Georgia Institute of Technology

CSIP Center for Signal & Image Processing

ECE/ML/CS/ISYE 8803

Approximate Inference in Graphical Models

Module 7: Part A
Particle-Based Methods

Faramarz Fekri

Center for Signal and Information Processing

Overview

- Particle-Based Inference
  - Direct Sampling
    - Example
  - Rejection Sampling
  - Likelihood Weighting
  - Importance Sampling
    - Normalized and Unnormalized cases

Read Chapter 12 of K&F

Inference

- Exact inference (already studied)
  - It is an NP-Hard problem in general
  - Is it hopeless?
    - No, approximate inference works efficiently with high accuracy

- Approximate inference
  - Particle-based methods
  - Global methods

Approximate Inference

- Particle-based methods
  - Generate instances (particles) that represent part of the probability mass
    - Random sampling method
    - Search deterministically for high probability assignments

- Global methods
  - Approximate the distribution in its entirety
    - Apply exact inference on a simpler network (e.g. meanfield)
    - Perform inference in the original network but approximate some steps of the process (e.g., either ignore or approximate some intermediate results)

Particle-Based Inference

Particle types:

- Full particles – complete assignments to all variables
- Distributional particles – assignment to part of the variables (Rao-Blackwellized particles)

Particle generation:

- Generate particles deterministically
- Generate particles by sampling

Particle-Based Method via Sampling

- Full particle methods
  - Sampling methods
    - Forward/direct sampling
      - Simple
      - Works only for easy distributions
    - Rejection sampling
      - Create samples like direct sampling
      - Only count samples consistent with given evidence
    - Importance sampling
      - Create samples like direct sampling
      - Assign weights to samples
    - Markov chain Monte Carlo
      - 1. Gibbs sampling, and 2. Metropolis-Hastings
        - Often used for high-dimensional problem
        - Use variables and its Markov blanket for sampling

Why Sampling?

- Previous inference tasks focus on obtaining the entire posterior distribution $P(X_i|e)$
  - Often we want to take expectations
    - Mean $\mu_{X_i|e} = E[X_i|e] = \int X_i P(X_i|e) dX_i$
    - Variance $\sigma^2_{X_i|e} = E\left[(X_i - \mu_{X_i|e})^2 | e \right] = \int (X_i - \mu_{X_i|e})^2 P(X_i|e) dX_i$
    - More general $E[f] = \int f(X) P(X|e) dX$, can be difficult to do it analytically
  - Key idea: approximate expectation by sample average

$$E[f] \approx \frac{1}{N} \sum_{i=1}^{N} f(x_i)$$

where $x_1, \ldots, x_N \sim P(X|e)$ independently and identically

Estimate P(y) by: $P(y) \approx \frac{1}{M} \sum_{m=1}^{M} \underbrace{\mathbf{1}\{x[m](y) = y\}}_{f(x[m])}$

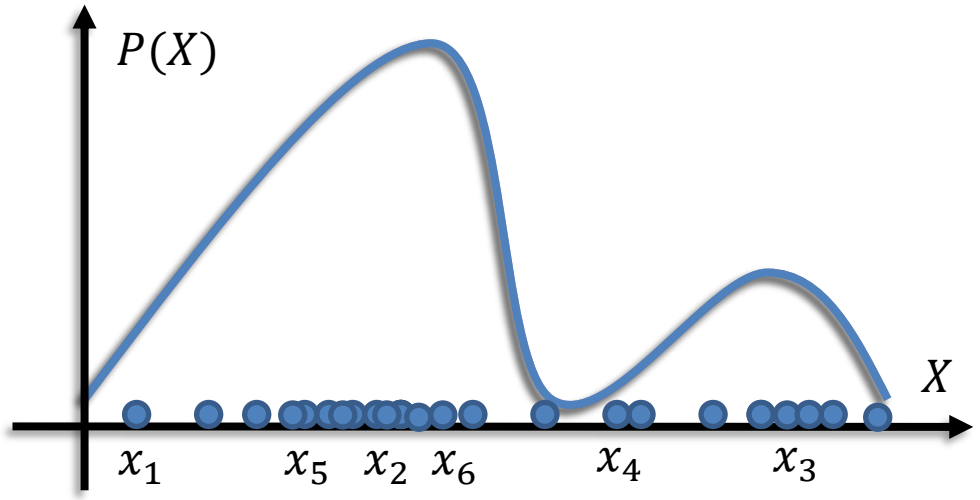
Sampling

General Framework:

- Generate samples (particles) x[1],...,x[M] from P
- Estimate function by $E_P(f) \approx \frac{1}{M} \sum_{m=1}^{M} f(x[m])$

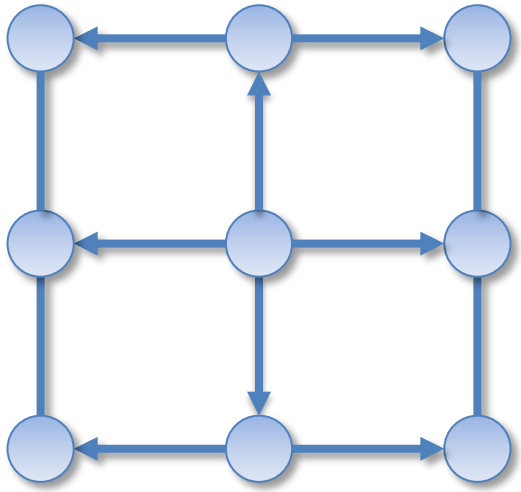
- Samples: points from the domain of a distribution $P(X)$
- The higher the $P(x)$, the more likely we see x in the sample

- Mean $\hat{\mu} = \frac{1}{N} \sum_i x_i$
- Variance $\hat{\sigma}^2 = \frac{1}{N} \sum_i (x_i - \hat{\mu})^2$



Challenges in Sampling

- How to draw sample from a given distribution?
  - Not all distributions can be trivially sampled



- How to make better use of samples (not all samples are equally useful)?

- How do know we've sampled enough?

Number of Samples Needed?

$$E_P(f) \approx \frac{1}{M}\sum_{m=1}^{M} f(x[m]) \equiv \hat{f}_{\mathcal{X}}$$

The subscript in $\hat{f}_{\mathcal{X}}$ emphasises that the approximation is dependent on the set of samples drawn. This sample approximation holds for both discrete and continuous variables.

