Exam Part A
Lecturer: R. Evans, F. Caron
Checked by: R. Evans, F. Caron

A12 Simulation and Statistical Programming
SOLUTIONS

Date of this version: 10/03/2016

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

1. (a) [5 marks] Classification: B

Simulate $U \sim U[0,1]$, with sampled value u. Set $X = k$ with k such that

$$\sum_{i=1}^{k-1} q_\lambda(i) < u \leqslant \sum_{i=1}^{k} q_\lambda(i)$$

where the lhs is equal to 0 for $k = 1$.
The algorithm is therefore as follows.
Simulate $U \sim U[0,1]$, with sampled value u. Set $k = 1$ and $c = q_\lambda(1)$.
While $u > c$

- Set $k \leftarrow k + 1$
- Set $c \leftarrow c + q_\lambda(k)$

Return $X = k$.

(b) [10 marks]

(i) [2 marks] Classification: S

$Y = g(Z)$ where $g(z) = e^{-z}$ is one-to-one and $g^{-1}(y) = -\log(y)$.

$$\begin{aligned} f_Y(y) &= f_Z(g^{-1}(y)) \left| \frac{\partial g^{-1}(y)}{\partial y} \right| \\ &= \lambda y^{\lambda-1} \end{aligned}$$

(ii) [2 marks] Classification: S

$$\begin{aligned} \mathbb{P}(\lceil Z \rceil = i) &= \int_{i-1}^{i} f_Z(z) dz \\ &= e^{-\lambda(i-1)} - e^{-\lambda i} = e^{-\lambda(i-1)}(1 - e^{-\lambda}) \end{aligned}$$

(iii) [2 marks] Classification: S

$$\begin{aligned} \mathbb{P}(X = i) &= \int_0^\infty \mathbb{P}(X = i | Y = y) f_Y(y) dy \\ &= \int_0^\infty \lambda y^{\lambda-1}(1-y)^{i-1} dy \\ &= \lambda \frac{\Gamma(i)\Gamma(\lambda+1)}{\Gamma(i+\lambda+1)} \int_0^\infty \frac{\Gamma(i+\lambda+1)}{\Gamma(i)\Gamma(\lambda+1)} y^{\lambda-1}(1-y)^{i-1} dy \quad = \lambda \frac{\Gamma(i)\Gamma(\lambda+1)}{\Gamma(i+\lambda+1)} \end{aligned}$$

(iv) [4 marks] Classification: S/N

1. Sample $U_1, U_2 \sim \mathrm{U}[0,1]$
2. Set $Z = -\log(U_1)/\lambda$ [$Z \sim \mathrm{Exp}(\lambda)$]
3. Set $Y = e^{-Z}$ [$Y \sim \mathrm{Beta}(\lambda, 1)$]
4. Set $Z_2 = \log(U_2)/\log(1-Y)$ [$Z_2 \sim \mathrm{Exp}(-\log(1-Y))$]
5. Set $X = \lceil Z_2 \rceil$

(c) [10 marks]

(i) [4 marks] Classification: S/N

We need the ratio

$$p_s(i)/q_\lambda(i) \propto \frac{\Gamma(i+\lambda+1)}{\Gamma(i) i^s}$$

to be upper bounded. As $\frac{\Gamma(i+\lambda+1)}{\Gamma(i)i^s} \sim i^{\lambda+1-s}$ as $i \to \infty$, we need $\lambda + 1 - s < 0$.

(ii) [6 marks] Classification: S

$$p_2(i) = \frac{6}{\pi^2}\frac{1}{i^2}$$
$$q_1(i) = \frac{1}{i(i+1)}$$
$$p_2(i)/q_1(i) = \frac{6}{\pi^2}\frac{(i+1)}{i} \leqslant \frac{12}{\pi^2} = M$$

The rejection sampler is as follows.

