

June 201

Christophoros Spyretos

2022-10-16

## Problem 1

### Task a

```
#given values
n <- 100
s <- 38
f <- n-s
alpha0 <- 16
beta0 <- 24

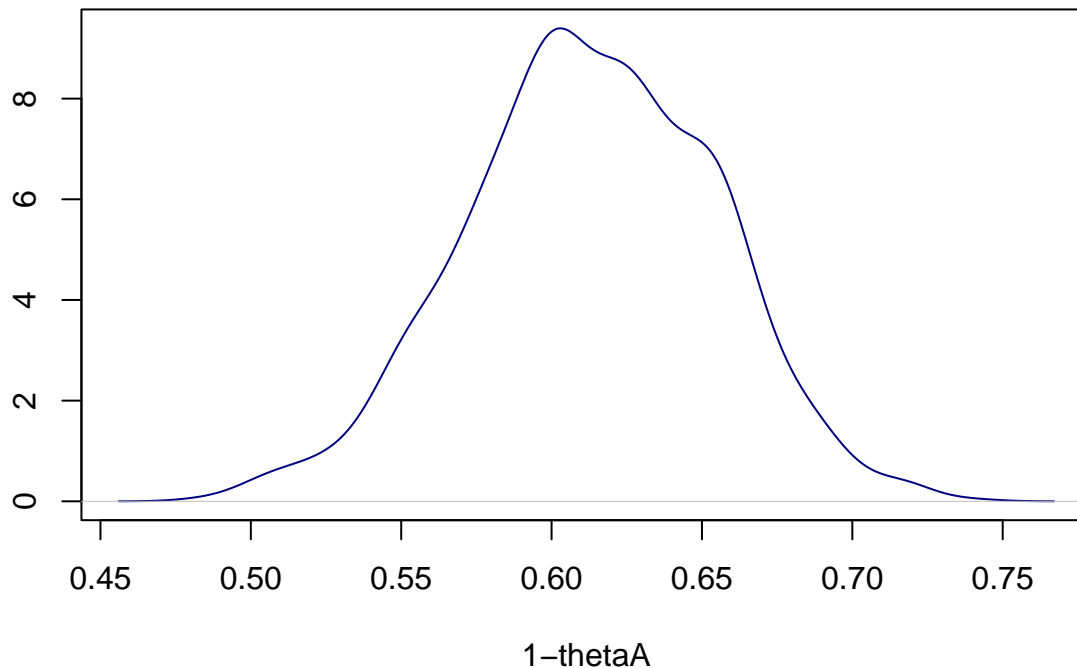
#posterior parameters
post_alpha <- alpha0 + s
post_beta <- beta0 + f

#draw observations
nDraws <- 1000
thetaA <- rbeta(n=nDraws, shape1 = post_alpha, shape2 = post_beta)

#calculate posterior probability
prob <- pbeta(q=0.4, shape1 = post_alpha, shape2 = post_beta, lower.tail = FALSE)

plot(density(1-thetaA), type = "l",
     main = "Posterior Distribution of 1-thetaA",
     xlab = "1-thetaA", ylab = "", col = "navy")
```

## Posterior Distribution of 1-thetaA



The posterior probability that  $\theta_A > 0.4$  is approximately 0.36.

### Task b

```
ratio <- (1-thetaA)/thetaA

interval <- quantile(ratio, probs = c(0.025,0.975))

df_intervals <- data.frame(lower_bound = interval[1], upper_bound = interval[2])
colnames(df_intervals) <- c("lower bound", "upper bound")
rownames(df_intervals) <- c("95% Equal Tail Credible Interval")
knitr::kable(df_intervals)
```

	lower bound	upper bound
95% Equal Tail Credible Interval	1.131894	2.219882

The ratio shows the odds of not choosing brand A. The 95% equal tail credible interval for the ratio describes the values of the ratio with 95 % probability.

### Task c

```
marginal_likelihood <- beta(post_alpha, post_beta)/beta(alpha0, beta0)
```

The marginal likelihood is approximately 7.55.

## Task d

```
counts <- c(38,27,35)
alpha_const <- 20
alpha <- alpha_const*c(1,1,1)

K <- length(alpha)
xDraws <- matrix(0, nrow = nDraws, ncol = K)
thetaDraws <- matrix(0, nrow = nDraws, ncol = K)

for (i in 1:K) {
  xDraws[,i] <- rgamma(nDraws, shape = alpha[i] + counts[i], rate = 1)
}

for (j in 1:nDraws) {
  thetaDraws[j,] <- xDraws[j,]/sum(xDraws[j,])
}

prob <- mean(thetaDraws[1,] > thetaDraws[3,])
```

The posterior probability that  $\theta_A > \theta_C$  is approximately 0.611.

