The Metropolis-Hastings algorithm by example

John Kerl

December 21, 2012

Abstract

The following are notes from a talk given to the University of Arizona Department of Mathematics Graduate Probability Seminar on February 14, 2008.

- Consider flipping a fair coin 100 times. The probability space has size 2^{100}, which is on the order of 10^{69}. Yet we can easily sample from this space: each event occurs with equal probability $(1/2)^{100}$, and all we have to do is conduct 100 independent Bernoulli experiments with $p = 1/2$.
- If the coin isn't fair, not all events occur with equal probability. Yet it's still easy to sample from this space: we conduct 100 independent Bernoulli experiments with $0 \leq p \leq 1$.
- Now suppose the coin flips aren't independent — they have some peer pressure. This system is called a *one-dimensional Ising model* and it's non-trivial to sample from this space.

I will describe how the Metropolis-Hastings algorithm allows us to conduct experiments using such a model.

This paper is under construction.

Contents

1 Introduction

Coin flips are used as a motivating example to describe why one would want to use the Metropolis-Hastings algorithm. The algorithm is presented, illustrated by example, and then proved correct. See [Kerl] for probability terminology and notation used in this paper.

2 Probability distributions

Consider flipping n coins. Encode a heads-up flip as $+1$ and a tails-up flip as -1. Then the result of the ith flip is $\omega_i \in \{+1, -1\}$. I'll discuss four scenarios, with increasing levels of complexity. Write ordered n-tuples

$$\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n).$$

Notice that there are $N = 2^n$ outcomes in the probability space

$$\Omega = \{(\omega_1, \ldots, \omega_n) : \omega_i = \pm 1\}.$$

For example, with $n = 3$ coins, there are $N = 8$ outcomes: HHH, HHT, HTH, HTT, THH, THT, TTH, and TTT.

2.1 IID fair flips

In the first case I consider, the flips are independent and identically distributed. Furthermore, each coin is fair, which is to say

$$P_i(+1) = 1/2 \qquad \text{and} \qquad P_i(-1) = 1/2.$$

Due to independence, the probabilities factor:

$$\begin{aligned} P(\boldsymbol{\omega}) &= P(\omega_1, \ldots, \omega_n) \\ &= P_1(\omega_1) \cdots P_n(\omega_n) \\ &= \prod_{i=1}^{n} \frac{1}{2} = \left(\frac{1}{2}\right)^n. \end{aligned}$$

For example, with $n = 3$, the probability of heads, tails, heads is

$$P(+1, -1, +1) = 1/8.$$

The other 7 possible outcomes also occur with probability 1/8. Here's the PMF:

$(\omega_1, \omega_2, \omega_3)$	$P(\omega_1, \omega_2, \omega_3)$
(1 1 1)	0.125
(1 1 -1)	0.125
(1 -1 1)	0.125
(1 -1 -1)	0.125
(-1 1 1)	0.125
(-1 1 -1)	0.125
(-1 -1 1)	0.125
(-1 -1 -1)	0.125

2.2 IID flips

In the second case I consider, the n flips are still IID but the coin isn't necessarily fair. Each lands heads-up with probability p, where $0 \le p \le 1$. Due to independence, the probabilities still factor. Let

$$\eta_i = \begin{cases} 0, & \omega_i = +1 \\ 1, & \omega_i = -1. \end{cases} \tag{2.1}$$

That is,

$$\omega_i = 1 - 2\eta_i \qquad \text{and} \qquad \eta_i = \frac{1-\omega_i}{2}. \tag{2.2}$$

Then

$$\begin{aligned} P(\boldsymbol{\omega}) &= P(\omega_1, \ldots, \omega_n) \\ &= P_1(\omega_1) \cdots P_n(\omega_n) \\ &= \prod_{i=1}^{n} p^{1-\eta_i}(1-p)^{\eta_i}. \end{aligned}$$

With $n = 3$, the probability of heads, tails, heads is

$$\begin{aligned} P(1,-1,1) &= \left(p^1(1-p)^0\right)\left(p^0(1-p)^1\right)\left(p^1(1-p)^0\right) \\ &= p^2(1-p). \end{aligned}$$

The PMF is as follows, with $p = 0.9$:

$(\omega_1, \omega_2, \omega_3)$	$P(\omega_1, \omega_2, \omega_3)$
(1 1 1)	0.729
(1 1 -1)	0.081
(1 -1 1)	0.081
(1 -1 -1)	0.009
(-1 1 1)	0.081
(-1 1 -1)	0.009
(-1 -1 1)	0.009
(-1 -1 -1)	0.001

2.3 Independent flips

In the third case I consider, the coin flips are still independent but they are not identically distributed: each has its own probability p_i of heads. That is,

$$P_i(\omega_i = +1) = p_i.$$

Then

$$\begin{aligned} P(\boldsymbol{\omega}) &= P(\omega_1, \ldots, \omega_n) \\ &= P_1(\omega_1) \cdots P_n(\omega_n) \\ &= \prod_{i=1}^{n} p_i^{1-\eta_i}(1-p_i)^{\eta_i}. \end{aligned} \tag{2.3}$$

With $n = 3$, the probability of heads, tails, heads is

$$P(1, -1, 1) = p_1(1 - p_2)p_3.$$

With $p_1 = 0.6$, $p_2 = 0.7$, and $p_3 = 0.8$, the PMF looks like this:

$(\omega_1, \omega_2, \omega_3)$	$P(\omega_1, \omega_2, \omega_3)$
(1 1 1)	0.336
(1 1 -1)	0.084
(1 -1 1)	0.144
(1 -1 -1)	0.036
(-1 1 1)	0.224
(-1 1 -1)	0.056
(-1 -1 1)	0.096
(-1 -1 -1)	0.024

2.4 Dependent flips

In the fourth and final case I consider, the coin flips are no longer independent — perhaps we are flipping magnetized coins, where the result of a flip is influenced by one or more of its neighbors. The PMF is a function with potentially 2^n different input/output values. There are all sorts of PMFs one could write down to define a probability model for dependent coin flips. For example:

$(\omega_1, \omega_2, \omega_3)$	$P(\omega_1, \omega_2, \omega_3)$
(1 1 1)	0.4
(1 1 -1)	0.0
(1 -1 1)	0.1
(1 -1 -1)	0.1
(-1 1 1)	0.0
(-1 1 -1)	0.1
(-1 -1 1)	0.1
(-1 -1 -1)	0.2

However, it's not clear what kind of physical situation could lead to such an ad-hoc PMF.

There is one particular family of models — the Ising family — which contains enough flexibility to describe plenty of dependent-flip situations. Furthermore, the Ising family has some physical plausibility.

