Part A Simulation and Statistical Programming HT14

Lecturer: Geoff Nicholls

University of Oxford

Lecture 10: MCMC!

Notes and Problem sheets are available at

www.stats.ox.ac.uk\ $\sim$nicholls\PartASSP

Denote by $P'_{i,j} = \mathbb{P}(X_{t-1} = j | X_t = i)$ the transition matrix for the time-reversed chain.

It seems clear that a Markov chain will be reversible if and only if $P = P'$, so that any particular transition occurs with equal probability in forward and reverse directions.

Theorem.
(I) If there is a probability mass function $\pi(i), i \in \Omega$ satisfying $\pi(i) \geq 0$, $\sum_{i \in \Omega} \pi(i) = 1$ and

"Detailed balance": $\pi(i) P_{i,j} = \pi(j) P_{j,i}$ for all pairs $i, j \in \Omega$,

then $\pi = \pi P$ so π is stationary for P.

(II) If in addition $p^{(0)} = \pi$ then $P' = P$ and the chain is reversible with respect to π.

Proof of (I): sum both sides of detailed balance equation over $i \in \Omega$. Now $\sum_i P_{j,i} = 1$ so $\sum_i \pi(i) P_{i,j} = \pi(j)$.

Proof of (II), we have π a stationary distribution of P so $\mathbb{P}(X_t = i) = \pi(i)$ for all $t = 1, 2, \ldots$ along the chain. Then

$$\begin{aligned} P'_{i,j} &= \mathbb{P}(X_{t-1} = j | X_t = i) \\ &= \mathbb{P}(X_t = i | X_{t-1} = j) \frac{\mathbb{P}(X_{t-1} = j)}{\mathbb{P}(X_t = i)} \text{ (Bayes rule)} \\ &= P_{j,i} \pi(j) / \pi(i) \text{ (stationarity)} \\ &= P_{i,j} \text{ (detailed balance).} \end{aligned}$$

Convergence and the Ergodic Theorem

If the (finite state space) MC is irreducible and aperiodic then the stationary distribution is unique and $p^{(t)} \to \pi$ as $t \to \infty$. If we simulate the MC $X_0, X_1, ...X_n$ to large enough n from any start $X_0 = x_0$ then since $X_t \sim p^t$ and $p^t \simeq \pi$ at large t, 'most' of the samples are 'nearly' distributed according to π.

We will use $\{X_t\}_{t=0}^{n-1}$ to estimate $\mathbb{E}_p(\phi(X))$. The 'obvious' estimator is

$$\hat{\phi}_n = \frac{1}{n} \sum_{t=0}^{n-1} \phi(X_t),$$

but the X_t are correlated and only converge in distribution to π.

Theorem. If $\{X_t\}_{t=0}^{\infty}$ is an irreducible and aperiodic Markov chain on a finite space of states Ω, with stationary distribution π then, as $n \to \infty$, for any bounded function $\phi : \Omega \to R$,

$$P(X_n = i) \to \pi(i) \text{ and } \hat{\phi}_n \to \mathbb{E}_p(\phi(X)).$$

We refer to such a chain as ergodic with equilibrium π.

In Part A Probability the Ergodic theorem asks for positive recurrent $X_0, X_1, X_2, \ldots$. It is simpler here because we are assuming a finite state space for the Markov chain.

We would like a CLT for $\hat{\phi}_n$, and confidence intervals $\pm\sqrt{\text{var}(\hat{\phi}_n)}$. A CLT does hold for all the examples discussed later but this is a little beyond us at this point.

The Metropolis-Hastings Algorithm

The Metropolis-Hastings (MH) algorithm determines a Markov Chain ergodic for your given target distribution (p say).

MCMC is an algorithm for simulating X_{t+1} given X_t. The algorithm determines the transition probabilities $P(X_{t+1} = j | X_t = i)$ and the transition matrix P. We have to choose the algorithm so that the MC is ergodic for p. The transition matrix it simulates should satisfy $pP = p$.

Let $p(x) = \tilde{p}(x)/Z_p$ be the pmf on finite state space $\Omega = \{1, 2, ..., m\}$. We will call p the (pmf of the) target distribution.

To set things up, first choose a 'proposal' transition matrix $q(y|x)$ which is simple to simulate, easy to calculate, irreducible, and satisfying $q(x|y) > 0 \Rightarrow q(y|x) > 0$.

Claim: The following algorithm simulates a Markov chain. If the the chain is irreducible and aperiodic then it is ergodic with equilibrium distribution p.

Let $X_t = x$. X_{t+1} is determined in the following way.