1. Sample $Y \sim q_1$ and $U \sim U[0,1]$.
2. If $U \leqslant \frac{Y+1}{2Y}$ then return $X = Y$. Otherwise go back to step 1.

Let N be the number of simulations of Y before acceptance. N follows a geometric distribution with parameter $1/M$ and $\mathbb{E}[N] = M$.

Note: If they don't give the exact expression of the normalizing constant $Z_{p_2} = \frac{\pi^2}{6}$ for p_2, they still have full marks.

2. (a) [12 marks]

   (i) [2 marks] Classification: B

   Let $\theta = \mathbb{E}_p[\phi(X)]$. The importance sampling estimator is defined as

$$\widehat{\theta}_n = \frac{1}{n}\sum_{i=1}^{n} \frac{p(Y_i)}{q_{m,r}(Y_i)}\phi(Y_i)$$

   where Y_i are iid from $q_{m,r}$.

   (ii) [5 marks] Classification: B/S

   For the importance sampling estimator to be unbiased and consistent, we need to have $q_{m,r}(i) > 0$ for all i such that $\phi(i)p(i) > 0$. So we need $m \leqslant a$.

$$\begin{aligned}
\mathbb{E}[\widehat{\theta}_n] &= \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_{q_{m,r}}\left[\frac{p(Y_i)}{q_{m,r}(Y_i)}\phi(Y_i)\right] \\
&= \mathbb{E}_{q_{m,r}}\left[\frac{p(Y_1)}{q_{m,r}(Y_1)}\phi(Y_1)\right] \\
&= \sum_{i=m}^{\infty} \frac{p(i)}{q_{m,r}(i)}\phi(i)q_{m,r}(i) \\
&= \sum_{i=m}^{\infty} \phi(i)p(i) = \mathbb{E}_p[\phi(X)]
\end{aligned}$$

   Let $Z_i = \frac{p(Y_i)}{q_{m,r}(Y_i)}\phi(Y_i)$. Z_i are iid with mean θ. From the law of large numbers, $\frac{1}{n}\sum_i Z_i \to \theta$ almost surely as $n \to \infty$, hence the estimator is strongly consistent.

   (iii) [5 marks] Classification: S

$$\mathbb{V}(\widehat{\theta}_n) = \frac{1}{n}\left(\mathbb{E}_p[p(X)/q_{m,r}(X)\phi(X)^2] - \theta^2\right)$$

   where

$$\theta = 2^{1-a}$$

   and

$$\mathbb{E}_p[p(X)/q_{m,r}(X)\phi(X)^2] = \sum_{i=a}^{\infty} p(i)^2/q_{m,r}(i)$$

$$\mathbb{E}_p[p(X)/q_{m,r}(X)\phi(X)^2] = r^{-1}(1-r)^m \sum_{i=a}^{\infty} (4(1-r))^{-i}$$

   It is finite for $4(1-r) > 1$ hence $r < 3/4$. In this case

$$\begin{aligned}
\mathbb{E}_p[p(X)/q_{m,r}(X)\phi(X)^2] &= r^{-1}(1-r)^m \frac{(4(1-r))^{-a}}{1-(4(1-r))^{-1}} \\
&= 4^{1-a}\frac{(1-r)^{m+1-a}}{r(3-4r)}
\end{aligned}$$

(b) [13 marks] (i) [3 marks] Classification: S

The acceptance probability is

$$\alpha(y|x) = \min\left(1, \frac{p(y)Q(x|y)}{p(x)Q(y|x)}\right) = \begin{cases} \min(1, 2^{x-y}) & \text{if } y > 0 \\ 0 & \text{otherwise} \end{cases}$$

Set $X_0 = x_0$. For $t = 1, \ldots, n-1$

- Sample $Z_t \sim U[0,1]$. If $Z_t < 1/2$, set $Y_t = X_{t-1} + 1$. Otherwise set $Y_t = X_{t-1} - 1$
- Sample $U_t \sim U[0,1]$. If $U_t < \alpha(Y_t|X_{t-1})$, set $X_t = Y_t$. Otherwise, set $X_t = X_{t-1}$.

