ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Advanced AI Techniques

I. Bayesian Networks / 1. Probabilistic Independence and Separation in Graphs

Wolfram Burgard, Luc de Raedt,
Bernhard Nebel, Lars Schmidt-Thieme

Institute of Computer Science
University of Freiburg
http://www.informatik.uni-freiburg.de/

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005

1/50

ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Outline of Part 1 : Learning Bayesian Networks

section	key concepts
I. Probabilistic Independence and Separation in Graphs	Prob. independence, separation in graphs, Bayesian Network
II. Learning Parameters	Max. likelihood estimation, max. a posterior estimation
III. Learning Parameters with Missing Values	EM algo.
IV. Learning Structure by Constrained-based Learning	Markov equivalence, graph patterns, PC algo.
V. Learning Structure by Local Search	Bayesian Information Criterion (BIC), K2 algo., GES algo.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005

1/50

Advanced AI Techniques / 1. Basic Probability Calculus

Joint probability distributions

ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Pain	Y				N			
Weightloss	Y		N		Y		N	
Vomiting	Y	N	Y	N	Y	N	Y	N
Adeno Y	0.220	0.220	0.025	0.025	0.095	0.095	0.010	0.010
N	0.004	0.009	0.005	0.012	0.031	0.076	0.050	0.113

Figure 1: Joint probability distribution $p(P, W, V, A)$ of four random variables P (pain), W (weightloss), V (vomiting) and A (adeno).

Joint probability distributions

ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Discrete JPDs are described by

- nested tables,

- multi-dimensional arrays,

- data cubes, or

- tensors

having entries in $[0,1]$ and summing to 1.

Marginal probability distributions

ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Definition 1. Let p be a the joint probability of the random variables $\mathcal{X} := \{X_1, \ldots, X_n\}$ and $\mathcal{Y} \subseteq \mathcal{X}$ a subset thereof. Then

$$p(\mathcal{Y} = y) := p^{\downarrow \mathcal{Y}}(y) := \sum_{x \in \operatorname{dom} \mathcal{X} \setminus \mathcal{Y}} p(\mathcal{X} \setminus \mathcal{Y} = x, \mathcal{Y} = y)$$

is a probability distribution of $\mathcal{Y}$ called **marginal probability distribution**.

Example 1. Marginal $p(V, A)$:

Vomiting	Y	N
Adeno Y	0.350	0.350
N	0.090	0.210

Pain	Y				N			
Weightloss	Y		N		Y		N	
Vomiting	Y	N	Y	N	Y	N	Y	N
Adeno Y	0.220	0.220	0.025	0.025	0.095	0.095	0.010	0.010
N	0.004	0.009	0.005	0.012	0.031	0.076	0.050	0.113

Figure 2: Joint probability distribution $p(P, W, V, A)$ of four random variables P (pain), W (weightloss), V (vomiting) and A (adeno).

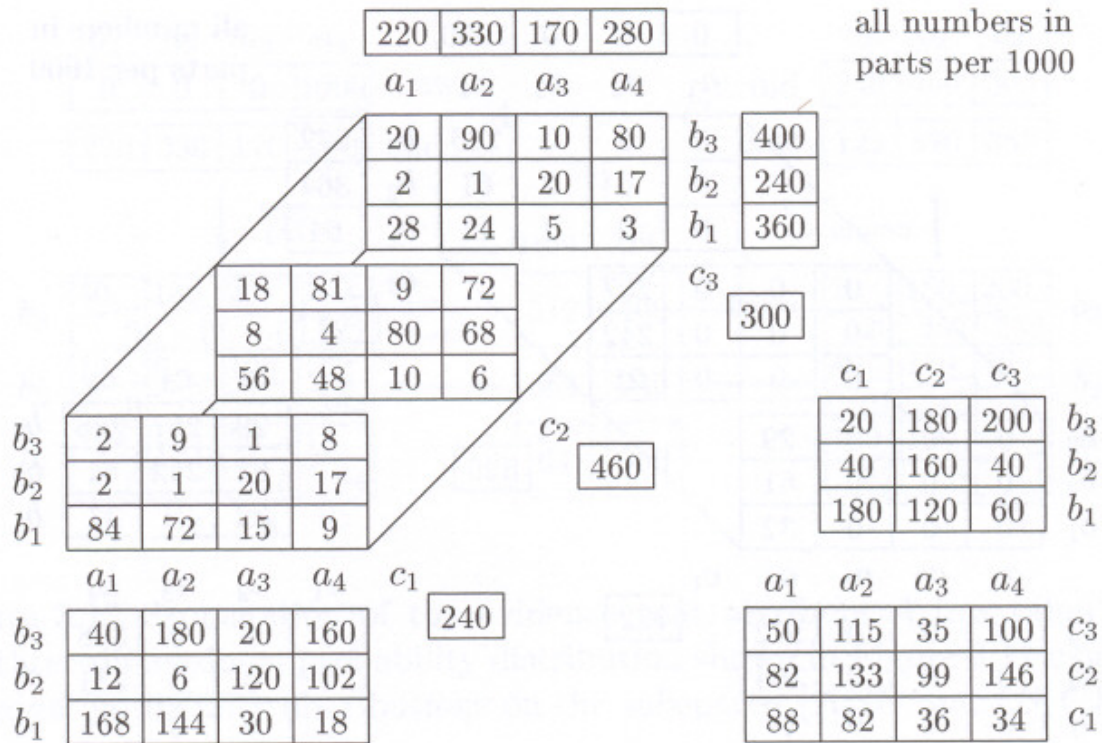## Marginal probability distributions / example

Figure 3: Joint probability distribution and all of its marginals [BK02, p. 75].

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
5/50

Extreme and non-extreme probability distributions

Definition 2. By $p > 0$ we mean

$$p(x) > 0, \quad \text{for all } x \in \prod \text{dom}(p)$$

Then p is called **non-extreme**.

Example 2.

$$\begin{pmatrix} 0.4 & 0.0 \\ 0.3 & 0.3 \end{pmatrix} \qquad \qquad \begin{pmatrix} 0.4 & 0.1 \\ 0.2 & 0.3 \end{pmatrix}$$

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
6/50

ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Definition 3. For a JPD p and a subset $\mathcal{Y} \subseteq \mathrm{dom}(p)$ of its variables with $p^{\downarrow\mathcal{Y}} > 0$ we define

$$p^{|\mathcal{Y}} := \frac{p}{p^{\downarrow\mathcal{Y}}}$$

as **conditional probability distribution of p w.r.t. $\mathcal{Y}$.**

A conditional probability distribution w.r.t. $\mathcal{Y}$ sums to 1 for all fixed values of $\mathcal{Y}$, i.e.,

$$(p^{|\mathcal{Y}})^{\downarrow\mathcal{Y}} \equiv 1$$

ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Example 3. Let p be the JPD

$$p := \begin{pmatrix} 0.4 & 0.1 \\ 0.2 & 0.3 \end{pmatrix}$$

on two variables R (rows) and C (columns) with the domains $\mathrm{dom}(R) = \mathrm{dom}(C) = \{1, 2\}$.

The conditional probability distribution w.r.t. C is

$$p^{|C} := \begin{pmatrix} 2/3 & 1/4 \\ 1/3 & 3/4 \end{pmatrix}$$

Chain rule

Lemma 1 (Chain rule). *Let p be a JPD on variables $X_1, X_2, \ldots, X_n$ with $p(X_1, \ldots, X_{n-1}) > 0$. Then*

$$p(X_1, X_2, \ldots, X_n) = p(X_n | X_1, \ldots, X_{n-1}) \cdots p(X_2 | X_1) \cdot p(X_1)$$

The chain rule provides a **factorization** of the JPD in some of its conditional marginals.

