

STAT 535 Lecture 4

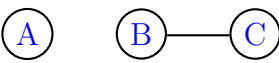
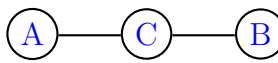
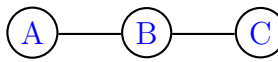
Graphical representations of conditional independence Part II Bayesian Networks

©Marina Meilă

mmp@stat.washington.edu

1 A limitation of Markov networks

Between two variables or subsets of V , there can be four combinations of independence statements.

Conditional Independence	Marginal Independence	
	$A \perp B$	$A \not\perp B$
$A \perp B C$		
$A \not\perp B C$	Not representable by MRF	

The table above gives an example of MRF structure for each of the three cases that are representable by MRF's. For the fourth case, an I-map is possible, and this is the graph that contains an edge $A - B$. Note that this I-map is trivial, since it does not represent the independence $A \perp B$.

This is a limitation of MRF's. We will show how serious this limitation is by an example.

Suppose that a variable called "burglar alarm (A)" becoming true can have two causes: a burglary (B) or an earthquake (E). A reasonable assumption is that burglaries and earthquakes occur independently ($B \perp E$). But, given that the alarm sounds $A = 1$, and hearing that an earthquake as taken place

($E = 1$), most people will believe that the burglary is less likely to have taken place than if there had been no earthquake ($E = 0$), hence E carries information about B when $A = 1$ and therefore B, E are dependent given A . In other words, it is reasonable to assume that

$$B \perp E \tag{1}$$

$$B \not\perp E|A. \tag{2}$$

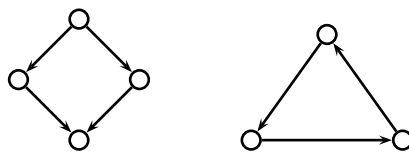
Markov nets cannot represent this situation.

Exercise Find other examples of the kind (i) $X \perp Y$ and $X \not\perp Y|Z$, i.e. two independent causes which can produce the same effect. Find examples that fit the other three possible combinations of marginal and conditional (in)dependence, i.e. (ii) $X \perp Y, X \perp Y|Z$, (iii) $X \not\perp Y, X \perp Y|Z$, (iv) $X \not\perp Y, X \not\perp Y|Z$. Can you describe them in words?

The case illustrated above is important enough to warrant the introduction of another formalism for encoding independencies in graphs, called **D-separation** and based on **directed graphs**.

2 Directed Acyclic Graphs (DAG's)

A **Directed Acyclic Graph (DAGs)** is a directed graph $\mathcal{G} = (V, \vec{\mathcal{E}})$ which contains no **directed cycles**.



this is a DAG this is not a DAG

Below is a somewhat more complicated textbook example (the “chest-clinic” example). In this example, the arrows coincide with “causal links” between

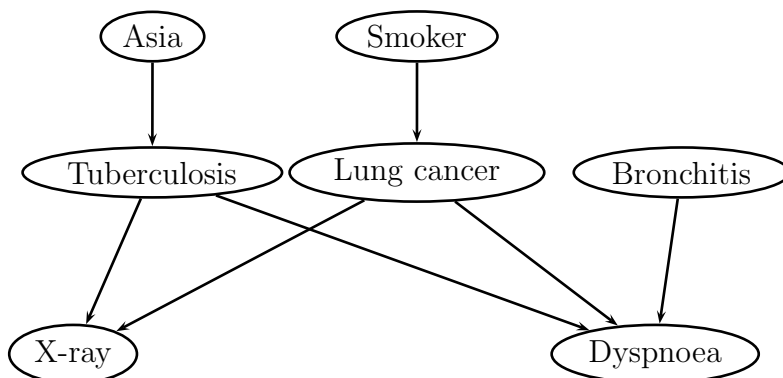


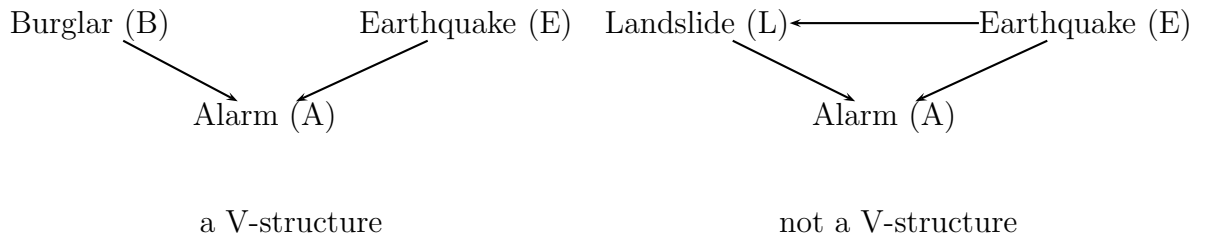
Figure 1: The “Chest clinic” DAG.

variables (i.e “Smoking causes Lung cancer”). This is not completely accidental. Bayesian networks are particularly fit for representing domains where there are causal relationships. It is therefore useful to think of causal relationships when we try to build a Bayes net that represents a problem. But the formalism we study here is not necessarily tied to causality. One does not have to interpret the arrows as cause-effect relations and we will not do so.

