

Lecture 41: The Gibbs Sampler

Charles A. Bouman

Purdue University

Friday, December 4, 2026

Outline

- 1 The Gibbs sampler as a special case of coordinatewise Hastings–Metropolis
- 2 The Gibbs sampler algorithm
- 3 The conditional distribution of a pixel in a Gibbs distribution
- 4 The proposal distribution $g_s(x_s|x_{\partial s})$
- 5 The acceptance probability is always $\alpha = 1$
- 6 The Ising model case
- 7 A 16×16 Ising simulation
- 8 Convergence: guaranteed, but sometimes extremely slow

Core idea

Use the pixel's own conditional distribution as the proposal. Then the acceptance probability is always one, every proposal is accepted, and sampling reduces to replacing each pixel in sequence with a draw from its conditional distribution given its neighbors.

Today's Big Ideas

- The Gibbs sampler is the coordinatewise Hastings–Metropolis sampler whose proposal for a pixel is that pixel's conditional distribution given its neighbors.
- Under a Gibbs distribution, that conditional distribution is built from only the cliques that contain the pixel, so it is cheap to evaluate.
- Because the proposal is the conditional distribution, every proposal is accepted, and sampling becomes the sequential replacement of each pixel by a draw from its conditional distribution.
- For the Ising model each pixel is redrawn from a distribution that counts how many of its neighbors disagree with it, and a simulation above the critical temperature settles into one of the two modes.
- The Gibbs sampler is guaranteed to converge, but its convergence can be extremely slow.

The Gibbs sampler

The **Gibbs sampler** is an important special case of the coordinatewise Hastings–Metropolis sampler that was first proposed by Geman and Geman [1984].

- It has since become widely used due to its **ease of implementation** and **relatively fast convergence for simple distributions**.
- With the Gibbs sampler, the proposal distribution is taken to be the **conditional distribution of the pixel X_s given its neighbors** under the Gibbs distribution.

That is, the update of pixel s is generated as

$$X_s^{(k+1)} \sim p_s(x_s | X_{\partial s}^{(k)}) .$$

The Gibbs sampler algorithm

Gibbs Sampler Algorithm:

Initialize $X^{(0)}$

Select ordering of pixels, $\{s_i\}_{i=0}^{N-1}$ // Can use raster order

Initialize $k \leftarrow 0$

For $l = 0$ to $L - 1$ {

For $i = 0$ to $N - 1$ {

// Generate update for s_i^{th} pixel

$X^{(k+1)} \leftarrow X^{(k)}$ // Initialize update

$W_{s_i} \sim g_{s_i}(w_{s_i} | X_{\partial s_i}^{(k)})$ // Generate sample from conditional distribution

$X_{s_i}^{(k+1)} \leftarrow W_{s_i}$ // Replace pixel with generated sample

$k \leftarrow k + 1$

}}

What the algorithm box says

The Gibbs sampler algorithm is a special case of the coordinatewise Hastings–Metropolis algorithm when the proposal distribution for each pixel is just its conditional distribution given its neighbors, that is, $X_s^{(k+1)} \sim p_s(x_s | X_{\partial s}^{(k)})$. In this case, the acceptance probability is always $\alpha = 1$, so all proposals are accepted.

- Compare with last lecture's coordinatewise Hastings–Metropolis pseudocode: there is no accept/reject step here at all.
- The inner loop does two things: draw W_s from the conditional distribution, and write it into the image.
- L complete iterations through the image; each full iteration performs a single update of each pixel.

Setting: X is an MRF with a Gibbs distribution

Assume that X is an MRF with a Gibbs distribution of the form

$$\begin{aligned} p(x) &= \frac{1}{z} \exp \{-u(x)\} \\ &= \frac{1}{z} \exp \left\{ - \sum_{c \in \mathcal{C}} V_c(x_c) \right\} , \end{aligned}$$

where $u(x)$ is the energy function and $V_c(x_c)$ are the potential functions for clique c .

Remark (book footnote)

Without loss of generality, X can be assumed to be an MRF because we can always take $\partial s = \{r : r \neq s\}$.

