Bayesian networks: Representation

Vibhav Gogate
(Some slides borrowed from Adnan Darwiche)

THE UNIVERSITY OF TEXAS AT DALLAS
Erik Jonsson School of Engineering and Computer Science

Motivation

- Explicit representation of the joint distribution is unmanageable
  - Computationally: Memory intensive to store and manipulate
  - Cognitively: Impossible to acquire so many numbers from human experts
  - Statistically: We will need ridiculously large amount of data to learn.
- Solution: Exploit Independence properties and Represent the distribution using a graph
- Trouble: Mapping the logic of probability theory into graph theory!

Properties of Independence

The statement $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$ means that $\mathbf{X}$ is independent of $\mathbf{Y}$ given $\mathbf{Z}$.
Namely, $\Pr(\mathbf{X}|\mathbf{Y}, \mathbf{Z}) = \Pr(\mathbf{X}|\mathbf{Z})$ and $\Pr(\mathbf{X}, \mathbf{Y}|\mathbf{Z}) = \Pr(\mathbf{X}|\mathbf{Z}) \Pr(\mathbf{Y}|\mathbf{Z})$

- **Symmetry** $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y}) \Rightarrow I(\mathbf{Y}, \mathbf{Z}, \mathbf{X})$
- **Decomposition** $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y} \cup \mathbf{W}) \Rightarrow I(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$
- **Weak Union** $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y} \cup \mathbf{W}) \Rightarrow I(\mathbf{X}, \mathbf{Z} \cup \mathbf{W}, \mathbf{Y})$
- **Contraction** $I(\mathbf{X}, \mathbf{Z} \cup \mathbf{Y}, \mathbf{W}) \& I(\mathbf{X}, \mathbf{Z}, \mathbf{Y}) \Rightarrow I(\mathbf{X}, \mathbf{Z}, \mathbf{Y} \cup \mathbf{W})$
- **Intersection** For any positive distribution:
  $I(\mathbf{X}, \mathbf{Z} \cup \mathbf{W}, \mathbf{Y}) \& I(\mathbf{X}, \mathbf{Z} \cup \mathbf{Y}, \mathbf{W}) \Rightarrow I(\mathbf{X}, \mathbf{Z}, \mathbf{Y} \cup \mathbf{W})$

Proof of Symmetry

- Assume that $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$ holds. This implies that:

$$\Pr(\mathbf{X}, \mathbf{Y}|\mathbf{Z}) = \Pr(\mathbf{X}|\mathbf{Z}) \times \Pr(\mathbf{Y}|\mathbf{Z}) \tag{1}$$

  i.e. $I(\mathbf{Y}, \mathbf{Z}, \mathbf{X})$ holds too (exchanging the positions of $\mathbf{X}$ and $\mathbf{Y}$).

Proof of Decomposition

- Assume that $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y} \cup \mathbf{W})$ holds. Then,

$$\Pr(\mathbf{X}, \mathbf{Y}, \mathbf{W} | \mathbf{Z}) = \Pr(\mathbf{X} | \mathbf{Z}) \times \Pr(\mathbf{Y}, \mathbf{W} | \mathbf{Z})$$

$$\begin{aligned}
\Pr(\mathbf{X}, \mathbf{Y} | \mathbf{Z}) &= \sum_{\mathbf{w}} \Pr(\mathbf{X}, \mathbf{Y}, \mathbf{w} | \mathbf{Z}) && (2) \\
&= \sum_{\mathbf{w}} \Pr(\mathbf{X} | \mathbf{Z}) \times \Pr(\mathbf{Y}, \mathbf{w} | \mathbf{Z}) && (3) \\
&= \Pr(\mathbf{X} | \mathbf{Z}) \sum_{\mathbf{w}} \Pr(\mathbf{Y}, \mathbf{w} | \mathbf{Z}) && (4) \\
&= \Pr(\mathbf{X} | \mathbf{Z}) \Pr(\mathbf{Y} | \mathbf{Z}) && (5)
\end{aligned}$$

i.e. $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$ holds too.

