HW#2 Calculating Posterior Probibilities and HDI's

Elijah Hall

April 25, 2018

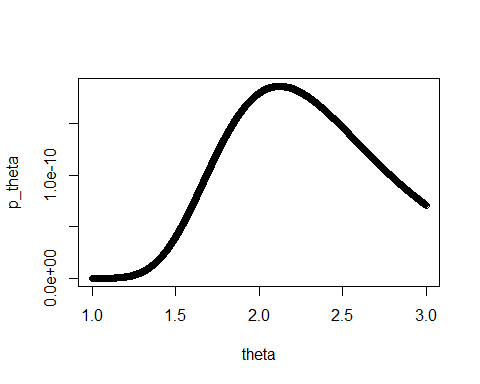
## Homework 2 Description

Suppose that a scale is known to be unbiased on average, but the standard deviation of its measurement error is unknown. A known 1 kg weight is placed on the scale 10 times, and the 10 resulting measurement errors (in mgs) are recorded as Y1,…,Y10. We assume that these 10 measurement errors are independently generated by a normal distribution with mean μ=0 and unknown standard deviation σ. Our prior uncertainty about the value of σ is represented by a uniform distribution over the interval [1,3].

### 1

Suppose I tell you that Y1,…,Y10 = -1.4132526, 0.8443594, -2.6628625, -3.5429051, 2.4482109, -3.7110128, -0.1110948, -1.5821031, -0.5346214,-0.3185383. Write an R function to compute the unnormalized posterior density of σ for arbitrary values of σ. (By unnormalized posterior density, we mean p(Y1,…,Y10|σ)p(σ), i.e. the posterior density with the denominator omitted.) For a fine grid of possible σ values between 1 and 3, compute the unnormalized posterior density and plot it.

#Error is N ~ (mu, σ)  
#mu = 0  
#σ = unknown  
  
#1)  
#likelyhood p(θ|σ)p(θ) s.t. p(θ) = 1/(3-1) since θ is uniformly distributed  
# σ prior U ~ {1,3} s.t. theta, θ, is probability distribution of σ's  
theta <- seq(from =1, to = 3, by =.0001)  
  
# calculating unnormalized p(θ|data) = p(data|θ)\*p(θ)  
  
#observed errors (data)  
Errors <- c(-1.4132526, 0.8443594, -2.6628625, -3.5429051, 2.4482109, -3.7110128,   
 -0.1110948, -1.5821031, -0.5346214,-0.3185383)  
  
#calculating unnormalized likelihood \* prior i.e. p(data|θ)\*p(θ) for every theta  
p\_theta<- sapply(seq(length(theta)), function(i) {  
prod(dnorm(Errors, mean = 0, sd = theta[i]))\*.5  
})  
  
#plot unnormalized distribution  
plot(theta, p\_theta)



### 2

What is the posterior probability that σ is between 1.5 and 2.5?

#normalize p\_theta   
norm <-p\_theta/sum(p\_theta)  
  
#calculate posterior probability  
sum(norm[theta <2.5 & theta >1.5])

## [1] 0.7169976

# 0.7169976

### 3

What is the 95% HDI of the posterior distribution of σ? The 50% HDI? Write your own function to compute HDIs.

names(norm)<-c(1:length(norm))  
  
head(norm)

## 1 2 3 4 5   
## 4.203970e-09 4.218716e-09 4.233508e-09 4.248347e-09 4.263233e-09   
## 6   
## 4.278165e-09

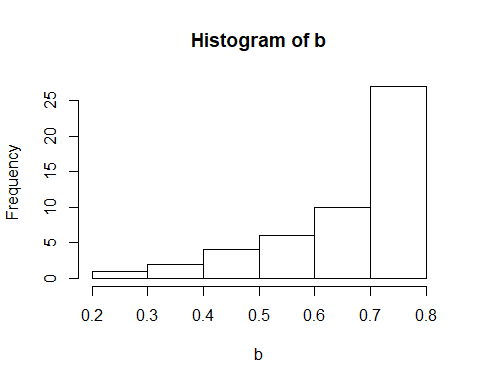
idx <- names(sort(norm, decreasing = TRUE))  
cumsums <- cumsum(norm[idx])  
  
idx\_min <- min(which(cumsums >.95))  
upper = max(theta[as.integer(idx[c(1:idx\_min)])])  
#2.9973  
lower = min(theta[as.integer(idx[c(1:idx\_min)])])  
#1.6022  
  
idx\_min <- min(which(cumsums >.5))  
upper = max(theta[as.integer(idx[c(1:idx\_min)])])  
#2.4446  
lower = min(theta[as.integer(idx[c(1:idx\_min)])])  
#1.8643  
  
