Lecture 6: Markov Chain Monte Carlo

Ziyu Shao

School of Information Science and Technology
ShanghaiTech University

April 13 & 15, 2020

Outline

Outline

Monte Carlo Method

- Proposed by the mathematician Stanislaw Ulam in 1946
- Whose uncle regularly gambled at the Monte Carlo casino in Monaco
- Generate random variables & stochastic processes
- Heart of many simulation techniques for increasing complex systems & models

Theory Justification: The Strong Law of Large Numbers

If $Y_1, Y_2, \cdots$ is an i.i.d. sequence with common mean $\mu < \infty$, then the strong law of large numbers says that, with probability 1,

$$\lim_{n \to \infty} \frac{Y_1 + \cdots + Y_n}{n} = \mu.$$

Theory Justification: The Strong Law of Large Numbers

Equivalently, let Y be a random variable with the same distribution as the Y_i and assume that r is a bounded, real-valued function. Then, $r(Y_1), r(Y_2), \cdots$ is also an i.i.d. sequence with finite mean, and, with probability 1,

$$\lim_{n\to\infty} \frac{r(Y_1) + \cdots + r(Y_n)}{n} = E(r(Y)).$$

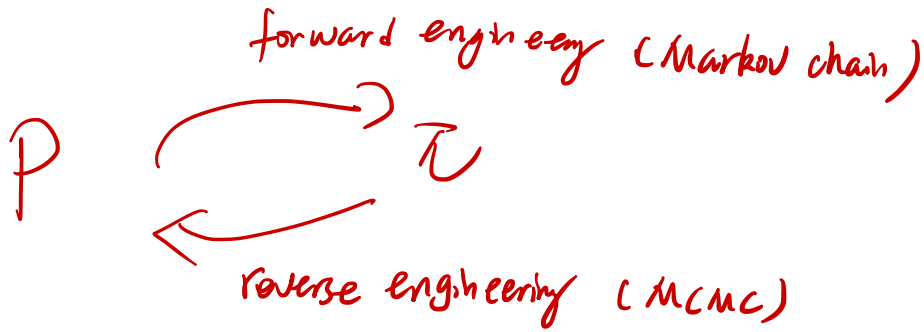
Major Limitation of the Monte Carlo Method

- we need to know how to generate $X_1, X_2, \ldots, X_n$
- Given the universality of the Uniform
- For many distributions such as beta distribution, find the CDF is difficult, let alone its inverse.

Markov Chain Monte Carlo (MCMC)

- Revolutionized statistics and scientific computation
- Expanding the range of possible distributions that we can simulate from, including joint distributions in high dimensions
- Basic idea: build your own Markov chain $(X_0, X_1, \ldots)$ so that the desired distribution π is the stationary distribution of the chain.
- Sampling from distribution π (running the chain for a long time and then sampling)
- Further do sample mean & sample variance & other sample functions
- Connections to optimization

MCMC Method



- Forward engineering: given the transition matrix P, find the stationary distribution of Markov chain.
- Reverse engineering: given a distribution π that we want to simulate, we will engineer a Markov chain whose stationary distribution is π. Then run this engineered Markov chain for a long time, the distribution of the chain will approach π.

MCMC Method

- Markov Chain Monte Carlo (MCMC) is a remarkable methodology, which utilizes Markov sequences to effectively simulate from what would otherwise be intractable distributions.
- All MCMC algorithms construct reversible (time-reversible) Markov chain: detailed balance equations help us.
- Two most widely used algorithms: Metropolis-Hastings & Gibbs Sampling

Theory Justification: Strong Law of Large Numbers for Markov Chains

Theorem

Assume that $X_0, X_1, \cdots$ is an ergodic Markov chain with stationary distribution π. Let r be a bounded, real-valued function. Let X be a random variable with distribution π. Then, with probability 1,

$$\lim_{n\to\infty} \frac{r(X_1) + \cdots + r(X_n)}{n} = E(r(X)).$$

where $E(r(X)) = \sum_j r(j)\pi_j$.

Example: Bob's Lunch

Bob's daily lunch choices at the cafeteria are described by a Markov chain with transition matrix

$$\boldsymbol{P} = \begin{array}{c} \\ \text{Yogurt} \\ \text{Salad} \\ \text{Hamburger} \\ \text{Pizza} \end{array} \begin{array}{c} \begin{array}{cccc} \text{Yogurt} & \text{Salad} & \text{Hamburger} & \text{Pizza} \end{array} \\ \begin{pmatrix} 0 & 0 & 1/2 & 1/2 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 0 & 1/4 & 1/2 \\ 1/4 & 0 & 1/4 & 1/2 \end{pmatrix} \end{array}.$$

Yogurt costs \$3.00, hamburgers cost \$7.00, and salad and pizza cost \$4.00 each. Over the long term, how much, on average, does Bob spend for lunch?

Solution

① X : state of the M.C. $\quad X \in \{$"Yogurt", "Salad", "Hamburger", "Piazza"$\}$

$$r(x) = \begin{cases} 3 & \text{if } X = \text{"Yogurt"} \\ 4 & \text{if } X = \text{"Salad" or "Piazza"} \\ 7 & \text{if } X = \text{"Hamburger"} \end{cases}$$

② The lunch chain is ergodic. with stationary distribution $\quad \pi P = \pi$

	Yogurt	Salad	Hamburger	Piazza
$\pi =$	$\frac{7}{65}$	$\frac{2}{13}$	$\frac{18}{65}$	$\frac{6}{13}$

③ SLLN for Markov chain. w.p.1, Bob's average lunch cost converges to $\sum_x r(x)\pi(x) = 3 \cdot \frac{7}{65} + 4\left(\frac{2}{13} + \frac{6}{13}\right) + 7\left(\frac{18}{65}\right)$

$= \$ 4.32$ per day.

Example: Binary Sequences with No Adjacent 1s

$n = 2$, there are $2^2 = 4$ binary sequences.

00 ✓
01 ✓
10 ✓
11 ×

} good.

good sequence. — state.

{ ↓ } → state space.

We construct an ergodic M.C. over such state space s.t. whose stationary distribution is uniform.

Consider sequences of length m consisting of 0s and 1s. Call a sequence good if it has no adjacent 1s. What is the expected number of 1s in a good sequence if all good sequences are equal likely?

Markov chain. $X_0, X_1, X_2, \ldots$

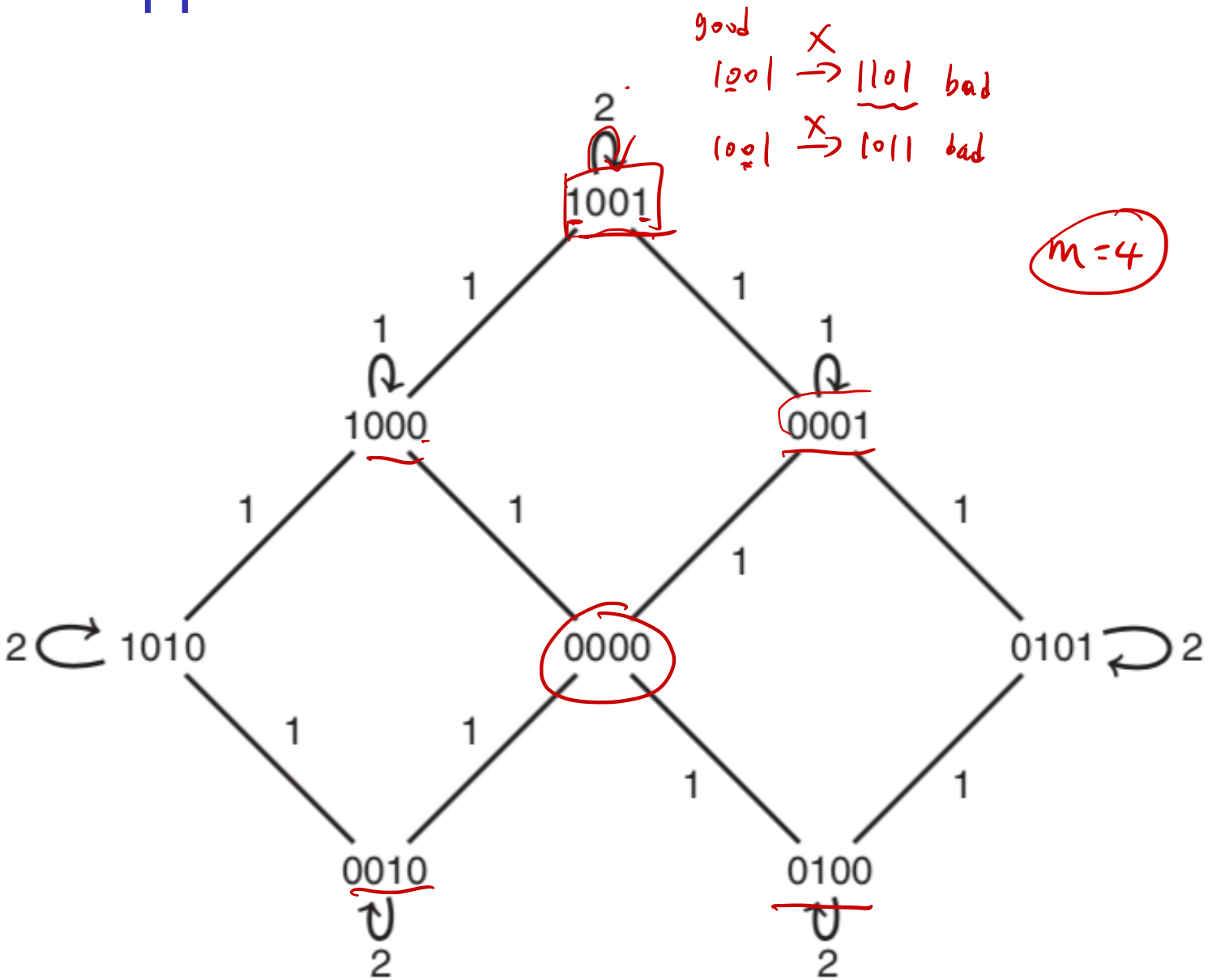
x : good sequence.