[1] Draw $y \sim q(\cdot|x)$ and $u \sim U[0,1]$.

[2] If

$$u \leq \alpha(y|x) \text{ where } \alpha(y|x) = \min\left\{1, \frac{\tilde{p}(y)q(x|y)}{\tilde{p}(x)q(y|x)}\right\}$$

set $X_{t+1} = y$, otherwise set $X_{t+1} = x$.

We initialise this with $X_0 = x_0, p(x_0) > 0$ and iterate for $t = 1, 2, 3, ...n$ to simulate the samples $X_0, X_1, X_2, ...X_n$ we need.

Proof: By the ergodic theorem it is sufficient to show that the Markov chain determined by the random MCMC update has p as a stationary distribution.

We will compute the transition matrix P and show it satisfies detailed balance,

$$P_{x,y}\, p(x) = P_{y,x}\, p(y),$$

since that implies $p = pP$.

We don't need to calculate $P_{x,y}$ when $x = y$ as DB is clear. Suppose $y \neq x$. If $X_t = x$, then the probability $P_{x,y}$ to move to $X_{t+1} = y$ at the next step is the probability to propose y at step 1 times the probability to accept it at step 2, so

$$P_{x,y} = P(X_{t+1} = y | X_t = x) = q(y|x)\alpha(y|x).$$

Now check DB:

$$\begin{aligned} p(x)P_{x,y} &= p(x)q(y|x)\alpha(y|x) \\ &= p(x)q(y|x)\min\left\{1, \frac{p(y)q(x|y)}{p(x)q(y|x)}\right\} \\ &= \min\{p(x)q(y|x), p(y)q(x|y)\} \\ &= p(y)q(x|y)\min\left\{\frac{p(x)q(y|x)}{p(y)q(x|y)}, 1)\right\} \\ &= p(y)q(x|y)\alpha(x|y) \\ &= p(y)P_{y,x} \end{aligned}$$

and we are done.

Example: Simulating a Discrete Distribution

Let $p(i) = i/Z_p$ with $Z_p = \sum_{i=1}^{m} i$.

Give a MH MCMC algorithm ergodic for $p(i), i = 1, 2, ..., m$.

Step 1: Choose a proposal distribution $q(j|i)$. It needs to be easy to simulate and determine a irreversible chain.

A simple distribution that 'will do' is $Y \sim U\{1, 2, ..., m\}$, so

$$q(i) = 1/m, \quad i = 1, 2, ..., m.$$

This proposal scheme is clearly irreducible (we can get from A to B in a single hop) and satisfies $q(x|y) > 0 \Rightarrow q(y|x) > 0$ (since q is constant).

Step 2: write down the algorithm.

If $X_t = x$, then X_{t+1} is determined in the following way.

[1] Simulate $y \sim U\{1, 2, ..., m\}$ and $u \sim U[0, 1]$.

[2] If

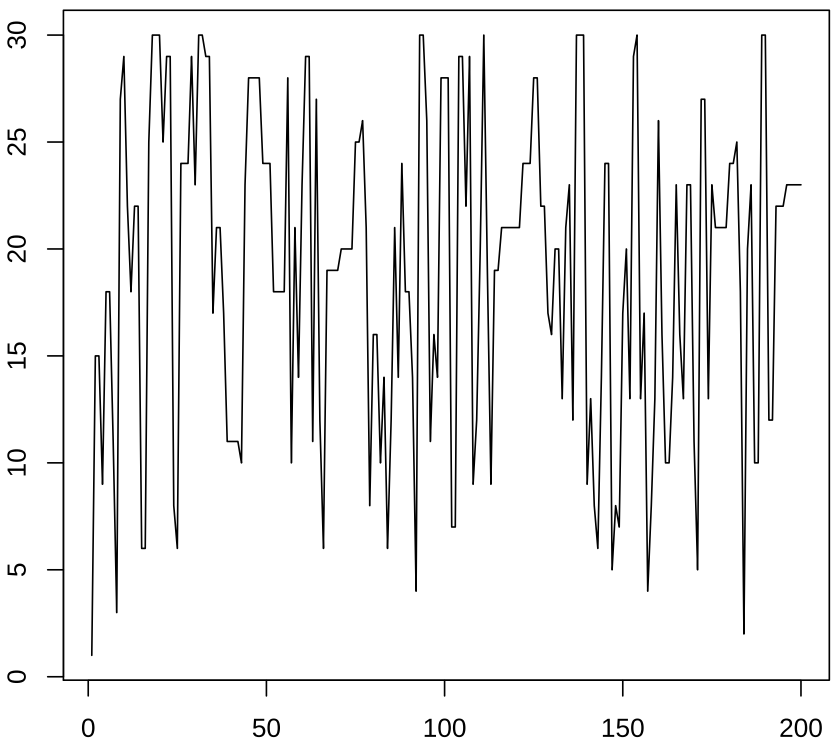
$$\begin{aligned} u &\leq \min\left\{1, \frac{\tilde{p}(y)q(x|y)}{\tilde{p}(x)q(y|x)}\right\} \\ &= \min\left\{1, \frac{y}{x}\right\} \end{aligned}$$

