CS 561 Artificial Intelligence
Lecture 6-7 Inference in Bayesian Networks

Rashmi Dutta Baruah

Department of Computer Science & Engineering

IIT Guwahati

भारतीय प्रौद्योगिकी संस्थान गुवाहाटी

Indian Institute of Technology Guwahati

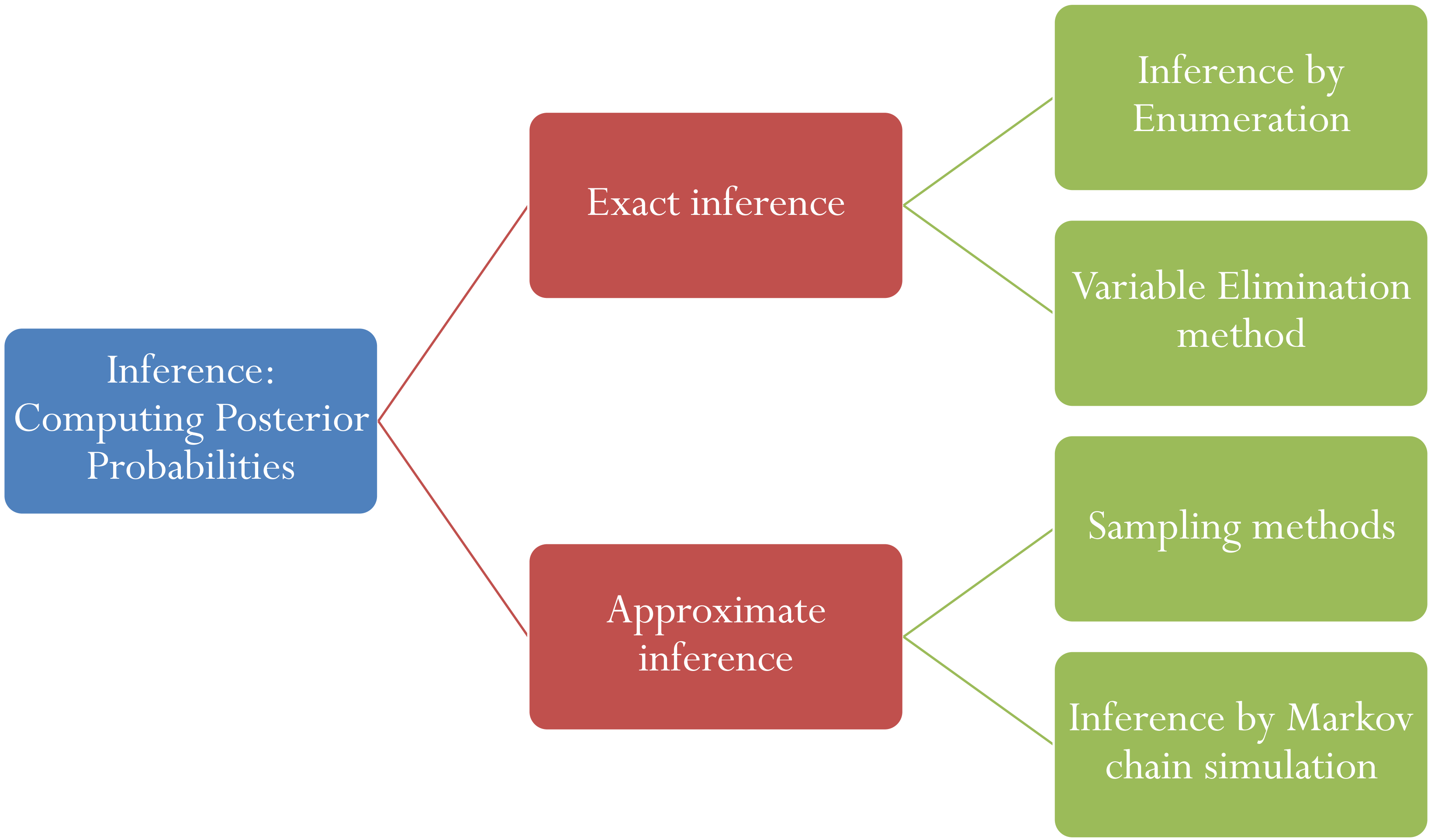
Guwahati - 781039, INDIA

Outline

- Approximate inference: Sampling
- Sampling
  - Direct Sampling methods
    - Forward sampling
    - Rejection sampling
    - Likelihood sampling
  - Markov chain sampling

  - Bayesian Networks with Continuous Variables

Inference in Bayesian Networks

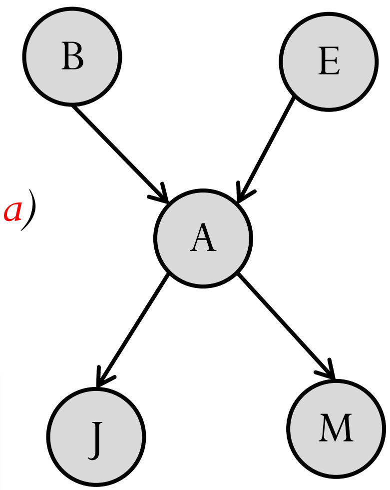


Summary Exact Inference

- Inference – compute posterior probability distribution for a set of query variables given a set of evidence variables that are observed.
- Exact methods :
  - **Inference by enumeration**
  - Simple query: $\mathbf{P}(B \mid j,m) = \alpha\, \mathbf{P}(B,j,m) = \alpha \sum \sum \mathbf{P}(B,j,m,e,a)$

$$P(B \mid j,m) = \alpha \sum_{e} \sum_{a} P(B)\, P(E)\, P(a \mid B,e)\, P(j \mid a)\, P(m \mid a)$$

Worst case time complexity: for n Boolean variable $O(n2^n)$, can be improved by moving the summation outside but still will be $O(2^n)$



  - **Variable elimination**
    - d^k entries computed for a factor over k variables with domain sizes d
    - ordering of elimination of hidden variables does matter- bad elimination order can generate large factors
    - Worst case running time exponential in the size of the Bayes' net (large multiply connected networks)

Approximate Inference

- Sampling based inference

- Basic idea: Generate random samples and compute required probabilities from samples.
  - Draw N samples from distribution
  - Estimate $P(X \mid \boldsymbol{E})$ from samples

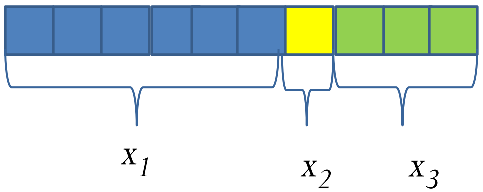
- What do we need to know?
  - How to generate a new sample ?
  - How many samples do we need ?
  - How to estimate P(X|E) ?

