > 0$.

- **Note:**

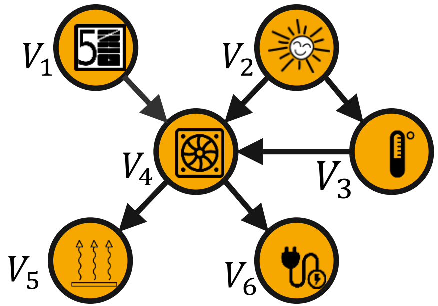
  It is possible to find X, Y which are conditionally independent given a variable Z but unconditionally dependent, and vice versa.

2. Structural Causal Models
Definition (Pearl)

- Directed Acyclic Graph (DAG) $G = (V, E)$
  - *Vertices* $V_1, \dots, V_n$
  - *Directed edges* $E = (V_i, V_j)$, i.e., $V_i \rightarrow V_j$,
  - *No cycles*
- Use kinship terminology, e.g., for path $V_i \rightarrow V_j \rightarrow V_k$
  - $V_i = Pa(V_j)$ *parent* of V_j
  - $\{V_i, V_j\} = Ang(V_k)$ *ancestors* of V_k
  - $\{V_j, V_k\} = Des(V_i)$ *descendants* of V_i
- Directed Edges encode *direct causes* via
  - $V_j = f_j(\mathrm{Pa}(\mathrm{V_j}), \mathrm{N_j})$ with independent noise $N_1, \dots, N_n$

➡ This forms the Causal Graphical Model

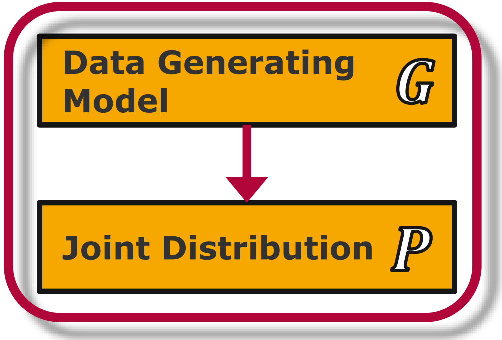
Cooling House Example:



- $V_1 = N(0,1)$
- $V_2 = N(0,1)$
- $V_3 = 3\,V_2 + N(0,1)$
- $V_4 = 4\,V_1 + 5\,V_2 + 0.7\,V_3 + N(0,1)$
- $V_5 = V_4 + N(0,1)$
- $V_6 + 1.2\,V_4 + N(0,1)$

2. Structural Causal Models
Connecting G and P

- Basic Assumption: *Causal Sufficiency*
  - All relevant variables are included in the DAG G



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

- Key Postulate: *(Local) Markov Condition*
- Essential mathematical concept: *d-separation*

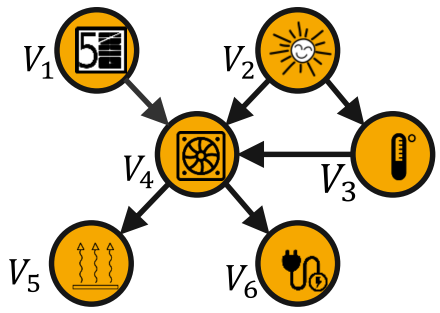
  (describes the conditional independences required by a causal DAG)

3. (Local) Markov Condition
Theorem

(Local) Markov Condition:
V_j statistically independent of nondescendants, given parents $Pa(V_j)$, i.e.,

$$\boldsymbol{V_j \perp V_{V/Des(V_j)} | Pa(V_j)}.$$

- I.e., every information exchange with its nondescendants involves its parents
- Example:



- $V_6 \perp \{V_1, V_2, V_3\} | V_4$
- $V_5 \perp \{V_1, V_2, V_3\} | V_4$

3. (Local) Markov Condition
Supplement (Lauritzen 1996)

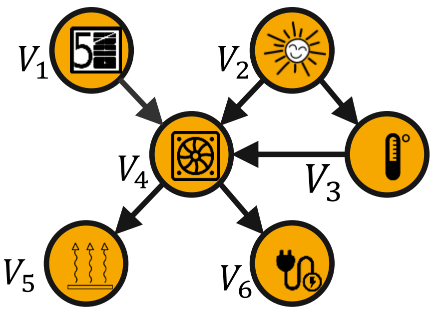
- Assume V_n has no descendants, then $ND_n = \{V_1, \ldots, V_{n-1}\}$.
- Thus the local Markov condition implies
$$V_n \perp \{V_1, \ldots, V_{n-1}\} | Pa(V_n).$$
- Hence, the general decomposition
$$p(v_1, \ldots, v_n) = p(v_n | v_1, \ldots, v_{n-1})\, p(v_1, \ldots, v_{n-1})$$
becomes
$$p(v_1, \ldots, v_n) = p(v_n | Pa(v_n))\, p(v_1, \ldots, v_{n-1}).$$
- Induction over n yields to
$$p(v_1, \ldots, v_n) = \prod_{i=1}^{n} p(v_i | Pa(v_i)).$$
- I.e., the graph shows us how to factor the joint distribution P_V.

