U20 Math 559 Bayesian Statistics Homework 1

Sami Cheong, Ph.D

Due: 2/20/2021

Instruction: Please type or write your answers clearly and show your work. You are encouraged to use the Rmarkdown version of this assignment as a template to submit your work. Unless stated otherwise, all programming references in the assignment will be in `R`, and the predefined `R` functions used for the problems can all be found on our Canvas site under `DBDA2Eprograms.zip`. For this assignment, problems roughly covers content from the first 3 lectures.

Problem 1

a) Suppose we have a four-sided die from a board game. On a tetrahedral die, each face is an equilateral triangle. When you roll the die, it lands with one face down and the other three faces visible as a three-sided pyramid. The faces are numbered 1-4, with the value of the bottom face printed (as clustered dots) at the bottom edges of all three visible faces. Denote the value of the bottom face as x. Consider the following three mathematical descriptions of the probabilities of x. *Model A:* $p(x) = 1/4$, *Model B:* $p(x) = x/10$, and *Model C:* $p(x) = 12/(25x)$. Describe what kind of bias (or lack of bias) is expressed by each model as values of x changes.

b) Suppose when we rolled the die 100 times we found these results: #1's = 48, #2's = 24, #3's = 16, #4's = 12. Now which model seems most likely?

Problem 2

Modify the coin flipping program in `RunningProportion.R` to simulate a biased coin that has p(H) = 0.8. Change the height of the reference line in the plot to match p(H). Comment your code. Hint: Read the help for the sample command.

```
source("../DBDA2E-utilities.R")
```

```
##
## *********************************************************************
## Kruschke, J. K. (2015). Doing Bayesian Data Analysis, Second Edition:
## A Tutorial with R, JAGS, and Stan. Academic Press / Elsevier.
## *********************************************************************
```

```
# Specify the total number of flips
n_flips <- 500
set.seed(2) # Set to constant random seed
# Generate a random sample
pheads = 0.8
flipsequence = sample(x=c(0,1), # Sample from coin (heads=1, tails=0)
                      prob=c(1-pheads,pheads), # Define a fair coin
                      size=n_flips, # Number of flips
                      replace=TRUE ) # Sample with replacement


# Compute the running proportion of heads
r = cumsum( flipsequence )
```

