Directed Graphical Models II

Independencies

Michael U. Gutmann

Probabilistic Modelling and Reasoning (INFR11134)
School of Informatics, The University of Edinburgh

Autumn Semester 2025

Recap

- Statistical independence assumptions limit the number of variables that are allowed to directly interact with each other; leads to factorisation of a pdf/pmf.
- Visualisation of factorised pdfs/pmfs as directed acyclic graphs (DAGs).
- DAGs to define sets of pdfs/pmfs in terms of a factorisation: directed graphical models (DGMs)
- The factors correspond to conditionals of the pdf/pmf, which defines a data generating process via ancestral sampling.
- DGMs satisfy conditional independencies $x_i \perp\!\!\!\perp (\mathrm{pre}_i \setminus \mathrm{pa}_i) \mid \mathrm{pa}_i$ for all i.

Program

1. Directed ordered Markov property
2. D-separation and the directed global Markov property
3. Further methods to determine independencies

Program

1. Directed ordered Markov property
   - Equivalence between factorisation and directed ordered Markov property
   - Examples

2. D-separation and the directed global Markov property

3. Further methods to determine independencies

Directed ordered Markov property

- A distribution $p(\mathbf{x})$ is said to satisfy the directed ordered Markov property relative to a DAG G if

$$x_i \perp\!\!\!\perp (\mathrm{pre}_i \setminus \mathrm{pa}_i) \mid \mathrm{pa}_i \text{ for all } i \tag{1}$$

  holds for all topological orderings of the variables wrt to G.
- Notation: $p(\mathbf{x})$ satisfies $M_o(G)$.
- We have seen that DGMs satisfy the directed ordered Markov property.
- Moreover, we have seen that satisfying the directed ordered Markov property with respect to a DAG G is equivalent to factorising over it:

$$F(G) \iff M_o(G) \tag{2}$$

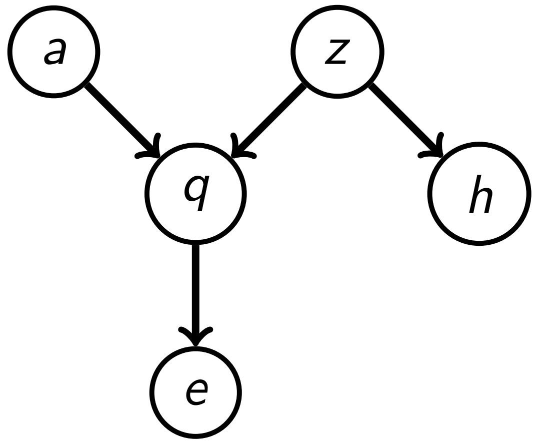
Two equivalent views on directed graphical models

1. Factorisation (generative) view of DGMs:
   - It is the set of models that you obtain by using different conditionals (tables, parametric models) $p(x_i|\mathrm{pa}_i)$ in the factorisation over G.
   - The data generating mechanisms that you obtain by using different conditionals in ancestral sampling wrt to a DAG G.
2. Independence (filtering) view of DGMs:
   - It is the set of models that satisfy $M_o(G)$.
   - The models that you obtain by filtering out from all possible models those that satisfy $M_o(G)$.

     (Similarly for further Markov properties that we will derive, the directed global Markov property $M_g(G)$ and the directed local Markov property $M_l(G)$.)

Example

DAG:



Topological ordering: (a, z, q, e, h)

Predecessor sets for the ordering:
$\mathrm{pre}_a = \varnothing, \mathrm{pre}_z = \{a\}, \mathrm{pre}_q = \{a, z\}, \mathrm{pre}_e = \{a, z, q\}, \mathrm{pre}_h = \{a, z, q, e\}$

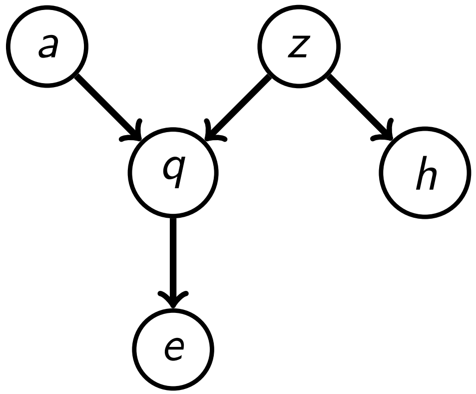
Parent sets
$\mathrm{pa}_a = \mathrm{pa}_z = \varnothing, \mathrm{pa}_q = \{a, z\}, \mathrm{pa}_e = \{q\}, \mathrm{pa}_h = \{z\}$

All models defined by the DAG satisfy $x_i \perp\!\!\!\perp (\mathrm{pre}_i \setminus \mathrm{pa}_i) \mid \mathrm{pa}_i$:

$$z \perp\!\!\!\perp a \qquad e \perp\!\!\!\perp \{a, z\} \mid q \qquad h \perp\!\!\!\perp \{a, q, e\} \mid z$$

Example (different topological ordering)

DAG:



Topological ordering: (a, z, h, q, e)

