Bayesian Methods - Assignment 2

Ryan Durfey

April 17, 2016

```
dataPath<-"//Users/rdurfey/R_misc/BayesianMethods"
source(paste(dataPath,"DBDA2Eprograms/DBDA2E-utilities.R",sep="/"))
```

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

## Loading required package: coda

## Linked to JAGS 4.2.0

## Loaded modules: basemod,bugs

## Warning: package 'runjags' was built under R version 3.2.4
```

```
source(paste(dataPath,"DBDA2Eprograms/BernGrid.R",sep="/"))
opar<-par(no.readonly = TRUE)
```

PART 1: Find posterior given initial data and uniform prior. Then find updated posterior with additional data.

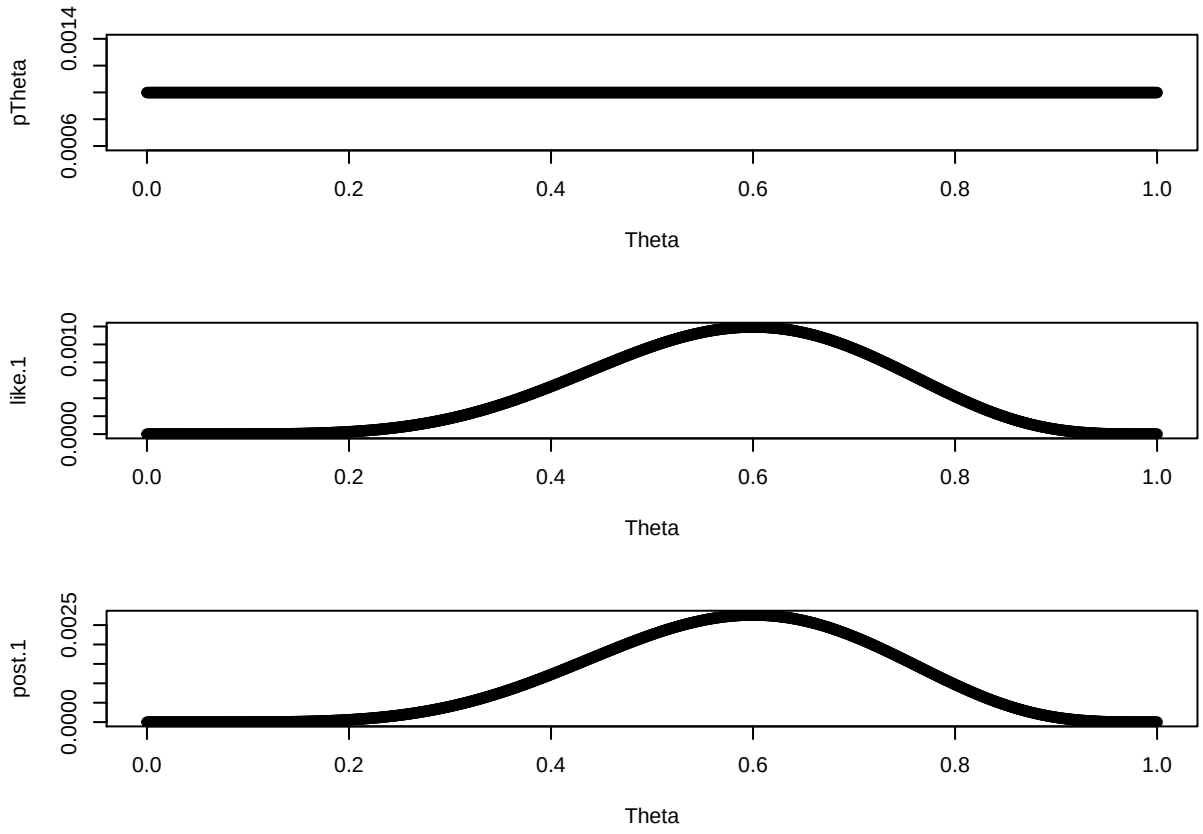
1. Find posterior probability for binomial model with uniform prior and data. Use the same seed 81 for simulation of theta.

```
# data w/ seed = 81
set.seed(81)
(data1<-rbinom(10,1,.71))
```

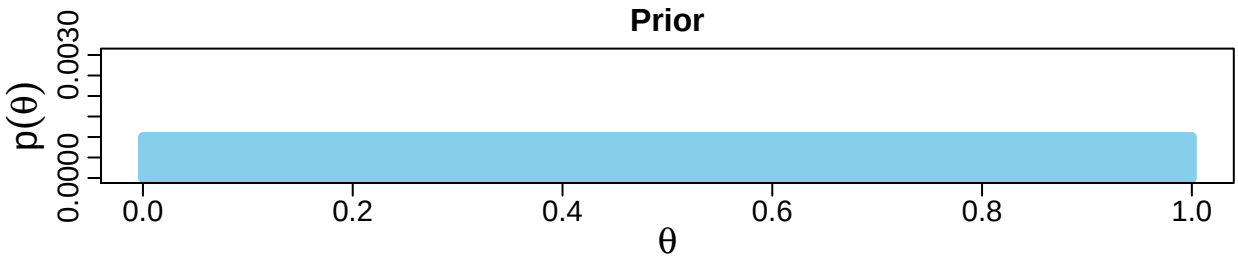
```
##  [1] 1 0 1 1 0 0 0 1 1 1
```

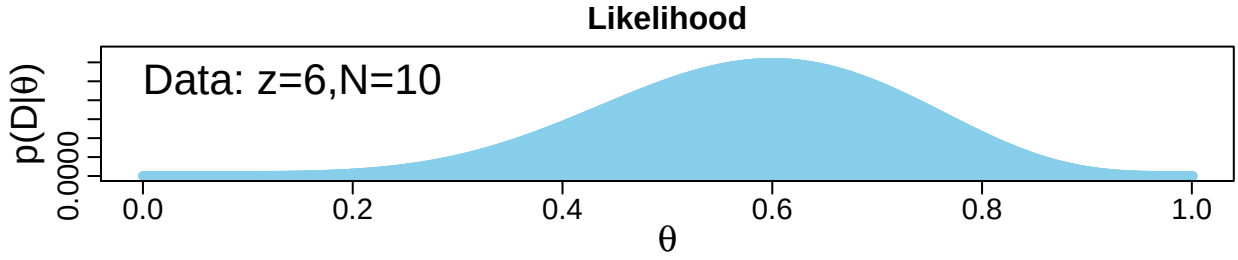
```
# Define likelihood function for binomial distribution.
likeli<-function(par,data){
  sdata<-sum(data)
  ldata<-length(data)
  return(par^sdata*(1-par)^(ldata-sdata))
}