Sampling

- Known distribution, single variable
- Consider a random variable X with $dom(X) = \{x_1, x_2, x_3\}$
- To generate a random sample for X
  - Select a random number y in the range [0,1) (we select y from a uniform distribution to ensure that each number between 0 and 1 has same chance of being chosen)
  - Convert this value of y to a sample from given distribution by associating each outcome $\{x_1, x_2, x_3\}$ to a given sub-interval with size proportional to P(X).
  - Example: suppose `random()` returns y = 0.3, 0.25, 0.45, 0.65 …corresponding samples will be $x_1, x_1, x_1, x_2, \ldots$

X	P(X)
x_1	0.6
x_2	0.1
x_3	0.3

$$0 \le y < 0.6 \rightarrow X = x_1$$
$$0.6 \le y < 0.7 \rightarrow X = x_2$$
$$0.7 \le y < 1 \rightarrow X = x_3$$



Sampling Methods

- Forward sampling
- Rejection sampling
- Likelihood weighting
- Gibbs Sampling (MCMC)

Sampling Methods

- Forward sampling / Prior sampling (without evidence)
  - Sample each variable in topological order
  - probability distribution from which the value is sampled is conditioned on the values already assigned to the variable's parents

```
Input: Bayesian network
     X= {X1,…,Xn}, n- #nodes, N - # samples
Output: N samples
Process nodes in topological order – first process
     the ancestors of a node, then the node itself
1.   For t = 1 to N
2.       For i = 1 to n
3.            Xi ← sample xi^t from P(xi|parents (Xi))
```

Forward Sampling

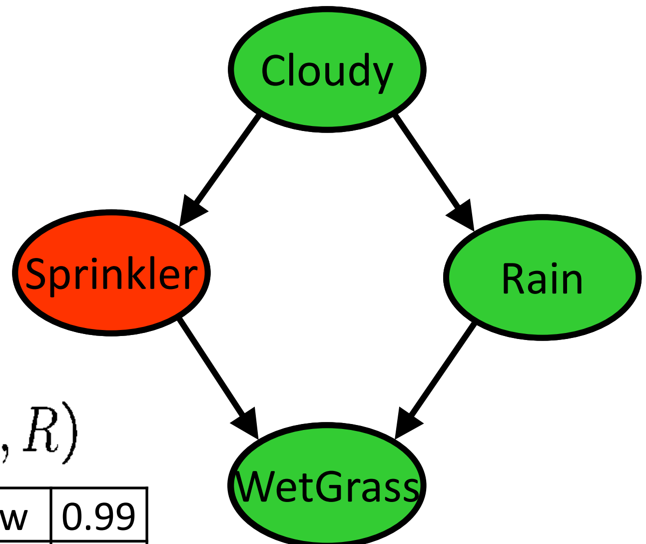
Assume ordering: Cloudy, Sprinkler, Rain, WetGrass

$P(C)$

+c	0.5
-c	0.5

$P(S|C)$

+c	+s	0.1
	-s	0.9
-c	+s	0.5
	-s	0.5



$P(R|C)$

+c	+r	0.8
	-r	0.2
-c	+r	0.2
	-r	0.8

$P(W|S,R)$

+s	+r	+w	0.99
		-w	0.01
	-r	+w	0.90
		-w	0.10
-s	+r	+w	0.90
		-w	0.10
	-r	+w	0.01
		-w	0.99

Samples:

+c, -s, +r, +w

-c, +s, -r, +w

…

Queries:

$\mathrm{P(w)} \approx \#w/\#samples$

$\mathrm{P(w \mid c, s)} \approx \frac{P(w,c,s)}{P(c,s)}$

Sampling methods

- Rejection Sampling (evidence available)
  - generate samples from the prior distribution specified by the network, reject all those samples that do not match the evidence.

```
Input: Bayesian network
    X= {X1,…,Xn}, n- #nodes
    E – evidence, N – # samples
Output: N samples consistent with E
1.  For t=1 to N
2.     For i=1 to n
3.         Xi ← sample xi^t from P(xi | parent(Xi))
4.         If Xi in E and Xi ≠ xi, reject sample:
5.               set i = 1 and go to step 2
```

Rejection Sampling

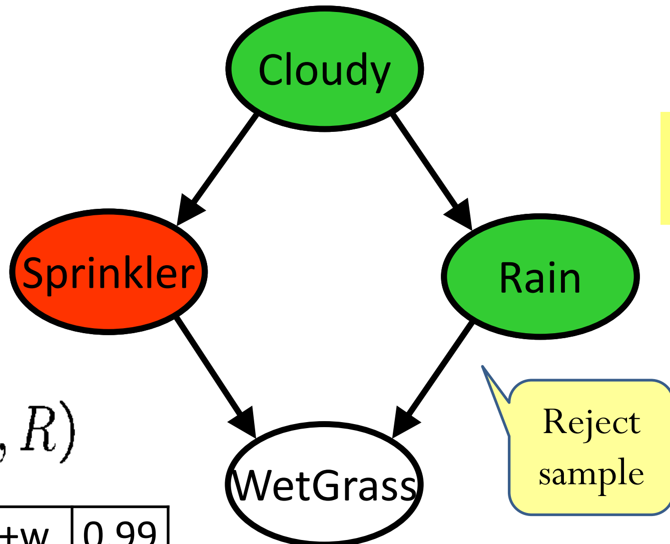
Assume ordering: Cloudy, Sprinkler, Rain, WetGrass

$P(C)$

+c	0.5
-c	0.5

$P(S|C)$

+c	+s	0.1
	-s	0.9
-c	+s	0.5
	-s	0.5



$P(R|C)$

+c	+r	0.8
	-r	0.2
-c	+r	0.2
	-r	0.8

$P(W|S,R)$

+s	+r	+w	0.99
		-w	0.01
	-r	+w	0.90
		-w	0.10
-s	+r	+w	0.90
		-w	0.10
	-r	+w	0.01
		-w	0.99

Samples:

+c, -s, +r, ---Reject

-c, +s, -r, +w

…

Evidence : R $= -r$

// generate sample k

1. Sample C from $P(C)$
2. Sample S from $P(S \mid C)$
3. Sample R from $P(R \mid C)$
4. If $R \neq$ -r, reject sample and start from 1, otherwise
5. Sample W from $P(W \mid S, R)$

Sampling Method

Problem of Rejection Sampling: for unlikely evidence, lots of samples rejected

- Likelihood Weighting
  - fix the values for evidence variables, and sample only non-evidence variable (not sampling from right distribution anymore)
  - Now weight the samples by evidence likelihood (probability of evidence given parents).

