

BDA - Assignment 4

Anonymous

Contents

| | |
|---------------|---|
| Load packages | 1 |
| Exercise 1 | 1 |

Load packages

```
library(aaltobda)
data("bioassay_posterior")
data("bioassay")
head(bioassay_posterior)
```

```
##           alpha      beta
## 1 -0.02050577 10.032841
## 2  1.21738518  4.504546
## 3  3.04829407 16.239424
## 4  1.32272770  4.924268
## 5  1.36274817 12.880561
## 6  1.08593225  5.943731
```

```
bioassay
```

```
##           x n y
## 1 -0.86 5 0
## 2 -0.30 5 1
## 3 -0.05 5 3
## 4  0.73 5 5
```

Exercise 1

a)

Given the prior distribution for each of the parameters as:

$$\alpha \propto N(0, 2^2)$$

$$\beta \propto N(10, 10^2)$$

And the correlation between them:

$$\text{corr}(\alpha, \beta) = 0.6$$

The full mean of the bivariate normal posterior is:

$$\bar{\mu} = (\mu_{\alpha}, \mu_{\beta}) = (0, 10)$$

The covariance matrix is:

$$\Sigma = \begin{pmatrix} \sigma_{\alpha}^2 & \rho\sigma_{\alpha}\sigma_{\beta} \\ \rho\sigma_{\alpha}\sigma_{\beta} & \sigma_{\beta}^2 \end{pmatrix} = \begin{pmatrix} 2^2 & 0.6 \cdot 2 \cdot 10 \\ 0.6 \cdot 2 \cdot 10 & 10^2 \end{pmatrix} = \begin{pmatrix} 4 & 12 \\ 12 & 100 \end{pmatrix}$$

So, the posterior distribution is:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = N \left[\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} \sigma_{\alpha}^2 & \rho\sigma_{\alpha}\sigma_{\beta} \\ \rho\sigma_{\alpha}\sigma_{\beta} & \sigma_{\beta}^2 \end{pmatrix} \right]$$

```
# Posterior distribution
mean_vector <- c(0,10);
covariance_matrix <- matrix(c(4, 12, 12, 100),2)
joint_posterior <- rmvnorm(4000, mean=mean_vector, sigma=covariance_matrix)
head(joint_posterior)

##           [,1]      [,2]
## [1,] -1.7062636 -1.9298382
## [2,]  0.3692508 10.3314005
## [3,] -1.4720117 15.3077838
## [4,] -3.4293404  0.1977571
## [5,] -3.7702181  7.4512873
## [6,] -2.5641813 -0.9646812

print(covariance_matrix)

##           [,1] [,2]
## [1,]      4   12
## [2,]     12  100

print(mean_vector)

## [1]  0 10
```

b)

```
alpha_samples <- bioassay_posterior$alpha
beta_samples <- bioassay_posterior$beta
S = length(alpha_samples)

mean_alpha <- mean(alpha_samples)
mean_beta <- mean(beta_samples)
var_alpha <- var(alpha_samples)
var_beta <- var(beta_samples)

#Quantiles:

alpha_5 <- quantile(alpha_samples, probs = 0.05)
alpha_95 <- quantile(alpha_samples, probs = 0.95)

beta_5 <- quantile(beta_samples, probs = 0.05)
```

```

beta_95 <- quantile(beta_samples, probs = 0.95)

#Mean MCSE
mean_alpha_mcse <- sqrt(var_alpha/S)
mean_beta_mcse <- sqrt(var_beta/S)

mean_alpha

## [1] 0.9852263
mean_beta

## [1] 10.59648
mean_alpha_mcse

## [1] 0.01482435
mean_beta_mcse

## [1] 0.07560016

#Quantile MCSE
alpha_5_mcse <- mcse_quantile(alpha_samples, prob=0.05)
alpha_95_mcse <- mcse_quantile(alpha_samples, prob=0.95)
alpha_90_mcse <- mcse_quantile(alpha_samples, prob=0.9)
beta_5_mcse <- mcse_quantile(beta_samples, prob=0.05)
beta_95_mcse <- mcse_quantile(beta_samples, prob=0.95)
beta_90_mcse <- mcse_quantile(beta_samples, prob=0.9)

#print("quantiles alpha")
#alpha_5
#alpha_95
#alpha_5_mcse
#alpha_95_mcse
#alpha_90_mcse

#print("quantiles beta")
#beta_5
#beta_95
#beta_5_mcse
#beta_95_mcse
#beta_90_mcse

```

RESULTS:

The mean for α is 0.9852263 and the MCSE is 0.0148243, so $\mu_\alpha = 1.0$
The 5% quantile for α is -0.4675914 and the MCSE is 0.0260041159750117, so $quantile_{\alpha_5} = -0.5$
The 95% quantile for α is 2.6102028 and the MCSE is 0.0420634167918579, so $quantile_{\alpha_{95}} = 2.6$

