Assignment3

2023-04-20

Ex 1

```r
library(bayestestR)
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✔ dplyr     1.1.0     ✔ readr     2.1.4
## ✔ forcats   1.0.0     ✔ stringr   1.5.0
## ✔ ggplot2   3.4.1     ✔ tibble    3.2.0
## ✔ lubridate 1.9.2     ✔ tidyr     1.3.0
## ✔ purrr     1.0.1
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the ]8;;http://conflicted.r-lib.org/conflicted package]8;; to force all conflicts to become errors
```

```r
n <- 20
r <- 7

prob <- seq(from =0.1, to =1000, by=1)/1000
like <- dbinom(x=r, size=n, prob)

Fprior <- 1
Jprior <- dbeta(x = prob, shape1 = 1/2 , shape2 = 1/2)
Sprior <- function(x){
  f <- c()
  for (i in x){
      if(i<0.2){
        f <- append(f,i)
      }
      else if(0.2<i & i<0.3){
        f <- append(f,0.2)
      }
      else if(0.3<i & i<0.5){
        f <- append(f,0.5-i)
      }
      else{
        f <- append(f,0)
      }
  }
  return(f)
}
conf <- function(x,delta,c){
  p <- 0
  j <- 0
  low <- 0
  high <- 0
  Z = sum(x*delta)
  for (i in x){
    p <- p+i*delta/Z
    if (p>=(1-c)/2){
      low <- j
      break
    }
    j <- j+1
  }
  p <- 0
  j <- 0
  for (i in x){
    p <- p+i*delta/Z
    if (p>=c+(1-c)/2){
      high <- j
      break
    }
    j <- j+1
  }
  return(c(low*delta,high*delta))
}
PartFunc <- function(x,delta){
  return(sum(x*1/1000))
}

sprior <- Sprior(prob)
ZF <- PartFunc(like*Fprior,1/1000)
ZJ <- PartFunc(like*Jprior,1/1000)
ZS <- PartFunc(like*sprior,1/1000)
Fpost <- like*Fprior/(ZF)
Jpost <- like*Jprior/(ZJ)
Spost <- like*sprior/(ZS)

cat("The confidence interval for the posterior of the Unif prior is ",conf(Fpost,1/1000, 0.95),"\n")
```

```
## The confidence interval for the posterior of the Unif prior is  0.181 0.57
```

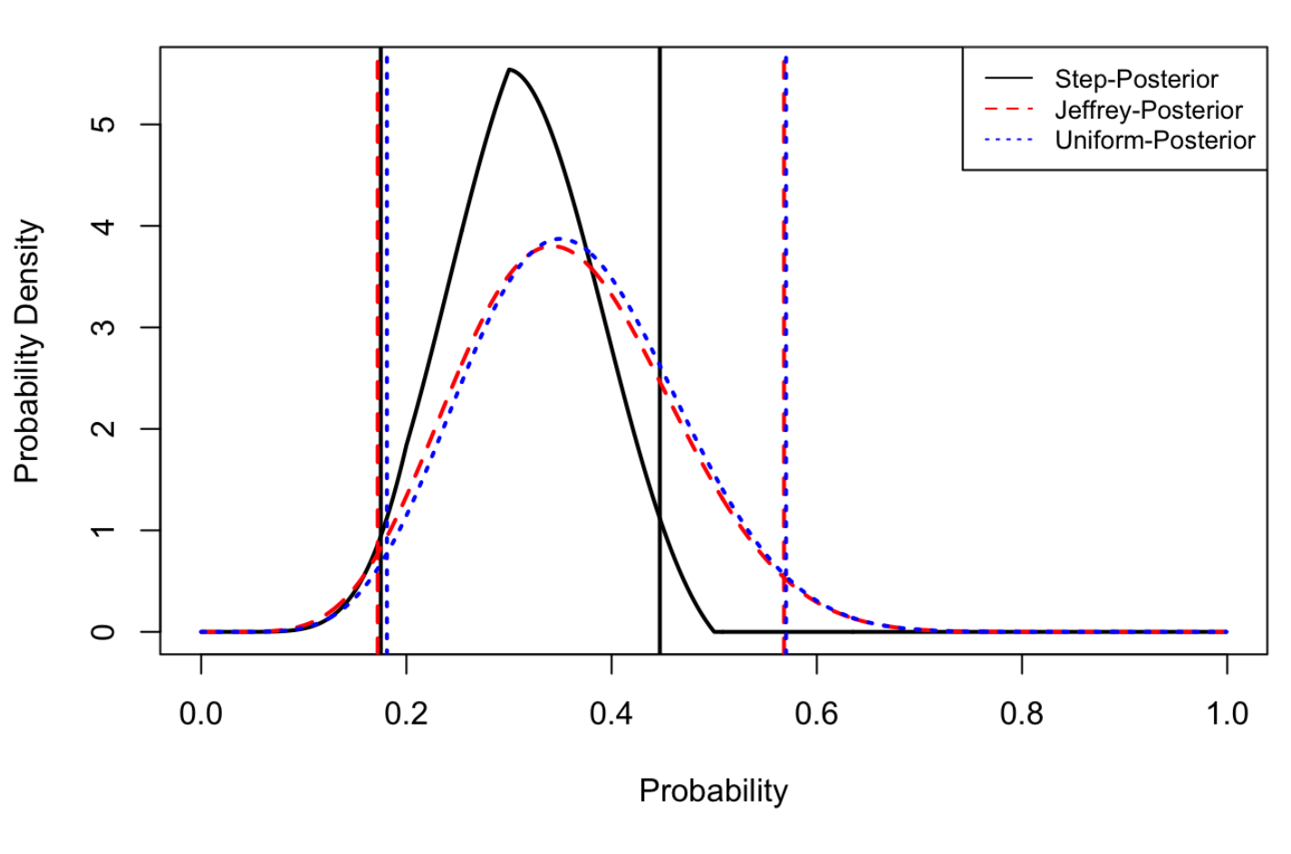
```r
cat("The confidence interval for the posterior of the Jeffrey prior is",conf(Jpost,1/1000, 0.95),"\n")
