STATS 225: Bayesian Analysis
Markov Chain Monte Carlo (MCMC)

Babak Shahbaba

Department of Statistics, UCI

Winter, 2015

Background

- We saw previously that in certain situations, the posterior distribution has a closed form (e.g., when the prior is conjugate), and the integrals are tractable.
- For many other problems, however, finding the posterior distribution and obtaining the expectation are far from trivial.
- Remember that even for the case of simple normal distribution with two parameters the posterior didn't have a closed form unless we were willing to use noninformative priors or tie the variance of the mean to the variance of the data.
- In this lecture, we focus on problems where the posterior distribution is not analytically tractable.
- For this, we need to learn about Monte Carlo methods and Markov chain stochastic processes.

Monte Carlo methods: A general framework

- Assume that are interested in finding integrals of the form $I = \int_a^b g(x)dx$.
- If we can draw iid samples, $x^{(1)}, x^{(2)}, ..., x^{(m)}$ uniformly from (a, b), we can approximate this integral as

$$\hat{I}_m = (b-a)\frac{1}{m}[g(x^{(1)}) + g(x^{(2)}) + ... + g(x^{(m)})]$$

- Based on the law of large numbers, we know that

$$\lim_{m\to\infty} \hat{I}_m = I, \qquad \text{with probability } 1$$

- And based on the central limit theorem

$$\sqrt{m}(\hat{I}_m - I) \to N(0, \sigma^2), \qquad \sigma^2 = \text{Var}(g(x))$$

Monte Carlo method for finite expectation

- Now, let's consider the problem of finding integrals of the form $\int_{\mathcal{X}} h(x)f(x)dx$, where $f(x)$ is a probability density function. Recall that this integral is in fact $\mu = E_f(h(x))$, i.e., the expectation of $h(x)$.

- Analogous to the above argument, we can approximate this integral (or expectation) by drawing iid samples $x^{(1)}, x^{(2)}, ..., x^{(m)}$ from the density $f(x)$ and then

$$\hat{\mu} = \frac{1}{m}[h(x^{(1)}) + h(x^{(2)}) + ... + h(x^{(m)})]$$

- For sampling x from f, we can sample $u \sim \text{Uniform}(0, 1)$, and set $x = F^{-1}(u)$, where F^{-1} is the inverse CDF of f.

- This would of course work if the CDF has a closed form and we can find its inverse. Otherwise, we need to use other methods (described below).

Example: sampling from exponential distribution

- For example, assume we want to find the expectation of the function $h(x) = \sqrt{x}$ with respect to the exponential distribution Exp(3) where $f(x) = \theta \exp(-\theta x)$ is the density and $F(x) = 1 - \exp(-\theta x)$ is the CDF.

- We can sample $u^{(i)} \sim \text{Uniform}(0, 1)$ for $i = 1, ..., m$, and set

$$x^{(i)} = -\frac{log(1 - u^{(i)})}{\theta}$$

- We can then estimate $E_f(\sqrt{x})$ as

$$\hat{\mu} = \frac{1}{m}[\sqrt{x^{(1)}} + \sqrt{x^{(2)}} + ... + \sqrt{x^{(m)}}]$$

- Of course, for well-known distributions such as the exponential this is not necessary anymore since their own specific random generating function is usually provided.

Rejection sampling

- If it is difficult or computationally intensive to sample directly from $f(x)$ (as described above), we need to use other strategies.

- Although it is difficult to sample from $f(x)$, suppose that we can evaluate the density at any given point up to a constant $f(x) = f^*(x)/Z$, where Z could be unknown (remember that this makes the computation convenient since in most cases we know the posterior distribution only up to a constant).

- Furthermore, assume that we can easily sample from another distribution with the density $g(x) = g^*(x)/Q$, where Q is also a constant.

- Now we choose the constants c such that $cg^*(x)$ becomes the envelope (blanket) function for $f^*(x)$:

$$cg^*(x) \geq f^*(x), \qquad \forall x$$

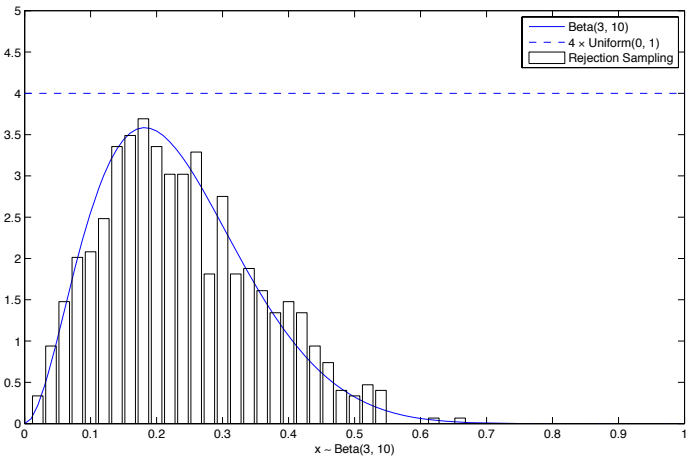
- Then, we can use a strategy known as *rejection sampling* in order to sample from $f(x)$ indirectly.

Rejection sampling

- The rejection sampling method works as follows:
  1. draw a sample x from g(x)
  2. generate $u \sim \text{Uniform}(0, cg^*(x))$
  3. if $u \leq f^*(x)$ we accept x as the new sample, otherwise, reject x (discard it) and start with a new sample from $g(x)$.

An illustrative example

- Assume that it is difficult to sample from the Beta(3, 10) distribution (this is not the case of course!).
- We use the Uniform(0, 1) distribution with $g(x) = 1, \forall x \in [0, 1]$, which has the envelop property: $4g(x) > f(x), \forall x \in [0, 1]$. The following graph shows the result after 3000 iterations.



- Finding an appropriate distribution $g(x)$ becomes very difficult (and sometimes impossible) as the dimensionally of x increases, and it might not be efficient in general if there is a high rejection rate.

Importance sampling

- Importance sampling is used to find the expectation of a function $h(x)$ with respect to a distribution, with the density $f(x)$, from which we cannot directly sample.
- Assume again that we can sample from another distribution with the density $g(x)$ that is close to $f(x)$.
- Note that unlike the rejection sampling, we do not need the envelop property.
- The only requirement is that $g(x)$ must not be zero anywhere that $f(x)$ is not zero.
- As before, we only need to know $f(x)$ and $g(x)$ up to a constant.

Importance sampling

- Now we can write $E_f(h(x))$ as follows:

$$\begin{aligned}
\mu = E_f(h(x)) &= \int_{\mathcal{X}} h(x) f(x) dx \\
&= \int_{\mathcal{X}} h(x) \frac{f(x)}{g(x)} g(x) dx \\
&= \int_{\mathcal{X}} [h(x) w(x)] g(x) dx \\
&= E_g(h(x) w(x))
\end{aligned}$$

Importance sampling

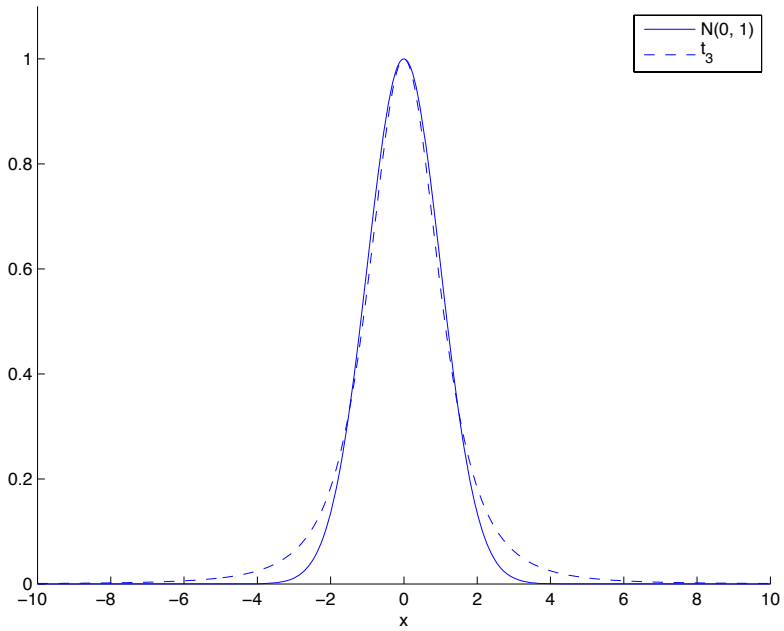
- We can then approximate the original expectation as follows:
  1. draw samples $x^{(1)}, ..., x^{(m)}$ from $g(x)$
  2. Find the *importance weight* $w^{(j)} = \frac{f(x^{(j)})}{g(x^{(j)})}$, where $j = 1, ..., m$
  3. Approximate the original expectation, $\mu = E_f(h(x))$, as follows

$$\hat{\mu} = \frac{[w^{(1)}h(x^{(1)}) + ... + w^{(m)}h(x^{(m)})]}{w^{(1)} + ... + w^{(m)}}$$

- In general, $f(x)$ and $g(x)$ do not need to be normalized. We only need to known them up to a constant. Whatever those constants are, they will be canceled out from the numerator and denominator.

An illustrative example

- We want to approximate a N(0, 1) distribution with $t(3)$ distribution:



- We use the unnormalized forms where $f(x) = \exp(-\frac{x^2}{2})$ and $g(x) = (1 + \frac{x^2}{3})^{-2}$.

- We generated 500 samples and estimated $\mu = E(x^2)$ as 0.97, which is close to the true value 1.

Potential problems

- The efficiency of this approach depends on how good $g(x)$ approximates $f(s)$.
- If the samples do not include the areas where f is large, or they include only a few samples from the high probability region, the estimation would not be accurate.
- To see this, as an exercise, repeat the above example, but this time approximate $t(3)$ with $N(0, 1)$.
- The estimate of $E(x^2)$ this time would be systematically smaller than the true value 3.

Possible difficulties and improved methods

- Instead of the methods discussed so far, we can use a Markov chain process to generate samples (which would not be independent anymore) and approximate the target distribution. This method is known as Markov chain Monte Carlo (MCMC) technique.

- However, we first need to discuss Markov chains and stochastic processes in general.

Stochastic processes; random walk

- Stochastic processes are dynamic random variables. They are indexed (usually by time), $\{X_n\}$. A discrete time, time-homogeneous stochastic process is simply a sequence of random variables, $X_0, X_1, ..., X_n$ defined on the same probability space.

