Markov Chain Monte Carlos

S. Vasishth

Potsdam

August 12, 2015

The goal

Our main goal is always to estimate properties of posterior probability distributions.
Suppose we have a random variable $X \sim f(x)$. What is its mean? We know how to calculate the mean analytically if f(x) is "solvable".

$$\mu = \int_{\infty}^{\infty} x f(x)\, dx \tag{1}$$

Example

The expectation of the standard normal random variable

$$E[Z] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} x e^{-x^2/2}\, dx$$

Let $u = -x^2/2$.
Then, $du/dx = -2x/2 = -x$. I.e., $du = -x\, dx$ or $-du = x\, dx$.
We can rewrite the integral as:

$$E[Z] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^u x\, dx$$

Replacing $x\, dx$ with $-du$ we get:

$$-\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^u\, du$$

which yields:

$$-\frac{1}{\sqrt{2\pi}} [e^u]_{-\infty}^{\infty}$$

Example

The expectation of the standard normal random variable

Replacing u with $-x^2/2$ we get:

$$-\frac{1}{\sqrt{2\pi}}[e^{-x^2/2}]_{-\infty}^{\infty} = 0$$

Using Monte Carlo Integration

Suppose that $f(x)$ is not solvable, but suppose that we can get samples from X: $x_1, \dots, x_n$.
We can now estimate μ:

$$\hat{\mu} = \frac{1}{n} \sum x_i \tag{2}$$

This estimate is unbiased:
$E(\hat{\mu}) = \frac{1}{n} \sum (E(X_i)) = \frac{1}{n} nE(X) = \mu$
For large n, $Var(\hat{\mu}) = \frac{Var(X)}{n}$. I.e., variance tends to zero as $n \rightarrow \infty$.

Example

Expectation of standard normal random variable using sampling

```
> x<-rnorm(1000,mean=0,sd=1)
> mean(x) ## cf. analytical value 0
[1] 0.06285426
```

We can also compute quantities like $P(X < 1.96)$ by sampling:

```
> counts<-table(x<1.96)[2]
> ## pretty close to the theoretical value:
> counts/length(x)
 TRUE
0.982
> ## theoretical value:
> pnorm(1.96)
[1] 0.9750021
```

The posterior distribution

In the bayesian setting, we often cannot derive the posterior distribution. *But we can always write it up to proportionality*:

$$f(\theta \mid x) \propto f(\theta) f(x \mid \theta) \tag{3}$$

We often can't figure out $\int f(\theta) f(x \mid \theta)\, d\theta$. **But maybe we can make it disappear**.

The goal again

The goal is to produce samples $\theta_1, \theta_2, \ldots$ from $f(\theta \mid x)$.
The MCMC approach will produce a sample even if we know $f(\theta \mid x)$ only up to proportionality.

Markov Chain sampling

We have been doing non-Markov chain sampling in the introductory course:

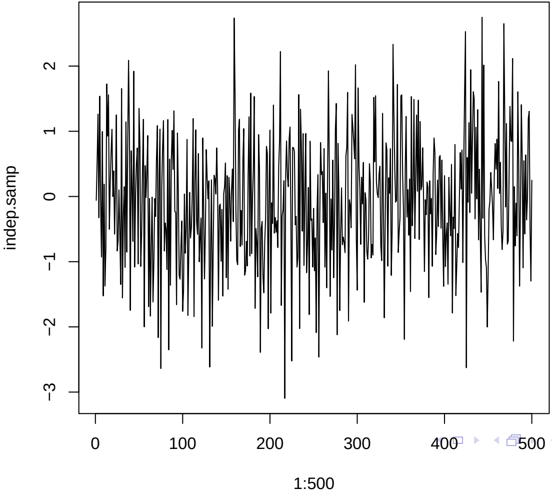
```
> indep.samp<-rnorm(500,mean=0,sd=1)
> head(indep.samp,n=3)

[1] -0.05879067  0.52808900  1.26280858
```

The vector of values sampled here are statistically independent.

