Question 1.

Solution 1. *We are going to assume that a, b, c, d are known through the solutions.*

For part (i)*:*

To find the posterior distribution of θ, we have:

$$\begin{aligned} p(\theta|x_1,\ldots,x_n) &\propto p(\theta)\cdot p(x_1\ldots,x_n|\theta) \\ &= \frac{b^a}{\Gamma(a)}\theta^{a-1}e^{-b\theta}\cdot\prod_{i=1}^{n}\frac{\theta^{x_i}}{x_i!}\cdot e^{-\theta} \\ &= \frac{b^a}{\Gamma(a)}\cdot\frac{1}{\prod_{i=1}^{n}x_i!}\cdot\underbrace{\theta^{a+\sum_{i=1}^{n}x_i-1}\cdot e^{-(b+n)\theta}}_{\textit{kernel of Gamma}(a+\sum_{i=1}^{n}x_i,b+n)} \end{aligned}$$

Thus we have:

$$\theta|x_1,\ldots,x_n\sim \textit{Gamma}(a+\sum_{i=1}^{n}x_i,b+n)$$

With the completely same work as above, we have:

$$\delta|y_1,\ldots,y_m\sim \textit{Gamma}(c+\sum_{j=1}^{m}y_j,d+m)$$

For part (ii)*:*

To compute the posterior mode of $p(\theta|x_1,\ldots,x_n)$, since we have:

$$p(\theta|x_1,\ldots,x_n)=\frac{(b+n)^{a+\sum_{i=1}^{n}x_i}}{\Gamma(a+\sum_{i=1}^{n}x_i)}\cdot\theta^{a+\sum_{i=1}^{n}x_i-1}\cdot e^{-(b+n)\theta}$$

we take derivative on θ and set it to 0:

$$\begin{aligned}\frac{d}{d\theta}p(\theta|x_1,\ldots,x_n)=\frac{(b+n)^{a+\sum_{i=1}^{n}x_i}}{\Gamma(a+\sum_{i=1}^{n}x_i)}\cdot\Big[(a+\sum_{i=1}^{n}x_i-1)\theta^{a+\sum_{i=1}^{n}x_i-2}\cdot e^{-(b+n)\theta}+ \\ \theta^{a+\sum_{i=1}^{n}x_i-1}\Big(-(b+n)\Big)\cdot e^{-(b+n)\theta}\Big]=0\end{aligned}$$

This leads to

$$\theta^{a+\sum_{i=1}^{n}x_i-2}\cdot e^{-(b+n)\theta}\Big[(a+\sum_{i=1}^{n}x_i-1)+\Big(-(b+n)\Big)\theta\Big]=0$$

We denote the mode by $\hat{\theta}$, which is the solution to the above equation. Since it is not practical to let $\theta=0$, the only solution to the above equation is when

$$-(b+n)\hat{\theta}=-(a+\sum_{i=1}^{n}x_i-1)$$

Guanlin Zhang

Thus

$$\hat{\theta} = \frac{a + \sum_{i=1}^n x_i - 1}{b + n}$$

With completely same steps, we also find the mode for $p(\delta|y_1, \ldots, y_m)$ *is:*

$$\hat{\delta} = \frac{c + \sum_{j=1}^m y_j - 1}{d + m}$$

For part (iii)*:*

To compute the derivative and second derivative of $\log p(\theta|x_1, \ldots, x_n)$, *Since*

$$p(\theta|x_1, \ldots, x_n) = \frac{(b+n)^{a+\sum_{i=1}^n x_i}}{\Gamma(a + \sum_{i=1}^n x_i)} \theta^{a+\sum_{i=1}^n x_i - 1} \cdot e^{-(b+n)\theta}$$

We have

$$\log p(\theta|x_1, \ldots, x_n) = \log \left[\frac{(b+n)^{a+\sum_{i=1}^n x_i}}{\Gamma(a + \sum_{i=1}^n x_i)} \right] + (a + \sum_{i=1}^n x_i - 1) \log \theta - (b+n)\theta$$

