Outline

- Continuation of Inference in Belief Networks
- Automated Belief propagation in PolyTrees
d-separation: Direction-Dependent Separation

• **Network construction**
  – Conditional independence of a node and its predecessors, given its parents
  – The absence of a link between two variables does not guarantee their *independence*

• **Effective inference needs to exploit all available conditional independences**
  – Which set of nodes X are conditionally independent of another set Y, given a set of evidence nodes E
    • \( P(X,Y|E) = P(X|E) \cdot P(Y|E) \)
    – Limits propagation of information
    – Comes directly from structure of network

---

**d-separation**

**Definition:** If X, Y and E are three disjoint subsets of nodes in a DAG, then E is said to **d-separate** X from Y if every undirected path from X to Y is **blocked** by E. A path is blocked if it contains a node Z such that:

1. Z has one incoming and one outgoing arrow; or
2. Z has two outgoing arrows; or
3. Z has two incoming arrows and neither Z nor any of its descendants is in E.
• **Property of belief networks**: if X and Y are d-separated by E, then X and Y are conditionally independent given E.

• An “if-and-only-if” relationship between the graph and the probabilistic model cannot always be achieved.
Whether there is Gas in the car and whether the car Radio plays are independent given evidence about whether the SparkPlugs fire [ignition] (case 1).

\[
P(R,G/I) = P(R/I) \cdot P(G/I) \\
P(G/I,R) = P(G/I)
\]

Gas and Radio are conditionally-independent if it is known if the battery works (case 2).

\[
P(R/B,G) = P(R/B); P(G/B,R)=P(G/B)
\]
Gas and Radio are independent given no evidence at all. But they are dependent given evidence about whether the car Starts. For example, if the car does not start, then the radio playing is increased evidence that we are out of gas. Gas and Radio are also dependent given evidence about whether the car Moves, because that is enabled by the car starting.

\[
P(\text{Gas}/\text{Radio}) = P(\text{Gas}); \quad P(\text{Radio}/\text{Gas}) = P(\text{Radio})
\]

\[
P(\text{Gas}/ \text{Radio, Start}) \neq P(\text{Gas}/\text{Start})
\]

Inference in Belief Networks

- BNs are fairly expressive and easily engineered representation for knowledge in probabilistic domains.
- They facilitate the development of inference algorithms.
- They are particularly suited for parallelization.
- Current inference algorithms are efficient and can solve large real-world problems.
Network Features Affect Reasoning

- Topology (trees, singly-connected, sparsely-connected, DAGs).
- Size (number of nodes).
- Type of variables (discrete, cont, functional, noisy-logical, mixed).
- Network dynamics (static, dynamic).

Belief Propagation in Polytrees

Polytree belief network, where nodes are singly connected
- Exact inference, Linear in size of network

Multiconnected belief network. This is a DAG, but not a polytree.
- Exact inference, Worst case NP-hard
Simple examples of 4 patterns of reasoning that can be handled by belief networks. \( E \) represents an evidence variable; \( Q \) is a query variable.

\[ P(Q/E) =? \]

Belief Network Calculation in Polytree: Evidence Above

- What is \( p(Y_5|Y_1,Y_4) \)
  - Define in terms of CPTs = \( p(Y_5,Y_4,Y_3,Y_2,Y_1) \)
  - \( p(Y_5|Y_3,Y_4)p(Y_4), p(Y_3|Y_1,Y_2), p(Y_2), p(Y_1) \)
  - \( p(Y_5|Y_1,Y_4) = p(Y_5,Y_1,Y_4)/p(Y_1,Y_4) \)
  - Use cpt to sum over missing variables
  - \( p(Y_5,Y_1,Y_4) = \text{Sum}(Y_2,Y_3) p(Y_5,Y_4,Y_3,Y_2,Y_1) \)
  - assuming variables take on only truth or falsity.

