CS281A/Stat241A: Statistical Learning Theory

Lecture 2 (8/28/02)

Lecturer: Michael I. Jordan *Scribes: Edwards and Lobay*

1 Overview

The following is a quick overview of the topics we will cover concerning Graphical Models:

1. **Representation:** conditional independence, families of probability distributions.
2. **Parameterization:** subfamilies of distributions (e.g., the exponential family).
3. **Inference:** computational issues in finding $p(x_A|x_B)$
4. **Estimation:** choosing the "best" parameters for a probability model, given a dataset.
5. **Model Selection:** finding the graph which best represents the implicit relationships between variables in a dataset.

2 Directed Graphical Models

We will look at directed graphical models in two ways. First, we parametrically define a family of probability distributions associated with the graph by assigning functions to each node that, when taken together, define a joint probability distribution. Next, we introduce an alternative way of defining a family of distributions by finding the conditional independencies that are asserted by a graph. In a later lecture, we show that these two approaches define identical families.

2.1 The Parametric Approach

Let $G(V, E)$ be a graph with vertex (or, node) set V and edge set E. For each node, define a function which is parameterized by that node and its parents. Define the joint probability as

$$p(x_1, x_2, ..., x_n) \triangleq \prod_{i=1}^{n} f_i(x_i, x_{\pi_i})$$

where π_i is the set of indices of the parents of the ith node. These functions are non-negative and must sum (or integrate) to one. It can be shown that this defines a valid joint probability distribution. Furthermore, these functions can be viewed as conditional probabilities $f_i(x_i, x_{\pi_i}) = p(x_i|x_{\pi_i})$ Thus, an alternative equation of the joint, derived from our original definition is

$$p(x_1, x_2, ..., x_n) \triangleq \prod_{i=1}^{n} p(x_i|x_{\pi_i})$$

2.2 The Conditional Independence Approach

Recall the two equivalent ways of establishing conditional independence:

$$\begin{aligned} \{X_A \perp\!\!\!\perp X_B | X_C\} &\iff p(x_A, x_B|x_C) = p(x_A|x_C)p(x_B|x_C) \\ &\iff p(x_A|x_B, x_C) = p(x_A|x_C) \end{aligned}$$

We will associate a set of conditional independence statements with a graph G of the form $\{X_i \perp\!\!\!\perp X_{\nu_i} | X_{\pi_i}\}$, where the nodes X_{π_i} are, as before, the parents of X_i and X_{ν_i} are those nodes which appear before i in a topological ordering.

We can prove that these conditional independencies are valid by derivation directly from the formula for the joint. The set of independencies derived by this method is not a complete list of conditional independencies associated with the graph. This list does however form a basis in the sense that all conditional independencies can be inferred from it.

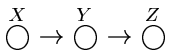
It's important to note that conditional independencies which are not associated with a graph can still be true in specific realizations of the parameters of the graph. However, it impossible to create a joint probability that *does not* respect the independencies in the list without leaving the family of distributions defined by the graph

2.3 Independence Queries and the Bayes Ball Algorithm

Our next task is to derive a graph-theoretic algorithm that will let us answer questions about conditional independencies of the form. In particular, given sets of nodes X_A,X_B and X_C, is $X_A \perp\!\!\!\perp X_B | X_C$?

Let's look at three canonical graphs in the sense that we can build any graph by stitching together these pieces.

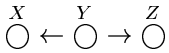
The first canonical graph is a chain:



We can show that the independence relation $X \perp\!\!\!\perp Z|Y$ holds for this graph since

$$p(z|x, y) = \frac{p(x, y, z)}{p(x, y)} = \frac{p(x)p(y|x)p(z|y)}{p(x)p(y|x)} = p(z|y)$$

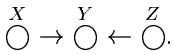
The second canonical graph is:



This is know as the "hidden cause" case. For example, let X be the presence or absence of gray hair, Z be shoe size, and Y be chronological age. Clearly, X and Y are highly correlated because children tend to have no gray hair and smaller feet. However, given that we know a person's age, knowing their hair color gives

us no additional information about their shoe size, and vice versa. Thus, given age, hair color and shoe size are independent.

The third canonical graph is:



In this graph, we have a marginal independence $X \perp\!\!\!\perp Z$ but we do *not* have the conditional independence $X \perp\!\!\!\perp Z|Y$. This is known as "explaining away". Let Y be the event that your friend is late to lunch, X be the event that aliens have abducted your friend, and Z be the event that your watch is slow. In general, whether or not your friend has been abducted by aliens provides no information about whether or not your watch is slow (i.e., marginal independence). Given that you know that you think your friend is late, however, the relative probabilities that you assign to the two possible causes may change drastically. In particular, if you realize that your watch is slow, then the chance that aliens abducted your friend decreases.

For sets X_A,X_B and X_C of nodes, $X_A \perp\!\!\!\perp X_B|X_C$ if X_C *d-separates* X_A and X_B. This is equivalent to asking whether X_A is reachable from X_B using a special set of rules that determine whether it is possible to pass through a given node. We will define these rules on the three canonical graphs listed above and construct an algorithm, the "Bayes Ball" algorithm, to answer queries regarding conditional independence.

(Bayes Ball is described in detail on p. 16 of the text.)

The general idea is to first shade the nodes X_C which we are conditioning on. Now, place balls at the nodes in some set X_B and allow them to bounce around the graph (as allowed by the rules of major league Bayes Ball). One of them either finds a node of set X_A, in which case you reply that the two sets X_B, X_A are not necessarily conditionally independent, or they all die out, in which case you reply that they are conditionally independent. It is possible to design an implementation of this algorithm that runs in linear time $O(V + E)$.

Finally, note that the algorithm had a slight bug and has been fixed. See updated version of the text.