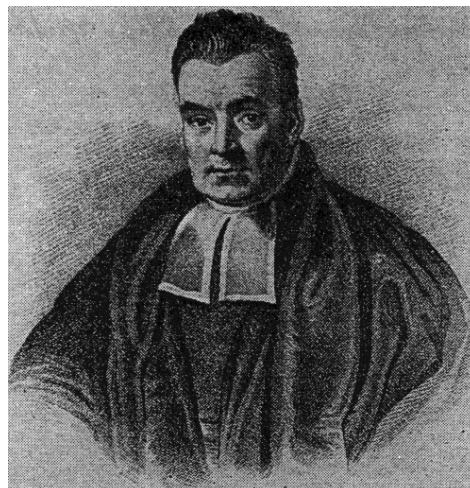The factorizations stemming from the chain rule are trivial as they have as many parameters as the original JPD:

$$\#\text{parameters} = 2^{n-1} + 2^{n-2} + \cdots + 2^1 + 2^0 = 2^n - 1$$

(example computation for all binary variables)

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
9/50

Bayes formula

Lemma 2 (Bayes Formula). *Let p be a JPD and $\mathcal{X}, \mathcal{Y}$ be two disjoint sets of its variables. Let $p(\mathcal{Y}) > 0$. Then*

$$p(\mathcal{X} \mid \mathcal{Y}) = \frac{p(\mathcal{Y} \mid \mathcal{X}) \cdot p(\mathcal{X})}{p(\mathcal{Y})}$$

Thomas Bayes (1701/2–1761)

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
10/50

Independent variables

Definition 4. Two sets $\mathcal{X}, \mathcal{Y}$ of variables are called **independent**, when

$$p(\mathcal{X} = x, \mathcal{Y} = y) = p(\mathcal{X} = x) \cdot p(\mathcal{Y} = y)$$

for all x and y or equivalently

$$p(\mathcal{X} = x | \mathcal{Y} = y) = p(\mathcal{X} = x)$$

for y with $p(\mathcal{Y} = y) > 0$.

Independent variables / example

Example 4. Let Ω be the cards in an ordinary deck and
$R = $ true, if a card is royal,

- $T = $ true, if a card is a ten or a jack,

- $S = $ true, if a card is spade.

Cards for a single color:

2 3 4 5 6 7 8 9 10 J Q K A

ROYALS

S	R	T	$p(R, T \| S)$
Y	Y	Y	1/13
		N	2/13
	N	Y	1/13
		N	9/13
N	Y	Y	3/39 = 1/13
		N	6/39 = 2/13
	N	Y	3/39 = 1/13
		N	27/39 = 9/13

R	T	$p(R, T)$
Y	Y	4/52 = 1/13
	N	8/52 = 2/13
N	Y	4/52 = 1/13
	N	36/52 = 9/13

Conditionally independent variables

Definition 5. Let $\mathcal{X}, \mathcal{Y}$, and $\mathcal{Z}$ be sets of variables.

$\mathcal{X}, \mathcal{Y}$ are called **conditionally independent given** $\mathcal{Z}$, when for all events $\mathcal{Z} = z$ with $p(\mathcal{Z} = z) > 0$ all pairs of events $\mathcal{X} = x$ and $\mathcal{Y} = y$ are conditionally independend given $\mathcal{Z} = z$, i.e.

$$p(\mathcal{X} = x, \mathcal{Y} = y, \mathcal{Z} = z) = \frac{p(\mathcal{X} = x, \mathcal{Z} = z) \cdot p(\mathcal{Y} = y, \mathcal{Z} = z)}{p(\mathcal{Z} = z)}$$

for all x, y and z (with $p(\mathcal{Z} = z) > 0$), or equivalently

$$p(\mathcal{X} = x | \mathcal{Y} = y, \mathcal{Z} = z) = p(\mathcal{X} = x | \mathcal{Z} = z)$$

We write $I_p(\mathcal{X}, \mathcal{Y} | \mathcal{Z})$ for the statement, that $\mathcal{X}$ and $\mathcal{Y}$ are conditionally independent given $\mathcal{Z}$.

Conditionally independent variables

Example 5. Assume S (shape), C (color), and L (label) be three random variables that are distributed as shown in figure 4.

We show $I_p(\{L\}, \{S\} | \{C\})$, i.e., that label and shape are conditionally independent given the color.

C	S	L	$p(L|C,S)$
black	square	1	2/6 = 1/3
		2	4/6 = 2/3
	round	1	1/3
		2	2/3
white	square	1	1/2
		2	1/2
	round	1	1/2
		2	1/2

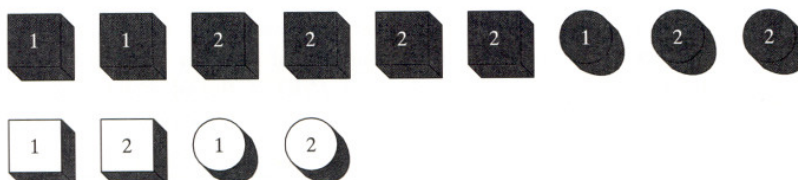| C	L	$p(L|C)$
black	1	3/9 = 1/3
	2	6/9 = 2/3
white	1	2/4 = 1/2
	2	2/4 = 1/2

Figure 4: 13 objects with different shape, color, and label [Nea03, p. 8].

1. Basic Probability Calculus

2. Separation in undirected graphs

3. Separation in directed graphs

4. Markov networks

5. Bayesian networks

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
15/50

Separation in graphs (u-separation)

Definition 6. Let $G := (V, E)$ be a graph. Let $Z \subseteq V$ be a subset of vertices. We say, two vertices $x, y \in V$ are **u-separated by Z in G**, if every path from x to y contains some vertex of Z ($\forall p \in G^* : p_1 = x, p_{|p|} = y \Rightarrow \exists i \in \{1, \ldots, n\} : p_i \in Z$).

Let $X, Y, Z \subseteq V$ be three disjoint subsets of vertices. We say, the vertices X and Y are **u-separated by Z in G**, if every path from any vertex from X to any vertex from Y is separated by Z, i.e., contains some vertex of Z.

We write $I_G(X, Y | Z)$ for the statement, that X and Y are u-separated by Z in G.
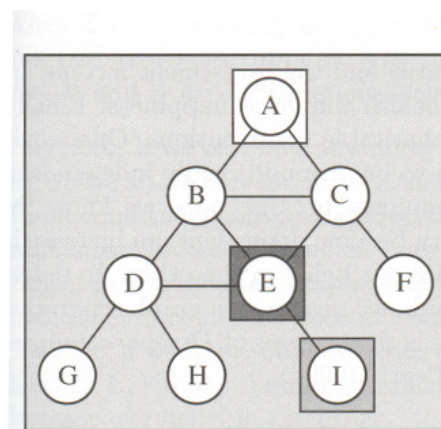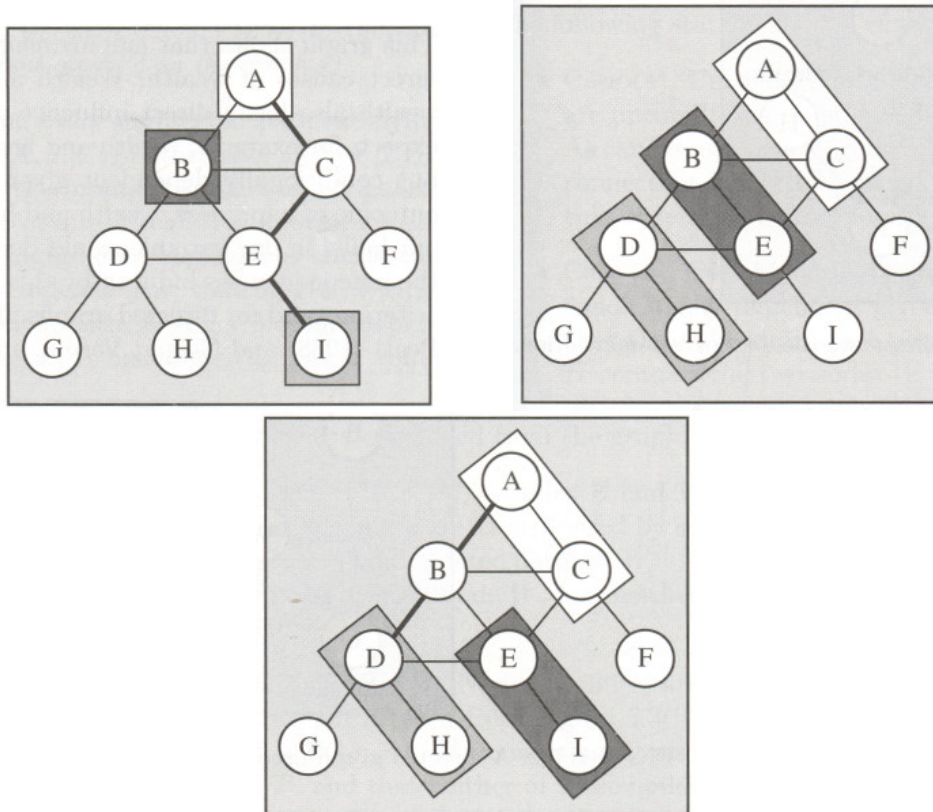
I_G is called **u-separation relation in G**.