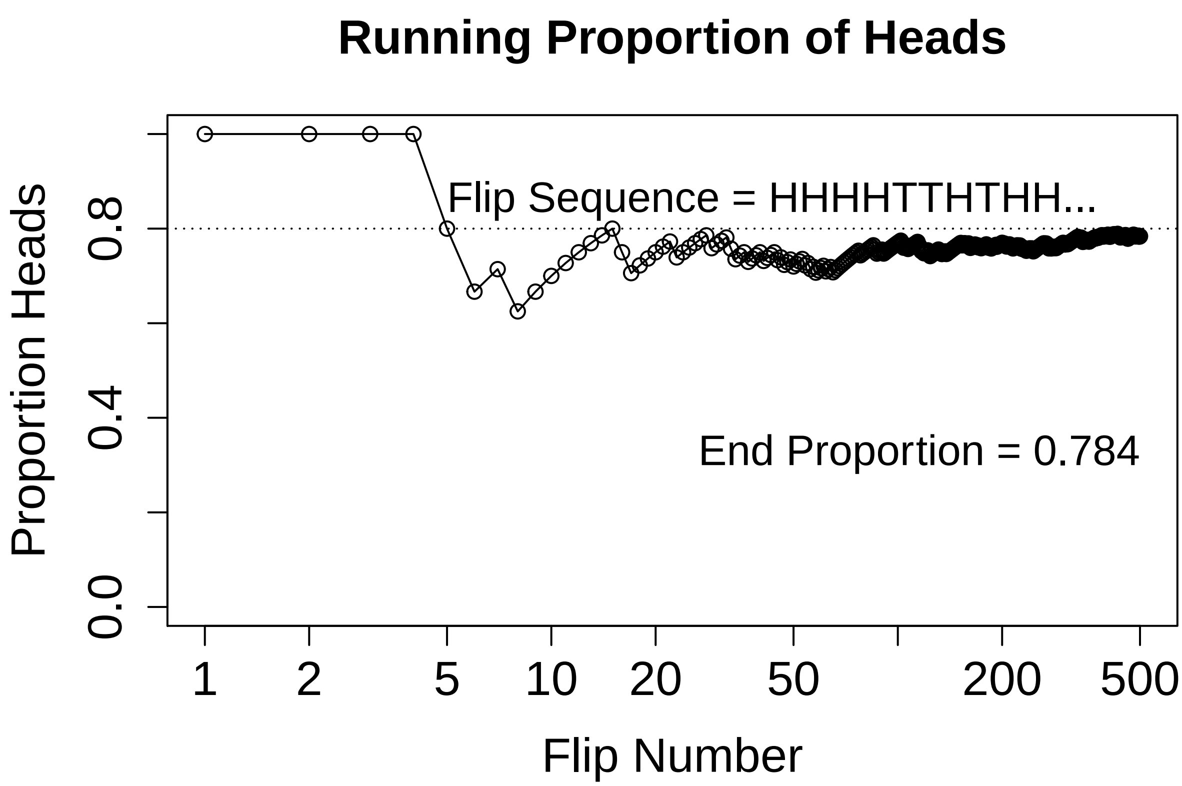
```r
n = 1:n_flips
runprop = r / n

# Graph the running proportion
plot( n , runprop , type="o" , log="x" ,
      xlim=c(1,n_flips) , ylim=c(0.0,1.0) , cex.axis=1.5 ,
      xlab="Flip Number" , ylab="Proportion Heads" , cex.lab=1.5 ,
      main="Running Proportion of Heads" , cex.main=1.5 )


# Plot a dotted horizontal line at y=pheads, just as a reference line:
abline( h = pheads, lty=3 )

# Display the beginning of the flip sequence. These string and character
# manipulations may seem mysterious, but you can de-mystify by unpacking
# the commands starting with the innermost parentheses or brackets and
# moving to the outermost.

flipletters = paste( c("T","H")[ flipsequence[ 1:10 ] + 1 ] , collapse="" )
displaystring = paste( "Flip Sequence = " , flipletters , "..." , sep="" )
text( 5 , .9 , displaystring , adj=c(0,1) , cex=1.3 )
# Display the relative frequency at the end of the sequence.
text(n_flips, .3 , paste("End Proportion =",runprop[n_flips]) , adj=c(1,0) , cex=1.3 )
```



Problem 3

School children were surveyed regarding their favorite foods. Of the total sample, 20% were 1st graders, 20% were 6th graders, and 60% were 11th graders. For each grade, the following table shows the proportion of respondents that chose each of three foods as their favorite:

	Ice cream	Fruit	French fries
1^{st} graders	0.3	0.6	0.1
6^{th} graders	0.6	0.3	0.1
11^{th} graders	0.3	0.1	0.6

From that information, construct a table of joint probabilities of grade and favorite food. Also, say whether grade and favorite food are independent or not, and how you ascertained the answer. Hint: You are given $p(\text{grade})$ and $p(\text{food}|\text{grade})$. You need to determine $p(\text{grade,food})$.

Answer

As the hint indicates, the table specifies $p(\text{food}|\text{grade})$, and the text provides $p(\text{grade})$. Next, remember that

$p(\text{food}|\text{grade}) = p(\text{food}, \text{grade})/p(\text{grade})$, which implies that $p(\text{food}, \text{grade}) = p(\text{food}|\text{grade}) * p(\text{grade})$

As a result, the table of joint probabilities can be calculated as:

