CSE250A Fall '12: Rejection Sampling and Likelihood Weighting

Aditya Menon (akmenon@ucsd.edu)

October 24, 2012

Setting. We are given a Bayesian network comprising N nodes $\{X_1, \ldots, X_N\}$. Recall that in a Bayesian network, for every node X_i, we have a CPT that specifies $\Pr[X_i|\text{Pa}(X_i)]$, where $\text{Pa}(X)$ denotes the parents of X.

The problem. We would like to know the conditional probability $\Pr[X_q|X_e]$ for some indices $q \neq e$. (This generalizes to the case of *sets* of nodes, but we'll stick to individual nodes here for simplicity.) We'll think of X_q as being the *query* node and X_e being the *evidence* node.

Exact solutions. If X_e is the sole parent of X_q, we just need to perform a lookup on the CPT of node X_q, and we're done.

But what if this is not the case? The straightforward strategy is to marginalize out all other random variables in the network, and appeal to conditional independences:

$$\Pr[X_q|X_e] = \sum_{\{X_i i \neq \{q,e\}\}} \Pr[X_1, \ldots, X_N|X_e] = \ldots \text{(some simplification depending on the structure of the network)}.$$

This is perfectly reasonable. (Indeed, it's how we've computed conditional probabilities thus far.) However, it is computationally expensive in general. In the worst case, if each node is binary, we would need to do $O(2^N)$ work to compute this conditional probability.

We'd like to know: can we get an *approximate* value for this probability with less effort?

Approximate solution I. Here is a simple alternative to exact marginalization.

1. Draw M samples from the joint distribution of the Bayesian network, $\Pr[X_1, \ldots, X_N]$. Call these $\{(x_1^{(i)}, \ldots, x_N^{(i)})\}_{i=1}^M$. Note that the process of drawing these samples is simple: we use the canonical decomposition
$$\Pr[X_1, \ldots, X_N] = \prod_{i=1}^N \Pr[X_i|\text{Pa}(X_i)],$$
and then perform CPT lookups to compute each of the individual probabilities.

2. Compute the quantity
$$\hat{\theta}(x_q, x_e) = \frac{\sum_{i=1}^M \mathbf{1}[x_q^{(i)} = x_q] \cdot \mathbf{1}[x_e^{(i)} = x_e]}{\sum_{i=1}^M \mathbf{1}[x_e^{(i)} = x_e]},$$
where (x_q, x_e) represent any pair of possible outcomes for the variables (X_q, X_e). For example, if all nodes are binary, then $x_q, x_e \in \{0, 1\}$.

The claim is that $\Pr[X_q = x_q|X_e = x_e] \approx \hat{\theta}(x_q, x_e)$ when M is large. More precisely, it can be shown that

$$\lim_{M \to \infty} \hat{\theta}(x_q, x_e) = \Pr[X_q = x_q|X_e = x_e].$$

Why is this so? First, let's write $\hat{\theta}$ as

$$\hat{\theta}(x_q, x_e) = \frac{N(x_q, x_e)}{N(x_e)},$$

where $N(x_q, x_e)$ is the number of samples where the X_q variable takes the value x_q *and* the X_e variable takes the value x_e, and $N(x_e)$ is the number of samples where X_e takes the value x_e. This is trivially equivalent to

$$\hat{\theta}(x_q, x_e) = \frac{N(x_q, x_e)}{M} \cdot \frac{M}{N(x_e)}.$$

Now remember that $\Pr[X_q = x_q | X_E = x_e] = \frac{\Pr[X_Q = x_q, X_E = x_e]}{\Pr[X_E = x_e]}$. It should be clear intuitively that $\Pr[X_Q = x_q, X_E = x_e] \approx \frac{N(x_q, x_e)}{M}$ and $\Pr[X_e = x_e] \approx \frac{N(x_e)}{M}$ when M is large. Thus, $\Pr[X_q = x_q | X_e = x_e] \approx \hat{\theta}(x_q, x_e)$.

Note that above, if a sample has $x_e^{(i)} \neq x_e$, we effectively just throw it away – this is because it doesn't contribute to either the numerator or denominator of $\hat{\theta}$. Thus, this method is known as *rejection sampling*. It should be clear that if $\Pr[E = e] \approx 0$, this method will need many samples to get a good approximation.

