10-708: Probabilistic Graphical Models 10-708, Spring 2016

2 : Directed GMs: Bayesian Networks

Lecturer: Eric P. Xing *Scribes: Lidan Mu, Lanxiao Xu*

1 Notation

The notations used in this course are defined as follows:

Variable, value and index: Variables are names representing some numbers, denoted by upper-case letters, such as V, S and the value of a variable is a realization of the variable, usually denoted by lower-case letters, such as v, s. Index corresponds to conditions used to distinguish between different scenarios, which can be subscript or superscript, such as V_i and v_i.

Random variable: Random variables are usually represented by upper-case letters such as X, Y, Z.

Random vector and matrix: Random vectors are multivariate distributions represented by vectors, usually denoted by bold upper-case letters such as $\mathbf{X} = \begin{pmatrix} X_1 \\ \vdots \\ X_n \end{pmatrix}$. Similarly andom matrices are denoted by bold upper-case letters such as $\mathbf{X} = \begin{pmatrix} X_{11} & \cdots & X_{1n} \\ X_{21} & \cdots & X_{2n} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{nn} \end{pmatrix}$.

Parameters: Parameters are in general represented by Greek letters such as α, β.

2 Example: The Dishonest Casino

Suppose that you are in a casino. The casino has two dice, where one is a fair die

$$P(1) = P(2) = P(3) = P(4) = P(5) = P(6) = \frac{1}{6},$$

and also a loaded die

$$P(1) = P(2) = P(3) = P(4) = P(5) = \frac{1}{10}, \quad P(6) = \frac{1}{2}.$$

The casino player switches back ans forth between the fair and loaded die once every 20 turns.

You first bet 1 dollar and roll with the fair die. The casino player also rolls a die, but it may be the fair die or the loaded die. The one with highest number wins 2 dollars from the game.

Given a sequence X of rolls by the casino player, there are a few questions that we are interested in:

Evaluation problem: How likely is this sequence, given our model of how the casino works? This should be given by $P(X|\theta)$ where θ represents the parameters of our model.

Decoding problem: What portion of the sequence was generated with the fair die, and what portion with the loaded die?

Learning problem: How "loaded" is the loaded die? Hoe "fair" is the fair die? How often does the casino player change from fair to loaded, and back? The model we learn should give us this information.

We can actually convert this casino game scenario into a graphical model problem using knowledge engineering. First we need to pick the variables which can be observed or hidden, discrete or continuous. Denote the result of the ith rolling as X_i and the indicator of which die is used as Y_i. Here notice that X_i's are observed during the game while the values of Y_i's are hidden. Then we pick the structure to represent the relationship between random variables we just choose. Since the probability of choosing a loaded die depends on the previous one, we can use a so called hidden Markov model. Finally we need to pick the probability distributions in the model. As we can easily see, the probability of seeing a particular result depends on the die we pick, and thus the model is shown in Figure 1.

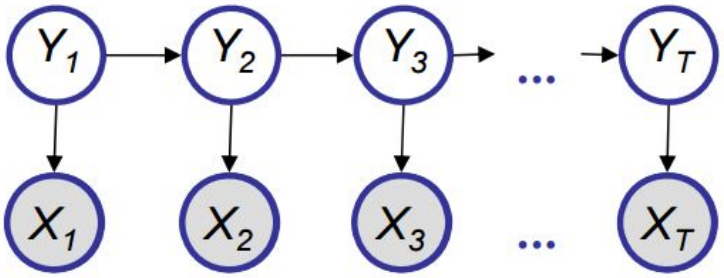

Figure 1: A hidden Markov model representation of the casino game.

Given a sequence $X = X_1 \cdots X_T$ and a parse $Y = Y_1 \cdots Y_T$, the graphical model helps us to answer questions we previously mentioned. If we want to find out how likely the parse given our hidden Markov model and the observed sequence, we can compute the joint probability (also called the complete probability) as

$$\begin{aligned} p(X,Y) &= p(X_1,\cdots,X_T,Y_1,\cdots,Y_T) \\ &= p(Y_1)p(X_1|Y_1)p(Y_2|Y_1)p(X_2|Y_2)\cdots p(Y_T|Y_{T-1})p(X_T|Y_T) \\ &= p(Y_1)p(Y_2|Y_1)\cdots p(Y_T|Y_{T-1}) \times p(X_1|Y_1)\cdots p(X_T|Y_T) \\ &= p(Y_1,\cdots,Y_T)p(X_1\cdots X_T|Y_1,\cdots,Y_T). \end{aligned}$$

And we can also get the marginal probability

$$p(X) = \sum_Y p(X,Y) = \sum_{Y_1}\sum_{Y_2}\cdots\sum_{Y_T} \pi_{Y_1} \prod_{t=2}^{T} a_{Y_{t-1},Y_T} \prod_{t=1}^{T} p(X_t|Y_t),$$

but it takes exponential time to do this summation. We will be talking about how to do this efficiently in future chapters.