Markov Chain sampling

Independent samples:

```
> plot(1:500,indep.samp,type="l")
```

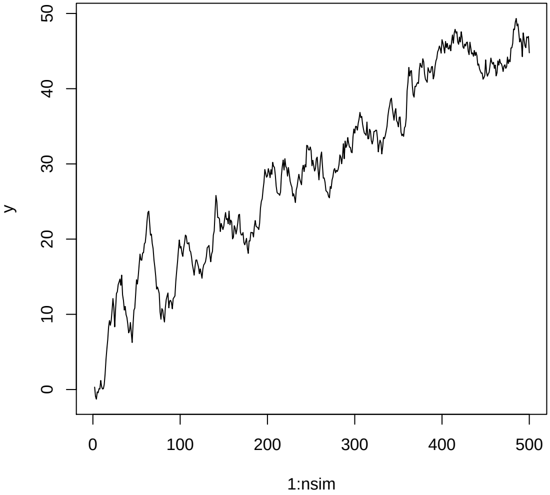


Markov Chain sampling

If the current value influences the next one, we have a Markov chain. Here is a Markov chain: the i-th draw is dependent on the i-1 th draw:

```
> nsim<-500
> x<-rep(NA,nsim)
> y<-rep(NA,nsim)
> x[1]<-rnorm(1) ## initialize x
> for(i in 2:nsim){
+ ## draw i-th value based on i-1-th value:
+ y[i]<-rnorm(1,mean=x[i-1],sd=1)
+ x[i]<-y[i]
+ }
> plot(1:nsim,y,type="l")
```

Markov Chain sampling



Formal definition of Markov Chains

Suppose we are given a sequence of random variables $X_1, X_2, \ldots$ with a **transition kernel** which tells us the probability distribution of X_{t+1} given X_t:

Transition kernel: $P(X_{t+1} \mid X_t)$.

In a Markov Chain: $P(X_{t+1} \mid X_1, \ldots, X_t) = P(X_{t+1} \mid X_t)$.

Stationarity

Suppose we have X_1. Let $P^t(X_t \mid X_1)$ be the distribution of X_t.
$P^t(X_t \mid X_1)$ will converge to a **stationary distribution** $\phi(X)$ if the Markov Chain has certain properties: if it is ergodic, i.e.,

- irreducible (can eventually visit every possible state)
- aperiodic (some cycle of values repeats only once),
- positive recurrent (will eventually return to any given start state with prob. 1).

Convergence in discrete Markov Chains

Recall that a Markov Chain defines a probabilistic move from one state to the next.
Suppose we have 6 states; a **transition matrix** can define the probabilities:

```
> ## Set up transition matrix:
> T<-matrix(rep(0,36),nrow=6)
> diag(T)<-0.5
> offdiags<-c(rep(0.25,4),0.5)
> for(i in 2:6){
+ T[i,i-1]<-offdiags[i-1]
+ }
> offdiags2<-c(0.5,rep(0.25,4))
> for(i in 1:5){
+ T[i,i+1]<-offdiags2[i]
+ }
```

Convergence

```
> T

     [,1] [,2] [,3] [,4] [,5] [,6]
[1,] 0.50 0.50 0.00 0.00 0.00 0.00
[2,] 0.25 0.50 0.25 0.00 0.00 0.00
[3,] 0.00 0.25 0.50 0.25 0.00 0.00
[4,] 0.00 0.00 0.25 0.50 0.25 0.00
[5,] 0.00 0.00 0.00 0.25 0.50 0.25
[6,] 0.00 0.00 0.00 0.00 0.50 0.50
```

Note that the rows sum to 1, i.e., the probability mass is distributed over all possible transitions from any one location:

```
> rowSums(T)

[1] 1 1 1 1 1 1
```

