10-708: Probabilistic Graphical Models, Spring 2015

2 : Directed GMs: Bayesian Networks

Lecturer: Eric P. Xing *Scribes: Yi Cheng, Cong Lu*

1 Notation

Here the notations used in this course are defined:

Random variables and values: Random variables are denoted by upper-case letters, such as A, B, C and the value of random variables are denoted by lower-case letters, such as $P(A = a|B = b, C = c)$, and here a, b, c are realizations for A, B, C.

Random vectors: Random vectors are also denoted by upper-case letters, but they are vectors, for example: $X = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix}$, sometimes, it's also denoted as $\overrightarrow{X}$. Here the subscript means the dimension of the random vector. For example X_1 is the first dimension of X.

Random matrix: Random matrix is usually denoted by bold upper-case letter, such as $\mathbf{X} = \begin{bmatrix} X_1^{(0)} & X_1^{(1)} \\ X_2^{(0)} & X_2^{(1)} \\ X_3^{(0)} & X_3^{(1)} \end{bmatrix}$. Usually the superscript denotes the data index. So $X_i^{(j)}$ denotes the i_{th} dimensional random variable of j_{th} data.

Parameters: Parameters are usually denoted by Greek letters, such as: $\alpha, \beta, ...$

2 Example: The Dishonest Casino

In the casino game, if you bet \$1, you and casino player rolls the dice and the one gets the higher number gets \$2. The casino player has two dice, one is fair and the other is loaded which gives a higher probability to get number six.

In this situation, when given a sequence of rolling result, there would be three questions that can be asked:

Evaluation: The probability of the sequence in our model, which is $P(X|m)$, where m is our model and X is the result.

Decoding: Which portion of the sequence was generated by fair die and which was generated by loaded die. This would be $P(y|X)$, where y denotes the die.

Learning: We want to learn that how skewed the loaded die is and how often the casino player changes from fair to loaded and back.

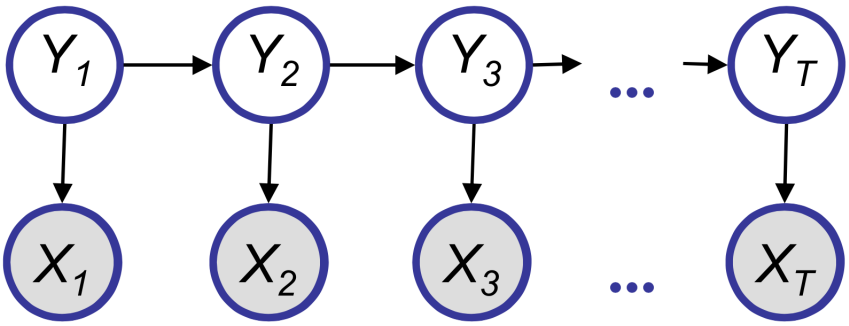

Figure 1: This is a hidden markov model. X_is are observed random variables and Y_is are hidden random variables. The probability of Y_i is determined by the previous Y_{i-1}. In this example, X_i denotes the rolling result in i_{th} turn and Y_i denotes the die in that turn.

Now let's convert these into graphic model problem. First we need to pick the randoms which can be observed or not observed in this situation. Here we pick X_i to denote the result of a rolling and y to denote which die was used in i_{th} rolling. As we can see, X_is are observed and y_is are hidden. Then we need to choose a structure to represent the model. The structure of the model denotes the causality and relationship of the random variables. Here we assume (or by statistics) the probability of choosing a loaded die is determined by the previous one. Actually we can also choose other structures. In the end, we need to pick the probability for the distribution encoded in this model. The final model is shown in Fig. 1.

Now given a sequence of $\mathbf{x} = x_1, ..., x_n$ and a parse $\mathbf{y} = y_1, ..., y_n$, we can make use of the graphic model defined above to answer questions like how likely is the parse.

