

August-2021-Exam

Christophoros Spyretos

Exercise 1

Task a

Hand written solution.

Task b

```
set.seed(12345)

y <- c(2.32,1.82,2.4,2.08,2.13)
n <- 5

N <- 10000

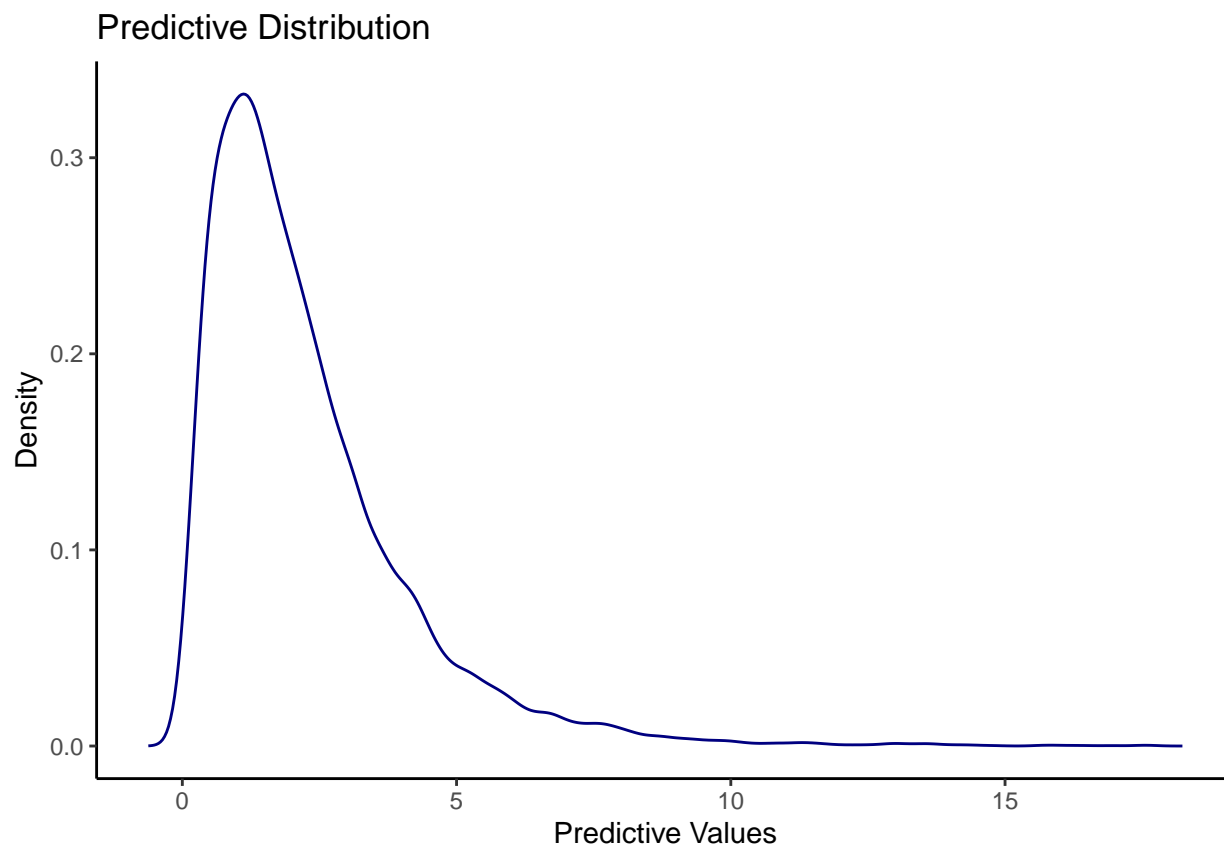
#generate thetas from the posterior distribution
theta <- rgamma(N, shape = 2*n + 1, rate = 0.5 + sum(y))

#generate predictive values from the given gamma density
y_tilde <- rgamma(N, shape = 2, rate = theta)

#calculate the density
dens_y_tilde <- density(y_tilde)

df_plot1 <- data.frame("x" = dens_y_tilde$x, "y" = dens_y_tilde$y)

ggplot(df_plot1) +
  geom_line(aes(x=x,y=y), color = "navy") +
  ggtitle("Predictive Distribution") +
  xlab("Predictive Values") +
  ylab("Density") +
  theme_classic()
```



