STAT 238 - Bayesian Statistics
Lecture Twenty Four

Spring 2026, UC Berkeley

Aditya Guntuboyina

30 March 2026

Our next topic is Bayesian computation. The two main classes of techniques are Markov Chain Monte Carlo (MCMC) and Variational Inference (VI). We will start discussing MCMC today.

1 Markov Chain Monte Carlo

The goal is to obtain samples from a target distribution with density π (in our applications, π will be the posterior density). We assume that π is known only upto a normalizing constant. The normalizing constant can be determined by integrating the unnormalized π (over its domain) but this integration is assumed to be difficult.

We want to generate $\theta^{(0)}, \theta^{(1)}, \dots, \theta^{(N)}$ such that

$$\frac{1}{N} \sum_{t=1}^{N} g(\theta^{(t)}) \approx \int g(\theta)\pi(\theta) d\theta. \tag{1}$$

for all functions g.

$\theta^{(t)}$ is generated sequentially or iteratively for $t = 0, 1, 2, \dots$ with $\theta^{(t)}$ (and some external randomness) used to generate $\theta^{(t+1)}$.

The simplest MCMC algorithm is the Random Walk Metropolis Algorithm.

2 The Random Walk Metropolis Algorithm

The random-walk Metropolis algorithm works by generating proposals via a random walk-step from the current state and then accepting or rejecting them with certain probability. Suppose that the current state is $\theta^{(t)}$. It then proposes to move to the state $\theta^{(t)} + Z$ where Z is a random vector having a symmetric density around 0. Note that the density of the proposal $\theta^{(t)} + Z$ is given by $q(\theta^{(t)}, \cdot)$ where

$$q(x, y) = f_Z(y - x).$$

with f_Z denoting the density of Z. Because f_Z is symmetric about 0, we have

$$q(x, y) = q(y, x). \tag{2}$$

The function $q(\cdot,\cdot)$ is commonly referred to as the candidate-generating density or proposal-generating density. We say that q is symmetric if (2) holds. Clearly random walk Metropolis with a symmetric f_Z uses symmetric proposal-generating densities. With this notation, the random-walk Metropolis algorithm can be rephrased as follows. For x and y, let

$$\alpha(x, y) := \min\left(\frac{\pi(y)}{\pi(x)}, 1\right). \tag{3}$$

$\alpha(x, y)$ is referred to as the **acceptance probability** while going from x to y.

1. Initialize with an arbitrary value $\theta^{(0)}$.
2. Repeat the following for $t = 1, \ldots, N$:
   a) Let $x = \theta^{(t)}$.
   b) Generate a candidate or proposal y from the **symmetric** candidate generating density $q(x, \cdot)$ and a uniform random number $u \sim \text{Unif}(0, 1)$.
   c) If $u \geq \alpha(x, y)$, then set $\theta^{(t+1)} = x$. We say in this case that the proposed move from x to y is rejected and we stay at x.
   d) If $u < \alpha(x, y)$, then set $\theta^{(t+1)} = y$. We say in this case that the proposed move from x to y is accepted.
3. Return the values $\theta^{(1)}, \ldots, \theta^{(N)}$.

The above algorithm is known as the **Metropolis Algorithm**. It works for any symmetric (i.e., satisfying (2)) proposal-generating density $q(\cdot,\cdot)$. The Random-Walk Metropolis algorithm is a special case of this where $q(x, y)$ is of the form $f_Z(x - y)$ for a symmetric (about 0) density f_Z.

The Metropolis algorithm can be further generalized to deal with non-symmetric proposal generating densities. This generalization is referred to as the Metropolis-Hastings algorithm. We shall look at this algorithm in the next lecture. We will try to answer the following questions in the coming lectures:

1. What is the intuition behind the form of these algorithms?
2. Why do they actually work and give samples satisfying (1)?

To answer these questions, we need to know a little bit about Markov Chains. The algorithms above output samples $\theta^{(0)}, \ldots, \theta^{(N)}$. Clearly, $\theta^{(t+1)}$ is generated using only the value $\theta^{(t)}$. In other words, the values $\theta^{(0)}, \ldots, \theta^{(t-1)}$ are not used at all in order to generate $\theta^{(t+1)}$. Thus $\theta^{(0)}, \ldots, \theta^{(N)}$ can be seen as a realization of a sequence of random vectors $\Theta^{(0)}, \ldots, \Theta^{(N)}$ which form a Markov Chain. Properties of Markov Chains are used to justify (1), as we shall see in the coming lectures.