- \( p(Y_5|Y_1,Y_4) = p(Y_5,Y_3,Y_1,Y_4) + p(Y_5, \text{not } Y_3|Y_1,Y_4) \)
  - Connect to parents of \( Y_5 \) not already part of expression, by marginalization

- \( = \text{SUM}(Y_3) p(Y_5,Y_3|Y_1,Y_4) \)
Continuation of Example Above

1. \(= \text{SUM}(Y3)(p(Y5|Y3, Y1, Y4) \cdot p(Y3|Y1, Y4))\)
   - \(p(s_i s_j|d) = p(s_i|s_j,d)p(s_j|d)\)

2. \(= \text{SUM}(Y3) \cdot p(Y5|Y3, Y4) \cdot p(Y3|Y1, Y4)\)
   - \(Y1\) conditionally independent of \(Y5\) given \(Y3\),
   - \(Y3\) represents all the contributions of \(Y1\) to \(Y5\),
   - Case 1: a node is conditionally independent of non-descendants given its parents

3. \(= \text{SUM}(Y3) \cdot p(Y5|Y3, Y4) \cdot p(Y3|Y1)\)
   - \(Y4\) conditionally independent of \(Y3\) given \(Y1\)
   - Case 3: \(Y3\) not a descendant of \(Y5\) which d-separates \(Y1\) and \(Y4\)

Continuation of Example Above

1. \(= \text{SUM}(Y3) \cdot p(Y5|Y3, Y4) \cdot \text{SUM}(Y2)p(Y3|Y1,Y2) \cdot p(Y2)\)
   - \(Y2\) independent of \(Y1\); \(p(Y2|Y1) = p(Y2)\)
   - Definition of Baysean network
Belief Network Calculation in Polytree: Evidence Below

- What is $p(Y_1|Y_5)$
  - $p(Y_1|Y_5) = \frac{p(Y_1, Y_5)}{p(Y_5)}$
  - $p(Y_1, Y_2, Y_3, Y_4, Y_5) = \text{in terms of cpt}$
  - $p(Y_5|Y_3, Y_4)p(Y_3|Y_1, Y_2)p(Y_1)p(Y_2)p(Y_4)$

- $p(Y_1|Y_5) = \frac{p(Y_5|Y_1)p(Y_1)}{p(Y_5)}$
  - Bayes Rule

- $= K \times p(Y_5|Y_1)p(Y_1)$

Continuation of Example Below

- $= K \times p(Y_5|Y_1)p(Y_1)$

- $= K \times (\text{SUM}(Y_3) \, p(Y_5|Y_3)p(Y_3|Y_1)) \, p(Y_1)$
  - Connect to $Y_3$ parent of $Y_5$ not already part of expression
  - $P(s_i|s_j) = \text{SUM}(d)P(s_i|s_j, d) \, P(d|s_j)$
  - $Y_1$ conditionally independent of $Y_5$ given $Y_3$
  - $p(Y_5|Y_3, Y_1) = p(Y_5|Y_3)$

- $= K \times (\text{SUM}(Y_3) \, (\text{SUM}(Y_4)p(Y_5|Y_3, Y_4)p(Y_4|Y_3))p(Y_3|Y_1)) \, p(Y_1)$
  - Connect to $Y_4$ parent of $Y_5$ not already part of expression
  - $P(s_i|s_j) = \text{SUM}(d)P(s_i|s_j, d) \, P(d|s_j)$

- $= K \times (\text{SUM}(Y_3) \, (\text{SUM}(Y_4)p(Y_5|Y_3, Y_4)p(Y_4))p(Y_3|Y_1)) \, p(Y_1)$
  - $Y_4$ independent of $Y_3$; $p(Y_4|Y_3) = p(Y_4)$
Continuation of Example Below

- \[ = K \ast (\text{SUM}(Y3) (\text{SUM}(Y4)p(Y5|Y3,Y4)p(Y4))p(Y3|Y1)) p(Y1) \]
- \[ = K \ast (\text{SUM}(Y3) (\text{SUM}(Y4)p(Y5|Y3,Y4)p(Y4))\text{SUM}(Y2)p(Y3|Y1,Y2)p(Y2))) p(Y1) \]
  - Connect to Y2 parent of Y3 not already part of expression
  - \[ P(s_i | s_j) = \text{SUM}(d)P(s_i | s_j, d) P(d | s_j) \]

- \[ = K \ast (\text{SUM}(Y3) (\text{SUM}(Y4)p(Y5|Y3,Y4)p(Y4))\text{SUM}(Y2)p(Y3|Y1,Y2)p(Y2))) p(Y1) \]
  - Y2 independent of Y1
  - Expression that can be calculated from cpt

