CSCI 360
Introduction to Artificial Intelligence

Spring 2016
Instructor: Wei-Min Shen
TA: Thomas Collins
Probability Reasoning

• Propositional logic + probability
  – Full joint distribution (example)
  – Inferences
    • Using full joint distributions (1\textsuperscript{st} part of product rule)
    • Marginalization or summing out
    • Using Bayesian rule (2\textsuperscript{nd} part of product rule)
  – \textbf{Bayesian Networks}: Topology + CPTs (fig 14.2)
    • Compare to truth-table format (2\textsuperscript{5} rows)
    • Inferences, exact vs approximate

• Relational and FOL + probability
• Other uncertain reasoning theories
  – Dempster-Shafer, Fuzzy sets and logic,
More Concise Representation

• A full joint distribution has $2^N$ entries
  – Can we do better than this?
• Yes, use conditional independence to reduce the number of entries
Product Rule (Bayes Rule)

Product rule \( P(A \land B) = P(A|B)P(B) = P(B|A)P(A) \)

\[ \Rightarrow \text{Bayes' rule} \quad P(A|B) = \frac{P(B|A)P(A)}{P(B)} \]

Why is this useful???

For assessing diagnostic probability from causal probability:

\[ P(Cause|Effect) = \frac{P(Effect|Cause)P(Cause)}{P(Effect)} \]

E.g., let \( M \) be meningitis, \( S \) be stiff neck:

\[ P(M|S) = \frac{P(S|M)P(M)}{P(S)} = \frac{0.8 \times 0.0001}{0.1} = 0.0008 \]

Note: posterior probability of meningitis still very small!
Independence

Two random variables \( A \) and \( B \) are (absolutely) independent iff
\[
P(A|B) = P(A)
\]
or
\[
P(A, B) = P(A|B)P(B) = P(A)P(B)
\]
e.g., \( A \) and \( B \) are two coin tosses

If \( n \) Boolean variables are independent, the full joint is
\[
P(X_1, \ldots, X_n) = \Pi_i P(X_i)
\]
hence can be specified by just \( n \) numbers

Absolute independence is a very strong requirement, seldom met
Conditional Independence

Consider the dentist problem with three random variables: 
Toothache, Cavity, Catch (steel probe catches in my tooth)

The full joint distribution has $2^3 - 1 = 7$ independent entries

If I have a cavity, the probability that the probe catches in it doesn’t depend on whether I have a toothache:

1. $P(\text{Catch}|\text{Toothache}, \text{Cavity}) = P(\text{Catch}|\text{Cavity})$

i.e., Catch is conditionally independent of Toothache given Cavity

The same independence holds if I haven’t got a cavity:

2. $P(\text{Catch}|\text{Toothache}, \neg\text{Cavity}) = P(\text{Catch}|\neg\text{Cavity})$
Conditional Independence

Equivalent statements to (1)

(1a) \( P(\text{Toothache}|\text{Catch, Cavity}) = P(\text{Toothache}|\text{Cavity}) \) Why??

(1b) \( P(\text{Toothache, Catch}|\text{Cavity}) = P(\text{Toothache}|\text{Cavity})P(\text{Catch}|\text{Cavity}) \) Why??

Full joint distribution can now be written as

\[
P(\text{Toothache, Catch, Cavity}) = P(\text{Toothache, Catch}|\text{Cavity})P(\text{Cavity})
= P(\text{Toothache}|\text{Cavity})P(\text{Catch}|\text{Cavity})P(\text{Cavity})
\]

i.e., \( 2 + 2 + 1 = 5 \) independent numbers (equations 1 and 2 remove 2)
Conditional Independence

Equivalent statements to (1)

(1a) \( P(\text{Toothache}|\text{Catch, Cavity}) = P(\text{Toothache}|\text{Cavity}) \)

\[
P(\text{Toothache}|\text{Catch, Cavity})
= P(\text{Catch}|\text{Toothache, Cavity})P(\text{Toothache}|\text{Cavity})/P(\text{Catch}|\text{Cavity})
= P(\text{Catch}|\text{Cavity})P(\text{Toothache}|\text{Cavity})/P(\text{Catch}|\text{Cavity}) \quad \text{(from 1)}
= P(\text{Toothache}|\text{Cavity})
\]

(1b) \( P(\text{Toothache, Catch}|\text{Cavity}) = P(\text{Toothache}|\text{Cavity})P(\text{Catch}|\text{Cavity}) \)

\[
P(\text{Toothache, Catch}|\text{Cavity})
= P(\text{Toothache}|\text{Catch, Cavity})P(\text{Catch}|\text{Cavity}) \quad \text{(product rule)}
= P(\text{Toothache}|\text{Cavity})P(\text{Catch}|\text{Cavity}) \quad \text{(from 1a)}
\]
Bayesian Networks

A simple, graphical notation for conditional independence assertions and hence for compact specification of full joint distributions

Syntax:
- a set of nodes, one per variable
- a directed, acyclic graph (link ≈ “directly influences”)
- a conditional distribution for each node given its parents:
  \[ P(X_i | \text{Parents}(X_i)) \]

In the simplest case, conditional distribution represented as a conditional probability table (CPT)
Example

I’m at work, neighbor John calls to say my alarm is ringing, but neighbor Mary doesn’t call. Sometimes it’s set off by minor earthquakes. Is there a burglar?

