Numerical Bayesian Techniques

outline

- How to evaluate integrals?
- Numerical integration
- Monte Carlo integration
- Importance sampling
- Metropolis algorithm
- Metropolis-Hastings algorithm
- Gibbs algorithm
- Convergence
- examples

How to evaluate integrals?

- In the previous lessons we have seen, how to choose priors and how to obtain the posterior distributions
- We, generally wish to evaluate some point estimates or predictive distributions based on the computed posterior
- This involves integration over some variables

- If the model is simple and nonhierarchical and involves conjugate distributions this may be simple.
- However, many cases are more complicated and it is difficult to evaluate the integrals analytically

Three cases of integrals

- the normalisation integral

$$\int_{\forall\theta} p(x \mid \theta)\, \pi(\theta)\; d\theta$$

Marginalisation integral

- Suppose $\theta = (\theta_1, \ldots, \theta_k)$ (multidimensional parameter)
- We have calculated $\pi(\theta \mid x)$ and we want to calculate the posterior for one parameter only

$$\pi(\theta_i \mid x) = \int \pi(\theta \mid x)\, d\theta_{-i},$$

- Where

$$\theta_{-i} = (\theta_1, \ldots, \theta_{i-1}, \theta_i, \ldots, \theta_k)$$

- This is a k-1 dimensional integration

- Expectation integral
  - That is, when we are trying to evaluate a point estimate

$$E(\theta \mid x) = \int \theta \, \pi(\theta \mid x) \, d\theta.$$

  - In many cases, it is very difficult or impossible to evaluate the integral analytically
  - This was one of the main problems with the Bayesian approach

Example: shape parameter of the Gamma distribution

$$f(x \mid \alpha, \beta) = \frac{\alpha^{\beta}}{\Gamma(\beta)} x^{\beta-1} e^{-\alpha x}$$

- Assume that α is known and we observe $x_1, \ldots, x_n$
- Take the prior as uniform distribution
- The posterior is proportional to:

$$\frac{\alpha^{n\beta}}{\Gamma(\beta)^n} \left(\prod_i^n x_i \right)^{\beta-1} e^{-\alpha \sum_i^n x_i}$$

- Difficult to have closed form itegrals over this integrand

Solution?

- Numerical integration techniques
- Suppose we wish to integrate

$$\int_0^\infty g(\theta)\, d\theta$$

- Various techniques exist.
- Simplest one: finite difference approximation

Finite difference approximation

- 1. Find a value of $\theta_{\max}$ beyond which $g(\theta)$ is negligible
- 2. Split $(0, \theta_{\max})$ into *N* equivalent intervals. Then

- Where $$\int_0^{\infty} g(\theta)\, d\theta \approx \sum_{i=1}^{N} g(i\,\delta\theta)\,\delta\theta$$

$$\delta\theta = \theta_{\max}/(N-1)$$

- Numerically inefficient since we need very large N for good approximation
- When we have multiparameters, you need to form grids
- Where the distributions are peaked you should use finer, that is nonregular grids!
- it gets too complicated to apply this method
- Alternative: Monte Carlo methods

What is sampling good for?

- MC techniques used for integration and optimization problems
  - Bayesian Inference and Learning
    - Normalization
    - Marginalization
    - Expectation
  - Statistical Mechanics
  - Optimization
  - Model Selection

The Monte Carlo Principle

- Draw an i.i.d set of samples $\{x^{(i)}\}_{i=1}^{N}$ from $p(x)$
- Approximate target density

$$p_N(x) = \frac{1}{N}\sum_{i=1}^{N}\delta_{x^{(i)}}(x)$$

- Approximate integrals

$$I_N(f) = \frac{1}{N}\sum_{i=1}^{N} f(x^{(i)}) \xrightarrow[N\to\infty]{a.s.} I(f) = \int_{\mathcal{X}} f(x)p(x)dx.$$

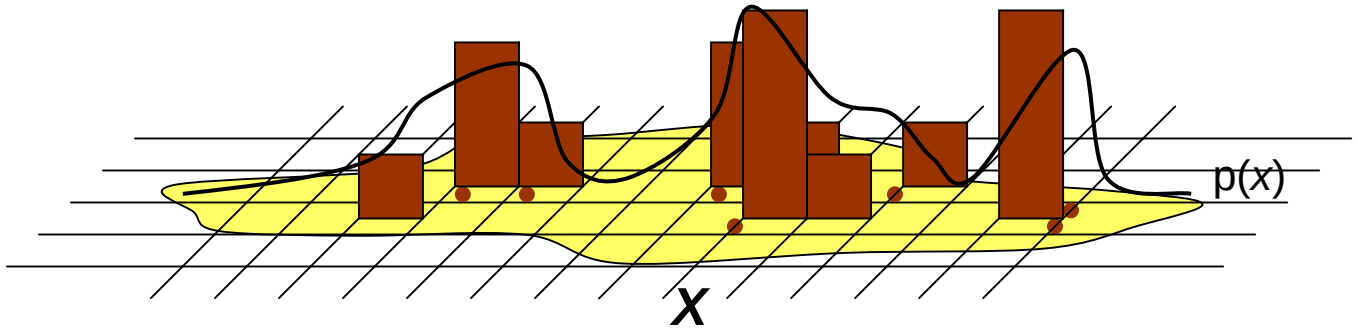
$$\mathrm{var}(I_N(f)) = \frac{\sigma_f^2}{N}$$

Monte Carlo methods

- We do not need to numerically calculate the posterior density function but try to generate values with its distribution.
- Then we use these values to approximate the density functions or estimates such as posterior means, variances, etc.
- Various ways:
  - Rejection sampling
  - Importance sampling

Monte Carlo principle

- Given a very large set X and a distribution $p(x)$ over it
- We draw i.i.d. a set of N samples
- We can then approximate the distribution using these samples



$$p_N(x) = \frac{1}{N}\sum_{i=1}^{N} 1(x^{(i)} = x) \quad \underset{N\to\infty}{\to} p(x)$$

Monte Carlo principle

- We can also use these samples to compute expectations

$$E_N(f) = \frac{1}{N}\sum_{i=1}^{N} f(x^{(i)}) \underset{N\to\infty}{\rightarrow} E(f) = \sum_x f(x)\,\mathrm{p}(x)$$

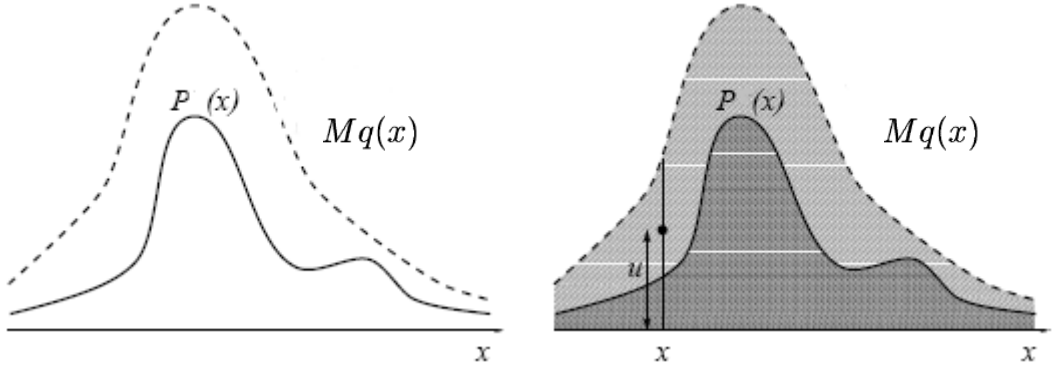
- And even use them to find a maximum

$$\hat{x} = \arg\max_{x^{(i)}}[\mathrm{p}(x^{(i)})]$$

Rejection sampling

- More generally, we would like to sample from p(x), but it's easier to sample from a *proposal distribution* q(x)
- q(x) satisfies p(x) < M q(x) for some $M<\infty$
- Procedure:
  - Sample $x^{(i)}$ from q(x)
  - Accept with probability p($x^{(i)}$) / Mq($x^{(i)}$)
  - Reject otherwise
- The accepted $x^{(i)}$ are sampled from p(x)!
- Problem: if M is too large, we will rarely accept samples
  - In the Bayes network, if the evidence $\boldsymbol{Z}$ is very unlikely then we will reject almost all samples

Rejection Sampling



Set $i = 1$

Repeat until $i = N$

1. Sample $x^{(i)} \sim q(x)$ and $u \sim \mathcal{U}_{(0,1)}$.
2. If $u < \frac{p\left(x^{(i)}\right)}{Mq\left(x^{(i)}\right)}$ then accept $x^{(i)}$ and increment the counter i by 1. Otherwise, reject.

Example

- Shape parameter of a Gamma distribution

$$\pi(\beta \mid x) \propto g(\beta) = \frac{\alpha^{n\beta}}{\Gamma(\beta)^n}\left(\prod_i x_i\right)^{\beta-1}, \quad 0 \le \beta \le \beta_{\max}$$

- Choosing uniform prior, the prior is bounded and on a finite interval
- We need to find a constant such that $g(\theta) \le C f(\theta)$
- In this case c is $c = \max g(\theta) / \beta_{\max} = 2.38 \times 10^{18}$

1. Generate a random number u_1. Proposal density is

$$f(y) = \frac{1}{1000},\ 0 \leq y \leq 1000.$$

Thus y is then $1000 \times u_1$ (by inverse transform).

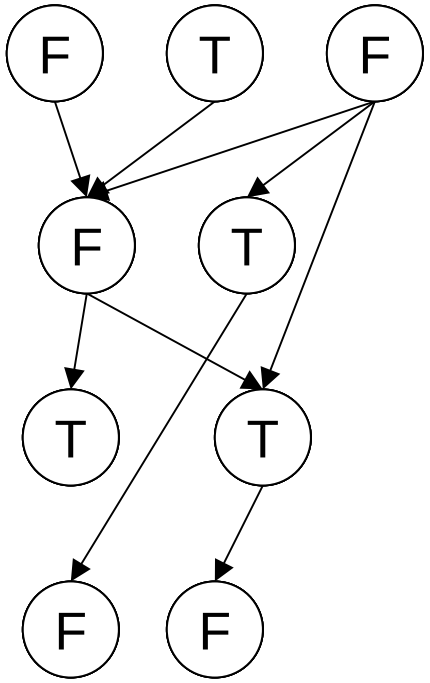
2. Generate a random number u_2. Accept y if

$$\frac{g(y)}{Cf(y)} \geq u_2$$

$$\frac{\frac{\alpha^{ny}}{\Gamma(y)^n}\left(\prod_i x_i\right)^{y-1}}{2.38 \times 10^{15}} \geq u_2$$

This is 500,000 proposals of which 557 (%0.11) are accepted.

Example: Bayes net inference



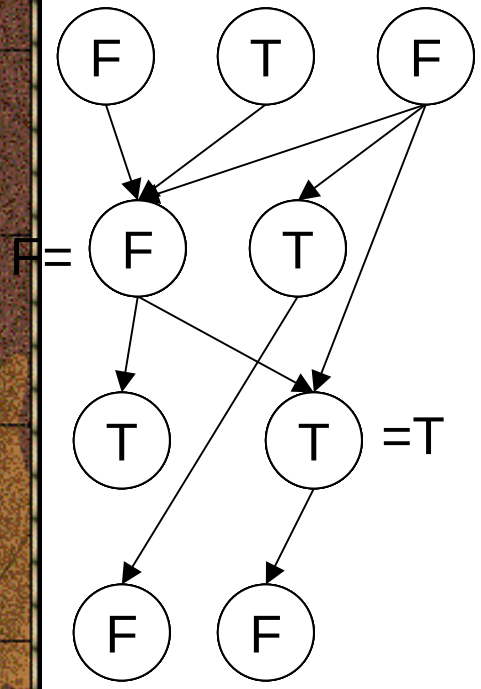
Sample 1: FTFTTTFFT

Sample 2: FTFFTTTFF

etc.

