Part A Simulation and Statistical Programming HT14

Lecturer: Geoff Nicholls

University of Oxford

Lecture 13: MCMC for Bayesian Inference; The Ising Model

Notes and Problem sheets are available at

www.stats.ox.ac.uk\ ~nicholls\PartASSP

Bayesian Inference

If $\lambda \sim \pi(\lambda)$ is a parameter and $x \sim p(x|\lambda)$ is data then it is natural to consider the posterior distribution of λ,

$$\pi(\lambda|x) = \frac{p(x|\lambda)\pi(\lambda)}{m(x)}.$$

Here $m(x) = \int p(x|\lambda)\pi(\lambda)d\lambda$ is a normalizing constant.

We learn about the unknown true value of the parameter, Λ say, by combining our prior knowledge ($\pi(\lambda)$) with what we learn from the data (in the likelihood $L(\lambda; x) = p(x|\lambda)$).

If $\lambda \sim \pi(\lambda)$ is a repeatable process then there is nothing much new here. However, in Bayesian inference, the prior $\pi(\lambda)$ often describes a state of knowledge: λ-values where $\pi(\lambda)$ is larger are a priori more likely to be the true values.

We can answer questions about Λ using $\pi(\lambda|x)$. If A is a set of λ-values then the probability that λ is in A (given the prior and observation models) is

$$Pr(\Lambda \in A|x) = \int_A \pi(\lambda|x) d\lambda.$$

The expected value of λ (given ...) is

$$E(\Lambda|x) = \int \lambda \pi(\lambda|x) d\lambda.$$

As we can see, many objects of importance for Bayesian inference are expectations.

Example Suppose Λ is the mean of a Poisson distribution, and we want to know if $\Lambda > 1$. Suppose the prior for λ is $\lambda \sim \text{Gamma}(a, b)$ with $a, b > 0$ given, and we observe $x \sim \text{Poisson}(\lambda)$.

The prior for λ is

$$\pi(\lambda) \propto \lambda^{a-1} e^{-b\lambda}.$$

The likelihood is

$$L(\lambda; x) \propto \lambda^x e^{-\lambda},$$

and the posterior is

$$\begin{aligned} \pi(\lambda|x) &\propto L(\lambda; x)\pi(\lambda) \\ &= \lambda^{a+x-1} e^{-(b+1)\lambda}, \end{aligned}$$

so the posterior distribution of λ is $\lambda \sim \text{Gamma}(a+x, b+1)$.

We want to estimate $\Pr(\Lambda > 1|x)$. We need to compute

$$\Pr(\Lambda > 1|x) = \frac{\int_1^\infty \tilde{\pi}(\lambda|x)}{\int_0^\infty \tilde{\pi}(\lambda|x)}$$

where $\lambda^{a+x-1}e^{-(b+1)\lambda}$,

This is quite tractable in this example, but in general these integrals will be a dead end for pencil and paper work.

MCMC and Bayesian inference

Bayesian inference is widely applied to complex multivariate priors and likelihoods. The posterior expectations $E(\phi(\Lambda)|X)$ are hopelessly complex to evaluate.

We run MCMC targeting the posterior $\pi(\lambda|x)$ and simulate

$$\Lambda_1 = \lambda_1, ..., \Lambda_n = \lambda_n$$

ergodic for $\pi(\lambda|x)$. Our estimate for $E(\phi(\Lambda)|X)$

$$n^{-1} \sum_{t=1}^{n} \phi(\Lambda_t) \longrightarrow E(\phi(\Lambda)|X)$$

converges as $n \to \infty$ by the ergodic theorem for Markov Chains.

Example...(cont)

In our example above

$$\pi(\lambda|x) \propto \lambda^{a+x-1} e^{-(b+1)\lambda}.$$

Suppose $a = 3$, $b = 4.2$, we observe $x = 0$ and we want to estimate $\Pr(\Lambda > 1|x)$. Give an MCMC algorithm targeting $\pi(\lambda|x)$ and use the simulation to form the estimate.

[Step 1] Choose a proposal density $q(\lambda'|\lambda)$ for the candidate state λ'. I will use

$$\lambda' \sim U(\lambda - d, \lambda + d)$$

I will start with $d = 1$ but may need to adjust d to get an efficient algorithm, as we have seen. Since $q(\lambda'|\lambda) > 0 = q(\lambda|\lambda')$ we clearly have $q(\lambda'|\lambda) > 0 \Leftrightarrow q(\lambda|\lambda') > 0$.

[Step 2] Write down the Metropolis Hastings MCMC algorithm. Let $\Lambda_t = \lambda$. Λ_{t+1} is determined in the following way.

