Tutorial on Bayesian Networks

Jack Breese
Microsoft Research
breese@microsoft.com

Daphne Koller
Stanford University
koller@cs.stanford.edu

First given as a AAAI’97 tutorial.
Probabilities

- Probability distribution $P(X|\xi)$
  - $X$ is a random variable
    - Discrete
    - Continuous
  - $\xi$ is background state of information
Discrete Random Variables

- Finite set of possible outcomes
  \[ X \in \{x_1, x_2, x_3, \ldots, x_n\} \]

  \[ P(x_i) \geq 0 \]

  \[ \sum_{i=1}^{n} P(x_i) = 1 \]

  \( X \) binary: \( P(x) + P(\overline{x}) = 1 \)
Continuous Random Variable

- Probability distribution (density function) over continuous values

\[ X \in [0,10] \quad P(x) \geq 0 \]

\[
\int_{0}^{10} P(x) \, dx = 1
\]

\[ P(5 \leq x \leq 7) = \int_{5}^{7} P(x) \, dx \]
Bayesian networks

- Basics
  - Structured representation
  - Conditional independence
  - Naïve Bayes model
  - Independence facts
Bayesian Networks

\[ S \in \{\text{no, light, heavy}\} \quad \text{Smoking} \rightarrow \quad \text{Cancer} \]

\[ C \in \{\text{none, benign, malignant}\} \]
Bayesian Networks

\[ S \in \{\text{no, light, heavy}\} \]

\[
\begin{array}{|c|c|}
\hline
P(S=\text{no}) & 0.80 \\
P(S=\text{light}) & 0.15 \\
P(S=\text{heavy}) & 0.05 \\
\hline
\end{array}
\]

\[ C \in \{\text{none, benign, malignant}\} \]
Bayesian Networks

\[ S \in \{no, \text{light}, \text{heavy}\} \]

\[
\begin{array}{c|c}
P(S=\text{no}) & 0.80 \\
P(S=\text{light}) & 0.15 \\
P(S=\text{heavy}) & 0.05 \\
\end{array}
\]

\[ C \in \{\text{none, benign, malignant}\} \]

\[
\begin{array}{|c|c|c|c|}
\hline
\text{Smoking} & \text{no} & \text{light} & \text{heavy} \\
\hline
P(C=\text{none}) & 0.96 & 0.88 & 0.60 \\
P(C=\text{benign}) & 0.03 & 0.08 & 0.25 \\
P(C=\text{malign}) & 0.01 & 0.04 & 0.15 \\
\hline
\end{array}
\]
Product Rule

\[ P(C, S) = P(C|S) \cdot P(S) \]

<table>
<thead>
<tr>
<th>S (\downarrow)</th>
<th>C (\Rightarrow)</th>
<th>none</th>
<th>benign</th>
<th>malignant</th>
</tr>
</thead>
<tbody>
<tr>
<td>no</td>
<td></td>
<td>0.768</td>
<td>0.024</td>
<td>0.008</td>
</tr>
<tr>
<td>light</td>
<td></td>
<td>0.132</td>
<td>0.012</td>
<td>0.006</td>
</tr>
<tr>
<td>heavy</td>
<td></td>
<td>0.035</td>
<td>0.010</td>
<td>0.005</td>
</tr>
</tbody>
</table>
## Marginalization

<table>
<thead>
<tr>
<th>S↓</th>
<th>C⇒</th>
<th>none</th>
<th>benign</th>
<th>malig</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>no</td>
<td>none</td>
<td>0.768</td>
<td>0.024</td>
<td>0.008</td>
<td>0.80</td>
</tr>
<tr>
<td>light</td>
<td>none</td>
<td>0.132</td>
<td>0.012</td>
<td>0.006</td>
<td>0.15</td>
</tr>
<tr>
<td>heavy</td>
<td>none</td>
<td>0.035</td>
<td>0.010</td>
<td>0.005</td>
<td>0.05</td>
</tr>
<tr>
<td>total</td>
<td>none</td>
<td>0.935</td>
<td>0.046</td>
<td>0.019</td>
<td></td>
</tr>
</tbody>
</table>

\[ P(\text{Smoke}) \]

\[ P(\text{Cancer}) \]

© Jack Breese (Microsoft) & Daphne Koller (Stanford)
Bayes Rule Revisited

\[ P(S \mid C) = \frac{P(C \mid S)P(S)}{P(C)} = \frac{P(C, S)}{P(C)} \]
Bayes Rule Revisited

\[ P(S \mid C) = \frac{P(C \mid S)P(S)}{P(C)} = \frac{P(C, S)}{P(C)} \]

<table>
<thead>
<tr>
<th>$S \downarrow$ C ⇒</th>
<th>none</th>
<th>benign</th>
<th>malign</th>
</tr>
</thead>
<tbody>
<tr>
<td>no</td>
<td>0.768/0.935</td>
<td>0.024/0.046</td>
<td>0.008/0.019</td>
</tr>
<tr>
<td>light</td>
<td>0.132/0.935</td>
<td>0.012/0.046</td>
<td>0.006/0.019</td>
</tr>
<tr>
<td>heavy</td>
<td>0.030/0.935</td>
<td>0.015/0.046</td>
<td>0.005/0.019</td>
</tr>
</tbody>
</table>
Bayes Rule Revisited

\[ P(S | C) = \frac{P(C | S)P(S)}{P(C)} = \frac{P(C, S)}{P(C)} \]

<table>
<thead>
<tr>
<th>(S \downarrow)</th>
<th>(C \Rightarrow)</th>
<th>none</th>
<th>benign</th>
<th>malign</th>
</tr>
</thead>
<tbody>
<tr>
<td>no</td>
<td></td>
<td>0.768/.935</td>
<td>0.024/.046</td>
<td>0.008/.019</td>
</tr>
<tr>
<td>light</td>
<td></td>
<td>0.132/.935</td>
<td>0.012/.046</td>
<td>0.006/.019</td>
</tr>
<tr>
<td>heavy</td>
<td></td>
<td>0.030/.935</td>
<td>0.015/.046</td>
<td>0.005/.019</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(\text{Cancer}=)</th>
<th>none</th>
<th>benign</th>
<th>malignant</th>
</tr>
</thead>
<tbody>
<tr>
<td>(P(S=\text{no}))</td>
<td>0.821</td>
<td>0.522</td>
<td>0.421</td>
</tr>
<tr>
<td>(P(S=\text{light}))</td>
<td>0.141</td>
<td>0.261</td>
<td>0.316</td>
</tr>
<tr>
<td>(P(S=\text{heavy}))</td>
<td>0.037</td>
<td>0.217</td>
<td>0.263</td>
</tr>
</tbody>
</table>
A Bayesian Network

Age → Exposure to Toxics → Cancer
Gender → Smoking → Cancer

Cancer → Serum Calcium
Cancer → Lung Tumor
Independence

Age and Gender are independent.
Independence

Age and Gender are independent.

\[ P(A,G) = P(G)P(A) \]
Independence

Age and Gender are independent.

\[ P(A, G) = P(G)P(A) \]
\[ P(A|G) = P(A) \quad A \perp G \]
\[ P(G|A) = P(G) \quad G \perp A \]
Independence

*Age* and *Gender* are independent.

\[
P(A, G) = P(G)P(A)
\]

\[
P(A|G) = P(A) \quad A \perp G
\]

\[
P(G|A) = P(G) \quad G \perp A
\]

\[
P(A, G) = P(G|A)P(A) = P(G)P(A)
\]

\[
P(A, G) = P(A|G)P(G) = P(A)P(G)
\]
Conditional Independence

Cancer is independent of Age and Gender given Smoking.
Conditional Independence

Cancer is independent of Age and Gender given Smoking.

\[ P(C|A,G,S) = P(C|S) \quad C \perp A, G \mid S \]
More Conditional Independence: Naïve Bayes

Serum Calcium and Lung Tumor are dependent
More Conditional Independence: Naïve Bayes

*Serum Calcium and Lung Tumor are dependent*

*Serum Calcium is independent of Lung Tumor, given Cancer*

\[ P(L|SC,C) = P(L|C) \]
Naïve Bayes in general

2n + 1 parameters:

\[ P(h) \]
\[ P(e_i \mid h), P(e_i \mid \bar{h}), \quad i = 1, \ldots, n \]
More Conditional Independence: Explaining Away

Exposure to Toxics and Smoking are independent

\[ E \perp S \]
More Conditional Independence: Explaining Away

Exposure to Toxics and Smoking are independent

\[ E \perp S \]

Exposure to Toxics is dependent on Smoking, given Cancer
More Conditional Independence: Explaining Away

Exposure to Toxics and Smoking are independent

\[ E \perp S \]

Exposure to Toxics is dependent on Smoking, given Cancer

\[
P(E = \text{heavy} \mid C = \text{malignant}) >
\]

\[
P(E = \text{heavy} \mid C = \text{malignant}, S=\text{heavy})
\]
Put it all together

\[ P(A,G,E,S,C,L,SC) = P(A) \cdot P(G) \cdot P(E | A) \cdot P(S | A,G) \cdot P(C | E,S) \cdot P(SC | C) \cdot P(L | C) \]
General Product (Chain) Rule for Bayesian Networks

\[ P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid Pa_i) \]

\[ Pa_i = \text{parents}(X_i) \]
Conditional Independence

A variable (node) is conditionally independent of its non-descendants given its parents.

\[ \text{Age} \rightarrow \text{Gender} \rightarrow \text{Cancer} \rightarrow \{\text{Serum Calcium, Lung Tumor}\} \]

\[ \text{Exposure to Toxics} \rightarrow \text{Smoking} \rightarrow \text{Cancer} \rightarrow \{\text{Age, Gender}\} \]

\[ \text{Cancer is independent of Age and Gender given Exposure to Toxics and Smoking.} \]
Another non-descendant

Cancer is independent of Diet given Exposure to Toxics and Smoking.
Independence and Graph Separation

- Given a set of observations, is one set of variables dependent on another set?
- Observing effects can induce dependencies.
- d-separation (Pearl 1988) allows us to check conditional independence graphically.
CPCS Network
Structuring
Structuring
Structuring
Structuring

- Exposure to Toxic
- Smoking
- Cancer
- Genetic Damage
- Lung Tumor
Structuring
Structuring

Network structure corresponding to “causality” is usually good.
Structuring

Network structure corresponding to “causality” is usually good.