Predecessor sets for the ordering:
$\mathrm{pre}_a = \varnothing, \mathrm{pre}_z = \{a\}, \mathrm{pre}_h = \{a, z\}, \mathrm{pre}_q = \{a, z, h\}, \mathrm{pre}_e = \{a, z, h, q\}$

Parent sets: as before
$\mathrm{pa}_a = \mathrm{pa}_z = \varnothing, \mathrm{pa}_h = \{z\}, \mathrm{pa}_q = \{a, z\}, \mathrm{pa}_e = \{q\}$

All models defined by the DAG satisfy $x_i \perp\!\!\!\perp (\mathrm{pre}_i \setminus \mathrm{pa}_i) \mid \mathrm{pa}_i$:

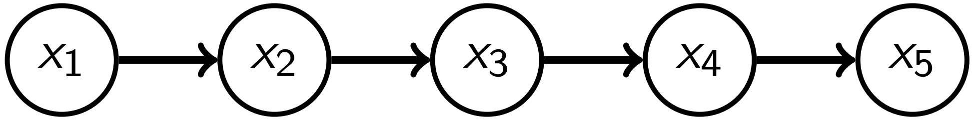
$$z \perp\!\!\!\perp a \qquad h \perp\!\!\!\perp a \mid z \qquad q \perp\!\!\!\perp h \mid a, z \qquad e \perp\!\!\!\perp \{a, z, h\} \mid q$$

Note: the models also satisfy those obtained with the previous ordering:

$$z \perp\!\!\!\perp a \qquad e \perp\!\!\!\perp \{a, z\} \mid q \qquad h \perp\!\!\!\perp \{a, q, e\} \mid z$$

Example: Markov chain

DAG:



There is only one topological ordering: $(x_1, x_2, \ldots, x_5)$

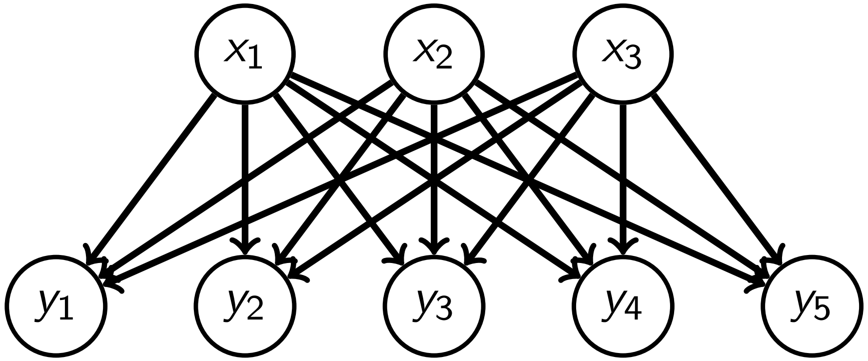
All models defined by the DAG satisfy: $x_{i+1} \perp\!\!\!\perp x_1, \ldots, x_{i-1} \mid x_i$

(future independent of the past given the present)

Example: Probabilistic PCA, factor analysis, ICA

(PCA: principal component analysis; ICA: independent component analysis)

DAG:



Topological ordering $(x_1, x_2, x_3, y_1, y_2, y_3, y_4, y_5)$

All models defined by the DAG satisfy:

$x_i \perp\!\!\!\perp x_j \qquad y_2 \perp\!\!\!\perp y_1 \mid x_1, x_2, x_3 \qquad y_3 \perp\!\!\!\perp y_1, y_2 \mid x_1, x_2, x_3$

$y_4 \perp\!\!\!\perp y_1, y_2, y_3 \mid x_1, x_2, x_3 \qquad y_5 \perp\!\!\!\perp y_1, y_2, y_3, y_4 | x_1, x_2, x_3$

y_5 is independent from all the other y_i given x_1, x_2, x_3. Using further topological orderings shows that all y_i are independent from each other given x_1, x_2, x_3.

(Marginally the y_i are not independent. The model explains possible dependencies between (observed) y_i through fewer (unobserved) x_i, see later.)

Remarks

- By using different topological orderings you can generate possibly different independence relations satisfied by the model.
- While they imply each other, deriving them from each other from the basic definition of independence may not be straightforward.
- Missing edges in a DAG cause the pa_i to be smaller than the pre_i, and thus lead to the independencies $x_i \perp\!\!\!\perp \mathrm{pre}_i \setminus \mathrm{pa}_i \mid \mathrm{pa}_i$.
- Instead of "directed ordered Markov property", we may just say "ordered Markov property" if it is clear that we are talking about DAGs.

