CS 188 Introduction to Artificial Intelligence
Spring 2023 Note 15

These lecture notes are based on notes originally written by Josh Hug and Jacky Liang. They have been heavily updated by Regina Wang.

Last updated: January 15, 2023

Probability Rundown

We're assuming that you've learned the foundations of probability in CS70, so these notes will assume a basic understanding of standard concepts in probability like PDFs, conditional probabilities, independence, and conditional independence. Here we provide a brief summary of probability rules we will be using.

A **random variable** represents an event whose outcome is unknown. A **probability distribution** is an assignment of weights to outcomes. Probability distributions must satisfy the following conditions:

$$0 \leq P(\omega) \leq 1$$
$$\sum_{\omega} P(\omega) = 1$$

For instance if A is a binary variable (can only take on two values) then $P(A=0) = p$ and $P(A=1) = 1-p$ for some $p \in [0,1]$.

We will use the convention that capital letters refer to random variables and lowercase letters refer to some specific outcome of that random variable.

We use the notation $P(A,B,C)$ to denote the **joint distribution** of the variables A,B,C. In joint distributions ordering does not matter i.e. $P(A,B,C) = P(C,B,A)$.

We can expand a joint distribution using the **chain rule**, also sometimes referred to as the product rule.

$$P(A,B) = P(A|B)P(B) = P(B|A)P(A)$$
$$P(A_1,A_2...A_k) = P(A_1)P(A_2|A_1)...P(A_k|A_1...A_{k-1})$$

The **marginal distribution** of A,B can be obtained by summing out all possible values that variable C can take as $P(A,B) = \sum_c P(A,B,C=c)$. The marginal distribution of A can also be obtained as $P(A) = \sum_b \sum_c P(A,B=b,C=c)$. We will also sometimes refer to the process of marginalization as "summing out".

When we do operations on probability distributions, sometimes we get distributions that do not necessarily sum to 1. To fix this, we **normalize**: take the sum of all entries in the distribution and divide each entry by that sum.

Conditional probabilities assign probabilities to events conditioned on some known facts. For instance $P(A|B=b)$ gives the probability distribution of A given that we know the value of B equals b. Conditional probabilities are defined as:

$$P(A|B) = \frac{P(A,B)}{P(B)}.$$

Combining the above definition of conditional probability and the chain rule, we get the **Bayes Rule**:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

To write that random variables A and B are **mutually independent**, we write $A \perp\!\!\!\perp B$. This is equivalent to $B \perp\!\!\!\perp A$.

When A and B are mutually independent, $P(A,B) = P(A)P(B)$. An example you can think of are two independent coin flips. You may be familiar with mutual independence as just 'independence' in other courses. We can derive from the above equation and the chain rule that $P(A|B) = P(A)$ and $P(B|A) = P(B)$.

To write that random variables A and B are **conditionally independent** given another random variable C, we write $A \perp\!\!\!\perp B|C$. This is also equivalent to $B \perp\!\!\!\perp A|C$.

If A and B are conditionally independent given C, then $P(A,B|C) = P(A|C)P(B|C)$. This means that if we have knowledge about the value of C, then B and A do not affect each other. Equivalent to the above definition of conditional independence are the relations $P(A|B,C) = P(A|C)$ and $P(B|A,C) = P(B|C)$. Notice how these three equations are equivalent to the three equations for mutual independence, just with an added conditional on C!

Probabilistic Inference

In artificial intelligence, we often want to model the relationships between various nondeterministic events. If the weather predicts a 40% chance of rain, should I carry my umbrella? How many scoops of ice cream should I get if the more scoops I get, the more likely I am to drop it all? If there was an accident 15 minutes ago on the freeway on my route to Oracle Arena to watch the Warriors' game, should I leave now or in 30 minutes? All of these questions (and many more) can be answered with **probabilistic inference**.

In previous sections of this class, we modeled the world as existing in a specific state that is always known. For the next several weeks, we will instead use a new model where each possible state for the world has its own probability. For example, we might build a weather model, where the state consists of the season, temperature and weather. Our model might say that $P(winter, 35°, cloudy) = 0.023$. This number represents the probability of the specific outcome that it is winter, 35°, and cloudy.

