BayesExam

2022-10-20

Problem 1

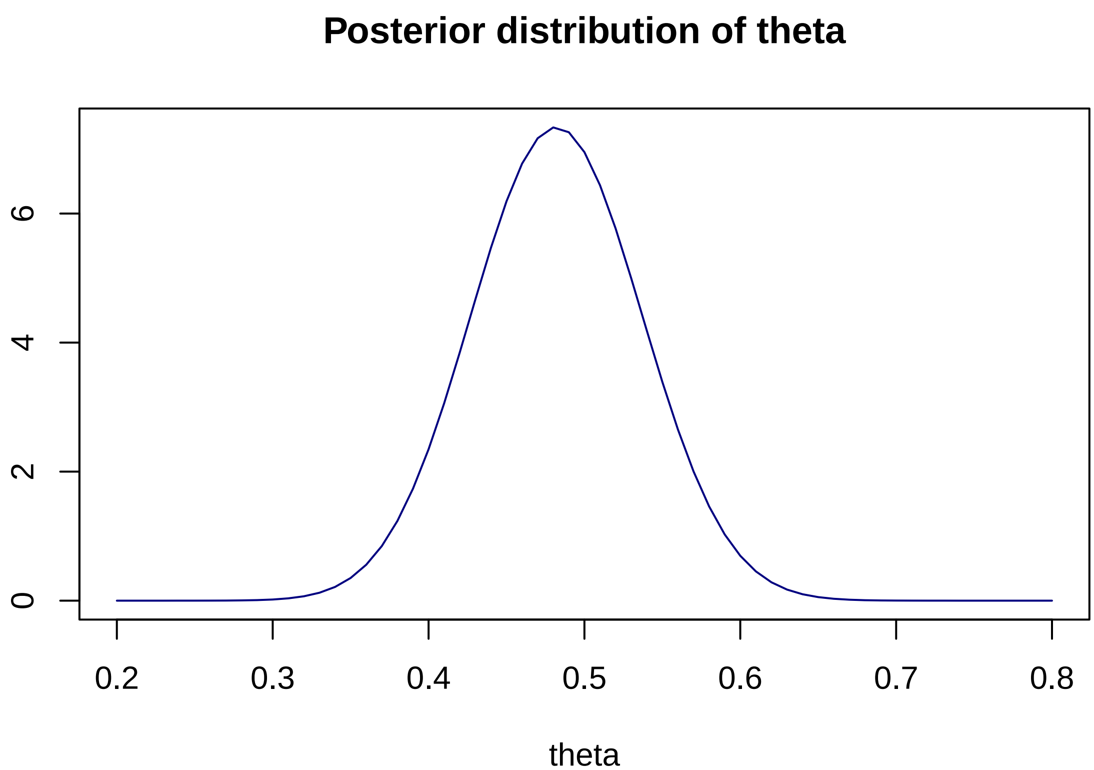
Task d

```
LogPost <- function(theta,n,N,sumx){
  LogLik <- sumx*log(theta) + (n*N-sumx)*log(1-theta)
  LogPrior <- log(theta) + 2*log(1-theta)
  res <- LogLik + LogPrior
  return(res)
}

obs <- c(13,8,11,7)
sumx <- sum(obs)
n <- 4
N <- 20
thetaGrid <- seq(0.2,0.8,0.01)

LogPost_propto <- exp(LogPost(thetaGrid,n,N,sumx))
LogPost_Dens <- LogPost_propto/(0.01*sum(LogPost_propto))


plot(thetaGrid,LogPost_Dens, type = "l", col = "navy",
     main = "Posterior distribution of theta", xlab = "theta", ylab = "")
```



