Recap: Inference Example

- Find $P(W|F=\text{bad})$
  - Restrict all factors
  - No hidden vars to eliminate (this time!)
  - Just join and normalize

\[
\begin{align*}
  f_W(W) &\quad f_R(W) \\
  f_{WR}(W) = f_W(W) \times f_R(W) &\quad P(W|R = \text{cloudy})
\end{align*}
\]
Decision Networks

- MEU: choose the action which maximizes the expected utility given the evidence

- Can directly operationalize this with decision diagrams
  - Bayes nets with nodes for utility and actions
  - Lets us calculate the expected utility for each action

- New node types:
  - Chance nodes (just like BNs)
  - Actions (rectangles, must be parents, act as observed evidence)
  - Utilities (depend on action and chance nodes)

Decision Networks

- Action selection:
  - Instantiate all evidence
  - Calculate posterior joint over parents of utility node
  - Set action node each possible way
  - Calculate expected utility for each action
  - Choose maximizing action
Example: Decision Networks

Umbrella = leave
EU(leave) = \sum_w P(w)U(\text{leave}, w)
= 0.7 \cdot 100 + 0.3 \cdot 0 = 70

Umbrella = take
EU(take) = \sum_w P(w)U(\text{take}, w)
= 0.7 \cdot 20 + 0.3 \cdot 70 = 35

Optimal decision = leave
MEU(\alpha) = \max_a EU(a) = 70

Example: Decision Networks

Umbrella = leave
EU(\text{leave}|\text{bad}) = \sum_w P(w|\text{bad})U(\text{bad}, w)
= 0.34 \cdot 100 + 0.66 \cdot 0 = 34

Umbrella = take
EU(\text{take}|\text{bad}) = \sum_w P(w|\text{bad})U(\text{take}, w)
= 0.34 \cdot 20 + 0.66 \cdot 70 = 53

Optimal decision = take
MEU(F = \text{bad}) = \max_a EU(a|\text{bad}) = 53
Value of Information

- Idea: compute value of acquiring each possible piece of evidence
  - Can be done directly from decision network

- Example: buying oil drilling rights
  - Two blocks A and B, exactly one has oil, worth k
  - Prior probabilities 0.5 each, mutually exclusive
  - Current price of each block is k/2
  - Probe gives accurate survey of A. Fair price?

- Solution: compute value of information
  - = expected value of best action given the information minus expected value of best action without information

- Survey may say "oil in A" or "no oil in A," prob 0.5 each
  = [0.5 * value of "buy A" given "oil in A"] +
  [0.5 * value of "buy B" given "no oil in A"] − 0
  = [0.5 * k/2] + [0.5 * k/2] − 0 = k/2

Value of Information

- Current evidence E=e, utility depends on S=s
  \[ \text{MEU}(e) = \max_a \sum_s P(s|e) \ U(s, a) \]

- Potential new evidence E': suppose we knew E' = e'
  \[ \text{MEU}(e, e') = \max_a \sum_s P(s|e, e') \ U(s, a) \]

- BUT E’ is a random variable whose value is currently unknown, so:
  - Must compute expected gain over all possible values

  \[ VPI_e(E') = \sum_{e'} P(e'|e) \left( \text{MEU}(e, e') - \text{MEU}(e) \right) \]

- (VPI = value of perfect information)
**VPI Example**

MEU with no evidence

\[ \text{MEU}(s) = \max_a \text{EU}(a) = 70 \]

MEU if forecast is bad

\[ \text{MEU}(F = \text{bad}) = \max_a \text{EU}(a|\text{bad}) = 53 \]

MEU if forecast is good

\[ \text{MEU}(F = \text{good}) = \max_a \text{EU}(a|\text{good}) = 95 \]

Forecast distribution

<table>
<thead>
<tr>
<th>F</th>
<th>P(F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>good</td>
<td>0.59</td>
</tr>
<tr>
<td>bad</td>
<td>0.41</td>
</tr>
</tbody>
</table>

\[ \text{VPI}_e(E') = \sum_{e'} P(e'|e) \left( \text{MEU}(e, e') - \text{MEU}(e) \right) \]

**VPI Properties**

- Nonnegative in expectation

\[ \forall j, e : \text{VPI}_e(E') \geq 0 \]

- Nonadditive --- consider, e.g., obtaining \( E_j \) twice

\[ \text{VPI}_e(E_j, E_k) \neq \text{VPI}_e(E_j) + \text{VPI}_e(E_k) \]

- Order-independent

\[ \text{VPI}_e(E_j, E_k) = \text{VPI}_e(E_j) + \text{VPI}_{e,E_j}(E_k) \]

\[ = \text{VPI}_e(E_k) + \text{VPI}_{e,E_k}(E_j) \]
VPI Scenarios

- Imagine actions 1 and 2, for which \( U_1 > U_2 \)
- How much will information about \( E_j \) be worth?