Variables: Burglar, Earthquake, Alarm, JohnCalls, MaryCalls

Network topology reflects “causal” knowledge:

Conditional Probability Table (CPT)

Note: \( \leq k \) parents \( \Rightarrow O(d^k n) \) numbers vs. \( O(d^n) \)
Semantics

“Global” semantics defines the full joint distribution as the product of the local conditional distributions:

\[ P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i|Parents(X_i)) \]

e.g., \( P(J \land M \land A \land \neg B \land \neg E) \) is given by??

= 

“Global” semantics defines the full joint distribution as the product of the local conditional distributions:

$$\mathbf{P}(X_1, \ldots, X_n) = \prod_{i=1}^{n} \mathbf{P}(X_i | \text{Parents}(X_i))$$

e.g., $P(J \land M \land A \land \neg B \land \neg E)$ is given by

$$= P(\neg B)P(\neg E)P(A | \neg B \land \neg E)P(J | A)P(M | A)$$

“Local” semantics: each node is conditionally independent of its nondescendants given its parents

Theorem: Local semantics $\Leftrightarrow$ global semantics
Markov Blanket

Each node is conditionally independent of all others given its Markov blanket: parents + children + children’s parents
Constructing Bayesian Networks

Need a method such that a series of locally testable assertions of conditional independence guarantees the required global semantics:

1. Choose an ordering of variables $X_1, \ldots, X_n$
2. For $i = 1$ to $n$
   - add $X_i$ to the network
   - select parents from $X_1, \ldots, X_{i-1}$ such that
     \[ P(X_i|\text{Parents}(X_i)) = P(X_i|X_1, \ldots, X_{i-1}) \]

This choice of parents guarantees the global semantics:
\[ P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i|X_1, \ldots, X_{i-1}) \] (chain rule)
\[ = \prod_{i=1}^{n} P(X_i|\text{Parents}(X_i)) \] by construction
Example: car diagnosis

Initial evidence: engine won’t start
Testable variables (thin ovals), diagnosis variables (thick ovals)
Hidden variables (shaded) ensure sparse structure, reduce parameters
Example: car insurance

Predict claim costs (medical, liability, property) given data on application form (other unshaded nodes)
Conditional Probability Distribution
(Compactness)

CPT grows exponentially with no. of parents
CPT becomes infinite with continuous-valued parent or child

Solution: canonical distributions that are defined compactly

Deterministic nodes are the simplest case:
\[ X = f(\text{Parents}(X)) \] for some function \( f \)

E.g., Boolean functions
\[ \text{NorthAmerican} \iff \text{Canadian} \lor US \lor \text{Mexican} \]

E.g., numerical relationships among continuous variables
\[ \frac{\partial \text{Level}}{\partial t} = \text{inflow} + \text{precipitation} - \text{outflow} - \text{evaporation} \]
Compact conditional distributions

Noisy-OR distributions model multiple noninteracting causes

1) Parents $U_1 \ldots U_k$ include all causes (can add leak node)
2) Independent failure probability $q_i$ for each cause alone

$$\Rightarrow P(X | U_1 \ldots U_j, \neg U_{j+1} \ldots \neg U_k) = 1 - \prod_{i=1}^{j} q_i$$

<table>
<thead>
<tr>
<th>Cold</th>
<th>Flu</th>
<th>Malaria</th>
<th>$P(\text{Fever})$</th>
<th>$P(\neg \text{Fever})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td>0.0</td>
<td>1.0</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>0.9</td>
<td>0.1</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td>0.8</td>
<td>0.2</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>0.98</td>
<td>0.02 = 0.2 $\times$ 0.1</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>0.4</td>
<td>0.6</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>0.94</td>
<td>0.06 = 0.6 $\times$ 0.1</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>0.88</td>
<td>0.12 = 0.6 $\times$ 0.2</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>0.988</td>
<td>0.012 = 0.6 $\times$ 0.2 $\times$ 0.1</td>
</tr>
</tbody>
</table>

Number of parameters linear in number of parents
Hybrid (Discrete+Continuous) Network

Discrete (*Subsidy?* and *Buys?*); continuous (*Harvest* and *Cost*)

Option 1: discretization—possibly large errors, large CPTs

Option 2: finitely parameterized canonical families

1) Continuous variable, discrete+continuous parents (e.g., *Cost*)
2) Discrete variable, continuous parents (e.g., *Buys?*)
Continuous Child Variables

Need one conditional density function for child variable given continuous parents, for each possible assignment to discrete parents.