So the first derivative is

$$\frac{d}{d\theta} \log p(\theta|x_1, \ldots, x_n) = \frac{a + \sum_{i=1}^n x_i - 1}{\theta} - (b+n)$$

and the second derivative is:

$$\frac{d^2}{d\theta^2} \log p(\theta|x_1 \ldots, x_n) = -\frac{a + \sum_{i=1}^n x_i - 1}{\theta^2}$$

With completely same steps, we have the first derivative of $\log p(\delta|y_1, \ldots, y_m)$ *as*

$$\frac{d}{d\delta} \log p(\delta|y_1, \ldots, y_m) = \frac{c + \sum_{j=1}^m y_j - 1}{\delta} - (d+m)$$

and the second derivative is:

$$\frac{d^2}{d\delta^2} \log p(\delta|y_1, \ldots, y_m) = -\frac{c + \sum_{j=1}^m y_j - 1}{\delta^2}$$

For part (iv):

To construct the normal approximation of $\theta|x_1, \ldots, x_n$, *observe the Taylor expansion on* $\log p(\theta|x_1, \ldots, x_n)$ *about the mode* $\hat{\theta}$*:*

$$\begin{aligned}
\log p(\theta|x_1, \ldots, x_n) &= \log p(\hat{\theta}|x_1, \ldots, x_n) + \frac{1}{2} \left[\frac{d^2}{d\theta^2} \log p(\theta|x_1, \ldots, x_n) \right]_{\theta=\hat{\theta}} (\theta - \hat{\theta})^2 + \cdots \\
&= \log p(\hat{\theta}|x_1, \ldots, x_n) + \frac{1}{-2\left(\frac{1}{-[\frac{d^2}{d\theta^2} \log p(\theta|x_1, \ldots, x_n)]_{\theta=\hat{\theta}}} \right)} (\theta - \hat{\theta})^2 + \cdots
\end{aligned}$$

This implies that $\log p(\theta|x_1,\dots,x_n)$ *has a quadratic estimation about the mode* $\hat{\theta}$ *and* $p(\theta|x_1,\dots,x_n)$ *has an estimation that is of normal density form. Namely:*

$$p(\theta|x_1,\dots,x_n) = Constant \cdot exp\Big\{\frac{1}{-2\Big(\frac{1}{-[\frac{d^2}{d\theta^2}\log p(\theta|x_1,\dots,x_n)]_{\theta=\hat{\theta}}}\Big)} \cdot (\theta-\hat{\theta})^2\Big\}$$

So $\theta|x_1,\dots,x_n$ *has a normal approximation with*

$$\begin{aligned}
\mu &= \hat{\theta} = \frac{a+\sum_{i=1}^n x_i - 1}{b+n} \\
\sigma^2 &= \frac{1}{-[\frac{d^2}{d\theta^2}\log p(\theta|x_1,\dots,x_n)]_{\theta=\hat{\theta}}} = -\frac{1}{-\frac{a+\sum_{i=1}^n x_i-1}{\theta^2}}\Big|_{\theta=\hat{\theta}} \\
&= \frac{\theta^2}{a+\sum_{i=1}^n x_i - 1}\Big|_{\theta=\hat{\theta}} \\
&= \frac{(a+\sum_{i=1}^n x_i - 1)^2}{(a+\sum_{i=1}^n x_i - 1)(b+n)^2} \\
&= \frac{a+\sum_{i=1}^n x_i - 1}{(b+n)^2}
\end{aligned}$$

With completely same steps, we conclude that $\delta|y_1,\dots,y_m$ *has a normal approximation with*

$$\begin{aligned}
\mu &= \frac{c+\sum_{j=1}^m y_j - 1}{d+m} \\
\sigma^2 &= \frac{c+\sum_{j=1}^m y_j - 1}{(d+m)^2}
\end{aligned}$$

Question 2.