Program

1. Directed ordered Markov property
   - Equivalence between factorisation and directed ordered Markov property
   - Examples
2. D-separation and the directed global Markov property
3. Further methods to determine independencies

Program

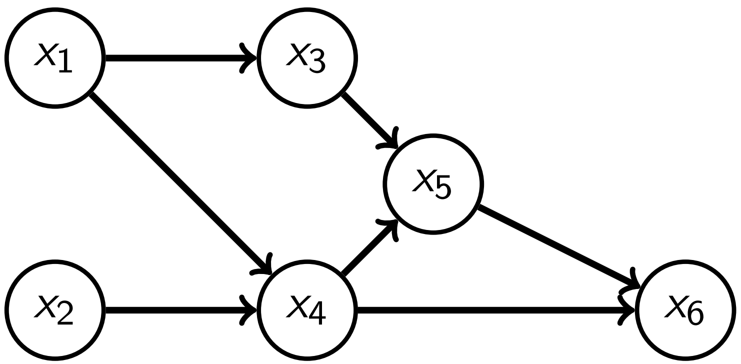
1. Directed ordered Markov property

2. D-separation and the directed global Markov property
   - Canonical connections
   - D-separation
   - Recipe and examples

3. Further methods to determine independencies

Further independence relations

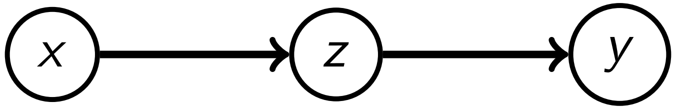
- Given the DAG below, what can we say about the independencies for the set of probability distributions that factorise over the graph?
- Is $x_1 \perp\!\!\!\perp x_2$? $\quad x_1 \perp\!\!\!\perp x_2 \mid x_6$? $\quad x_2 \perp\!\!\!\perp x_3 \mid \{x_1, x_4\}$?
- Ordered Markov properties give some independencies.
- Limitations: (1) it only allows us to condition on parent sets (2) you need to pick a topological ordering
- Directed separation (d-separation) gives further independencies.



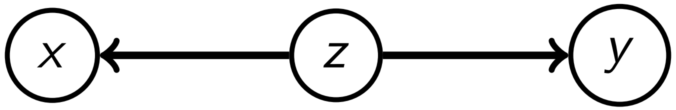
Three canonical connections in a DAG

In a DAG, two nodes x, y can be connected via a third node z in three ways:

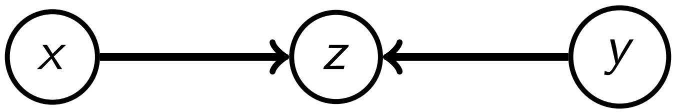
1. Serial connection (chain, head-tail or tail-head)



2. Diverging connection (fork, tail-tail)



3. Converging connection (collider, head-head, v-structure)



Note: in any case, the sequence x, z, y forms a trail

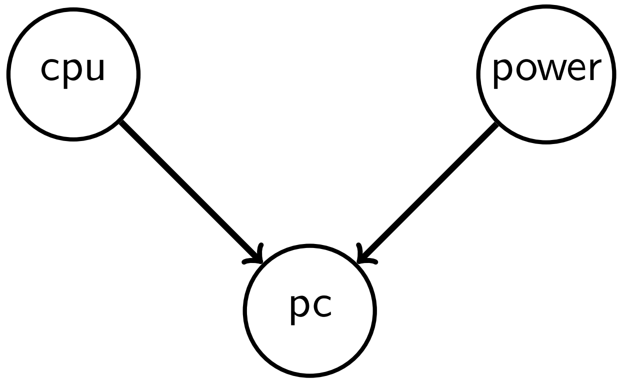
Independencies for the three canonical connections

Connection	$p(x, y)$	$p(x, y \mid z)$	z node
$x \rightarrow z \rightarrow y$	$x \not\perp\!\!\!\perp y$	$x \perp\!\!\!\perp y \mid z$	default: open instantiated: closed
$x \leftarrow z \rightarrow y$	$x \not\perp\!\!\!\perp y$	$x \perp\!\!\!\perp y \mid z$	default: open instantiated: closed
$x \rightarrow z \leftarrow y$	$x \perp\!\!\!\perp y$	$x \not\perp\!\!\!\perp y \mid z$	default: closed with evidence: opens

Think of the z node as a valve or gate through which evidence (probability mass) can flow. Depending on the type of the connection, it's default state is either open or closed.
Instantiation/conditioning/evidence acts as a switch on the valve.

Colliders model "explaining away"

Example:



- One day your computer does not start and you bring it to a repair shop. You think the issue could be the power unit or the cpu.
- Investigating the power unit shows that it is damaged. Is the cpu fine?
- Without further information, finding out that the power unit is damaged typically reduces our belief that the cpu is damaged

$$\text{power} \not\perp\!\!\!\perp \text{cpu} \mid \text{pc}$$

- Finding out about the damage to the power unit *explains away* the observed start-issues of the computer.

D-separation

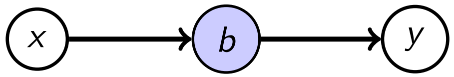
Let $X = \{x_1, \ldots, x_n\}$, $Y = \{y_1, \ldots, y_m\}$, and $Z = \{z_1, \ldots, z_r\}$ be three disjoint sets of nodes in the graph. Assume all z_i are observed (instantiated).