# Define values of parameter theta and prior distribution.
Theta = seq( .00001 , 1-.00001 , length=1001 ) # Fine teeth for Theta.
```

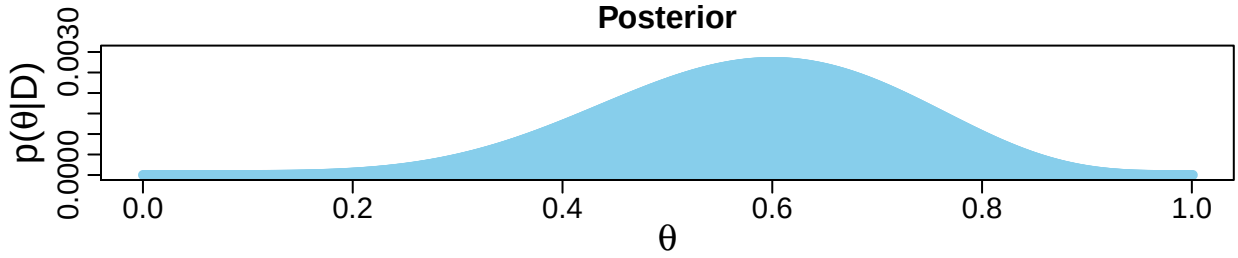
```r
pTheta = rep(1,length(Theta))       # Uniform (horizontal) shape for pTheta.
pTheta = pTheta/sum(pTheta)         # Make pTheta sum to 1.0
# plot(Theta,pTheta)

# likelihood
like.1<-likeli(Theta,data1)

# posterior
post.1<-like.1*pTheta
post.1<-post.1/sum(post.1)

# plot
par(mfrow=c(3,1),mar=c(5.1,4.1,1.7,2.1))
plot(Theta,pTheta)
plot(Theta,like.1)
plot(Theta,post.1)
```



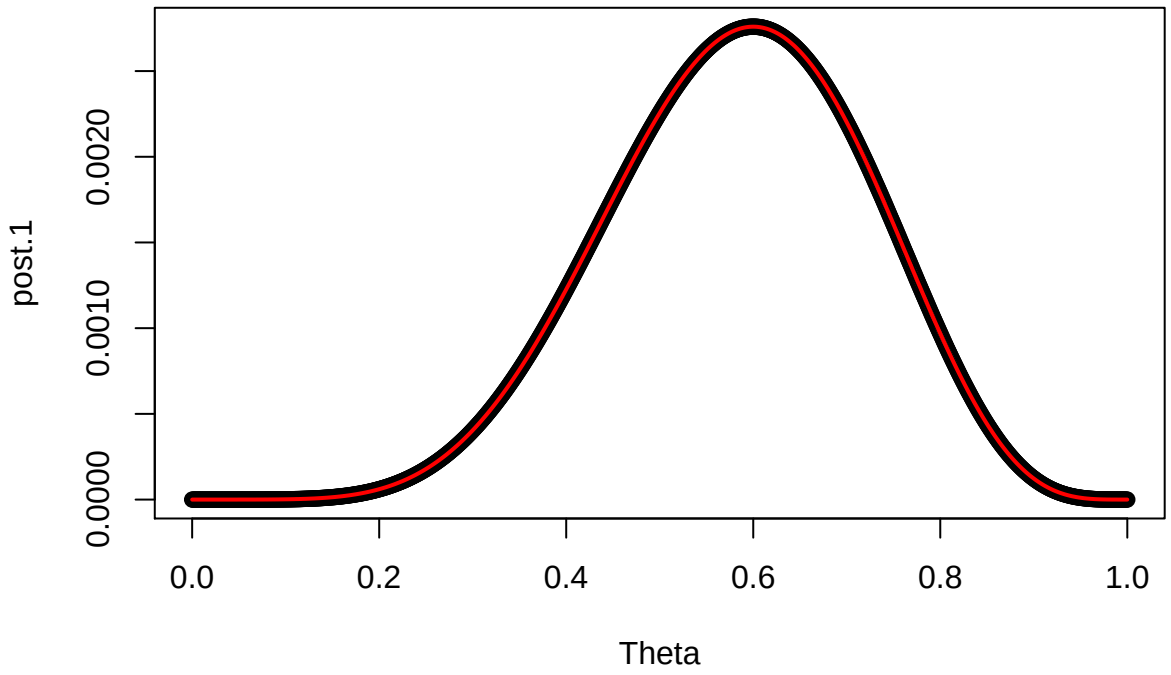

```r
par(opar)


# confirm plot shape w/ BernGrid() function
posterior = BernGrid( Theta, pTheta , data1 , plotType="Bars" ,
                      showCentTend="None" , showHDI=FALSE , showpD=FALSE )
```








```r
par(opar)

# approximate posterior distribution function
post.1.distr<-approxfun(Theta,post.1,method="linear")
plot(Theta,post.1)
lines(Theta,post.1.distr(Theta),col="red",lwd=2)
```




```
# mode of posterior
(mode1<-Theta[which.max(post.1.distr(Theta))])
```

```
## [1] 0.599998
```

```
# mean of posterior
(mean1<-Theta%*%post.1.distr(Theta)/sum(post.1.distr(Theta)))
```

