BE/Bi 103: Data Analysis in the Biological Sciences

Justin Bois

Caltech

Fall, 2016

3 The theory of Markov chain Monte Carlo

3.1 Why MCMC?

When doing Bayesian analysis, our goal is very often to compute a posterior distribution. For most inference problems, the posterior itself is the holy grail. However, just having an analytical expression for the posterior is of little use if we cannot understand and properties about it. Importantly, we often want to marginalized the posterior; that is, we want to integrate over parameters we are not interested in and get simpler distributions for those we are. This is often necessary to understand all but the simplest models. Doing these marginalizations requires what David MacKay calls "macho integration," which is often impossible to do analytically.

MCMC allows use to **sample** out of an arbitrary probability distribution, which includes pretty much any posterior we could write down.[12] We can trivially perform marginalizations from these samples and can generate histograms to plot various marginalizations. All we have to do is specify the distribution we want to sample from, and a good MCMC algorithm will take care of the rest.

Generating samples that actually come from the probability distribution of interest is not a trivial matter. We will discuss how this is accomplished through MCMC.

3.2 The basic idea behind MCMC

We often draw *independent* samples from a **target distribution**. For example, we could use `np.random.uniform(0, 1, 100)` to draw 100 independent samples from a uniform distribution on the domain $[0, 1]$. Generating independent samples for complicated target distributions is difficult.

But, the samples need not be independent! Instead, we only need that the samples be generated from a process that generates samples from the target distribution in the correct proportions. In the case of the parameter estimation problem, this distribution is the posterior distribution for model A parametrized by $\mathbf{a}$, $P(\mathbf{a} \mid D, A, I)$. For notational simplicity, since we know we are always talking about a posterior distribution, we will use $P(\mathbf{a})$ for shorthand notation.

The approach of MCMC is to take random walks in parameter space such that the probability that a walker arrives at point $\mathbf{a}$ is proportional to $P(\mathbf{a})$. This is the main concept and is

[12] Well, not *any*. For some cases, we may not be able to make a transition kernel that satisfies the necessary properties, which I describe in the following pages.

important enough to repeat.

> The approach of MCMC is to take random walks in parameter space such that the probability that a walker arrives at point $\mathbf{a}$ is proportional to $P(\mathbf{a})$.

If we can achieve such a walk, we can just take the walker positions as samples from the distributions. To implement this random walk, we define a **transition kernel**, $T(\mathbf{a}_{i+1} \mid \mathbf{a}_i)$, the probability of a walker stepping from position $\mathbf{a}_i$ in parameter space to position $\mathbf{a}_{i+1}$. The transition kernel defines a **Markov chain**, which you can think of as a random walker whose next step depends only on where the walker is right now; i.e., it has no memory.

The condition that the probability of arrival at point $\mathbf{a}_{i+1}$ is proportional to $P(\mathbf{a}_{i+1})$ may be stated as

$$P(\mathbf{a}_{i+1}) = \int \mathrm{d}\mathbf{a}_i \, T(\mathbf{a}_{i+1} \mid \mathbf{a}_i) \, P(\mathbf{a}_i). \tag{3.1}$$

Here, we have taken $\mathbf{a}$ to be continuous. Were it discete, we just replace the integral with a sum. When this relation holds, it is said that the target distribution is an **invariant distribution** or **stationary distribution** of the transition kernel. When this invariant distribution is unique, it is called a **limiting distribution**. We want to choose our transition kernel $T(\mathbf{a}_{i+1} \mid \mathbf{a}_i)$ such that $P(\mathbf{a})$ is limiting. This is the case if equation (3.1) holds and the chain is **ergodic**. An ergodic Markov chain has the following properties:

1. It is **aperiodic**. A periodic Markov chain can only return to a given point in parameter space after k, $2k$, $3k, \ldots$ steps, where k is the period. An aperiodic chain is not periodic.
2. It is **irreducible**, which means that any point in parameter space is accessible to the walker from any other.
3. It is **positive recurrent**, which means that the walker will surely come revisit any point in parameter space in a finite number of steps.

So, if our transition kernel satisfies this checklist and equation (3.1), it will eventually sample the posterior distribution. We will discuss how to come up with such a transition kernel in a moment, for for now we focus on the important concept of "eventually" in the preceding sentence.

