Mid Term

Md Kamrul Hasan Khan

March 31, 2017

Answer to the Question Number 1

Since samples are uniformly distributed, the joint pdf will be

$$f(x,y) = \frac{1}{2}; \qquad |\, x \,| + |\, y \,| \leq 1$$

The marginal distribution of x is

$$f(x) = \int_y f(x,y)dy = \int_{|x|-1}^{1-|x|} \frac{1}{2} dy = 1 - |\, x \,| = \begin{cases} 1+x; & -1 < x < 0 \\ 1-x; & 0 < x < 1 \end{cases}$$

Now,

$$f(y|x) = \frac{f(x,y)}{f(x)} = \frac{1}{2(1-|\, x \,|)}; \qquad -(1-|\, x \,|) < y < (1-|\, x \,|)$$

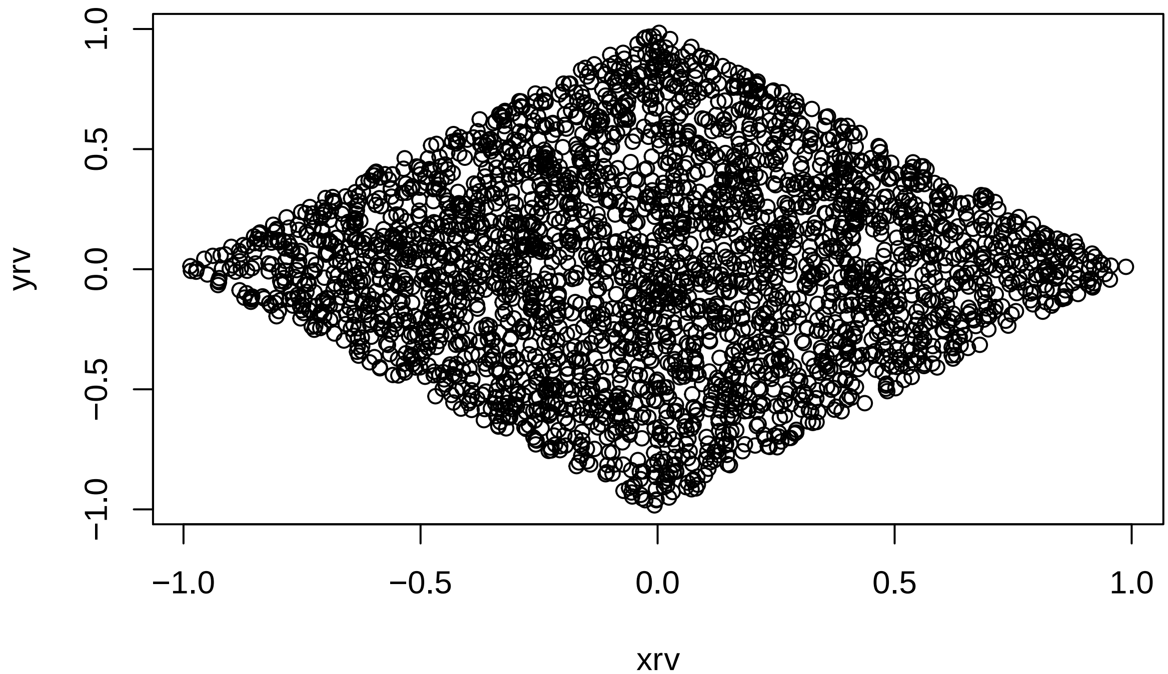
So, $y|x \sim Unif(-(1-|\, x \,|), (1-|\, x \,|))$

The cdf of x is

$$F(x) = \begin{cases} \frac{(1+x)^2}{2}; & -1 < x < 0 \\ 1 - \frac{(1-x)^2}{2}; & 0 < x < 1 \end{cases}$$

So, to generate sample from the given dimond shaped area first generate x from $\begin{cases} \sqrt{2u} - 1; & 0 < u < 0.5 \\ 1 - \sqrt{2(1-u)}; & 0.5 < u < 1 \end{cases}$; where $u \sim Unif(0,1)$ and then $y|x$ from $Unif(-(1-|\, x \,|), (1-|\, x \,|))$

```
diamond_sampling <- function(n)
{
 xrv = matrix(0,n,1)
 yrv = matrix(0,n,1)
 for (i in 1:n)
{
 u = runif(1)
 if (u<0.5) {  x = sqrt(2*u) -1} else {x = 1-sqrt(2*(1-u))}

  y_give_x = runif(1, -(1-abs(x)), (1-abs(x)))
  xrv[i,] = x
  yrv[i,] = y_give_x
  }
plot(xrv,yrv)
return(cbind(xrv,yrv))
}
y = diamond_sampling(3000)
```