3 Bayesian Network

A Bayesian Network (BN) is a kind of probabilistic graphical model that represents a set of random variables and their conditional dependencies via a directed acyclic graph (DAG). The nodes in BN represents the random variables and edges represent the direct influence of one variable on another. When a node is connected to the other node, such as $A \rightarrow B$, it means that A causes B. With BN, the distribution can be factorized by the graph given, which will be introduced in the next section. A random variable is independent with its non-descendant random variables given its parents.

3.1 Factorization Theorem

Given a BN with structure G, a distribution of it can be factorized as $P(\mathbf{X}) = \Pi_{i=1:d} P(X_i|\mathbf{X}_{\pi_i})$, where $\mathbf{X}_{\pi_i}$ denotes the set of parents of X_i in the graph. For example, when given the graph in Figure 2, the join distribution can be written as following:

$$P(\mathbf{X_1}, \mathbf{X_2}, \mathbf{X_3}, \mathbf{X_4}, \mathbf{X_5}, \mathbf{X_6}, \mathbf{X_7}, \mathbf{X_8}) = P(\mathbf{X_1})P(\mathbf{X_2})P(\mathbf{X_3}|\mathbf{X_1})P(\mathbf{X_4}|\mathbf{X_2})P(\mathbf{X_5}|\mathbf{X_2})P(\mathbf{X_6}|\mathbf{X_3}, \mathbf{X_4})P(\mathbf{X_7}|\mathbf{X_6})P(\mathbf{X_8}|\mathbf{X_5}, \mathbf{X_6}) \quad (1)$$

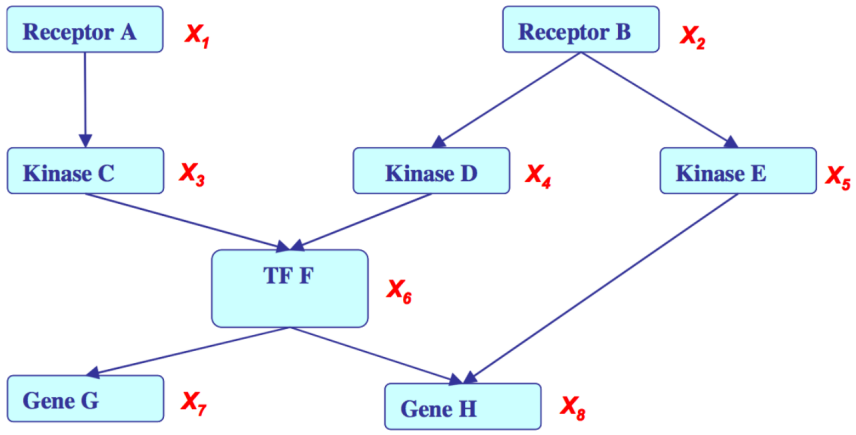

Figure 2: Example of factorization theorem.

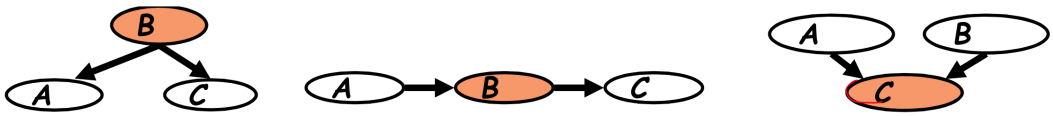

Figure 3: The left is the structure of common parent, and encode $A \perp\!\!\!\perp C|B$. The middle one is cascade structure and encode $A \perp\!\!\!\perp C|B$. The last one is v-structure and encode $A \not\perp\!\!\!\perp B|C$

3.2 Local Structures and Independencies

For a graph in BN, there are three typical local structures: *a*) common parent; *b*) cascade; *c*) V-structure . The three kinds of structures are shown in Fig. 3

Common parent: In this structure, two nodes has a same parent node. When given their parents, these nodes are independent. This is used in factorization theorem. A proof of $P(AC|B) = P(A|B)P(C|B)$ is as following:

$$\begin{aligned} P(A, C|B) &= \frac{P(A, B, C)}{P(B)} \\ &= \frac{P(B)P(A|B)P(C|B)}{P(B)} \\ &= P(A|B)P(C|B) \end{aligned} \tag{2}$$