We quickly explain how rejection sampling would work for the network shown in Figure 1.

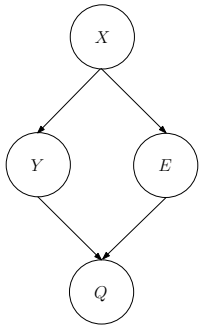

Figure 1: Example Bayesian network.

Say we want to estimate $\Pr[Q = q | E = e]$. We do the following.

- Draw M samples from $\Pr[X, Y, Q, E]$, using the decomposition

  $$\Pr[X, Y, Q, E] = \Pr[X] \Pr[Y|X] \Pr[E|X] \Pr[Q|E, Y],$$

  i.e. for a given sample, first draw $x \sim \Pr[X]$, then $y \sim \Pr[Y | X = x]$, and so on. Call the samples $\{(x^{(i)}, y^{(i)}, q^{(i)}, e^{(i)})\}_{i=1}^{M}$.
- Throw out all samples where $e^{(i)} \neq e$.
- Compute $\hat{\theta}(q, e) = \frac{N(q,e)}{N(e)}$.

Approximate solution II. Here is an alternative strategy that can converge faster than rejection sampling. First, to simplify things, we let $X_{-e} := X_1, \ldots, X_{e-1}, X_{e+1}, \ldots, X_N$, and similarly $x_{-e}^{(i)} = (x_1^{(i)}, \ldots, x_{e-1}^{(i)}, x_{e+1}^{(i)}, \ldots, x_N^{(i)})$.

1. Draw M samples from the network as follows. For every node $X_i \neq X_e$, draw X_i based on its CPT, i.e. based on $\Pr[X_i | \text{Pa}(X_i)]$. By this, we mean that if X_1 is the "root" of the network, then we first draw $x_1 \sim \Pr[X_1]$, then draw $x_2 \sim \Pr[X_2 | X_1 = x_1]$, and so on. Importantly, for all children of X_e, the value of X_e when we look up the CPT is *clamped* to the value x_e. Call the resulting samples $\{x_{-e}^{(i)}\}_{i=1}^{M}$. (We'll go over an example of this shortly.)

2. Compute the quantity

$$\hat{\theta}(x_q, x_e) = \frac{\sum_{i=1}^{M} \mathbf{1}[x_q^{(i)} = x_q] \cdot \Pr[X_e = x_e | (\text{Pa}(X_e))^{(i)}]}{\sum_{i=1}^{M} \Pr[X_e = x_e | (\text{Pa}(X_e))^{(i)}]}, \tag{1}$$

where as before (x_q, x_e) represent any pair of possible outcomes for the variables (X_q, X_e).

We claim that, as before, $\lim_{M\to\infty} \hat{\theta}_{qe} = \Pr[Q = q | E = e]$. The proof is non-trivial, so we defer it for the moment. But just intuitively, we can think of this method as forcing $X_e = x_e$ in the network, getting samples for X_q and counting the fraction of times that $X_q = x_q$ happens. However, when we count, we must take into account the fact that the X_q samples are *not* from the correct distribution, namely, $\Pr[X_q | X_e = x_e]$. It turns out however that applying $\Pr[X_e = x_e | \text{Pa}(X_e)]$ as a weighting factor removes the bias. Thus, the method is called *likelihood weighting*.

We sketch how we would apply likelihood weighting for the network of Figure 1.

1. Draw M samples from the network as follows:
   - First draw $x \sim \Pr[X]$
   - Next, draw $y \sim \Pr[Y | X = x]$
   - Do *not* draw a sample from $\Pr[E|X]$. We assume that $E = e$ for all our samples.
   - Finally, draw $q \sim \Pr[Q | E = e, Y = y]$. Note that here, we clamped the value of E.

   Call the samples $\{(x^{(i)}, y^{(i)}, q^{(i)})\}_{i=1}^{M}$.
2. Compute $\Pr[E = e | X = x^{(i)}]$ for every sample, since X is the sole parent of E.
3. Compute

$$\hat{\theta}(q, e) = \frac{\sum_{i=1}^{M} \mathbf{1}[q^{(i)} = q] \cdot \Pr[E = e | X = x^{(i)}]}{\sum_{i=1}^{M} \Pr[E = e | X = x^{(i)}]}.$$

We note that compared to rejection sampling, we don't bother sampling E, and instead clamp it down to always have the value e. Instead, we need to compute $\Pr[E = e | x^{(i)}]$, and then use that to weight each of our counts.