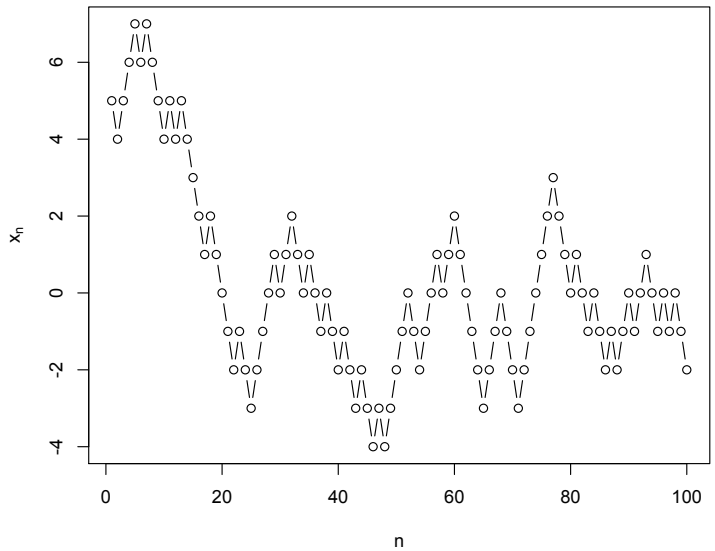
- One of the simplest stochastic processes (and one of the most useful ones) is the simple random walk.

- Consider a sequence of iid random variables $\{Z_i\}$ such that $P(Z_i = 1) = p$ and $P(Z_i = -1) = 1 - p = q$. The stochastic process represented by $\{X_n\}$ where $X_0 = a$ and $X_n = a + Z_1 + ... + Z_n$ is called a random walk process.

- This stochastic process could be interpreted as a gambler's fortune (in dollars) at time n, where the gambler starts with a dollars, makes \$1 bets repeatedly, and the probability of winning is p.

Random walk

- A random walk with $a = 5$ and $p = 0.4$.



- To obtain the distribution of such process note that $P(X_n = a + k) = 0$ unless $-n \leq k \leq n$ and $n + k$ is even. To get to $a + k$ after n steps, we need $\frac{n+k}{2}$ of $Z = 1$ and $\frac{n-k}{2}$ of $Z = -1$. Therefore:

$$P(X_n = a + k) = \binom{n}{\frac{n+k}{2}} p^{\frac{n+k}{2}} q^{\frac{n-k}{2}}$$

Discrete time, discrete space, time-homogenous Markov chains

- The above simple random walk is a special case of another well-known stochastic process called *Markov chains*.

- A Markov chain represent the stochastic movement of some particle from one state in the space S to another state in S. The particle initially starts from state i with probability $\pi_i^{(0)}$, and every time it is in state i, there is a p_{ij} chance it moves to state j in its next step.

- Therefore, a Markov chain has three main elements: 1- a state space S, which is a finite or countable set, 2- an initial distribution $\{\pi^{(0)}\}_i \in S$ which are non-negative numbers assigned to each state and they are summing to 1, and 3- transition probabilities $\{p_{ij}\}$ which are non-negative numbers representing the probability of going from state i to j, and $\sum_j p_{ij} = 1$.

Markov chains: An example

- For example, in the above simple random walk, we have
  - State space: $S = \mathcal{Z}$, the set of all integers
  - Initial distribution: $\pi^{(0)} = \delta_a$, i.e., a point mass at a such that $\pi_a = 1$ and $\pi_i = 0$, where $i \neq a$.
  - Transition probabilities: $p_{i,i+1} = p$, $p_{i,i-1} = 1 - p$ for all $i \in \mathcal{Z}$, and $p_{i,j} = 0$ when $j \neq i \pm 1$.

Markov chains: Formal definition

- A Markov chain is formally defined as a sequence of random variables, $X_0, X_1, ..., X_n$, representing states in the set S such that

$$P(X_0 = i_0, X_1 = i_1, ..., X_n = i_n) = \pi^{(0)}_{i_0} p_{i_0 i_1} ... p_{i_{n-1} i_n}.$$

- The above definition is based on the joint distribution, which can be used to derive the conditional distributions of the form

$$\begin{aligned} P(X_{k+1} = j | X_k = i) &= \frac{P(X_k = i, X_{k+1} = j)}{P(X_k = i)}, \quad P(X_k = i) > 0 \\ &= \frac{\sum_{i_0, i_1, ..., i_{k-1}} \pi^{(0)}_{i_0} p_{i_0 i_1} p_{i_1 i_2} ... p_{i_{k-1} i} p_{ij}}{\sum_{i_0, i_1, ..., i_{k-1}} \pi^{(0)}_{i_0} p_{i_0 i_1} p_{i_1 i_2} ... p_{i_{k-1} i}} \\ &= p_{ij} \end{aligned}$$

Markov property

- As we can see, this does not depend on k (i.e, it's time homogeneous).
- Also, it does not depend on the prior steps to get to $X_k = i$ (a.k.a the Markov property).
- This latter property of Markov process has been used as the formal definition of Markov process, however, we do not use this definition to avoid the complexity due to situations with $P(X_k = i) = 0$.

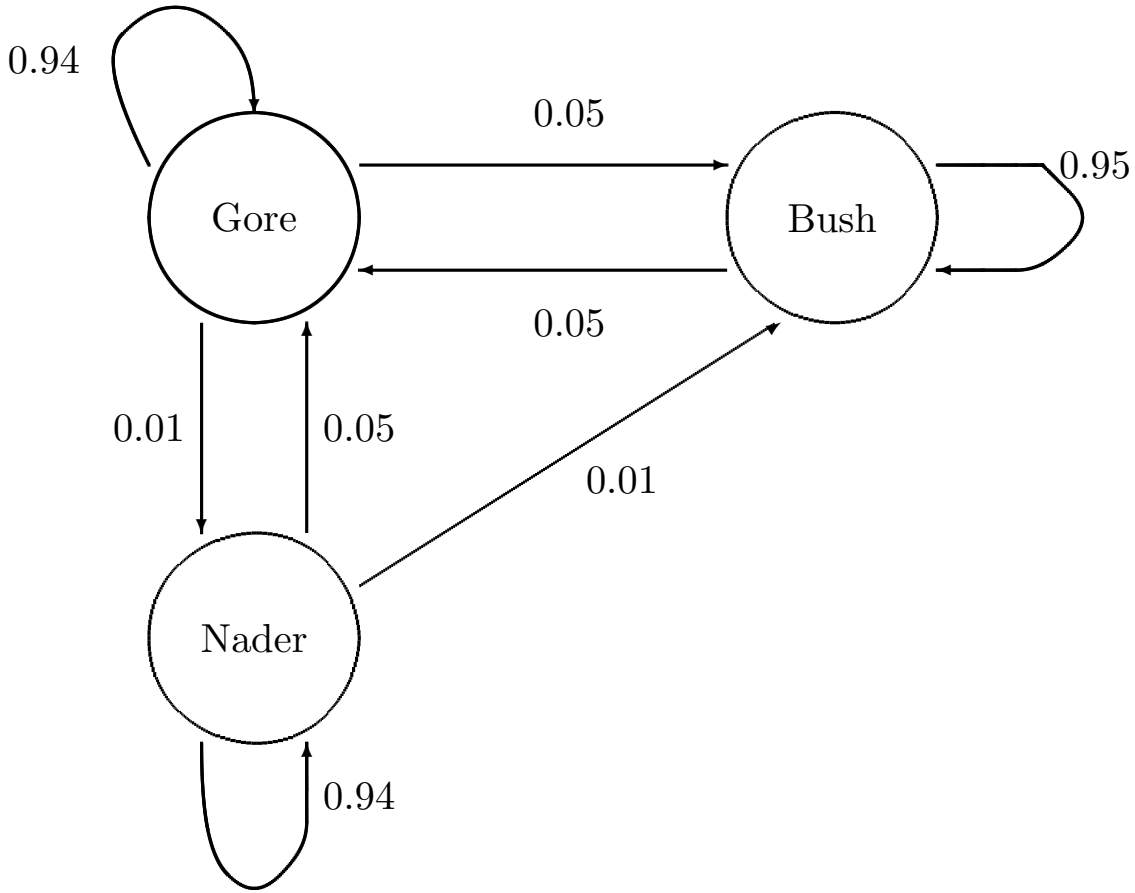
Example: 2000 presidential election

- Consider the 2000 US presidential election with three candidates: Gore, Bush and Nader (note that this is just an illustrative example and does not reflect the reality of that election).
- We assume that the initial distribution of votes (i.e., probability of winning) was $\pi = (0.49, 0.45, 0.06)$ for Gore, Bush and Nader respectively.
- Further, we assume the following transition probability matrix:

	Gore	*Bush*	*Nader*
Gore	0.94	0.05	0.01
Bush	0.05	0.95	0
Nader	0.05	0.01	0.94

Example: 2000 presidential election

- We can present the above Markov chain schematically as follows:



Example: 2000 presidential election

- If we represent the initial distribution $\pi^{(0)}$ by a row vector, and the transition probability by a square matrix P such that the element in row i and column j is p_{ij}, we can obtain the distribution of states in step n, $\pi^{(n)}$, as follows

$$\begin{aligned} \pi^{(1)} &= \pi^{(0)} P \\ \pi^{(2)} &= \pi^{(1)} P = \pi^{(0)} P^2 \\ &\dots \\ \pi^{(n)} &= \pi^{(0)} P^n \end{aligned}$$

- For the above example, we have

$$\begin{aligned} \pi^{(0)} &= (0.4900, 0.4500, 0.0600) \\ \pi^{(10)} &= (0.4656, 0.4655, 0.0689) \\ \pi^{(100)} &= (0.4545, 0.4697, 0.0758) \\ \pi^{(200)} &= (0.4545, 0.4697, 0.0758) \end{aligned}$$

Stationary distribution

- As we can see last, after some steps, the above Markov chain converges to a distribution, (0.4545, 0.4697, 0.0758), and does not moved afterwards.

- The chain would have reached this distribution (eventually) regardless of what initial distribution $\pi^{(0)}$ we chose. Therefor, $\pi = (0.4545, 0.4697, 0.0758)$ is the *stationary distribution* for the above Markov chain.

- Stationary distribution: A distribution of Markov chain states is called to be stationary if it remain the same in the next time step(s). That is, if the current distribution is $\{\pi_i\}$, we have $[\pi][p] = [\pi]$, where $[p]$ is the transition probability matrix. By induction, this also means $[\pi][p]^{(n)} = [\pi]$. In other words $\sum_i \pi_i p_{ij}^{(n)} = \pi_j$ for any $n \in \mathcal{N}$.

Stationary distribution

- How can we find out whether such distribution exists?
- Also, how do we know whether the chain would converge to this distribution?
- To find out the answer, we briefly discuss some properties of Markov chains.
- Finding the stationary distribution is an interesting problem on its own, but as we will see later, this would not be our concern in this course.

Recurrent vs. transient

- Recurrent states: a state i is called *recurrent* (or persistent) if starting from i, we will eventually visit i again with probability 1. All the states in the following MC are recurrent.

	Gore	*Bush*	*Nader*
Gore	0.94	0.05	0.01
Bush	0.05	0.95	0
Nader	0.05	0.01	0.94

- Transient state: a state i is called transient if starting from i, the probability of re-visiting it is less than 1. Nader in the following MC is a transient state:

	Gore	*Bush*	*Nader*
Gore	0.95	0.05	0
Bush	0.05	0.95	0
Nader	0.05	0.01	0.94

Irreducibility

- Irreducible: A Markov chain is irreducible if the chain can move from any state to another state. For example, the simple random walk is irreducible. The following chain is however reducible since Nader does not communicate with the other two states (Gore and Bush).

