Lecture 4: January 26, 2015
cs 573: Probabilistic Reasoning
Professor Nevatia

Spring 2015

Review

- Assignment #1 due today
- Assignment #2 to be posted in two parts; part (a) to be posted today
  - Please start early, exercises should help in reading of the book material
- Previous lecture:
  - Bayesian Network Representation
  - Local Independencies
  - I-Map and factorization
- Today's objective
  - Global dependencies
  - Distributions to graphs

Naïve Bayes Discussion

- Used for classification tasks such as disease based on symptoms, object based on some features etc
- Naïve Bayes assumption (evidence conditionally independent given cause) is typically not valid but many examples show very good results, often hard to beat by including some dependencies
  - Possible reasons: easier to estimate the smaller number of parameters and to elicit them from an expert.
  - Errors in estimation may dominate the more accurate models constructed by more complex Bayes networks.

Factorization Theorems

Let $\mathcal{G}$ be a BN graph over the variables $X_1, \ldots, X_n$. We say that a distribution P over the same space factorizes *according to $\mathcal{G}$ if P can be expressed as a product*

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid \mathrm{Pa}^{\mathcal{G}}_{X_i}). \qquad (3.17)$$

Thm 3.1

Let $\mathcal{G}$ be a BN structure over a set of random variables $\mathcal{X}$, and let P be a joint distribution over the same space. If $\mathcal{G}$ is an I-map for P, then P factorizes according to $\mathcal{G}$.

Thm 3.2

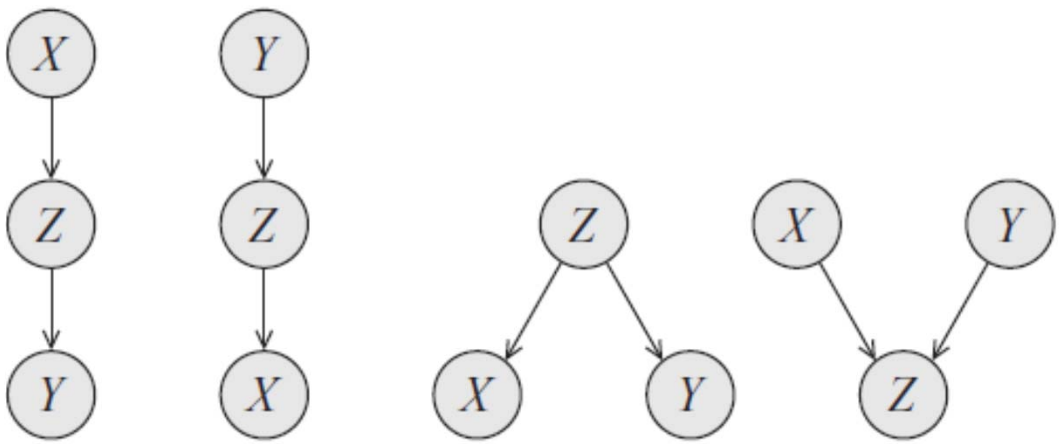
Let $\mathcal{G}$ be a BN structure over a set of random variables $\mathcal{X}$ and let P be a joint distribution over the same space. If P factorizes according to $\mathcal{G}$, then $\mathcal{G}$ is an I-map for P.

Global independencies

- Local independencies are defined just for a node and its ancestors given the parents
- Global dependencies define conditional independencies between any sets of nodes given values of another set of nodes
- Consider the “student” network:
  - is “L” independent of “S” given “G”?
  - is “D” independent of “I” given “L”?
  - ….
- We will next consider an algorithmic way to decide on such independencies.

Influence Flow: Short Trails

- Consider if influence can flow from X to Y *via* Z, *i.e.* consider if *trail* (note, we ignore directions) X, Z, Y is *active*
- Four cases (from Fig. 3.5)
- (a): Causal Trail: X → Z →Y; active *iff* Z is **not** observed
- (b): Evidential Trail: X ← Z ← Y; active *iff* Z is **not** observed
- (c): Common Cause: X ← Z →Y; active *iff* Z is **not** observed
- (d): Common Effect: X → Z ← Y; active *iff* Z or one (or more) of Z's descendants **is** observed



(a) (b) (c) (d)

Note: (d) is called a v-structure

Active Trails and d-separation

- Let $(X_1, X_2, \ldots X_n)$ be a trail in graph G. For influence to flow from X_1 to X_n , it needs to flow through every node in the trail
- Let **Z** be a subset of observed variables; trail $(X_1, X_2, \ldots X_n)$ is said to be active if:

  Whenever we have a v-structure of the form $X_{i-1} \rightarrow X_i \leftarrow X_{i+1}$, X_i or one of its descendants is in **Z**.