The conditional distribution of a pixel

Define $\mathcal{C}_s$ to be the subset of all cliques containing x_s (Chapter 14). Then we know that the conditional distribution of a pixel, x_s , given all other pixels has the form

$$P\{X_s = x_s | X_r \text{ } r \neq s\} = \frac{\exp\{-u_s(x_s | x_{\partial s})\}}{\sum_{x_s \in \Omega} \exp\{-u_s(x_s | x_{\partial s})\}},$$

where

$$u_s(x_s | x_{\partial s}) = \sum_{c \in \mathcal{C}_s} V_c(x_c).$$

- From Chapter 14: this came out of the constructive proof that a Gibbs distribution corresponds to an MRF. The cliques not containing s cancel between numerator and denominator.

The proposal distribution

So for a Gibbs sampler, the key idea is to use the conditional distribution of a pixel as the proposal distribution.

More specifically, the proposal distribution is given by

$$\begin{aligned} g_s(x_s | x_{\partial s}) &= P\{X_s = x_s | X_i = x_i \text{ } i \neq s\} \\ &= \frac{\exp\{-u_s(x_s | x_{\partial s})\}}{\sum_{x'_s \in \Omega} \exp\{-u_s(x'_s | x_{\partial s})\}}. \end{aligned}$$

- The sum in the denominator runs over the $|\Omega|$ values of a single pixel, so this distribution is cheap to evaluate.

The acceptance probability

In this case, the acceptance probability can be computed in closed form to be

$$\begin{aligned}\alpha &= \min \left\{ 1, \frac{g_s(X_s^{(k)} | W_{\partial s})}{g_s(W_s | X_{\partial s}^{(k)})} \exp \left\{ -[u(W) - u(X^{(k)})] \right\} \right\} \\ &= \min \left\{ 1, \frac{g_s(X_s^{(k)} | W_{\partial s})}{g_s(W_s | X_{\partial s}^{(k)})} \frac{\exp \left\{ -u(W_s | X_{\partial s}^{(k)}) \right\}}{\exp \left\{ -u(X_s^{(k)} | X_{\partial s}^{(k)}) \right\}} \right\} \\ &= \min \{1, 1\} = 1 .\end{aligned}$$

- Line 1 is the coordinatewise Hastings–Metropolis acceptance probability from the previous section.
- Only pixel s changes between W and $X^{(k)}$, so the energy difference reduces to the single-pixel terms.
- The ratio of exponentials is exactly the reciprocal of the ratio of proposal densities, so the product is one.

Consequence: there are no rejections

So the acceptance probability for the Gibbs sampler is always one. Without rejections, the algorithm becomes the simple sequential replacement of pixels from their assumed conditional distribution.

- This is why the pseudocode has no accept/reject step: the proposal W_{s_i} is always written into $X^{(k+1)}$.
- Everything proved last lecture still applies. The Gibbs sampler is a coordinatewise Hastings–Metropolis sampler, so the convergence theorem for that algorithm covers it.

The Ising model case

For example, with the Ising model of Chapter 14, the conditional distribution of a pixel given its neighbors becomes

$$g_s(x_s | x_{\partial s}) = \frac{\exp \left\{ -\beta \sum_{r \in \partial s} \delta(x_s \neq x_r) \right\}}{\sum_{x_s=0}^{M-1} \exp \left\{ -\beta \sum_{r \in \partial s} \delta(x_s \neq x_r) \right\}} .$$

