EE3731C Statistical Signal 3.5

BT Thomas Yeo

ECE, CIRC, Sinapse, Duke-NUS, HMS

Recap (1)

- Discrete time Markov process

$$p(x_1, \cdots, x_{n-1}, x_{n+1}, \cdots, x_N | x_n) = p\underbrace{(x_1, \cdots, x_{n-1}|x_n)}_{\text{PAST}} p\underbrace{(x_{n+1}, \cdots, x_N|x_n)}_{\text{FUTURE}}$$

- Discrete time, discrete state Markov process: $\pi_{n+1} = \pi_n T$, where $\pi_n =$ probability of different states at time n

$$\pi_{n+1} = [\pi_n(1)\ \pi_n(2)\ \pi_n(3)] \times \begin{bmatrix} 0.9 & 0.075 & 0.025 \\ 0.15 & 0.8 & 0.05 \\ 0.25 & 0.25 & 0.5 \end{bmatrix}$$

Recap (2)

- From previous slide,

$$\begin{aligned}\pi_{n+1} &= \pi_n T \\ \implies \pi_{n+2} &= \pi_{n+1} T = \pi_n T^2 \\ \implies \pi_{n+k} &= \pi_n T^k\end{aligned}$$

- If $\pi^* = \pi^* T$, then π^* is stationary distribution of T.
  - Because $\pi_{n_0} = \pi^* \implies$ probability of any state constant (stationary) for $n \geq n_0$
  - π^* is the left eigenvector of T with eigenvalue 1
- Fundamental Theorem of Markov Chains
  - If there is n_0, such that $T^n(i,j) > 0$ for all i, j and $n > n_0$, then markov chain has unique stationary distribution π^*.
  - For any π_1, as $n \to \infty$, $\pi_1 T^n \to \pi^*$

Recap (2)

- From previous slide,

$$\pi_{n+1} = \pi_n T$$
$$\implies \pi_{n+2} = \pi_{n+1} T = \pi_n T^2$$
$$\implies \pi_{n+k} = \pi_n T^k$$

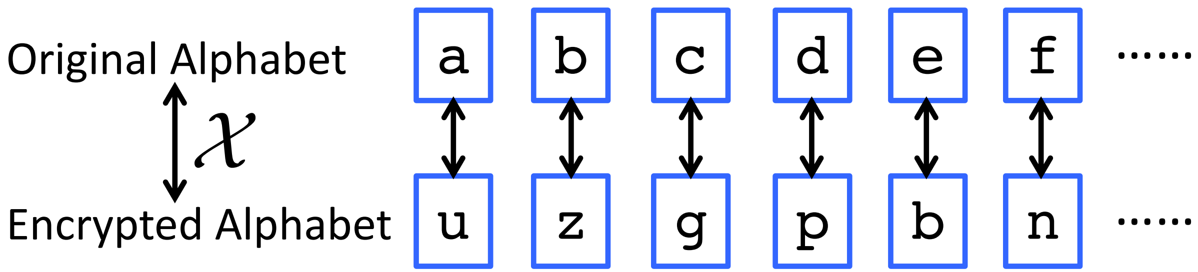
- If $\pi^* = \pi^* T$, then π^* is <u>stationary</u> distribution of T.
  - Because $\pi_{n_0} = \pi^* \implies$ probability of any state constant (stationary) for $n \geq n_0$
  - π^* is the left eigenvector of T with eigenvalue 1
- Fundamental Theorem of Markov Chains (in plain English)
  - Suppose we start at an arbitrary state i at time 1. If there is non-zero probability of being in any state any time after finite time n_0, then Markov chain has unique stationary distribution.
  - Any initial state will reach stationary distribution given sufficient time

Markov Chain Monte Carlo (MCMC)

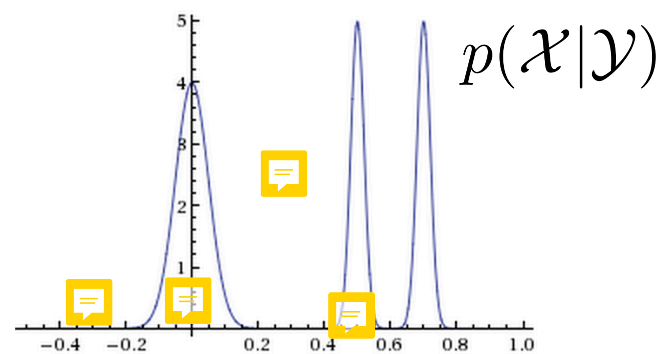
Markov Chain Monte Carlo (MCMC)