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

The posterior predictive probability is roughly 0.53.

```
set.seed(12345)
```

```
weeks <- 30
```

```
weights <- matrix(NA, nrow = N, ncol = weeks)
```

```
for (i in 1:N){
  #generate thetas from the posterior distribution
  theta <- rgamma(weeks, shape = 2*n + 1, rate = 0.5 + sum(y))

  #generate predictive values form the given gamma density
  weights[i,]<- t(rgamma(weeks, shape = 2, rate = theta))
}
```

```
overweight_weeks <- 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 about 10.5.

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

```

a <- runif(1000,1,10)
loss_results <- rep(NA,length(a),1)
count <- 0

for (i in a){
  count <- count + 1
  loss_results[count] = loss_function(i, weights)
}

optimal_a <- a[which.min(loss_results)]

```

The optimal build cost (a) is approximately 7.3 thousands of kilos.

Exercise 2

```

#loading data & code
source("ExamData.R")

```

Task a

```

set.seed(12345)

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

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

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

betas <- PosteriorDraws$betaSample

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

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

```

	lower bound	upper bound
99% Equal Tail Credible Interval	-0.3714471	1.825996

It is 99 % posterior probability that β_1 is on the above interval.

Task b

```
x <- c(1,1,1,0.5,0,1,0,1)
mu <- betas %*% x
sigma <- sqrt(PosteriorDraws$sigma2Sample)
CV <- median(sigma/mu)
```

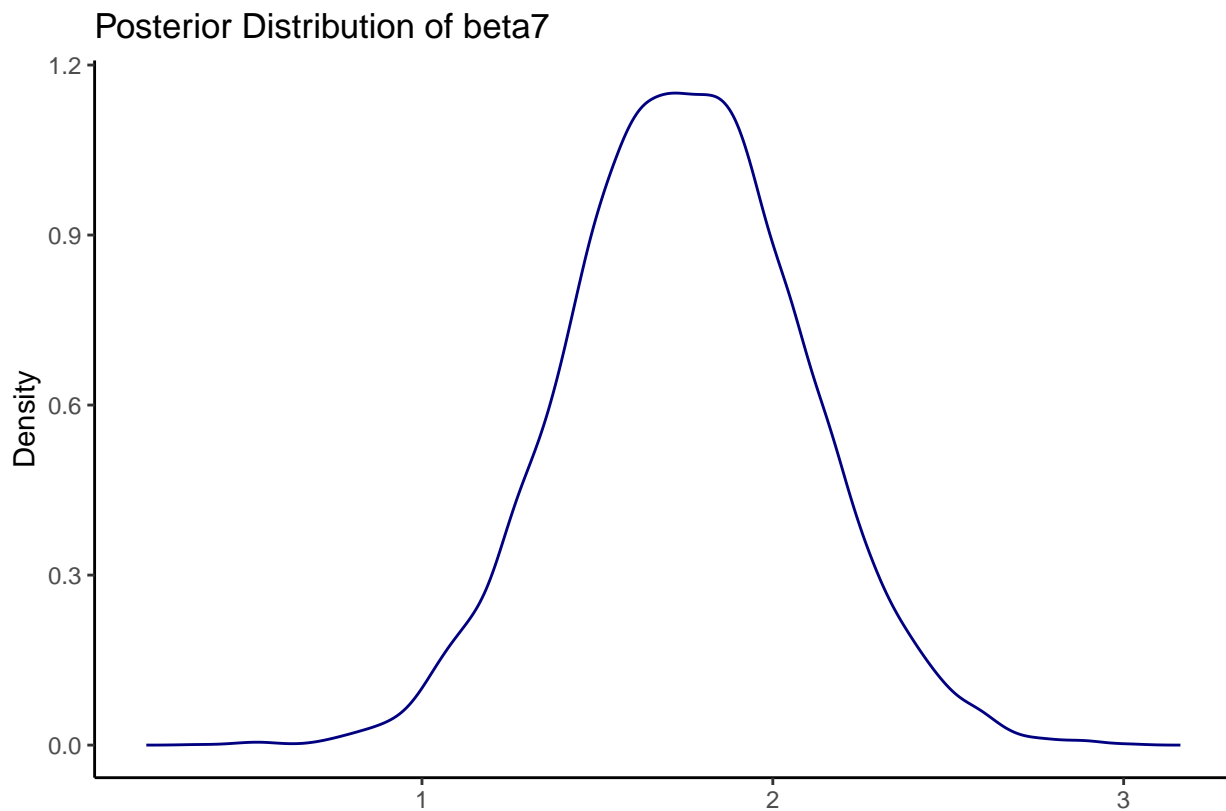
The median is about 1.816.

Task c