Extending the conversation.
Local Structure
Local Structure

- Causal independence: from $2^n$ to $n+1$ parameters
Local Structure

- Causal independence: from $2^n$ to $n+1$ parameters
- Asymmetric assessment: similar savings in practice.
Local Structure

- Causal independence: from $2^r$ to $n+1$ parameters
- Asymmetric assessment: similar savings in practice.
- Typical savings (#params):
Local Structure

- Causal independence: from 2" to \( n+1 \) parameters
- Asymmetric assessment: similar savings in practice.
- Typical savings (#params):
  - 145 to 55 for a small hardware network;
Local Structure

- Causal independence: from 2" to \( n+1 \) parameters
- Asymmetric assessment: similar savings in practice.
- Typical savings (\#params):
  - 145 to 55 for a small hardware network;
  - 133,931,430 to 8254 for CPCS !!
Course Contents

- Concepts in Probability
- Bayesian Networks
  - Inference
- Decision making
- Learning networks from data
- Reasoning over time
- Applications
Inference

- Patterns of reasoning
- Basic inference
- Exact inference
- Exploiting structure
- Approximate inference
Predictive Inference

How likely are elderly males to get malignant cancer?

\[ P(C=\text{malignant} \mid \text{Age} \geq 60, \text{Gender} = \text{male}) \]
Combined

How likely is an elderly male patient with high Serum Calcium to have malignant cancer?

$$P(\text{C=malignant} \mid \text{Age}>60, \text{Gender= male, Serum Calcium = high})$$
Explaining away

- If we see a lung tumor, the probability of heavy smoking and of exposure to toxics both go up.
Explaining away

- If we see a lung tumor, the probability of heavy smoking and of exposure to toxics both go up.
- If we then observe heavy smoking, the probability of exposure to toxics goes back down.
Inference in Belief Networks

- Find $P(Q=q | E = e)$
  - $Q$ the query variable
  - $E$ set of evidence variables

\[
P(q | e) = \frac{P(q, e)}{P(e)}
\]

$X_1, \ldots, X_n$ are network variables except $Q, E$

\[
P(q, e) = \sum_{x_1, \ldots, x_n} P(q, e, x_1, \ldots, x_n)
\]
Basic Inference

\[ P(c) = ? \]
Basic Inference

\[ P(c) = ? \]

- \[ P(C,S) = P(C|S) \cdot P(S) \]
Basic Inference

$P(c) = ?$

$P(C,S) = P(C|S)P(S)$

<table>
<thead>
<tr>
<th>Smoking</th>
<th>no</th>
<th>light</th>
<th>heavy</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P(C=none)$</td>
<td>0.96</td>
<td>0.88</td>
<td>0.60</td>
</tr>
<tr>
<td>$P(C=benign)$</td>
<td>0.03</td>
<td>0.08</td>
<td>0.25</td>
</tr>
<tr>
<td>$P(C=malig)$</td>
<td>0.01</td>
<td>0.04</td>
<td>0.15</td>
</tr>
</tbody>
</table>
Basic Inference

\[ P(c) = ? \]

\[ P(C,S) = P(C|S) \ P(S) \]

<table>
<thead>
<tr>
<th>Smoking</th>
<th>no</th>
<th>light</th>
<th>heavy</th>
</tr>
</thead>
<tbody>
<tr>
<td>P(C=none)</td>
<td>0.96</td>
<td>0.88</td>
<td>0.60</td>
</tr>
<tr>
<td>P(C=benign)</td>
<td>0.03</td>
<td>0.08</td>
<td>0.25</td>
</tr>
<tr>
<td>P(C=malig)</td>
<td>0.01</td>
<td>0.04</td>
<td>0.15</td>
</tr>
</tbody>
</table>

\begin{align*}
P(S=\text{no}) & = 0.80 \\
P(S=\text{light}) & = 0.15 \\
P(S=\text{heavy}) & = 0.05
\end{align*}
Basic Inference

\[ P(c) = ? \]

\[ P(C, S) = P(C | S) P(S) \]

<table>
<thead>
<tr>
<th>Smoking</th>
<th>no</th>
<th>light</th>
<th>heavy</th>
</tr>
</thead>
<tbody>
<tr>
<td>P(C=none)</td>
<td>0.96</td>
<td>0.88</td>
<td>0.60</td>
</tr>
<tr>
<td>P(C=benign)</td>
<td>0.03</td>
<td>0.08</td>
<td>0.25</td>
</tr>
<tr>
<td>P(C=malign)</td>
<td>0.01</td>
<td>0.04</td>
<td>0.15</td>
</tr>
</tbody>
</table>

| P(S=no) | 0.80 |
| P(S=light) | 0.15 |
| P(S=heavy) | 0.05 |

<table>
<thead>
<tr>
<th>S(\downarrow)</th>
<th>C(\Rightarrow)</th>
<th>none</th>
<th>benign</th>
<th>malign</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>no</td>
<td>0.768</td>
<td>0.024</td>
<td>0.008</td>
<td>0.80</td>
<td></td>
</tr>
<tr>
<td>light</td>
<td>0.132</td>
<td>0.012</td>
<td>0.006</td>
<td>0.15</td>
<td></td>
</tr>
<tr>
<td>heavy</td>
<td>0.035</td>
<td>0.010</td>
<td>0.005</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td>total</td>
<td>0.935</td>
<td>0.046</td>
<td>0.019</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\(P(Cancer)\)
Basic Inference
Basic Inference

A → B → C
Basic Inference

\[ P(b) = \sum_a P(a, b) = \sum_a P(b \mid a) P(a) \]
Basic Inference

\[ P(b) = \sum_a P(a, b) = \sum_a P(b \mid a) P(a) \]

\[ P(c) = \sum_b P(c \mid b) P(b) \]
Basic Inference

\[ P(b) = \sum_a P(a, b) = \sum_a P(b \mid a) P(a) \]

\[ P(c) = \sum_b P(c \mid b) P(b) \]
Basic Inference

\[ P(b) = \sum_{a} P(a, b) = \sum_{a} P(b \mid a) P(a) \]

\[ P(c) = \sum_{b} P(c \mid b) P(b) \]

\[ P(c) = \sum_{b,a} P(a, b, c) \]
Basic Inference

\[ P(b) = \sum_a P(a, b) = \sum_a P(b \mid a) \ P(a) \]

\[ P(c) = \sum_b P(c \mid b) \ P(b) \]

\[ P(c) = \sum_{b,a} P(a, b, c) = \sum_{b,a} P(c \mid b) \ P(b \mid a) \ P(a) \]
Basic Inference

\[ P(b) = \sum_a P(a, b) = \sum_a P(b \mid a) \, P(a) \]

\[ P(c) = \sum_b P(c \mid b) \, P(b) \]

\[ P(c) = \sum_{b,a} P(a, b, c) = \sum_{b,a} P(c \mid b) \, P(b \mid a) \, P(a) \]

\[ = \sum_b P(c \mid b) \sum_a P(b \mid a) \, P(a) \]
Basic Inference

\[ P(b) = \sum_a P(a, b) = \sum_a P(b \mid a) P(a) \]

\[ P(c) = \sum_b P(c \mid b) P(b) \]

\[ P(c) = \sum_{b,a} P(a, b, c) = \sum_{b,a} P(c \mid b) P(b \mid a) P(a) \]

\[ = \sum_b P(c \mid b) \sum_a P(b \mid a) P(a) \]

\[ P(b) \]
Inference in trees

\[
P(x) = \sum_{y_1, y_2} P(x \mid y_1, y_2) \ P(y_1, y_2)
\]
Inference in trees

\[ P(x) = \sum_{y_1, y_2} P(x \mid y_1, y_2) \cdot P(y_1, y_2) \]

because of independence of \( Y_1, Y_2 \):

\[ = \sum_{y_1, y_2} P(x \mid y_1, y_2) \cdot P(y_1) \cdot P(y_2) \]
Polytrees

A network is *singly connected* (a polytree) if it contains no undirected loops.