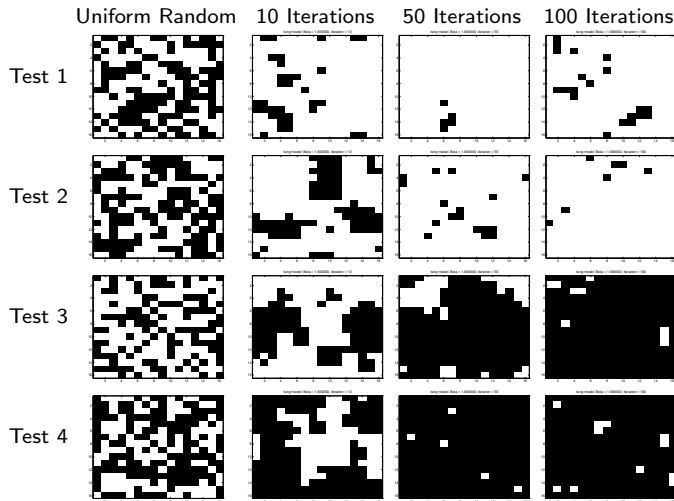
- So for this case, the Gibbs sampler would simply replace each pixel in sequence with a new pixel drawn from this conditional distribution.
- Evaluating it requires only counting how many neighbors disagree with each candidate value of x_s .

Why accepting everything helps

- An important advantage of the Gibbs sampler is that, since all proposals are accepted, it **tends to converge faster**.
- Intuitively, each pixel is replaced with a sample having the desired distribution.
- So it is quite reasonable that repeated updating of pixels with this “correct” distribution should converge to the entire random field having the correct distribution.

Contrast with the Metropolis sampler of the previous lecture, where a proposal drawn from some other distribution may be rejected and the pixel left unchanged.

A 16×16 Ising simulation



A Gibbs sampler simulation of an Ising model with inverse temperature parameter of $\beta = 1.0$ for a 16×16 binary array using circular boundary conditions.

Reading the simulation

- The distribution sampled is the conditional distribution above, but with a **four-point neighborhood** and $M = 2$.
- Each row represents an independently run experiment using a different seed for the pseudorandom number generator, with an independent initial condition of i.i.d. binary random variables with equal probability of 1 (white) and -1 (black).
- The columns are the results after 0, 10, 50, and 100 full iterations through the image.

What to notice

After a large number of iterations, the result becomes either mostly all 1's or mostly all -1 's. This happens because β is above the critical value, so we know that the distribution of the Ising model is bimodally distributed, with one mode consisting of mostly all 1's, and the other mode consisting of mostly all -1 's.

From Lecture 38: this is the critical temperature behavior of the Ising model, now seen in actual samples.

Convergence: guaranteed, but sometimes very slow

For this example, the Gibbs sampler does a good job of generating representative samples from the distribution of the Ising model, **but**:

- As the size of the image grows larger, or as the value of β grows larger, one can easily imagine that the Gibbs sampler can become “trapped” in a deep local maximum of the distribution.
- In fact, while the Gibbs sampler is **guaranteed to converge**, in many practical cases its convergence can be **extremely slow**.
- In some cases, the convergence is so slow that **no practical computer could generate the desired result**.

The book returns to this issue in Chapter 16, which is not covered in this course.

Summary

- The Gibbs sampler is the coordinatewise Hastings–Metropolis sampler whose proposal for pixel s is the conditional distribution of X_s given its neighbors, $X_s^{(k+1)} \sim p_s(x_s | X_{\partial s}^{(k)})$ (Geman and Geman, 1984).
- For a Gibbs distribution $p(x) = \frac{1}{Z} \exp\{-\sum_{c \in \mathcal{C}} V_c(x_c)\}$, that conditional distribution is built from $u_s(x_s | x_{\partial s}) = \sum_{c \in \mathcal{C}_s} V_c(x_c)$.
- With this choice, $\alpha = \min\{1, 1\} = 1$: all proposals are accepted, and the algorithm is the simple sequential replacement of pixels from their conditional distribution.
- Ising model: replace each pixel in sequence from $g_s(x_s | x_{\partial s}) \propto \exp\{-\beta \sum_{r \in \partial s} \delta(x_s \neq x_r)\}$.
- 16×16 simulation at $\beta = 1.0$: after 100 iterations each run is mostly all 1's or mostly all -1 's, because β is above the critical value.
- Convergence is guaranteed, but it can be extremely slow.

Reading: Section 15.5. Next class (Dec. 7): course review and final exam preparation. Final exam: finals week, Dec. 14–18.