# XIAO_Lin_Solutions_lab7

*Lin Xiao*

*October 27, 2015*

In class last week, you saw the Beta-Bernoulli model. We will now use this to solve a very real problem! Assume that I as a statistician need to determine whether the probability that a student will fake an illness is truly 1%. Your task is to assist me!

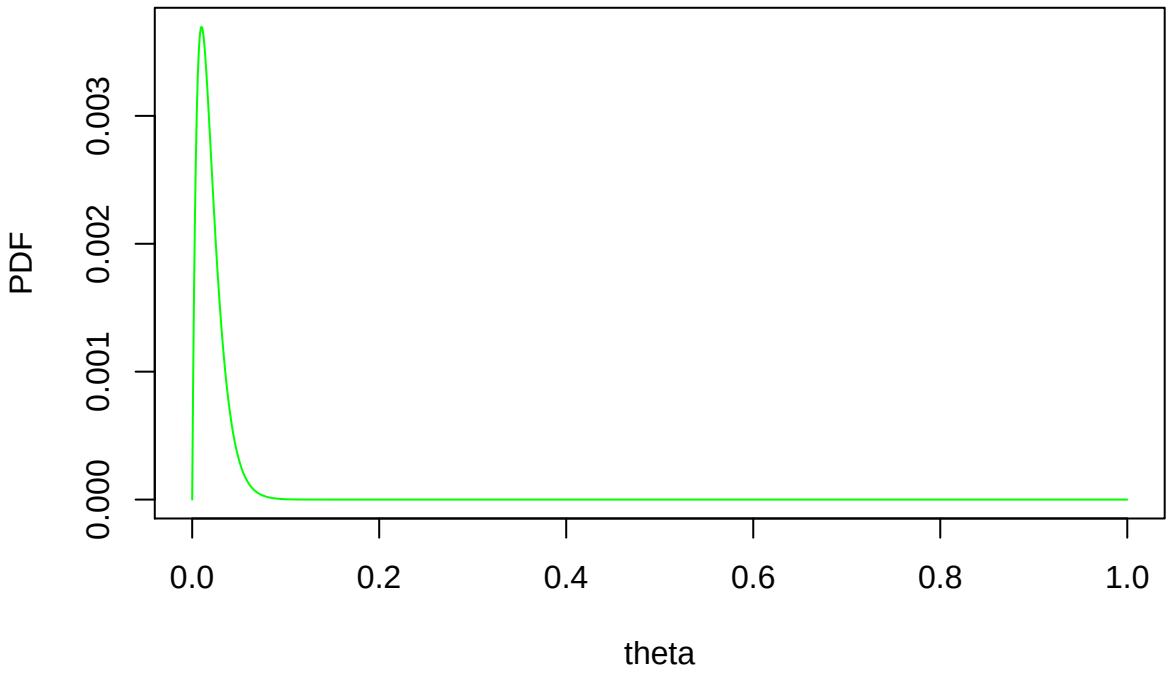
1. simulate some data using the rbinom function of size n = 100 and probability equal to 1%. Remember to set.seed(123) so that I can replicate your results.

```
set.seed(123)
dat <- rbinom(100, 1, 0.01)

# To see how many 1's
sum(dat)
```

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

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

```
theta <- seq(0, 1, length.out = 1000)

# Calculate-likelihood function
# Input simulated data(data), and a sequence of theta
# Output is the likelihood values based on Binomial Likelihood,
# simulated data and theta values
fun1 <- function(data, theta_seq){
  lhd <- c()
  for(i in 1:1000){
    lhd[i] <- (theta_seq[i])^sum(data)*(1-theta_seq[i])^(length(data)-sum(data))
  }
  return(lhd)
}

# Plot the sequence and its corresponding Likelihood function
plot(theta, fun1(dat, theta), type = "l", col = "green", ylab = "PDF")
```




3. 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 the posterior parameters for a non-informative prior i.e. (a,b) = (1,1) and for an informative case (a,b) = (3,1)

```
# Input simulated data and the prior parameters
# Output is a combine of posterior parameters
fun2 <- function(data, a, b){
  return(c("post_a" = a+sum(data), "post_b" = b+length(data)-sum(data)))
}

# Posterior parameters for (a,b)=(1,1)
fun2(dat,1,1)
```

```
## post_a post_b
##      2    100
```