- Suppose we have a Bayesian network with variables ***X***
- Our state space is the set of all possible assignments of values to variables
- Computing the joint distribution is in the worst case NP-hard
- However, note that you can draw a sample in time that is linear in the size of the network
- Draw *N* samples, use them to approximate the joint

Rejection sampling



Sample 1: FTFTTTFFT reject

Sample 2: FTFFTTTFF accept

etc.

- Suppose we have a Bayesian network with variables ***X***
- We wish to condition on some evidence ***Z*** ϵ ***X*** and compute the posterior over ***Y***=***X***-***Z***
- Draw samples, rejecting them when they contradict the evidence in ***Z***
- Very inefficient if the evidence is itself improbable, because we must reject a large number of samples

Importance sampling

- The problem with the rejection sampling is that we have to be clever in choosing the proposal density!
- Importance sampling avoids difficult choices and generates random numbers economically.

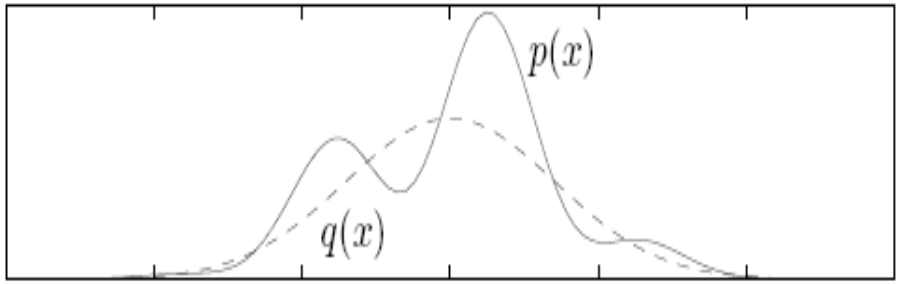
Importance Sampling

In importance sampling we generate N samples $\{x^{(i)}\}_{i=1}^{N}$ from q(x)

To account for the fact we sampled from the wrong distribution we introduce weights

$$w(x) \triangleq \frac{p(x)}{q(x)}$$

Then $$I(f) = \int f(x)\, w(x)\, q(x)\, dx$$



Monte Carlo estimate of ***I(f)***

$$\widehat{I}_N(f) = \sum_{i=1}^{N} f\left(x^{(i)}\right) w(x^{(i)})$$

Choose proposal distribution to minimize variance of the estimator

$$\mathrm{var}_{q(x)}\left(f(x)w(x)\right) = \mathbb{E}_{q(x)}\left(f^2(x)w^2(x)\right) - I^2(f)$$

Optimal proposal distribution $$q^{\star}(x) = \frac{|f(x)|p(x)}{\int |f(x)|p(x)dx}$$

Importance sampling

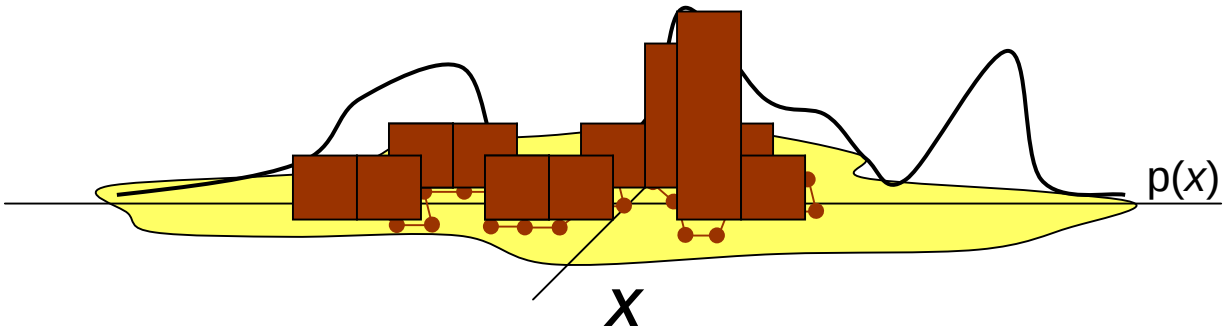
- Also called biased sampling
- It is a variance reduction sampling technique
- Introduced by Metropolis in 1953
- Instead of choosing points from a uniform distribution, they are now chosen from a distribution which concentrates the points where the function being integrated is large.

$$I = \int_a^b \frac{f(x)}{g(x)} g(x) dx$$

- Sample from g(x) and evaluate f(x)/g(x)
- The new integrand, *f*/*g*, is close to unity and so the variance for this function is much smaller than that obtained when evaluating the function by sampling from a uniform distribution. Sampling from a non-uniform distribution for this function should therefore be more efficient than doing a crude Monte Carlo calculation without importance sampling

Markov Chain sampling

- Recall again the set X and the distribution $p(x)$ we wish to sample from
- Suppose that it is hard to sample $p(x)$ and difficult to suggest a proposal distribution but that it is possible to "walk around" in X using only local state transitions
- Insight: we can use a "random walk" to help us draw random samples from $p(x)$



- That is, if our sampling follows a well defined structure, i.e. if our sample at one instant depends on our sampling the step before there are things we gain
- This is Markov Chain Monte Carlo Sampling and we will see how we benefit from a “random walk”
- MCMC theory basically says that if you sample using a Markov Chain, eventually your samples will be coming from the *stationary distribution* of the Markov Chain.

- The key to Markov Chain sampling's success is the fact that at every iteration you sample from a better distribution which eventually converges to your target distribution while in importance sampling, you always sample from the same (and wrong) distribution.
- We start by reviewing the Markov process theory.

MARKOV CHAINS

Markov chain is a process

We are dealing with a sequence of random variables

$$\{X_n\}_{n=0}^{\infty}$$

and give them a (parametric) statistical model, which permits dependencies between different random variables. Models like this are called **stochastic processes**. The index n is here taken to indicate 'time'.

Consider a set $S = \{E_1, E_2, \ldots, E_J\}$ and sequence of random variables $X_0, X_1, \ldots, X_n, \ldots$, assuming values in S. The symbols E_j are called *states* (and can designate all kinds of things) and S is also called the state space. We give the state E_j the label j and take for simplicity of typing $S = \{1, 2, \ldots, J\}$.

Markov Chain

A sequence of random variables $\{X_n\}_{n=0}^{\infty}$ is called a **Markov chain,(MC)**, if for all $n \geq 1$ and $j_0, j_1, \ldots, j_n \in S$,

$$P\left(X_n = j_n | X_0 = j_0, X_1 = j_1, \ldots, X_{n-1} = j_{n-1}\right) =$$

$$P\left(X_n = j_n | X_{n-1} = j_{n-1}\right).$$

■

The condition is known as the *Markov property*.

A.A. Markov 1856-1922

Russian
mathematician

Memory

The significance of an MC lies in the fact that if $X_n = j_n$ is a future event, then the conditional probability of this event given the past history $X_0 = j_0, X_1 = j_1, \ldots, X_{n-1} = j_{n-1}$ depends only upon the immediate past $X_{n-1} = j_{n-1}$ and not upon the remote past $X_0 = j_0, X_1 = j_1, \ldots, X_{n-2} = j_{n-2}$.

Other properties

The Markov property above is perhaps straightforward to state, since there is a natural order for integers n.

- There are various different Markov properties, e.g., for probabilities on directed acyclic graphs (Bayesian networks).
- There are definitions of Markov property for random fields, e.g., relevant (?) in image analysis.

Chain rule of probability

By iteration of the definition of conditional probability we get the following identity valid for any sequence of random variables

$$P\left(X_1 = x_{l_1}, \cdots, X_m = x_{l_m}\right) =$$

$$= \prod_{i=1}^{m} P\left(X_i = x_{l_i} \mid X_1 = x_{l_1} \ldots X_{i-1} = x_{l_{i-1}}\right)$$

where

$$P\left(X_1 = x_{l_1} \mid X_0 = x_{l_0}\right) = P\left(X_1 = x_{l_1}\right).$$

Modelling by chain rule

The model

$$P\left(X_1 = x_{l_1}, \cdots, X_m = x_{l_m}\right) =$$

$$= \prod_{i=1}^{m} P\left(X_i = x_{l_i} \mid X_1 = x_{l_1} \ldots X_{i-1} = x_{l_{i-1}}\right)$$

where

$$P\left(X_1 = x_{l_1} \mid X_0 = x_{l_0}\right) = P\left(X_1 = x_{l_1}\right).$$

is for obvious reasons unpractical: There will be too many parameters (conditional probabilities).

Let $\{X_n\}_{n=0}^{\infty}$ be Markov chain. If $X_n = j$, we say that the *the chain is in state j at time n* or that *the chain visits the state j at time n*. The conditional probabilities

$$p_{i|j} = P\left(X_n = j | X_{n-1} = i\right), n \geq 1, i, j \in S$$

are assumed to be independent of n (temporally homogeneous), i.e.,

$$p_{i|j} = P\left(X_1 = j | X_0 = i\right), i, j \in S$$

and are called *(stationary) one-step transition probabilities.*

Transition matrix

The numbers $p_{i|j}$ are taken as entries in a matrix

$$P = \left(p_{i|j}\right)_{i=1,j=1}^{J,J}$$

	to E_1	to E_2	$\dots$	to E_{J-1}	to E_J	
from E_1	$p_{1\|1}$	$p_{1\|2}$	$\cdots$	$p_{1\|J-1}$	$p_{1\|J}$	
from E_2	$p_{2\|1}$	$p_{2\|2}$	$\cdots$	$p_{2\|J-1}$	$p_{2\|J}$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\dots$	$\vdots$	$\vdots$
from E_{J-1}	$p_{J-1\|1}$	$p_{J-1\|2}$	$\cdots$	$p_{J-1\|J-1}$	$p_{J-1\|J}$	
from E_J	$p_{J\|1}$	$p_{J\|2}$	$\cdots$	$p_{J\|J-1}$	$p_{J\|J}$	

$$P = \left(p_{i|j}\right)_{i=1,j=1}^{J,J}$$

$$P = \begin{pmatrix} p_{1|1} & p_{1|2} & \cdots & p_{1|J} \\ p_{2|1} & p_{2|2} & \cdots & p_{2|J} \\ \vdots & \vdots & \vdots & \vdots \\ p_{J|1} & p_{J|2} & \cdots & p_{J|J} \end{pmatrix}.$$

Thus P is an $J \times J$ matrix to be called a *transition matrix*.

The $i:th$ row of P is the conditional probability distribution of X_{n+1} given that $X_n = i$ (or, as well, distribution of X_1 given that $X_0 = i$). Clearly the following properties hold true:

$$p_{i|j} \geq 0, \sum_{j=1}^{J} p_{i|j} = 1.$$

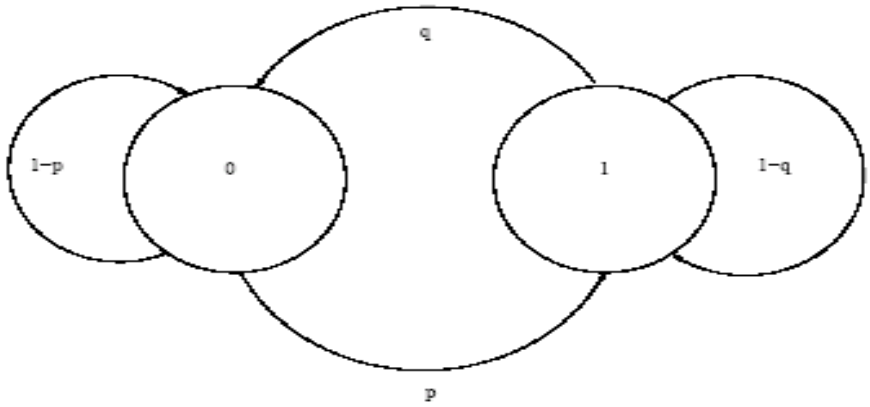
Example

A **binary Markov chain** has a state space designated by $\{0, 1\}$. If at some stage 0 is seen, then at the next stage 1 will be seen with probability p and 0 will be seen with probability $1 - p$. If a 1 is seen, then at a next stage 0 will be seen with probability q and 1 will be seen with probability $1 - q$. This corresponds to the transition matrix

$$P = \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix}$$

State transition graph

$$P = \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix}$$



The structure of a state transition graph without the probabilities is known as the **topology of the graph**.