More precisely, our model is a **joint distribution**, i.e. a table of probabilities which captures the likelihood of each possible **outcome**, also known as an **assignment** of variables. As an example, consider the table below:

Season	Temperature	Weather	Probability
summer	hot	sun	0.30
summer	hot	rain	0.05
summer	cold	sun	0.10
summer	cold	rain	0.05
winter	hot	sun	0.10
winter	hot	rain	0.05
winter	cold	sun	0.15
winter	cold	rain	0.20

This model allows us to answer questions that might be of interest to us, for example:

- What is the probability that it is sunny? $P(W = sun)$
- What is the probability distribution for the weather, given that we know it is winter? $P(W|S = winter)$
- What is the probability that it is winter, given that we know it is rainy and cold? $P(S = winter|T = cold, W = rain)$
- What is the probability distribution for the weather and season give that we know that it is cold? $P(S,W|T = cold)$

Inference By Enumeration

Given a joint PDF, we can trivially compute any desired probability distribution $P(Q_1 \dots Q_m|e_1 \dots e_n)$ using a simple and intuitive procedure known as **inference by enumeration**, for which we define three types of variables we will be dealing with:

1. **Query variables** Q_i, which are unknown and appear on the left side of the conditional (|) in the desired probability distribution.
2. **Evidence variables** e_i, which are observed variables whose values are known and appear on the right side of the conditional (|) in the desired probability distribution.
3. **Hidden variables**, which are values present in the overall joint distribution but not in the desired distribution.

In Inference By Enumeration, we follow the following algorithm:

1. Collect all the rows consistent with the observed evidence variables.
2. Sum out (marginalize) all the hidden variables.
3. Normalize the table so that it is a probability distribution (i.e. values sum to 1)

For example, if we wanted to compute $P(W|S = winter)$ using the above joint distribution, we'd select the four rows where S is winter, then sum out over T and normalize. This yields the following probability table:

W	S	Unnormalized Sum	Probability
sun	winter	$0.10 + 0.15 = 0.25$	$0.25/(0.25 + 0.25) = 0.5$
rain	winter	$0.05 + 0.20 = 0.25$	$0.25/(0.25 + 0.25) = 0.5$

Hence $P(W = sun|S = winter) = 0.5$ and $P(W = rain|S = winter) = 0.5$, and we learn that in winter there's a 50% chance of sun and a 50% chance of rain.

So long as we have the joint PDF table, inference by enumeration (IBE) can be used to compute any desired probability distribution, even for multiple query variables $Q_1 ... Q_m$.

Bayesian Network Representation

While inference by enumeration can compute probabilities for any query we might desire, representing an entire joint distribution in the memory of a computer is impractical for real problems — if each of n variables we wish to represent can take on d possible values (it has a **domain** of size d), then our joint distribution table will have d^n entries, exponential in the number of variables and quite impractical to store!

Bayes nets avoid this issue by taking advantage of the idea of conditional probability. Rather than storing information in a giant table, probabilities are instead distributed across a number of smaller conditional probability tables along with a **directed acyclic graph** (DAG) which captures the relationships between variables. The local probability tables and the DAG together encode enough information to compute any probability distribution that we could have computed given the entire large joint distribution. We will see how this works in the next section

We formally define a Bayes Net as consisting of:

- A directed acyclic graph of nodes, one per variable X.
- A conditional distribution for each node $P(X|A_1 \dots A_n)$, where A_i is the i^{th} parent of X, stored as a **conditional probability table** or CPT. Each CPT has $n+2$ columns: one for the values of each of the n parent variables $A_1 \dots A_n$, one for the values of X, and one for the conditional probability of X given its parents.

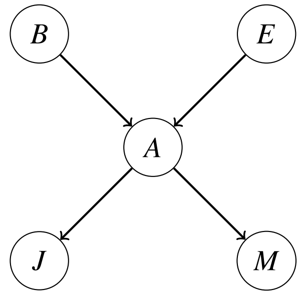
The structure of the Bayes Net graph encodes conditional independence relations between different nodes. These conditional independences allow us to store multiple small tables instead of one large one.

It is important to remember that the edges between Bayes Net nodes do not mean there is specifically a *causal* relationship between those nodes, or that the variables are necessarily dependent on one another. It just means that there may be *some* relationship between the nodes.

As an example of a Bayes Net, consider a model where we have five binary random variables described below:

- B: Burglary occurs.
- A: Alarm goes off.
- E: Earthquake occurs.
- J: John calls.
- M: Mary calls.

Assume the alarm can go off if either a burglary or an earthquake occurs, and that Mary and John will call if they hear the alarm. We can represent these dependencies with the graph shown below.



In this Bayes Net, we would store probability tables $P(B), P(E), P(A|B,E), P(J|A)$ and $P(M|A)$.

Given all of the CPTs for a graph, we can calculate the probability of a given assignment using the following rule:

$$P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | parents(X_i))$$

For the alarm model above, we can actually calculate the probability of a joint probability as follows: $P(-b,-e,+a,+j,-m) = P(-b) \cdot P(-e) \cdot P(+a|-b,-e) \cdot P(+j|+a) \cdot P(-m|+a)$.

We will see how this relation holds in the next section.

