Foundations of AI | **Fall 2024**
Professors Greenwald and Ewing | **Knowledge Representation**

Bayesian Networks

All models are wrong; some are useful.

– George Box

An overarching goal of AI is to build artificial agents that can reason about the world around them. Doing so requires representing an agent's knowledge about its surroundings in some way. For example, knowledge about how to play a game can be represented as an adversarial search problem, by defining states, actions, transitions, etc.

More generally, a logical language (e.g., propositional logic) can be used to represent an agent's knowledge, beyond the realm of any specific search problem. For example, an agent's knowledge base might include sentences such as "CS 410 lectures are on Mondays, Wednesdays, and Fridays," and "Today is Friday," based on which the agent could infer that "a CS 410 lecture is happening today."

Logical inference—deriving new sentences from old—can be very powerful in a *certain* world, e.g., one in which CS 410 lectures are never cancelled. But says tomorrow is Monday, and there is a 90% chance of 1 foot of snow tomorrow and Brown cancels classes with probability 50% whenever there is 1 foot of snow. Inference under uncertainty is our next topic.

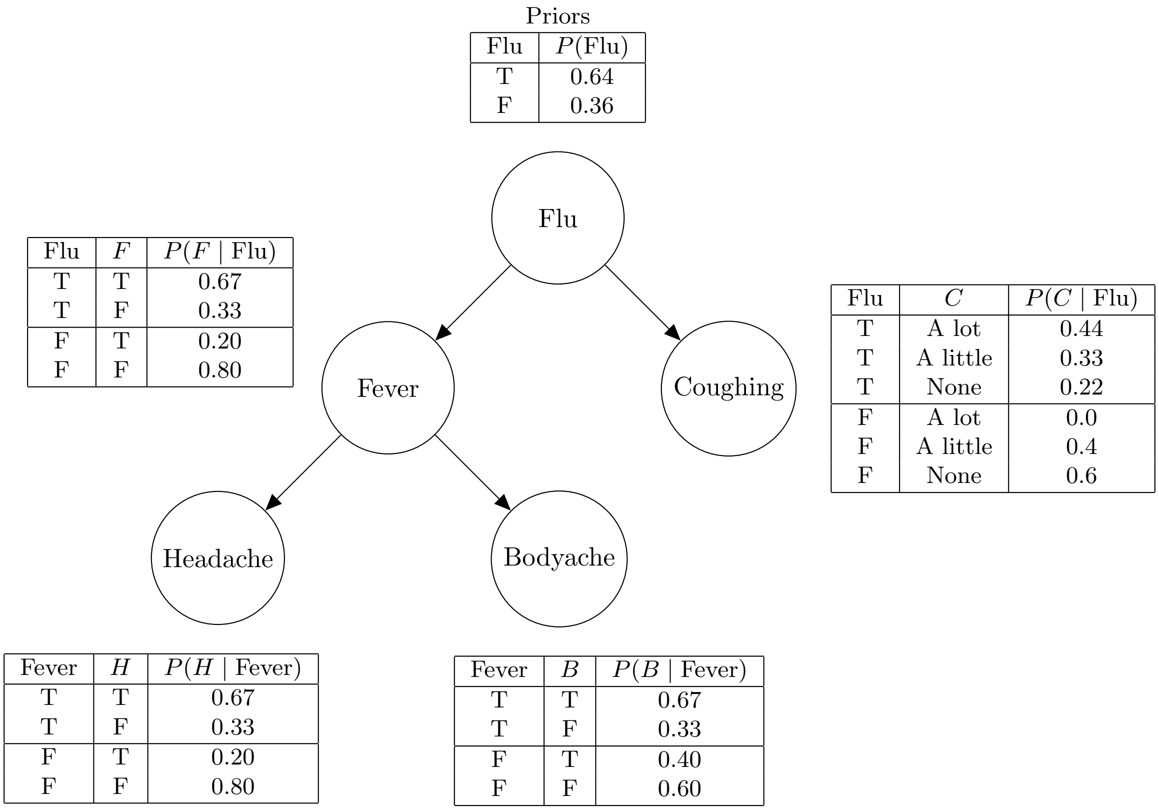
In AI, we build **probabilistic models** to reason about uncertainty. The basic reasoning unit in these models is the **random variable**, which may take on one of many values, each with some probability. For example, a die may take on one of six values (1 through 6), each with equal probability. Alternatively, a biased coin may take on one of two values (heads or tails), heads with probability p, and tails with probability $1 - p$.