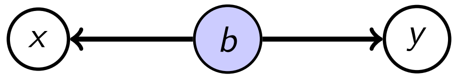
- Two nodes x_i and y_j are said to be d-separated by Z if all trails between them are blocked by Z.
- The sets X and Y are said to be d-separated by Z if every trail from any variable in X to any variable in Y is blocked by Z.

D-separation

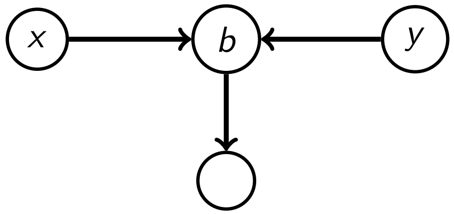
A trail between nodes x and y is blocked by Z if there is a node b on the trail such that

1. either b is part of a head-tail or tail-tail connection along the trail and b is in Z,





2. or b is part of a head-head (collider) connection along the trail and neither b nor any of its descendants are in Z.



We treat each segment of the trail as a canonical connection.

D-separation and conditional independence

Theorem: If X and Y are d-separated by Z
then $X \perp\!\!\!\perp Y \mid Z$ for all probability distributions that factorise over the DAG.

For those interested: A proof can be found in Section 2.8 of *Bayesian Networks – An Introduction* by Koski and Noble (not examinable)

Important because:

1. the theorem allows us to read off (conditional) independencies from the graph
2. no restriction on the sets X, Y, Z
3. the theorem shows that statistical independencies detected by d-separation, which is purely a graph-based criterion, do always hold. They are "true positives" ("soundness of d-separation").

Directed global Markov property $M_g(G)$

- Distributions $p(\mathbf{x})$ are said to satisfy the directed global Markov property with respect to the DAG G, or $M_g(G)$, if for any triple X, Y, Z of disjoint subsets of nodes such that X and Y are d-separated by Z in G, we have $X \perp\!\!\!\perp Y | Z$.
- *Global* Markov property because we do not restrict the sets X, Y, Z.
- The theorem says that $F(G) \Longrightarrow M_g(G)$.
- We thus have so far $M_o(G) \iff F(G) \Longrightarrow M_g(G)$.

What if two sets of nodes are not d-separated?

Theorem: If X and Y are not d-separated by Z then $X \not\perp\!\!\!\perp Y \mid Z$ in some probability distributions that factorise over the DAG.

For those interested: A proof sketch can be found in Section 3.3.1 of *Probabilistic Graphical Models* by Koller and Friedman (not examinable).

"not d-separated" is also called "d-connected"

$\not\perp\!\!\!\perp$ means statistically dependent

What if two sets of nodes are not d-separated?

- However, it can also be that d-connected variables are independent for some distributions that factorise over the graph.
- Example (Koller, Example 3.3): $p(x, y)$ with $x, y \in \{0, 1\}$ and

$$p(y = 0|x = 0) = a \quad p(y = 0|x = 1) = a$$

  for $a > 0$ and some non-zero $p(x = 0)$.
- Graph has arrow from x to y. Variables are not d-separated.

- $p(y = 0) = ap(x = 0) + ap(x = 1) = a$,
  which is $p(y = 0|x)$ for all x.
- $p(y = 1) = (1 - a)p(x = 0) + (1 - a)p(x = 1) = 1 - a$,
  which is $p(y = 1|x)$ for all x.
- Hence: $p(y|x) = p(y)$ so that $x \perp\!\!\!\perp y$.

D-separation is not complete

- This means that d-separation does generally not reveal all independencies in all probability distributions that factorise over the graph.
- In other words, individual probability distributions that factorise over the graph may have further independencies not included in the set obtained by d-separation. This is because the graph criteria do not operate on the numerical values of the factors but only on "whom affects whom", i.e. the parent-children relationships.
- We say that d-separation is not "complete" ("recall-rate" is not guaranteed to be 100%).

Recipe to determine whether two nodes are d-separated

1. Determine all trails/paths between x and y (note: direction of the arrows does here not matter).
2. For each trail:
   i Determine the default state of all nodes on the trail.
      - open if part of a head-tail or a tail-tail connection
      - closed if part of a head-head connection

   ii Check whether the set of observed nodes Z switches the state of the nodes on the trail.

   iii The trail is blocked if it contains a closed node.
3. The nodes x and y are d-separated if all trails between them are blocked.

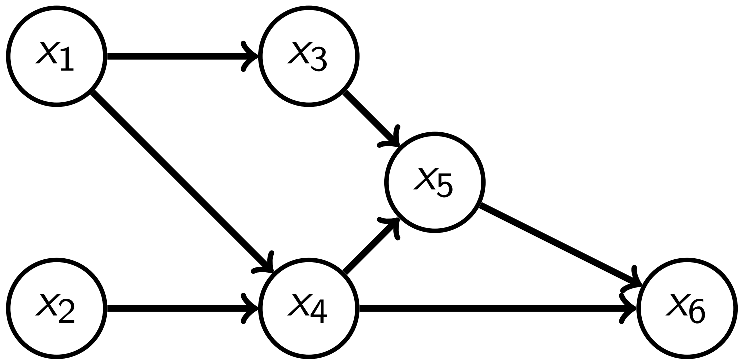
Example: Are x_1 and x_2 d-separated?