Solution 2. For part (i)*:*

To estimate $P(\theta < \delta|x_1,\dots,x_n,y_1,\dots,y_m)$ *with the posterior distribution we computed before, We will use rgamma function to simulate* 10000 *times of posterior* θ *and* δ*. Since it is a relative large trial, we are just going to assume* $a=b=c=d=0.001$*. Given the data we have:*

$$\sum_{i=1}^n x_i = 57 \text{ and } \sum_{j=1}^m y_j = 50$$

Hence we have the following posterior distributions:

$$\begin{aligned}
\theta|x_1,\dots,x_n &\sim Gamma(a+\sum_{i=1}^n x_i, b+n) = Gamma(57.001, 169.001) \\
\delta|y_1,\dots,y_m &\sim Gamma(c+\sum_{j=1}^m y_j, d+m) = Gamma(50.001, 178.001)
\end{aligned}$$

With R simulation we got the following results (code and output in the Appendix section as required:)

```
## The posterior probability that theta less than delta is:  0.17
```

The result says the non-supplment group has only 17% *probability of having less adverse events than the supplment group, which means the supplment is effective.*

For part (ii)*:*

To estimate $P(\theta < \delta | x_1, \dots, x_n, y_1, \dots, y_m)$ *with normal approximation we computed before, we are just going to generate* 10000 *random normal samples for* θ *and* δ *each and compute the proportion of* $\theta < \delta$ *among samples.*

Based on the results before, we have

$$\mu_\theta = \frac{a + \sum_{i=1}^n x_i - 1}{b+n} = \frac{0.001+57-1}{169.001}$$
$$\mu_\delta = \frac{c + \sum_{j=1}^m y_j - 1}{d+m} = \frac{0.001+50-1}{178.001}$$
$$\sigma_\theta^2 = \frac{a + \sum_{i=1}^n x_i - 1}{(b+n)^2} = \frac{0.001+57-1}{169.001^2}$$
$$\sigma_\delta^2 = \frac{c + \sum_{j=1}^m y_j - 1}{(d+m)^2} = \frac{0.001+50-1}{178.001^2}$$

Our simulation shows the following results

```
## The normal approximated
## posterior probability that theta less than delta is:
##  0.1741
```

So the posterior probability of $\theta < \delta$ *under normal approximation is also* 17.4%*.*

For part (iii)*:*

Our answers from (i) and (ii) are the same up to the first two decimal places.The reason is that we have a relative large trial here and our n *and* m *are large enough such that the exact posterior distributions get close enough to their normal approximations.*

Question 3.

Solution 3. *In order to draw* $\lambda_1, \lambda_2, \theta | y$ *using Gibbs sampling(successive substitution sampling), we need to know the conditional posterior distribution of* $\lambda_1 | \lambda_2, \theta, y$*,* $\lambda_2 | \lambda_1, \theta, y$ *and* $\theta | \lambda_1, \lambda_2, y$*.*

Notice that we have:

$$\begin{aligned}
p(\theta, \lambda_1, \lambda_2 | y) &\propto p(\theta, \lambda_1, \lambda_2, y) = p(\theta, \lambda_1, \lambda_2) \cdot p(y | \theta, \lambda_1, \lambda_2) \\
&= p(\theta) \cdot p(\lambda_1) \cdot p(\lambda_2) \cdot p(y | \lambda_1, \lambda_2, \theta) \\
&= \frac{1}{8} \cdot \Big(\frac{0.001^{0.001}}{\Gamma(0.001)} \cdot \lambda_1^{0.001-1} \cdot e^{-0.001\lambda_1} \Big) \cdot \Big(\frac{0.001^{0.001}}{\Gamma(0.001)} \cdot \lambda_2^{0.001-1} \cdot e^{-0.001\lambda_2} \Big) \\
&\quad \cdot \Big(\prod_{i=1}^{3+\theta} \frac{\lambda_1^{y_i}}{y_i!} \cdot e^{-\lambda_1} \Big) \Big(\prod_{i=4+\theta}^{12} \frac{\lambda_2^{y_i}}{y_i!} \cdot e^{-\lambda_2} \Big)
\end{aligned}$$