```
##           [,1]
## [1,] 0.5833333
```

```
# variance of posterior
(var1<-((Theta-mean1)^2)%*%post.1.distr(Theta)/sum(post.1.distr(Theta)))
```

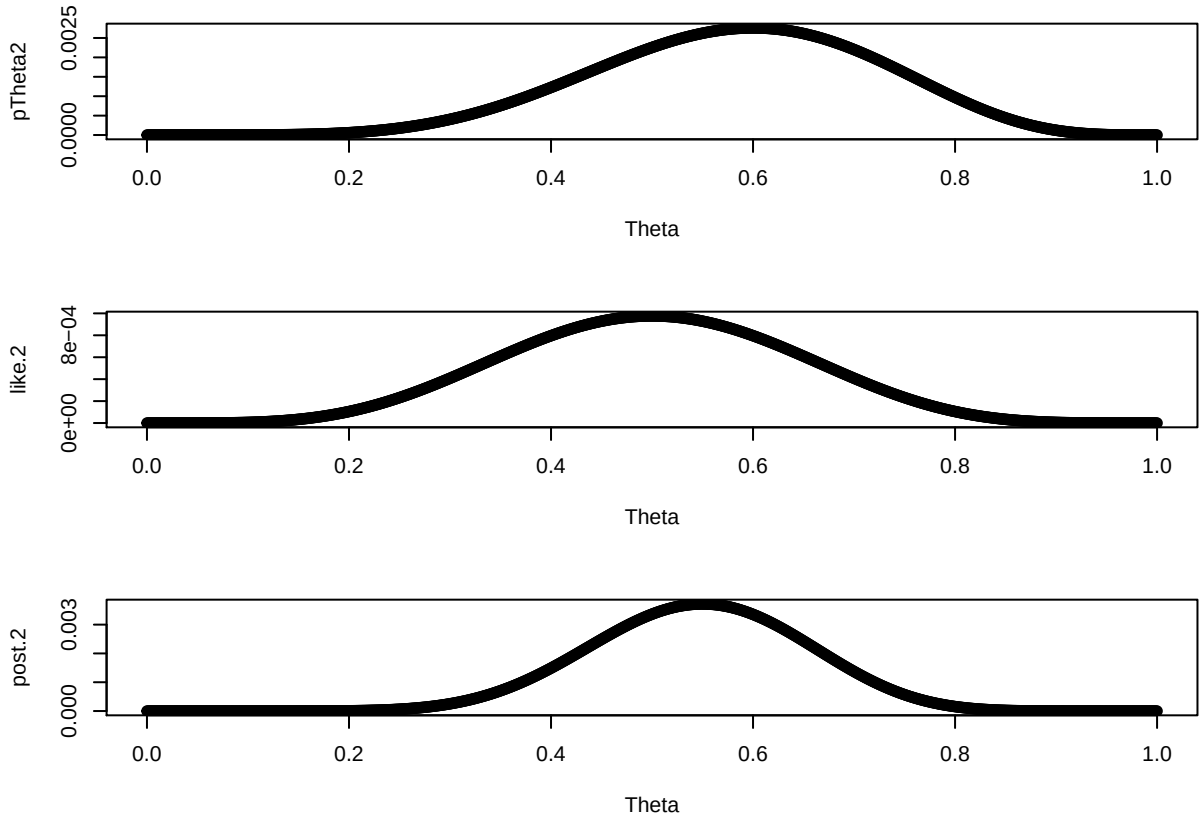
```
##            [,1]
## [1,] 0.01869658
```

2. Add more data and recalculate posterior distribution. Use the same seed 97 for second simulation of theta.

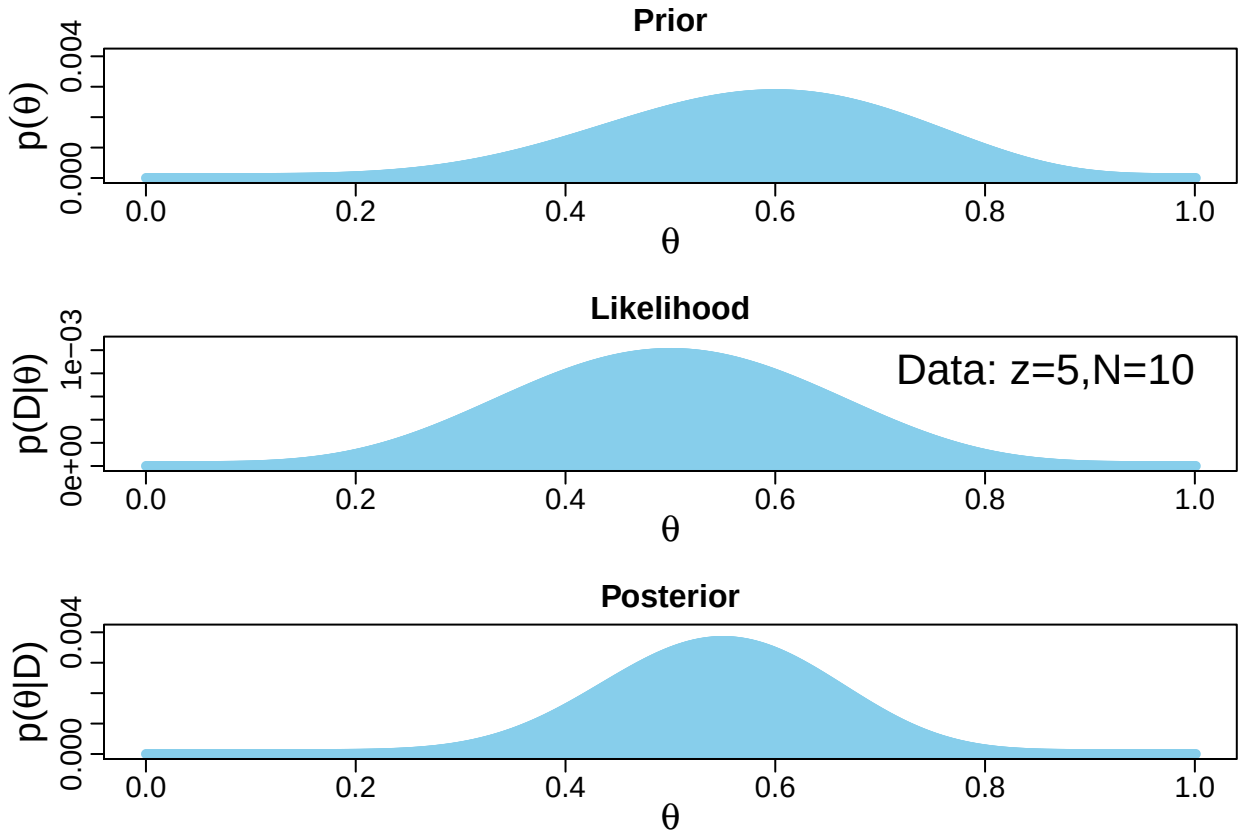
```
set.seed(97)
(data2<-rbinom(10,1,.71))
```

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

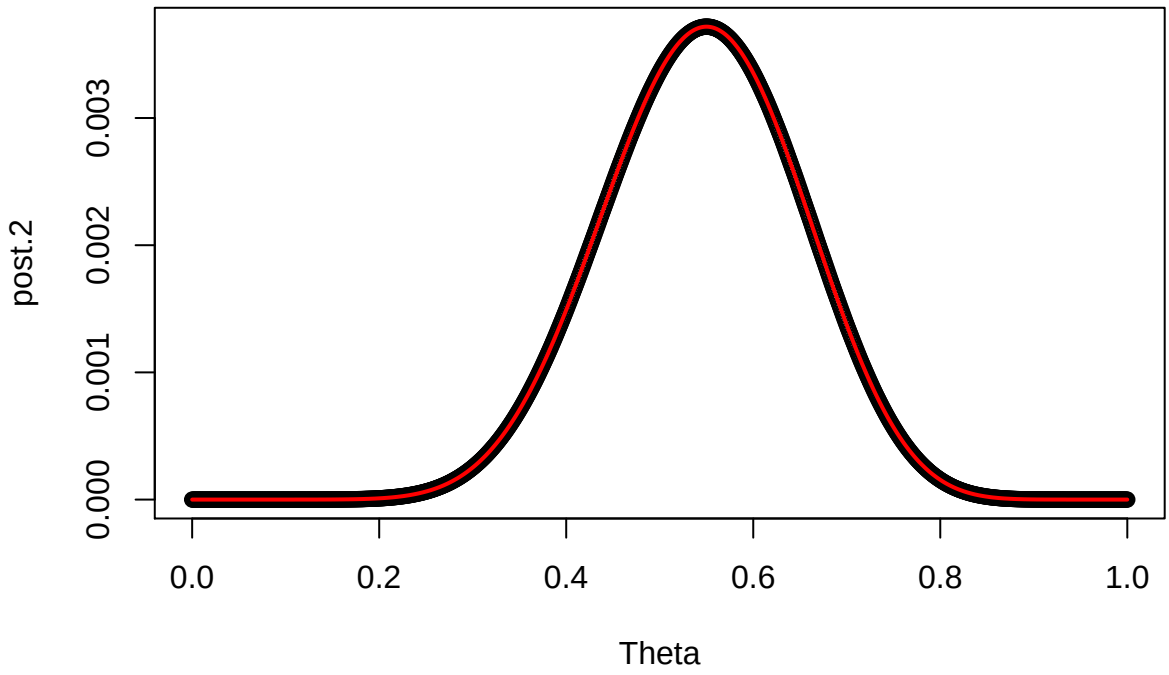
```r
# Replace prior distribution with the posterior distribution and add new data
pTheta2<-post.1.distr(Theta)/sum(post.1.distr(Theta))

