CSE 150A-250A AI: Probabilistic Models

Lecture 6

Fall 2025

Trevor Bonjour
Department of Computer Science and Engineering
University of California, San Diego

Slides adapted from previous versions of the course (Prof. Lawrence, Prof. Alvarado, Prof Berg-Kirkpatrick)

Agenda

Review

Approximate inference in loopy BNs

- Rejection sampling
- Likelihood weighting

Review

Inference

- Problem

  Given a set E of evidence nodes, and a set Q of query nodes, how to compute the posterior distribution $P(Q|E)$?

- More precisely

  How to express $P(Q|E)$ in terms of the CPTs $P(X_i|pa(X_i))$ of the BN, which are assumed to be given?

- Tools at our disposal

Bayes rule	marginal independence
marginalization	conditional independence
product rule	

Exact Inference: Variable Elimination

- **Idea**: Eliminate redundant calculations by storing intermediate results in “factors”.
- A **factor** is a function that takes in values of random variables, and produces a number.
- VE works by eliminating all variables in turn until there is a factor with only the query variable.
- To eliminate a variable:
  - *join* all factors containing that variable.
  - *sum* out the influence of the variable on the new factor.

VE Example

$$
\begin{aligned}
P(J) &= \sum_{M,A,B,E} P(J, M, A, B, E) \\
&= \sum_{M,A,B,E} P(J|A)P(M|A)P(B)P(A|B,E)P(E) \\
&= \sum_{A} P(J|A) \sum_{M} P(M|A) \sum_{B} P(B) \sum_{E} P(A|B,E)P(E) \\
&= \sum_{A} P(J|A) \sum_{M} P(M|A) \sum_{B} (P(B) f1(A,B) \\
&= \sum_{A} P(J|A) \sum_{M} P(M|A) f2(A) \\
&= \sum_{A} P(J|A) f3(A) \\
&= f4(J)
\end{aligned}
$$

VE Example

$$
\begin{aligned}
P(J) &= \sum_{M,A,B,E} P(J, M, A, B, E) \\
&= \sum_{M,A,B,E} P(J|A)P(M|A)P(B)P(A|B,E)P(E) \\
&= \sum_{A} P(J|A) \sum_{M} P(M|A) \sum_{B} P(B) \sum_{E} P(A|B,E)P(E) \\
&= \sum_{A} P(J|A) \sum_{M} P(M|A) \sum_{B} (P(B) f1(A,B) \\
&= \sum_{A} P(J|A) \sum_{M} P(M|A) f2(A) \\
&= \sum_{A} P(J|A) f3(A) \\
&= f4(J)
\end{aligned}
$$

Q. What is the elimination order?

A. M,A,B,E

B. E,B,A,M

C. A,M,B,E

VE Example

$$
\begin{aligned}
P(J) &= \sum_{M,A,B,E} P(J, M, A, B, E) \\
&= \sum_{M,A,B,E} P(J|A)P(M|A)P(B)P(A|B, E)P(E) \\
&= \sum_{A} P(J|A) \sum_{M} P(M|A) \sum_{B} P(B) \sum_{E} P(A|B, E)P(E) \\
&= \sum_{A} P(J|A) \sum_{M} P(M|A) \sum_{B} (P(B) f1(A, B) \\
&= \sum_{A} P(J|A) \sum_{M} P(M|A) f2(A) \\
&= \sum_{A} P(J|A) f3(A) \\
&= f4(J)
\end{aligned}
$$

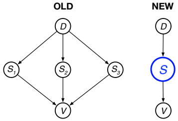
Q. What if we first eliminate M ?

Exact Inference: Variable Elimination

- Time and space of VE is dominated by the **largest** factor created.
  - Can be exponential in the size of BN in the worst case.
- **Heuristic:** Eliminate the variable that will lead to the smallest next factor being created
  - In a **polytree** this leads to **linear** time inference (in size of largest CPT).

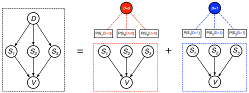
Exact Inference in Loopy BNs

- Transform a loopy BN into a polytree.
- Run the exact inference algorithm.
  - Node Clustering



CPTs grow exponentially when nodes are clustered.

  - Cutset Conditioning



Number of runs grows exponentially with the size of the cutset.

- No efficient algorithm to get clustering or the minimal cutset leads to maximally efficient inference.

Approximate inference in loopy BNs

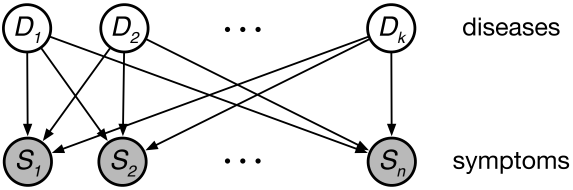
Motivation

- **Formal results**

  Exact inference in belief networks is **NP-hard**.
  Actually it is **#P-hard** (even worse).

