Solution to computer exam in Bayesian learning

Bertil Wegmann

2021-06-03

First load all the data into memory by running the R-file given at the exam

```
rm(list=ls())
source("ExamData.R")
```

```
## Warning: package 'geoR' was built under R version 4.0.5
```

```
## --------------------------------------------------------------
##  Analysis of Geostatistical Data
##  For an Introduction to geoR go to http://www.leg.ufpr.br/geoR
##  geoR version 1.8-1 (built on 2020-02-08) is now loaded
## --------------------------------------------------------------
```

```
set.seed(1)
```

Problem 1

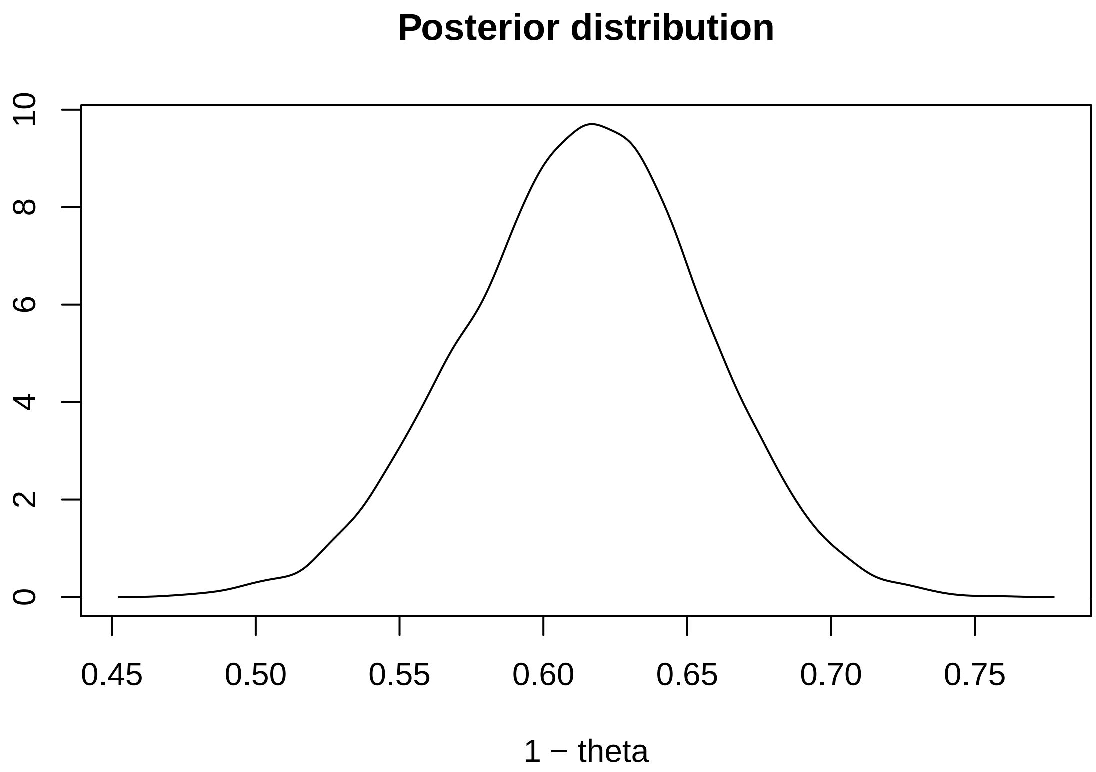
1a

```
alpha <- 16
beta <- 24
n <- 100
s <- 38
f <- n - s
post_alpha <- alpha + s
post_beta <- beta + f
theta_A <- rbeta(1e4, post_alpha, post_beta)

pbeta(0.4, post_alpha, post_beta, lower.tail = FALSE)
```

```
## [1] 0.3600383
```

```
plot(density(1-theta_A),type="l",main="Posterior distribution",xlab="1 - theta",ylab="")
```



The posterior probability is 0.36. The posterior distribution is plotted above.