For $k = 1$ to N

For each each X_i in topological order $o = (X_1, \ldots, X_n)$:

$w_k = 1$

if $X_i \notin E$

$X_i \leftarrow$ sample x_i from $P(x_i \mid parents(X_i))$

else

assign $X_i = e_i$

$w_k = w_k \bullet P(e_i \mid parents(Xi))$

Likelihood Weighting

Assume ordering: Cloudy, Sprinkler, Rain, WetGrass

Query: P(_| W)

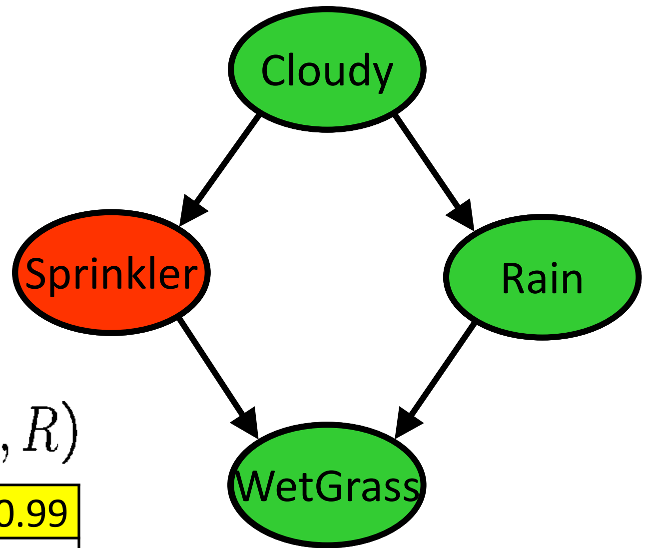
Evidence: W = +w

$P(C)$

+c	0.5
-c	0.5

$P(S|C)$

+c	+s	0.1
	-s	0.9
-c	+s	0.5
	-s	0.5



$P(R|C)$

+c	+r	0.8
	-r	0.2
-c	+r	0.2
	-r	0.8

$P(W|S,R)$

+s	+r	+w	0.99
		-w	0.01
	-r	+w	0.90
		-w	0.10
-s	+r	+w	0.90
		-w	0.10
	-r	+w	0.01
		-w	0.99

Samples:

+c, -s, +r, +w 0.90 (weight)

-c, +s, -r, +w 0.90

…

Sampling Method: MCMC

- Markov Chain Monte Carlo Sampling
- MCMC techniques often applied to solve integration and optimisation problems in large dimensional spaces
- So where do we need this in Bayesian Inference?

(So far considered only discrete state space, yet to see continuous space)

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)} \qquad P(X) = \sum_y P(X,y) \qquad P(X) = \int_y P(X,y)$$

Normalisation

$$P(Y|X) = \textstyle\sum_z P(Y,z|X) \text{ OR } \int_z P(Y,z|X)$$

Marginal Posterior

Expectations

$$E[f(x)] = \int_x f(x)p(x)d(x)$$

Sampling Method: MCMC

- Example Monte Carlo Approximation: Compute the distribution of a function of a random variable, $y = f(x)$.
- Suppose $x \sim \text{Unif}(-1, 1)$ and $y = x^2$. We can approximate $p(y)$ by drawing many samples from $p(x)$, squaring them, and computing the resulting empirical distribution

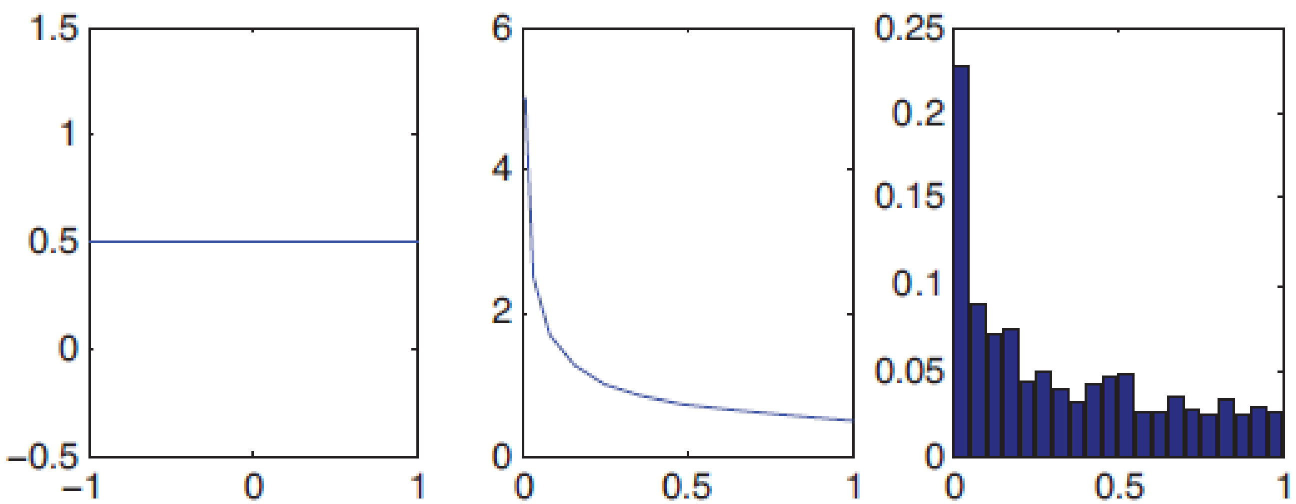


Figure: Computing the distribution of $y = x^2$, where $p(x)$ is uniform (left). The analytic result is shown in the middle, and the Monte Carlo approximation is shown on the right.
(From the book Kevin P. Murphy, ML a probabilistic perspective)

Sampling Method: MCMC

- Example 2: estimating π by Monte Carlo integration
  - Draw a square, then inscribe a circle within it
  - Uniformly scatter a given number of points over the square
  - Count the number of points inside the circle (C) and inside the square (S).
  - The ratio of the inside-count and the total-sample-count is an estimate of the ratio of the two areas.

$$\frac{C}{S} \approx \frac{\pi r^2}{(2r)^2} \qquad \pi \approx \frac{4C}{S}$$

$x^2 + y^2 = r^2$: Equation of a circle

πr^2 Area of a circle

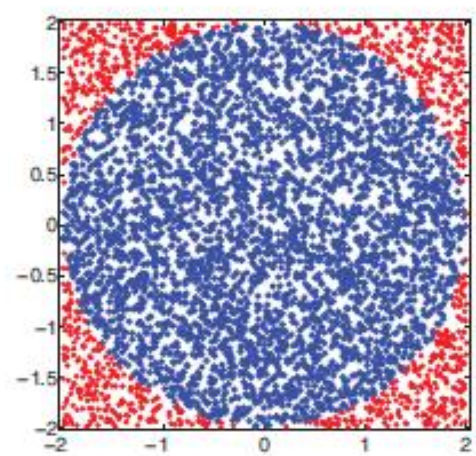


Figure: Estimating π by Monte Carlo integration. Blue points are inside the circle, red crosses are outside.

Introduction to Markov Chains

- A Markov chain is a sequence of random variables $\boldsymbol{X}_0, \boldsymbol{X}_1, \boldsymbol{X}_2, \ldots$ with Markov property that the probability of moving to the next state depends only on the current state of the random variable.

$$\Pr\left(\boldsymbol{X_{t+1}} = \boldsymbol{s_j} \middle| \boldsymbol{X_0} = \boldsymbol{s_k}, \ldots, \boldsymbol{X_t} = \boldsymbol{s_i}\right) = \Pr(\boldsymbol{X_{t+1}} = \boldsymbol{s_j} | \boldsymbol{X_t} = \boldsymbol{s_i})$$

- For simplicity, $\Pr\left(X_{t+1} = s_j \middle| X_0 = s_k, \ldots, X_t = s_i\right) = \Pr(X_{t+1} = s_j | X_t = s_i)$
- A particular chain is defined by its transition probabilities , a transition probability the probability that the chain at state s_i moves to state s_j in a single step and can be given as:

$$P(i \to j) = \Pr(X_{t+1} = s_j | X_t = s_i)$$

- Let $\pi_i(t) = \Pr(X_t = s_i)$ be the probability that the chain is in state i at time t and $\boldsymbol{\pi}(t)$ denote the vector of the state space probabilities at time step t.