$r(x)$: # of "1" in sequence x.

$$\mu \approx \frac{r(x_1) + r(x_2) + \cdots + r(x_n)}{n}$$

$$\mu \overset{\text{a.s.}}{=} \lim_{n \to \infty}$$

MCMC Approach



MCMC Approach

1^0. Markov chain **irreducible**. relay 0000 all "b"s - sequence.

aperiodic.

finite. $(2m)$

$\leq 2m$ steps.

$a \to b$.

$b \to a$

2^0. detailed balance equation.

$$\pi_i p_{ij} = \pi_j p_{ji} \quad i \neq j.$$

$i \neq j$. $p_{ij} = p_{ji} = \frac{1}{m}$.

$\Leftrightarrow \pi_i = \pi_j \quad \not\Leftarrow \pi_i = \frac{1}{C}$, $|C|$: # of all good sequences.

$|$State space$|$

$\Rightarrow$ Uniform distribution over state space

all good sequences.

$m = 100$; $n = 100000$ $\Rightarrow \mu \approx 27.833$

2^{100} binary (1021 good sequences)

Theory. $\mu = 27.7921$

Outline

Basic Idea

- Proposed by Nicholas Metropolis in 1953 & further developed by Wilfred Keith Hastings in 1970.
- Start with any irreducible Markov chain on the state space of interest
- Then modify it into a new Markov chain with desired stationary distribution
- Modification: moves are proposed according to the original chain, but the proposal may or may not accepted.
- Art: choice of the probability of accepting the proposal

Algorithm for DTMC

Let $\pi = (\pi_1, ..., \pi_M)$ be a desired stationary distribution on state space $\{1, ..., M\}$. Assume that $\pi_i > 0$ for all i (if not, just delete any states i with $s_i = 0$ from the state space). Suppose that $P = (p_{ij})$ is the transition matrix for a Markov chain on state space $\{1, ..., M\}$. Intuitively, P is a Markov chain that we know how to run but that doesn't have the desired stationary distribution. Now we will modify P to construct a Markov chain.

- Use the original transition probabilities p_{ij} to propose where to go next
- Then accept the proposal with probability a_{ij}
- Staying in the current state in the event of a rejection.
- Normalization of π does not need to be known.

Algorithm 1 Metropolis-Hastings

Require:

Stationary distribution $\pi = (\pi_1, ..., \pi_M)$;

Original transition matrix $P = (p_{ij})$;

State X_0 (chosen randomly or deterministically);

Ensure:

Modified transition matrix $P' = (p'_{ij})$;

1: **repeat**

2: If $X_n = i$, propose a new state j using the transition probabilities in the ith row of the original transition matrix P;

3: Compute the acceptance probability $a_{ij} = \min\left(\frac{\pi_j p_{ji}}{\pi_i p_{ij}}, 1\right)$;

4: Flip a coin that lands Heads with probability a_{ij};

5: If the coin lands Heads, accept the proposal (i.e., go to j), setting $X_{n+1} = j$. Otherwise, reject the proposal (i.e., stay at i), setting $X_{n+1} = i$;

6: **until** Convergence;

7: **return** P';

Metropolis–Hastings Algorithm

1°. irreducible.

aperiodic

2°. detailed balance equation

$p_{ij}' = p_{ij} a_{ij}$ $\qquad \pi_i p_{ij}' \overset{?}{=} \pi_j p_{ji}'$

Theorem

The sequence $X_0, X_1, \cdots$ constructed by the Metropolis–Hastings algorithm is a reversible Markov chain with stationary distribution π.

$$\pi_i p_{ij}' = \pi_i p_{ij} a_{ij} = \pi_i p_{ij} \min\left(\frac{\pi_j p_{ji}}{\pi_i p_{ij}}, 1\right)$$

$$= \min(\pi_j p_{ji}, \pi_i p_{ij})$$

$$\pi_j p_{ji}' = \pi_j p_{ji} a_{ji} = \min(\pi_i p_{ij}, \pi_j p_{ji})$$

Remarks

- The exact form of π is not necessary to implement Metropolis–Hastings. The algorithm only uses ratios of the form $\frac{\pi_j}{\pi_i}$. Thus, π needs only to be specified up to proportionality.
- If the proposal transition matrix P is symmetric, $a_{ij} = \min\left(\frac{\pi_j}{\pi_i}, 1\right)$.
- The algorithm works for any irreducible proposal chain. Thus, the user has wide latitude to find a proposal chain that is efficient in the context of their problem.
- If the proposal chain is ergodic (irreducible and aperiodic in the finite case), then the resulting Metropolis–Hastings chain is also ergodic with limiting(stationary) distribution π.

Remarks

- The generated sequence $X_0, X_1, \ldots, X_n$ gives an approximate sample from π.
- If the chain requires many steps to get close to stationarity, there may be initial bias.
- Burn-in: the practice of discarding the initial iterations and retaining $X_m, X_{m+1}, \ldots, X_n$, for some m.
- The strong laws of large numbers for Markov chains:

$$\lim_{n\to\infty} \frac{r(X_m) + \cdots + r(X_n)}{n - m + 1} = E(r(X)) = \sum_x r(x)\pi_x.$$

Example: Power-law Distribution

Power-law distributions are positive probability distributions of the form $\pi_i \propto i^S$, for some constant S. Unlike distributions with exponentially decaying tails (e.g., Poisson, geometric, exponential, normal), power-law distributions have *fat tails*, and thus are often used to model skewed data. Let

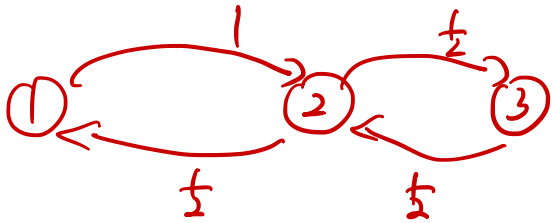
$$\pi_i = \frac{i^{-3/2}}{\sum_{k=1}^{\infty} k^{-3/2}}, \text{ for } i = 1, 2, \cdots$$

Implement a Metropolis–Hastings algorithm to simulate from $\boldsymbol{\pi}$.

Solution

1°. original chain: Proposal chain: Simple Symmetric random walk on the positive integers with reflecting boundary.