1b

```
Ratio <- (1-theta_A)/theta_A
quantile(Ratio,probs=c(0.025,0.975))
```

```
##     2.5%    97.5%
## 1.136435 2.259215
```

The ratio is the odds of not choosing brand A, i.e. it describes how many more times likely it is to not choose brand A compared to choosing brand A. The credible interval shows the values of the ratio with 95 % probability.

1c

```
beta(post_alpha,post_beta)/beta(alpha,beta) # Ratio of beta functions
```

```
## [1] 7.556771e-30
```

The marginal likelihood is given above (see for example lecture 11, slide 3).

1d

```
#########################################################################################
# Generate samples from the joint posterior distribution of theta=(theta_1,...,theta_K)
# for the multinomial model with K categories and a Dirichlet prior for theta.
#########################################################################################
```

```r
Dirichlet <- function(NDraws,y,alpha){
  K <- length(alpha)
  xDraws <- matrix(0,NDraws,K)
  thetaDraws <- matrix(0,NDraws,K) # Matrix where the posterior draws of theta are stored
  for (j in 1:K){
    xDraws[,j] <- rgamma(NDraws,shape=alpha[j]+y[j],rate=1)
  }
  for (ii in 1:NDraws){
    thetaDraws[ii,] <- xDraws[ii,]/sum(xDraws[ii,])
  }
  return(thetaDraws)
}
y_count <- c(38,27,35) # Data of counts for each category
alpha_const <- 20
alpha <- alpha_const*c(1,1,1) # Dirichlet prior hyperparameters
NDraws <- 1e5 # Number of posterior draws

###########   Posterior sampling from Dirichlet  #################
thetaDraws <- Dirichlet(NDraws,y_count,alpha)
mean(thetaDraws[,1] > thetaDraws[,3])
```

```
## [1] 0.61056
```

The posterior probability is 0.611.