- **Practical tools**

  But many large loopy BNs remain useful models.
  In these BNs, we must resort to approximate methods.



Approximate inference

There are many strategies for approximate inference.
We will focus on **stochastic simulation methods**.

This lecture:

1. Prior sampling
2. Rejection sampling
3. Likelihood weighting
4. Markov chain Monte Carlo (MCMC) - Gibbs Sampling

But before we can describe the above methods,
we must review some basic ideas in sampling ...

Sampling Intuition

Suppose we have a (biased) coin but we don't know the $P(H)$.

- How can we estimate $P(H)$?
- When will we get the correct probability?

Sampling

Basic Idea:

- Draw N samples from a sampling distribution S.
- Compute an approximate posterior probability.
- Show that this converges to the true probability P.

We can apply similar intuition for inference in BNs -> Generate samples from a BN.

Sampling a discrete random variable $X \sim P(X)$

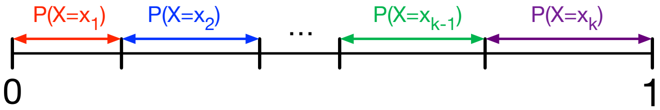
Let X be a discrete random variable with probabilities $P(X=x_i)$. Suppose we can generate random numbers uniformly in $[0, 1)$ (e.g. random() in python).

- **Problem**

  How to sample values of X so that repeated samples are distributed according to $P(X)$?

- **Solution**

  Note that $P(X=x_i)$ defines a partition of unity, which maps a random number $r \in [0, 1)$ into a discrete value of X.



Sampling a discrete random variable $X \sim P(X)$

C	P(C)
orange	0.6
blue	0.1
green	0.3

1. Get sample u from uniform distribution over [0,1).
2. Assign discrete value of C.
   - $0 \le u < 0.6, C \leftarrow orange$
   - $0.6 \le u < 0.7, C \leftarrow blue$
   - $0.7 \le u < 1, C \leftarrow green$

8 samples generated:

orange, green, orange, blue, orange, orange, green, orange

$$P(C = green)?$$

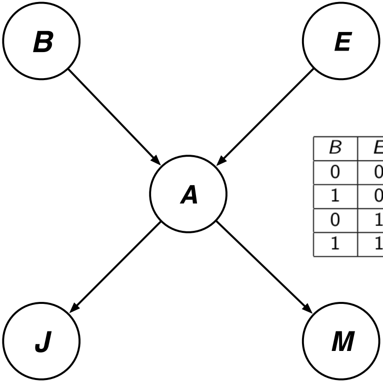
Generative Model

Belief Network is a generative model. We can sample from the distribution represented by the BN.

Idea: Generate one variable at a time in topological order.

$P(B=1) = 0.001$

$P(E=1) = 0.002$

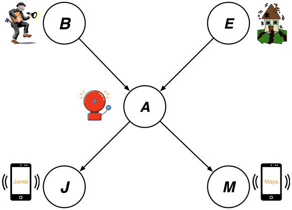


B	E	$P(A=1 \mid B, E)$
0	0	0.001
1	0	0.94
0	1	0.29
1	1	0.95

A	$P(J=1 \mid A)$
0	0.05
1	0.9

A	$P(M=1 \mid A)$
0	0.01
1	0.7

Alarm example



- **Joint sample**

  To draw a joint sample $\{b, e, a, j, m\}$ from $P(B, E, A, J, M)$, it is enough to draw the individual samples:

$$\begin{aligned} b &\sim P(B) \\ e &\sim P(E) \\ a &\sim P(A|B=b, E=e) \\ j &\sim P(J|A=a) \\ m &\sim P(M|A=a) \end{aligned}$$

- **Convergence in the limit**

$$P(b, e, a, j, m) = \lim_{N \to \infty} \frac{\text{count}(B=b, E=e, A=a, J=j, M=m)}{N}.$$

Prior Sampling from the joint distribution in a discrete BN

Let $\{X_1, X_2, \ldots, X_m\}$ be discrete random variables in a BN.

- **Problem**

  How to sample values of $\{X_1, X_2, \ldots, X_m\}$ so that repeated samples are distributed according to $P(X_1, X_2, \ldots, X_m)$?

- **Solution**

  The BN defines a **generative model** with joint distribution

$$P(X_1, X_2, \ldots, X_m) = \prod_{i=1}^{m} P(X_i | \mathrm{pa}(X_i)).$$

  To draw samples, we can simply take $X_i \sim P(X_i | \mathrm{pa}(X_i))$.

