Monte Carlo methods for Statistical Inference: Markov Chain and Related Simulation Methods

Hung Chen

hchen@math.ntu.edu.tw

Department of Mathematics

National Taiwan University

9th March 2004

Meet at NS 104 On Wednesday from 9:10 to 12.

Outline

- Bayesian Analysis
  - Introduction
  - Slice Sampler
- Mixture Distribution
- Markov Chain
  - State space
  - Transition probability
  - Basic limit theorem
- Markov Chain Monte Carlo Method
  - Metropolis Algorithm
  - Metropolis-Hastings Algorithm

- **Gibbs Sampling**
  - **Algorithm**
  - **Example**

Bayesian Analysis

In Bayesian inference, the parameter is assumed to be random with some known distribution and its estimate ends up being an expected value under the posterior distribution.

- If θ is a random parameter with some known distribution, then according to the Bayes' rule, the posterior distribution is given by

  $$p(\theta \mid \mathbf{y}) = \frac{P(\mathbf{y} \mid \theta)P(\theta)}{P(\mathbf{y})} \propto P(\mathbf{y} \mid \theta)P(\theta)$$

  where y is the dependent random variable.

  - The first Bayesians, Bayes and Laplace, chose a constant prior distribution for θ.

  - 20 **years ago, one often heard the refrain that** *Bayesian analysis is nice conceptually; too bad it is not possible to compute Bayesian answers in realistic situations*.

- **Given an observation y, the posterior is proportional to the product of the data likelihood and the prior probability on the parameter θ.**

- **The next question is: what should be the criterion for estimating θ from observed y?**
  - **Under the posterior probability, there are several cost functions that can be minimized, resulting in different values of the estimate.**
  - **We briefly mention a few commonly used cost functions. Let $\|\cdot\|$ denote the Euclidean distance**

function on R^n.

1. **The maximizer of the posterior density is called the maximum a-posterior (MAP) estimate:**

   $$\hat{\theta}_{MAP} = arg \max_{\theta} P(\theta \mid \mathbf{y}).$$

   The MAP estimate can be found using techniques which were described to find the maximum-likelihood estimate, except the posterior density is maximized instead of the likelihood function.

2. **Let θ be the value of the estimator. Then the expected squared error is given by**

   $$\int_{\theta_1} \|\theta - \theta_1\|^2 P(\theta_1 \mid \mathbf{y}) d\theta_1.$$

   The value of θ which minimizes this expected squared error is called the minimum mean squared

error estimator (MMSE).

$$\hat{\theta}_{MMSE} = arg\min_{\theta} \textbf{Expected Squared Error}.$$

It can be shown that for Euclidean parameters, the MMSE estimator is given by the mean under the posterior (conditional) distribution.

- This estimator involves computing integral of the posterior density.
- In case the analytical solutions are difficult to evaluate numerical techniques for integration can be used.

3. The disadvantage of minimizing expected squared error is that it results in an average: the solution is the average of high posterior probability points.

- · Even the individual points may be high probability, their average itself can be a low-probability point and hence a bad estimate.
- · Therefore, very often it is the expected error which is minimized instead of the expected squared error.
- · For Euclidean parameters the minimum expected absolute error criterion results in the median of the posterior probability.

- Consider the example in motivating EM algorithm.
  - For MLE $\hat{\theta}$,

$$L(\theta \mid \mathbf{y}) \propto (2+\theta)^{125}(1-\theta)^{38}\theta^{34}.$$

The MLE is 0.62682115.

– **Assume a uniform prior for θ and a squared loss function.**

– **Posterior pdf is**

$$p(\theta \mid \mathbf{y}) \propto \pi(\theta) L(\theta \mid \mathbf{y}) = L(\theta \mid \mathbf{y}).$$

Not a recognizable distribution.

– **The Bayes estimator is its posterior mean**

$$E(\theta \mid \mathbf{y}) = \frac{\int_0^1 \theta \cdot (2+\theta)^{125}(1-\theta)^{38}\theta^{34} d\theta}{(2+\theta)^{125}(1-\theta)^{38}\theta^{34} d\theta}.$$

No explicit formula available, need a computer integration routine.