	Ice cream	Fruit	French fries
1^{st} graders	0.3*0.2 = 0.06	0.6*0.2 = 0.12	0.1*0.2 = 0.02
6^{th} graders	0.6*0.2 = 0.12	0.3*0.2 = 0.06	0.1*0.2 = 0.02
11^{th} graders	0.3*0.6 = 0.18	0.1*0.6 = 0.06	0.6*0.6 = 0.36

Grade and food are not independent. This can be proven by exhibiting any cell for which p(food,grade) does not equal $p(\text{food}) * p(\text{grade})$. Consider Ice cream in 1st grade: $p(\text{Ice cream}, 1^{st}\text{grade})$ is 0.06, while $p(\text{Ice cream}) * p(1^{st}\text{grade})$ = 0.36 * 0.20 = 0.072.

Problem 4

For this exercise, use the R function `BernBeta.R`. (Don't forget to source the function before calling it.) Notice that the function returns the posterior beta values each time it is called, so you can use the returned values as the prior values for the next function call.

a) Start with a prior distribution that expresses some uncertainty that a coin is fair: $beta(\theta|4, 4)$. Flip the coin once; suppose we get a head. What is the posterior distribution?

Answer

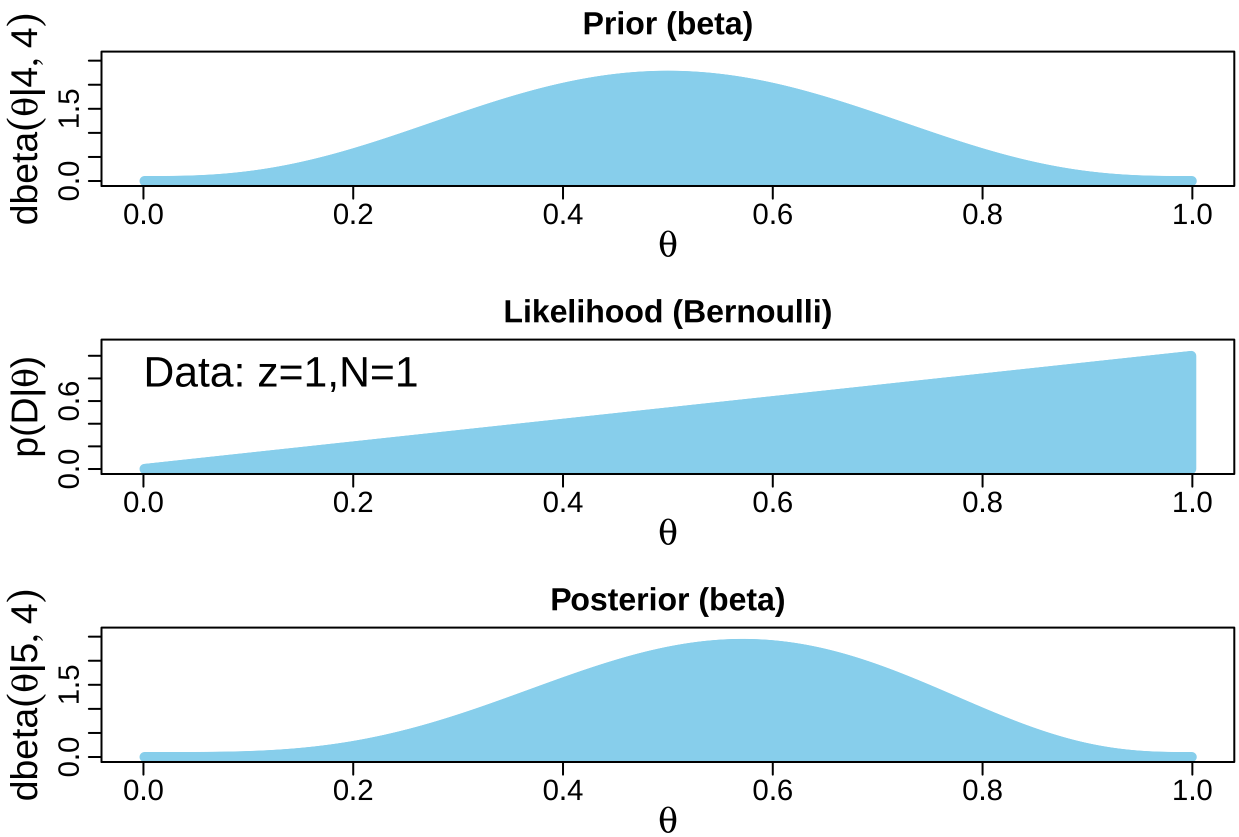
The posterior is dbeta(theta|5,4); see the label on the y-axis of the posterior distribution or type show(post) or just use the simple formula for updating a beta distribution.

```
source("../DBDA2E-utilities.R")  # Load definitions of graphics functions etc.
```

```
##
## *********************************************************************
## Kruschke, J. K. (2015). Doing Bayesian Data Analysis, Second Edition:
## A Tutorial with R, JAGS, and Stan. Academic Press / Elsevier.
## *********************************************************************
```