```
difference_price <- (betas[,5] + betas[,7]) - (betas[,6] + betas[,8])

dens_difference_price <- density(difference_price)
df_plot <- data.frame("x"= dens_difference_price$x, "y"= dens_difference_price$y)

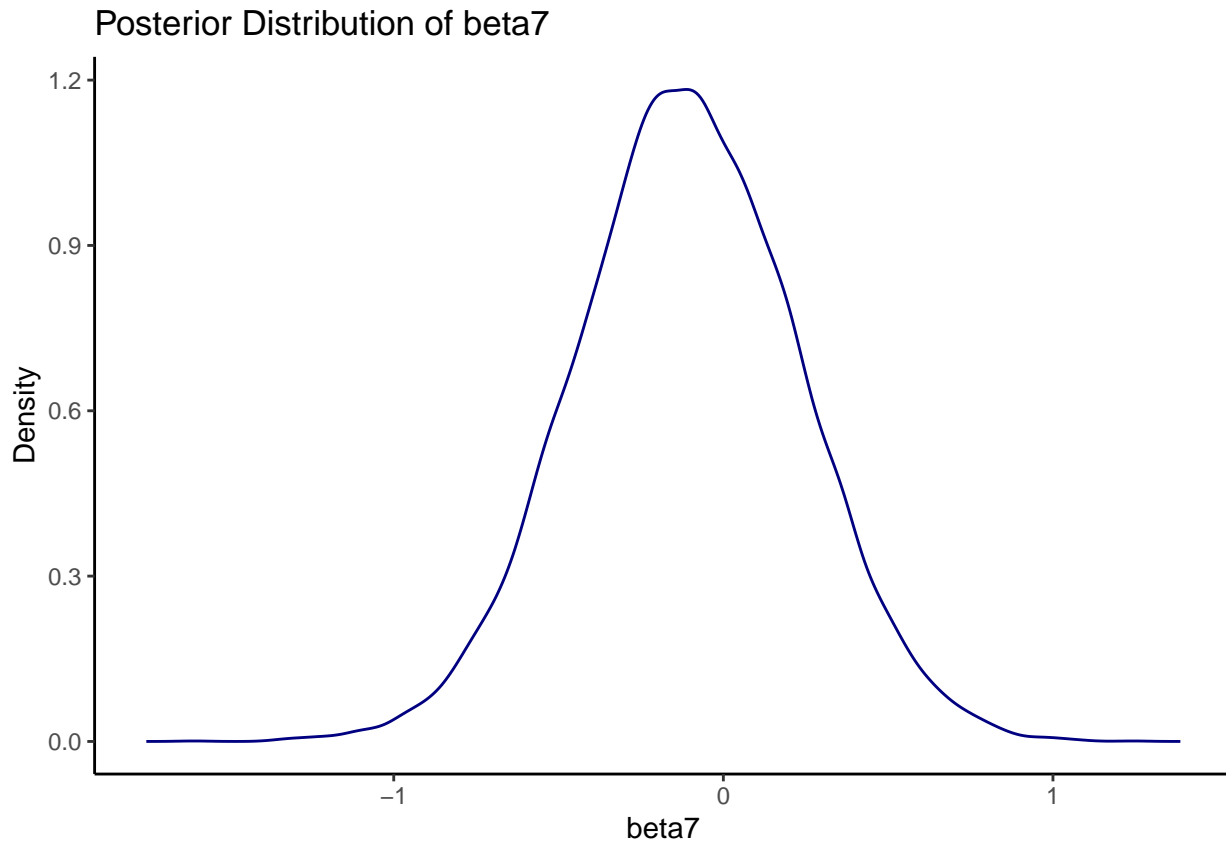
ggplot(df_plot) +
  geom_line(aes(x=x, y=y), color = "navy") +
  ggtitle("Posterior Distribution of beta7") +
  xlab("") +
  ylab("Density") +
  theme_classic()
```



The expected selling price is different for apartments in the inner city compared to apartments on the south side of the city. There is substantial probability mass that the expected price is higher for apartments in the inner city compared to the south side of the city.