– **Note that it is equal to**

$$\frac{0.14683870067 \times 10^{29}}{0.2357695165 \times 10^{29}} = 0.6228061319.$$

Slice Sampler

Consider the posterior distribution with uniform prior of the linkage model

$$p(\theta \mid \mathbf{y}) \propto (2+\theta)^{125}(1-\theta)^{38}\theta^{34}.$$

Slice Sampler Algorithm

- **If**

$$p(x) \propto \prod_{j=1}^{k} f_j(x), \quad f_j(x) \geq 0, \quad 1 \leq j \leq k,$$

  then $p(x)$ **can be completed into**

$$g(z_1, z_2, \cdots, z_k, x) \propto \prod_{j=1}^{k} I(0 \leq z_j \leq f_j(x)).$$

- **Every conditional pdf is a uniform distribution!**

- **For this linkage problem, define**

$$f_1(\theta) = (2+\theta)^{125}, \;\; f_2(\theta) = (1-\theta)^{38}, \;\; f_3(\theta) = \theta^{34}.$$

  - **Generate** $z_1 \sim U(0, (2+\theta)^{125})$**,**
  - **Generate** $z_2 \sim U(0, (1-\theta)^{38})$**,**
  - **Generate** $z_3 \sim U(0, \theta^{34})$**.**
  - **For each** z_1, z_2, z_3 **generated, find the range of** θ**:**

$$\theta \geq z_1^{1/125} - 2, \;\; \theta \leq 1 - Z_2^{1/38}, \;\; \theta \geq z_3^{1/34}.$$

  - **Generate** $\theta^{new} \sim U(A, B)$**, where**

$$A = \max(z_1^{1/125} - 2, z_3^{1/34}), \;\; B = 1 - Z_2^{1/38}.$$

- **Why does it work?**

Mixture Distribution

Suppose that we have iid observations from a mixture of exponential distribution with density

$$f(x;\theta) = \sum_{j=1}^{k} p_j \lambda_j \exp(-\lambda_j x).$$

Here k is assumed known.

- **unknown parameter $\theta = (p_1, \ldots, p_k, \lambda_1, \ldots, \lambda_k)$**
- **Let Λ be a random variable, taking on the value λ_i with probability p_i.**
- **The likelihood can be written as**

$$L(\theta) = \prod_{i=1}^{n} f(x_i;\theta) = E\prod_{i=1}^{n} \Lambda_i \exp(-\Lambda_i x_i).$$

- **The Gibbs sampler draws, given θ, vectors $\mathbf{\Lambda} = (\Lambda_1, \ldots, \Lambda_n)$.**
- **Note that the Λ_i are iid, the Gibbs sampler just repeatedly generates iid $\theta^{(i)}$, $i = 1, \ldots, N$, and then estimates the likelihood by averaging**

$$\hat{L}(\theta) = \frac{1}{N} \sum_{j=1}^{N} \prod_{i=1}^{n} \Lambda_i^{(j)} \exp(-\Lambda_i^{(j)} x_i).$$

  - **Rather large values of N may be needed to evaluate the likelihood precisely enough.**
  - **The original maximization problem can be unpleasant.**

Markov Chain

A Markov chain describes a system whose state changes over time.

- The changes are not completely predictable, but rather are governed by probability distributions.
- These probability distributions incorporate a simple sort of dependence structure, where the conditional distribution of future states of the system, given some information about past states, depends only on the most recent piece of information.
  - That is, what matters in predicting the future of the system is its present state, and not the path by which the system got to its present state.

- **What is a Markov chain?**
  It is a sequence $\{X_0, X_1, X_2, \ldots\}$ **of random variables that has the "Markov property."**

- **Think about how to simulate a Markov chain, a typical "sample path."**

- **How do I tell you which particular Markov chain I want you to simulate?**

  - **State space** $\mathcal{S}$**:** $\mathcal{S}$ **is a finite or countable set of states, that is, values that the random variables** X_i **may take on.**
    For definiteness, and without loss of generality, label the states as follows: either $\mathcal{S} = \{1, 2, \ldots, N\}$ **for some finite** N**, or** $\mathcal{S} = \{1, 2, \ldots\}$**, which we may think of as the case** $N = \infty$**?**