	Gore	*Bush*	*Nader*
Gore	0.95	0.05	0
Bush	0.05	0.95	0
Nader	0	0	1

Aperiodicity

- Period: the period of a state i is the greatest common divisor of the times at which it is possible to move from i to i. All the states in the following MC have period 3.

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

- Aperiodic: a Markov chain is said to be aperiodic if the period of each state is 1, otherwise the chain is periodic.

- Ergodic: a state is *ergodic* if it is aperiodic and recurrent. A Markov chain is said to be ergodic if all of its states are ergodic.

Reversibility (very important)

- Reversibility: a Markov chain is said to be *reversible* with respect to a probability distribution $\{\pi_i\}$ if $\pi_i p_{ij} = \pi_j p_{ji}$.

- We can show that if a Markov chain is reversible with respect to $\{\pi_i\}$, then $\{\pi_i\}$ is a stationary distribution:

$$\begin{aligned} \sum_i \pi_i p_{ij} &= \sum_i \pi_j p_{ji} \\ &= \pi_j \sum_i p_{ji} \\ &= \pi_j \end{aligned}$$

  since for all Markov chains $\sum_i p_{ji} = 1$.

- The above condition is also called *detailed balance*.

Discrete time, general space, time-homogeneous Markov chains

- We can define a Markov chain on a general state space $\mathcal{X}$ with initial distribution $\pi^{(0)}$ and transition probabilities $P(x, A)$ interpreted as when at point $x \in \mathcal{X}$, this is the probability of jumping to the subset A.

- As before, a Markov chain $X_0, X_1, ...$ is defined by

$$P(X_0 \in A_0, X_1 \in A_1, ..., X_n \in A_n) =$$
$$\int_{A_0} \pi^{(0)}(dx_0) \int_{A_1} P(x_0, dx_1) ... \int_{A_n} P(x_{n-1}, dx_n)$$

- As an example, consider a Markov chain the real line as its state space, $N(1, 1)$ as its initial distribution, and $P(x, .) = N(\frac{x}{2}, \frac{3}{4})$ as its transition probability.

Discrete time, general space, time-homogeneous Markov chains

- In this case, although we cannot talk about irreducibility and period as we discussed for discrete space since on a continuous the probability of visiting the same point is zero, we can still talk about irreducibility for sets A with non-zero measure, ϕ, on $\mathcal{X}$.

- *A chain in ϕ-irreducible if for all $A \subseteq \mathcal{X}$ with $\phi(A) > 0$ and for all $x \in \mathcal{X}$, there exists a positive integer n such that $P^n(x, A) > 0$.*

- Similarly, we need to modify our definition of period.

Discrete time, general state space, time-homogeneous Markov chains

- A distribution π is a stationary distribution if

$$\pi(A) = \int_{\mathcal{A}} \pi(dx)P(x, A)$$

- As for the discrete case, a continuous space is reversible with respect to π if

$$\pi(dx)P(x, dy) = \pi(dy)P(y, dx)$$

Discrete time, general state space, time-homogeneous Markov chains

- Also, if the chain is reversible with respect to π then, π is its stationary distribution:

$$\begin{aligned}
\int_{x\in\mathcal{X}} \pi(dx)P(x,dy) &= \\
\int_{x\in\mathcal{X}} \pi(dy)P(y,dx) &= \\
\pi(dy)\int_{x\in\mathcal{X}} P(y,dx) &= \\
\pi(dy)
\end{aligned}$$

- The above Markov chain with $N(1,1)$ initial distribution and $P(x,.) = N(\frac{x}{2},\frac{3}{4})$ transition probability converges to N(0, 1).

Main convergence theorem of Markov chains

- The main convergence theorem: If a Markov chain is ϕ-irreducible and aperiodic and has a stationary distribution π, then for all measurable $A \subseteq \mathcal{X}$, we have $\lim_{n\to\infty} P^n(x, A) = \pi(A)$.

- That is, regardless of the initial state, the chain converges to its stationary distribution.

- The convergence is in terms of the "total variation distance" defined between two probability measures as follows:

$$\|\nu_1(.) - \nu_2(.)\| = \sup_A |\nu_1(A) - \nu_2(A)|$$

- Therefore, the main convergence theorem indicates that

$$\lim_{n\to\infty} \|P^n(x, .) = \pi(.)\| = 0$$

Markov chain Monte Carlo

- Now suppose that we are interested in sampling from a distribution π (e.g., with density f, if it is continuous).
- Markov chain Monte Carlo is a method where we sample $x^{(1)}, x^{(2)}, \ldots$ from a Markov chain whose stationary distribution is π. We do this mainly by fixing π and finding an appropriate transition probability.
- We can then sample from m independent Markov chains and use Monte Carlo method for approximation.
- Alternatively (this is the usual application), we can run the chain for a long time (until it converges to π), discard the first s pre-convergence samples, and use the remaining m samples for Monte Carlo approximation.
- MCMC was first suggested by physicists Metropolis, Rosenbluth, Rosenbluth, Teller and Teller.

The Metropolis algorithm

- Suppose that we are interested in sampling from a distribution π. We known the density of π up to a constant, i.e., $cf(x)$.
- We can construct a Markov chain with a transition probability (a.k.a, *proposal distribution*) $g(x, y)$ which is symmetric; that is, $g(x, y) = g(y, x)$.
- For example, $N(x, 1)$ is symmetric since

$$\exp(-\frac{-(y-x)^2}{2}) = \exp(-\frac{-(x-y)^2}{2}).$$

The Metropolis algorithm

- Now follow these steps
  1. Given our current state $X^{(n)} = x$, we propose a new state $Y^{(n+1)} = y$ according to the transition probability (for example, if we chose $N(x, 1)$, we sample from a normal centered at x with variance 1).
  2. Calculated the acceptance probability
  $$a(x, y) = \min(1, \frac{f(y)}{f(x)})$$
  3. Accept the proposed state y as the new state with probability $a(x, y)$ or remain at state x. That is, sample $u \sim Unif(0, 1)$ and set
  $$X^{(n+1)} = \begin{cases} y & u < a(x, y) \\ x & \text{otherwise} \end{cases}$$
- Caution! when you reject the proposed state, you consider the current state as a new sample (i.e., you might have multiple copies of the same state x in your final sample).

The Metropolis algorithm

- How do we know that the above chain is going to converge to π?
- We use reversibility to prove this:

$$
\begin{aligned}
\pi(dx)P(x,dy) &= [f(x)dx][g(x,y)a(x,y)dy] \\
&= f(x)g(x,y)\ \min(1, \frac{f(y)}{f(x)})\, dx\, dy \\
&= \min(f(x)g(x,y), f(y)g(x,y))\, dx\, dy \\
&= \min(f(x)g(y,x), f(y)g(y,x))\, dx\, dy \\
&= f(y)g(y,x)\min(1, \frac{f(x)}{f(y)})\, dx\, dy \\
&= [f(y)dy][g(y,x)a(y,x)dx] \\
&= \pi(dy)P(y,dx)
\end{aligned}
$$

The Metropolis-Hastings algorithm

- Hastings later on generalized the above algorithm by showing that symmetrical proposal distribution is not necessary if we change the acceptance probability to

$$a(x, y) = \min(1, \frac{f(y)g(y, x)}{f(x)g(x, y)})$$

- We can use a similar procedure as above to show that reversibility (detailed balance) is preserved.

Proposal distribution

- The choice of proposal distribution is important since it determines the speed of convergence to π and the efficiency of sampling.
- The proposal distribution could be independent of current state.
- For example, we might sample from $y|x \sim N(0, 100^2)$ regardless of where we are at any given time. Note that this is not a symmetric proposal.
- Alternatively, we can use a proposal that depends on our current state.
- For example, if at any time we are at point x, we propose our next step by sampling from $N(x, \delta^2)$, or Unifrom$(x - \delta, x + \delta)$.
- Finding a good δ is sometimes challenging.

Example 1: Normal model with known variance

- Recall the univariate normal model with known variance

$$\begin{aligned} y &\sim N(\theta, \sigma^2) \\ P(y|\theta,\sigma) &= \prod_i^n \frac{1}{\sqrt{2\pi}\sigma} \exp[-\frac{(y-\theta)^2}{2\sigma^2}] \end{aligned}$$

- We used a conjugate $N(\mu_0, \tau_0^2)$ prior for θ, and showed that the posterior distribution, $P(\theta|y)$, has a closed form and is in fact a normal distribution.

- Now let's not use the closed form and sample from the posterior distribution using a Markov chain.

Example 1: Normal model with known variance

- We can of course write the posterior distribution up to a constant:

$$\begin{aligned} P(\theta|y) &\propto \exp\left(-\frac{(\theta-\mu_0)^2}{2\tau_0^2}\right)\prod_i^n \exp\left[-\frac{(y-\theta)^2}{2\sigma^2}\right] \\ &= f(\theta) \end{aligned}$$

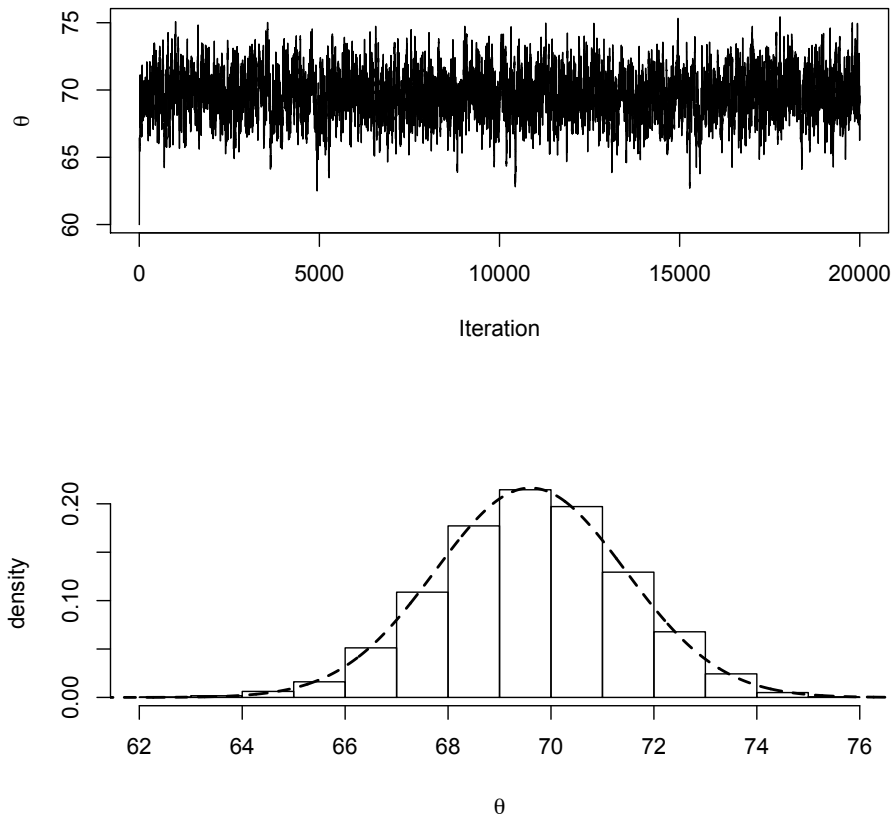
- To use the Metrolpolis algorithm, we need a symmetric proposal distribution. Here, we use $N(\theta^{(i)}, 1)$, which is a normal distribution around our current point, to propose the next step.