3 The one-dimensional Ising model

I'll refer to **spins** rather than **flips**: heads and tails will be replaced with spin-up and spin-down, respectively. A sequence such as HHTH will now be $++-+$, and I will call that a **spin configuration**.

In the one-dimensional Ising model, there are n spins and $N = 2^n$ possible states (spin configurations), and the individual spins in a configuration are not necessarily independent.

Let $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ as before. We define an **energy function** on the spins ω_i by

$$E(\boldsymbol{\omega}) = \sum_{i=1}^{n} \sum_{j=1}^{n} S_{ij} \omega_i \omega_j + \sum_{i=1}^{n} h_i \omega_i.$$

The S_{ij}'s are **coupling coefficients** which specify the dependence of one spin on another. The h_i's are the **magnetic field term**, which breaks the symmetry.

We get a probability model using the statistical-mechanical notation that *energy is the logarithm of probability.* The probability of each configuration $\boldsymbol{\omega}$ is a Boltzmann distribution with inverse temperature $\beta = 1/kT$ (where k is Boltzmann's constant):

$$P(\boldsymbol{\omega}) \propto e^{-\beta E(\boldsymbol{\omega})}. \tag{3.1}$$

To normalize this, so that $\sum_{\boldsymbol{\omega}} P(\boldsymbol{\omega}) = 1$, we divide by

$$Z = \sum_{k=1}^{N} e^{-\beta E(\boldsymbol{\omega}^{(k)})}$$

where $\boldsymbol{\omega}^{(k)}$ runs over all possible states. This denominator has a special name: it is called a **partition function**. Thus,

$$P(\boldsymbol{\omega}^{(j)}) = \frac{e^{-\beta E(\boldsymbol{\omega}^{(j)})}}{\sum_{k=1}^{N} e^{-\beta E(\boldsymbol{\omega}^{(k)})}}.$$

Some handy notation: Write the **probability numerator** as

$$P^*(\boldsymbol{\omega}^{(j)}) = e^{-\beta E(\boldsymbol{\omega}^{(j)})}. \tag{3.2}$$

Then

$$P(\boldsymbol{\omega}^{(j)}) = \frac{P^*(\boldsymbol{\omega}^{(j)})}{Z}. \tag{3.3}$$

We will see (if I get to it) that the diagonal terms S_{ii} do not affect spin probabilities. Here are some examples of S matrices, with $n = 3$.

Non-interacting:

$$S = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Mean field:

$$S = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

Aligning nearest-neighbor:

$$S = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

Anti-aligning nearest-neighbor:

$$S = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{pmatrix}$$

Aligning nearest-neighbor
with periodic boundary conditions:

$$S = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

Anti-aligning nearest-neighbor
with periodic boundary conditions:

$$S = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ -1 & 0 & 0 \end{pmatrix}$$

Here are the PMFs for the first three of those, with $h = 0$:

$(\omega_1, \omega_2, \omega_3)$	$P(\omega_1, \omega_2, \omega_3)$ Non-interacting	$P(\omega_1, \omega_2, \omega_3)$ Mean field	$P(\omega_1, \omega_2, \omega_3)$ Aligning nearest neighbor
(1 1 1)	0.125	0.4994973	0.3879017
(1 1 -1)	0.125	0.0001676	0.0524968
(1 -1 1)	0.125	0.0001676	0.0071047
(1 -1 -1)	0.125	0.0001676	0.0524968
(-1 1 1)	0.125	0.0001676	0.0524968
(-1 1 -1)	0.125	0.0001676	0.0071047
(-1 -1 1)	0.125	0.0001676	0.0524968
(-1 -1 -1)	0.125	0.4994973	0.3879017

4 Independence and CDF sampling

In this section we discuss two sampling methods which are simpler than Metropolis-Hastings: independence sampling, which works for the independent cases, and CDF sampling which works for all four examples of section 2 but doesn't scale well as n increases.

4.1 Sample proportions

Suppose we flip four fair coins. Getting all four heads, or all four tails, is no huge surprise — but getting two heads and two tails is more likely. On the other hand, if we flip a thousand fair coins, all heads or all tails are very unlikely and we expect about 500 heads in general. It is quite possible to quantify these expectations precisely, in terms of sample mean and variance, using the binomial theorem as described in [Kerl]: IID fair flips lend themselves to neat theoretical analysis. For purposes of this paper, though, the goal is to describe experimental methods which are useful for probability models which are *not* theoretically neat. It is more important to look at sample proportions, which allow us to tabulate our experimental results.

Suppose we conduct (in a manner which subsequent sections will make precise) M experiments of an n-coin flip, using any of the four models of section 2, with outcomes

$$\boldsymbol{\omega}^{(1)}, \ldots, \boldsymbol{\omega}^{(M)}.$$

Then, for $j = 1, \ldots, N = 2^n$, define Q_i to be the fraction of $\boldsymbol{\omega}$'s which landed in state i. I also sometimes write, for brevity,

$$\mathbf{P} = (P_1, \ldots, P_N) \quad \text{and} \quad \mathbf{Q} = (Q_1, \ldots, Q_N).$$

For example, if I flip $N = 3$ fair coins $M = 10$ times, I might get the following:

i	$\boldsymbol{\omega}^{(i)}$
1	(1 1 1)
2	(-1 1 -1)
3	(-1 1 1)
4	(-1 1 -1)
5	(-1 -1 1)
6	(1 -1 1)
7	(1 -1 1)
8	(-1 -1 1)
9	(1 -1 -1)
10	(1 1 1)

Then I can tabulate the sample proportions, which I can compare against the PMF:

$\boldsymbol{\omega}$	$P(\boldsymbol{\omega})$	$Q(\boldsymbol{\omega})$
(1 1 1)	0.125	0.2
(1 1 -1)	0.125	0.0
(1 -1 1)	0.125	0.2
(1 -1 -1)	0.125	0.1
(-1 1 1)	0.125	0.1
(-1 1 -1)	0.125	0.2
(-1 -1 1)	0.125	0.2
(-1 -1 -1)	0.125	0.0

If I re-run this experiment with $M = 10000$ trials, I might get the following sample proportions:

$\boldsymbol{\omega}$	$P(\boldsymbol{\omega})$	$Q(\boldsymbol{\omega})$
(1 1 1)	0.125	0.1247
(1 1 -1)	0.125	0.1191
(1 -1 1)	0.125	0.1250
(1 -1 -1)	0.125	0.1281
(-1 1 1)	0.125	0.1237
(-1 1 -1)	0.125	0.1307
(-1 -1 1)	0.125	0.1231
(-1 -1 -1)	0.125	0.1256

The **law of large numbers** tells us that each of the eight Q_i's approach the respective P_i's as the number M of trials increases; the **central limit theorem** tells us about the distribution of the Q_i's.