The mean for β is 10.5964813 and the MCSE is 0.0756002, so $\mu_{\beta} = 10.6$
The 5% quantile for β is 3.9914032 and the MCSE is 0.0704312509185216, so $quantile_{\beta_5} = 4.0$
The 95% quantile for β is 19.3403654 and the MCSE is 0.241212887707373, so $quantile_{\beta_{95}} = 19$

c)

```
# Function for computing the log importance ratios
log_importance_weights <- function(alpha, beta){
  weights_array <- array(0, dim=c(1,length(alpha)))
  for (i in 1:length(alpha)) {
    weights_array[i] <- bioassaylp(alpha[i], beta[i], bioassay$x, bioassay$y, bioassay$n)
  }
  return(c(weights_array))
}

alpha <- c(1.896, -3.6, 0.374, 0.964, -3.123, -1.581)
beta <- c(24.76, 20.04, 6.15, 18.65, 8.16, 17.4)
weights <- round(log_importance_weights(alpha, beta),2)
head(weights)
```

```
## [1] -8.95 -23.47 -6.02 -8.13 -16.61 -14.57
```

The first six log importance ratios are -8.95, -23.47, -6.02, -8.13, -16.61, -14.57.

d)

```
# Function for computing the normalized importance ratios
normalized_importance_weights <-function(alpha,beta){
  weights_array_norm <- array(0, dim=c(1,length(alpha)))
  for (i in 1:length(alpha)) {
    weights_array_norm[i] <-
      exp(bioassaylp(alpha[i], beta[i], bioassay$x, bioassay$y, bioassay$n))
  }
  weights_array_norm <- c(weights_array_norm)
  sum_weights <- sum(weights_array_norm)
  weights_array_norm_final <- array(0, dim=c(1,length(alpha)))
  for (i in 1:length(weights_array_norm)){
    weights_array_norm_final[i] <- weights_array_norm[i]/sum_weights
  }
  weights_norm <- c(weights_array_norm_final)
  return(weights_norm)
}

alpha <- c(1.896, -3.6, 0.374, 0.964, -3.123, -1.581)
beta <- c(24.76, 20.04, 6.15, 18.65, 8.16, 17.4)
weights_normalized <-
  round(normalized_importance_weights(alpha = alpha, beta = beta),3)
head(weights_normalized)
```

```
## [1] 0.045 0.000 0.852 0.103 0.000 0.000
```

The first six normalized log importance ratios are 0.045, 0, 0.852, 0.103, 0, 0

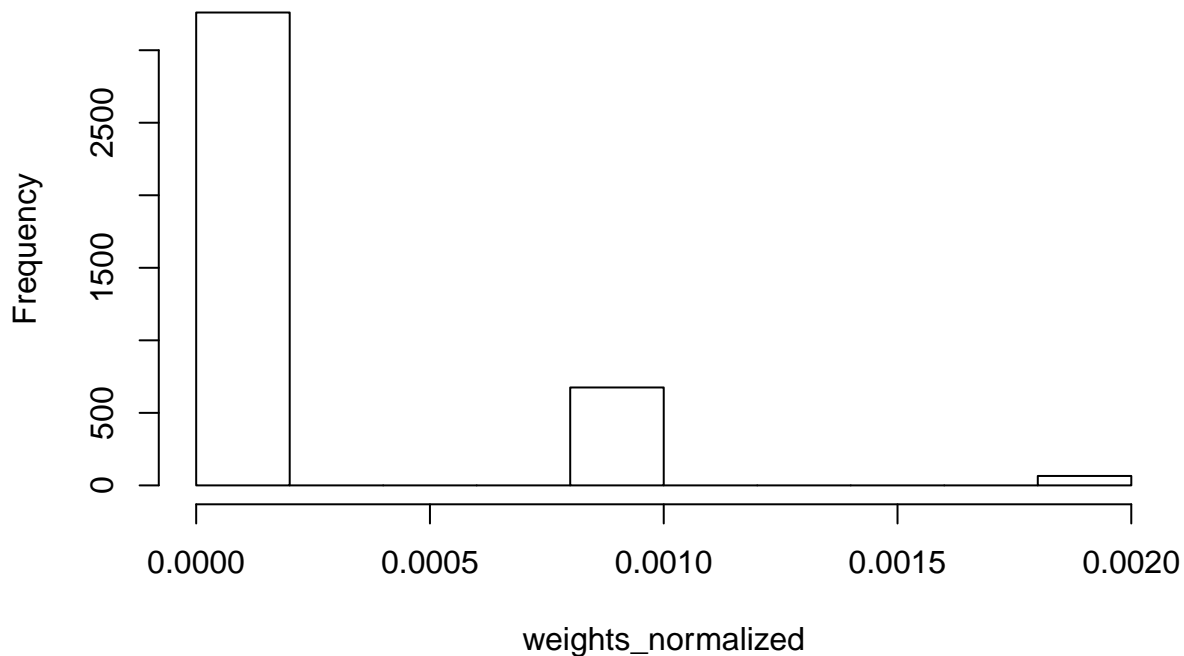
e)

```

n <- 4000
#alpha_samples <- rnorm(n, 0, 2)
#beta_samples <- rnorm(n, 10, 10)
alpha_samples <- (joint_posterior[,1])
beta_samples <- (joint_posterior[,2])
weights <- round(log_importance_weights(alpha_samples, beta_samples),2)
weights_normalized <- round(normalized_importance_weights(alpha = alpha_samples, beta = beta_samples),3)
hist(weights_normalized, main = "Histogram of the normalized weights")

