Department of Statistics and Actuarial Science
University of Hong Kong

Tutorial 8: MCMC - Real Examples

ZHANG Yan

October 30, 2021

Overview

Random Disks

Random Disks

Problem formulation - a visualization

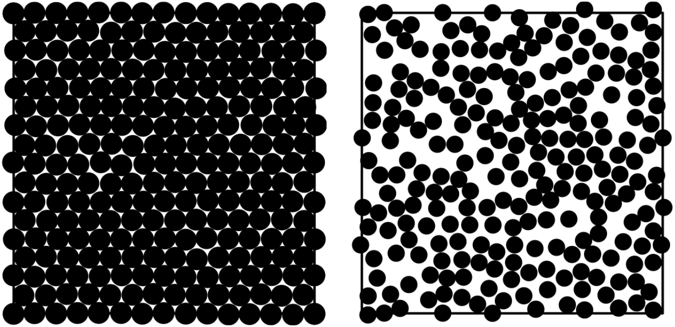


Figure 11.1: The left panel shows $N = 224$ disks of diameter about 0.0692 closely packed into the unit square. The boundary of the square is visible in a few places. The boundary is periodic, and so a disk that intersects an edge is plotted twice, and a disk intersecting the corner is plotted four times. The right panel shows 224 disks of diameter about 0.0536.

- The problem that motivated Metropolis to invent MCMC.

Random Disks
Problem formulation - assumptions and goals

Assumptions:

- **224 equally large circular disks** are packed into a **unit square**.
- **Wraparound boundary** (better approximation to an enormous system).
- **No overlap** (disks are not independent).

Goals:

- **Draw samples**, where each sample is the 224 disks randomly distributed on the square.
- Do inference based on samples, like the **distribution of distance** from each disk to its nearest neighbor.
- Consider **different sizes** of disks.

Difficulties:

- A **448 dimensional** sampling problem.
- Sampling sequentially (choosing centers one at a time and never placing one overlapping with a previous point), but we may have **no room to place the late points**.

Solution:

- Do random perturbations sequentially based on MCMC (a technique called the **Metropolis within Gibbs**).

Random Disks

Sampling strategy - the target distribution

- Assume the diameter $d_0 = d_{\max}(1 - 2^{v-8})$, $0 \le v \le 7$, $d_{\max} = 1/14$. Each sample is a 224×2 matrix $\boldsymbol{x} = [x_{jk}]$, $j = 1, \ldots, 224$, $k = 1, 2$.

- Define the **distance in the unit square**:

$$d((x_{j1}, x_{j2}), (x_{j'1}, x_{j'2})) = \sqrt{d_W(x_{j1}, x_{j'1})^2 + d_W(x_{j2}, x_{j'2})^2},$$

  where d_W is the **wraparound distance**

$$d_W(x_{jk}, x_{j'k}) = \frac{1}{2} - \left||x_{jk} - x_{j'k}| - \frac{1}{2}\right|.$$

- The (unnormalized) **target distribution** is

$$\pi_u(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \min_{j,j'} d((x_{j1}, x_{j2}), (x_{j'1}, x_{j'2})) \ge d_0, \\ 0 & \text{otherwise.} \end{cases}$$

Random Disks

Sampling strategy - the Metropolis within Gibbs

- Initialize the simulation at $\boldsymbol{x}_0$, which has the disks centered on a **grid ensuring no overlap**.
- Given $\boldsymbol{x}$, for the disk j:

$$x_j' = (x_{j1}, x_{j2}) + \mathcal{U}([-\alpha, \alpha]^2), \alpha = d_{\max} - d_0,$$

  and denote $\boldsymbol{x}'$ as the old $\boldsymbol{x}$ with row j replaced by x_j'.
- The **M-H acceptance probability** is

$$\min(1, \frac{\pi_u(\boldsymbol{x}')}{\pi_u(\boldsymbol{x})}) = \min(1, \pi_u(\boldsymbol{x}')) = \pi_u(\boldsymbol{x}').$$

  So, we accept this move if there is no overlap in $\boldsymbol{x}'$.
- The 224 updates complete one iteration of this sampler, called the **Metropolis within Gibbs**.