Markov Chains of k-th order

A sequence of random variables $\{X_n\}_{n=0}^{\infty}$ is called a **k:th order Markov chain** , if for all $n \geq 1$ and $j_0, j_1, \ldots, j_n \in S$,

$$P\left(X_n = j_n | X_0 = j_0, X_1 = j_1, \ldots, X_{n-1} = j_{n-1}\right) =$$

$$= P\left(X_n = j_n | X_{n-k} = j_{n-k}, \ldots, X_{n-1} = j_{n-1}\right),$$

for a positive integer k.

The MC in our primary definition is called a first order Markov chain. An I.I.D process assuming values in S would in this respect be called a Markov chain of zero order.

Joint probability distribution of a Markov Chain

By successive iterations of the defi nition of conditional probability and by successive uses of the chain rule and of the Markov property

$$P(X_0 = j_0, \ldots, X_{n-1} = j_{n-1}, X_n = j_n) =$$

$$P(X_n = j_n | X_0 = j_0, \ldots, X_{n-1} = j_{n-1}) \cdot P(X_0 = j_0, \ldots, X_{n-1} = j_{n-1}) =$$

$$P(X_n = j_n | X_{n-1} = j_{n-1}) \cdot P(X_0 = j_0, \ldots, X_{n-1} = j_{n-1}) =$$

$$p_{j_{n-1}|j_n} \cdot P\left(X_{n-1} = j_n | X_0 = j_0, \ldots, X_{n-2} = j_{n-2}\right) \cdot P\left(X_0 = j_0, \ldots, X_{n-2} = j_{n-2}\right) =$$

$$\vdots$$

$$= p_{j_{n-1}|j_n} \cdot p_{j_{n-2}|j_{n-1}} \cdots \cdot p_{j_0|j_1} \cdot p_{X_0}\left(j_0\right) =$$

$$= p_{X_0}\left(j_0\right) \cdot p_{j_0|j_1} \cdot \cdots p_{j_{n-2}|j_{n-1}} \cdot p_{j_{n-1}|j_n} .$$

If $\{X_n\}_{n=0}^{\infty}$ is a Markov chain with stationary transition probabilities, then

$$P\left(X_0 = j_0, X_1 = j_1, \ldots, X_n = j_n\right) = p_{X_0}\left(j_0\right) \prod_{l=1}^{n} p_{j_{l-1}|j_l}.$$

This shows that the probabilistic properties of a Markov chain are completely determined by its one-step transition probability matrix and the probability distribution at 0.

Parametric statistical model

We write

$$\{X_n\}_{n=0}^{\infty} \sim \text{Markov}\left(P, p_{X_0}\right),$$

where

$$p_{X_0} = (p_1, \ldots, p_J).$$

This is a parametric model. One way to assign parameters is

$$\Theta = (P, p_{X_0})$$

by which we mean the $J \cdot J - J$ (free) parameters in P and the $J - 1$ parameters in p_{X_0}.

N-step transition probabilities

The conditional probabilities

$$p_{i|j}(n) = P\left(X_{m+n} = j | X_m = i\right), n \geq 1, i, j \in S$$

are also independent of m. The probabilities $p_{i|j}(n)$ are called the n *-step transition probabilities*. Then

$$P(n) = \left(p_{i|j}(n)\right)_{i=1,j=1}^{J,J}$$

is the $n-$*step transition matrix*. We define

$$p_{i|j}(0) = \begin{cases} 1 & \text{if } j = i \\ 0 & \text{if } j \neq i. \end{cases}$$

$$P(n) = \left(p_{i|j}(n)\right)_{i=1,j=1}^{J,J}$$

is the n *-step transition matrix*. Then

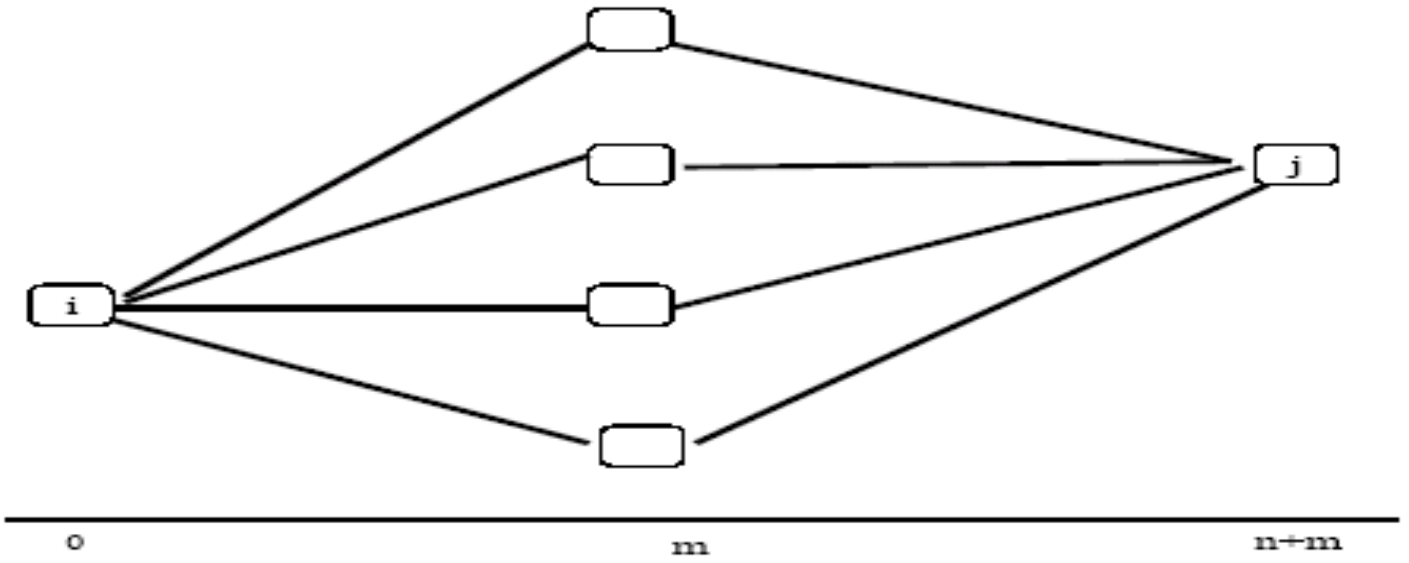
$$P(1) = P$$

as defined above.

Chapman-Kolmogorov equations

For all $m, n \geq 1$ and $i, j \in S$,

$$p_{i|j}(m+n) = \sum_{k=1}^{J} p_{i|k}(m) \cdot p_{k|j}(n).$$



In matrix form:

Using a matrix notation we can write the Chapman - Kolmogorov equation as the following matrix multiplication

$$P(n+m) = P(m) \cdot P(n).$$

Proof: straigthforward but lengthy. Look at Thomas Cover: *Elements of Information Theory*

Power

$$P(n) = P^n.$$

Proof: This is easily proved by induction. The case $n = 1$ follows by definitions

$$P(1) = P = P^1$$

Assume the claim holds for n, i.e., $P(n) = P^n$. Then by Chapman-Kolmogorov

$$P(n+1) = P \cdot P(n),$$

and by induction assumption

$$= P \cdot P^n = P^{n+1}$$

Chapman - Kolmogorov equation can be written as

$$P(n+m) = P^m \cdot P^n.$$

State probabilities

Let the distribution of X_0 be denoted by $\phi(0)$. In other words,

$$\phi(0) = (p_{X_0}(1), \ldots, p_{X_0}(J)).$$

This will be called the *initial distribution*. Let us denote by

$$\phi(n) = (p(X_n = 1), \ldots, p(X_n = J))$$

the $1 \times J$ vector of the probabilities that the chain visits state j at time n.

By marginalization

$$p\left(X_n = j\right) = \sum_{k=1}^{J} p_{k|j} \cdot p\left(X_{n-1} = k\right).$$

This we write using a matrix notation as

$$\phi(n) = \phi(n-1)P.$$

A Markov chain $\{X_n\}_{n=0}^{\infty}$ may be such that the probability $p(X_n = j)$ is independent of n for all j in the state space. A distribution π an *invariant* or *stationary distribution*, with

$$\pi = (\pi_1, \ldots, \pi_J),$$

if $p(X_0 = j) = \pi_j$ for all j implies that $p(X_1 = j) = \pi_j$ for all j.

Let $\{X_n\}_{n=0}^{\infty} \sim \text{Markov}\,(P, \phi(0))$. Every stationary (invariant) distibution satisfies the equation

$$\pi = \pi P$$

(π is a row vector) with the constraints

$$\sum_{j=1}^{J} \pi_j = 1, \pi_j \geq 0.$$

Proof

Assume first that π is an invariant distribution. Then $\sum_{j=1}^{J} \pi_j = 1$ and $\pi_j \geq 0$ are clear. Since π is invariant, by the definition above we must have $\phi(0) = \pi$ and $\phi(1) = \pi$. But since

$$\phi(n) = \phi(n-1)P,$$

we get that

$$\pi = \pi P.$$

Assume now that π satisfies $\pi = \pi P$ and the other constraints. Let $\phi(0) = \pi$. Then

$$\phi(1) = \phi(0)P = \pi P = \pi$$

and π is an invariant distribution. ■

Existence of a stationary distribution

Every MC with a fi nite state space has *at least one* invariant distribution *Proof:* We give only an outline of the proof. Let p be an arbitrary probability distribution on S. Set

$$p^{(n)} = \frac{1}{n}\left(p + pP + pP^2 + \ldots + pP^{n-1}\right)$$

This is a sequence of probability distributions, i.e. vectors with components with values between zero and one. Thus the well known theorem of Bolzano and Weierstrass shows that we can pick a convergent subsequence $p^{(n_v)}$ which converges componentwise to the vector ϕ. We can show that ϕ is a probability distribution.

By our construction we have the recursion relations

$$p^{(n+1)} = \frac{n}{n+1} p^{(n)} + \frac{1}{n+1} p P^n$$

and

$$p^{(n+1)} = \frac{n}{n+1} p^{(n)} P + \frac{1}{n+1} p.$$

From the recursion above we get that

$$p^{(n_v+1)} \to \pi$$

and then we get that

$$\pi = \pi P,$$

which proves the claim.

The components in the stationary distribution can be interpreted as the asymptotic percentages of ‘time’ the chain spends in each of the states.

Stationary distribution

$$P = \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix}$$

Then $\pi = \pi P$ is solved by

$$\pi = \left(\frac{q}{p+q}, \frac{p}{p+q} \right)$$

The components in the stationary distribution are perhaps more explicitly visualized as the asymptotic percentages of time the chain spends in each of the states.

Is there convergence to a stationary distribution for any $\phi(0)$? Let $\{X_n\}_{n=0}^{\infty} \in \text{Markov}\,(P, \phi(0))$. Let us assume that

$$\lim_{n\to\infty} \phi(n) = a,$$

where $a = (a_1, \ldots, a_J)$ is a probability distribution. Then a is an invariant distribution.

Taking of limits yields

$$a = \lim_{n\to\infty} \phi(n) = \lim_{n\to\infty} \phi(n+1) =$$

$$= \lim_{n\to\infty} (\phi(n)P) = \left(\lim_{n\to\infty} \phi(n)\right) P = aP.$$

(a) An MC is **aperiodic**, if there is no state such that return to that state is possible only after t_0, $2t_0$, $3t_0$ ... steps later.

(b) An MC is irreducible means that every state can be reached from any other state, if not in one step, but then after several steps.