Answer to the Question Number 2

Given,

$\beta_i^{Ridge} = \underset{\beta \in \mathbb{R}^n}{\text{argmin}}[(y_i - \beta_i)^2 + \lambda \beta_i^2]$

$\Rightarrow \frac{\partial \beta_i^{Ridge}}{\partial \beta_i} = -2(y_i - \beta_i) + 2\lambda\beta_i = 0$

$\Rightarrow -y_i + \beta_i + \lambda\beta_i = 0$

$\Rightarrow \hat{\beta}_i^{Ridge} = \frac{y_i}{1+\lambda}$

And

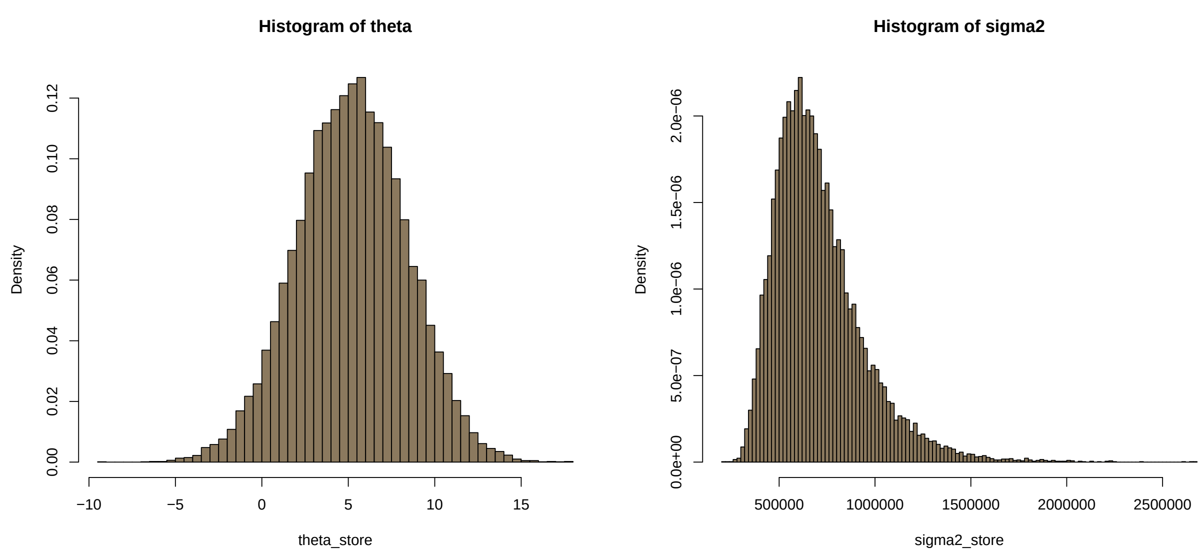
$\frac{\partial^2 \hat{\beta}_i^{Ridge}}{\partial \beta_i^2} = 1 + \lambda > 0$

So, $\hat{\beta}_i^{Ridge} = \frac{y_i}{1+\lambda}; i = 1, 2, ..., n$

Answer to the Question Number 2

```
set.seed(7)
x = c(91, 504, 557, 609, 693, 727, 764, 803, 857, 929, 970, 1043, 1089, 1195, 1384, 1713)
n = length(x)
nmc = 20000
a = b = 3
tau2 = 10
theta0 = 5
sigma2 = var(x)
theta_store = matrix(0,nmc,1)
```

```r
sigma2_store = matrix(0,nmc,1)
for(iter in 1:nmc)
{
theta_var = 1/((n/sigma2) + (1/tau2))
theta_mean = theta_var*( (n*mean(x))/sigma2 + (theta0/tau2))
theta = rnorm(1, theta_mean, sqrt(theta_var))

sigma2_shape = n/2 + a
sigma2_rate = 0.5 *((n-1)*var(x) + n*(mean(x)-theta)^2) + b
sigma2 = 1/rgamma(1,sigma2_shape, sigma2_rate)

theta_store[iter,] = theta
sigma2_store[iter,] = sigma2
}
par(mfrow=c(1,2))
hist(theta_store, "FD", freq = F, main= "Histogram of theta",col = "navajowhite4")
hist(sigma2_store, "FD", freq = F, main= "Histogram of sigma2",col = "navajowhite4")
```




```r
min(theta_store)
```

```
## [1] -9.161741
```

```r
max(theta_store)
```

```
## [1] 17.86951
```

