

Teaching Bayes: The Exponential Lab

Rebecca C. Steorts

June 9, 2016

Task 1

Note that there are multiple observed values. Assuming that the observations are independent, we have

$$\begin{aligned} p(\mathbf{x}|a, b) &= \prod_{i=1}^n p(x_i|a, b) \\ &= \prod_{i=1}^n ab \exp(-abx_i) \\ &= (ab)^n \exp\left(-ab \sum_{i=1}^n x_i\right). \end{aligned}$$

The function is symmetric for a and b , so we only need to derive $p(a|\mathbf{x}, b)$. We find that

$$\begin{aligned} p(a|\mathbf{x}, b) &\propto_a p(a, b, \mathbf{x}) \\ &= p(\mathbf{x}|a, b)p(a, b) \\ &= (ab)^n \exp\left(-ab \sum_{i=1}^n x_i\right) (\exp(-a - b)I(a, b > 0)) \\ &= a^n b^n \exp\left(-ab \sum_{i=1}^n x_i\right) \exp(-a) \exp(-b)I(a, b > 0) \\ &\propto_a a^n \exp\left(-a \left(b \sum_{i=1}^n x_i + 1\right)\right) I(a > 0), \end{aligned}$$

which is the kernel of a Gamma distribution with shape $n + 1$ and rate $b \sum_{i=1}^n x_i + 1$.

Thus,

$$a|\mathbf{x}, b \sim \text{Gamma}\left(\text{shape} = n + 1, \text{rate} = b \sum_{i=1}^n x_i + 1\right).$$

By the symmetry of the distribution we also find

$$b|\mathbf{x}, a \sim \text{Gamma}\left(\text{shape} = n + 1, \text{rate} = a \sum_{i=1}^n x_i + 1\right).$$

This will always be a valid Gamma distribution because we have $a, b > 0$ and $x_i > 0$ for all i , which will ensure that the parameters of the Gamma are always positive.

Task 2

Now that we have our conditional distributions, we can program our Gibbs sampler.

```
# read in the data #
dataExp = read.csv("Assignment4Exponential.csv", header = FALSE)
# check number of observations in 30 #
nrow(dataExp)

## [1] 30

# set seed for reproducibility #
set.seed(666)

#####
### Function to do gibbs sampling for exponential ###
### inputs are number of simulations nsim          ###
### a & b values and data someData                 ###
#####
exp_gibbs <- function(nSims, a, b, someData) {
  # define n as number of data points #
  n = nrow(someData)
  # define x as sum of data points #
  x = sum(someData)
  # create matrix for results #
  results = matrix(NA, nrow = nSims, ncol = 2)
  # set starting points #
  results[1,] = c(a,b)

  for (i in 1:(nSims-1)) {
    results[i+1,1] = rgamma(1, shape = n+1, rate = 1 + x*results[i, 2])
    results[i+1,2] = rgamma(1, shape = n+1, rate = 1 + x*results[i+1, 1])
  }
  colnames(results) = c("conditionalA", "conditionalB")
  return(results)
}

condPost = exp_gibbs(10000, 2, 2, dataExp)
```