```

```
## The confidence interval for the posterior of the Jeffrey prior is 0.172 0.568
```

```r
cat("The confidence interval for the posterior of the Step prior is",conf(Spost,1/1000, 0.95),"\n")
```

```
## The confidence interval for the posterior of the Step prior is 0.175 0.447
```

```r
plot(prob,Spost, col="black",type="l",lwd=2 ,lty=1,xlab = "Probability",ylab = "Probability Density")
abline(v=conf(Spost,1/1000, 0.95), col=c("black", "black"), lty=c(1,1), lwd=c(2, 2))

lines(prob,Jpost, col="red",lty=2,lwd=2)
abline(v=conf(Jpost,1/1000, 0.95), col=c("red", "red"), lty=c(2,2), lwd=c(2, 2))

lines(prob,Fpost, col="blue",lty=3,lwd=2)
abline(v=conf(Fpost,1/1000, 0.95), col=c("blue", "blue"), lty=c(3,3), lwd=c(2, 2))

legend(x="topright",legend=c("Step-Posterior","Jeffrey-Posterior","Uniform-Posterior"), col=c("black","red","blu
e"),lty=c(1,2,3), cex=0.8)
```



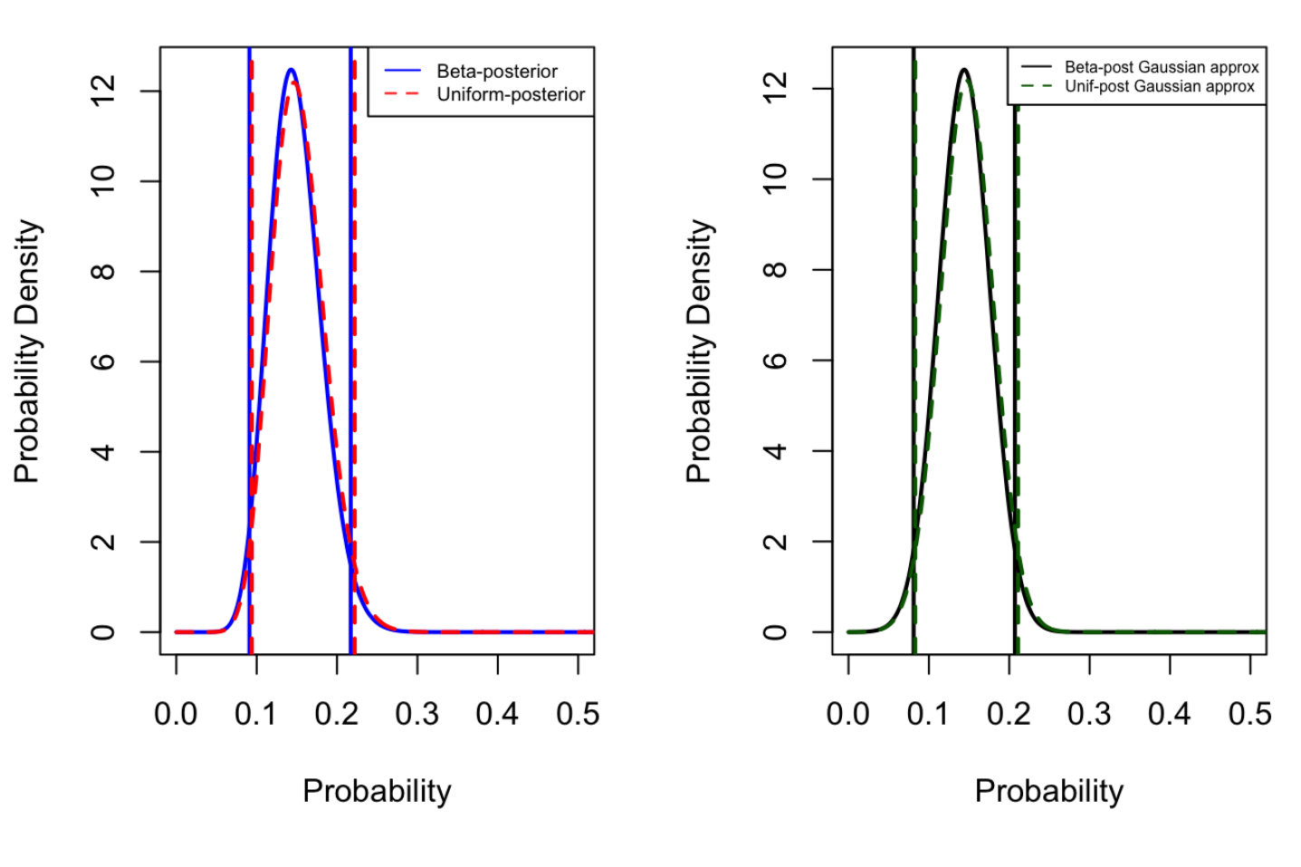
```r
x <- c("Mean","Lower 95% bound","Upper 95% bound")
jeffrey_post <- c(which.max(Jpost)*1/1000,conf(Jpost,1/1000, 0.95))
uniform_post <- c(which.max(Fpost)*1/1000,conf(Fpost,1/1000, 0.95))
step_post <- c(which.max(Spost)*1/1000,conf(Spost,1/1000, 0.95))
summ <- data.frame(x,jeffrey_post, uniform_post, step_post)
print(summ)
```

```
##                 x jeffrey_post uniform_post step_post
## 1            Mean        0.343        0.351     0.301
## 2 Lower 95% bound        0.172        0.181     0.175
## 3 Upper 95% bound        0.568        0.570     0.447
```

Ex 2

```r
n <- 116
r <- 17
N <- 1000
prob <- seq(from =0.1, to =N, by=1)/N
like <- dbinom(x=r, size=n, prob)

Uprior <- 1
Bprior <- dbeta(prob, shape1 = 1, shape2 = 4)

Zu <- PartFunc(like*Uprior, 1/N)
Zb <- PartFunc(like*Bprior, 1/N)

Upost <- like*Uprior/Zu
Bpost <- like*Bprior/Zb

Ufirst <- sum(prob*Upost*1/N)
Usecond <- sum(prob*prob*Upost*1/N)
Umax_ind <- which.max(Upost)
