Introduction to Computational Biology
Lecture # 14: MCMC - Markov Chain Monte Carlo

Assaf Weiner

Tuesday , March 13, 2007

1 Introduction

Today we will return to the motif finding problem, in lecture 10 we have built an HMM model that describes a sequence with one or many occurrence of the motif, and estimated the model's parameters using the EM algorithm. Today we will show a different approach for solving the motif finding problem using Markov chain Monte Carlo (MCMC) method.

We have a probability distribution $P(x)$, we would like to compute the expectation of a function on a data set $E_p[f(X)]$. Suppose that we can generate i.i.d samples $x_1, ..., x_n$ from $P(x)$ then

$$E_p[f(X)] \approx \frac{1}{n}\sum_{i=1}^{n} f(x_i) \tag{1}$$

is a MC estimator of $E_p[f(x)]$, the next question is how to draw i.i.d samples from some $P(x)$?

2 Markov Chains

Before introducing the Metropolis-Hastings algorithm and the Gibbs sampler, a few introductory comments on Markov chains.

Markov Process: Let $X^{(t)}$ denote the value of a random variable at time t, and let the **state space** refer to the range of possible X values. The random variable is a ***Markov process*** if the transition probabilities between different values in the state space depend only on the random variables current state, i.e.,

$$Pr(X^{(t+1)} = s_{t+1}|X^{(t)} = s_t, ..., X^{(1)} = s_1) = Pr(X^{(t+1)} = s_{t+1}|X^{(t)} = s_t) \tag{2}$$

Markov Chain: A *Markov chain* is a sequence of random variables $X^{(1)}, ...X^{(n)}$ with the Markov property (each state depends only on the previous state). A particular chain is defined by its ***transition probabilities***, $Pr(i, j) = Pr(i \to j)$, which is the probability that a process at state space s_i moves to state s_j in a single step,

$$P(i, j) = P(i \to j) = Pr(X^{(t+1)} = s_j|X^{(t)} = s_i) \tag{3}$$

Homogeneous Markov Chain: A markov chain is called *homogeneous* (Markov chains with homogeneous transition probabilities) if transition from one state to another is not time-dependent, Formally

$$Pr(X^{(t+1)}|X^{(t)}) = Pr(X^{(t'+1)}|X^{(t')}) \tag{4}$$

for all t, t'.

irreducible A Markov chain is said to be *irreducible* if it is possible to get to any state from any state with a positive probability

Acyclic process A state is called *periodic* with period k if any return to state i must occur in some multiple of k steps and k is the largest number with this property

$$\exists k \quad Pr(X^{(n+d)} = s | X^{(n)} = s) > 0 \iff d \bmod k = 0$$

A process is called *acyclic* if for every state X, X is not periodic. Put another way, the chain is not forced into some cycle of fixed length between certain states.

Definition 2.1 *Ergodic Markov chain - A Markov chain will be called* ergodic *if it is homogeneous, acyclic and irreducible.*

3 Sampling using Markov Chain

Our problem is to obtain samples from some complex probability distribution $p(x)$. In order to achieve that we will build a Markov Chain with some transition probability $Q(X^{(t)}|X^{(t+1)})$ that when $t \to \infty$ converge to some *stationary distribution*

$$Q(X^{(t)}|X^{(0)}) \longrightarrow^{t\to\infty} Q^{\infty}(X^{(t)}) \quad s.t. \quad P(X = x) = Q^{\infty}(X^{(t)} = x) \tag{5}$$

In order to use markov chain for sampling we want to reach a steady state that will represent our probability distribution, we'll denote this probability as $Q^{\infty}(X = x) = P(X = x)$ (stationary distribution).

Stationary Distribution: A distribution on the states such that the distribution at time n + 1 is the same as the distribution at the time n is called a *stationary distribution*. The conditions for a stationary distribution is that the chain is ***irreducible*** and ***aperiodic*** (*ergodic*). When a chain is periodic, it can cycle in a deterministic fashion between states and hence never settles down to a stationary distribution.

