SECOND PUBLIC EXAMINATION

Honour School of Mathematics and Statistics Part A: Paper A12

Simulation and Statistical Programming

SPECIMEN EXAM #1 FOR NEW COURSE

SAMPLE SOLUTIONS

TRINITY TERM 2014
DAY, DATE, TIME

This paper contains three questions of which at least two should be attempted. (The best two answers will be counted).

Do not turn this page until you are told that you may do so

1. Let X and Y be discrete random variables with probability mass functions $p(x) = \tilde{p}(x)/Z_p$ and $q(y) = \tilde{q}(y)/Z_q$ given in terms of unnormalised functions $\tilde{p}(x)$ and $\tilde{q}(y)$.

   (a) [6 marks] Write down the rejection algorithm simulating X. State any additional conditions you impose on $q(y)$. You may assume simulation of $Y \sim q$ is available. However, your algorithm should not require evaluation of Z_p and Z_q.

   ANS: Suppose we can find M so that $M \geqslant \tilde{p}(x)/\tilde{q}(x)$ for all x.

   Rejection Algorithm simulating $X \sim p$:

   [1] Simulate $y \sim q(y)$ and $u \sim U(0,1)$.

   [2] If $u < \tilde{p}(y)/M\tilde{q}(y)$ return $X = y$, otherwise goto [1].

   (b) [7 marks] Prove that the rejection algorithm returns $X \sim p(x)$.

   ANS: Let $a(x) = \tilde{p}(x)/\tilde{q}(x)M$. The probability to reject at [2] is $r = 1 - \sum_y q(y)r(y) = 1 - Z_p/MZ_q$. We must propose x and it accept it. We could propose and reject any number of times before that.

$$\begin{aligned}
\Pr(X = x) &= q(x)a(x) + rq(x)a(x) + r^2 q(x)a(x) + ... \\
&= q(x)a(x)/(1 - r) \\
&= \tilde{p}(x)/MZ_q \;\times\; MZ_q/Z_p \\
&= p(x).
\end{aligned}$$

   as supposed.

   (c) [12 marks] Let θ be a discrete parameter with prior probability mass function $\pi(\theta)$ and let $z \sim f(z|\theta)$ be a discrete observation with probability mass function $f(z|\theta)$. The likelihood for θ is $L(\theta; z) = f(z|\theta)$. The posterior for θ is

$$\pi(\theta|z) = \frac{f(z|\theta)\pi(\theta)}{m(z)}$$

   where $m(z) = \sum_\theta f(z|\theta)\pi(\theta)$.

   (i) [7 marks] Prove that the following algorithm simulates $\Theta \sim \pi(\theta|z)$.

   [1] Simulate $y \sim \pi(y)$ and $u \sim U[0, 1]$.

   [2] If $u \leqslant L(y; z)$ set $\Theta = y$ and stop, otherwise, return to step [1] and repeat.

   ANS: this is a rejection algorithm with $p(\theta) = \pi(\theta|z)$, $q(\theta) = \pi(\theta)$ and $M = 1/m(y)$. The acceptance probability $p/Mq = \pi(\theta|z)m(y)/\pi(\theta) = L(\theta; z)$ and since θ is discrete, $L(\theta; z) \leqslant 1$ as it is a pmf.

   (ii) [5 marks] Let N be the random number of simulations needed at step [1] for each acceptance at step [2]. Show that $E(N) = 1/m(z)$.

   ANS: the number of trials is geometric with success probability $1 - r$ since each trial is independent. Now $1 - r = \sum_y q(y) \times p(y)/Mq(y) = m(z)$ so $E(N) = 1/m$.

2. (a) [9 marks] Let $X = (X_1, X_2)$ be a pair of random variables $X \in \Re^2$ with joint density

$$p(x) \propto \exp(-|x_1| - |x_2| - |x_1 - x_2|), \qquad \text{where } x = (x_1, x_2).$$

   Give an MCMC algorithm targeting $p(x)$.

ANS: 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:

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

with $a > 0$ a fixed constant. That is easy to sample, and certainly $q(x'|x) > 0 \Leftrightarrow q(x|x') > 0$ since $q(x'|x) = q(x|x') = 1/4a^2$.

Step 2. The algorithm is, given $X^{(n)} = x$,

[1] for $i = 1, 2$ simulate $x_i' \sim U(x_i - a, x_i + a)$

[2] with probability

$$\alpha(x'|x) = \min\left\{1,\ \frac{\exp(-|x_1'| - |x_2'| - |x_1' - x_2'|)}{\exp(-|x_1| - |x_2| - |x_1 - x_2|)}\right\}$$

set $X^{(n+1)} = x'$ otherwise set $X^{(n+1)} = x$.

(b) [8 marks] Write an R function implementing the code you wrote down in part (a). Your function should take as input the number of MCMC steps, n say, and return as output a $2 \times n$ matrix X of simulated values.

```
f<-function(x) {
  return(exp(-sum(abs(x)))-abs(diff(x)))
}

a=1; n=100000
X=matrix(NA,2,n); X[,1]=x=c(0,0)
for (t in 1:(n-1)) {
   xp<-x+(2*runif(2)-1)*a
   MHR<-f(xp)/f(x)
   if (runif(1)<MHR) x<-xp
   X[,t+1]<-x
}
```

(c) [8 marks] Let

$$A = \{\ x \in \Re^2 :\ 2 < x_2 < 2.01\ \}.$$

Write down an MCMC algorithm targeting $p(x|X \in A)$ giving the acceptance probability as an explicit function of x_1, x_2. In choosing the proposal distribution you should consider the efficiency of your algorithm.