4. Factorization
Definition

Factorization:

$$p(v_1, \dots, v_n) = \prod_{i=1}^{n} p(v_i | Pa(v_i)).$$

- I.e., conditionals as causal mechanisms generating statistical dependence
- Example:



$$
\begin{aligned}
&p(V)\\
&= p(v_1, \dots, v_n)\\
&= p(v_1) \cdot p(v_2)\\
&\quad \cdot p(v_3|v_2) \cdot p(v_4|v_1, v_2, v_3)\\
&\quad \cdot p(v_5|v_4) \cdot p(v_6|v_4)\\
&= \prod_{i=1}^{n} p(v_i|Pa(v_i))
\end{aligned}
$$

5. Global Markov Condition
D-Separation (Pearl 1988)

- *Path* = sequence of pairwise distinct vertices where consecutive ones are adjacent

- A path q is said to be *blocked* by a set S if
  - q contains a *chain* $V_i \to V_j \to V_k$ or a *fork* $V_i \leftarrow V_j \to V_k$ such that the middle node is in S, or
  - q contains a *collider* $V_i \to V_j \leftarrow V_k$ such that the middle node is not in S and such that no descendant of V_j is in S.

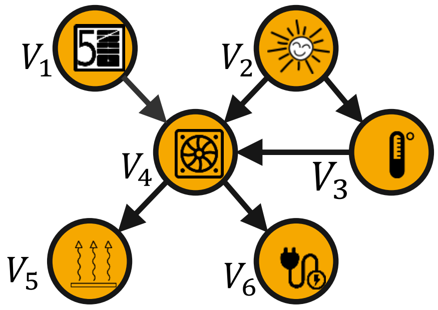
D-separation:
S is said to **d-separate** X **and** Y in the DAG G, i.e.,

$$(X \perp Y | S)_G,$$

if S blocks every path from a vertex in X to a vertex in Y.

5. Global Markov Condition
Examples of d-Separation

- Example:



- The path from V_1 to V_6 is blocked by V_4.
- V_1 and V_6 are d-separated by V_4.

- The path $V_2 \to V_3 \to V_4 \to V_6$ is blocked by V_3 or V_4 or both.
- But: V_2 and V_6 are d-separated only by V_4 or $\{V_3, V_4\}$.

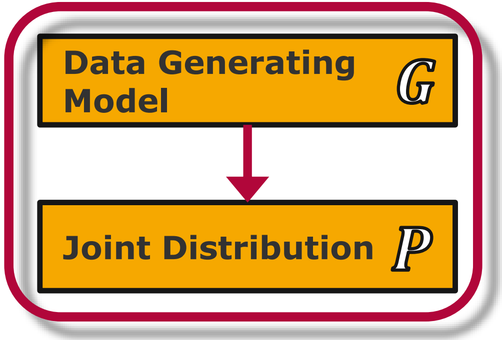
- V_1 and V_2 are not blocked by V_4.

5. Global Markov Condition
Theorem

Global Markov Condition:
For all disjoint subsets of vertices X, Y and Z we have that

$$X, Y \text{ d-separated by } Z \Rightarrow (X \perp Y \mid Z)_P .$$

- I.e., we have $(X \perp Y | Z)_G \Rightarrow (X \perp Y | Z)_P$



6. Functional Model and Markov Conditions
Theorem (Lauritzen 1996, Pearl 2000)

Theorem:

The following are equivalent:

- Existence of a *functional causal model* G;
- *Local Causal Markov condition*: V_j statistically independent of nondescendants, given parents
  (i.e.: every information exchange with its nondescendants involves its parents)
- *Global Causal Markov condition*: d-separation
  (characterizes the set of independences implied by local Markov condition)
- *Factorization:* $p(v_1, \dots, v_n) = \prod_{i=1}^{n} p(v_i | Pa(v_i))$.