- **Initial distribution π_0: This is the probability distribution of the Markov chain at time 0. For each state $i \in \mathcal{S}$, we denote by $\pi_0(i)$ the probability $P\{X_0 = i\}$ that the Markov chain starts out in state i where**

$$\pi_o(i) \geq 0 \text{ for all } \mathcal{S} \text{ and } \quad \sum_i \pi_0(i) = 1.$$

- **Probability transition rule: This is specified by giving a matrix $P = (P_{ij})$.**
  If $\mathcal{S}$ is the finite set $\{1, \ldots, N\}$, say, then P is an $N \times N$ matrix.
  - **The interpretation of the number P_{ij} is the conditional probability, given that the chain is in state i at time n, say, that the chain jumps to**

the state j at time $n+1$. That is,

$$P_{ij} = P\{X_{n+1} = j \mid X_n = i\}.$$

- $*$ **Note that we have written this probability as a function of just i and j, but of course it could depend on n as well.**
- $*$ **The** *time homogeneity* **restriction is just the assumption that this probability does not depend on the time n, but rather remains constant over time.**
- $*$ **A probability transition matrix is an $N \times N$ matrix whose entries are all nonnegative and whose rows sum to 1.**

- **Simulate** *Markov frog*.
  Think of a frog hopping among three lily ponds.

– Here state refers to those three lily ponds. The state will change over time.

– To start, he chooses his initial position X_0 according to the specified initial distribution $\pi_0 = (1/2, 1/4, 1/4)$. We can use computer simulation to generate his initial position by generating a uniformly distributed random number $U_0 \sim Unif(0,1)$, and then taking

$$X_0 = \begin{cases} 1 \text{ if } 0 \le U_0 \le 1/2 \\ 2 \text{ if } 1/2 < U_0 \le 3/4 \\ 3 \text{ if } 3/4 < U_0 \le 1 \end{cases}$$

– How does the Markov frog choose a path?

Assume that the probability transition matrix is

$$P = \begin{pmatrix} 0 & 1 & 0 \\ 1/3 & 0 & 2/3 \\ 1/3 & 1/3 & 1/3 \end{pmatrix}$$

Suppose that U_0 comes out to be 0.8419, so that $X_0 = 3$.

Then the frog chooses X_1 according to the probability distribution in row 3 of P. He paws his computer again to generate $U_1 \sim Unif(0,1)$ independently of U_0, and takes

$$X_1 = \begin{cases} 1 \textbf{ if } 0 \leq U_1 \leq 1/3 \\ 2 \textbf{ if } 1/3 < U_1 \leq 2/3 \\ 3 \textbf{ if } 2/3 < U_1 \leq 1 \end{cases}$$

Suppose he happens to get $U_1 = 0.1234$, so that

$X_1 = 1$. Then he chooses X_2 according to row 1 of P, so that $X_2 = 2$; there's no choice this time.

- Next, he chooses X_3 according to row 2 of P. And so on $\ldots$.

- The Markov property: Clearly, in the previous example, if I told you that we came up with the values $X_0 = 3$, $X_1 = 1$, and $X_2 = 2$, then the conditional probability distribution for X_3 is

$$P(X_3 = j \mid X_0 = 3, X_1 = 1, X_2 = 2) = \begin{cases} 1/3 & \textbf{for } j = 1 \\ 0 & \textbf{for } j = 2 \\ 2/3 & \textbf{for } j = 3, \end{cases}$$

which is also the conditional probability distribution for X_3 given only the information that $X_2 = 2$.

- Markov chain is a particular family of discrete time

stochastic processes.

