A Queueing Theorist Looks at MCMC

Myron Hlynka

Department of Mathematics and Statistics
University of Windsor
Windsor, ON, Canada.

June 16, 2015
Outline

- Usual Markov Chain Monte Carlo
- Fibonacci Distribution
- Birth and Death process
- Markov chain matrix
- Output, Graphs and Comments
Usual MCMC

- Rosenbluth-Hastings (aka Metropolis-Hastings)
  (see Press, B. Opinionated Lessons in Statistics. #39 MCMC and Gibbs Sampling. Video lecture. (history at 15:45 in video)
Usual MCMC


- Start with a distribution $\tilde{\pi}$ that we wish to simulate for which we might know only ratios $\pi_i/\pi_j$. 

Let $Q = [q_{ij}]$ be irreducible matrix of a MC. Let $\alpha_{ij} = \min(\pi_jq_{ji}/\pi_iq_{ij}, 1)$. Then $P = [p_{ij}]$ is transition matrix with stationary vector $\tilde{\pi}$. 

We can use $P$ to simulate values of $\tilde{\pi}$. 

Myron Hlynka (University of Windsor)  
A Queueing Theorist Looks at MCMC  
June 16, 2015 3 / 17
Usual MCMC

- Start with a distribution $\vec{\pi}$ that we wish to simulate for which we might know only ratios $\pi_i/\pi_j$.
- Let $Q = [q_{ij}]$ be irreducible matrix of a MC. Let $\alpha_{ij} = \text{min}(\frac{\pi_j q_{ji}}{\pi_i q_{ij}}, 1)$. Let $p_{ij} = \alpha_{ij} q_{ij}$
Usual MCMC

- Start with a distribution \( \vec{\pi} \) that we wish to simulate for which we might know only ratios \( \pi_i/\pi_j \).
- Let \( Q = [q_{ij}] \) be irreducible matrix of a MC. Let \( \alpha_{ij} = \min(\frac{\pi_j q_{ji}}{\pi_i q_{ij}}, 1) \).
  - Let \( p_{ij} = \alpha_{ij} q_{ij} \)
- Then \( P = [p_{ij}] \) is transition matrix with stationary vector \( \vec{\pi} \).
Usual MCMC

- Start with a distribution \( \vec{\pi} \) that we wish to simulate for which we might know only ratios \( \pi_i / \pi_j \).
- Let \( Q = [q_{ij}] \) be irreducible matrix of a MC. Let \( \alpha_{ij} = min(\pi_j q_{ji} / \pi_i q_{ij}, 1). \)
  Let \( p_{ij} = \alpha_{ij} q_{ij} \)
- Then \( P = [p_{ij}] \) is transition matrix with stationary vector \( \vec{\pi} \).
- We can use \( P \) to simulate values of \( \vec{\pi} \)
Difficulties

- What should we choose for \( Q \)?

I would not have thought of R-H.

Myron Hlynka (University of Windsor)

A Queueing Theorist Looks at MCMC

June 16, 2015 4 / 17
Difficulties

- What should we choose for $Q$?
- What is the intuition behind R-H?
Difficulties

- What should we choose for $Q$?
- What is the intuition behind R-H?
- I would not have thought of R-H.
The Fibonacci distribution has values \( f(3) = f(2) = 1/4 \).
In general \( f(n) = F_{n-1}/2^n \) for \( n = 2, 3, \ldots \),
where \( F_0 = F_1 = 1 \), \( F_n = F_{n-1} + F_{n-2} \)
so \( \{F_i\} = 1, 1, 2, 3, 5, 8, 13, \ldots \)
It turns out that \( f(n) \) is the probability that
exactly \( n \) steps are required for a random walk to
reach state 2 from state 0 if the system moves
to the right with probability .5 on each step and moves
left (or remains the same at 0) with probability .5
Suppose we wish to simulate observations from the Fibonacci distribution using MCMC. First we compute the ratio of consecutive Fibonacci probabilities.

\[
\frac{f(x+1)}{f(x)} = \frac{F_x}{2^{x+1}} \div \frac{F_{x-1}}{2^x} = \frac{F_x}{2F_{x-1}}.
\] (1)
\[ 2F_{x-1} f(x + 1) = F_x f(x). \]