Variable Elimination

- Can remove a lot of re-calculation/multiplications in expression

- K \ast (\text{SUM}(Y3) (\text{SUM}(Y4)p(Y5|Y3,Y4)p(Y4))\text{SUM}(Y2)p(Y3|Y1,Y2)p(Y2))) p(Y1)

- Summations over each variable are done only for those portions of the expression that depend on variable

- Save results of inner summing to avoid repeated calculation
  - Create Intermediate Functions
  - F-Y2(Y3,Y1)= (SUM(Y2)p(Y3|Y1,Y2)p(Y2))
Evidence Above and Below for Polytrees

If there is evidence both above and below $P(Y_3 \mid Y_5, Y_2)$ we separate the evidence into above, $\mathcal{E}^+$, and below, $\mathcal{E}^-$, portions and use a version of Bayes’ rule to write

$$p(Q \mid \mathcal{E}^+, \mathcal{E}^-) = \frac{p(\mathcal{E}^- \mid Q, \mathcal{E}^+) p(Q \mid \mathcal{E}^+)}{p(\mathcal{E}^- \mid \mathcal{E}^+)}$$

We treat $\frac{1}{p(\mathcal{E}^- \mid \mathcal{E}^+)} = k_2$ as a normalizing factor and write

$$p(Q \mid \mathcal{E}^+, \mathcal{E}^-) = k_2 p(\mathcal{E}^+ \mid \mathcal{E}^-) p(Q \mid \mathcal{E}^-)$$

$Q$ d-separates $\mathcal{E}^-$ from $\mathcal{E}^+$, so

$$p(Q \mid \mathcal{E}^+, \mathcal{E}^-) = k_2 p(\mathcal{E}^+ \mid Q) p(Q \mid \mathcal{E}^-)$$

We calculate the first probability in this product as part of the top-down procedure for calculating $p(Q \mid \mathcal{E}^-)$. The second probability is calculated directly by the bottom-up procedure.

Other types of queries

- Most probable explanation (MPE) or most likely hypothesis: The instantiation of all the remaining variables $U$ with the highest probability given the evidence

  $$\text{MPE}(U \mid e) = \arg\max_u P(u, e)$$

- Maximum a posteriori (MAP): The instantiation of some variables $V$ with the highest probability given the evidence

  $$\text{MAP}(V \mid e) = \arg\max_v P(v, e)$$

  Note that the assignment to $A$ in $\text{MAP}(A \mid e)$ might be completely different from the assignment to $A$ in $\text{MAP}(\{A, B\} \mid e)$.
  - sum over values of $B$ vs individual values of $B$

- Other queries: probability of an arbitrary logical expression over query variables, decision policies, information value, seeking evidence, information gathering planning, etc.
Incremental Updating of BN: Pearl’s message passing algorithm

**Notation:**

\[ M_{y|x} \]  
Conditional probability matrix

\[ e \]  
The evidence

\[ \text{Bel}(x) = P(x | e) \]  
Posterior distribution of \( x \)

\[ f(x) \cdot M_{y|x} = \sum_x f(x) M_{y|x} \]

---

Simple chains

\[ e^+ \rightarrow T \rightarrow \ldots \rightarrow U \rightarrow X \rightarrow Y \rightarrow \ldots \rightarrow e^- \]

\( e = \{ e^+, e^- \} \)

\( e^+ \) Represents the “causal” evidence

\( e^- \) Represents the “evidential” evidence

Need to compute \( \text{Bel}(x) \)
\( Bel(x) = P(x \mid e^+ e^-) \)
\[
= \frac{P(e^- \mid x e^+) P(x \mid e^+)}{P(e^- \mid e^+)} \quad \text{Bayes rule}
\]
\[
= \alpha P(e^- \mid x e^+) P(x \mid e^+) \quad \text{Normalization}
\]
\[
= \alpha P(e^- \mid x) P(x \mid e^+) \quad x \text{ d-sep } e^+ e^-
\]
\[
= \alpha \cdot \lambda(x) \cdot \pi(x)
\]

**The \( \lambda(x) \) and \( \pi(x) \) Messages**

\( \lambda(x) \) represents the degree to which \( x \) might explain the evidential support. -- \( P(e^-/X) \)