Most common is the linear Gaussian model, e.g.,:

$$P(Cost = c | Harvest = h, Subsidy? = true)$$

$$= N(a_t h + b_t, \sigma_t)(c)$$

$$= \frac{1}{\sigma_t \sqrt{2\pi}} \exp \left( -\frac{1}{2} \left( \frac{c - (a_t h + b_t)}{\sigma_t} \right)^2 \right)$$

Mean $Cost$ varies linearly with $Harvest$, variance is fixed.
Linear variation is unreasonable over the full range but works OK if the likely range of $Harvest$ is narrow.
Continuous child variables

All-continuous network with LG distributions
⇒ full joint is a multivariate Gaussian

Discrete + continuous LG network is a conditional Gaussian network i.e., a multivariate Gaussian over all continuous variables for each combination of discrete variable values
Discrete variable w/ continuous parents

Probability of \( Buys? \) given \( Cost \) should be a “soft” threshold:

![Graph showing the relationship between probability and cost]

Probit distribution uses integral of Gaussian:

\[
\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2} dx
\]

\[
P(Buys? = true \mid Cost = c) = \Phi((-c + \mu)/\sigma)
\]

Can view as hard threshold whose location is subject to noise.
Discrete Variables

Sigmoid (or logit) distribution also used in neural networks:

\[
P(Buys? = true \mid Cost = c) = \frac{1}{1 + exp\left(-2\frac{-c+\mu}{\sigma}\right)}
\]

Sigmoid has similar shape to probit but much longer tails:
Inference in Belief Networks

• Exact inference by the network
• Exact inference by enumeration
• Exact inference by variable elimination
• Approximate inference by stochastic simulation
• Approximate inference by Markov chain Monte Carlo (MCMC)
What is the probability that the alarm has sounded but neither a burglary nor an earthquake has occurred and both John and Mary call?

\[ P(j \land m \land a \land \neg b \land \neg e) \]

**Example: Full Joint Distribution**
\[ P(j \land m \land a \land \neg b \land \neg e) = P(j | a) P(m | a) P(a | \neg b \land \neg e) P(\neg b) P(\neg e) \]
\[ = 0.90 \times 0.70 \times 0.001 \times 0.999 \times 0.998 \]
\[ = 0.00062 \]
We find that storms can also set off alarms. We add that into our CPT. Notice that JohnCalls and MaryCalls stay the same since Storms were always there but were just unaccounted for. John and Mary did not change! However, we have better precision at P(A).

What if we find a new variable?
What if we inject a new cause that was not there before. We pay a crazy guy to set off the alarm frequently, JohnCalls and MaryCalls may no longer be valid since we may have changed the behaviors. For instance the alarm goes off so often now that John and Mary are more likely to ignore it.

What if we cause a new variable? (or a new variable just occurs)
If the introduced variable is highly erratic, it can invalidate even more of the CPT than we would like.

What if we cause a new variable?
What if we *cause* a new variable?

However, some changes to the CPT may be absurd so we may never have to worry about them.
Things in the Model can Change

• We can account for change in the model over time (a more advanced topic)
  • John and Mary may be more or less likely to call at certain times. Cyclical repetition may be not too difficult to model
  • People may become tired of their job and be less likely call over longer periods

• This may be easy or difficult to model.
  • Crime picks up. If the trend is slow enough, the model may be able to adjust online even if we have never observed crime picking up before. However, this may easily and totally throw our model off

• However, keep in mind, we may get good enough results for our model even without accounting for changes over time
How to Apply this to Robotics?

The CPT can describe other events and probabilities such as action success given observations.
Exact Inference in BN

What if we want to make an inference such as: what is the probability of a tree in the path given that the robot has stopped and turned. This might be useful to a robot which can judge if there is a tree in a path based on the behavior of another robot. So if robot A sees robot B turn or stop it might infer that there is a tree in the path.

\[
P(t|s,u) = \alpha P(t) \sum_r P(r) \sum_o P(o|t,r)P(s|o)P(u|o)
\]

Normalized we get:

\[
P(T|s,u) = \left\langle \frac{P(t|s,u)}{P(t|s,u)+P(-t|s,u)} , \frac{P(-t|s,u)}{P(t|s,u)+P(-t|s,u)} \right\rangle
\]
We compute $P$ of tree and $P$ of not tree and normalize. This is essentially an enumeration of all situations for tree and not tree.
\[ P(t|s,u) = 0.051975 + 0.00315 + 0.002025 + 0.00285 \]

\[ P(\neg t|s,u) = 0.0002835 + 0.05481 + 0.0134865 + 0.00639 \]

\[
P(T|s,u) = \left\langle \frac{P(t|s,u)}{P(t|s,u) + P(\neg t|s,u)}, \frac{P(\neg t|s,u)}{P(t|s,u) + P(\neg t|s,u)} \right\rangle
\]

\[
= \left\langle \frac{0.06}{0.06 + 0.07497}, \frac{0.07497}{0.06 + 0.07497} \right\rangle
\]

\[
= \langle 0.4445, 0.5555 \rangle
\]

The P of a tree in the path is 0.4445
What about hidden variables?

Initial evidence: engine won’t start
Testable variables (thin ovals), diagnosis variables (thick ovals)
Hidden variables (shaded) ensure sparse structure, reduce parameters
Example: car diagnosis

Initial evidence: engine won’t start
Testable variables (thin ovals), diagnosis variables (thick ovals)
Hidden variables (shaded) ensure sparse structure, reduce parameters