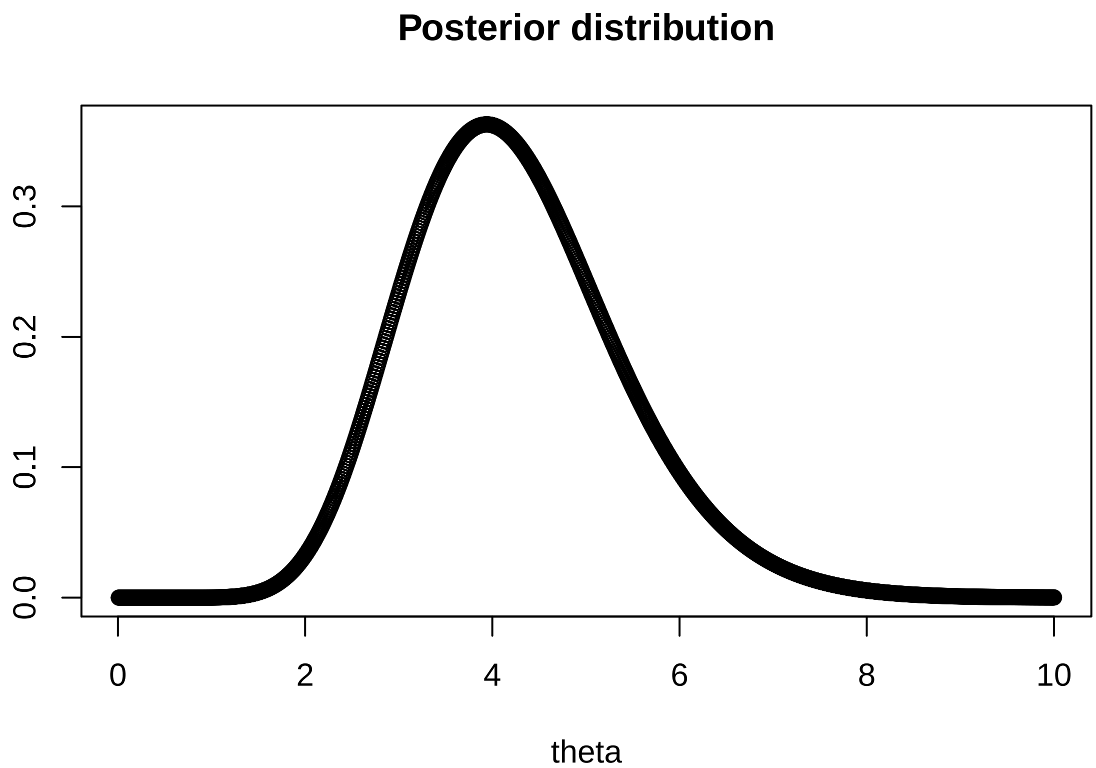
Problem 2

2d

```r
LogPost <- function(theta,n,Sumx2){

  logLik <- n*log(theta) -  Sumx2*theta;
  logPrior <- -0.5*theta;

  return(logLik + logPrior)
}
theta_grid <- seq(0.01,10,0.01)
PostDens_propto <- exp(LogPost(theta_grid,13,2.8))
PostDens <- PostDens_propto/(0.01*sum(PostDens_propto))
plot(theta_grid,PostDens,main="Posterior distribution",xlab="theta", ylab="")
```

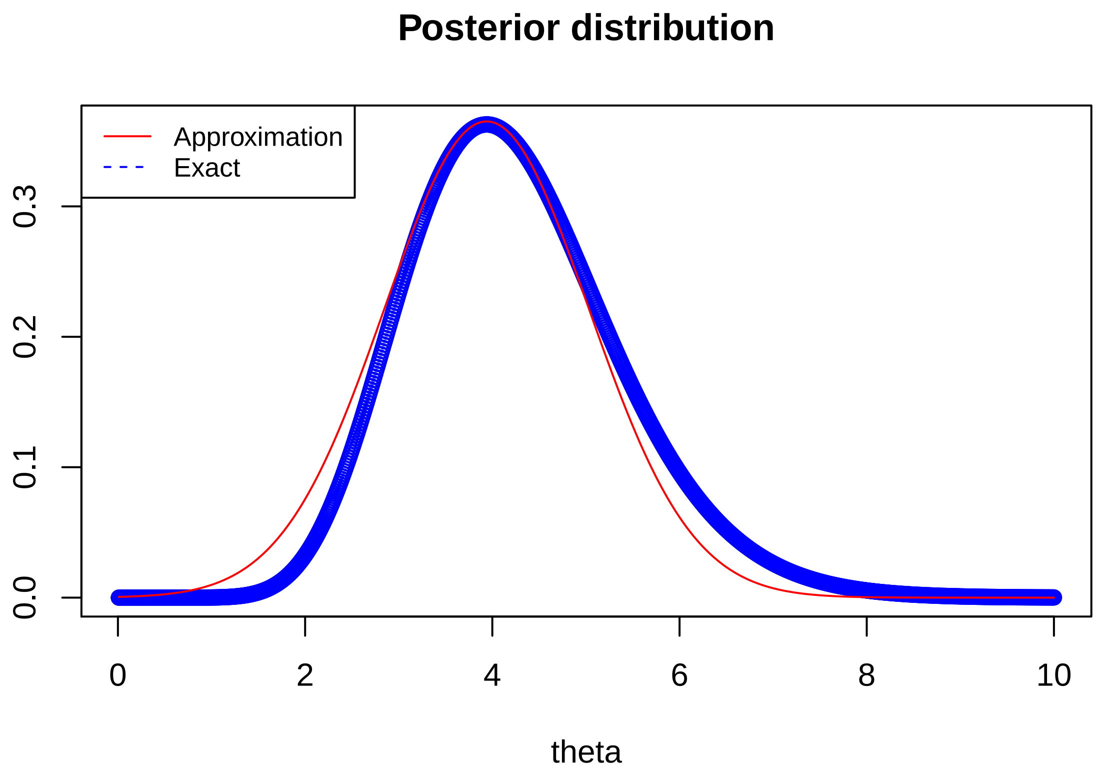


The posterior distribution is given above.