Convergence

We can represent a current state as a probability vector: e.g., in state one, the transition probabilities for possible moves are:

```
> T[1,]
[1] 0.5 0.5 0.0 0.0 0.0 0.0
```

We can also simulate a non-determinsitic random walk based on a state like T[1,]:

```
> sample(1:6,size=1,prob=T[1,])
[1] 1
> sample(1:6,size=1,prob=T[1,])
[1] 1
> sample(1:6,size=1,prob=T[1,])
[1] 2
```

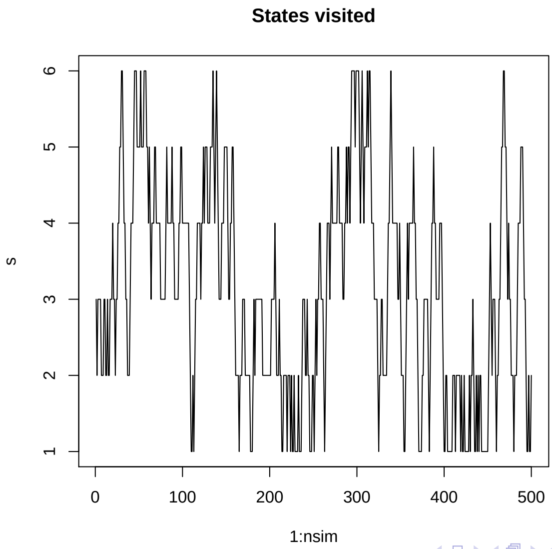
Convergence

A non-deterministic random walk:

```
> nsim<-500
> s<-rep(0,nsim)
> ## initialize:
> s[1]<-3
> for(i in 2:nsim){
+   s[i]<-sample(1:6,size=1,prob=T[s[i-1],])
+ }
> plot(1:nsim,s,type="l",main="States visited")
```

Convergence

A non-deterministic random walk:



Convergence

This Markov chain converges to a particular distribution of probabilities of visiting states 1 to 6. We can see the convergence happen by examining the proportions of visits to each state after blocks of steps that increase by 500 steps.

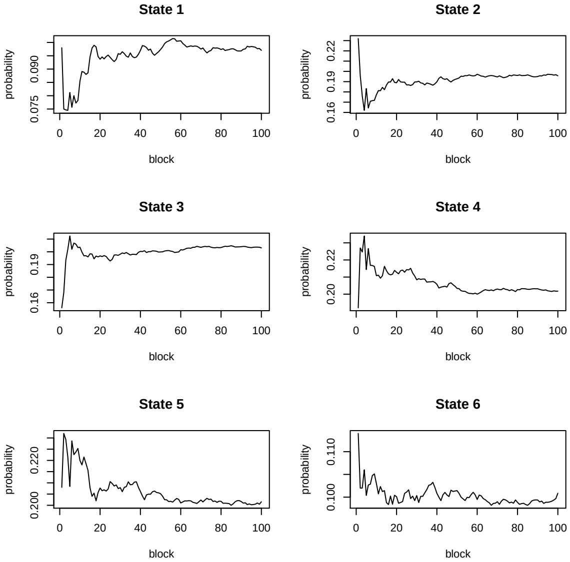
Convergence

```
> nsim<-50000
> s<-rep(0,nsim)
> ## initialize:
> s[1]<-3
> for(i in 2:nsim){
+   s[i]<-sample(1:6,size=1,prob=T[s[i-1],])
+ }
> blocks<-seq(500,50000,by=500)
> n<-length(blocks)
> ## store transition probs over increasing blocks:
> store.probs<-matrix(rep(rep(0,6),n),ncol=6)
> ## compute relative frequencies over increasing blocks:
> for(i in 1:n){
+   store.probs[i,]<-table(s[1:blocks[i]])/blocks[i]
+ }
```

Convergence


```
> op <- par(mfrow=c(3,2))
> for(i in 1:6){
+ plot(1:n,store.probs[,i],type="l",lty=1,xlab="block",
+       ylab="probability",main=paste("State ",i,sep=""))
+ }
```