# likelihood
like.2<-likeli(Theta,data2)

# posterior
post.2<-like.2*pTheta2
post.2<-post.2/sum(post.2)

# plot
par(mfrow=c(3,1),mar=c(5.1,4.1,1.7,2.1))
plot(Theta,pTheta2)
plot(Theta,like.2)
plot(Theta,post.2)
```




```r
par(opar)

# confirm plot shape w/ BernGrid() function
posterior = BernGrid( Theta, pTheta2 , data2 , plotType="Bars" ,
                      showCentTend="None" , showHDI=FALSE , showpD=FALSE )
```




```
par(opar)

# approximate posterior distribution function
post.2.distr<-approxfun(Theta,post.2,method="linear")
plot(Theta,post.2)
lines(Theta,post.2.distr(Theta),col="red",lwd=2)
```




```
# mode of posterior
(mode2<-Theta[which.max(post.2.distr(Theta))])
```

```
## [1] 0.549999
```

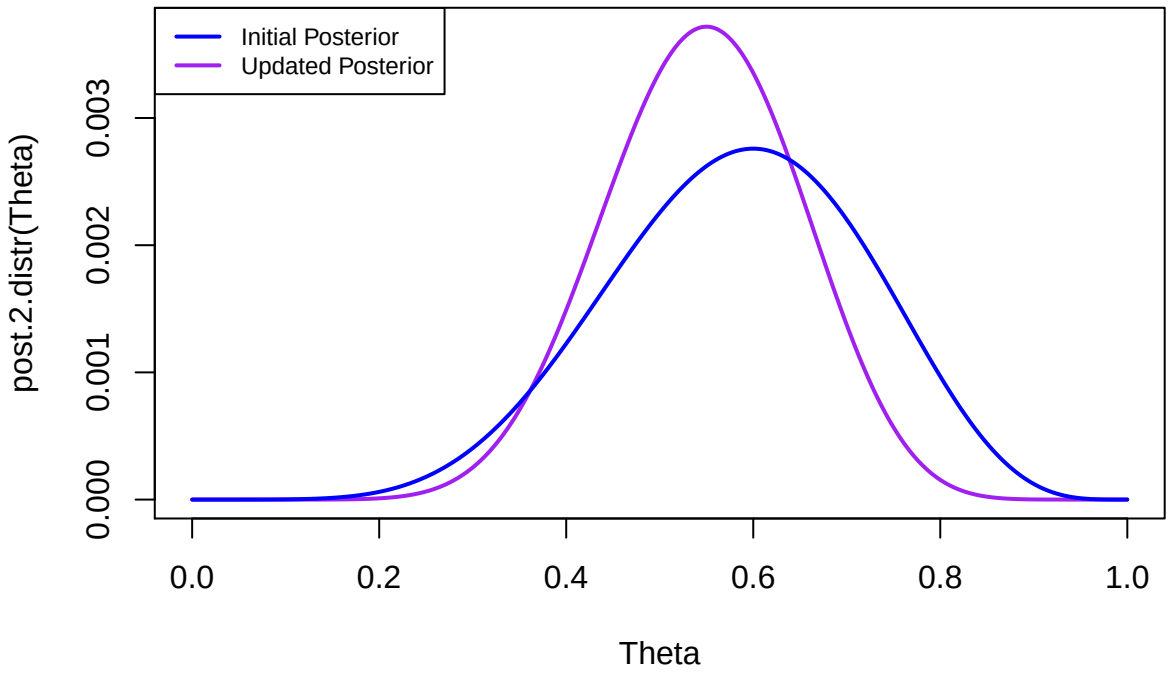
```
# mean of posterior
(mean2<-Theta%*%post.2.distr(Theta)/sum(post.2.distr(Theta)))
```

```
##           [,1]
## [1,] 0.5454545
```

```
# variance of posterior
(var2<-((Theta-mean2)^2)%*%post.2.distr(Theta)/sum(post.2.distr(Theta)))
```