State space $\{1, 2, \dots \infty\}$



$$P_{ij} = \begin{cases} \frac{1}{2} & \text{if } j = i \pm 1 \quad i > 2 \\ 1 & \text{if } i = 1, j = 2 \\ 0 & \text{otherwise.} \end{cases}$$

$$\pi_i = \frac{i^{-\frac{3}{2}}}{\sum_{k=1}^{\infty} k^{-\frac{3}{2}}},$$ the acceptance function is

$$a_{ij} = \min\left(\frac{\pi_j p_{ji}}{\pi_i p_{ij}}, 1\right)$$

$$= \min\left(\frac{j^{-\frac{3}{2}} p_{ji}}{i^{-\frac{3}{2}} p_{ij}}, 1\right)$$

Solution

2^0. $a_{12} = \min\left(\frac{\pi_2 p_{21}}{\pi_1 p_{12}}, 1\right) = \min\left(\frac{2^{-\frac{3}{2}} \cdot 1}{1 \cdot 1}, 1\right) = 2^{-\frac{5}{2}}$

$a_{21} = 1 = \min(2^{\frac{5}{2}}, 1)$

3^0. $i, j \geq 2 \Rightarrow a_{i,i+1} = \min\left(\frac{(i+1)^{-\frac{3}{2}}}{i^{-\frac{3}{2}}}, 1\right) = \left(\frac{i}{i+1}\right)^{\frac{3}{2}}$

$a_{i+1,i} = \min\left(\frac{i^{-\frac{3}{2}}}{(i+1)^{-\frac{3}{2}}}, 1\right) = 1$

Solution

TABLE 5.1 Comparison of Markov chain Monte Carlo Estimates with Exact Probabilities for Power-Law Distribution

i	1	2	3	4	5	6	7	8	≥ 9
Simulation	0.389	0.137	0.075	0.048	0.034	0.026	0.021	0.017	0.252
Exact	0.383	0.135	0.074	0.048	0.034	0.026	0.021	0.017	0.262

Example: Zipf Distribution Simulation

Let $M \geq 2$ be an integer. An r.v. X has the *Zipf distribution* with parameter $a > 0$ if its PMF is

$$P(X = k) = \frac{1/k^a}{\sum_{i=1}^{M} (1/j^a)},$$

for $k = 1, 2, ..., M$ (and 0 otherwise). This distribution is widely used in linguistics for studying frequencies of words.
Create a Markov chain $X_0, X_1, ...$ whose stationary distribution is the Zipf distribution, and such that $|X_{n+1} - X_n| \leq 1$ for all n. Your answer should provide a simple, precise description of how each move of the chain is obtained, i.e., how to transition from X_n to X_{n+1} for each n.

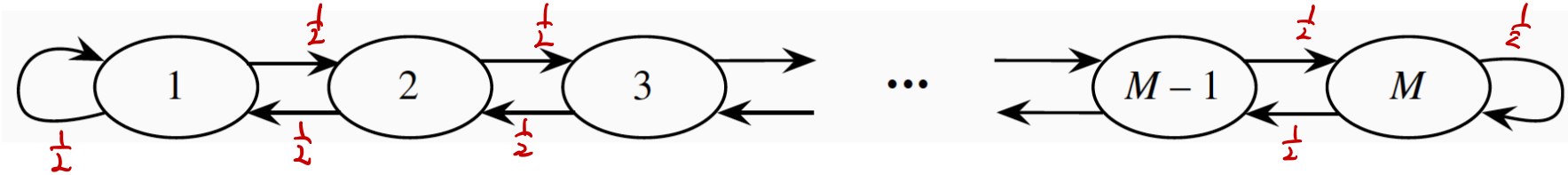
Solution

Proposal chain : State space $\{1, 2, \dots, M\}$.

Simple random walk.

$$p_{ij} = \begin{cases} \frac{1}{2} & \text{if } j = i \pm 1, \ 2 \le i \le M-1 \\ \frac{1}{2} & i = 1, j = 2; \ i = M, j = M-1; \\ \frac{1}{2} & i = 1, j = 1; \ i = M, j = M; \\ 0 & \text{otherwise} \end{cases}$$

$$\begin{cases} \frac{1}{2} & |i-j| \le 1 \\ 0 & \text{otherwise.} \end{cases}$$



$$a_{ij} = \min\left(\frac{\pi_j p_{ji}}{\pi_i p_{ij}}, 1\right) = \min\left(\frac{i^a}{j^a}, 1\right) \qquad |i-j| \le 1$$

$$\Rightarrow \quad a_{i,i+1} = \min\left(\left(\frac{i}{i+1}\right)^a, 1\right) = \left(\frac{i}{i+1}\right)^a, \quad 1 \le i \le M-1$$

$$a_{i,i-1} = \min\left(\left(\frac{i}{i-1}\right)^a, 1\right) = 1, \quad 2 \le i \le M$$

$$a_{1,1} = a_{M,M} = 1$$

Solution

Example: Knapsack Problem

- m treasures with labels from 1 to m, where the jth treasure is worth g_j gold pieces and weighs w_j pounds.
- The maximum weight we can carry is w pounds.
- We must choose a vector $x = (x_1, ..., x_m)$, where x_j is 1 if we choose the jth treasure and 0 otherwise, such that the total weight of the treasures j with $x_j = 1$ is at most w.
- Let C be the space of all such vectors, so C consists of all binary vectors $(x_1, ..., x_m)$ with $\sum_{j=1}^{m} x_j w_j \leq w$.
- We wishes to maximize the total worth of the treasure we takes.
- Finding an optimal solution is an extremely difficult problem, known as the knapsack problem, which has a long history in computer science.
- A brute force solution would be completely infeasible in general. How about MCMC?

Example: Knapsack Problem

(a) Consider the following Markov chain. Start at $(0, 0, ..., 0)$. One move of the chain is as follows. Suppose the current state is $x = (x_1, ..., x_m)$. Choose a uniformly random J in $\{1, 2, ..., m\}$, and obtain y from x by replacing x_J with $1 - x_J$ (i.e., toggle whether that treasure will be taken). If y is not in C, stay at x; if y is in C, move to y. Show that the uniform distribution over C is stationary for this chain.

(b) Show that the chain from (a) is irreducible, and that it may or may not be aperiodic (depending on $w, w_1, ..., w_m$).

Solution

$|\mathcal{C}|$: size of $\mathcal{C}$.

(a). $x \leftrightarrow y$, $P_{x,y} = \frac{1}{m} = P_{y,x}$.

$\pi P = \pi$

$\pi_x = \pi_y = \cdots = \frac{1}{|\mathcal{C}|}$.

uniform is a stationary distribution

(b) Markov chain irreducible.

x, y , $x \longleftrightarrow (0,0,\ldots 0) \longleftrightarrow y$

$x \longrightarrow (0,\ldots 0) \longrightarrow y$

dropping treasures one at a time at most m steps

picking up treasures one at a time. at most m steps

at most $2m$ steps.

$x \longrightarrow y$

Solution

periodicity. $w, w_1, \dots w_m$

1^0. $w_1 + \dots + w_m < w$.

All binary vectors of length-m are allowed.

the period of $(0, \dots 0)$ is 2 (pick up and put down)

2^0. Now we consider $w_1 > w$.

$\forall x \in C$, there is a $\frac{1}{m}$ chance the chain will try to pick up the first treasure., and if that happens, the Markov chain will stay at x.

$P_{x,x} \neq 0$

$\longrightarrow$ period of each state is 1

$\longrightarrow$ aperiodic Markov chain.

Example: Knapsack Problem

(c) The chain from (a) is a useful way to get approximately *uniform* solutions, but Bilbo is more interested in finding solutions where the value (in gold pieces) is high. In this part, the goal is to construct a Markov chain with a stationary distribution that puts much higher probability on any particular high-value solution than on any particular low-value solution. Specifically, suppose that we want to simulate from the distribution

$$s(x) \propto e^{\beta V(x)},$$

where $V(x) = \sum_{j=1}^{m} x_j g_j$ is the value of x in gold pieces and β is a positive constant. The idea behind this distribution is to give exponentially more probability to each high-value solution than to each low-value solution. Create a Markov chain whose stationary distribution is as desired.

Solution

1° desired distribution

$\pi(x) \propto e^{\beta V(x)}$

$\pi(x) = \dfrac{e^{\beta V(x)}}{\sum_{y \in C} e^{\beta V(y)}}, \quad x \in C.$

$V(x) = x \cdot g = \sum_{j=1}^{n} x_j g_j$

2°. Proposal Chain in (a)

acceptance probability

$\alpha_{x,y} = \min\left(\dfrac{\pi_y P_{y,x}}{\pi_x P_{x,y}}, 1\right) = \min\left(\dfrac{e^{\beta V(y)}}{e^{\beta V(x)}}, 1\right)$

$= \min\left(e^{\beta(V(y)-V(x))}, 1\right)$

Solution

(c) We can apply Metropolis-Hastings using the chain from (a) to make proposals. Start at $(0, 0, \ldots, 0)$. Suppose the current state is $x = (x_1, \ldots, x_m)$. Then:

1. Choose a uniformly random J in $\{1, 2, \ldots, m\}$, and obtain y from x by replacing x_J with $1 - x_J$.

2. If y is not in C, stay at x. If y is in C, flip a coin that lands Heads with probability $\min\left(1, e^{\beta(V(y)-V(x))}\right)$. If the coin lands Heads, go to y; otherwise, stay at x.