```
dens_beta7 <- density(betas[,8])
df_plot <- data.frame("x"= dens_beta7$x, "y"= dens_beta7$y)
```

```
ggplot(df_plot) +
  geom_line(aes(x=x, y=y), color = "navy") +
  ggtitle("Posterior Distribution of beta7") +
  xlab("beta7") +
  ylab("Density") +
  theme_classic()
```



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. It could be seen that the substantial probability mass is on both sides of 0.

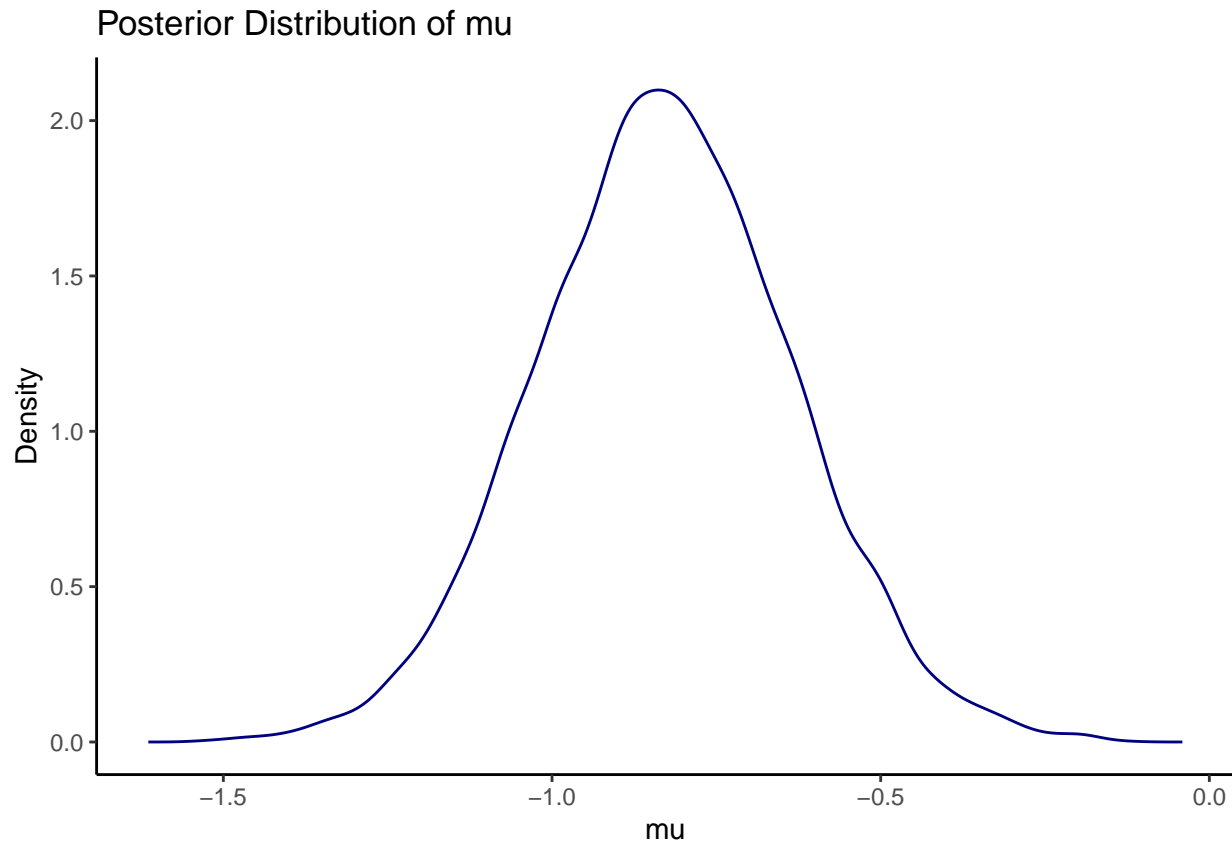
Task d

```
x <- c(1,-0.5,-0.5,0,0,1,0,-0.5)
mu <- betas %*% x

mu_density <- density(mu)

plot_df <- data.frame("x"=mu_density$x, "y"=mu_density$y)

ggplot(plot_df) +
  geom_line(aes(x=x, y=y), color = "navy") +
  ggtitle("Posterior Distribution of mu") +
  xlab("mu") +
  ylab("Density") +
  theme_classic()
```



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

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

Task e

```
set.seed(12345)