## Problem 2

Task a,b and c are hand written

## Task d

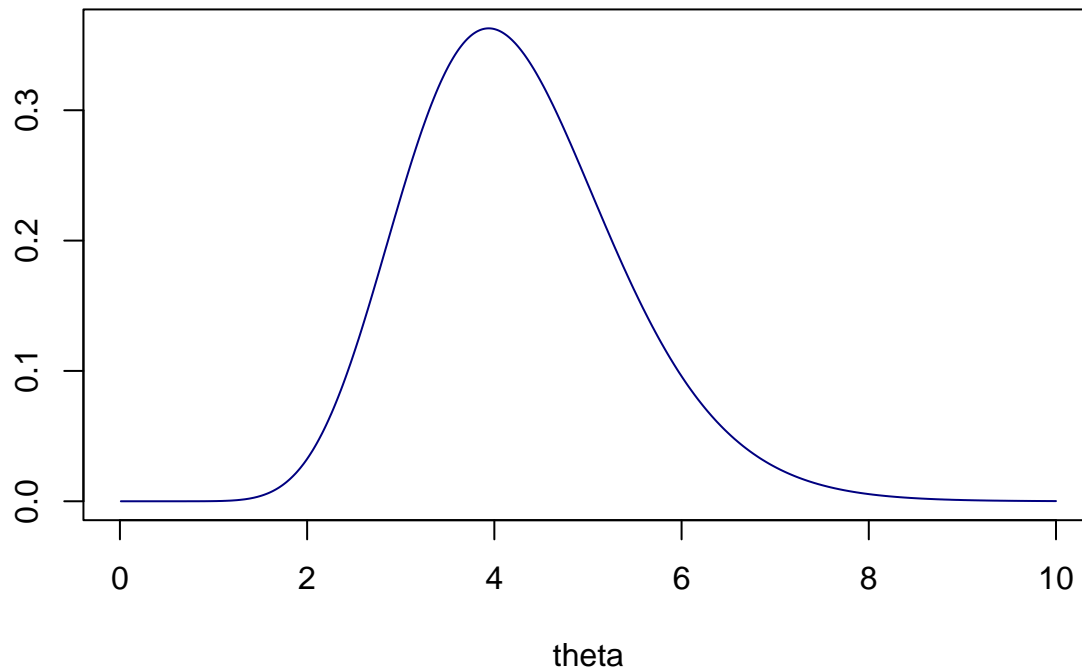
```
LogPost <- function(theta,n,sumx2) {
  res <- n*log(theta) - theta*(0.5 + sumx2)
  return(res)
}

n <- 13
thetaGrid <- seq(0.01,10,0.01)
sumx2 <- 2.8

postDens_propto <- exp(LogPost(thetaGrid,n,sumx2))
postDens <- postDens_propto/(0.01*sum(postDens_propto))

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

## Posterior Distribution of theta



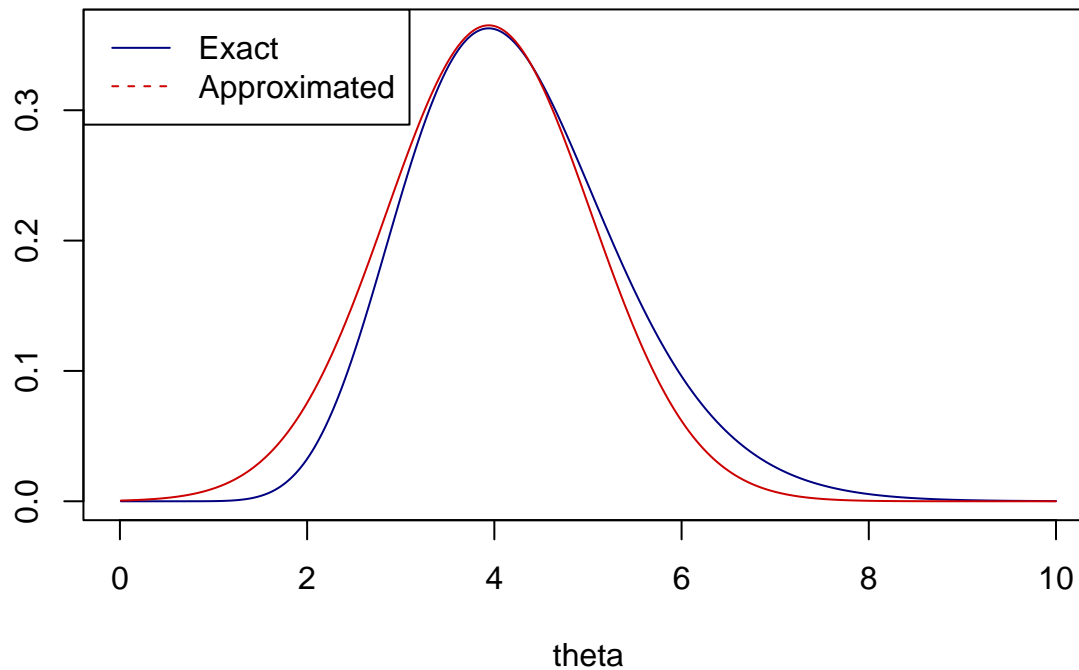
### Task d