Solution

$$a_{x,y} = \min\left(e^{\beta(V(y)-V(x))}, 1\right)$$

Comments on β.

$\beta >> 1$, close to optimal π x^*.

exploitation

Slow convergence.

Struck in local optimal.

if x is local optimal, $V(y) < V(x)$.

$\beta \to \infty$.

$a_{x,y} = \min(0, 1) = 0$

$\beta \to 0$.

exploration

far from optimal.

Fast convergence.

$\beta = 0$, $a_{x,y} = \min(1, 1) = 1$

β gradually increase over time

β is small, explore the state space

β is larger, exploitation.

→ best solution

Simulated Annealing.

$\beta = \frac{1}{\text{temperature}}$.

Continuous State Space

- MCMC can also be used in the continuous case, when π is a probability density function.
- For a continuous state space, Markov process a transition *function* replaces the transition matrix.
- P_{ij} is the value of a conditional density function given $X_0 = i$.

Example: Beta Simulation

PDF $f(x) = \frac{1}{B(a,b)} x^{a-1}(1-x)^{b-1}$ $0<x<1$, $a>0, b>0$

Suppose that we want to generate $W \sim \mathrm{Beta}(a, b)$. What we have available are i.i.d. $\mathrm{Unif}(0, 1)$ r.v.s. How can we generate W which is *approximately* $\mathrm{Beta}(a, b)$ if a and b are any positive real numbers, with the help of a Markov chain on the continuous state space $(0, 1)$?

Solution

1^0. Proposal Chain: independent sampler.

independent Unif(0,1) r.v.s.

2^0. If the chain is currently at state w (a real number $\in (0,1)$)

then

(a). generate a proposal u by drawing a Unif(0,1) r.v.

(b) accept the proposal with probability $\min\left(\frac{\pi_u \cdot 1}{\pi_w \cdot 1}, 1\right)$

$$= \min\left(\frac{u^{a-1}(1-u)^{b-1}}{w^{a-1}(1-w)^{b-1}}, 1\right)$$

if the proposal is accepted, go to u.

Otherwise, stay at w.

Solution

Let W_0 be any starting state, and generate a chain $W_0, W_1, \ldots$ as follows. If the chain is currently at state w (a real number in $(0,1)$), then:

1. Generate a proposal u by drawing a Unif$(0,1)$ r.v.

2. Accept the proposal with probability $\min\left(\frac{u^{a-1}(1-u)^{b-1}}{w^{a-1}(1-w)^{b-1}}, 1\right)$. If the proposal is accepted, go to u; otherwise, stay at w.

Example: Normal-Normal Conjugacy

Let $Y|\theta \sim \mathcal{N}(\theta, \sigma^2)$, where σ^2 is known but θ is unknown. Using the Bayesian framework, we treat θ as a random variable, with prior given by $\theta \sim \mathcal{N}(\mu, \tau^2)$ for some known constants μ and τ^2. That is, we have the two-level model

$$\theta \sim \mathcal{N}(\mu, \tau^2)$$
$$Y|\theta \sim \mathcal{N}(\theta, \sigma^2)$$

Describe how to use the Metropolis-Hastings algorithm to find the posterior mean and variance of θ after observing the value of Y.

Solution

1°. $f_{\theta|Y}(\theta|y) \propto f_{Y|\theta}(y|\theta) f_\theta(\theta)$

$\propto e^{-\frac{1}{2\sigma^2}(y-\theta)^2} \cdot e^{-\frac{1}{2\tau^2}(\theta-\mu)^2}$

Normal is the conjugate prior of the normal.

$$\theta|Y=y \sim N\left(\frac{\frac{1}{\sigma^2}}{\frac{1}{\sigma^2}+\frac{1}{\tau^2}} y + \frac{\frac{1}{\tau^2}}{\frac{1}{\sigma^2}+\frac{1}{\tau^2}} \mu, \frac{1}{\frac{1}{\sigma^2}+\frac{1}{\tau^2}}\right)$$

posterior mean ↑ ; ↓ posterior variance.

posterior mean of θ $E(\theta|Y=y)$

(a) $\tau^2 << 1$, $\frac{1}{\tau^2} >> 1$, more weight on prior mean μ.

(b) $\sigma^2 << 1$, $\frac{1}{\sigma^2} >> 1$, ... data y.

Solution

2^0. M-H. Construct M. C. whose stationary distribution $f_{\theta|Y}(\theta|y)$

generates $\theta_0, \theta_1, \ldots$

(a) if $\theta_n = x$, propose a new state x', $x' = x + \varepsilon_n$, $\varepsilon_n \sim N(0, d^2)$

d is given in practice

(b) the acceptance probability.

$$a(x, x') = \min\left(\frac{\pi_{x'} p(x', x)}{\pi_x p(x, x')}, 1\right)$$

π is the desired stating PDF $f_{\theta|Y}$

$p(x, x')$: probability density of transitity from $x \to x'$

$\varepsilon_n = x' - x$, $p(x, x') = \frac{1}{\sqrt{2\pi}\, d} e^{-\frac{1}{2d^2}(x'-x)^2} = p(x', x)$

$p(x, x') = f(x'|x) = f(x + \varepsilon_n | x) = f(\varepsilon_n | x) = f(\varepsilon_n)$

Solution

$$\Rightarrow a(x, x') = \min\left(\frac{f_{\theta|Y}(x'|y)}{f_{\theta|Y}(x|y)}, 1\right)$$

$$= \min\left(\frac{e^{\frac{1}{2\sigma^2}(y-x')^2} \cdot e^{-\frac{1}{2\tau^2}(x'-\mu)^2}}{e^{-\frac{1}{2\sigma^2}(y-x)^2} \cdot e^{-\frac{1}{2\tau^2}(x-\mu)^2}}, 1\right)$$

Solution

Simulation Results with MCMC

$$E[\theta|Y=y] = \frac{\frac{1}{\sigma^2}}{\frac{1}{\sigma^2}+\frac{1}{\tau^2}} y + \frac{\frac{1}{\tau^2}}{\frac{1}{\sigma^2}+\frac{1}{\tau^2}} \mu$$

$$= \frac{1}{1+\frac{1}{4}} \cdot 3 + \frac{\frac{1}{4}}{1+\frac{1}{4}} \cdot 0$$

$$= 2.4 \checkmark$$

$$\text{var}[\theta|Y=y] = \frac{1}{\frac{1}{\sigma^2}+\frac{1}{\tau^2}} = \frac{1}{1+\frac{1}{4}}$$

$$= 0.8 \checkmark$$

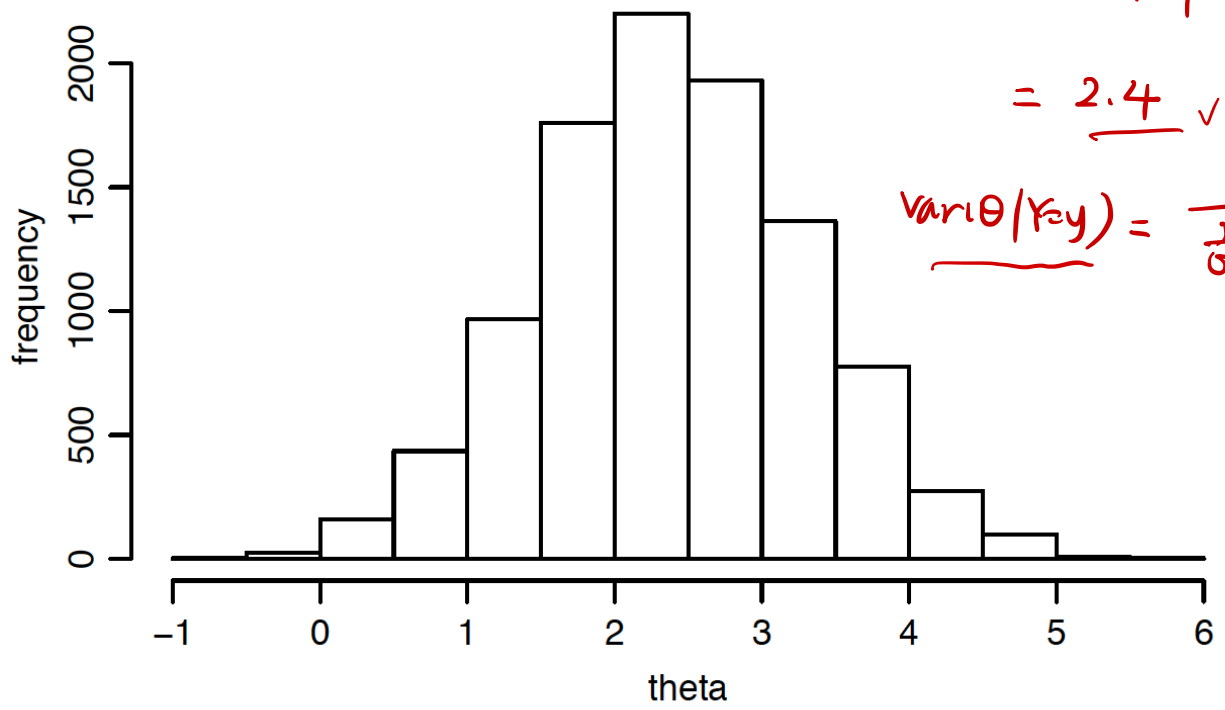


FIGURE 12.2
Histogram of 10^4 draws from the posterior distribution of θ given $Y = 3$, obtained using Metropolis-Hastings with $\mu = 0$, $\sigma^2 = 1$, and $\tau^2 = 4$. The sample mean is 2.4 and the sample variance is 0.8, in agreement with the theoretical values.