(ii) [4 marks] Classification:

```
mcmc <- function(x0,n) {
xsamples <- numeric(n)
x <- x0
for(t in 1:n)
{
  z <- sample(c(1,-1), 1)
  y <- x + z

  u = runif(1)
  if (y > 0){
    if (u < 2^{x-y})
    {
      x <- y
    }
  }
  xsamples[t] <- x
  }
return(xsamples)
}
```

(iii) [3 marks] Classification: S

$$P_{i,j} = Q(j|i)\alpha(j|i) + \left(1 - \sum_{k=1}^{\infty} Q(k|i)\alpha(k|i)\right)\mathbb{I}(i = j)$$

So, for $i \geqslant 2$

$$P_{i,j} = \begin{cases} 1/2 & \text{if } j = i-1 \\ 1/4 & \text{if } j = i+1 \\ 1/4 & j = i \\ 0 & \text{otherwise} \end{cases}$$

and

$$P_{1,j} = \begin{cases} 1/4 & \text{if } j = 2 \\ 3/4 & \text{if } j = 1 \\ 0 & \text{otherwise} \end{cases}$$

For reversibility, we need to check that P and p verify detailed balance, i.e. $p(i)P_{i,j} = p(j)P_{j,i}$ for all i, j. For $i = j$ this is trivially true. For $i \geqslant 1$,

$$p(i)P_{i,i+1} = 1/2^{i+2} = p(i+1)P_{i+1,i}$$

(iv) [3 marks] Classification: N

The MH transition matrix P verifies detailed balance and thus admits p as invariant distribution. Therefore $pP^t = p$ for any $t = 1, 2, \ldots$. As $X_0 \overset{d}{=} X$, it follows that $X_t \overset{d}{=} X$ and $\mathbb{E}[X_t] = \mathbb{E}_p[X]$. Therefore

$$\mathbb{E}\left[\widehat{\theta}_n^{\text{MH}}\right] = \frac{1}{n}\sum_{t=1}^{n-1} \mathbb{E}[X_t] = \mathbb{E}_p[X]$$

3. (a) [5 marks] Consider the linear system

$$X\beta = y, \tag{1}$$

where X is an $n \times p$ matrix of full column rank, y is a known vector of length n, and β is a vector of unknown parameters.

(i) What does is mean to say that the linear system (1) is *overdetermined*? Bookwork.

[1 mark] A linear system is overdetermined if it comprises more equations than variables, so $n > p$.

(ii) Consider the related least squares problem of minimising

$$R(\beta) = (y - X\beta)^T(y - X\beta). \tag{2}$$

Show that $\widehat{\beta}$, the solution to the least squares problem, satisfies the equation

$$X^T X\widehat{\beta} = X^T y.$$

[4 marks] Differentiating R w.r.t. β_j gives On prob. sheet.

$$\frac{\partial R}{\partial \beta_j} = -2\sum_{i=1}^{n} x_{ij}(y_i - x_{i1}\beta_1 - \cdots - x_{ip}\beta_p)$$
$$\frac{\partial R}{\partial \beta_j} = -2x^{(j)}y - x^{(j)}X\beta,$$

where $x^{(j)}$ is a row vector formed from the jth row of X^T. Writing the equations for every j out gives the matrix equation.

(b) [13 marks] Let A be a non-singular $n \times n$ upper triangular matrix, and b a vector of length n. Consider the linear system of equations

$$Ax = b. \tag{3}$$

(i) By writing

$$A = \begin{pmatrix} \widetilde{A} & c_n \\ 0 & a_{nn} \end{pmatrix},$$

where $\widetilde{A}$ is a matrix of dimensions $(n-1) \times (n-1)$ and c_n a column vector of length $n-1$, give a recursive algorithm for solving (3).