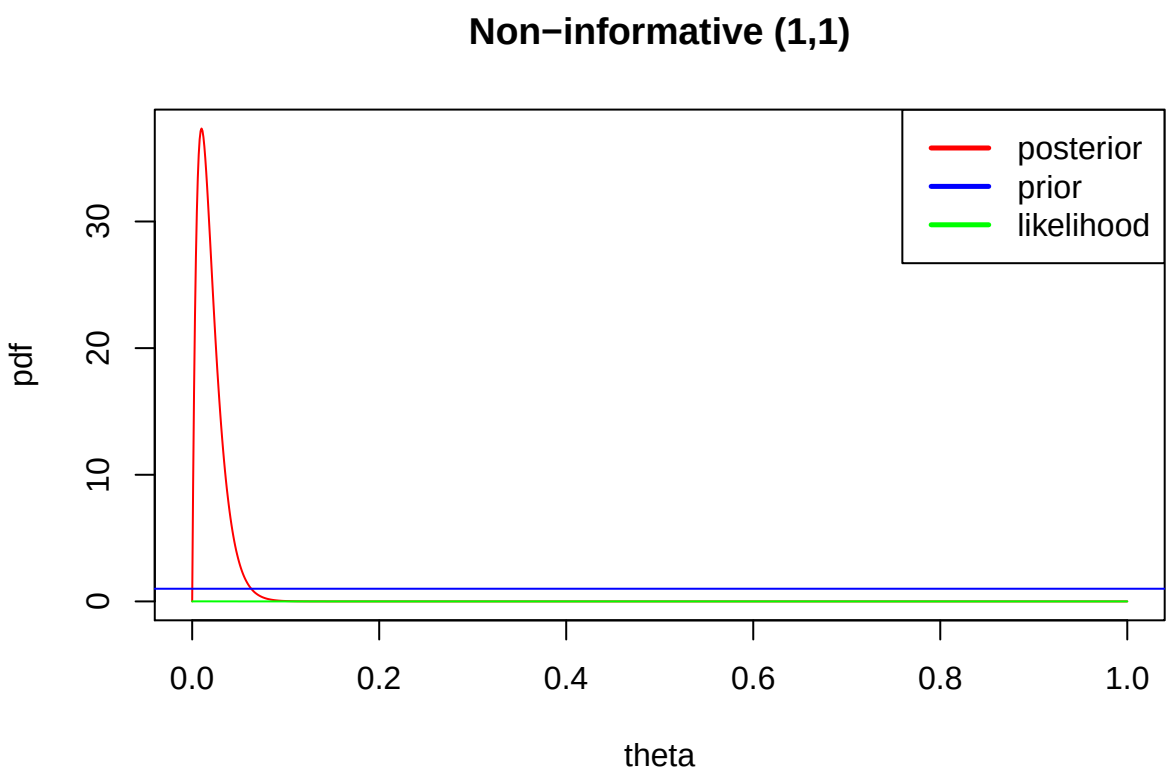
```
# Posterior parameters for (a,b)=(3,1)
fun2(dat,3,1)
```

```
## post_a post_b
##      4    100
```

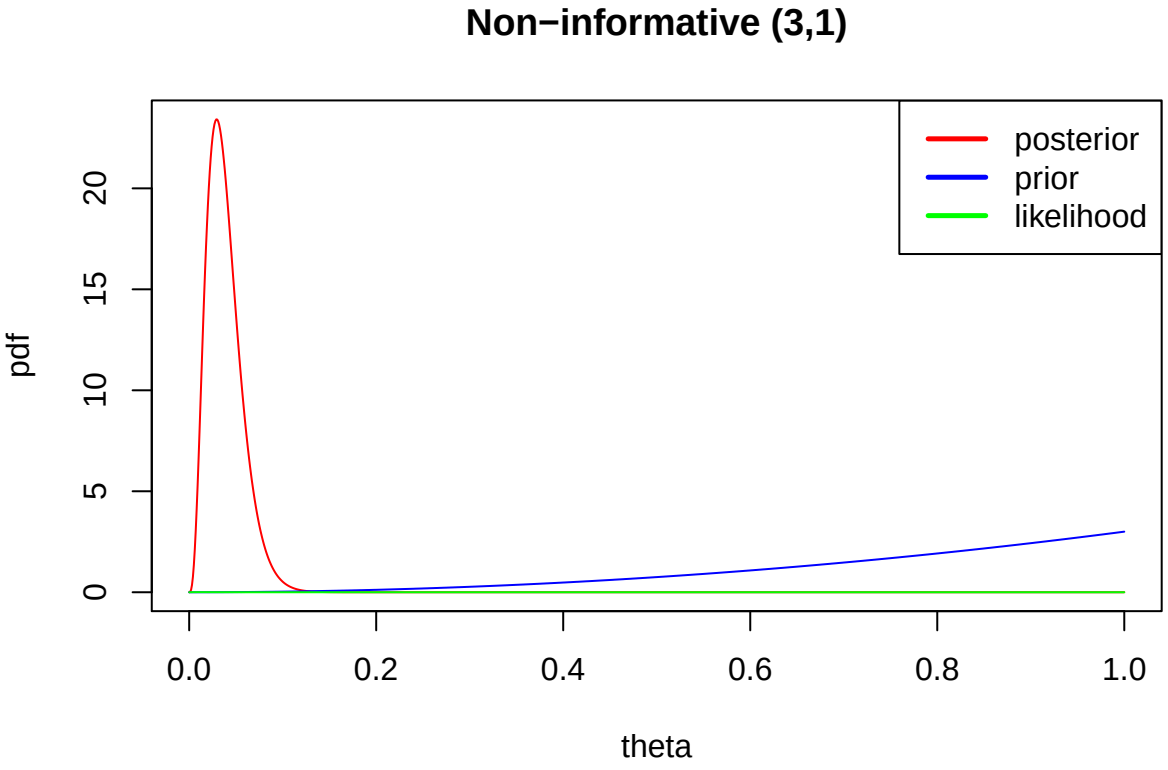
4. Create two plots, one for the informative and one for the non-informative case to show the posterior distribution and superimpose the prior distributions on each along with the likelihood. What do you see?

