Bayesian Networks with Loops

- A direct application of sum-product can be problematic
- Can be converted to a junction tree, size can be exponential
- Focus on approximate inference techniques:
  - Stochastic inference, based on sampling
  - Deterministic inference, based on approximations
Inference by Stochastic Simulation

- **Basic idea:**
  - Draw \( N \) samples from a sampling distribution
  - Compute an approximate posterior probability \( P' \)
  - Show this converges to the true probability \( P \)

- **Sampling approaches:**
  - Sampling from an empty network
  - Rejection sampling
  - Likelihood weighting
  - Markov chain Monte Carlo (MCMC)
Sampling from an empty network

Consider a Bayesian Network $P(X_1,\ldots,X_n)$

The joint distribution factorizes as

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

For $i = 1,\ldots,n$

Assume $\text{Parents}(X_i)$ have been instantiated

Draw a sample $x_i$ following $P(X_i|\text{Parents}(X_i))$

$(x_1,\ldots,x_n)$ forms a sample from the Bayesian Network
Example

Cloudy

Sprinkler

Rain

Wet Grass

P(C) = 0.5

P(R|C) = 0.8

P(S|C) = 0.2

P(C) = 0.50

P(S) = 0.5

P(R) = 0.8

P(S) = 0.2

P(W|S,R) =

| S | R | P(W|S,R) |
|---|---|----------|
| T | T | 0.99     |
| T | F | 0.90     |
| F | T | 0.90     |
| F | F | 0.01     |
Example

```
| C | P(S|R) |
|---|--------|
| T | .10    |
| F | .5     |

| C | P(R|C) |
|---|-------|
| T | .10   |
| F | .5    |

| S | R | P(W|S,R) |
|---|---|---------|
| T | T | .99     |
| T | F | .90     |
| F | T | .90     |
| F | F | .01     |

P(C) = .50

P(R) = .80

P(S) = .20
```
Example

- **Cloudy**
  - Probability of Cloudy: 0.5

- **Sprinkler**
  - Probability of Sprinkler:
    - T: 0.1
    - F: 0.5

- **Rain**
  - Probability of Rain:
    - T: 0.8
    - F: 0.2

- **Wet Grass**
  - Probability of Wet Grass:
    - T T: 0.99
    - T F: 0.90
    - F T: 0.90
    - F F: 0.01

- **P(C)**
  - Probability of Cloudy:
    - T: 0.99
    - F: 0.01
Example

<table>
<thead>
<tr>
<th></th>
<th>Cloudy</th>
<th>Wet Grass</th>
<th>Sprinkler</th>
<th>Rain</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>P(C)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>.50</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>.99</td>
<td>.90</td>
<td>.90</td>
<td>.90</td>
</tr>
<tr>
<td></td>
<td>.01</td>
<td>.01</td>
<td>.01</td>
<td></td>
</tr>
</tbody>
</table>

|   | T | F | C | P(R|C) |
|---|---|---|---|------|
| S |   |   |   |      |
|   | T | T | .90| .80 |
| R | T | F | .90| .20 |
|   | F | T | .90|      |
|   | F | F | .01|      |
Example

\[
P(C) = 0.50 \\
\]

\[
P(S|R) = \begin{array}{cccc}
T & T & 0.99 \\
T & F & 0.90 \\
F & T & 0.90 \\
F & F & 0.01 \\
\end{array}
\]

\[
P(R) = \begin{array}{cccc}
T & T & 0.80 \\
T & F & 0.20 \\
\end{array}
\]
Example
Sampling from an Empty Network (Contd.)

- Probability of generating \((x_1, \ldots, x_n) = P(x_1, \ldots, x_n)\)
  - Sampling following true prior probability
  - How to estimate \(P(x_1, \ldots, x_n)\) from samples?
- Let \(N(x_1 \ldots x_n) = \# \text{ samples of } (x_1, \ldots, x_n)\)
- Then
  \[
  \lim_{N \to \infty} \hat{P}(x_1, \ldots, x_n) = \lim_{N \to \infty} \frac{N(x_1, \ldots, x_n)}{N} = P(x_1, \ldots, x_n)
  \]
  
- Estimates derived from samples are consistent
  \[
  \hat{P}(x_1, \ldots, x_n) \approx P(x_1, \ldots, x_n)
  \]
Rejection Sampling

- $\hat{P}(X|e)$ estimates from samples agreeing with $e$
  - Draw sample $x$ from the Bayesian network
  - If $x$ is consistent with $e$, increment $N(x)$
  - Obtain $\hat{P}(X|e)$ by normalization
- Example
  - Estimate $\hat{P}(\text{Rain}|\text{Sprinkler} = \text{true})$ using 100 samples
  - 27 samples have Sprinkler = true
  - Of these, 8 have Rain = true and 19 have Rain = false

$$\hat{P}(\text{Rain} = \text{true} \mid \text{Sprinkler} = \text{true}) = \frac{8}{27}$$
Analysis of Rejection Sampling