A sufficient condition for a unique stationary distribution is that the **detailed balance** equation holds, for big enough t the probability to reach X_t and transit to X_{t+1} is:

$$P(X^{(t)}, X^{(t+1)}) = Q^{\infty}(X_t)Q(X^{(t+1)}|X^{(t)}) \tag{6}$$

A Markov process is said to show ***detailed balance*** if the transition rates between each pair of states a and b in the state space obey

$$Q(X^{(t)} = a|X^{(t+1)} = b) \cdot Q^{\infty}(X_t = a) = Q(X^{(t)} = b|X^{(t+1)} = a) \cdot Q^{\infty}(X_t = b) \tag{7}$$

Reversible Markov chains: A *reversible Markov chain* is a process in which you can generate the same trajectory whether you walk forward or backward in the process. Another way of phrasing it is a process in which we can't tell the order between states, if there are two given states we can't tell which one came before the other (diffusion in a solution is an example for such a reversible process). In reversible Markov chains:

$$P(X = a)Q(X^{(t+1)} = b|X^{(t)} = a) = P(X = b)Q(X^{(t+1)} = a|X^{(t)} = b) \tag{8}$$

so the ratio between the stationary probability of two states is:

$$\frac{P(X = a)}{P(X = b)} = \frac{Q(X^{(t+1)} = a|X^{(t)} = b)}{Q(X^{(t+1)} = b|X^{(t)} = a)} \tag{9}$$

if this holds for every a and b then the *stationary distribution* is the only distribution the holds this condition, and we have,

$$P(X = x) = Q^{\infty}(X^{(t)} = x) \quad \Longrightarrow P = Q^{\infty} \tag{10}$$

We will introduce two methods to construct a Markov Chain (defining Q) that has the desired distribution as its stationary distribution.

4 Metropolis-Hastings Algorithm

Our goal is to draw samples from some distribution $p(x)$ where $p(x) = f(x)/K.$ the normalizing constant K may not be known, and very difficult to compute. The Metropolis-Hastings algorithm (Metropolis and Ulam 1949, Metropolis et al. 1953, Hastings 1970) generates a sequence of draws from this distribution is as follows:

4.1 The Algorithm

1. Specify an initial value $\theta^{(0)}$ satisfying $f(\theta^{(0)}) > 0$.
2. Repeat for $t = 1, 2, ..., M$
   (a) Using current $\theta^{(t)}$ value, sample a candidate point θ' from some proposal distribution $r(\theta'|\theta^{(t)})$. This distribution is also referred to as the jumping or candidate-generating distribution. The only restriction on the proposal density in the Metropolis algorithm is that it is symmetric, i.e., $r(a|b) = r(b|a)$.
   (b) Sample $q \sim U(0, 1)$.
   (c) Let $\theta^{(t+1)} = \begin{cases} \theta', & \text{if } \frac{P(\theta^{(t)})r(\theta'|\theta^{(t)})}{P(\theta')r(\theta^{(t)}|\theta')} > q \text{ ;} \\ \theta^{(t)}, & \text{othrerwise.} \end{cases}$
3. Return the values $\theta^{(1)}, \theta^{(2)}, ..., \theta^{(M)}$

We can summarize the Metropolis sampling as first computing

$$\alpha(b|a) = \min(\frac{P(b)r(a|b)}{P(a)r(b|a)}, 1) \tag{11}$$

and then accepting a candidate point with probability α (the **probability of a move**). This generates a Markov chain $(\theta^{(0)}, ..., \theta^{(k)}, ...)$ as the transition probabilities from $\theta^{(t)}$ to $\theta^{(t+1)}$ depends only on $\theta^{(t)}$ and not $(\theta^{(0)}, ..., \theta^{(t-1)})$. Following a sufficient **burn-in period** (of, say, k steps), the chain approaches its stationary distribution and samples from the vector $(\theta^{(k+1)}, ..., \theta^{(k+n)})$ are samples from $p(x)$.

Metropolis-Hasting Sampling as a Markov Chain To demonstrate that the Metropolis-Hasting sampling generates a Markov chain whose equilibrium density is that candidate density $p(x)$, it is sufficient to show that the Metropolis-Hasting transition probability satisfy the detailed balance equation with $p(x)$.