Simulation Results with MCMC

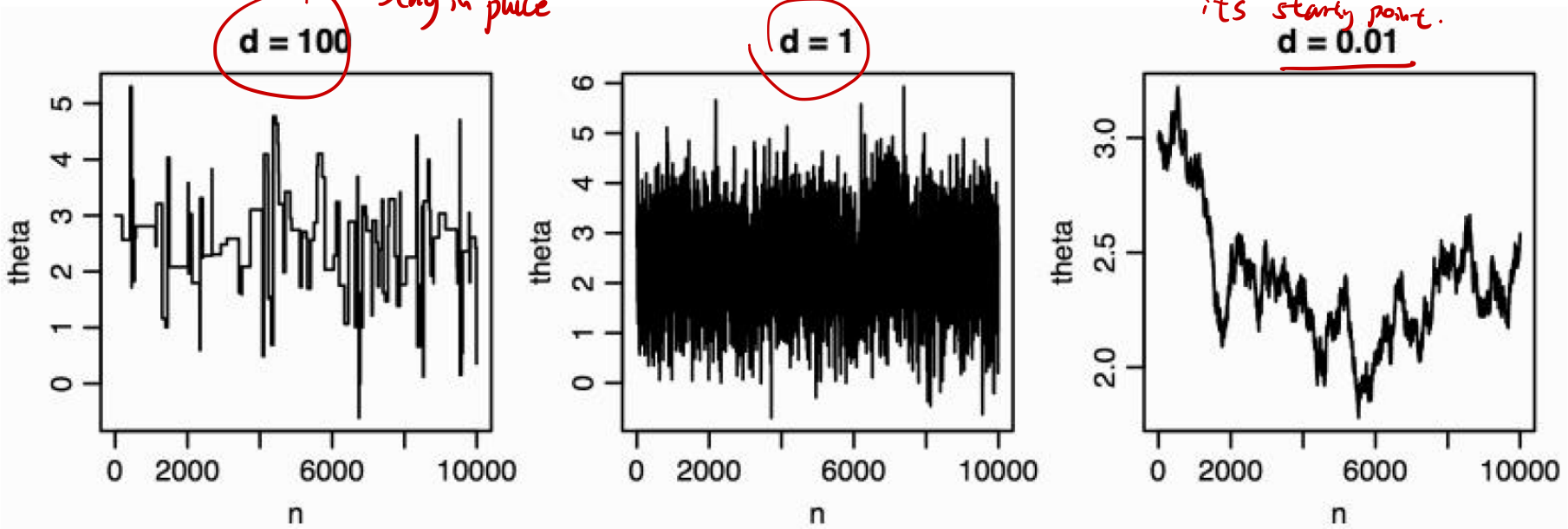


FIGURE 12.3
Trace plots of θ_n as a function of the iteration number n, for $d = 100, 1, 0.01$.

Outline

Basic Idea

- Proposed by brothers Stuart and Donald Geman in 1984.
- Named after the physicist Josiah Willard Gibbs, in reference to an analogy between the sampling algorithm and statistical physics.
- Obtaining approximate draws from a joint distribution, based on sampling from conditional distributions one at a time
- Especially useful in problems where these conditional distributions are pleasant to work with.

Basic Idea

- At each stage, one variable is updated (keeping all the other variables fixed) by drawing from the conditional distribution of that variable given all the other variables.
- Two major kinds of Gibbs sampler:
  - systematic scan: the updates sweep through the components in a deterministic order.
  - random scan: a randomly chosen component is updated at each stage.

practice ✓

Algorithm: Systematic Scan Gibbs Sampler

Let X and Y be discrete r.v.s with joint PMF $p_{X,Y}(x, y) = P(X = x, Y = y)$. We wish to construct a two-dimensional Markov chain (X_n, Y_n) whose stationary distribution is $p_{X,Y}$. The systematic scan Gibbs sampler proceeds by updating the X-component and the Y-component in alternation. If the current state is $(X_n, Y_n) = (x_n, y_n)$, then we update the X-component while holding the Y-component fixed, and then update the Y-component while holding the X-component fixed.

Algorithm 2 Systematic Scan Gibbs Sampler

Require:

Joint PMF $p_{X,Y}$;

Initial state (X_0, Y_0);

Ensure:

Two-dimensional Markov chain (X_n, Y_n);

1: **repeat**

2: Draw a value x_{n+1} from the conditional distribution of X given $Y = y_n$, and set $X_{n+1} = x_{n+1}$;

3: Draw a value y_{n+1} from the conditional distribution of Y given $X = x_{n+1}$, and set $Y_{n+1} = y_{n+1}$;

4: **return** (X_{n+1}, Y_{n+1});

5: **until** $n \geq N$;

Algorithm: Random Scan Gibbs Sampler

As above, let X and Y be discrete r.v.s with joint PMF $p_{X,Y}(x, y)$. We wish to construct a two-dimensional Markov chain (X_n, Y_n) whose stationary distribution is $p_{X,Y}$. Each move of the random scan Gibbs sampler picks a uniformly random component and updates it, according to the conditional distribution given the other component.

Algorithm 3 Random scan Gibbs sampler

Require:

Joint PMF $p_{X,Y}$;

Initial state (X_0, Y_0);

Ensure:

Two-dimensional Markov chain (X_n, Y_n);

1: **repeat**

2: Choose which component to update, with equal probabilities;

3: If the X-component was chosen, draw a value x_{n+1} from the conditional distribution of X given $Y = y_n$, and set $X_{n+1} = x_{n+1}, Y_{n+1} = y_n$. Similarly, if the Y-component was chosen, draw a value y_{n+1} from the conditional distribution of Y given $X = x_n$, and set $X_{n+1} = x_n$, $Y_{n+1} = y_{n+1}$;

4: **return** (X_{n+1}, Y_{n+1});

5: **until** $n \geq N$;

Random Scan Gibbs as Metropolis-Hastings

Theorem

The random scan Gibbs sampler is a special case of the Metropolis-Hastings algorithm, in which the proposal is always accepted. In particular, it follows that the stationary distribution of the random scan Gibbs sampler is as desired.

Gibbs Sampling vs. Metropolis-Hastings

- Gibbs sampling emphasizes conditional distributions.
- Metropolis-Hastings emphasizes acceptance probabilities.

Gibbs Sampler for Continuous State Space

In the Gibbs sampler, the target distribution $\boldsymbol{\pi}$ is an m-dimensional joint density

$$\boldsymbol{\pi}(\mathbf{x}) = \pi(x_1, \cdots, x_m).$$

A multivariate Markov chain is constructed whose limiting distribution is $\boldsymbol{\pi}$, and which takes values in an m-dimensional space. The algorithm generates elements by iteratively updating each component of an m-dimensional vector conditional on the other $m-1$ components.

Example: Bivariate Standard Normal Distribution

Joint PDF $f(x,y) = \frac{1}{2\pi\sqrt{1-\rho^2}} e^{-\frac{x^2-2\rho xy+y^2}{2(1-\rho^2)}}$. $-\infty < x, y < \infty$, $-1 < \rho < 1$

$f(x|Y=y) \sim N(\rho y, 1-\rho^2)$

$f(Y|X=x) \sim N(\rho x, 1-\rho^2)$

Example: Bivariate Standard Normal Distribution

The Gibbs sampler is implemented to simulate (X, Y) from a bivariate standard normal distribution with correlation ρ.