Figure 5: Example for u-separation [CGH97, p. 179].

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
15/50

Separation in graphs (u-separation)

Figure 6: More examples for u-separation [CGH97, p. 179].

Checking u-separation

To test, if for a given graph $G = (V, E)$ two given sets $X, Y \subseteq V$ of vertices are u-separated by a third given set $Z \subseteq V$ of vertices, we may use standard breadth-first search to compute all vertices that can be reached from X (see, e.g., [OW02], [CLR90]).

For checking u-separation we have to tweak the algorithm

1. not to add vertices from Z to the border and

2. to stop if a vertex of Y has been reached.

```
1  breadth-first search(G, X) :
2  border := X
3  reached := ∅
4  while border ≠ ∅ do
5      reached := reached ∪ border
6      border := fan_G(border) \ reached
7  od
8  return reached
```

```
1  check-u-separation(G, X, Y, Z) :
2  border := X
3  reached := ∅
4  while border ≠ ∅ do
5      reached := reached ∪ border
6      border := fan_G(border) \ reached \ Z
7      if border ∩ Y ≠ ∅
8          return false
9      fi
10 od
11 return true
```

Figure 7: Breadth-first search algorithm for enumerating all vertices reachable from X.

Figure 8: Breadth-first search algorithm for checking u-separation of X and Y by Z.

1. Basic Probability Calculus

2. Separation in undirected graphs

3. Separation in directed graphs

4. Markov networks

5. Bayesian networks

Chains

Definition 7. Let $G := (V, E)$ be a directed graph. We can construct an **undirected skeleton** $u(G) := (V, u(E))$ **of** G by dropping the directions of the edges:

$$u(E) := \{\{x, y\} \mid (x, y) \in E \text{ or } (y, x) \in E\}$$

The paths on $u(G)$ are called **chains of** G:

$$G^{\blacktriangle} := u(G)^*$$

i.e., a chain is a sequence of vertices that are linked by a forward or a backward edge. If we want to stress the directions of the linking edges, we denote a chain $p = (p_1, \ldots, p_n) \in G^{\blacktriangle}$ by

$$p_1 \leftarrow p_2 \rightarrow p_3 \leftarrow \cdots \leftarrow p_{n-1} \rightarrow p_n$$

The notions of **length**, **subchain**, **interior** and **proper** carry over from undirected paths to chains.
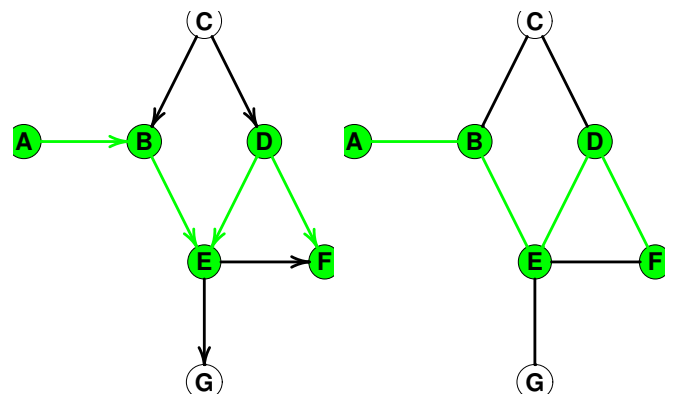

Figure 9: Chain (A, B, E, D, F) on directed graph and path on undirected skeleton.

Blocked chains

Definition 8. Let $G := (V, E)$ be a directed graph. We call a chain

$$p_1 \to p_2 \leftarrow p_3$$

a **head-to-head meeting**.

Let $Z \subseteq V$ be a subset of vertices. Then a chain $p \in G^{\blacktriangle}$ is called **blocked at position** i **by** Z, if for its subchain (p_{i-1}, p_i, p_{i+1}) there is

$$\begin{cases} p_i \in Z, & \text{if not } p_{i-1} \to p_i \leftarrow p_{i+1} \\ p_i \notin Z \cup \text{anc}(Z), & \text{else} \end{cases}$$

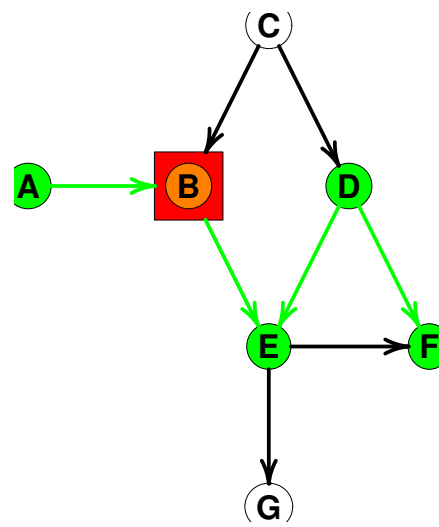Figure 10: Chain (A, B, E, D, F) is blocked by $Z = \{B\}$ at 2.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
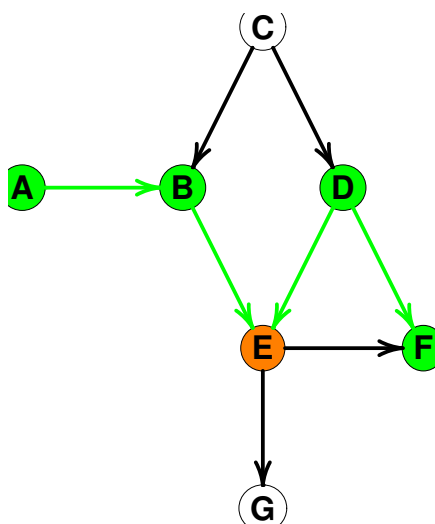19/50

Blocked chains / more examples

Figure 11: Chain (A, B, E, D, F) is blocked by $Z = \emptyset$ at 3.
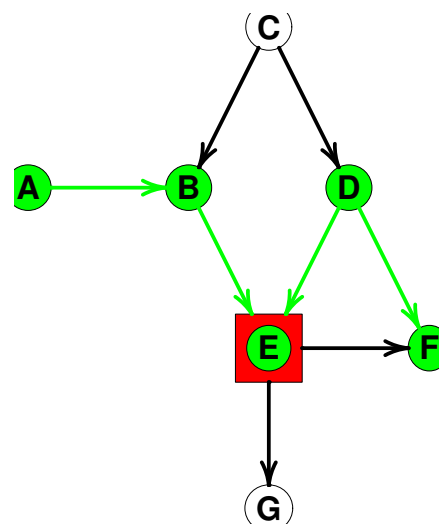
Figure 12: Chain (A, B, E, D, F) is **not** blocked by $Z = \{E\}$ at 3.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
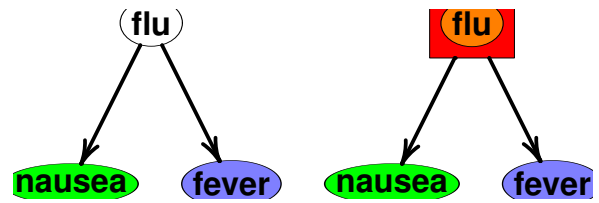Course on Advanced AI Techniques, winter term 2005
20/50

Blocked chains / rationale

The notion of blocking is choosen in a way so that chains model "flow of causal influence" through a causal network where the states of the vertices Z are already know.