Since this is conditioned on data y, we could further simplify the above expression and get

$$p(\theta, \lambda_1, \lambda_2|y) \propto \lambda_1^{0.001-1} \cdot e^{-0.001\lambda_1} \cdot \lambda_2^{0.001-1} \cdot e^{-0.001\lambda_2} \cdot \Big(\prod_{i=1}^{3+\theta} \lambda_1^{y_i} \cdot e^{-\lambda_1}\Big)\Big(\prod_{i=4+\theta}^{12} \lambda_2^{y_i} \cdot e^{-\lambda_2}\Big)$$
$$= \lambda_1^{(0.001-1)+\sum_{i=1}^{3+\theta} y_i} \cdot \lambda_2^{(0.001-1)+\sum_{i=4+\theta}^{12} y_i} \cdot e^{-(0.001+3+\theta)\lambda_1} \cdot e^{-(0.001+12-(3+\theta))\lambda_2}$$

We will define (or call) the last line of the above equation as L_θ if we regard it as a function of θ that is conditioned on λ_1, λ_2, y.

Particularly, we have

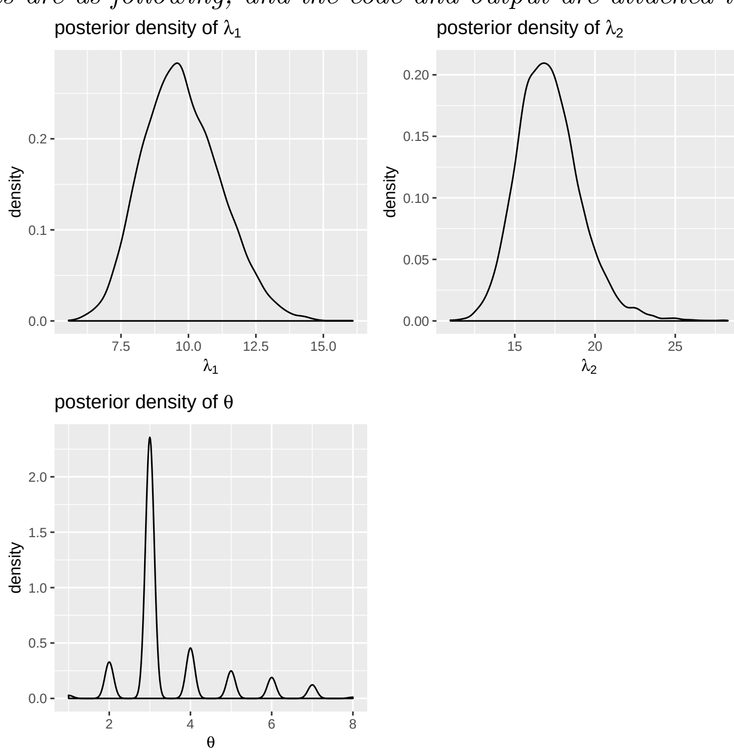
$$p(\theta|\lambda_1, \lambda_2, y) = \frac{L_\theta}{\sum_{j=1}^{8} L_j}, \theta = 1, 2, 3, \ldots, 8$$

Also, from the expression above, we have:

$$\lambda_1|\lambda_2, \theta, y \sim Gamma(0.001 + \sum_{i=1}^{3+\theta} y_i, 0.001 + 3 + \theta)$$
$$\lambda_2|\lambda_1, \theta, y \sim Gamma(0001 + \sum_{i=4+\theta}^{12} y_i, 0.001 + 9 - \theta)$$

With those conditional posterior distribution above, we are now ready to draw with Gibbs sampling method.

The density plots are as following, and the code and output are attached in the appendix.



Question 4.