Follows from ordered Markov property, but let us answer it with d-separation.

1. Determine all trails between x_1 and x_2
2. For trail x_1, x_4, x_2
   - i default state
   - ii conditioning set is empty
   - iii $\Rightarrow$ Trail is blocked

   For trail x_1, x_3, x_5, x_4, x_2
   - i default state
   - ii conditioning set is empty
   - iii $\Rightarrow$ Trail is blocked

   Trail $x_1, x_3, x_5, x_6, x_4, x_2$ is blocked too (same arguments).
3. $\Rightarrow$ x_1 and x_2 are d-separated.

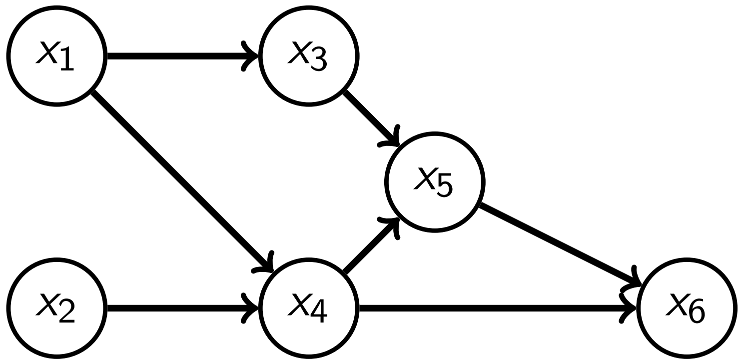


$x_1 \perp\!\!\!\perp x_2$ for all probability distributions that factorise over the graph.

Example: Are x_1 and x_2 d-separated by x_6?

1. Determine all trails between x_1 and x_2
2. For trail x_1, x_4, x_2
   i default state
   ii influence of x_6
   iii $\Rightarrow$ Trail not blocked

No need to check the other trails: x_1 and x_2 are not d-separated by x_6



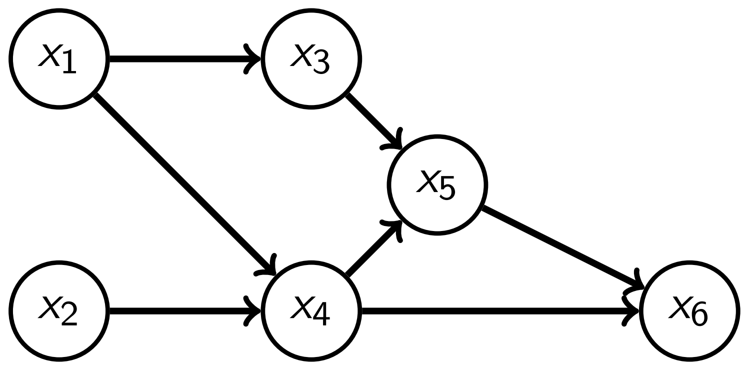
$x_1 \perp\!\!\!\perp x_2 \mid x_6$ does not hold for all probability distributions that factorise over the graph.

Example: Are x_2 and x_3 d-separated by x_1 and x_4?

1. Determine all trails between x_2 and x_3
2. For trail x_3, x_1, x_4, x_2
   i default state
   ii influence of $\{x_1, x_4\}$
   iii $\Rightarrow$ Trail blocked

   For trail x_3, x_5, x_4, x_2
   i default state
   ii influence of $\{x_1, x_4\}$
   iii $\Rightarrow$ Trail blocked

   Trail x_3, x_5, x_6, x_4, x_2 is blocked too (same arguments).
3. $\Rightarrow$ x_2 and x_3 are d-separated by x_1 and x_4.

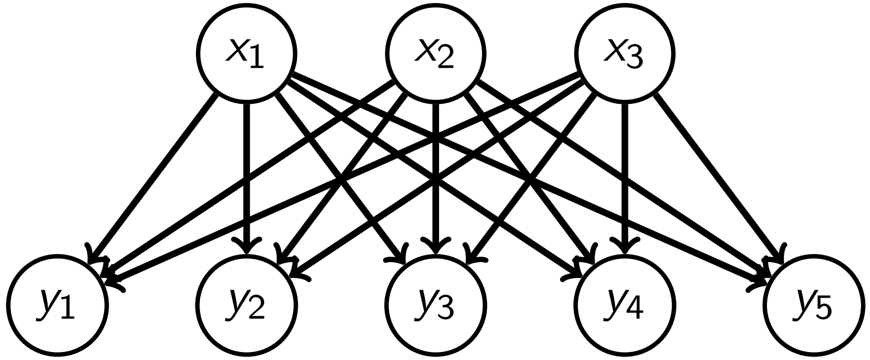


$x_2 \perp\!\!\!\perp x_3 \mid \{x_1, x_4\}$ for all probability distributions that factorise over the graph.

