

# Course EPIB-675 - Bayesian Analysis in Medicine

## Assignment 3 - R and Numerical Integration - Solutions

1. The course web page provides a function for doing Bayesian analysis of a normal mean. Using similar ideas, here you will create your own R function that does Bayesian analysis for beta prior binomial likelihood functions. I suggest that you proceed along the steps outlined below, but you just need to hand in a printout of your final program and the answer to part (e).

(a) Start by creating an R function that takes as input the information needed to provide the posterior distribution given beta prior parameters and number of successes and trial of the binomial data. Use parameter names that will help the user of your program understand the inputs required. The output should be the parameters of the beta posterior distribution.

(b) Add some graphics to your function, in particular the beta prior curve and the beta posterior curve. Here are some hints:

- Note the difference between the plot command (to start a new graph) versus points (to add points to graph). Note also the use of the lty option for making different line types, and the type option for making continuous lines (e.g., lty="1").
- To avoid cutting off the tops of the graphs, think about which order you want to make plots. Starting with the tallest plot (i.e., the posterior) may be a good idea.
- Use the range of (0,1) for your x-axis in all graphs. [A more sophisticated function might first examine all curves in the tri-plot, and automatically decide on a range based on this information ... I will leave this as an optional exercise.]

(c) Add a 95% credible interval to your output. (Hint: Investigate the use of the qbeta function, which gives quantiles of the beta density. For example perhaps using qbeta(0.025, alpha, beta) is useful for the lower limit, etc.

(d) Test your function on a trial data set and trial prior distribution.

(e) Run your program with a beta(5, 25) prior, and a data set with  $x = 10$  successes in  $n = 70$  trials. Print out all of the outputs (posterior, plot, and 95% interval).

*One program that works is:*

```
beta.binomial.analysis <- function(alpha.prior, beta.prior, x, n, level=0.95)
{
#####
# R function for Bayesian analysis of binomial data      #
#                                                         #
# Parameters included are:                                #
#                                                         #
# Inputs:                                                #
#                                                         #
# alpha.prior = 1st parameter of prior beta dist. for success rate #
# beta.prior  = 2nd parameter of prior beta dist. for success rate #
# x = number of observed successes in the binomial data set      #
# n = sample size in the data set                               #
# level = desired probability of the credible set                #
#                                                         #
#                                                         #
# Outputs:                                                #
#                                                         #
# alpha.post = 1st parameter of post beta dist. for success rate #
# beta.post  = 2nd parameter of post beta dist. for success rate #
#                                                         #
# lower.post = lower limit of posterior credible set           #
# upper.post = upper limit of posterior credible set           #
#                                                         #
# A tri-plot is also output                                   #
#                                                         #
#####

alpha.post <- alpha.prior + x
beta.post <- beta.prior + n - x
lower.post <- qbeta((1-level)/2, alpha.post, beta.post)
```

```

upper.post <- qbeta(1-(1-level)/2, alpha.post, beta.post)

x.axis <- seq(0,1,by=.001)

plot(x.axis, dbeta(x.axis, alpha.post, beta.post), type="l", lty=1,
     main = "Tri-Plot", xlab="probability of success", ylab="Density",
     ylim=c(0,dbeta(alpha.post/(alpha.post+beta.post),alpha.post,beta.post)*1.5) )
points(x.axis, dbeta(x.axis, alpha.prior, beta.prior), type="l", lty=2)
points(x.axis, dbeta(x.axis, x+1, (n-x)+1), type="l", lty=3)
legend(x=0.01, y=dbeta(alpha.post/(alpha.post+ beta.post),alpha.post,
     beta.post)*1.48,
     legend=c("Posterior", "Prior", "Likelihood") , lty=c(1,2,3))

text1 <- "Posterior distribution is: Beta("
text2 <- " , "
text3 <- " ) "
text4 <- " Percent Credible Interval is:"

cat(text1, alpha.post, text2, beta.post, text3, "\n", level*100, text4,
     lower.post, text2, upper.post )
}

```

*Note a “trick” used to place the legend in the graph so that it does not interfere much with the curve: I changed the ylim option (the limits of the y-axis) to be 50% that needed, so that there is room for the legend.*