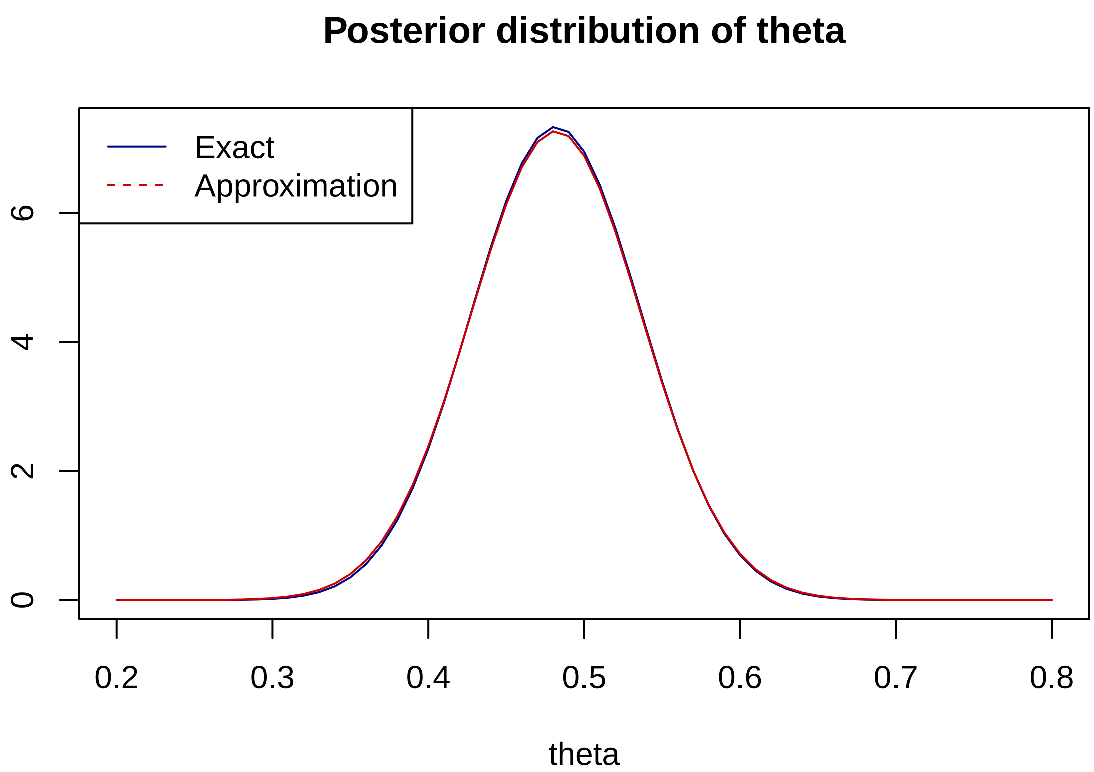
Task e

```
set.seed(12345)

OptimRes <- optim(0.85, LogPost, gr = NULL, n, N, sumx,
                  method = c("L-BFGS-B"), control = list(fnscale = -1),
                  lower = 0.2, hessian = TRUE)

approx <- dnorm(thetaGrid, OptimRes$par, sqrt(diag(-solve(OptimRes$hessian))))

plot(thetaGrid,LogPost_Dens, type = "l", col = "navy",
     main = "Posterior distribution of theta", xlab = "theta", ylab = "")
lines(thetaGrid,approx, col = "red3")
legend("topleft", legend = c("Exact", "Approximation"), col = c("navy", "red3"),
       lty = 1:2)
```



The posterior approximation is very accurate.

Problem 2

```
source("ExamData.R")
```

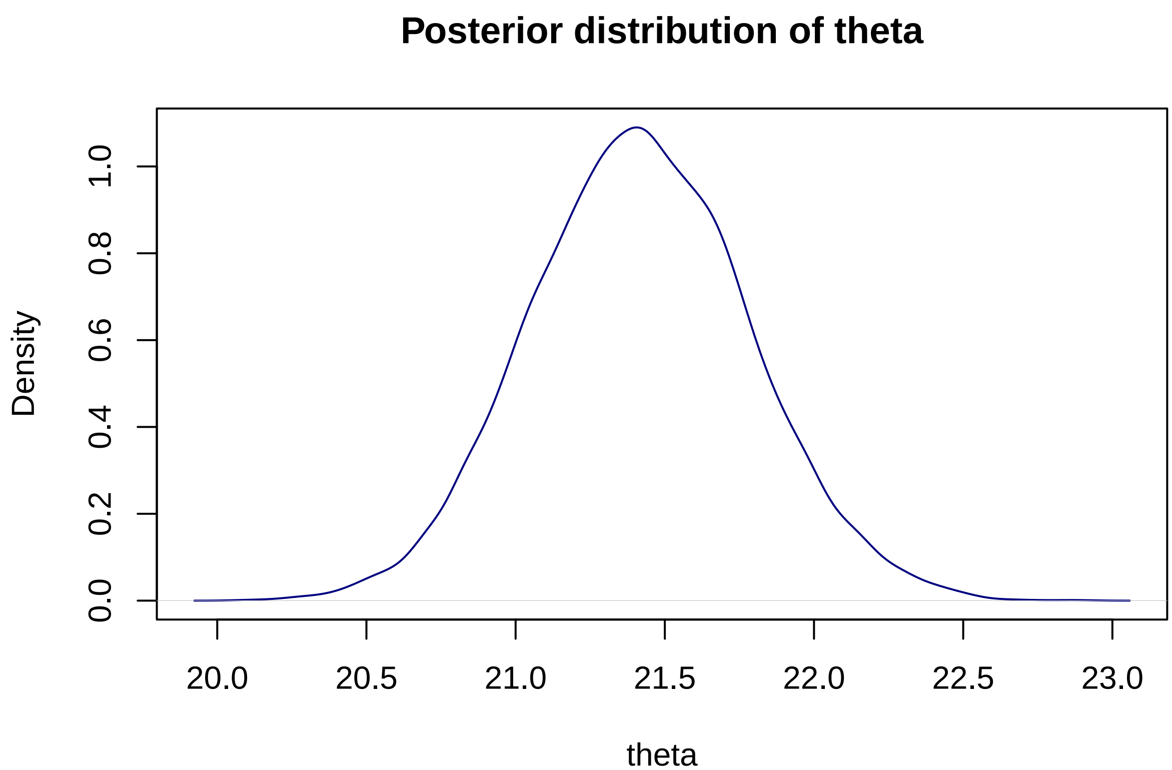
Task b

```
set.seed(12345)

nDraws <- 10000
n <- 156
a <- 40
sumy <- sum(Wolves[,2])

theta <- rgamma(nDraws,a+sumy,n+2)

plot(density(theta), col = "navy",
     main = "Posterior distribution of theta",
     xlab = "theta")
```




```
prob2b <- mean(theta < 21)
```