- **Denote a discrete time stochastic process as:** $\{X_t, t = t_1, t_2, \ldots\}$**. Such a process can be characterized by** n**th- order joint probability density function, for any** $n \geq 1$**,**

$$P_{X_{t_1}, X_{t_2}, \ldots, X_{t_n}}(x_1, x_2, \ldots, x_n),$$

Using the rule for total probability, we can factor the joint density function as:

$$P_{X_{t_1}, X_{t_2}, \ldots, X_{t_n}}(x_1, x_2, \ldots, x_n) = P_{X_{t_1}}(x_1) P_{X_{t_2} | X_{t_1}}(x_2 \mid x_1) \\ \cdots P_{X_{t_n} | X_{t_{n-1}}, \ldots, X_{t_1}}(x_n \mid x_{n-1}, \ldots, x_1)$$

where the first term on the right side is the marginal density of the process at time t_1**, and the remaining terms are the conditional densities.**

- **A stochastic process is called a Markov process if**

  $P_{X_{t_n}|X_{t_{n-1}},\ldots,X_{t_1}}(x_n \mid x_{n-1},\ldots,x_1) = P_{X_{t_n}|X_{t_{n-1}}}(x_n \mid x_{n-1})$.

  – **The issue addressed by the Markov property is the dependence structure among random variables.**

  – **The simplest dependence structure for** $X_0, X_1, \ldots$ **is no dependence at all, that is, independence.**

  – **The Markov property could be said to capture the next simplest sort of dependence: in generating the process** $X_0, X_1, \ldots$ **sequentially, the "next" state** X_{n+1} **depends only on the "current" value** X_n**, and not on the "past" values** $X_0, \ldots, X_{n-1}$**.**

- **The Markov property implies a simple expression for the probability of our Markov chain taking any spec-**

ified path, as follows:

$$\begin{aligned}&P(X_0 = i_0, X_1 = i_1, \ldots, X_n = i_n)\\&= P(X_0 = i_0)P(X_1 = i_1 \mid X_0 = i_0)P(X_2 = i_2 \mid X_1 = i_1)\\&\cdots P(X_n = i_n \mid X_{n-1} = i_{n-1})\\&= \pi_0(i_0)P(i_0, i_1)P(i_1, i_2)\cdots P(i_{n-1}, i_n).\end{aligned}$$

- **The joint density function can be written as a product of one-step conditional densities.**
- **If we only observe** $X_0 = i_0$ **and** $X_2 = i_2$**, what is** $P(X_0 = i_0$ **and** $X_2 = i_2)$**?**

$$\pi_0(i_0)\sum_{i_1} P(i_0, i_1)P(i_1, i_2)$$

  Recall the matrix multiplication $a_{ik} = \sum_j a_{ij}a_{jk}$**.**

- **Recall that** $\pi_0(i) = P(X_0 = i)$ **which describes the initial distribution of the chain.**

Let π_n denote the distribution of the chain at time n, that is, $\pi_n(i) = P(X_n = i)$.

– **Assume that the state space is finite with N states.**

– **Denote the transition matrix by**

$$P = (P_{ij}) = (P(i,j)),$$

– **The law of total probability gives**

$$\pi_{n+1}(j) = \sum_{i=1}^{N} \pi_n(i) P(i,j).$$

In matrix notation, it is just the equation

$$\pi_{n+1} = \pi_n P,$$

where

$$\pi_n = (\pi_n(1), \ldots, \pi_n(N)).$$

By induction, we have

$$\pi_n = \pi_0 P^n.$$

- In principle, we can find the answer to any question about the probabilistic behavior of a Markov chain by doing matrix algebra, finding powers of matrices, etc.
- In practice, it is another story.

  For example, the state space for a Markov chain that describes repeated shuffling of a deck of cards contains $52!$ elements.

  The probability transition matrix that describes the effect of a single shuffle is a $52! \times 52!$ matrix.

- The joint density function is called translation invariant if for any $c \geq 0$, we have

$P_{X_{t_1},X_{t_2},\ldots,X_{t_n}}(x_1, x_2, \ldots, x_n) = P_{X_{t_1+c},X_{t_2+c},\ldots,X_{t_n+c}}(x_1, x_2, .$

– A stochastic process is called *stationary* if its nth-order joint density function is translation invariance, for all $n \geq 1$.