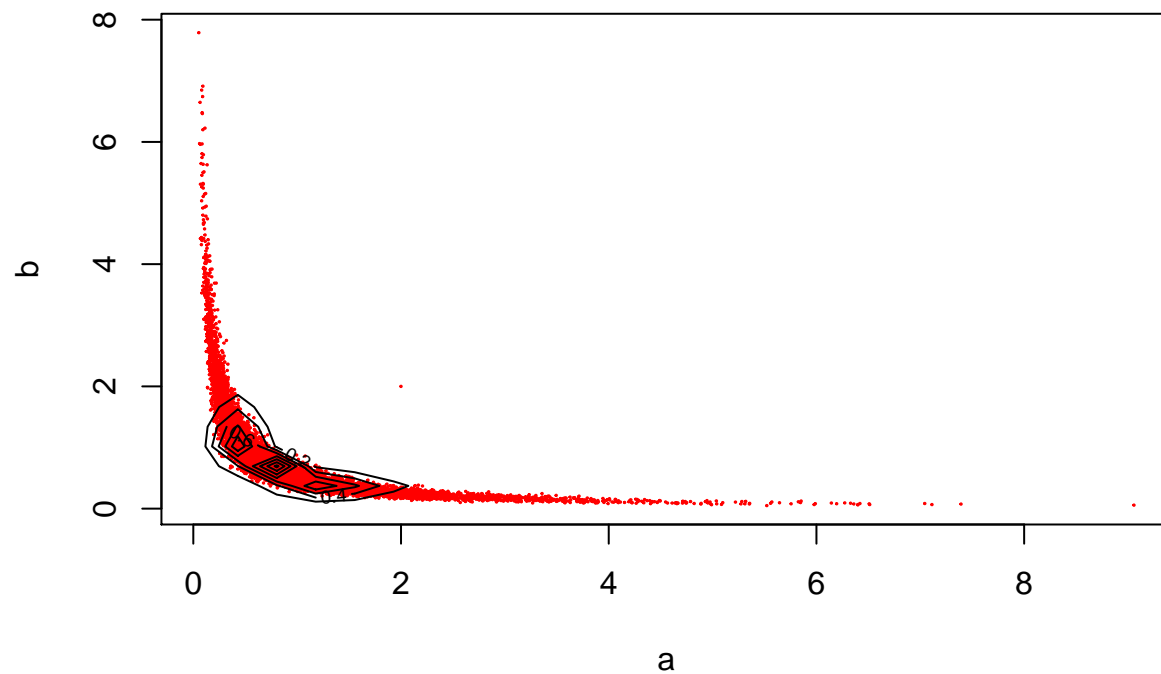
Task 3

We plot the joint draws with a density superimposed.

```
require("MASS")

## Loading required package: MASS

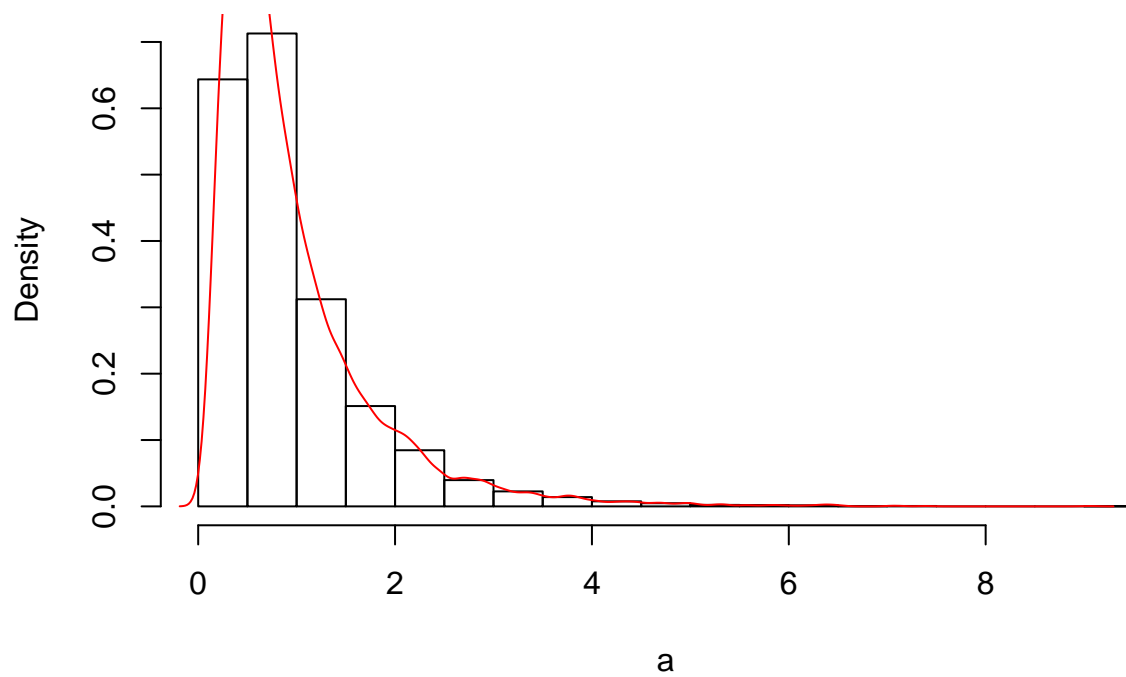
Z <- kde2d(condPost[,1],condPost[,2])
plot(condPost[,1],condPost[,2], pch = 19, cex = .1, col = 2,
      xlab = expression(a),
      ylab = expression(b))
contour(Z, add = TRUE)
```