- Why stationary distribution? MCMC are methods for sampling from $p(x)$ by constructing markov chain whose stationary distribution $\pi^*(x) = p(x)$

- Why sample? Given observations $\mathcal{Y}$, we often want samples from conditional distribution $p(\mathcal{X}|\mathcal{Y})$
  - e.g., in programming assignment, $\mathcal{Y}$ is encrypted text, $\mathcal{X}$ is mapping between the original and encrypted alphabets.



  - If posterior $p(\mathcal{X}|\mathcal{Y})$ is very "peaky", then samples of $\mathcal{X}$ likely close to peaks (high probability) and good decryption candidates
  - MAP $\approx$ sample with largest posterior
  - MMSE $\approx$ average of all samples

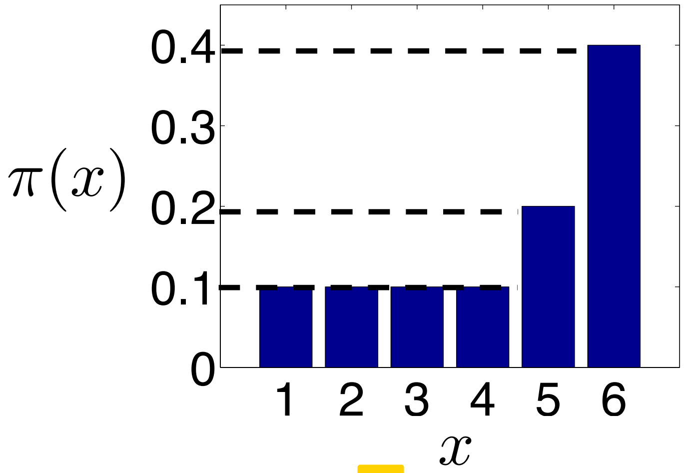


Sampling $\pi(\mathcal{X})$ with Metropolis Algorithm

1. Start with any $x = x_0$
2. Sample new x' using any proposal distribution $q(x'|x)$
   - Only constraint is $q(x'|x) = q(x|x')$
3. $$\begin{cases} \text{if } \pi(x') \geq \pi(x) & \text{Replace } x \text{ with } x' \\ \text{if } \pi(x') < \pi(x) & \text{Replace } x \text{ with } x' \text{ with probability } \pi(x')/\pi(x) \\ & \text{Keep } x \text{ with probability } 1 - \pi(x')/\pi(x) \end{cases}$$
4. Repeat steps 2 and 3 for a "long" time before stopping
   - Current $x \sim \pi(\mathcal{X})$
5. Will explain why this works later. First see example.

Biased Dice Metropolis Example

- Biased dice



- Metropolis:

1. Let $x = 1$
2. Generate x' with $q(x'|x) = 1/6$ for $x' = 1, 2, 3, 4, 5, 6$
3. $$\begin{cases} \text{if } \pi(x') \geq \pi(x) & \text{Replace } x \text{ with } x' \\ \text{if } \pi(x') < \pi(x) & \text{Replace } x \text{ with } x' \text{ with probability } \pi(x')/\pi(x) \\ & \text{Keep } x \text{ with probability } 1 - \pi(x')/\pi(x) \end{cases}$$
4. Repeat steps 2 and 3 ten times and return current value of x as one sample from $\pi(x)$

10 iterations of Biased Dice Example

Iter 1: Current x = 1, new x’ = 5

pi(x’) >= pi (x): definitely accept

Iter 2: Current x = 5, new x’ = 3

pi(x’) < pi (x): accept with probability pi(x’)/pi(x)

Coin toss with p = pi(x’)/pi(x) = 0.5; head; accept x’