Convergence to a unique invariant distribution

If a finite MC is aperiodic and irreducible, then for any $\phi(0)$

$$\lim_{n\to\infty} \phi(n) = \pi,$$

where π is a unique probability distribution that satisfies

$$\pi = \pi P.$$

Markov Chains for sampling

- To sample from p(x), we require

$$\mu(x^{(1)})T^t \underset{t\to\infty}{\to} p(x)$$

- The stationary distribution of the Markov chain must be p(x)

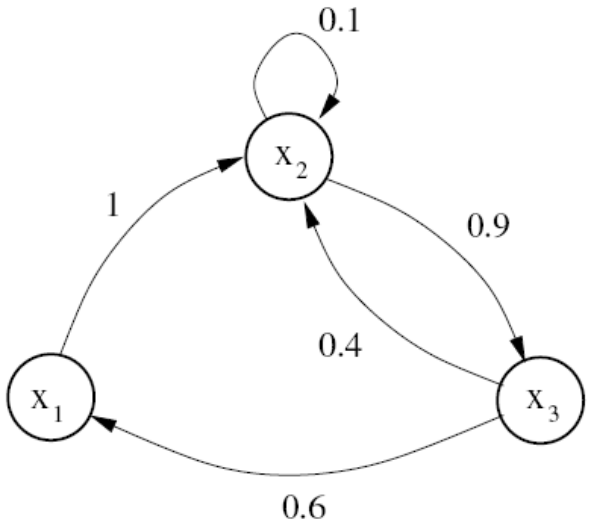
$$p\mathbf{T} = p$$

- If this is the case, we can start in an arbitrary state, use the Markov chain to do a random walk for a while, and stop and output the current state $x^{(t)}$
- The resulting state will be sampled from p(x)

Markov Chain Monte Carlo

- Strategy for generating samples $x^{(i)}$, while exploring the state space X using a Markov chain mechanism

$$p(x^{(i)}|x^{(i-1)}, \ldots, x^{(1)}) = T(x^{(i)}|x^{(i-1)})$$

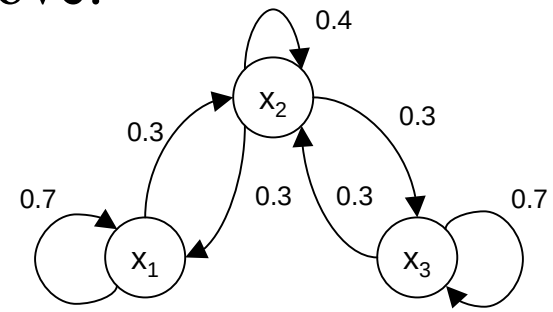


$$T = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0.1 & 0.9 \\ 0.6 & 0.4 & 0 \end{bmatrix}$$

Stationary distribution

- Consider the Markov chain given above:

$$T = \begin{pmatrix} 0.7 & 0.3 & 0 \\ 0.3 & 0.4 & 0.3 \\ 0 & 0.3 & 0.7 \end{pmatrix}$$



- The stationary distribution is

$$\begin{pmatrix} 0.33 & 0.33 & 0.33 \end{pmatrix} \times \begin{pmatrix} 0.7 & 0.3 & 0 \\ 0.3 & 0.4 & 0.3 \\ 0 & 0.3 & 0.7 \end{pmatrix} = \begin{pmatrix} 0.33 & 0.33 & 0.33 \end{pmatrix}$$

- Some samples:

1,1,2,3,2,1,2,3,3,**2**
1,2,2,1,1,2,3,3,3,**3**
1,1,1,2,3,2,2,1,1,**1**
1,2,3,3,3,2,1,2,2,**3**
1,1,2,2,2,3,3,2,1,**1**
1,2,2,2,3,3,3,2,2,**2**

Empirical Distribution:

$$\begin{pmatrix} 0.33 & 0.33 & 0.33 \end{pmatrix}$$

Reversibility

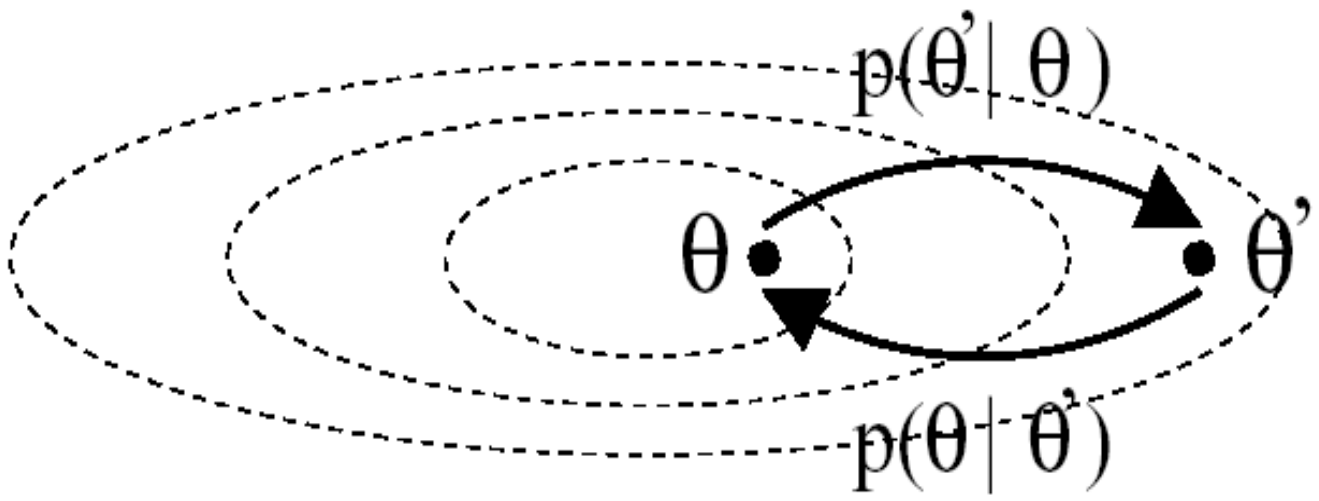
It can be difficult to work with irreducibility and aperiodicity so more often we simply consider *reversibility*.

Reversibility is sufficient but not necessary.

The distribution π is invariant for $p(. \mid .)$ if we have detailed balance (reversibility):

$$\pi(\boldsymbol{\theta})\, p\left(\boldsymbol{\theta}' \mid \boldsymbol{\theta}\right) = \pi\left(\boldsymbol{\theta}'\right) p\left(\boldsymbol{\theta} \mid \boldsymbol{\theta}'\right)$$

Think of reversibility as requiring a balance in the flow of probability



Detailed Balance

- Claim: To ensure that the stationary distribution of the Markov chain is p(x) it is sufficient for p and *T* to satisfy the *detailed balance* (*reversibility*) condition:

$$p(x^{(i)})T(x^{(i-1)} \mid x^{(i)}) = p(x^{(i-1)})T(x^{(i)} \mid x^{(i-1)})$$

Summing over (i-1)

$$\sum_{i-1} p(x^{(i)})T(x^{(i-1)} \mid x^{(i)}) = \sum_{i-1} p(x^{(i-1)})T(x^{(i)} \mid x^{(i-1)})$$

$$p(x^{(i)}) = \sum_{i-1} p(x^{(i-1)})T(x^{(i)} \mid x^{(i-1)})$$

Ergodicity

- Claim: To ensure that the chain converges to a unique stationary distribution the following conditions are sufficient:
  - *Irreducibility*: every state is eventually reachable from any start state; for all $x,y \epsilon \boldsymbol{X}$ there exists a t such that
$$p_x^{(t)}(y) > 0$$
  - *Aperiodicity*: the chain doesn't get caught in cycles; for all $x,y \epsilon \boldsymbol{X}$ it is the case that
$$\gcd\{t : p_x^{(t)}(y) > 0\} = 1$$
- The process is *ergodic* if it is both irreducible and aperiodic
- This claim is easy to prove, but involves eigenstuff!

How to pick a suitable Markov chain for our distribution?

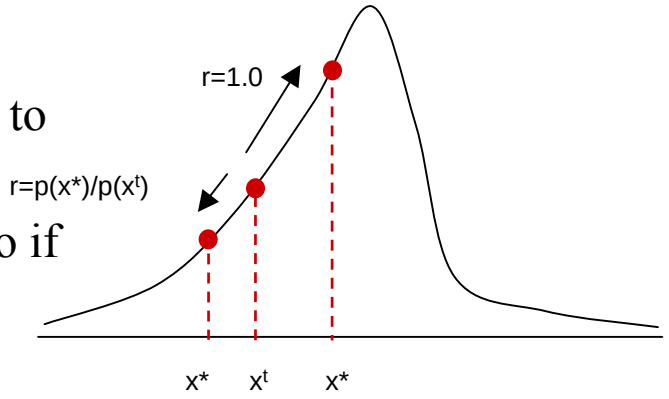
- Suppose our distribution p(x) is easy to sample, and easy to compute *up to a normalization constant*, but hard to compute exactly
  - e.g. a Bayesian posterior P(M|D) α P(D|M)P(M)
- We define a Markov chain with the following process:
  - Sample a candidate point x* from a *proposal distribution* $q(x^*|x^{(t)})$ which is *symmetric*: q(x|y)=q(y|x)
  - Compute the *importance ratio* (this is easy since the normalization constants cancel)

$$r = \frac{p(x^*)}{p(x^{(t)})}$$

  - With probability min(r,1) transition to x*, otherwise stay in the same state

Metropolis intuition

- Why does the Metropolis algorithm work?
  - Proposal distribution can propose anything it likes (as long as it can jump back with the same probability)
  - Proposal is always accepted if it's jumping to a more likely state
  - Proposal accepted with the importance ratio if it's jumping to a less likely state
- The acceptance policy, combined with the reversibility of the proposal distribution, makes sure that the algorithm explores states in proportion to p(x)!



Metropolis convergence

- Claim: The Metropolis algorithm converges to the target distribution $p(x)$.
- Proof: It satisfies detailed balance
  - For all $x,y \in \mathbf{X}$, wlog assuming $p(x)<p(y)$

$$p(x)T(x,y) = p(x)q(y \mid x)$$ candidate is always accepted b/c p(x)<p(y)

$$= p(x)q(x \mid y)$$ q is symmetric

$$= p(y)q(x \mid y)\frac{p(x)}{p(y)}$$

$$= p(y)T(y,x)$$ transition prob b/c p(x)<p(y)

Metropolis-Hastings

- The symmetry requirement of the Metropolis proposal distribution can be hard to satisfy
- Metropolis-Hastings is the natural generalization of the Metropolis algorithm, and the most popular MCMC algorithm
- We define a Markov chain with the following process:
  - Sample a candidate point x^* from a proposal distribution $q(x^*|x^{(t)})$ which is not necessarily symmetric
  - Compute the importance ratio:

$$r = \frac{p(x^*)\,q(x^{(t)} \mid x^*)}{p(x^{(t)})\,q(x^* \mid x^{(t)})}$$

  - With probability $\min(r,1)$ transition to x^*, otherwise stay in the same state $x^{(t)}$