ANS: The algorithm is similar to the last but we should adjust the proposal, take $X_0 \in A$ and reject if we leave A.

Step 1. Take $a = (1, 0.005)$. For $i, 1, 2$,

$$x_i' \sim U(x_i - a_i, x_i + a_i).$$

so we make small jumps in the constrained variable. It remains the case that $q(x'|x) > 0 \Leftrightarrow q(x|x') > 0$ since each variable by itself still satisfies the same generic proposal as before.

Step 2. The algorithm is, given $X^{(n)} = x$,

[1] for $i = 1, 2$ simulate $x_i' \sim U(x_i - a_i, x_i + a_i)$

[2] with probability

$$\alpha(x'|x) = \min\left\{1,\ \mathbb{I}_{2 < x_2' < 2.01} \frac{\exp(-|x_1'| - |x_2'| - |x_1' - x_2'|)}{\exp(-|x_1| - |x_2| - |x_1 - x_2|)}\right\}$$

set $X^{(n+1)} = x'$ otherwise set $X^{(n+1)} = x$.

3. (a) [15 marks] Suppose A is a $n \times n$ symmetric positive definite matrix. The Cholesky factorization of A is $A = LL^T$ with L a lower triangular matrix with positive diagonal elements.

(i) [6 marks] Give an algorithm computing L. ANS: Chop A and L up into blocks

$$A = \left(\begin{array}{c|c} a_{11} & A_{21}^T \\ \hline A_{21} & A_{22} \end{array}\right) = \left(\begin{array}{c|c} 1 \times 1 & 1 \times (n-1) \\ \hline (n-1) \times 1 & (n-1) \times (n-1) \end{array}\right)$$

and similarly

$$L = \left(\begin{array}{c|c} L_{11} & 0_{1\times(n-1)} \\ \hline L_{21} & L_{22} \end{array}\right).$$

Since L is lower triangular it is zero above the diagonal, and in particular all the entries in the top row except the first are zero. Since $A = LL^T$,

$$\begin{pmatrix} a_{11} & A_{21}^T \\ A_{21} & A_{22} \end{pmatrix} = \begin{pmatrix} L_{11} & 0_{1\times(n-1)} \\ L_{21} & L_{22} \end{pmatrix} \begin{pmatrix} L_{11} & {L_{21}}^T \\ 0_{(n-1)\times 1} & L_{22}^T \end{pmatrix}$$

$$= \left(\begin{array}{c|c} L_{11}^2 & L_{11}{L_{21}}^T \\ \hline L_{11}L_{21} & L_{22}{L_{22}}^T + L_{21}{L_{21}}^T \end{array}\right)$$

so $L_{11} = \sqrt{a_{11}}$, $L_{21} = A_{21}/\sqrt{a_{11}}$ and the A_{22} block gives

$$\begin{aligned} A_{22} - L_{21}{L_{21}}^T &= L_{22}{L_{22}}^T \\ \tilde{A} &= \tilde{L}\tilde{L}^T \qquad \text{now } (n-1) \times (n-1) \end{aligned}$$

To solve for L_{22}, we need the Cholesky factorization of the $(n-1) \times (n-1)$ matrix $\tilde{A} = A_{22} - L_{21}{L_{21}}^T$, so we have reduced the problem by one dimension. Finally, if $n = 1$ so A is a scalar, $L = \sqrt{A}$ terminates the recursion.

(ii) [9 marks] Write an R function implementing the algorithm you wrote part (i). Your function should take as input A and return L as output.

```
#Cholesky
my.chol<-function(A) {
   n=dim(A)[1] #assume nxn
   if (n==1) return(sqrt(A))

   L=matrix(0,n,n)
   L[1,1]=sqrt(A[1,1])
   L[2:n,1]=A[2:n,1]/L[1,1]
   L[1,2:n]=rep(0,n-1)

   A22=A[2:n,2:n,drop=FALSE]
   newA=A22-L[2:n,1]%*%t(L[2:n,1])
```

```
    L[2:n,2:n]=my.chol(newA)

    return(L)
}
```

(b) [10 marks] Let $Z = (Z_1, Z_2, ..., Z_K)^T$ be a vector of K jointly independent $N(0, 1)$ random variables. Let $\mu \in \Re^n$ be an $n \times 1$ vector and let Σ be a non-singular symmetric positive definite $K \times K$ matrix.

(i) [5 marks] Show that if L is a $K \times K$ matrix satisfying $\Sigma = LL^T$ and $X = \mu + LZ$ then $X \sim N(\mu, \Sigma)$. ANS: since X is a linear combination of normals it is also normal. Its mean is $E(X) = \mu + LE(Z) = \mu$. It covariance matrix is $E((\mu - X)(\mu - X)^T) = E(LZZ^T L^T) = LE(ZZ^T)L^T = \Sigma$ since $E(ZZ^T)$ is the identity matrix as it is the covariance of Z.

(ii) [5 marks] Write an R function simulating $X \sim N(\mu, \Sigma)$. Your function should take as input μ and Σ and return as output a simulation of X.

```
my.mvnorm<-function(mu,Sigma) {
L<-my.chol(Sigma)
X<-mu+L%*%rnorm(length(mu))
return(X)
}
```