- We then start from an initial point $\theta^{(0)}$ and propose the next step $\theta' \sim N(\theta^{(0)}, 1)$, we either accept this value with probability $a(\theta^{(0)}, \theta')$, or reject and stay where we are.

- We continue these steps for many iterations (see the provided R code for details).

Example 1: Normal model with known variance

- As we can see, the posterior distribution we obtain using the Metropolis algorithm is very similar to the one based on the closed form.



Trace plot and posterior distribution of θ.

Example 2: Binomial model with Beta prior

- Recall the binomial model:

$$P(y|n,\theta) = \binom{n}{y}\theta^y(1-\theta)^{n-y}$$

- Assuming the conjugate prior Beta(α,β) for θ, we saw that the posterior distribution is Beta$(\alpha+y,\beta+n-y)$.

- For the election example, we mentioned that out of 100 people surveyed, 39 said they are going to vote for A. We used a conjugate Beta(1, 1) prior and obtained Beta(40, 62) as the posterior distribution for θ.

- Now let's not use the closed form of the posterior distribution and use the Metropolis algorithm instead.

Example 2: Binomial model with Beta prior

- We first need to find the posterior distribution (up to a constant).
- The prior distribution is of course uniform: $P(\theta) = 1$.
- The likelihood is (i.e., based on the binomial sampling distribution)

$$P(y|\theta) \propto \theta^y (1-\theta)^{n-y}$$

where $n = 100$ and $y = 39$.

- Therefore, using the Bayes' theorem, the posterior is

$$\begin{aligned} P(\theta|y) &\propto P(\theta)P(y|\theta) \\ &\propto \theta^y (1-\theta)^{n-y} \\ &\propto \theta^{39}(1-\theta)^{61} \end{aligned}$$

Example 2: Binomial model with Beta prior

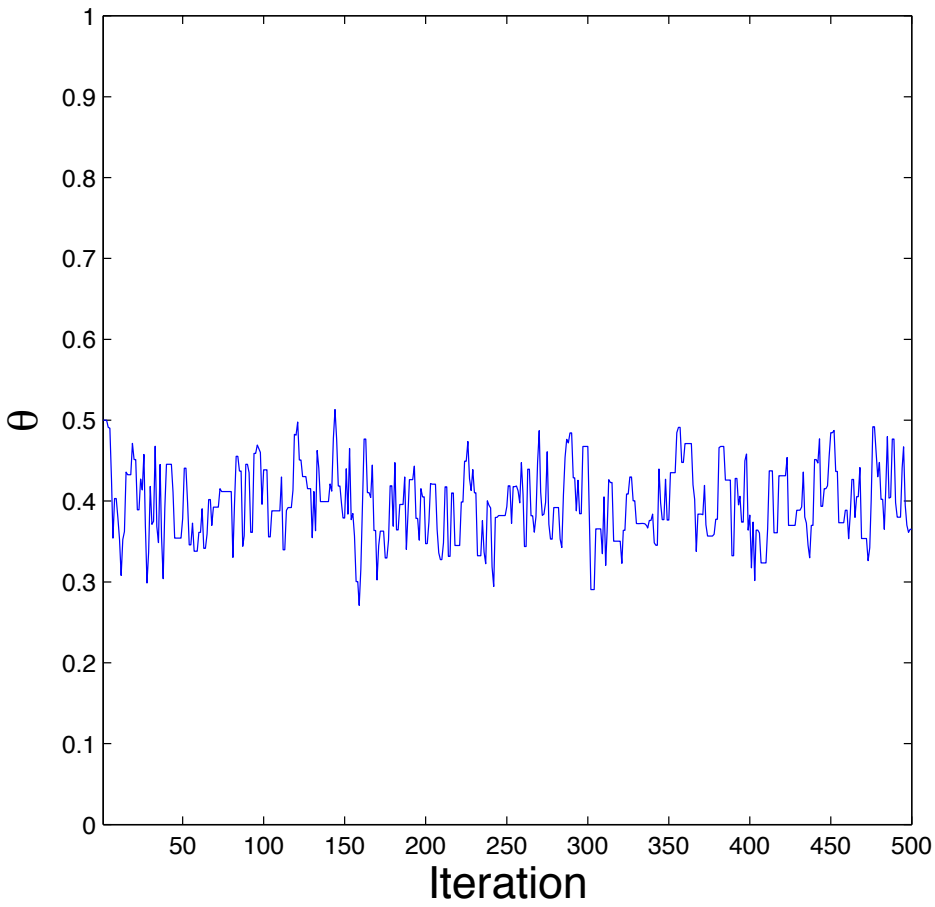
- Next, we need to choose a transition (i.e., proposal) distribution.
- Let's use Unifrom(0, 1). This is of course symmetric since $g(x, y) = g(y, x) = 1$.
- Now we start from $x_0 = 0.5$ and for $i = 1, ..., S$, we repeat the following steps:
  1. sample θ' from Uniform(0, 1).
  2. calculate the acceptance probability

$$a(\theta^{(i)}, \theta') = \min\left[1, \frac{(\theta')^{39}(1-\theta')^{61}}{(\theta^{(i)})^{39}(1-\theta^{(i)})^{61}}\right]$$

  3. Accept the proposed value with probability $a(\theta^{(i)}, \theta')$. For this, we can sample $u \sim Uniform(0, 1)$ and set

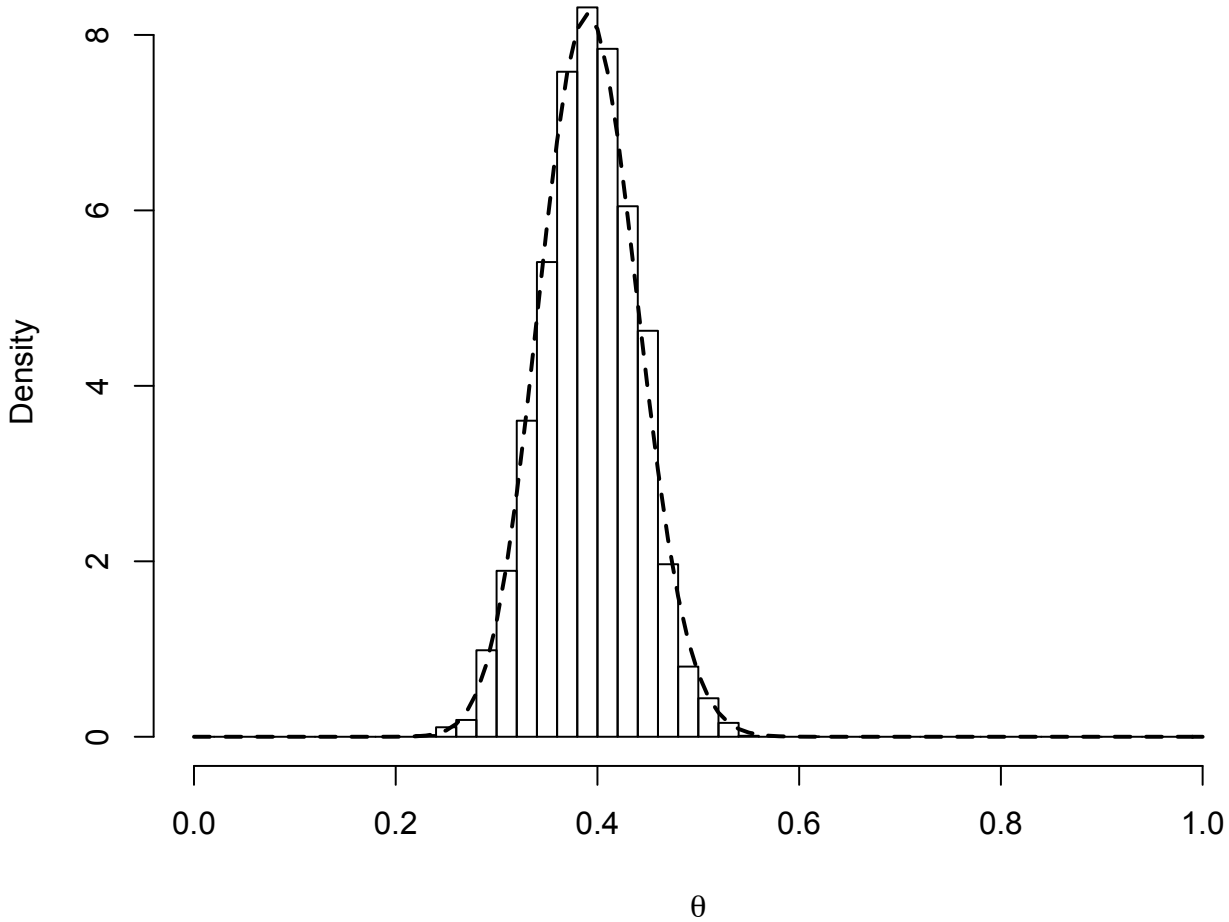
$$\theta^{(i+1)} = \begin{cases} \theta' & u < a(\theta^{(i)}, \theta') \\ \theta^{(i)} & \text{otherwise} \end{cases}$$

Example 2: Binomial model with Beta prior



Trace plot of samples from the posterior distribution of θ, the parameter of the binomial model.

Example 2: Binomial model with Beta prior



Posterior distribution of θ using the closed form (dashed line) and MCMC (histogram).

Example 3: Poisson model with Gamma prior

- Recall the Beckham's example. We modeled the number of goals (y_i) he scores in a game using a Poisson model

$$y_i \sim \text{Poisson}(\theta)$$

- He scored 0 and 1 goals in the first two games respectively.

- We used Gamma(1.4, 10) prior (Gamma(1.4, 0.1) if you are using MATLAB) for θ, and because of conjugacy, the posterior distribution also had a Gamma distribution

$$\theta|y \sim \text{Gamma}(2.4, 12)$$

- Again, let's ignore the closed form and use MCMC for sampling the posterior distribution.

Example 3: Poisson model with Gamma prior

- The prior is

$$P(\theta) \propto \theta^{0.4} \exp(-10\theta)$$

- The likelihood is

$$P(y|\theta) \propto \prod_{i=1}^{2} \theta^{y_i} \exp(-\theta)$$

  where $y_1 = 0$ and $y_2 = 1$.