Metropolis Hastings

1. Initialise $x^{(0)}$.
2. For $i = 0$ to $N - 1$
   - Sample $u \sim \mathcal{U}_{[0,1]}$.
   - Sample $x^\star \sim q(x^\star | x^{(i)})$.
   - If $u < \mathcal{A}(x^{(i)}, x^\star) = \min\left\{1, \frac{p(x^\star)q(x^{(i)}|x^\star)}{p(x^{(i)})q(x^\star|x^{(i)})}\right\}$

     $$x^{(i+1)} = x^\star$$

     else

     $$x^{(i+1)} = x^{(i)}$$

MH convergence

- Claim: The Metropolis-Hastings algorithm converges to the target distribution $p(x)$.
- Proof: It satisfies detailed balance
  - For all $x,y \in \boldsymbol{X}$, wlog assume $p(x)q(y|x)=p(y)q(x|y)$

$$p(x)T(x,y) = p(x)q(y \mid x)$$

candidate is always accepted b/c p(x)q(y|x)=p(y)q(x|y)

$$= p(x)q(y \mid x)\frac{p(y)q(x \mid y)}{p(y)q(x \mid y)}$$

$$= p(y)q(x \mid y)\frac{p(x)q(y \mid x)}{p(y)q(x \mid y)}$$

$$= p(y)T(y,x)$$

transition prob b/c p(x)q(y|x)=p(y)q(x|y)

advantages

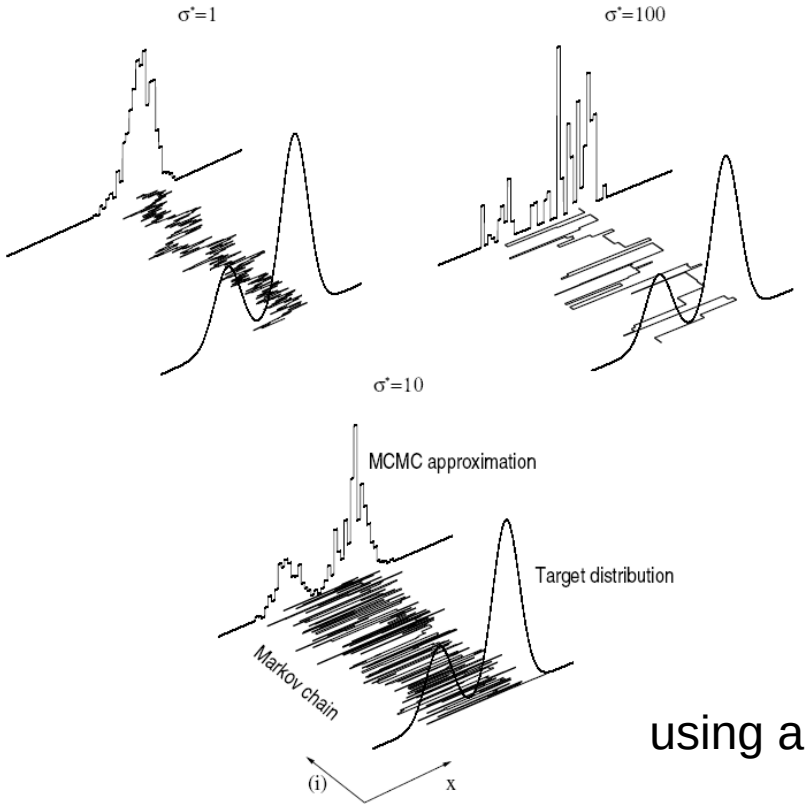
- The symmetry requirement is avoided.
- Allowing asymmetric jumping rules can be useful in increasing the speed of the random walk

- A good jumping distribution has the following properties:
  - For any θ, it is easy to sample from $J(\theta^* | \theta)$
  - It is easy to compute the ratio of importance ratios r
  - Each jump goes a reasonable distance in the parameter space (otherwise the random walk moves too slowly)
  - The jumps are not rejected too frequently (otherwise the random walk wastes too much time standing still)

Metropolis Hastings

If $A(x^i, x^*) = 1$, then the new state is accepted

Otherwise, the new state is accepted with probability $\frac{p(x^*)q(x^{(i)} \mid x^*)}{p(x^{(i)})q(x^* \mid x^{(i)})}$



using a good proposal distribution is important

The Transition Kernel

The transition kernel for MH algorithm

$$K_{\mathrm{MH}}(x^{(i+1)}|x^{(i)}) = q(x^{(i+1)}|x^{(i)})\mathcal{A}(x^{(i)}, x^{(i+1)}) + \delta_{x^{(i)}}(x^{(i+1)})r(x^{(i)})$$

Rejection term -

$$r(x^{(i)}) = \int_{\mathcal{X}} q(x^{\star}|x^{(i)}) \left(1 - \mathcal{A}(x^{(i)}, x^{\star})\right) dx^{\star}$$

$$r(x^{(i)}) = \sum_{X} q(x^{*} \mid x^{(i)})(1 - A(x^{(i)}, x^{*}))$$

Special cases of MH algorithm

- Independent sampler

$$\mathcal{A}(x^{(i)}, x^{\star}) = \min\left\{1, \frac{p(x^{\star})q(x^{(i)})}{p(x^{(i)})q(x^{\star})}\right\} = \min\left\{1, \frac{w(x^{\star})}{w(x^{(i)})}\right\}$$

- Metropolis algorithm

$$\mathcal{A}(x^{(i)}, x^{\star}) = \min\left\{1, \frac{p(x^{\star})}{p(x^{(i)})}\right\}$$

- Gibbs algorithm

Gibbs sampling

- A special case of Metropolis-Hastings which is applicable to state spaces in which we have a factored state space, and access to the full conditionals:

$$p(x_j \mid x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n)$$

- Perfect for Bayesian networks!
- Idea: To transition from one state (variable assignment) to another,
  - Pick a variable,
  - Sample its value from the conditional distribution
  - That's it!
- We'll show in a minute why this is an instance of MH and thus must be sampling from the full joint

Gibbs Sampling

- Gibbs sampling is the simplest and most easily implemented sampling method for MCMC. However, the problem has to have a particular form in order for it to work.
- The idea is as follows. Consider a problem with two parameters, θ_1 and θ_2. Suppose we have available the *conditional* distributions

$$p(\theta_1 \mid \theta_2, D) \quad \text{and} \quad p(\theta_2 \mid \theta_1, D)$$

where D is the data (not needed). Then, starting at some initial point $(\theta_1^{(0)}, \theta_2^{(0)})$ in parameter space, generate a *random walk*, a sequence $(\theta_1^{(k)}, \theta_2^{(k)})$ as follows:

Gibbs Sampling

- For k=1,…,n define

$$\theta_1^{(k)} \sim p(\theta_1 \mid \theta_2^{(k-1)}, D),$$

$$\theta_2^{(k)} \sim p(\theta_2 \mid \theta_1^{(k)}, D)$$

  where '~' means here that we draw the value in question from the indicated distribution.

- The resulting sequence of values is a *Markov chain*; the values at the (k+1)st step depend only on the values at the kth step and are independent of previous values

- The Markov chain will in general tend to a stationary distribution, and the stationary distribution will be the desired $p(\theta_1, \theta_2 | D)$

Gibbs Sampling

- The method generalizes to a large number of variables, e.g.,

$$\theta_1^{(k)} \sim p(\theta_1 \mid \theta_2^{(k-1)}, \theta_3^{(k-1)}, \ldots, \theta_m^{(k-1)}, D),$$
$$\theta_2^{(k)} \sim p(\theta_2 \mid \theta_1^{(k)}, \theta_3^{(k-1)}, \ldots, \theta_m^{(k-1)}, D)$$
$$\vdots$$
$$\theta_m^{(k)} \sim p(\theta_2 \mid \theta_1^{(k)}, \theta_2^{(k)}, \ldots, \theta_{m-1}^{(k)}, D)$$

Gibbs sampling

- More formally, the proposal distribution is

$$q(x^* \mid x^{(t)}) = \begin{cases} p(x^*_j \mid x^{(t)}_{-j}) & \text{if } x^*_{-j} = x^{(t)}_{-j} \\ 0 & \text{otherwise} \end{cases}$$

- The importance ratio is

$$r = \frac{\mathrm{p}(y)\,\mathrm{q}(x \mid y)}{\mathrm{p}(x)\,\mathrm{q}(y \mid x)}$$

$$= \frac{\mathrm{p}(y)\,\mathrm{p}(x_j \mid x_{-j})}{\mathrm{p}(x)\,\mathrm{p}(y_j \mid y_{-j})}$$

Definition of proposal distribution

$$= \frac{\mathrm{p}(y)\,\mathrm{p}(x_j, x_{-j})\,\mathrm{p}(y_{-j})}{\mathrm{p}(x\ )\,\mathrm{p}(y_j, y_{-j})\,\mathrm{p}(x_{-j})}$$

Definitionn of conditional probability

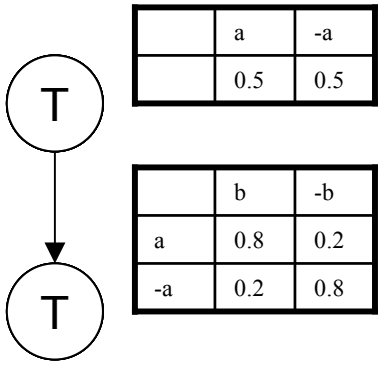
$$= \frac{\mathrm{p}(y_{-j})}{\mathrm{p}(x_{-j})} = 1$$

B/c we didn't change other vars

- So we always accept!

Gibbs sampling example

- Consider a simple, 2 variable Bayes net



	a	-a
	0.5	0.5

	b	-b
a	0.8	0.2
-a	0.2	0.8

	b	-b
a	1	1
-a	1	1

- Initialize randomly
- Sample variables alternately

Practical issues

- How many iterations?
- How to know when to stop?
- What's a good proposal function?

Gibbs Sampling

- One particular choice of cycle kernel
  - Make each variable a block
  - Proposal distribution: $p(x_i|x_{j\setminus i})$
  - For a Bayes Net, this is $p(x_i|\textit{markov blanket}(x_i))$

$$q(x^\star|x^{(i)}) = \begin{cases} p(x_j^\star|x_{-j}^{(i)}) & \text{If } x_{-j}^\star = x_{-j}^{(i)} \\ 0 & \text{Otherwise.} \end{cases}$$

- This choice of q yields an acceptance probability of 1 ("deterministic scan")
- Sampling from q may require additional MH steps

Gibbs Sampling

- Suppose we have normally distributed estimates X_i, i=1,…,N, of a parameter x, with unknown variance σ^2. The likelihood is

$$p(X|x,\sigma) \propto \sigma^{-N}\exp(-\Sigma(X_i-x)^2/2\sigma^2)$$

- Assume a flat (uniform) prior for x and a "Jeffreys" prior $1/\sigma^2$ for σ^2. The posterior is proportional to prior times likelihood:

$$p(x,\sigma|X) \propto \sigma^{-(N+2)}\exp(-\Sigma(X_i-x)^2/2\sigma^2)$$

(The Jeffreys prior is $d\sigma/\sigma \propto d\sigma^2/\sigma^2$; it is commonly used as a prior for *scale* variables. For technical reasons having to do with the distributions available in R it's best to think about sampling σ^2 instead of σ so we use $d\sigma^2/\sigma^2$)

Gibbs Sampling

- The posterior distribution can be simplified using the trick of "completing the square"

$$p(x,\sigma^2 \mid X) \propto \sigma^{-(N+2)} \exp\left(-\frac{\sum (X_i - \bar{x} + \bar{x} - x)^2}{2\sigma^2}\right)$$

$$= \sigma^{-(N+2)} \exp\left(-\frac{\sum (X_i - \bar{x})^2}{2\sigma^2}\right) \exp\left(-\frac{\sum (\bar{x} - x)^2}{2\sigma^2}\right)$$

$$= \sigma^{-(N+2)} \exp\left(-\frac{S_{xx}}{2\sigma^2}\right) \exp\left(-\frac{N(\bar{x} - x)^2}{2\sigma^2}\right)$$

$$p(x \mid \sigma^2, X) = p(x,\sigma^2 \mid X) / p(\sigma^2 \mid X)$$

Depends only on σ^2

Gibbs Sampling

- When sampling x, σ will be fixed, so we can ignore factors dependent only on σ. Let $\bar{x}$ be the sample mean as we defined it before. Then the conditional distribution of x can be rewritten, apart from constant factors

$$p(x \mid \sigma^2, X) \propto \exp\left(-\frac{N(\bar{x}-x)^2}{2\sigma^2}\right)$$

- This is readily seen to be normal with mean $\bar{x}$ and variance σ^2/N. So that's the distribution we need to use for sampling x in each Gibbs step.

Gibbs Sampling

- The conditional distribution for $\chi^2=\Sigma(X_i-x)^2/\sigma^2$ is even easier, going back to the *original* posterior (3 pages back):

$$p(\chi^2 \mid x, X) \propto (\chi^2)^{N/2+1} \exp(-\chi^2/2)\frac{d\sigma^2}{d\chi^2}$$

$$\propto (\chi^2)^{N/2-1} \exp(-\chi^2/2)$$

$$\sigma^2 = \frac{1}{\chi^2} \quad \text{so} \quad \frac{d\sigma^2}{d\chi^2} = \frac{1}{(\chi^2)^2}$$

- This is a standard chi-square distribution on N degrees of freedom, and R has a function for drawing samples from that distribution. We can then get a value of σ^2 by dividing $\Sigma(X_i-x)^2$ by the value of χ^2 that we sampled. That allows us to sample σ^2 at each Gibbs step

Recall previous example in which we have data $X_i \sim N\left(\theta, \sigma^2\right)$ with independent priors for θ and σ^2:

$$\begin{aligned} \theta &\sim N\left(\mu, \sigma^2\right) \\ \sigma^{-2} &\sim \Gamma(\alpha, \beta) \end{aligned}$$

Although we saw that the resulting posterior was not in standard form it is possible to find full conditionals for the parameters θ and σ^2.

