MCV172, HW#3

Oren Freifeld

May 6, 2017

Contents

1 Gibbs Sampling in the Ising Model

We will test the Gibbs sampler by computing expectations and comparing with the exact results (which we obtained using dynamic programming) from HW#2. We will also pretend that the Ising model is a good model for images, and study image restoration from Ising-model samples corrupted by noise.

Remark 1 ***Before you start coding*** *a single line in this assignment, do yourself a favor and* ***read the entire document****.* $\diamond$

Reminder: Gibbs Sampling from the Ising-Model Prior

Recall the Ising-model prior, $p(x)$, is

$$p(x) \propto \exp\left(\frac{1}{\text{Temp}}\sum_{s\sim t} x_s x_t\right) \qquad \text{Temp} > 0\,, \tag{1}$$

and that

$$\begin{aligned} p(x_s|_s x) &= \frac{p(x_s, {}_s x)}{p({}_s x)} = \frac{p(x)}{p({}_s x)} = \frac{\frac{1}{Z}\exp\left(\frac{1}{\text{Temp}}\sum_{s\sim t} x_s x_t\right)}{\sum_{x'_s}\frac{1}{Z}\exp\left(\frac{1}{\text{Temp}}\sum_{s\sim t} x'_s x_t\right)} \\ &= \frac{\exp\left(\frac{1}{\text{Temp}}\sum_{t:t\in\eta_s} x_s x_t\right)}{\sum_{x'_s}\exp\left(\frac{1}{\text{Temp}}\sum_{t:t\in\eta_s} x'_s x_t\right)} = \frac{\exp\left(\frac{1}{\text{Temp}}\sum_{t:t\in\eta_s} x_s x_t\right)}{\exp\left(-\frac{1}{\text{Temp}}\sum_{t:t\in\eta_s} x_t\right) + \exp\left(\frac{1}{\text{Temp}}\sum_{t:t\in\eta_s} x_t\right)}\,. \end{aligned} \tag{2}$$

Thus,

$$p(x_s|_s x) = p(x_s|x_{\eta_s}) \tag{3}$$

and

$$\begin{cases} p(x_s = 1|_s x) = \frac{\exp\left(\frac{1}{\text{Temp}} \sum_{t:t\in\eta_s} x_t\right)}{\exp\left(-\frac{1}{\text{Temp}} \sum_{t:t\in\eta_s} x_t\right)+\exp\left(\frac{1}{\text{Temp}} \sum_{t:t\in\eta_s} x_t\right)} \propto \exp\left(\frac{1}{\text{Temp}} \sum_{t:t\in\eta_s} x_t\right) \\ p(x_s = -1|_s x) = \frac{\exp\left(-\frac{1}{\text{Temp}} \sum_{t:t\in\eta_s} x_t\right)}{\exp\left(-\frac{1}{\text{Temp}} \sum_{t:t\in\eta_s} x_t\right)+\exp\left(\frac{1}{\text{Temp}} \sum_{t:t\in\eta_s} x_t\right)} \propto \exp\left(-\frac{1}{\text{Temp}} \sum_{t:t\in\eta_s} x_t\right) \end{cases} . \tag{4}$$

So sampling from this conditional distribution is just flipping a coin, where the bias on the coin is affected by the states of the neighbors of x_s.

Reminder: Gibbs Sampling from the Ising-Model Posterior, Assuming Gaussian IID Additive Noise

Under this assumption, letting x denote the sample from the Ising-model prior and y denote the noise,

$$p(x|y) \propto \exp\left(\frac{1}{\text{Temp}}\left(\sum_{s\sim t} x_s x_t\right) - \frac{1}{2\sigma^2}\sum_s (y_s - x_s)^2\right) \tag{5}$$

and

$$\begin{aligned} p(x_s|_s x, y) &= \frac{p(x_s, {}_s x|y)}{p({}_s x|y)} = \frac{p(x|y)}{p({}_s x|y)} \\ &= \frac{\exp\left(\frac{1}{\text{Temp}}\left(\sum_{s\sim t} x_s x_t\right) - \frac{1}{2\sigma^2}\sum_s (y_s - x_s)^2\right)}{\sum_{x'_s} \exp\left(\frac{1}{\text{Temp}}\left(\sum_{s\sim t} x'_s x_t\right) - \frac{1}{2\sigma^2}\sum_s (y_s - x'_s)^2\right)} \\ &\propto \exp\left(\frac{1}{\text{Temp}}\left(\sum_{t:t\in\eta_s} x_s x_t\right) - \frac{1}{2\sigma^2}(y_s - x_s)^2\right) \end{aligned} \tag{6}$$