[You do not have to write your answer as computer code.] Forward case in lec. backward case in prac.

[6 marks] The last equation $a_{nn}x_n = b_n$ gives $x_n = b_n/a_{nn}$; we must have $a_{nn} \neq 0$ since otherwise the matrix is singular. Letting $\widetilde{x} = (x_1, \ldots, x_{n-1})$, the rest becomes

$$\widetilde{A}\widetilde{x} + c_n x_n = b_{1:n-1}$$
$$\widetilde{A}\widetilde{x} = b_{1:n-1} - c_n x_n \equiv \widetilde{b},$$

so we have an $(n-1) \times (n-1)$ matrix $\widetilde{A}$. This gives the following algorithm: Function: Solve(A, b)

Input: Non-singular upper triangular matrix A, $n \times n$, vector b of length n.

Output: x, a vector satisfying $Ax = b$.

1. Set $x_n = b_n/a_{nn}$;
2. If $n = 1$, then this is the solution, so return x_n and stop;
3. Otherwise, let $\widetilde{A} = A_{1:(n-1),1:(n-1)}$, and $\widetilde{b} = b_{1:n-1} - A_{1:(n-1),n}x_n$;

4. Find $\widetilde{x}$ as Solve($\widetilde{A}, \widetilde{b}$);
5. Return $(\widetilde{x}, x_n)$.

(ii) Write an R function to implement this algorithm. Your function should take arguments A and b, and return a vector x. Several similar examples in practicals.

[4 marks] For example:

```
solveMat <- function(A, b) {
  n <- nrow(A)

  x = numeric(n)
  x[n] = b[n]/A[n,n]

  if (n == 1) return(x)

  b2 <- b[-n] - A[-n,n]*x[n]
  A2 <- A[-n,-n,drop=FALSE]
  x[-n] <- solveMat(A2, b2)

  x
}
```

Total penalty for all trivial mistakes limited to 2 marks.

(iii) What is the computational complexity of your algorithm? Justify your answer. Standard calculation.

[3 marks] Step 1 involves 1 computation, and Step 3 involves $2(n-1)$. Then if the complexity for an $n \times n$ matrix is $g(n)$ we have $g(n) = 2n - 1 + g(n-1)$, so $g(n) = \sum_{i=1}^{n}(2i-1) = O(n^2)$ is quadratic in n. [If candidates worry about the complexity of constructing $\widetilde{A}$ they may conclude that the complexity is cubic, in which case (if justified) this is an acceptable answer.]

(c) [7 marks] (i) Define the QR decomposition of the $n \times p$ matrix X from (a). Bookwork.

[2 marks] The QR decomposition of an $n \times p$ matrix X is a pair Q, R of matrices such that $X = QR$, where Q is $n \times p$ with orthogonal columns, and R is $p \times p$ and upper triangular with non-negative diagonal entries (positive since X has full column rank).

(ii) Let (Q, R) be the QR decomposition of the matrix $X = QR$. Explain how to use this to find the solution of the least squares problem (2). *[2 marks] We have* In notes.

$$X^T X\beta = X^T y$$
$$R^T Q^T QR\beta = R^T Q^T y$$
$$R^T R\beta = R^T Q^T y$$

so $R\beta = Q^T y$. We can then use the algorithm from (b) to solve this linear system.

(iii) Deduce a necessary and sufficient condition, in terms of Q and y only, for the existence of a unique solution to (1). Harder and unseen.

[3 marks] Any solution to (1) must also be the solution to (2). Given $\beta = R^{-1}Qy$ as the solution to (2) we can plug this back in to (1), and find $QRR^{-1}Q^T y = y$ which implies that $QQ^T y = y$. This condition is necessary and sufficient for the solution to (2) to satisfy (1). [Of course, the candidate may not make the connection to (ii) and just plug $X = QR$ directly into (1), which is also fine.]