CS 188: Artificial Intelligence
Fall 2011

Lecture 16: Bayes Nets IV
10/18/2011

Dan Klein – UC Berkeley

Announcements

- **Midterm**
  - Next **TUESDAY, 10/25, 5-8pm**
  - Prep page is on the web (practice exams, etc)
  - Topical review sessions: see prep page
  - Overall review: in class Thursday
  - If you have a conflict, we should already know about it!

- **Written 3**
  - Due this Friday but fixes not due until NEXT Friday

- P1, P2, W1 in glookup

Approximate Inference

- **Simulation has a name: sampling**
- Sampling is a hot topic in machine learning, and it’s really simple
- **Basic idea:**
  - Draw N samples from a sampling distribution \( S \)
  - Compute an approximate posterior probability
  - Show this converges to the true probability \( P \)

- **Why sample?**
  - Learning: get samples from a distribution you don’t know
  - Inference: getting a sample is faster than computing the right answer (e.g. with variable elimination)

Prior Sampling

- This process generates samples with probability:
  \[
P_S(x_1 \ldots x_n) = \prod_{i=1}^{n} P(x_i | \text{Parents}(X_i)) = P(x_1 \ldots x_n)
\]
  \[\ldots \text{i.e. the BN’s joint probability}\]

- Let the number of samples of an event be \( N_P(x_1 \ldots x_n) \)

- Then
  \[
  \lim_{N \to \infty} P_S(x_1 \ldots x_n) = \lim_{N \to \infty} \frac{N_P(x_1 \ldots x_n)}{N} = S_{BP}(x_1 \ldots x_n) = \frac{P(x_1 \ldots x_n)}{N}
  \]

- I.e., the sampling procedure is consistent
Example

- First: Get a bunch of samples from the BN:
  - \(-C, -S, +T, +W\)
  - \(-C, +S, +T, +W\)
  - \(+C, +S, +T, +W\)
  - \(-C, -S, +T, +W\)

- Example: we want to know \(P(W)\)
  - We have counts \(-w: 4, \ w: 1\)
  - Normalize to get approximate \(P(W) = -w: 0.8, \ w: 0.2\)
  - This will get closer to the true distribution with more samples
  - Can estimate anything else, too
  - What about \(P(C|+w)\)? \(P(C|+w)\) vs \(P(C|+w)\)?
  - Faster: can use fewer samples if less time (what’s the drawback?)

Sampling Example

- There are 2 cups.
  - The first contains 1 penny and 1 quarter
  - The second contains 2 quarters

- Say I pick a cup uniformly at random, then pick a coin randomly from that cup. It's a quarter (yes!). What is the probability that the other coin in that cup is also a quarter?

Rejection Sampling

- Let’s say we want \(P(C)\)
  - No point keeping all samples around
  - Just tally counts of \(C\) as we go

- Let’s say we want \(P(C|+s)\)
  - Same thing: tally \(C\) outcomes, but ignore (reject) samples which don’t have \(S=+s\)
  - This is called rejection sampling
  - It is also consistent for conditional probabilities (i.e., correct in the limit)

Likelihood Weighting

- Problem with rejection sampling:
  - If evidence is unlikely, you reject a lot of samples
  - You don’t exploit your evidence as you sample
  - Consider \(P(B|+a)\)

- Idea: fix evidence variables and sample the rest

- Problem: sample distribution not consistent!

- Solution: weight by probability of evidence given parents

Likelihood Weighting

- Sampling distribution if \(z\) sampled and \(e\) fixed evidence
  \[S_{Wz}(z, e) = \prod_{i=1}^{j} P(z_i|\text{Parents}(z_i))\]

- Now, samples have weights
  \[w(z, e) = \prod_{i=1}^{m} P(z_i|\text{Parents}(E_i))\]

- Together, weighted sampling distribution is consistent
  \[S_{Wz}(z, e) \cdot w(z, e) = \prod_{i=1}^{j} P(z_i|\text{Parents}(z_i)) \prod_{i=1}^{m} P(z_i|\text{Parents}(E_i))\]

\[= P(z, e)\]
Likelihood Weighting

- Likelihood weighting is good
  - We have taken evidence into account as we generate the sample
  - E.g. here, W's value will get picked based on the evidence values of S, R
  - More of our samples will reflect the state of the world suggested by the evidence
- Likelihood weighting doesn't solve all our problems
  - Evidence influences the choice of downstream variables, but not upstream ones (C isn't more likely to get a value matching the evidence)
- We would like to consider evidence when we sample every variable

Markov Chain Monte Carlo*

- Idea: instead of sampling from scratch, create samples that are each like the last one.
  - Procedure: resample one variable at a time, conditioned on all the rest, but keep evidence fixed. E.g., for P(B|c):
  - Properties: Now samples are not independent (in fact they're nearly identical), but sample averages are still consistent estimators!
  - What's the point: both upstream and downstream variables condition on evidence.