Thus,

$$p(x_s|_s x, y) = p(x_s|x_{\eta_s}, y) = p(x_s|x_{\eta_s}, y_s) \tag{7}$$

and

$$p(x_s = 1|_s x, y) \propto \exp\left(\frac{1}{\text{Temp}}\left(\sum_{t:t\in\eta_s} x_t\right) - \frac{1}{2\sigma^2}(y_s - 1)^2\right) \tag{8}$$

$$p(x_s = -1|_s x, y) \propto \exp\left(\frac{1}{\text{Temp}}\left(-\sum_{t:t\in\eta_s} x_t\right) - \frac{1}{2\sigma^2}(y_s + 1)^2\right) . \tag{9}$$

Again, the sampling is nothing more than flipping a coin, where here the bias on the coin is affected by both (i) the states of the neighbors of x_s and (ii) y_s, the measurement associated with x_s.

1.1 Estimation: Comparisons with the Nearly-exact Values

In HW #2, we used exact sampling to estimate

$$E_{\text{Temp}}(X_{(1,1)}X_{(2,2)}) \text{ and } E_{\text{Temp}}(X_{(1,1)}X_{(8,8)})$$

through their empirical expectations,

$$\widehat{E}_{\text{Temp}}(X_{(1,1)}X_{(2,2)}) \triangleq \frac{1}{10000}\sum_{n=1}^{10000} x_{(1,1)}(n)x_{(2,2)}(n) \tag{10}$$

$$\widehat{E}_{\text{Temp}}(X_{(1,1)}X_{(8,8)}) \triangleq \frac{1}{10000}\sum_{n=1}^{10000} x_{(1,1)}(n)x_{(8,8)}(n) \tag{11}$$

where $x(1), x(2), \ldots, x(10000)$ were 10,000 exact samples from the Ising model. We refer to these expectations as "nearly-exact" for the following reason: the term "exact" stems from the fact the samples were obtained using exact sampling, while the "nearly" qualifier is added since an empirical expectation, *i.e.*, the average over the samples (*i.e.*, AKA the sample mean) is a random quantity which, by the Law of Large Numbers, converges to the true expectation as the number of samples tends to infinity.

Now, in the current HW assignment, instead of using exact sampling, we will produce the samples using MCMC, particularly the Gibbs-sampling method.

Computer Exercise 1 *Build a Gibbs sampler for the* 8×8 *Ising model at temperatures* $\text{Temp} \in \{1, 1.5, 2\}$. *Compute the empirical expectations of*

$$E_{\text{Temp}}(X_{(1,1)}X_{(2,2)}) \text{ and } E_{\text{Temp}}(X_{(1,1)}X_{(8,8)}) \tag{12}$$

using two different methods; see below. Compare the estimates to the nearly-exact values computed in the previous HW assignment; see the table below.
Programming Note: *A convenient way to handle boundaries is to embed the* $n \times n$ *lattice in the interior of an* $(n+2) \times (n+2)$ *lattice whose boundary values are set to zero, and then visit only the* $n \times n$ *interior sites of the larger lattice.* $\diamondsuit$

Method 1. **Independent Samples.** At each temperature, draw 10,000 (approximated) samples from the Ising-model prior,

$$p(x) \propto \exp\left(\frac{1}{\text{Temp}}\sum_{s\sim t} x_s x_t\right), \tag{13}$$

using the Gibbs sampler. For each sample, initiate the Gibbs sampler at a random configuration, and update sites in deterministic, raster-scan, order (fixed site-visitation schedule). The sample is the obtained configuration after 25 passes through the entire graph (*i.e.*, after 25 "sweeps" – where each sweep involves $8 \times 8 = 64$ single-site updates). Putting it differently, you are asked to run 10,000 such Markov Chains, where each chain has 25×64 iterations. Let the obtained 10,000 (approximated) samples be denoted by $x(1), \ldots, x(10000)$. Your estimates are then

$$\frac{1}{10000}\sum_{n=1}^{10000} x_{(1,1)}(n)x_{(2,2)}(n) \text{ and } \frac{1}{10000}\sum_{n=1}^{10000} x_{(1,1)}(n)x_{(8,8)}(n)\,. \tag{14}$$