- Rejection sampling estimates $N(X,e)$ and $N(e)$
- The conditional probability estimate

$$\hat{P}(X|e) = \alpha N(X,e) = \frac{N(X,e)}{N(e)} \approx \frac{P(X,e)}{P(e)} = P(X|e)$$

- Obtains consistent posterior estimates
- $P(e)$ drops off exponentially with number of evidence variables
- What if $P(e)$ is very small
  - Need large number of samples to get reliable estimates
Likelihood Weighting

- **Main Idea**
  - Fix evidence variables, sample only non-evidence variables
  - Weigh each sample by the likelihood of the evidence

- Set $w = 1$. For $i = 1$ to $n$
  - If $X_i$ is a non-evidence variable, sample $P(X_i | \text{Parents}(X_i))$
  - If $X_i$ is an evidence variable $E_i$, $w \leftarrow w \times P(E_i | \text{Parents}(E_i))$

- Then $(X, w)$ forms a weighted sample
Example

| C | P(S|C) |
|---|-------|
| T | .10   |
| F | .5    |

| C | P(R|C) |
|---|-------|
| T | .80   |
| F | .20   |

| S | R     | P(W|S,R) |
|---|-------|---------|
| T | T     | .99     |
| T | F     | .90     |
| F | T     | .90     |
| F | F     | .01     |
Example

Set \( w = 1 \). For \( i = 1 \) to \( n \)
- If \( X_i \) is a non-evidence variable, sample \( P(X_i | \text{Parents}(X_i)) \)
- If \( X_i \) is an evidence variable \( E_i \), \( w \leftarrow w \times P(E_i | \text{Parents}(E_i)) \)
Example

Sample Cloudy
Sample Cloudy

$w=1.0$
Example

Sample Cloudy

$w=1.0$
Sprinkler is an evidence node
\[ w = 1.0 \times 0.1 \]
Example

Sample Rain given Cloudy

| C | P(S|C) | P(R|C) |
|---|-------|-------|
| T | .10   |       |
| F | .5    |       |

| S | R | P(W|S,R) |
|---|---|---------|
| T | T | .99     |
| T | F | .90     |
| F | T | .90     |
| F | F | .01     |

| C | P(R|C) |
|---|-------|
| T | .80   |
| F | .20   |
Example

Sample Rain given Cloudy
Final Sample weight
\[ w = 1.0 \times 0.1 \times 0.99 = 0.099 \]
Likelihood Weighting Analysis

- Sampling probability for non-evidence component $z$
  \[
  S(z, e) = \prod_{i=1}^{l} P(z_i|\text{Parents}(Z_i))
  \]
- Sample weight from evidence component $e$
  \[
  w(z, e) = \prod_{i=1}^{m} P(e_i|\text{Parents}(E_i))
  \]
- Weighted sampling probability is
  \[
  S(z, e)w(z, e) = \prod_{i=1}^{l} P(z_i|\text{Parents}(Z_i)) \prod_{i=1}^{m} P(e_i|\text{Parents}(E_i))
  = P(z, e)
  \]
- Likelihood weighting returns consistent estimates

*Performance degrades with lots of evidence variables*
Approximate Inference using MCMC

- Construct a Markov chain based on the Bayesian network
- “State” of network = current assignment to all variables
- Generate next state by sampling one variable given Markov blanket
- Sample each variable in turn, keeping evidence fixed
  - More general sampling schedules are admissible
The Markov chain

With Sprinkler = true, WetGrass = true, there are four states:

[Diagram showing the four states: Cloudy, Rain, Sprinkler, Wet Grass.]

Wander about for a while, average what you see.
MCMC Example (Contd.)

- Problem: Estimate
  \[ P(\text{Rain}|\text{Sprinkler} = \text{true}, \text{WetGrass} = \text{true}) \]

- Sample Cloudy or Rain given its Markov blanket, repeat

- Count number of times Rain is true and false in the samples

- Example: Visit 100 states
  - 31 have Rain = true, 69 have Rain = false
  \[
  P(\text{Rain} = \text{true}|\text{Sprinkler} = \text{true}, \text{WetGrass} = \text{true}) = \frac{31}{100}
  \]

- Theorem: Markov chain approaches stationary distribution
  - Long-run fraction is proportional to posterior probability
Markov Blanket Sampling

- Markov blanket of *Cloudy* is *Sprinkler* and *Rain*
- Markov blanket of *Rain* is *Cloudy*, *Sprinkler*, and *WetGrass*
- Probability given the Markov blanket is calculated as
  \[ P(x_i|MB(X_i)) \propto P(x_i|Parents(X_i)) \prod_{Z_j \in \text{Children}(X_i)} P(z_j|Parents(Z_j)) \]

- Main computational problems
  - Difficult to tell if convergence has been achieved
  - Can be wasteful if Markov blanket is large