How close $\hat{f}_{\mathcal{X}}$ gets to $E_P(f)$?

$$P(y_0) = E_p[1\{y = y_0\}] = \sum_m p(m)\ 1\{y[m] = y_0\}$$

Hoeffding's inequality: $P[\ \widehat{P}(y_0) \notin \{P(y_0) \pm \varepsilon\}] \le 2\, e^{-2M\varepsilon^2}$

How many samples are required to achieve an estimate whose error is bounded by ε, with probability at least $1 - \delta$.

$$\Rightarrow \quad M \geq \frac{\ln(2/\delta)}{2\epsilon^2}$$

Number of Samples Needed?

$$\hat{P}(y_0) = E_p[1\{y = y_0\}] = \sum_m p(m)\ 1\{y[m] = y_0\}$$

Using Chernoff bound:

$$P_{\mathcal{D}}(\hat{P}_{\mathcal{D}}(\boldsymbol{y}) \notin P(\boldsymbol{y})(1 \pm \epsilon)) \leq 2e^{-MP(\boldsymbol{y})\epsilon^2/3}$$

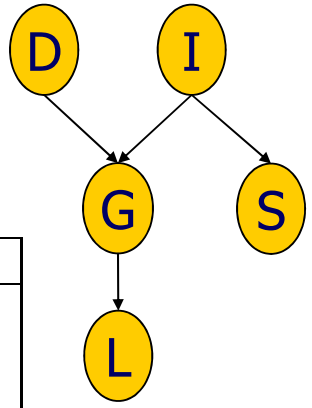
- To get a relative error < ε, with probability 1-δ, we need

$$M \geq 3\frac{\ln(2/\delta)}{P(\mathbf{y})\varepsilon^2}$$

  - Note that number of samples grows inversely with P(y)
  - For small P(y) we need many samples, otherwise we report P(y)=0

Example: Direct Sampling (I)

d^0	d^1
0.4	0.6



i^0	i^1
0.7	0.3

I	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

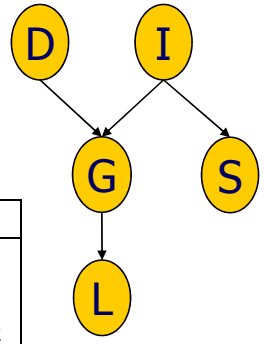
D	***I***	g^0	g^1	g^2
d^0	i^0	0.3	0.4	0.3
d^0	i^1	0.05	0.25	0.7
d^1	i^0	0.9	0.08	0.02
d^1	i^1	0.5	0.3	0.2

G	l^0	l^1
g^0	0.1	0.9
g^1	0.4	0.6
g^2	0.99	0.01

Samples

D	***I***	***G***	***S***	***L***

d^0	d^1
0.4	0.6



i^0	i^1
0.7	0.3

I	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

D	***I***	g^0	g^1	g^2
d^0	i^0	0.3	0.4	0.3
d^0	i^1	0.05	0.25	0.7
d^1	i^0	0.9	0.08	0.02
d^1	i^1	0.5	0.3	0.2

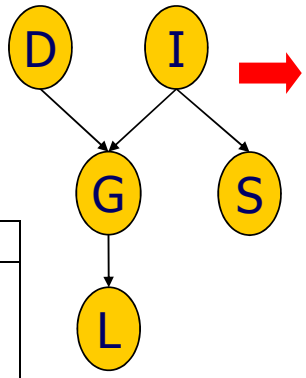
G	l^0	l^1
g^0	0.1	0.9
g^1	0.4	0.6
g^2	0.99	0.01

Samples

D	***I***	***G***	***S***	***L***
d^1				

Example: Direct Sampling (II)

d^0	d^1
0.4	0.6



i^0	i^1
0.7	0.3

I	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

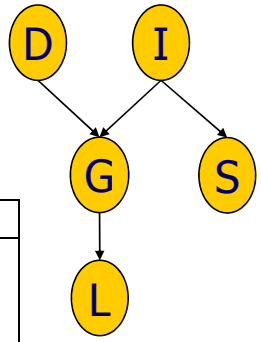
D	***I***	g^0	g^1	g^2
d^0	i^0	0.3	0.4	0.3
d^0	i^1	0.05	0.25	0.7
d^1	i^0	0.9	0.08	0.02
d^1	i^1	0.5	0.3	0.2

G	l^0	l^1
g^0	0.1	0.9
g^1	0.4	0.6
g^2	0.99	0.01

Samples

D	***I***	***G***	***S***	***L***
d^1	i^1			

d^0	d^1
0.4	0.6



i^0	i^1
0.7	0.3

I	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

D	***I***	g^0	g^1	g^2
d^0	i^0	0.3	0.4	0.3
d^0	i^1	0.05	0.25	0.7
d^1	i^0	0.9	0.08	0.02
d^1	i^1	0.5	0.3	0.2

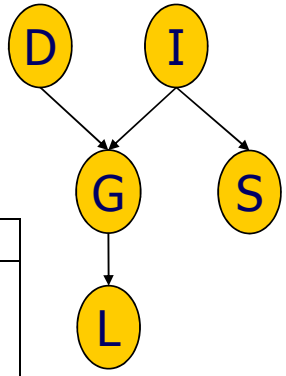
G	l^0	l^1
g^0	0.1	0.9
g^1	0.4	0.6
g^2	0.99	0.01

Samples

D	***I***	***G***	***S***	***L***
d^1	i^1	g^2		

Example: Direct Sampling (III)

d^0	d^1
0.4	0.6



i^0	i^1
0.7	0.3

I	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

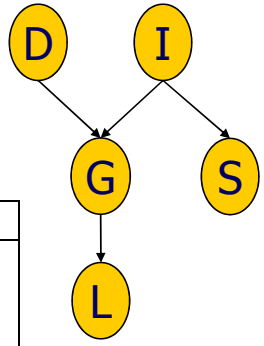
D	***I***	g^0	g^1	g^2
d^0	i^0	0.3	0.4	0.3
d^0	i^1	0.05	0.25	0.7
d^1	i^0	0.9	0.08	0.02
d^1	i^1	0.5	0.3	0.2

G	l^0	l^1
g^0	0.1	0.9
g^1	0.4	0.6
g^2	0.99	0.01

Samples

D	***I***	***G***	***S***	***L***
d^1	i^1	g^2	s^1	

d^0	d^1
0.4	0.6



i^0	i^1
0.7	0.3

I	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

D	***I***	g^0	g^1	g^2
d^0	i^0	0.3	0.4	0.3
d^0	i^1	0.05	0.25	0.7
d^1	i^0	0.9	0.08	0.02
d^1	i^1	0.5	0.3	0.2