- Therefore, the posterior is proportional to

$$\begin{aligned} P(\theta|y) &\propto \theta^{0.4} \exp(-10\theta) \prod_{i=1}^{2} \theta^{y_i} \exp(-\theta) \\ &= f(\theta) \end{aligned}$$

Example 3: Poisson model with Gamma prior

- For sampling from this posterior distribution, we can use the Metropolis algorithm with a symmetric proposal distribution such as Uniform$(\theta^{(i)} - \delta, \theta^{(i)} + \delta)$ or $N(\theta^{(i)}, \delta^2)$, where $\theta^{(i)}$ is our current state at iteration i, and δ is a constant.

- However, these proposals might not be very efficient since they could propose negative values which we know would be rejected for sure.

- Alternatively, we can use a non-symmetric proposal distribution such as Uniform$(0, \theta^{(i)} + \delta)$ and use the Metropolis-Hastings (MH) algorithm instead of Metropolis.

- Here, we set $\delta = 1$.

Example 3: Poisson model with Gamma prior

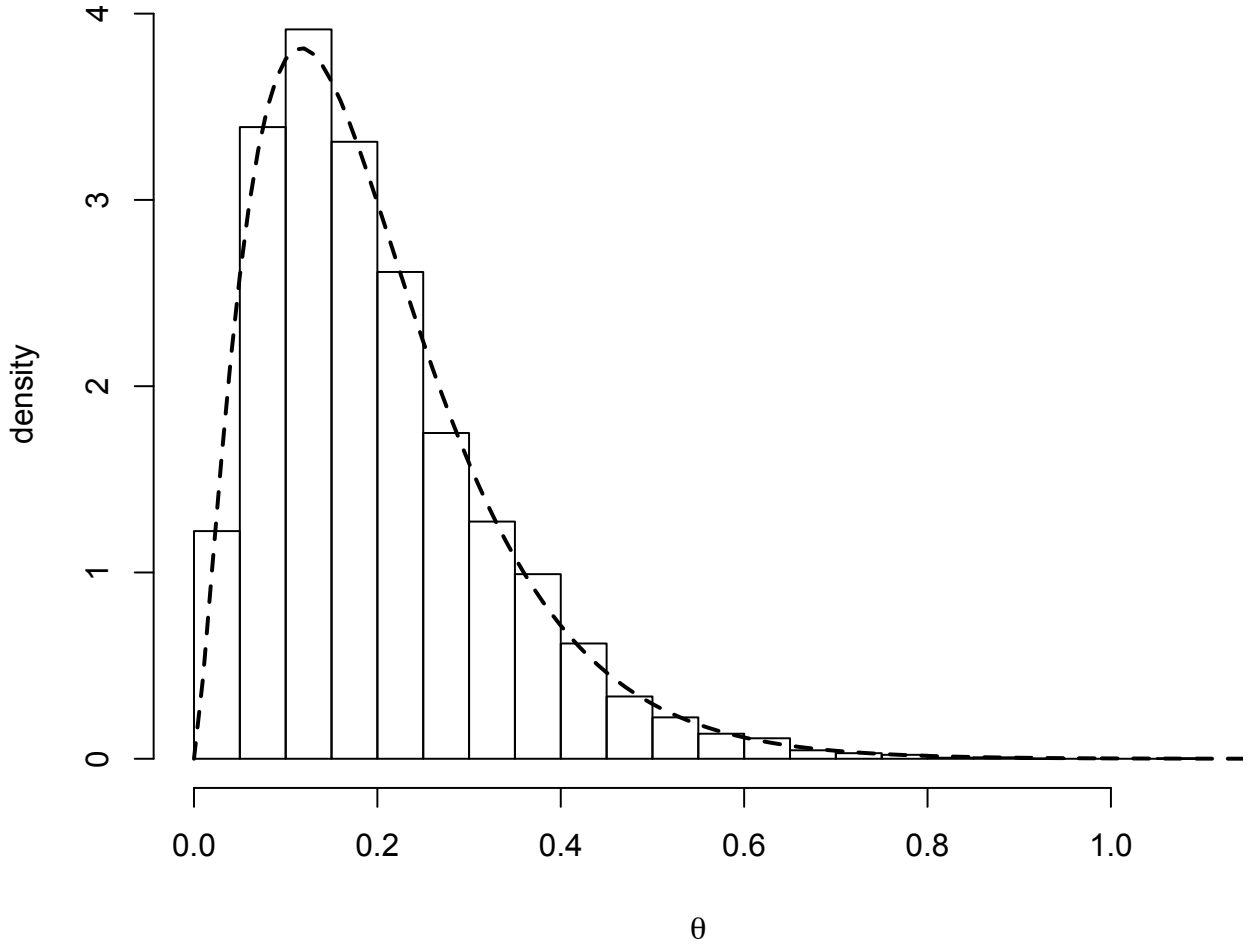
- For the MH sampling, start from $\theta_0 = 1$ and follow these steps for $i = 1, ..., S$:
  1. Sample θ' from Uniform$(0, \theta^{(i)} + 1)$.
  2. Calculate the acceptance probability

$$a(\theta^{(i)}, \theta') = \min\left[1, \frac{f(\theta')\,\text{Uniform}(\theta^{(i)}|0, \theta' + 1)}{f(\theta^{(i)})\,\text{Uniform}(\theta'|0, \theta^{(i)} + 1)}\right]$$

  3. Sample $u \sim Uniform(0, 1)$ and set

$$\theta^{(i+1)} = \begin{cases} \theta' & u < a(\theta^{(i)}, \theta') \\ \theta^{(i)} & \text{otherwise} \end{cases}$$

Example 3: Poisson model with Gamma prior



Posterior distribution of θ using the closed form (dashed line) and MCMC (histogram).

Multivariate distributions

- What if the distribution is multidimensional, i.e., $x = (x_1, x_2, ..., x_d)$.
- We can still use the Metropolis algorithm (or MH), with a multivariate proposal distribution, i.e., we now propose $y = (y_1, y_2, ..., y_d)$.
- For example, we can use a multivariate normal $N_d(x, \delta I_{d \times d})$, or a d-dimensional uniform distribution around the current point.

Example 4: Multivariate normal model

- Assume that we want to model the joint distribution of the expression levels for two genes, $y = (y_1, y_2)$, using a multivariate (bivariate in this case) normal distribution

$$y \sim N(\mu, \Sigma)$$

- For now, assume that the covariance is $\Sigma = I_2$.

- We assume a bivariate normal prior for μ

$$\mu = (\mu_1, \mu_2) \sim N(\mu_0, \Lambda_0)$$

and is the known covariance matrix. where $\mu_0 = (\mu_{01}, \mu_{02})$ and $\Lambda_0 = \sigma_0 I$.

Example 4: Multivariate normal model

- This is, of course, a conjugate prior, and we know that the posterior distribution of μ is also multivariate normal with mean and covariance

$$\begin{aligned} \mu_n &= (\Lambda_0^{-1} + n\Sigma^{-1})^{-1}(\Lambda_0^{-1}\mu_0 + n\Sigma^{-1}\overline{y}) \\ \Lambda_n^{-1} &= \Lambda_0^{-1} + n\Sigma^{-1} \end{aligned}$$

- If we ignore the conjugacy situation and wish to use the Metropolis (or MH) algorithm, we can use a multivariate (bivariate in this case) normal proposal distribution such as $N((\mu_1^{(i)}, \mu_2^{(i)}), 0.5I)$, which is a symmetric proposal.

- Everything else in the Metropolis algorithm remains as before.

Example 4: Multivariate normal model

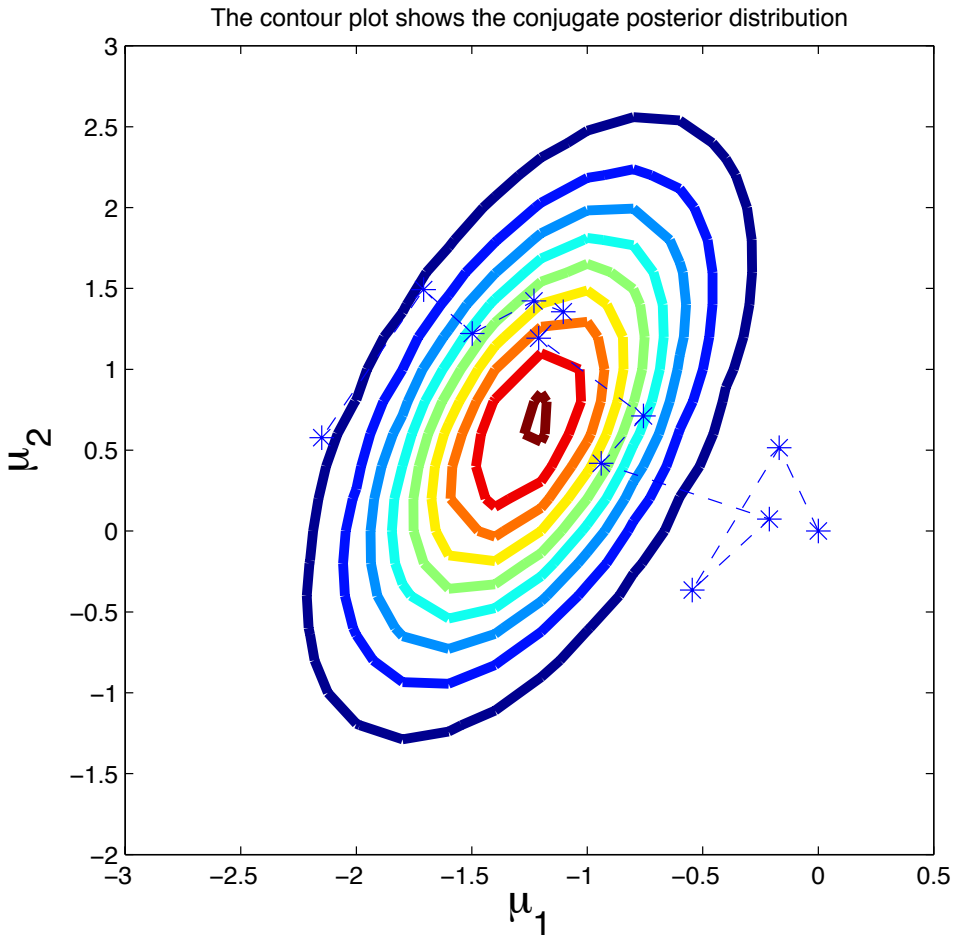
- Let's assume the prior is $N((0,0), 10I_2)$, and we have observed the following data, y:

Gene1	-1.2	-0.5	-2.1
Gene2	2.3	0.7	-1

- The posterior is

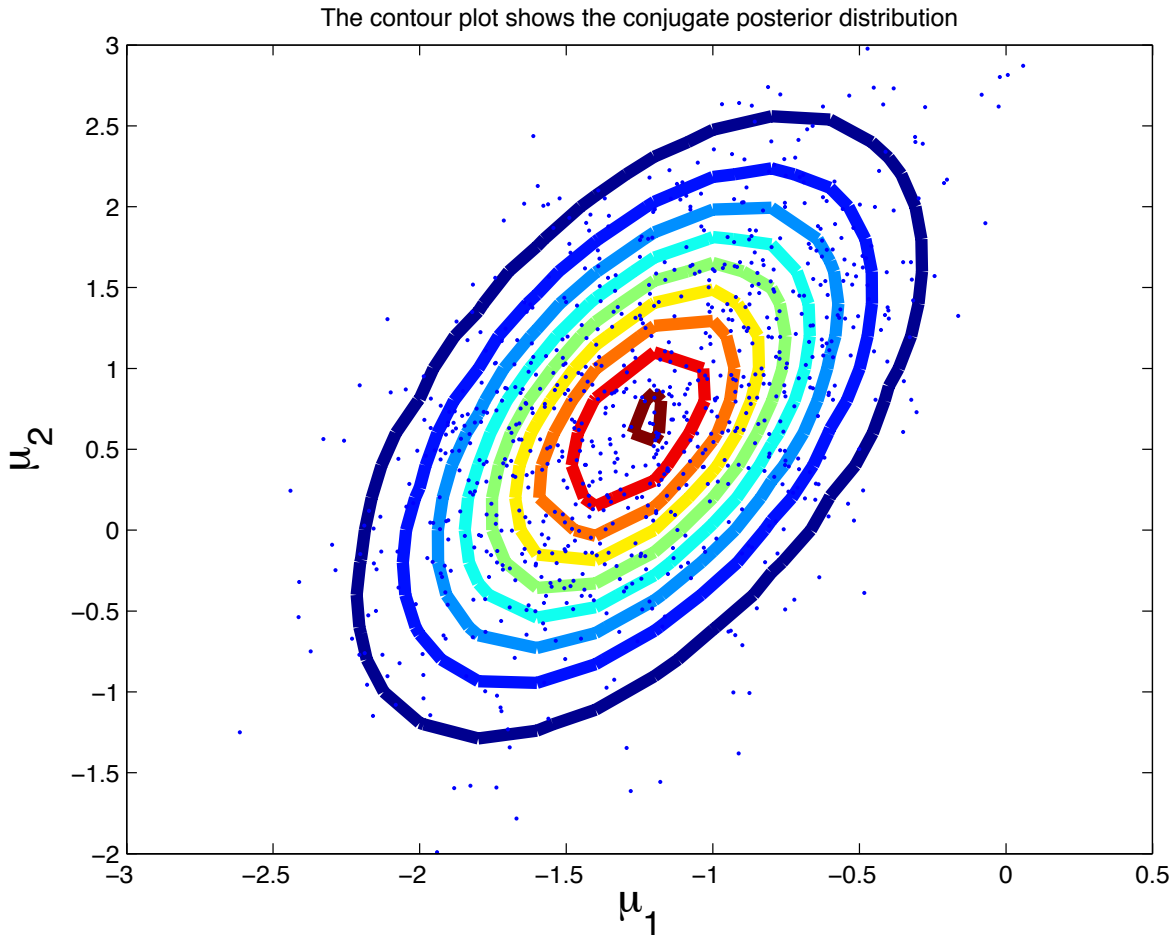
$$P(\mu|y, \Sigma) \propto N(\mu|(0,0), 10I_2)\, N(y|\mu, I_2)$$

Example 4: Multivariate normal model



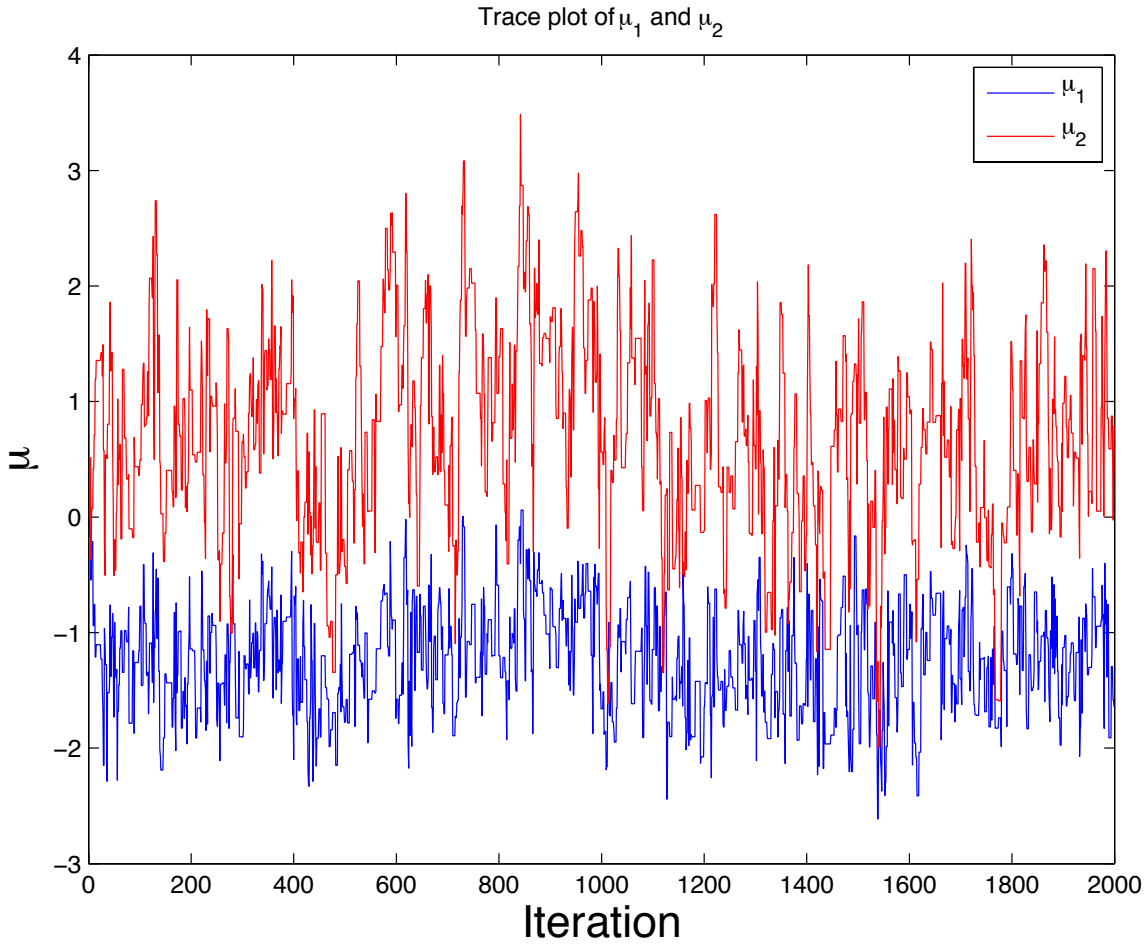
The first few samples from the posterior distribution of $\mu = (\mu_1 \mu_2)$, using a bivariate normal proposal. The contour plot shows the closed form of the posterior distribution.

Example 4: Multivariate normal model



Posterior samples for $\mu = (\mu_1 \mu_2)$, the mean of the bivariate normal model.

Example 4: Multivariate normal model



Trace plot of posterior samples for $\mu = (\mu_1 \mu_2)$, the mean of the bivariate normal model.

Decomposing the parameter space

- Sometimes, it is easier to decompose the parameter space into several components, and use the Metropolis (or MH) algorithm for one component at a time.
- Assume that we want to use the Metropolis algorithm for sampling from $P(x_1, x_2, ...x_d)$.
- To do this, we keep all but one component fixed at their current states, and use a univariate proposal distribution to update component.

Decomposing the parameter space

- At iteration i, given our current state $(x_1^{(i)}, x_2^{(i)}, ..., x_d^{(i)})$ we follow these steps:

1. Sample $Y_1^{(i+1)} = y_1$ from the univariate proposal distribution $g_1(x_1^{(i)}, y_1)$
2. Accept this new value and set $x_1^{(i+1)} = y_1$ with probability

$$a(x_1^{(i)}, y_1) = \min\left[1, \frac{p(y_1, x_2^{(i)}, ..., x_d^{(i)})}{p(x_1^{(i)}, x_2^{(i)}, ..., x_d^{(i)})}\right]$$

   or reject it and set $x_1^{(i+1)} = x_1^{(i)}$
3. Now sample $Y_2^{(i+1)} = y_2$ from the univariate proposal distribution $g_2(x_2^{(i)}, y_2)$.
4. Accept this new value for x_2 with probability

$$a(x_2^{(i)}, y_2) = \min\left[1, \frac{p(x_1^{(i+1)}, y_2, ..., x_d^{(i)})}{p(x_1^{(i+1)}, x_2^{(i)}, ..., x_d^{(i)})}\right]$$

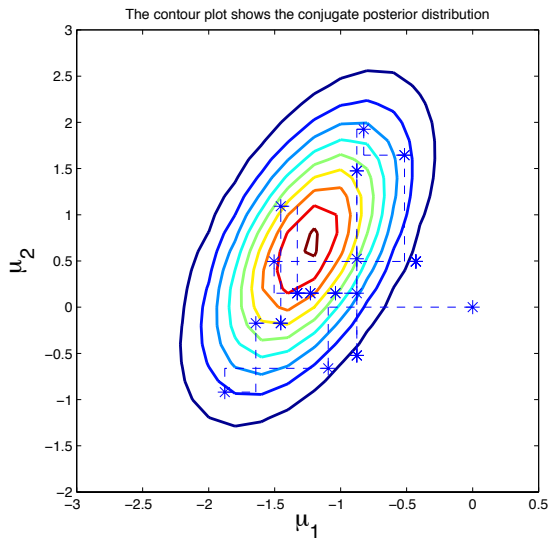
   or reject it and set $x_1^{(i+1)} = x_1^{(i)}$
5. Continue to update all d components.

Decomposing the parameter space

- Note that in general, we can decompose the space of random variable into blocks of components.
- Also, we can use transition distributions g_1, g_2, ... sequentially, or pick them randomly.
- As long as each transition probability individually leaves the target distribution invariant, their sequence would leave the target distribution invariant.
- In Bayesian model, this is especially useful if it is easier and computationally less intensive to evaluate the posterior distribution when one subset of parameters change at a time.

Example 4: Multivariate normal model

- In the example of multivariate normal with known covariance, we can sample μ_1 and μ_2 one at a time.
- The following figure shows the first few steps of sampling from a bivariate normal.



The first few samples from the posterior distribution of $\mu = (\mu_1, \mu_2)$, using a univariate normal proposal distribution sequentially.

Avoiding underflow

- To avoid *undreflow*, use the log transformation of posterior distribution, i.e.,

$$\log(P(\theta|y)) \quad = \quad \log[P(\theta)] + \log[P(y|\theta)] + c$$

And calculate the acceptance probability $a(\theta, \theta')$ in the Metropolis algorithm as

$$\min\left[1, \exp(\log[P(\theta')] + \log[P(y|\theta')] - \log[P(\theta)] - \log[P(y|\theta)])\right]$$

The Gibbs sampler

- As the dimensionality of the parameter space increases (sometimes to hundreds and thousands of components) it becomes difficult to find an appropriate proposal distributions (e.g., with appropriate step seize) for the Metropolis algorithm.