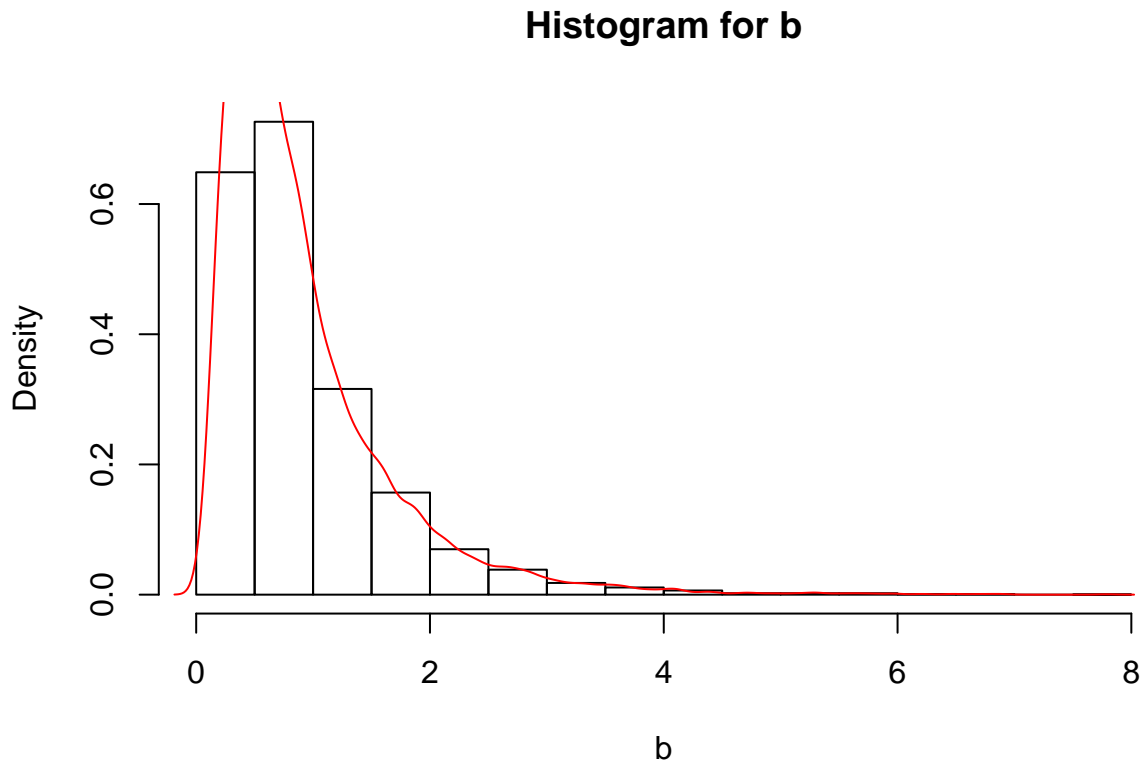
We now plot histograms of the marginal posteriors below.

```
hist(condPost[,1], prob = TRUE, main = "Histogram for a",
     xlab = expression(a))
lines(density(condPost[,1]), col = "red")
```

Histogram for a



```
hist(condPost[,2], prob = TRUE, main = "Histogram for b",
     xlab = expression(b))
lines(density(condPost[,2]), col = "red")
```