Given a set of random variables, a probabilistic model expresses probabilistic relationships among them. That is, a probabilistic model represents a joint distribution. For example, the random variables in a probabilistic model may represent symptoms and diseases. A doctor can query such a model to discover the probability of a disease given symptoms observed in a patient. A doctor can also conduct counterfactual reasoning using such a model: e.g., she can query the model as if she had administered a diagnostic test to see how much, if any, uncertainty would be resolved by administering the test.

Reasoning about the relationships among random variables in a large probabilistic model can quickly become unwieldy. In many cases, however, it is not necessary to fully represent a joint distribution, because of independence relations among the random variables. For example, in the case of n coin flips, we can explicitly enumerate all 2^n outcomes, and compute probabilities for each, which we can then look up in a table to find the probability of a particular outcome (e.g., HTHHTT, assuming $n = 6$).[1] If the coin flips are independent, however, we can more compactly represent this same joint distribution using only n numbers, each corresponding to the probability p_i of heads of the ith coin, which we then multiply accordingly to query a particular outcome: e.g., $p_1(1 - p_2)p_3p_4(1 - p_5)(1 - p_6)$ for outcome HTHHTT.[2]

While independence may be a reasonable assumption for coin flips, it is far too strong of an assumption in many real-world applications. For example, when trying to diagnose a patient with flu-like symptoms, it may not be reasonable to assume that a headache is independent of a tired and achy body, or that a cough is independent of a sore throat. On the other hand, a **conditional independence** relationship may hold among some of these symptoms. That is, knowing that the patient has the flu may render the symptoms

[1] Exponential space; constant-time lookup.
[2] Linear space; linear-time lookup.



Priors

Flu	$P(\text{Flu})$
T	0.64
F	0.36

Flu	F	$P(F \mid \text{Flu})$
T	T	0.67
T	F	0.33
F	T	0.20
F	F	0.80

Flu	C	$P(C \mid \text{Flu})$
T	A lot	0.44
T	A little	0.33
T	None	0.22
F	A lot	0.0
F	A little	0.4
F	None	0.6

Fever	H	$P(H \mid \text{Fever})$
T	T	0.67
T	F	0.33
F	T	0.20
F	F	0.80

Fever	B	$P(B \mid \text{Fever})$
T	T	0.67
T	F	0.33
F	T	0.40
F	F	0.60

Figure 1: An example of a Bayesian network with conditional probability tables (CPTs).

"cough" and "sore throat" independent of each other. Similarly, knowing that the patient has a fever may render "headache" and "body ache" independent of each other. Conditional independence assumptions are more likely to hold than independence assumptions, and also lead to compact probabilistic models.

Bayesian networks (*a.k.a.* Bayes' nets) are probabilistic models that can compactly represent joint probability distributions over random variables by encoding not only independence relations, but conditional independence relations as well. An example appears in Figure 1. Whereas explicitly representing this joint distribution would require $2^4 \cdot 3$ numbers (3 is the size of the "Coughing" domain), this Bayesian network, which encodes conditional independence assumptions, requires only 20 numbers.[3]

The graphical structure of a Bayes' net, which is typically devised by domain experts as it encodes a great deal of domain knowledge, assumes causal relationships among the variables, such as "the flu may cause a fever", and "knowing that I have the flu, whether or not I have a cough is independent of the whether or not I have a fever." Conditional probability tables (CPTs) then specify the relevant probabilities, given the assumed graphical structure. Global relationships (i.e., relationships among non-neighboring nodes in the graph) can be inferred based on the specified local interactions (CPTs), using the laws of probability, together with algorithmic enhancements that improve computational efficiency.