⋮

Iter 9: Current x = 6, new x’ = 6

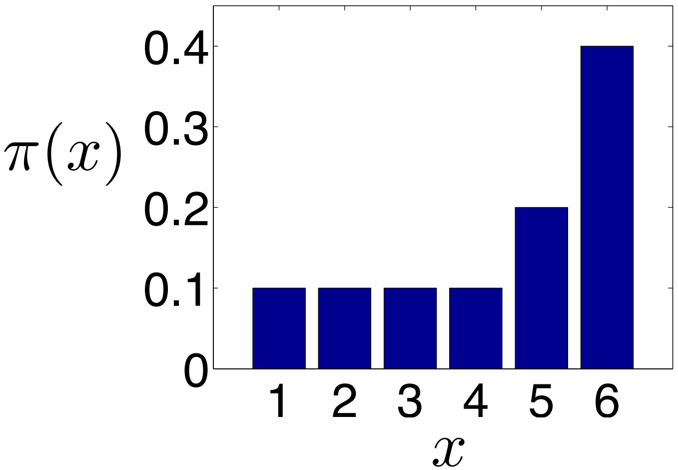
pi(x’) >= pi(x): definitely accept

Iter 10: Current x = 6, new x’ = 5

pi(x’) < pi(x): accept with probability pi(x’)/pi(x)

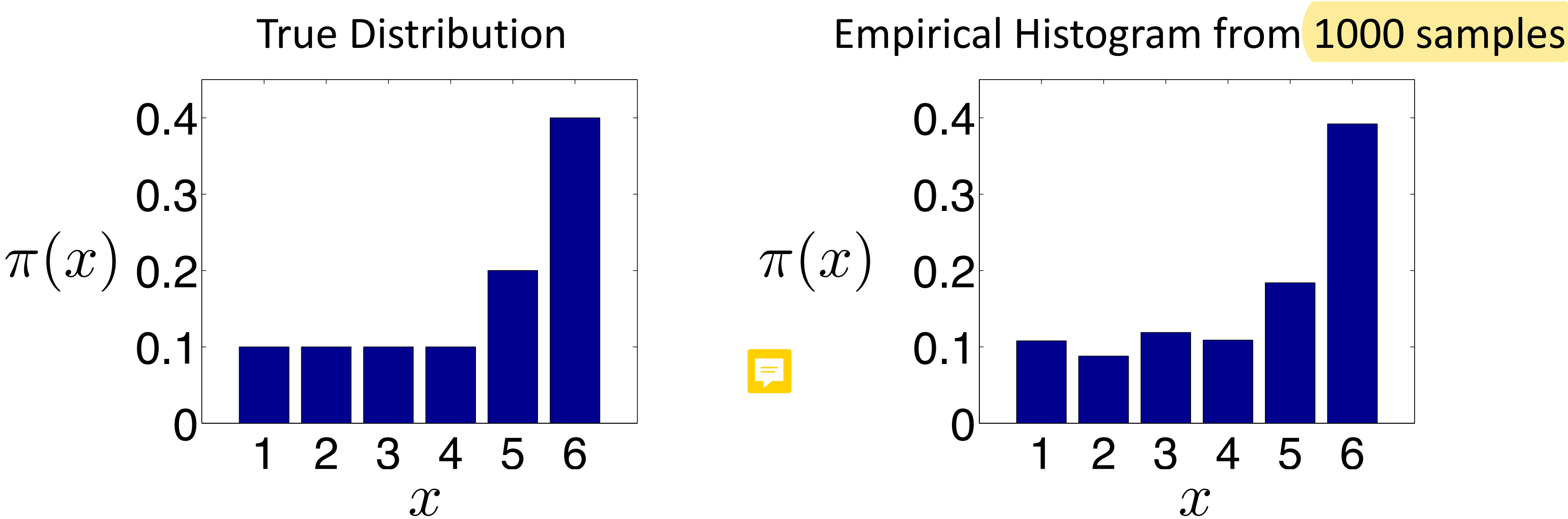
Coin toss with p = pi(x’)/pi(x) = 0.5; tail; reject x’

1st sample of p(x) is 6



See MetropolisBiasedDice.m on IVLE

Repeat 1000 times to get 1000 samples



- What can we use samples from $\pi(x)$ for?
  - Can estimate $E(x)$ by averaging samples. In dice example, empirical mean = 4.349 (true mean $E(x) = 4.4$)
  - In real applications, we often sample from $p(x|y)$. Suppose $\pi(x) = p(x|y)$, then if we average samples, we get approximation of $E(x|y)$, i.e., MMSE estimate of x given y

See MetropolisBiasedDice.m on IVLE

Why do we want to sample from p(x | y)?