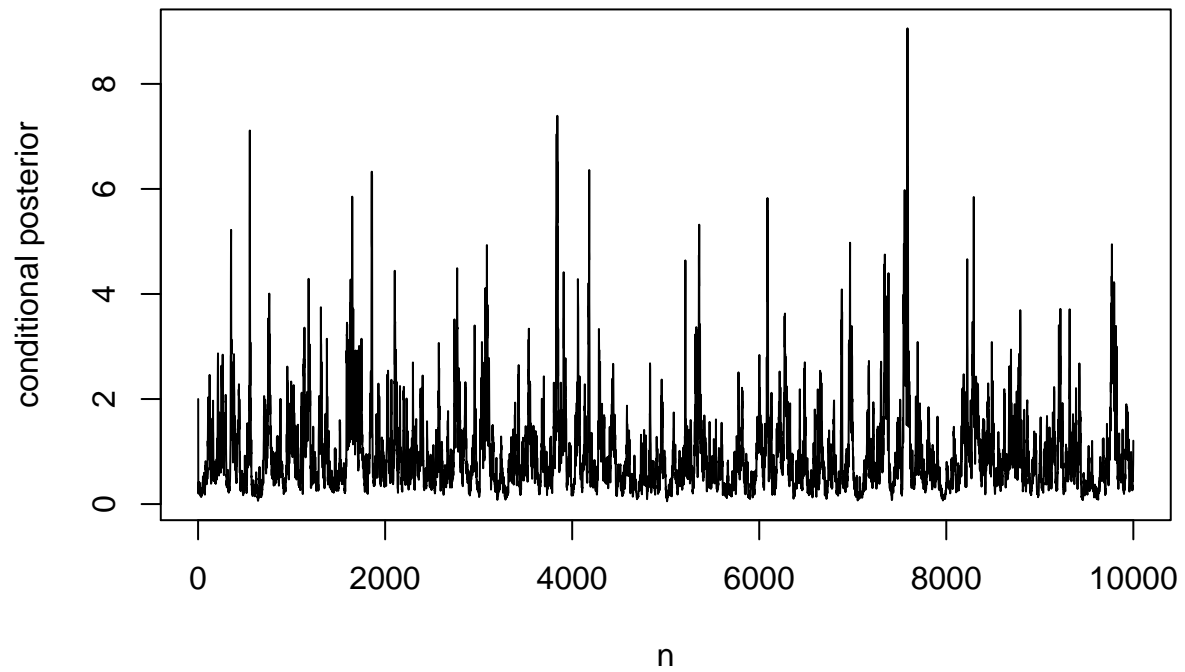
The points seem to be converging to a line of the form $ab = c$ for some constant c . This is reasonable, as the data could have been generated from a distribution with a fixed value of ab . This model is an example of a model that is not identifiable; it is not possible to uniquely determine a and b . We cannot predict to which values of a and b our Gibbs sampler will converge, only that the product of the means should be approximately constant.

Above, we see the histograms of our conditional and joint posterior densities.