2e

```
n <- 13
Sumx2 <- 2.8
OptRes <- optim(3,LogPost,gr=NULL,n,Sumx2,method=c("L-BFGS-B"),lower=0.1,
                control=list(fnscale=-1),hessian=TRUE)

plot(theta_grid,PostDens,col="blue",main="Posterior distribution",xlab="theta", ylab="")
lines(theta_grid,dnorm(theta_grid,mean = OptRes$par,sd = sqrt(-1/OptRes$hessian)),col="red")
legend("topleft", legend=c("Approximation", "Exact"), col=c("red", "blue"), lty=1:2, cex=0.8)
```



The posterior approximation is quite accurate, but the exact posterior distribution is skewed to the right.

Problem 3

3a

```
mu_0 <- as.vector(rep(0,7))
Omega_0 <- (1/25)*diag(7)
v_0 <- 1
sigma2_0 <- 4
nIter <- 10000
library(geoR)
library(mvtnorm)
```

```
## Warning: package 'mvtnorm' was built under R version 4.0.5
```

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

Betas <- PostDraws$betaSample

Means <- colMeans(Betas)
CredInt <- matrix(0,7,2)
for (j in 1:7){
  CredInt[j,] <- quantile(Betas[,j],probs=c(0.025,0.975))
}
PostRes <- matrix(0,7,3)
PostRes[,1] <- t(Means)
```

```r
PostRes[,2:3] <- CredInt
PostRes
```

```
##             [,1]        [,2]        [,3]
## [1,]  1.30705333  1.14948644  1.46210744
## [2,]  0.70118560  0.52783800  0.87101547
## [3,]  0.15752730  0.05067731  0.26694297
## [4,]  0.42825292  0.02847996  0.83707772
## [5,] -0.16180732 -0.36597564  0.03880249
## [6,]  0.07499406 -0.28416227  0.43722247
## [7,] -0.24218483 -0.45147799 -0.03627859
```

It is 95 % posterior probability that beta_1 is on the interval (0.528,0.876).

3b

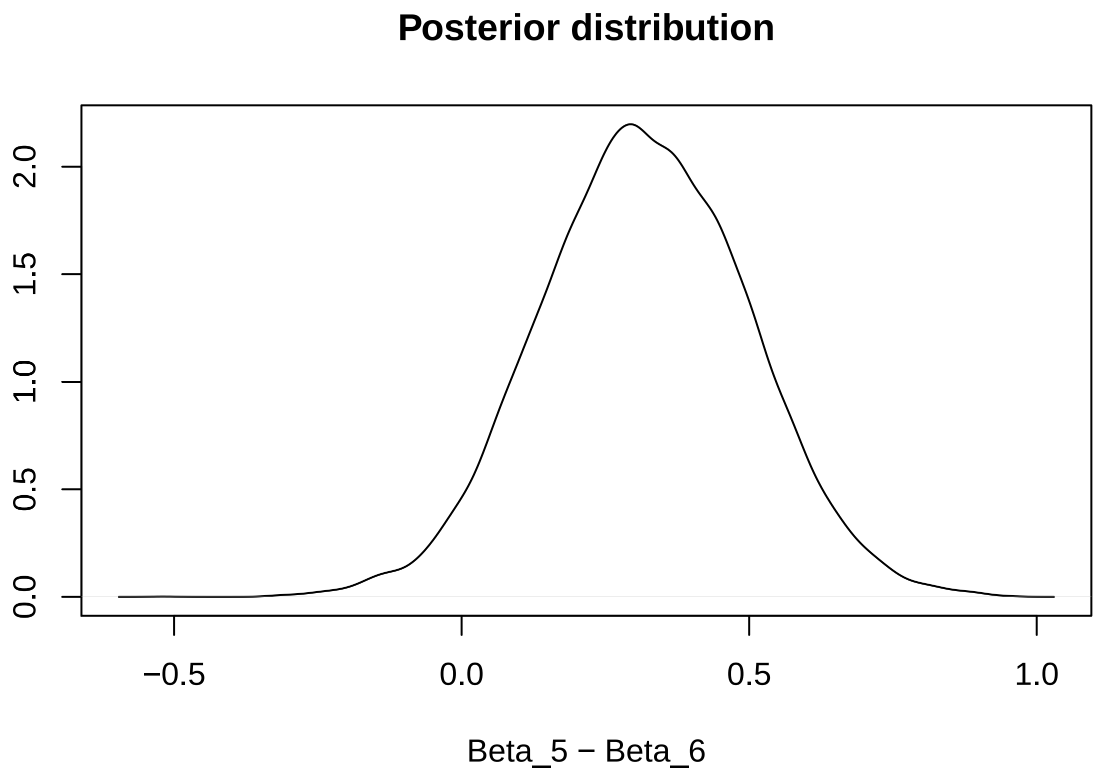
```r
Sigma2 <- PostDraws$sigma2Sample
median(sqrt(Sigma2))
```