1) Serial connection / intermediate cause:
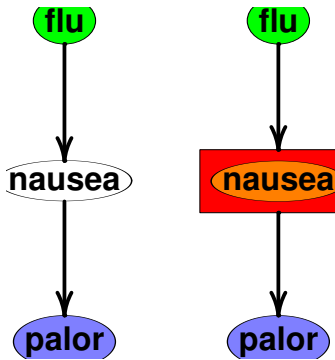
2) Diverging connection / common cause:

3) Converging connection / common effect:
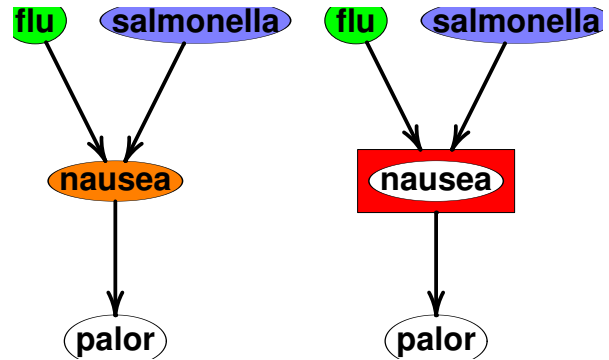
Models "discounting" [Nea03, p. 51].

The moral graph

Definition 9. Let $G := (V, E)$ be a DAG.

As the **moral graph of** G we denote the undirected skeleton graph of G plus additional edges between each two parents of a vertex, i.e. $\mathrm{moral}(G) := (V, E')$ with

$$E' := u(E) \cup \{\{x, y\} \mid \exists z \in V : x, y \in \mathrm{pa}(z)\}$$

Figure 14: DAG and its moral graph.
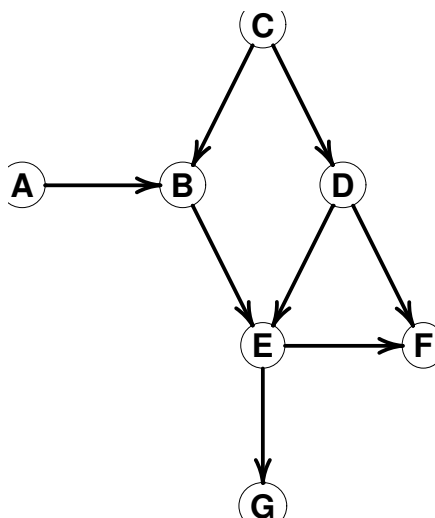
Separation in DAGs (d-separation)

Let $G := (V, E)$ be a DAG.

Let $X, Y, Z \subseteq V$ be three disjoint subsets of vertices. We say, the vertices X and Y are **separated by** Z **in** G, if

(i) every chain from any vertex from X to any vertex from Y is blocked by Z
or equivalently

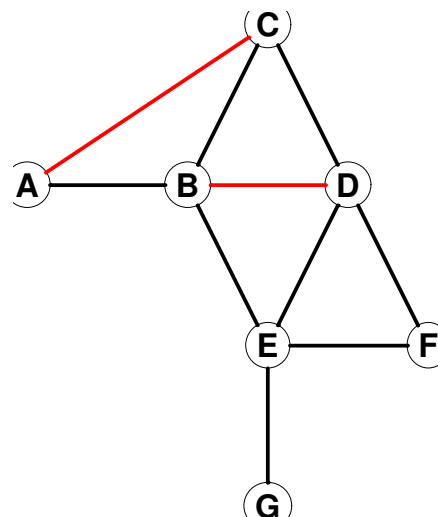(ii) X and Y are u-separated by Z in the moral graph of the ancestral hull of $X \cup Y \cup Z$.

We write $I_G(X, Y | Z)$ for the statement, that X and Y are separated by Z in G.
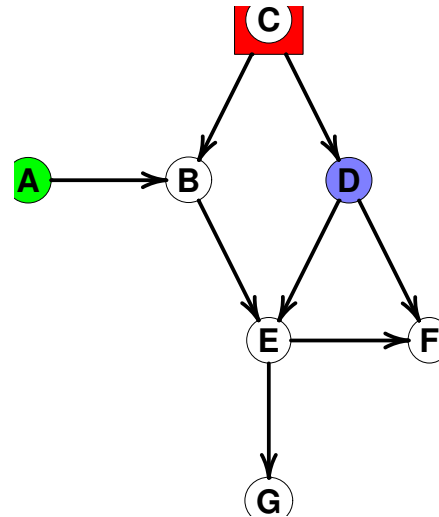
Figure 15: Are the vertices A and D separated by C in G?

Separation in DAGs (d-separation) / examples
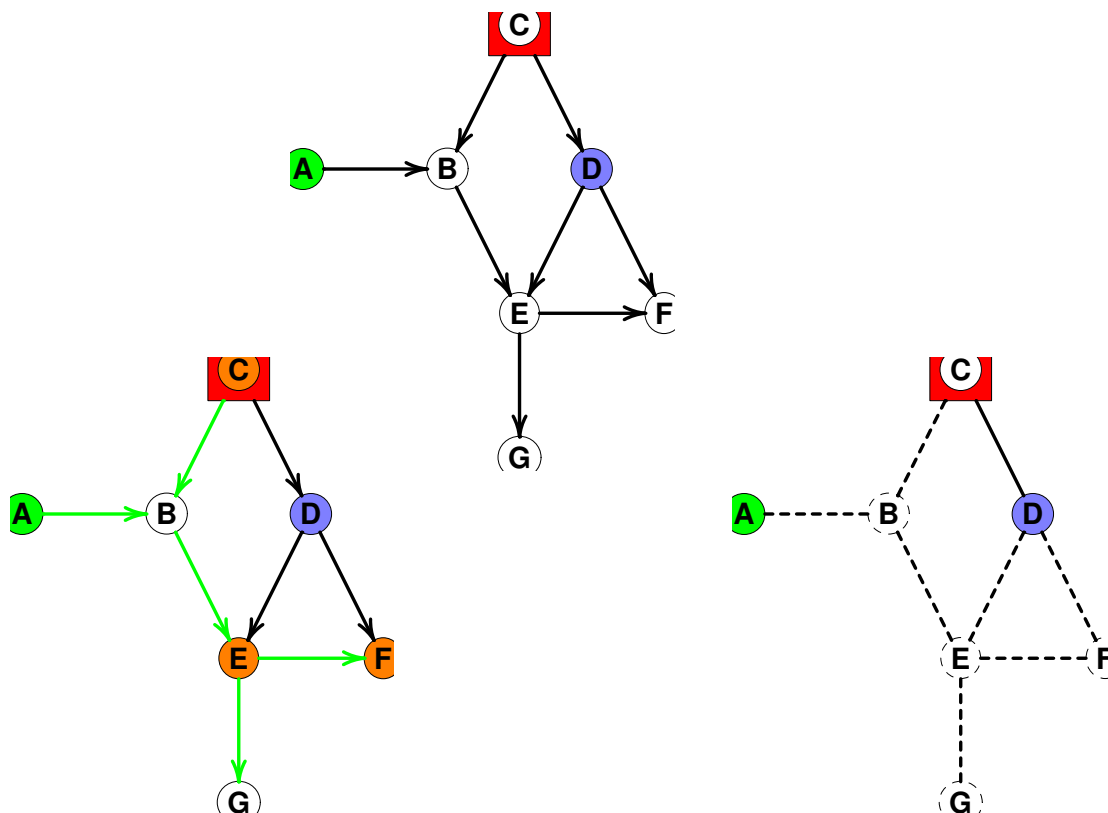
Figure 16: A and D are separated by C in G.