*Using the program for part (e) gives:*

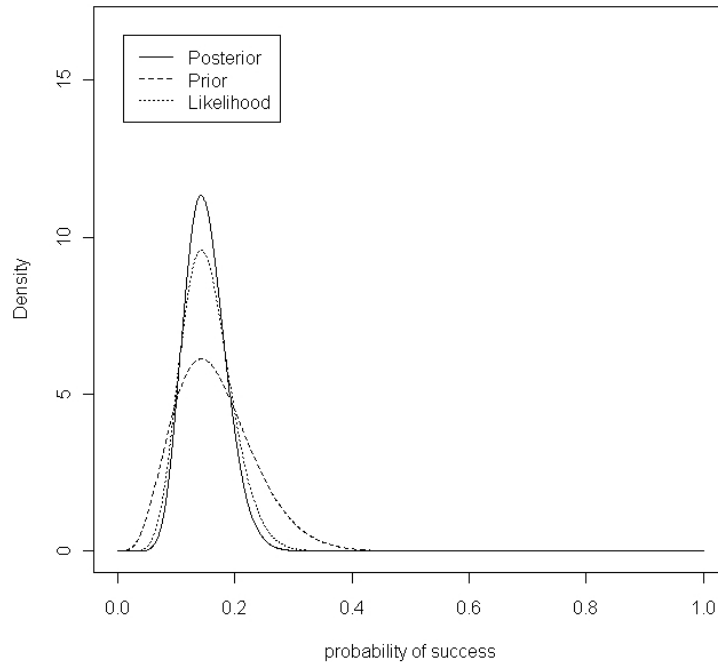
```

> beta.binomial.analysis(5, 25, 10, 70)
Posterior distribution is: Beta( 15 , 85 )
95 Percent Credible Interval is: 0.08735483 , 0.2258735

```

*with graph*

Tri-Plot



2. Suppose we have a parameter  $\theta$  that has a posterior distribution that is  $\text{beta}(20, 100)$ . What is the probability that  $\theta$  is between 0.10 and 0.20? The answer is of course given by the definite integral

$$Pr\{0.10 < \theta < 0.20\} = \int_{0.10}^{0.20} \frac{1}{B(20, 100)} \theta^{20-1} (1 - \theta)^{100-1} d\theta$$

We will investigate two different ways to solve this problem.

(a) Use the R function “integrate”, to directly integrate the above function, which is a  $\text{dbeta}(20,100)$  function. [Hint: The help on integrate (type ?integrate from the R command line) shows examples from the normal density, follow the example but change the density to a beta.]

*The following two lines of R code is what is needed. The first line defines the function to integrate, the second does the integration.*

```
> integrand <- function(x) {dbeta(x, 20,100)}  
> integrate(integrand, .1, .2)
```

0.8233355 with absolute error < 9.1e-15

(b) The R function `pbeta` gives you the probability of being less than any given value for any beta density. Use the difference between two `pbeta`'s to solve the integral. Check that your answers in (a) and (b) are very similar.

*The following single line of R code is what is needed.*

```
> pbeta(.2, 20,100) - pbeta(.1, 20,100)
[1] 0.8233355
```

*Results from two different methods in R are identical.*

3. In this problem we will use Monte Carlo integration to find the variance of a beta density. Suppose we have a `beta(50, 30)` density. What is its variance? We will do this in three ways:

(a) First, calculate the variance “by hand”, using the formula for the variance of a beta density,

$$\sigma^2 = \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)}$$

Of course, you can do this calculation in R, or any hand calculator, etc.

*Plugging the  $\alpha$  and  $\beta$  values into the equation gives 0.0028935.*

(b) Next, remember that the variance is just an integral, so the R command `integrate` can be used, similar to problem 2 above, but with a different definite integral, defined as a beta density (using `dbeta` as before), but with an extra term in front,  $(\theta - 0.625)^2$ , where 0.625 is the mean of this beta density.

*The following two lines of R code is what is needed. The first line defines the function to integrate, the second does the integration.*

```
> integrand <- function(x) {(x-0.625)^2}*dbeta(x, 50,30)}
> integrate(integrand, 0, 1)
0.002893519 with absolute error < 2.4e-05
```

*Once again, the answer is the same.*

(c) Finally, this problem can be solved using Monte Carlo integration. Here, the R command `rbeta` can be used, and see the class notes for hints on how to calculate variances using Monte Carlo integration.

*Simply get a random sample from a  $\text{beta}(50, 30)$  distribution, and then ask for the variance of that sample. R code is (I have taken a large sample of 1,000,000 for greater accuracy):*

```
> sbeta <- rbeta(1000000, 50,30)
> var(sbeta)
[1] 0.002890365
```

(d) Compare your answers from (a), (b), and (c) to ensure that they are all the same (or at least, very close).

*Yes, all methods seems to work very well, they all agree to about five decimal places.*

4. You have just gone shopping, and received a quarter in change from the cashier.

(a) Assume that your prior probability that the coin will come up heads in any given toss can be expressed by a beta distribution with appropriately chosen  $\alpha$  and  $\beta$  parameters. State your prior distribution. (Note: There is no “correct” answer, since each individual will have their own prior distribution. However, you should justify your answer in terms of your prior mean and variance (or standard deviation), that is, check to ensure that the values of  $\alpha$  and  $\beta$  give reasonable means and variances. You may wish to imagine a 95% probability interval, and consider that the mean is in the center of that interval, and that four times the standard deviation will equal the length of that interval.)