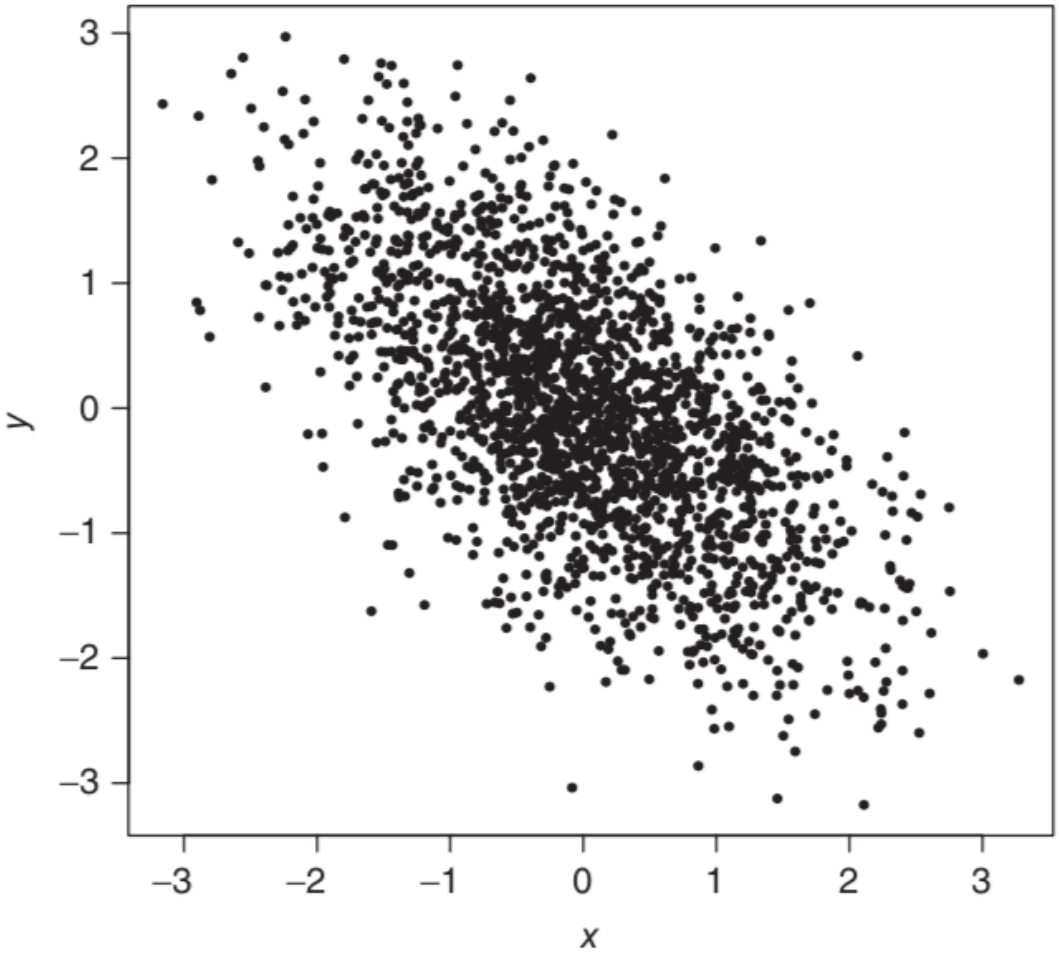
1. Initialize: $(x_0, y_0) \leftarrow (0, 0)$, $m \leftarrow 1$.
2. Generate x_m from the conditional distribution of X given $Y = y_{m-1}$. That is, simulate from a normal distribution with mean ρy_{m-1} and variance $1 - \rho^2$.
3. Generate y_m from the conditional distribution of Y given $X = x_m$. That is, simulate from a normal distribution with mean ρx_m and variance $1 - \rho^2$.
4. $m \leftarrow m + 1$.
5. Return to Step 2.

$f(x|Y=y_{m-1}) \sim N(\rho y_{m-1}, 1-\rho^2)$

$f(Y|X=x_m) \sim N(\rho x_m, 1-\rho^2)$

Simulation Results with MCMC

$\rho = -0.6$. run for 2000.



Example: Chicken-Egg with Unknown Parameters

A chicken lays N eggs, where $N \sim \text{Pois}(\lambda)$. Each egg hatches with probability p, where p is unknown; we let $p \sim \text{Beta}(a, b)$. The constants λ, a, b are known.

Here's the catch: we don't get to observe N. Instead, we only observe the number of eggs that hatch, X. Describe how to use Gibbs sampling to find $E(p|X = x)$, the posterior mean of p after observing x hatched eggs.

Solution

1°. $N \sim \text{pois}(\lambda)$. $X|P \sim \text{pois}(\lambda P)$

$$f(p|X=x) \propto P(X=x|p) f(p)$$

$$\propto e^{-\lambda p} \cdot (\lambda p)^x \, p^{a-1}(1-p)^{b-1}.$$

we can use M-H to generate samples, ...

2°. Conditioning on $N=n$, and know true value of P.

$$X|N=n, P \sim \text{Bin}(n, P)$$

$P \sim \text{Beta}(a, b)$, Beta-binomial conjugacy.

$$P|X=x, N=n \sim \text{Beta}(x+a, n-x+b)$$

$f(p|X=x)$ — hard.

$f(p, N|X=x)$ — easy.

$f(p|N=n, X=x) \sim \text{Beta}(x+a, n-x+b)$

$f(N|P=p, X=x) \overset{-n}{\sim} X + \text{pois}(\lambda(1-p))$

shifted poisson.

Solution

Solution

We make an initial guess for p and N, then iterate the following steps:

1. Conditional on $N = n$ and $X = x$, draw a new guess for p from the $\text{Beta}(x + a, n - x + b)$ distribution.

2. Conditional on p and $X = x$, the number of unhatched eggs is $Y \sim \text{Pois}(\lambda(1 - p))$ by the chicken-egg story, so we can draw Y from the $\text{Pois}(\lambda(1 - p))$ distribution and set the new guess for N to be $N = x + Y$.