Here I first draw the original plots, but because of the scale we can't see the shape clearly, so by using par(new=T) and removing overlapping the axes and titles we can see their shapes clearly.

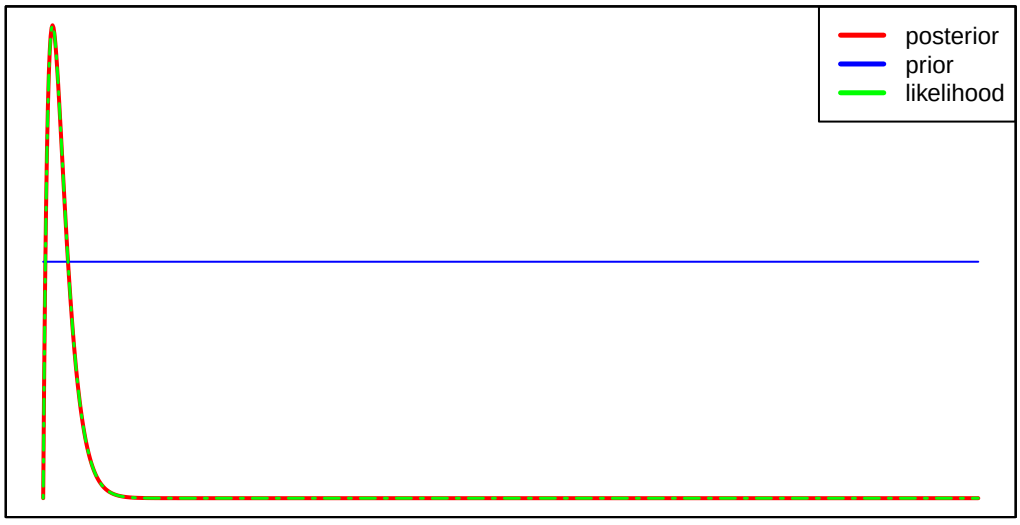
```r
# These plots are all having the same plotting order: posterior, prior and likelihood

# Orignial plots
par(new = F, yaxt="s", xaxt="s", ann=T) #Reset par()
plot(theta, dbeta(theta, fun2(dat,1,1)[[1]], fun2(dat,1,1)[[2]]),type="l",
     ylab="pdf", col="red", main="Non-informative (1,1)")
abline(h=1, col="blue")
lines(theta, fun1(dat, theta), col="green")
legend("topright", c("posterior","prior","likelihood"),lty=c(1,1,1),
       lwd=c(2.5,2.5,2.5), col=c("red", "blue", "green"))
```