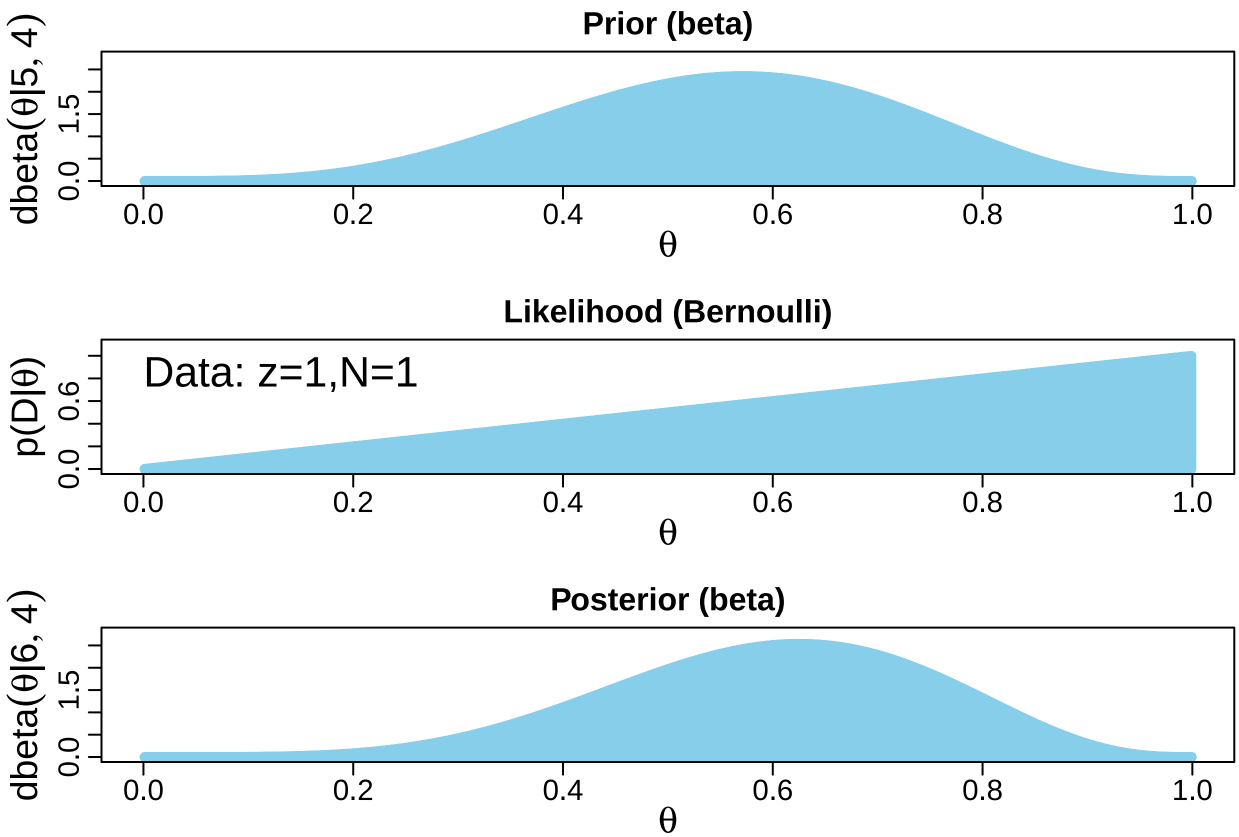
```
source("../BernBeta.R")          # Load the definition of the BernBeta function

post = BernBeta( priorBetaAB=c(4,4) , Data=c(1) )
```