Terminology:

parent	Asia is parent of Tuberculosis
$pa(\text{variable})$	the set of parents of a variable
	$pa(\text{X-ray}) = \{ \text{Lung cancer, Tuberculosis} \}$
child	Lung cancer is child of Smoker
ancestor	Smoker is ancestor of Dyspnoea
descendent	Dyspnoea is descendent of Smoker
family	a node and its parents
	$\{ \text{Dyspnoea, Tuberculosis, Lung cancer, Bronchitis} \}$ are a family

But perhaps the most important concept in DAG’s is the **V-structure**, which denotes a variable having two parents which are *not connected* by an edge. The Burglar-Earthquake-Alarm example of the previous section is the V-structure.



In figure 1, (T, X, L) , (T, D, L) , (T, D, B) are V-structures.

3 D-separation

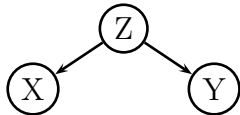
In a DAG, independence is encoded by the relation of d-separation, define below.

$$A \perp B \mid C \iff A \text{ d-separated from } B \text{ by } C$$

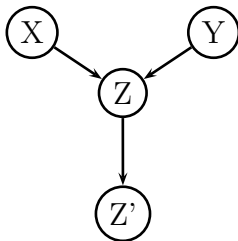
D-separation : A is **d-separated** from B by C if all the paths between sets A and B are blocked by elements of C . The three cases of d-separation:



if $Z \in C$ the path is blocked, otherwise open



if $Z \in C$ the path is blocked, otherwise open



if Z or one of its descendants $\in C$ the path is open, otherwise blocked

The **directed Markov property**: $X \perp \text{its non-descendants} \mid pa(X)$

Theorem For any DAG $\mathcal{G} = (V, \vec{\mathcal{E}})$ and any probability distribution P_V over V , if P_V satisfies the Directed Markov Property w.r.t the DAG \mathcal{G} , then any D-separation relationship that is true in \mathcal{G} corresponds to a true independence relationship in P_V .

(In other words, \mathcal{G} is an I-map of P_V).

3.1 The Markov blanket

In a DAG, the Markov blanket of a node X is the union of all parents, children and other parents of X 's children.

$$\text{Markov blanket}(X) = pa(X) \cup ch(X) \cup \left(\bigcup_{Y \in ch(X)} pa(Y) \setminus \{X\} \right)$$

For example, in the DAG in Figure 1, the Markov blanket of L is the set $\{S$ (parent), X, D (children), T, B (parents of the children)}, and the Markov blanket of A is T .

3.2 Equivalent DAG's

Two DAG's are said to be (**likelihood**) **equivalent** if they encode the same set of independencies. For example, the two graphs below are equivalent (encoding no independencies).

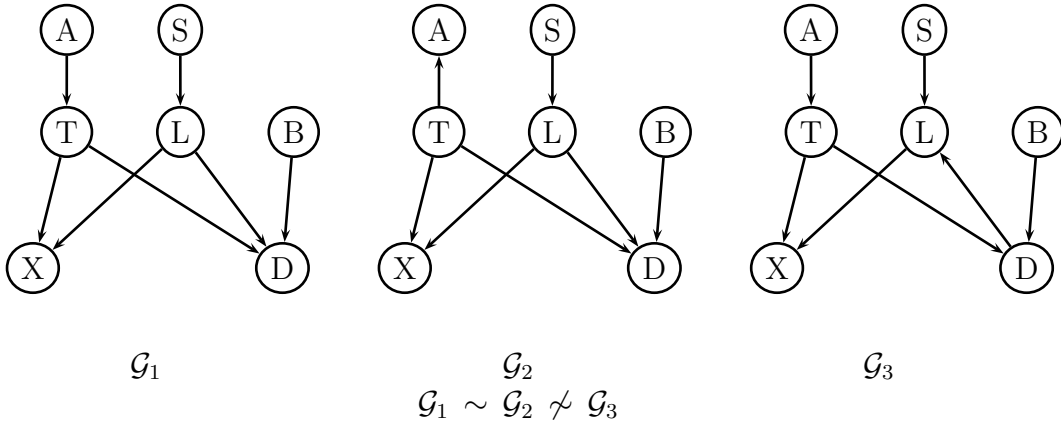
$$A \longrightarrow B \quad A \longleftarrow B$$

If the arrows represented causal links, then the two above graphs would not be equivalent!

