Introduction to Graphical Models

Readings in Prince textbook: Chapters 10 and 11 but mainly only on directed graphs at this time
Bayes Nets for representing and reasoning about uncertainty

Andrew W. Moore
Associate Professor
School of Computer Science
Carnegie Mellon University

Copyright © 2001, Andrew W. Moore
Oct 15th, 2001
Review: Probability Theory

- Sum rule (marginal distributions)
  \[ p(x) = \sum_y p(x, y) \]

- Product rule
  \[ p(x, y) = p(x|y)p(y) \]

- From these we have Bayes’ theorem
  \[ p(y|x) = \frac{p(x|y)p(y)}{p(x)} \]
  – with normalization factor
  \[ p(x) = \sum_y p(x|y)p(y) \]

Christopher Bishop, MSR
Review: Conditional Probability

- Conditional Probability (rewriting product rule)
  \[
  P(A \mid B) = \frac{P(A, B)}{P(B)}
  \]

- Chain Rule
  \[
  P(A, B, C, D) = \frac{P(A)}{P(A)} \cdot \frac{P(A, B)}{P(A, B)} \cdot \frac{P(A, B, C)}{P(A, B, C)} \cdot \frac{P(A, B, C, D)}{P(A, B, C, D)}
  = P(A) \cdot P(B \mid A) \cdot P(C \mid A, B) \cdot P(D \mid A, B, C)
  \]

- Conditional Independence
  \[
  P(A, B \mid C) = P(A \mid C) \cdot P(B \mid C)
  \]
  – statistical independence
  \[
  P(A, B) = P(A) \cdot P(B)
  \]
Overview of Graphical Models

- Graphical Models model conditional dependence/independence
- Graph structure specifies how joint probability factors
- Directed graphs

\[ p(x_1, \ldots, x_D) = \prod_{i=1}^{D} p(x_i | pa_i) \quad \text{Example: HMM} \]

- Undirected graphs

\[ p(x) = \frac{1}{Z} \prod_C \psi_C(x_C) \quad \text{Example: MRF} \]

- Inference by message passing: belief propagation
  - Sum-product algorithm
  - Max-product (Min-sum if using logs)

We will focus mainly on directed graphs right now.
The Joint Distribution

Example: Boolean variables A, B, C

Recipe for making a joint distribution of M variables:
Recipe for making a joint distribution of $M$ variables:

1. Make a truth table listing all combinations of values of your variables (if there are $M$ Boolean variables then the table will have $2^M$ rows).

\[
\begin{array}{ccc}
A & B & C \\
0 & 0 & 0 \\
0 & 0 & 1 \\
0 & 1 & 0 \\
0 & 1 & 1 \\
1 & 0 & 0 \\
1 & 0 & 1 \\
1 & 1 & 0 \\
1 & 1 & 1 \\
\end{array}
\]
The Joint Distribution

Recipe for making a joint distribution of M variables:

1. Make a truth table listing all combinations of values of your variables (if there are M Boolean variables then the table will have $2^M$ rows).
2. For each combination of values, say how probable it is.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.30</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0.05</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0.10</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0.05</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0.05</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0.10</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0.25</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.10</td>
</tr>
</tbody>
</table>

Example: Boolean variables A, B, C
Recipe for making a joint distribution of M variables:

1. Make a **truth table** listing all combinations of values of your variables (if there are M Boolean variables then the table will have $2^M$ rows).
2. For each combination of values, say how probable it is.
3. If you subscribe to the axioms of probability, those numbers must sum to 1.

**Example: Boolean variables A, B, C**

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.30</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0.05</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0.10</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0.05</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0.05</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0.10</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0.25</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.10</td>
</tr>
</tbody>
</table>

Andrew Moore, CMU
Joint distributions

• Good news

Once you have a joint distribution, you can answer all sorts of probabilistic questions involving combinations of attributes
Using the Joint

\[
P(E) = \sum_{\text{rows matching } E} P(\text{row})
\]

<table>
<thead>
<tr>
<th>gender</th>
<th>hours_worked</th>
<th>wealth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>v0:40.5-</td>
<td>poor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rich</td>
</tr>
<tr>
<td></td>
<td>v1:40.5+</td>
<td>poor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rich</td>
</tr>
<tr>
<td>Male</td>
<td>v0:40.5-</td>
<td>poor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rich</td>
</tr>
<tr>
<td></td>
<td>v1:40.5+</td>
<td>poor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rich</td>
</tr>
</tbody>
</table>

\[P(\text{Poor Male}) = 0.4654\]
Using the Joint

<table>
<thead>
<tr>
<th>gender</th>
<th>hours_worked</th>
<th>wealth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>v0:40.5-</td>
<td>poor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rich</td>
</tr>
<tr>
<td>Female</td>
<td>v1:40.5+</td>
<td>poor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rich</td>
</tr>
<tr>
<td>Male</td>
<td>v0:40.5-</td>
<td>poor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rich</td>
</tr>
<tr>
<td>Male</td>
<td>v1:40.5+</td>
<td>poor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rich</td>
</tr>
</tbody>
</table>

\[
P(Poor) = 0.7604
\]

\[
P(E) = \sum_{\text{rows matching } E} P(\text{row})
\]
Inference with the Joint

computing conditional probabilities

$$P(E_1 \mid E_2) = \frac{P(E_1 \land E_2)}{P(E_2)} \quad = \quad \frac{\sum P(\text{row})}{\sum P(\text{row})}$$