Separation in DAGs (d-separation) / more examples
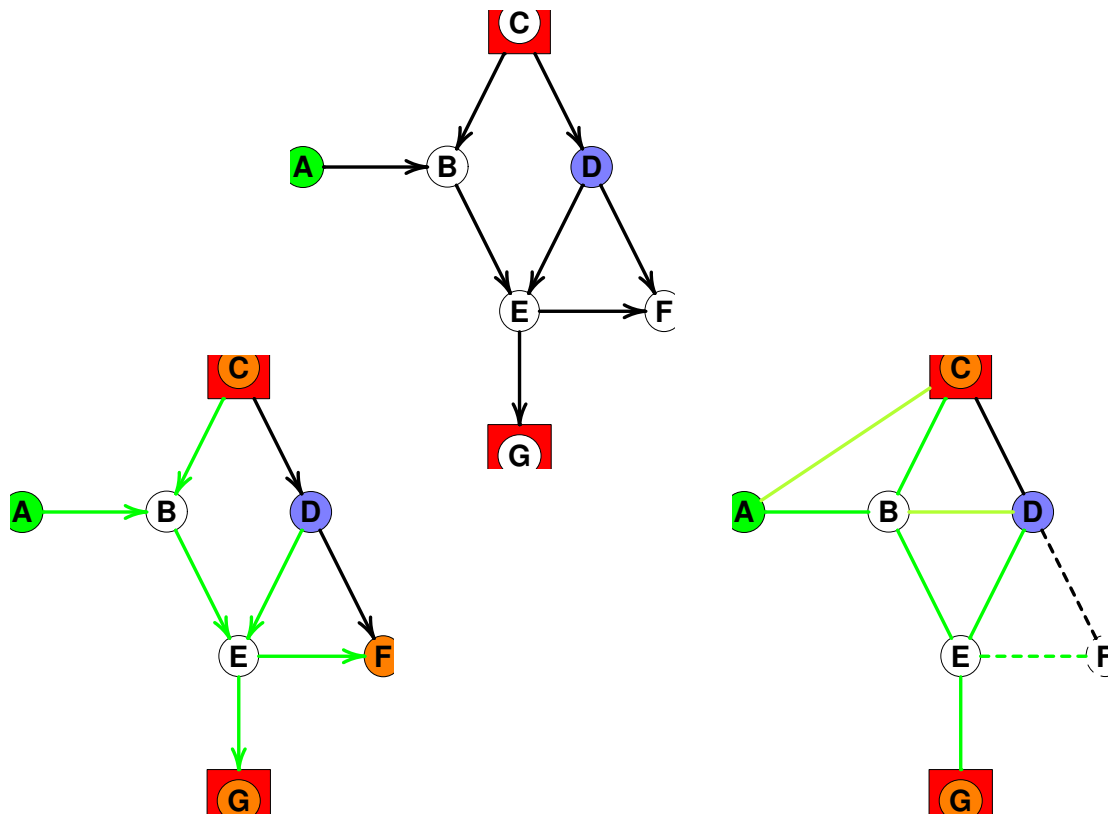
Figure 17: A and D are not separated by $\{C, G\}$ in G.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
25/50

Checking d-separation

To test, if for a given graph $G = (V, E)$ two given sets $X, Y \subseteq V$ of vertices are d-separated by a third given set $Z \subseteq V$ of vertices, we may

- build the moral graph of the ancestral hull and

- apply the u-separation criterion.

> *1* check-d-separation(G, X, Y, Z) :
> *2* $G' := \mathrm{moral}(\mathrm{anc}_G(X \cup Y \cup Z))$
> *3* **return** check-u-separation(G', X, Y, Z)

Figure 18: Algorithm for checking d-separation via u-separation in the moral graph.

A drawback of this algorithm is that we have to rebuild the moral graph of the ancestral hull whenever X or Y changes.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
26/50

Checking d-separation

Instead of constructing a moral graph, we can modify **a breadth-first search for chains** to find all vertices not d-separated from X by Z in G.

The breadth-first search must not hop over head-to-head meetings with the middle vertex not in Z nor having an descendent in Z.
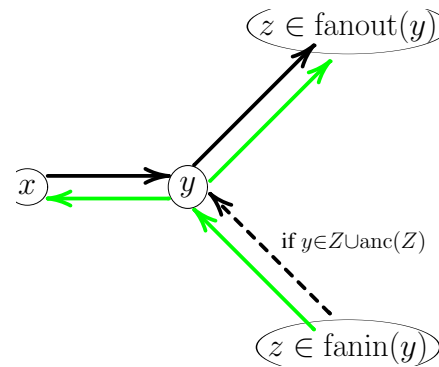
Figure 19: Restricted breadth-first search of non-blocked chains.

```
 1  enumerate-d-separation(G = (V, E), X, Z) :
 2  borderForward := ∅
 3  borderBackward := X \ Z
 4  reached := ∅
 5  while borderForward ≠ ∅ or borderBackward ≠ ∅ do
 6      reached := reached ∪ (borderForward \ Z) ∪ borderBackward
 7      borderForward := fanout_G(borderBackward ∪ (borderForward \ Z)) \ reached
 8      borderBackward := fanin_G(borderBackward ∪ (borderForward ∩ (Z ∪ anc(Z)))) \ Z \ reached
 9  od
10  return V \ reached
```

Figure 20: Algorithm for enumerating all vertices d-separated from X by Z in G via restricted breadth-first search (see [Nea03, p. 80–86] for another formulation).

1. Basic Probability Calculus

2. Separation in undirected graphs

3. Separation in directed graphs

4. Markov networks

5. Bayesian networks

Complete graphs, orderings

Definition 10. An undirected graph $G := (V, E)$ is called **complete**, if it contains all possible edges (i.e. if $E = \mathcal{P}^2(V)$).

Definition 11. Let $G := (V, E)$ be a directed graph.
A bijective map

$$\sigma : \{1, \ldots, |V|\} \to V$$

is called an **ordering of (the vertices of)** G.

We can write an ordering as enumeration of V, i.e. as $v_1, v_2, \ldots, v_n$ with $V = \{v_1, \ldots, v_n\}$ and $v_i \neq v_j$ for $i \neq j$.
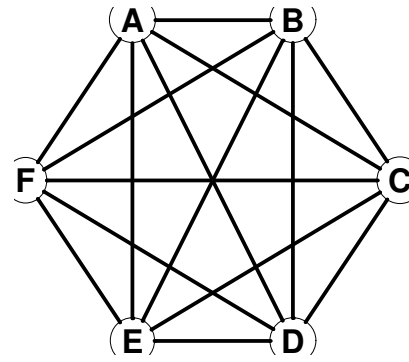
Figure 21: Undirected complete graph with 6 vertices.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
28/50

Topological orderings (1/2)

Definition 12. An ordering $\sigma = (v_1, \ldots, v_n)$ is called **topological ordering** if

(i) all parents of a vertex have smaller numbers, i.e.

$$\mathrm{fanin}(v_i) \subseteq \{v_1, \ldots, v_{i-1}\}, \quad \forall i = 1, \ldots, n$$

or equivalently

(ii) all edges point from smaller to larger numbers

$$(v, w) \in E \Rightarrow \sigma^{-1}(v) < \sigma^{-1}(w), \quad \forall v, w \in V$$

The reverse of a topological ordering – e.g. got by using the fanout instead of the fanin – is called **ancestral numbering**.