- If we are lucky (in many situations we are!), the conditional distribution of one component, x_j, given all other components, x_{-j} is tractable and has a closed form so we can directly sample from it.

The Gibbs sampler

- If that's the case, we can sample for each component one at a time using their corresponding conditional distributions $P(x_j|x_{-j})$.
- This is known as the Gibbs sampler or "heat bath" (Geman and Geman, 1984).
- Again, note that in Bayesian analysis, we are mainly interested in sampling from $P(\theta|y)$.
- Therefore, we use the Gibbs sampler when $P(\theta_j|y, \theta_{-j})$ has a closed form, e.g., there is a conditional conjugacy.
- For example, recall the univariate normal model. We saw that given σ (i.e., when it's fixed), the posterior distribution $P(\mu|y, \sigma^2)$ has a closed form (using the conjugate normal prior), and given μ, the posterior distribution of $P(\sigma^2|\mu, y)$ also has a closed form (using the conjugate Inv-χ^2 prior for σ^2 or using χ^2 prior for σ^{-2}).

The Gibbs sampler

- The Gibbs sampler works as follows:
- At iteration i, given our current state $(x_1^{(i)}, x_2^{(i)}, ..., x_d^{(i)})$ we cycle through the components one at a time:
  - Sample $x_1^{(i+1)}$ from the conditional distribution $P(x_1|x_2^{(i)}, x_3^{(i)}, ..., x_d^{(i)})$
  - Sample $x_2^{(i+1)}$ from the conditional distribution $P(x_2|x_1^{(i+1)}, x_3^{(i)}, ..., x_d^{(i)})$

    $\vdots$
  - Sample $x_j^{(i+1)}$ from the conditional distribution $P(x_j|x_1^{(i+1)}, ..., x_{j-1}^{(i+1)}, x_{j+1}^{(i)}, ..., x_d^{(i)})$

    $\vdots$
  - Sample $x_d^{(i+1)}$ from the conditional distribution $P(x_d|x_1^{(i+1)}, x_2^{(i+1)}, ..., x_{d-1}^{(i+1)})$

The Gibbs sampler

- Note that we are not just proposing anymore, we are directly sampling.
- Or you can look at it as a proposal that will be accepted with probability 1.
- Looking at it this way, the Gibbs sampler is a special case of the Metropolis-Hastings algorithm (note that the transition distribution is not symmetric)

$$
\begin{aligned}
a(x_j^{(i)}, x_j^{(i+1)}) &= \min\left[1, \frac{p(x_j^{(i+1)}, x_{-j}^{(i)})p(x_j^{(i)}|x_{-j}^{(i)})}{p(x_j^{(i)}, x_{-j}^{(i)})p(x_j^{(i+1)}|x_{-j}^{(i)})}\right] \\
&= \min\left[1, \frac{p(x_j^{(i+1)}|x_{-j}^{(i)})p(x_{-j}^{(i)})p(x_j^{(i)}|x_{-j}^{(i)})}{p(x_j^{(i)}|x_{-j}^{(i)})p(x_{-j}^{(i)})p(x_j^{(i+1)}|x_{-j}^{(i)})}\right] \\
&= 1
\end{aligned}
$$

Example 5: univariate normal model

- We can now use the Gibbs method to simulate samples from the posterior distribution of the parameters of a univariate normal $y \sim N(\mu, \sigma^2)$ model, where

$$\begin{aligned} \mu &\sim N(\mu_0, \tau_0^2) \\ \sigma^2 &\sim \text{Inv-}\chi^2(\nu_0, \sigma_0^2) \end{aligned}$$

- Given $[\sigma^{(i)}]^2$ at the i^{th} iteration, we sample $\mu^{(i+1)}$ from

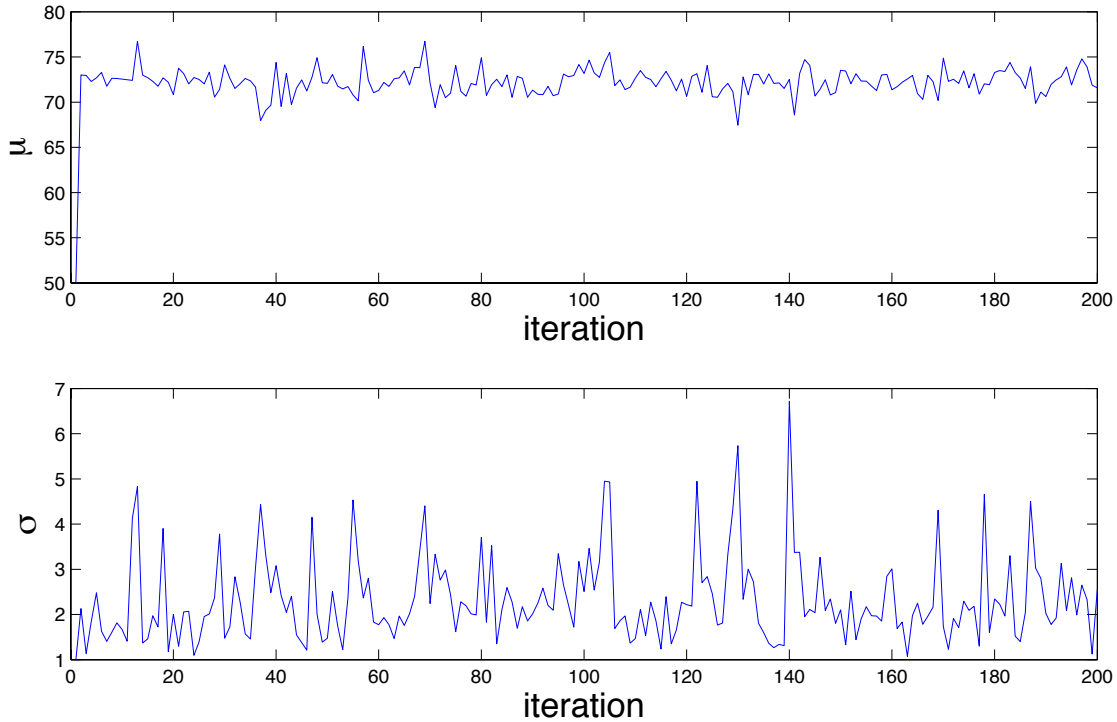
$$\mu^{(i+1)} \sim N\left(\frac{\frac{\mu_0}{\tau_0^2} + \frac{n\bar{y}}{[\sigma^{(i)}]^2}}{\frac{1}{\tau_0^2} + \frac{n}{[\sigma^{(i)}]^2}}, \frac{1}{\frac{1}{\tau_0^2} + \frac{n}{[\sigma^{(i)}]^2}}\right)$$

- Given $\mu^{(n+1)}$, we sample a new σ^2 from

$$[\sigma^{(i+1)}]^2 \sim \text{Inv-}\chi^2\left(\nu_0 + n, \frac{\nu_0\sigma_0^2 + \nu n}{\nu_0 + n}\right) \qquad \nu = \frac{1}{n}\sum_{i=1}^{n}(y_i - \mu^{(i+1)})^2$$

Example 5: univariate normal model

- The following graphs show the trace plots of posterior samples based on student heights example:



Trace plots of μ and σ, the mean and standard deviation of the normal model.

Combining Metropolis with Gibbs

- For more complex models, we might have conditional conjugacy for some parameters and not for the others.
- In such situations, we can combine the Gibbs sampler with the Metropolis method.
- That is, we update the components with conditionally conjugate priors using the Gibbs sampler and update the rest with Metropolis (or MH) steps.

Slice sampling

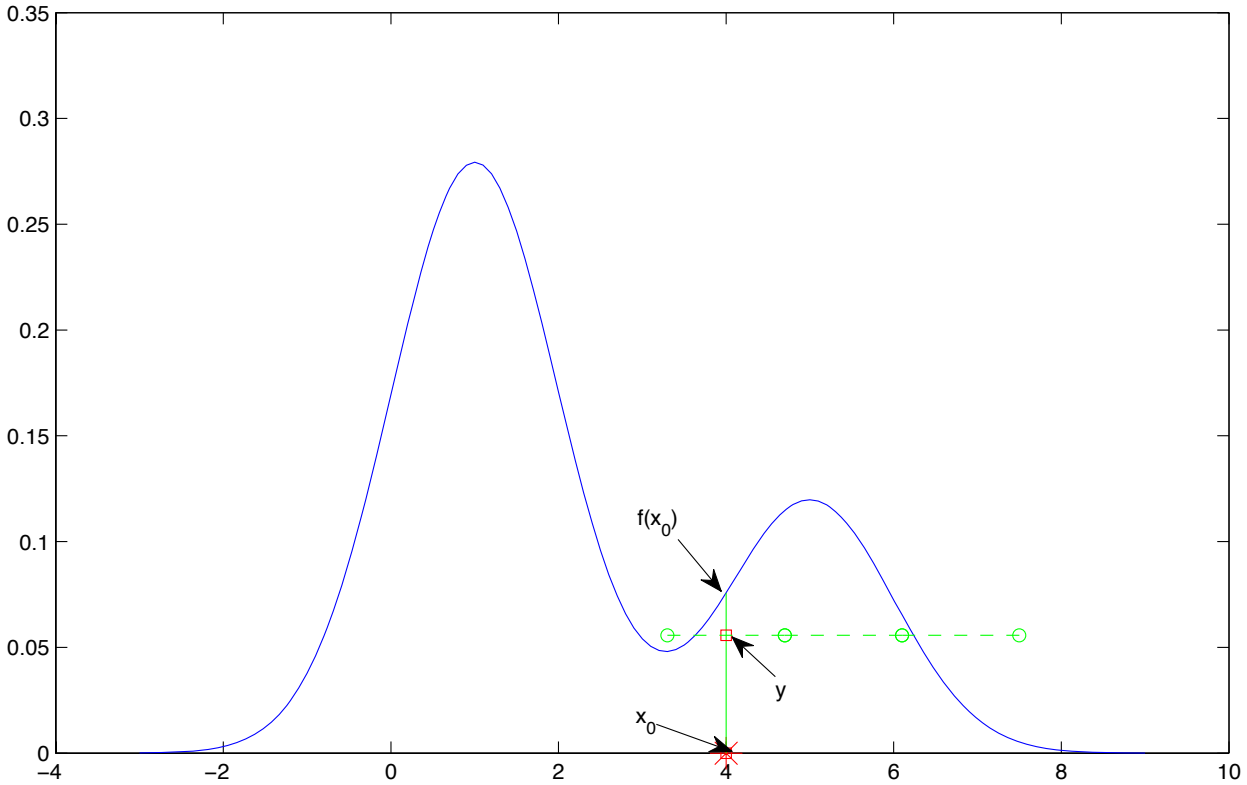
- Slice sampling was introduced by Neal (2003) as an improvement to Metropolis by reducing the sensitivity to the choice of step size.
- On the other hand, it is similar to the Gibbs sampler since it always produces samples that are accepted. However, unlike the Gibbs method, it does not require a closed form for the conditional distribution of one parameter given all others.
- This method is based on the idea that in order to sample from a distribution, we can sample uniformly from the region under the corresponding density function, $f(x)$.
- For this purpose, slice sampling alternates between two steps. Given the current state of the Markov chain, x, we uniformly sample a new point, y, from the interval $[0, f(x)]$. Next, given the current value of y, we uniformly sample from the region $S = \{x : y < f(x)\}$, which is referred to as the "slice" defined by y.