Andrew Moore, CMU
Inference with the Joint

\[
P(E_1 \mid E_2) = \frac{P(E_1 \land E_2)}{P(E_2)} = \frac{\sum P(\text{row}) \mid \text{rows matching } E_1 \text{ and } E_2}{\sum P(\text{row}) \mid \text{rows matching } E_2}
\]

\[
P(\text{Male} \mid \text{Poor}) = \frac{0.4654}{0.7604} = 0.612
\]

Andrew Moore, CMU
Joint distributions

• Good news
  Once you have a joint distribution, you can answer all sorts of probabilistic questions involving combinations of attributes

• Bad news
  Impossible to create JD for more than about ten attributes because there are so many numbers needed when you build the thing.

For 10 binary variables you need to specify $2^{10} - 1$ numbers = 1023.

(question for class: why the -1?)
How to use Fewer Numbers

• Factor the joint distribution into a product of distributions over subsets of variables

• Identify (or just assume) independence between some subsets of variables

• Use that independence to simplify some of the distributions

• Graphical models provide a principled way of doing this.
Factoring

• Consider an arbitrary joint distribution

\[ p(x, y, z) \]

• We can always factor it, by application of the chain rule

\[
p(x, y, z) = p(x)p(y, z|x) = p(x)p(y|x)p(z|x, y)\]

what this factored form looks like as a graphical model
Directed versus Undirected Graphs

Directed Graph Examples:
- Bayes nets
- HMMs

Undirected Graph Examples
- MRFS

Note: The word “graphical” denotes the graph structure underlying the model, not the fact that you can draw a pretty picture of it (although that helps).
Graphical Model Concepts

- Nodes represent random variables.
- Edges (or lack of edges) represent conditional dependence (or independence).
- Each node is annotated with a table of conditional probabilities wrt parents.

\[
P(s)=0.3
\]

\[
P(M)=0.6
\]

\[
P(R|M)=0.3
P(R|\neg M)=0.6
\]

\[
P(T|L)=0.3
P(T|\neg L)=0.8
\]

\[
P(L|M^S)=0.05
P(L|M^\neg S)=0.1
P(L|\neg M^S)=0.1
P(L|\neg M^\neg S)=0.2
\]
Graphical Model Concepts

Note: The word “graphical” denotes the graph structure underlying the model, not the fact that you can draw a pretty picture of it using graphics.
Directed Acyclic Graphs

- Directed acyclic means we can’t follow arrows around in a cycle.
- Examples: chains; trees
- Also, things that look like this:

We can “read” the factored form of the joint distribution immediately from a directed graph

$$p(x_1, \ldots, x_D) = \prod_{i=1}^{D} p(x_i | pa_i)$$

where $pa_i$ denotes the parents of $i$
Factoring Examples

- Joint distribution

\[ p(x_1, \ldots, x_D) = \prod_{i=1}^{D} p(x_i | \text{pa}_i) \]

where \( \text{pa}_i \) denotes the parents of \( i \)

\[ P(x| \text{parents of } x) \cdot P(y| \text{parents of } y) \]
Factoring Examples

• Joint distribution

\[ p(x_1, \ldots, x_D) = \prod_{i=1}^{D} p(x_i | \text{pa}_i) \]

where \( \text{pa}_i \) denotes the parents of \( i \)

\[ p(x,y) = p(x)p(y|x) \]

\[ p(x,y) = p(y)p(x|y) \]

\[ p(x,y) = p(x)p(y) \]
Factoring Examples

- We can “read” the form of the joint distribution directly from the directed graph

\[ p(x_1, \ldots, x_D) = \prod_{i=1}^{D} p(x_i | \text{pa}_i) \]

where \( \text{pa}_i \) denotes the parents of \( i \)

\[
P(L| \text{parents of } L) \ P(M| \text{parents of } M) \ P(R| \text{parents of } R)
\]
Factoring Examples

