STAT580 HOMEWORK5

Xingche Guo

3/26/2017

Problem1

(a)

```
set.seed(580580)
Y <- rpois(100,2)
R <- sample(c(0,1), size = 100, replace = TRUE, prob = c(0.7,0.3))
X <- Y*R
```

(b)

(i)

$$\because \ f(\lambda|p,\boldsymbol{r},\boldsymbol{x}) = \frac{f(\lambda;p,\boldsymbol{r},\boldsymbol{x})}{f(p,\boldsymbol{r},\boldsymbol{x})} \propto f(\lambda;p,\boldsymbol{r},\boldsymbol{x})$$

$$\therefore \ f(\lambda|p,\boldsymbol{r},\boldsymbol{x}) \propto \lambda^{(a-1+\sum_{i=1}^{n} x_i)} e^{-(b+\sum_{i=1}^{n} r_i)\lambda}$$

$$\therefore \ f(\lambda|p,\boldsymbol{r},\boldsymbol{x}) \sim Gamma(a+\sum_{i=1}^{n} x_i, b+\sum_{i=1}^{n} r_i)$$

(ii)

$$\because \ f(p|\lambda,\boldsymbol{r},\boldsymbol{x}) = \frac{f(p;\lambda,\boldsymbol{r},\boldsymbol{x})}{f(\lambda,\boldsymbol{r},\boldsymbol{x})} \propto f(p;\lambda,\boldsymbol{r},\boldsymbol{x})$$

$$\therefore \ f(p|\lambda,\boldsymbol{r},\boldsymbol{x}) \propto p^{\sum_{i=1}^{n} r_i}(1-p)^{n-\sum_{i=1}^{n} r_i}$$

$$\therefore \ f(p|\lambda,\boldsymbol{r},\boldsymbol{x}) \sim Beta(\sum_{i=1}^{n} r_i + 1, n+1-\sum_{i=1}^{n} r_i)$$

(iii)

$$\because \ f(r_i|\lambda,p,x_i) = \frac{f(r_i;\lambda,p,x_i)}{f(\lambda,p,x_i)} \propto f(r_i;\lambda,p,x_i)$$

First, we need to note that:

$$\because \ (x_i|\boldsymbol{r},\lambda,p) \sim Poisson(\lambda r_i)$$

$$\therefore \ if \ x_i \neq 0, \ then: \ r_i \neq 0$$

$$\therefore\ if\ x_i \neq 0,\ then:\ r_i = 1$$

When $x_i = 0$, we have:

$$f(r_i|\lambda, p, x_i) \propto (pe^{-\lambda})^{r_i}(1-p)^{1-r_i} \propto (\frac{pe^{-\lambda}}{pe^{-\lambda}+1-p})^{r_i}(\frac{1-p}{pe^{-\lambda}+1-p})^{1-r_i}$$

$$\therefore\ f(r_i|\lambda, p, x_i) \sim Bernoulli(\frac{pe^{-\lambda}}{pe^{-\lambda}+1-p}),\ when\ x_i = 0.$$

$$\therefore\ f(r_i|\lambda, p, x_i) \sim Bernoulli(\frac{pe^{-\lambda}}{pe^{-\lambda}+(1-p)I(x_i=0)})$$

(c)

```
#####The MLE of the lambda & p.
meanL_mle <- mean(Y)
meanP_mle <- mean(R)
meanL_mle
```

```
## [1] 1.96
```

```
meanP_mle
```

```
## [1] 0.25
```

```
##(c)
P <- c()
Lambda <- c()
a <- 1
b <- 1
p <- 0.5
l <- 1
label <- X==0
n <- sum(label)
sum_X <- sum(X)

N = 100000
for (i in 1:N){
  r <- rep(1,100)
  p0 <- (p*exp(-l))/(p*exp(-l)+1-p)
  r[label] <- sample(c(0,1), size = n, replace = TRUE, prob = c(1-p0,p0))
  sum_r <- sum(r)
  l <- rgamma(1, a+sum_X, b+sum_r)
  Lambda <- c(Lambda,l)
  p <- rbeta(1, 1+sum_r, 101-sum_r)
  P <- c(P,p)
}
```

```
M = 30000
d = 10
P1 <- P[-(1:M)]
P2 <- P1[(1:(N-M)) %% d == 0 ]
Lambda1 <- Lambda[-(1:M)]
```

```r
Lambda2 <- Lambda1[(1:(N-M)) %% d == 0 ]

meanL <- mean(Lambda2)
L_95 <- quantile(Lambda2,c(0.025,0.975))
meanP <- mean(P2)
P_95 <- quantile(P2,c(0.025,0.975))
meanL
```

```
## [1] 1.961752
```

```r
L_95
```

```
##     2.5%    97.5%
## 1.348822 2.693497
```

