ISC 5228
Markov Chain Monte Carlo

Metropolis Monte Carlo for Bayesian Inference

1 Goal

Use the Metropolis MCMC method to sample the posterior distribution that arises in Bayesian inference.

2 Introduction

Surveys that measure the prevalence of dishonesty are notoriously unreliable, because dishonest subjects often lie again on such surveys.[1]

For concreteness, consider a class with N students who take an exam. Suppose an unknown fraction p_{true} of the students cheat. After the exam, each student is asked (with complete amnesty) whether they cheated. Suppose X students answer "Yes". Our goal is to infer p_{true} from this data.[2]

However, this experimental design is flawed, because no student would admit to cheating. Fortunately there is an elegant design solution to this dishonesty problem that encourages honesty while protecting the privacy of the subject.

> During the interview process, the student flips a fair coin, hidden from the interviewer. The student agrees to answer honestly if the coin comes up heads.
>
> Otherwise, if the coin comes up tails, the student (secretly) flips the coin again, and answers "Yes, I did cheat" if the coin flip lands heads, and "No, I did not cheat", if the coin flip lands tails.
>
> This way, the interviewer does not know if a "Yes" was the result of a guilty plea, or a heads on a second coin toss. Thus privacy is preserved and the researchers receive honest answers.

Think about the setup, and note that due to this experimental design the probability of a cheater answering "No" or a non-cheater answering "Yes" are nonzero.

	cheater	non-cheater
"Yes"	0.75	0.25
"No"	0.25	0.75

Therefore, we have added noise to anonymize the surveys. One could criticize that the interviewers are still receiving contaminated data as some "Yes" answers are not confessions but instead randomness.

[1] This example in this lab is borrowed from this tutorial (chapter 2) on pyMC - a python package that uses MCMC for statistical inference.

[2] The inference is similar to example of a potentially biased coin considered in class.

However, the key advance is that we now have a systematic data generation process that can be modeled. More importantly, this method eliminates the primary need for deceitful answers. We can use Bayesian inference and Metropolis MCMC to dig through this noisy model, and find a posterior distribution for the true distribution p_{true}.

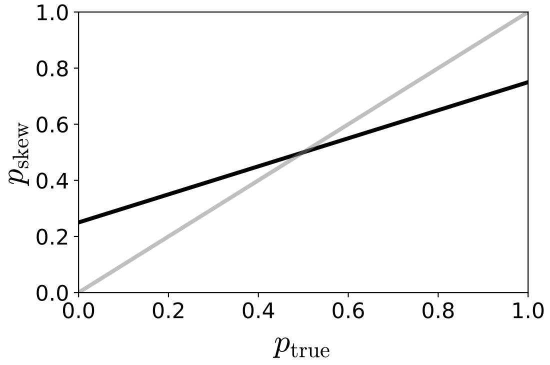
2.1 Model

We begin by asking what is the probability that a student will answer "Yes".

There are two paths that lead to a "Yes": (i) a cheater who flips heads in the first round, and (ii) a student (cheater or not) who first flips a tails, and then a heads. Thus,

$$\begin{aligned} P(\text{“Yes”}) =& P(\text{heads on } 1^{\text{st}} \text{ coin}) P(\text{cheater}) + P(\text{tails on } 1^{\text{st}} \text{ coin}) P(\text{heads on } 2^{\text{nd}} \text{ coin}) \\ =& \frac{1}{2} \cdot p_{\text{true}} + \frac{1}{2} \cdot \frac{1}{2} \\ p_{\text{skew}} =& \frac{p_{\text{true}}}{2} + \frac{1}{4} \end{aligned} \tag{1}$$

Thus, knowing p_{true} we know the probability a student will respond "Yes" which we label p_{skew}. Thus, p_{skew} has some information about p_{true}, but is contaminated with noise.



When there are no cheaters $p_{\text{skew}} = 0.25$, and when everyone cheats $p_{\text{skew}} = 0.75$.

The key insight is that the number of students who answer "Yes" (X) follow a binomial distribution with this skewed probability.

$$X \sim \text{Binomial}(N, p_{\text{skew}}) \tag{2}$$

This informs our likelihood model in Bayesian inference.

3 Bayesian Inference Model

For concreteness, let $N = 100$ and $X = 35$. What does this tell us about p_{true}? Is it greater than $X/N = 0.35$ or smaller? What is the associated uncertainty?

Suppose, we have no idea about p_{true}, so we assume a noninformative prior $\pi(p_{\text{true}}) = U[0, 1] = 1$. The likelihood is given by the binomial distribution eqn 2 as

$$\pi(X|p_{\text{true}}) = {}^{N}C_{X}\, p_{\text{skew}}^{X} (1 - p_{\text{skew}})^{N-X} \tag{3}$$

The posterior distribution is

$$\pi(p_{\text{true}}|X) \propto \pi(X|p_{\text{true}})\pi(p_{\text{true}}) = {}^{N}C_{X}\, p_{\text{skew}}^{X}(1-p_{\text{skew}})^{N-X} \times 1$$

Fortunately, MCMC does not require the target distribution to be normalized. As usual, we take the 'log' on both sides to avoid overflow/underflow, and drop all terms that do not depend on p_{true}.

$$\log \pi(p_{\text{true}}|X) = X \log p_{\text{skew}} + (N-X)\log(1-p_{\text{skew}}) + \text{constant} \tag{4}$$

4 Exercises

Write down functions for (i) the target distribution,[3] (ii) proposal for p_{true} from a uniform or normal distribution,[4] (iii) acceptance or rejection based on Metropolis criterion, and (iv) driver.

(i) Start with p_{true}^{0} = 0.35, and perform a MCMC simulation of 5000 steps. Report the acceptance ratio, a traceplot, and a histogram of the target distribution.

(ii) Vary the step size "δ" in the proposal, and comment on the acceptance ratio, and the ability to sample the target distribution. Select a δ that yields an acceptance ratio of 0.4 $\pm$ 0.1.

(iii) Run the MCMC simulation long enough to ensure convergence ($\hat{R} < 1.1$, say).

(iv) Based on the samples of the posterior distribution, what is your estimate of the p_{true}. Provide a mean and standard deviation.

[3] make sure to return $\exp\left(\log \pi(p_{\text{true}})\right)$ only for $0 \leq p_{\text{true}} \leq 1$.

[4] obtain p_{skew} using eqn 1