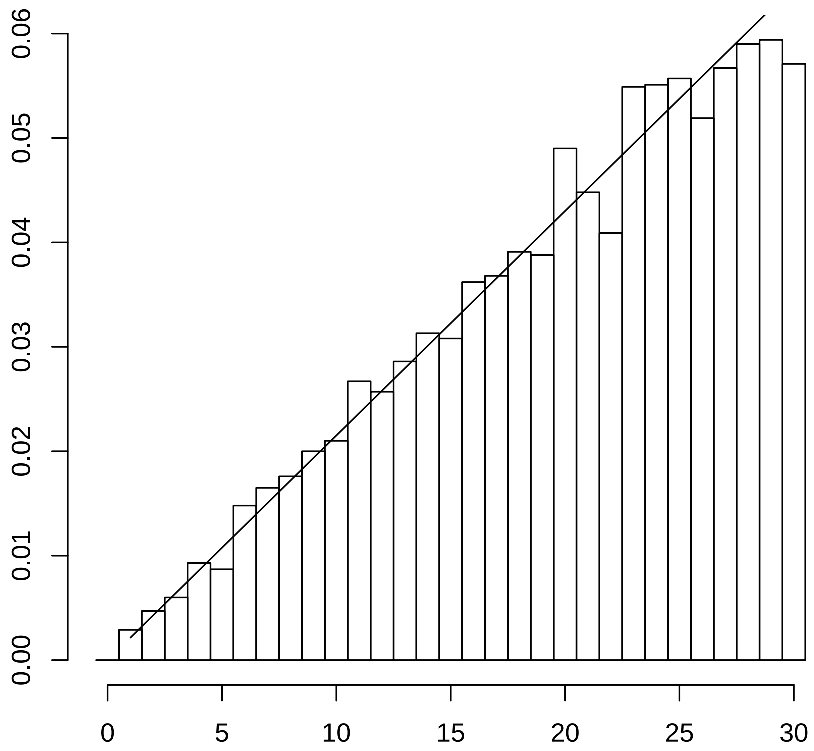
set $X_{t+1} = y$, otherwise set $X_{t+1} = x$.

```
#MCMC simulate X_t according to p=[1:m]/sum(1:m).
m<-30
n<-10000; X<-rep(NA,n); X[1]<-1
for (t in 1:(n-1)) {
   x<-X[t]
   y<-ceiling(m*runif(1))
   a<-min(1,y/x)
   U<-runif(1)
   if (U<=a) {
      X[t+1]<-y
   } else {
      X[t+1]<-x
   }
}
```

Left: x-axis is Markov chain step counter $t = 1, 2, 3 \ldots 200$ and y-axis is Markov chain state X_t for $\tilde{p}(i) = i, i = 1, 2, \ldots, m$, $m = 30$.

Right: histogram of $X_1, X_2, \ldots, X_n$ for $n = 1000$.





Example: Simulating a Poisson Distribution

We want to simulate $p(x) = e^{-\lambda}\lambda^x/x! \propto \lambda^x/x!$

Step 1: Define the proposal. We want something simple to simulate with a simple density. We use

$$q(y|x) = \begin{cases} \frac{1}{2} & \text{for } y = x \pm 1 \\ 0 & \text{otherwise,} \end{cases}$$

i.e. toss a coin and add or subtract 1 to x to obtain y.

Step 2: Write down the algorithm, giving the acceptance probability $\alpha(y|x)$ in closed form.

If $X_t = x$, then X_{t+1} is determined in the following way.

[1] Simulate $y \sim U\{x-1, x+1\}$ and $u \sim U[0,1]$.

[2] The acceptance probability is $\alpha(y|x) = \min\left\{1, \frac{\tilde{p}(y)q(x|y)}{\tilde{p}(x)q(y|x)}\right\}$.
If $y = -1$ then $\alpha = 0$. Otherwise,

$$\alpha(y|x) = \begin{cases} \min\left(1, \frac{\lambda}{x+1}\right) & \text{if } y = x+1 \\ \min\left(1, \frac{x}{\lambda}\right) & \text{if } y = x-1 \end{cases}$$

If $u \leq \alpha(y|x)$, set $X_{t+1} = y$, otherwise set $X_{t+1} = x$.