The posterior probability that θ is smaller than 21 is 0.1272.

Task c

```
set.seed(12345)

indexA <- which(Wolves[,1]==1)
regionA <- Wolves[indexA,]

indexB <- which(Wolves[,1]==0)
regionB <- Wolves[indexB,]

thetaA <- rgamma(nDraws,a + sum(regionA[,2]), length(indexA) + 2)
yvalsA <- rpois(nDraws,thetaA)

thetaB <- rgamma(nDraws,a + sum(regionB[,2]), length(indexB) + 2)
yvalsB <- rpois(nDraws,thetaB)

prob2c <- mean(yvalsB>yvalsA)
```

The probability that $y_B, j > y_A, j$ on a randomly picked future week j is 0.6871.

Task d

The result from task 2c indicate that in a future week there is approximately a 68% probability that more wolves would appear in region B. From the given data, it could be assumed that it could be expected, because 99 observations out of 156 are placed in region B, which is 2/3 of the data. Thus, the assumption made from the wolf expert is correct.

Problem 3

Task a

```
set.seed(12345)

mu_0 <- as.vector(rep(0,14))
Omega_0 <- (1/100)*diag(14)
v_0 <- 1
sigma2_0 <- 16
nIter <- 10000

PostDraws <- BayesLinReg(y, X, mu_0, Omega_0, v_0, sigma2_0, nIter)
```

```
Betas <- PostDraws$betaSample
```

```
medianBetas <- matrix(0,14,1)
for (i in 1:14) {
 medianBetas[i,] <- median(Betas[,i])
}

medianBetas <- as.data.frame(medianBetas)
colnames(medianBetas) <- c("Median Values")
rownames(medianBetas) <- covNames
knitr::kable(medianBetas)
```

	Median Values
intercept	35.9724185
crim	-0.1080825
zn	0.0466186
indus	0.0202944
chas	2.6816863
nox	-17.5151100
rm	3.8362272
age	0.0005358
dis	-1.4672535
rad	0.3056082
tax	-0.0123877
ptratio	-0.9456100
black	0.0093943
lstat	-0.5231485

```
intervalsBetas <- matrix(0,14,2)

for (i in 1:14) {
  intervalsBetas[i,] <- quantile(Betas[,i], probs = c(0.025,0.975))
}

intervalsBetas <- data.frame(lower_bound = intervalsBetas[,1], upper_bound = intervalsBetas[,2])
colnames(intervalsBetas) <- c("Lower Bound", "Upper Bound")
rownames(intervalsBetas) <- covNames
knitr::kable(intervalsBetas)
```

	Lower Bound	Upper Bound
intercept	26.1245509	45.7564426
crim	-0.1720449	-0.0448405
zn	0.0202294	0.0727987
indus	-0.0975178	0.1403917
chas	0.9982866	4.3711828
nox	-24.9354507	-10.2264910
rm	3.0353465	4.6564832
age	-0.0254087	0.0257099
dis	-1.8591248	-1.0864302
rad	0.1767813	0.4346677
tax	-0.0197151	-0.0051103
ptratio	-1.1965399	-0.6902015
black	0.0041115	0.0146530
lstat	-0.6224810	-0.4228977

The 95% posterior probability that the regression coefficient on per capita crime rate by town is on the interval (-0.17,-0.044).