$$\pi_i(t+1) = \Pr(\mathrm{X}_{t+1} = \mathrm{s}_i) = \textstyle\sum_k \Pr(X_{t+1} = s_i,\ \ X_t = s_k)$$

$$= \sum_k \Pr(X_{t+1} = s_i, | X_t = s_k) \Pr(X_t = s_k) = \sum_k P(k \to i)\pi_k(t)$$

Introduction to Markov Chains

$$\pi_i(t+1) = \sum_k P(k \to i)\pi_k(t) \quad \ldots\ldots\ldots\ldots\ldots(1)$$

- This can be written in matrix forms if a transition probability matrix **P** is defined whose i, jth element is $P(i \to j)$ also the row sums to 1 i.e $\sum_j P(i \to j) = 1$
- Now, equation (1) becomes $\boldsymbol{\pi}(t+1) = \boldsymbol{\pi}(t)\mathbf{P}$

$$\boldsymbol{\pi}(t) = \boldsymbol{\pi}(t-1)\,\mathbf{P} = (\boldsymbol{\pi}(t-2)\,\mathbf{P})\,\mathbf{P} = \boldsymbol{\pi}(t-2)\mathbf{P}^2 = \ \ldots = \boldsymbol{\pi}(t-t)\mathbf{P}^t$$

$$\boldsymbol{\pi}(t) = \boldsymbol{\pi}(0)\mathbf{P}^t$$

- Let the n-step transition probability be p_{ij}^n i.e. the probability that the chain is at state j given that it was in state i, n time steps ago. It can be determined from the matrix $\mathbf{P}^n$ as it is just the i, jth element of the matrix.

Introduction to Markov Chains

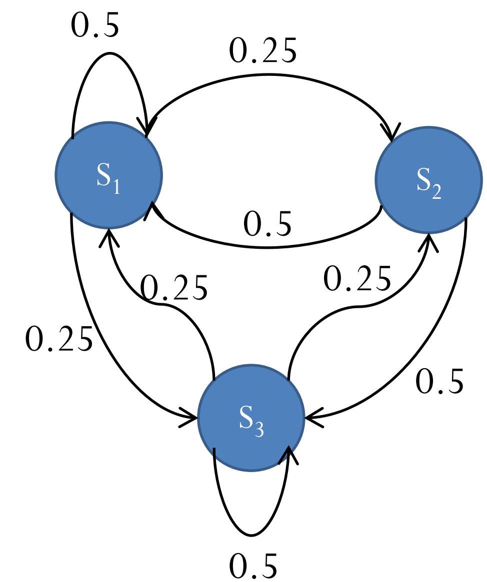
$$\pi_i(t+1) = \sum_k P(k \to i)\pi_k(t)$$

$$\boldsymbol{\pi}(t+1) = \boldsymbol{\pi}(t)\mathbf{P}$$

$$\boldsymbol{\pi}(t) = \boldsymbol{\pi}(0)\mathbf{P}^t$$

$$p_{ij}^n = \Pr\left(X_{t+n} = s_j \,\middle|\, X_t = s_i\right)$$

$= i, jth$ element of the matrix $\mathbf{P}^n$



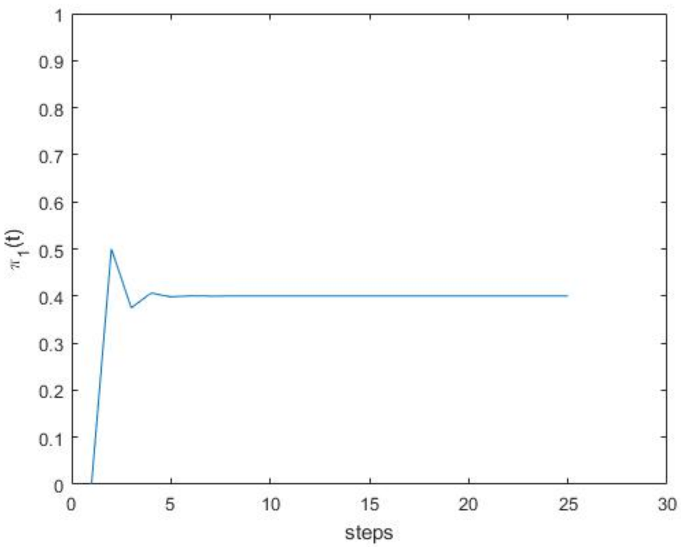
$Let\ \boldsymbol{\pi}(0) = [0\ 1\ 0]$

$\boldsymbol{\pi}(1) = \boldsymbol{\pi}(0)\mathbf{P}$ = [0.5 0 0.5]
$\boldsymbol{\pi}(2) = \boldsymbol{\pi}(1)\mathbf{P}$ = [0.375 0.25 0.375]
….

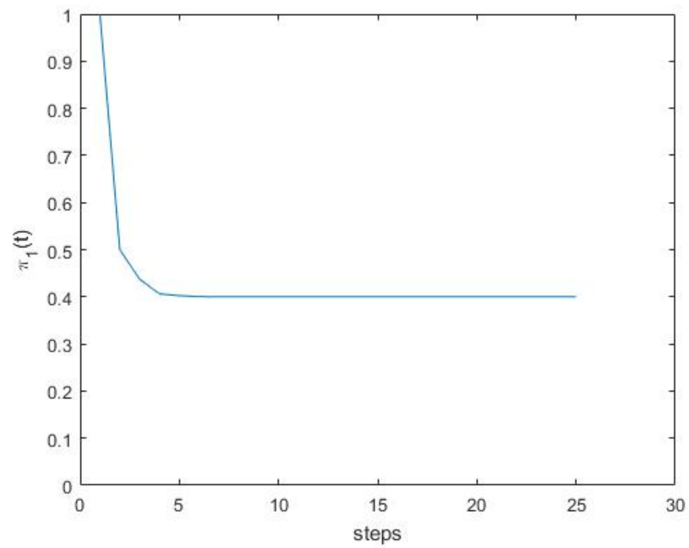
$\boldsymbol{\pi}(7) = \boldsymbol{\pi}(0)\mathbf{P}^7$ = [0.4 0.2 0.4]

$$\mathbf{P} = \begin{array}{cc} \overbrace{}^{X_{t+1}} & \\ \begin{bmatrix} 0.5 & 0.25 & 0.25 \\ 0.5 & 0 & 0.5 \\ 0.25 & 0.25 & 0.5 \end{bmatrix} & \left.\vphantom{\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}}\right\} X_t \end{array}$$

Introduction to Markov Chains



$for\ \boldsymbol{\pi}(0) = [0\ 1\ 0]$



$for\ \boldsymbol{\pi}(0) = [1\ 0\ 0]$

- After a sufficient amount of time, the probability values are independent of actual starting value and we say that the chain has reached a stationary distribution.
- As the process converges, it is expected that
  - $\boldsymbol{\pi}(t) = \boldsymbol{\pi}(t+1) = \boldsymbol{\pi}(t)\mathbf{P}$ i.e. $\boldsymbol{\pi}^* = \boldsymbol{\pi}^*\mathbf{P}$: a distribution satisfying this condition is called stationary distribution.

