# Monte Carlo Methods — Why Do They Work? — Part III

In the previous two "Why Do They Work?" write-ups, I proved that the Metropolis algorithm worked and that the Metropolis-Hastings algorithm worked:

* https://brianhill.github.io/bayesian-statistics/resources/MonteCarloMethodsWhyDoTheyWork-I.nb.pdf
* https://brianhill.github.io/bayesian-statistics/resources/MonteCarloMethodsWhyDoTheyWork-II.nb.pdf

Both proofs required the Principle of Detailed Balance. Now we are going to prove that the Gibbs Sampling (GS) algorithm works, but most of the hard work was actually in the second write-up, because Gibbs sampling can be viewed a special case of Metropolis-Hastings (and Metropolis is a special case of Metropolis-Hastings too).

To get going, and to leverage our previous write-up, we need to understand the relationship between Metropolis-Hastings and Gibbs sampling.

## The Metropolis-Hastings and Gibbs Sampling Algorithms

The core of the Metropolis-Hastings algorithm (repeated *ad nauseam)* was:

**Step 1:** You are in bin $i$. You propose a random move from bin $i$ to some other bin $j$. The probability of proposing the random move is denoted $g(j \mid i)$, which is read "$g$ of $j$ given $i$." (Note: If you want to recover Metropolis as a special case of Metropolis-Hastings, you put in that $g(i + 1 \mid i) = 0.5$ and $g(i - 1 \mid i) = 0.5$. In other words, you have 50% chance of moving to either of the nearest neighbors, and in Metropolis there is no chance of moving to anything but a nearest neighbor.)

**Step 2:** Compute the ***clamped*** appropriate ratio. The **clamped** appropriate ratio for going from $i \to j$ is $\min\left(\frac{p_j}{p_i}\ \frac{g(i \mid j)}{g(j \mid i)}, 1\right)$.

**Step 3:** Generate a random number between 0 and 1. If the number is less than clamped appropriate ratio, move to the proposed bin, and make a tally there. Otherwise stay in the current bin and make another tally in the current bin. Whatever bin you made the tally in is now the current bin, and you go back to step 1.

"Hold on to your papers, fellow scholars [apologies to Dr. Karoly Zsolnai-Feher]," here is what Gibbs sampling is:

You just choose $g(i \mid j) = p_i$.

Really!? That is almost laughably simple and it is also ridiculous. I'll get to why it is ridiculous if you don't already see why. Anyway, let's see what happens to the clamped appropriate ratio.

We put $g(i \mid j) = p_i$ and $g(j \mid i) = p_j$ into the clamped appropriate ratio and get:

$$\min\left(\frac{p_j}{p_i}\,\frac{g(i\,|\,j)}{g(j\,|\,i)},\,1\right) = \min\left(\frac{p_j}{p_i}\,\frac{p_i}{p_j},\,1\right) = \min(1,\,1) = 1$$

The clamped appropriate ratio is always 1! So you always move. So let us summarize the Gibbs sampling algorithm (to be repeated *ad nauseam):*

**Step 1:** The current bin is bin *i*. You propose a random move from bin *i* to some other bin, bin *j*. The probability of the proposed random move to bin *j* is just $p_j$, where $p_j$ is the probability distribution you are trying to sample.

**Step 2:** You accept the proposed move and make a tally in the proposed bin. The proposed bin is now the current bin, and you go back to step 1.

Now I'll say why this laughably simple algorithm is also ridiculous. The whole point of any Monte Carlo method is to generate a representative set of samples ***for a probability distribution that is hard to sample.*** The Gibbs sampling algorithm says to ***make the proposed move by drawing a representative sample from that very same probability distribution*** that you already had resigned yourself as being hard to sample. So it has "solved" the problem by requiring you to know the solution to the problem. ***This appears to be a circular solution with no benefit.***

Of course it isn't ridiculous. The actual genius of the Gibbs sampling algorithm comes next.

## High-Dimensional Probability Distributions

Back on Nov. 22, I launched the entire section on Monte Carlo Methods with my "Monte Carlo Methods Introduction" write-up:

* https://brianhill.github.io/bayesian-statistics/resources/MonteCarloMethodsIntroduction.nb.pdf

My example was a probability distribution that has 50 dimensions! It was the electoral college outcome. Now each state can only go one of two ways, so each axis was as simple as an axis can get. But still the total number of outcomes was $2^{50}$. We did the math and saw that there is absolutely no way to exhaustively sample that space with any reasonable amount of computer time.

As a second example, I looked ahead to what will be our case study, which was a study of 106 babies vaccinated for Hepatitis B. For each baby, there are multiple blood samples, and the hepatitis "titer" is measured in each blood sample. The blood sample data is going to be fitted by a slope and an intercept for each baby (in the study, they actually take the logarithm of the titer before doing the linear

fit). So that is 212 parameters. There is no way you conclude an epidemiological paper by reporting 212 different numbers and error bars on each of them, so the authors introduced five more parameters that capture the distributions of those 212 parameters and the uncertainty in the titer data. So in total there are 217 parameters. The bottom line is that in this second example, you have a space with 217 parameters (instead of an electoral college with 50 states), and these are now continuous parameters (not just binary wins and losses). You can see that this is going to be a far worse probability distribution to sample than the electoral college problem, and that one was already practically impossible to exhaustively sample.