Usig <- 1/sqrt(-1*(log(Upost[Umax_ind-1])-2*log(Upost[Umax_ind])+log(Upost[Umax_ind+1]))/(1/(N*N)))

Bfirst <- sum(prob*Bpost*1/N)
Bsecond <- sum(prob*prob*Bpost*1/N)
Bmax_ind <- which.max(Bpost)
Bsig <- 1/sqrt(-1*(log(Bpost[Bmax_ind-1])-2*log(Bpost[Bmax_ind])+log(Bpost[Bmax_ind+1]))/(1/(N*N)))


UGauss <- dnorm(prob, mean =Umax_ind*1/N, sd= Usig)
BGauss <- dnorm(prob, mean =Bmax_ind*1/N, sd= Bsig)
```

```r
par(mfrow=c(1,2))

plot(prob,Bpost, col="blue",type="l",lwd=2 ,lty=1, xlim=c(0,0.5),xlab = "Probability",ylab = "Probability Densit
y")
abline(v=conf(Bpost,1/1000, 0.95), col=c("blue", "blue"), lty=c(1,1), lwd=c(2, 2))

lines(prob,Upost, col="red",lty=2,lwd=2)
abline(v=conf(Upost,1/1000, 0.95), col=c("red", "red"), lty=c(2,2), lwd=c(2, 2))
legend(x="topright",legend=c("Beta-posterior","Uniform-posterior"), col=c("blue","red"),lty=c(1,2), cex=0.6)

plot(prob,BGauss, col="black",type="l",lwd=2 ,lty=1, xlim=c(0,0.5),xlab = "Probability",ylab = "Probability Densi
ty")
abline(v=conf(BGauss,1/1000, 0.95), col=c("black", "black"), lty=c(1,1), lwd=c(2, 2))

lines(prob,UGauss, col="darkgreen",lty=2,lwd=2)
abline(v=conf(UGauss,1/1000, 0.95), col=c("darkgreen", "darkgreen"), lty=c(2,2), lwd=c(2, 2))
legend(x="topright",legend=c("Beta-post Gaussian approx","Unif-post Gaussian approx"), col=c("black","darkgree
n"),lty=c(1,2), cex=0.5)
```



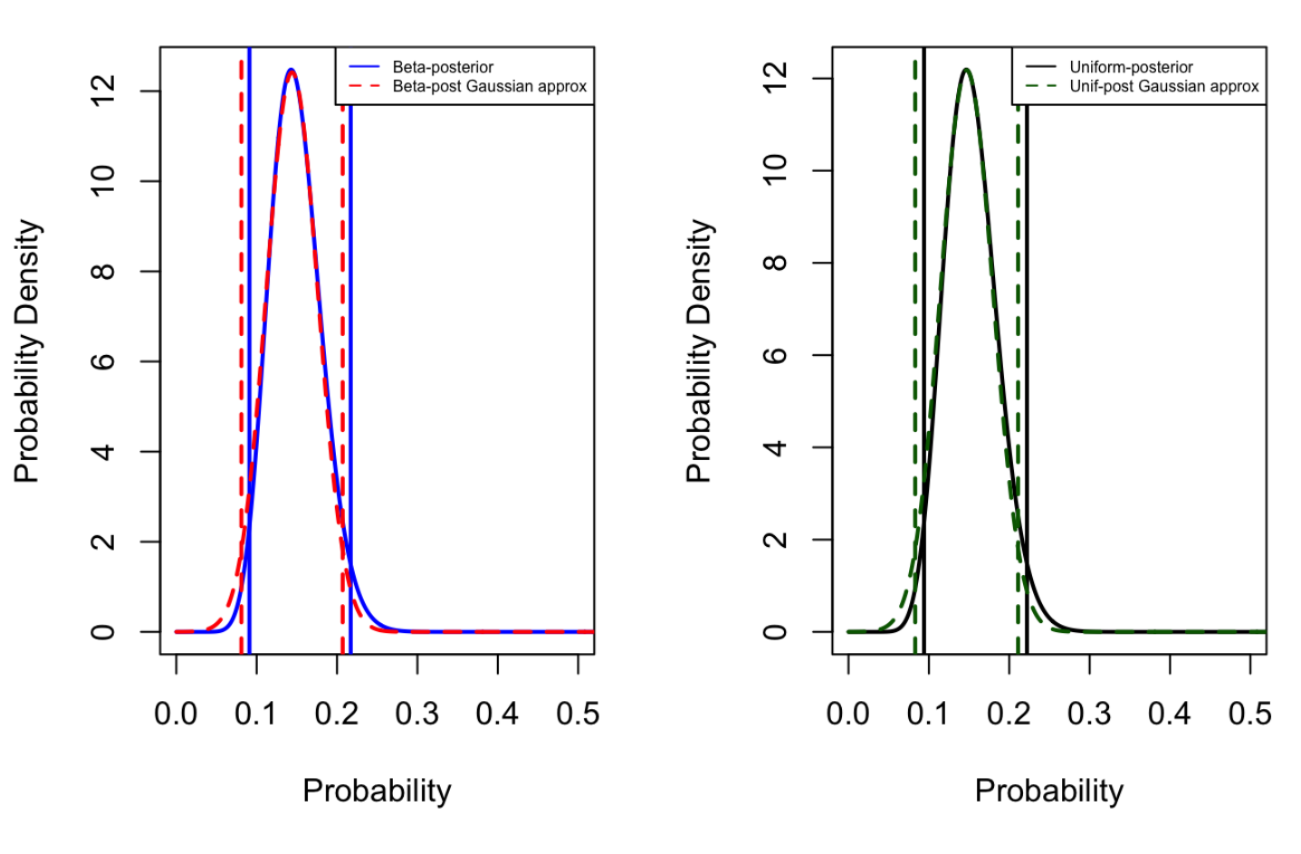
```r
cat("The first and the second moment of the posterior of the unif prior are:", Ufirst,",", Usecond, "\n")
```

```
## The first and the second moment of the posterior of the unif prior are: 0.1525424 , 0.0243555
```

```r
cat("The first and the second moment of the posterior of the beta prior are:", Bfirst," ", Bsecond, "\n")
```

```
## The first and the second moment of the posterior of the beta prior are: 0.1487603   0.02316759
```

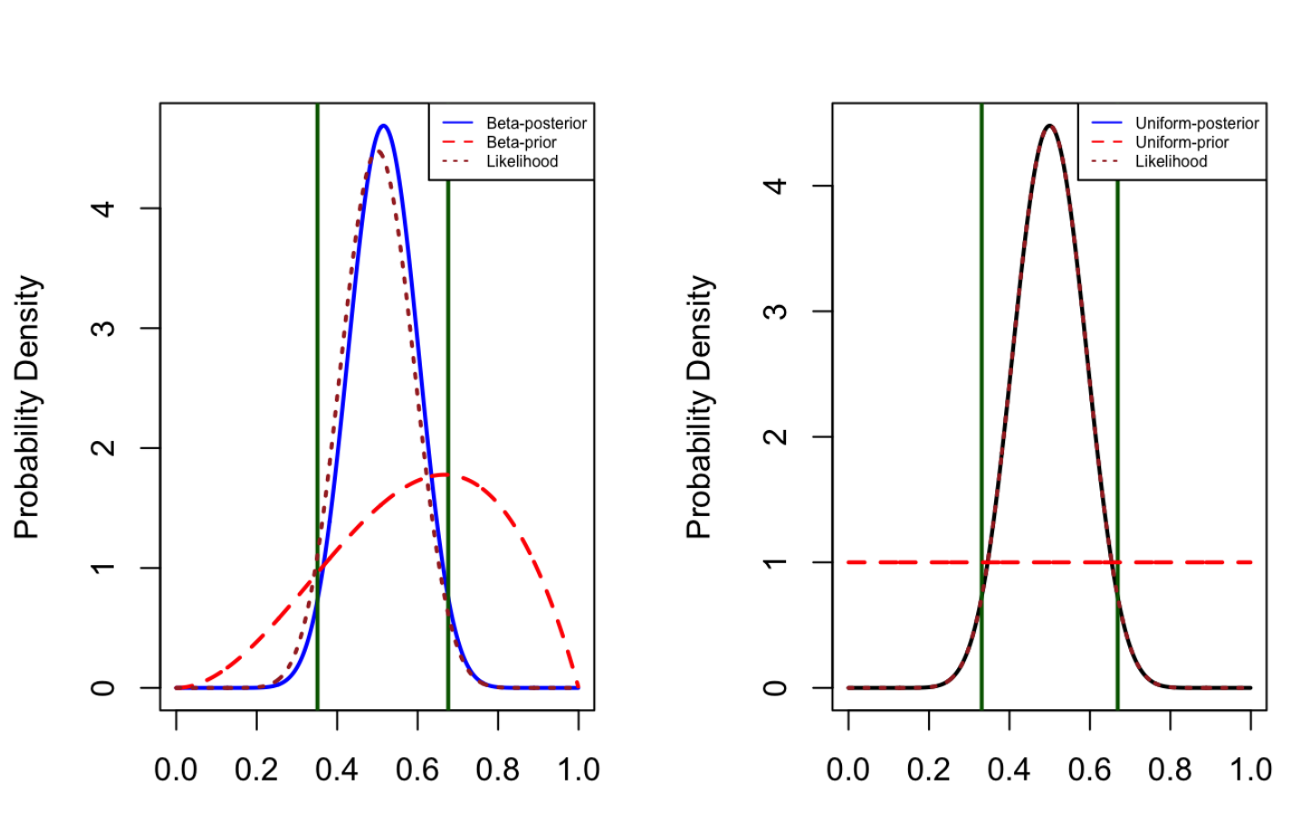
