Metropolis Algorithm for Sampling a Probability Distribution

March 26, 2009

There are two sections here. The first section gives a brief overview of Markov chains and the Metropolis step. The second section provides a step-by-step algorithm for Metropolis sampling.

0.0.1 Markov chains and the Metropolis step

Briefly, a Markov chain is a discrete-time stochastic process, $\left\{\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots\right\}$, with the property that the conditional probability distribution of the current state, $\mathbf{x}^{(t)}$, given all previous states, $\left\{\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(t-1)}\right\}$, depends only on $\mathbf{x}^{(t-1)}$:

$$P\left(\mathbf{x}^{(t)}|\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(t-1)}\right) = P\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) \tag{1}$$

where $P\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right)$ is called the transition probability. After a large number of steps, the chain converges to the probability density, $p(\mathbf{x})$, for any staring value, $\mathbf{x}^{(0)}$, as long as the transition probability satisfies certain conditions [e.g., *Gamerman*, 1997; *MacKay*, 2003]. In other words, after a large number of steps the chain can be regarded as a set of samples drawn from $p(\mathbf{x})$. The Metropolis algorithm [e.g., *Metropolis et al.*, 1953; *Hastings*, 1970; *Gamerman*, 1997; *MacKay*, 2003] is one such MCMC method that satisfies convergence conditions.

Assuming the current state of a Markov chain, $\mathbf{x}^{(t)}$, the Metropolis algorithm generates the next state of the Markov chain, $\mathbf{x}^{(t+1)}$, by a two-step procedure. The first step generates a candidate state, $\mathbf{x}'$, using a proposal probability density, $q(\mathbf{x}'|\mathbf{x}^{(t)})$, which depends only on the current state and is assumed to be symmetric: $q(\mathbf{x}'|\mathbf{x}^{(t)}) = q(\mathbf{x}^{(t)}|\mathbf{x}')$. In the second step, the candidate $\mathbf{x}'$ is accepted as the next state (i.e., $\mathbf{x}^{(t+1)} = \mathbf{x}'$) with probability

$$P_{\text{accept}} = \min\left(1, \frac{p(\mathbf{x}')}{p(\mathbf{x}^{(t)})}\right). \tag{2}$$

Otherwise, the next state remains at the current state: $\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)}$. Iterations of the two-step procedure from a starting state $\mathbf{x}^{(0)}$ produce the Markov chain $\{\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots\}$. Early steps in the Markov chain may be influenced by a starting state $\mathbf{x}^{(0)}$. After sufficiently large number of steps, t_0, the chain $\{\mathbf{x}^{(t_0)}, \mathbf{x}^{(t_0+1)}, \dots\}$ can be considered as a set of samples drawn from $p(\mathbf{x})$.

0.0.2 Algorithm for sampling the posterior distribution

This explanation is a modified version of explanations provided in Fukuda and Johnson [2008], and in the Appendix of the paper in proof by Fukuda and Johnson, both of which I handed out in class.

We use the Metropolis algorithm to generate samples from a probability distribution, $p(\mathbf{x})$. Let M be the dimension of $\mathbf{x}$. As the proposal probability density, $q(\mathbf{x}'|\mathbf{x}^{(t)})$, we use the M-dimensional uniform distribution centered on $\mathbf{x}^{(t)}$:

$$q(\mathbf{x}'|\mathbf{x}^{(t)}) \propto \prod_{i=1}^{M} U\left(x_i^{(t)} - \Delta x_i, x_i^{(t)} + \Delta x_i\right) \tag{3}$$

where $U\left(x_i^{(t)} - \Delta x_i, x_i^{(t)} + \Delta x_i\right)$ is a 1-dimensional uniform distribution over the interval $\left[x_i^{(t)} - \Delta x_i, x_i^{(t)} + \Delta x_i\right]$, and Δx_i scales the step size of the Markov chain along the ith coordinate direction in the parameter space.

The algorithm proceeds as follows.

1. Initialize the counter $t = 0$ and set the initial state $\mathbf{x}^{(0)}$. That is, choose an initial value, $\mathbf{x}^{(0)}$.

2. Generate a candidate state $\mathbf{x}'$ from the proposal density $q(\mathbf{x}'|\mathbf{x}^{(t)})$. That is, let

$\mathbf{x}' = \mathbf{x}^{(t)} + \Delta x_i$ where Δx_i is a uniform random number selected from the interval $[-r, r]$ and r is a pre-specified "step size".

3. Compute the acceptance probability $P_{\text{accept}} = \min\left(1, \frac{p(\mathbf{x}')}{p(\mathbf{x}^{(t)})}\right)$.

4. Generate a random number u from a uniform distribution over the interval [0,1], $U(0,1)$: $u \sim U(0,1)$.

5. If $u \leq P_{\text{accept}}$, then accept the candidate state $\mathbf{x}'$, that is $\mathbf{x}^{(t+1)} = \mathbf{x}'$. If $u > P_{\text{accept}}$, then reject the candidate state and remain at the current state, that is $\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)}$.

6. Increase the counter from t to $t + 1$ and return to step 2.

The Markov chain is initially dependent on the starting value, $\mathbf{x}^{(0)}$, and therefore there is a burn-in stage during which the samples of the Markov chain do not sample $p(\mathbf{x})$. We discard these burn-in samples and regard $\mathbf{x}^{(t+1)}$ as a sample drawn from $p(\mathbf{x})$ only after the Markov chain converges.

The width of the proposal density, $q(\mathbf{x}'|\mathbf{x}^{(t)})$, which is determined by Δx_i (and by the step size, r), controls the efficency of the Metropolis algorithm. The step size, r, is effectively a tuning parameter that you will determine emperically. If r is too large, $p(\mathbf{x}')$ is likely to be low and the acceptance probability, P_{accept}, is low. In this case, the Markov chain is likely to remain at a state for many steps, reducing the efficiency of the random walk. If the step size, r, is too small, it takes a large number of steps for the Markov chain random walk to explore the entire high probability region.

Consider the burn-in samples to be the first $t_0 - 1$ samples. Discard these samples. Then the chain $\left\{\mathbf{x}^{(t_0)}, \mathbf{x}^{(t_0+1)}, \ldots\right\}$ can be considered as a set of samples drawn from $p(\mathbf{x})$. You can display the estimate of $p(\mathbf{x})$ by plotting a histogram (for 1-dimensional $\mathbf{x}$). You can compute the mean and standard deviation of the probability distribution with the mean and standard deviation of the set of samples.