Bayes' nets are a general tool for modeling *noisy, causal processes.* As such, they are used to model many real-world phenomena beyond medical diagnoses, such as conservation efforts, to assess the impact of envi-

[3] In fact, this Bayesian network has only 11 parameters, since $P(\text{Flu} = F)$ can be inferred from $P(\text{Flu} = T)$, for example. Likewise, explicitly representing the joint distribution requires only $2^4 \cdot 3 - 1$ parameters.

ronmental factors on ecosystems; and risk management, to assess the impact of economic indices on markets. Dynamic (or iterated) Bayesian networks are used to model—and make predictions based on—time-series, or sequential, data. Hidden Markov models (HMMs), which power many speech recognition systems, are one such example: the goal is to identify phonemes based on a series of signals (sounds). Anomaly (e.g., credit card fraud) detection is another common use case.

1 Conditional Independence

Independence is a very strong assumption. When it is raining, I may carry an umbrella and at the same time, there may be traffic on the road. These two events—me carrying my umbrella and traffic on the road—are not independent, as they have a common cause.

They are, however, **conditionally independent**, given that it is raining. Intuitively, the probability that I carry my umbrella given that it is raining does not depend on whether or not there is traffic on the road. Similarly, the probability that there is traffic on the road given that it is raining does not depend on whether or not I carry my umbrella.

Like independence, conditional independence can be understood in various ways. See Table 1.

Independence	Conditional Independence
$P(X \mid Y) = P(X)$	$P(X \mid Y, Z) = P(X \mid Z)$
$P(X, Y) = P(X)P(Y)$	$P(X, Y \mid Z) = P(X \mid Z)P(Y \mid Z)$
$P(Y \mid X) = P(Y)$	$P(Y \mid X, Z) = P(Y \mid Z)$

Table 1: Independence vs. Conditional Independence

When X is conditionally independent of Y given Z, we write $X \perp\!\!\!\perp Y \mid Z$ (or $Y \perp\!\!\!\perp X \mid Z$). In particular, Traffic $\perp\!\!\!\perp$ Umbrella | Rain. Likewise, Umbrella $\perp\!\!\!\perp$ Traffic | Rain.

2 Simplifying the Chain Rule

Recall the chain rule, which can be used to compute a joint probability.

$$\begin{aligned} P(X_1, \ldots, X_n) &= P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_2, X_1) \ldots P(X_n \mid X_{n-1}, \ldots, X_1) \\ &= \prod_{i=1}^{n} P(X_i \mid X_{i-1} \ldots X_1) \end{aligned}$$

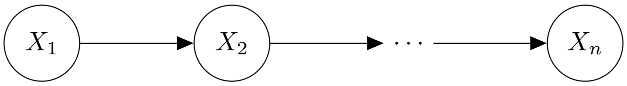


Figure 2: A linear Bayesian network. $X_3 \perp\!\!\!\perp X_1 \mid X_2$. $X_4 \perp\!\!\!\perp X_1, X_2 \mid X_3$. $X_n \perp\!\!\!\perp X_1, \ldots, X_{n-2} \mid X_{n-1}$. For example, assume $n = 3$, and let X_1 be the random variable "it is raining," X_2, the random variable "there is traffic," and X_3, the random variable "the professor is late for class." $X_3 \perp\!\!\!\perp X_1 \mid X_2$, since whether the professor is late for class depends only on whether there is traffic, and not on the root cause of the traffic.

A Bayes net encodes conditional independence assumptions that simplify this computation. For example, consider the linear Bayes net, as shown in Figure 2. Under these conditional independence assumptions,

$$\begin{aligned}\prod_{i=1}^{n} P(X_i \mid X_{i-1} \dots X_1) &= \prod_{i=1}^{n} P(X_i \mid X_{i-1}) \\ &= P(X_n \mid X_{n-1}) P(X_{n-1} \mid X_{n-2}) \dots P(X_2 \mid X_1) P(X_1)\end{aligned}$$

Without any conditional independence assumptions, the CPT $P(X_i \mid X_{i-1} \dots X_1)$ is exponential in $i-1$. Assuming Bernoulli random variables, this CPT requires $2 \cdot 2^{i-1} - 1$ probabilities. A linear Bayes net, in contrast, with n variables, requires only $O(2n)$ probabilities, 2 per variable—one to describe $P(X_i \mid 1)$ and a second to describe $P(X_i \mid 0)$, for all $1 \leq i \leq n$.