```r
par(mfrow=c(1,2))

plot(prob,Bpost, col="blue",type="l",lwd=2 ,lty=1, xlim=c(0,0.5),xlab = "Probability",ylab = "Probability Densit
y")
abline(v=conf(Bpost,1/1000, 0.95), col=c("blue", "blue"), lty=c(1,1), lwd=c(2, 2))

lines(prob,BGauss, col="red",lty=2,lwd=2)
abline(v=conf(BGauss,1/1000, 0.95), col=c("red", "red"), lty=c(2,2), lwd=c(2, 2))
legend(x="topright",legend=c("Beta-posterior","Beta-post Gaussian approx"), col=c("blue","red"),lty=c(1,2), cex=
0.5)

plot(prob,Upost, col="black",type="l",lwd=2 ,lty=1, xlim=c(0,0.5),xlab = "Probability",ylab = "Probability Densit
y")
abline(v=conf(Upost,1/1000, 0.95), col=c("black", "black"), lty=c(1,1), lwd=c(2, 2))

lines(prob,UGauss, col="darkgreen",lty=2,lwd=2)
abline(v=conf(UGauss,1/1000, 0.95), col=c("darkgreen", "darkgreen"), lty=c(2,2), lwd=c(2, 2))
legend(x="topright",legend=c("Uniform-posterior","Unif-post Gaussian approx"), col=c("black","darkgreen"),lty=c
(1,2), cex=0.5)
```



Ex 3

```r
n <- 30
r <- 15
N <- 1000
prob <- seq(from = 0.1, to = N, by = 1)/N
like <- dbinom(x=r, size=n, prob)
like <- like/PartFunc(like,1/N)


Uprior <- rep(1,N)
Bprior <- dbeta(prob, shape1 = 3, shape2 = 2)

Zu <- PartFunc(like*Uprior, 1/N)
Zb <- PartFunc(like*Bprior, 1/N)

Upost <- like*Uprior/Zu
Bpost <- like*Bprior/Zb

Ufirst <- sum(prob*Upost*1/N)
Usecond <- sum(prob*prob*Upost*1/N)

Bfirst <- sum(prob*Bpost*1/N)
Bsecond <- sum(prob*prob*Bpost*1/N)
```

```r
par(mfrow=c(1,2))

plot(prob,Bpost, col="blue",type="l",lwd=2 ,lty=1,xlab = "Probability",ylab = "Probability Density")
abline(v=conf(Bpost,1/1000, 0.95), col=c("darkgreen", "darkgreen"), lty=c(1,1), lwd=c(2, 2))

lines(prob,Bprior, col="red",lty=2,lwd=2)
lines(prob,like, col="brown",lty=3,lwd=2)
legend(x="topright",legend=c("Beta-posterior","Beta-prior","Likelihood"), col=c("blue","red","brown"),lty=c(1,2,
3), cex=0.5)