3.3 Burn-in

Imagine for a moment that we devised a transition kernel that satisfies the above properties. Say we start a walker at position $\mathbf{a}_0$ in parameter space and it starts walking according to the transition kernel. It may not reach a place in parameter space where it is sampling from the limiting distribution. This is because the invariance condition, equation (3.1), does not hold for every set of parameter values. Once the walker reaches the limiting distribution, it is indeed sampling from it. So, we need to let the walker walk for a while without keeping track of the samples so that it can arrive at the limiting distribution. This is called **burn-in**, though Andrew Gelman and coauthors (in their famous book, *Bayesian Data Analysis*) prefer to call it "warm up."

There is no general way to tell if a walker has reached the limiting distribution, so we do not know how many burn-in steps are necessary. There are several heuristics. For example, Gelman and coauthors proposed generating several burn-in chains and computing the R statistic,

$$R = \frac{\text{variance between the chains}}{\text{mean variance within the chains}}. \tag{3.2}$$

Limiting chains have $R \approx 1$, so you can use this as a metric for having achieved stationarity.

3.4 Generating a transition kernel: The Metropolis-Hastings algorithm

The **Metropolis-Hastings algorithm** covers a widely used class of algorithms for MCMC sampling. I will first state the algorithm here, and then we will show that it satisfies the necessary conditions for the walkers to be sampling out of the target posterior distribution.

3.4.1 The algorithm/kernel

Say our walker is at position $\mathbf{a}_i$ in parameter space.

1. We randomly choose a candidate position $\mathbf{a}'$ to step next from an arbitrary **proposal distribution** $K(\mathbf{a}' \mid \mathbf{a}_i)$.
2. We compute the **Metropolis ratio**,

$$r = \frac{P(\mathbf{a}')\, K(\mathbf{a}_i \mid \mathbf{a}')}{P(\mathbf{a}_i)\, K(\mathbf{a}' \mid \mathbf{a}_i)}. \tag{3.3}$$

3. If $r \geq 1$, accept the step and set $\mathbf{a}_{i+1} = \mathbf{a}'$. Otherwise, accept the step with probability r. If we do reject the step, set $\mathbf{a}_{i+1} = \mathbf{a}_i$.

The last two steps are used to define the transition kernel $T(\mathbf{a}_{i+1} \mid \mathbf{a}_i)$. We can define the acceptance probability of the proposal step as

$$\alpha(\mathbf{a}_{i+1} \mid \mathbf{a}_i) = \min(1, r) = \min\left(1, \frac{P(\mathbf{a}_{i+1})\, K(\mathbf{a}_i \mid \mathbf{a}_{i+1})}{P(\mathbf{a}_i)\, K(\mathbf{a}_{i+1} \mid \mathbf{a}_i)}\right). \tag{3.4}$$

Then, the transition kernel is

$$T(\mathbf{a}_{i+1} \mid \mathbf{a}_i) = \alpha(\mathbf{a}_{i+1} \mid \mathbf{a}_i)\, K(\mathbf{a}_{i+1} \mid \mathbf{a}_i). \tag{3.5}$$

3.4.2 Detailed balance

This algorithm seems kind of nuts! How on earth does this work? To investigate this, we consider the joint probability, $P(\mathbf{a}_{i+1}, \mathbf{a}_i)$, that the walker is at $\mathbf{a}_i$ and $\mathbf{a}_{i+1}$ at sequential steps. We can write this in terms of the transition kernel,