The joint posterior is given by:

$$\pi\left(\theta, \sigma^{-2} \mid \underline{x}\right) \propto \left(\sigma^2\right)^{-\alpha-n/2-a} \times \exp\left\{-\frac{\beta}{\sigma^2} - \frac{(\theta-\mu)^2}{2\tau^2} - \frac{\sum (x_i-\theta)^2}{2\sigma^2}\right\}$$

If we consider this to be a function of θ the the full conditional of θ must be proportiona to

$$\exp\left\{-\frac{1}{2}\left(\theta^2\left(\frac{n}{\sigma^2}+\frac{1}{\tau^2}\right) - 2\theta\left(\frac{n\bar{x}}{\sigma^2}+\frac{\mu}{\tau^2}\right)\right)\right\}$$

Hence the full conditional is:

$$\theta \mid \sigma^2, \underline{x} \sim N\left(\frac{\sigma^{-2}\sum x_i + \kappa\epsilon}{\sigma^{-2}n+\kappa}, \frac{1}{\sigma^{-2}n+\kappa}\right)$$

Similarly, the full conditional of σ^{-2} must be proportional to

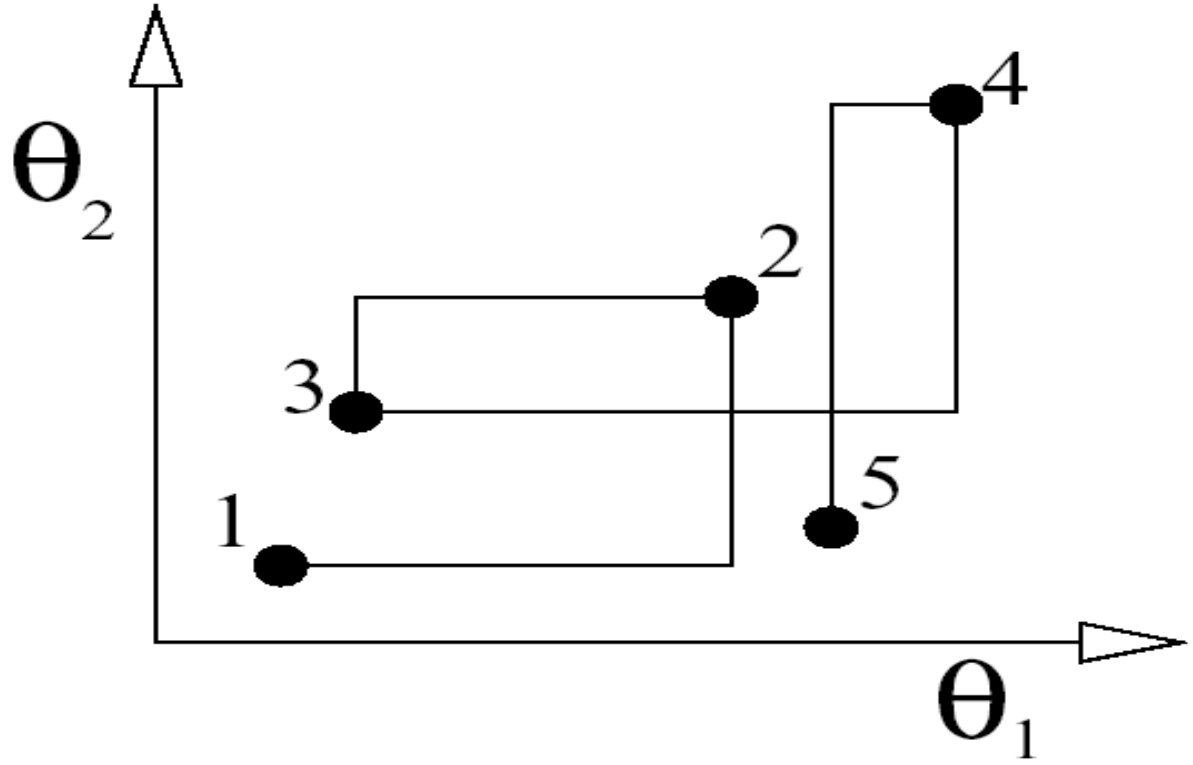
$$\left(\sigma^{-2}\right)^{\alpha+n/2+1} \exp\left\{-\sigma^{-2}\frac{1}{2}\left(\beta-\sum\left(x_i-\theta\right)\right)\right\}$$

Hence the full conditional is:

$$\sigma^{-2} \mid \theta, \underline{x} \;\sim\; \Gamma\left(\alpha+n/2, \beta+\sum\left(Y_i-\mu\right)^2/2\right)$$

We implement the Gibbs sampler by alternately drawing θ and σ^{-2} from these distributions.

In two dimensions (k=2) the sample path of the Gibbs sampler should look something like this.



Example: Fitting straight line

We have a set of 5 observed (x, y) pairs $(1,1), (2,3), (3,3), (4,3), (5,5)$. We shall fit a simple linear regression of y on x using the notation

$$\begin{aligned} Y_i &\sim N\left(\theta_i, \sigma^2\right) \\ \theta_i &= \alpha + \beta\left(x_i - \bar{x}\right) \end{aligned}$$

Classical unbiased estimates are

$$\begin{array}{lcccl} \hat{\alpha} & = & \bar{y} & = & 3.00 \\ \hat{\beta} & = & \sum_i y_i\left(x_i - \bar{x}\right) / \sum_i \left(x_i - \bar{x}\right)^2 & = & 0.80 \\ \hat{\sigma}^2 & = & \sum \left(y_i - \hat{y}_i\right)^2 / (n-2) & = & 0.533 \\ \text{Var}(\hat{\alpha}) & = & \hat{\sigma}^2/n & = & 0.107 \\ \text{Var}(\hat{\beta}) & = & \hat{\sigma}^2 / \sum \left(x_i - \bar{x}\right)^2 & = & 0.053 \end{array}$$

$$\begin{array}{lcl} 95\% \text{ interval for } \alpha & = & (1.96, 4.04) \\ 95\% \text{ interval for } \beta & = & (0.07, 1.53) \\ 95\% \text{ interval for } \sigma & = & (0.41, 2.67) \end{array}$$

The full conditionals for the parameters α, β and σ^{-2} are

$$\begin{aligned}
\alpha \mid \beta, \sigma^{-2} &\sim N\left(\beta \frac{1}{n}(n-1)\,\bar{x} - \bar{y}, \frac{1}{n}\sigma^2\right) \\
\beta \mid \alpha, \sigma^{-2} &\sim N\Big(\alpha \textstyle\sum (x_i - \bar{x}) \,/\, \sum (x_i - \bar{x})^2 \\
&\qquad - \textstyle\sum y_i (x_i - \bar{x}) \,/\, \sum (x_i - \bar{x})^2 \\
&\qquad , \sigma^2 / \textstyle\sum (x_i - \bar{x})^2\Big) \\
\sigma^{-2} \mid \alpha, \beta &\sim \Gamma\left(\frac{n}{2} + 1, \frac{1}{2}\sum (y_i - \alpha - \beta\,(x_i - \bar{x}))^2\right)
\end{aligned}$$

Updates are conducted by sampling from each of these three conditional distributions in turn.

Advanced Topics

- Simulated annealing, for global optimization, is a form of MCMC
- Mixtures of MCMC transition functions
- Monte Carlo EM (stochastic E-step)
- Reversible jump MCMC for model selection
- Adaptive proposal distributions

Simulated Annealing

1. Initialise $x^{(0)}$ and set $T_0 = 1$.
2. For $i = 0$ to $N - 1$
   - Sample $u \sim \mathcal{U}_{[0,1]}$.
   - Sample $x^\star \sim q(x^\star | x^{(i)})$.
   - If $u < \mathcal{A}(x^{(i)}, x^\star) = \min\left\{1, \frac{p^{\frac{1}{T_i}}(x^\star) q(x^{(i)}|x^\star)}{p^{\frac{1}{T_i}}(x^{(i)}) q(x^\star|x^{(i)})}\right\}$

     $$x^{(i+1)} = x^\star$$

     else

     $$x^{(i+1)} = x^{(i)}$$
   - Set T_{i+1} according to a chosen cooling schedule.

MCMC Mixtures and Cycles

- If K_1 and K_2 are transition kernels for $p(x)$, then the following are valid kernels:
  - $K_1 K_2$ (cycle hybrid kernel)
  - $\nu K_1 + (1-\nu) K_2, 0 \le \nu \le 1$ (mixture hybrid kernel)
- What's the Point?
  - K_1 and K_2 can have different behavior
    - Global vs. local (mixture)
    - One set of variables vs. others (cycles)

Using a Mixture of Kernels

- Use u to decide between K_1 and K_2
  - Global K locks into peaks
  - Local K does random walk in that area

1. Initialise $x^{(0)}$.
2. For $i = 0$ to $N - 1$
   - Sample $u \sim \mathcal{U}_{[0,1]}$.
   - If $u < \nu$

     Apply the MH algorithm with a global proposal.
   - else

     Apply the MH algorithm with a random walk proposal.

from Andrieu et al. An Introduction to MCMC for Machine Learning. *Machine Learning*, 2002.

Cycles of Kernels

- Split multivariate state into blocks for separate updating
- Update block b_i given all other blocks b_j and previous value of b_i

1. Initialise $x^{(0)}$.
2. For $i = 0$ to $N - 1$
   - Sample the block $x_{b_1}^{(i+1)}$ according to an MH step with proposal distribution $q_1(x_{b_1}^{(i+1)}|x_{-[b_1]}^{(i+1)}, x_{b_1}^{(i)})$ and invariant distribution $p(x_{b_1}^{(i+1)}|x_{-[b_1]}^{(i+1)})$.
   - Sample the block $x_{b_2}^{(i+1)}$ according to an MH step with proposal distribution $q_2(x_{b_2}^{(i+1)}|x_{-[b_2]}^{(i+1)}, x_{b_2}^{(i)})$ and invariant distribution $p(x_{b_2}^{(i+1)}|x_{-[b_2]}^{(i+1)})$.

   ⋮

   - Sample the block $x_{b_{n_b}}^{(i+1)}$ according to an MH step with proposal distribution $q_{n_b}(x_{b_{n_b}}^{(i+1)}|x_{-[b_{n_b}]}^{(i+1)}, x_{b_{n_b}}^{(i)})$ and invariant distribution $p(x_{b_{n_b}}^{(i+1)}|x_{-[b_{n_b}]}^{(i+1)})$.

from Andrieu et al. An Introduction to MCMC for Machine Learning. *Machine Learning*, 2002.

Monte Carlo EM (MCEM)

- The EM algorithm
  - E: Compute expected log lhood over *p(h|v,t)*

$$Q(\theta) = \int_{\mathcal{X}_h} \log\left(p(x_h, x_v|\theta)\right) p(x_h|x_v, \theta^{(\text{old})}) dx_h,$$

  - M: Maximize log lhood wrt parameters theta
- The integral over all variables can be hard
  - Approximate via sampling (E-step)
  - Do M Step as usual

MCEM: Procedure

1. Initialise $(x_h^{(0)}, \theta^{(0)})$ and set $i = 0$.
2. Iteration i of EM
   - Sample $\{x_h^{(j)}\}_{j=1}^{N_i}$ with any suitable MCMC algorithm. For example, one could use an MH algorithm with acceptance probability

$$\mathcal{A} = \min\left\{1, \frac{p(x_v|x_h^\star, \theta^{(i-1)})p(x_h^\star|\theta^{(i-1)})q(x_h^{(j)}|x_h^\star)}{p(x_v|x_h^{(j)}, \theta^{(i-1)})p(x_h^{(j)}|\theta^{(i-1)})q(x_h^\star|x_h^{(j)})}\right\}$$

   - **E step**: Compute

$$\widehat{Q}(\theta) = \frac{1}{N_i}\sum_{j=1}^{N_i} \log p(x_h^{(j)}, x_v|\theta)$$

   - **M step**: Maximise $\theta^{(i)} = \arg\max_\theta \widehat{Q}(\theta)$.
3. $i \leftarrow i + 1$ and go to 2.

from Andrieu et al. An Introduction to MCMC for Machine Learning. *Machine Learning*, 2002.