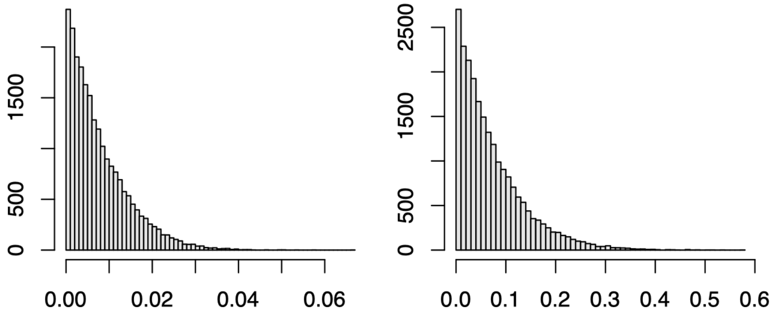
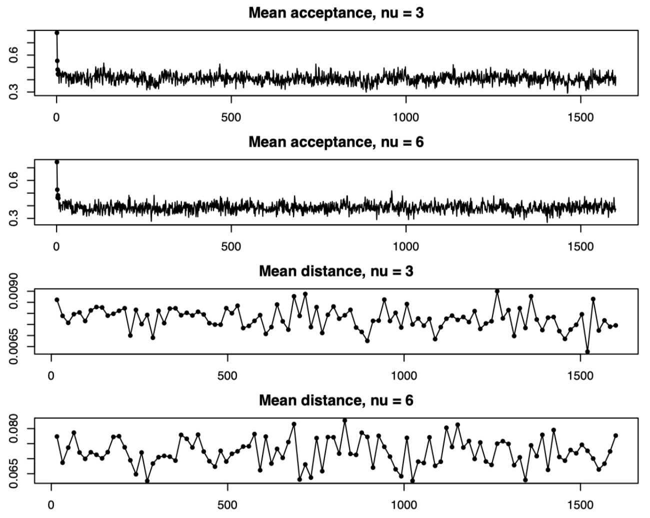


Figure 11.8: This figure shows histograms of the distance from each disk to its nearest neighbor, as a multiple of the disk diameter. On the left the diameters are given by (11.28) with $\nu = 3$ and on the right $\nu = 6$.

- 1600 iterations with thinning parameter 16.
- 22400 distances in each histogram.

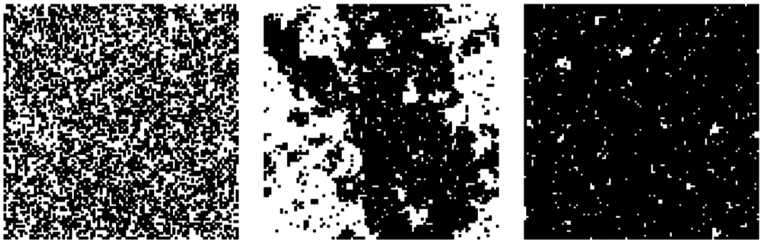
Random Disks
Simulation results - trajectories



Ising Model

Ising Model

Problem formulation - a visualization



T = 8.0 T = 2.269 T = 2.0

Figure 11.2: The Ising model with $J = 1$ and $B = 0$, sampled at 3 temperatures T on a 100×100 grid, with periodic boundary conditions. The left panel is roughly half black but some renderings make it look like a higher fraction black.

- An example from physics.

- At each of $N = 100^2$ **grid points** there is a dichotomous variable with charge **1 or -1**, or equivalently we have $\boldsymbol{x} \in \{-1, 1\}^{100^2}$.
- Define the **Hamiltonian energy** function

$$H(\boldsymbol{x}) = -J \sum_{j \sim k} x_j x_k - B \sum_{j=1}^{N} x_j,$$

  where $\sum_{j \sim k} x_j x_k$ is the number of neighbor pairs with matching signs minus the number that differ. Assume **wraparound**.
- Based on the **Boltzmann's law**:

$$\pi(\boldsymbol{x}) \propto \exp(-\frac{H(\boldsymbol{x})}{T}),$$

  where T is the called the **temperature**.

The energy function:

$$H(\boldsymbol{x}) = -J \sum_{j \sim k} x_j x_k - B \sum_{j=1}^{N} x_j.$$

- When $J > 0$, matching neighbors result in lower energy $H(\boldsymbol{x})$ thereby raising the probability $\pi(\boldsymbol{x})$ and mismatched neighbors have the opposite effect. This is known as the **ferromagnetic case**.
- If $J < 0$ then we have the **antiferromagnetic case** and the neighbors tend to differ from each other.
- If $B \neq 0$ then the model is **biased** towards more charges of the same sign as B.

The probability function:

$$\pi(\boldsymbol{x}) \propto \exp(-\frac{H(\boldsymbol{x})}{T}) = \exp(-H(\boldsymbol{x}))^{1/T}.$$

- About T, in a very **hot system**, the Ising model is nearly a uniform distribution.
- In a very **cold system**, the Ising model puts almost all of its probability on states that achieve the minimum value of $H(\boldsymbol{x})$, which are called the **ground states**. (annealing)
- Actually, the interesting temperatures are intermediate.