The next question is how to actually conduct such an experiment in a computer program.

4.2 Sampling with the independence method

For independent flips, e.g. any of the first three models in section 2, it's easy to produce samples. Recall that p_j is the probability that the jth flip is heads, which we encode as $+1$. Thus, the algorithm is straightforward.

```
*Initialize the histogram bins:*
For $i = 1, \ldots, N$ (number of possible outcomes):
    counts$_i = 0$.

*Generate states and collect data:*
For $i = 1, \ldots, M$ (number of trials):
    For $j = 1, \ldots, n$ (number of coins):
        $U$ = uniform random on $[0, 1)$.
        If $U < p_j$ then $\omega_j = +1$ else $\omega_j = -1$.
    Find the index $k$ of the state $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ which was just generated.
    Increment counts$_k$ by 1.

*Scale the histogram bins down to sample proportions:*
For $i = 1, \ldots, N$ (number of possible outcomes):
    Divide counts$_i$ by $M$.
```

The output of such an experiment was shown in section 4.1.

The main limitation of this algorithm is obvious: it only works for independent-flip models.

4.3 Sampling with the CDF method

For all four models of section 2 — independent or not — it is possible to sample using a technique which is almost as simple as the independence sampling of the previous section.

The key concept is the **cumulative distribution function**. That is, we write down the N possible outcomes in some order $\boldsymbol{\omega}^{(1)}$ through $\boldsymbol{\omega}^{(N)}$, then tabulate at the jth slot the probability of obtaining states $\boldsymbol{\omega}^{(1)}$ through $\boldsymbol{\omega}^{(j)}$. That is, the CDF, as a function of j from 1 to N, is

$$C(j) = P(\{\boldsymbol{\omega}^{(1)}, \ldots, \boldsymbol{\omega}^{(j)}\}).$$

E.g.

$$\begin{aligned} C(1) &= P(\boldsymbol{\omega}^{(1)} \\ C(2) &= P(\boldsymbol{\omega}^{(1)} + P(\boldsymbol{\omega}^{(2)} \\ C(3) &= P(\boldsymbol{\omega}^{(1)} + P(\boldsymbol{\omega}^{(2)} + P(\boldsymbol{\omega}^{(3)} \\ &\vdots \\ C(N) &= P(\boldsymbol{\omega}^{(1)} + \ldots + P(\boldsymbol{\omega}^{(N)}. \end{aligned}$$

For example, using the IID-fair model, we have

$\boldsymbol{\omega}$	$P(\boldsymbol{\omega})$	CDF
(1 1 1)	0.125	0.125
(1 1 -1)	0.125	0.250
(1 -1 1)	0.125	0.375
(1 -1 -1)	0.125	0.500
(-1 1 1)	0.125	0.625
(-1 1 -1)	0.125	0.750
(-1 -1 1)	0.125	0.875
(-1 -1 -1)	0.125	1.000

Likewise, using the aligning nearest-neighbor Ising model from section 3, with $h = 0.1$, we have

$\boldsymbol{\omega}$	$P(\boldsymbol{\omega})$	CDF
(1 1 1)	0.3879017	0.3879017
(1 1 -1)	0.0524968	0.4403985
(1 -1 1)	0.0071047	0.4475032
(1 -1 -1)	0.0524968	0.5000000
(-1 1 1)	0.0524968	0.5524968
(-1 1 -1)	0.0071047	0.5596015
(-1 -1 1)	0.0524968	0.6120983
(-1 -1 -1)	0.3879017	1.0000000

The algorithm is as follows.

```
Turn the PMF into a CDF:
Sum = 0.
For $i = 1, \ldots, N$ (number of possible states):
    Sum := sum + $p_i$.
    CDF$_i$ = sum.

Initialize the histogram bins:
For $i = 1, \ldots, N$ (number of possible outcomes):
    $\text{counts}_i = 0$.

*Generate states and collect data:*
For $i = 1, \ldots, M$ (number of trials):
    $U$ = uniform random on $[0, 1)$.
    For $j = 1, \ldots, N$:
        If $U < \text{CDF}_j$:
            Increment $\text{counts}_j$ by 1; go the next iteration of the $i$ loop.

*Scale the histogram bins down to sample proportions:*
For $i = 1, \ldots, N$ (number of possible outcomes):
    Divide $\text{counts}_i$ by $M$.
```

Example results are as follows, with $M = 10000$ trials:

$\boldsymbol{\omega}$	$P(\boldsymbol{\omega})$	$Q(\boldsymbol{\omega})$	Difference
(1 1 1)	0.3879017	0.3832000	0.0047017
(1 1 -1)	0.0524968	0.0480000	0.0044968
(1 -1 1)	0.0071047	0.0070000	0.0001047
(1 -1 -1)	0.0524968	0.0506000	0.0018968
(-1 1 1)	0.0524968	0.0557000	-0.0032032
(-1 1 -1)	0.0071047	0.0063000	0.0008047
(-1 -1 1)	0.0524968	0.0522000	0.0002968
(-1 -1 -1)	0.3879017	0.3970000	-0.0090983

The limitation of this CDF-sampling algorithm is a bit subtle: it requires computing probabilities for all N states. In the coin-flip example, $N = 2^n$. For, say, $n = 100$, $2^n \approx 10^{69}$ which is (as I recall) on the order of the number of protons in the universe. One cannot even write down all possible states. Yet 100 coins isn't a lot so it seems like we should be able to do *something*.

There is even worse news. Computing the partition function Z requires a sum over all N states, which is also infeasible. Not only can we not list the probability of all states, we can't even compute the (normalized) probability of even *one* state, since (from equations 3.2 and 3.3) $P(\boldsymbol{\omega}) = P^*(\boldsymbol{\omega})/Z$. It would be nice if we had something which only depended on the probability numerator $P^*(\boldsymbol{\omega}) = e^{-\beta E(\boldsymbol{\omega})}$, which is easy to compute.

We will see in the following sections that the Metropolis-Hastings algorithm overcomes both these problems. First we will see what that algorithm is and what it does; then, we'll see why it works.

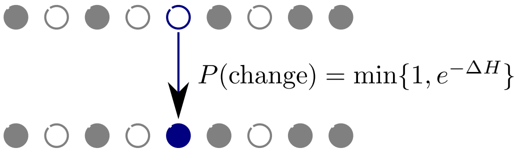
5 Sampling with Metropolis-Hastings

Here we present the essential components of the Metropolis-Hastings algorithm, in pseudocode and a worked example. Important, non-negligible practical considerations are deferred until section 6; correctness is proved in section 8.