Cascade: In this structure node A has an edge to B and B has an edge to C. It's shown in the middle graph of Fig. 3. In this structure, when given the middle node B, A and C are independent. A proof

of $P(AC|B) = P(A|B)P(C|B)$ is as following:

$$\begin{aligned} P(A, C|B) &= \frac{P(A, B, C)}{P(B)} \\ &= \frac{P(A)P(B|A)P(C|B)}{P(B)} \\ &= \frac{P(A, B)P(C|B)}{P(B)} \\ &= P(A|B)P(C|B) \end{aligned} \tag{3}$$

V-structure: In this structure, node C has two parents A and B. The shape of the structure is like letter V, and so it's called V-structure. When C is not given, A and B are independent. But when C is given, A and B is no longer independent. This can be thought as that when the result C is given, A and B are competing to explain it. Here is an example of it. Let A denote *the clock is incorrect*, B denote *There is a traffic jam* and C denote *Eric came to class late*. Here we can see *there is a traffic jam* is independent with *clock is incorrect*. But when we know *Eric come to class late*, the event *there is traffic jam* and *clock is incorrect* is dependent and they are connected by *Eric come to class late*. The reason *Eric come to class late* is maybe because *the clock is incorrect* or *there is traffic jam*.

3.3 Introduction to I-maps

A graph G is an I-map of a distribution P if every independence assertion implied by G is also implied by P. However, P may have additional independence assertion that are not reflected in G.

More formally, 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, and similarly $I(K)$ to be the set of independence assertions associated with graph object K. We say that K is an I-map for a set of independences I if $I(K) \subseteq I$. Based on these definitions, we now say that G is an I-map for P if G is an I-map for $I(P)$, where $I(G)$ is the set of independences associated with G.

3.4 Markovian Independence Assumptions

The structure of G asserts independence relations that reflect both local Markov assumptions and global Markov assumptions. Let us consider G, a directed acyclic graph whose nodes represent random variables $X_1, ..., X_n$. The local Markov assumption is that each node X_i is independent of its non-descendants given its parents. More formally, Let P_{aX_i} denote the parents of X_i in G, and $Non - descendants_{X_i}$ denote the variables in the graph that are not descendants of X_i. Then G encodes the following set of local conditional independence assumptions $I_l(G)$: $X_i \perp Non - descendants_{X_i} \mid P_{aX_i} : \forall i$.

The global Markov assumptions are related to the notion of D-separation. Two nodes X and Y are D-separated given node Z if they are conditional independent given Z. There are two equivalent ways to establish D-separation in a graph. The first involves derivation of the moralized ancestral graph, while the other requires classifying trails as either active or inactive.

To get the moralized ancestral graph, we first construct the ancestral graph which includes only variables X, Y, Z, and all the ancestors of any of these variables (their parents, their parents parents, etc.). We then moralize the ancestral graph by marrying the parents, inserting an undirected edge between any two variables in the ancestral graph that have a common child.

A trail is a path between three variables, for simplicity X, Y, and Z. There are exactly four ways this trail can be active:

1. Causal Trail: $X \rightarrow Z \rightarrow Y$, active if and only if Z is not observed
2. Evidential Trail: $X \leftarrow Z \leftarrow Y$, active if and only if Z is not observed
3. Common Cause: $X \leftarrow Z \rightarrow Y$, active if and only if Z is not observed
4. Common Effect: $X \rightarrow Z \leftarrow Y$, active if and only if Z (or any of its descendants) is observed

Notice that common effect behaves differently than the other kinds of active trails. The others are activated by having the middle node unobserved, but common effect is activated by observations. With either criterion for establishing D-separation, we can finally define $I_g(G)$ to be all the independence properties that correspond to D-separation: $I(G) = X \perp Y|Z : dsepG(X \perp Y|Z)$.