Approximate inference

- **Problem**

  Let Q denote a set of query nodes.
  Let E denote a set of evidence nodes.
  How can we ~~calculate~~ **estimate** $P(Q|E)$?

- **Challenge**

  While easy to sample from the BN's joint distribution,
  it may be much harder to sample directly from $P(Q|E)$.

- **Solutions**

  1. rejection sampling (very inefficient)
  2. likelihood weighting (more efficient)
  3. MCMC (most efficient)

Rejection sampling in BNs

- **Problem**

  Let Q denote a set of query nodes.
  Let E denote a set of evidence nodes.
  How to estimate $P(Q=q|E=e)$?

- **Solution**

  Generate N samples from the BN's joint distribution.
  Discard (or **reject**) the samples where $E \neq e$.
  Count the samples $N(q,e)$ where $Q=q$ and $E=e$.
  Count the samples $N(e)$ where $E=e$.
  Take the ratio of these counts:

$$\boxed{P(Q=q|E=e) \approx \frac{N(q,e)}{N(e)}} \quad \textbf{where} \quad \boxed{N(q,e) \leq N(e) \leq N}$$

Example for rejection sampling

Problem: Estimate $P(a_0|c_1, d_1)$

Samples:

a_1, b_1, c_0, d_0

a_0, b_1, c_1, d_0

a_1, b_1, c_0, d_1

a_1, b_1, c_1, d_1

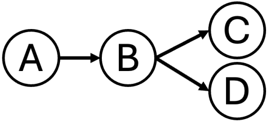
a_1, b_0, c_0, d_0

a_0, b_1, c_1, d_1

Q. How many samples will be rejected?

A. 6 **B.** 2 **C.** 4

D. 3 **E.** 0



A	P(A)
a_0	1/5
a_1	4/5

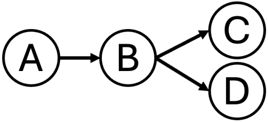
A	B	P(B|A)
a_0	b_0	1/4
a_0	b_1	3/4
a_1	b_0	1/3
a_1	b_1	2/3

B	C	P(C|B)
b_0	c_0	1/5
b_0	c_1	4/5
b_1	c_0	3/5
b_1	c_1	2/5

B	D	P(D|B)
b_0	d_0	3/4
b_0	d_1	1/4
b_1	d_0	1/3
b_1	d_1	2/3

Example for rejection sampling

Problem: Estimate $P(a_0|c_1, d_1)$



Samples:

a_1, b_1, c_0, d_0

a_0, b_1, c_1, d_0

a_1, b_1, c_0, d_1

a_1, b_1, c_1, d_1

a_1, b_0, c_0, d_0

a_0, b_1, c_1, d_1

A	P(A)
a_0	1/5
a_1	4/5

A	B	P(B|A)
a_0	b_0	1/4
a_0	b_1	3/4
a_1	b_0	1/3
a_1	b_1	2/3

B	C	P(C|B)
b_0	c_0	1/5
b_0	c_1	4/5
b_1	c_0	3/5
b_1	c_1	2/5

B	D	P(D|B)
b_0	d_0	3/4
b_0	d_1	1/4
b_1	d_0	1/3
b_1	d_1	2/3

Q.Estimate of $P(a_0|c_1, d_1)$ using rejection sampling?

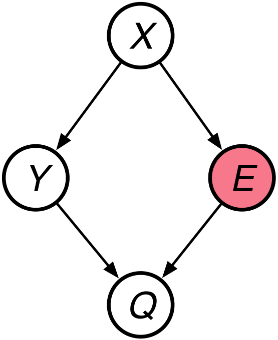
A. 1/2 **B.** 2/3 **C.** 1/4

D. 1/3 **E.** 0

Example for rejection sampling

- Sample N times:

$$\begin{aligned} x_i &\sim P(X) \\ y_i &\sim P(Y|X{=}x_i) \\ e_i &\sim P(E|X{=}x_i) \\ q_i &\sim P(Q|Y{=}y_i, E{=}e_i) \end{aligned}$$



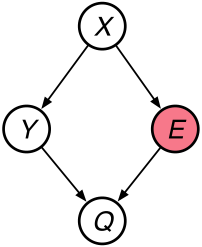
- Define the indicator function:

$$I(z, z') = \begin{cases} 1 & \text{if } z{=}z' \\ 0 & \text{if } z{\neq}z' \end{cases}$$

- Estimate from ratio:

$$P(Q{=}q|E{=}e) \approx \frac{N(q,e)}{N(e)} = \frac{\sum_{i=1}^{N} I(q, q_i) I(e, e_i)}{\sum_{i=1}^{N} I(e, e_i)}$$

Properties of rejection sampling



- It converges in the limit:

$$\lim_{N \to \infty} \frac{N(q,e)}{N(e)} = P(Q{=}q|E{=}e)$$

- But it is extremely inefficient:

  It discards all samples without $E = e$.
  It converges **very slowly** for rare evidence.