MCMC for state spaces which are not finite Does this work for continuous rv? Computers use the "computer measure". The reals are discretised.

Let x^* be the computer truncation of x and $\delta x = \{y : x^* = x\}$. The length $|\delta x|$ of the cell containing x is not constant. Roughly $|\delta x|/x \simeq 10^{-15}$. $\pi(x)$ is approximated by $[\pi(x)]^*$.

The Hastings ratio we compute is

$$\begin{aligned}\frac{[\tilde{p}(y)]^*[q(x|y)]^*}{[\tilde{p}(x)]^*[q(y|x)]^*} &= \frac{[p(y)]^*|\delta y|[q(x|y)]^*|\delta x|}{[p(x)]^*|\delta x|[q(y|x)]^*|\delta y|} \\ &\simeq \frac{\Pr(Y \in \delta y)\Pr(X \in \delta x|Y = y)}{\Pr(X \in \delta x)\Pr(Y \in \delta y|X = x)}\end{aligned}$$

since $\Pr(X \in \delta x) \simeq [p(x)]^*|\delta x|$ etc. If we apply this to densities, we simulate the approximate distribution. Our discussion of Markov chains on finite spaces is relevant.

MCMC for the Normal distribution

Suppose want to simulate the standard normal distribution $X \sim N(0,1)$. The target density is

$$\tilde{p}(x) \propto \exp(-x^2/2).$$

Step 1: Choose the proposal distribution. We need something simple and irreducible. Fix a constant $a > 0$ and choose a new point uniformly at random in a window of length $2a$ centred at x. The proposal density is

$$q(y|x) = \frac{1}{2a}\mathbb{I}_{x-a<y<x+a}$$

Notice that $q(y|x) = q(x|y)$.

Step 2: give the MCMC algorithm. If $X_t = x$ then X_{t+1} is determined in the following way:

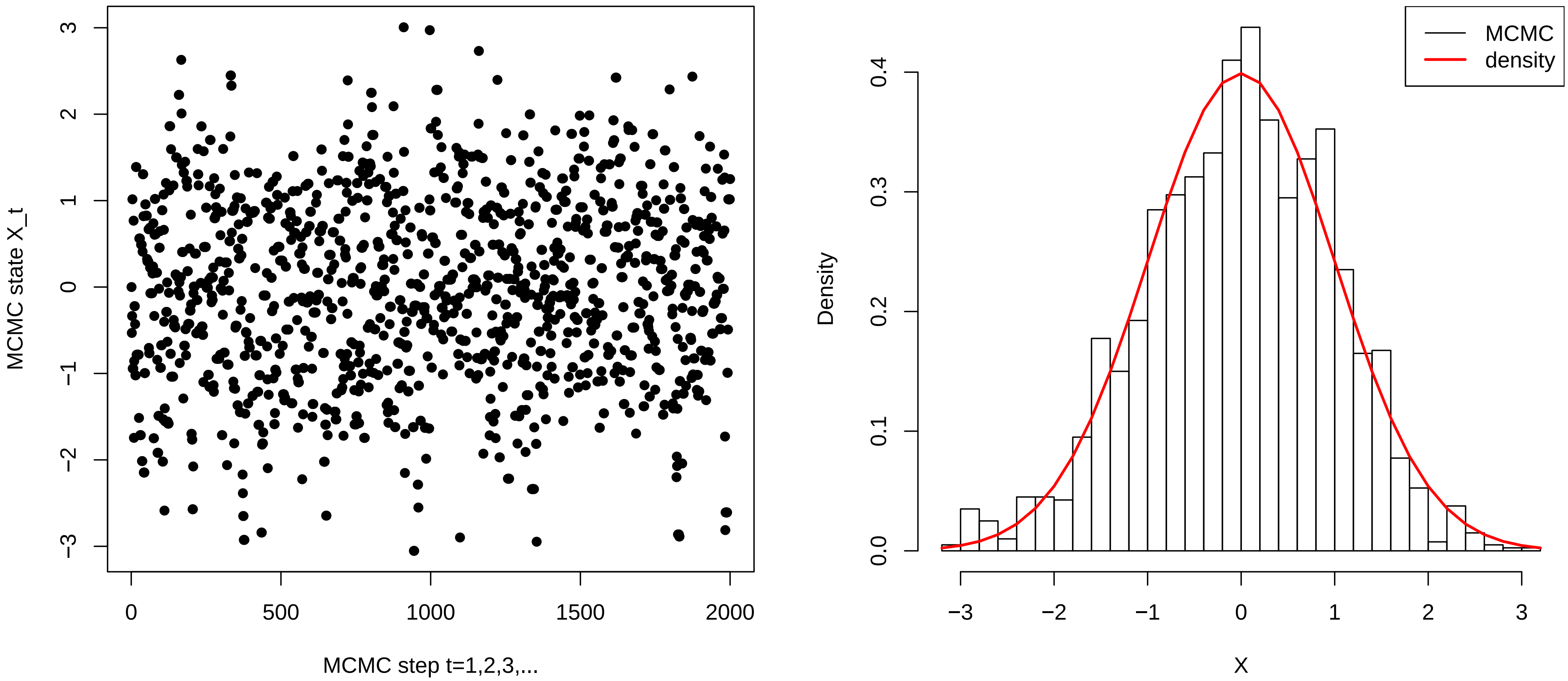
[1] Simulate $Y \sim U(x-a, x+a)$ and $U \sim U(0,1)$.

