

August 2021

Christophoros Spyretos

2022-10-16

Problem 1

Task a

Hand written solution

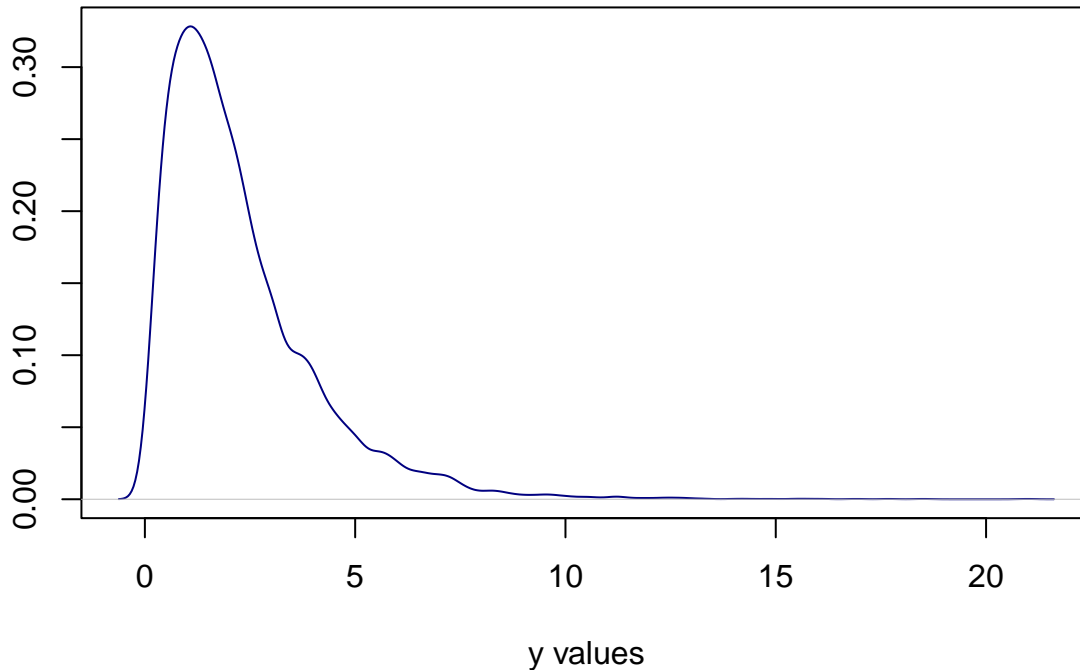
Task b

```
obs <- c(2.32,1.82,2.4,2.08,2.13)
n <- length(obs)
Ndraws <- 10000

thetaDraws <- rgamma(n = Ndraws, shape = 2*n + 1, rate = 0.5 + sum(obs))
y <- rgamma(n = Ndraws, shape = 2, rate = thetaDraws)

plot(density(y), type = "l", col = "navy",
     main = "Posterior Distribution",
     xlab = "y values", ylab = "")
```

Posterior Distribution



```
prob <- mean(y<1.9)
```

The $Pr(Y_6 < 1.9|y_1, \dots, y_5)$ is 0.5313.

Task c

```
Nweeks <- 30
weights <- matrix(0, Ndraws, Nweeks)

for (i in 1:Ndraws) {
  thetaDraws <- rgamma(n = Nweeks, shape = 2*n + 1, rate = 0.5 + sum(obs))
  weights[i,]<- rgamma(n = Nweeks, shape = 2, rate = thetaDraws)
}

ExceedWeights <- mean(rowSums(weights>2.4))
```

The expected number of weeks out of the future 30 weeks in which the maximal weight will exceed 2.4 thousands of kilos is approximately 10.5.