Solution 4. *Given* $g(\theta) = \frac{e^{2\theta}}{(1+exp(2\theta))^2}$*, we are asked to draw sample from* $\pi(\theta) = g(\theta)/c$.

We are going to use all three methods, which are rejection sampling, weighted bootstrap sampling, and Metropolis sampling.

First of all, observe that $g(\theta)$ *is an even function (symmetric about y axis):*

$$\begin{aligned} g(\theta) &= \frac{e^{2\theta}}{1+2\cdot e^{2\theta}+e^{4\theta}} \\ g(-\theta) &= \frac{e^{-2\theta}}{1+2\cdot e^{-2\theta}+e^{-4\theta}} \\ &= \frac{e^{-2\theta}\cdot e^{4\theta}}{e^{4\theta}(1+2\cdot e^{-2\theta}+e^{-4\theta})} = \frac{e^{2\theta}}{1+2\cdot e^{2\theta}+e^{4\theta}} \\ &= g(\theta) \end{aligned}$$

Also notice that if we assume $\theta > 0$*, since* $e^{2\theta}+1 > e^{2\theta}$*, hence we have*

$$g(\theta) = \frac{e^{2\theta}}{(1+e^{2\theta})^2} < \frac{e^{2\theta}}{e^{4\theta}} = e^{-2\theta}$$

So overall for any $\theta \in \mathbb{R}$*, we have*

$$g(\theta) < e^{-2|\theta|}$$

Notice that the right hand side $w(\theta) = e^{-2|\theta|}$ *is the probability density function of Laplace distribution with* $\mu = 0$ *and* $\sigma = \frac{1}{2}$*. So we can use this function as our envelope funtion for rejection sampling, as well as the weight function for weighted bootstrap sampling.*

On the other hand, for Metropolis sampling, $g(\theta)$ *as the target function is our invariant distribution for the markov process. We choose the candidate density for transition probability of the markov process as normal (which according to notes, is our choise most of the time), and in the mean time, to control the rejection rate in the range of between* 30% *and* 50%*, we pick the standard deviation of the normal candidate as* $\sigma = 1$ *(there are other choices in this case that also meet the criteria, for example,* $\sigma = 1.5$*). We also pick the starting point as* $x[1] = 0$*, which is the mean value of the target distribution.*

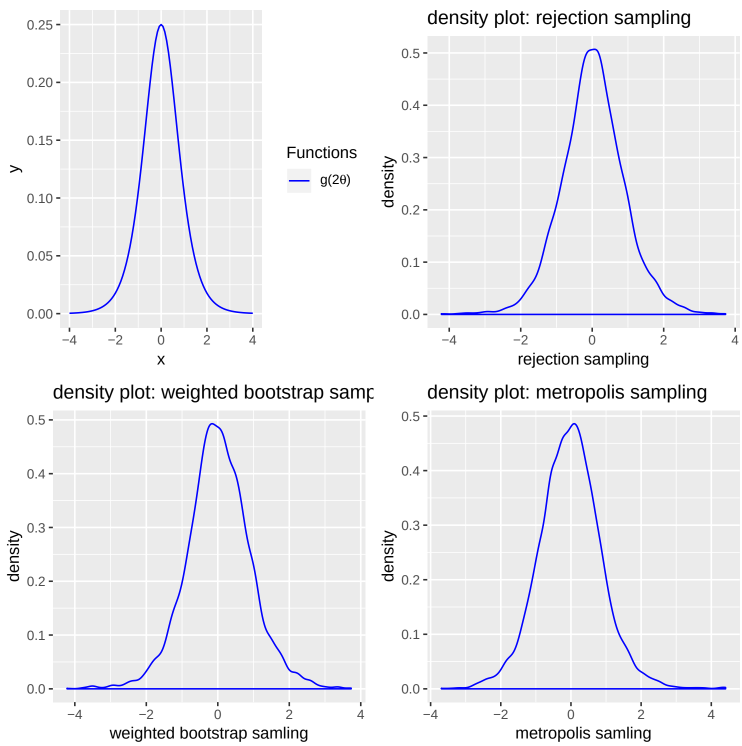
The code and detailed output are attached in the appendix.