Example: Probabilistic PCA, factor analysis, ICA

(PCA: principal component analysis; ICA: independent component analysis)

DAG:



- From ordered Markov property: e.g. $y_5 \perp\!\!\!\perp y_1, y_2, y_3, y_4 | x_1, x_2, x_3$.
- Via d-separation: $y_i \not\perp\!\!\!\perp y_k$ since the x are in a tail-tail connection with the y's.
- Via d-separation: $x_i \perp\!\!\!\perp x_j$ since all trails between them go through y's that are in a collider configuration.
- Via d-separation: $x_i \not\perp\!\!\!\perp x_j \mid y_k$ for any $i, j, k, (i \neq j)$. This is the "explaining away" phenomenon.

Program

1. Directed ordered Markov property
2. D-separation and the directed global Markov property
    - Canonical connections
    - D-separation
    - Recipe and examples
3. Further methods to determine independencies

Program

1. Directed ordered Markov property
2. D-separation and the directed global Markov property
3. Further methods to determine independencies
   - Directed local Markov property
   - Equivalences
   - Markov blanket

Directed local Markov property

- The independencies that you can obtain with the ordered Markov property depend on the topological ordering chosen while d-separation may require you to check multiple trails.
- We next introduce the "directed local Markov property" that only depends on the graph and is easier to use.
- We say that $p(\mathbf{x})$ satisfies the directed local Markov property, $M_l(G)$ with respect to DAG G if

$$x_i \perp\!\!\!\perp (\mathrm{nondesc}(x_i) \setminus \mathrm{pa}_i) \mid \mathrm{pa}_i$$

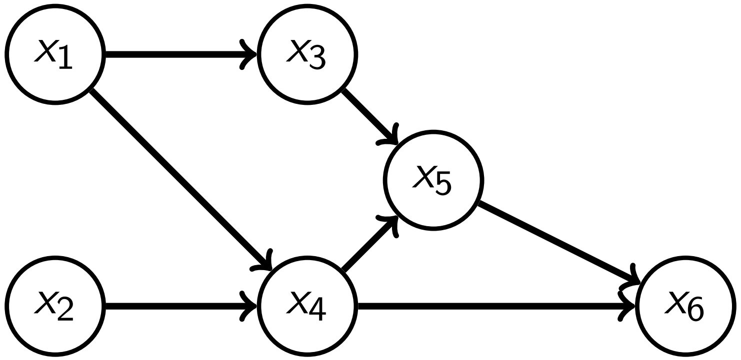
  holds for all i, where pa_i denotes the parents and $\mathrm{nondesc}(x_i)$ the non-descendants of x_i.
- Independence means that $p(x_i|\mathrm{nondesc}(x_i)) = p(x_i|\mathrm{pa}_i)$
- We have that DGMs satisfy $M_l(G)$ (see later for a proof)

Example: do we have $x_3 \perp\!\!\!\perp \{x_2, x_4\}|x_1$?

- We note that $\mathrm{nondesc}(x_3) = \{x_1, x_2, x_4\}$ and hence

$$x_3 \perp\!\!\!\perp \{x_1, x_2, x_4\} \setminus x_1 \mid x_1$$

  so that $x_3 \perp\!\!\!\perp \{x_2, x_4\}|x_1$.
- We could have used d-separation checking three trails.
- We could have considered the topological ordering: $x_1, x_2, x_4, x_3, x_5, x_6$ so that $\mathrm{pre}_3 = \{x_1, x_2, x_4\}$ from which the independence follows from $M_o(G)$.



- The topological ordering $x_1, x_2, x_3, x_4, x_5, x_6$ only gives $\mathrm{pre}_3 = \{x_1, x_2\}$ and hence $x_3 \perp\!\!\!\perp x_2|x_1$ but not $x_3 \perp\!\!\!\perp \{x_2, x_4\}|x_1$.

Equivalences between factorisation and Markov properties

For a DAG G with nodes (random variables) x_i and parent sets pa_i, we have the following equivalences:

$p(\mathbf{x})$ satisfies $F(G)$ — $p(\mathbf{x}) = \prod_{i=1}^{d} k(x_i|\mathrm{pa}_i) = \prod_{i=1}^{d} p(x_i|\mathrm{pa}_i)$

$\Updownarrow$

$p(\mathbf{x})$ satisfies $M_o(G)$ — $x_i \perp\!\!\!\perp (\mathrm{pre}_i \setminus \mathrm{pa}_i) \mid \mathrm{pa}_i$ for all i and any topol. ordering

$\Updownarrow$

$p(\mathbf{x})$ satisfies $M_l(G)$ — $x_i \perp\!\!\!\perp (\mathrm{nondesc}(x_i) \setminus \mathrm{pa}_i) \mid \mathrm{pa}_i$ for all i

$\Updownarrow$

$p(\mathbf{x})$ satisfies $M_g(G)$ — independencies asserted by d-separation

F: factorisation property, M_o: directed ordered MP, M_l: directed local MP, M_g: directed global MP (MP: Markov property)