Reasoning over Time

- Often, we want to reason about a sequence of observations
  - Speech recognition
  - Robot localization
  - User attention
  - Medical monitoring

- Need to introduce time into our models
- Basic approach: hidden Markov models (HMMs)
- More general: dynamic Bayes’ nets
Markov Models

- A Markov model is a chain-structured BN
  - Each node is identically distributed (stationarity)
  - Value of $X$ at a given time is called the state
  - As a BN:

    $X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow X_4 \rightarrow \cdots$

    $P(X_1) \quad P(X|X_{-1})$

- Parameters: called transition probabilities or dynamics, specify how the state evolves over time (also, initial probs)

Conditional Independence

- Basic conditional independence:
  - Past and future independent of the present
  - Each time step only depends on the previous
  - This is called the (first order) Markov property

- Note that the chain is just a (growing) BN
  - We can always use generic BN reasoning on it (if we truncate the chain)
Example: Markov Chain

- **Weather:**
  - States: $X = \{\text{rain, sun}\}$
  - Transitions:
    - Initial distribution: 1.0 sun
    - What’s the probability distribution after one step?

\[
P(X_2 = \text{sun}) = P(X_2 = \text{sun}|X_1 = \text{sun})P(X_1 = \text{sun}) + P(X_2 = \text{sun}|X_1 = \text{rain})P(X_1 = \text{rain})
\]

\[
0.9 \cdot 1.0 + 0.1 \cdot 0.0 = 0.9
\]

Mini-Forward Algorithm

- **Question:** probability of being in state $x$ at time $t$?
- **Slow answer:**
  - Enumerate all sequences of length $t$ which end in $s$
  - Add up their probabilities

\[
P(X_t = \text{sun}) = \sum_{x_1 \ldots x_{t-1}} P(x_1, \ldots x_{t-1}, \text{sun})
\]

\[
P(X_1 = \text{sun})P(X_2 = \text{sun}|X_1 = \text{sun})P(X_3 = \text{sun}|X_2 = \text{sun})P(X_4 = \text{sun}|X_3 = \text{sun})
\]

\[
P(X_1 = \text{sun})P(X_2 = \text{rain}|X_1 = \text{sun})P(X_3 = \text{sun}|X_2 = \text{rain})P(X_4 = \text{sun}|X_3 = \text{sun})
\]

\[
\vdots
\]
Mini-Forward Algorithm

- Better way: cached incremental belief updates

\[
P(x_t) = \sum_{x_{t-1}} P(x_t|x_{t-1})P(x_{t-1})
\]

\[P(x_1) = \text{known}\]

Forward simulation

Example

- From initial observation of sun

\[
\begin{pmatrix}
1.0 \\
0.0
\end{pmatrix}
\begin{pmatrix}
0.9 \\
0.1
\end{pmatrix}
\begin{pmatrix}
0.82 \\
0.18
\end{pmatrix}
\begin{pmatrix}
0.5 \\
0.5
\end{pmatrix}
\]

\[P(X_1) \quad P(X_2) \quad P(X_3) \quad P(X_\infty)\]

- From initial observation of rain

\[
\begin{pmatrix}
0.0 \\
1.0
\end{pmatrix}
\begin{pmatrix}
0.1 \\
0.9
\end{pmatrix}
\begin{pmatrix}
0.18 \\
0.82
\end{pmatrix}
\begin{pmatrix}
0.5 \\
0.5
\end{pmatrix}
\]

\[P(X_1) \quad P(X_2) \quad P(X_3) \quad P(X_\infty)\]
Stationary Distributions

- If we simulate the chain long enough:
  - What happens?
  - Uncertainty accumulates
  - Eventually, we have no idea what the state is!

- Stationary distributions:
  - For most chains, the distribution we end up in is independent of the initial distribution
  - Called the stationary distribution of the chain
  - Usually, can only predict a short time out

Web Link Analysis

- PageRank over a web graph
  - Each web page is a state
  - Initial distribution: uniform over pages
  - Transitions:
    - With prob. c, uniform jump to a random page (dotted lines)
    - With prob. 1-c, follow a random outlink (solid lines)

- Stationary distribution
  - Will spend more time on highly reachable pages
  - E.g. many ways to get to the Acrobat Reader download page!
  - Somewhat robust to link spam
  - Google 1.0 returned the set of pages containing all your keywords in decreasing rank, now all search engines use link analysis along with many other factors
Most Likely Explanation

- **Question:** most likely sequence ending in x at t?
  - E.g. if sun on day 4, what’s the most likely sequence?
  - Intuitively: probably sun all four days
- **Slow answer:** enumerate and score
  
  \[ P(X_t = \text{sun}) = \max_{x_1, \ldots, x_{t-1}} P(x_1, \ldots, x_{t-1}, \text{sun}) \]
  
  \[ P(X_1 = \text{sun})P(X_2 = \text{sun}|X_1 = \text{sun})P(X_3 = \text{sun}|X_2 = \text{sun})P(X_4 = \text{sun}|X_3 = \text{sun}) \]
  
  \[ P(X_1 = \text{sun})P(X_2 = \text{rain}|X_1 = \text{sun})P(X_3 = \text{sun}|X_2 = \text{rain})P(X_4 = \text{sun}|X_3 = \text{sun}) \]
  
  \[ \vdots \]