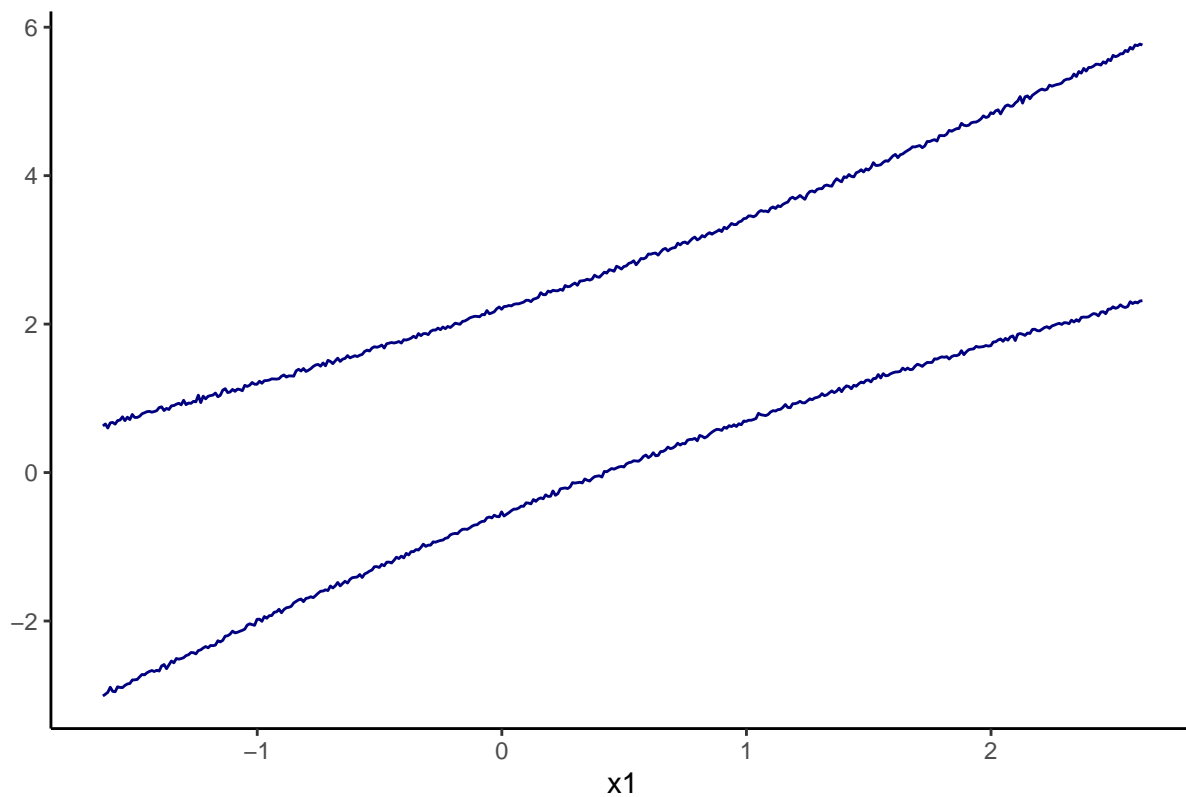
x1 <- seq(min(X[,2]), max(X[,2]), 0.01)
intervals <- matrix(0, nrow = length(x1), ncol = 2)

for (i in 1:length(x1)){
  mu <- betas %% c(1,x1[i],1,0.5,1,0,x1[i]*1,0)
  y_pred <- rnorm(nIter, mu, sigma)
  intervals[i,] <- quantile(y_pred, probs = c(0.025,0.975))
}

df_plot <- data.frame("x1"= x1, "low" = intervals[,1], "upper" = intervals[,2])

ggplot(df_plot) +
  geom_line(aes(x=x1, y=low), color= "navy") +
  geom_line(aes(x=x1, y=upper), color= "navy") +
  ggtitle("95 % posterior predictive intervals as a function of x1")+
  ylab("") +
  theme_classic()
```

95 % posterior predictive intervals as a function of x1



Exercise 3

Task a,b and c are hand written.

Task d

```
set.seed(1)

logPost <- function(theta,n,sumlogx){

  logLik <- -n*theta**2 + 2*theta*sumlogx

  return(logLik)
}

thetaGrid <- seq(-1,1.7,0.01)
n <- 5
sumlogx <- 2

posterior <- exp(logPost(thetaGrid,n,sumlogx))

posterior <- posterior/(0.01*sum(posterior))

df_plot2 <- data.frame("posterior"=posterior, "theta"= thetaGrid)

ggplot(df_plot2) +
```

```
geom_line(aes(x=theta, y=posterior), colour = "red4", size = 1) +  
ggtitle("Posterior Distribution of Theta") +  
xlab("Theta Values") +  
ylab("Density") +  
theme_classic()
```