Another example of equivalence between DAGs was seen in lecture 3: a Markov chain or a HMM can be represented by a directed graph with "forward" arrows, as well as by one with "backward" arrows. (Note, in passing, that an HMM can also be represented as an undirected graph, demonstrating equivalence between a DAG and a MRF).

Yet another example is below. On the left, the "chest clinic" DAG. In the

middle, an equivalent DAG. On the right, another DAG which is *not* equivalent with the chest clinic example. [**Exercise:** verify that the graphs are/are not equivalent by looking at what independence relationships hold in the three graphs.]



Theorem 1 (Chickering) *Two DAG's are equivalent iff they have the same undirected skeleton and the same V-structures.*

Consequently, we can invert an arrow in a DAG and preserve the same independencies, only if that arrow is not part of a V-structure. In other words, $A \rightarrow B$ can be reversed iff for any arc $\vec{C}A$ pointing at A , there is an arc $\vec{C}B$ pointing into B and viceversa. Or, $pa(B) = \{A\} \cup pa(A)$.

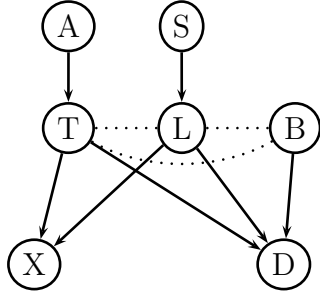
It can be proved that one can traverse the class of all equivalent DAG's by successive arrow reversals.

3.3 D-separation as separation in an undirected graph

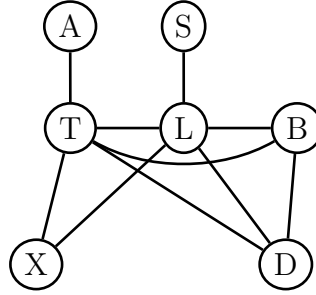
Here we show that D-separation in a DAG is equivalent to separation in an undirected graph obtained from the DAG and the variables we are interested in. First two definitions, whose meaning will become clear shortly.

Moralization is the graph operation of connecting the parents of a V-structure. A DAG is **moralized** if all nodes that share a child have been

connected. After a graph is moralized, all edges, be they original edges or new edges added by moralization, are considered as undirected. If \mathcal{G} is a DAG the graph obtained by moralizing \mathcal{G} is denoted by \mathcal{G}^m and is called the **moral graph** of \mathcal{G} .



marrying the parents



dropping the directions

For any variable X the set $\text{an}(X)$ denotes the ancestors of X (including X itself). Similarly, if A is a set of nodes, $\text{an}(A)$ denotes the set of all ancestors of variables in A .

$$\text{an}(A) = \bigcup_{X \in A} \text{an}(X)$$

The **ancestral graph** of a set of nodes $A \subseteq V$ is the graph $\mathcal{G}_{\text{an}(A)} = (\text{an}(A), E_A)$ obtained from \mathcal{G} by removing all nodes not in $\text{an}(A)$.

Now we can state the main result.

Theorem 2 *Let $A, B, S \subseteq V$ be three disjoint sets of nodes in a DAG \mathcal{G} . Then A, B are D -separated by S in \mathcal{G} iff they are separated by S in the moral ancestral graph of A, B, S .*

$$A \perp B \mid S \text{ in } \mathcal{G} \quad \text{iff} \quad A \perp B \mid S \text{ in } (\mathcal{G}_{\text{an}(A \cup B \cup S)})^m$$

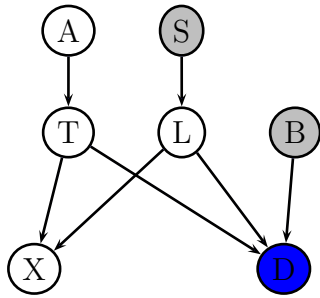
The intuition is that observing/conditioning on a variable creates a dependence between its parents (if it has any). Moralization represents this link. Now why the ancestral graph? Note that an unobserved descendent cannot produce dependencies between its ancestors (ie cannot open a path in a directed graph). So we can safely remove all descendents of A, B that are not

in S . The descendants of S itself that are not in A, B , and all the nodes that are not ancestors of A, B, S can be removed by a similar reasoning. Hence, first the graph is pruned, then dependencies between parents are added by moralization. Now directions on edges can be removed, because DAG's are just like undirected graphs if it weren't for the V-structures, and we have already dealt with those.

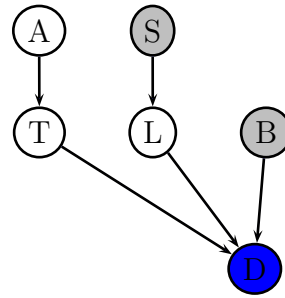
The Theorem immediately suggests an algorithm for testing D-separation using undirected graph separation.