b) Use the posterior from the previous flip as the prior for the next flip. Suppose we flip again and get a head. Now what is the new posterior? (Hint: If you type `post = BernBeta( c(4,4) , c(1) )` for the first part, then you can type `post = BernBeta( post , c(1) )` for the next part.)

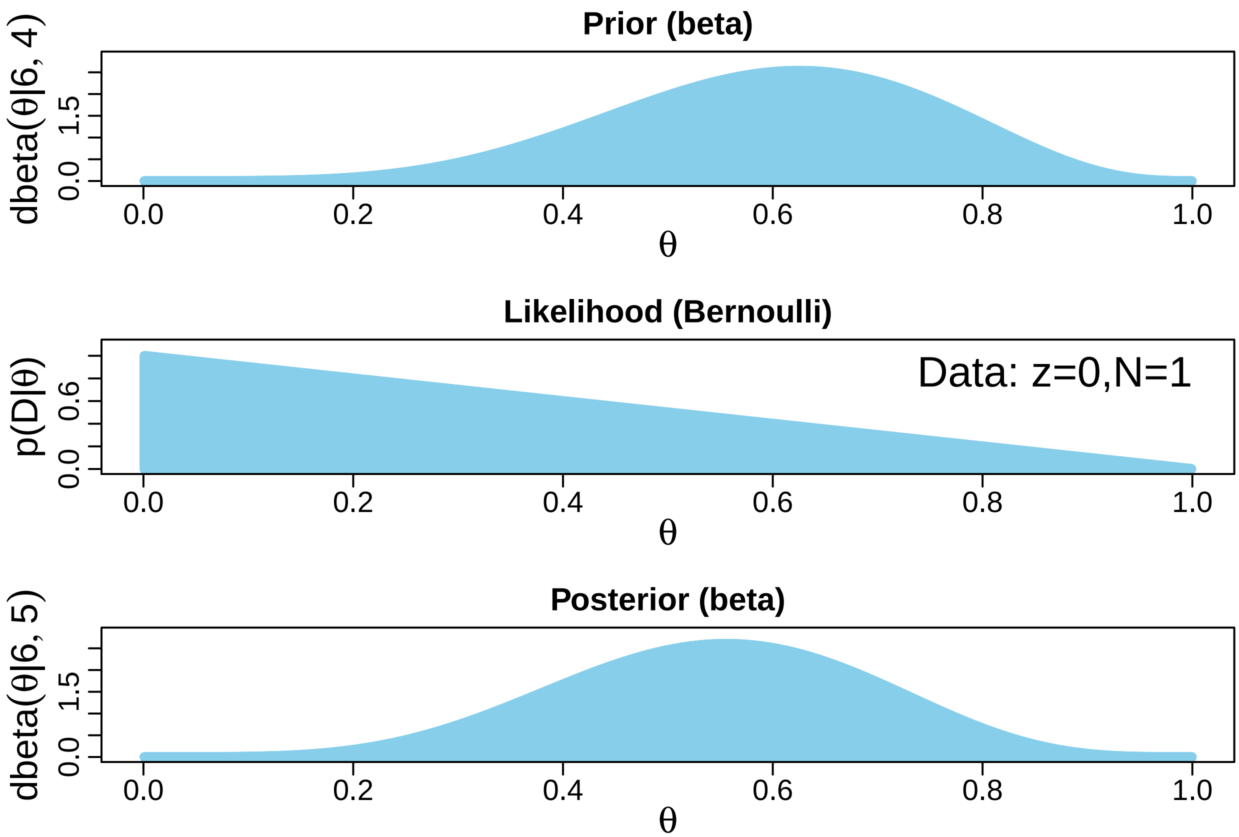
```
post = BernBeta( priorBetaAB=post , Data=c(1) )
```



The posterior is `dbeta(theta|6,4)`.

c) Using that posterior as the prior for the next flip, flip a third time and get a tail. Now what is the new posterior? (Hint: Type post = BernBeta(post , c(0)).)

