Lab 2 – Beta-Binomial Distribution

Rebecca C. Steorts

January 2018

In class, you saw the Binomial-Beta model. We will now use this to solve a very real problem! Suppose I wish to determine whether the probability that a worker will fake an illness is truly 1%. Your task is to assist me! Tasks 1–3 will be completed in lab and tasks 3–5 should be completed in your weekly homework assignment. You should still upload task 3 even though this will be worked through in lab!

Task 1

Let's start by quickly deriving the Beta-Binomial distribution.

We assume that

$$X \mid \theta \sim \text{Binomial}(\theta)$$

,

$$\theta \sim \text{Beta}(a, b),$$

where a, b are assumed to be known parameters. What is the posterior distribution of $\theta \mid X$?

$$\begin{aligned}
p(\theta \mid X) &\propto p(X \mid \theta) p(\theta) &(1)\\
&\propto \theta^{x}(1-\theta)^{(n-x)} \times \theta^{(a-1)}(1-\theta)^{(b-1)} &(2)\\
&\propto \theta^{x+a-1}(1-\theta)^{(n-x+b-1)}. &(3)
\end{aligned}$$

This implies that

$$\theta \mid X \sim \text{Beta}(x+a, n-x+b).$$

Task 2

Simulate some data using the rbinom function of size $n = 100$ and probability equal to 1%. Remember to set.seed(123) so that you can replicate your results.

The data can be simulated as follows:

```r
# set a seed
set.seed(123)
# create the observed data
obs.data <- rbinom(n = 100, size = 1, prob = 0.01)
# inspect the observed data
head(obs.data)
```

```
## [1] 0 0 0 0 0 0
```

```r
tail(obs.data)
```

```
## [1] 0 0 0 0 0 0
```

```r
length(obs.data)
```

```
## [1] 100
```

Task 3

Write a function that takes as its inputs that data you simulated (or any data of the same type) and a sequence of θ values of length 1000 and produces Likelihood values based on the Binomial Likelihood. Plot your sequence and its corresponding Likelihood function.

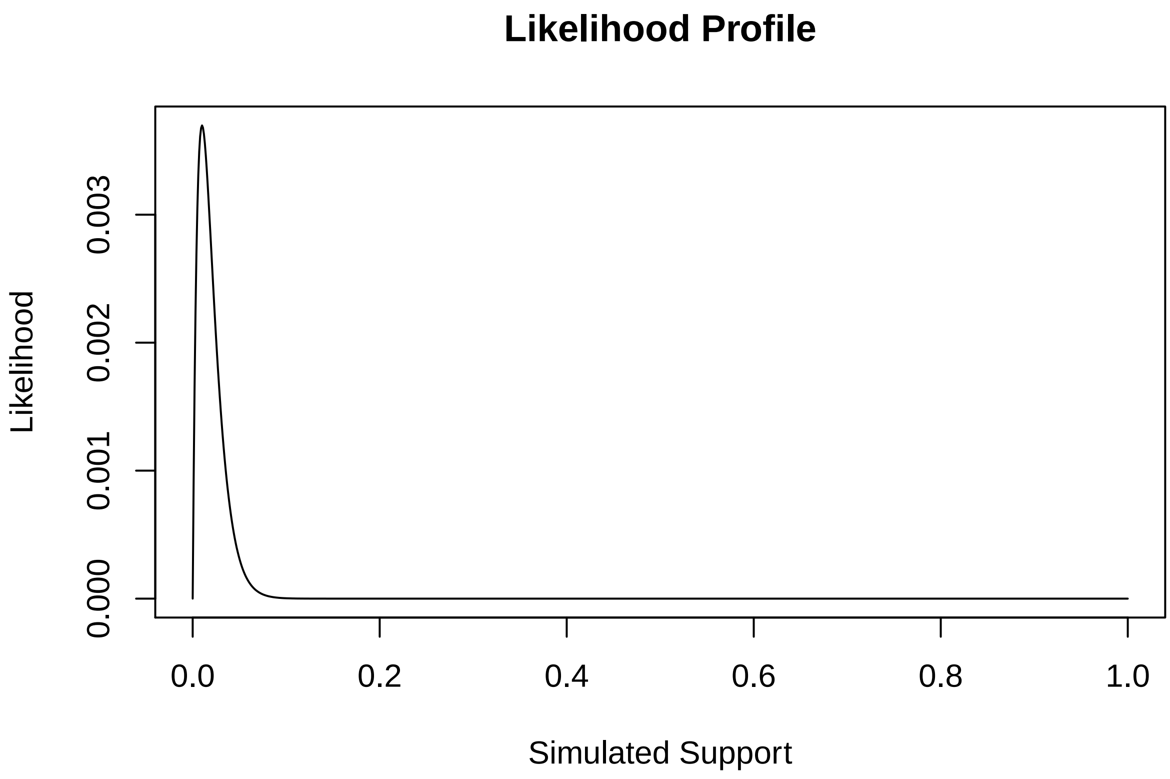
The likelihood function is given below. Since this is a probability and is only valid over the interval from $[0, 1]$ we generate a sequence over that interval of length 1000.