HDI <- function(data, theta, percentage ) {  
   
 p\_theta<- sapply(seq(length(theta)), function(i) {  
 prod(dnorm(data, mean = 0, sd = theta[i]))\*.5  
 })  
 names(p\_theta) <- c(1:length(p\_theta))  
 norm <-p\_theta/sum(p\_theta)  
   
names(norm)<-c(1:length(norm))  
idx <- names(sort(norm, decreasing = TRUE))  
cumsums <- cumsum(norm[idx])  
  
idx\_min <- min(which(cumsums >percentage))  
upper = max(theta[as.integer(idx[c(1:idx\_min)])])  
  
lower = min(theta[as.integer(idx[c(1:idx\_min)])])  
  
return(c(upper,lower))  
  
}

### 4

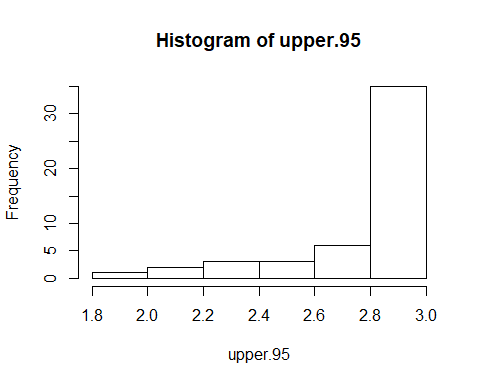
I generated Y1,…,Y10 using the R function rnorm(10,mean=0,sd=2), i.e. I chose 2 as the real value of σ. Repeat problems (a)-(c) 50 times simulating new data using rnorm(10,mean=0,sd=2) each time. Plot histograms of your 50 answers to (b) and your 50 lower and upper bounds of the HDIs from (c). Are these estimates volatile, i.e. do they vary a lot depending on the particular sample we observe?

Hints: For parts (b) and (c) create a normalized vector of densities by dividing each element of your grid of unnormalized densities by the sum of the entire vector. For part (d), use a for-loop.

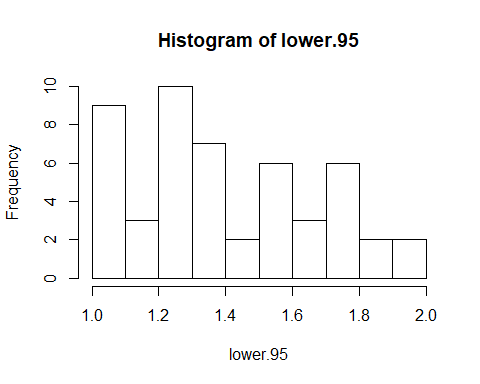
b <- c()  
upper.95<- c()  
lower.95 <- c()  
upper.5<-c()  
lower.5<-c()  
  
  
for(i in 1:50){  
 #generate 10 observations  
 set.seed(i)  
 Ys<- rnorm(10, 0,2)  
   
 #3)b. output  
 p\_theta<- sapply(seq(length(theta)), function(x) {  
 prod(dnorm(Ys, mean = 0, sd = theta[x]))\*.5  
 })  
 names(p\_theta) <- c(1:length(p\_theta))  
 norm <-p\_theta/sum(p\_theta)  
 b[i] <- sum(norm[theta <2.5 & theta >1.5])  
   
 #3)c. outputs  
HDI.95 <- HDI(Ys, theta, .95)  
upper.95[i]<- HDI.95[1]  
lower.95[i] <- HDI.95[2]  
  
HDI.5 <- HDI(Ys, theta, .5)  
upper.5[i]<-HDI.5[1]  
lower.5[i]<-HDI.5[2]  
}  
  
df\_3.d <-data.frame(b,upper.95,lower.95,upper.5,lower.5)  
  
hist(b)



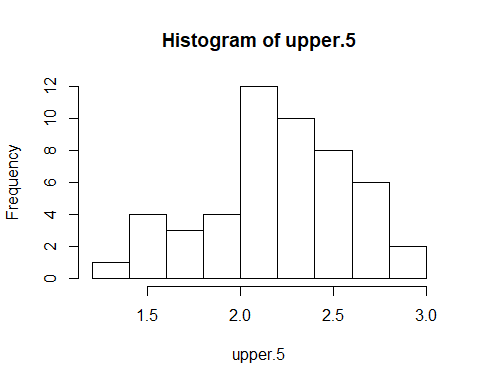
hist(upper.95)



hist(lower.95)



hist(upper.5)



hist(lower.5)