plot(prob,Upost, col="black",type="l",lwd=2 ,lty=1,xlab = "Probability",ylab = "Probability Density")
abline(v=conf(Upost,1/1000, 0.95), col=c("darkgreen", "darkgreen"), lty=c(1,1), lwd=c(2, 2))

lines(prob,Uprior, col="red",lty=2,lwd=2)
lines(prob,like, col="brown",lty=3,lwd=2)
legend(x="topright",legend=c("Uniform-posterior","Uniform-prior","Likelihood"), col=c("blue","red","brown"),lty=c
(1,2,3), cex=0.5)
```



```r
cat("The first and the second moment of the posterior of the unif prior are:", Ufirst,",", Usecond, "\n")
```

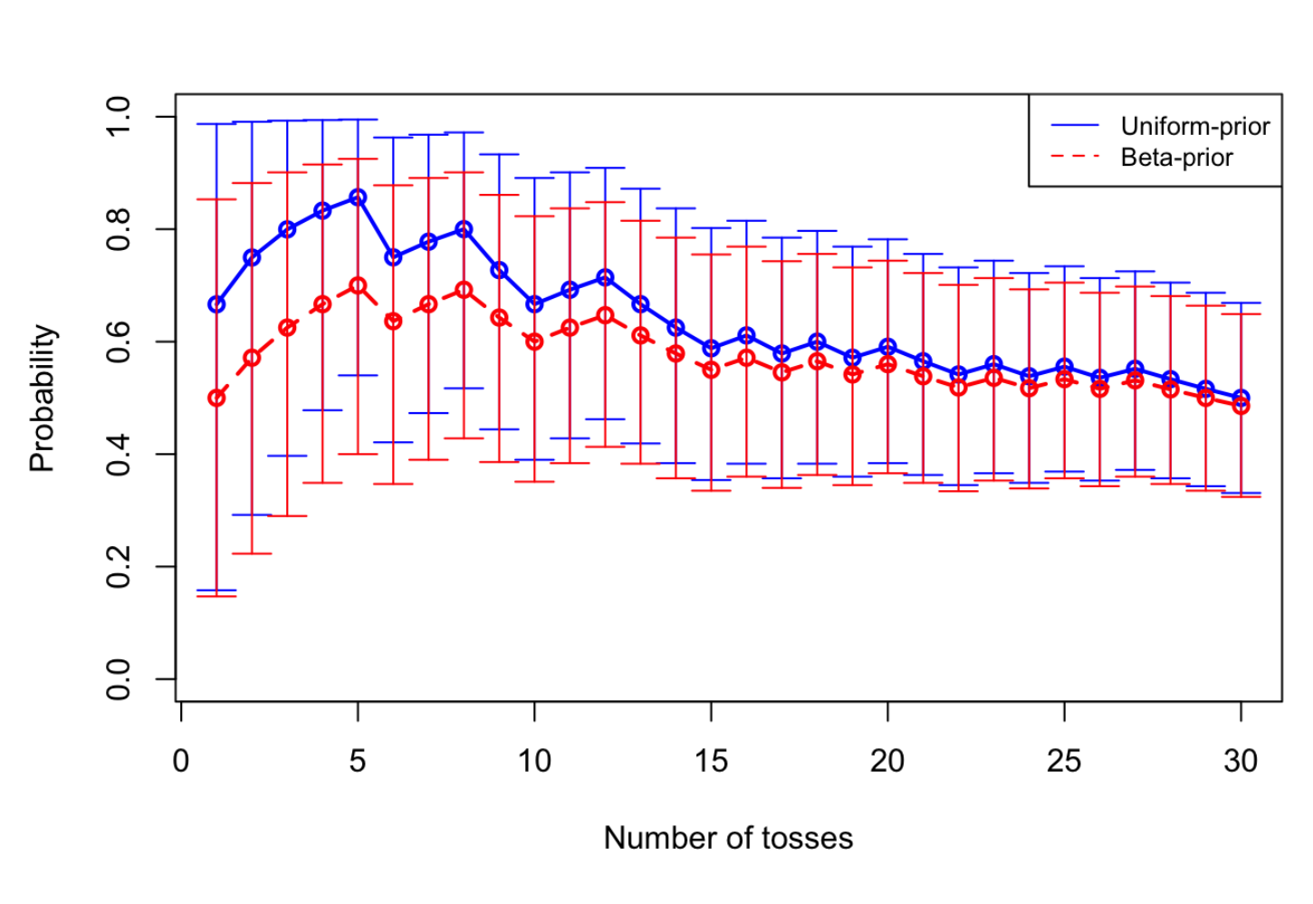
```
## The first and the second moment of the posterior of the unif prior are: 0.5 , 0.2575758
```

```r
cat("The first and the second moment of the posterior of the beta prior are:", Bfirst," ", Bsecond, "\n")
```

```
## The first and the second moment of the posterior of the beta prior are: 0.5142857   0.2714286
```

```r
tosses <- c(1, 1, 1, 1, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0,0, 0)
Uvalues <- c()
Bvalues <- c()
ULowt <- c()
UUpt <- c()
BLowt <- c()
BUpt <- c()
r <- 0
vers1 = FALSE
for (i in 1:length(tosses)){
  n <- 1
  r <- tosses[i]
  N <- 1000

  prob <- seq(from = 0.1, to = N, by = 1)/N
  like <- dbinom(x=r, size=n, prob)
  like <- like/PartFunc(like,1/N)

  if (i==1){
    Uprior <- 1
    Bprior <- dbeta(prob, shape1 = 2, shape2 = 3)
  }

  Zu <- PartFunc(like*Uprior, 1/N)
  Zb <- PartFunc(like*Bprior, 1/N)

  Upost <- like*Uprior/Zu
  Bpost <- like*Bprior/Zb

  Ufirst <- sum(prob*Upost*1/N)
  Usecond <- sum(prob*prob*Upost*1/N)
  Uvalues <- append(Uvalues, Ufirst)
  ULowt <- append(ULowt,conf(Upost,1/N,0.95)[1])
  UUpt <- append(UUpt,conf(Upost,1/N,0.95)[2])

  Bfirst <- sum(prob*Bpost*1/N)
  Bsecond <- sum(prob*prob*Bpost*1/N)
  Bvalues <- append(Bvalues, Bfirst)
  BLowt <- append(BLowt,conf(Bpost,1/N,0.95)[1])
  BUpt <- append(BUpt,conf(Bpost,1/N,0.95)[2])