1. remove all nodes not in $\text{an}(A \cup B \cup S)$ to get $\mathcal{G}_{\text{an}(A \cup B \cup S)}$
2. moralize the remaining graph to get $(\mathcal{G}_{\text{an}(A \cup B \cup S)})^m$
3. remove all nodes in S from $(\mathcal{G}_{\text{an}(A \cup B \cup S)})^m$ to get \mathcal{G}'
4. test if there is a path between A and B in \mathcal{G}'

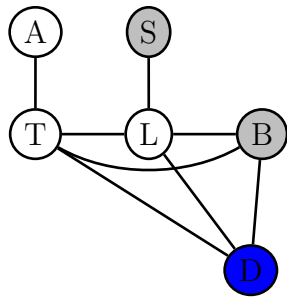
For example, test if $S \perp B \mid D$ in the chest clinic DAG.



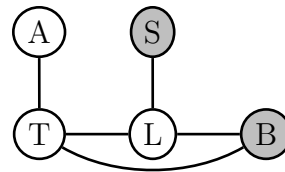
show nodes of interest



ancestral graph



moralize



eliminate conditioning nodes

4 Factorization

Now we construct joint probability distributions that have the independencies specified by a given DAG. Assume the set of discrete variables is $V = \{X_1, X_2, \dots, X_n\}$ and that we are given a DAG $\mathcal{G} = (V, \mathcal{E})$. The goal is to construct the family of distributions that are represented by the graph. This family is given by

$$P(X_1, X_2, \dots, X_n) = \prod_{i=1}^n P(X_i | \text{pa}(X_i)) \quad (3)$$

In the above $P(X_i | \text{pa}(X_i))$ represents the conditional distribution of variable X_i given its parents. Because the factors $P(X_i | \text{pa}(X_i))$ involve a variable

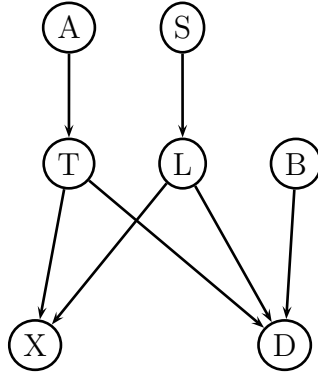


Figure 2: The “chest clinic” DAG, with shorter variable names.

and its parents, that is, nodes closely connected in the graph structure, we often call them **local** probability tables (or local distributions).

Note that the parameters of each local table are (functionally) independent of the parameters in the other tables. We can choose them separately, and the set of all parameters for all conditional probability distributions form the family of distributions for which the DAG \mathcal{G} is an I-map.

If a distribution can be written in the form (3) we say that the distribution **factors according to the graph** \mathcal{G} . A joint distributions that factors according to some DAG \mathcal{G} is called a **Bayes net**.

Note that any distribution is a Bayes net in a trivial way: by taking \mathcal{G} to be the complete graph, with no missing edges. In general, we want a Bayes net to be as sparse as possible, because representing independences explicitly has many computational advantages.

The Bayes net described by this graph is

$$P(A, S, T, L, B, X, D) = P(A)P(S)P(T|A)P(L|S)P(B)P(X|T, L)P(D|T, L, B)$$

A way of obtaining this decomposition starting from the graph is

1. Construct a topological ordering of the variables. A **topological or-**

dering is an ordering of the variables where the parents of each variable are always before the variable itself in the ordering.

A, S, T, L, B, X, D is a topological ordering for the graph above. This ordering is not unique; another possible topological ordering is A, B, T, S, L, D, X . In general, there can be exponentially many topological orderings for a given DAG.

2. Apply the chain rule following the topological ordering.

$$P(A, S, T, L, B, X, D) = P(A)P(S|A)P(T|A, S)P(L|A, S, T)P(B|A, S, T, L)P(X|A, S, T, L, B)P(D|A, S, T, L, B, X, S)$$

3. Use the directed Markov property to simplify the factors

$$\begin{aligned} P(S|A) &= P(S) \\ P(T|A, S) &= P(T|A) \\ P(L|A, S, T) &= P(L|S) \\ P(B|A, S, T, L) &= P(B), \text{ etc.} \end{aligned}$$

Let us now look at the number of parameters in such a model. Assume that in the example above all variables are binary. Then the number of unconstrained parameters in the model is

$$1 + 1 + 2 + 2 + 1 + 4 + 8 = 19$$

The number of parameters in a 7 way contingency table is $2^7 - 1 = 127$ so we are saving 118 parameters. As we shall see, there are also other computational advantages to joint distribution representations of this form.