In general there are several topological orderings of a DAG.
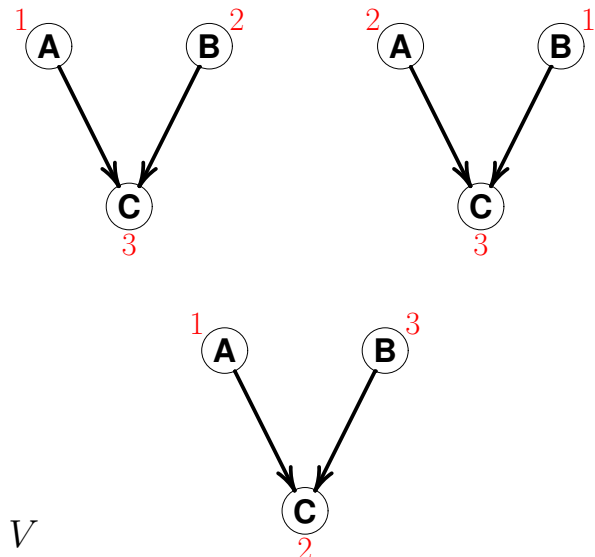
Figure 22: DAG with different topological orderings: $\sigma_1 = (A, B, C)$ and $\sigma_2 = (B, A, C)$. The ordering $\sigma_3 = (A, C, B)$ is not topological.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
29/50

Lemma 3. *Let G be a directed graph. Then*

$$G \text{ is acyclic (a DAG)} \Leftrightarrow G \text{ has a topological ordering}$$

1 topological-ordering$(G = (V, E))$:
2 choose $v \in V$ with $\text{fanout}(v) = \emptyset$
3 $\sigma(|V|) := v$
4 $\sigma|_{\{1,\ldots,|V|-1\}} := \text{topological-ordering}(G \setminus \{v\})$
5 **return** σ

Figure 23: Algorithm to compute a topologcial ordering of a DAG.

Exercise: write an algorithm for checking if a given directed graph is a acyclic.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
30/50

Definition 13. A DAG $G := (V, E)$ is called complete, if

(i) it has a topological ordering $\sigma = (v_1, \ldots, v_n)$ with
$$\text{fanin}(v_i) = \{v_1, \ldots, v_{i-1}\}, \quad \forall i = 1, \ldots, n$$
or equivalently

(ii) it has exactly one topological ordering
or equivalently

(iii) every additional edge introduces a cycle.
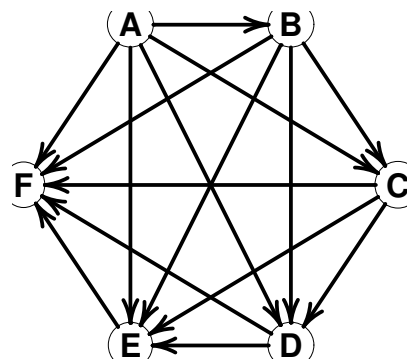
Figure 24: Complete DAG with 6 vertices. Its topological ordering is $\sigma = (A, B, C, D, E, F)$.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
31/50

Graph representations of ternary relations on $\mathcal{P}(V)$

Definition 14. Let V be a set and I a ternary relation on $\mathcal{P}(V)$ (i.e. $I \subseteq \mathcal{P}(V)^3$). In our context I is often called an **independency model**.

Let G be a graph on V (undirected or DAG).
G is called a **representation of** I, if

$$I_G(X, Y|Z) \Rightarrow I(X, Y|Z) \quad \forall X, Y, Z \subseteq V$$

A representation G of I is called **faithful**, if

$$I_G(X, Y|Z) \Leftrightarrow I(X, Y|Z) \quad \forall X, Y, Z \subseteq V$$

Representations are also called **independency maps of** I or **markov w.r.t.** I, faithful representations are also called **perfect maps of** I.
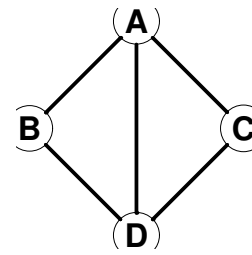
Figure 25: Non-faithful representation of

$$I := \{(A, B|\{C, D\}), (B, C|\{A, D\}),$$
$$(B, A|\{C, D\}), (C, B|\{A, D\})\}$$
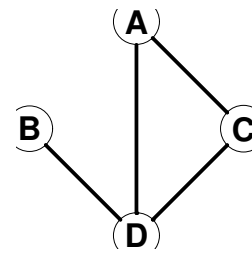
Figure 26: Faithful representation of I. Which I?

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
32/50

Faithful representations

In G also holds
$I_G(B, \{A, C\}|D), I_G(B, A|D), I_G(B, C|D), \ldots$
so G is not a representation of
$$I := \{(A, B|\{C, D\}), (B, C|\{A, D\}),$$
$$(B, A|\{C, D\}), (C, B|\{A, D\})\}$$
at all. It is a representation of
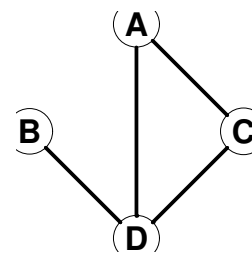
Figure 27: Faithful representation of J.

$$J := \{(A, B|\{C, D\}), (B, C|\{A, D\}), (B, \{A, C\}|D), (B, A|D), (B, C|D),$$
$$(B, A|\{C, D\}), (C, B|\{A, D\}), (\{A, C\}, B|D), (A, B|D), (C, B|D)\}$$

and as all independency statements of J hold in G, it is faithful.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
33/50

ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Trivial representations

For a complete undirected graph or a complete DAG $G := (V, E)$ there is

$$I_G \equiv \text{false},$$

i.e. there are no triples $X, Y, Z \subseteq V$ with $I_G(X, Y | Z)$. Therefore G represents any independency model I on V and is called **trivial representation**.

There are independency models without faithful representation.
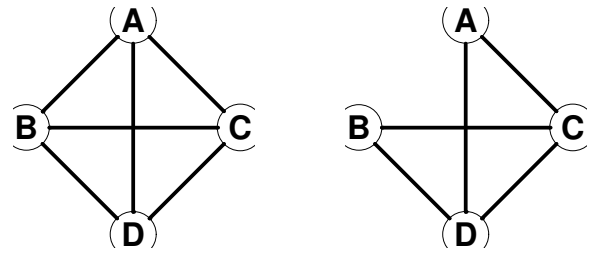
Figure 28: Independency model

$$I := \{(A, B | \{C, D\})\}$$

without faithful representation.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
34/50

ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Minimal representations

Definition 15. A representation G of I is called **minimal**, if none of its subgraphs omitting an edge is a representation of I.
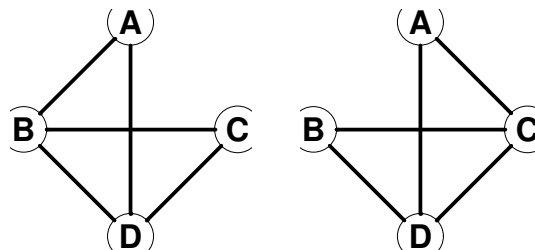
Figure 29: Different minimal undirected representations of the independency model

$$I := \{(A, B | \{C, D\}), (A, C | \{B, D\}),$$
$$(B, A | \{C, D\}), (C, A | \{B, D\})\}$$

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
35/50

Minimal representations

Lemma 4 (uniqueness of minimal undirected representation).
An independency model I has exactly one minimal undirected representation, if and only if it is

> *(i) symmetric: $I(X, Y|Z) \Rightarrow I(Y, X|Z)$.*

> *(ii) decomposable: $I(X, Y|Z) \Rightarrow I(X, Y'|Z)$ for any $Y' \subseteq Y$*

> *(iii) intersectable: $I(X, Y|Y' \cup Z)$ and $I(X, Y'|Y \cup Z) \Rightarrow I(X, Y \cup Y'|Z)$*

Then this representation is $G = (V, E)$ with

$$E := \{\{x, y\} \in \mathcal{P}^2(V) \,|\, not\ I(x, y|V \setminus \{x, y\}\}$$

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
36/50

Minimal representations (2/2)

Example 6.

$$I := \{(A, B|\{C, D\}), (A, C|\{B, D\}), (A, \{B, C\}|D), (A, B|D), (A, C|D),$$
$$(B, A|\{C, D\}), (C, A|\{B, D\}), (\{B, C\}, A|D), (B, A|D), (C, A|D)\}$$

is symmetric, decomposable and intersectable.