Slice sampling

- Since sampling an independent point uniformly from S might be difficult, we can substitute this step by any update that leaves the uniform distribution over S invariant.
- There are several methods to perform this task.
- Here, we discuss a simple but effective procedure that consists of two phases: "stepping-out" procedure for finding an interval around the current point, and "shrinkage" for sampling from the interval obtained.
- For a detail description of these methods see Neal (2003).

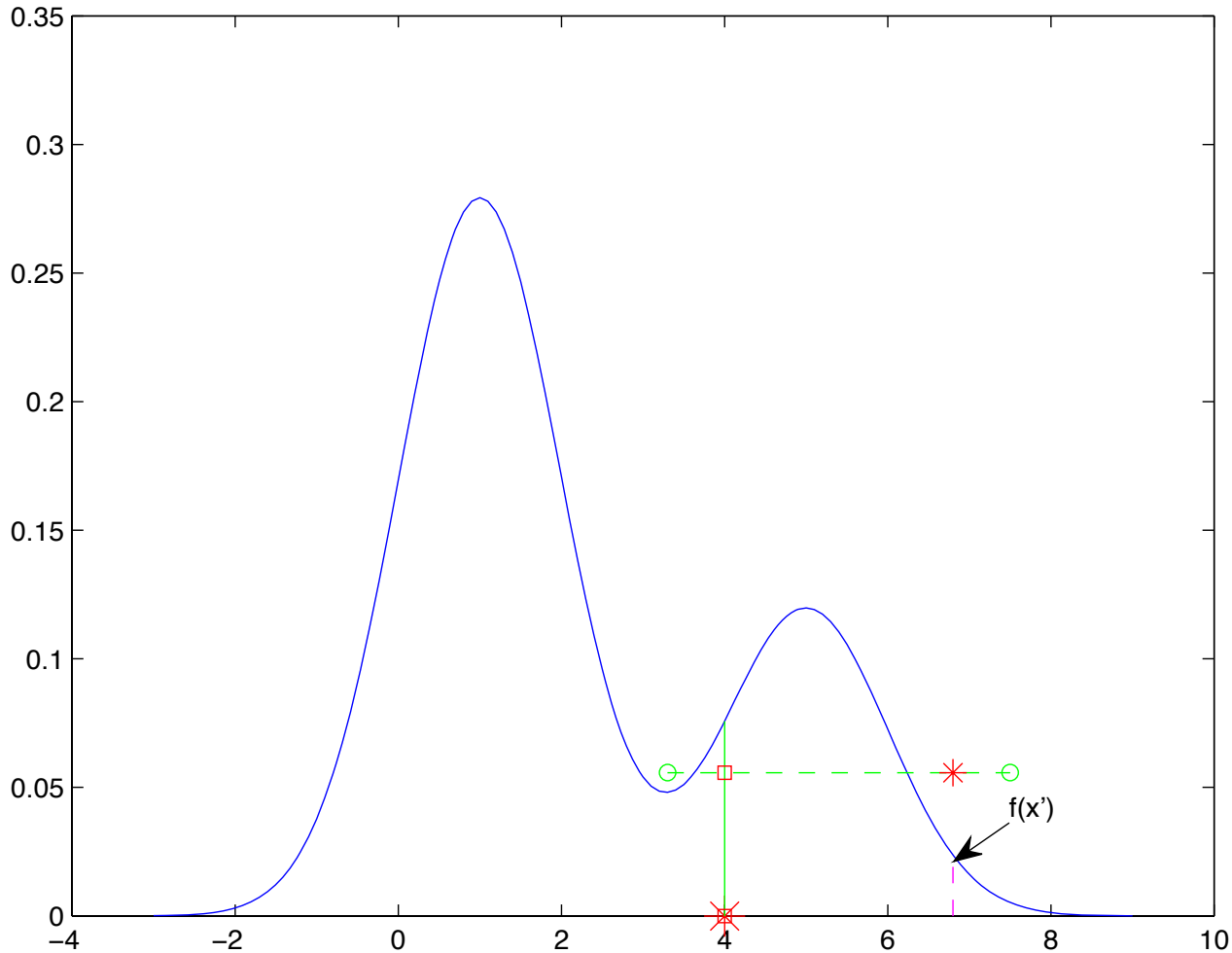
Slice sampling- Illustration

- Sampling $y \sim \text{Uniform}(0, f(x_0))$ and stepping out (of size w) until we reach points outside the area under the density.



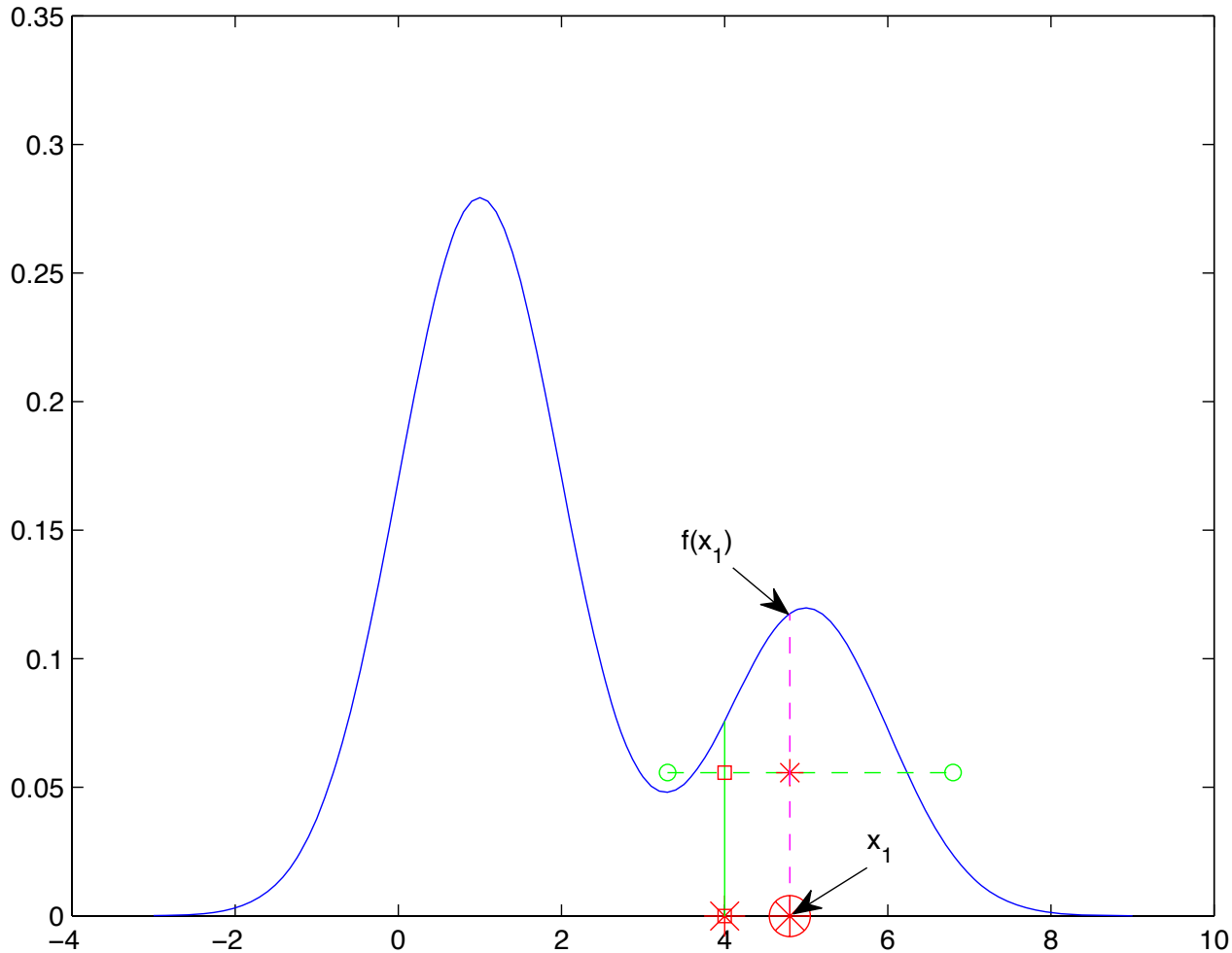
Slice sampling- Illustration

- Shrinkage of interval to a point, x', which is sampled (uniformly) from the interval but it has $f(x') < y$.



Slice sampling- Illustration

- Continuing shrinkage until we reach a point x_1 such that $y < f(x_1)$. We accept x_1 as our new sample.



Slice sampling- Stepping out

- The following figure (provided in Neal, 2003), shows the stepping out procedure to create an interval $[L, R]$ around our current point x_0, with $y \sim \text{Uniform}(0, f(x_0))$.

Input: f = function proportional to the density

x_0 = the current point

y = the vertical level defining the slice

w = estimate of the typical size of a slice

m = integer limiting the size of a slice to mw

Output: (L, R) = the interval found

$U \sim \text{Uniform}(0, 1)$
$L \leftarrow x_0 - w * U$
$R \leftarrow L + w$
$V \sim \text{Uniform}(0, 1)$
$J \leftarrow \text{Floor}(m * V)$
$K \leftarrow (m - 1) - J$

repeat while $J > 0$ and $y < f(L)$:

$L \leftarrow L - w$
$J \leftarrow J - 1$

repeat while $K > 0$ and $y < f(R)$:

$R \leftarrow R + w$
$K \leftarrow K - 1$

The stepping out procedure to create interval $[L, R]$ around x_0.

Slice sampling- Shrinkage

- The following figure (provided in Neal, 2003) shows how we can sample a new point from the interval $[L, R]$ around x_0.

Input: f = function proportional to the density

x_0 = the current point

y = the vertical level defining the slice

(L, R) = the interval to sample from

Output: x_1 = the new point

$\bar{L} \leftarrow L, \quad \bar{R} \leftarrow R$

Repeat:

$U \sim \text{Uniform}(0, 1)$

$x_1 \leftarrow \bar{L} + U * (\bar{R} - \bar{L})$

if $y < f(x_1)$ and Accept (x_1) then

exit loop

if $x_1 < x_0$ then $\bar{L} \leftarrow x_1$

else $\bar{R} \leftarrow x_1$

The shrinkage procedure to create a new sample x_1.