- Random walk on a clock: Consider a clock with 6 numbers on it: $0, 1, 2, 3, 4, 5$.
  – Suppose we perform a random walk by moving clockwise, moving counterclockwise, and staying in place with probabilities $1/3$ each at every time n. That is,

$$P(i, j) = \begin{cases} 1/3 \text{ if } j = i - 1 \textbf{ mod } 6 \\ 1/3 \text{ if } j = i \\ 1/3 \text{ if } j = i + 1 \textbf{ mod } 6 \end{cases}$$

– **Suppose we start out at $X_0 = 2$, say. That is,**

$$\pi_0 = (0, 0, 1, 0, 0, 0).$$

Then $\pi_1 = (0, 1/3, 1/3, 1/3, 0, 0)$**,** $\pi_2 = (1/9, 2/9, 1/3, 2/9, 1/$
and $\pi_3 = (3/27, 6/27, 7/27, 6/27, 3/27, 2/27)$**.**
It is intuitively clear that π_n will approach the uniform distribution as $n \to \infty$.
Does the initial starting state 2 matter?

– **It means that the random walk has essentially "forgotten" that it started out in state 2 at time 0.**

π_n approaches a limit that does not depend upon the initial distribution π_0.

- **Basic Limit Theorem. Let $X_0, X_1, \ldots$ be an irreducible, aperiodic Markov chain having a stationary**

distribution $\pi(\cdot)$. Let X_0 have the distribution π_0, an arbitrary initial distribution. Then

$$\lim_{n\to\infty} \pi_n(i) = \pi(i) \quad \textbf{for all states } i.$$

- *irreducible*: **All states communicate with each other for the corresponding transition matrix P.**
    - $*$ **Irreducibility implies that it is possible to visit from any state to any state in a finite number of steps.**
- *aperiodic*: **An irreducible Markov chain is called aperiodic if its period is one.**
    - $*$ **Aperiodicity implies that the Markov chain does not cycle around in the states with a finite period.**
- *stationary distribution*: **Suppose a distribution π**

on $\mathcal{S}$ is such that, if our Markov chain starts out with initial distribution $\pi_0 = \pi$, then we also have $\pi_1 = \pi$. Then π is called a stationary distribution for the Markov chain.

- **Stationary distribution**
  - **If the $N \times N$ probability transition matrix P is symmetric, then the uniform distribution is stationary.**
  - **The uniform distribution is stationary if the matrix P is doubly stochastic, that is, the column-sums of P are 1 (we already know the row-sums of P are all 1).**
  - **Computing stationary distributions is an algebra problem.**

Markov Chain Monte Carlo Method

Goal: Estimate $E[g(X)]$, where $X \sim \pi(x)$ and $g(X)$ is a function of X.

- **If $\pi(x)$ is a well-known distribution, one may find an algorithm to generate iid $X_1, X_2, \dots, X_n, \dots$**

$$\hat{E}[g(X)] = \frac{1}{n}\sum_{i=1}^{n} g(X_i) = \hat{g}(X).$$

  - **What if no such algorithm is available?**
  - **e.g. $p(\theta \mid \mathbf{y}) \propto (2+\theta)^{125}(1-\theta)^{38}\theta^{34}$.**

- **Generate $X_0, X_1, X_2, \dots$ to form a Markov Chain.**
  - **Markov Chain: X_i depends on $X_0, X_1, X_2, \dots, X_{i-2}, X_{i-1}$ only through X_{i-1}.**

– **Under some general conditions, this Markov Chain will converge to a stationary distribution with pdf** $\pi(x)$**.**

$$\mu = E_\pi[g(X_i)] = \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} g(X_i).$$

where π **is the** *equilibrium distribution***, also called invariant distribution, stationary distribution, or ergodic limit of the Markov chain (assuming such exists).**

- **When** n **is large,**

$$Var(\hat{g}(X)) \approx \frac{1+\rho_1}{1-\rho_1}\frac{Var(g(X))}{n},$$

where ρ_1 **is the lag** 1 **autocorrelation of the Markov Chain.**

How do we pick up the right chain so that the variance is small?

- **Cause and effect of bigger ρ_1:**
  - **smaller movement (or lots of repeats) in the Markov Chain.**
  - **larger variance for its estimator**
- **MCMC Theory: It is just like IIDMC theory (except MC replaces IID).**
  - **The Markov chain law of large numbers (LLN) says**

$$\hat{\mu}_n \overset{a.s.}{\to} \mu, \quad n \to \infty.$$

  - **The Markov chain central limit theorem (CLT)** says

$$\sqrt{n}(\hat{\mu}_n - \mu) \xrightarrow{D} Normal(0, \sigma^2), \quad n \to \infty$$

where

$$\sigma^2 = Var[g(X_i)] + 2 \sum_{k=1}^{\infty} Cov[g(X_i), g(X_{i+k})].$$

See Chan and Geyer (1994) for assumptions.

