




Learning Problems in Real Applications & Approximate Inference

Lecture 14 – May 11, 2011
CSE 515, Statistical Methods, Spring 2011

Instructor: Su-In Lee
University of Washington, Seattle

Outline

- Learning problems in real applications: Robotics, AI, Natural Language Processing, Computational Biology, Computer Vision...
 - Robotic mapping
 - Part-of-speech tagging
 - Peptide identification in MSMS
 - Finding tumor-specific mutations
 - Collaborative filtering 
 - Discovering user clusters
 - Computer vision
 - Learning spatial context: using stuff to find things
 - Machine learning
 - Structured prediction
- Particle-based approximate inference

PARTICLE-BASED APPROXIMATE INFERENCE

CSE 515 – Statistical Methods – Spring 2011

35

Inference Complexity Summary

- NP-Hard
 - Exact inference
 - Approximate inference
 - with relative error
 - with absolute error < 0.5 (given evidence)
- Hopeless?
 - No, we will see many network structures that have provably efficient algorithms and we will see cases when approximate inference works efficiently with high accuracy

36

Approximate Inference

- **Particle-based methods**
 - Create **instances (particles)** that represent part of the probability mass
 - Random sampling
 - Deterministically search for high probability assignments
- **Global methods**
 - Approximate the distribution in its entirety
 - Use exact inference on a simpler (but close) network (e.g. meanfield)
 - Perform inference in the original network but approximate some steps of the process (e.g., ignore certain computations or approximate some intermediate results)

37

Particle-Based Methods

- **Particle definition**
 - Full particles – complete assignments to all variables
 - Distributional particles – assignment to part of the variables
- **Particle generation process**
 - Generate particles deterministically
 - Generate particles by sampling

General framework

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

38

Particle-Based Methods Overview

- Full particle methods
 - Sampling methods
 - ➔ ■ Forward sampling
 - Importance sampling
 - Markov chain Monte Carlo
 - Deterministic particle generation
- Distributional particles

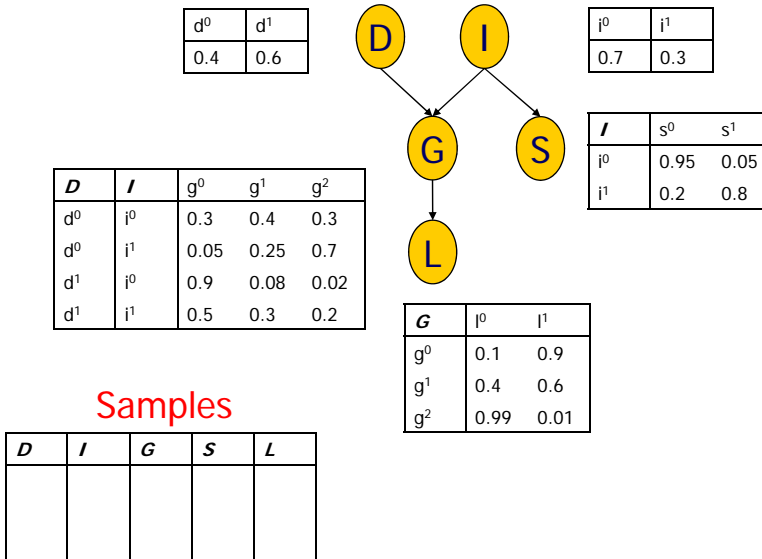
39

Forward Sampling

- Generate random samples from $P(X)$
 - Use the Bayesian network to generate samples
- Estimate function by $E_p(f) \approx \frac{1}{M} \sum_{m=1}^M f(x[m])$