```

Histogram of the normalized weights



```
head(weights_normalized)
```

```
## [1] 0.000 0.001 0.000 0.000 0.000 0.000
```

The first six log importance ratios are -27.41, -6.6, -13.3, -30.86, -19.39, -28.04.
The first six normalized log importance ratios are 0, 0.001, 0, 0, 0, 0.

f)

```

#alpha <- c(1.896, -3.6, 0.374, 0.964, -3.123, -1.581)
#beta <- c(24.76, 20.04, 6.15, 18.65, 8.16, 17.4)
weights <- round(log_importance_weights(alpha_samples, beta_samples),2)
weights_normalized <- round(normalized_importance_weights(alpha = alpha, beta = beta),3)
S_eff <- function(alpha, beta){
  weights_normalized <- normalized_importance_weights(alpha = alpha, beta = beta)
  den <- 0
  for (i in weights_normalized){
    den <- den + i^2
  }
}

```

```

    seff <- 1/den
    return(seff)
}

s_eff <- round(S_eff(alpha = alpha_samples, beta = beta_samples),3)
s_eff

## [1] 1165.868

```

The importance sampling effective sample size S_{eff} is 1165.868.

g)

If we had the target distribution, the effective sample size would be the number of independent samples drawn from the distribution so we would get an equivalent model. When used for computing an estimate, it means that the quality of the estimate would be the same using a number of samples equal to S_{eff} drawn from the target distribution than the one obtained using the proposal distribution with a certain number of samples (from which depends the value of S_{eff}). The effective sampling can be extracted from the histogram in exercise (e) looking at the frequency for those importance weights different from zero (frequency of 0.001 plus the frequency of 0.002).

h)

The expression for computing the posterior mean using importance sampling is:

$$E(h(\theta|y)) = \frac{\frac{1}{S} \sum_{S=1}^S h(\theta^S) w(\theta^S)}{\frac{1}{S} \sum_{S=1}^S w(\theta^S)}$$

```

# Function that computes the posterior mean using importance sampling
posterior_mean <- function(alpha, beta){
  weights_normalized <- round(normalized_importance_weights(alpha = alpha, beta = beta),3)
  alpha_posterior_mean <- 0
  beta_posterior_mean <- 0
  for (i in 1:length(alpha)){
    i
    alpha_posterior_mean <- alpha_posterior_mean + alpha[i]*weights_normalized[i]
    beta_posterior_mean <- beta_posterior_mean + beta[i]*weights_normalized[i]
  }
  return(c(alpha_posterior_mean, beta_posterior_mean))
}

alpha <- c(1.896, -3.6, 0.374, 0.964, -3.123, -1.581)
beta <- c(24.76, 20.04, 6.15, 18.65, 8.16, 17.4)
posterior_means <- posterior_mean(alpha=alpha_samples, beta=beta_samples)

# MEANS
posterior_means

## [1] 0.7641836 7.8746934

# MCSE
squared_alpha <- alpha_samples^2
squared_beta <- beta_samples^2

```

```

posterior_means_square <- posterior_mean(alpha=squared_alpha, beta=squared_beta)
var_alpha <- posterior_means_square[1]-posterior_means[1]^2
var_beta <- posterior_means_square[2]-posterior_means[2]^2

mcse_alpha <- sqrt(var_alpha/s_eff)
mcse_beta <- sqrt(var_beta/s_eff)

mcse_alpha

## [1] NaN

mcse_beta

## [1] NaN

```

Given the previous results:
 The posterior mean for alpha is 0.7641836 and the MCSE NaN, so $PosteriorMean_{\alpha} = 0.9$
 The posterior mean for beta is 7.8746934 and the MCSE NaN, so $PosteriorMean_{\beta} = 8$