[1] Simulate $\lambda' \sim U(\lambda - d, \lambda + d)$ and $u \sim U(0, 1)$.
[2] If $u < \alpha(\lambda'|\lambda)$ set $\Lambda_{t+1} = \lambda'$ else set $\Lambda_{t+1} = \lambda$.

[Step 3] Calculate $\alpha(\lambda'|\lambda)$. If $\lambda' < 0$, $\alpha(\lambda'|\lambda) = 0$, so we reject if we leave $[0, \infty)$. Otherwise, if $\lambda' > 0$,

$$\begin{aligned} \alpha(\lambda'|\lambda) &= \min\left\{1,\ \frac{\pi(\lambda'|x)q(\lambda|\lambda')}{\pi(\lambda|x)q(\lambda'|\lambda)}\right\} \\ &= \min\left\{1,\ (\lambda'/\lambda)^{a+x-1}e^{-(b+1)(\lambda'-\lambda)}\right\}. \end{aligned}$$

There is a [step 4]: check irreduciblity (in computer measure, at least). If this is not obvious (as here, where the random-walk proposal can reach any part of $[0, \infty)$, and α is never zero in the space) then check carefully.

```
bayes.example<-function(n,a,b,x,lm0=1,d=1) {
  Lambda<-numeric(n)
  lm<-lm0
  for (k in 1:n) {
    lm.p<-runif(1,lm-d,lm+d)
    MHR<-(lm.p/lm)^(x+a-1)*exp(-(b+1)*(lm.p-lm))
    if (runif(1)<MHR*(lm.p>0)) lm<-lm.p
    Lambda[k]<-lm
  }
  return(Lambda)
}
> Lm<-bayes.example(n=1000,a=3,b=4.2,x=0)
> #convert boolean to numeric
> v<-as.numeric(Lm>1);
> mean(v)
[1] 0.077
#see file for std error & checking
```

The Ising Model Let

$$X = (X_{i_1,i_2})_{i_1=1:n}^{i_2=1:n}, \quad X_{i_1,i_2} \in \{0,1\}$$

be a collection of binary rv. The set C of cell indices is

$$C = \{(i_1, i_2) : i_1, i_2 \in \{1, 2, ..., n\}\}.$$

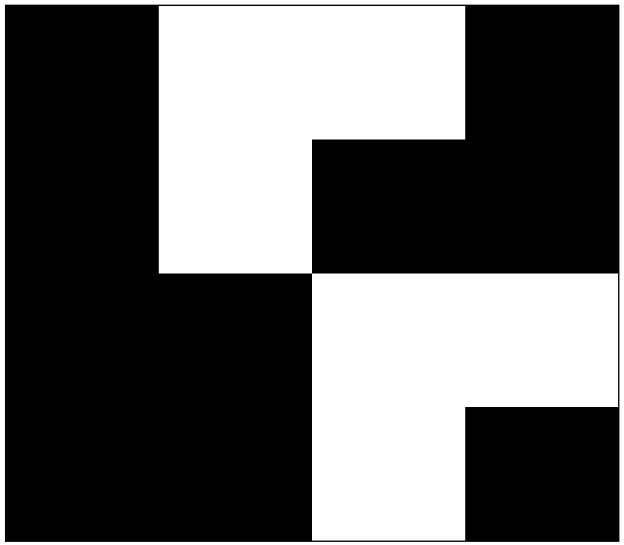
These variables live on an $n \times n$ square lattice so we can represent X by a black and white image. If $i, j \in C$ are two cells on the lattice, $i \sim j$ indicates the relation "cell i is a neighbor of cell j on the lattice".

Two neighboring cells $i \sim j$ *agree* if $X_{i_1,i_2} = X_{j_1,j_2}$ and otherwise they *disagree*.

Let $X = x$ be a particular realization of X. Let

$$\#x = \sum_{i \in C} \sum_{j \sim i} \mathbb{I}(X_{i_1,i_2} \neq X_{j_1,j_2}).$$

$\#x$ is a function of x giving the number of disagreeing neighbours in the image x. For example, in this realization of X, $\#x = 12$.



Denote by $\Omega = \{0,1\}^{n^2}$ the set of all binary images X. The *Ising model* is the following distribution over Ω:

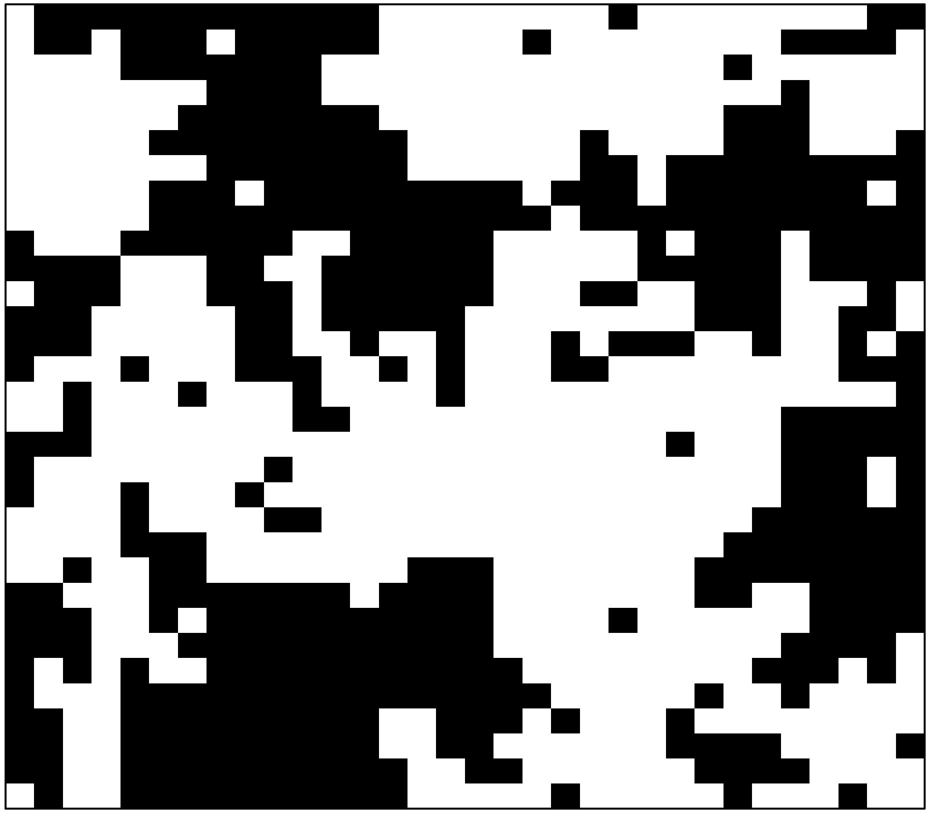
$$\pi(x) = \exp(-\theta \# x)/Z.$$

Here θ is a *smoothing parameter* which is usually taken to be greater than zero. Z is a normalizing constant, given by

$$Z = \sum_{x \in \Omega} \exp(-\theta \# x).$$

Expectations in $X \sim \pi(x)$ are typically hopelessly intractable and Z can not be evaluated for large n on the lattice we have here (it can be evaluated for certain special "boundary conditions").

Here is a sample $x \sim \pi(x)$, from the Ising model distribution $\pi(x) = \exp(-\theta \# x)/Z$ with $n = 32$ and $\theta = 0.8$.



MCMC for the Ising Model

Here is an MCMC algorithm simulating the Ising Model. We will simulate a sequence $X^{(1)}, X^{(2)}, ...$ of "Ising" images, targeting $\pi(x) \propto \exp(-\theta \# x)$. Suppose $X^{(t)} = x$.

[Step 1] Choose an update. Here is something simple. Choose a cell $i = (i_1, i_2)$ at random from C. Set $x'_{i_1,i_2} = 1 - x_{i_1,i_2}$ and $x'_{j_i,j_2} = x_{j_i,j_2}$ for $j \neq i$. Notice that $q(x'|x) = q(x|x') = 1/n^2$ for x', x differing at exactly one cell.

[Step 2] Write down the algorithm. Let $X^{(t)} = x$. $X^{(t+1)}$ is determined in the following way.

[1] Simulate $x' \sim q(x'|x)$ as above, and $u \sim U(0,1)$.

[2] If $u < \alpha(x'|x)$ set $X^{(t+1)} = x'$ and otherwise set $X^{(t+1)} = x$.

[Step 3] Calculate α. The q's cancel as usual, so

$$\begin{aligned}\alpha(x'|x) &= \min\left\{1,\ \frac{\pi(x')q(x|x')}{\pi(x)q(x'|x)}\right\} \\ &= \min\left\{1,\ \exp(-\theta(\#x' - \#x))\right\}\end{aligned}$$

It is clear the algorithm is irreducible (q is irreducible and α is never zero) and aperiodic (rejection is possible), so it is ergodic for $\pi(x)$.

```
n<-32; theta<-0.8
X<-matrix(runif(n^2)>1/2,n,n) #random start state
hashX<-sum(abs(diff(X))+abs(diff(t(X))))
N<-20000
for (j in 1:N) {
  i<-1+floor(runif(1)*n^2)
  Xp<-X
  Xp[i]<-1-X[i]
  hashXp<-sum(abs(diff(Xp))+abs(diff(t(Xp))))
  if (runif(1)<exp(theta*(hashX-hashXp))) {
    X<-Xp
    hashX<-hashXp
  }
  image(X,col=gray(0:255/255),axes=F); box()
}
```