Task 4

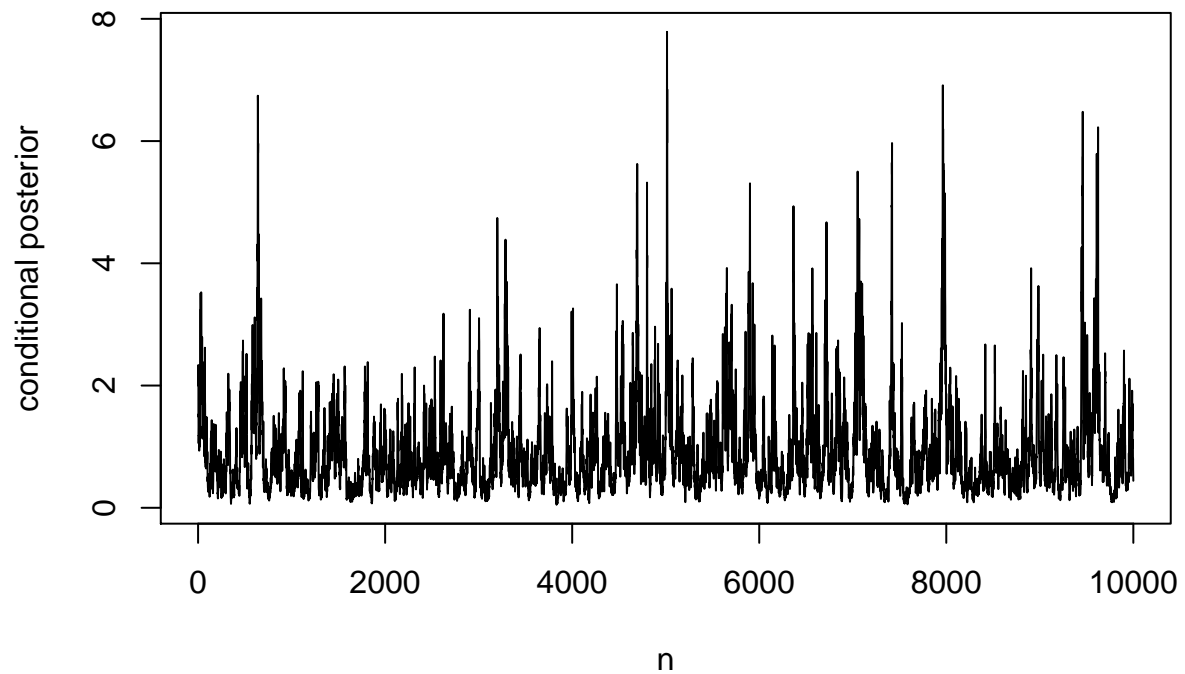
```
# plot trace plot for A #
plot(1:10000, condPost[,1], type = "l",
     main="Trace plot for conditional on A",
     ylab="conditional posterior", xlab="n")
```

Trace plot for conditional on A



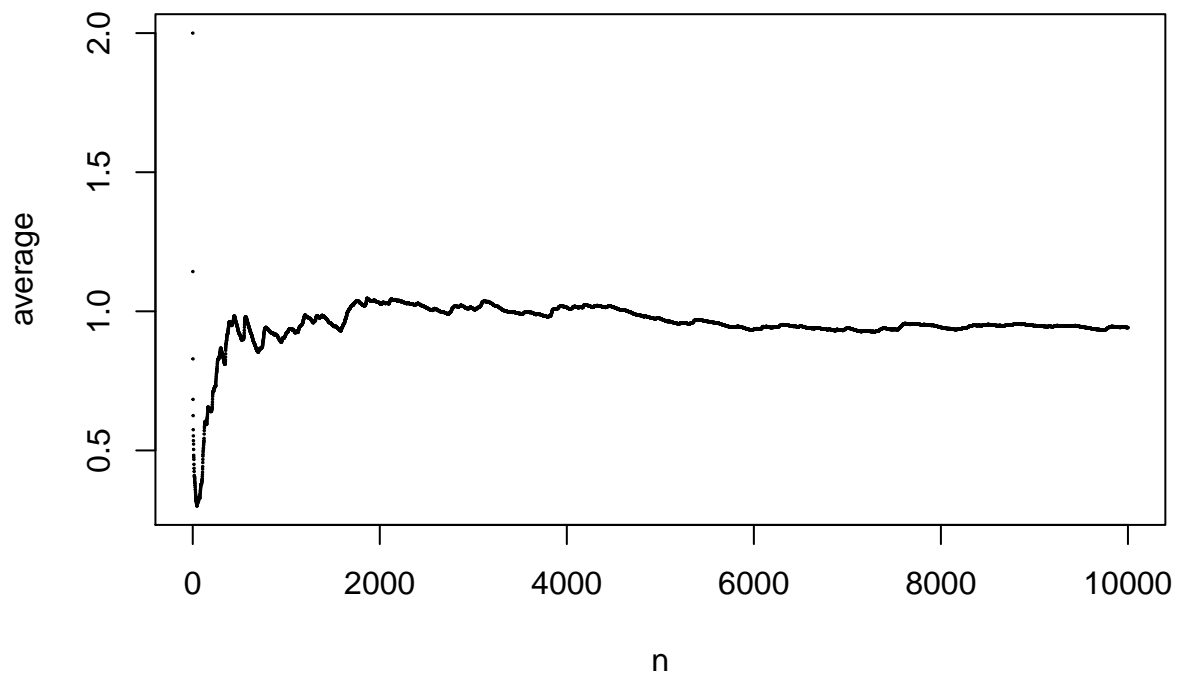
```
# plot trace plot for B #  
plot(1:10000, condPost[,2], type = "l",  
     main="Trace plot for conditional on B",  
     ylab="conditional posterior", xlab="n")
```