Proof of Weak Union

- Assume that $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y} \cup \mathbf{W})$ holds. Then,

$$\Pr(\mathbf{X}, \mathbf{Y}, \mathbf{W}|\mathbf{Z}) = \Pr(\mathbf{X}|\mathbf{Z}) \times \Pr(\mathbf{Y}, \mathbf{W}|\mathbf{Z})$$

$$\begin{aligned}
\Pr(\mathbf{X}, \mathbf{Y}|\mathbf{W}, \mathbf{Z}) &= \frac{\Pr(\mathbf{X}, \mathbf{Y}, \mathbf{W}|\mathbf{Z})}{\Pr(\mathbf{W}|\mathbf{Z})} && (6)\\
&= \frac{\Pr(\mathbf{X}|\mathbf{Z}) \times \Pr(\mathbf{Y}, \mathbf{W}|\mathbf{Z})}{\Pr(\mathbf{W}|\mathbf{Z})} && (7)\\
&= \Pr(\mathbf{X}|\mathbf{Z}) \times \Pr(\mathbf{Y}|\mathbf{W}, \mathbf{Z}) && (8)
\end{aligned}$$

I stopped here: Homework problem

Which of the previous properties can you use to prove that:

$$\Pr(\mathbf{X}, \mathbf{Y}|\mathbf{W}, \mathbf{Z}) = \Pr(\mathbf{X}|\mathbf{W}, \mathbf{Z}) \times \Pr(\mathbf{Y}|\mathbf{W}, \mathbf{Z})$$

Bayesian networks: Directed-Graphs

- Mapping a distribution to a Graph!!

The graph can be viewed in two different ways:

- As a data structure to represent the joint distribution compactly
- As a compact representation of a set of conditional independence assumptions about a distribution

The two views are equivalent

Bayesian networks: Data Structure view

- Bayesian network $=$ Use Chain rule $+$ Conditional Independence properties
- Chain rule:

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | X_1, \ldots, X_{i-1})$$

Bayesian networks: Data Structure view

- Chain rule: $P(X_1, \dots, X_n) = \prod_{i=1}^{n} P(X_i | X_1, \dots, X_{i-1})$

Use the chain rule to represent the joint distribution rather than a giant table!

Example

Intelligence "I" and SAT Score "S"

$$P(I, S) = P(I)P(S|I)$$

	s^0	s^1
i^0	0.95*0.7	0.05 *0.7
i^1	0.2*0.3	0.8*0.3

$=$

i^0	i^1
0.7	0.3

$\times$

	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

Bayesian networks: Data Structure view

- However, we don't gain anything by using the chain rule. Space complexity is the same.
- Exploit conditional independence properties
- What if I tell you that you are representing a joint distribution over 2 coin tosses?

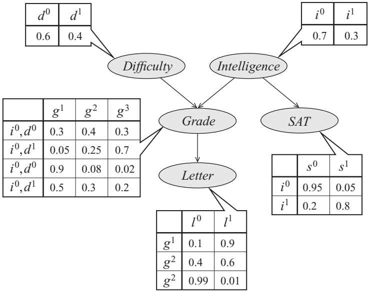
$$Chain\ rule : P(X_1, X_2) = P(X_1)P(X_2|X_1)$$

$$Conditional\ Independence : P(X_1, X_2) = P(X_1)P(X_2)$$

Bayesian networks: Data Structure view

- Chain rule $P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | X_1, \ldots, X_{i-1})$ as a directed graph.
- $X_1, \ldots, X_{i-1}$ are the parents of X_i. Complete Graph
- If we know that $P(X_i | X_1, \ldots, X_{i-1}) = P(X_i | \mathbf{Y}_i)$ where $\mathbf{Y_i}$ is a subset of $\{X_1, \ldots, X_{i-1}\}$. Then, we get a sparse graph (a Bayesian network).