- We can “read” the form of the joint distribution directly from the directed graph

\[ p(x_1, \ldots, x_D) = \prod_{i=1}^{D} p(x_i | pa_i) \]

where \( pa_i \) denotes the parents of \( i \)

\[ P(L,R,M) = P(M) P(L | M) P(R | M) \]
Factoring Examples

- We can “read” the form of the joint distribution directly from a directed graph

\[ p(x_1, \ldots, x_D) = \prod_{i=1}^{D} p(x_i | pa_i) \]

where \( pa_i \) denotes the parents of \( i \)

\[ P(L \mid \text{parents of } L) \ P(M \mid \text{parents of } M) \ P(R \mid \text{parents of } R) \]
Factoring Examples

• We can “read” the form of the joint distribution directly from a directed graph

\[ p(x_1, \ldots, x_D) = \prod_{i=1}^{D} p(x_i | p_a_i) \]

where \( p_a_i \) denotes the parents of \( i \)

Note: \( P(L,R,M) = P(L|R,M)P(R)P(M) \)
Graphical Model Concepts

How about this one?

\[ P(L, M, R, S, T) = \]
Graphical Model Concepts

- How about this one?

\[ P(L,M,R,S,T) = P(S)P(M)P(L|S,M)P(R|M)P(T|L) \]

Note to mention \( P(T|L) + P(T|\sim L) \)
Factoring Examples

- Joint distribution

\[ p(x_1, \ldots, x_D) = \prod_{i=1}^{D} p(x_i | \text{pa}_i) \]

where \( \text{pa}_i \) denotes the parents of \( i \)

What about this one?

No directed cycles
Factoring Examples

• How many probabilities do we have to specify/learn (assuming each $x_i$ is a binary variable)?

if fully connected, we would need $2^7 - 1 = 127$

but, for this connectivity, we need $1 + 1 + 1 + 8 + 4 + 2 + 4 = 21$

Note: If all nodes were independent, we would only need $7!$
Important Case: Time Series

Consider modeling a time series of sequential data \(x_1, x_2, \ldots, x_N\)

These could represent

- locations of a tracked object over time
- observations of the weather each day
- spectral coefficients of a speech signal
- joint angles during human motion
Modeling Time Series

Simplest model of a time series is that all observations are independent.

\[ x_1 \quad x_2 \quad x_3 \quad x_4 \quad \ldots \]

This would be appropriate for modeling successive tosses \{heads,tails\} of an unbiased coin.

However, it doesn’t really treat the series as a sequence. That is, we could permute the ordering of the observations and not change a thing.
Modeling Time Series

In the most general case, we could use chain rule to state that any node is dependent on all previous nodes...

\[ P(x_1, x_2, x_3, x_4, \ldots) = P(x_1)P(x_2|x_1)P(x_3|x_1, x_2)P(x_4|x_1, x_2, x_3) \ldots \]

Look for an intermediate model between these two extremes.
Modeling Time Series

Markov assumption:
\[ P(x_n \mid x_1,x_2,\ldots,x_{n-1}) = P(x_n \mid x_{n-1}) \]
that is, assume all conditional distributions depend only on the most recent previous observation.

The result is a first-order Markov Chain

\[ P(x_1,x_2,x_3,x_4,\ldots) = P(x_1)P(x_2\mid x_1)P(x_3\mid x_2)P(x_4\mid x_3)\ldots \]
Modeling Time Series

Generalization: State-Space Models

You have a Markov chain of latent (unobserved) states

Each state generates an observation

\[ x_1 \rightarrow x_2 \rightarrow \cdots \rightarrow x_{n-1} \rightarrow x_n \rightarrow x_{n+1} \]

\[ y_1 \rightarrow y_2 \rightarrow \cdots \rightarrow y_{n-1} \rightarrow y_n \rightarrow y_{n+1} \]

Goal: Given a sequence of observations, predict the sequence of unobserved states that maximizes the joint probability.
Modeling Time Series

Examples of State Space models
  • Hidden Markov model
  • Kalman filter
Modeling Time Series

\[ P(x_1, x_2, x_3, x_4, \ldots, y_1, y_2, y_3, y_4, \ldots) = \]
\[ P(x_1)P(y_1|x_1)P(x_2|x_1)P(y_2|x_2)P(x_3|x_2)P(y_3|x_3)P(x_4|x_3)P(y_4|x_4)\ldots \]
Example of a Tree-structured Model

Confusion alert: Our textbook uses “w” to denote a world state variable and “x” to denote a measurement. (we have been using “x” to denote world state and “y” as the measurement).
Message Passing
Message Passing: Belief Propagation

- Example: 1D chain

\[ p(x_i) = \sum_{x_1} \ldots \sum_{x_{i-1}} \sum_{x_{i+1}} \ldots \sum_{x_L} p(x_1, \ldots, x_L) \]

- for M-state nodes, cost is \( O(M^L) \)
- exponential in length of chain
- but, we can exploit the graphical structure (conditional independences)