To summarize, we are motivated to study high-dimensional probability distributions with Monte Carlo methods because (a) they show up in interesting, real-world examples, and (b) they are impossible to exhaustively sample.

To study high-dimensional probability distributions, we need some new notation. Instead of bins labeled by an index $i$, the bins will now be labeled by $D$ indices, $i_1$, $i_2$, $i_3$, ..., $i_D$. So the probability distribution we are trying to sample is now denoted:

$$p_{i_1\, i_2\, i_3 \ldots\, i_D}$$

The simplest case that has more than one-dimension would have $D = 2$, and the $p$'$s$ would look like:

$$p_{i_1\, i_2}$$

To make that even more concrete, we could have a really simple example in mind. How about the first axis represents whether a nodule that comes out of the potato harvester and is destined for the kitchen is a rock or a potato, and the second axis represents whether it is small, medium, or large. Then the six $p$'s would be:

$p_{\text{rock small}}$
$p_{\text{rock medium}}$
$p_{\text{rock large}}$
$p_{\text{potato small}}$
$p_{\text{potato medium}}$
$p_{\text{potato large}}$

Of course, the total of all 6 $p$'s has to add up to 1 (assuming there aren't other things in the sack going to the kitchen, like rotten turnips).

Anyway, examples aside, it is for multi-dimensional probability distributions that Gibbs sampling has one more twist, why it is not just a circular solution with no benefit, and why it is in widespread use.

## Gibbs Sampling of High-Dimensional Probability Distributions

Our plan of attack has two parts: (1) We are going to move along only one axis at a time in the Gibbs sampling, and (2) assume that a substantial burden of calculating $p_{i_1\, i_2\, i_3 \ldots\, i_d}$ is simplified by the fact that along at least some of the axes, the $p$'$s$ factor.

### 1A. Conditional Movement

We introduce the idea of "conditional movement." Specifically, we are going to propose a movement that keeps the indices of all the axes the same, except for one of them. If we are in the bin with indices

$$i_1\, i_2 \ldots i_d \ldots\, i_D$$

we will propose a movement such that only the $d$th axis changes and the new $d$th index will be $i_d$'.

This is called "conditional movement" because we have the "condition" that all the other indices stay the same. We need a new set of $p$'s that captures this idea, which we'll denote:

$$p_{i_d'|i_-}$$

You can read $i_-$ as "all the $i$'s except the $d$th one." You can read $i_d$' as the "proposed $d$th bin." You can read the whole expression as "the probability of moving to the proposed $d$th bin given that all the bins except the $d$th one stay the same." Also, "conditioned on" and "given" are synonyms, so you will sometimes see this read as "the probability of moving to the proposed $d$th bin conditioned on all the bins but the $d$th one staying the same."

NOTE: It is definitely not the case that $p_{i_d'|i_-}$ is the same as $p_{i_1\, i_2 \ldots\, i_d' \ldots\, i_D}$. But there is a relation:

$$p_{i_d'|i_-} = \frac{p_{i_1\, i_2 \ldots\, i_d' \ldots\, i_D}}{\sum_{i_d} p_{i_1\, i_2 \ldots\, i_d \ldots\, i_D}}$$

### 1B. The Principle of Detailed Balance

This is a new rule for movement, and you might be worried that our proof that it works still applies. Or to put it another way, this no longer seems like a specialization of Metropolis-Hastings as was originally claimed. But you can see that the proof definitely does still apply along the chosen axis. You bounce around on that axis using this rule, and you will properly sample that axis.

The leap of faith is that you can change up what axes you bounce around on, and over time you will build a sample of the multi-dimensional distribution that works along all axes.