Its unique minimal undirected representation is
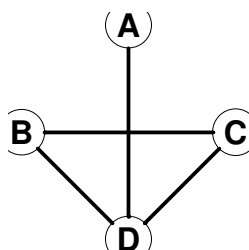
If a faithful representation exists, obviously it is the unique minimal representation, and thus can be constructed by the rule in lemma 4.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
37/50

Representation of conditional independency

Definition 16. We say, a graph **represents a JPD** p, if it represents the conditional independency relation I_p of p.

General JPDs may have several minimal undirected representations (as they may violate the intersection property).

Non-extreme JPDs have a unique minimal undirected representation.

To compute this representation we have to check $I_p(X, Y | V \setminus \{X, Y\})$ for all pairs of variables $X, Y \in V$, i.e.

$$p \cdot p^{\downarrow V \setminus \{X,Y\}} = p^{\downarrow V \setminus \{X\}} \cdot p^{\downarrow V \setminus \{Y\}}$$

Then the minimal representation is the complete graph on V omitting the edges $\{X, Y\}$ for that $I_p(X, Y | V \setminus \{X, Y\})$ holds.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
38/50

Representation of conditional independency

Example 7. Let p be the JPD on $V := \{X, Y, Z\}$ given by:

Z	X	Y	$p(X, Y, Z)$
0	0	0	0.024
0	0	1	0.056
0	1	0	0.036
0	1	1	0.084
1	0	0	0.096
1	0	1	0.144
1	1	0	0.224
1	1	1	0.336

Checking $p \cdot p^{\downarrow V \setminus \{X,Y\}} = p^{\downarrow V \setminus \{X\}} \cdot p^{\downarrow V \setminus \{Y\}}$ one finds that the only independency relations of p are $I_p(X, Y | Z)$ and $I_p(Y, X | Z)$.

Its marginals are:

Z	X	$p(X, Z)$
0	0	0.08
0	1	0.12
1	0	0.24
1	1	0.56

Z	Y	$p(Y, Z)$
0	0	0.06
0	1	0.14
1	0	0.32
1	1	0.48

X	Y	$p(X, Y)$
0	0	0.12
0	1	0.2
1	0	0.26
1	1	0.42

X	$p(X)$
0	0.32
1	0.68

Y	$p(Y)$
0	0.38
1	0.62

Z	$p(Z)$
0	0.2
1	0.8

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
38/50

Representation of conditional independency

Example 7 (cont.).

Z	X	Y	$p(X,Y,Z)$
0	0	0	0.024
0	0	1	0.056
0	1	0	0.036
0	1	1	0.084
1	0	0	0.096
1	0	1	0.144
1	1	0	0.224
1	1	1	0.336

Checking $p \cdot p^{\downarrow V \setminus \{X,Y\}} = p^{\downarrow V \setminus \{X\}} \cdot p^{\downarrow V \setminus \{Y\}}$ one finds that the only independency relations of p are $I_p(X,Y|Z)$ and $I_p(Y,X|Z)$.

Thus, the graph
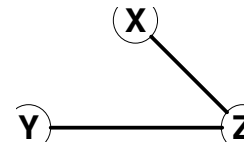
represents p, as its independency model is $I_G := \{(X,Y|Z), (Y,X|Z)\}$.

As for p only $I_p(X,Y|Z)$ and $I_p(Y,X|Z)$ hold, G is a faithful representation.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
39/50

Markov networks

Definition 17. A pair $(G, (\psi_C)_{C \in \mathcal{C}_G})$ consisting of

(i) an undirected graph G on a set of variables V and

(ii) a set of potentials

$$\psi_C : \prod_{X \in C} \mathrm{dom}(X) \to \mathbb{R}_0^+, \quad C \in \mathcal{C}_G$$

on the cliques[1] of G (called **clique potentials**)

is called a **markov network**.

[1] on the product of the domains of the variables of each clique.

Thus, a markov network encodes

(i) a joint probability distribution factorized as

$$p = (\prod_{C \in \mathcal{C}_G} \psi_C)^{|\emptyset}$$

and

(ii) conditional independency statements

$$I_G(X,Y|Z) \Rightarrow I_p(X,Y|Z)$$

G represents p, but not necessarily faithfully.

Under some regularity conditions (not covered here), ψ_{C_i} can be choosen as conditional probabilities $p^{\downarrow C_i | S_i}$.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
40/50

Markov networks / examples

Z	X	$p(X, Z)$
0	0	0.08
0	1	0.12
1	0	0.24
1	1	0.56

Z	Y	$p(Y|Z)$
0	0	0.3
0	1	0.7
1	0	0.4
1	1	0.6
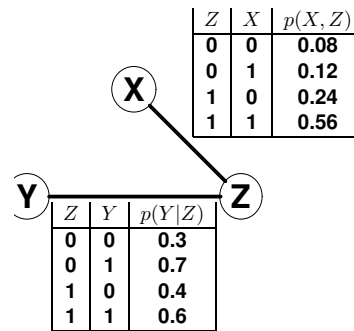
Figure 30: Example for a markov network.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
41/50

1. Basic Probability Calculus

2. Separation in undirected graphs

3. Separation in directed graphs

4. Markov networks

5. Bayesian networks

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
42/50

DAG-representations

Lemma 5 (criterion for DAG-representation). *Let p be a joint probability distribution of the variables V and G be a graph on the vertices V. Then:*

G represents $p \Leftrightarrow v$ and $\mathrm{nondesc}(v)$ are conditionally independent given $\mathrm{pa}(v)$ for all $v \in V$, i.e.,

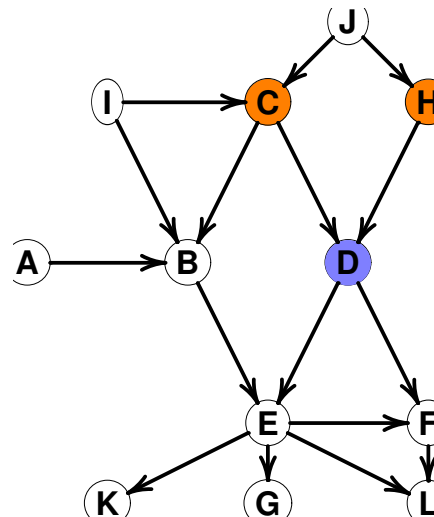$$I_p(\{v\}, \mathrm{nondesc}(v)|\,\mathrm{pa}(v)), \quad \forall v \in V$$

Figure 31: Parents of a vertex (orange).

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
42/50

Example for a not faithfully DAG-representable independency model

Probability distributions may have no faithful DAG-representation.