In general, the edges in a Bayes net indicate dependencies among variables, so that

$$P(X_1, \dots, X_n) = \prod_{i=1}^{n} P(X_i \mid \text{parents}(X_i))$$

A more complex Bayes net is shown in Figure 3. Your burglar alarm is ringing while you are at the office. When this happens, sometimes your neighbor John calls you, and sometimes your neighbor Mary calls you. If you receive calls from one or both of them, how likely is it that your an intruder has entered your house? Or was it just a small earthquake that caused the alarm to sound?

As per the edges in this graph, $E \perp\!\!\!\perp B$, $J \perp\!\!\!\perp E, B \mid A$, and $M \perp\!\!\!\perp J, E, B \mid A$. Therefore,

$$\begin{aligned}P(B,E,A,J,M) &= P(B)P(E \mid B)P(A \mid E,B)P(J \mid A,E,B)P(M \mid J,A,E,B) && (1) \\ &= P(B)P(E)P(A \mid E,B)P(J \mid A)P(M \mid A) && (2)\end{aligned}$$

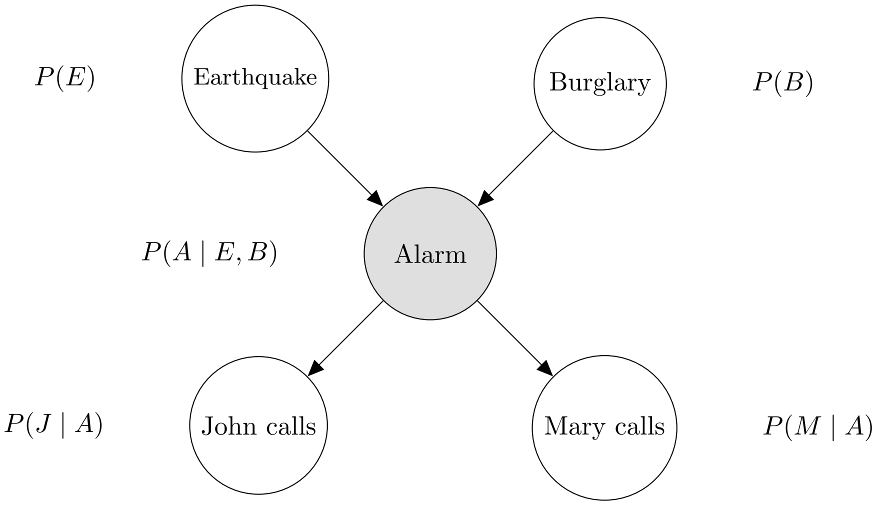


Figure 3: Textbook Example

3 Inference

It is typical to divide the random variables in a Bayes net into three categories: the **evidence** variables E, which are observed, the **query** variables Q of interest, and everything else, which are often referred to as the **latent** variables L (meaning the hidden, or unobserved, variables). We can then compute $P(Q \mid E)$ as per the definition of conditional probability, namely $P(Q, E)$ divided by $P(E)$, where $P(Q, E)$ is obtained by marginalizing over the latent variables: i.e., $P(Q, E) = \sum_L P(Q, E, L)$. For example, we can compute the probability a patient has the flu, given the symptoms observed.