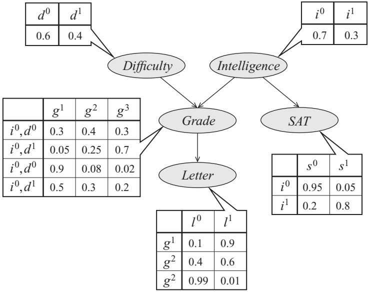
Bayesian networks: Data structure view



- Random variables are nodes and edges represent direct influence of one variable on other
- Each node is associated with a conditional probability table (CPT).

- The joint distribution is product of all CPTs
  $P(D, I, G, S, L) = P(D)P(I)P(G|D, I)P(L|G)P(S|I)$
- What is $P(i^1, d^0, g^2, s^1, l^0)$?

Bayesian networks: Data structure view



d^0	d^1
0.6	0.4

i^0	i^1
0.7	0.3

	g^1	g^2	g^3
i^0,d^0	0.3	0.4	0.3
i^0,d^1	0.05	0.25	0.7
i^0,d^0	0.9	0.08	0.02
i^0,d^1	0.5	0.3	0.2

	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

	l^0	l^1
g^1	0.1	0.9
g^2	0.4	0.6
g^2	0.99	0.01

Space complexity?

- Assume each variable has d values in its domain.
- $O(d^{k+1})$ for each variable having k parents.

Graph terms

$\text{Parents}(V)$

variables N with an edge from N to V

$\text{Descendants}(V)$

variables N with a directed path from V to N.
V is said to be an ancestor of N

$\text{Non\_Descendants}(V)$

variables other than V, $\text{Parents}(V)$ and $\text{Descendants}(V)$

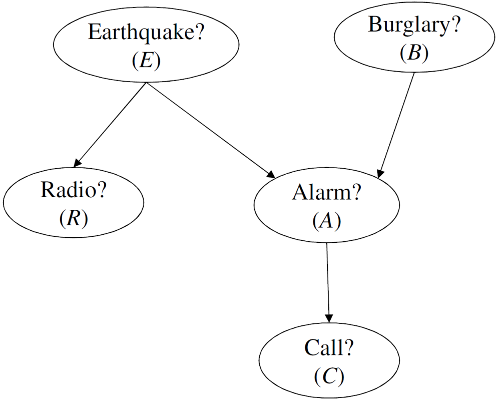
Bayesian networks: Compact Representation of Conditional Independence statements view

- A directed acyclic graph G represents the following independence statements

$$Markov(G) \;=\; I(V, Parents(V), Non - Descendants(V))$$

- Parents(V) denote the direct causes of V and Descendants(V) denote the effects of V
- Given the direct causes of a variable, our beliefs in that variable become independent of its non-effects.

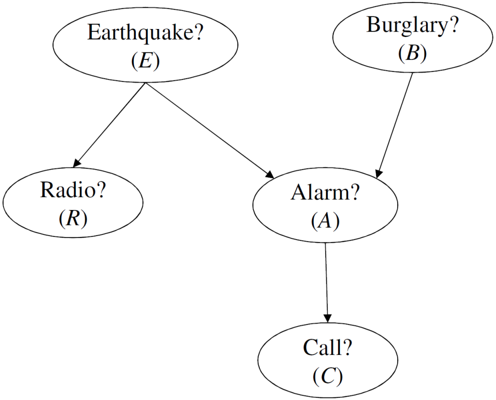
Bayesian networks: Compact Representation of Conditional Independence statements view



Markovian assumptions?

- 1.
- 2.
- 3.
- 4.
- 5.