```
OptimRes <- optim(3, LogPost, gr=NULL, n, sumx2,
                 method=c("L-BFGS-B"), lower=0.1,
                 control=list(fnscale=-1), hessian=TRUE)

approx <- dnorm(thetaGrid, mean = OptimRes$par, sd = sqrt(-1/OptimRes$hessian))

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

## Posterior Distribution of theta



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

## Problem 3

### Task a

```
library(mvtnorm)

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

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

Betas <- PostDraws$betaSample

BetasMean <- colMeans(Betas)

BetasMean <- as.data.frame(BetasMean)
rownames(BetasMean) <- c("Beta0", "Beta1", "Beta2", "Beta3",
                        "Beta4", "Beta5", "Beta6")
knitr::kable(BetasMean)
```

	BetasMean
Beta0	1.3086342
Beta1	0.7013319

	BetasMean
Beta2	0.1563998
Beta3	0.4260635
Beta4	-0.1639961
Beta5	0.0751615
Beta6	-0.2409906

```
intervals <- matrix(0,7,2)

for (i in 1:7){
  intervals[i,] <- quantile(Betas[,i], probs = c(0.025,0.975))
}

intervals <- as.data.frame(intervals)
colnames(intervals) <- c("Lower Bound", "Upper Bound")
rownames(intervals) <- c("Beta0", "Beta1", "Beta2", "Beta3",
  "Beta4", "Beta5", "Beta6")
knitr::kable(intervals, caption = "95% Equal Tail Credible Interval")
```

Table 3: 95% Equal Tail Credible Interval

	Lower Bound	Upper Bound
Beta0	1.1535737	1.4657085
Beta1	0.5247313	0.8749398
Beta2	0.0457782	0.2661933
Beta3	0.0319548	0.8189005
Beta4	-0.3680022	0.0405376
Beta5	-0.2853119	0.4336701
Beta6	-0.4515462	-0.0277552

It is 95 % posterior probability that  $\beta_{1,1}$  is on the interval (0.528,0.876).

## Task b

```
PostSigma2Median <- median(sqrt(PostDraws$sigma2Sample))
```

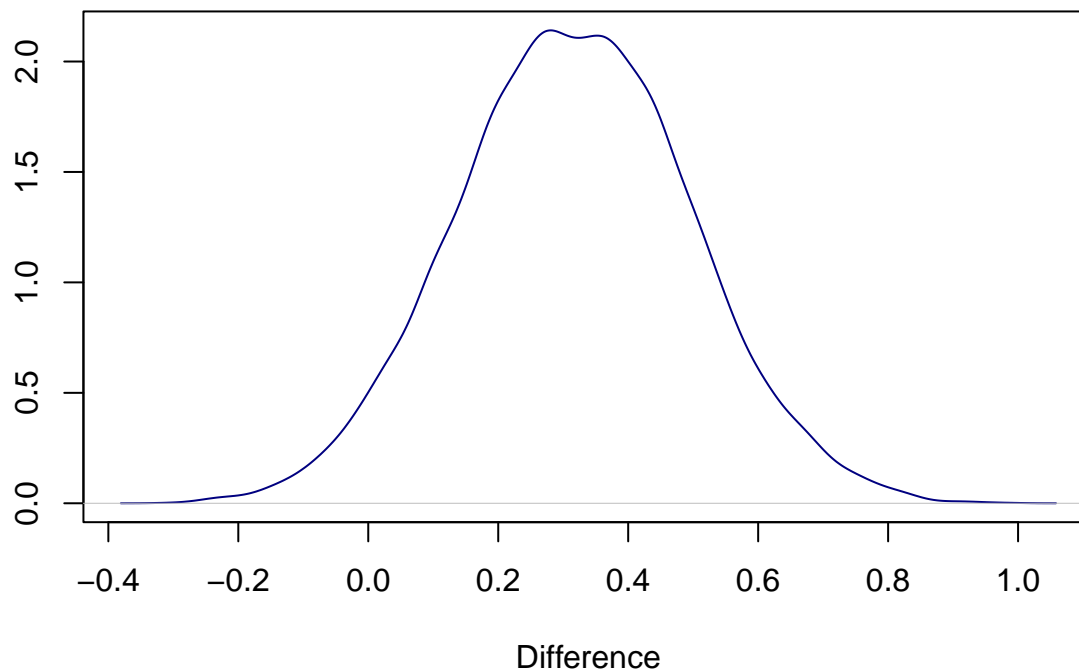
The posterior median of the standard deviation  $\sigma$  is 0.639.