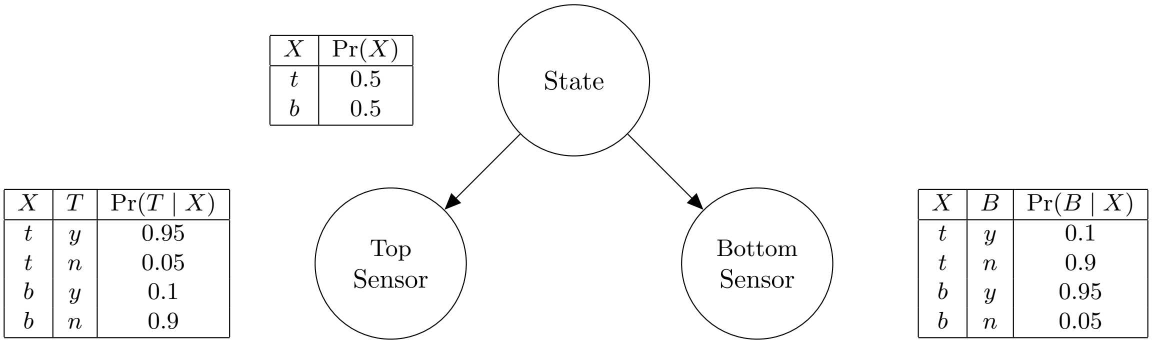
We can similarly compute $\arg\max_{q \in Q} P(Q \mid E)$ to find the most likely explanation for the evidence observed. For example, we can compute the most likely cause of a patient's symptoms, be it the flu, COVID, or just a common cold.

Next, we present an example of a Bayes net that represents a robot's state estimation problem.

Imagine a robot operating in a (very small) state space comprising only two states, top and bottom. Imagine further two sensors, one per state, which report whether the corresponding state is occupied or not. The robot can prompt these sensors for information about its location, which the sensors usually, but not always, report accurately. In particular, when the robot is in the top state, the top sensor reports this information accurately with probability 0.95. In addition, when the robot is in the *bottom* state, the *top* sensor reports this information accurately with probability only 0.9.

A Bayesian network for this scenario is depicted in Figure 4. We use X to denote the robot's state, T to denote the top sensor's output, and B to denote the bottom sensor's output. This network structure encodes the fact that the top and bottom sensors are independent of each other, given the robot's state. That is, if the robot's state is known, no further information can be gleaned from an additional sensor. We can express these conditional independence assumptions as $B \perp\!\!\!\perp T \mid X$ and $T \perp\!\!\!\perp B \mid X$. They enable us to simplify the computation of the joint probability distribution this Bayes' nets encodes, as follows:

$$\begin{aligned} P(X, T, B) &= P(X) P(T \mid X) P(B \mid T, X) && (3) \\ &= P(X) P(T \mid X) P(B \mid X) && (4) \end{aligned}$$



X	$\Pr(X)$
t	0.5
b	0.5

X	T	$\Pr(T \mid X)$
t	y	0.95
t	n	0.05
b	y	0.1
b	n	0.9

X	B	$\Pr(B \mid X)$
t	y	0.1
t	n	0.9
b	y	0.95
b	n	0.05

Figure 4: A robot's state estimation problem.

As an example, we will compute the probability that the robot is in the top state, given that both sensors report that it is in the top state, namely $P(X = t \mid T = y, B = n)$. In other words, X is our query, and the top and bottom sensors provide our evidence. (There are no latent variables in this small example.)

By definition,

$$P(X = t \mid T = y, B = n) = \frac{P(X = t, T = y, B = n)}{P(T = y, B = n)}$$

Now, as per Equation 4,

$$\begin{aligned}
P(X = t, T = y, B = n) &= P(X = t)P(T = y \mid X = t)P(B = n \mid X = t) \\
&= (0.5)(0.95)(0.9) \\
&= 0.4275
\end{aligned}$$

and

$$\begin{aligned}
P(X = b, T = y, B = n) &= P(X = b)P(T = y \mid X = b)P(B = n \mid X = b) \\
&= (0.5)(0.1)(0.05) \\
&= 0.0025
\end{aligned}$$

Therefore,

$$\begin{aligned}
P(T = y, B = n) &= \sum_{x \in \{t,b\}} P(X = x, T = y, B = n) \\
&= P(X = t, T = y, B = n) + P(X = b, T = y, B = n) \\
&= 0.4275 + 0.0025
\end{aligned}$$

Finally,

$$\begin{aligned}
P(X = t \mid T = y, B = n) &= \frac{P(X = t, T = y, B = n)}{P(T = y, B = n)} \\
&= \frac{0.4275}{0.4275 + 0.0025} \\
&\sim .99
\end{aligned}$$