Bayesian networks: Compact Representation of Conditional Independence statements view



Markovian assumptions, Markov(G):

$$I(C, A, \{B, E, R\})$$
$$I(R, E, \{A, B, C\})$$
$$I(A, \{B, E\}, R)$$
$$I(B, \emptyset, \{E, R\})$$
$$I(E, \emptyset, B)$$

Expanding Markov (G) using properties of probabilistic independence

- $Markov(G)$ is not comprehensive. We can expand it using properties such as symmetry, decomposition, weak-union, contraction and intersection.

- Given $\mathbf{W} \subseteq Non - Descendants(X))$, how can we strengthen $Markov(G) = I(X, Parents(X), Non - Descendants(X))$
- Decomposition: $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y} \cup \mathbf{W}) \Rightarrow I(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$

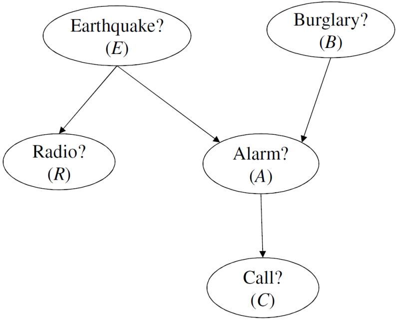
$$I(X, Parents(X), \mathbf{W})$$

- Weak Union: $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y} \cup \mathbf{W}) \Rightarrow I(\mathbf{X}, \mathbf{Z} \cup \mathbf{W}, \mathbf{Y})$

$$I(X, Parents(X) \cup \mathbf{W}, Non - Decendants(X) \setminus \mathbf{W})$$

- and so on.

Expanding Markov(G) using Symmetry



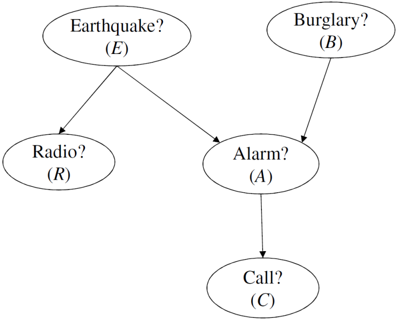
$I_{\Pr}(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$ iff $I_{\Pr}(\mathbf{Y}, \mathbf{Z}, \mathbf{X})$

Learning **y** does not influence our belief in **x** iff learning **x** does not influence our belief in **y**

Example

From $\mathrm{Markov}(G)$, we have $I_{\Pr}(A, \{B, E\}, R)$. Using Symmetry, we get $I_{\Pr}(R, \{B, E\}, A)$ which is not part of $\mathrm{Markov}(G)$

Expanding Markov(G) using Weak-union



Markov(G) gives

$I(C, A, \{B, E, R\})$

By Weak Union

$I(C, \{A, B, E\}, R)$ which is not part of Markov(G)

Graphoid Axioms

Triviality: $I_{\Pr}(\mathbf{X}, \mathbf{Z}, \emptyset)$

Symmetry, Decomposition, Weak Union, and Contraction, combined with Triviality, are known as the graphoid axioms.

With Intersection, the set is known as the positive graphoid axioms.

Decomposition, Weak Union, and Contraction in one statement

$I_{\Pr}(\mathbf{X}, \mathbf{Z}, \mathbf{Y} \cup \mathbf{W})$ iff $I_{\Pr}(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$ and $I_{\Pr}(\mathbf{X}, \mathbf{Z} \cup \mathbf{Y}, \mathbf{W})$

The terms semi-graphoid and graphoid are sometimes used instead of graphoid and positive graphoid, respectively.

Capturing independence graphically

- Question: Is there a purely graphical test that can find all of these independence statements, namely $Markov(G)$ plus the ones inferred using the properties of conditional independence?
  - YES, it is called d-separation.