G	l^0	l^1
g^0	0.1	0.9
g^1	0.4	0.6
g^2	0.99	0.01

Samples

D	***I***	***G***	***S***	***L***
d^1	i^1	g^2	s^1	l^0

Direct Sampling Pseudocode

- Let $X_1, \ldots, X_n$ be a topological order of the variables
- For i = 1,...,n
  - Sample x_i from $P(X_i \mid pa_i)$
    - (Note: since $Pa_i \subseteq \{X_1,\ldots,X_{i-1}\}$, we already assigned values to them)
- return $x_1, \ldots, x_n$

- Estimate function by: $E_P(f) \approx \frac{1}{M}\sum_{m=1}^{M} f(x[m])$

- Estimate P(y) by: $P(y) \approx \frac{1}{M}\sum_{m=1}^{M} \mathbf{1}\{x[m](y) = y\}$

Rejection Sampling (I)

Motivation:

- to sample from distribution: $P(X) = \frac{1}{Z} f(X)$
  - But $P(X)$ is difficult to sample, while f(X) is easy to evaluate.

- Key idea: sample from a simpler distribution ***Q(X)***, and then select samples: (proof next page)

Rejection sampling (choose M s.t. $f(X) \leq M\ Q(X)$)

- $i = 1$
- Repeat until $i = N$
  - $x \sim Q(X), u \sim U[0,1]$
  - if $u \leq \frac{f(X)}{M\ Q(X)}$, accept x and $i = i + 1$; otherwise, reject x

Reject sample with probability $1 - \frac{f(X)}{M\ Q(X)}$

Proof of Rejection Sampling

$$P(X) = \frac{1}{Z}\tilde{p}(x) \qquad\qquad Mq(x) \geq \tilde{p}(x)$$

q(x) is the proposal distribution, which is simple to sample.

Define:

$$S = \{(x,u) : u \leq \tilde{p}(x)/Mq(x)\},\ S_0 = \{(x,u) : x \leq x_0, u \leq \tilde{p}(x)/Mq(x)\}$$

Then the CDF of the accepted points is given by:

$$\begin{aligned} P(x \leq x_0 | x \text{ accepted}) &= \frac{P(x \leq x_0, x \text{ accepted})}{P(x \text{ accepted})} \\ &= \frac{\int\int \mathbb{I}((x,u) \in S_0) q(x) du dx}{\int\int \mathbb{I}((x,u) \in S) q(x) du dx} = \frac{\int_{-\infty}^{x_0} \tilde{p}(x) dx}{\int_{-\infty}^{\infty} \tilde{p}(x) dx} \end{aligned}$$

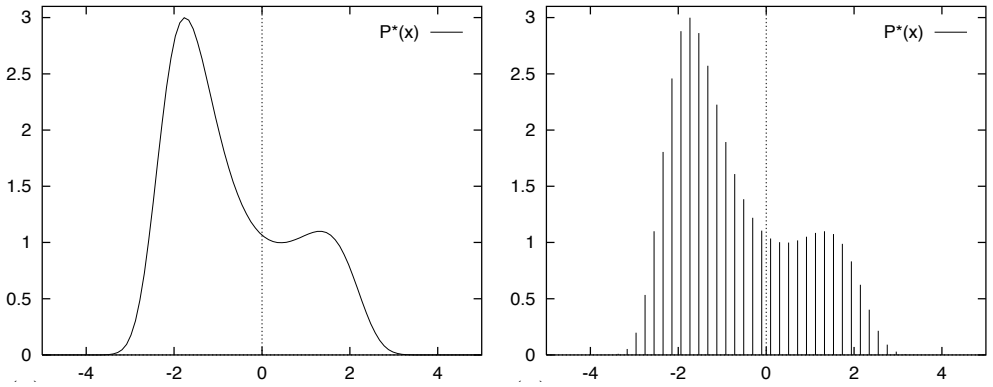
which is the CDF of p(x).

Example: Rejection Sampling

- Imagine that we wish to draw samples from the density P(x) = P*(x)/Z where

$$P^*(x) = \exp\left[0.4(x-0.4)^2 - 0.08x^4\right], \quad x \in (-\infty, \infty)$$

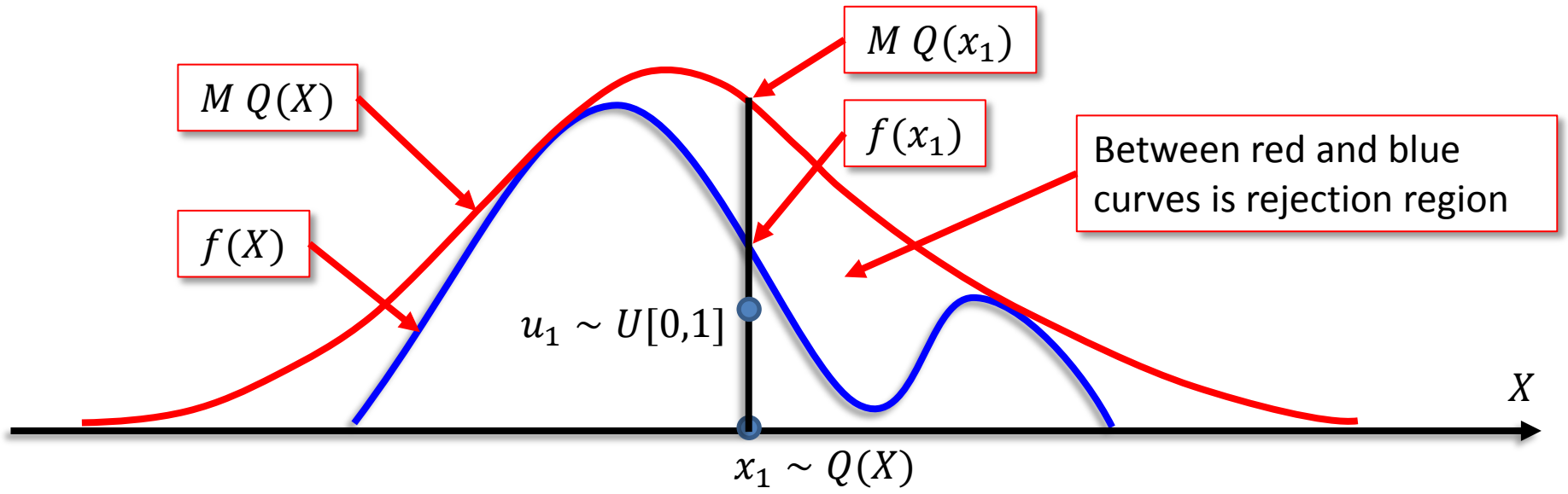
- We can plot P*(x) but that does not mean we can draw samples from it.
- We don't know the normalizing constant Z.