Solution

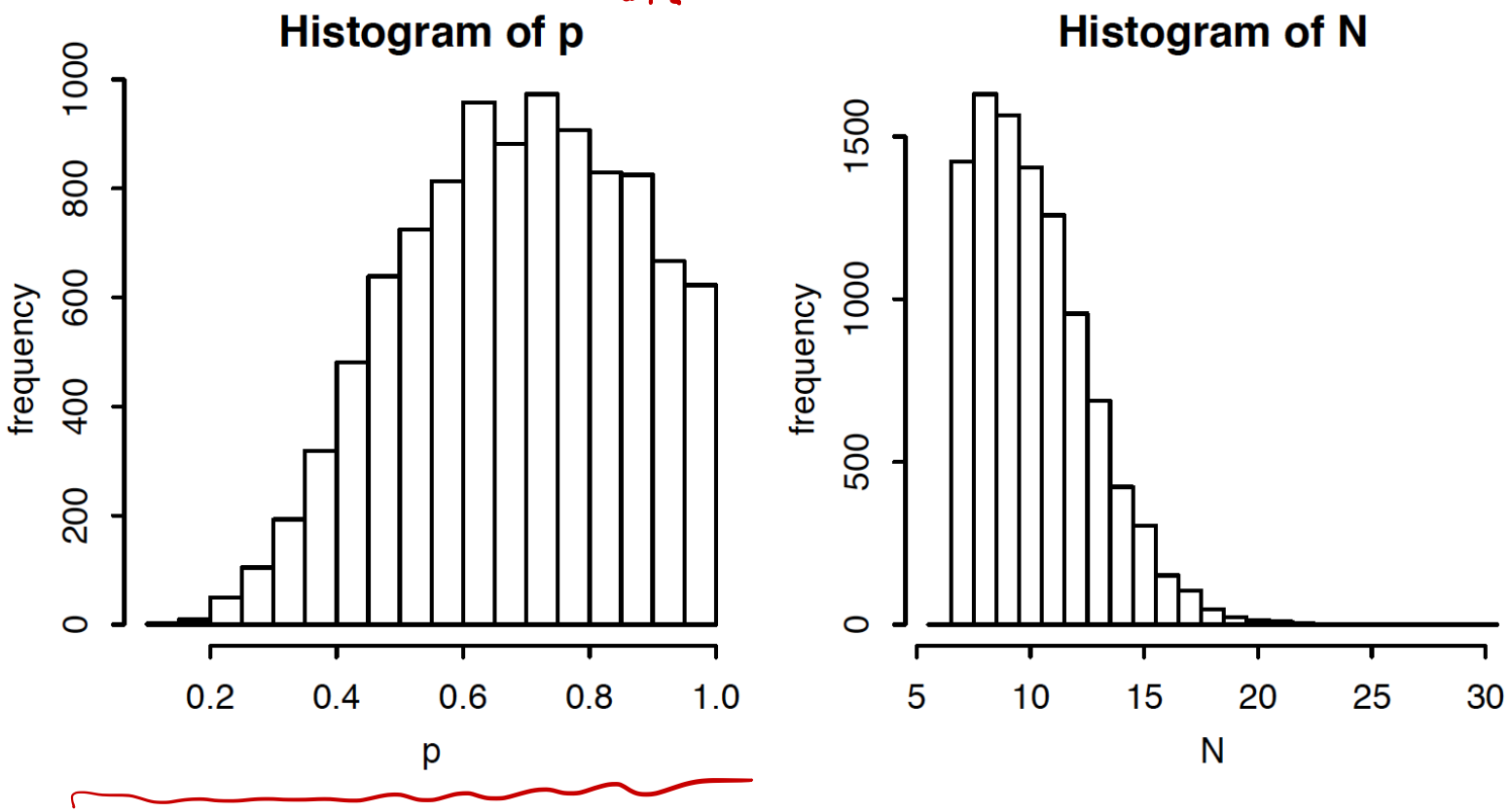


FIGURE 12.5
Histograms of 10^4 draws from the posterior distributions of p and N, where $\lambda = 10$, $a = 1$, $b = 1$, and we observe $X = 7$.

Example: Three-dimensional Joint Distribution

① PMF $f(x \mid N=n, P=p) \propto \binom{n}{x} p^x (1-p)^{n-x}$ $\sim bin(n, p)$

PDF $f(p \mid X=x, N=n) \propto p^x (1-p)^{n-x}$ $\sim Beta(x+1, n-x+1)$

PMF $f(N \mid X=x, P=p) \propto \binom{n}{x} \cdot (1-p)^{n-x} \cdot \frac{\varphi^n}{n!}$ $n = x, x+1, \ldots$

Shifted Poisson process, $\phi(1-p)$.

$\hookrightarrow = x + Z$, $Z \sim Pois(\phi(1-p))$

Random variables X, P and N have joint density

$$\pi(x, p, n) \propto \binom{n}{x} p^x (1-p)^{n-x} \frac{4^n}{n!}$$

for $x = 0, 1, \ldots, n$, $0 < p < 1$, $n = 0, 1, \ldots$. The p variable is continuous; x and n are discrete.

Solution

Solution

The Gibbs sampler, with arbitrary initial value, is implemented as follows:

1. Initialize: $(x_0, p_0, n_0) \leftarrow (1, 0.5, 2)$
   $m \leftarrow 1$
2. Generate x_m from a binomial distribution with parameters n_{m-1} and p_{m-1}.
3. Generate p_m from a beta distribution with parameters $x_m + 1$ and $n_{m-1} - x_m + 1$.
4. Let $n_m = z + x_m$, where z is simulated from a Poisson distribution with parameter $4(1 - p_m)$.
5. $m \leftarrow m + 1$
6. Return to Step 2.

The output of the Gibbs sampler is a sequence of samples

$$(X_0, P_0, N_0), (X_1, P_1, N_1), (X_2, P_2, N_2), \ldots$$

Simulation Results with MCMC

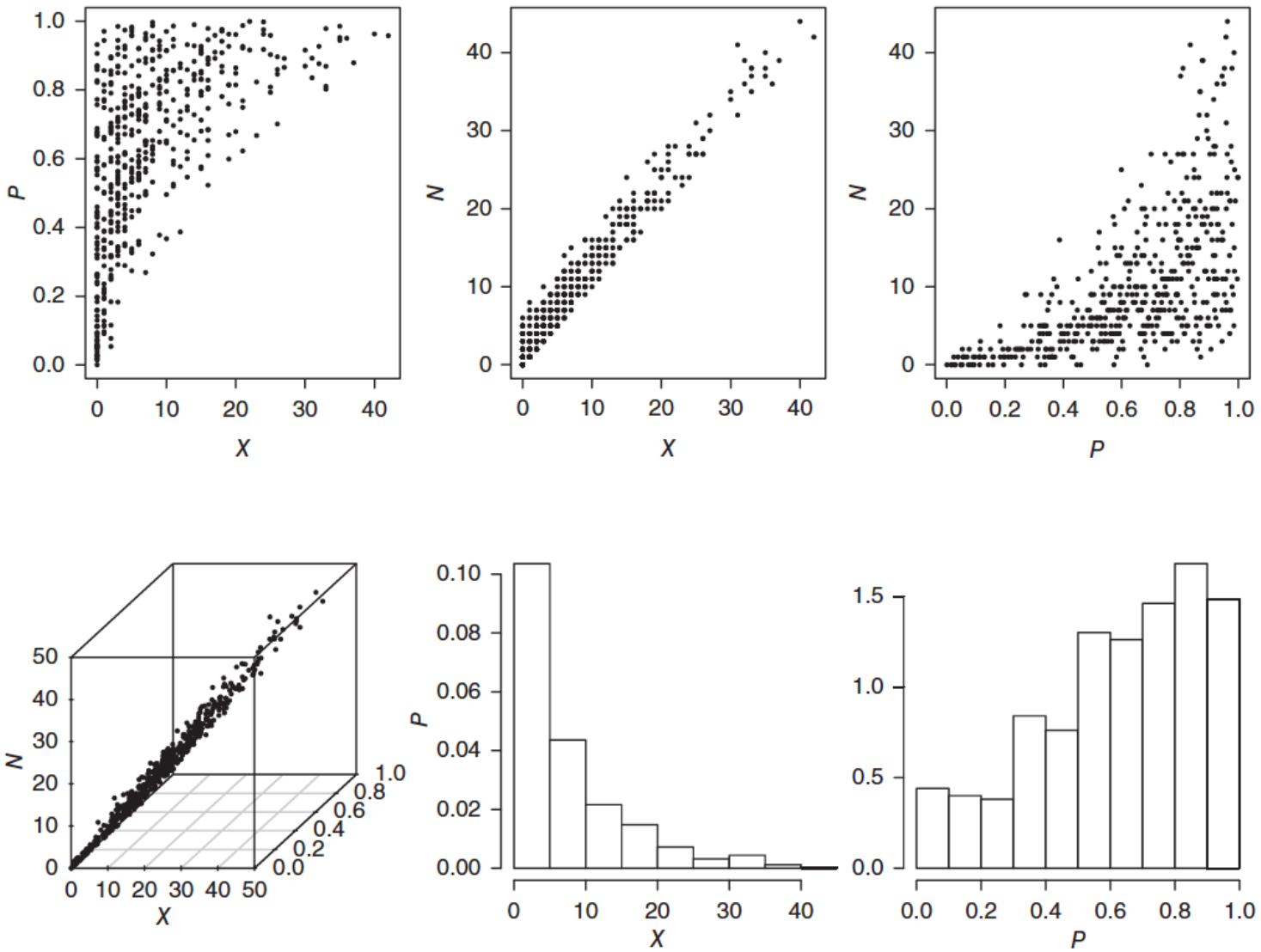
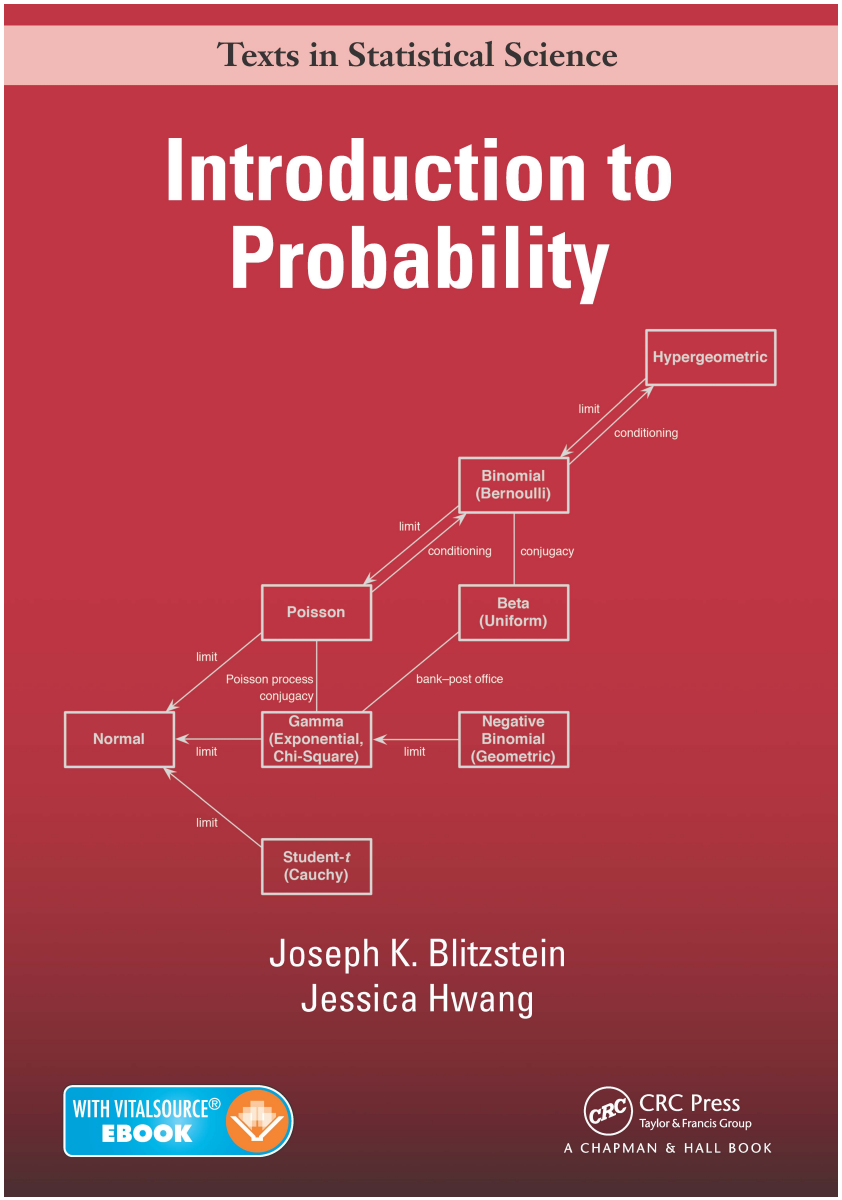