Here's a sketch of what exactly the method is doing for this network. Consider that

$$\begin{aligned}
\Pr[Q = q, E = e] &= \sum_{X,Y} \Pr[Q = q, E = e, X, Y] \text{ by marginalization} \\
&= \sum_{X,Y} \Pr[Q = q | X, Y, E = e] \Pr[E = e | X, Y] \Pr[X, Y] \text{ by the product rule} \\
&= \sum_{X,Y} \Pr[Q = q | Y, E = e] \Pr[E = e | X] \Pr[X, Y] \text{ by d-separation} \\
&= \mathbb{E}_{x,y\sim\Pr[X,Y]} \left[\Pr[Q = q | Y = y, E = e] \Pr[E = e | X = x]\right] \\
&= \mathbb{E}_{x,y\sim\Pr[X,Y]} \left[\mathbb{E}_{q'\sim\Pr[Q|Y=y,E,=e]} \left[\mathbf{1}[q' = q] \Pr[E = e | X = x]\right]\right].
\end{aligned}$$

In the last two lines, we used the fact that $\Pr[A = a | B = b] = \mathbb{E}_{a'\sim\Pr[A|B=b]} \mathbf{1}[a' = a]$.

Now observe that we may approximate the expectation by drawing random samples from the appropriate distributions, and averaging. Given M samples $\{(x^{(i)}, y^{(i)})\}$ from $\Pr[X, Y]$ and $\{q^{(i)}\}$ from $\Pr[Q | Y = y^{(i)}, E = e]$, we have that

$$\Pr[Q = q, E = e] \approx \frac{1}{M} \sum_{i=1}^{M} \mathbf{1}[q^{(i)} = q] \Pr[E = e | X = x^{(i)}].$$

Similarly, using the fact that

$$\Pr[E = e] = \sum_X \Pr[X] \Pr[E = e|X = x] = \mathbb{E}_{x \sim \Pr[X]} \Pr[E = e|X = x],$$

we may write

$$\Pr[E = e] \approx \frac{1}{M} \sum_{i=1}^{M} \Pr[E = e|X = x^{(i)}].$$

Now using the definition of conditional probability, we get

$$\Pr[Q = q|E = e] \approx \frac{\sum_{i=1}^{M} \mathbf{1}[q^{(i)} = q] \cdot \Pr[E = e|X = x^{(i)}]}{\sum_{i=1}^{M} \Pr[E = e|X = x^{(i)}]}.$$

The idea can be generalized. Suppose WLOG that in the topological sort of the nodes in the Bayesian network, X_e appears before X_q. Now consider that

$$\begin{aligned}
\Pr[X_q = x_q, X_e = x_e] &= \sum_{\{x_i : 1 \le i \le q-1, i \ne e\}} \Pr[X_1 = x_1, \ldots, X_q = x_q] \\
&= \sum_{\{x_i : 1 \le i \le q-1, i \ne e\}} \Pr[X_1 = x_1, \ldots, X_{e-1} = x_{e-1}] \cdot \Pr[X_e = x_e | \text{Pa}(X_e) = \pi_e] \cdot \\
&\qquad \Pr[X_{e+1} = x_{e+1}, \ldots, X_{q-1} = x_{q-1} | X_1 = x_1, \ldots, X_e = x_e] \cdot \Pr[X_q = x_q | \text{Pa}(X_q) = \pi_q] \\
&= \mathbb{E}_{x_1, \ldots, x_{q-1}, x'_q} \Pr[X_e = x_e | \text{Pa}(X_e) = \pi_e] \cdot \mathbf{1}[x_q = x'_q]
\end{aligned}$$

where the expectation is over the appropriate distribution for each variable, i.e.

$$\begin{aligned}
(x_1, \ldots, x_{e-1}) &\sim \Pr[X_1, \ldots, X_{e-1}] \\
(x_{e+1}, \ldots, x_{q-1}) &\sim \Pr[X_{e+1}, \ldots, X_{q-1} | X_1, \ldots, X_e = x_e] \\
x'_q &\sim \Pr[X_q | \text{Pa}(X_q)].
\end{aligned}$$

As before, the expectation may be approximated by taking the average of the inner quantity based on a number of random samples.