Broadly speaking, the graph serves two related purposes:

1. it tells us how distributions factorise
2. it represents the independence assumptions made

While equivalent they have different powers

- If a distribution $p(\mathbf{x})$ satisfies $M_o(G)$ then it must satify $M_l(G)$ and $M_o(G)$ and vice versa.
- This is about what a distribution $p(\mathbf{x})$ obeys, not about what each criterion can infer.
- Local/ordered Markov properties give a small subset of conditional independencies (just enough to ensure the full factorization).
- They don't show all independencies; just enough to (theoretically) derive the rest.
- D-separation gives the entire set of independencies implied by the graph—it is the most complete tool for reading off conditional independencies.
- While most expressive, d-separation may also take longest to use as you have to identify all trails.

What can we do with the equivalences?

- If we know the factorisation of a $p(\mathbf{x})$ in terms of conditional pdfs/pmfs, we can build a graph G such that $p(\mathbf{x})$ satisfies $F(G)$ and then use the graph to determine independencies that $p(\mathbf{x})$ satisfies.
- Similarly, if for some ordering of the random variables, we know the independencies $x_i \perp\!\!\!\perp (\mathrm{pre}_i \setminus \pi_i) \mid \pi_i$ that $p(\mathbf{x})$ satisfies, where π_i is a minimal subset of the predecessors, we can obtain a graph G by identifying the π_i with the parents pa_i in a graph. By construction, $p(\mathbf{x})$ satisfies $M_o(G)$. From the graph we can obtain the factorisation of $p(\mathbf{x})$ and further independencies.
- We can start with the graph and check which independencies it implies, and, when happy, define a set of pdfs/pdfs that all satisfy the specified independencies.

Some topics we haven't covered

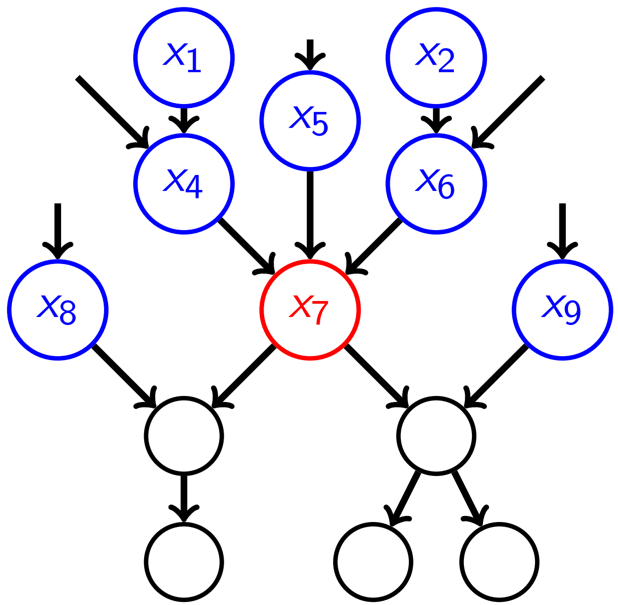
- How to determine a graph G from an arbitrary set of independencies
- How to learn the graph from samples from $p(\mathbf{x})$ (structure learning)
- These are difficult topics:
  - Multiple DAGs may express the same independencies and there may be no DAG that expresses all desired independencies (see later)
  - Learning the graph from samples involves independence tests which are not 100% accurate and errors propagate and may change the structure of the resulting DAG.
- Areas of active research, in particular in the field of causality.

Proof for equivalence (not examinable)

- We have already shown that $M_o(G) \iff F(G) \Longrightarrow M_g(G)$.
- We now show that $M_o(G) \iff M_l(G)$.
- We would like to show that:

$$x_i \perp\!\!\!\perp (\mathrm{pre}_i \setminus \mathrm{pa}_i) \,|\mathrm{pa}_i \iff x_i \perp\!\!\!\perp (\mathrm{nondesc}(x_i) \setminus \mathrm{pa}_i) \,|\mathrm{pa}_i$$

$$x_i \equiv x_7$$
$$\mathrm{pa}_7 = \{x_4, x_5, x_6\}$$
$$\mathrm{pre}_7 = \{x_1, x_2, \ldots, x_6\}$$
$\mathrm{nondesc}(x_7)$ in blue



Proof for equivalence (not examinable)

$x_i \perp\!\!\!\perp \mathrm{pre}_i \setminus \mathrm{pa}_i | \mathrm{pa}_i \Leftarrow x_i \perp\!\!\!\perp \mathrm{nondesc}(x_i) \setminus \mathrm{pa}_i | \mathrm{pa}_i$ follows because
(1) $\{x_1, \ldots, x_{i-1}\} \subseteq \mathrm{nondesc}(x_i)$ for all topological orderings, and
(2) $x \perp\!\!\!\perp \{y, w\} \mid z$ implies that $x \perp\!\!\!\perp y \mid z$ and $x \perp\!\!\!\perp w \mid z$.