Example 8. The independency model

$$I := \{I(x, y|z), I(y, x|z), I(x, y|w), I(y, x|w)\}$$

does not have a faithful DAG-representation. [CGH97, p. 239]

Exercise: compute all minimal DAG-representations of I using lemma 6 and check if they are faithful.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
43/50

Minimal DAG-representations

Lemma 6 (construction and uniqueness of minimal DAG-representation, [VP90])

Let I be an independence model of a JPD p. Then:

(i) A minimal DAG-representation can be constructed as follows: Choose an arbitrary ordering $\sigma := (v_1, \ldots, v_n)$ of V. Choose a minimal set $\pi_i \subseteq \{v_1, \ldots, v_{i-1}\}$ of σ-precursors of v_i with

$$I(v_i, \{v_1, \ldots, v_{i-1}\} \setminus \pi_i | \pi_i)$$

Then $G := (V, E)$ with

$$E := \{(w, v_i) \mid i = 1, \ldots, n, w \in \pi_i\}$$

is a minimal DAG-representation of p.

(ii) If p also is non-extreme, then the minimal representation G is unique up to ordering σ.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
44/50

Minimal DAG-representations / example

$$I := \{(A, C|B), (C, A|B)\}$$
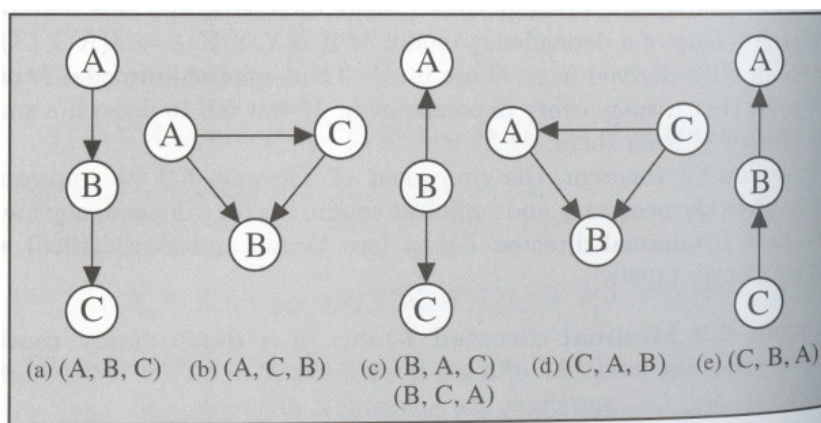
Figure 32: Minimal DAG-representations of I [CGH97, p. 240].

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005
45/50

ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Minimal representations / conclusion

Representations always exist (e.g., trivial).

Minimal representations always exist
(e.g., start with trivial and drop edges successively).

	Markov network (undirected)		Bayesian network (directed)	
	minimal	faithful	minimal	faithful
general JPD	may not be unique	may not exist	may not be unique	may not exist
non-extreme JPD	unique	may not exist	unique up to ordering	may not exist

ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Bayesian Network

Definition 18. A pair $(G := (V, E), (p_v)_{v \in V})$ consisting of

(i) a directed graph G on a set of variables V and

(ii) a set of conditional probability distributions

$$p_X : \mathrm{dom}(X) \times \prod_{Y \in \mathrm{pa}(X)} \mathrm{dom}(Y) \to \mathbb{R}_0^+$$

at the vertices $X \in V$ conditioned on its parents (called **(conditional) vertex probability distributions**)

is called a **bayesian network**.

Thus, a bayesian network encodes

(i) a joint probability distribution factorized as

$$p = \prod_{X \in V} p(X \mid \mathrm{pa}(X))$$

and

(ii) conditional independency statements

$$I_G(X, Y | Z) \Rightarrow I_p(X, Y | Z)$$

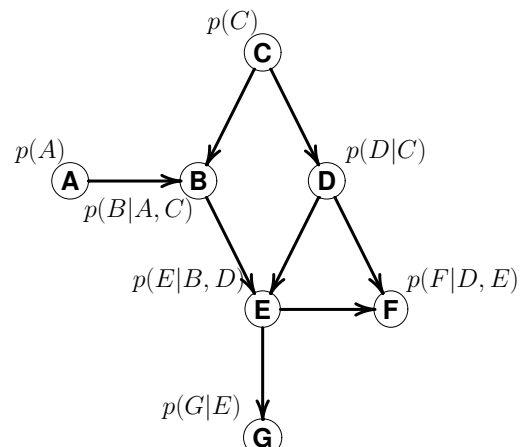G represents p, but not necessarily faithfully.

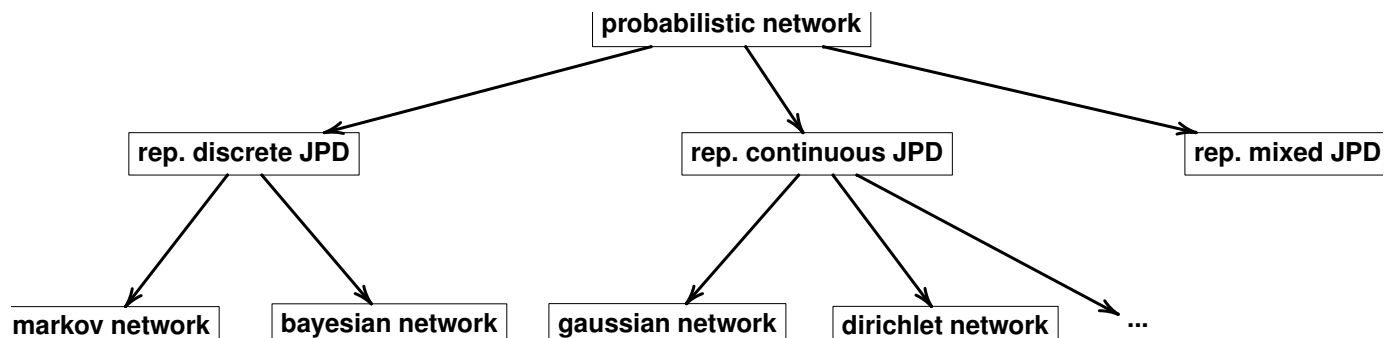Figure 33: Example for a bayesian network.

Types of probabilistic networks

```
                        probabilistic network
              ┌──────────────────┼──────────────────────┐
              ▼                   ▼                      ▼
      rep. discrete JPD    rep. continuous JPD      rep. mixed JPD
         ┌────┴────┐        ┌────────┼────────┐
         ▼         ▼        ▼        ▼         ▼
   markov network  bayesian network  gaussian network  dirichlet network  ...
```

Figure 34: Types of probabilistic networks.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005

48/50

References

[BK02]   Christian Borgelt and Rudolf Kruse. *Graphical Models*. Wiley, New York, 2002.

[CGH97] Enrique Castillo, José Manuel Gutiérrez, and Ali S. Hadi. *Expert Systems and Probabilistic Network Models*. Springer, New York, 1997.

[CLR90] Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. *Introduction to Algorithms*. MIT Press, Cambridge, Massachusetts, 1990.

[Nea03] Richard E. Neapolitan. *Learning Bayesian Networks*. Prentice Hall, 2003.

[OW02]   Thomas Ottmann and Peter Widmayer. *Algorithmen und Datenstrukturen*. Spektrum Verlag, Heidelberg, 2002.

[VP90]   Thomas Verma and Judea Pearl. Causal networks: semantics and expressiveness. In Ross D. Shachter, Tod S. Levitt, Laveen N. Kanal, and John F. Lemmer, editors, *Uncertainty in Artificial Intelligence 4*, pages 69–76. North-Holland, Amsterdam, 1990.

Wolfram Burgard, Luc de Raedt, Bernhard Nebel, Lars Schmidt-Thieme, Institute of Computer Science, University of Freiburg, Germany,
Course on Advanced AI Techniques, winter term 2005

49/50