Convergence



Convergence

Note that each of the rows of the `store.probs` matrix is a probability mass function, which defines the probability distribution for the 6 states:

```
> store.probs[1,]
```

```
[1] 0.098 0.232 0.156 0.192 0.208 0.114
```

This distribution is settling down to a particular set of values; that's what we mean by convergence. This particular set of values is:

```
> (w<-store.probs[n,])
```

```
[1] 0.09704 0.19572 0.20306 0.20176 0.20156 0.10086
```

Convergence

w is called a **stationary** distribution. If $wT = w$, then w is the stationary distribution of the Markov chain.

```
> round(w%*%T,digits=2)

     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]  0.1  0.2  0.2  0.2  0.2  0.1

> round(w,digits=2)

[1] 0.1 0.2 0.2 0.2 0.2 0.1
```

This discrete example gives us an intuition for what will happen in continuous distributions: we will devise a Markov chain such that the chain will converge to the distribution we are interested in sampling from.

Convergence

If by time T the Markov Chain reaches its stationary distribution (reaches equilibrium) then $X_{T+1},\ldots,X_{T+n}$ will be samples from the density function $\phi(X)$.
Once we are at equilibirum, we can use $\frac{1}{n}\sum X_{T+i}$ to estimate $E(X)$.

Independence

Note that X_{T+i} and X_{T+i+1} will still not be independent.
But the samples will be from $\phi(X)$.
As $n \rightarrow \infty$, the estimator will converge to $E(X)$.

A key result

For any pdf $f(\theta \mid x)$, it is possible to construct a Markov Chain $\theta_1, \theta_2, \ldots$ whose stationary distribution is $\phi(X) = f(\theta \mid x)$.

The method: Metropolis-Hastings

1. Let $f(X)$ be the target density.
2. Let X_t be the value of the Markov Chain at time t.
3. We need to define the transition kernel $P(X_{t+1} \mid X_t)$ which states the probability with which the different X_{t+1} values would be generated.

The method: Metropolis-Hastings

We have an initial value X_t.

1. Define a proposal density $q(Y \mid X_t)$. Example: $N(mean = X_t, sd = 1)$.
2. Generate a candidate point Y from $q(Y \mid X_t)$.
3. Set
   3.1 $X_{t+1} \leftarrow Y$ with probability $\alpha(X_t, Y)$,
   3.2 $X_{t+1} \leftarrow X_t$ with probability $1 - \alpha(X_t, Y)$

   where $\alpha(X_t, Y) = min\{1, \frac{f(Y)}{f(X_t)} \frac{q(X_t|Y)}{q(Y|X_t)}\}$.

   To make the above decision, sample $U \sim Unif(0,1)$.

   3.1 If $U \leq \alpha(X_t, Y)$, set $X_{t+1} \leftarrow Y$.
   3.2 If $U > \alpha(X_t, Y)$, set $X_{t+1} \leftarrow X_t$.

Implementing Metropolis-Hastings

Cauchy random variables

$$f(x) = \frac{k}{1+\theta^2} \quad \text{where } k = \pi^{-1} \text{ (k will cancel out)} \tag{4}$$

Proposal density $q(\theta \mid \theta_t)$, let this be $N(\theta_t, 1)$, where θ_t is the current value of θ.
It follows that

$$q(\theta \mid \theta_t) = \frac{1}{\sqrt{2\pi}} \exp -\frac{1}{2} \frac{(\theta - \theta_t)^2}{1} \tag{5}$$

Note that $q(\theta \mid \theta_t) = q(\theta_t \mid \theta)$. This means that

$$\alpha(X_t, Y) = min\{1, \frac{f(Y)}{f(X_t)} \frac{q(X_t \mid Y)}{q(Y \mid X_t)}\} = min\{1, \frac{f(Y)}{f(X_t)}\} \tag{6}$$