- Discretize variable x and ask for samples from the discrete probability distribution over a finite set of uniformly spaced points {xi}.
- If we evaluate p*i = P*(xi) at each x_i, we can compute Z.
- What if N dim for large N?

Rejection Sampling (II)

- Sample $x \sim Q(X)$ and reject with probability $1 - \frac{f(X)}{M\,Q(X)}$



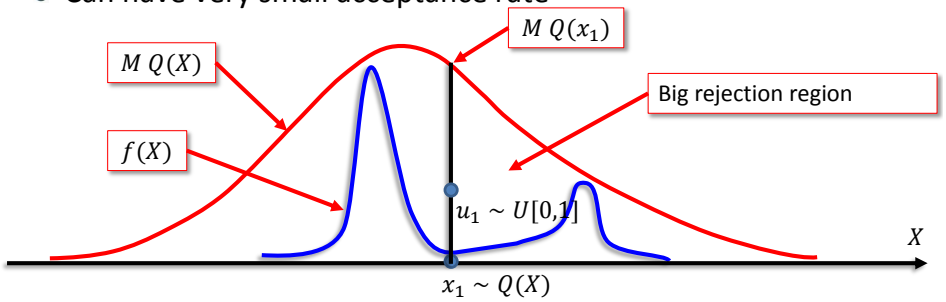
- How efficient is this method?

Average acceptance rate: $P\{\text{acceptance}\} = \int \frac{f(z)}{MQ(z)} Q(z).dz = \frac{1}{M} \int f(z).dz$

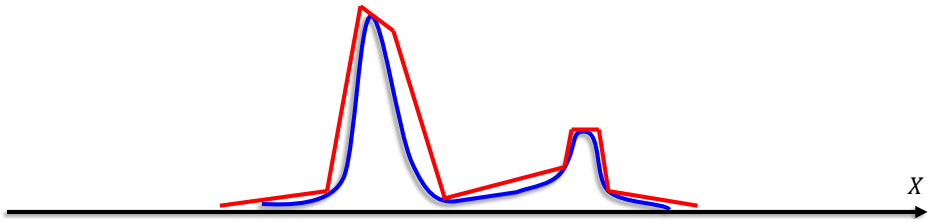
- Need to choose M as small as possible while still satisfying $f(X) \leq M\,Q(X)$

Drawback of rejection sampling

- Can have very small acceptance rate

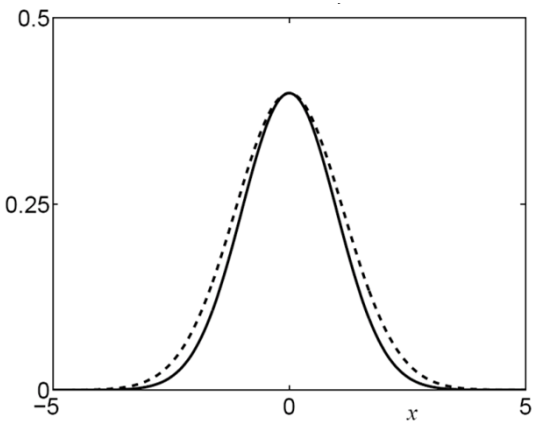


- Adaptive rejection sampling: use envelop function for Q



Pitfall:

- Using Q=$\mathcal{N}(\mu,\sigma_q^{2/d})$ to sample P=$\mathcal{N}(\mu,\sigma_p^{2/d})$
- If σ_q exceeds σ_p by 1%, and dimensional=1000,
- The optimal acceptance rate k=$(\sigma_q/\sigma_p)^d \approx 1/20{,}000$
- Big waste of samples!



Rejection sampling is a potentially useful method of drawing independent samples in very low dimensions, but is likely to be impractical/inefficient in higher dimensions.

Sampling with Likelihood Weighting (I)

- Motivation:
- Wish to sample ***P(Y/e)***.
  - We can do so with rejection sampling:
    - Generate samples as in forward sampling
    - Reject samples in which ***E≠e***
    - Estimate function from accepted samples
  - Problem: if evidence is unlikely (e.g., ***P(e)=0.001***)) then we generate many rejected samples

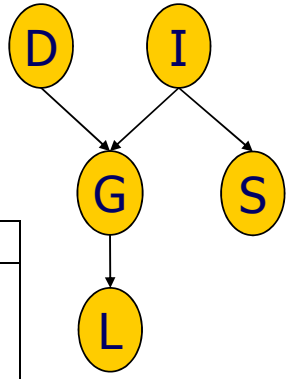
- Question: Can we ensure that all of our samples satisfy ***E=e*** in sample generation?

Sampling with Likelihood Weighting (II)

- Idea: when sampling a variable $Y \in E$, set $Y=e$
- Problem: we wish to sample from the posterior $P(X/e)$. However, our sampling process still samples from $P(X)$.
- Solution: weigh each sample by the joint probability of setting each variable to its evidence/observed value.
- Normalization, eventually we are sampling from $P(X,e)$, which can be then normalized to obtain $P(X/e)$ for a query of interest.

Example: Likelihood Weighting (I)

d^0	d^1
0.4	0.6



i^0	i^1
0.7	0.3

I	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

D	***I***	g^0	g^1	g^2
d^0	i^0	0.3	0.4	0.3
d^0	i^1	0.05	0.25	0.7
d^1	i^0	0.9	0.08	0.02
d^1	i^1	0.5	0.3	0.2

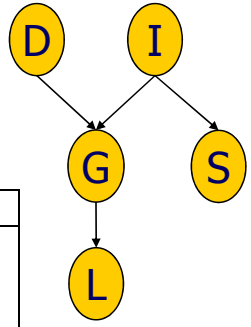
G	l^0	l^1
g^0	0.1	0.9
g^1	0.4	0.6
g^2	0.99	0.01

$E = \{S=s^1, G=g^2\}$

Samples

D	***I***	***G***	***S***	***L***	***W***

d^0	d^1
0.4	0.6



i^0	i^1
0.7	0.3

I	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

D	***I***	g^0	g^1	g^2
d^0	i^0	0.3	0.4	0.3
d^0	i^1	0.05	0.25	0.7
d^1	i^0	0.9	0.08	0.02
d^1	i^1	0.5	0.3	0.2