$$\begin{aligned}
P(\mathbf{a}_{i+1}, \mathbf{a}_i) &= P(\mathbf{a}_i)\, T(\mathbf{a}_{i+1} \mid \mathbf{a}_i) \\
&= P(\mathbf{a}_i)\, \alpha(\mathbf{a}_{i+1} \mid \mathbf{a})\, K(\mathbf{a}_{i+1} \mid \mathbf{a}_i) \\
&= P(\mathbf{a}_i)\, K(\mathbf{a}_{i+1} \mid \mathbf{a})\, \min\left(1, \frac{P(\mathbf{a}_{i+1})\, K(\mathbf{a}_i \mid \mathbf{a}_{i+1})}{P(\mathbf{a}_i)\, K(\mathbf{a}_{i+1} \mid \mathbf{a}_i)}\right) \\
&= \min\left[P(\mathbf{a}_i)\, K(\mathbf{a}_{i+1} \mid \mathbf{a}_i), P(\mathbf{a}_{i+1})\, K(\mathbf{a}_i \mid \mathbf{a}_{i+1})\right] \\
&= P(\mathbf{a}_{i+1})\, K(\mathbf{a}_i \mid \mathbf{a}_{i+1})\, \min\left(1, \frac{P(\mathbf{a}_i)\, K(\mathbf{a}_{i+1} \mid \mathbf{a}_i)}{P(\mathbf{a}_{i+1})\, K(\mathbf{a}_i \mid \mathbf{a}_{i+1})}\right) \\
&= P(\mathbf{a}_{i+1})\, \alpha(\mathbf{a}_i \mid \mathbf{a}_{i+1})\, K(\mathbf{a}_i \mid \mathbf{a}_{i+1}) \\
&= P(\mathbf{a}_{i+1})\, T(\mathbf{a}_i \mid \mathbf{a}_{i+1}).
\end{aligned} \tag{3.6}$$

Thus, we have

$$P(\mathbf{a}_i)\, T(\mathbf{a}_{i+1} \mid \mathbf{a}_i) = P(\mathbf{a}_{i+1})\, T(\mathbf{a}_i \mid \mathbf{a}_{i+1}). \tag{3.7}$$

This says that the rate of transition from $\mathbf{a}_i$ to $\mathbf{a}_{i+1}$ is equal to the rate of transition from $\mathbf{a}_{i+1}$ to $\mathbf{a}_i$. In this case, the transition kernel is said to satisfy **detailed balance**.

Any transition kernel that satisfies detailed balance has $P(\mathbf{a})$ as an invariant distribution. This is easily shown.

$$\int \mathrm{d}\mathbf{a}_i\, P(\mathbf{a}_i)\, T(\mathbf{a}_{i+1} \mid \mathbf{a}_i) = \int \mathrm{d}\mathbf{a}_i\, P(\mathbf{a}_{i+1})\, T(\mathbf{a}_i \mid \mathbf{a}_{i+1})$$

$$
\begin{aligned}
&= P(\mathbf{a}_{i+1}) \left[\int \mathrm{d}\mathbf{a}_i \, T(\mathbf{a}_i \mid \mathbf{a}_{i+1})\right] \\
&= P(\mathbf{a}_{i+1}),
\end{aligned}
\tag{3.8}
$$

since the bracketed term is unity because the transition kernel is a probability.

Note that all transition kernels that satisfy detailed balance have an invariant distribution. (If the chain is ergodic, this is a limiting distribution.) But not all kernels that have an invariant distribution satisfy detailed balance. So, detailed balance is a sufficient condition for a transition kernel having an invariance distribution.

3.4.3 Choosing the transition kernel

There is an art to choosing the transition kernel. The original Metropolis algorithm (1953), took $K(\mathbf{a}_{i+1} \mid \mathbf{a}_i) = 1$. As a rule of thumb, you want to choose a proposal distribution such that you get an acceptance rate of about 0.4. If you accept every step, the walker just wander around and it takes a while to get to the limiting distribution. If you reject too many steps, the walkers never move, and it again takes a long time to get to the limiting distribution. There are tricks to "tune" the walkers to achieve this target acceptance rates.

Gibbs sampling, which is popular, though we will not go into the details, is a special case of a Metropolis-Hastings sampler, as is the No U-turn sampler (NUTS). These both result in significant performance improvements over important subclasses of problems. The sampler employed by `emcee`, the affine invariant ensemble sampler (Goodman and Weare, *J. Comp. Sci.*, **5**, 65–80, 2000), utilizes many walkers walking at the same time, sharing information between them. It is technically not a Metropolis-Hastings sampler, but many of the ideas presented in this lecture there apply for ensuring that the sampler is indeed sampling the appropriate posterior distribution.

Finally, importantly, the No U-Turn sampler and the affine invariant sample can only handle continuous variables; they cannot sample discrete variables. Depending on your problem, this could be a serious limitation.