However we summarize our findings here as well:

	rejection	*weighted bootstrap*	*metropolis*
mean	0.02	0.02	-0.02
sd	0.89	0.90	0.89
95% *posterior interval*	$(-1.74, 1.84)$	$(-1.78, 1.86)$	$(-1.81, 1.77)$

As we can see that all three algorithms generate similar statistics such as mean, standard deviation and 95% *posterior intervals.*

We give a plot of the original function $g(\theta)$*, as well as density plots of all three algorithms:*



As we can observe that the density plots of all three algorithms are approaching the target function $g(\theta)$ pretty well. The difference on the peak between the original function $g(\theta)$(close to 0.25*) and the samplings(close to* 0.5*) is due to the fact that $g(\theta)$ is not normalized.*

Appendix

Code and outcome for Question 2:

```
#BIOS902 Midterm

#Question 2

#Input data
#non-supplement group
x<- c(0,1,0,0,0,0,1,0,0,0,1,0,0,1,0,0,1,2,0,0,0,0,0,0,0,0,0,0,0,
    0,1,0,0,0,0,1,0,0,0,1,0,1,0,0,0,1,1,0,0,1,0,0,0,2,0,0,0,0,
    0,0,1,1,0,0,0,0,0,1,1,0,0,1,3,0,0,1,0,0,0,0,0,0,0,0,1,0,0,
    0,0,2,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,3,0,1,2,
    0,0,3,0,0,0,2,0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,0,0,4,0,0,0,0,
    0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,5)
#supplement group
y <- c(0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,2,4,0,0,0,0,0,0,0,1,
    0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,1,1,1,0,0,0,0,0,
    0,0,0,0,0,0,1,0,0,0,0,0,1,0,1,1,0,0,0,0,0,0,0,0,0,3,0,0,0,
    0,0,0,0,1,0,0,0,0,0,0,2,0,1,0,0,0,0,0,1,0,0,0,1,0,0,1,0,1,
```

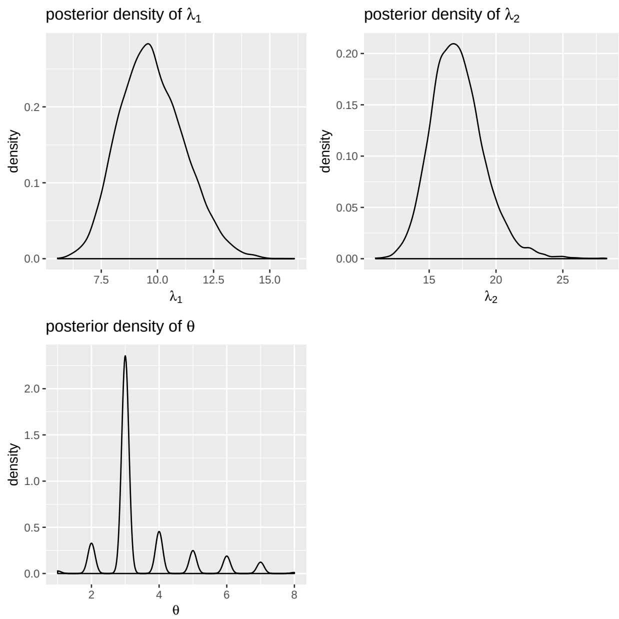
```
    0,0,0,1,0,0,0,0,0,0,1,0,1,0,0,0,1,0,0,0,0,0,0,0,3,0,0,0,0,
    1,1,0,0,0,1,0,1,0,0,1,0,1,1,0,1,0,0,0,0,0,1,0,0,2,0,1,0,0,
    0,1,1,0)
n <- length(x); m <- length(y)
#generate necessary quantities for data analysis
a <- 0.001; b <- 0.001; c <- 0.001; d <- 0.001 #for large trials
sum_x <- sum(x); sum_y <- sum(y) #preparing data for posterior
#mean for normal approximation
mu_x <- (a + sum_x - 1)/(b + n); mu_y <- (c + sum_y - 1)/(d + m)
#variance for normal approximation
var_x <- (a + sum_x - 1)/(b + n)^2
var_y <- (c + sum_y - 1)/(d + m)^2

#part 1: estimate prob of theta < delta based on posterior
set.seed(34) #set seed number as 34
S = 10000 #set simulation size as S
theta <- rgamma(n = S, shape = a + sum_x, rate = b + n)
delta <- rgamma(n = S, shape = c + sum_y, rate = d + m)
prob <- mean(theta < delta)
cat("The posterior probability that theta less than delta is: ",prob)

## The posterior probability that theta less than delta is:  0.17

#part 2: estimate prob of theta < delta based on normal approximation
theta_n <- rnorm(n = S, mean = (a + sum_x - 1)/(b + n),
                 sd = sqrt((a + sum_x - 1)/(b + n)^2))
delta_n <- rnorm(n = S, mean = (c + sum_y - 1)/(d + m),
                 sd = sqrt((c + sum_y - 1)/(d + m)^2))
prob_n <- mean(theta_n < delta_n)
cat("The normal approximated
posterior probability that theta less than delta is: \n",prob_n)

## The normal approximated
## posterior probability that theta less than delta is:
##  0.1741
```