Applicable to both directed and undirected graphs.
Key Idea of Message Passing

multiplication distributes over addition

\[ a \cdot b + a \cdot c = a \cdot (b + c) \]

as a consequence:

\[
\sum_i \sum_j \sum_k a_i b_j c_k = \sum_i \sum_j a_i b_j \left( \sum_k c_k \right) \\
= \sum_i a_i \left[ \sum_j b_j \left( \sum_k c_k \right) \right]
\]
For message passing, this principle is applied to functions of random variables, rather than the variables as done here.
Message Passing

In the next several slides, we will consider an example of a simple, four-variable Markov chain.

\[ P(x_1, x_2, x_3, x_4) = P(x_1) \ P(x_2|x_1) \ P(x_3|x_2) \ P(x_4|x_3) \]
Message Passing

Now consider computing the marginal distribution of variable $x_3$

$$P(x_1, x_2, x_3, x_4) = P(x_1) \ P(x_2|x_1) \ P(x_3|x_2) \ P(x_4|x_3)$$

$$P(x_3) = \sum_{x_1} \sum_{x_2} \sum_{x_4} P(x_1, x_2, x_3, x_4)$$

$$= \sum_{x_1} \sum_{x_2} \sum_{x_4} P(x_1) \ P(x_2|x_1) \ P(x_3|x_2) \ P(x_4|x_3)$$
Message Passing

Multiplication distributes over addition...

\[
P(x_3) = \sum_{x_1} \sum_{x_2} \sum_{x_4} P(x_1) \ P(x_2|x_1) \ P(x_3|x_2) \ P(x_4|x_3)
\]

\[
= \sum_{x_1} \sum_{x_2} P(x_1) \ P(x_2|x_1) \ P(x_3|x_2) \left[ \sum_{x_4} P(x_4|x_3) \right]
\]

\[
= \sum_{x_2} \left[ \sum_{x_1} \left( P(x_1) \ P(x_2|x_1) \right) P(x_3|x_2) \right] \left[ \sum_{x_4} P(x_4|x_3) \right]
\]
Message Passing, aka Forward-Backward Algorithm

Can view as sending/combining messages...

\[
\begin{align*}
M_{\text{Forw}} &= \sum_{x_2} \left( \sum_{x_1} P(x_1) P(x_2|x_1) P(x_3|x_2) \right) \\
M_{\text{Back}} &= \sum_{x_4} P(x_4|x_3)
\end{align*}
\]
Forward-Backward Algorithm

- Express marginals as product of messages evaluated forward from ancestors of \( x_i \) and backwards from descendents of \( x_i \)

\[
p(x_i) = \frac{1}{Z} m_\alpha(x_i)m_\beta(x_i)
\]

- Recursive evaluation of messages

\[
m_\alpha(x_i) = \sum_{x_{i-1}} \psi(x_{i-1}, x_i)m_\alpha(x_{i-1})
\]

\[
m_\beta(x_i) = \sum_{x_{i+1}} \psi(x_i, x_{i+1})m_\beta(x_{i+1})
\]

- Find \( Z \) by normalizing \( p(x_i) \)

Works in both directed and undirected graphs

---

Christopher Bishop, MSR
Specific numerical example

Note: these are conditional probability tables, so values in each row must sum to one.
Specific numerical example

\[
\begin{pmatrix}
.4 & .6 \\
.5 & .5
\end{pmatrix}
\begin{pmatrix}
.8 & .2 \\
.2 & .8
\end{pmatrix}
\begin{pmatrix}
.5 & .5 \\
.7 & .3
\end{pmatrix}
\]

Note: these are conditional probability tables, so values in each row must sum to one.

Interpretation

\[
P(x_1, x_2, x_3, x_4) = P(x_1) \cdot P(x_2|x_1) \cdot P(x_3|x_2) \cdot P(x_4|x_3)
\]

\[
P(x_2|x_1) = \begin{pmatrix}
.4 & .6 \\
.5 & .5
\end{pmatrix}
\]

\[
P(x_2=2|X_1=1) = 0.6
\]
Specific numerical example

Sample computations:

\[ P(x_1=1, x_2=1, x_3=1, x_4=1) = (0.7)(0.4)(0.8)(0.5) = 0.112 \]

\[ P(x_1=2, x_2=1, x_3=2, x_4=1) = (0.3)(0.5)(0.2)(0.7) = 0.021 \]
### Specific numerical example

#### Joint Probability, represented in a truth table

<table>
<thead>
<tr>
<th>x1</th>
<th>x2</th>
<th>x3</th>
<th>x4</th>
<th>P(x1,x2,x3,x4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.1120</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.1120</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0392</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0168</td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0420</td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.0420</td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.2352</td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.1008</td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0600</td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.0600</td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0210</td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0090</td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0150</td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0840</td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0360</td>
</tr>
</tbody>
</table>
Specific numerical example