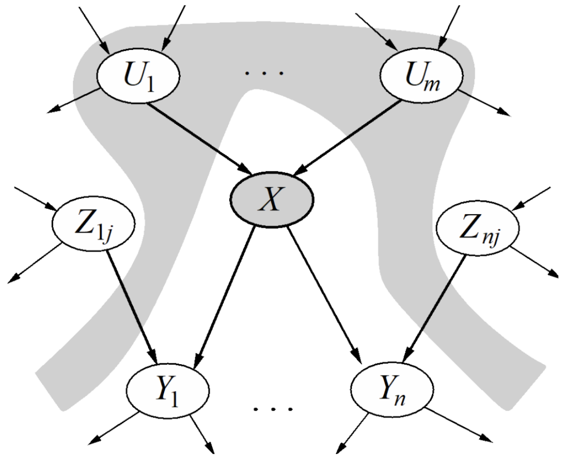
As a reality check, it's important to internalize that Bayes Nets are only a type of model. Models attempt to capture the way the world works, but because they are always a simplification they are always wrong. However, with good modeling choices they can still be good enough approximations that they are useful for solving real problems in the real world.

In general, a good model may not account for every variable or even every interaction between variables. But by making modeling assumptions in the structure of the graph, we can produce incredibly efficient inference techniques that are often more practically useful than simple procedures like inference by enumeration.

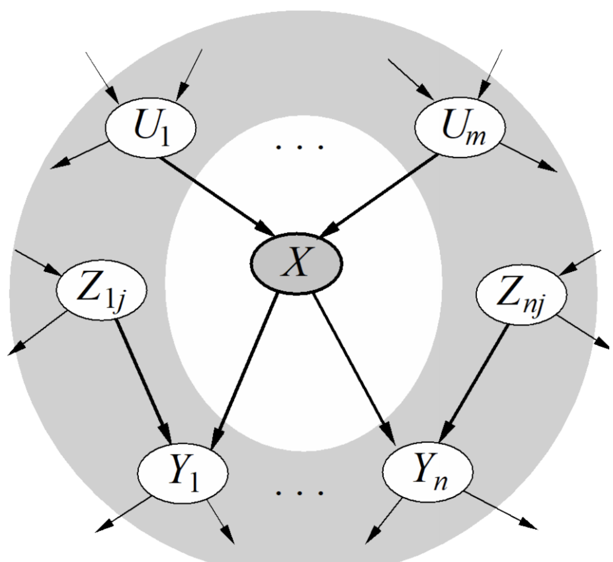
Structure of Bayes Nets

In this class, we will refer to two rules for Bayes Net independences that can be inferred by looking at the graphical structure of the Bayes Net:

- **Each node is conditionally independent of all its ancestor nodes (non-descendants) in the graph, given all of its parents**.



- **Each node is conditionally independent of all other variables given its Markov blanket.** A variable's Markov blanket consists of parents, children, children's other parents.

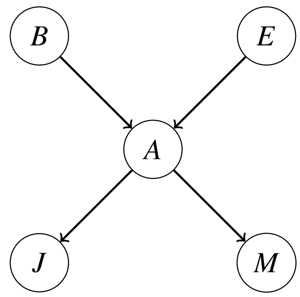


Using these tools, we can return to the assertion in the previous section: that we can get the joint distribution of all variables by joining the CPTs of the Bayes Net.

$$P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | parents(X_i))$$

This relation between the joint distribution and the CPTs of the Bayes net works because of the conditional independence relationships given by the graph. We will prove this using an example.

Let's revisit the previous example. We have the CPTs $P(B), P(E), P(A|B,E), P(J|A)$ and $P(M|A)$, and the following graph:



For this Bayes net, we are trying to prove the following relation:

$$P(B,E,A,J,M) = P(B)P(E)P(A|B,E)P(J|A)P(M|A) \tag{1}$$

We can expand the joint distribution another way: using the chain rule. If we expand the joint distribution with topological ordering (parents before children), we get the following equation

$$P(B,E,A,J,M) = P(B)P(E|B)P(A|B,E)P(J|B,E,A)P(M|B,E,A,J) \tag{2}$$

Notice that in Equation (1) every variable is represented in a CPT $P(var|Parents(var))$, while in Equation (2), every variable is represented in a CPT $P(var|Parents(var), Ancestors(var))$.

We rely on the first conditional independence relation above, that **each node is conditionally independent of all its ancestor nodes in the graph, given all of its parents**[1].

Therefore, in a Bayes net, $P(var|Parents(var), Ancestors(var)) = P(var|Parents(var))$, so Equation (1) and Equation (2) are equal. The conditional independences in a Bayes Net allow for multiple smaller conditional probability tables to represent the entire joint probability distribution.

[1] Elsewhere, the assumption may be defined as "a node is conditionally independent of its *non-descendants* given its parents." We always want to make the minimum assumption possible and prove what we need, so we will use the ancestors assumption.