Task d

```
loss_function <- function(a,weights){
  res <- a + mean(rowSums(weights>0.9*log(a)))
  return(res)
}

aGrid <- seq(0.01,10,0.01)

expected_loss <- matrix(0,length(aGrid),1)
```

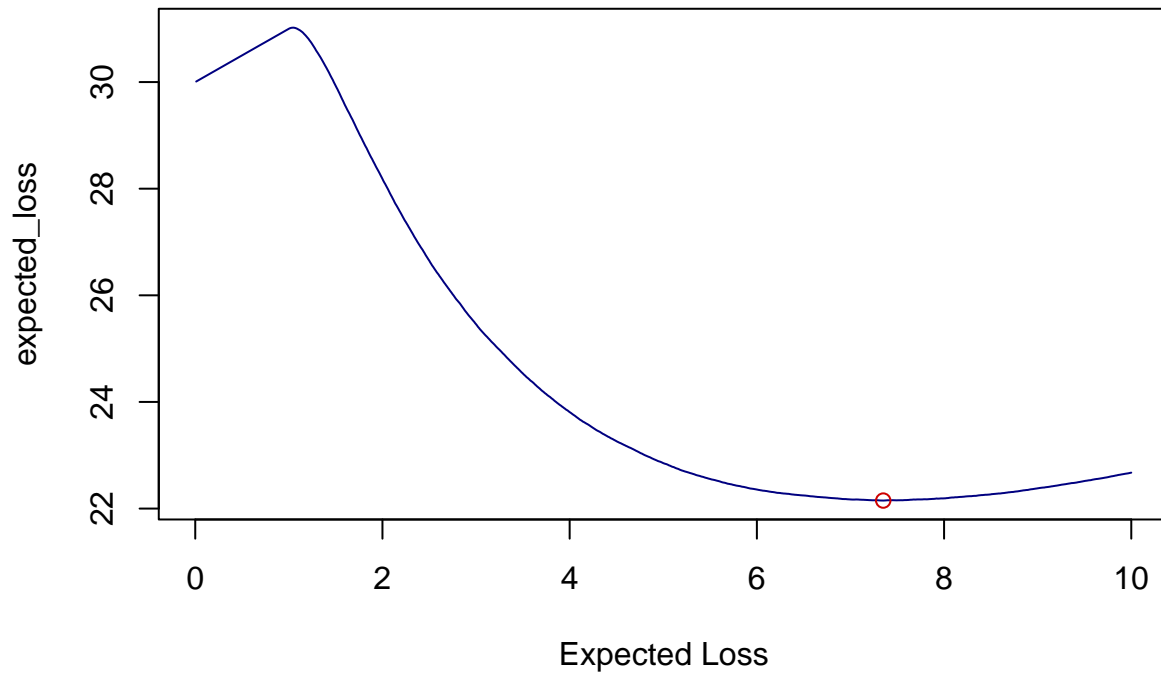
```

for (i in 1:length(aGrid)){
  expected_loss[i,] <- loss_function(aGrid[i],weights)
}

aOpt <- aGrid[which.min(expected_loss)]

plot(aGrid,expected_loss, type = "l", col = "navy",
      xlab = "Expected Loss")
points(aOpt, loss_function(aOpt,weights),col = "red3")

```



The optimal build cost is 7.39.

Problem 2

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

Task a

```

library(mvtnorm)

nIter <- 10000
mu_0 <- as.vector(rep(0,8))
Omega_0 <- 1/9 * diag(8)
v_0 <- 1
sigma2_0 <- 9
X <- as.matrix(X)

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

```

```

BetaDraws <- PostDraws$betaSample

interval <- quantile(BetaDraws[,2], probs = c(0.005,0.995))

interval <- data.frame(lower_bound = interval[1], upper_bound = interval[2])
colnames(interval) <- c("lower bound", "upper bound")
rownames(interval) <- c("99% Equal Tail Credible Interval")
knitr::kable(interval)

```

| | lower bound | upper bound |
|----------------------------------|-------------|-------------|
| 99% Equal Tail Credible Interval | -0.3437328 | 1.852771 |

It is 99 % posterior probability that β_1 is on the interval (-0.33,1.84).

Task b

```

mu <- BetaDraws[,1] + BetaDraws[,2] + BetaDraws[,3] + BetaDraws[,4] * 0.5 +
  BetaDraws[,6] + BetaDraws[,8]

CV <- sqrt(PostDraws$sigma2Sample)/mu

MedianCV <- median(CV)

```

The median of the coefficient of variation is approximately 1.83.