## Task c

```
effectB <- Betas[,2] + Betas[,6]
effectC <- Betas[,2] + Betas[,7]
diff <- effectB - effectC

plot(density(diff), type = "l", col = "navy",
  main = "x1 Effect on High School B and High School C",
  xlab = "Difference", ylab = "")
```

## x1 Effect on High School B and High School C



```
intervalDiff <- quantile(diff, probs = c(0.025, 0.975))
# table for the interval
intervalDiff <- data.frame(lower_bound = intervalDiff[1],
                           upper_bound = intervalDiff[2])
colnames(intervalDiff) <- c("lower bound", "upper bound")
rownames(intervalDiff) <- c("95% Equal Tail Credible Interval")
knitr::kable(intervalDiff)
```

	lower bound	upper bound
95% Equal Tail Credible Interval	-0.0335425	0.6710474

From the plot it seems that the effect on  $y$  from  $x_1$  is greater in high school B compared to high school C. However, the 95% equal tail credible interval for the difference of the slopes of  $x_1$  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.

### Task d

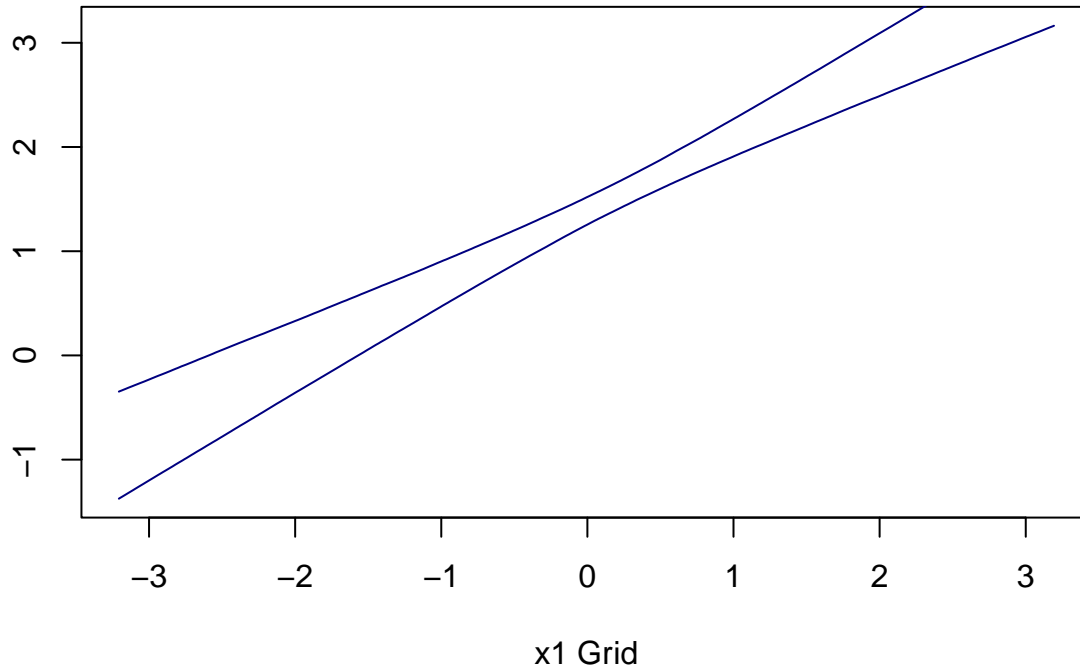
```
x1Grid <- seq(min(X[,2]), max(X[,2]), 0.01)
intervalsMu <- matrix(0, length(x1Grid), 2)

for (i in 1:length(x1Grid)) {
  mu <- Betas[,1] + Betas[,2] * x1Grid[i] + Betas[,3] * 0.5
  intervalsMu[i,] <- quantile(mu, probs=c(0.05, 0.95))
}

plot(x1Grid, intervalsMu[,1], type = "l", col = "navy",
     main = "90% Equal Tail Posterior Probability Intervals",
```

```
xlab = "x1 Grid", ylab = "")
lines(x1Grid, intervalsMu[,2], col = "navy")
```

## 90% Equal Tail Posterior Probability Intervals



### Task e

```
mu <- Betas[,1] + Betas[,2] * 0.4 + Betas[,3] +
  Betas[,4] + Betas[,6] * 0.4
sigma <- sqrt(PostDraws$sigma2Sample)

values <- rnorm(nIter, mu, sigma)

plot(density(values), type = "l", col = "navy",
     main = "Posterior Predictive Distribution of y for a New Student",
     xlab = "values", ylab = "")
```



## Posterior Predictive Distribution of $y$ for a New Student