  No other node in the trail is in **Z**.
- Let **X**, **Y**, **Z** be three sets of nodes in G; we say **X** and **Y** are d-separated given **Z**, d-sepG (**X**;**Y**|**Z**) if there is no active trail between any node X in **X** and any node Y in **Y**, given Z.
- Define I(G) = {(**X**⊥ **Y**| **Z**); d-sepG(**X**;**Y**|**Z**)}
  - these define the **global** independencies of graph G

Soundness and Completeness of d-separation

- *Soundness*: If two nodes X and Y are d-separated given some **Z**, then we are guaranteed that they are also conditionally independent given Z (d-separation does not make "wrong" assertions): See Theorem 3.3 in the book. Proof is in chapter 4.
- *Completeness*: d-separation detects *all* possible independencies for all possible distributions that factor over the graph *i.e.* I(P) = I(G).
  - Does not hold in general but holds for *almost all* distributions P over G
  - *almost all* is formally defined in *measure theory* (*measure 0*): set of all rationals in [0,1] is of measure zero

Computing d-separation

- We can enumerate all trails between two nodes and check if they are active. This can be exponential in the size of the graph.
- Algorithm 3.1 computes all nodes reachable from a node X in linear time.
  - Basic idea:

    1. Going from leaves to the roots, mark all nodes that are in **Z** or are ancestors of **Z** (have descendants in **Z**).

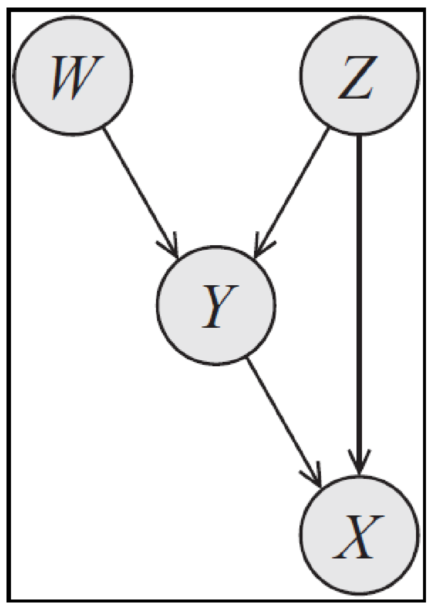
    2. Traverse, breadth-first, from X to Y: stop traversal if

       we hit a middle node of a v-structure that is *not* marked in step 1,

       or is not such a node and is in **Z**.
  - Details of the algorithm omitted; essentially implements the above rules carefully.

d-separation Example with a Loop

- Let Y be the observed variable.
- Start from X, Go to Y, trail is blocked at Y
- Go from X to Z, then to Y and W; this trail is not blocked (due to v-structure)
- Need to keep track if node traversal is from bottom or from top



Markov Blanket

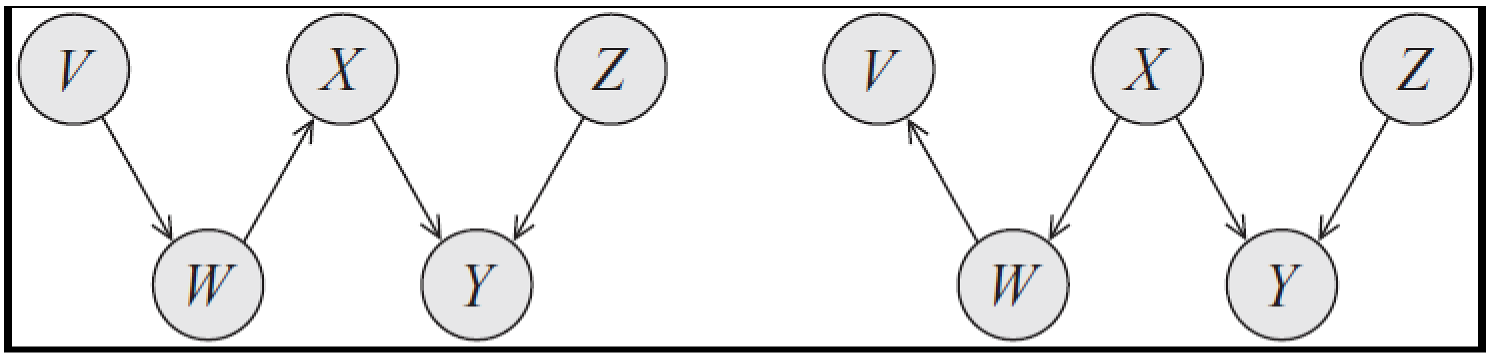
- (Covered much later in the book)
- MB of a node: parents of the node, children of the node and parents of the children
- Given MB, node is conditional independent of all other nodes

$$(X_i \perp \mathbf{X}_{-i} - \mathbf{MB}(X_i) \mid \mathbf{MB}(X_i))$$

- An Example

I-equivalence

- Two graphs are *I-equivalent* if they represent the same independencies
- *Skeleton* of a graph: undirected graph by removing directions on edges
- Two graphs with same skeleton and same set of *v-structures* are I-equivalent; see example below



- This is a sufficient but not a necessary condition
  - A necessary and sufficient definition follows

Immorality

- *Immorality*: If the parents of a node in v-structure are not connected (*married*), there is an immorality.
  - *If* there is an edge between parents, it is called a *covering* edge.
- Two graphs have the same skeleton and same set of immoralities *if and only if* they are I-equivalent
  - Note: *iff* above implies a necessary and sufficient condition