3 Bayesian Network

A **Bayesian network** (BN) is a directed graph whose nodes represent the random variables and whose edges represent direct influence of one variable on another. When a node is connected to the other node with a directed arrow such as $X \to Y$, it means that Y is caused by X.

It is a data structure that provides the skeleton for representing a **joint distribution** compactly in a **factorized** way and also offers a compact representation for a set of **conditional independence assumptions** about a distribution.

We can view the graph as encoding a generative sampling process executed by nature where the value for each variable is selected by nature using a distribution that depends only on its parents. In other words, each variable is a stochastic function of its parents.

3.1 Factorization Theorem

Theorem: Given a DAG, the most general form of the probability distribution that is **consistent with** the graph factors according to "**node given its parents**"

$$P(X) = \prod_{i=1\cdots d} P(X_i|X_{\pi_i}),$$

where X_{π_i} is the set of parents of X_i and d is the number of nodes (random variables) in the graph.

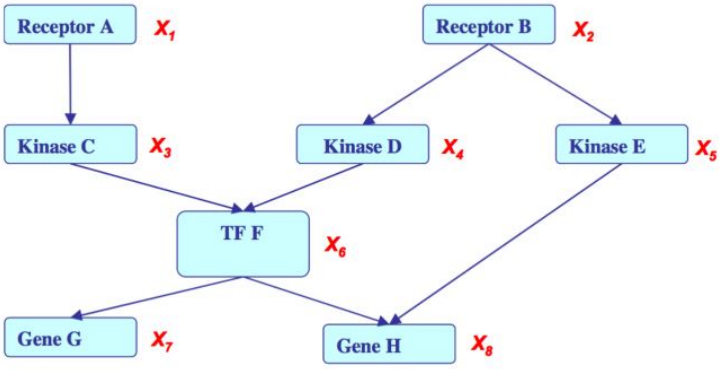

Figure 2: An example of factorization theorem.

For example the following joint probability is derived from Figure 2:

$$\begin{aligned} &P(X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8) \\ &= P(X_1)P(X_2)P(X_3|X_1)P(X_4|X_2)P(X+5|X_2)P(X_6|X_3, X_4)P(X_7|X_6)P(X_8|X_5, X_6) \end{aligned}$$

3.2 Local Structures and Independencies

There are three types of local structures in graphical models (Fgure 3).

3.2.1 Common parent

In this structure, two nodes A and C share the same parents. Fixing B decouples A and C, that is, A and C are independent given B.

This can be justified by

$$\begin{aligned} P(A, C|B) &= \frac{P(A, B, C)}{P(B)} \\ &= \frac{P(B)P(A|B)P(C|B)}{P(B)} \\ &= P(A|B)P(C|B) \end{aligned}$$

3.2.2 Cascade

In this structure, node A has an edge to node B, which has an edge to node C. Fixing B again decouples A and C, that is, A and C are independent given B.

We can justify it by

$$\begin{aligned} P(A, C|B) &= \frac{P(A, B, C)}{P(B)} \\ &= \frac{P(A)P(B|A)P(C|B)}{P(B)} \\ &= \frac{P(A, B)P(C|B)}{P(B)} \\ &= P(A|B)P(C|B) \end{aligned}$$

3.2.3 V-structure

In this structure, node C has two parents A and B. Knowing C would couple A and B, meaning that A and B are originally independent if we don't know C.

This can be justified by thinking of a real world example. Let A denote the fact that the clock in the classroom is 5 minutes fast, B represent the statement that there is a traffic jam on Highland Park Bridge, and C denote the observation that Eric is late for class. Apparently having a traffic jam is independent of any problem with the clock. However, if we know that Eric comes to class late, knowing that there is no traffic jam means a higher probability of the clock being fast. Therefore, the two events now become dependent.

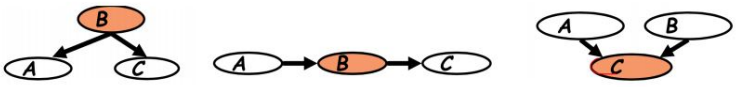

Figure 3: Three types of local structures.

3.3 I-maps

Definition: Let P be a distribution over X. We define $I(P)$ to be the set of independence assertions of the form $X \perp Y|Z$ that hold in P (however we set the parameter values).

Definition: Let K be any graph object associated with a set of independencies $I(K)$. We say that K is an **I-map** for a set of independencies I, if $I(K) \subseteq I$. We now say that G is an I-map for P if G is an I-map for $I(P)$, where we use $I(G)$ as the set of independencies associated.

3.3.1 Facts about I-map

For G to be an I-map of P, any independence that G asserts must also hold in P. Conversely, P may have additional indepndencies that are not reflected in G.