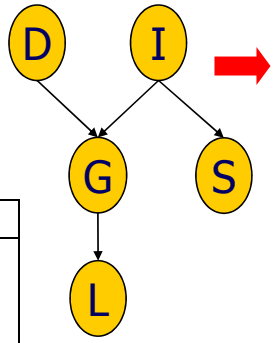
G	l^0	l^1
g^0	0.1	0.9
g^1	0.4	0.6
g^2	0.99	0.01

Samples

D	***I***	***G***	***S***	***L***	***W***
d^1					1

Example: Likelihood Weighting (II)

d^0	d^1
0.4	0.6



i^0	i^1
0.7	0.3

I	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

D	***I***	g^0	g^1	g^2
d^0	i^0	0.3	0.4	0.3
d^0	i^1	0.05	0.25	0.7
d^1	i^0	0.9	0.08	0.02
d^1	i^1	0.5	0.3	0.2

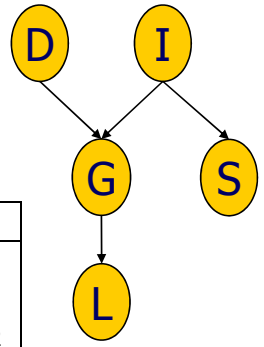
G	l^0	l^1
g^0	0.1	0.9
g^1	0.4	0.6
g^2	0.99	0.01

$E = \{S=s^1, G=g^2\}$

Samples

D	***I***	***G***	***S***	***L***	***W***
d^1	i^1				1

d^0	d^1
0.4	0.6



i^0	i^1
0.7	0.3

I	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

D	***I***	g^0	g^1	g^2
d^0	i^0	0.3	0.4	0.3
d^0	i^1	0.05	0.25	0.7
d^1	i^0	0.9	0.08	0.02
d^1	i^1	0.5	0.3	0.2

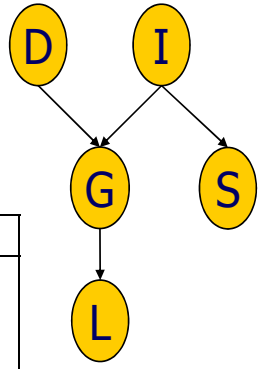
G	l^0	l^1
g^0	0.1	0.9
g^1	0.4	0.6
g^2	0.99	0.01

Samples

D	***I***	***G***	***S***	***L***	***W***
d^1	i^1	g^2			0.2

Example: Likelihood Weighting (III)

d^0	d^1
0.4	0.6



i^0	i^1
0.7	0.3

I	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

D	*I*	g^0	g^1	g^2
d^0	i^0	0.3	0.4	0.3
d^0	i^1	0.05	0.25	0.7
d^1	i^0	0.9	0.08	0.02
d^1	i^1	0.5	0.3	0.2

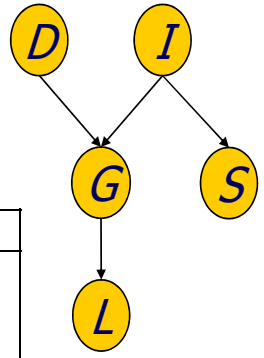
G	l^0	l^1
g^0	0.1	0.9
g^1	0.4	0.6
g^2	0.99	0.01

$E = \{S=s^1, G=g^2\}$

Samples

D	*I*	*G*	*S*	*L*	*W*
d^1	i^1	g^2	s^1		0.16

d^0	d^1
0.4	0.6



i^0	i^1
0.7	0.3

I	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

D	*I*	g^0	g^1	g^2
d^0	i^0	0.3	0.4	0.3
d^0	i^1	0.05	0.25	0.7
d^1	i^0	0.9	0.08	0.02
d^1	i^1	0.5	0.3	0.2

G	l^0	l^1
g^0	0.1	0.9
g^1	0.4	0.6
g^2	0.99	0.01

Samples

D	*I*	*G*	*S*	*L*	*W*
d^1	i^1	g^2	s^1	l^0	0.16

Likelihood Weighting Pseudocode

$w_i = 1$

- Let $X_1, \ldots, X_n$ be a topological order of the variables
- For i = 1,...,n
  - If $X_i \notin E$
    - Sample x_i from $P(X_i \mid pa_i)$
  - If $X_i \in E$
    - Set $X_i = E[x_i]$
    - Set $w_i = w_i \cdot P(E[x_i] \mid pa_i)$
- return w_i and $x_1, \ldots, x_n$

- Estimate P(y|E) by: $P(y \mid e) \approx \frac{\sum_{m=1}^{M} w[m]\mathbf{1}\{x[m](y) = y\}}{\sum_{m=1}^{M} w[m]}$

(Unnormalized) Importance Sampling (I)

- Generalization of likelihood weighting sampling.
- Suppose sampling from $P(X)$ is hard
- Idea: to estimate a function relative to $P(X)$, rather than sampling from the distribution $P(X)$, sample from another simpler distribution Q(X)
- If $Q(X)$ dominates P(X) (i.e., $Q(X) > 0$ whenever $P(X) > 0$), we can sample from Q and re-weight.