Using the Metropolis-Hasting algorithm, we sample from,

$$Q(b|a) = r(b|a)\alpha(b|a) = r(b|a) \cdot \min(1, \frac{P(b)r(a|b)}{P(a)r(b|a)}) \tag{12}$$

Thus if the transition probability satisfies $P(X = a)Q(X^{(t+1)} = b|X^{(t)} = a) = P(X = b)Q(X^{(t+1)} = a|X^{(t)} = b)$ then that stationary distribution corresponds to draws from the target distribution. We assume w.l.o.g that $(1 < \frac{P(b)r(a|b)}{P(a)r(b|a)})$ for $a \neq b$

$$\frac{Q(b|a)}{Q(a|b)} = \frac{r(b|a) \cdot \frac{P(b)r(a|b)}{P(a)r(b|a)}}{r(a|b) \cdot 1} = \frac{r(b|a)P(b)r(a|b)}{r(a|b)P(a)r(b|a)} = \frac{P(b)}{P(a)} \tag{13}$$

and the detailed balance equation holds.

5 Gibbs Sampling

The Gibbs sampler (introduced in the context of image processing by Geman and Geman 1984), is a special case of Metropolis-Hastings sampling wherein the random value is always accepted. The task remains to specify how to construct a Markov Chain whose values converge to the target distribution.

We will present this algorithm with the *motif finding problem*, Lets first recall the motif finding problem: given a set of n DNA sequences each of length t, find the profile that maximizes the consensus score.

5.1 Algorithm

1. Input $\vec{a} =< a_1, ..., a_n >$
2. Repeat for $t = 1, 2, ..., M$
   (a) Sample random position in the array $i \sim r(n)$
   (b) Set $\vec{b} =< b_1, ..., b_n > s.t.$
      i. $b_j = a_j \quad if \quad j \neq i$
      ii. $b_i \sim P(X_i|X_1 = a_1, ..., X_{i-1} = a_{i-1}, X_{i+1} = a_{i+1}, ..., X_n = a_n)$
3. Return $\vec{b} =< b_1, ..., b_n >$

This algorithm produces detailed balanced Markov chains. We are interested only in transitions from $\vec{a}$ to $\vec{b}$ where $a \neq b$ and when $\vec{a}$ and $\vec{b}$ differ only in one coordinate (result of the Gibbs algorithm).

Denote $\vec{a}_{-i} =< a_1, ..., a_{i-1}, a_{i+1}, ..., a_n >$

$$Q(b|a) = r(i)P(X_i = b_i|\vec{X}_{-i} = \vec{a}_{-i}) \tag{14}$$

$$Q(a|b) = r(i)P(X_i = a_i|\vec{X}_{-i} = \vec{b}_{-i}) \tag{15}$$

so the ratio between the stationary probability of two states is:

$$\frac{Q(b|a)}{Q(a|b)} = \frac{P(X_i = b_i|\vec{X}_{-i} = \vec{a}_{-i})}{P(X_i = a_i|\vec{X}_{-i} = \vec{b}_{-i})} \tag{16}$$

since $\vec{a}_{-i} = \vec{b}_{-i}$,

$$\frac{Q(b|a)}{Q(a|b)} = \frac{P(X_i = b_i|\vec{X}_{-i} = \vec{b}_{-i}) \cdot P(\vec{X}_{-i} = \vec{b}_{-i})}{P(X_i = a_i|\vec{X}_{-i} = \vec{a}_{-i}) \cdot P(\vec{X}_{-i} = \vec{a}_{-i})} = \frac{P(\vec{b})}{P(\vec{a})} \tag{17}$$

The Gibbs sampler is somewhat easier to implement than the Metropolis-Hasting algorithm since we don't have to create the proposal distribution. Furthermore, computing $p(x_1|x_2..x_n)$ can be done using the equation

$$p(x_1|x_2..x_n) = \frac{p(x_1..x_n)}{\sum_{x_1'} p(x_1', x_2..x_n)}$$

Most of the times the numerator and the denominator can be expressed as a product and then most of the elements will be reduced.

A sketch algorithm for the motif finding problem

- **Initialization**:
  - Select random locations in sequences $x_1, ..., x_N$
  - Compute an initial model M from these locations
- **Sampling Iterations:**
  - Remove one sequence x_i
  - Recalculate model
  - Pick a new location of motif in x_i according to the model