3.3.2 local Markov assumptions

Given a Bayesian network structure G whose nodes represent random variables $X_1, \cdots, X_n$.

Definition: Let Pa_{X_i} denote the parents of X_i in G, and $NonDescendatas_{X_i}$ denote the variables in the graph that are not descendants of X_i. Then G encodes the following set of **local conditional independence assumptions** $I_l(G)$

$$I_l(G) : \{X_i \perp NonDescendats_{X_i} | Pa_{X_i} : \forall i\}.$$

In other words, each node X_i is independent of its nondescendants given its parents.

3.4 Global Markov assumptions

The global Markov assumptions are related to the concept of D-separation (D stands for Directed edges). Let X, Y, Z be three sets of nodes in G. That X and Y are d-separated given Z, if they are conditionally independent given Z. There are two ways to define d-separation:

(1) D-separation by Moralized Ancestral Graph

Definition: Variables x and y are D-separated (conditionally independent) given z if they are separated in the *moralized* ancestral graph.

There are two steps to generate a moralized ancestral graph from the original BN (illustrated in Figure 4). In the first step we construct an ancestral graph from the original graph by deleting the descendants of the query nodes (in the case of Figure 4 x, y and z are the query nodes). In the second step we moralized the ancestral graph by creating undirected edges between nodes that are not connected yet and have at least one common child.

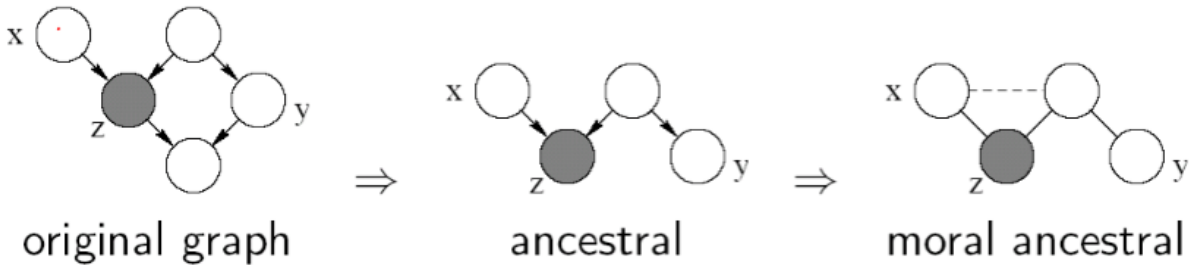

Figure 4: An Illustration of Constructing Moralized Ancestral Graph

(2) D-separation by Bayes Ball Algorithm

Definition: X is D-separated from Y given Z if we can't send a ball from any node in X to any node in Y using the "*Bayes ball*" algorithm, which means that there are no active trail between any node $x \in X$ and $y \in Y$ given Z.

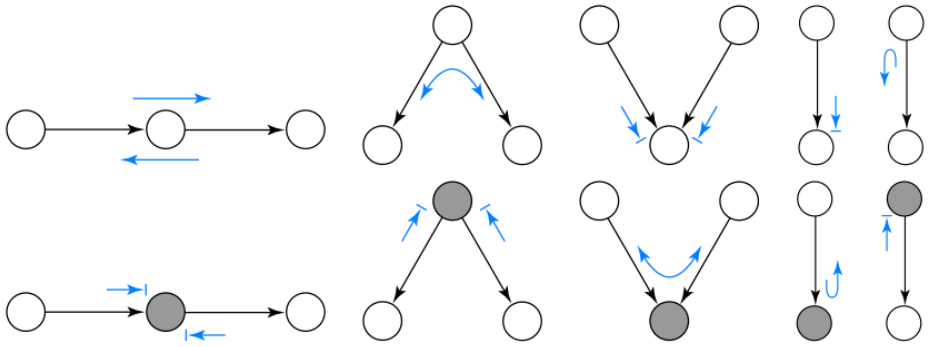

Figure 5: An Illustration of Bayes Ball Algorithm

Figure 5 shows an illustration of the Bayes ball algorithm. An undirected trail is active if a Bayes ball travelling along the graph and never encounters the stop symbol.

D-separation can be used as an approach to reveal the conditional independencies characterized by a graph. Let $I(G)$ denote all independence properties that correspond to d-separation. In the example shown in Figure 6, the elements of $I(G)$ are:

$$x_1 \perp x_2 \qquad x_1 \perp x_2 \mid x_4 \qquad x_2 \perp x_4 \qquad x_2 \perp x_4 \mid x_1 \qquad x_3 \perp x_4 \mid x_1$$

From the example above, we reach a conclusion that separation properties in the graph imply independence properties about the associated variables.