**Theorem:** Inference in a singly connected network can be done in linear time*.

Main idea: in variable elimination, need only maintain distributions over single nodes.

* in network size including table sizes.
The problem with loops

The grass is dry only if no rain and no sprinklers.

\[ P(\overline{g}) = P(\overline{r}, \overline{s}) \sim 0 \]
The problem with loops contd.

\[ P(\bar{g}) = \]
The problem with loops contd.

\[
P(\overline{g}) = P(\overline{g} \mid r, s) \ P(r, s) + P(\overline{g} \mid r, \overline{s}) \ P(r, \overline{s}) \\
+ P(\overline{g} \mid \overline{r}, s) \ P(\overline{r}, s) + P(\overline{g} \mid \overline{r}, \overline{s}) \ P(\overline{r}, \overline{s})
\]
The problem with loops contd.

\[ P(\bar{g}) = \underbrace{P(\bar{g} \mid r, s)}_{0} P(r, s) + \underbrace{P(\bar{g} \mid r, \bar{s})}_{0} P(r, \bar{s}) \]
\[ + \underbrace{P(\bar{g} \mid \bar{r}, s)}_{0} P(\bar{r}, s) + \underbrace{P(\bar{g} \mid \bar{r}, \bar{s})}_{1} P(\bar{r}, \bar{s}) \]
The problem with loops contd.

\[
P(\bar{g}) = \begin{cases} 
0 & P(\bar{g} \mid r, s) P(r, s) + 0 \\
0 & P(\bar{g} \mid r, \bar{s}) P(r, \bar{s}) \\
& + P(\bar{g} \mid \bar{r}, s) P(\bar{r}, s) + P(\bar{g} \mid \bar{r}, \bar{s}) P(\bar{r}, \bar{s}) \\
& + 0 & P(\bar{g} \mid \bar{r}, \bar{s}) P(\bar{r}, \bar{s}) \\
& + 1 & P(\bar{g} \mid \bar{r}, \bar{s}) P(\bar{r}, \bar{s}) \\
\end{cases}
\]

\[= P(\bar{r}, \bar{s}) \]
The problem with loops contd.

\[
\begin{align*}
P(\bar{g}) &= \underbrace{P(\bar{g} \mid r, s)}_{0} P(r, s) + \underbrace{P(\bar{g} \mid r, \bar{s})}_{0} P(r, \bar{s}) \\
& \quad + \underbrace{P(\bar{g} \mid \bar{r}, s)}_{0} P(\bar{r}, s) + \underbrace{P(\bar{g} \mid \bar{r}, \bar{s})}_{1} P(\bar{r}, \bar{s}) \\
&= P(\bar{r}, \bar{s}) \\
&= P(\bar{r}) P(\bar{s}) \sim 0.5 \cdot 0.5 = 0.25
\end{align*}
\]
The problem with loops contd.

\[
P(\overline{g}) = \underbrace{P(\overline{g} \mid r, s) P(r, s)}_{0} + \underbrace{P(\overline{g} \mid r, \overline{s}) P(r, \overline{s})}_{0} + \underbrace{P(\overline{g} \mid \overline{r}, s) P(\overline{r}, s)}_{0} + \underbrace{P(\overline{g} \mid \overline{r}, \overline{s}) P(\overline{r}, \overline{s})}_{1}
\]

\[= P(\overline{r}, \overline{s}) \sim 0 \]

\[= P(\overline{r}) P(\overline{s}) \sim 0.5 \cdot 0.5 = 0.25\]
The problem with loops contd.

\[
P(\overline{g}) = \frac{0}{P(\overline{g} \mid r, s) P(r, s)} + \frac{0}{P(\overline{g} \mid r, \overline{s}) P(r, \overline{s})} \\
+ \frac{0}{P(\overline{g} \mid \overline{r}, s) P(\overline{r}, s)} + \frac{1}{P(\overline{g} \mid \overline{r}, \overline{s}) P(\overline{r}, \overline{s})}
\]

\[= P(\overline{r}, \overline{s}) \sim 0 \]

\[\neq P(\overline{r}) P(\overline{s}) \sim 0.5 \cdot 0.5 = 0.25 \]

problem
Variable elimination

\[ P(c) = \sum_b P(c \mid b) \sum_a P(b \mid a) P(a) \]

\[ \sum_a P(b) \]

© Jack Breese (Microsoft) & Daphne Koller (Stanford)
Variable elimination

\[ P(c) = \sum_b P(c \mid b) \sum_a P(b \mid a) P(a) \]

\[ P(A) \quad P(B \mid A) \]

\[ P(B, A) \]
Variable elimination

\[ P(c) = \sum_b P(c \mid b) \sum_a P(b \mid a) P(a) \]

\[ P(A) \quad P(B \mid A) \]

\[ P(B, A) \rightarrow P(B) \]

© Jack Breese (Microsoft) & Daphne Koller (Stanford)
Variable elimination

\[ P(c) = \sum_b P(c \mid b) \sum_a P(b \mid a) P(a) \]

\[ P(A) \quad P(B \mid A) \]

\[ P(B, A) \rightarrow \sum_A P(B) \quad P(C \mid B) \]

\[ P(C, B) \]

© Jack Breese (Microsoft) & Daphne Koller (Stanford)
Variable elimination

\[
P(c) = \sum_b P(c \mid b) \sum_a P(b \mid a) P(a)
\]

\[
P(A) \quad P(B \mid A)
\]

\[
P(B, A) \quad \Sigma_A P(B) \quad P(C \mid B)
\]

\[
P(C, B) \quad \Sigma_B P(C)
\]
Inference as variable elimination
Inference as variable elimination

- A factor over $X$ is a function from $val(X)$ to numbers in $[0,1]$: 
Inference as variable elimination

- A **factor** over $X$ is a function from $\text{val}(X)$ to numbers in $[0,1]$:
  - A CPT is a factor
Inference as variable elimination

- A **factor** over $X$ is a function from $\text{val}(X)$ to numbers in $[0,1]$:
  - A CPT is a factor
  - A joint distribution is also a factor
Inference as variable elimination

- A factor over $X$ is a function from $\text{val}(X)$ to numbers in $[0,1]$:
  - A CPT is a factor
  - A joint distribution is also a factor

- BN inference:
Inference as variable elimination

- A **factor** over $X$ is a function from $val(X)$ to numbers in $[0,1]$:
  - A CPT is a factor
  - A joint distribution is also a factor

- BN inference:
  - factors are multiplied to give new ones
Inference as variable elimination

- A **factor** over $X$ is a function from $\text{val}(X)$ to numbers in $[0,1]$:
  - A CPT is a factor
  - A joint distribution is also a factor
- BN inference:
  - factors are multiplied to give new ones
  - variables in factors summed out
Inference as variable elimination

- A **factor** over $X$ is a function from $\text{val}(X)$ to numbers in $[0,1]$:
  - A CPT is a factor
  - A joint distribution is also a factor
- BN inference:
  - factors are multiplied to give new ones
  - variables in factors summed out
- A variable can be summed out as soon as all factors mentioning it have been multiplied.
Variable Elimination with loops

Age

Gender

Exposure to Toxics

Smoking

Cancer

Serum Calcium

Lung Tumor
Variable Elimination with loops

\[ P(A) \quad P(G) \quad P(S \mid A, G) \]

\[ P(A, G, S) \]
Variable Elimination with loops

\[ P(A) \quad P(G) \quad P(S \mid A,G) \]

\[ P(A, G, S) \rightarrow \sum_{G} P(A, S) \]
Variable Elimination with loops

\[
P(A) \quad P(G) \quad P(S \mid A, G) \\
P(A, G, S) \quad \sum_G P(A, S) \quad P(A, E, S)
\]
Variable Elimination with loops

Age \rightarrow Gender \rightarrow Smoking \rightarrow Cancer

Exposure to Toxics

Serum Calcium \rightarrow Lung Tumor

\[ P(A) \rightarrow P(G) \rightarrow P(S \mid A, G) \]

\[ P(A, G, S) \rightarrow \sum_G P(A, S) \rightarrow P(A, E, S) \]

\[ P(E \mid A) \]

\[ P(E, S) \]
Variable Elimination with loops

- Age
- Gender
- Smoking
- Cancer
- Serum Calcium
- Lung Tumor

Graph:

- $P(A)$
- $P(G)$
- $P(S \mid A,G)$
- $P(E \mid A)$
- $P(A,G,S)$
- $P(A,S)$
- $P(A,E,S)$
- $P(E,S)$
- $P(C \mid E,S)$
- $P(E,S,C)$
Variable Elimination with loops

\[ P(A), P(G), P(S \mid A,G) \]

\[ P(E \mid A) \]

\[ P(A,G,S) \rightarrow \sum_G P(A,S) \rightarrow \times P(A,E,S) \]

\[ \sum_A P(E,S) \rightarrow P(C \mid E,S) \]

\[ P(E,S,C) \rightarrow \sum_{E,S} P(C) \]
Variable Elimination with loops

\[ P(A) \quad P(G) \quad P(S \mid A, G) \]
\[ P(E \mid A) \quad P(A, G, S) \quad P(A, S) \quad P(A, E, S) \]
\[ P(E, S) \quad P(C \mid E, S) \]
\[ P(E, S, C) \quad P(C) \quad P(C, L) \]
\[ P(L \mid C) \quad P(L) \]

© Jack Breese (Microsoft) & Daphne Koller (Stanford)
Variable Elimination with loops

Complexity is exponential in the size of the factors
Join trees*

A join tree is a partially precompiled factorization

* aka junction trees, Lauritzen-Spiegelhalter, Hugin alg., ...
Computational complexity
Computational complexity

- **Theorem:** Inference in a multi-connected Bayesian network is NP-hard.
Computational complexity

- **Theorem**: Inference in a multi-connected Bayesian network is NP-hard.