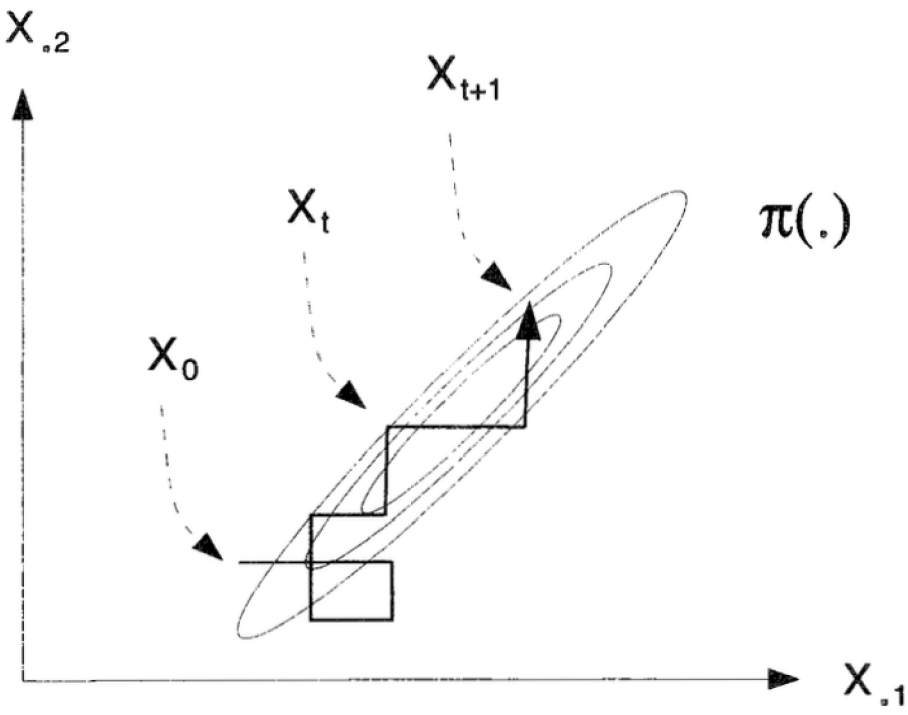
You can think of this as another detailed balance proof. Is it not the case, that along a given axis the tallies will settle down to the right ratios? And is it not the case that if we cycle through all the axes of movement enough times that along any given axis the tallies will settle down to the right ratios? So it must be the case that all the tallies along all the axes must settle down to the right ratios.

## 1C. Visualizing Gibbs Sampling

At this point in our study of Monte Carlo, we have left Donovan and Mickey behind, because I don't think their presentation particularly illuminates what I claim is a laughably simple method. For Gibbs sampling, I am instead using as my reference the first two chapters of the book edited by Gilks, Richardson, and Spiegelhalter. David Spiegelhalter is the author of the program that launched this methodology into the mainstream of epidemiology, **B**ayesian Inference **U**sing **G**ibbs **S**ampling (BUGS). The other authors are also leaders in Markov Chain Monte Carlo analyses. On this web page, they have made a "Preview PDF" of most of the first two chapters of their book readily available:

* https://www.taylorfrancis.com/books/mono/10.1201/b14835

Here is their Figure 1.2 showing this axis-by-axis movement for the two-dimensional case:




The diagonal ellipses represent the region where the joint probability distribution is high. The meandering path consisting of straight-line movements in one axis and then the other represents movements such that the Gibbs sampling algorithm discovers and representatively samples the high

probability region.

## 2. Factorization of the *p*'s

We now are going to remove a very substantial burden in sampling

$$p_{i_d'|i-} = \frac{p_{i_1 i_2 i_3 \ldots i_d' \ldots i_D}}{\sum_{i_d} p_{i_1 i_2 i_3 \ldots i_d \ldots i_D}}$$

by making some assumptions. We are going to assume (at least along most axes), that the very complicated expressions for the *p*'s factor into a bunch of terms, each of which is much simpler. Furthermore, once the *p*'s have been shown to factor, we will see something very similar happen to what happened back in Chapters 11 and 12 of Donovan and Mickey. Specifically, we will see that despite all the complexity, if we start off with Gaussian- and Gamma-distributed priors, and if we multiply by Gaussian likelihoods, then we get Gaussian- and Gamma-distributed posteriors.

The whole goal of all this Gibbs sampling technology is to sample the posterior, and I am claiming that viewed along one axis at a time, every one of the of the $p_{i_d'|i-}$ will turn out to either a one-dimensional Gaussian, or a one-dimensional Gamma-distribution, and those we already know how to sample. This only works if you imagine that all the other axes are constant, and you only move along one axis at a time.

Frankly, I don't think I can state the simplifying assumptions in all generality. It is easier for me to tell you how it is going to work out in the case of the babies immunized for Hepatitis B that I introduced back on Nov. 22nd and re-introduced earlier in this document in the section titled "High-Dimensional Probability Distributions." Again, there were 217 axes. Unlike the electoral college example, where every state was either a win or a loss, each of these axes is a continuous parameter. So not only have we gone from 50 axes to 217 axes, each axis now has an infinity of values. Even if we binned the continuous values of each parameter arbitrarily into something like 5 bins per parameter, we would still have $5^{217}$ outcomes, each with a different probability. So we have gone from something that we estimated would take 30 years of computing time to something that would take $1.3 * 10^{138}$ years of computing time. If you don't think that is an impossible amount of computing time, remember that the universe is only $1.3 * 10^{10}$ years old.

# Conclusions

The goals of this third and last of my "Why Do They Work?" write-ups were to (a) explain Gibbs sampling, (b) convince you that it works, (c) suggest that with some simplifying assumptions, instead of being a ridiculous and circular solution, it could be efficient. I hope you now believe those three points. If so, for our last class on Tuesday Dec. 10th you are in a good position to appreciate the "Monte Carlo Methods Case Study" which I will present then.