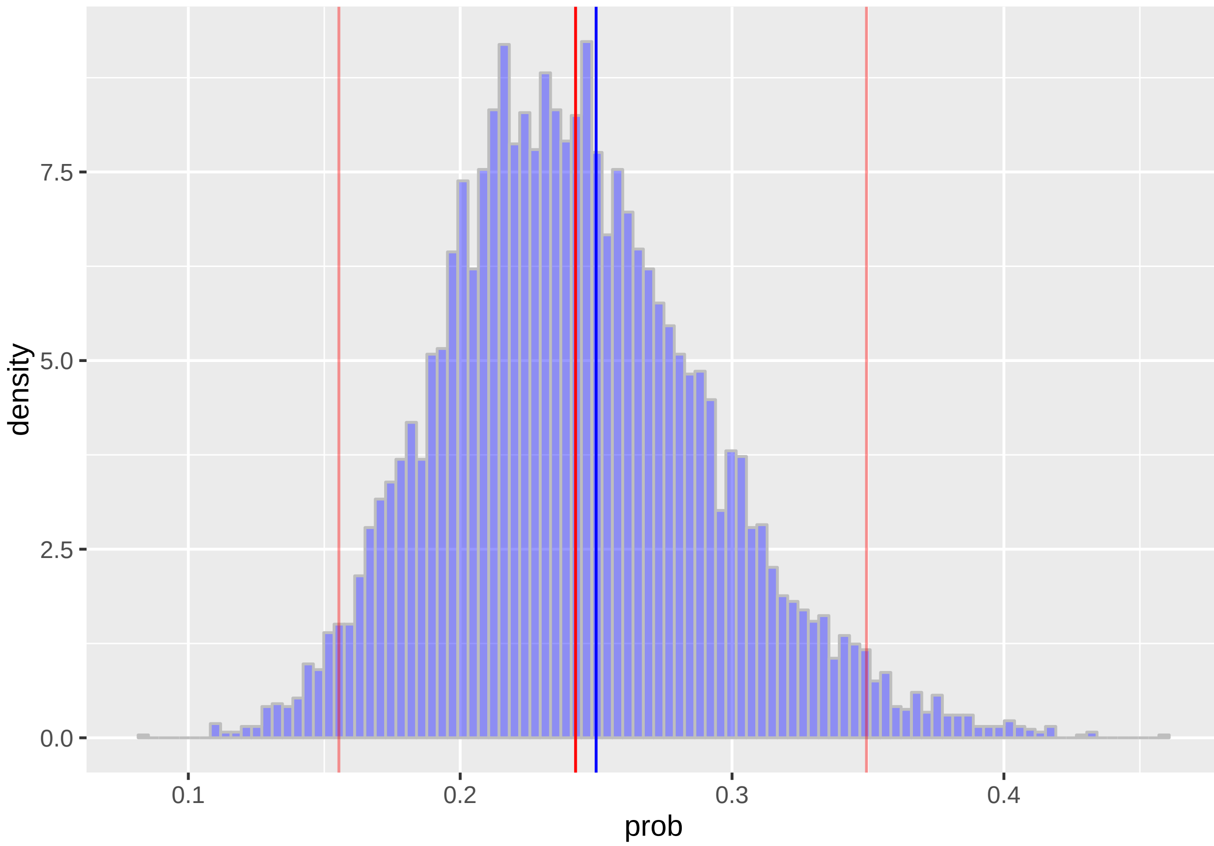
```r
meanP
```

```
## [1] 0.242447
```

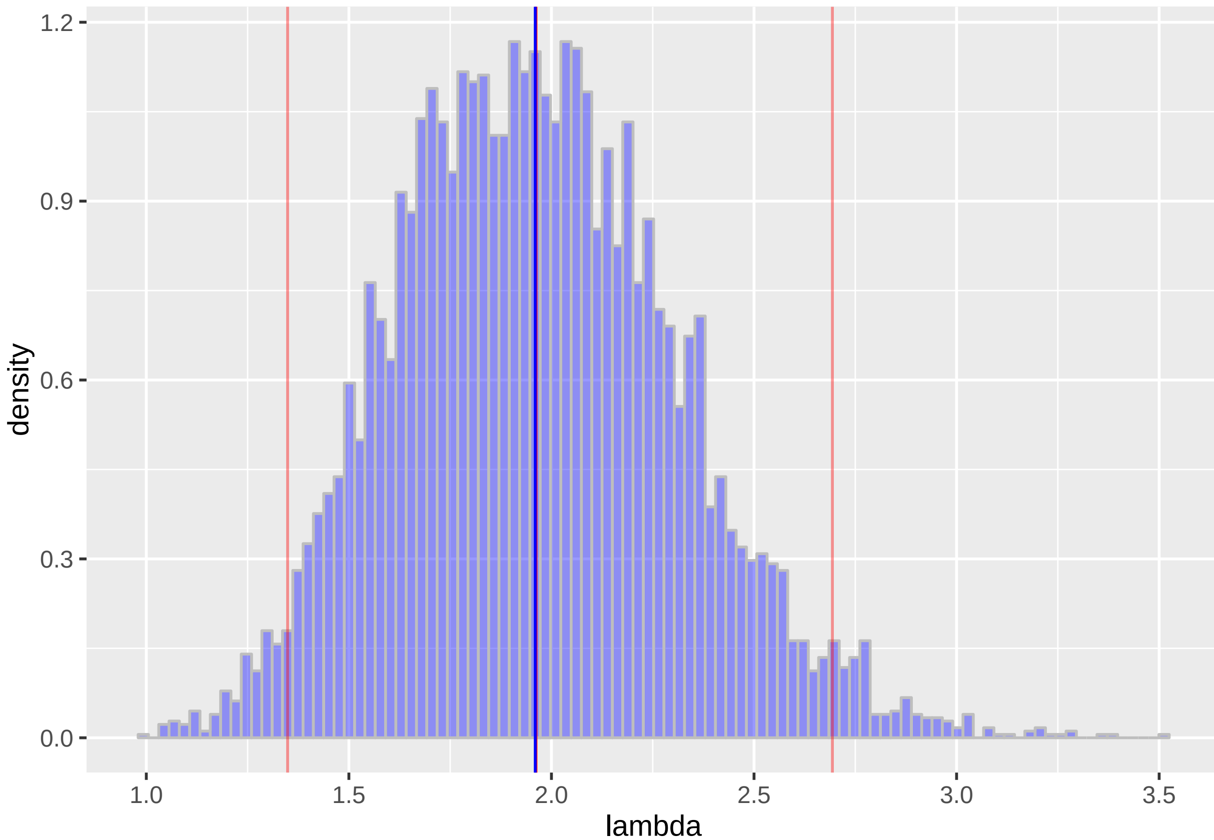
```r
P_95
```

```
##      2.5%     97.5%
## 0.1553752 0.3494559
```

```r
d <- data.frame(prob = P2, lambda = Lambda2)
library(ggplot2)
ggplot(data = d, aes(x = prob,..density..))+
  geom_histogram(bins = 100, fill = "blue", alpha = 0.4, colour = "grey")+
  geom_vline( xintercept = meanP, colour = "red")+
  geom_vline( xintercept = meanP_mle, colour = "blue")+
  geom_vline( xintercept = P_95, colour = "red", alpha = 0.4)
```




```r
ggplot(data = d, aes(x = lambda,..density..))+
  geom_histogram(bins = 100, fill = "blue", alpha = 0.4, colour = "grey")+
  geom_vline( xintercept = meanL, colour = "red")+
  geom_vline( xintercept = meanL_mle, colour = "blue")+
  geom_vline( xintercept = L_95, colour = "red", alpha = 0.4)
```



Note: The change of a and b are shown using the code in Problem 3.