Implementing Metropolis-Hastings

Cauchy random variables

It follows that

$$\begin{aligned} \alpha(\theta_t,\theta) =& min\{1, \frac{f(\theta)}{f(\theta_t)}\} \\ =& min\{1, \frac{1+\theta_t^2}{1+\theta^2}\} \end{aligned} \tag{7}$$

In other words:

$$\alpha(\theta_t,\theta) = \begin{cases} 1 & \text{if } \mid \theta_t \mid > \mid \theta \mid, \\ \frac{1+\theta_t^2}{1+\theta^2}, & \text{otherwise} \end{cases} \tag{8}$$

Implementing Metropolis-Hastings

Cauchy random variables

As a starting value, choose the mode of a cauchy: $\theta_1 = 0$.

```
Markov Chain so far: <0>
```

Implementing Metropolis-Hastings

Cauchy random variables

Generate candidate point Y from $q(\theta \mid theta_1 = 0)$.

```
> ## initial value:
> theta.t<-0
> ## always give the same result:
> set.seed(43210)
> ## generate candidate:
> (Y<-rnorm(1,mean=theta.t,sd=1))
[1] -0.4311743
```

Since $\mid Y \mid > \mid \theta_1 \mid$, we have

$$\begin{aligned}\alpha(\theta_t, Y) =& \frac{1+\theta_t^2}{1+Y^2} \\ =& \frac{1+0}{1+(-0.43117)^2} \\ =& 0.84323\end{aligned} \tag{9}$$

Implementing Metropolis-Hastings

Cauchy random variables

Next, generate a uniform random variable:

```
> (U <- runif(1,0,1))
[1] 0.3535107
```

This step allows us to decide whether to probabilistically accept or reject Y:

(a) If $U \leq \alpha(X_t, Y)$, set $X_{t+1} \leftarrow Y$.
(b) If $U > \alpha(X_t, Y)$, set $X_{t+1} \leftarrow X_t$.

Here, the first condition holds, so we set X_{t+1} to Y.

```
Markov chain: <0,-0.43117>
```

Now start over, with $\theta_t = -0.43117$ instead of $\theta_t = 0$.

Implementing Metropolis-Hastings

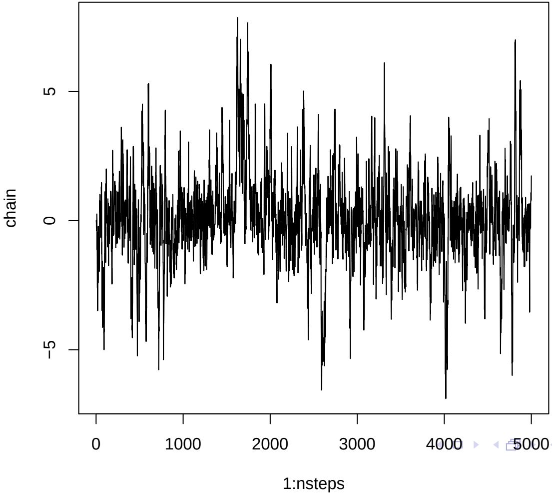
In-class Exercise

1. Implement the Metropolis-Hastings algorithm for the cauchy case illustrated above. Run 5,000 simulations and assess convergence visually ("fat hairy caterpillars").
2. Once you have the algorithm working properly, run three chains, each with three different initial values (-100,0,100), and plot them together in one figure.
3. Discard the first 2000 runs (burn-in or warm-up) and compute $P(0 < \theta < 1)$ from the sampled chains.

Implementing Metropolis-Hastings

Exercise

```
> plot(1:nsteps,chain,type="l")
```



Implementing Metropolis-Hastings

Exercise

```
> plot(1:nsteps,chains[1,],type="l",ylim=c(-100,100))
> lines(1:nsteps,chains[2,],col="red")
> lines(1:nsteps,chains[3,],col="orange")
```