Boolean 3CNF formula $\phi = (u \lor \overline{v} \lor w) \land (\overline{u} \lor \overline{w} \lor y)$
Computational complexity

- **Theorem:** Inference in a multi-connected Bayesian network is NP-hard.

Boolean 3CNF formula $\phi = (u \lor \overline{v} \lor w) \land (u \lor \overline{w} \lor y)$
Computational complexity

- **Theorem:** Inference in a multi-connected Bayesian network is NP-hard.

Boolean 3CNF formula $\phi = (u \lor \overline{v} \lor w) \land (\overline{u} \lor \overline{w} \lor y)$

Probability $(\cdot) = \frac{1}{2^n} \cdot \# \text{satisfying assignments of } \phi$
Stochastic simulation

\[
P(b) = 0.03, \quad P(e) = 0.001
\]

\[
\begin{array}{cccc}
    b & e & b & e \\
    \hline
    0.98 & 0.7 & 0.4 & 0.01
\end{array}
\]

\[
P(c) = 0.8, \quad P(n) = 0.3
\]

= c
Stochastic simulation

- **Burglary**
  - $P(b) = 0.03$

- **Earthquake**
  - $P(e) = 0.001$

- **Alarm**
  - $P(a) = \begin{bmatrix} 0.98 & 0.7 & 0.4 & 0.01 \end{bmatrix}$

- **Call**
  - $P(c) = \begin{bmatrix} 0.8 & 0.05 \end{bmatrix}$

- **Newscast**
  - $P(n) = \begin{bmatrix} 0.3 & 0.001 \end{bmatrix}$

**Samples:**

- B
- E
- A
- C
- N
Stochastic simulation

\[ P(b) = 0.03 \]

\[ \begin{array}{c|c|c|c|c}
  & b & \bar{b} & b & \bar{b} \\
\hline
 P(a) & 0.98 & 0.7 & 0.4 & 0.01 \\
\end{array} \]

\[ \text{Burglary} \]

\[ \text{Earthquake} \]

\[ P(e) = 0.001 \]

\[ \begin{array}{c|c|c|c|c|c|c|c}
  & e & \bar{e} & e & \bar{e} & e & \bar{e} & e & \bar{e} \\
\hline
 P(n) & 0.3 & 0.001 & 0.3 & 0.001 & 0.3 & 0.001 & 0.3 & 0.001 \\
\end{array} \]

\[ \text{Alarm} \]

\[ \text{Newscast} \]

\[ = c \]

\[ \begin{array}{c|c|c}
  & a & \bar{a} \\
\hline
 P(c) & 0.8 & 0.05 \\
\end{array} \]

\[ \text{Call} \]

\[ \text{Samples:} \]

© Jack Breese (Microsoft) & Daphne Koller (Stanford)
Stochastic simulation

\[
P(b) = 0.03
\]

\[
P(a) = \begin{array}{cccc}
  b & e & \bar{b} & \bar{e} \\
  0.98 & 0.7 & 0.4 & 0.01
\end{array}
\]

\[
P(c) = \begin{array}{cc}
  a & \bar{a} \\
  0.8 & 0.05
\end{array}
\]

\[
P(n) = \begin{array}{cc}
  e & \bar{e} \\
  0.3 & 0.001
\end{array}
\]

Samples:

\[
\begin{array}{cccccc}
  B & E & A & C & N \\
  \bar{b} & & & & \\
\end{array}
\]
Stochastic simulation

\begin{align*}
P(b) &= 0.03 \\
P(e) &= 0.001 \\
P(a) &= \begin{pmatrix} b & e \\ \bar{b} & \bar{e} \end{pmatrix} \begin{pmatrix} 0.98 & 0.7 \\ 0.4 & 0.01 \end{pmatrix} \\
P(c) &= \begin{pmatrix} a & \bar{a} \end{pmatrix} \begin{pmatrix} 0.8 & 0.05 \end{pmatrix} \\
P(n) &= \begin{pmatrix} e & \bar{e} \end{pmatrix} \begin{pmatrix} 0.3 & 0.001 \end{pmatrix}
\end{align*}

Call = c

Samples:

\begin{align*}
& B \quad E \quad A \quad C \quad N \\
& \bar{b}
\end{align*}
Stochastic simulation

\[ \begin{align*}
\text{Burglary} & \quad \text{Earthquake} \\
\Pr(b) & = 0.03 & \Pr(e) & = 0.001 \\
\Pr(a|b) & = 0.98 & \Pr(a|\overline{b}) & = 0.01 \\
\Pr(c|a) & = 0.8 & \Pr(c|\overline{a}) & = 0.05 \\
\end{align*} \]

\[ \begin{align*}
\text{Alarm} & \quad \text{Newscast} \\
\Pr(n|e) & = 0.3 & \Pr(n|\overline{e}) & = 0.001 \\
\end{align*} \]

Samples:

\[ \begin{array}{cccccc}
B & E & A & C & N \\
\overline{b} & \overline{e} & \\
\end{array} \]
Stochastic simulation

\[ P(b) = 0.03 \]

\[ P(a) = \begin{array}{cccc} b & e & \bar{b} & \bar{e} \\ 0.98 & 0.7 & 0.4 & 0.01 \end{array} \]

\[ P(c) = \begin{array}{cc} a & \bar{a} \\ 0.8 & 0.05 \end{array} \]

\[ P(e) = 0.001 \]

\[ P(n) = \begin{array}{cc} e & \bar{e} \\ 0.3 & 0.001 \end{array} \]

\[ \text{Call} = c \]

Samples:

\[ \begin{array}{cccc} B & E & A & C & N \\ \bar{b} & e \end{array} \]
Stochastic simulation

\[
P(b) = 0.03, \quad P(e) = 0.001
\]

\[
P(a) = \begin{array}{cccc}
b & e & b & e \\
0.98 & 0.7 & 0.4 & 0.01
\end{array}
\]

\[
P(c) = \begin{array}{cc}
a & \bar{a} \\
0.8 & 0.05
\end{array}
\]

\[
P(n) = \begin{array}{cc}
e & \bar{e} \\
0.3 & 0.001
\end{array}
\]

Samples:

\[
\begin{array}{cccccc}
B & E & A & C & N \\
\bar{b} & e & a
\end{array}
\]
Stochastic simulation

\[ P(b) = 0.03 \]

\[ P(a) = \begin{array}{cc} b & e \\ 0.98 & 0.7 \end{array} \]

\[ P(c) = \begin{array}{cc} a & \bar{a} \\ 0.8 & 0.05 \end{array} \]

\[ P(e) = 0.001 \]

\[ P(n) = \begin{array}{cc} e & \bar{e} \\ 0.3 & 0.001 \end{array} \]

Samples:

\[ \bar{b} \ e \ a \]
Stochastic simulation

\[
P(b) = 0.03
\]

\[
P(a)
\begin{array}{llll}
  b  & e  & \bar{b} & \bar{e} \\
  0.98 & 0.7 & 0.4 & 0.01 \\
\end{array}
\]

\[
P(c)
\begin{array}{ll}
  a  & \bar{a} \\
  0.8 & 0.05 \\
\end{array}
\]

\[
P(e) = 0.001
\]

\[
P(n)
\begin{array}{llll}
  e  & \bar{e} \\
  0.3 & 0.001 \\
\end{array}
\]

Samples:

\[
\begin{array}{llll}
  B & E & A & C & N \\
  \bar{b} & e & a & c \\
\end{array}
\]

© Jack Breese (Microsoft) & Daphne Koller (Stanford)
Stochastic simulation

$P(b) = 0.03$

$P(e) = 0.001$

$P(a | b) = 0.98, 0.7, 0.4, 0.01$

$P(a | \overline{b}) = 0.01$

$P(c | a) = 0.8, 0.05$

$P(c | \overline{a}) = 0.05$

$P(n | e) = 0.3, 0.001$

$P(n | \overline{e}) = 0.001$

Samples:

$B \ E \ A \ C \ N$

$\overline{b} \ e \ a \ c$
Stochastic simulation

$P(b) = 0.03$

$P(a | b) = 0.98$
$P(a | \bar{b}) = 0.7$
$P(a) = 0.98$

$P(e | b) = 0.01$
$P(e | \bar{b}) = 0.001$
$P(e) = 0.001$

$P(c | a) = 0.8$
$P(c | \bar{a}) = 0.05$
$P(c) = 0.8$

$P(n | e) = 0.3$
$P(n | \bar{e}) = 0.001$
$P(n) = 0.3$

Samples:

$B E A C N$
$\bar{b} e a c \bar{n}$
Stochastic simulation

\[
P(b) = 0.03, \quad P(a) = 0.98, \quad P(c) = 0.8, \quad P(n) = 0.3
\]

\[
P(e) = 0.001, \quad P(\neg e) = 0.001
\]

\[
P(a|b) = 0.7, \quad P(a|\neg b) = 0.4, \quad P(\neg a|b) = 0.1, \quad P(\neg a|\neg b) = 0.6
\]

\[
P(c|a) = 0.8, \quad P(c|\neg a) = 0.2
\]

\[
P(n|e) = 0.3, \quad P(n|\neg e) = 0.001
\]

\[
P(\neg n|e) = 0.01, \quad P(\neg n|\neg e) = 0.99
\]

Samples:

\[
\begin{array}{cccccc}
B & E & A & C & N \\
\hline
\bar{b} & e & a & c & \bar{n} \\
\hline
\bar{b} & \bar{e} & a & \bar{c} & n \\
\end{array}
\]
Stochastic simulation

$P(b) = 0.03$

$P(a|b) = 0.98, \quad P(a|\bar{b}) = 0.01$

$P(c|a) = 0.8, \quad P(c|\bar{a}) = 0.05$

$P(e|b) = 0.001$

$P(e|\bar{b}) = 0.001$

$P(n|e) = 0.3, \quad P(n|\bar{e}) = 0.001$

Samples:

$\begin{array}{cccccc}
 B & E & A & C & N \\
\bar{b} & e & a & c & \bar{n} \\
\bar{b} & \bar{e} & a & \bar{c} & \bar{n}
\end{array}$

$P(a) = 0.7, \quad P(e) = 0.001$

$P(n) = 0.3, \quad P(\bar{n}) = 0.7$

$P(c) = 0.8, \quad P(\bar{c}) = 0.2$

$P(b) = 0.03, \quad P(\bar{b}) = 0.97$

$P(e) = 0.001, \quad P(\bar{e}) = 0.999$

$P(n) = 0.3, \quad P(\bar{n}) = 0.7$

$P(c) = 0.8, \quad P(\bar{c}) = 0.2$

$P(b) = 0.03, \quad P(\bar{b}) = 0.97$

$P(e) = 0.001, \quad P(\bar{e}) = 0.999$

$P(n) = 0.3, \quad P(\bar{n}) = 0.7$

$P(c) = 0.8, \quad P(\bar{c}) = 0.2$

$P(b) = 0.03, \quad P(\bar{b}) = 0.97$

$P(e) = 0.001, \quad P(\bar{e}) = 0.999$

$P(n) = 0.3, \quad P(\bar{n}) = 0.7$

$P(c) = 0.8, \quad P(\bar{c}) = 0.2$
Stochastic simulation

P(b) = 0.03

P(a) = 0.98 0.7 0.4 0.01

P(c) = 0.8 0.05

P(e) = 0.001

Samples:

© Jack Breese (Microsoft) & Laphne Koller (Stanford)
Stochastic simulation

\[ P(b) = 0.03 \]

\[ P(a) = \begin{bmatrix} 0.98 & 0.7 & 0.4 & 0.01 \end{bmatrix} \]

\[ P(c) = \begin{bmatrix} 0.8 & 0.05 \end{bmatrix} \]

- Burglary
- Earthquake
- Alarm
- Call
- Newscast

\[ \text{Call} = c \]

\[ P(n) = \begin{bmatrix} 0.3 & 0.001 \end{bmatrix} \]

Samples:

\[
P(b|c) \sim \frac{\text{# of live samples with } B=b}{\text{total # of live samples}}
\]

© Jack Breese (Microsoft) & Laphne Koller (Stanford)
Likelihood weighting

<table>
<thead>
<tr>
<th></th>
<th>$a$</th>
<th>$\bar{a}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P(c)$</td>
<td>0.8</td>
<td>0.05</td>
</tr>
<tr>
<td>$P(\bar{c})$</td>
<td>0.2</td>
<td>0.95</td>
</tr>
</tbody>
</table>

Samples:

<table>
<thead>
<tr>
<th>B</th>
<th>E</th>
<th>A</th>
<th>C</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Likelihood weighting

<table>
<thead>
<tr>
<th></th>
<th>( a )</th>
<th>( \bar{a} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P(c) )</td>
<td>0.8</td>
<td>0.05</td>
</tr>
<tr>
<td>( P(\bar{c}) )</td>
<td>0.2</td>
<td>0.95</td>
</tr>
</tbody>
</table>

\[ \text{Call} = c \]

Samples:

<table>
<thead>
<tr>
<th>B</th>
<th>E</th>
<th>A</th>
<th>C</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \bar{b} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Likelihood weighting

<table>
<thead>
<tr>
<th>a</th>
<th>(\bar{a})</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.8</td>
<td>0.05</td>
</tr>
<tr>
<td>0.2</td>
<td>0.95</td>
</tr>
</tbody>
</table>

Samples:

\[
\begin{array}{cccccc}
B & E & A & C & N \\
\bar{b} & e & & & \\
\end{array}
\]
Likelihood weighting

Burglary

Earthquake

Alarm

Call = c

Newscast

Samples:

B E A C N

\( \bar{b} \ e \ a \)

\[
\begin{array}{cc}
 \text{a} & \bar{a} \\
 P(c) & 0.8 & 0.05 \\
 P(\bar{c}) & 0.2 & 0.95
\end{array}
\]
Likelihood weighting

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>(\bar{a})</th>
</tr>
</thead>
<tbody>
<tr>
<td>(P(c))</td>
<td>0.8</td>
<td>0.05</td>
</tr>
<tr>
<td>(P(\bar{c}))</td>
<td>0.2</td>
<td>0.95</td>
</tr>
</tbody>
</table>

Samples:

<table>
<thead>
<tr>
<th>B</th>
<th>E</th>
<th>A</th>
<th>C</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\bar{b})</td>
<td>e</td>
<td>a</td>
<td>c</td>
<td></td>
</tr>
</tbody>
</table>

\(\text{Call} = c\)

\(\text{Burglary} \rightarrow \text{Alarm} \rightarrow \text{Newscast}\)
Likelihood weighting

<table>
<thead>
<tr>
<th></th>
<th>$a$</th>
<th>$\bar{a}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P(c)$</td>
<td>0.8</td>
<td>0.05</td>
</tr>
<tr>
<td>$P(\bar{c})$</td>
<td>0.2</td>
<td>0.95</td>
</tr>
</tbody>
</table>

Samples:

<table>
<thead>
<tr>
<th>$B$</th>
<th>$E$</th>
<th>$A$</th>
<th>$C$</th>
<th>$N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\bar{b}$</td>
<td>$e$</td>
<td>$a$</td>
<td>$c$</td>
<td>$\bar{n}$</td>
</tr>
</tbody>
</table>

$\text{Burglary} \rightarrow \text{Alarm} \rightarrow \text{Newscast}$

$\text{Call} = c$
Likelihood weighting

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>( \bar{a} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P(c) )</td>
<td>0.8</td>
<td>0.05</td>
</tr>
<tr>
<td>( P(\bar{c}) )</td>
<td>0.2</td>
<td>0.95</td>
</tr>
</tbody>
</table>

Samples:

<table>
<thead>
<tr>
<th>B E A C N</th>
<th>weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \bar{b} ) e a c ( \bar{n} )</td>
<td>0.8</td>
</tr>
</tbody>
</table>

Call = \( c \)

Burglary

Earthquake

Alarm

Newscast

© Jack Breese (Microsoft) & Daphne Koller (Stanford)
Likelihood weighting

Samples:

<table>
<thead>
<tr>
<th>B</th>
<th>E</th>
<th>A</th>
<th>C</th>
<th>N</th>
<th>weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>_b</td>
<td>e</td>
<td>a</td>
<td>c</td>
<td>_n</td>
<td>0.8</td>
</tr>
<tr>
<td>b</td>
<td>_e</td>
<td>a</td>
<td>c</td>
<td>n</td>
<td>0.05</td>
</tr>
</tbody>
</table>
Likelihood weighting

Samples:

<table>
<thead>
<tr>
<th>B</th>
<th>E</th>
<th>A</th>
<th>C</th>
<th>N</th>
<th>weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>(b)</td>
<td>(e)</td>
<td>(a)</td>
<td>(c)</td>
<td>(n)</td>
<td>0.8</td>
</tr>
<tr>
<td>(b)</td>
<td>(e)</td>
<td>(_)</td>
<td>(a)</td>
<td>(c)</td>
<td>(n)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>
Likelihood weighting

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>N</th>
<th>weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>b</td>
<td>e</td>
<td>a</td>
<td>c</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>e</td>
<td>a</td>
<td>c</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$P(b|c) = \frac{\text{weight of samples with } B=b}{\text{total weight of samples}}$
Markov Chain Monte Carlo
MCMC with Gibbs Sampling

Fix the values of observed variables
Set the values of all non-observed variables randomly
Perform a random walk through the space of complete variable assignments. On each move:

1. Pick a variable X
2. Calculate $\Pr(X=\text{true} \mid \text{all other variables})$
3. Set X to true with that probability

Repeat many times. Frequency with which any variable X is true is its posterior probability.