5.1 The algorithm

The goal is to select M samples $\boldsymbol{\omega}^{(1)}, \ldots, \boldsymbol{\omega}^{(M)}$ from the PMF of the 1D Ising model with specified S, h, and β. The states have probability numerator $P^*(\boldsymbol{\omega}) = \exp(-\beta E(\boldsymbol{\omega}))$ as given by equation 3.2. Here, I display spins as up (filled) or down (hollow):

- There is an *inverse temperature* β: For physical systems, this has some meaning and, presumably, would be varied under some controlled pattern. For the independent models, β cancels and has no effect (as we'll see in section 6.7) so you can give it any non-zero value, e.g. $\beta = 1$.
- Pick an *initial configuration.* Typically, there are three choices: (1) Start with all spins down, i.e. $\boldsymbol{\omega} = (-1, \ldots, -1)$. (2) Start with all spins up, i.e. $\boldsymbol{\omega} = (+1, \ldots, +1)$. (3) Start with $\boldsymbol{\omega}$ selected from a uniform probability distribution on Ω. (See section 6.2 for more information on choice of initial state.)
- Select a *site* j and decide whether to flip ω_j to $-\omega_j$. (See section 6.3 for more information on site selection.)



- This decision is made using the *Metropolis prescription*, namely:
  - Compute the change in energy $\Delta E = E(\boldsymbol{\omega}') - E(\boldsymbol{\omega})$ which would be obtained if $\boldsymbol{\omega}$ were sent to $\boldsymbol{\omega}'$ by flipping ω_j.
  - One may compute ΔE by separately computing $E(\boldsymbol{\omega}')$ and $E(\boldsymbol{\omega})$ and subtracting the two. However, since the only change is at the site j, one may do some ad-hoc algebra (sections 6.7 and 6.8) to derive an expression for ΔE which is less computationally expensive.
  - Accept the change with probability
    $$\min\{1, e^{-\Delta E}\}.$$
    Otherwise, the change is said to be rejected.

  This is called a *Metropolis step.*
- Looping through all n sites from $j = 1$ to $j = n$, performing a Metropolis step at each site j, is called a *Metropolis sweep.*
- Doing M sweeps is a complete experiment. (See section 6.1 for more information on choice of M.)

- If one realizes a random variable $X(\boldsymbol{\omega})$ at each of M sweeps, averaging X over the M sweeps, one obtains an approximation $\overline{X}$ for the expectation $E[X]$. (See section 5.5 for more information on choice of random variable.)

The above may be rephrased in pseudocode as follows:

```
*Pick the initial state $\boldsymbol{\omega}$:*
For $i = 1, \dots, n$ (number of spins):
    $\omega_i = 0$ (spins-down)
OR
For $i = 1, \dots, n$ (number of spins):
    $\omega_i = 1$ (spins-up)
OR
For $i = 1, \dots, n$ (number of spins):
    $\omega_i = \pm 1$ with probabilities $1/2, 1/2$ (uniform random initial state).

*Generate states and collect data:*
For $i = 1, \dots, M$ (number of trials):
    *One Metropolis sweep:*
    For $j = 1, \dots, n$ (number of spins):
        *One Metropolis step:*
        $\Delta E = E(\boldsymbol{\omega}') - E(\boldsymbol{\omega})$ where $\boldsymbol{\omega}'$ has opposite spin at site $j$.
        Let $V = \min\{1, e^{-\beta \Delta E}\}$.
        Generate $U$ uniformly distributed on $[0, 1)$.
        If $U < V$:
            Accept the new state: $\boldsymbol{\omega} := \boldsymbol{\omega}'$.
        Else:
            Reject the new state: keep $\boldsymbol{\omega}$.
        Find the index $k$ of the state $\boldsymbol{\omega} = (\omega_1, \dots, \omega_n)$ which was just generated.
        Increment counts$_k$ by 1. (Do this whether the new state was accepted or not.)

*Scale the histogram bins down to sample proportions:*
For $k = 1, \dots, N$ (number of possible outcomes):
    Divide counts$_k$ by $M$.
```

Notice that a *step* of this algorithm touches one of the n sites; a *sweep* touches all n sites. We collect data only once per sweep, since only then have all sites been given a chance to change.

This is the basic algorithm. Important caveats, though, are presented in section 6.

5.2 Walk-through

Here are two sweeps with $n = 3$, $\beta = 1$, aligning nearest-neighbor model, initial state $(1, 1, 1)$, and $h = 0.1$. These are the energies and probability numerators:

State number	$\boldsymbol{\omega}$	$E(\boldsymbol{\omega})$	$P^*(\boldsymbol{\omega})$
1	(1 1 1)	-1.7	5.4739474
2	(1 1 -1)	0.1	0.9048374
3	(1 -1 1)	2.1	0.1224564
4	(1 -1 -1)	-0.1	1.1051709
5	(-1 1 1)	0.1	0.9048374
6	(-1 1 -1)	1.9	0.1495686
7	(-1 -1 1)	-0.1	1.1051709
8	(-1 -1 -1)	-2.3	9.9741825

```
Initial state:       ( 1  1  1 )

Sweep 1:
    omega       j  state#  Delta E  U         V         Accept/reject New E
  ------------ --- ------  -------  --------- --------- ------------- -----
  ( 1  1  1 )   1   1      1.8      0.9731406 0.1652989 Reject        -1.7
  ( 1  1  1 )   2   1      3.8      0.3204335 0.0223708 Reject        -1.7
  ( 1  1 -1 )   3   2      1.8      0.1590330 0.1652989 Accept         0.1
State after sweep 1: ( 1  1 -1 )

Sweep 2:
    omega       j  state#  Delta E  U         V         Accept/reject New E
  ------------ --- ------  -------  --------- --------- ------------- -----
  ( 1  1 -1 )   1   2      1.8      0.6781967 0.1652989 Reject         0.1
  ( 1 -1 -1 )   2   4     -0.2      0.9644570 1.0000000 Accept        -0.1
  ( 1 -1 -1 )   3   4      2.2      0.2357908 0.1108032 Reject        -0.1
State after sweep 2: ( 1 -1 -1 )
```

5.3 Numerical results

Example results are as follows, $n = 3$, aligning nearest-neighbor model, $h = 0.1$, and $M = 10000$ trials:

$\boldsymbol{\omega}$	$P(\boldsymbol{\omega})$	$Q(\boldsymbol{\omega})$	Difference
(1 1 1)	0.3879017	0.3875000	0.0004017
(1 1 -1)	0.0524968	0.0514000	0.0010968
(1 -1 1)	0.0071047	0.0066000	0.0005047
(1 -1 -1)	0.0524968	0.0523000	0.0001968
(-1 1 1)	0.0524968	0.0530000	-0.0005032
(-1 1 -1)	0.0071047	0.0070000	0.0001047
(-1 -1 1)	0.0524968	0.0523000	0.0001968
(-1 -1 -1)	0.3879017	0.3899000	-0.0019983

5.4 Heuristics

Recall from section 5.1 that the Metropolis transition probability is

$$\min\{1.0, \exp(-\beta\Delta_E)\}.$$

So, the Metropolis algorithm certainly transitions to states with non-negative $\exp(-\beta\Delta E)$, and randomly transitions to states with negative $\exp(-\beta\Delta E)$. The certain transition occurs when

$$\begin{aligned}\exp(-\beta\Delta E) &\geq 1\\ -\beta\Delta E &\geq \log(1) = 0\\ \beta\Delta E &\leq 0\\ \Delta E &\leq 0.\end{aligned}$$

Thus, we may also say that the Metropolis algorithm certainly transitions to lower-energy or same-energy states, and randomly transitions to higher-energy states.

Certain transition	$\exp(-\beta\Delta E) \geq 1$	$\Delta E \leq 0$
Random transition	$\exp(-\beta\Delta E) < 1$	$\Delta E > 0$

5.5 Random variables for 1D Ising

Here, the RVs are counts Q_i as described in section 4.1. This is just for didactic purposes. Often people look at *sample means* (averages) of some quantity, taken at each sample. One can also define energy, total magnetization, correlations between specified pairs of sites, etc.

6 Design choices and caveats for Metropolis-Hastings

For clarity, I presented the basic algorithm in section 5.1. Here are additional practical considerations which must be addressed in any serious use of Metropolis-Hastings. Correctness of the algorithm is proved in section 8.

6.1 Number of trials

Choice of the number of iterations M: How to do this wisely is a story in itself. For this paper, I suggest trying a few runs with each M, for varying M's. For example, do half a dozen runs with $M = 10000$, another half-dozen runs with $M = 20000$, etc. Then choose an M which reduces the standard deviation of the sample mean to your desired number of decimal places.

6.2 Initial state

Choice of the initial state is as described in section 5.1: typically lowest-energy/highest-energy (e.g. here the all-down/all-up states), or random configuration. Results should be independent of initial state. Failure to achieve this may be due to inadequate thermalization (section 6.4).

6.3 Sequential vs. random sweeps

The decision here is whether to sweep through all sites from (say) left to right, or to move through them randomly. There is some debate in the literature on this topic (for which I lack citations here).

6.4 Thermalization

One should first run L Metropolis sweeps of the system, discarding any data collected, before running the M sweeps in which data are accumulated. The L sweeps are called the *thermalization phase*; the M sweeps are called the *accumulation phase*.

xxx clarify with numerical/graphical examples (describe in terms of P plots in section xxx): The underlying concern is the convergence of the Metropolis probability distribution to the stable distribution of its implicit Markov chain.

Put up E plots?

According to Tom Kennedy, there is no general method to determine whether the system has thermalized. xxx discuss E-plot idea (if it works out ...); manual approaches.

6.5 Separation of states

xxx highly annp — the distribution is supported on all-spins-up and all-spins-down. The Metro one-at-a-time method will always give rejects. Generalize, and cite some further reading.

6.6 Downsampling to avoid correlation

Downsampling and batched means are discussed thoroughly in appendix B of [Kerl2010]. The discussion there also includes information about Berg's autocorrelation detection.

6.7 ΔE computations for the independent model

As noted in section 5.1, it is usually possible to compute ΔE at less computational cost than $E(\boldsymbol{\omega}') - E(\boldsymbol{\omega})$. That is, computing $E(\boldsymbol{\omega})$ requires visiting all n sites of a spin configuration, as does computing $E(\boldsymbol{\omega}')$, but since the Metropolis algorithm only modifies *one* spin, we can skip most of the redundant values which will only cancel out anyway.

When you use a Metropolis-Hastings on a new problem, you will likely be doing some calculations of the type shown here.

xxx note β shouldn't affect probabilities — it shouldn't matter whether we're flipping hot coins or cold ones. then do the algebra and show why this intuition is right.

xxx independence from temperature.

Using equations 3.1 and 2.3,

$$\begin{aligned}
P(\boldsymbol{\omega}) &\propto e^{-\beta E(\boldsymbol{\omega})} \\
&xxx \\
E(\boldsymbol{\omega}) &= -\frac{1}{\beta}\log(P(\boldsymbol{\omega})) \\
&= -\frac{1}{\beta}\log\left(\prod_{i=1}^{n} p_i^{1-\eta_i}(1-p_i)^{\eta_i}\right) \\
&= -\frac{1}{\beta}\sum_{i=1}^{n}\left((1-\eta_i)\log(p_i) + \eta_i\log(1-p_i)\right).
\end{aligned}$$

xxx flip only at jth slot: $\omega_j' = -\omega_j$ and so $\eta_j' = 1 - \eta_j$:

$$\begin{aligned}
\Delta E &= -\frac{1}{\beta}\left(\eta_j\log(p_j) + (1-\eta_j)\log(1-p_j) - (1-\eta_j)\log(p_j) - \eta_j\log(1-p_j)\right) \\
&= -\frac{1}{\beta}\left((1-2\eta_j)\log(1-p_j) - (1-2\eta_j)\log(p_j)\right) \\
&= -\frac{1}{\beta}(1-2\eta_j)\log\left(\frac{1-p_j}{p_j}\right) \\
&= \frac{\omega_j}{\beta}\log\left(\frac{p_j}{1-p_j}\right).
\end{aligned}$$

xxx (or perhaps just state that the β's cancel):

$$\exp\left(-\beta\Delta E\right) = \exp\left(-\omega_j\log\left(\frac{p_j}{1-p_j}\right)\right) = \left(\frac{1-p_j}{p_j}\right)^{\omega_j}.$$

xxx certain-transition case is

$$\left(\frac{1-p_j}{p_j}\right)^{\omega_j} \geq 1.$$

Thus, when $\omega_j = 1$, there is a certain transition when $p_j \leq 1/2$; when $\omega_j = 1$, there is a certain transition when $p_j \geq 1/2$. Since p was $P(\omega_j = 1)$ — the probability of heads at the jth slot — the Metropolis algorithm certainly sends individual spins to their more likely states, and randomly sends them to their less likely states.