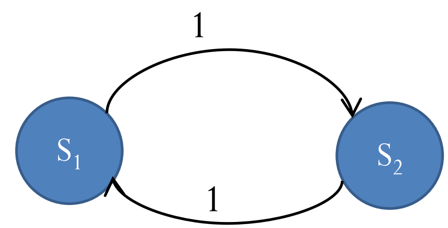
Introduction to Markov Chains

- In general, there is no guarantee that a chain will converge to stationary distribution.
- Example : for the given Markov chain if $\boldsymbol{\pi}(0) = [1\ 0]$ then

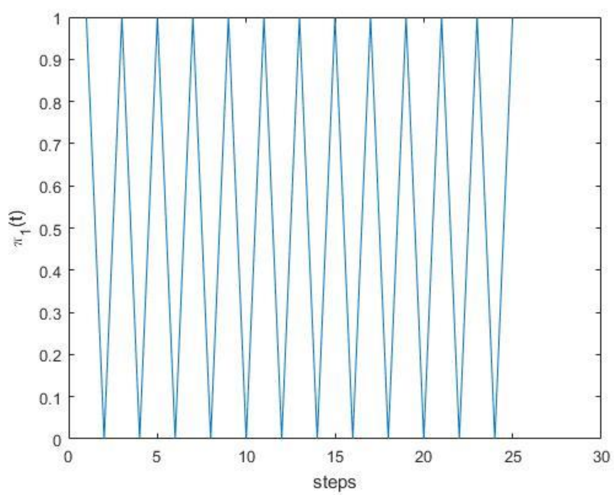
$$\boldsymbol{\pi}(t) = \boldsymbol{\pi}(0)\mathbf{P}^t$$

If t is even then $\mathbf{P}^t = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and $\boldsymbol{\pi}(t) = [1\ 0]$

If t is odd then $\mathbf{P}^t = \mathbf{P}$ and $\boldsymbol{\pi}(t) = [0\ 1]$



$$\mathbf{P} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$



Markov chains like this, which exhibits fixed cyclic behaviour, are called periodic Markov chains.

Introduction to Markov Chains

- For any starting value, the chain will converge to the stationary distribution, as long as the following conditions are satisfied:
  - Aperiodicity: the chain should not get trapped in cycles between certain states.
  - Irreducibility: there exists a positive integer k such that for every i, j the probability of getting from s_i to s_j in k steps is greater than 0 i.e $p_{ij}^{k} > 0$. (all states can be visited from every other state, although it may take more than one step). If this is satisfied then the chain has **unique** stationary distribution.

- A finite-state Markov chain is reversible if there exists a unique distribution $\boldsymbol{\pi}^*$ such that for all $\boldsymbol{i, j}$

$$\pi_i^* P(i \to j) = \pi_j^* P(j \to i)$$: **Detailed balance equation**

Introduction to Markov Chains

If the a Markov chain is irreducible and it satisfies the detailed balance equation relative to $\boldsymbol{\pi}^*$, then $\boldsymbol{\pi}^*$ is the unique stationary distribution.

- MCMC samplers are irreducible and aperiodic Markov chains that have the target distribution as the stationary distribution.
- How to construct such MCMC samplers?
  - One way is to ensure that the detailed balance is satisfied

MCMC Methods

- Markov Chain Monte Carlo (MCMC) methods
  - Unlike direct sampling methods that we discussed, MCMC methods do not generate samples from scratch. Each sample is generated by making a random change to the preceding sample.
  - Say, MCMC algorithm is in a particular current state specifying a value for every variable, it generates a next state by making random changes to the current state.
  - generates samples while exploring the state space of random variables using a Markov chain such that it draws samples from target distribution.
    - cycle mimics the distribution by spending more time in the most important regions (with high probability in the distribution)
- MCMC methods :
    - Metropolis-Hastings algorithm, Gibbs Sampling

The Metropolis-Hastings Algorithm

- Metropolis-Hastings (MH) algorithm involves two distributions
  - proposal distribution : q(.) a simple distribution from where samples can be drawn directly)
  - Target distribution: p($\mathbf{x}$)

Note the change in notations

- Sample from proposal distribution while keeping track of current state $\mathbf{x}^{(t)}$

- the proposal distribution $q(\mathbf{x}|\ \mathbf{x}^{(t)})$ depends on this current state, and so the sequence of samples $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \mathbf{x}^{(3)}, \ldots$.forms a Markov chain.

- Assumption: if we write $p(\mathbf{x}) = \dfrac{\tilde{p}(\mathbf{x})}{Z}$ where Z is the normalization constant, then $\tilde{p}(\mathbf{x})$ can be evaluated for any given value of $\mathbf{x}$, although the value of Z may be unknown or difficult to compute (as it involves high dimension integration or summation).

The Metropolis-Hastings Algorithm

- At each cycle of the algorithm, a candidate sample $\mathbf{x}^{(cand)}$ is generated from the proposal distribution and then accepted according to an appropriate criterion.

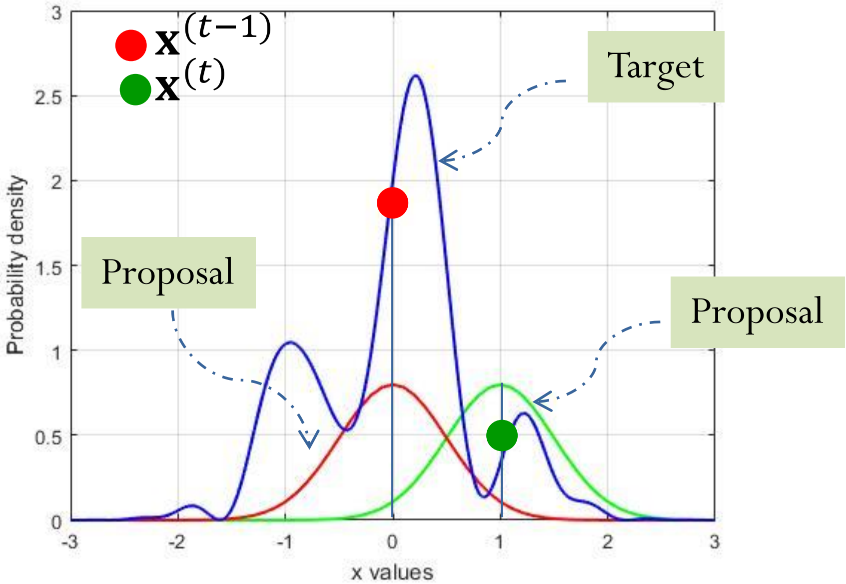
- **MH Algorithm**
- Initialize $\mathbf{x}^{(0)} \sim \mathrm{q}(\mathbf{x})$
- for iteration t = 1,2,3 … do
  - Propose: $\mathbf{x}^{(cand)} \sim q\left(x^{(t)} \middle| x^{(t-1)}\right)$
  - Compute acceptance probability:
    - $\alpha(\mathbf{x}^{(cand)} \mid x^{(t-1)}) = \min\left\{1, \frac{q\left(x^{(t-1)} \middle| x^{cand}\right)\tilde{p}\left(x^{cand}\right)}{q\left(x^{cand} \middle| x^{(t-1)}\right)\tilde{p}\left(x^{(t-1)}\right)}\right\}$
    - $u \sim Uniform\,(0,1)$
    - if $u < \alpha$ then
      - Accept the proposal: $x^{(t)} \leftarrow x^{cand}$
    - else
      - Reject the proposal: $x^{(t)} \leftarrow x^{(t-1)}$
    - end if
- end for