Figure 5.7 Joint and marginal distributions for trivariate distribution.

Outline

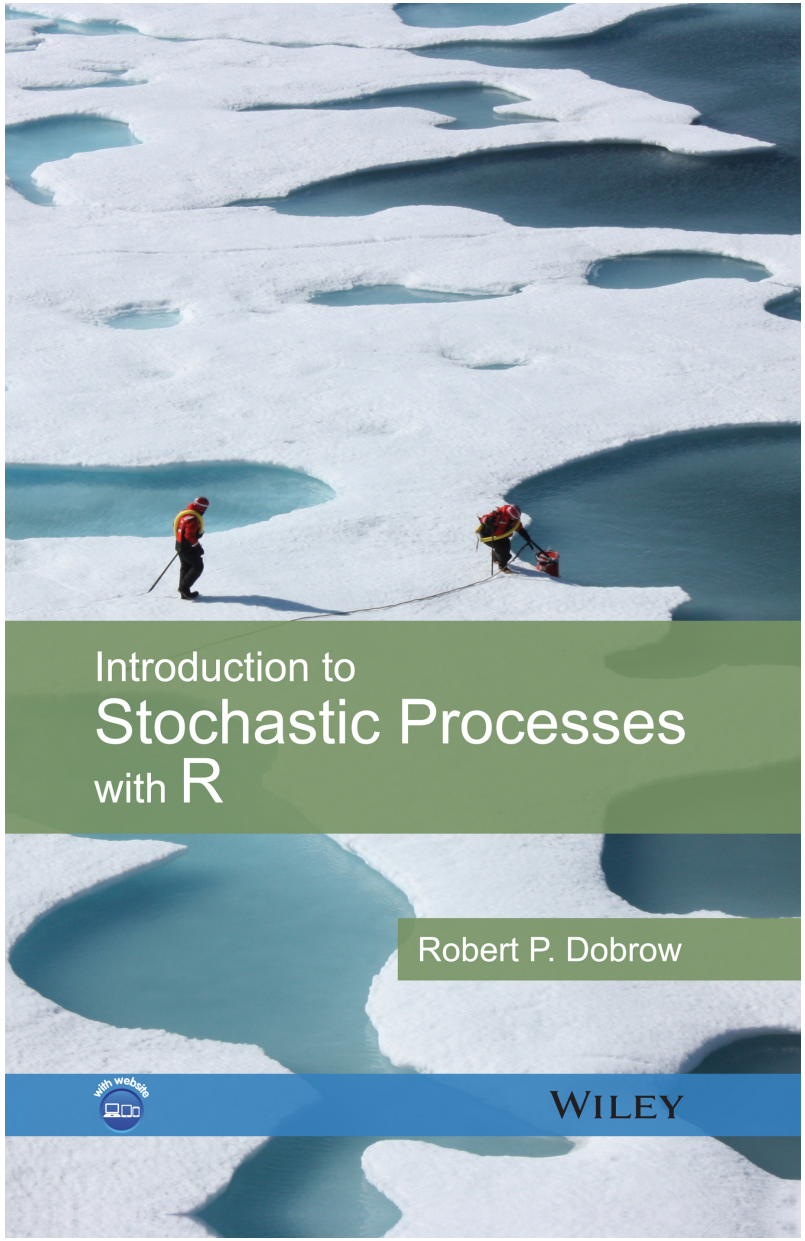
BH



BH

- Introduction to Probability
- Chapman & Hall/CRC, 2014.
- Chapman & Hall/CRC, 2019.
- Chapter 12

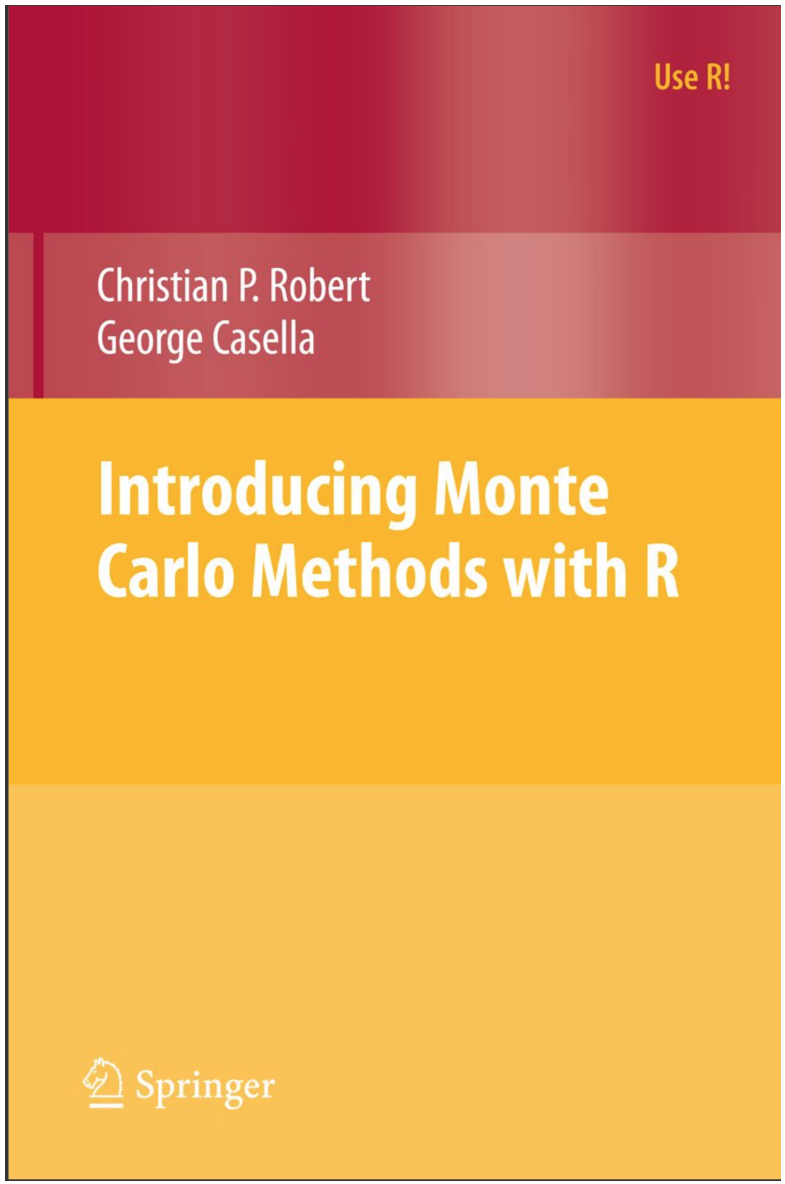
SRR



SPR

- Introduction to Stochastic Processes with R
- John Wiley & Son, 2016.
- Chapter 5

RC



RC

- Introducing Monte Carlo Methods with R
- Springer, 2010.
- All chapters including Chapter 8 (when to stop MCMC algorithm)