6.8 ΔE computations for the 1D Ising model

As discussed at the beginning of the previous section, and as mentioned in section 5.1, we can compute ΔE at less computational cost than $E(\boldsymbol{\omega}') - E(\boldsymbol{\omega})$.

The current state is $\boldsymbol{\omega}$ and the energy is

$$E = -\sum_{ij} S_{ij}\omega_i\omega_j + \sum_i h_i\omega_i.$$

The candidate new state $\boldsymbol{\omega}'$ has opposite spin ω_k. For example, with $n = 4$, $k = 2$, and

$$\boldsymbol{\omega} = (+1, +1, +1, +1),$$

we have (numbering states from 0 up to $n - 1$):

$$\boldsymbol{\omega}' = (+1, +1, -1, +1).$$

Then the new energy is

$$E' = -\sum_{ij} S_{ij}\omega'_i\omega'_j + \sum_i h_i\omega'_i$$

and the energy change is

$$\begin{aligned}\Delta E &= E' - E \\ &= -\sum_{ij} S_{ij}\omega'_i\omega'_j + \sum_{ij} S_{ij}\omega_i\omega_j + \sum_i h_i(\omega'_i - \omega_i).\end{aligned}$$

Now, in my computer program I'm going to be computing this energy difference very many times, so I want to omit as much redundant computation as possible. For $i \neq k$ we have $\omega'_i = \omega_i$, and for $i = k$ we have $\omega'_k = -\omega_k$. So the last term is

$$h_k(\omega'_k - \omega_k) = -2h_k\omega_k.$$

The sums split into four pieces: (1) $i \neq k$ and $j \neq k$, (2) $i = k$ and $j \neq k$, (3) $i \neq k$ and $j = k$, and (4) $i = k$ and $j = k$.

The first piece is

$$-\sum_{i,j\neq k} S_{ij}\omega'_i\omega'_j + \sum_{i,j\neq k} S_{ij}\omega_i\omega_j = -\sum_{i,j\neq k} S_{ij}\omega_i\omega_j + \sum_{i,j\neq k} S_{ij}\omega_i\omega_j = 0.$$

The second piece is

$$-\sum_{j\neq k} S_{kj}\omega'_k\omega'_j + \sum_{j\neq k} S_{kj}\omega_k\omega_j = \sum_{j\neq k} S_{kj}\omega_k\omega_j + \sum_{j\neq k} S_{kj}\omega_k\omega_j = 2\sum_{j\neq k} S_{kj}\omega_k\omega_j.$$

The third piece is

$$-\sum_{i\neq k} S_{ik}\omega'_i\omega'_k + \sum_{i\neq k} S_{ik}\omega_i\omega_k = 2\sum_{i\neq k} S_{ik}\omega_i\omega_k.$$

The fourth piece is

$$-S_{kk}\omega'_k\omega'_k + S_{kk}\omega_k\omega_k = -S_{kk}\omega_k\omega_k + S_{kk}\omega_k\omega_k = 0.$$

Thus the energy difference is

$$2\left(\sum_{j\neq k} S_{kj}\omega_k\omega_j + \sum_{i\neq k} S_{ik}\omega_i\omega_k - h_k\omega_k\right).$$

7 Markov chains

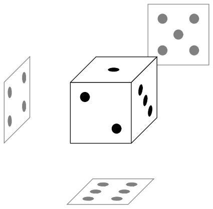
In order to prove the correctness of Metropolis-Hastings, we first need to understand Markov chains. We get Markov chains by putting together two concepts: **stochastic processes** and **conditional probability**. I preface discussion of Markov chains with the **die-tipping example** which helps motivate the discussion.

Throughout, I assume elementary probability terminology as in [Kerl, Lawler, GS]. Namely: sample space; probability measure; probability mass function (PMF); probability density function (PDF); random variable.

7.1 The die-tipping experiments

Most of this document uses coin-flipping and the 1D Ising model as a running example. Here, though, I describe two even simpler experiments (experiment A and experiment B), which involve tipping a single die. As simple as these are, they encapsulate all the central points I want to make about **Markov chains**. I'll return to coin flips and the Ising model in section xxx.

First recall that the pips on opposite faces of a die add to 7:



Experiment A: I set the die on the table with the one-face up. Then I close my eyes momentarily and tip it, so the one-face is now on the side. I then close my eyes and tip it again, and so on, until I've tipped it 20 times.

If I repeat this experiment — placing the side and doing the 20 tips — many times, what's the probability distribution for the initial state, before tipping? I always start with 1 on top, so we have

$$\mathbf{P}^{(0)} = (1, 0, 0, 0, 0, 0).$$

(The six-tuple notation is the probability of 1-face up, 2-face up, and so on. The superscript on the **P** indicates how many tips have occurred.)

What are the possibilities for the new top face after the first tip? The 1-face was tipped over, and the 6-face was on the bottom so it can't be on top. But 2, 3, 4, and 5 are equally likely. So the probability distribution is

$$\mathbf{P}^{(1)} = \left(0, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, 0\right).$$

Note in particular that the up-face after each tip is not independent of the up-face before that tip: **given** that the current up-face is, say, 3, there are **transition probabilities** for what the next up-face will be.

What about $\mathbf{P}^{(2)}$, $\mathbf{P}^{(3)}$, and so on? It's an elementary, but somewhat tedious, exercise in cases to show that

$$\mathbf{P}^{(2)} = \left(\frac{1}{4}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{4}\right).$$

One can continue this pattern, but one of my goals is to make an advertisement for the computational advantages of Markov matrices. We'll find out in section 7.5 how to easily compute $\mathbf{P}^{(k)}$.

Even before computing that, though, I certainly expect that after many tips, the die won't remember that the starting position was with the 1-face up. So, $\mathbf{P}^{(k)}$ ought to begin to look uniform.

Experiment B: The only difference here is that instead of starting with the 1-face up, I instead first roll the die. Then the initial distribution is

$$\mathbf{P}^{(0)} = \left(\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}\right).$$

Again, you can go through the cases and find that

$$\mathbf{P}^{(1)} = \left(\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}\right).$$

and so on. Again, Markov matrices in section 7.5 will make this easier.

In experiment A, the probability distribution started out supported only on 1-up but (we expect) begins to approach uniform; in experiment B, the probability distribution is already uniform and stays that way.