```
##            [,1]
## [1,] 0.01077973
```

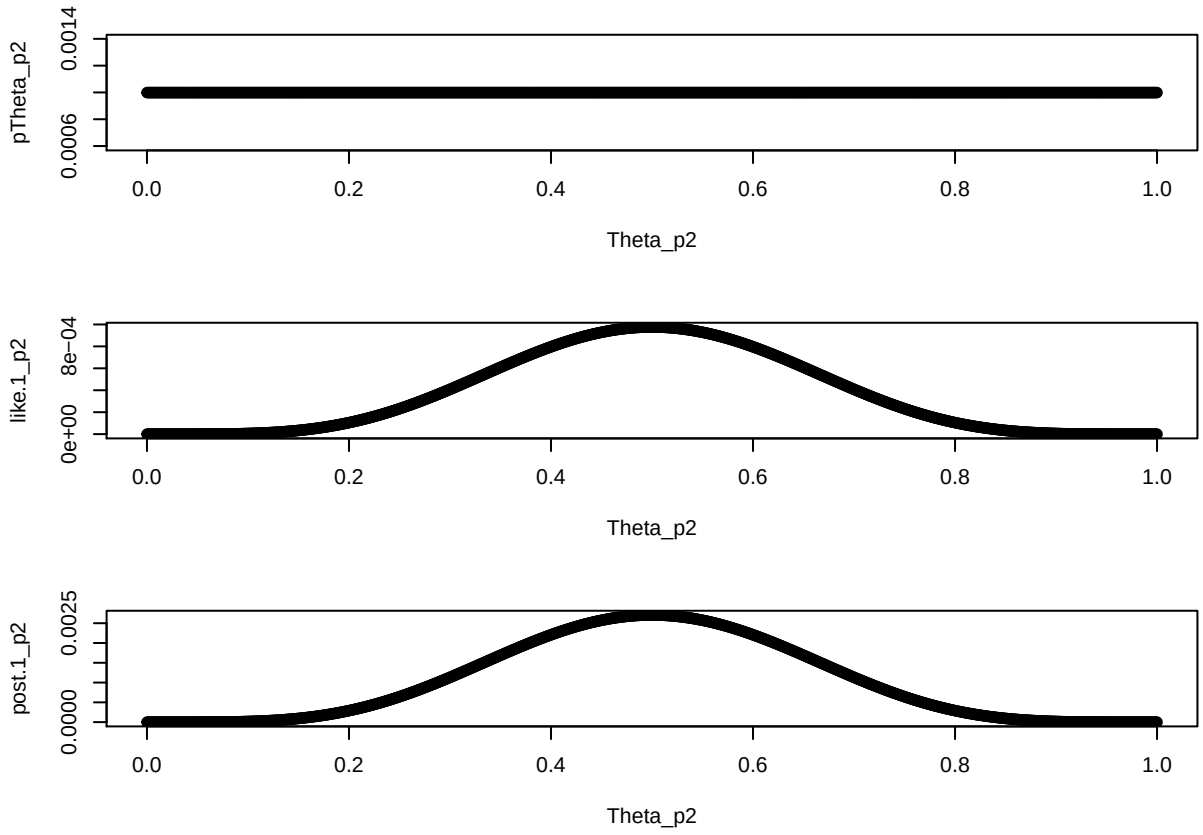
```
### Comparison of Initial Posterior and Updated Posterior
plot(Theta,post.2.distr(Theta),col='purple',type='l',lwd=2)
lines(Theta,post.1.distr(Theta),col='blue',type='l',lwd=2)
legend("topleft",c("Initial Posterior","Updated Posterior"),
       lty=c(1,1),lwd=c(2,2),col=c("blue","purple"),cex=0.75)
```



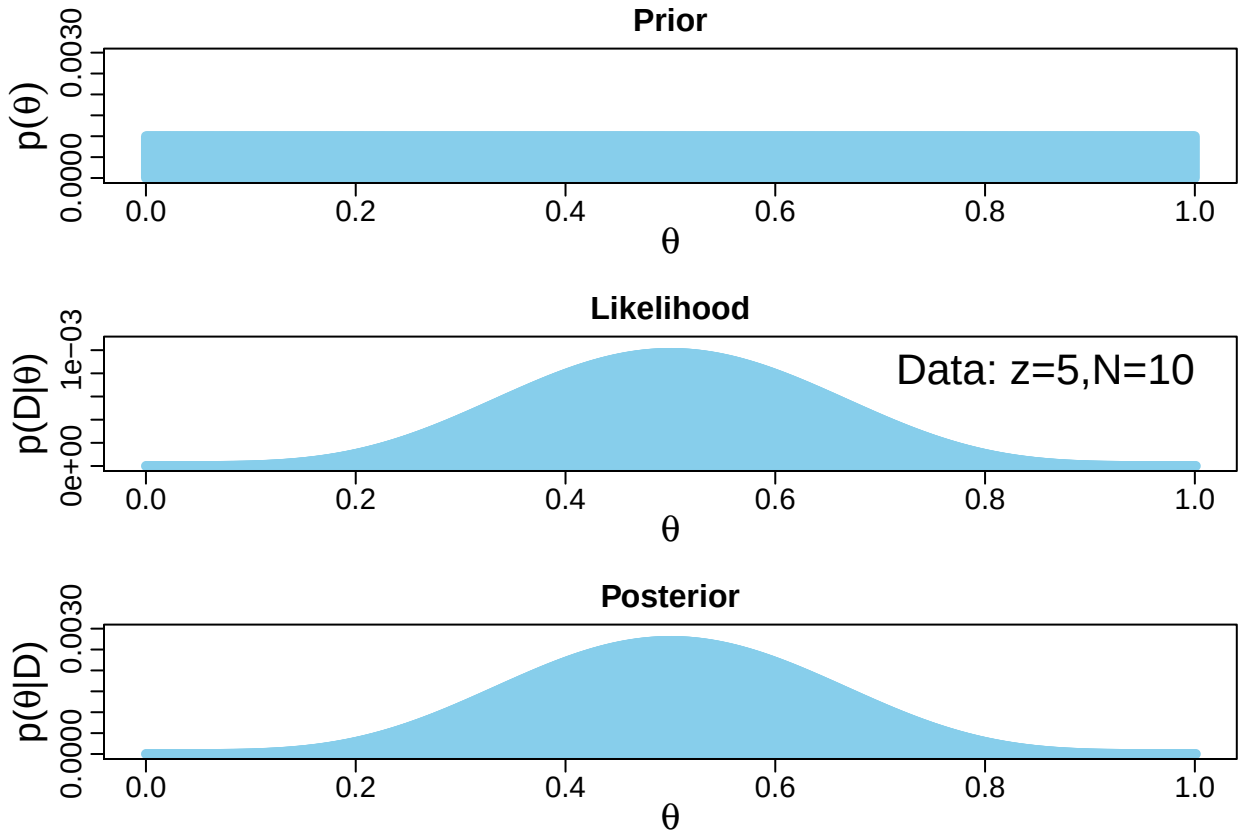
PART 2: Repeat steps of Part 1, but using data1 and data2 in reverse order.

1. Find posterior probability for binomial model with uniform prior and data

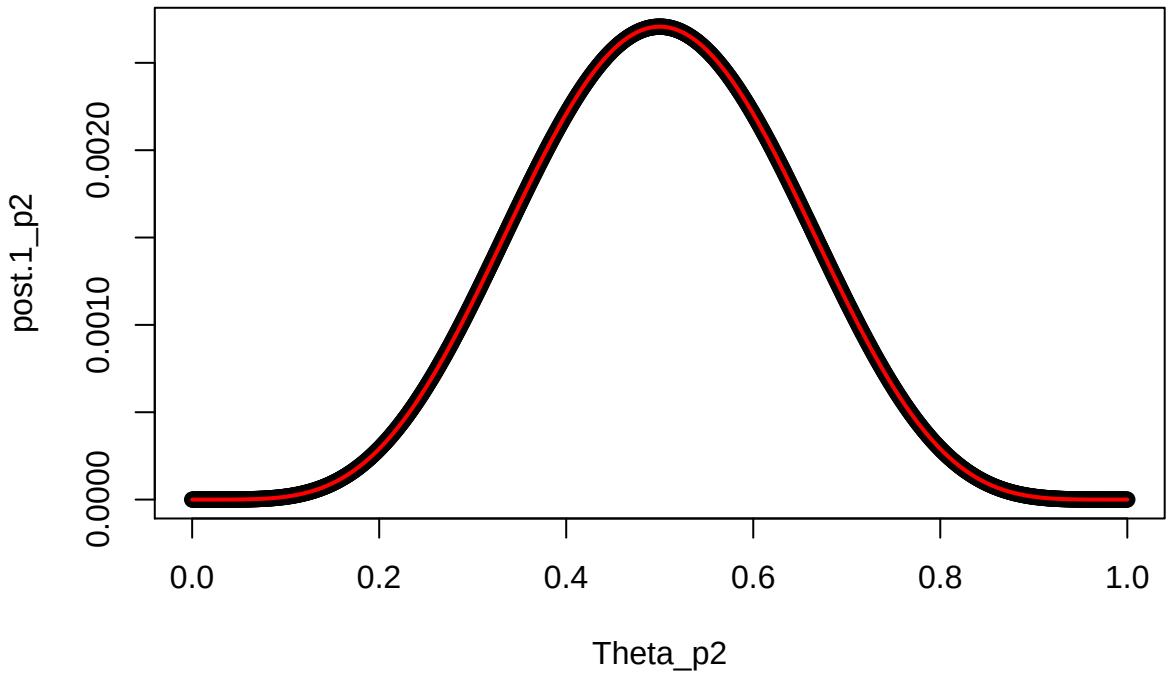
```
# Define values of parameter theta and prior distribution.
Theta_p2 = seq( .00001 , 1-.00001 , length=1001 ) # Fine teeth for Theta.
pTheta_p2 = rep(1,length(Theta))      # Uniform (horizontal) shape for pTheta.
pTheta_p2 = pTheta_p2/sum(pTheta_p2)        # Make pTheta sum to 1.0
# plot(Theta_p2,pTheta_p2)