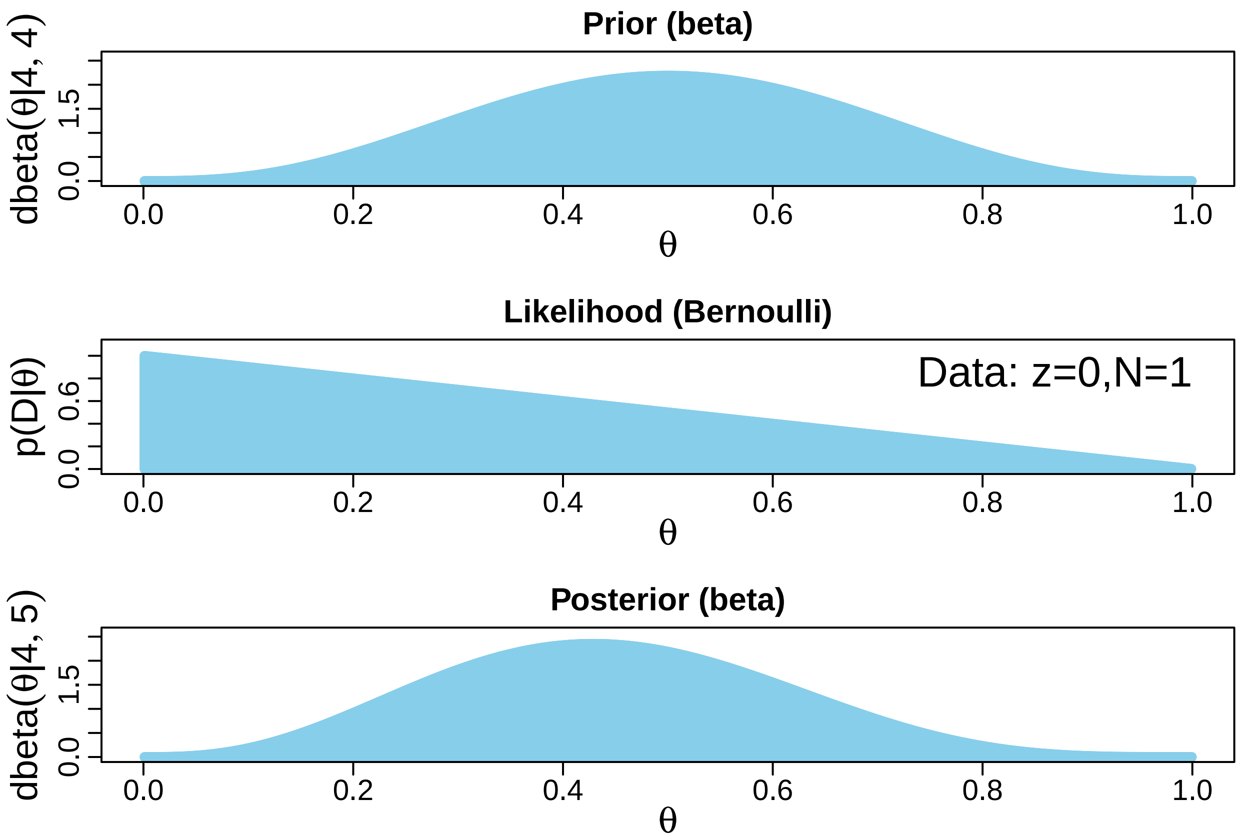
```
post = BernBeta( priorBetaAB=post , Data=c(0) )
```