3.5 Quantitative Specification of Probability Distributions

Separation properties in the graph imply independence properties about the associated variables. The equivalence theorem states that:

Theorem For a graph G, let D_1 denote the family of all distributions that satisfy $I(G)$. Let D_2 denote the family of all distributions that factor according to G. Then $D_1 \equiv D_2$.

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. Because of the Equivalence Theorem, we can specify distributions P as a set of conditional probability density (CPD) functions, one for each variable conditioned on its parents. A CPD function for a variable allows us to compute its probability distribution given the values of its parents. For discrete-valued distributions, these can be represented as the familiar conditional probability tables (CPTs).

3.6 Soundness and Completeness:Distributional equivalence and I-equivalence

D-separation is sound with respect to the Bayesian Network Factorization Theorem. Soundness is a desirable property because it means that if a distribution P factorizes according to G, then $I(G) \subseteq I(P)$. We would also like the completeness property: for any distribution P that factorizes over G, if $(X \perp Y|Z) \in I(P)$, then $dsepG(X \perp Y|Z)$. Equivalently, we may ask: if X and Y are not d-separated given Z in G, then are X and Y are dependent in all distributions P that factorize over G? This is false, because a distribution that factorizes over G may yet contain additional accidental independences not asserted by the graph structure. Thus we have the following theorems:

Theorem 1 Let G be a Bayesian Network 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 2 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 parameterizations, we have that $I(P) = I(G)$.

Very different BN graphs can actually be equivalent, in the sense that they encode precisely the same set of conditional independence assertions. For example, these three following graphs:

1. $X \to Z \to Y$,
2. $Y \to Z \to X$,
3. $X \leftarrow Z \to Y$,

they all represent the same independence assumption $(X \perp Y|Z)$. Two BN graphs G_1 and G_2 over X are I-equivalent if $I(G_1) = I(G_2)$. Specifically,

- The set of all graphs over X is partitioned into a set of mutually exclusive and exhaustive I-equivalence classes, which are the set of equivalence classes induced by the I-equivalence relation.
- Any distribution P that can be factorized over one of these graphs can be factorized over the other.
- Furthermore, there is no intrinsic property of P that would allow us associate it with one graph rather than an equivalent one.
- This observation has important implications with respect to our ability to determine the directionality of influence.

To detect I-equivalence, we can apply the following theorem, where the skeleton of a Bayesian network graph G over V is an undirected graph over V that contains an edge X, Y for every edge (X, Y) in G.

Theorem : Let G_1 and G_2 be two graphs over V . If G_1 and G_2 have the same skeleton and the same set of V-structure then they are I-equivalent.

To find an I-map for distribution P, one simple way is to use the a complete graph, which is a (trivial) I-map for any distribution, yet it does not reveal any of the independence structure in the distribution. This intuition leads to the following definition of a minimal I-map:

Definition: Minimal I-map A graph object G is a minimal I-map for a set of independences 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.

We should also notice that the minimum I-map is not unique.

3.7 An example of Bayesian Network: Hidden Markov Model

One simple example of Bayesian Networks is the Hidden Markov Model (HMM). HMM is a dynamic mixture model as shown in Figure 1, where Y_i is the hidden state and X_i is the observed variable. Given a sequence $X = X_1, ..., X_T$ and a parse $Y = Y_1, ..., Y_T$, we can find how likely the parse is. According to the graph we can factorize the joint probability as:

$$\begin{aligned} p(x, y) &= p(x_1...x_T, y_1...y_T) \\ &= p(y_1)p(x_1|y_1)p(y_2)p(x_2|y_2)...p(y_T|y_{T-1})p(x_T|y_T) \\ &= p(y_1, ..., y_T)p(x_1, ..., x_T|y_1, ..., y_T) \end{aligned} \tag{4}$$

3.8 Some Statements of Graphical Models

- Graphical Models require a localist semantics for the nodes. **True**
- Graphical Models require a causal semantics for the edges. **False**
- Graphical Models are necessarily Bayesian. **False**