\( \pi(x) \) represents the direct causal support for \( x \). -- \( P(X/e^+) \)

Both \( \lambda(x) \) and \( \pi(x) \) can be calculated in terms of the \( \lambda \) and \( \pi \) values of the neighbors of \( x \).
Computing $\lambda(x)$ based on $\lambda(y)$

\[ \lambda(x) = P(e^- | x) \]
\[ = \sum_{y} P(e^- | x, y) P(y | x) \]
\[ = \sum_{y} P(e^- | y) P(y | x) \]
\[ = \sum_{y} \lambda(y) P(y | x) \]
\[ = \lambda(y) \cdot M_{y|x} \]

Computing $\pi(x)$ based on $\pi(u)$

\[ \pi(x) = P(x | e^+) \]
\[ = \sum_{u} P(x | u, e^+) P(u | e^+) \]
\[ = \sum_{u} P(x | u) P(u | e^+) \]
\[ = \sum_{u} P(x | u) \pi(u) \]
\[ = \pi(u) \cdot M_{x|u} \]
**Update scheme for chains**

\[ e^+ \rightarrow T \rightarrow \cdots \rightarrow U \rightarrow X \rightarrow Y \rightarrow \cdots \rightarrow e^- \]

\[ \pi(u) \rightarrow M_{x|u} \rightarrow \pi(x) \]

\[ \lambda(x) \leftarrow M_{y|x} \leftarrow \lambda(y) \]

**Belief Propagation in Trees**

- Each node must combine the impact of \( \lambda \)-messages from several children.
- Each node must distribute a separate \( \pi \)-message to each child.
Propagation in Polytrees

Decomposing the evidence

The evidence $E$ can be decomposed into two subsets:

- $E_i^+$, the subset of $E$ that can be accessed from $X_i$ through its parents.
- $E_i^-$, the subset of $E$ that can be accessed from $X_i$ through its children.
Parameters:

The current strength of the causal support, $\pi$, contributed by each incoming link $U_i \rightarrow x$:

$$\pi_x(U_i) = P(U_i \mid e_{uix}^+)$$

The current strength of the diagnostic support, $\lambda$, contributed by each outgoing link $x \rightarrow Y_j$:

$$\lambda y_j(x) = P(e_{xyj}^- \mid x)$$

The fixed conditional probability matrix

$$P(x \mid u_1, K, u_n))$$

Propagation Process

Step 1: Belief updating: Inspect msgs from parents & children and compute:

$$Bel(x) = \alpha \lambda(x) \pi(x) \text{ where :}$$

$$\lambda(x) = \prod_j \lambda y_j(x)$$

$$\pi(x) = \sum_{u_1, K, u_n} P(x \mid u_1 K, u_n) \prod_i \pi_x(u_i)$$

Step 2: Bottom-up propagation: Compute $\lambda_x$ msgs to send up.

$\lambda_x(U_i)$ is the msg $X$ sends to parent $U_i$.

$$\lambda_x = \beta \sum_x \lambda(x) \sum_{U_i = k \neq i} P(x \mid U_1 \ldots U_n) \prod_{k \neq i} \pi_x(U_k)$$

$\beta$ is an arbitrary constant (factor out contributions to bel(x) from $U_i$)
Step 3: Top-Down Propagation:

Compute $\pi$ msgs to send down.

$\pi_{Y_j}(x)$ is sent from $x$ to child $Y_j$

$\pi_{Y_j}(x) = \alpha \left[ \prod \lambda_{Y_k}(x) \right] \sum \limits_{u_i \neq u_j} P(x \mid U_1 \ldots U_n) \prod \pi_i(U_i)$

$= \alpha \cdot \frac{\text{Bel}(x)}{\lambda_{Y_j}(x)}$ (factor out contributions to bel(x) from $Y_j$)

Boundary Conditions:

1. Root $\Rightarrow \pi(x)$ is the prior prob. dist.
2. Childless node $\Rightarrow \lambda(x) = (1, \ldots, 1)$
3. Evidence node $\Rightarrow \lambda(x) = (0, \ldots, 1, \ldots, 0)$
• Approximate inference techniques

• Alternative approaches to uncertain reasoning