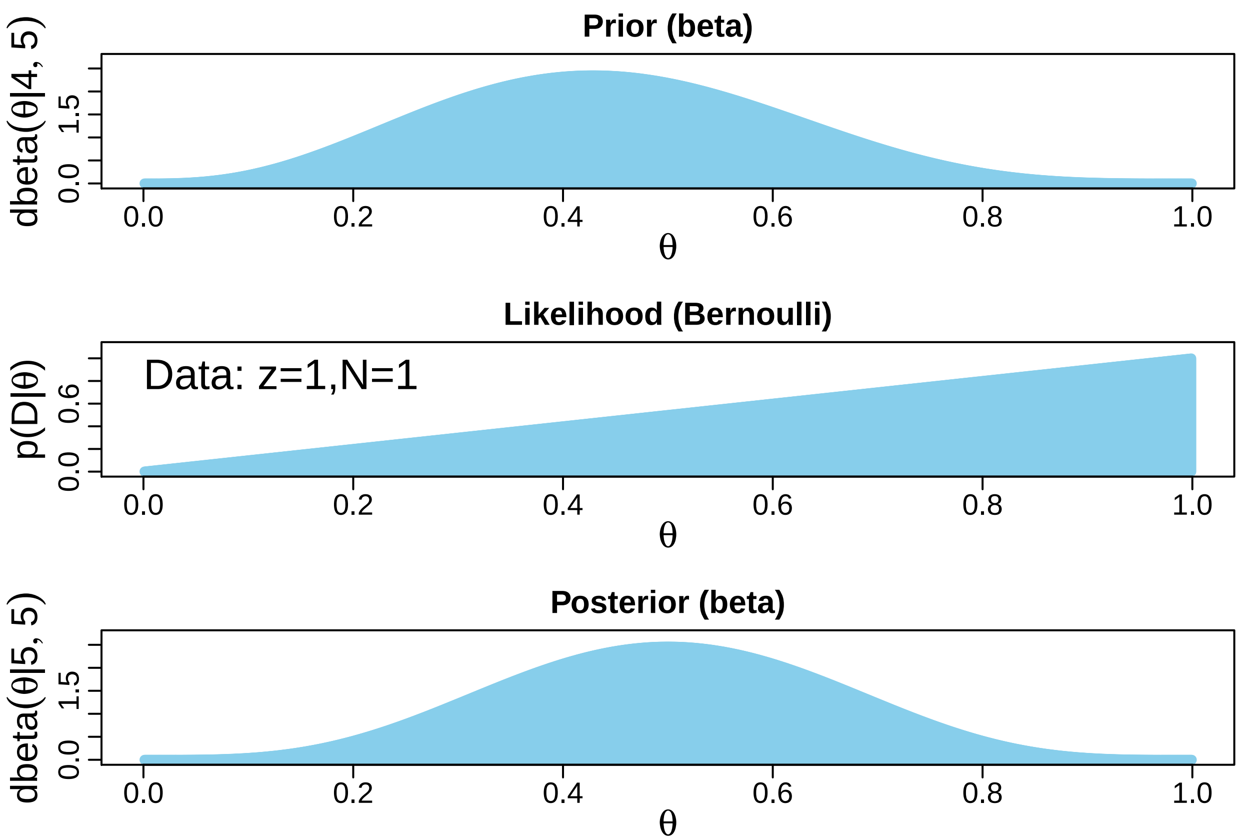
The posterior is `dbeta(theta|6,5)`.

d) Do the same three updates but in the order T, H, H instead of H, H, T. Is the final posterior distribution the same for both orderings of the flip results?

```
post = BernBeta( priorBetaAB=c(4,4) , Data=c(0) )
```




```
post = BernBeta( priorBetaAB=post , Data=c(1) )
```



```r
post = BernBeta( priorBetaAB=post , Data=c(1) )
```