Method 2. **Ergodicity.** Beginning with a random configuration, run the Gibbs sampler for 25,000 sweeps of the lattice (*i.e.*, a single Markov Chain

temperature	$\hat{E}_{\text{Temp}}(X_{(1,1)}X_{(2,2)})$	$\hat{E}_{\text{Temp}}(X_{(1,1)}X_{(8,8)})$
1	0.95	0.9
1.5	0.77	0.55
2	your result from HW2	your result from HW2

Table 1: Expectations estimated from exact samples using dynamic programming

of 25,000 × 64 iterations). Use the empirical averages of $x_{(1,1)}x_{(2,2)}$ and $x_{(1,1)}x_{(8,8)}$ over all but the first 100 sweeps (*i.e.*, following a so-called "burn-in period" of 100 sweeps, use the 24,900 configurations obtained at the end of each of the remaining sweeps to compute the empirical averages).

Remark 2 *Note that for computing the average of, say, such 24,900 quantities, you do not need to store 24,900 values in memory. In effect, suppose you have a sequence of N values, denoted by $z_1, z_2, \dots, z_N$ and you want to compute $\frac{1}{N}\sum_{n=1}^{N} z_i$. Then you can discard each z_i once you iteratively update the current estimate of the mean:*

$$\mu^{[1]} = z_1 \tag{15}$$

$$\mu^{[n]} = \frac{1}{n}\sum_{m=1}^{n} z_m = \frac{1}{n}\left(\left(\sum_{m=1}^{n-1} z_m\right) + z_n\right) \tag{16}$$

$$= \frac{1}{n}\left(\frac{n-1}{n-1}\left(\sum_{m=1}^{n-1} z_m\right) + z_n\right) = \frac{(n-1)\mu^{[n-1]} + z_n}{n} \qquad n \in 2, 3, \dots, N\,. \tag{17}$$

◊

Exercise 1 *Compare the results obtained from the two methods, and compare the results to the values from exact sampling. Speculate about the reasons for any substantial discrepancies.* ◊

1.2 Image Restoration

These experiments are to be performed on a 100×100 lattice.

Computer Exercise 2 *At each of the three temperatures* Temp $\in \{1, 1.5, 2\}$*:*

1. *Using Gibbs sampling, generate a single* 100×100 *sample, x, from the* 100×100 *Ising model using 50 raster-scan sweeps.*

2. *Add to the sample an array of* 100×100 *Gaussian noise values, sampled IID from $\mathcal{N}(0, 2^2 = 4)$. We denote this noise array by η. Here is one way to achieve this in Python:*

```
eta = 2*np.random.standard_normal(size=(100,100))
y = x + eta
```

3. *Using Gibbs sampling, generate a sample from the posterior distribution of the uncorrupted sample given the corrupted sample,*

$$x \sim p(x|y)\,, \tag{18}$$

again using 50 sweeps.

4. *For comparison, compute the "ICM" (Iterated Conditional Mode) restoration: visit sites, in raster order, replacing at each site the current spin by its most-likely value (of the two possibilities, ± 1), under the posterior distribution and conditioned on the four neighbors:*

$$x_s^{new} = \underset{x_s \in \{-1,1\}}{\arg\max}\; p(x_s|_s x, y) \tag{19}$$

where

$$p(x_s|_s x, y) = p(x_s|x_{\eta_s}, y) = p(x_s|x_{\eta_s}, y_s)\,. \tag{20}$$

This "greedy" algorithm will converge within a few raster cycles.

5. *Display all four images (from the previous 4 steps) on a single plot. (see the subplot and imshow commands).* ◊