Auxiliary Variable Sampling

- Sometimes easier to sample *p(x,u)* than *p(x)*
  - Obtain samples of *p(x)* by ignoring *u*
- Hybrid Monte Carlo
  - Use gradient of target distribution
  - $p(x,u) = p(x)N(u; O, I_{nx})$
  - Take L "frog leaps" in *u* and *x*

$$\Delta x \propto \partial \log p(x) / \partial x$$

$$x_l = x_{l-1} + \rho u_{l-1}$$
$$u_l = u_{l-1} + \rho_l \Delta(x_l)$$

  - Lth value is the proposal candidate for MH with *p(x,u)*

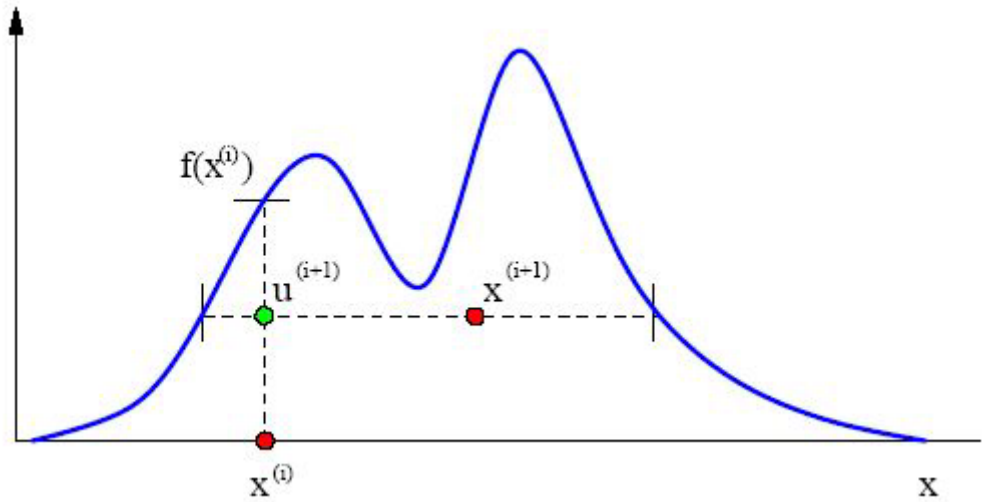
Auxiliary Variable Sampling (II)

- The Slice Sampler
  - General version of Gibbs Sampler

$$p^\star(x,u) = \begin{cases} 1 & \text{if } 0 \le u \le p(x) \\ 0 & \text{otherwise.} \end{cases} \quad \text{where} \quad \int p^\star(x,u)du = \int_0^{p(x)} du = p(x).$$

  - Conditionals:

$$p(u|x) = \mathcal{U}_{[0,p(x)]}(u)$$
$$p(x|u) = \mathcal{U}_A(x) \qquad A = \{x; p(x) \ge u\}$$



from Andrieu et al. An Introduction to MCMC for Machine Learning. *Machine Learning*, 2002.

Convergence

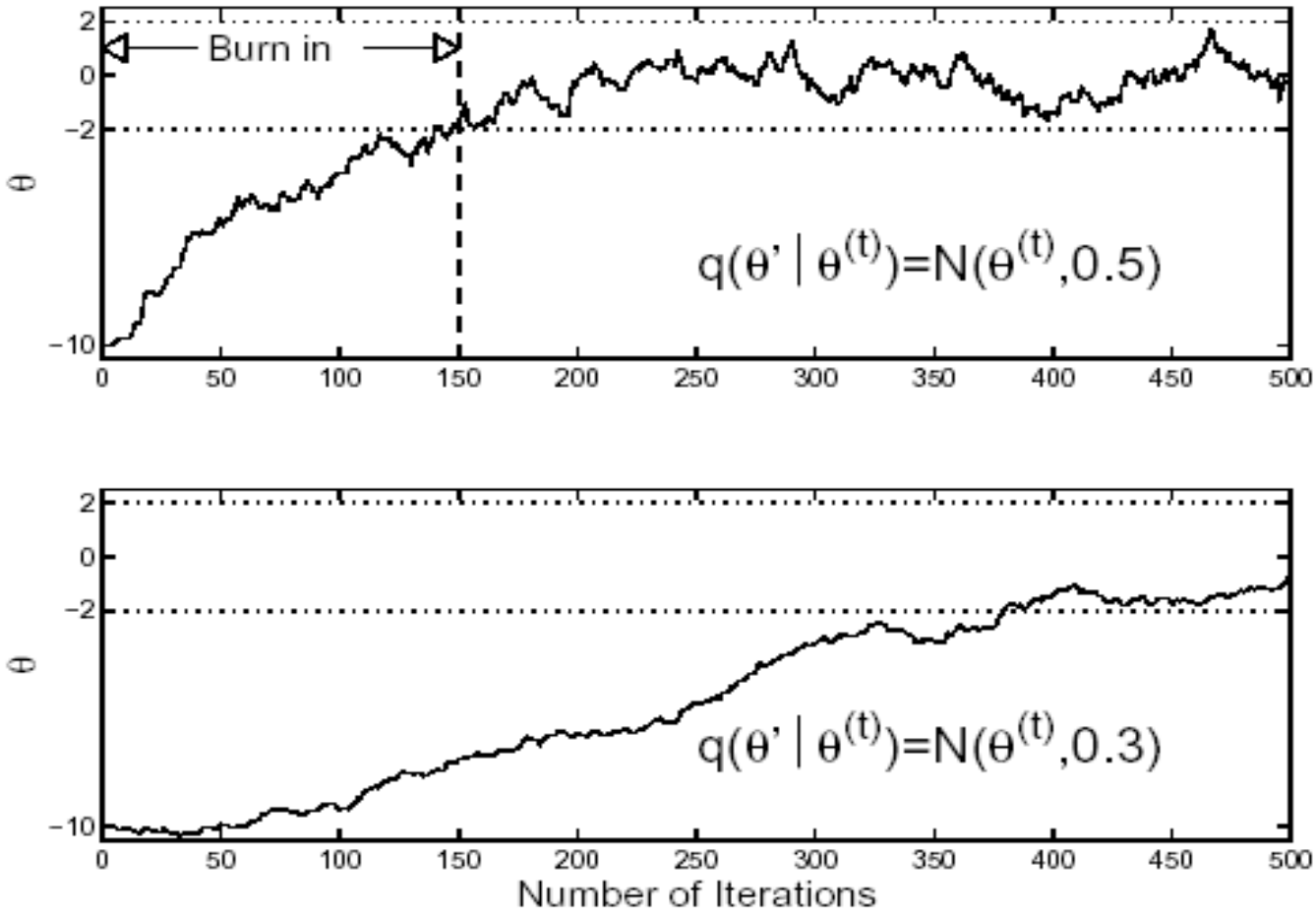
We know that our Markov chain will only resemble our posterior distribution $\pi(.)$ once it has converged to its stationary distribution.

How do we decide when convergence has taken place?

We might hope to define some measure of similarity between $P^{(t)}(. \mid .)$ and $\pi(.)$. Unfortunately this is almost always impossible.

Most common approach is visual inspection of the Monte Carlo output.

Visual inspection



autocorrelations

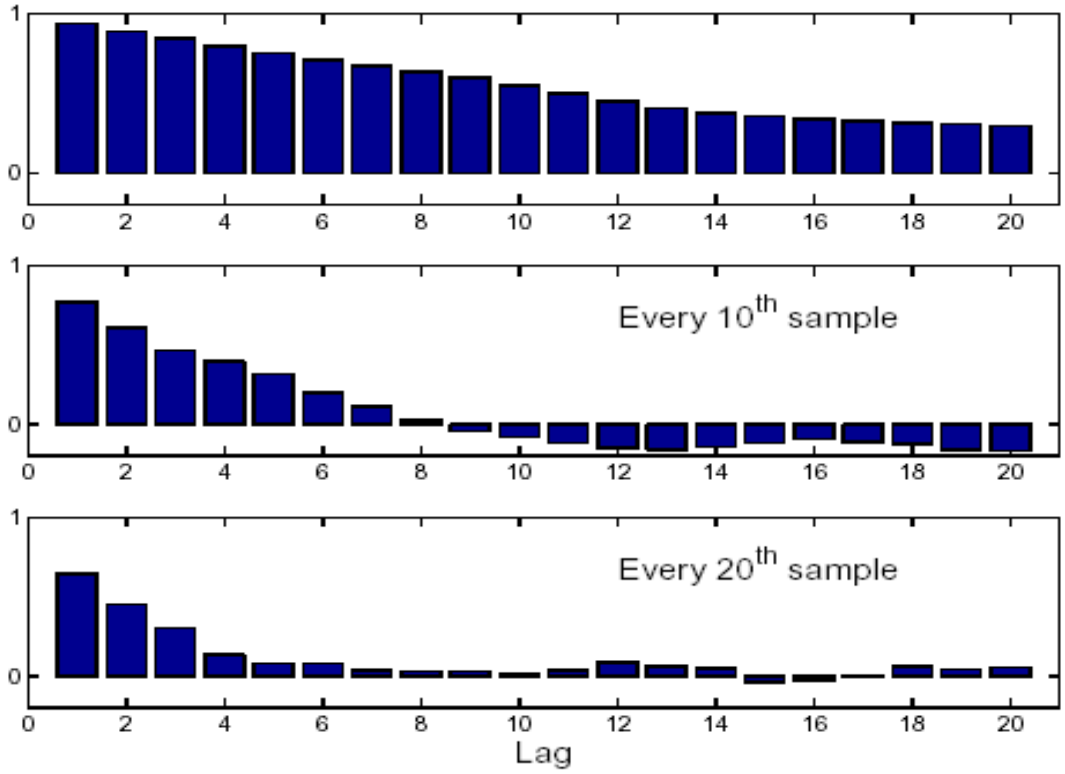
Another useful diagnostic is to look at auto-correlation plots.

Samples produced by a Markov Chain will be dependent.

Too much dependence is indicative of poor mixing.

We look at the chain's autocorrelation, high autocorrelation will indicate poor mixing (and hence poor convergence).

We take lags $1, 2, 3, \ldots$ and for each we take the set of all pairs separated by these lags and calculate the autocorrelation coefficient.



One way that we can reduce the autocorrelation of our sample is to use only every k^{th} sample, discarding the rest.

example

Data on 10 power plant pumps.

Interested in the number of failures of each pump.

We observe for each pump, i, the number of failures x_i that occur within a time t_i (hence failures occur with observed rate $\rho_i = x_i/t_i$).

Assume that the number of failures for pump i, x_i, follows a Poisson distribution:

$$x_i \sim Poisson\left(\theta_i t_i\right), \text{ so } P\left(x_i\right) = \frac{e^{-\theta_i t_i}\left(\theta_i t_i\right)^{x_i}}{x_i!}.$$

where θ_i is the true failure rate for pump i.

Pump	1	2	3	4	5
t_i	94.3	15.7	62.9	126	5.24
x_i	5	1	5	14	3

Pump	6	7	8	9	10
t_i	31.4	1.05	1.05	2.1	10.5
x_i	19	1	1	4	22

We assume a gamma distribution for the failure rates:

$$\theta_i \sim G(\alpha, \beta)$$

so

$$P(\theta_i) = \frac{\theta_i^{\alpha-1} e^{-\theta_i/\beta}}{\beta^{\alpha}\Gamma(\alpha)}$$

Note that the θ_i are i.i.d. and are therefore exchangeable (see lecture 1).