Converges to true posterior when frequencies stop changing significantly

- stable distribution, mixing
Markov Blanket Sampling

How to calculate $\Pr(X=\text{true} \mid \text{all other variables})$?

Recall: a variable is independent of all others given it’s Markov Blanket

- parents
- children
- other parents of children

So problem becomes calculating $\Pr(X=\text{true} \mid \text{MB}(X))$

- We solve this sub-problem exactly
- Fortunately, it is easy to solve

$$P(X) = \alpha P(X \mid Parents(X)) \prod_{Y \in \text{Children}(X)} P(Y \mid Parents(Y))$$
Example

\[ P(X) = \alpha P(X \mid \text{Parents}(X)) \prod_{Y \in \text{Children}(X)} P(Y \mid \text{Parents}(Y)) \]

\[
P(X \mid A, B, C) = \frac{P(X, A, B, C)}{P(A, B, C)}
\]

\[
= \frac{P(A)P(X \mid A)P(C)P(B \mid X, C)}{P(A, B, C)}
\]

\[
= \left[ \frac{P(A)P(C)}{P(A, B, C)} \right] P(X \mid A)P(B \mid X, C)
\]

\[
= \alpha P(X \mid A)P(B \mid X, C)
\]
Example

Smoking

<table>
<thead>
<tr>
<th>P(s)</th>
<th>s</th>
<th>0.6</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>~s</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Heart disease

<table>
<thead>
<tr>
<th>H</th>
<th>G</th>
<th>P(b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>h</td>
<td>g</td>
<td>0.9</td>
</tr>
<tr>
<td>h</td>
<td>~g</td>
<td>0.8</td>
</tr>
<tr>
<td>~h</td>
<td>g</td>
<td>0.7</td>
</tr>
<tr>
<td>~h</td>
<td>~g</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Lung disease

<table>
<thead>
<tr>
<th>P(g)</th>
</tr>
</thead>
<tbody>
<tr>
<td>s</td>
</tr>
<tr>
<td>~s</td>
</tr>
</tbody>
</table>
Example

**Smoking**

- $P(s)$: 0.2
- $P(g | s)$: 0.8
- $P(g | \sim s)$: 0.1

**Heart disease**

- $P(s | H)$: 0.6
- $P(s | \sim H)$: 0.1

**Lung disease**

- $P(g)$: 0.8

**Shortness of breath**

- $P(b | h, g)$: 0.9
- $P(b | h, \sim g)$: 0.8
- $P(b | \sim h, g)$: 0.7
- $P(b | \sim h, \sim g)$: 0.1

Evidence: $s$, $b$
Randomly set: $h$, $b$
Example

Smoking

Heart disease

Lung disease

Shortness of breath

Evidence: s, b
Randomly set: h, g
Sample H using P(H|s,g,b)
Example

Evidence: s, b
Randomly set: ~h, g
Sample H using P(H|s,g,b)
⇒ Suppose result is ~h
Example

Evidence: s, b
Randomly set: ~h, g
Sample H using P(H|s,g,b)
Suppose result is ~h
Sample G using P(G|s,~h,b)
Example

Evidence: s, b
Randomly set: ~h, g
Sample H using P(H|s,g,b)
Suppose result is ~h
Sample G using P(G|s,~h,b)
⇒ Suppose result is g
Example

**Evidence:** s, b
Randomly set: ~h, g
Sample H using P(H|s,g,b)
Suppose result is ~h
Sample G using P(G|s,~h,b)
⇒Suppose result is g
Sample G using P(G|s,~h,b)
Example

Evidence: s, b
Randomly set: ~h, g
Sample H using P(H|s,g,b)
Suppose result is ~h
Sample G using P(G|s,~h,b)
⇒Suppose result is g
Sample G using P(G|s,~h,b)
⇒Suppose result is ~g
Gibbs MCMC Summary

Advantages:

- No samples are discarded
- No problem with samples of low weight
- Can be implemented very efficiently
  - 10K samples @ second

Disadvantages:

- Can get stuck if relationship between two variables is deterministic
- Many variations have been devised to make MCMC more robust
Gibbs MCMC Summary

\[ P(X|E) = \frac{\text{number of samples with } X=x}{\text{total number of samples}} \]

Advantages:
- No samples are discarded
- No problem with samples of low weight
- Can be implemented very efficiently
  - 10K samples @ second

Disadvantages:
- Can get stuck if relationship between two variables is deterministic
- Many variations have been devised to make MCMC more robust
Other approaches

- Search based techniques
  - search for high-probability instantiations
  - use instantiations to approximate probabilities

- Structural approximation
  - simplify network
    - eliminate edges, nodes
    - abstract node values
    - simplify CPTs
  - do inference in simplified network
Course Contents

- Concepts in Probability
- Bayesian Networks
- Inference
- Decision making
  » Learning networks from data
- Reasoning over time
- Applications
Learning networks from data

- The learning task
- Parameter learning
  - Fully observable
  - Partially observable
- Structure learning
- Hidden variables
The learning task

Input: training data
The learning task

Input: training data  Output: BN modeling data
The learning task

Input: training data  Output: BN modeling data

- Input: fully or partially observable data cases?
The learning task

Input: training data
Output: BN modeling data

- Input: fully or partially observable data cases?
- Output: parameters or also structure?
Parameter learning: one variable
Parameter learning: one variable

- Unfamiliar coin:
Parameter learning: one variable

- Unfamiliar coin:
  - Let $\theta =$ bias of coin (long-run fraction of heads)
Parameter learning: one variable

- Unfamiliar coin:
  - Let $\theta =$ bias of coin (long-run fraction of heads)
- If $\theta$ known (given), then
Parameter learning: one variable

- Unfamiliar coin:
  - Let $\theta = \text{bias of coin (long-run fraction of heads)}$

- If $\theta$ known (given), then
  - $P(X = \text{heads} \mid \theta) =$
Parameter learning: one variable

- Unfamiliar coin:
  - Let $\theta =$ bias of coin (long-run fraction of heads)

- If $\theta$ known (given), then
  - $P(X = \text{heads} \mid \theta) = \theta$
Parameter learning: one variable

- Unfamiliar coin:
  - Let $\theta =$ bias of coin (long-run fraction of heads)
- If $\theta$ known (given), then
  - $P(X = \text{heads} \mid \theta) = \theta$
- Different coin tosses independent given $\theta$
  - $P(X_1, \ldots, X_n \mid \theta) =$
Parameter learning: one variable

- Unfamiliar coin:
  - Let $\theta = \text{bias of coin (long-run fraction of heads)}$

- If $\theta$ known (given), then
  - $P(X = \text{heads} \mid \theta) = \theta$

- Different coin tosses independent given $\theta$
  - $P(X_1, \ldots, X_n \mid \theta) = h \text{ heads}, t \text{ tails}$
Parameter learning: one variable

- Unfamiliar coin:
  - Let $\theta =$ bias of coin (long-run fraction of heads)

- If $\theta$ known (given), then
  - $P(X = \text{heads} \mid \theta) = \theta$

- Different coin tosses independent given $\theta$
  \[ P(X_1, \ldots, X_n \mid \theta) = \theta^h (1-\theta)^t \]
  - $h$ heads, $t$ tails
Maximum likelihood
Maximum likelihood

- Input: a set of previous coin tosses
  \[ X_1, ..., X_n = \{H, T, H, H, H, T, T, H, \ldots, H\} \]
Maximum likelihood

- Input: a set of previous coin tosses
  \[ X_1, \ldots, X_n = \{H, T, H, H, H, T, T, H, \ldots, H\} \]
  \[ h \text{ heads, } t \text{ tails} \]
Maximum likelihood

- Input: a set of previous coin tosses
  \[ X_1, \ldots, X_n = \{H, T, H, H, H, T, T, H, \ldots, H\} \]
  \[ h \text{ heads, } t \text{ tails} \]

- Goal: estimate \( \theta \)
Maximum likelihood

- Input: a set of previous coin tosses
  \[ X_1, \ldots, X_n = \{H, T, H, H, H, T, T, H, \ldots, H\} \]
  \( h \) heads, \( t \) tails

- Goal: estimate \( \theta \)

- The likelihood
  \[ P(X_1, \ldots, X_n \mid \theta) = \theta^h (1-\theta)^t \]
Maximum likelihood

- Input: a set of previous coin tosses
  \[ X_1, \ldots, X_n = \{H, T, H, H, H, T, T, H, \ldots, H\} \]
  \( h \) heads, \( t \) tails

- Goal: estimate \( \theta \)

- The likelihood \( P(X_1, \ldots, X_n \mid \theta) = \theta^h (1-\theta)^t \)

- The maximum likelihood solution is:
Maximum likelihood

- Input: a set of previous coin tosses
  \[ X_1, \ldots, X_n = \{H, T, H, H, H, T, T, H, \ldots, H\} \]
  \( h \) heads, \( t \) tails

- Goal: estimate \( \theta \)