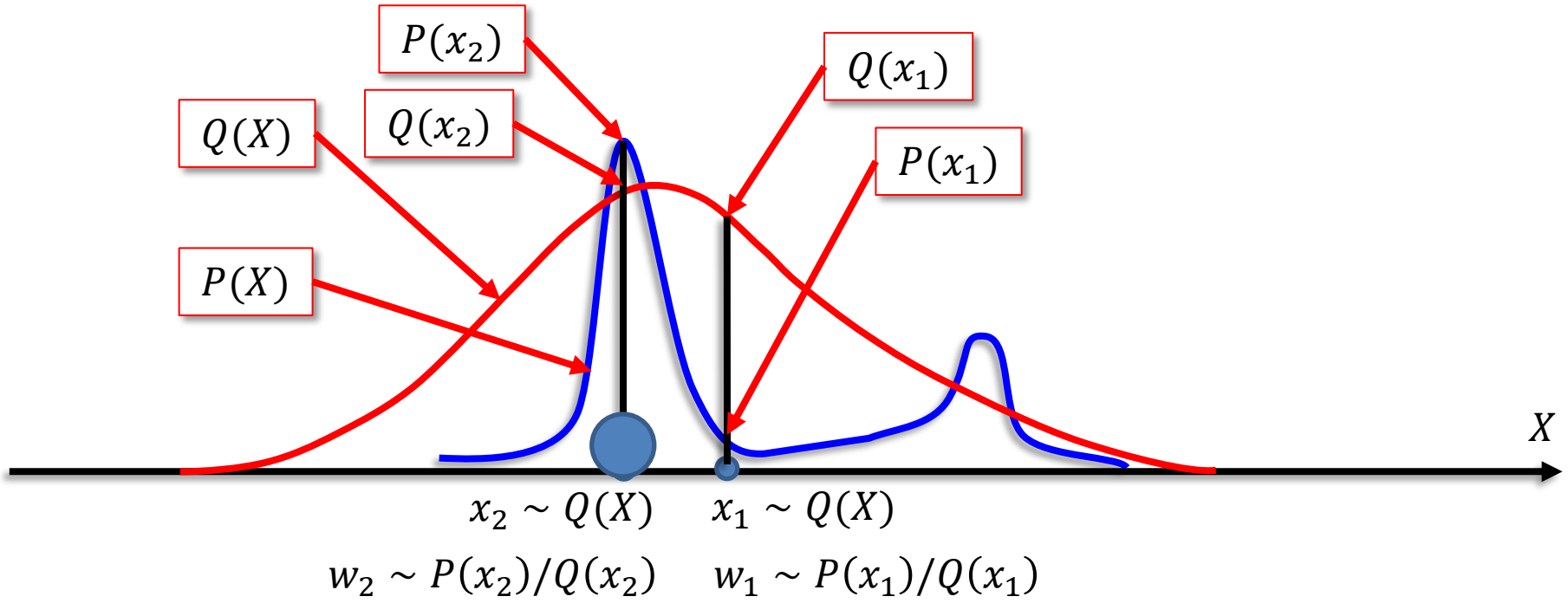
- P is called the target distribution
- Q is called the proposal or the sampling distribution
- Requirement from Q: P(x) > 0 → Q(x) > 0

(Unnormalized) Importance Sampling (II)

- Q does not 'ignore' any non-zero probability events in P
- In practice, performance depends on similarity between Q and P

- Suppose, we wish to estimate a function $f(X)$ relative to P(X):
  - $E_P[f(X)] = \int f(X)P(X)dX) = \int f(X)\frac{P(X)}{Q(X)}Q(X)dX$
    - Sample $x_i \sim Q(X)$
    - Reweight sample x_i with $w_i = \frac{P(x_i)}{Q(x_i)}$
    - approximate expectation with $\frac{1}{N}\sum_i f(x_i)w_i$
      - Can show that estimator variance decreases with more samples N
      - Can show that Q=P is the lowest variance estimator

(Unnormalized) Importance Sampling (III)



Instead of rejecting samples, we properly weight samples

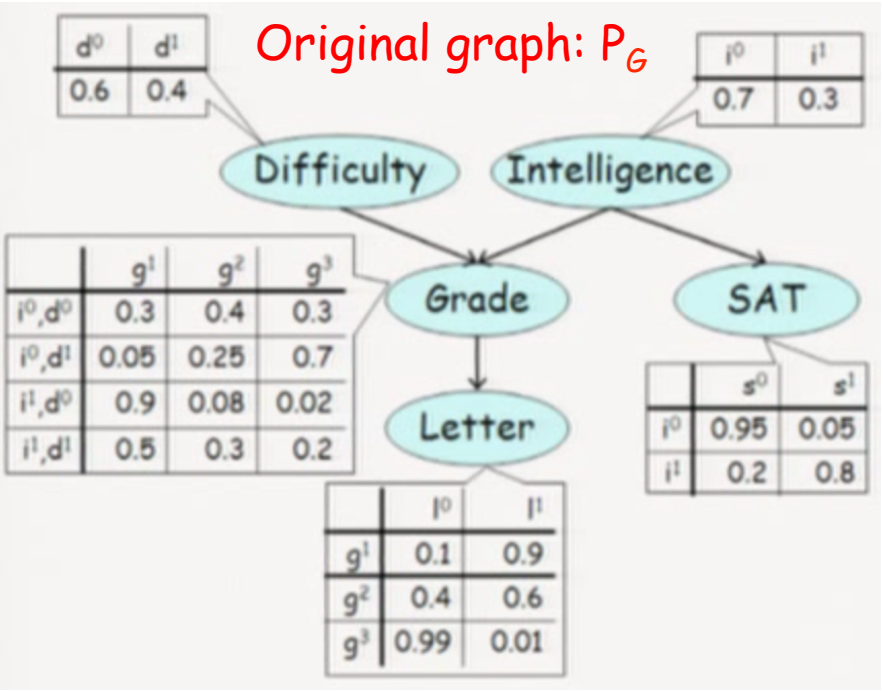
Normalized Importance Sampling

- Un-normalized importance sampling assumes it can evaluate ***P*** at ***X***.
  - Usually we know ***P*** up to a constant:
  - e.g., Posterior distribution ***P(X|E=e) = P(X,E=e)/a*** where ***a=P(E=e)***.
  - e.g., in MRF, $P(X) = \frac{1}{Z} f(X)$, and don't know ***Z*** (and ***Z*** is difficult to compute).

- To compute $E_P[g(X)]$, We can get around the normalization of ***Z*** by normalizing the importance weight: (Assume: $P(X) = \frac{1}{Z} f(X)$)
  - $r(X) = \frac{f(X)}{Q(X)} \Rightarrow E_Q[r(X)] = \int \frac{f(X)}{Q(X)} Q(X)dX = \int f(X)dX = Z$
  - $E_P[g(X)] = \int g(X)P(X)dX = \frac{1}{Z}\int g(X)\frac{f(X)}{Q(X)}Q(X)dX = \frac{\int g(X)r(X)Q(X)dX}{\int r(X)Q(X)dX} \approx \frac{\sum_i g(x_i)r(x_i)}{\sum_i r(x_i)}$