Key points:

- We have a choice of **initial state** for the die-tipping experiment: for example, one-up, or roll first, or others we could have used.
- We have a **probability distribution $\mathbf{P}^{(0)}$** for the initial state.
- We have probability distributions $\mathbf{P}^{(k)}$ for the **subsequent states**. These distributions evolve in k.
- We have **transition probabilities** constraining which face can appear on each step of the experiment. These connect the probability distributions $\mathbf{P}^{(k)}$ and $\mathbf{P}^{(k+1)}$. In section 7.5 we will describe the evolution and the transition probabilities in terms of **Markov chains**.
- Some distributions evolve, or converge, toward a **limiting distribution**; this is called **mixing** or **thermalization**. The limiting distribution (here, uniform) doesn't evolve. This limiting distribution is called a **stationary distribution** for the experiment.
- For this experiment, it looks various distributions evolve toward the uniform distribution. We'll see in section xxx how, and when, this happens.

7.2 Stochastic processes

Definition 7.1. Given a probability space Ω and a sequence $P^{(k)}$ of probability measures for $k = 0, 1, 2, \ldots$, a **discrete-time stochastic process** is a sequence of random variables[1] $X^{(k)}$ from Ω to a **state space** S.

For concreteness, think of Ω as all $N = 2^n$ heads/tails configuration of n coin flips, or all $N = 6$ results of the landing of a single die, where we allow the probability distributions $P^{(k)}$ to vary in time.

The state space might be another copy of Ω, or $\mathbb{R}$, etc. For example, $X^{(k)} : \Omega \to \mathbb{R}$ might count the number of heads in a set of n coin flips. Or, $W^{(k)} : \Omega \to \Omega$ might be the identity function.

(I should point out that many, if not most, authors define a stochastic process with one "long Ω", e.g. the probablity space is all *sequences* of n-coin-flip experiments.)

We categorize stochastic processes by their inputs and outputs:

[1] For the measure-theoretically inclined reader, $X^{(k)}$ is a measurable function from $(\Omega, \mathcal{F}, P^{(k)})$ to $(S, \mathcal{G})$ where $\mathcal{F}$ and $\mathcal{G}$ are sigma algebras on Ω and S, respectively.

- If $k = 0, 1, 2, \ldots$, as above, then we have a **discrete-time stochastic process**, consisting of countably many random variables $X^{(k)}$. If $k \in [0, +\infty)$, then we have a **continuous-time stochastic process**, consisting of uncountably many random variables $X^{(k)}$.
- If each $X^{(k)}$ has only finitely or countably infinitely many possible output values, then we have a **discrete-valued stochastic process**, and the distribution of $X^{(k)}$ is a PMF. If each $X^{(k)}$ is a continuous random variable, then we have a **continuous-valued stochastic process**, and the distribution of $X^{(k)}$ is a PDF.

For this paper, I am considering discrete-time, discrete-valued stochastic processes; Brownian motion is an example of a continuous-time, continuous-valued stochastic process.

7.3 Conditional probability

Forgetting for just a moment about stochastic process, consider two discrete random variables X and Y with joint PMF

$$f_{X,Y}(x, y) = P(X = x, Y = y).$$

We can sum over x's or y's to obtain the marginal distributions of X and Y:

$$P(X = x_i) = \sum_j P(Y = y_j, X = x_i) \qquad \text{and} \qquad P(Y = y_j) = \sum_i P(Y = y_j, X = x_i).$$

Recall the definition of conditional probability:

$$P(Y = y_j) = \frac{P(Y = y_j, X = x_i)}{P(X = x_i)} \quad (\text{if } P(X = x_i) \neq 0, \text{ otherwise } 0).$$

Multiply both sides by the denominator and sum over i:

$$P(Y = y_j, X = x_i) = P(Y = y_j \mid X = x_i) P(X = x_i).$$

Then we have

$$P(Y = y_j) = \sum_i P(Y = y_j \mid X = x_i) P(X = x_i). \tag{7.1}$$

For example, take X to be the starting up-face of the die in experiment A or B of section 7.1, and take Y to be the up-face after the first tip. Then the intuition is that equation 7.1 is the following. How likely is it that Y will be in state j? Add up all the likelihoods of all previous states i, weighted by the probability that state i transitions to state j.

In fact, if X and Y take finitely many values, say n each, we can write down an even nicer equation. Take $n = 2$ for example. Then equation 7.1, for $j = 1, 2$ simultaneously, is just a matrix product:

$$(P(Y = y_1), P(Y = y_2)) = (P(X = x_1), P(X = x_2)) \begin{pmatrix} P(Y = y_1 \mid X = x_1) & P(Y = y_1 \mid X = x_1) \\ P(Y = y_1 \mid X = x_1) & P(Y = y_1 \mid X = x_1) \end{pmatrix}$$

For shorthand, write

$$P_i = P(X = x_i), \qquad P'_j = P(Y = y_j), \qquad \text{and} \qquad M_{ij} = P(Y = y_j \mid X = x_i).$$

Then equation 7.1 is

$$P'_j = \sum_{i=1}^{n} P_i M_{ij}. \tag{7.2}$$

or, more compactly,

$$\mathbf{P}' = \mathbf{P}M. \tag{7.3}$$

This matrix M of transition probabilities is called a stochastic matrix, which means the following.

Definition 7.2. A **stochastic matrix** has all entries between 0 and 1, and all row sums equal to 1.

The reason we choose the row-sum condition is that, given $X = x_i$, it is certain that Y has to go to some y_j:

$$\sum_{j=1}^{n} M_{ij} = \sum_{j=1}^{n} P(Y = y_j \mid X = x_i) = 1.$$

7.4 Markov chains

Remember that a (discrete-time) stochastic process is just a sequence of random variables on the same probability space Ω. Those random variables can be related or unrelated. We are particularly interested in the case when the $(k+1)$st random variable depends only on the previous one.

Definition 7.3. A **Markov chain** is a stochastic process $X^{(k)}$ such that

$$P(X_{n+1} = x \mid X_n = x_n, \ldots, X_1 = x_1) = P(X_{n+1} = x \mid X_n = x_n).$$

The intuition is that a Markov process has (at most) only one level of memory. The die-tipping example of section 7.1 is a Markov process. If the up-face at the 19th step is a 2, then the up-face at the 20th step can be 1, 3, 4, or 6. But it doesn't matter at all *how* the 19th face got to be a 2.

(I said "at most" since a sequence of independent random variables is also a Markov process: for example, replace the die-tipping experiment with a die-rolling experiment.)

(An example of a non-Markov process is a modified die-tipping experiment wherein I tip to a new face that avoids the previous one. E.g. if the 10th face is a 3 and the 9th face was a 2, I would choose the 11th face with equal likelihood from 1, 2, and 6.)