Distributions to Graphs

- How to find optimal (at least good) graphs for a distribution?
  - Note that the distribution is rarely given in full joint distribution form
- Any complete graph is an I-map for any distribution
- Minimal I-map: no edges can be removed to retain I-map property
- Given a variable ordering, we can construct a minimal I-map (may not be unique). Different orderings may give graph structures of very different complexity (see Fig 3.8, slide follows)
  - Examine each variable in turn, by its ordering
  - Assign all previous nodes as parents of current node, remove those that can be removed without violating the independence property that $(X_i \perp \{X_1, X_2 \ldots X_{i-1}\} - \mathbf{U} | \mathbf{U})$, **U** is a subset of potential parents
  - Example on next slide

Examples

- Consider ordering L, S, G, I, D
- L is root
- S: should L be a parent? Yes, because ($S \perp L$) does not hold
- G: parents are subsets of {L, S}. Clearly G depends on L. G is independent of S *given* I but not independent *given* L, so L must also be a parent
- I: parents in set {L,S,G}. Can remove L if G is a parent of I….

$$(L \perp I, D, S \mid G)$$

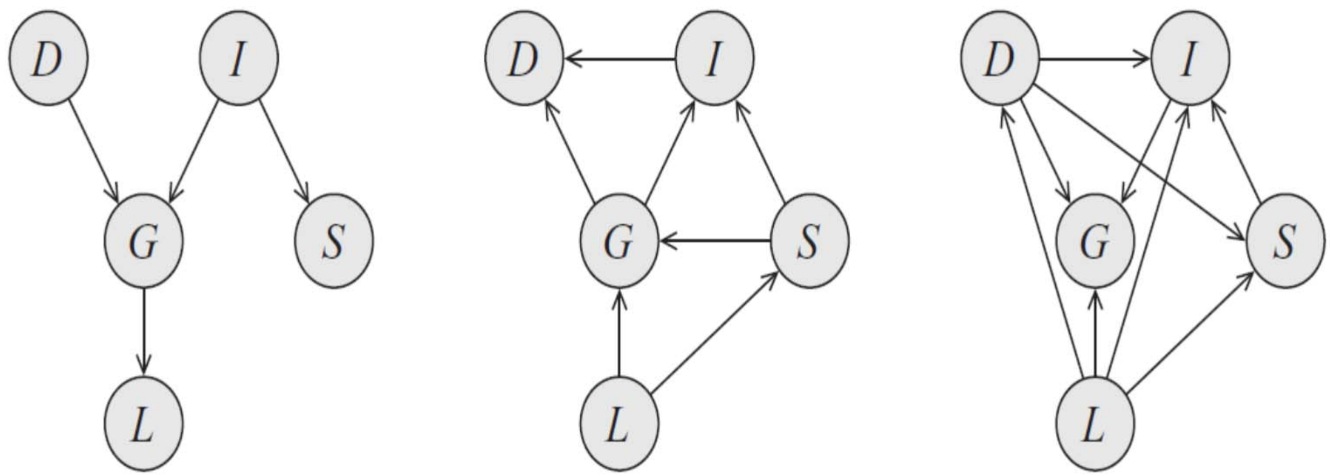
$$(S \perp D, G, L \mid I)$$

$$(G \perp S \mid I, D)$$

$$(I \perp D).$$

$$(D \perp I, S)$$

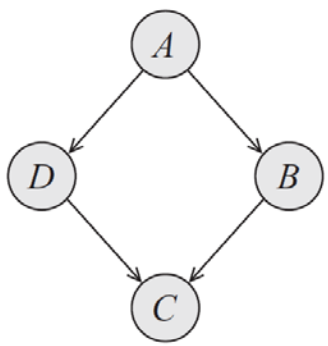
Figure 3.8

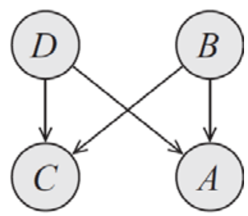


(a) (b) (c)

P-Maps

- Minimal I-maps may not capture all the independencies of P. *Perfect Maps* (P-map) do this.
- There may not be a P-map for every distribution
- Consider (A⊥C| B,D) and (B ⊥D|A, C)
- Neither of the figures below satisfy both
  - First one satisfies (A⊥C| B,D) but (B⊥D|A), not (B ⊥D|A, C)
  - In second version, B and D are independent





- We skip algorithm to discover P-maps (when they do exist), see section 3.4.3 (marked with a *), if interested.

How to go from Distributions to Graphs?

- What is the form of the input?
  - Are the independences given explicitly or must we infer them from data?
- Construct the structure from independences
  - Structure may not be unique; choose the "simplest" one
- Learn the parameters
  - Ask a domain expert?
  - From "training" data
    - Not all variable values may be given in the training data (learn from missing data)
- Parameter learning to be studied in later part of the course
  - Structure learning if time permits.

BN Summary

- Graphical representations that capture conditional independencies in distributions
  - Compact, intuitive
- Inference by enumeration (compute entries in the joint) and marginalization
  - More efficient algorithms to follow
- I-maps and P-maps
- d-separation and Markov Blanket

Next Class

- Read sections 5.3.1.1, 5.4 and 5.5 of the KF book