The Metropolis-Hastings Algorithm

$$\alpha(\mathbf{x}^{(cand)} \mid x^{(t-1)}) = \min\left\{1, \frac{q\left(x^{(t-1)} \middle| x^{cand}\right)\tilde{p}\left(x^{cand}\right)}{q\left(x^{cand} \middle| x^{(t-1)}\right)\tilde{p}\left(x^{(t-1)}\right)}\right\}$$

- Expectations from the sampler
  - visit high probability regions in the distribution- this can be achieved by the ratio $\frac{\tilde{p}\left(x^{cand}\right)}{\tilde{p}\left(x^{(t-1)}\right)}$
  - explore the state space and avoid getting stuck in one region – this can be achieved by the ratio $\frac{q\left(x^{(t-1)} \middle| x^{cand}\right)}{q\left(x^{cand} \middle| x^{(t-1)}\right)}$

- Proposal distribution: symmetric or asymmetric distribution
  - If $q(\mathbf{x}^{(t)} \mid \mathbf{x}^{(t-1)}) = q(\mathbf{x}^{(t-1)} \mid \mathbf{x}^{(t)})$ then the distribution is symmetric
  - Example of symmetric distributions: Gaussian distributions or Uniform distribution centred at current state of the chain.



The Metropolis-Hastings Algorithm

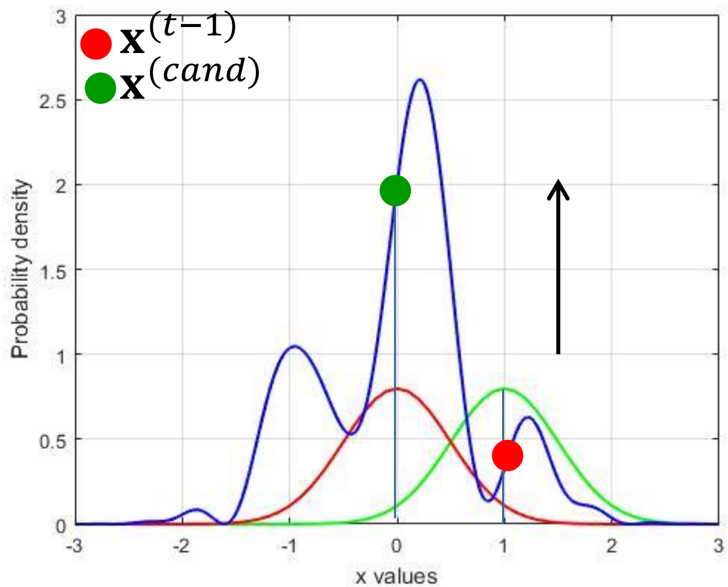


Fig. (a)

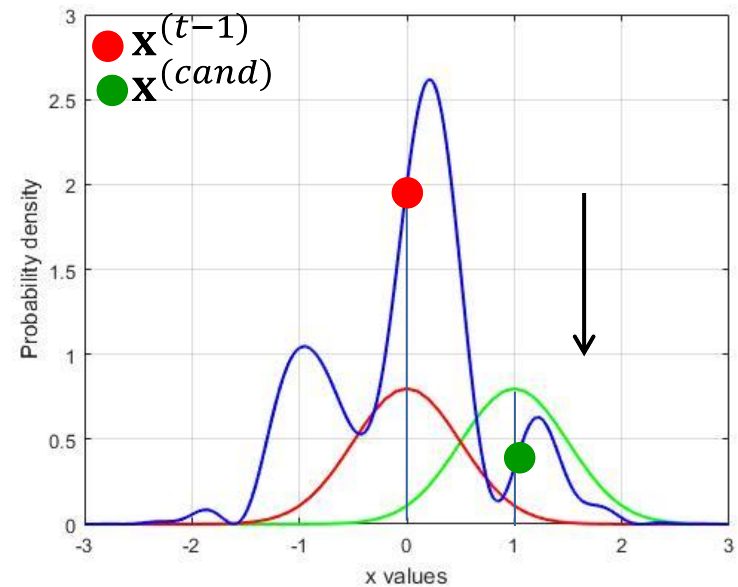


Fig. (b)

For symmetric proposal distribution: $\mathrm{q}(\mathbf{x}^{(cand)} | \mathbf{x}^{(t-1)}) = \mathrm{q}(\mathbf{x}^{(t-1)} | \mathbf{x}^{(cand)})$

$$\frac{q\left(x^{(t-1)} \middle| x^{cand}\right)}{q\left(x^{cand} \middle| x^{(t-1)}\right)} = 1$$

Metropolis Algorithm

for Fig. (a) $\frac{\tilde{p}\left(x^{cand}\right)}{\tilde{p}\left(x^{(t-1)}\right)} = 6.06$: accepted

for Fig. (b) $\frac{\tilde{p}\left(x^{cand}\right)}{\tilde{p}\left(x^{(t-1)}\right)} = 0.17$:accepted with probability 0.17

The Metropolis-Hastings Algorithm

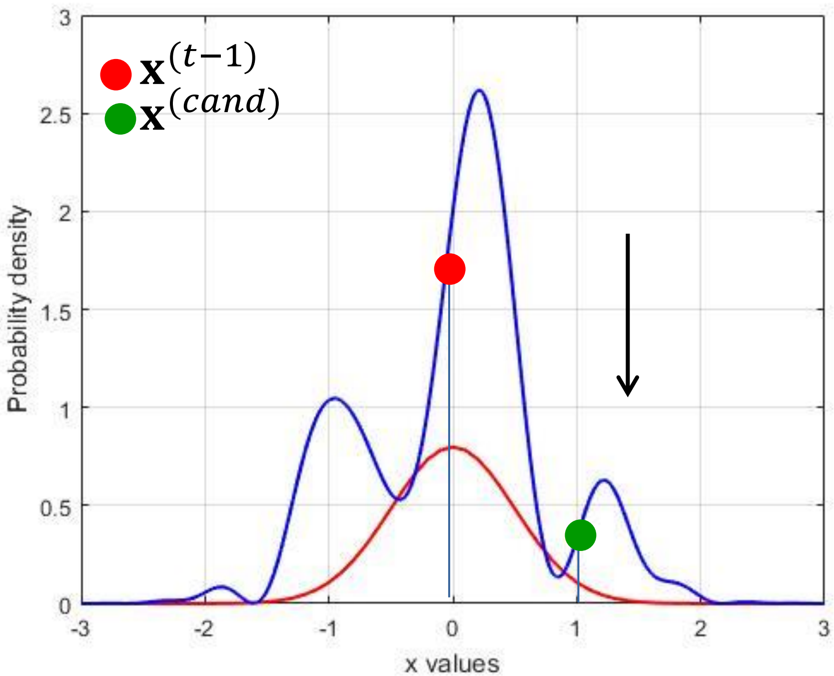


Fig. (c)

In Fig (c) the proposal distribution is asymmetric [fixed at normal(0,0.5)]

$$\frac{\tilde{p}\left(x^{cand}\right)}{\tilde{p}\left(x^{(t-1)}\right)} < 1$$

$$\frac{q\left(x^{(t-1)} \middle| x^{cand}\right)}{q\left(x^{cand} \middle| x^{(t-1)}\right)} > 1$$

So, $\mathbf{x}^{(cand)}$ in this case will be accepted or rejected?