Joint Probability

<table>
<thead>
<tr>
<th></th>
<th>x1</th>
<th>x2</th>
<th>x3</th>
<th>x4</th>
<th>P(x1,x2,x3,x4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.1120</td>
<td></td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.1120</td>
<td></td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0392</td>
<td></td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0420</td>
<td></td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.0420</td>
<td></td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.2352</td>
<td></td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.1008</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0600</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.0600</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0210</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0090</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0150</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.0150</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0840</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0360</td>
<td></td>
</tr>
</tbody>
</table>

Compute marginal of x3:

P(x3=1) = 0.458
P(x3=2) = 0.542
Specific numerical example now compute via message passing

\[
P(x_1)
\begin{pmatrix}
.4 & .6 \\
.5 & .5
\end{pmatrix}
P(x_2|x_1)
\begin{pmatrix}
.8 & .2 \\
.2 & .8
\end{pmatrix}
P(x_3|x_2)
\begin{pmatrix}
.5 & .5 \\
.7 & .3
\end{pmatrix}
P(x_4|x_3)
\]

\[
P(x_3) = \sum_{x_2} \left( \sum_{x_1} P(x_1) P(x_2|x_1) \right) P(x_3|x_2) \left( \sum_{x_4} P(x_4|x_3) \right)
\]

\[
\sum_{x_1} \begin{pmatrix}
.7 & .4 \\
.3 & .5
\end{pmatrix} P(x_1=1)P(x_2=1|x_1=1)
\begin{pmatrix}
.7 & .6 \\
.3 & .5
\end{pmatrix} P(x_1=1)P(x_2=2|x_1=1)
\]

\[
\begin{pmatrix}
.7 & .5 \\
.3 & .5
\end{pmatrix} P(x_1=2)P(x_2=1|x_1=2)
\begin{pmatrix}
.7 & .6 \\
.3 & .5
\end{pmatrix} P(x_1=2)P(x_2=2|x_1=2)
\]

\[
(0.43) + (0.57)
\]

0.43 0.57
Specific numerical example now compute via message passing

\[
P(x_1) = \begin{pmatrix} 0.7 & 0.3 \end{pmatrix}
\]

\[
P(x_2|x_1) = \begin{pmatrix} 0.4 & 0.6 \\ 0.5 & 0.5 \end{pmatrix}
\]

\[
P(x_3|x_2) = \begin{pmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{pmatrix}
\]

\[
P(x_4|x_3) = \begin{pmatrix} 0.5 & 0.5 \\ 0.7 & 0.3 \end{pmatrix}
\]

\[
P(x_3) = \sum_{x_2} \left( \sum_{x_1} \left( P(x_1) P(x_2|x_1) \right) P(x_3|x_2) \right) \sum_{x_4} P(x_4|x_3)
\]

\[
\sum_{x_1} \begin{pmatrix} .7 & .4 \\ .3 & .5 \end{pmatrix} \begin{pmatrix} .4 & .6 \\ .5 & .5 \end{pmatrix} = \begin{pmatrix} .43 & .57 \end{pmatrix}
\]

simpler way to compute this...

\[
\begin{pmatrix} .7 & .3 \\ .5 & .5 \end{pmatrix} = \begin{pmatrix} .43 & .57 \end{pmatrix}
\]

i.e. matrix multiply can do the combining and marginalization all at once!!!!
Specific numerical example

Now compute via message passing

\[
P(x_1) = \begin{pmatrix} .7 & .3 \end{pmatrix}
\]

\[
P(x_2|x_1) = \begin{pmatrix} .4 & .6 \\ .5 & .5 \end{pmatrix}
\]

\[
P(x_3|x_2) = \begin{pmatrix} .8 & .2 \\ .2 & .8 \end{pmatrix}
\]

\[
P(x_4|x_3) = \begin{pmatrix} .5 & .5 \\ .7 & .3 \end{pmatrix}
\]

\[
P(x_3) = \sum_{x_2} \left( \sum_{x_1} \left( \begin{pmatrix} .7 & .3 \end{pmatrix} \begin{pmatrix} .4 & .6 \\ .5 & .5 \end{pmatrix} \begin{pmatrix} .8 & .2 \\ .2 & .8 \end{pmatrix} \right) \right) \begin{pmatrix} .5 & .5 \\ .7 & .3 \end{pmatrix}
\]

\[
= \begin{pmatrix} .43 & .57 \end{pmatrix} \begin{pmatrix} .8 & .2 \\ .2 & .8 \end{pmatrix}
\]

\[
= \begin{pmatrix} .458 & .542 \end{pmatrix}
\]
Specific numerical example

Now compute via message passing

\[ P(x1) = \begin{pmatrix} .7 & .3 \\ .5 & .5 \end{pmatrix} \]

\[ P(x2|x1) = \begin{pmatrix} .4 & .6 \\ .5 & .5 \end{pmatrix} \]

\[ P(x3|x2) = \begin{pmatrix} .8 & .2 \\ .2 & .8 \end{pmatrix} \]

\[ P(x4|x3) = \begin{pmatrix} .5 & .5 \\ .7 & .3 \end{pmatrix} \]

\[ P(x3) = \sum_{x2} \left( \sum_{x1} P(x1) P(x2|x1) P(x3|x2) \right) \sum_{x4} P(x4|x3) \]