$define\ w_i = \frac{r(x_i)}{\sum_i r(x_i)}\ as\ the\ new\ importance\ weight$

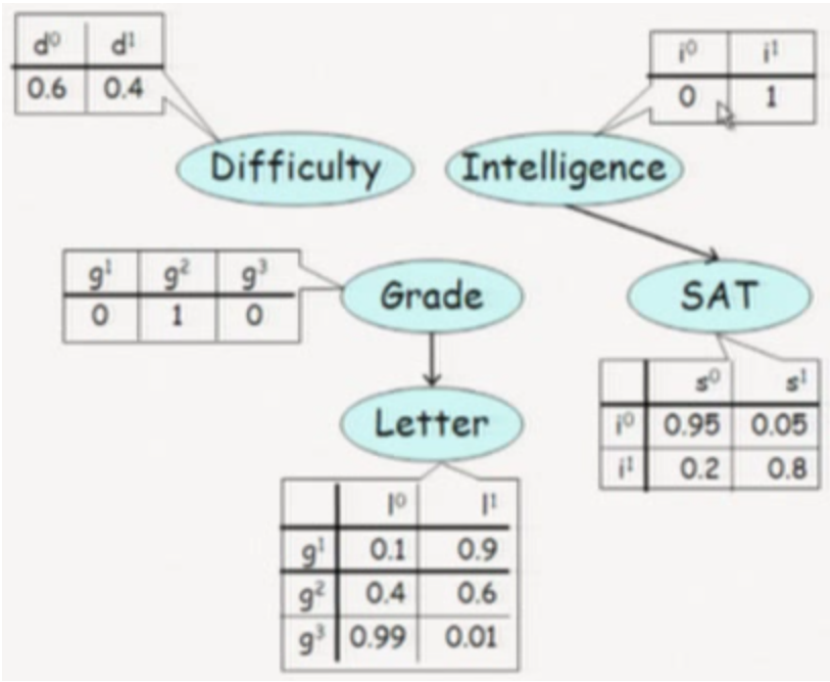
Example: Importance Sampling



In this example, importance sampling helps us to generate samples from P_G that are also consistent with evidence:

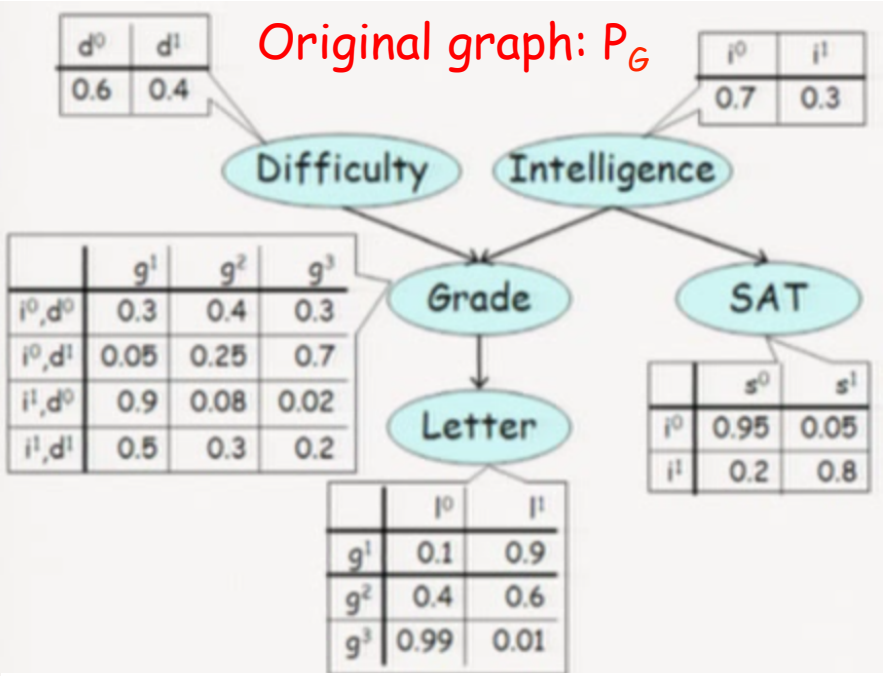
Evidence: $e=\{g^2, i^1\}$

Mutilated graph: P_M



Proposal distribution P_M can be obtained by applying chain rule on the mutilated Bayesian network

Finishing Example: Importance Sampling



Wish to get samples from $P_G(X|E=e)$ so that for example to compute:

$$P(X = x \mid E = e) = \frac{P(X = x, e)}{P(E = e)} \quad X: \text{all variables except } E$$

Evidence e={g^2, i^1}

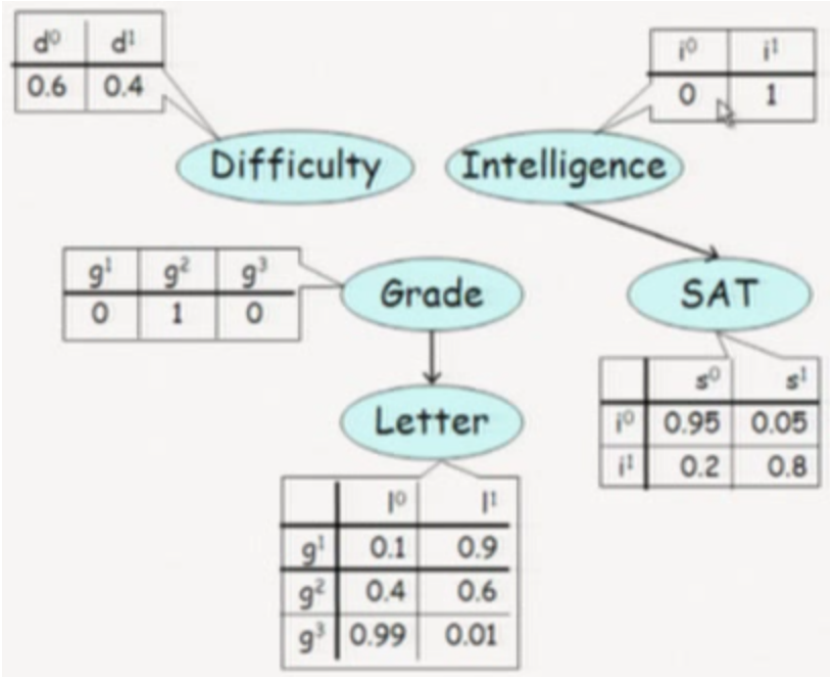
The unnormalized posterior is P'(x) = P(x, e)

Draw M samples from proposal dist. P_M via mutilated graph.

$$r_m = \frac{P'(x_m)}{P_M(x_m)} \quad \text{for each sample } x_m$$

$$\hat{P}(X = x^{(0)} \mid E = e) = \frac{\sum_m r_m \, 1\{X = x^{(0)}\}}{\sum_m r_m}$$

Mutilated graph: P_M



Likelihood Weighting as Importance Sampling in BN

- Target distribution P'(**X**,**e**) in graph P_G Likelihood Weighting
- Proposal distribution Q(**X**,**e**)=$P_{E=e}$(**X**,**e**) Mutilated graph P_M
- Then in Bayesian Networks: "Likelihood weighting" is equivalent to "normalized importance sampling" with the above proposal distributions

Proof:

- LW estimate: $$P(y \mid e) \approx \frac{\sum_{m=1}^{M} w[m]\mathbf{1}\{x[m](Y) = y\}}{\sum_{m=1}^{M} w[m]}$$
- IS estimate: $$P(y \mid e) \approx \frac{\sum_{m=1}^{M} P'(x[m]) / Q(x[m])\mathbf{1}\{x[m](Y) = y\}}{\sum_{m=1}^{M} P'(x[m]) / Q(x[m])}$$
- But w[m] in LW precisely comes out to be P'(x[m])/Q(x[m])
  - Since

$$\frac{P'(x[m])}{Q(x[m])} = \frac{\prod_{e_i \in \mathbf{E}} P(e_i \mid Pa(e_i)) \prod_{X_i \notin \mathbf{E}} P(X_i \mid Pa(X_i))}{\prod_{X_i \notin \mathbf{E}} P(X_i \mid Pa(X_i))} = \prod_{e_i \in \mathbf{E}} P(e_i \mid Pa(e_i)) = w[m]$$

Difference between Normalized and Unnormalized

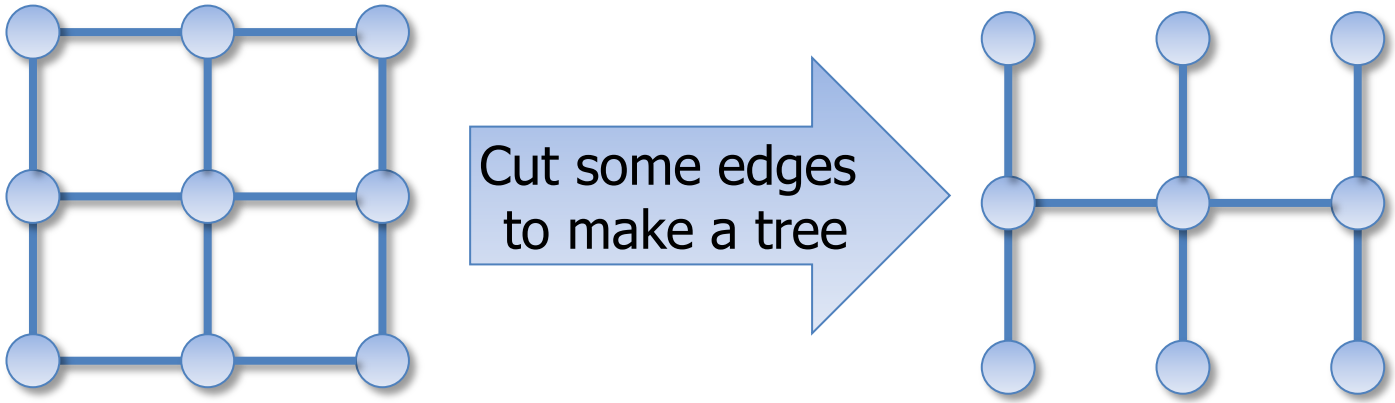
- Unnormalized importance sampling is unbiased.
- Normalized importance sampling is biased, e.g., for N=1

$$E_Q\left[\frac{g(x_1)w_1}{w_1}\right] = E_Q[g(x_1)] \neq E_P[g(x_1)]$$

- However, the normalized importance sampling usually has a lower variance than unnormalized counterpart.
- Most importantly, normalized importance sampling work for unnormalized distributions.
- For normalized sampling, one can also do resampling based on w_i (multinomial distribution with probability w_i for x_i)
  (Page 822 Kevin Murphy Book)

Example: Sample from MRF

- Use tree distribution Q as the proposal distribution



$$P(X_1, \dots, X_n) \propto \exp\left(\sum_{(ij)\in E} \theta_{ij} X_i X_j + \sum_{i\in V} \theta_i X_i\right)$$

$$Q(X_1, \dots, X_n) \propto \exp\left(\sum_{(ij)\in T} \theta_{ij} X_i X_j + \sum_{i\in V} \theta_i X_i\right)$$

has fewer terms

- Then use rejection sampling or importance sampling

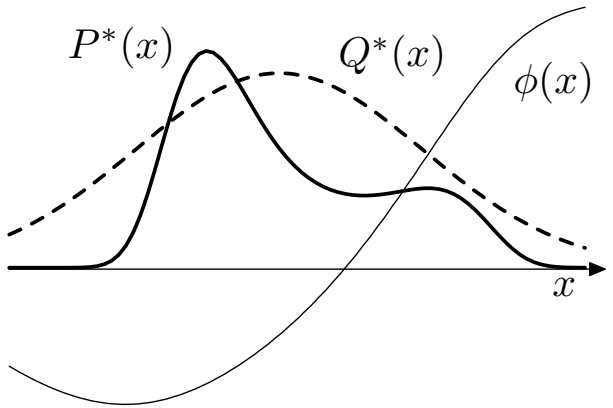
Normalized Importance Sampling

- Note that we do not need full access to Q(X)= Q*(X)/Z_Q, we only need to be able to draw samples from Q(X) and be able to evaluate Q*(X).
- Let P (x) = P*(x)/Z, and assume Z is unknown, and P (x) hard to sample but P*(x) is easy to evaluate at any x.
- Wish to compute $E_P[\varphi(x)]$ (Using normalized importance sampling):

$$w_r \equiv \frac{P^*(x(r))}{Q^*(x(r))}$$

Where sample x(r) is drawn from Q(X).

$$\hat{\Phi} \equiv \frac{\sum_r w_r \phi(x(r))}{\sum_r w_r}$$



Main Issues of Importance Sampling

- It is hard to estimate how reliable the estimator $\hat{\Phi}$ is.
- Depends on how well Q matches P
- If the proposal density Q(x) is small in a region where $|\varphi(x)P^*(x)|$ is large then it is quite possible, even after many points x(r) have been generated, that none of them will have fallen in that region.
- In this case, the estimate of $E_P[\varphi(x)]$ would be drastically wrong, and there would be no indication in the empirical variance that the true variance of the estimator $\hat{\Phi}$ is large.

Solution

- Use heavy tail Q.
- Weighted resampling (Sampling importance resampling (SIR))

1. Draw N samples from Q: $X_1 \quad X_N$
2. Constructing weights: $\hat{w}_1 \quad \hat{w}_N$,
3. Sub-sample x from $\{X_1 \quad X_N\}$ w.p. $(\hat{w}_1 \quad \hat{w}_N)$

$$\hat{w}_r = \frac{w_r}{\sum_r w_r}$$

Solving MAP via Sampling

- Choose a proper sampling method.
- Generate sufficient samples from the posterior
- For each X_i, the answer for MAP is the majority assignment from samples.

Example: Binary image