Task b

```
Sigma <- sqrt(PostDraws$sigma2Sample)

MeanSD <- mean(Sigma)
MeadianSD <- median(Sigma)
```

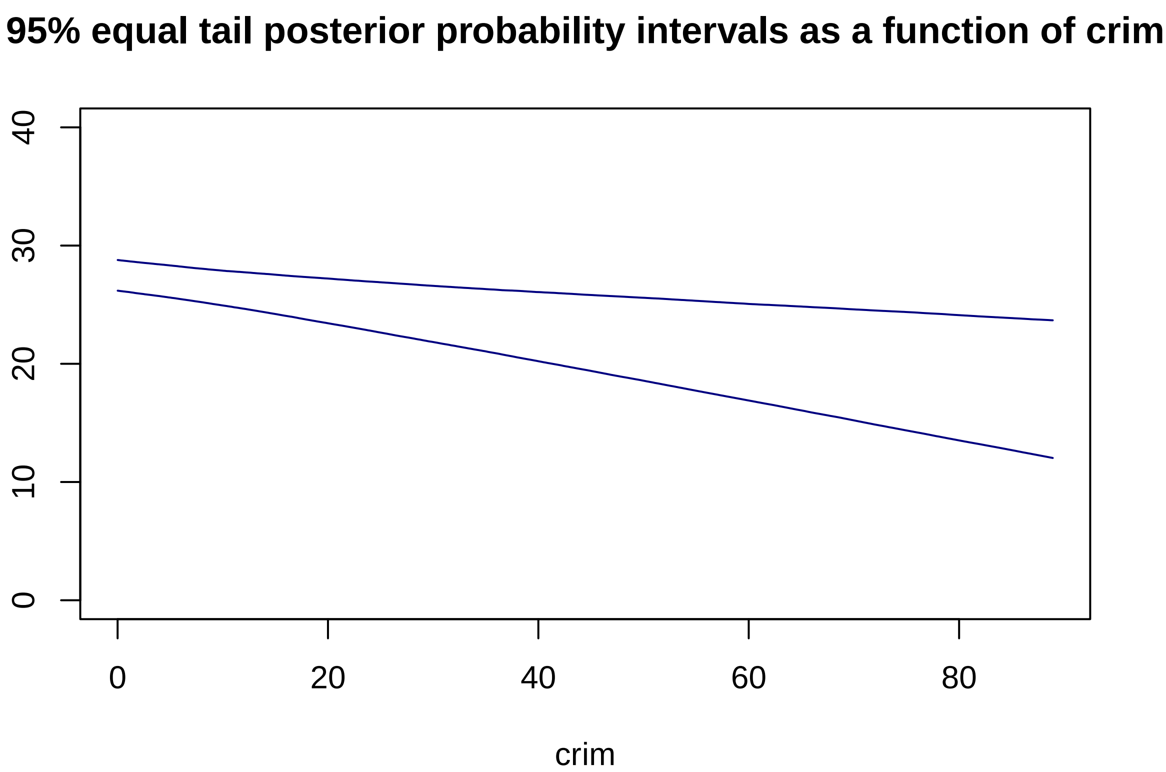
The posterior mean and posterior median of the standard deviation are approximately 4.688 and 4.683, respectively.

Task c

```
crimGrid <- seq(min(X[,2]),max(X[,2]),0.1)
intervalsMU <- matrix(0,length(crimGrid),2)


for (i in 1:length(crimGrid)){
  mu <- Betas %*% c(XNewHouse[1],crimGrid[i],XNewHouse[3:14])
  intervalsMU[i,] <- quantile(mu, probs = c(0.025,0.975))
}

plot(crimGrid, intervalsMU[,1], type = "l", col = "navy",
     main = "95% equal tail posterior probability intervals as a function of crim",
     xlab = "crim", ylab ="", ylim = c(0,40))
lines(crimGrid, intervalsMU[,2], type = "l", col = "navy")
```



Task d

```
mu3d <- XNewHouse%*%t(X)

prob3d <- mean(mu3d>20000)
```

There is 100% probability that the house would be sold for more than 20000$.

Task e

```
set.seed(12345)

nSim <- 10000
T_y_rep <- matrix(0,nSim,1)
mu <- Betas%*%t(X)

for (i in 1:nSim){
  yvals <- rnorm(length(y), mean = mu, sd =Sigma)
  T_y_rep[i,] <- max(yvals)
}

prob3e <- mean(T_y_rep > max(y))
```

The posterior predictive p-value is 0.0068. The model does not replicate the value of the most expensive house in this data, because the p-value is too small.