Compute sum along rows of \( P(x4|x3) \)

Can also do that with matrix multiply:

\[ \begin{pmatrix} .5 & .5 \\ .7 & .3 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \]

\[ = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \]

Note: this is not a coincidence
Message Passing

Can view as sending/combining messages...

\[
\begin{bmatrix}
M_{\text{Forw}} & \ast \\
M_{\text{Back}}
\end{bmatrix}
\]

= \begin{bmatrix} 0.458 & 0.542 \end{bmatrix}

= \begin{bmatrix} 1 \\ 1 \end{bmatrix}

Belief that x3=1, from front part of chain

Belief that x3=2, from front part of chain

Belief that x3=1, from back part of chain

Belief that x3=2, from back part of chain

How to combine them?
Message Passing

Can view as sending/combining messages...

\[
p(x_i) = \frac{1}{Z} m_\alpha(x_i) m_\beta(x_i)
\]

\[
= [ (.458)(1) \quad (.542)(1) ]
= [.458 \quad .542]
\]

These are the same values for the marginal P(x3) that we computed from the raw joint probability table. Whew!!!
Specific numerical example

If we want to compute all marginals, we can do it in one shot by cascading, for a big computational savings.

We need one cascaded forward pass, one separate cascaded backward pass, then a combination and normalization at each node.
Specific numerical example

Then combine each by elementwise multiply and normalize.
Specific numerical example

\[ \begin{bmatrix} 0.7 & 0.3 \end{bmatrix} \]

\[ \begin{bmatrix} 0.4 & 0.6 \\
0.5 & 0.5 \end{bmatrix} \begin{bmatrix} 0.8 & 0.2 \\
0.2 & 0.8 \end{bmatrix} \begin{bmatrix} 0.5 & 0.5 \\
0.7 & 0.3 \end{bmatrix} \]

forward pass

backward pass

Then combine each by elementwise multiply and normalize

Forward:

\[ \begin{bmatrix} 0.7 & 0.3 \end{bmatrix} \begin{bmatrix} 0.43 & 0.57 \\
0.458 & 0.542 \end{bmatrix} \begin{bmatrix} 0.6084 & 0.3916 \end{bmatrix} \]

Backward:

\[ \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \end{bmatrix} \]

combined+ normalized

\[ \begin{bmatrix} 0.7 & 0.3 \end{bmatrix} \begin{bmatrix} 0.43 & 0.57 \\
0.458 & 0.542 \end{bmatrix} \begin{bmatrix} 0.6084 & 0.3916 \end{bmatrix} \]
Specific numerical example

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.1120</td>
<td></td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.1120</td>
<td></td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0392</td>
<td></td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0168</td>
<td></td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0420</td>
<td></td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.0420</td>
<td></td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.2352</td>
<td></td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.1008</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0600</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.0600</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0210</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0090</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0150</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.0150</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0840</td>
<td></td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0360</td>
<td></td>
</tr>
</tbody>
</table>

num truth table entries = 16
Computation using joint prob table took 0.062000 sec
marginal x1: 0.700000 0.300000
marginal x2: 0.430000 0.570000
marginal x3: 0.458000 0.542000
marginal x4: 0.608400 0.391600
Computation using BP sum-product took 0.000000 sec
marginal x1: 0.700000 0.300000
marginal x2: 0.430000 0.570000
marginal x3: 0.458000 0.542000
marginal x4: 0.608400 0.391600
Specific numerical example

Note: In this example, a directed Markov chain using true conditional probabilities (rows sum to one), only the forward pass is needed. This is true because the backward pass sums along rows, and always produces $[1 \ 1]'$.

We didn’t really need forward AND backward in this example.

Forward: $\begin{bmatrix} .7 & .3 \\ .43 & .57 \\ .458 & .542 \end{bmatrix}$ $\begin{bmatrix} .6084 & .3916 \end{bmatrix}$

Backward: $\begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix}$

combined+ normalized $\begin{bmatrix} .7 & .3 \\ .43 & .57 \\ .458 & .542 \end{bmatrix}$ $\begin{bmatrix} .6084 & .3916 \end{bmatrix}$

these are already the marginals for this example

Did’t need to do these steps
Max Marginals

What if we want to know the most probably state (mode of the distribution)? Since the marginal distributions can tell us which value of each variable yields the highest marginal probability (that is, which value is most likely), we might try to just take the argmax of each marginal distribution.