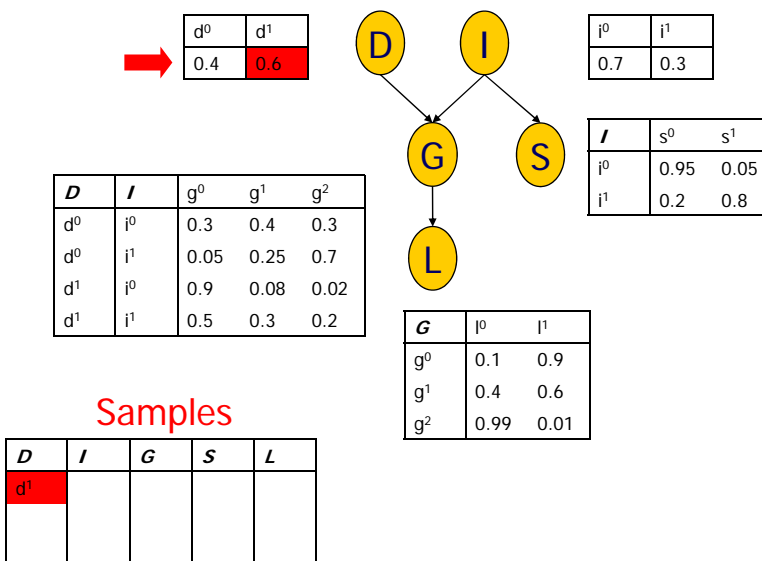
40

Forward Sampling



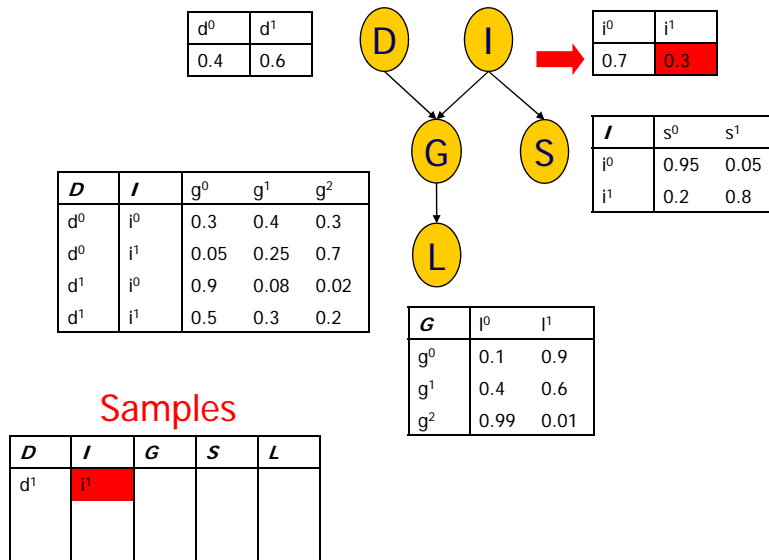
41

Forward Sampling



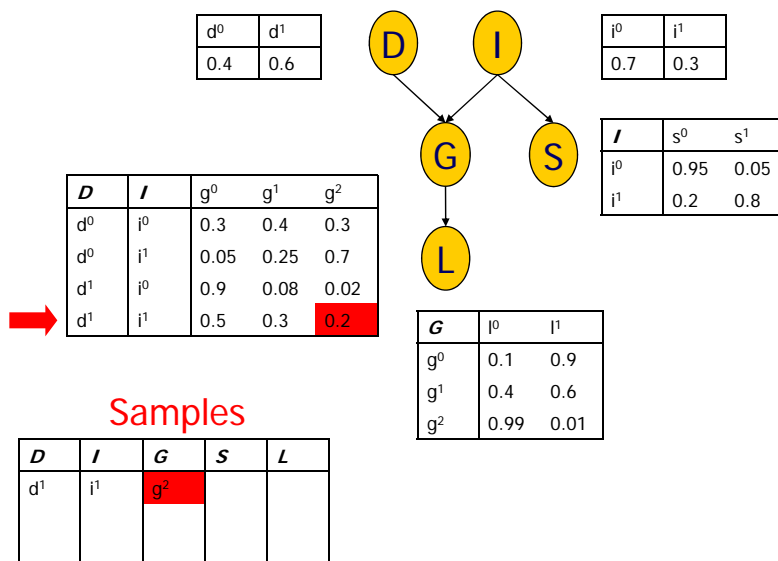
42

Forward Sampling



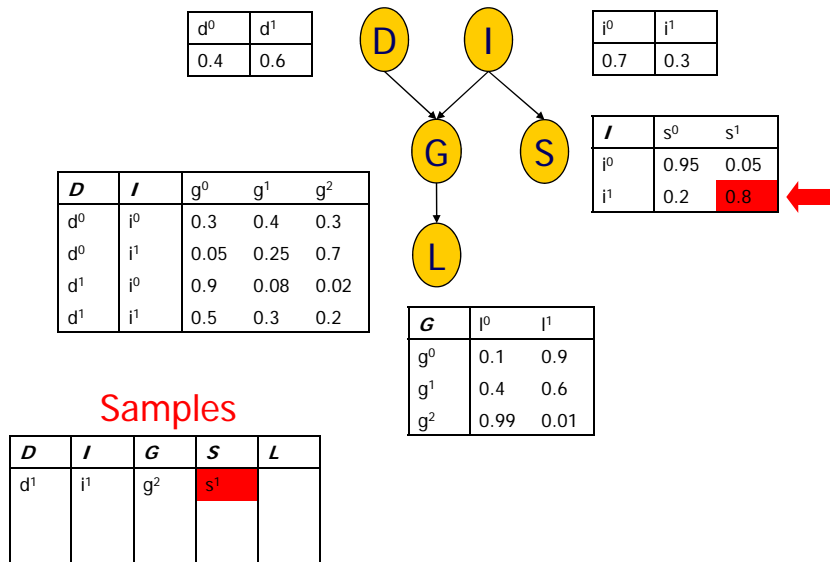
43

Forward Sampling



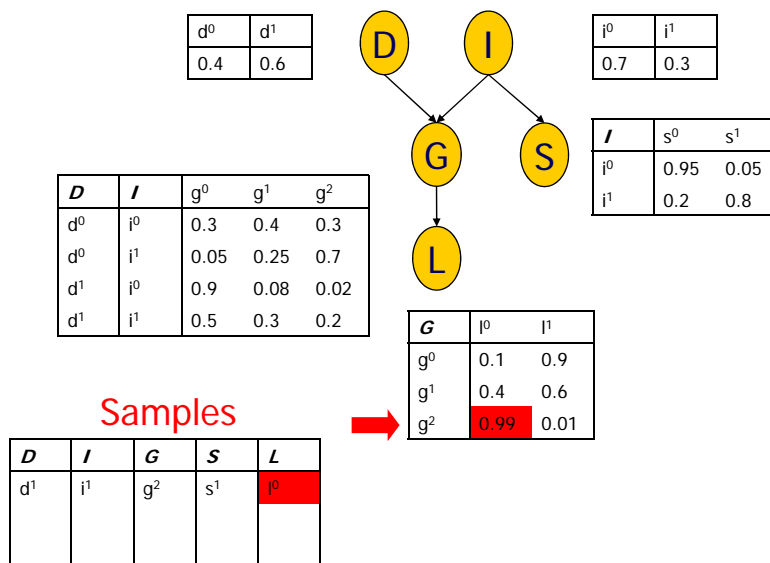
44

Forward Sampling



45

Forward Sampling



46

Forward Sampling

- Let X_1, \dots, X_n be a topological order of the variables
- For $i = 1, \dots, n$
 - Sample x_i from $P(X_i \mid \text{pa}_i)$
 - (Note: since $\text{pa}_i \subseteq \{X_1, \dots, X_{i-1}\}$, we already assigned values to them)
- return x_1, \dots, x_n
- Estimate function by: $E_P(f) \approx \frac{1}{M} \sum_{m=1}^M f(x[m])$
- Estimate $P(y)$ by: $P(y) \approx \frac{1}{M} \sum_{m=1}^M \mathbf{1}\{x[m](y) = y\}$

47

Forward Sampling

- **Sampling cost**
 - Per variable cost: $O(\log(\text{Val}|X_i|))$
 - Sample uniformly in $[0,1]$
 - Find appropriate value of all $\text{Val}|X_i|$ values that X_i can take
 - Per sample cost: $O(n \log(d))$ ($d = \max_i \text{Val}|X_i|$)
 - **Total cost: $O(Mn \log(d))$**
- **Number of samples needed**
 - To get a relative error $< \epsilon$, with probability $1-\delta$, we need
$$M \geq 3 \frac{\ln(2/\delta)}{P(\mathbf{y})\epsilon^2}$$
 - Note that number of samples grows inversely with $P(\mathbf{y})$
 - For small $P(\mathbf{y})$ we need many samples, otherwise we report $P(\mathbf{y})=0$