D-separation

$\mathbf{X}$ and $\mathbf{Y}$ are d-separated by $\mathbf{Z}$, written $\mathrm{dsep}_G(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$

iff every path between a node in $\mathbf{X}$ and a node in $\mathbf{Y}$ is blocked by $\mathbf{Z}$

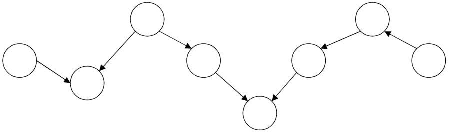
The definition of d-separation relies on

the notion of blocking a path by a set of variables $\mathbf{Z}$

$\mathrm{dsep}_G(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$ implies $I_{\mathrm{Pr}}(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$

for every probability distribution Pr induced by G

D-separation



View the path as a pipe

and view each variable W on the path as a valve.

A valve W is either open or closed

depending on some conditions that we state later.

If at least one of the valves on the path is closed

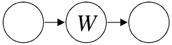
the whole path is blocked. Otherwise, the path is not blocked.

D-separation

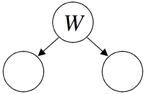
The type of a valve

is determined by its relationship to its neighbors on the path.

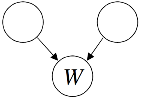
sequential $\rightarrow W \rightarrow$



divergent $\leftarrow W \rightarrow$

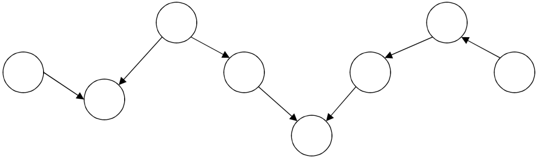


convergent $\rightarrow W \leftarrow$



D-separation

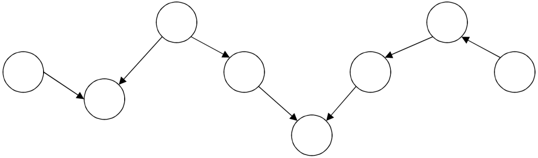
A path with 6 valves



From left to right

D-separation

A path with 6 valves



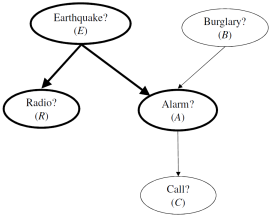
From left to right

convergent, divergent, sequential, convergent, sequential, and sequential.

D-separation

Given that we know $\mathbf{Z}$

when is a divergent valve closed?



Valve $R \leftarrow E \rightarrow A$ is closed iff

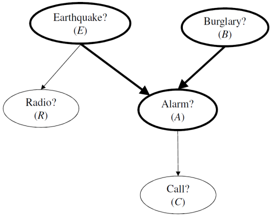
we know the value of variable E, otherwise a radio report on an earthquake may change our belief in the alarm triggering.

A divergent valve $\leftarrow W \rightarrow$ is closed iff variable W appears in $\mathbf{Z}$

D-separation

Given that we know $\mathbf{Z}$

when is a convergent valve closed?



Valve $E \rightarrow A \leftarrow B$ is closed iff

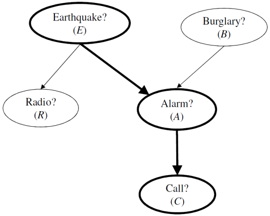
neither the value of variable A nor the value of C are known, otherwise, a burglary may change our belief in an earthquake.

A convergent valve $\rightarrow W \leftarrow$ is closed iff neither variable W nor any of its descendants appears in $\mathbf{Z}$

D-separation

Given that we know $\mathbf{Z}$

when is a sequential valve closed?



Valve $E \rightarrow A \rightarrow C$ is closed iff

we know the value of variable A, otherwise an earthquake E may change our belief in getting a call C.