– We do not have to approximate the rather obnoxious formula for asymptotic variance.

  ∗ If b is large, then

$$\sqrt{b}(\hat{\mu}_b - \mu) \approx Normal(0, \sigma^2)$$

  and $b(\hat{\mu}_b - \mu)^2 \approx \sigma^2$.

  ∗ If b is small compared to n, then

$$b(\hat{\mu}_b - \hat{\mu}_n)^2 \approx \sigma^2.$$

    - $*$ **If** $1 \ll b \ll n$ **the sample average of** $b(\hat{\mu}_b - \hat{\mu}_n)^2$ **over** *batches of length* b **estimates** σ^2**.**

  - **Demo**
    - $*$ **For a demo problem, which was a PhD take-home exam question at Minnesota, get at http://www.stat.umn.edu/geyer/PhD/F03/q1.pdf**
    - $*$ **The problem is to do Bayesian logistic regression with normal prior on the parameters with four predictor variables plus constant (five parameters).**
    - $*$ **Geyer's program can be downloaded at http://www.stat.umn.edu/geyer/mcmc.**

- **References**
  - **Chan, K.S. and Geyer, C.J. (1994). Discussion of**

the paper by Tierney. AS, 22, 1747-1758.

– Green, P.J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82, 711-732.

– Hastings, W.K. (1970), Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57, 97-109.

– Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H. and Teller, E. (1953). Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, 21, 1087-1092.

MCMC History

MCMC is a remarkable tour de force.

- It dates back to the dawn of the computer age (Metropolis, et al., 1953), but is highly non-obvious, even in its original incarnation, which was calculating ergodic limits for models of physical systems.
- What is obvious is run the (model of the) physical system and average over time (that's what an ergodic limit is).
- The tour de force is the realization that any other Markov process with the same ergodic limit will also do.
- Metropolis, et al. (1953) realized this and provided a simple algorithm for constructing a Markov chain

having a specified equilibrium distribution (the Metropolis algorithm).

  - The Metropolis algorithm, as generalized by Hastings (1970) and Green (1995), called the Metropolis-Hastings-Green algorithm, is the only known method of MCMC.
  - Every MCMC-like method is either a special case of the MHG algorithm, or is bogus.

- Many researchers have invented almost-but-not-quite MCMC algorithms.
  - But there is no theory about almost-but-not-quite Markov chains or about Markov chains having almost-but-not quite a specified equilibrium distribution.

- If you're going to do MCMC, do real MCMC, not bogo-MCMC.
- The first task in any MCMC project is to verify that your computer code actually implements a Markov chain having the specified equilibrium distribution.

Metropolis Algorithm

It is the first Markov chain Monte carlo method introduced by Metropolis et al. (1953).
Suppose that we want to generate a random variable with density $f(x)$.
Idea: By the Basic Limit Theorem, one way to do this (approximately) is to find an irreducible, aperiodic probability transition matrix P satisfying $\pi P = \pi$, and then run a Markov chain according to P for a sufficiently long time.

1. Start with any y_0, $X_0 = y_0$.
2. At the $(i+1)$-th stage, generate

$$y_{i+1} = X_i + s, \quad s \sim U(-a, a).$$

- To run the random walk, at each time, we choose a random neighbor and go there.

3. Generate u from $U(0, 1)$.

4. If

$$r = \frac{f(y_{i+1})}{f(X_i)} \geq u,$$

then $X_{i+1} = y_{i+1}$, else $X_{i+1} = X_i$.

   - If $r < 1$ go to y_{i+1} with probability r, and stay at x_i with probability $1 - r$.

It generates a random walk and performs an acceptance/rejection based on p evaluated at successive steps in the walk.
Under some regularity conditions,

- $X_0, X_1, X_2, \ldots$ forms a Markov Chain.