```
## [1] 0.6402451
```

3c

```r
Effect_B <- Betas[,2] + Betas[,6]
Effect_C <- Betas[,2] + Betas[,7]
Diff <- Effect_B - Effect_C

plot(density(Diff),main="Posterior distribution",xlab="Beta_5 - Beta_6", ylab="")
```




```
quantile(Diff,probs=c(0.025,0.975))
```

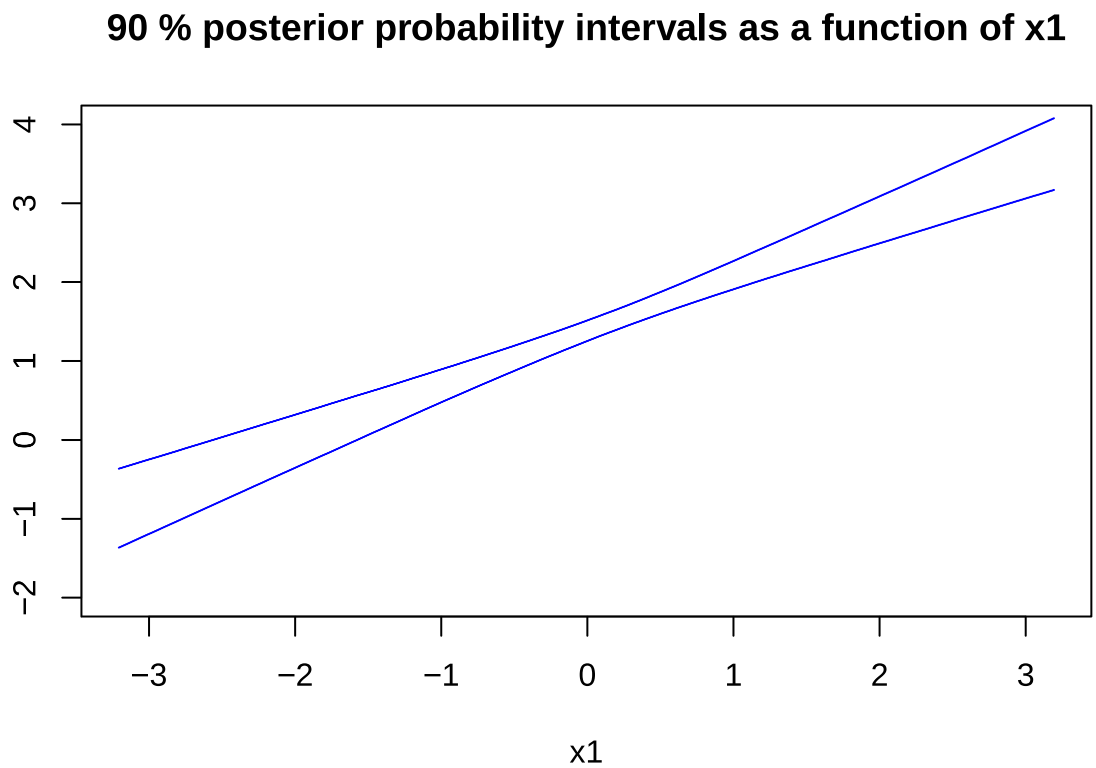
```
##        2.5%       97.5%
## -0.03344531  0.66779927
```

There is substantial probability mass that the effect on y from x1 is larger in high school B compared to high school C. However, the 95 % equal tail credible interval for the difference of the slopes of x1 between the high schools reveals that the difference can be either negative or positive. Hence, the probability is not that high that this effect in high school B is larger than in high school C.

3d