  Uprior <- Upost
  Bprior <- Bpost
}
```

```r
Num_toss <- seq(from=1, to=length(Uvalues), by=1)

plot(Num_toss,Uvalues, col="blue",type="o",lwd=2 ,lty=1, ylim=c(0,1), ylab = "Probability", xlab = "Number of tos
ses")
arrows(x0=Num_toss, y0=ULowt, x1=Num_toss, y1=UUpt, code=3, angle=90, length=0.1, col = 'blue')
lines(Num_toss,Bvalues, col="red",type="o",lty=2,lwd=2)
arrows(x0=Num_toss, y0=BLowt, x1=Num_toss, y1=BUpt, code=3, angle=90, length=0.1, col = 'red')
legend(x="topright",legend=c("Uniform-prior","Beta-prior"), col=c("blue","red"),lty=c(1,2), cex=0.8)
```

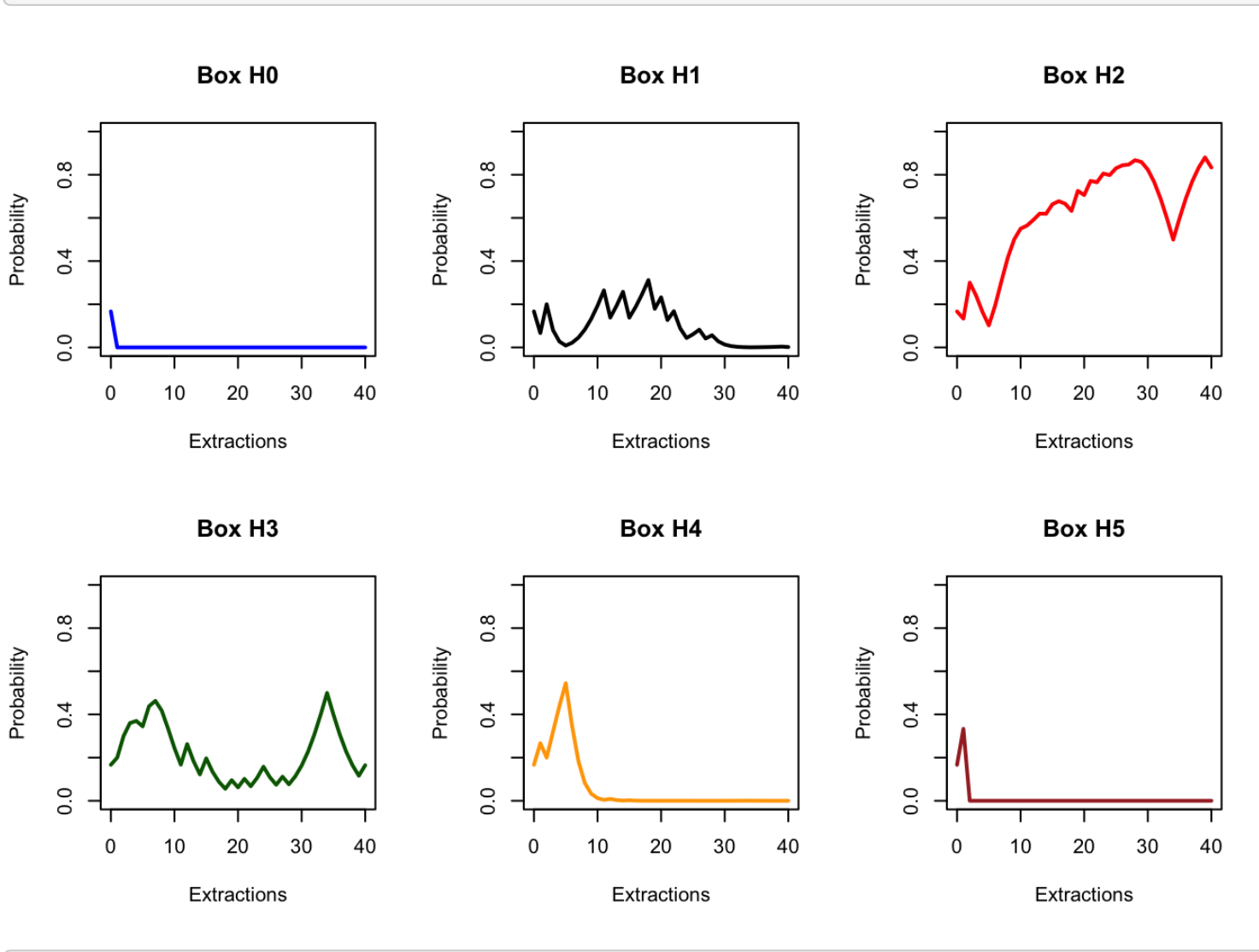


```r
cat("The result of the sequential analysis is", Bvalues[length(Bvalues)] ,Uvalues[length(Uvalues)], "with respect
to 0.5 of a one-step analysis, so obviously they are the same")
```

```
## The result of the sequential analysis is 0.4857143 0.5 with respect to 0.5 of a one-step analysis, so obviousl
y they are the same
```

Ex 4

```r
mat <- cbind(c(1,1,1,1,1),c(1,1,1,1,0),c(1,1,1,0,0),c(1,1,0,0,0),c(1,0,0,0,0),c(0,0,0,0,0))

box <- sample(1:6,1)
ph_e <- c(1,1,1,1,1,1)/6
ph_et <- c(1,1,1,1,1,1)/6
e <- 40
ex_balls <- c("N")
for (j in 1:e){
  ball <- sample(1:5,1)
  ext <- mat[ball,box]
  if(ext == 1){
    ex_balls <- append(ex_balls,"B")
  }
  else{
    ex_balls <- append(ex_balls,"W")
  }

  for (i in 1:6){
    ph_e[i] <- length(mat[,i][mat[,i] == ext])/5 * ph_e[i]
  }
  ph_e <- ph_e/sum(ph_e)
  ph_et <- rbind(ph_et,ph_e)
}

par(mfrow=c(2,3))

Extractions <- seq(from =0, to = e, by=1)

plot(Extractions,ph_et[,1], col="blue",type="l",lwd=2 , ylim=c(0,1) ,lty=1,main = "Box H0", ylab = "Probability")

plot(Extractions,ph_et[,2], col="black",type="l",lwd=2 , ylim=c(0,1),lty=1,main ="Box H1" , ylab = "Probability")

plot(Extractions,ph_et[,3], col="red",type="l",lwd=2 , ylim=c(0,1),lty=1,main ="Box H2", ylab = "Probability" )