(subject to technical conditions)

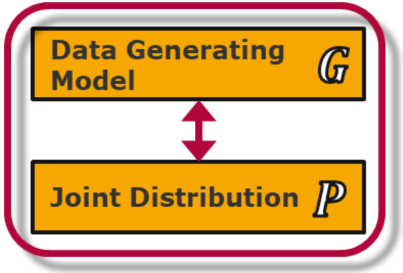
I.e., $(X \perp Y | Z)_G \Rightarrow (X \perp Y | Z)_P$

7. Causal Faithfulness
The key-postulate

Causal Faithfulness:
p is called faithful relative to G if only those independencies hold true that are implied by the Markov condition, i.e.,

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

- I.e., we assume that any population P produced by this causal graph G has the independence relations obtained by applying d-separation to it
- Seems like a hefty assumption, but it really isn't: It assumes that whatever independencies occur in it arise not from incredible coincidence but rather from structure, i.e., data generating model G.
- Hence:



8. Constraint-based Causal Inference
Concept (Spirtes, Glymor, Scheines and Pearl)

- **Assumptions:**
  - Causal Sufficiency
  - Global Markov Condition
  - Causal Faithfulness

- **Causal Structure Learning:**
  - Accept only those DAG's G as causal hypothesis for which

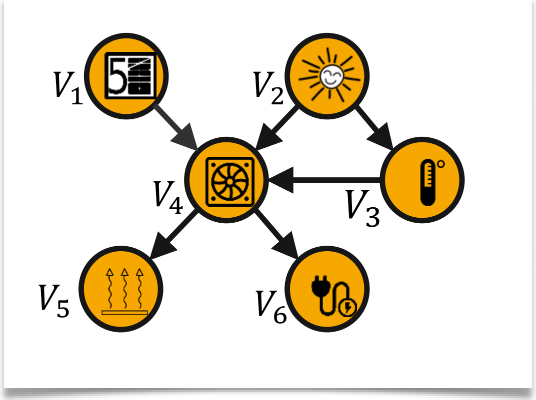
    $$(X \perp Y \mid Z)_G \Leftrightarrow (X \perp Y \mid Z)_P.$$

  - Defines the basis of constraint-based causal structure learning
  - Identifies causal DAG up to Markov equivalence class (DAGs that imply the same conditional independencies)

9. Markov Equivalence Class
Theorem (Verma and Pearl)

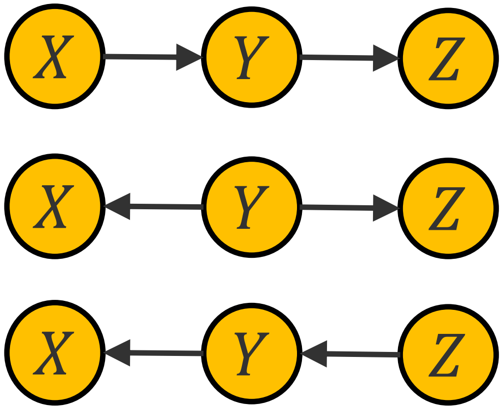
Theorem:
Two DAGs are Markov equivalent if and only if they have the same skeleton and the same v-structures

- *Skeleton:*
  corresponding undirected graph
- *v-structure:*
  substructure $X \rightarrow Y \leftarrow Z$ with no edges between X and Z.



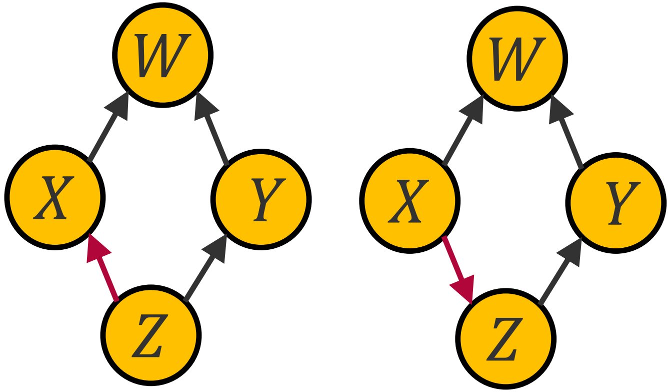
9. Markov Equivalence Class
Examples

- Same skeleton, no v-structure



$$X \perp Z \mid Y$$

- Same skeleton, same v-structure at W



10. Summary
Causal Structural Models

- Causal Structures formalized by DAG (directed acyclic graph) G with random variables $V_1, \dots, V_n$ as vertices.