For this example we will assume that the hyperparameter α is fixed and that the prior for the hyperparameters β is inverse gamma

$$\beta \sim IG(\gamma, \delta)$$

so

$$P(\beta) = \frac{\delta^{\gamma} e^{-\delta/\beta}}{\beta^{\gamma+1}\Gamma(\gamma)}$$

So we have a hierarchical model (see lecture 1).

The "hierarchy" of probability models that we have here is

$$
\begin{aligned}
x_i &\sim \text{Poisson}\,(\theta_i t_i) \\
\theta_i &\sim \text{Gamma}\,(\alpha, \beta) \\
\beta &\sim \text{Inverse Gamma}\,(\gamma, \delta)
\end{aligned}
$$

The joint posterior distribution is then given by

$$\pi(\theta_1,\ldots,\theta_{10},\beta \mid \boldsymbol{x}) \propto \prod_{j=1}^{10} \frac{e^{-\theta_j t_j}\left(\theta_j t_j\right)^{x_j}}{x_j!} \times \prod_{j=1}^{10} \frac{\theta_j^{\alpha-1} e^{-\theta_j/\beta}}{\beta^{\alpha}\Gamma(\alpha)} \times \frac{\delta^{\gamma} e^{-\delta/\beta}}{\beta^{\gamma+1}\Gamma(\gamma)}$$

We will use a Gibbs sampler to construct a Markov chain that will enable us to draw samples from the joint posterior distribution of $\theta_1,\ldots,\theta_{10}$ and β.

To do this we need to determine full conditional distributions of each of the unknown parameters...

The full conditional of θ_i is proportional to the joint posterior distribution, so

$$P(\theta_i \mid \boldsymbol{\theta}_{-i}, \beta) \propto \prod_{j=1}^{10} \frac{e^{-\theta_j t_j} \left(\theta_j t_j\right)^{x_j}}{x_j!} \times \prod_{j=1}^{10} \frac{\theta_j^{\alpha-1} e^{-\theta_j/\beta}}{\beta^{\alpha}\Gamma(\alpha)} \times \frac{\delta^{\gamma} e^{-\delta/\beta}}{\beta^{\gamma+1}\Gamma(\gamma)}$$

As we're thinking of this as a function of θ_i we can divide through by all those terms which don't contain θ_i, hence

$$P(\theta_i \mid \boldsymbol{\theta}_{-i}, \beta) \propto e^{-\theta_i t_i} \theta_i^{x_i} \times \theta_i^{\alpha-1} e^{-\theta_i/\beta}$$

So, rearranging, we have

$$P\left(\theta_i \mid \boldsymbol{\theta}_{-i}, \beta\right) \propto e^{-\theta_i t_i - \theta/\beta} \theta_i^{x_i + \alpha - 1}$$

This is the general form for a Gamma distribution. Hence the full conditional for θ_i is given by

$$\theta_i \mid \beta, \boldsymbol{x} \sim G\left(\alpha + x_i, \left(t_i + \frac{1}{\beta}\right)^{-1}\right)$$

Similarly, we find that the full conditional for β is given by

$$\beta \mid \theta_1 \ldots, \theta_{10}, \boldsymbol{x} \sim IG\left(\gamma + \alpha p, \sum \theta_j + \delta\right)$$

So the Gibbs sampler updates $\theta_1, \ldots, \theta_{10}$ and β in turn by drawing new values from their conditional distributions.

Before starting we need to give values to the fixed parameters α, γ and δ. We choose to use

$$\begin{aligned} \gamma &= 0.1 \\ \delta &= 1.0 \\ \alpha &= \bar{\rho}^2 / \left(S_p^2 - p^{-1} \bar{\rho} \sum t_i^{-1} \right) \end{aligned}$$

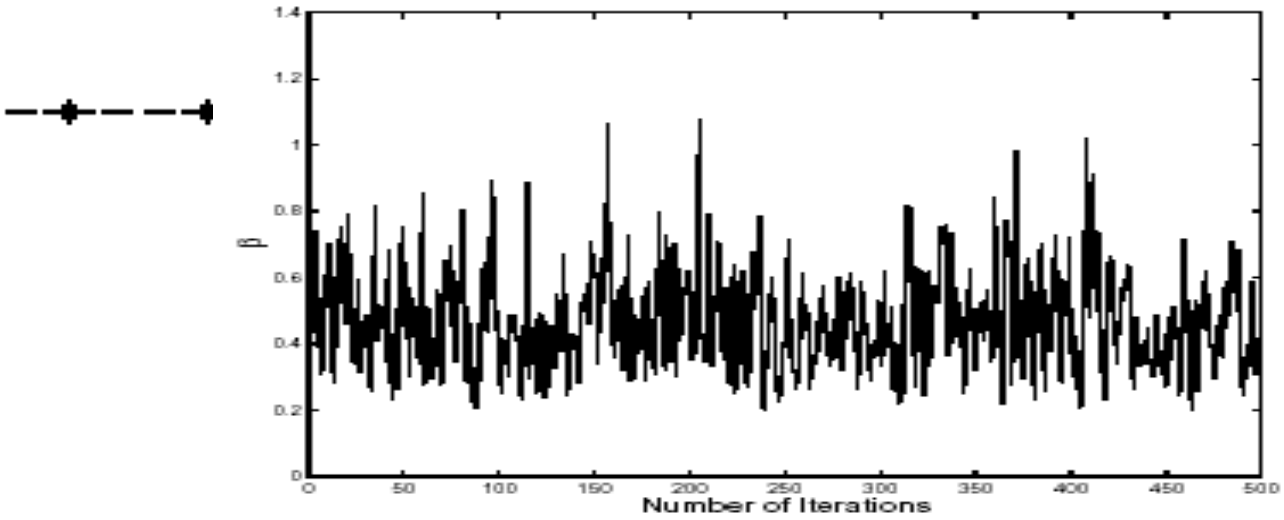
The choice of α comes from estimates of the mean and variance of the θ_i calculated from the data:

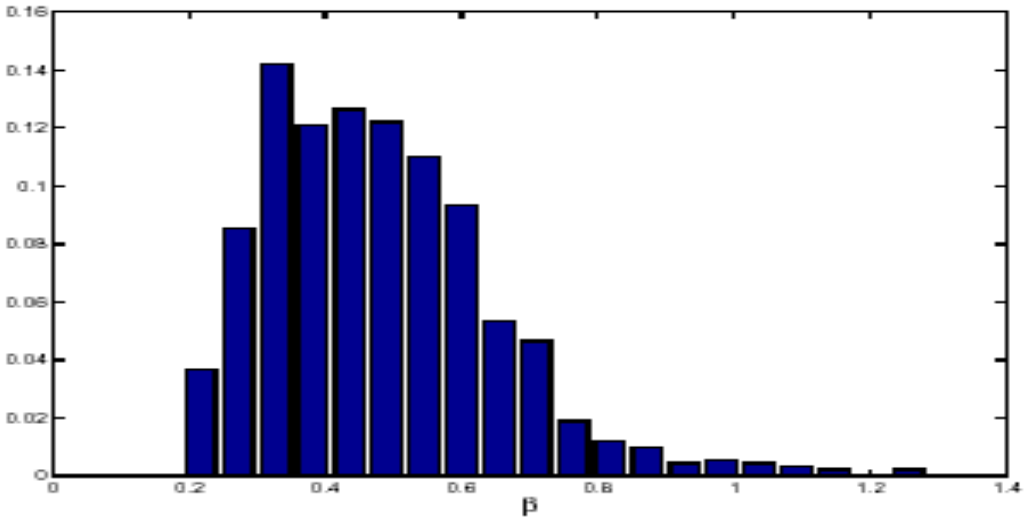
$$\begin{aligned} \mathbb{E}(\theta_i) &= \alpha/\beta \\ &\approx \bar{\rho} \\ V(\theta_i) &= \left(\alpha/\beta^2\right) + (\alpha/\beta t_i) \\ &\approx S_p^2 = p^{-1} \sum (\rho_i - \bar{\rho})^2 \end{aligned}$$

We need to choose starting values for $\theta_1, \ldots, \theta_{10}$ and β.

In this case sensible starting points are

$$\theta_i^{(0)} \approx \rho_i = x_i/t_i$$
$$\beta^{(0)} \approx \alpha/\bar{\rho} = \alpha p/\sum \theta_i^{(0)}$$


β
Number of Iterations


β

Mean value of sample of posterior distribution for β is 0.4795 and median is 0.4588.

We can use these as our best point estimates of β.

We can also use the MCMC sample to give a confidence interval for β.

To do this we order the sample from β and pick out the 2.5 and 97.5 percentiles.

In this case the confidence interval would be $[0.2330, 0.7830]$.

Convergence of MCMC

- Determing length of chain is hard
  - Initial set of samples discarded (burn-in)
  - Tests exist for stabilization, but unsatisfactory
- Trying to bound the mixing time
  - minimum # of steps for distribution of K to be close to *p(x)*

measure closeness with the total variation norm $\Delta_x(t)$, where

$$\Delta_x(t) = \|K^{(t)}(\cdot|x) - p(\cdot)\| = \frac{1}{2}\int \left(K^{(t)}(y|x) - p(y)\right) dy,$$

then the mixing time is

$$\tau_x(\epsilon) = \min\left\{t : \Delta_x(t') \leq \epsilon \text{ for all } t' \geq t\right\}.$$

Convergence of MCMC (cont'd)

$$\Delta_x(t) \leq \frac{1}{2\sqrt{p(x)}}\lambda_\star^t,$$

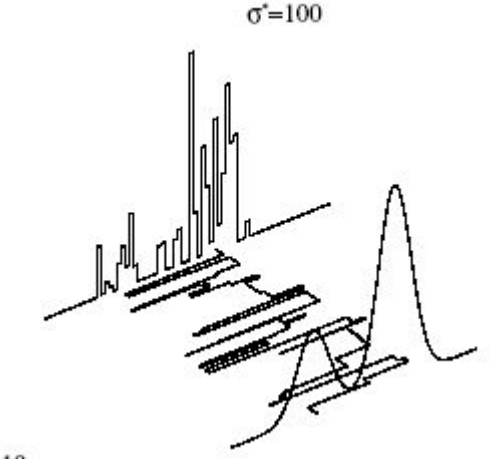
- Bound on total variation norm:
- Second eigenvalue can also be bounded
- Implications: simple MCMC algorithms
  - (such as Metropolis)
  - Run in time polynomial in dim(state space)
  - Polynomial algorithm scenarios:
    - Volume of convex body for large # dimensions
    - Sampling from log-concave distributions
    - Sampling from truncated multivariate Gaussians
    - Sampling matches from a bipartite graph (stereo)

Perfect Sampling

- Algorithms guaranteed to produce an independent sample from $p(x)$
- Current limited, computationally expensive
- Some work on general perfect samplers
  - perfect slice samplers

Adaptive MCMC: Motivation

- We would like to stay in "good" states for a long time, thus reduce variance of proposal distribution
- Automate choosing of proposal distribution such that (one of):
  - q is closer to the target distribution
  - We ensure a good acceptance rate
  - Minimize the variance of the estimator
- Too much adaptation is bad
  - Violates Markov property of K
  - $p(x_i|x_0...x_{i-1})$ no longer becomes $p(x_i|x_{i-1})$



from Andrieu et al. An Introduction to MCMC for Machine Learning. *Machine Learning*, 2002.

Adaptive MCMC: Methods

- Preserve Markov property by adapting only during initial fixed # steps
  - Then use standard MCMC to ensure convergence
  - Gelfand and Sahu 1994
    - Run several chains in parallel and use sampling-importance-resampling to multiply kernels doing well, suppress others
    - Monitor transition kernel and change components (like q) to improve mixing
- Other methods allow continuous adaptation
  - Retaining Markov Property
  - Delayed rejection, Parallel chains, Regeneration
  - Inefficient