Decision Networks

- MEU: choose the action which maximizes the expected utility given the evidence
- Can directly operationalize this with decision networks
  - Bayes nets with nodes for utility and actions
  - Let's calculate the expected utility for each action
- New node types:
  - Chance nodes (just like BNs)
  - Actions (rectangles, cannot have parents, act as observed evidence)
  - Utility node (diamond, depends on action and chance nodes)

Example: Decision Networks

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

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

Optimal decision = leave

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

Decisions as Outcome Trees

- Almost exactly like expectimax / MDPs
- What's changed?
Evidence in Decision Networks

- **Find** $P(W|F=\text{bad})$
- **Select** for evidence
- **First**, we join $P(W)$ and $P(\text{bad}|W)$
- **Then**, we normalize

<table>
<thead>
<tr>
<th>Weather</th>
<th>P(W)</th>
</tr>
</thead>
<tbody>
<tr>
<td>sun</td>
<td>0.7</td>
</tr>
<tr>
<td>rain</td>
<td>0.3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Forecast</th>
<th>P(F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>good</td>
<td>0.8</td>
</tr>
<tr>
<td>bad</td>
<td>0.2</td>
</tr>
</tbody>
</table>

$P(\text{bad}) = P(F=\text{bad})$

**Example: Decision Networks**

**Umbrella = leave**

$\text{EU(leave|bad)} = \sum_e P(e|\text{bad})U(\text{leave}, e)$

$= 0.34 \cdot 100 + 0.66 \cdot 0 = 34$

**Umbrella = take**

$\text{EU(take|bad)} = \sum_e P(e|\text{bad})U(\text{take}, e)$

$= 0.34 \cdot 20 + 0.66 \cdot 70 = 53$

**Optimal decision** = take

$\text{MEU}(F=\text{bad}) = \max_e \text{EU}(\text{e|bad}) = 53$

Decisions as Outcome Trees

<table>
<thead>
<tr>
<th>Weather</th>
<th>P(W)</th>
</tr>
</thead>
<tbody>
<tr>
<td>sun</td>
<td>0.7</td>
</tr>
<tr>
<td>rain</td>
<td>0.3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Forecast</th>
<th>P(F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>good</td>
<td>0.8</td>
</tr>
<tr>
<td>bad</td>
<td>0.2</td>
</tr>
</tbody>
</table>

**Value of Information**

- **Assume** we have evidence $E=e$. **Value** if we act now:
  \[ \text{MEU}(e) = \max_a \sum_{s} P(s|e) U(s,a) \]
- **Assume** we see that $E'=e'$. **Value** if we act then:
  \[ \text{MEU}(e,e') = \max_a \sum_{s} P(s|e,e') U(s,a) \]
- **BUT** $E'$ is a random variable whose value is unknown, so we don't know what $e'$ will be:
- **Expected value** if $E'$ is revealed and then we act:
  \[ \text{MEU}(e,E') = \sum_{e'} P(e'|e) \text{MEU}(e,e') \]
- **Value of information**: how much MEU goes up by revealing $E'$ first then acting, over acting now:
  \[ \text{VPI}(E'|e) = \text{MEU}(e,E') - \text{MEU}(e) \]

**VPI Example: Weather**

**MEU with no evidence**

$\text{MEU}(\text{e}) = \max_a \text{EU}(\text{e|a}) = 70$

$\text{MEU if forecast is bad}$

$\text{MEU}(F=\text{bad}) = \max_a \text{EU}(\text{e|bad}) = 53$

$\text{MEU if forecast is good}$

$\text{MEU}(F=\text{good}) = \max_a \text{EU}(\text{e|good}) = 95$

<table>
<thead>
<tr>
<th>Forecast</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>

**Forecast distribution**

$\text{VPI}(E'|\text{e}) = \sum_{e'} P(e'|\text{e}) \text{MEU}(e,e') - \text{MEU}(\text{e})$
VPI Properties

- Nonnegative
  \[ \forall E', e : \text{VPI}(E'|e) \geq 0 \]
- Nonadditive – consider, e.g., obtaining \( E_j \) twice
  \[ \text{VPI}(E_j, E_k|e) \neq \text{VPI}(E_j|e) + \text{VPI}(E_k|e) \]
- Order-independent
  \[ \text{VPI}(E_j, E_k|e) = \text{VPI}(E_j|e) + \text{VPI}(E_k|e, E_j) \]
  \[ = \text{VPI}(E_k|e) + \text{VPI}(E_j|e, E_k) \]

Quick VPI Questions

- The soup of the day is either clam chowder or split pea, but you wouldn’t order either one. What’s the value of knowing which it is?
- There are two kinds of plastic forks at a picnic. One kind is slightly sturdier. What’s the value of knowing which?
- You’re playing the lottery. The prize will be $0 or $100. You can play any number between 1 and 100 (chance of winning is 1%). What is the value of knowing the winning number?