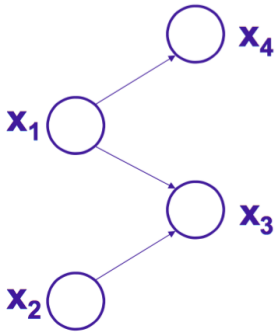

Figure 6

3.5 Quantitative Specification of Probability Distributions

The Equivalence Theorem: For a graph G,
Let $\mathcal{D}_1$ denote the family of all distributions that satisfy $I(G)$,
Let $\mathcal{D}_2$ denote the family of all distributions that factor according to G,

$$P(X) = \prod_{i=1:d} P(X_i | X_{\pi_i})$$

Then $\mathcal{D}_1 \equiv \mathcal{D}_2$.

For the graph to be useful, any conditional independence properties we can derive from the graph should hold for the probability distribution that the graph represents. According to the equivalence theorem, for one distribution, we only need to test whether it can be factored according to G instead of testing every independence conditions in $I(G)$ against it. For a Bayesian network (G, P), where P factorizes over G, we only need to specify P as a set of conditional probability tables (CPTs) for discrete random variables or a set of conditional probability density functions (CPDs) for continuous random variables.

3.6 Soundness and Completeness

D-separation is sound and "complete" w.r.t BN factorization law. The **soundness** property states that:

$$\text{If a distribution } P \text{ factorizes according to } G, \text{ then } I(G) \subseteq I(P).$$

That is, the graph G is the I-map for $I(P)$ and can not generate more independencies than those implied by the distribution. The **completeness** property states that:

$$\text{For any distribution } P \text{ that factorizes over } G, \text{ if } (X \perp Y \mid Z) \in I(P), \text{ then d-}sep_G(X;Y|Z)$$

And naturally, we need to ask the validity of the contrapositive of the completeness statement: If X and Y are not d-separated given Z in G, then are X and Y dependent in all distributions P that factorize over G? The answer is no. Even if a distribution factorizes over G, it can still contain additional indepencencies that are not reflected in the structure. For example, there is a graph with two nodes A and B, where there is a directed edge from A to B. The graph implies that A and B are dependent. However, there exist $P(A, B)$ in Table 1 which satisfies $P(A, B) = P(A)P(B)$. So the distribution implies that A and B are not dependent.

A	b^0	b^1
a^0	0.08	0.32
a^1	0.12	0.48

Table 1

Theorem: Let G be a BN graph. If X and Y are not d-separated given Z in G, then X and Y are dependent in **some** distribution P that factorizes over G.

Theorem: For almost all distributions P that factorize over G, i.e., for all distributions except for a set of "measure zero" in the space of CPD parametrization, we have that $I(P) = I(G)$.

3.7 I-equivalence

It should be noted that very different BN graphs can actually be equivalent, in that they encode precisely the same set of conditional independence assertions. As shown in Figure 7, all of the BNs encode the same conditional independence: $X \perp Y \mid Z$.

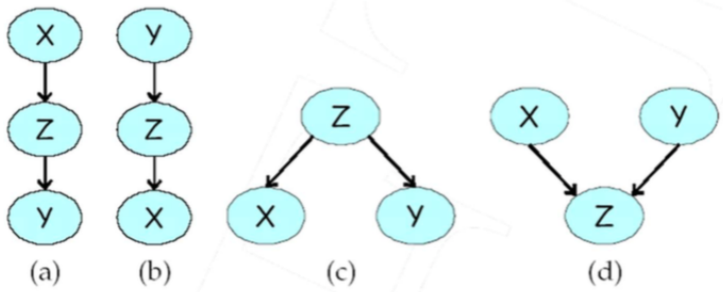

Figure 7

Definition: Two BN graphs $G1$ and $G2$ over X are I-equivalent if $I(G1) = I(G2)$.

Recalling the previous discussions about I-map, we can see that a complete graph is a (trivial) I-map for any distribution, yet it does not reveal any of the independence structure in the distribution. Naturally, we want to find an I-map of $I(P)$ such that it reveals as many independence relationships as possible, yet still $\subseteq I(P)$.

Definition: A graph object G is a minimal I-map for a set of independencies I if it is an I-map for I, and if the removal of even a single edge from G renders it not an I-map.

Note: Minimum I-map is not unique, there can exist multiple minimum I-maps for a single set of independencies I.

4 Summary

- **Definition:** A Bayesian network is a pair (G, P) where P factorizes over G, and where P is specified as set of local conditional probability distributions (CPDs). CPDs are associated with G's nodes.
- A BN captures "causality", "generative schemes", "asymmetric influences", etc., between entities.
- Local and global independence properties are identifiable via d-separation criteria (Bayes ball).
- Computing joint likelihood amounts multiplying CPDs, but computing marginal and conducting inference can be hard.
- **True:** Graphical Models require a localist semantics for the nodes.
- **False:** Graphical Models require a causal semantics for the edges.
- **False:** Graphical Models are necessarily Bayesian.