```
x1_grid <- seq(min(X[,2]),max(X[,2]),0.01)
Mu_draws <- matrix(0,length(x1_grid),2)
for (ii in 1:length(x1_grid)){
  CurrMu <- Betas[,1] + Betas[,2]*x1_grid[ii] + Betas[,3]*0.5
  Mu_draws[ii,] <- quantile(CurrMu,probs=c(0.05,0.95))
}

plot(x1_grid,Mu_draws[,1],"n",main="90 % posterior probability intervals as a function of x1",
     xlab="x1", ylab="",ylim=c(-2,4))
lines(x1_grid,Mu_draws[,1],col="blue")
lines(x1_grid,Mu_draws[,2],col="blue")
```

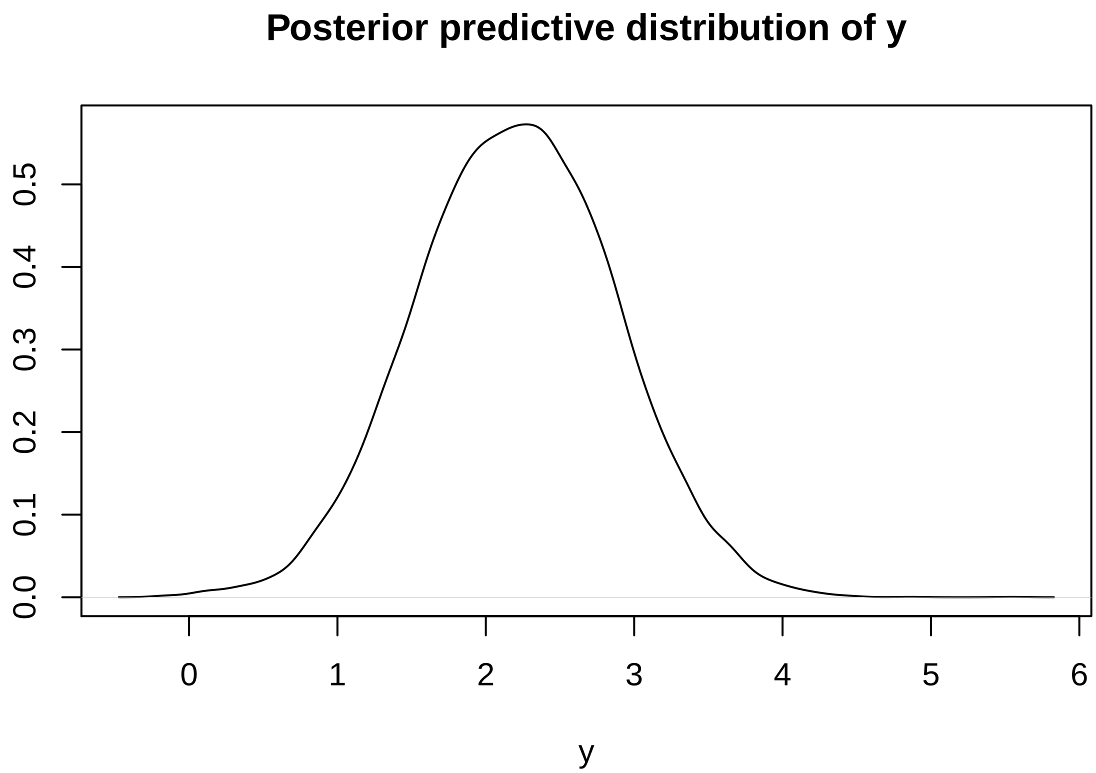


The limits of the posterior probability intervals as a function of x1 are plotted above.

3e

```
Mu <- Betas[,1] + Betas[,2]*0.4 + Betas[,3]*1 + Betas[,4]*1 + Betas[,6]*0.4
Sigma <- sqrt(Sigma2)

y_Vals <- rnorm(10000,Mu,Sigma)
plot(density(y_Vals),main="Posterior predictive distribution of y",xlab="y", ylab="")
```



The posterior predictive distribution of y is plotted above.