Problem2

```
set.seed(521521)
a1 <- 1.5
a2 <- 2
f <- function(z,a1,a2) z^(-3/2)*exp( -a1*z -a2/z )

a <-1
b <-1
N <- 20000
x <- 1
X <- c(x)
for (i in 1:N){
  y <- rgamma(1,a,b)
  r0 <- ( f(y,a1,a2)/f(x,a1,a2) ) * ( dgamma(x,a,b)/dgamma(y,a,b) )
  r <- min(r0,1)
  u <- runif(1)
  if (u <= r){
    x <- y
  }
  X <- c(X,x)
}
```

```r
M = 10000
d = 10
X1 <- X[-(1:M)]
X2 <- X1[(1:(N-M)) %% d == 0 ]


mean(X2)
```

```
## [1] 1.168295
```

```r
sqrt(a2/a1)
```

```
## [1] 1.154701
```

```r
mean(1/X2)
```

```
## [1] 1.105579
```

```r
sqrt(a1/a2)+1/(2*a2)
```

```
## [1] 1.116025
```

```r
### TRUE density function
Int <- integrate(f,0,Inf,a1=a1,a2=a2)$value
f1 <- function(z,a1,a2){
  (1/Int)*z^(-3/2)*exp( -a1*z -a2/z )
}
########################

d <- data.frame( x = X2 )
ggplot(data = d)+
  stat_function(fun = f1, args = list(a1=a1, a2=a2), colour = "red")+
  geom_histogram(aes(x = x,..density..),bins = 100,
                 fill = "green", alpha = 0.4, colour = "grey")
```

Problem3

(Using Rcpp)

```
library(Rcpp)
```

```
## Warning: package 'Rcpp' was built under R version 3.3.2
```

```
cppFunction('NumericMatrix Gibbs(NumericVector X, double a, double b, int N, double p, double l) {

NumericMatrix mat(N,2);
int n, m = X.size();
LogicalVector label(m);
NumericVector r(m);
int sum_x, sum_r, i,j;
double p0, u;

sum_x = sum(X);
mat(0,0) = l;
mat(0,1) = p;
label = (X==0);
n = sum(label);

GetRNGstate();
for (i = 1; i < N; i++){
```

```
  r = rep(1,m);
  p0 = (p*exp(-l))/(p*exp(-l)+1-p);

  for (j = 0; j < m; j++){
    if ( label(j) == TRUE){
      u = runif(1)[0];
      if ( u < 1-p0 ) r(j) = 0;
    }
  }

  sum_r = sum(r);
  l = rgamma(1, a+sum_x,  1/(b+sum_r) )[0];
  mat(i,0) = l;
  p = rbeta(1, 1+sum_r, m+1-sum_r)[0];
  mat(i,1) = p;
}
PutRNGstate();

return(mat);

}')


rcppGibbs <- function(X,a,b,p,l,N,M,d){
  Rout <- Gibbs(X,a,b,N,p,l)
  Rout1 <- Rout[-(1:M),]
  Rout2 <- Rout1[(1:(N-M)) %% d == 0 , ]
  Out <- data.frame(Lambda = Rout2[,1], P = Rout2[,2])
  apply(Out,2,quantile,c(0.025,0.5,0.975))
}




set.seed(580580)
Y <- rpois(100,2)
R <- sample(c(0,1), size = 100, replace = TRUE, prob = c(0.7,0.3))
X <- Y*R


a <- 1
b <- 1
p <- 0.5
l <- 1
N <- 100000
M = 30000
d = 10

rcppGibbs(X,a,b,p,l,N,M,d)
```

```
##         Lambda         P
## 2.5%  1.338804 0.1530810
## 50%   1.937348 0.2403613
## 97.5% 2.670967 0.3482589
```