48

Rejection Sampling

- In general we need to compute $P(Y|e)$
- We can do so with *rejection sampling*
 - Generate samples as in forward sampling
 - **Reject** samples in which $E \neq e$
 - Estimate function from accepted samples
- **Problem:** if evidence is unlikely (e.g., $P(e)=0.001$) then we generate many rejected samples

49

Particle-Based Methods Overview

- Full particle methods
 - Sampling methods
 - Forward sampling
 -  Importance sampling
 - Markov chain Monte Carlo
 - Deterministic particle generation
- Distributional particles

50

Likelihood Weighting

- Can we ensure that all of our samples satisfy $E=e$?
 - **Solution:** when sampling a variable $X \in E$, set $X=e$
 - **Problem:** we are trying to sample from the posterior $P(X|e)$ but 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
 - In effect, we are sampling from $P(X,e)$ which we can normalize to then obtain $P(X|e)$ for a query of interest

51

Likelihood Weighting

Bayesian network structure: $D \rightarrow G$, $I \rightarrow G$, $I \rightarrow S$, $G \rightarrow L$. Evidence: $E = \{S=s^1, G=g^2\}$.

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

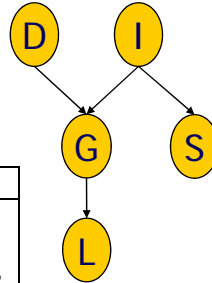
D	I	G	S	L	W

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

52

Likelihood Weighting

d ⁰	d ¹
0.4	0.6



i ⁰	i ¹
0.7	0.3

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

I	s ⁰	s ¹
i ⁰	0.95	0.05
i ¹	0.2	0.8

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