Code and outcome for Question 3:

```
#Question 3

#Input data
y <- c(7, 7, 15, 11, 7, 9, 16, 15, 15, 16, 22, 18)

###matrices
S <- 10000
```

```
lam1=rep(NA,S,1)
lam2=rep(NA,S,1)
theta=rep(NA,S,1)

###Starting values
lam1[1]=lam2[1]=mean(y)
theta[1]=4

###conditional post for theta
Post=function(lam1,lam2)
{
  IL=matrix(NA,8,1) #this is log of L_theta
  for (j in 1:8)
  {
    IL[j]=((0.001 - 1) + sum(y[1: (3 + j)]))*log(lam1) +
      ((0.001 - 1) + sum(y[(4 + j):12]))*log(lam2) -
      (0.001 + 3 + j)*lam1 - (0.001 + 12 - (3 + j))*lam2
  } #log of the joinst posterior
  exp(IL)/sum(exp(IL)) #this gives conditional post for theta
}

###Successive Substitution Sampling
set.seed(34)
for (i in 2:S){
  lam1[i]=rgamma(1, 0.001 + sum(y[1:(3 + theta[i - 1])]),
                 0.001 + 3 + theta[i - 1])
  lam2[i]=rgamma(1, 0.001 + sum(y[(4 + theta[i - 1]):12]),
                 0.001 + 9 - theta[i - 1])
  theta[i]=sample(seq(1:8), 1, replace= TRUE, prob=Post(lam1[i], lam2[i]))}
####Posterior

#make density plot with ggplot
library(ggplot2)
library(gridExtra)
den_plot <- data.frame(lam1, lam2, theta)
plam1 <- ggplot(data = den_plot, aes(x = lam1)) +
          geom_density() +
          labs(title = expression(paste("posterior density of ",lambda[1])),
               x = expression(lambda[1]))
plam2 <- ggplot(data = den_plot, aes(x = lam2)) +
  geom_density() +
  labs(title = expression(paste("posterior density of ",lambda[2])),
       x = expression(lambda[2]))
ptheta <- ggplot(data = den_plot, aes(x = theta)) +
```

```
  geom_density() +
  labs(title = expression(paste("posterior density of ",theta)),
       x = expression(theta))
grid.arrange(plam1, plam2, ptheta, nrow = 2)
```