- – **The next state only depends on the previous state, so that this is a Markov chain.**
- $X_n \to X \sim f(x)$**.**

Advantages:

- **No normalizing constant of** $f(x)$ **needed.**
- **No** $g(x)$ **or** M **needed.**

Metropolis-Hastings Algorithm

One of the most popular MCMC technique used in approximate sampling from complicated distributions is the Metropolis-Hastings algorithm.
The setup is as earlier:

- We are interested in generating samples of a random variable X distributed according to the density $f(x)$.
- In addition to $f(x)$, we will assume having a density $q(y \mid x)$ that satisfies the following properties:
  1. It is easy to sample from $q(\cdot \mid x)$ for all x.
  2. The support of q contains the support of $f(x)$.
  3. The functional form of $q(y \mid x)$ is known or $q(y \mid x)$ is symmetric in y and x. As shown later, it is

not necessary to know the normalizing constant in $q(y \mid x)$ as long as it does not depend upon x.

- Given $f(x)$ and a choice of $q(y \mid x)$, that satisfies the above mentioned properties, the Metropolis-Hastings algorithm can be stated as follows: Choose an initial condition X_0 in the support of $f(x)$. The Markov chain $X_1, X_2, \ldots X_n$ is constructed iteratively according to the steps:

  1. Generate a candidate $Y \sim q(y \mid X_t)$.
  2. Update the state to X_{t+1} according to:

$$X_{t+1} = \begin{cases} Y & \textbf{with probability } \rho(X_t, Y) \\ X_t & \textbf{with probability } 1 - \rho(X_t, Y) \end{cases},$$

  where $\rho(x, y) = \min\{f(y)q(x \mid y)/[f(x)q(y \mid x)], 1\}$.

  $q(y \mid x)$ is called the *proposal* density and $\rho(x, y)$ is

called the *acceptance-rejection* **function.**

Discussions on this algorithm:

- **Consider first the case where the ratio**

  $$f(y)q(x \mid y)/[f(x)q(y \mid x)$$

  values more than one and hence the acceptance-rejection function takes the value one.
  In this case, we set $X_{t+1} = Y$ with probability one.
  In other words, whenever this ratio exceeds once we change the state to the candidate.

- **In case this ratio goes below one, we set X_{t+1} to Y with probability $\rho(X_t, Y)$.**
  Higher the value of $\rho(X_t, Y)$ is, higher are the chances of accepting Y as the new state.

- **Note that the normalizing constants in the two densities f and q cancel out and hence are not explicitly needed.**
  - **If the normalizing constant for $q(y \mid x)$ depends upon x, then it does not cancel out and is needed in the expression for ρ.**
  - **$q(y \mid x_t)$ is the transition probability from x_t to y.**
- **In the algorithm, one generates samples from the proposal q at every step independently but the elements of the chain are not independent of each other. In fact, many times it is possible to have X_t and X_{t+1} be identical.**
- **Popular choices of $q(y \mid x_t)$:**

  1. **Independence chain: In cases where the proposal**

density $q(y \mid x_t)$ is independent of the current state x_t, i.e. $q(y \mid x_t) = q(y)$.

2. Random walk chain: In some applications, it is useful to generate proposals using a random walk. That is, the proposal is obtained using the equation:

   $$Y = X_t + \epsilon,$$

   where ϵ has density $g(\epsilon)$ that is symmetric and unimodal at zero.

   The proposal density is symmetric:

   $$q(y \mid x_t) = g(y - x)$$

   and the algorithm simplifies.

   Common choices are the uniform, normal, and t distributions.

Gibbs Sampler

Gibbs sampler is another commonly used tool for generating Markov chains with suitable asymptotic properties.

- **By construction, Gibb's sampler applies only to the problem of sampling from multivariate densities.**
  - **Recall the difficulty of generating multivariate random variable by rejection method.**
  - **It uses a sequence of univariate random numbers from conditional univariate distributions that combine to yield the desired multivariate distribution.**
- **Let $\mathbf{X} = (X_1, X_2, \dots, X_p) \in R^p$ be a vector of random variables with the joint density function given by**

$f(x_1, x_2, \ldots, x_p)$.