Samples

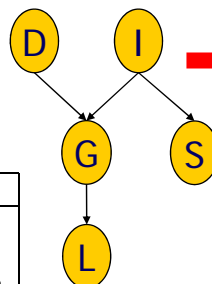
D	I	G	S	L	W
d ¹					1

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

53

Likelihood Weighting

d ⁰	d ¹
0.4	0.6



i ⁰	i ¹
0.7	0.3

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

I	s ⁰	s ¹
i ⁰	0.95	0.05
i ¹	0.2	0.8

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

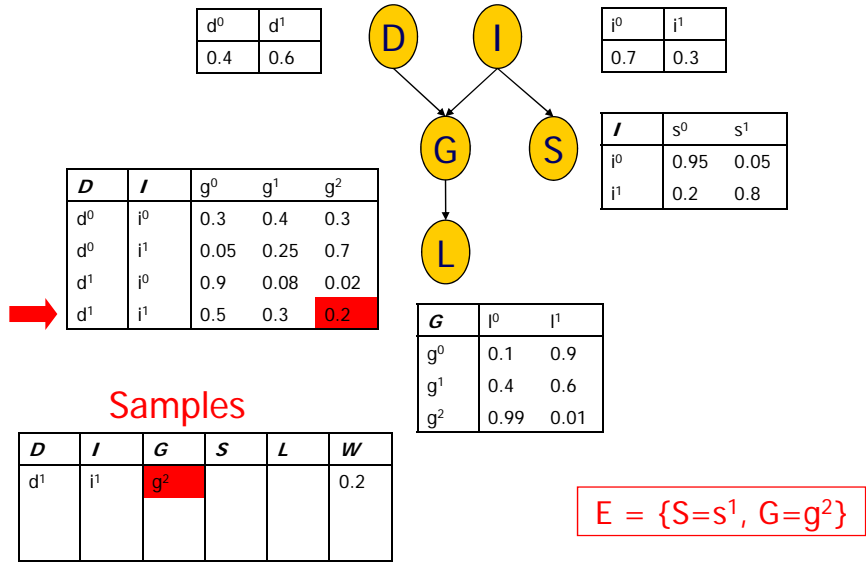
Samples

D	I	G	S	L	W
d ¹	i ¹				1

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

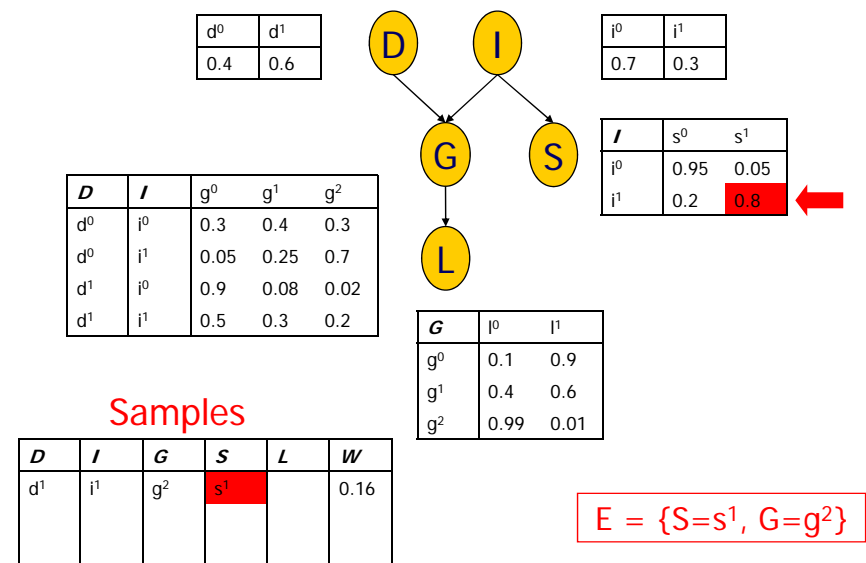
54

Likelihood Weighting



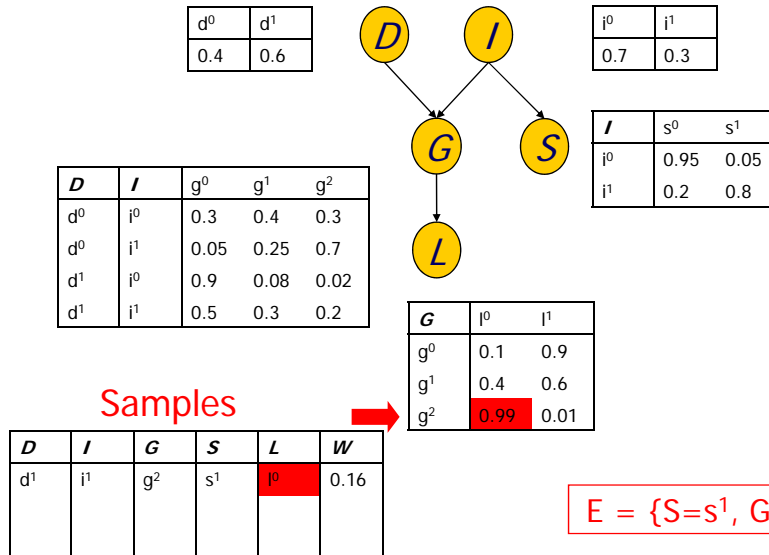
55

Likelihood Weighting



56

Likelihood Weighting



57

Likelihood Weighting

- Let X_1, \dots, X_n be a topological order of the variables
- For $i = 1, \dots, n$
 - 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, \dots, x_n

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

58

Importance Sampling

- Generalization of likelihood weighting sampling
- **Idea:** to estimate a function relative to P , rather than sampling from the distribution P , sample from another distribution Q
 - P is called the **target** distribution
 - Q is called the **proposal** or the **sampling** distribution
 - Requirement from Q : $P(x) > 0 \rightarrow Q(x) > 0$
 - Q does not 'ignore' any non-zero probability events in P
 - In practice, performance depends on similarity between Q and P

59

Acknowledgement

- These lecture notes were generated based on the slides from Prof Eran Segal.