- In example, I used Metropolis algorithm to generate 1000 samples from biased dice $\pi(x)$, where $\pi(1) = \pi(2) = \pi(3) = \pi(4) = 0.1$, $\pi(5) = 0.2$ and $\pi(6) = 0.4$.

- Since we already know $\pi(x)$, why sample from $\pi(x)$?

- More specifically, suppose $p(x|y) = \pi(x)$ from the dice example. We can compute the MAP estimate of x given y easily. For example, $x_{MAP} = 6$ because $\pi(6)$ has the highest value. Why bother to generate 1000 samples from $\pi(x)$?

- The reason is computation of MAP and MMSE estimates is difficult when x is multi-dimensional.

Why do we want to sample from p(x | y)?

- Let $\vec{x} = \{x_1, \cdots, x_N\}$. Let's assume $x_n \in \{1, \cdots, K\}$
- Given observation y: $\vec{x}_{MAP} = \text{argmax}_{\vec{x}}\, p(\vec{x}|y)$
- To find MAP estimate, we can iterate through every possible value of $\vec{x}$ and keep track of the value of $\vec{x}$ with the highest posterior probability.
- However, there are K^N possible values of $\vec{x}$ $\implies$ need to compare K^N possible values of $p(\vec{x}|y)$ to find the best $\vec{x}$.
- In programming assignment, $\vec{x}$ is mapping between original alphabets and encrypted alphabets. Therefore, $N = 27$ and $K = 27$ $\implies$ "straightforward" approach of computing $\vec{x}_{MAP}$ requires us to compare 27^{27} instances of $p(\vec{x}|y)$.

Why do we want to sample from p(x | y)?

- Instead, we can use Metropolis algorithm to generate M samples from $p(\vec{x}|y)$.
- Denote these samples as $\vec{x}^{(1)}, \cdots, \vec{x}^{(M)}$. Then

$$\vec{x}_{MAP} \approx \underset{\vec{x}^{(m)}}{\operatorname{argmax}} \, p(\vec{x}^{(m)}|y),$$

  which only requires us to compare M instances of $p(\vec{x}|y)$.
- In programming assignment, we run Metropolis algorithm for 15000 iterations to generate 1 sample. As will be seen in assignment, this 1 sample is good enough to unscramble the encrypted message. We do not need to generate multiple samples to approximate the MAP estimate (even though we could).

Why does Metropolis work?

Why does Metropolis Algorithm Work?

1. Start with any $x = x_0$
2. Sample new x' using <u>any</u> proposal distribution $q(x'|x)$
   - Only constraint is $q(x'|x) = q(x|x')$
3.

$$\begin{cases} \text{if } \pi(x') \geq \pi(x) & \text{Replace } x \text{ with } x' \\ \text{if } \pi(x') < \pi(x) & \text{Replace } x \text{ with } x' \text{ with probability } \pi(x')/\pi(x) \\ & \text{Keep } x \text{ with probability } 1 - \pi(x')/\pi(x) \end{cases}$$

4. Repeat steps 2 and 3 for a "long" time before stopping
   - Current $x \sim \pi(\mathcal{X})$

Why does Metropolis Algorithm Work?

- Running Metropolis equivalent to following Markov Chain:
  - Initial state: $x_1 = x_0$
  - Transition probability:

    For $x \neq x'$

$$T(x,x') = \begin{cases} q(x'|x) & \text{if } \pi(x') \geq \pi(x) \\ q(x'|x)\pi(x')/\pi(x) & \text{if } \pi(x') < \pi(x) \end{cases}$$

    and

$$T(x,x) = C_x \text{ to ensure } \sum_z T(x,z) = 1$$

- If T satisfies Fundamental Theorem of Markov Chain (which is quite easy), then Markov chain has stationary distribution $\pi(\mathcal{X})$