The Metropolis-Hastings Algorithm

- After a sufficient time (say k steps – burn-in period), the chain approaches the stationary distribution and the generated samples $\mathrm{x}^{(k+1)}, \mathrm{x}^{(k+2)}, \ldots$ are from the target distribution.

- We can show that p(x) is the stationary distribution of the Markov chain defined by the Metropolis-Hastings algorithm by showing that the detailed balance is satisfied .

$$\frac{q\left(x^{(t-1)} \middle| x^{cand}\right)}{q\left(x^{cand} \middle| x^{(t-1)}\right)} = \frac{\tilde{p}\left(x^{cand}\right)}{\tilde{p}\left(x^{(t-1)}\right)}$$: **Detailed balance equation**

Gibbs Sampling

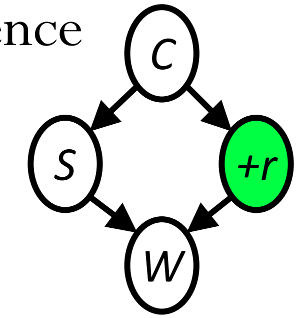
- MCMC algorithm and special case of the Metropolis-Hastings algorithm
- Let $p(x_1, x_2, .. x_n | e_1, \dots, e_m)$ denote the joint distribution of a set of random variables $(x_1, x_2, .. x_n)$ conditioned on a set of evidence variables $(e_1, \dots, e_m)$. A sequence of samples can be generated from such joint probability distribution using Gibbs sampling.
- The method resamples one variable at a time, conditioned on the rest, but keeps the evidence fixed

```
• Intialize {x_i : i = 1:n}
• For t = 1,2,…
    • Pick a variable x_i uniformly at random
    • Sample x_i from p(x_i | x_(-i)^(t-1), e)
    • Let x^t = (x_(-i), x_i)
• End for
```

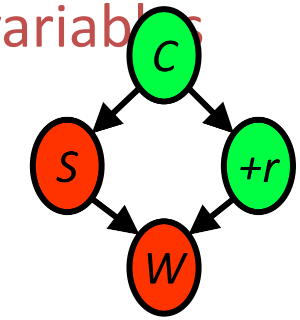
- Intialize $\{x_i : i = 1{:}n\}$
- For t = 1,2,…
  - Pick a variable x_i uniformly at random
  - Sample x_i from $p(x_i | x_{(-i)}^{(t-1)}, \mathbf{e})$
  - Let $x^t = (x_{(-i)}, x_i)$
- End for

Gibbs Sampling Example: P(S | +r)

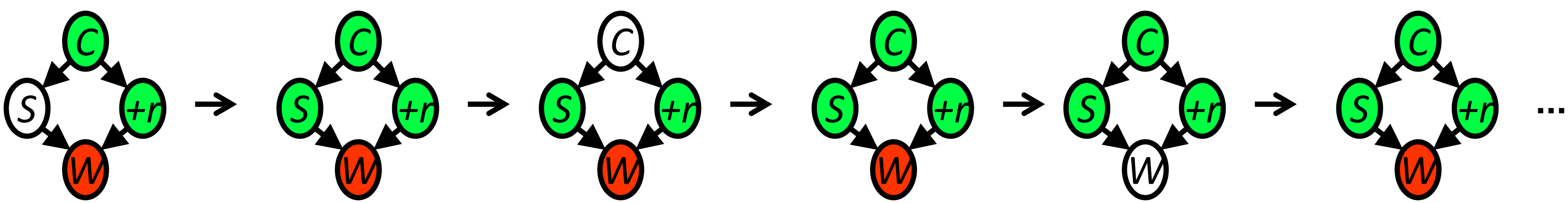
- Step 1: Fix evidence
  - R = +r



- Step 2: Initialize other variables
  - Randomly



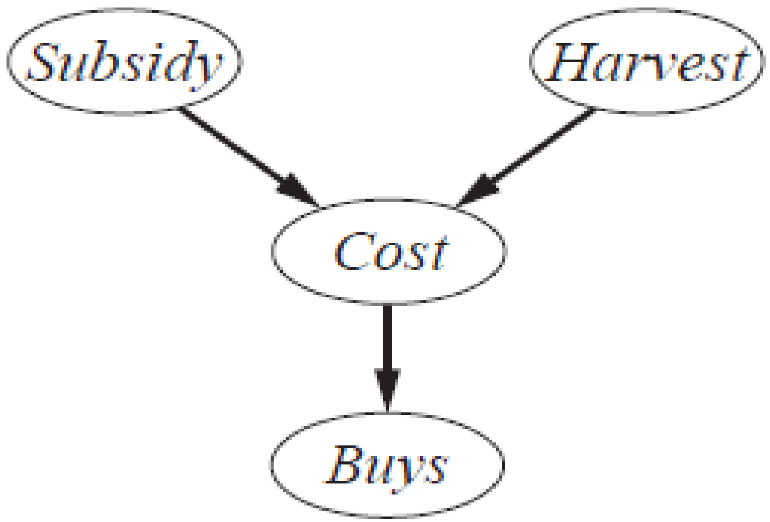
- Steps 3: Repeat
  - Choose a non-evidence variable X
  - Resample X from P(X | all other variables)



Sample from $P(S|+c,-w,+r)$ Sample from $P(C|+s,-w,+r)$ Sample from $P(W|+s,+c,+r)$

This slide is adapted from UC Berkeley CS188 Intro to AI Course Materials

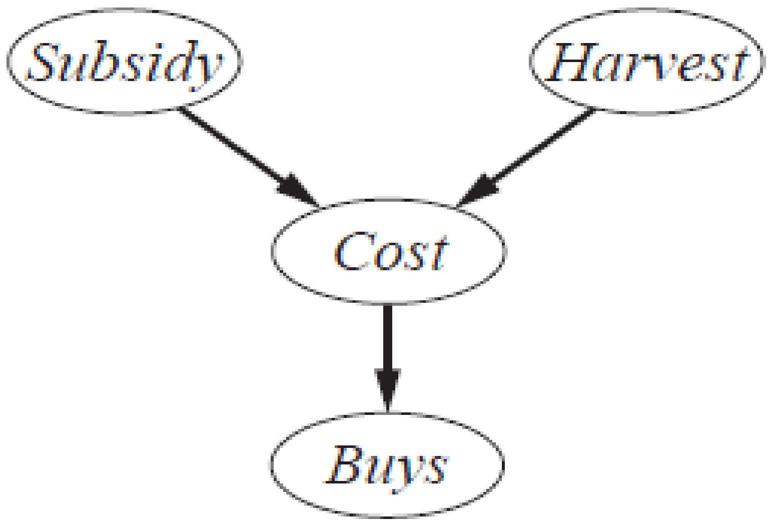
Digression- BN Continuous Variables



- Buys and Subsidy:
  - Discrete variables
- Harvest and Cost:
  - Continuous Variables