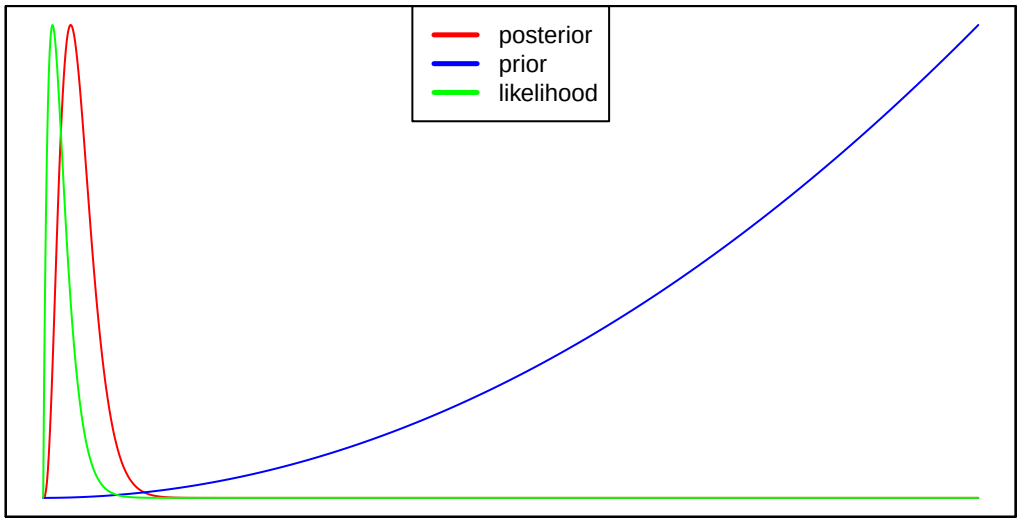

```r
plot(theta, dbeta(theta, fun2(dat,3,1)[[1]], fun2(dat,3,1)[[2]]),type="l",
     ylab="pdf", col="red", main="Non-informative (3,1)")
lines(theta,dbeta(theta, 3, 1),col="blue")
lines(theta, fun1(dat, theta), col="green")
legend("topright", c("posterior","prior","likelihood"),lty=c(1,1,1),
       lwd=c(2.5,2.5,2.5), col=c("red", "blue", "green"))
```




```
# Rescale the lines so we can see their shapes under
# their scales but plotting in the same graph
par(yaxt="n", xaxt="n", ann=FALSE)
plot(theta, dbeta(theta,fun2(dat,1,1)[[1]],fun2(dat,1,1)[[2]]),
     type="l", col="red",lty=1, lwd=2)
par(new = T)
plot(theta, dbeta(theta,1,1),type="l",col="blue")
par(new = T)
plot(fun1(dat, theta), col="green",type="l",lty=6, lwd=1.5)
legend("topright", c("posterior","prior","likelihood"),lty=c(1,1,1),
       lwd=c(2.5,2.5,2.5), col=c("red", "blue", "green"),cex=0.75)
```



```r
plot(theta, dbeta(theta,fun2(dat,3,1)[[1]],fun2(dat,3,1)[[2]]),type="l", col="red")
par(new = T)
plot(theta, dbeta(theta,3,1),type="l",col="blue")
par(new = T)
plot(fun1(dat, theta), col="green",type="l")
legend("top", c("posterior","prior","likelihood"),lty=c(1,1,1),
       lwd=c(2.5,2.5,2.5), col=c("red", "blue", "green"), cex=0.75)
```




First we see that the first two plots can't provide us much information on the relationship among posterior, prior and likelihood. Then we use par(new=T) to draw the plots in the same graph. We can see from the plots that each posterior looks like a weighted combination of prior and likelihhood. For example, for the non-informative prior, it's just the line y=1 with support on [0,1], it doesn't affect the shape of likelihood so we have overlapping lines.

5. Based on the informative case, generate a 95% credible interval and a 95% confidence interval for your parameter of interest, and use xtable to output these. What is the problem?

```r
library(xtable)
# 95% Credible Interval
Cred_I <- qbeta(c(0.025,0.975),4,100)
# 95% Confidence Interval
Conf_I <- c(0.01+sd(dat)*qnorm(0.025),0.01+sd(dat)*qnorm(0.975))
dat_CI <- data.frame(Cred_I,Conf_I)
colnames(dat_CI) <- c("Credible Interval", "Confidence Interval")
tb <- xtable(dat_CI)
digits(tb)[c(2,3)] <- c(6,6)
print(tb, comment=F)
```

| | Credible Interval | Confidence Interval |
|---|---|---|
| 1 | 0.010681 | -0.185996 |
| 2 | 0.082765 | 0.205996 |

The confidence interval of theta is out of bounds, with -0.1859964<0, which makes no sense.

6. Based on the data you simulated, do you conclude that the true value higher or lower than 1%?

Since the mode of beta(a=4, b=100) is

$$\frac{a-1}{a+b-2} = \frac{3}{102} > 0.01$$

And 0.01 is lower than the lower bound of credible interval, we can conclude that the true value is higher than 1%.