Task c

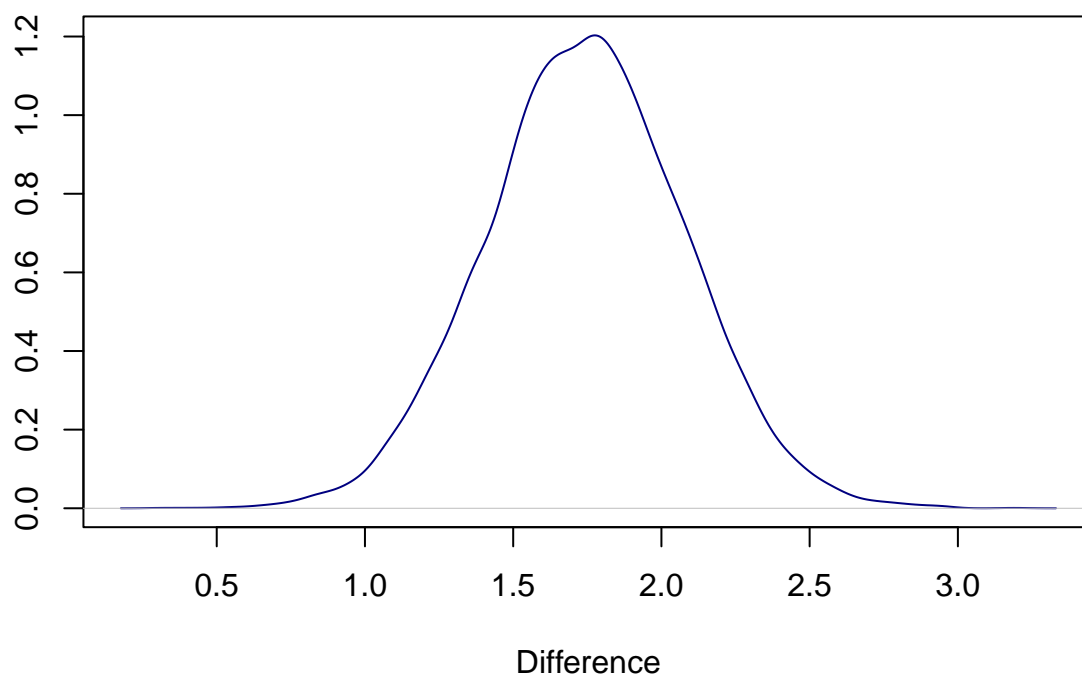
```

Effect_Inner <- BetaDraws[,1] + BetaDraws[,2] +
  BetaDraws[,5] + BetaDraws[,7]
Effect_South <- BetaDraws[,1] + BetaDraws[,2] +
  BetaDraws[,6] + BetaDraws[,8]

Diff <- Effect_Inner - Effect_South
plot(density(Diff), type = "l", col = "navy",
     main = "Inner vs South Apartments",
     xlab = "Difference", ylab = "")