To apply conditional probability to Markov processes, simply replace X and Y of the previous section with sequence of $W^{(k)}$ which are identity functions from $\Omega \to \Omega$. The $W^{(k)}$ is simply the random variable which is the state (of the coin flips, die-tips, etc.) at the kth time step. Also replace the $\mathbf{P}$'s of the previous section with a sequence of $\mathbf{P}^{(k)}$'s. Then we have probablity measures

$$P^{(k+1)}(\{\omega^{(j)}\}) \qquad \text{and} \qquad P^{(k)}(\{\omega^{(i)}\}).$$

on the sample space Ω, we have transition probabilities

$$M_{ij}^{(k,k+1)} = P(W_{k+1} = \omega^{(j)} \mid W^{(k)} = \omega^{(i)}),$$

and the PMFs at each successive time step are related by matrix multiplication: just as in equation 7.3:

$$P_j^{(k+1)} = \sum_i P_i^{(k)} M_{ij}^{(k,k+1)}$$

i.e.

$$\mathbf{P}^{(k+1)} = \mathbf{P}^{(k)} M^{(k,k+1)}. \tag{7.4}$$

If additionally (as is the case in this paper) the transition probabilities are the same for each k, then we have a **time-homogeneous Markov chain**. Then the successive PMFs are related by

$$\mathbf{P}^{(k+1)} = \mathbf{P}^{(k)} M \tag{7.5}$$

and so

$$\mathbf{P}^{k} = \mathbf{P}^{(0)} M^{k}. \tag{7.6}$$

Thus, it will be interesting below [xxx] to see what happens to powers M^k of the Markov matrix M.

For our purposes, think of a Markov chain as a fixed matrix M along with with the sequence $\mathbf{P}^{(k)}$ of PMFs, remembering that there's only the choice of $\mathbf{P}^{(0)}$ after which point subsequent PMFs are obtained by matrix multiplication.

When implementing such things in software (the electronics don't have the error-correcting intuition your brain does), it's especially important to get the i's and j's straight:

- $M_{ij} = P(W_{k+1} = \omega^{(j)} \mid W^{(k)} = \omega^{(i)})$.
- $M_{ij} = P(\omega^{(i)} \mapsto \omega^{(j)})$.
- When looking for the transition probability *from* state i *to* state j, look in row i and column j of M.

7.5 Die-tipping revisited

Now that we know about the evolution of PMFs using Markov matrices, it's much easier to compute the PMFs for the die-tipping experiments, as promised in section 7.1. In both experiments, the transition probabilities are

$$M = \begin{pmatrix} 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\ \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & 0 & 0 & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & 0 & 0 & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} \\ 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \end{pmatrix}$$

E.g. if the 1-face is up, afte the die-tip the up-face can't be 1 or 6 but it can be any of the other four with equal likelihood.

In experiment A, the initial PMF was

$$\mathbf{P}^{(0)} = (1, 0, 0, 0, 0, 0).$$

Then we can just turn the crank and compute

k	$\mathbf{P}^{(k)}$					
0	1.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
1	0.0000000	0.2500000	0.2500000	0.2500000	0.2500000	0.0000000
2	0.2500000	0.1250000	0.1250000	0.1250000	0.1250000	0.2500000
3	0.1250000	0.1875000	0.1875000	0.1875000	0.1875000	0.1250000
4	0.1875000	0.1562500	0.1562500	0.1562500	0.1562500	0.1875000
5	0.1562500	0.1718750	0.1718750	0.1718750	0.1718750	0.1562500
6	0.1718750	0.1640625	0.1640625	0.1640625	0.1640625	0.1718750
7	0.1640625	0.1679688	0.1679688	0.1679688	0.1679688	0.1640625
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
19	0.1666660	0.1666670	0.1666670	0.1666670	0.1666670	0.1666660
20	0.1666670	0.1666665	0.1666665	0.1666665	0.1666665	0.1666670

xxx comment, and plots.

Likewise, for experiment B,

k	$\mathbf{P}^{(k)}$					
0	0.1666667	0.1666667	0.1666667	0.1666667	0.1666667	0.1666667
1	0.1666667	0.1666667	0.1666667	0.1666667	0.1666667	0.1666667
2	0.1666667	0.1666667	0.1666667	0.1666667	0.1666667	0.1666667
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

xxx compute higher powers of M.

7.6 Markov-chain properties

These are discussed thoroughly in chapter 4 of [Kerl2010]:

- *irreducibility*
- *aperiodicity*
- *ergodicity*
- *reversibility*
- *stationarity*

Numerical examples from handwritten notes.

xxx plot them with and without transposing — informative!

xxx note that this is ideal for detecting thermalization, but for large N, writing down the entire PMF is precisely what we cannot do. Hence the art of thermalization detection.

8 Markov chain Monte Carlo

8.1 Metropolis choice of Markov matrix

Given P, find M. Various possibilities; "good" convergence rate.

Def the particular M.

Prove it satisfies DB by construction.

Prove irr.

Prove aper. Not true for uniform P_0! Need something else there ....

Note high correlation of $X^{(k)}$'s.

8.2 Sampling theory

Thm: Sample mean over chain converges to true mean, when M is finite, irr, and aper.

Prove this.

$c/\sqrt{n}$ convergence rate.

9 Generalizations

xxx to do. Citations.

9.1 2D Ising

9.2 Non-example: percolation

9.3 Numerical integration

xxx continuous here . . . .

xxx emph large number of variables; Riemann mesh impossible.

References

[GS] Grimmett, G. and Stirzaker, D. *Probability and Random Processes*, 3rd ed. Oxford, 2001.

[Huang] Huang, K. *Introduction to Statistical Physics.* CRC Press, 2001.

[Kerl] Kerl, J. *Probability notes.*
http://math.arizona.edu/˜kerl/doc/prb.pdf

[Kerl2010] Kerl, J. *Critical behavior for the model of random spatial permutations* (doctoral dissertation)
http://math.arizona.edu/˜kerl/rcm/kerl-dis-gc-ssp.pdf

[Lawler] Lawler, G. *Introduction to Stochastic Processes* (2nd ed.). Chapman & Hall/CRC, 2006.

[Mackay] MacKay, D.J.C. Introduction to Monte Carlo Methods.
http://www.cs.toronto.edu/˜mackay/p0.html#BayesMC.html

[NR] Press, W. et al. *Numerical Recipes* (2nd ed.). Cambridge, 1992.

[Wik] Articles on *Ising Model*, *Statistical Ensemble*, and *Statistical Mechanics.*
http://en.wikipedia.org.

Index