You have a rough sketch of what you should do for this part of the assignment. Try this out in lab on your own.

```r
### Bernoulli LH Function ###
# Input: obs.data, theta
# Output: bernoulli likelihood

myBernLH <- function(obs.data, theta) {
  N <- length(obs.data)
  x <- sum(obs.data)
  LH <- (theta^x) * ((1-theta)^(N-x))
  return(LH)
}

### Plot LH for a grid of theta values ###
# Create the grid #
theta.sim <- seq(from=0, to=1, length.out=1000)
# Store the LH values
sim.LH <- myBernLH(obs.data, theta = theta.sim)
# Create the Plot
plot(theta.sim, sim.LH,
     type='l',
     main='Likelihood Profile',
     xlab='Simulated Support',
     ylab='Likelihood')
```



Task 4 (To be completed for homework)

Write a function that takes as its inputs prior parameters a and b for the Beta-Bernoulli model and the observed data, and produces the posterior parameters you need for the model. **Generate and print** the posterior parameters for a non-informative prior i.e. (a,b) = (1,1) and for an informative case (a,b) = (3,1)}.

```
posterior <- function(a, b, obs.data){
  N <- length(obs.data)
  x <- sum(obs.data)
  return(list('a' = a + x, 'b' = b + N - x))
}

noninformative <- posterior(a=1, b=1, obs.data)
informative <- posterior(a=3, b=1, obs.data)

print(noninformative)
```

```
## $a
## [1] 2
##
## $b
## [1] 100
```

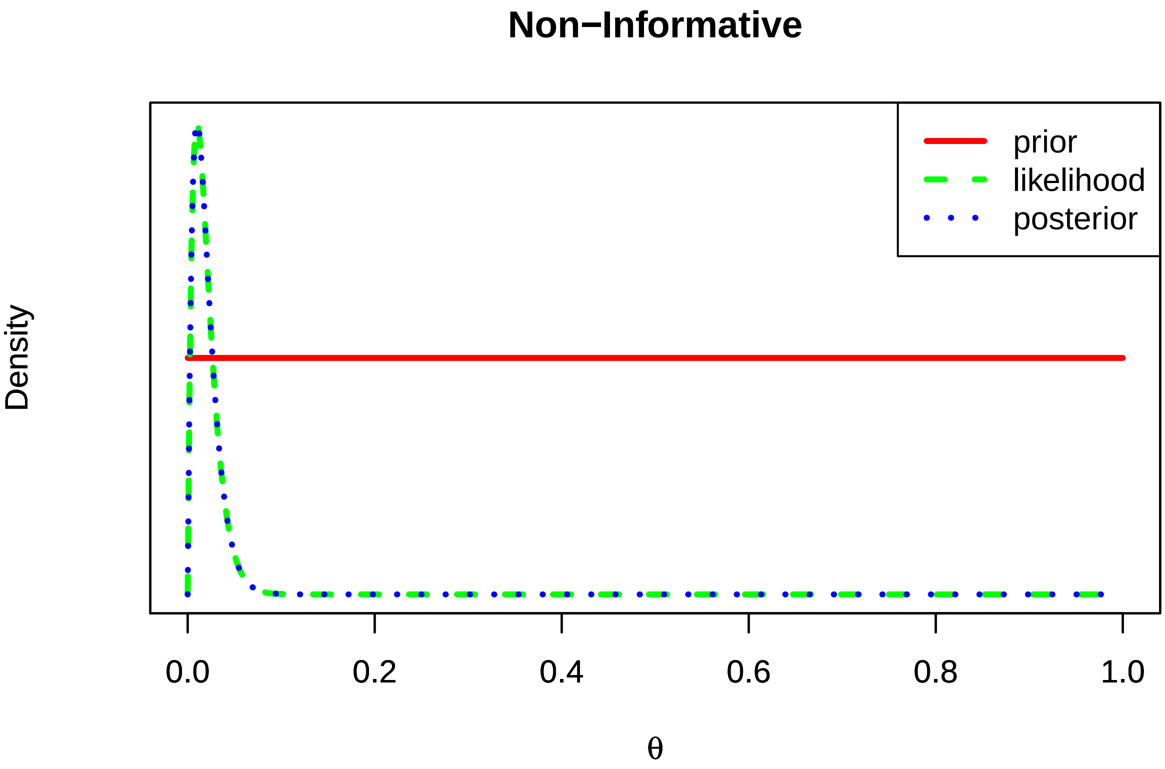
```
print(informative)
```

```
## $a
## [1] 4
##
## $b
## [1] 100
```