- Our goal is to generate samples from f and we will do so by constructing a Markov chain on R^p.
- In order to use Gibbs sampling we make the following assumption: we know the conditional densities
$$f_j(x_i \mid x_j, j \neq i), i = 1, 2, \ldots, p,$$
and have method(s) to sample from each of these. These conditional densities are called the full conditionals and have the simplicity of univariate densities.

An algorithm for Gibbs sampler is as follows:

0. Let $\mathbf{X}^{(t)} = (X_1^{(t)}, X_2^{(t)}, \ldots, X_p^{(t)}) \in R^p$ be the value of Markov chain at time t.

Following steps describe an update from $X^{(t)}$ to $X^{(t+1)}$.

1. Generate $X_1^{(t+1)} \sim f_1(x_1 \mid X_2^{(t)}, X_3^{(t)}, \ldots, X_p^{(t)})$.
2. Generate $X_2^{(t+1)} \sim f_2(x_2 \mid X_1^{(t+1)}, X_2^{(t)}, \ldots, X_p^{(t)})$.

   ⋮

3. Generate $X_p^{(t+1)} \sim f_p(x_p \mid X_1^{(t+1)}, X_2^{(t+1)}, \ldots, X_{p-1}^{(t+1)})$.

The p-dim random vector $\mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots$ form a Markov Chain with limiting distribution $f(x_1, x_2, \ldots, x_p)$.

- An important property of the Gibbs sampler is that even for large values of p, one samples from a univariate density at each step. This makes Gibbs sampler very attractive for large dimensional problems

such as image analysis. Consider the Ising model in Geman and Geman (1984).

- **Gibbs sampler is most useful when its full conditionals are easy to find and simulate.**
  - **multivariate normal distribution, multivariate exponential distribution**

Consider a specific case of Gibbs sampler for $p = 2$ which is called the bivariate Gibbs sampler.

- **Let X and Y be two scalar random variables with the joint density function $f(x, y)$ and the full conditionals: $f_1(x \mid y)$ and $f_2(y \mid x)$.**
- **Gibbs sampler can be constructed as follows:** **Start with some initial condition (x_0, y_0) and iterate according to:**

1. **Generate** $x_{t+1} \sim f_1(x \mid y_t)$**.**
2. **Generate** $y_{t+1} \sim f_2(y \mid x_{t+1})$**.**
3. **Set** $t = t + 1$ **and go to Step 1.**

- **Example: Consider generating a bivariate normal random vector with density**

$$(X, Y) \sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}\right),$$

with the full conditionals:

$$f_1(x, y) = N(\rho y, 1 - \rho^2), \quad f_2(y \mid x) = N(\rho x, 1 - \rho^2).$$

Given y_t**, generate**

$$\begin{aligned} X_{t+1} \mid y_t &\sim N(\rho y_t, 1 - \rho^2), \\ Y_{t+1} \mid x_{t+1} &\sim N(\rho x_{t+1}, 1 - \rho^2). \end{aligned}$$

- **Gibbs sampler by construction applies only to multivariate densities.**
  What if we are interested in sampling from a univariate density $f(x)$ with $p = 1$?
  - **We can** *complete* **it to a d-dimensional problem and then apply Gibbs sampler.**
  - **$g(x, z)$ is called a** *completion* **of pdf $p(x)$, if**

$$\int_Z g(x, z) dz = f(x).$$

  - **There are many choices of g, we choose g such that Gibbs algorithm is easy to implement on g.**
- **Example: Generate a truncated normal distribution**

$$f(x) \propto \exp[-(x - \mu)^2/2\sigma^2] I_{\{x \geq c\}}.$$

It is inefficient to use standard generator when x is large.
Consider Gibbs sampler with the completion

$$g(x,z) \propto I_{\{x \geq c\}} I_{\{0 \leq z \leq \exp[-(x-\mu)^2/2\sigma^2]\}}.$$

- Show that the marginal of g is indeed $f(x)$.
- $Z \mid X = x \sim U(0, \exp(-(x-\mu)^2/2\sigma^2))$
- $X \mid Z = z \sim U(a,b)$, with $a = c$ and $b = \mu + \sqrt{-2\sigma^2 \log(z)}$

- Recall Slice Sampler!