We take this equation to be the balance equation for a continuous time Markov process. The LHS is the rate from state \( x + 1 \) to state \( x \) and is the product of the limiting probability \( f(x + 1) \) of being in state \( x + 1 \) times the rate \( 2F_{x-1} \) of moving to state \( x \) given that the system is in state \( x + 1 \). The RHS is the rate from state \( x \) to state \( x + 1 \) which is the product of the limiting probability \( f(x) \) of being in state \( x \) times the rate \( F_x \) of moving to state \( x + 1 \) given that the system is in state \( x \).
Our birth and death transition diagram looks like

\[
\begin{array}{cccc}
F_2 & F_3 & F_4 & F_5 \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow \\
2 & 3 & 4 & 5 & 6 \ldots \\
\leftarrow & \leftarrow & \leftarrow & \leftarrow \\
2F_1 & 2F_2 & 2F_3 & 2F_4
\end{array}
\]
Fibonacci and MCMC IV

\[
\begin{align*}
\frac{F_2}{F_2 + 2F_1} & \quad \rightarrow \\
\frac{F_3}{F_3 + 2F_2} & \quad \rightarrow \\
\frac{F_4}{F_4 + 2F_3} & \quad \rightarrow \\
\frac{F_5}{F_5 + 2F_4} & \quad \rightarrow \\
2F_1 & \quad \leftarrow \\
2F_2 & \quad \leftarrow \\
2F_3 & \quad \leftarrow \\
2F_4 & \quad \leftarrow \\
2F_5 & \quad \leftarrow
\end{align*}
\]
The corresponding infinitesimal generator matrix for states 2, 3, … (with the pairs of rates appearing in the off-diagonal positions) is

\[
Q = \begin{bmatrix}
  a & \frac{F_2}{F_2 + 2F_1} & 0 & 0 & 0 & \cdots \\
  \frac{2F_1}{F_2 + 2F_1} & b & \frac{F_3}{F_3 + 2F_2} & 0 & 0 & \cdots \\
  0 & \frac{2F_2}{F_3 + 2F_2} & c & \frac{F_4}{F_4 + 2F_3} & 0 & \cdots \\
  0 & 0 & \frac{2F_3}{F_4 + 2F_3} & d & \frac{F_5}{F_5 + 2F_4} & \cdots \\
  \vdots & \vdots & \vdots & \vdots & \vdots & \ddots 
\end{bmatrix},
\]

where \(a, b, c, d, \ldots\) are negative values chosen so that each row sums to 0.
\[ \vec{0} = \vec{\pi} Q. \text{ Let } Q^* = Q/2. \text{ Then } \vec{0} = \vec{\pi} Q^*. \text{ All entries of } Q^*, \text{ excluding the diagonal entries, are less than } 0.5 \text{ in absolute value.} \]

Next we add \( \vec{\pi} \) to both sides to get \( P \) satisfying \( \vec{\pi} = \vec{\pi}(I + Q^*) = \vec{\pi}P \). Here \( P = I + Q^* \) so

\[
P = \begin{bmatrix}
1 + 0.5a & \frac{0.5F_2}{F_2 + 2F_1} & 0 & 0 & 0 & \ldots \\
\frac{F_1}{F_2 + 2F_1} & 1 + 0.5b & \frac{0.5F_3}{F_3 + 2F_2} & 0 & 0 & \ldots \\
0 & \frac{F_2}{F_3 + 2F_2} & 1 + 0.5c & \frac{0.5F_4}{F_4 + 2F_3} & 0 & \ldots \\
0 & 0 & \frac{F_3}{F_4 + 2F_3} & 1 + 0.5d & \frac{0.5F_5}{F_5 + 2F_4} & \ldots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots
\end{bmatrix}
\]
\[ P \approx \begin{bmatrix}
0.833 & 0.167 & 0 & 0 & 0 & \ldots \\
0.333 & 0.417 & 0.25 & 0 & 0 & \ldots \\
0 & 0.25 & 0.536 & 0.214 & 0 & \ldots \\
0 & 0 & 0.286 & 0.48 & 0.227 & \ldots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots
\end{bmatrix} \]
x=rep(2,1000000);
#This is the start vector with all values 2
u=runif(1000001);
#This generates 1000001 random uniform (0,1) values.
F=rep(1,150)
for (i in 4:150) {F[i]=F[i-1]+F[i-2]}  
#generate first 150 Fibonacci numbers
for (i in 1:1000000)  
{  
a=(x[i]>2)*(u[i+1]<F[x[i]-1]/(F[x[i]]+2*F[x[i]-1]))
  b=+1*(u[i+1]>(1-(.5*F[x[i]+1]/(F[x[i]+1]+2*F[x[i]]))))
  x[i+1]=x[i]-a+b
}
R output

\begin{verbatim}
x[1:100]
  [1] 2 2 2 2 2 2 3 4 3 3 3 2 2 2 2 2 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 3 3 4 4 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 4 4 4 5 5 4 5 5 6 5 6 6 5 5 4 4 5 5 5 4 4 5 5 5 5 4 4 5 5 5
[82] 4 4 4 3 2 2 2 3 4 4 4 4 4 3 3 2 2 2 2 2 2
\end{verbatim}
Figure: MCMC estimated probabilities
The End.
Thank you.