plot(Extractions,ph_et[,4], col="darkgreen",type="l",lwd=2 , ylim=c(0,1),lty=1,main = "Box H3", ylab = "Probabili
ty")

plot(Extractions,ph_et[,5], col="orange",type="l",lwd=2 , ylim=c(0,1),lty=1,main ="Box H4", ylab = "Probability"
)

plot(Extractions,ph_et[,6], col="brown",type="l",lwd=2 , ylim=c(0,1),lty=1,main = "Box H5", ylab = "Probability")
```



```r
cat("The real box tossed is H",box-1,"\n")
```

```
## The real box tossed is H 2
```

```r
cat("Probabilities of each box at the",e,"th trial is\nH0 =",ph_et[e,1],"\nH1 =",ph_et[e,2],"\nH2 =",ph_et[e,
3],"\nH3 =",ph_et[e,4],"\nH4 =",ph_et[e,5],"\nH5 =",ph_et[e,6])
```

```
## Probabilities of each box at the 40 th trial is
## H0 = 0
## H1 = 0.003765096
## H2 = 0.8803061
## H3 = 0.1159251
## H4 = 3.676851e-06
## H5 = 0
```

```r
library(tidyverse)
df <- data.frame(round(ph_et,3))
Extraction <- seq(from=0, to=e, by=1)
df <- cbind(df, Extraction)
df <- cbind(df, ex_balls)
df <- rename(df,H0=X1,
             H1=X2,
             H2=X3,
             H3=X4,
             H4=X5,
             H5=X6,
             BALL=ex_balls)
print(df)
```

```
##             H0    H1    H2    H3    H4    H5 Extraction BALL
## ph_et   0.167 0.167 0.167 0.167 0.167 0.167          0    N
## ph_e    0.000 0.067 0.133 0.200 0.267 0.333          1    W
## ph_e.1  0.000 0.200 0.300 0.300 0.200 0.000          2    B
## ph_e.2  0.000 0.080 0.240 0.360 0.320 0.000          3    W
## ph_e.3  0.000 0.027 0.164 0.370 0.438 0.000          4    W
## ph_e.4  0.000 0.009 0.102 0.345 0.545 0.000          5    W
## ph_e.5  0.000 0.022 0.195 0.438 0.346 0.000          6    B
## ph_e.6  0.000 0.046 0.309 0.463 0.183 0.000          7    B
## ph_e.7  0.000 0.082 0.418 0.418 0.082 0.000          8    B
## ph_e.8  0.000 0.132 0.501 0.334 0.033 0.000          9    B
## ph_e.9  0.000 0.193 0.550 0.245 0.012 0.000         10    B
## ph_e.10 0.000 