```

Inner vs South Apartments

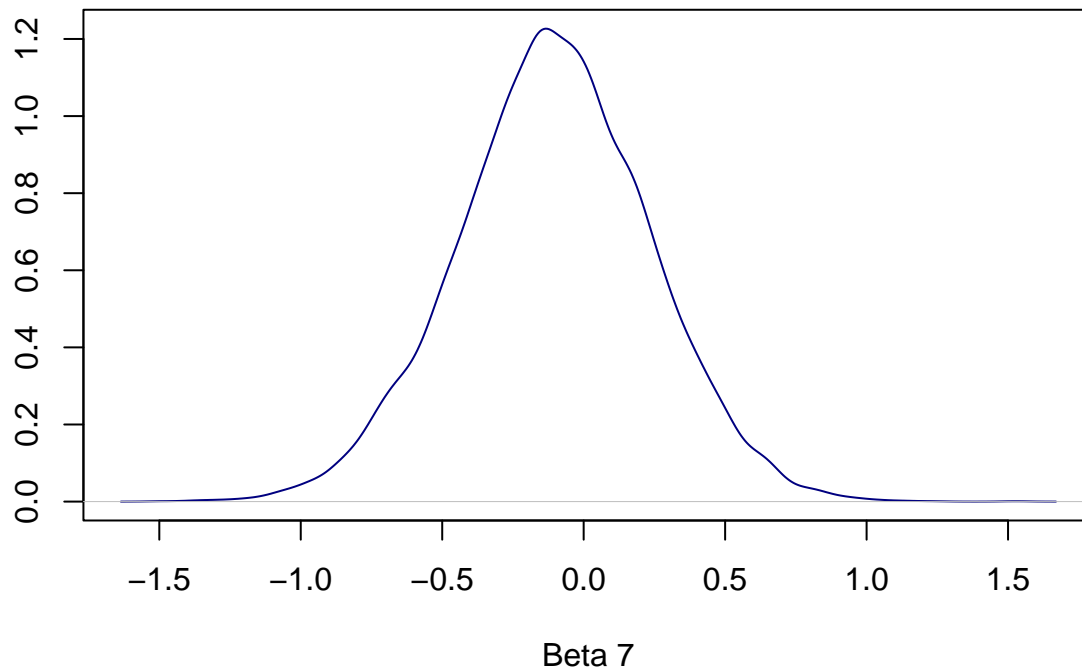


```
intervalDiff <- quantile(Diff, probs = c(0.025,0.975))
intervalDiff <- data.frame(lower_bound = intervalDiff[1],
                           upper_bound = intervalDiff[2])
colnames(intervalDiff) <- c("lower bound", "upper bound")
rownames(intervalDiff) <- c("95% Equal Tail Credible Interval")
knitr::kable(intervalDiff)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

| | lower bound | upper bound |
|----------------------------------|-------------|-------------|
| 95% Equal Tail Credible Interval | 1.090269 | 2.399985 |

There is a high probability that the apartments in the inner city have a higher price than the apartments on the south side of the city. The 95% equal tail credible interval has positive bounds which strengthens the assumption.

```
plot(density(BetaDraws[,8]), type = "l", col = "navy",
     main = "", xlab = "Beta 7", ylab = "")
```



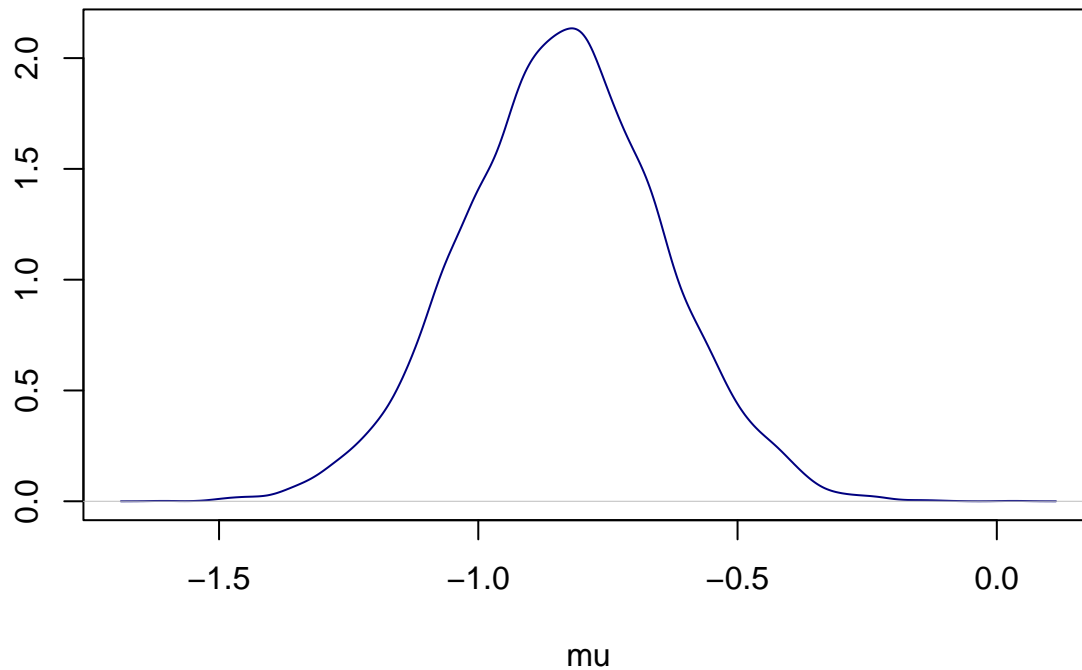
There is a substantial probability mass on both sides of 0. Thus, the effect on the selling price y from x_1 is not different for apartments on the south side of the city compared to apartments which are neither in the inner city nor on the south side of the city.

Task d