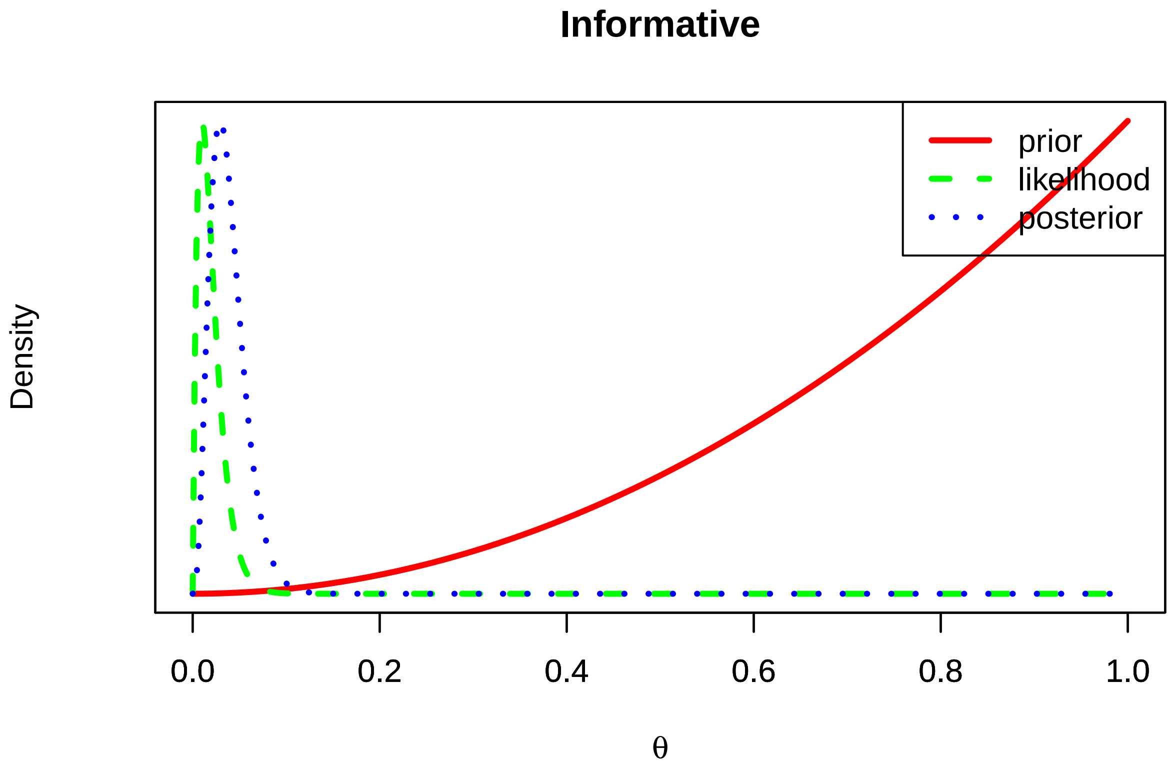
Task 5 (To be completed for homework)

Create two plots, one for the informative and one for the non-informative case to show the posterior distribution and superimpose the prior distributions on each along with the likelihood. What do you see? Remember to turn the y-axis ticks off since superimposing may make the scale non-sense.

```r
th = seq(0, 1, length=500)
like = dbeta(th, sum(obs.data)+1, length(obs.data)-sum(obs.data)+1)
a = 1
b = 1
prior = dbeta(th, a, b)
post = dbeta(th, sum(obs.data)+a, length(obs.data)-sum(obs.data)+b)
plot(th, prior, type='l', ylab='Density', lty = 1, lwd = 3, xlab = expression(theta),col='red', yaxt='n
par(new=TRUE)
plot(th, like, type='l', ylab='Density', lty = 2, lwd = 3, xlab = expression(theta),col='green', yaxt='
par(new=TRUE)
plot(th, post, type='l', ylab='Density', lty = 3, lwd = 3, xlab = expression(theta),col='blue', yaxt='n
legend('topright', lty=c(1,2,3), lwd=c(3,3,3), col=c('red','green','blue'), legend=c('prior', 'likeliho
```



```r
a = 3
b = 1
prior = dbeta(th, a, b)
post = dbeta(th, sum(obs.data)+a, length(obs.data)-sum(obs.data)+b)
plot(th, prior, type='l', ylab='', lty = 1, lwd = 3, xlab = '', col='red', yaxt='n')
par(new=TRUE)
plot(th, like, type='l', ylab='', lty = 2, lwd = 3, xlab = '', col='green', yaxt='n')
par(new=TRUE)
plot(th, post, type='l', ylab='Density', lty = 3, lwd = 3, xlab = expression(theta), col='blue', yaxt='
legend('topright', lty=c(1,2,3), lwd=c(3,3,3), col=c('red','green','blue'), legend=c('prior', 'likeliho
```



We see for the non-informative prior (the beta(1,1) distribution that is essentially a uniform distribution), the posterior and prior distributions are the same. For the informative prior that weighs towards a higher theta, the posterior is averaged between the likelihood and prior, but with much higher weightig for the likelihood and is nearly equivalent to it. The posterior for the non-informative prior returns a distribution to the right of the distribution for the informative prior, due to the required averaging. The importance of the likelihood is higher for a non-informative prior compared to an informative prior.