- The likelihood \( P(X_1, \ldots, X_n \mid \theta) = \theta^h (1-\theta)^t \)

- The maximum likelihood solution is:
  \[ \theta^* = \frac{h}{h+t} \]
General parameter learning

- A multi-variable BN is composed of several independent parameters ("coins").
General parameter learning

- A multi-variable BN is composed of several independent parameters ("coins").
General parameter learning

- A multi-variable BN is composed of several independent parameters ("coins").

\[ A \rightarrow B \quad \text{Three parameters:} \]
General parameter learning

- A multi-variable BN is composed of several independent parameters ("coins").

\[ \theta_A, \theta_B|a, \theta_B|\overline{a} \]
General parameter learning

- A multi-variable BN is composed of several independent parameters ("coins").

\[ A \rightarrow B \]

Three parameters:
\[ \theta_A, \theta_B | a, \theta_B | \overline{a} \]

- Can use same techniques as one-variable case to learn each one separately
General parameter learning

- A multi-variable BN is composed of several independent parameters ("coins").

\[
\theta_A, \theta_B|a, \theta_B|\overline{a}
\]

- Can use same techniques as one-variable case to learn each one separately

Max likelihood estimate of \(\theta_B|\overline{a}\) would be:
General parameter learning

- A multi-variable BN is composed of several independent parameters ("coins").

  \[ A \rightarrow B \]
  
  Three parameters: \( \theta_A, \theta_B | a, \theta_B | \bar{a} \)

- Can use same techniques as one-variable case to learn each one separately

  Max likelihood estimate of \( \theta_B | \bar{a} \) would be:

  \[
  \theta^*_B | \bar{a} = \frac{\# \text{data cases with } b, \bar{a}}{\# \text{data cases with } \bar{a}}
  \]
Partially observable data

- Fill in missing data with “expected” value
  - expected = distribution over possible values
  - use “best guess” BN to estimate distribution
Intuition

- In fully observable case:
Intuition

- In fully observable case:

$$\theta^*_n|e = \frac{\# \text{data cases with } n, e}{\# \text{data cases with } e}$$
Intuition

- In fully observable case:

\[
\theta^*_n|e = \frac{\# \text{data cases with } n, e}{\# \text{data cases with } e} = \frac{\sum_j I(n, e \mid d_j)}{\sum_j I(e \mid d_j)}
\]

\[
I(e \mid d_j) = \begin{cases} 
1 & \text{if } E-e \text{ in data case } d_j \\
0 & \text{otherwise}
\end{cases}
\]
Intuition

■ In fully observable case:

\[ \Theta_{n|e}^* = \frac{\# \text{data cases with } n, e}{\# \text{data cases with } e} = \frac{\sum_j I(n, e \mid d_j)}{\sum_j I(e \mid d_j)} \]

\[ I(e \mid d_j) = \begin{cases} 
1 & \text{if } E-e \text{ in data case } d_j \\
0 & \text{otherwise} 
\end{cases} \]

■ In partially observable case I is unknown.
Intuition

■ In fully observable case:

\[ \theta^*_n|e = \frac{\# \text{data cases with } n, e}{\# \text{data cases with } e} = \frac{\sum_j I(n,e \mid d_j)}{\sum_j I(e \mid d_j)} \]

\[ I(e \mid d_j) = \begin{cases} 1 & \text{if } E-e \text{ in data case } d_j \\ 0 & \text{otherwise} \end{cases} \]

■ In partially observable case \( I \) is unknown.

Best estimate for \( I \) is:
Intuition

- In fully observable case:

\[
\theta^*_{n|e} = \frac{\text{#data cases with } n, e}{\text{#data cases with } e} = \frac{\sum_j I(n,e \mid d_j)}{\sum_j I(e \mid d_j)}
\]

\[
I(e \mid d_j) = \begin{cases} 
1 & \text{if } E-e \text{ in data case } d_j \\
0 & \text{otherwise}
\end{cases}
\]

- In partially observable case \( I \) is unknown.

Best estimate for \( I \) is:

\[
\hat{I}(n,e \mid d_j) = P_{\theta^*}(n,e \mid d_j)
\]
Intuition

■ In fully observable case:

\[
\theta^*_{n|e} = \frac{\# \text{data cases with } n, e}{\# \text{data cases with } e} = \frac{\sum_j I(n,e \mid d_j)}{\sum_j I(e \mid d_j)}
\]

\[
I(e \mid d_j) = \begin{cases} 
1 & \text{if } E-e \text{ in data case } d_j \\
0 & \text{otherwise}
\end{cases}
\]

■ In partially observable case \( I \) is unknown.

Best estimate for \( I \) is:

\[
\hat{I}(n,e \mid d_j) = P_{\theta^*}(n,e \mid d_j)
\]

Problem: \( \theta^* \) unknown.
Expectation Maximization (EM)
Expectation Maximization (EM)

- Expectation (E) step
  - Use current parameters $\theta$ to estimate filled in data.
Expectation Maximization (EM)

- Expectation (E) step
  - Use current parameters $\theta$ to estimate filled in data.
  
  $$ \hat{I}(n, e \mid d_j) = P_\theta(n, e \mid d_j) $$
Expectation Maximization (EM)

- **Expectation (E) step**
  - Use current parameters $\theta$ to estimate filled in data.
  \[ \hat{I}(n, e \mid d_j) = P_{\theta} (n, e \mid d_j) \]

- **Maximization (M) step**
  - Use filled in data to do max likelihood estimation
Expectation Maximization (EM)

- Expectation (E) step
  - Use current parameters $\theta$ to estimate filled in data.
    $$\hat{I}(n, e | d_j) = P_\theta (n, e | d_j)$$

- Maximization (M) step
  - Use filled in data to do max likelihood estimation
    $$\tilde{\theta}_{n|e} = \frac{\sum_j \hat{I}(n, e | d_j)}{\sum_j \hat{I}(e | d_j)}$$
Expectation Maximization (EM)

- **Expectation (E) step**
  - Use current parameters $\theta$ to estimate filled in data.
  \[
  \hat{I}(n, e | d_j) = P_\theta (n, e | d_j)
  \]

- **Maximization (M) step**
  - Use filled in data to do max likelihood estimation
  \[
  \tilde{\theta}_{n|e} = \frac{\sum_j \hat{I}(n, e | d_j)}{\sum_j \hat{I}(e | d_j)}
  \]

- Set: $\theta := \tilde{\theta}$
Expectation Maximization (EM)

Repeat:

- **Expectation (E) step**
  - Use current parameters $\theta$ to estimate filled in data.

  $$\hat{I}(n, e \mid d_j) = P_\theta(n, e \mid d_j)$$

- **Maximization (M) step**
  - Use filled in data to do max likelihood estimation

  $$\tilde{\theta}_{n|e} = \frac{\sum_j \hat{I}(n, e \mid d_j)}{\sum_j \hat{I}(e \mid d_j)}$$

- Set: $\theta := \tilde{\theta}$

until convergence.
Structure learning

Goal: find “good” BN structure (relative to data)

Solution: do heuristic search over space of network structures.
Search space

Space = network structures
Operators = add/reverse/delete edges
Heuristic search

Use scoring function to do heuristic search (any algorithm). Greedy hill-climbing with randomness works pretty well.
Scoring
Scoring

- Fill in parameters using previous techniques & score completed networks.
Scoring

- Fill in parameters using previous techniques & score completed networks.
- One possibility for score:
Scoring

- Fill in parameters using previous techniques & score completed networks.
- One possibility for score:
  likelihood function: $Score(B) = P(data \mid B)$
Scoring

- Fill in parameters using previous techniques & score completed networks.
- One possibility for score:
  likelihood function: \( \text{Score}(B) = P(\text{data} \mid B) \)
Scoring

- Fill in parameters using previous techniques & score completed networks.
- One possibility for score:
  likelihood function: \( \text{Score}(B) = P(data \mid B) \)
  Example: \( X, Y \) independent coin tosses
  typical data = (27 h-h, 22 h-t, 25 t-h, 26 t-t)
Scoring

- Fill in parameters using previous techniques & score completed networks.
- One possibility for score:
  
  likelihood function: \( \text{Score}(B) = P(\text{data} \mid B) \)

  Example: \( X, Y \) independent coin tosses
  
  typical \( \text{data} = (27 \ h-h, 22 \ h-t, 25 \ t-h, 26 \ t-t) \)
  
  Maximum likelihood network structure:
Scoring

- Fill in parameters using previous techniques & score completed networks.
- One possibility for score:
  likelihood function: \( \text{Score}(B) = P(data \mid B) \)

  Example: \( X, Y \) independent coin tosses
  typical data = (27 h-h, 22 h-t, 25 t-h, 26 t-t)

  Maximum likelihood network structure:

  Max. likelihood network typically fully connected
Scoring

- Fill in parameters using previous techniques & score completed networks.
- One possibility for score:
  
  likelihood function: \( \text{Score}(B) = P(\text{data} \mid B) \)  
  
  Example: \( X, Y \) independent coin tosses  
  
  typical \( \text{data} = (27 \text{ h-h, 22 h-t, 25 t-h, 26 t-t}) \)  
  
  Maximum likelihood network structure:

- Max. likelihood network typically fully connected

  \[ \text{This is not surprising: maximum likelihood always overfits...} \]
Better scoring functions
Better scoring functions

- MDL formulation: balance fit to data and model complexity (\# of parameters)
Better scoring functions

- MDL formulation: balance fit to data and model complexity (# of parameters)

\[ \text{Score}(B) = P(\text{data} \mid B) - \text{model complexity} \]
Better scoring functions

- MDL formulation: balance fit to data and model complexity (# of parameters)
  \[
  \text{Score}(B) = P(\text{data} \mid B) - \text{model complexity}
  \]
- Full Bayesian formulation
Better scoring functions

- MDL formulation: balance fit to data and model complexity (# of parameters)

\[ \text{Score}(B) = P(\text{data} \mid B) - \text{model complexity} \]

- Full Bayesian formulation
  - prior on network structures & parameters
Better scoring functions

- MDL formulation: balance fit to data and model complexity (# of parameters)

  \[ \text{Score}(B) = P(\text{data} \mid B) - \text{model complexity} \]

- Full Bayesian formulation
  - prior on network structures & parameters
  - more parameters $\Rightarrow$ higher dimensional space
Better scoring functions

- MDL formulation: balance fit to data and model complexity (# of parameters)
  
  \[ \text{Score}(B) = P(\text{data} \mid B) - \text{model complexity} \]

- Full Bayesian formulation
  - prior on network structures & parameters
  - more parameters \(\Rightarrow\) higher dimensional space
  - get balance effect as a byproduct*
Better scoring functions

- MDL formulation: balance fit to data and model complexity (# of parameters)

\[ \text{Score}(B) = P(\text{data} \mid B) - \text{model complexity} \]

- Full Bayesian formulation
  - prior on network structures & parameters
  - more parameters \( \Rightarrow \) higher dimensional space
  - get balance effect as a byproduct*

* with Dirichlet parameter prior, MDL is an approximation to full Bayesian score.
Course Contents

- Concepts in Probability
- Bayesian Networks
- Inference
- Decision making
- Learning networks from data
- Reasoning over time

» Applications
Applications

- Medical expert systems
  - Pathfinder
  - Parenting MSN
- Fault diagnosis
  - Ricoh FIXIT
  - Decision-theoretic troubleshooting
- Vista
- Collaborative filtering
Why use Bayesian Networks?

- Explicit management of uncertainty/tradeoffs
- Modularity implies maintainability
- Better, flexible, and robust recommendation strategies
Pathfinder

- Pathfinder is one of the first BN systems.
- It performs diagnosis of lymph-node diseases.
- It deals with over 60 diseases and 100 findings.
- Commercialized by Intellipath and Chapman Hall publishing and applied to about 20 tissue types.
On Parenting: Selecting problem

- Diagnostic indexing for Home Health site on Microsoft Network
- Enter symptoms for pediatric complaints
- Recommends multimedia content
On Parenting: MSN

Original Multiple Fault Model
RICOH Fixit

- Diagnostics and information retrieval
What is Collaborative Filtering?

- A way to find cool websites, news stories, music artists etc
- Uses data on the preferences of many users, not descriptions of the content.
- **Firefly, Net Perceptions** (GroupLens), and others offer this technology.
Bayesian Clustering for Collaborative Filtering

- Probabilistic summary of the data
- Reduces the number of parameters to represent a set of preferences
- Provides insight into usage patterns.
- Inference:

\[ P(\text{Like title } i \mid \text{Like title } j, \text{Like title } k) \]
Applying Bayesian clustering

user classes

<table>
<thead>
<tr>
<th>title 1</th>
<th>title 2</th>
<th>...</th>
<th>title n</th>
</tr>
</thead>
<tbody>
<tr>
<td>title1</td>
<td>p(like)=0.2</td>
<td>p(like)=0.8</td>
<td></td>
</tr>
<tr>
<td>title2</td>
<td>p(like)=0.7</td>
<td>p(like)=0.1</td>
<td></td>
</tr>
<tr>
<td>title3</td>
<td>p(like)=0.99</td>
<td>p(like)=0.01</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
MSNBC Story clusters

Readers of commerce and technology stories (36%):

- E-mail delivery isn't exactly guaranteed
- Should you buy a DVD player?
- Price low, demand high for Nintendo
MSNBC Story clusters

Readers of commerce and technology stories (36%):
- E-mail delivery isn't exactly guaranteed
- Should you buy a DVD player?
- Price low, demand high for Nintendo

Readers of top promoted stories (29%):
- 757 Crashes At Sea
- Israel, Palestinians Agree To Direct Talks
- Fuhrman Pleads Innocent To Perjury
MSNBC Story clusters

Readers of commerce and technology stories (36%):
- E-mail delivery isn't exactly guaranteed
- Should you buy a DVD player?
- Price low, demand high for Nintendo

Readers of top promoted stories (29%):
- 757 Crashes At Sea
- Israel, Palestinians Agree To Direct Talks
- Fuhrman Pleads Innocent To Perjury

Sports Readers (19%):
- Umps refusing to work is the right thing
- Cowboys are reborn in win over eagles
- Did Orioles spend money wisely?
MSNBC Story clusters

Readers of commerce and technology stories (36%):
- E-mail delivery isn't exactly guaranteed
- Should you buy a DVD player?
- Price low, demand high for Nintendo

Readers of top promoted stories (29%):
- 757 Crashes At Sea
- Israel, Palestinians Agree To Direct Talks
- Fuhrman Pleads Innocent To Perjury

Sports Readers (19%):
- Umps refusing to work is the right thing
- Cowboys are reborn in win over eagles
- Did Orioles spend money wisely?

Readers of “Softer” News (12%):
- The truth about what things cost
- Fuhrman Pleads Innocent To Perjury
- Real Astrology
Top 5 shows by user class

Class 1
- Power rangers
- Animaniacs
- X-men
- Tazmania
- Spider man

Class 2
- Young and restless
- Bold and the beautiful
- As the world turns
- Price is right
- CBS eve news

Class 3
- Tonight show
- Conan O’Brien
- NBC nightly news
- Later with Kinnear
- Seinfeld

Class 4
- 60 minutes
- NBC nightly news
- CBS eve news
- Murder she wrote
- Matlock

Class 5
- Seinfeld
- Friends
- Mad about you
- ER
- Frasier
Richer model

Age ➔ Gender ➔ Likes soaps ➔ User class

Watches Seinfeld ➔ Watches NYPD Blue ➔ Watches Power Rangers
What’s old?

Decision theory & probability theory provide:
- principled models of belief and preference;
- techniques for:
  - integrating evidence (conditioning);
  - optimal decision making (max. expected utility);
  - targeted information gathering (value of info.);
  - parameter estimation from data.
What’s new?
What’s new?

Bayesian networks exploit domain structure to allow compact representations of complex models.
What’s new?

Bayesian networks exploit domain structure to allow compact representations of complex models.
What’s new?

Bayesian networks exploit domain structure to allow compact representations of complex models.

Knowledge Acquisition

Structured Representation
What’s new?

Bayesian networks exploit domain structure to allow compact representations of complex models.

Knowledge Acquisition

Structured Representation

Inference

© Jack Breeze (Microsoft) & Daphne Koller (Stanford)
What’s new?

Bayesian networks exploit domain structure to allow compact representations of complex models.

Knowledge Acquisition  Structured Representation  Learning

Inference
What’s in our future?
What’s in our future?

- Better models for:
What’s in our future?

- Better models for:
  - preferences & utilities;
What’s in our future?

- Better models for:
  - preferences & utilities;
  - not-so-precise numerical probabilities.
What’s in our future?

- Better models for:
  - preferences & utilities;
  - not-so-precise numerical probabilities.
- Inferring causality from data.
What’s in our future?

- Better models for:
  - preferences & utilities;
  - not-so-precise numerical probabilities.
- Inferring causality from data.
- More expressive representation languages:
What’s in our future?

- Better models for:
  - preferences & utilities;
  - not-so-precise numerical probabilities.
- Inferring causality from data.
- More expressive representation languages:
  - structured domains with multiple objects;
What’s in our future?

- Better models for:
  - preferences & utilities;
  - not-so-precise numerical probabilities.
- Inferring causality from data.
- More expressive representation languages:
  - structured domains with multiple objects;
  - levels of abstraction;
What’s in our future?

- Better models for:
  - preferences & utilities;
  - not-so-precise numerical probabilities.
- Inferring causality from data.
- More expressive representation languages:
  - structured domains with multiple objects;
  - levels of abstraction;
  - reasoning about time;
What’s in our future?

- Better models for:
  - preferences & utilities;
  - not-so-precise numerical probabilities.
- Inferring causality from data.
- More expressive representation languages:
  - structured domains with multiple objects;
  - levels of abstraction;
  - reasoning about time;
  - hybrid (continuous/discrete) models.
What’s in our future?

- Better models for:
  - preferences & utilities;
  - not-so-precise numerical probabilities.
- Inferring causality from data.
- More expressive representation languages:
  - structured domains with multiple objects;
  - levels of abstraction;
  - reasoning about time;
  - hybrid (continuous/discrete) models.