```
mu <- BetaDraws[,1] + BetaDraws[,2] * (-0.5) + BetaDraws[,3] * (-0.5) +
  BetaDraws[,6] + BetaDraws[,8] * (-0.5)

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

Posterior Distribution of mu



```
prob <- mean(mu>0)
```

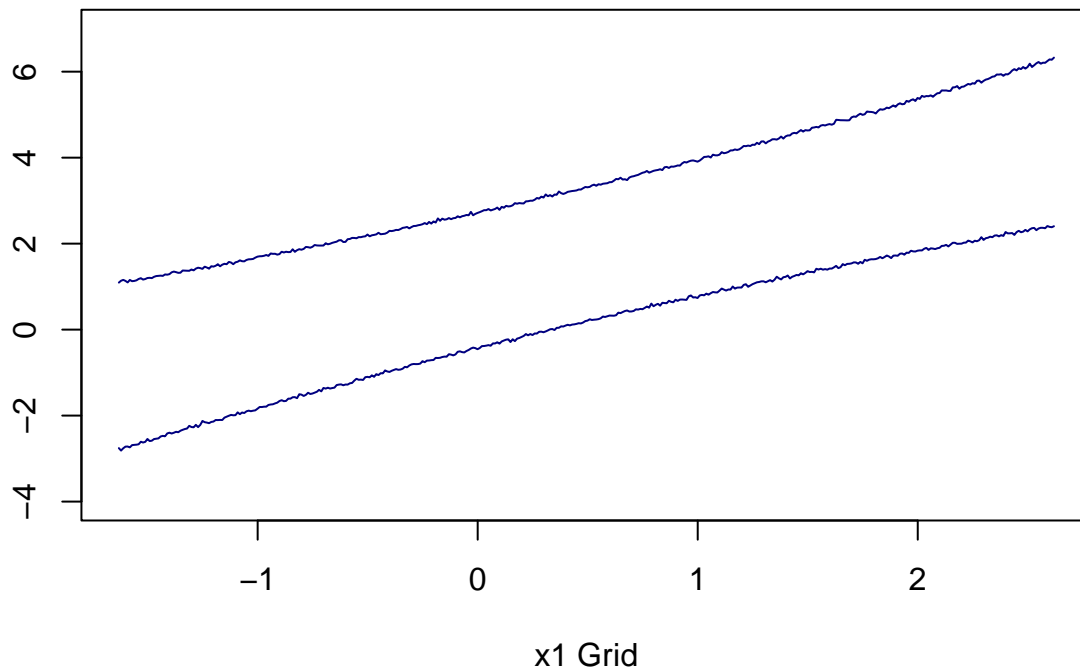
The posterior probability that $\mu > 0$ is 0.

Task e

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

for (i in 1:length(x1Grid)) {
  mu <- BetaDraws[,1] + BetaDraws[,2] * x1Grid[i] + BetaDraws[,3] +
    BetaDraws[,4] * 0.5 + BetaDraws[,5] + BetaDraws[,7] * x1Grid[i]
  intervals[i,] <- quantile(rnorm(nIter, mu, sqrt(PostDraws$sigma2Sample)),
    probs = c(0.025,0.975))
}

plot(x1Grid,intervals[,1], type = "l", col = "navy",
     main = "", xlab = "x1 Grid", ylab = "", ylim=c(-4,7))
lines(x1Grid, intervals[,2], type = "l", col = "navy")
```



Problem 3

Task a,b,c

Hand written solution.

Task d

```
LogPost <- function(theta, n, sumlogx){
  res <- 2*theta*sumlogx - n*(theta^2)
  return(res)
}

thetaGrid <- seq(-1,2,0.01)
n <- 5
sumlogx <- 2

PostDens_propto <- exp(LogPost(thetaGrid,n,sumlogx))
PostDens <- PostDens_propto/0.01*sum(PostDens_propto)

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