Trace plot for conditional on B



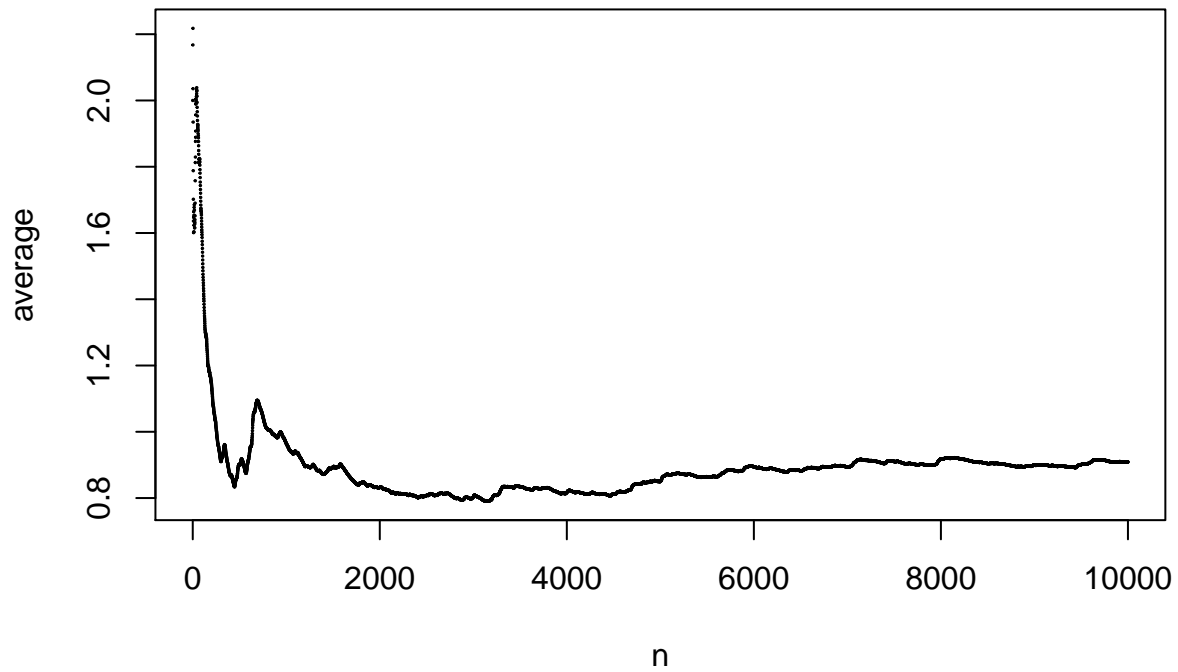
```
# plot moving average trace plot for A #
movAvgA = cumsum(condPost[,1])/(1:10000)
plot(1:length(movAvgA), movAvgA, cex = 0.1,
     main="Moving average for conditional on A",
     xlab="n", ylab="average")
```

Moving average for conditional on A



```
# plot moving average trace plot for B #
movAvgB = cumsum(condPost[,2])/(1:10000)
plot(1:length(movAvgB), movAvgB, cex = 0.1,
     main="Moving average for conditional on B",
     xlab="n", ylab="average")
```

Moving average for conditional on B



Examining our plots above we see that the trace plots are somewhat hard to interpret because of the proliferation of values. However, examining our moving average plots, after the initial period our average values appear to converge indicating the our Gibbs Sampler has converged.