- Causal Sufficiency, Causal Faithfulness and Markov Condition imply
$$(X \perp Y | Z)_G \Leftrightarrow (X \perp Y | Z)_P.$$

- Local Markov Condition states that the density $p(\mathrm{v}_1, \dots, v_n)$ then factorizes into
$$p(v_1, \dots, v_n) = \prod_{i=1}^{n} p(v_i | Pa(v_i)).$$

- Causal conditional $p(v_j | Pa(v_j))$ represent causal mechanisms.

11. Excursion: Maximal Ancestral Graphs
Motivating Example

- Suppose, we are given the following list of conditional independencies among X, Y, Z and W:
  - $X \perp\!\!\!\perp Z$,
  - $Y \perp\!\!\!\perp W$,
  - $X \perp\!\!\!\perp W$.
  - $X \not\!\perp\!\!\!\perp Y$,
  - $Y \not\!\perp\!\!\!\perp Z$,
  - $Z \not\!\perp\!\!\!\perp W$.
- Which DAG could have generated these, and only these, independencies and dependencies?
- The pattern of dependencies must be:

$$X - Y - Z - W$$

- And there must be the following colliders:

$$X \longrightarrow Y \longleftarrow Z$$
$$Y \longrightarrow Z \longleftarrow W$$

- There is no orientation of Y–Z that is consistent with the independencies.

11. Excursion: Maximal Ancestral Graphs
DAG Models and Marginalization

- Let’s include an additional variable U:

$$X \longrightarrow Y \leftarrow U \rightarrow Z \longleftarrow W$$

- This DAG model generates a probability distribution $P_{\{X,Y,Z,W,U\}}$ in which:
  - $X \perp\!\!\!\perp Z$,
  - $Y \perp\!\!\!\perp W$,
  - $X \perp\!\!\!\perp W$.
  - $X \not\!\perp\!\!\!\perp Y$,
  - $Y \not\!\perp\!\!\!\perp Z$,
  - $Z \not\!\perp\!\!\!\perp W$.

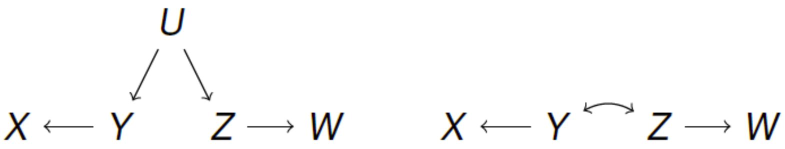
- The marginal distribution $P_{\{X,Y,Z,W\}} = P_{\{X,Y,Z,W,U\}}du$ must adhere the same independencies. But: this marginal distribution cannot be faithfully generated by any DAG.

➡ DAG models are not closed under marginalization!

11. Excursion: Maximal Ancestral Graphs
Ancestral Graphs (informally)

- **Ancestral Graph (AG)**

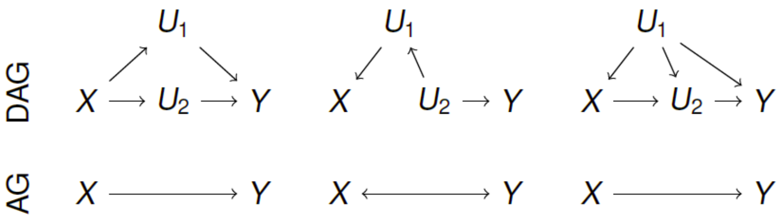
  is a graph containing both directed and bi-directed edges, where the bi-directed edges stand for *latent variables, e.g.,*



- **m-Separation**

  If S m-separates X and Y in an ancestral graph M, then X ⊥ Y | S in every density p that factorizes according to any DAG G that is represented by the AG M.

- **Example**



11. Excursion: Maximal Ancestral Graphs
DAGs vs. AGs

- **Advantages of AGs**
  - AGs can faithfully represent more probability distributions than DAGs.
  - AG models are closed under marginalization.
  - AGs can (implicitly) represent unobserved variables, which exist in many (possibly almost all) applications.

- **Disadvantages of AGs**
  - Parameterization is difficult in the general case.
  - Markov equivalence is difficult.

References

Literature

- Pearl, J. (2009). *Causal inference in statistics: An overview*. Statistics Surveys, 3:96-146.
- Pearl, J. (2009). *Causality: Models, Reasoning, and Inference.* Cambridge University Press.
- Spirtes, P., Glymour, C., and Scheines, R. (2000). Causation, Prediction, and Search. The MIT Press.

Thank you
for your attention!