[2] If $U \leq \alpha(y|x)$ set $X_{t+1} = y$ and otherwise set $X_{t+1} = x$. Here

$$\begin{aligned}\alpha(y|x) &= \min\left(1,\ \frac{\tilde{p}(y)q(x|y)}{\tilde{p}(x)q(y|x)}\right) \\ &= \min\left(1, \exp(-y^2/2 + x^2/2)\right).\end{aligned}$$

```
#MCMC simulate X_t ~ N(0,1)
a=3; n=2000
X=numeric(n); X[1]=0;
for (t in 1:(n-1)) {
   x<-X[t]
   y<-x+(2*runif(1)-1)*a
   if (runif(1)<exp((x^2-y^2)/2)) {
      X[t+1]<-y
   } else {
      X[t+1]<-x
   }
}
```

(see the associated R-file for plotting commands)



MH example: an equal mixture of bivariate normals

$$\pi(\theta) = (2\pi)^{-1}\left(0.5e^{-(\theta-\mu_1)\Sigma_1^{-1}(\theta-\mu_1)/2} + 0.5e^{-(\theta-\mu_2)\Sigma_2^{-1}(\theta-\mu_2)/2}\right)$$

with $\theta = (\theta_1, \theta_2)$. Use $\mu_1 = (1,1)^T$, $\mu_2 = (5,5)^T$ and $\Sigma_1 = \Sigma_2 = I_2$ for this illustration.

Step 1. For a proposal distribution q we want something simple to sample. The simplest thing I can think of is the same as before:

$$\theta_i' \sim U(\theta_i - a, \theta_i + a)$$

with a a fixed constant. Note that this time we are proposing in a box of side $2a$. That is easy to sample, and certainly $q(\theta'|\theta) > 0 \Leftrightarrow q(\theta|\theta') > 0$ since $q(\theta'|\theta) = q(\theta|\theta') = 1/4a^2$.

Step 2. The algorithm is, given $\theta^{(n)} = \theta$,
[1] for $i = 1, 2$ simulate $\theta'_i \sim U(\theta_i - a, \theta_i + a)$
[2] with probability

$$\alpha(\theta'|\theta) = \min\left\{1,\ \frac{\pi(\theta')}{\pi(\theta)}\right\}$$

set $\theta^{(n+1)} = \theta'$ otherwise set $\theta^{(n+1)} = \theta$.

This algorithm is ergodic for any $a > 0$ but we will see that the choice of a makes a difference to efficiency.

```
#MCMC simulate X_t according to a mixture of normals
f<-function(x,mu1,mu2,S1i,S2i,p1=0.5) {
  #mixture of normals, density up to constant factor
  c1<-exp(-t(x-mu1)%*%S1i%*%(x-mu1))
  c2<-exp(-t(x-mu2)%*%S2i%*%(x-mu2))
  return(p1*c1+(1-p1)*c2)
}
a=3; n=2000
mu1=c(1,1); mu2=c(5,5); S=diag(2); S1i=S2i=solve(S);
X=matrix(NA,2,n); X[,1]=x=mu1
for (t in 1:(n-1)) {
   y<-x+(2*runif(2)-1)*a
   MHR<-f(y,mu1,mu2,S1i,S2i)/f(x,mu1,mu2,S1i,S2i)
   if (runif(1)<MHR) x<-y
   X[,t+1]<-x
}
```

(see the associated R-file for plotting commands)