How can we do better?

2. Likelihood weighting

- Key idea

  Instantiate evidence nodes instead of sampling them.
  Weight each sample using CPTs at evidence nodes.

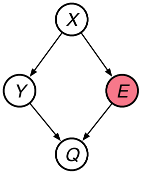
- Intuition

  “Cheat” by fixing the evidence nodes to their desired values.
  “Correct” for cheating by penalizing especially unlikely values.

- Benefits

  No discarding of uninformative samples.
  No wasted computation.
  Faster convergence.

How to estimate $P(Q = q | E = e)$?



- Sample N times:

$$
\begin{aligned}
x_i &\sim P(X) \\
y_i &\sim P(Y | X = x_i) \\
q_i &\sim P(Q | Y = y_i, E = e)
\end{aligned}
$$

Note: E is fixed to e.

- Estimate from ratio:

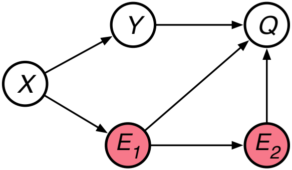
$$P(Q = q | E = e) \approx \frac{\sum_{i=1}^{N} I(q, q_i) \overbrace{P(E = e | X = x_i)}^{\text{likelihood weight}}}{\sum_{i=1}^{N} P(E = e | X = x_i)}$$

- Compare to rejection sampling:

$$P(Q = q | E = e) \approx \frac{\sum_{i=1}^{N} I(q, q_i) I(e, e_i)}{\sum_{i=1}^{N} I(e, e_i)}$$

Example with multiple evidence nodes

How to estimate $P(Q{=}q|E_1{=}e, E_2{=}e')$?



- Sample N times:

$$\begin{aligned} x_i &\sim P(X) \\ y_i &\sim P(Y|X{=}x_i) \\ q_i &\sim P(Q|Y{=}y_i, E_1{=}e, E_2{=}e') \end{aligned}$$

Note: (E_1, E_2) fixed to (e, e')

- Estimate from ratio:

product of likelihood weights

$$P(Q{=}q|E_1{=}e, E_2{=}e') \approx \frac{\sum_{i=1}^{N} I(q, q_i)\, \overbrace{P(E_1{=}e|X{=}x_i)\, P(E_2{=}e'|E_1{=}e)}^{\text{product of likelihood weights}}}{\sum_{i=1}^{N} P(E_1{=}e|X{=}x_i)\, P(E_2{=}e'|E_1{=}e)}$$

- Compare to rejection sampling:

$$P(Q{=}q|E{=}e) \approx \frac{\sum_{i=1}^{N} I(q, q_i)\, I(e, e_{1i})\, I(e', e_{2i})}{\sum_{i=1}^{N} I(e, e_{1i})\, I(e', e_{2i})}$$

Example for likelihood weighting sampling

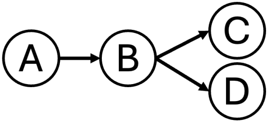
Problem: Estimate $P(a_0|c_1, d_1)$

Samples:

a_0, b_1, c_1, d_1

a_1, b_0, c_1, d_1

a_0, b_1, c_1, d_1



A	P(A)
a_0	1/5
a_1	4/5

A	B	P(B|A)
a_0	b_0	1/4
a_0	b_1	3/4
a_1	b_0	1/3
a_1	b_1	2/3

Q. Estimate of $P(a_0|c_1, d_1)$ using likelihood weighting?

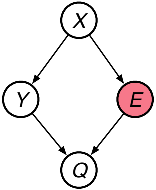
B	C	P(C|B)
b_0	c_0	1/5
b_0	c_1	4/5
b_1	c_0	3/5
b_1	c_1	2/5

B	D	P(D|B)
b_0	d_0	3/4
b_0	d_1	1/4
b_1	d_0	1/3
b_1	d_1	2/3

Properties of likelihood weighting

- It converges in the limit:

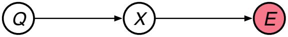
$$\lim_{N\to\infty} \frac{\sum_{i=1}^{N} I(q, q_i)\, P(E=e|X=x_i)}{\sum_{i=1}^{N} P(E=e|X=x_i)} = P(Q=q|E=e)$$



- It's more efficient than rejection sampling:

  No samples need to discarded.
  Descendants of evidence nodes are conditioned on evidence.

- But it can still be very slow:



The worst case for likelihood weighting is when rare evidence is descended from query nodes.

That's all folks!