# likelihood
like.1_p2<-likeli(Theta,data2) # use data2 first

# posterior
post.1_p2<-like.1_p2*pTheta
post.1_p2<-post.1_p2/sum(post.1_p2)

# plot
par(mfrow=c(3,1),mar=c(5.1,4.1,1.7,2.1))
plot(Theta_p2,pTheta_p2)
plot(Theta_p2,like.1_p2)
plot(Theta_p2,post.1_p2)
```




```
par(opar)

# confirm plot shape w/ BernGrid() function
posterior = BernGrid( Theta_p2, pTheta_p2, data2, plotType="Bars" ,
                      showCentTend="None" , showHDI=FALSE , showpD=FALSE )
```




```r
par(opar)

# approximate posterior distribution function
post.1.distr_p2<-approxfun(Theta_p2,post.1_p2,method="linear")
plot(Theta_p2,post.1_p2)
lines(Theta_p2,post.1.distr_p2(Theta),col="red",lwd=2)
```




```
# mode of posterior
(mode1_p2<-Theta_p2[which.max(post.1.distr_p2(Theta_p2))])
```

```
## [1] 0.5
```

```
# mean of posterior
(mean1_p2<-Theta_p2%*%post.1.distr_p2(Theta_p2)/sum(post.1.distr_p2(Theta_p2)))
```

```
##      [,1]
## [1,]  0.5
```

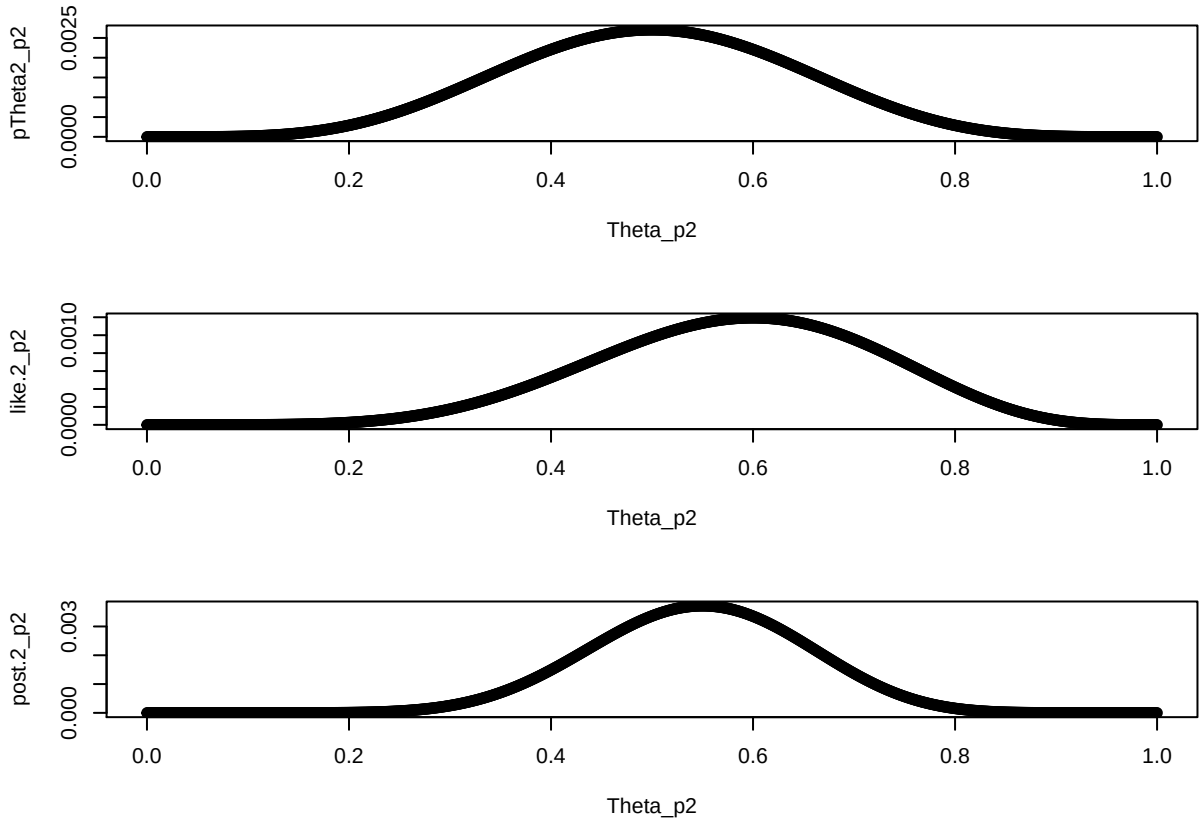
```
# variance of posterior
(var1_p2<-((Theta_p2-mean1_p2)^2)%*%post.1.distr_p2(Theta_p2)/sum(post.1.distr_p2(Theta_p2)))
```