Why Stationary Distribution is $\pi(\mathcal{X})$ (1)

- First, show $\pi(x)T(x,x') = \pi(x')T(x',x)$
- Proof is at end of slides
- Property is known as "detailed balance". Markov chain is considered "reversible"

Why Stationary Distribution is $\pi(\mathcal{X})$ (2)

- From previous slide $\pi(x)T(x,x') = \pi(x')T(x',x)$
- Let's verify $\pi(\mathcal{X})T = \pi(\mathcal{X})$. To evaluate k-th element of $\pi(\mathcal{X})T$, consider

$T(:,k) = k$-th column of T

$$\begin{aligned}
\pi(\mathcal{X})T(:,k) \\
&= \sum_{x\in\mathcal{X}} \pi(x)T(x,k) \\
&= \sum_{x\in\mathcal{X}} \pi(k)T(k,x) \\
&= \pi(k)\sum_{x\in\mathcal{X}} T(k,x) \\
&= \pi(k)
\end{aligned}$$

Definition of vector multiplication

$\pi(x)T(x,k) = \pi(k)T(k,x)$

$\pi(k)$ does not depend on x

$\sum_{x\in\mathcal{X}} T(k,x) = 1$

- k-th element of $\pi(\mathcal{X})T$ equals $\pi(k)$, so $\pi(\mathcal{X})$ is stationary distribution

Specifying Proposal Distribution $q(x' \mid x)$

Specifying proposal distribution q(x’ | x)

- Fundamental Theorem easy to satisfy
  - Exists n_0, such that $T^n(i,j) > 0$ for all i, j and $n > n_0$
  - Dice example: $q(x'|x) \neq 0$ for all $x' \implies T(x,x') \neq 0$ for all $x' \implies n_0 = 1$
- To sample $\pi(\mathcal{X}) = \frac{1}{Z} f(\mathcal{X})$, observe Z cancels out in the Metropolis algorithm, so replace π with f
  - Useful when Z difficult to compute (e.g., programming assignment)
- Can just specify procedure for generating random x' from x. No need to specify $q(x'|x)$
- For continuous $p(x)$, replace discrete states with continuous states

Summary

- Monte Carlo Markov Chain
  - Metropolis algorithm
  - Detailed balance: $\pi(x)T(x,x') = \pi(x')T(x',x)$
- Probabilistic signal detection
  - Rather than single estimate (ML, MAP, MMSE), sample posterior distribution $p(x|y)$
  - Average samples $\approx$ MMSE
  - Sample with biggest probability $\approx$ MAP

Further Optional Readings

- IVLE: Persi Diaconis, The Markov Chain Monte Carlo Revolution (MCMCRev.pdf)

- Another example of Metropolis algorithm: http://www.youtube.com/watch?v=Dzx5xNT79TI

- Search for terms on Wikipedia like “Markov Chain”, “Metropolis-Hastings”, “MCMC”

Additional Material

Why Stationary Distribution is $\pi(\mathcal{X})$ (1)

- First, show $\pi(x)T(x,x') = \pi(x')T(x',x)$
- If $x' = x$, obviously true
- If $x' \neq x$, two cases:
- If $\pi(x') \geq \pi(x)$

$$
\begin{aligned}
&\pi(x)T(x,x') && T(x,x') \text{ definition} \\
&= \pi(x)q(x'|x) && q(x'|x) = q(x|x') \\
&= \pi(x)q(x|x') && \text{Multiply } \frac{\pi(x')}{\pi(x')} \\
&= \pi(x')\frac{\pi(x)}{\pi(x')}q(x|x') && T(x',x) \text{ definition} \\
&= \pi(x')T(x',x)
\end{aligned}
$$

- If $\pi(x') < \pi(x)$

$$
\begin{aligned}
&\pi(x)T(x,x') && T(x,x') \text{ definition} \\
&= \pi(x)q(x'|x)\frac{\pi(x')}{\pi(x)} \\
&= q(x|x')\pi(x') && q(x'|x) = q(x|x') \\
&= T(x',x)\pi(x') && T(x',x) \text{ definition}
\end{aligned}
$$

- Property known as "detailed balance". Markov chain is considered "reversible"