Using R&C Interface

C code

```
#include <R.h>
#include <Rinternals.h>
#include <Rmath.h>


SEXP RCGibbs(SEXP X, SEXP a_, SEXP b_, SEXP N_, SEXP p_, SEXP l_){

double a,b;
double p,l;
int N, n, m = length(X);
int r[m], label[m];
int sum_x, sum_r, i,j,k;
double p0, u;

a = asReal(a_);
b = asReal(b_);
N = asInteger(N_);
p = REAL(p_)[0];
l = REAL(l_)[0];

SEXP P = PROTECT(allocVector(REALSXP, N));
SEXP Lambda = PROTECT(allocVector(REALSXP, N));
SEXP Rout = PROTECT(allocVector(VECSXP, 2));
SET_VECTOR_ELT(Rout, 0, P);
SET_VECTOR_ELT(Rout, 1, Lambda);


sum_x = 0;
for (k=0; k<m; k++){
sum_x = sum_x + REAL(X)[k];
}

REAL(Lambda)[0] = l;
REAL(P)[0] = p;

for (k=0; k<m; k++){
if (REAL(X)[k] == 0){
label[k] = 1;
}
else{
label[k] = 0;
}
}

n = 0;
for (k=0; k<m; k++){
n = n + label[k];
}
```

```
GetRNGstate();

for (i = 1; i < N; i++){

for (j=0; j<m; j++){
r[j] = 1;
}
                    p0 = (p*exp(-l))/(p*exp(-l)+1-p);

                    for (j = 0; j < m; j++){
                    if ( label[j] == 1){
                    u = runif(0,1);
                    if ( u < 1-p0 ) r[j] = 0;
                    }
                    }


                    sum_r = 0;
                    for (j=0; j<m; j++){
                      sum_r = sum_r + r[j];
                    }
                    l = rgamma(a+sum_x,  1/ (b+sum_r) );
                    REAL(Lambda)[i] = l;
                    p = rbeta(1+sum_r, m+1-sum_r);
                    REAL(P)[i] = p;
}

PutRNGstate();


UNPROTECT(3);
return Rout;
}
```

R code

```
rc_int_Gibbs <- function(X,a,b,p,l,N,M,d){
  dyn.load("/Users/apple/Desktop/ISU 2017 spring/STAT580/Homework/hw5/RC_gibbs.so")
  Rout <- .Call("RCGibbs", X,a,b,N,p,l )
  Rout <- cbind(Rout[[1]],Rout[[2]])
  Rout1 <- Rout[-(1:M),]
  Rout2 <- Rout1[(1:(N-M)) %% d == 0 , ]
  Out <- data.frame(P = Rout2[,1], Lambda = Rout2[,2])
  apply(Out,2,quantile,c(0.025,0.5,0.975))
}



set.seed(580580)
Y <- rpois(100,2)
R <- sample(c(0,1), size = 100, replace = TRUE, prob = c(0.7,0.3))
X <- Y*R
```

```r
a <- 1
b <- 1
p <- 0.5
l <- 1
N <- 100000
M = 30000
d = 10

rc_int_Gibbs(X,a,b,p,l,N,M,d)
```

```
##                P   Lambda
## 2.5%  0.1530810 1.338804
## 50%   0.2403613 1.937348
## 97.5% 0.3482589 2.670967
```

```r
rc_int_Gibbs(X,1/5,b,p,l,N,M,d)
```

```
##                P   Lambda
## 2.5%  0.1585840 1.294789
## 50%   0.2424292 1.892053
## 97.5% 0.3539858 2.635815
```

```r
rc_int_Gibbs(X,5,b,p,l,N,M,d)
```

```
##                P   Lambda
## 2.5%  0.1504485 1.546576
## 50%   0.2313331 2.182862
## 97.5% 0.3348732 2.912532
```

```r
rc_int_Gibbs(X,a,1/5,p,l,N,M,d)
```

```
##                P   Lambda
## 2.5%  0.1510250 1.405560
## 50%   0.2345514 2.040225
## 97.5% 0.3399908 2.822157
```

```r
rc_int_Gibbs(X,a,5,p,l,N,M,d)
```

```
##                P   Lambda
## 2.5%  0.1691298 1.070302
## 50%   0.2612833 1.561024
## 97.5% 0.3818402 2.167759
```

```r
rc_int_Gibbs(X,a,b,0.2,1.5,N,M,d)
```

```
##                P   Lambda
## 2.5%  0.1557964 1.347843
## 50%   0.2395652 1.937652
## 97.5% 0.3452756 2.671255
```