- In the **random scan Gibbs sampler**, the component to update is chosen at random.
- To draw samples from the Ising Model, given $\boldsymbol{x}$, we randomly picking a j and **change** x_j **to** $-x_j$, and denote $\boldsymbol{x}'$ as the old $\boldsymbol{x}$ with the jth element changed.
- The **M-H acceptance probability** is

$$\begin{aligned}\min(1, \frac{\pi(\boldsymbol{x}')}{\pi(\boldsymbol{x})}) &= \min(1, \exp(\frac{H(\boldsymbol{x}) - H(\boldsymbol{x}')}{T})) \\ &= \min(1, \exp(-\frac{2x_j(J\sum_{k:k\sim j} x_k + B)}{T})).\end{aligned}$$

- One **sweep** corresponds to $N = 100^2$ updates.

- The strategy to use the **Metropolis within Gibbs** in the discrete setting in this way is called the **Metropolized Gibbs sampler**.

- If we directly use the Gibbs here. The probability to move would be smaller:

$$\frac{\pi(\boldsymbol{x}')}{\pi(\boldsymbol{x}') + \pi(\boldsymbol{x})} \leq \frac{\pi(\boldsymbol{x}')}{\max(\pi(\boldsymbol{x}'), \pi(\boldsymbol{x}))} = \min(1, \frac{\pi(\boldsymbol{x}')}{\pi(\boldsymbol{x})}).$$

Ising Model

Simulation results - trajectories

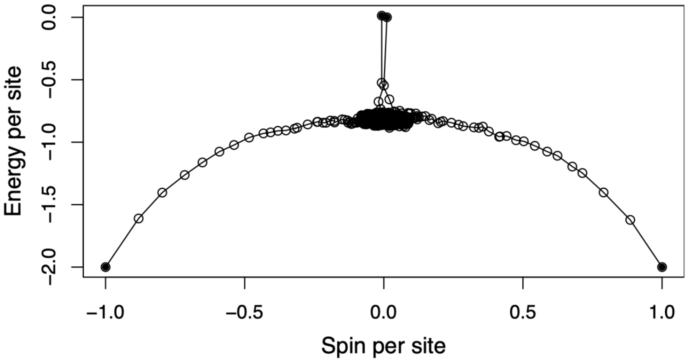


Figure 11.10: Mean energy versus mean spin for the Ising model with $J = 1$ and $B = 0$, temperature $T = 8.0$ on a 100×100 grid. Four trajectories of 500 sweeps are shown as described in the text. The starting points are solid.

- The **mean spin** is $\sum_{j=1}^{N} x_j/N$.
- The **mean energy** is $H(\boldsymbol{x})/N$.

Ising Model

Simulation results - the critical temperature

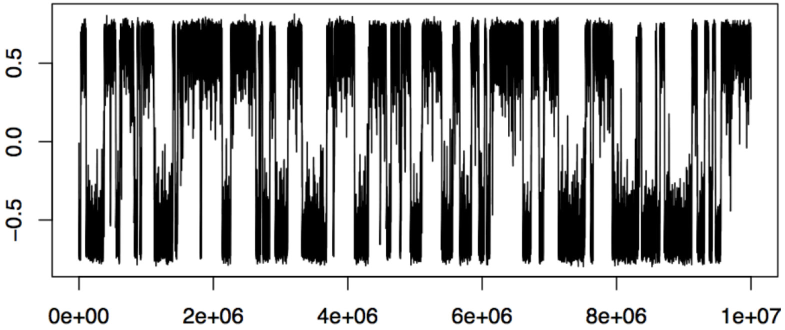


Figure 11.12: This figure shows the mean spin per site in $\boldsymbol{x}_i$, versus the simulation index i, after every 200'th sweep, for the Ising simulation at the critical temperature $T_c = 2.269$.

- The **critical temperature** is $T_c = 2.269$.

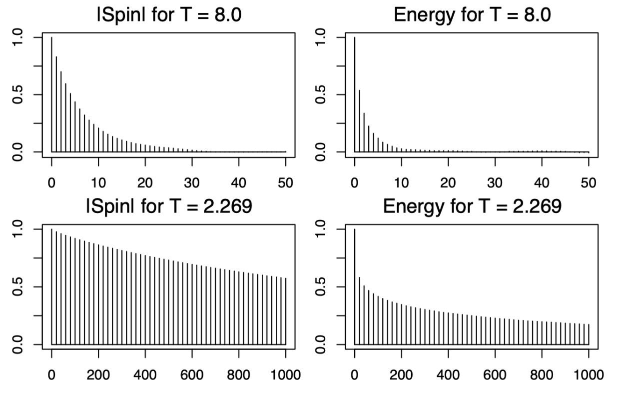


Figure 11.11: Autocorrelation functions for the Ising model at temperatures 8 and $T_c = 2.269$. The ACFs for mean absolute spin are on the left and the ACFs for mean energy are on the right. The lags for $T = 8$ go up to 50 while those for $T = T_c$ go up to 1000 in steps of 20.