- Probability tables for Continuous Variables:
  - Use discretization
  - Define standard families of PDF specified by finite number of parameters
    - Example: Gaussian (or normal distribution) $N(\mu, \sigma^2)(x)$

Digression- BN Continuous Variables



- Buys and Subsidy:
  - Discrete variables
- Harvest and Cost:
  - Continuous Variables

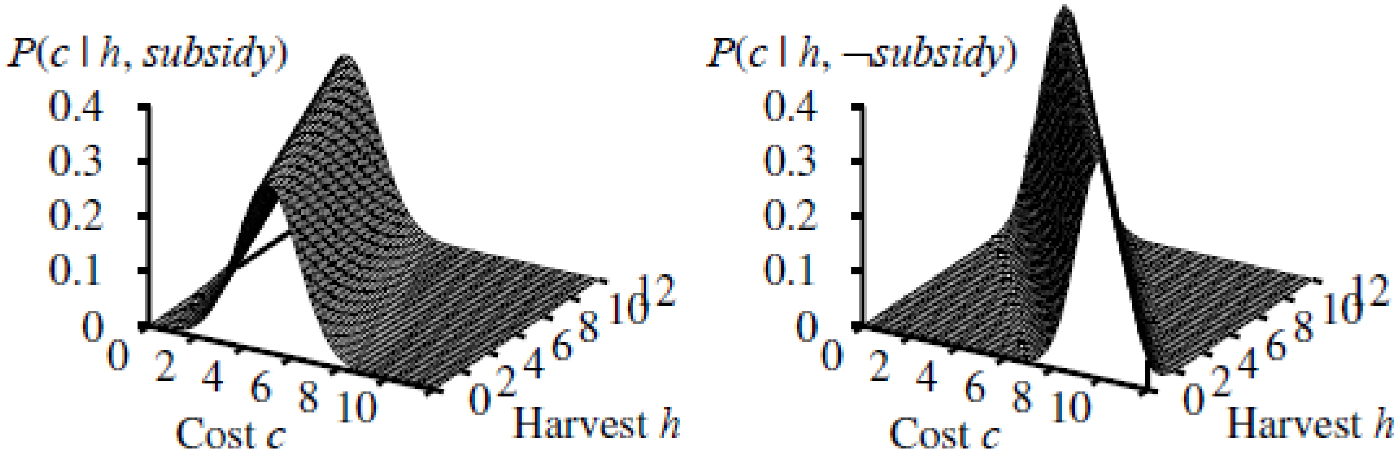
- CPT continuous variable and discrete/continuous parent

  $\mathbf{P}(Cost \mid Harvest, Subsidy)$

  $P(Cost \mid Harvest, subsidy) \qquad P(Cost \mid Harvest, \neg subsidy).$

- We will use linear Gaussian distribution to specify how the distribution over c (Cost) depends on h (Harvest).

Digression- BN Continuous Variables

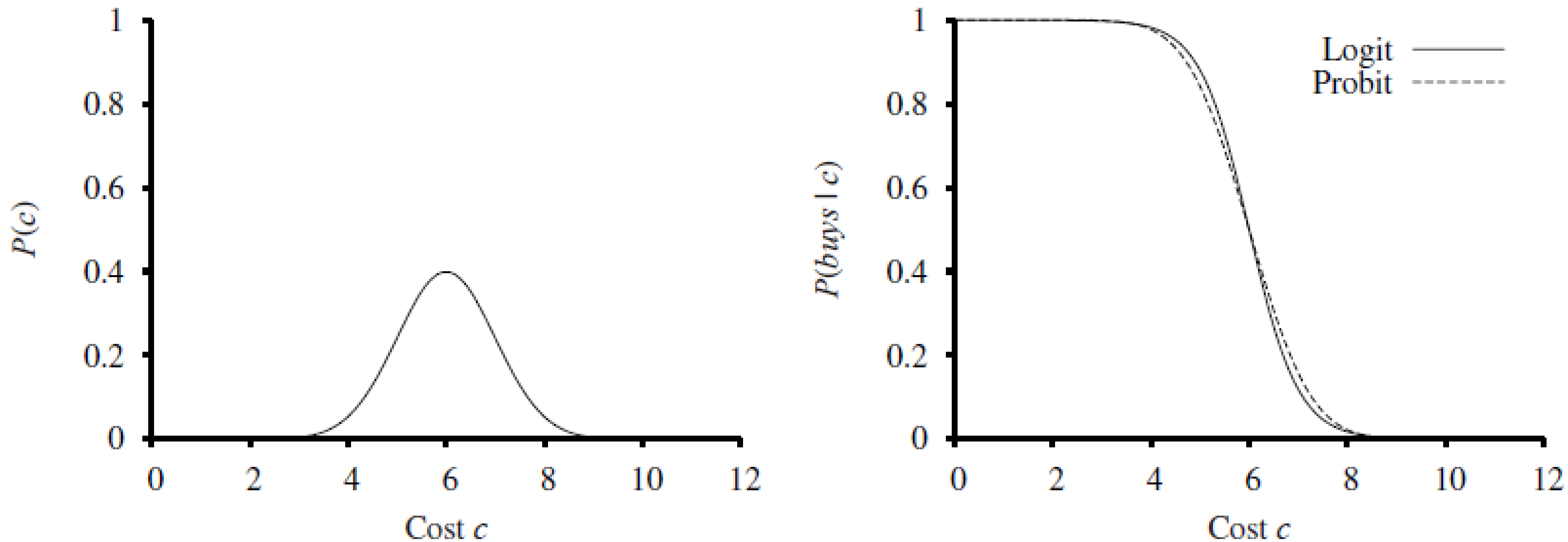


(a) (b)

$$P(c \mid h, subsidy) \;=\; N(a_t h + b_t, \sigma_t^2)(c) = \frac{1}{\sigma_t \sqrt{2\pi}} \, e^{-\frac{1}{2}\left(\frac{c-(a_t h + b_t)}{\sigma_t}\right)^2}$$

$$P(c \mid h, \neg subsidy) \;=\; N(a_f h + b_f, \sigma_f^2)(c) = \frac{1}{\sigma_f \sqrt{2\pi}} \, e^{-\frac{1}{2}\left(\frac{c-(a_f h + b_f)}{\sigma_f}\right)^2} .$$

Digression- BN Continuous Variables



(a) (b)

logistic function $1/(1+e^{-x})$ $\quad P(buys \mid Cost = c) = \dfrac{1}{1 + exp(-2\frac{-c+\mu}{\sigma})}$.

Summary

- In this lecture we discussed
  - Approximate inference using sampling
  - What is Monte Carlo Approximation?
  - What is Markov Chain?
  - What is Markov Chain Monte Carlo Approximation?
  - Working of Metropolis-Hastings Algorithm
  - Working of Gibbs Sampling