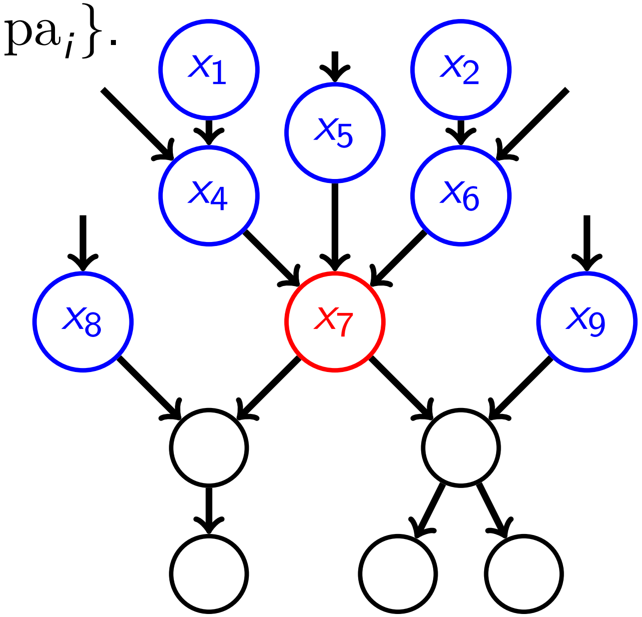
For $\Rightarrow$, assume $p(\mathbf{x})$ follows the ordered Markov property. It then factorises over the graph and hence satisfies $M_g(G)$, and we can use d-separation to establish independence.

Consider all trails from x_i to $\{\mathrm{nondesc}(x_i) \setminus \mathrm{pa}_i\}$.

Two cases: move upwards or downwards:

(1) upward trails are blocked by the parents

(2) downward trails must contain a head-head (collider) connection because the $x_j \in \{\mathrm{nondesc}(x_i) \setminus \mathrm{pa}_i\}$ is a non-descendant. These paths are blocked because the collider node or its descendants are never part of pa_i.



The result follows because all paths from x_i to all elements in $\{\mathrm{nondesc}(x_i) \setminus \mathrm{pa}_i\}$ are blocked.

Proof for equivalence (not examinable)

- For a DAG G, we have established the following relationships:

$$M_g(G) \Longleftarrow F(G) \iff M_o(G) \iff M_l(G)$$

- We can close the loop by showing that $M_g(G) \Longrightarrow M_l(G)$.
- If $p(\mathbf{x})$ satisfies $M_g(G)$ we can use d-separation to read our dependencies.
- The same reasoning as in the second part of the proof above shows that $x_i \perp\!\!\!\perp (\mathrm{nondesc}(x_i) \setminus \mathrm{pa}_i) \mid \mathrm{pa}_i$ holds.
- Hence $M_g(G) \Longrightarrow M_l(G)$ and thus:

$$M_g(G) \iff F(G) \iff M_o(G) \iff M_l(G)$$

Markov blanket

What is the minimal set of variables such that knowing their values makes x independent from the rest?

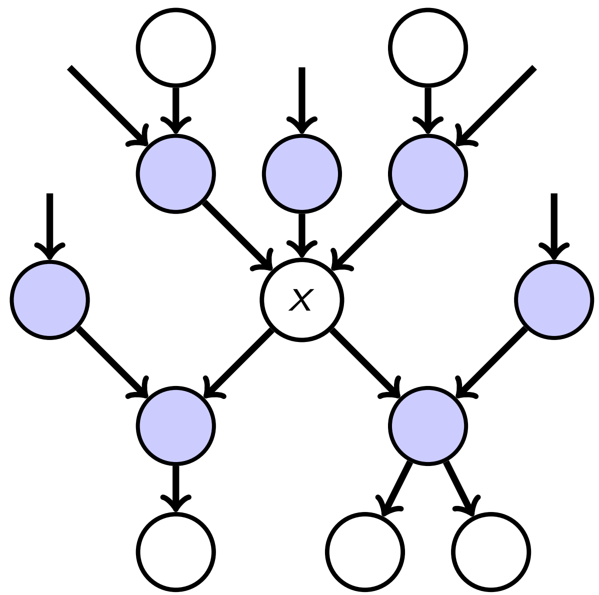
From d-separation:

- Isolate x from its ancestors

  $\Rightarrow$ condition on parents
- Isolate x from its descendants

  $\Rightarrow$ condition on children
- Deal with collider connection

  $\Rightarrow$ condition on co-parents (other parents of the children of x)



In directed graphical models, the parents, children, and co-parents of x are called its Markov blanket, denoted by $\mathrm{MB}(x)$. We have $x \perp\!\!\!\perp \{\text{all vars} \setminus x \setminus \mathrm{MB}(x)\} \mid \mathrm{MB}(x)$.

Program recap

1. Directed ordered Markov property
   - Equivalence between factorisation and directed ordered Markov property
   - Examples
2. D-separation and the directed global Markov property
   - Canonical connections
   - D-separation
   - Recipe and examples
3. Further methods to determine independencies
   - Directed local Markov property
   - Equivalences
   - Markov blanket