Although that's correct in this example, it isn't always the case.
Max Marginals can Fail to Find the MAP

However, the max marginals find most likely values of each variable treated individually, which may not be the combination of values that jointly maximize the distribution.

max marginals:  
\[ w_1=4, \; w_2=2 \]

actual MAP solution:  
\[ w_1=2, \; w_2=4 \]
Max-product Algorithm

• Goal: find

\[ x^{\text{MAP}} = \arg \max_x p(x) \]

  – define the “max marginal”

\[ M(x_i) = \max_{x_1} \cdots \max_{x_{i-1}} \max_{x_{i+1}} \cdots \max_{x_L} p(x_1, \ldots, x_L) \]

  – then

\[ x_i^{\text{MAP}} = \arg \max_{x_i} \phi(x_i) \]

• Message passing algorithm with “sum” replaced by “max”

• Generalizes to any two operations forming a semiring
Computing MAP Value

\[
M(x_i) = \max_{x_1} \cdots \max_{x_{i-1}} \max_{x_{i+1}} \cdots \max_{x_L} p(x_1, \ldots, x_L)
\]

Can solve using message passing algorithm with “sum” replaced by “max”.

In our chain, we start at the end and work our way back to the root (x1) using the max-product algorithm, keeping track of the max value as we go.

\[
\max_i (a_i \max_j (b_j \max_k (c_k)))
\]
Specific numerical example  

\[
P(x1) \begin{pmatrix} .4 & .6 \\ .5 & .5 \end{pmatrix} P(x2|x1) \begin{pmatrix} .8 & .2 \\ .2 & .8 \end{pmatrix} P(x3|x2) \begin{pmatrix} .5 & .5 \\ .7 & .3 \end{pmatrix} P(x4|x3)
\]

\[M(x3) = \max_{x2} \left( \max_{x1} \left( P(x1) \cdot P(x2|x1) \cdot P(x3|x2) \cdot \max_{x4} P(x4|x3) \right) \right)\]

max marginal

\[\max_{x1} \left( \max_{x2} \right) \begin{pmatrix} .7 & .4 \\ .3 & .5 \end{pmatrix} \begin{pmatrix} .7 & .6 \\ .3 & .5 \end{pmatrix} \]

\[\max[0.28, 0.42]\]
Specific numerical example

\[ P(x1) = \begin{pmatrix} .7 & .3 \\ .5 & .5 \end{pmatrix} \]

\[ P(x2|x1) = \begin{pmatrix} .4 & .6 \\ .2 & .8 \end{pmatrix} \]

\[ P(x3|x2) = \begin{pmatrix} .8 & .2 \\ .7 & .3 \end{pmatrix} \]

\[ P(x4|x3) = \begin{pmatrix} .5 & .5 \\ .2 & .8 \end{pmatrix} \]

\[ M(x3) = \max_{x2} \left( \max_{x1} \left( P(x1) P(x2|x1) P(x3|x2) \right) \right) \]

\[ \max_{x4} P(x4|x3) \]

\[ M(x3) = \max_{x2} \left( \max_{x1} \left( P(x1) P(x2|x1) P(x3|x2) \right) \right) \]

\[ \begin{pmatrix} .28 & .42 \end{pmatrix} \]

\[ \begin{pmatrix} .8 & .2 \\ .2 & .8 \end{pmatrix} \]

\[ \max_{x2} \begin{pmatrix} (0.28)(0.8) & (0.28)(0.2) \\ (0.42)(0.2) & (0.42)(0.8) \end{pmatrix} \]

\[ = \begin{pmatrix} .224 & .336 \end{pmatrix} \]

Note that this is no longer matrix multiplication, since we are not summing down the columns but taking max instead...
Specific numerical example

\[ P(x1) = \begin{pmatrix} .4 & .6 \\ .5 & .5 \end{pmatrix} \]

\[ P(x2|x1) = \begin{pmatrix} .8 & .2 \\ .2 & .8 \end{pmatrix} \]

\[ P(x3|x2) = \begin{pmatrix} .5 & .5 \\ .7 & .3 \end{pmatrix} \]

\[ P(x4|x3) \]

\[ M(x3) = \max_{x2} \left( \max_{x1} \begin{pmatrix} P(x1) & P(x2|x1) \end{pmatrix} P(x3|x2) \right) \max_{x4} P(x4|x3) \]

\[ \begin{pmatrix} .28 & .42 \end{pmatrix} \]

\[ \begin{pmatrix} (28)(.8) & (28)(.2) \\ (.42)(.2) & (.42)(.8) \end{pmatrix} \]

\[ = \begin{pmatrix} .224 & .336 \end{pmatrix} \]

\[ \max_{x4} \begin{pmatrix} .5 & .5 \\ .7 & .3 \end{pmatrix} = \begin{pmatrix} .5 \\ .7 \end{pmatrix} \]
Specific numerical example