Code and outcome for Question 4

```
#Question 4

library(ggplot2)
library(gridExtra)
library(rmutil) #need this package for laplace distribution

#define g(theta)
g<- function(x){
  exp(2*x)/(1 + exp(2*x))^2
}

# laplace distribution with mu = 0, sigma = 0.5 as envelope
w<- function(theta) {
  d<-dlaplace(theta, m = 0, s = 0.5)
}

#Rejection Sampling
set.seed(34)
theta<- rlaplace(10000, m = 0, s = 0.5) #Candidate thetas drawn from w(theta)
ratio<- g(theta)/w(theta) #Ratio for evaluation
```

```
u<- runif(10000)
post_rejection<- theta[u<ratio] ####Accept or reject?

length(post_rejection)

## [1] 4927

mean(post_rejection)

## [1] 0.02094486

sd(post_rejection)

## [1] 0.8889

quantile(post_rejection, c(.025, .975))

##      2.5%     97.5%
## -1.738235  1.839349

#weighted bootstrap sampling
#use the same envelope function as in rejection sampling
#so we already drew theta
q<- g(theta)/w(theta)/sum(g(theta)/w(theta))######Ratio wts
post_bootstrap=sample(theta, 10000, replace=TRUE, prob=q)
mean(post_bootstrap)

## [1] 0.0216452

sd(post_bootstrap)

## [1] 0.9036935

quantile(post_bootstrap, c(.025, .975))

##      2.5%     97.5%
## -1.771840  1.858301

##Metropolis sampling

####10,000 simulations & setup matrices to be filled
post_metropolis=matrix(NA, 10000,1)
alpha=matrix(NA, 10000,1)
reject=matrix(1,10000,1)

###Width for candidate density (tunes rejection)
sigma <- 1 #based on controling rejection rate 30%-50%
```

```
######Starting value
post_metropolis[1] <- 0  ###the mean of the target distribution

set.seed(34)
### Metropolis Algorithm
for (i in 1:9999)
{
  ####Candidate density: normal
  y=rnorm(1,post_metropolis[i],sigma)

  ####Ratio
  alpha[i]=min((g(y)*dnorm(post_metropolis[i],y,sigma))/
               (g(post_metropolis[i])*dnorm(y,post_metropolis[i],sigma)),1)
  ####Keep last one
  post_metropolis[i+1]=post_metropolis[i]
  ###Change to candidate draw if rule works and track "rejections"
  if(runif(1)<alpha[i]) {post_metropolis[i+1]=y
  reject[i]=0}
}

###Want rejection rate 30-50% or so
mean(reject)

## [1] 0.3454

mean(post_metropolis)

## [1] -0.02247845

sd(post_metropolis)

## [1] 0.8908291

quantile(post_metropolis, c(0.025, 0.975))

##      2.5%     97.5%
## -1.813374  1.766967

#plot density plots
p_g <- ggplot(data = data.frame(x = 0), mapping = aes(x = x)) +
  stat_function(fun=g, geom="line", aes(colour="g")) +
  scale_x_continuous(limits = c(-4, 4)) +
  scale_color_manual(name = "Functions",
                     values = "blue",
                     labels = expression(paste("g(2",theta,")")))
```

```
den_plot_4 <- data.frame(post_rejection)
den_plot_4_2 <- data.frame(post_bootstrap, post_metropolis)
p_rej <- ggplot(data = den_plot_4, aes(x = post_rejection)) +
  geom_density(colour = "blue") +
  labs(title = "density plot: rejection sampling",
       x = "rejection sampling")
p_boot <- ggplot(data = den_plot_4_2, aes(x = post_bootstrap)) +
  geom_density(colour = "blue") +
  labs(title = "density plot: weighted bootstrap sampling",
       x = "weighted bootstrap samling")
p_metro <- ggplot(data = den_plot_4_2, aes(x = post_metropolis)) +
  geom_density(colour = "blue") +
  labs(title = "density plot: metropolis sampling",
       x = "metropolis samling")
grid.arrange(p_g, p_rej, p_boot, p_metro, nrow = 2)
```