*I have no reason to believe that the coin is unfair, but also no strong reason to think it is perfectly balanced, either. I will therefore use a  $\text{beta}(1000,1000)$  prior distribution, which has 95% interval of about  $(0.48, 0.52)$ , which allows for some imbalance. Other choices, of course, can also be defended, depending on prior views.*

(b) Suppose that the coin is now tossed 5 times, and there are no heads. Use your program from question 1 of this assignment to calculate the posterior probability for the probability of heads for that coin and provide a 95% credible interval.

*The lines needed to run the program are:*

```
beta.binomial.analysis(1000,1000,0,5) Posterior distribution is:
Beta( 1000 , 1005 )
95 Percent Credible Interval is: 0.4768764 , 0.5206322
```

(c) If you were using a frequentist approach to analyse the same data (i.e., five tails in a row), what would the 95% confidence interval be? [Note: the normal approximation does not work well here, so you may consider using an “exact” approach. There is an R program for exact binomial inferences on the course web page.]

*Using the program on the course website gives:*

```
> ci.prop(0,5)
      x n x/n level p_L      p_U
[1,] 0 5  0 0.95  0 0.4507197
```

*So the interval in fact misses 0.5, going from 0 to 0.45.*

(d) Provide interpretations of the intervals you calculated in parts (b) and (c). Which of the intervals given in (c) or (d) do you prefer? Why?

*I personally trust the interval from part (b) more than (c). I have much experience in coin tossing, and the physics of the coin toss indicates that there should be some balance between heads and tails, even if imperfect. I suspect that the five heads in a row were unlucky, and the frequentist interval, which ignores prior experience, reflects unlucky data more than it reflects a good interval for the true probability. A few hundred more tosses should settle the issue, but if all I have is the result of five tosses, I prefer the Bayesian interval.*

5. Two researchers are looking at preliminary results for a new surgical technique. One researcher was very optimistic about the new technique, and had a prior distribution for the success rate of the technique of  $\text{beta}(40,10)$ . The second researcher was less confident in the new technique, but also less sure of his opinion, and so had a  $\text{beta}(5,5)$  prior. In the first ten trials of the use of this technique, the surgery was successful 7 times.

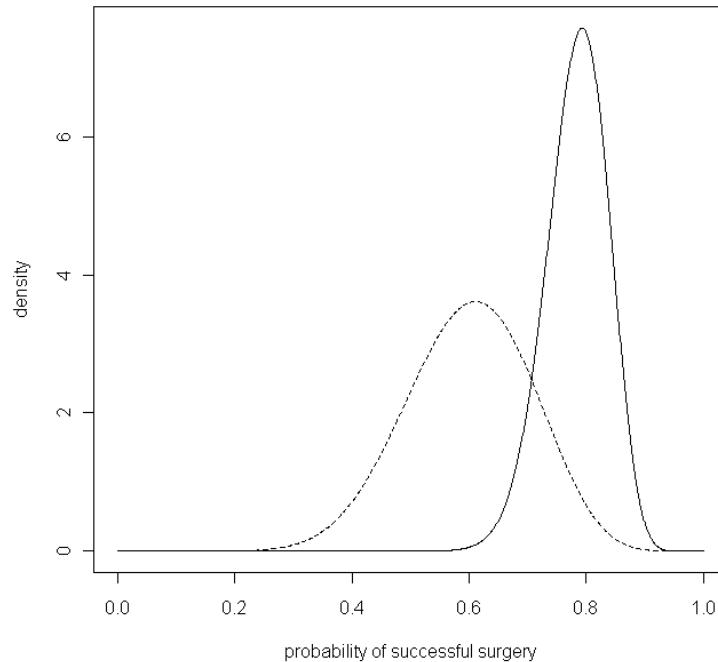
(a) What are the two researchers posterior distributions?

*Posterior from the first researcher is  $\text{beta}(47, 13)$ , and from the second it is  $\text{beta}(12,8)$ .*

(b) Plot the two posterior distributions on the same graph in R.

*Using the commands listed, the graph follows below.*

```
> plot(seq(0,1,by=0.001), dbeta(seq(0,1,by=0.001), 47,13), type="l",  
      xlab="probability of successful surgery", ylab="density", lty=1)  
> points(seq(0,1,by=0.001), dbeta(seq(0,1,by=0.001), 12,8), type="l", lty=2)
```



(c) Using Monte Carlo integration, compare the two posterior distributions by calculating the probability that the rate from the first researcher is greater than the rate of the second researcher.

Using the commands listed, the result follows below. Note the use of the length function, so see how often post1 was larger than post2.

```
> post1<- rbeta(100000, 47, 13)
> post2<- rbeta(100000, 12, 8)
> diff <- post1 - post2
> length(diff[diff > 0])/100000
[1] 0.94081
```

So about 94% of the posterior distribution from the first researcher lies to the right of the posterior distribution of researcher 2. This was to be expected if we look at the graph.