The β part of the posterior distribution of σ^2 is very large. Therefore, the values of σ^2 should be very large. That means the mean and variance of the posterior distribution of θ should very very close to prior mean (θ_0) and variance (τ^2). The prior mean (θ_0) of θ is 5 and mean of data is 870.5, which does not make any sense. Here $\tau^2 = 10$, which is also very small. That means these initial values does not make any sense here.

Here, $f(X_i|\sigma^2, \theta_0, \tau^2) \sim N(\theta_0, \sigma^2 + \tau^2)$

So, a good choice of initial values is Empirical Bayes estimators which are $\theta_0 = \bar{x}$ and $\tau^2 + \sigma^2 = \text{Var(x)} \Rightarrow \tau^2 = \text{Var(x)} - \sigma^2$.

Now we need to choose a initial value for σ^2 which cannot be greater than Var(x). A good choice is truncated posterior distribution of σ^2 with limit 0 to Var(x).

```r
set.seed(7)

x = c(91, 504, 557, 609, 693, 727, 764, 803, 857, 929, 970, 1043, 1089, 1195, 1384, 1713)
a = b = 3
n = length(x)
theta0 = mean(x)
alpha= n/2 + a
beta = .5*(n-1)*var(x)+b
library(invgamma) ## for truncated distribution
```

```
## Warning: package 'invgamma' was built under R version 3.2.5
```

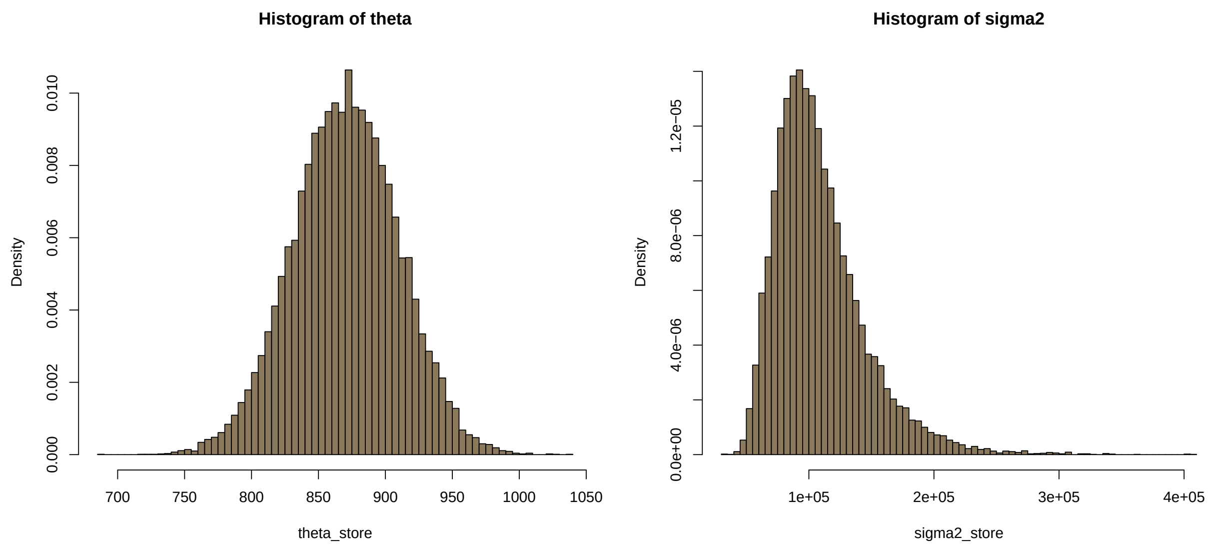
```r
rtrunc_inv_gamma = function(a, b, shape, rate)
  {
    F_a = pinvgamma (a, shape = shape, rate = rate)
    F_b = pinvgamma (b, shape = shape, rate = rate)
    x = qinvgamma((F_a + runif(1)*(F_b - F_a)), shape = shape, rate = rate)
        return(x)
  }
sigma2 = rtrunc_inv_gamma(0,var(x), alpha, beta)
tau2 = var(x) - sigma2
nmc = 20000

theta_store = matrix(0,nmc,1)
sigma2_store = matrix(0,nmc,1)
for(iter in 1:nmc)
{
theta_var = 1/((n/sigma2) + (1/tau2))
theta_mean = theta_var*( (n*mean(x))/sigma2 + (theta0/tau2))
theta = rnorm(1, theta_mean, sqrt(theta_var))

sigma2_shape = n/2 + a
sigma2_rate = 0.5 *((n-1)*var(x) + n*(mean(x)-theta)^2) + b
sigma2 = 1/rgamma(1,sigma2_shape, sigma2_rate)

theta_store[iter,] = theta
sigma2_store[iter,] = sigma2
}

min(theta_store)
```

```
## [1] 689.9411
```