**max product**

**message passing**

\[
P(x1) = \begin{pmatrix} .4 & .6 \\ .5 & .5 \end{pmatrix} \quad P(x2|x1) = \begin{pmatrix} .8 & .2 \\ .2 & .8 \end{pmatrix} \quad P(x3|x2) = \begin{pmatrix} .5 & .5 \\ .7 & .3 \end{pmatrix} \quad P(x4|x3) = \begin{pmatrix} .7 \end{pmatrix}
\]

\[
M(x3) = \max_{x2} \left( \max_{x1} \left( P(x1) P(x2|x1) \right) P(x3|x2) \right) \max_{x4} P(x4|x3)
\]

Interpretation:
The mode of the joint distribution is .2352, and the value of variable \(x3\), in the configuration that yields the mode value, is 2.
### Specific numerical example

**Joint Probability, represented in a truth table**

<table>
<thead>
<tr>
<th>x1</th>
<th>x2</th>
<th>x3</th>
<th>x4</th>
<th>P(x1,x2,x3,x4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.1120</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.1120</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0392</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0168</td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0420</td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.2352</td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0420</td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.1600</td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.0600</td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0210</td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0090</td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0150</td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.0150</td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0849</td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0360</td>
</tr>
</tbody>
</table>

Largest value of joint prob = mode = MAP

Interpretation:
The mode of the joint distribution is 0.2352, and the value of variable x3, in the configuration that yields the mode value, is 2.
Computing Arg-Max of MAP Value

\[ x_i^{\text{MAP}} = \arg \max_{x_i} \phi(x_i) \]

Chris Bishop, PRML:
“At this point, we might be tempted simply to continue with the message passing algorithm [sending forward-backward messages and combining to compute argmax for each variable node]. However, because we are now maximizing rather than summing, it is possible that there may be multiple configurations of \(x\) all of which give rise to the maximum value for \(p(x)\). In such cases, this strategy can fail because it is possible for the individual variable values obtained by maximizing the product of messages at each node to belong to different maximizing configurations, giving an overall configuration that no longer corresponds to a maximum. The problem can be resolved by adopting a rather different kind of message passing...”

Essentially, the solution is to write a dynamic programming algorithm based on max-product.
Specific numerical example: MAP Estimate

\[
P(x_1) = \begin{pmatrix} .7 & .3 \end{pmatrix}
\]
\[
P(x_2|x_1) = \begin{pmatrix} .4 & .6 \\ .5 & .5 \end{pmatrix}
\]
\[
P(x_3|x_2) = \begin{pmatrix} .8 & .2 \\ .2 & .8 \end{pmatrix}
\]
\[
P(x_4|x_3) = \begin{pmatrix} .5 & .5 \\ .7 & .3 \end{pmatrix}
\]

DP State Space Trellis
Specific numerical example: MAP Estimate

\[
P(x_1) = \begin{pmatrix} .7 & .3 \end{pmatrix} \quad P(x_2|x_1) = \begin{pmatrix} .4 & .6 \\ .5 & .5 \end{pmatrix} \quad P(x_3|x_2) = \begin{pmatrix} .8 & .2 \\ .2 & .8 \end{pmatrix} \quad P(x_4|x_3) = \begin{pmatrix} .5 & .5 \\ .7 & .3 \end{pmatrix}
\]

DP State Space Trellis
Specific numerical example

Joint Probability, represented in a truth table

<table>
<thead>
<tr>
<th>x1</th>
<th>x2</th>
<th>x3</th>
<th>x4</th>
<th>P(x1,x2,x3,x4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.1120</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.1120</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0392</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0168</td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0420</td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.0420</td>
</tr>
<tr>
<td>**1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td><strong>0.2352</strong></td>
</tr>
<tr>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.1008</td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0600</td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.0600</td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0210</td>
</tr>
<tr>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0090</td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0150</td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>2.0000</td>
<td>0.0150</td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>1.0000</td>
<td>0.0840</td>
</tr>
<tr>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.0360</td>
</tr>
</tbody>
</table>

Largest value of joint prob = mode = MAP achieved for x1=1, x2=2, x3=2, x4=1
Belief Propagation Summary

- Definition can be extended to general tree-structured graphs
- Works for both directed AND undirected graphs
- Efficiently computes marginals and MAP configurations
- At each node:
  - form product of *incoming* messages and local evidence
  - marginalize to give *outgoing* message
  - one message in each direction across every link
- Gives exact answer in any acyclic graph (no loops).

Christopher Bishop, MSR
Loopy Belief Propagation

- BP applied to graph that contains loops
  - needs a propagation “schedule”
  - needs multiple iterations
  - might not converge

- Typically works well, even though it isn’t supposed to

- State-of-the-art performance in error-correcting codes