A sequential valve $\rightarrow W \rightarrow$ is closed iff variable W appears in $\mathbf{Z}$

D-separation

$\mathbf{X}$ and $\mathbf{Y}$ are d-separated by $\mathbf{Z}$, written $\mathrm{dsep}_G(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$, iff

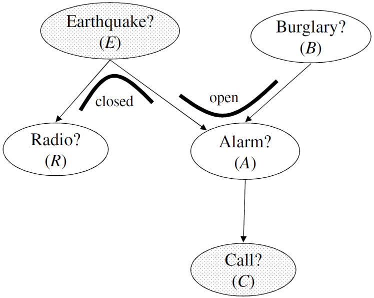
every path between a node in $\mathbf{X}$ and a node in $\mathbf{Y}$ is blocked by $\mathbf{Z}$

A path is blocked by $\mathbf{Z}$ iff

at least one valve on the path is closed given $\mathbf{Z}$

A path with no valves (i.e., $X \longrightarrow Y$) is never blocked.

D-separation

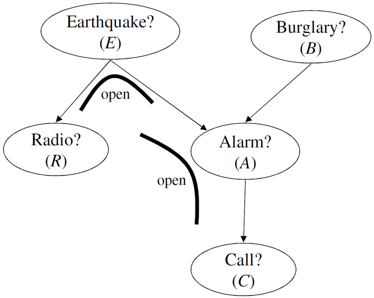


Are B and R d-separated by E and C?

Yes

The closure of only one valve is sufficient to block the path, therefore, establishing d-separation.

D-separation

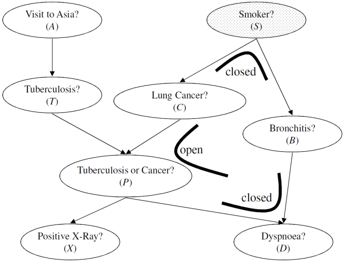


Are C and R d-separated?

No

Both valves are open. Hence, the path is not blocked and d-separation does not hold.

D-separation



Are C and B d-separated by S?

Yes

Both paths between them are blocked by S.

D-separation

The definition of d-separation, $\mathrm{dsep}_G(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$, calls for

considering all paths connecting a node in $\mathbf{X}$ with a node in $\mathbf{Y}$. The number of such paths can be exponential, yet one can implement the test without having to enumerate these paths explicitly.

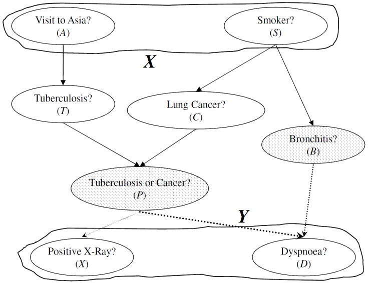
Deciding $\mathrm{dsep}_G(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$ is equivalent to testing whether $\mathbf{X}$ and $\mathbf{Y}$ are disconnected in a new DAG G' obtained by pruning DAG G

- Delete any leaf node W from DAG G as long as W not in $\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$. Repeat until no more nodes can be deleted.
- Delete all edges outgoing from nodes in $\mathbf{Z}$.

Decided in time and space that are linear in the size of DAG G

D-separation

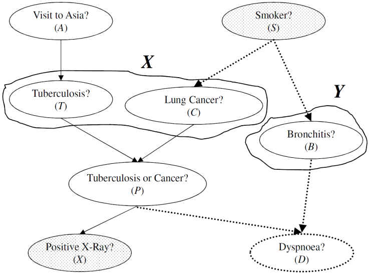
Nodes in **Z** are shaded. Pruned nodes and edges are dotted.



Is $\mathbf{X} = \{A, S\}$ d-separated from $\mathbf{Y} = \{D, X\}$ by $\mathbf{Z} = \{B, P\}$?

D-separation

Nodes in **Z** are shaded. Pruned nodes and edges are dotted.



Is $\mathbf{X} = \{T, C\}$ d-separated from $\mathbf{Y} = \{B\}$ by $\mathbf{Z} = \{S, X\}$?

D-separation

- The d-separation test is sound
  If distribution Pr is induced by Bayesian network G, then

$$dsep_G(\mathbf{X}, \mathbf{Z}, \mathbf{Y}) \textit{ only if } I_{\mathrm{Pr}}(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$$

- The proof of soundness is constructive showing that every independence claimed by d-separation can indeed be derived using the graphoid axioms.