```
##            [,1]
## [1,] 0.01923077
```

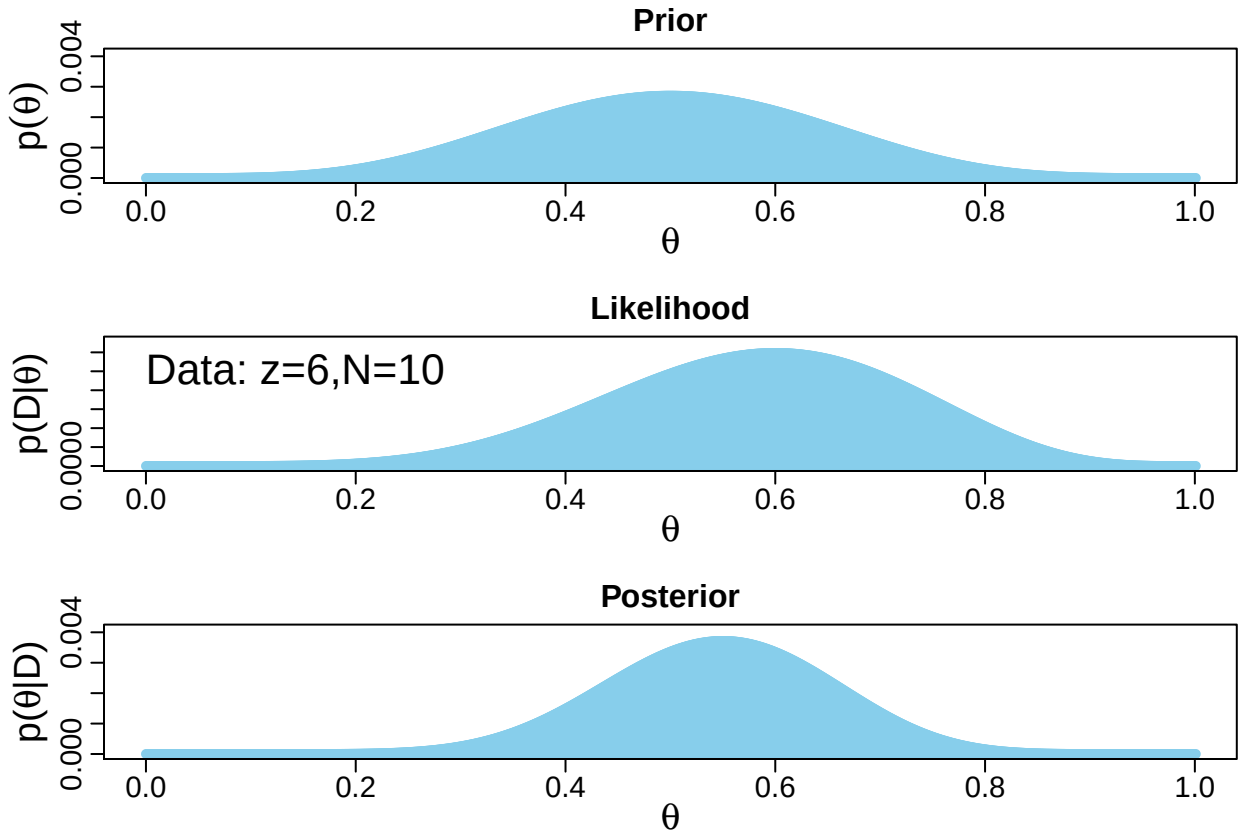
2. Add more data and recalculate posterior distribution.

```
# Replace prior distribution with the posterior distribution and add new data
pTheta2_p2<-post.1.distr_p2(Theta_p2)/sum(post.1.distr_p2(Theta_p2))

# likelihood
like.2_p2<-likeli(Theta_p2,data1) # now throw in data1

# posterior
post.2_p2<-like.2_p2*pTheta2_p2
post.2_p2<-post.2_p2/sum(post.2_p2)
```

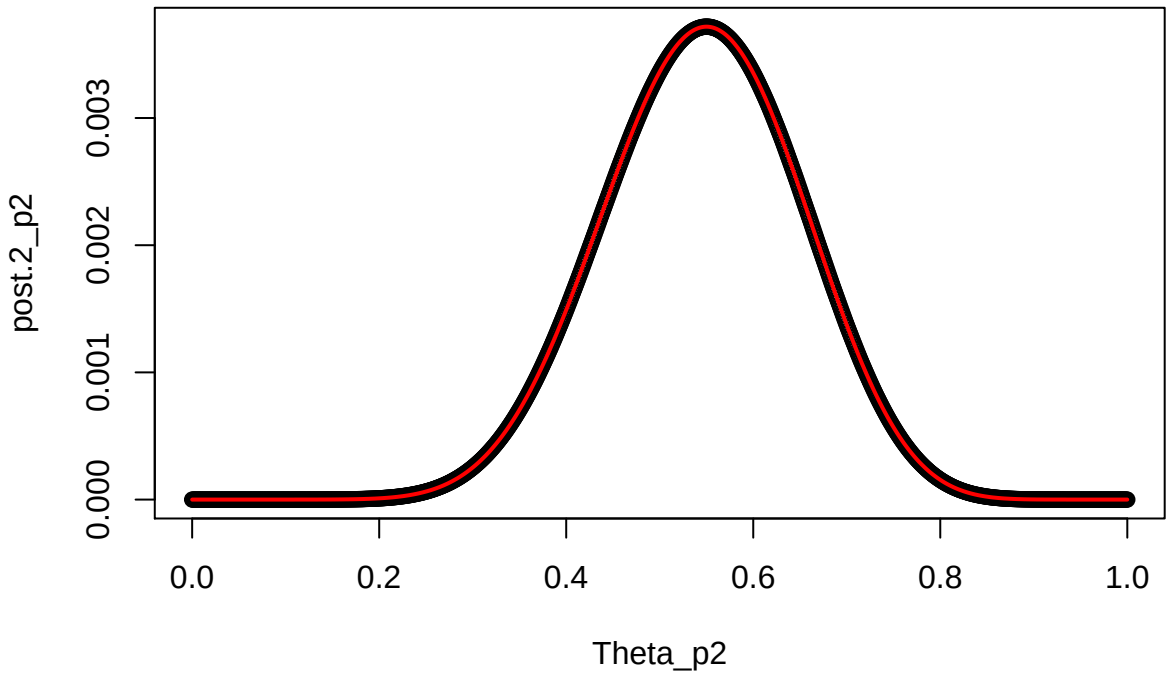
```
# plot
par(mfrow=c(3,1),mar=c(5.1,4.1,1.7,2.1))
plot(Theta_p2,pTheta2_p2)
plot(Theta_p2,like.2_p2)
plot(Theta_p2,post.2_p2)
```




```
par(opar)

# confirm plot shape w/ BernGrid() function
posterior = BernGrid( Theta_p2, pTheta2_p2, data1, plotType="Bars" ,
                      showCentTend="None" , showHDI=FALSE , showpD=FALSE )
```




```
par(opar)

# approximate posterior distribution function
post.2.distr_p2<-approxfun(Theta_p2,post.2_p2,method="linear")
plot(Theta_p2,post.2_p2)
lines(Theta_p2,post.2.distr_p2(Theta),col="red",lwd=2)
```




```
# mode of posterior
(mode2_p2<-Theta_p2[which.max(post.2.distr_p2(Theta_p2))])
```