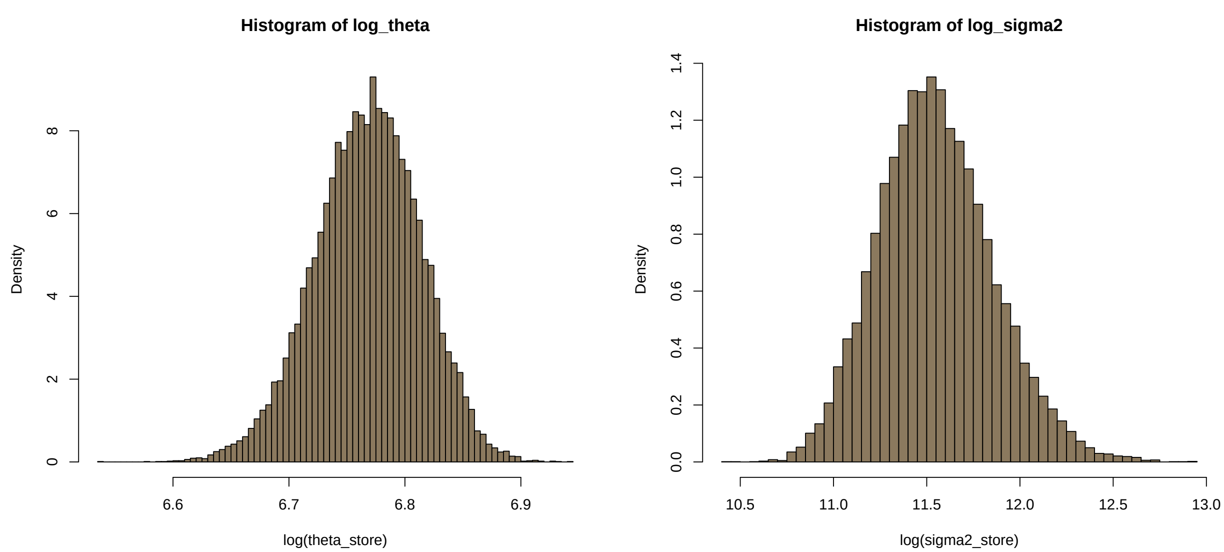
```r
max(theta_store)
```

```
## [1] 1036.16
```

```r
par(mfrow=c(1,2))
hist(theta_store, "FD", freq = F, main= "Histogram of theta",col = "navajowhite4")
hist(sigma2_store, "FD", freq = F, main= "Histogram of sigma2",col = "navajowhite4")
```




```
par(mfrow=c(1,2))
hist(log(theta_store), "FD", freq = F, main= "Histogram of log_theta",col = "navajowhite4")
hist(log(sigma2_store), "FD", freq = F, main= "Histogram of log_sigma2",col = "navajowhite4",)
```




```
quantile(log(theta_store), c(.05,0.95))
```

```
##       5%      95%
## 6.691481 6.840963
```

```
quantile(log(sigma2_store), c(.05,0.95))
```

```
##       5%      95%
## 11.06456 12.08825
```