Mini-Viterbi Algorithm

- **Better answer:** cached incremental updates

  ![Diagram showing transitions between sun and rain]

- **Define:**
  
  \[ m_t[x] = \max_{x_1:t-1} P(x_1:t-1, x) \]
  
  \[ a_t[x] = \arg\max_{x_1:t-1} P(x_1:t-1, x) \]

- **Read best sequence off of m and a vectors**
Mini-Viterbi

\[ m_t[x] = \max_{x_1:t-1} P(x_{1:t-1}, x) \]

\[ = \max_{x_1:t-1} P(x_{1:t-1}) P(x|x_{t-1}) \]

\[ = \max_{x_{t-1}} P(x_t|x_{t-1}) \max_{x_1:t-2} P(x_{1:t-1}) \]

\[ = \max_{x_{t-1}} P(x_t|x_{t-1}) m_{t-1}[x] \]

\[ m_1[x] = P(x_1) \]

Example

\[
\begin{array}{c|c|c}
 s & \pi(s) & s' & s' & t(s, s') \\
\hline
<s> & 1 & <s> & rain & 3/5 \\
 & & rain & rain & 2/5 \\
 & & rain & sun & 1/5 \\
 & & sun & rain & 4/5 \\
 & & sun & sun & 2/3 \\
\end{array}
\]

\[
\begin{array}{c}
<s> \rightarrow S_i \rightarrow \cdots \rightarrow S_i \\
\end{array}
\]
Hidden Markov Models

- Markov chains not so useful for most agents
  - Eventually you don’t know anything anymore
  - Need observations to update your beliefs

- Hidden Markov models (HMMs)
  - Underlying Markov chain over states $S$
  - You observe outputs (effects) at each time step
  - As a Bayes’ net:

```plaintext
\( X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow X_4 \rightarrow \cdots \)
```

```plaintext
\( E_1 \rightarrow E_2 \rightarrow E_3 \rightarrow E_4 \)
```
An HMM is
- Initial distribution: $P(X_1)$
- Transitions: $P(X|X_{-1})$
- Emissions: $P(E|X)$

HMMs have two important independence properties:
- Markov hidden process, future depends on past via the present
- Current observation independent of all else given current state

Quiz: does this mean that observations are independent given no evidence?
- [No, correlated by the hidden state]
Forward Algorithm

- Can ask the same questions for HMMs as Markov chains
- Given current belief state, how to update with evidence?
  - This is called monitoring or filtering
- Formally, we want: $P(X_t = x_t|e_{1:t})$

$$P(x_t|e_{1:t}) \propto P(x_t, e_{1:t})$$
$$= \sum_{x_{t-1}} P(x_{t-1}, x_t, e_{1:t})$$
$$= \sum_{x_{t-1}} P(x_{t-1}, e_{1:t-1})P(x_t|x_{t-1})P(e_t|x_t)$$
$$= P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1})P(x_{t-1}, e_{1:t-1})$$

Example

$$P(x_t|e_{1:t}) \propto f_t[x_t] = P(x_t, e_{1:t})$$

$$f_t[x_t] = P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1})f_{t-1}[x_{t-1}]$$
Viterbi Algorithm

- Question: what is the most likely state sequence given the observations?
  - Slow answer: enumerate all possibilities
  - Better answer: cached incremental version

\[ x^*_1:T = \arg \max_{x_1:T} P(x_1:T | e_{1:T}) \]
\[ m_t[x_t] = \max_{x_{1:t-1}} P(x_{1:t-1}, x_t, e_{1:t}) \]
\[ = \max_{x_{1:t-1}} P(x_{1:t-1}, e_{1:t-1}) P(x_t | x_{t-1}) P(e_t | x_t) \]
\[ = P(e_t | x_t) \max_{x_{t-1}} P(x_t | x_{t-1}) \max_{x_{1:t-2}} P(x_{1:t-1}, e_{1:t-1}) \]
\[ = P(e_t | x_t) \max_{x_{t-1}} P(x_t | x_{t-1}) m_{t-1}[x_{t-1}] \]

Example
Real HMM Examples

- **Speech recognition HMMs:**
  - Observations are acoustic signals (continuous valued)
  - States are specific positions in specific words (so, tens of thousands)

- **Machine translation HMMs:**
  - Observations are words (tens of thousands)
  - States are translation positions (dozens)

- **Robot tracking:**
  - Observations are range readings (continuous)
  - States are positions on a map (continuous)