```
## [1] 0.549999
```

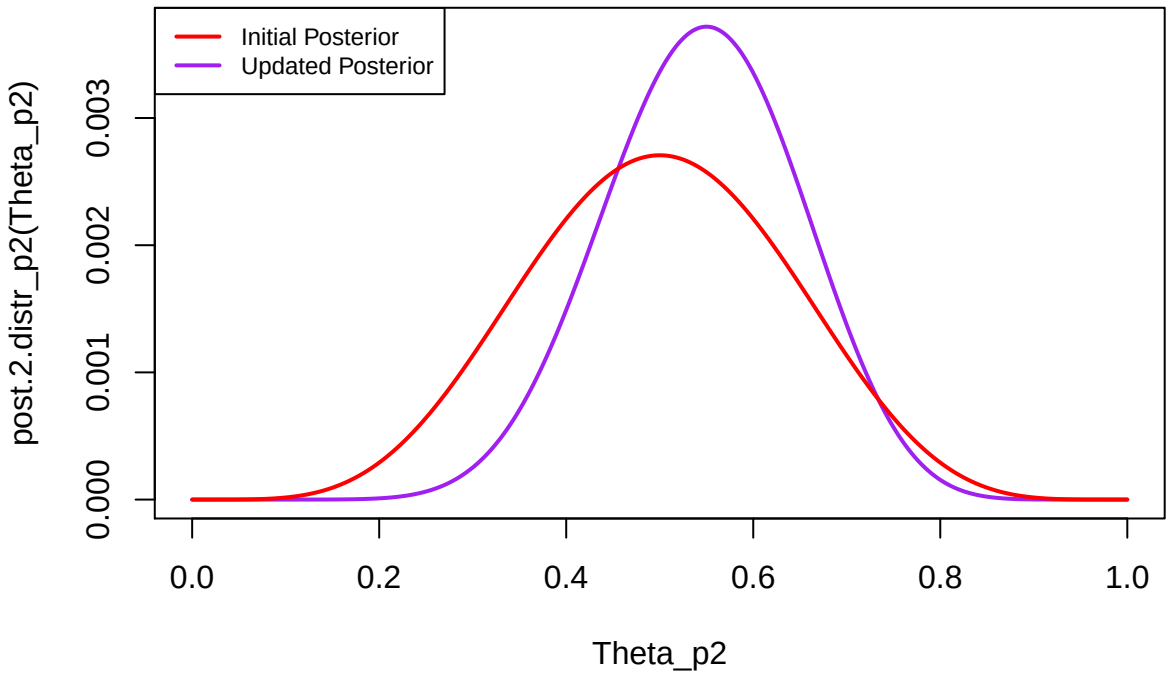
```
# mean of posterior
(mean2_p2<-Theta_p2%*%post.2.distr_p2(Theta)/sum(post.2.distr_p2(Theta_p2)))
```

```
##           [,1]
## [1,] 0.5454545
```

```
# variance of posterior
(var2_p2<-((Theta_p2-mean2_p2)^2)%*%post.2.distr_p2(Theta_p2)/sum(post.2.distr_p2(Theta_p2)))
```

```
##            [,1]
## [1,] 0.01077973
```

```
### Comparison of Initial Posterior and Updated Posterior
plot(Theta_p2,post.2.distr_p2(Theta_p2),col='purple',type='l',lwd=2)
lines(Theta_p2,post.1.distr_p2(Theta_p2),col='red',type='l',lwd=2)
legend("topleft",c("Initial Posterior","Updated Posterior"),
       lty=c(1,1),lwd=c(2,2),col=c("red","purple"),cex=0.75)
```

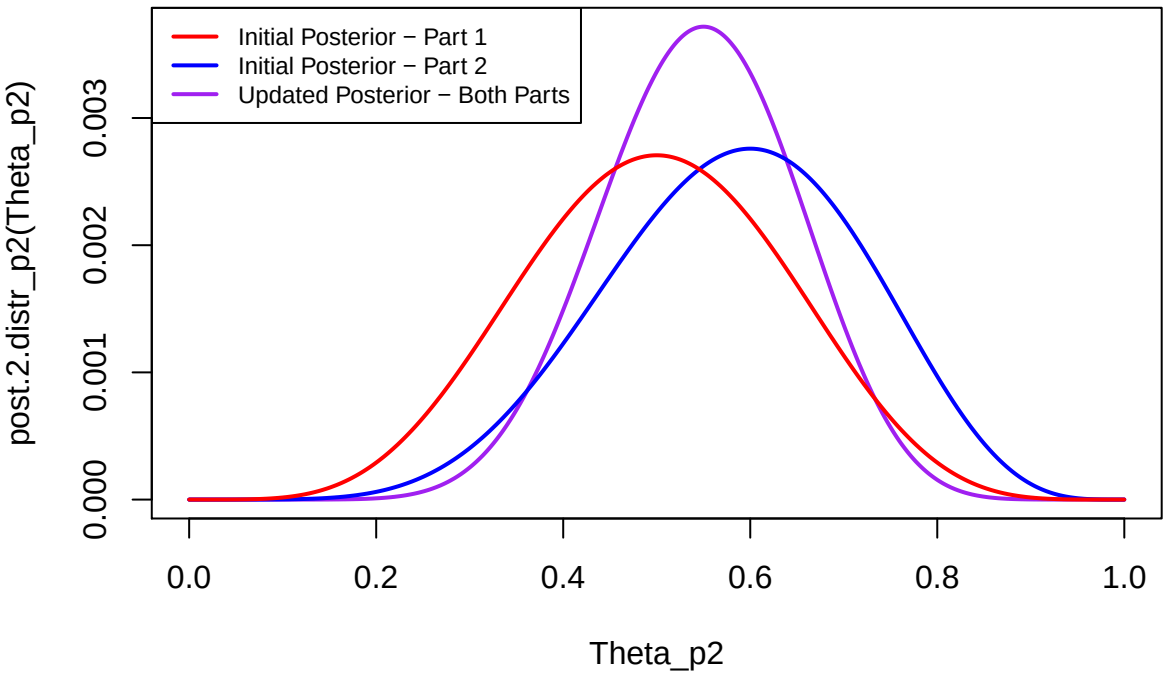


PART 3: Comparison of updated posteriors from Parts 1 and 2

```
# Comparison of Mean, Mode, and Variance of Updated Posteriors
df<-data.frame(rbind(c(mode2,mean2,var2),c(mode2_p2,mean2_p2,var2_p2)))
names(df)<-c('Mode','Mean','Variance')
df
```

```
##       Mode      Mean   Variance
## 1 0.549999 0.5454545 0.01077973
## 2 0.549999 0.5454545 0.01077973
```

```
### Comparison of Both Updated Posterior
plot(Theta_p2,post.2.distr_p2(Theta_p2),col='purple',type='l',lwd=2)
lines(Theta,post.1.distr(Theta),col='blue',type='l',lwd=2)
lines(Theta_p2,post.1.distr_p2(Theta_p2),col='red',type='l',lwd=2)
legend("topleft",c("Initial Posterior - Part 1","Initial Posterior - Part 2",
                   "Updated Posterior - Both Parts"),
       lty=c(1,1),lwd=c(2,2),col=c("red","blue","purple"),cex=0.75)
```



Seen above, the updated posteriors for both Parts 1 & 2 are identical. This is to be expected, because each method utilized the same data and initial prior. The only difference was the order in which the data were applied. However, in the end, the order doesn't matter and the resulting posteriors are the same.