I-map, D-map and P-map

- A directed acyclic graph (a Bayesian network) describes a set of conditional independence assumptions I_G.
- It is an I-map of a distribution I_{Pr} if $I_G \Rightarrow I_{\mathrm{Pr}}$ or $I_G \subseteq I_{\mathrm{Pr}}$
- It is a D-map if $I_{\mathrm{Pr}} \Rightarrow I_G$ or $I_{\mathrm{Pr}} \subseteq I_G$
- It is a P-map if it is both an I-map and a D-map. Namely, $I_G = I_{\mathrm{Pr}}$
- I-maps and D-maps can be constructed trivially. Therefore, we enforce minimality.

Minimal I-maps

Given a distribution Pr, how can we construct a DAG G which is guaranteed to be a minimal I-MAP of Pr?

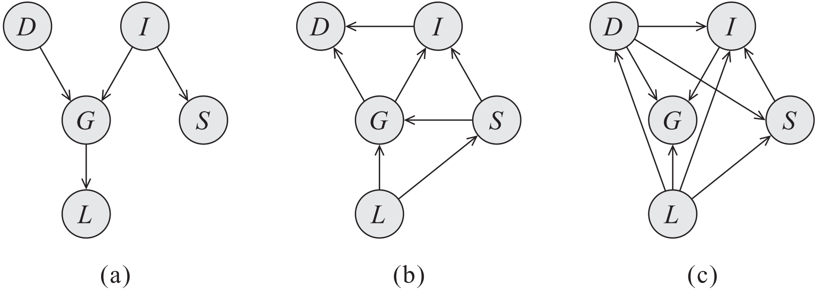
Given an ordering $X_1, \ldots, X_n$ of the variables in Pr:

- Start with an empty DAG G (no edges)
- Consider the variables X_i one by one, for $i = 1, \ldots, n$
- For each variable X_i, identify a minimal subset $\mathbf{P}$ of the variables in $X_1, \ldots, X_{i-1}$ such that
  - $I_{\mathrm{Pr}}(X_i, \mathbf{P}, \{X_1, \ldots, X_{i-1}\} \setminus \mathbf{P})$
  - Make $\mathbf{P}$ the parents of X_i in DAG G

The resulting DAG is a minimal I-MAP of Pr

Minimal I-maps

- Minimal I-maps are not unique.
- Different orderings give rise to different I-maps



Ordering for (b): (L, S, G, I, D)
Ordering for (c): (L, D, S, I, G)

Blankets and Boundaries

A Markov blanket for variable X

is a set of variables which, when known, will render every other variable irrelevant to X

A Markov blanket $\mathbf{B}$ is minimal iff

no strict subset of $\mathbf{B}$ is also a Markov blanket.

A minimal Markov blanket

is called a Markov Boundary.

The Markov Boundary is not unique

unless the distribution is strictly positive.

Blankets and Boundaries

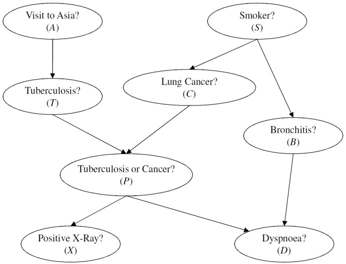
If distribution $\Pr$ is induced by DAG G

then a Markov blanket for variable X with respect to $\Pr$ can be constructed using its parents, children, and spouses in DAG G

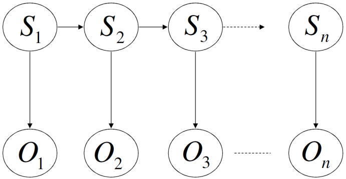
Variable Y is a spouse of X iff

the two variables have a common child in DAG G

Blankets and Boundaries



Markov blanket for C



Markov blanket for S_t, $t > 1$

S_{t-1}, S_{t+1}, O_t