Informatics 2D: Reasoning and Agents

Alex Lascarides

School of informatics
THE UNIVERSITY OF EDINBURGH

Lecture 25a: Approximate inference in BNs
Direct sampling methods

Where are we?

Last time ...

- Inference in Bayesian Networks
- Exact methods: enumeration, variable elimination algorithm
- Computationally intractable in the worst case

Today ...

- **Approximate Inference in Bayesian Networks**

Approximate inference in BNs

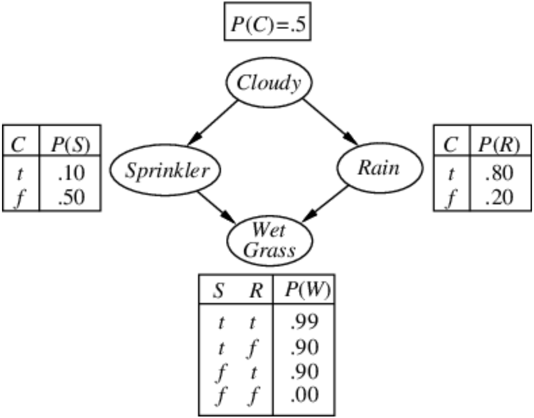
- Exact inference computationally very hard
- Approximate methods important, here randomised sampling algorithms
- **Monte Carlo** algorithms
- We will talk about two types of MC algorithms:
  1. Direct sampling methods (today)
  2. Markov chain sampling (next time)

Direct sampling methods

- Basic idea: generate samples from a known probability distribution
- Consider an unbiased coin as a random variable – sampling from the distribution is like flipping the coin
- It is possible to sample any distribution on a single variable given a set of random numbers from [0,1]
- Simplest method: generate events from network without evidence
  - Sample each variable in 'topological order'
  - Probability distribution for sampled value is conditioned on values assigned to parents

Example

- Consider the following BN and ordering $[Cloudy, Sprinkler, Rain, WetGrass]$:



Example

- Direct sampling process:
  - Sample from $\mathbf{P}(Cloudy) = \langle 0.5, 0.5 \rangle$, suppose this returns *true*
  - Sample from $\mathbf{P}(Sprinkler|Cloudy = true) = \langle 0.1, 0.9 \rangle$, suppose this returns *false*
  - Sample from $\mathbf{P}(Rain|Cloudy = true) = \langle 0.8, 0.2 \rangle$, suppose this returns *true*
  - Sample from $\mathbf{P}(WetGrass|Sprinkler = false, Rain = true) = \langle 0.9, 0.1 \rangle$, suppose this returns *true*
- Event returned=$[true, false, true, true]$

Direct sampling methods

- Generates samples with probability $S(x_1, \ldots, x_n)$

$$S(x_1, \ldots, x_n) = P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

i.e. in accordance with the distribution

- Answers are computed by counting the number $N(x_1, \ldots, x_n)$ of the times event $x_1, \ldots, x_n$ was generated and dividing by total number N of all samples
- In the limit, we should get

$$\lim_{n \to \infty} \frac{N(x_1, \ldots, x_n)}{N} = S(x_1, \ldots, x_n) = P(x_1, \ldots, x_n)$$

- If the estimated probability $\hat{P}$ becomes exact in the limit we call the estimate **consistent** and we write "$\approx$" in this sense, e.g.

$$P(x_1, \ldots, x_n) \approx N(x_1, \ldots, x_n)/N$$

Rejection sampling

- Purpose: to produce samples for hard-to-sample distribution from easy-to-sample distribution
- To determine $P(X|\mathbf{e})$ generate samples from the prior distribution specified by the BN first
- Then reject those that do not match the evidence
- The estimate $\hat{P}(X = x|\mathbf{e})$ is obtained by counting how often $X = x$ occurs in the remaining samples
- Rejection sampling is consistent because, by definition:

$$\hat{P}(X|\mathbf{e}) = \frac{\mathbf{N}(X,\mathbf{e})}{N(\mathbf{e})} \approx \frac{\mathbf{P}(X,\mathbf{e})}{P(\mathbf{e})} = \mathbf{P}(X|\mathbf{e})$$

Back to our example

- Assume we want to estimate $\mathbf{P}(Rain|Sprinkler = true)$, using 100 samples
  - 73 have $Sprinkler = false$ (rejected), 27 have $Sprinkler = true$
  - Of these 27, 8 have $Rain = true$ and 19 have $Rain = false$
- $\mathbf{P}(Rain|Sprinkler = true) \approx \alpha\langle 8, 19\rangle = \langle 0.296, 0.704\rangle$
- True answer would be $\langle 0.3, 0.7\rangle$
- But the procedure rejects too many samples that are not consistent with $\mathbf{e}$ (exponential in number of variables)
- Not really usable (similar to naively estimating conditional probabilities from observation)

Summary

Approximate inference in BNs

- Direct sampling
- rejection sampling
  - Can answer query of the form $\mathbf{P}(X|e)$
  - But wasteful. . .
- Next time: **likelihood weighting and MCMC**