0.264 0.564 0.167 0.004 0.000         11    B
## ph_e.11 0.000 0.138 0.591 0.262 0.009 0.000         12    W
## ph_e.12 0.000 0.193 0.620 0.184 0.003 0.000         13    B
## ph_e.13 0.000 0.258 0.619 0.122 0.001 0.000         14    B
## ph_e.14 0.000 0.138 0.663 0.197 0.002 0.000         15    W
## ph_e.15 0.000 0.188 0.677 0.134 0.001 0.000         16    B
## ph_e.16 0.000 0.246 0.666 0.088 0.000 0.000         17    B
## ph_e.17 0.000 0.312 0.632 0.056 0.000 0.000         18    B
## ph_e.18 0.000 0.179 0.725 0.096 0.000 0.000         19    W
## ph_e.19 0.000 0.232 0.706 0.062 0.000 0.000         20    B
## ph_e.20 0.000 0.127 0.771 0.102 0.000 0.000         21    W
## ph_e.21 0.000 0.168 0.765 0.067 0.000 0.000         22    B
## ph_e.22 0.000 0.088 0.806 0.106 0.000 0.000         23    W
## ph_e.23 0.000 0.044 0.798 0.158 0.000 0.000         24    W
## ph_e.24 0.000 0.061 0.830 0.109 0.000 0.000         25    B
## ph_e.25 0.000 0.082 0.844 0.074 0.000 0.000         26    B
## ph_e.26 0.000 0.041 0.847 0.112 0.000 0.000         27    W
## ph_e.27 0.000 0.056 0.867 0.076 0.000 0.000         28    B
## ph_e.28 0.000 0.028 0.859 0.113 0.000 0.000         29    W
## ph_e.29 0.000 0.013 0.824 0.163 0.000 0.000         30    W
## ph_e.30 0.000 0.006 0.767 0.227 0.000 0.000         31    W
## ph_e.31 0.000 0.003 0.690 0.307 0.000 0.000         32    W
## ph_e.32 0.000 0.001 0.599 0.399 0.000 0.000         33    W
## ph_e.33 0.000 0.001 0.499 0.499 0.001 0.000         34    W
## ph_e.34 0.000 0.001 0.599 0.400 0.000 0.000         35    B
## ph_e.35 0.000 0.001 0.691 0.307 0.000 0.000         36    B
## ph_e.36 0.000 0.002 0.770 0.228 0.000 0.000         37    B
## ph_e.37 0.000 0.003 0.833 0.165 0.000 0.000         38    B
## ph_e.38 0.000 0.004 0.880 0.116 0.000 0.000         39    B
## ph_e.39 0.000 0.002 0.834 0.165 0.000 0.000         40    W
```