- The **absolute mean spin** is $|\sum_{j=1}^{N} x_j/N|$.

Slice Sampler

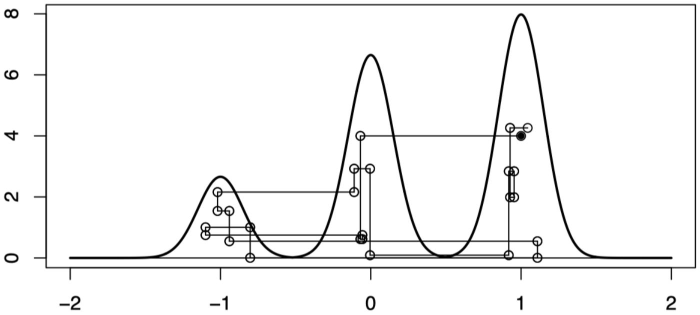


Figure 12.7: This figure illustrates the slice sampler. The unnormalized density π_u is given by the thick line. The slice sampler starts at the solid point $(1, 4)$ and from there executes 25 steps of the Gibbs sampler for the uniform distribution under π_u over the interval $[-2, 2]$.

- An important variation of the Gibbs sampler.

The **full conditional distributions**:

- To draw samples from $\pi(\boldsymbol{x})$, $\boldsymbol{x} \in \mathbb{R}^d$, we define the region $\mathcal{R} = \{(\boldsymbol{x}, z) : 0 \leq z \leq \pi(\boldsymbol{x})\}$. If $(\boldsymbol{X}, Z) \sim \mathcal{U}(\mathcal{R})$, $\boldsymbol{X} \sim \pi(\boldsymbol{x})$.
- Given $\boldsymbol{x}$, the full conditional distribution of Z is $\mathcal{U}(\mathcal{R}(\boldsymbol{x}))$, where $\mathcal{R}(\boldsymbol{x}) = \{z : 0 \leq z \leq \pi(\boldsymbol{x})\}$ is an one dimensional set.
- Given $\boldsymbol{x}_{-j}$ and z, the full conditional distribution of X_j is $\mathcal{U}(\mathcal{R}(\boldsymbol{x}_{-j}, z))$, where $\mathcal{R}(\boldsymbol{x}_{-j}, z) = \{x_j : z \leq \pi(\boldsymbol{x})\}$.

The algorithm:

- So, the **slice sampler** is simply to draw samples by the Gibbs sampler with these one dimensional uniform distributions.
- To draw one sample from $\mathcal{U}(\mathcal{R}(\boldsymbol{x}_{-j}, z))$, one naive method is to repeatedly draw $\mathcal{U}([L, R])$ for small L and big R and accept the first one inside $\mathcal{R}(\boldsymbol{x}_{-j}, z)$, but it is **too wasteful**.
- Instead, we will consider a **two stage strategy**.

The **stepping out procedure** (for the jth substep):

- Given x_j and $\mathcal{R}(\boldsymbol{x}_{-j}, z)$.
- Set hyperparameters W (the **initial estimated slice size**) and M (the **limit on iterations**).
- Do initialization: $L = x_j - \mathcal{U}([0, W])$, $R = L + W$; $J = \lfloor \mathcal{U}([0, M]) \rfloor$, $K = M - 1 - J$.
- While $J > 0$ or $L \in \mathcal{R}(\boldsymbol{x}_{-j}, z)$: $L = L - W$, $J = J - 1$.
- While $K > 0$ or $R \in \mathcal{R}(\boldsymbol{x}_{-j}, z)$: $R = R + W$, $K = K - 1$.
- Return $[L, R]$.

The **shrinkage sampling procedure** (for the jth substep):

- Given x_j, $\mathcal{R}(\boldsymbol{x}_{-j}, z)$ and $[L, R]$.
- Loop:
  - $x_j' = \mathcal{U}([L, R])$.
  - If $x_j' \in \mathcal{R}(\boldsymbol{x}_{-j}, z)$: Return x_j'.
  - If $x_j' < x_j$: $L = x_j'$.
  - If $x_j' > x_j$: $R = x_j'$.

Some comments:

- If the hyperparameters are carefully tuned, combining these two stages will result in **detailed balance.**
- The biggest practical issue is to pick the **width** W **for different** j.

References

- Owen, A. B. (2013). Monte Carlo theory, methods and examples.
- Neal, R. M. (2003). Slice sampling. The annals of statistics, 31(3), 705-767.

Thanks!