Bayesian Computation Ch 2 Assignment

March 14, 2019

1. Simulate a sample from a posterior distribution when you have a histogram prior.

Solution: Let's start by defining a function to generate the values for the histogram prior:

```
#Function to generate prior values for each value in x
get_histprior_value <- function(x, histprior){
  vec <- rep(NA, length=length(x))
  for(i in 1:length(vec)){
    new_vec <- histprior[histprior[,1]>=x[i],]

    #Check if new_vec is matrix or vector and index accordingly
    if(!is.null(ncol(new_vec))){
      vec[i] <- new_vec[1,2]
    } else{
      vec[i] <- new_vec[2]
    }
  }

  return(vec)
}
```

Now let's prepare the data for input into our new `get_histprior_value` function:

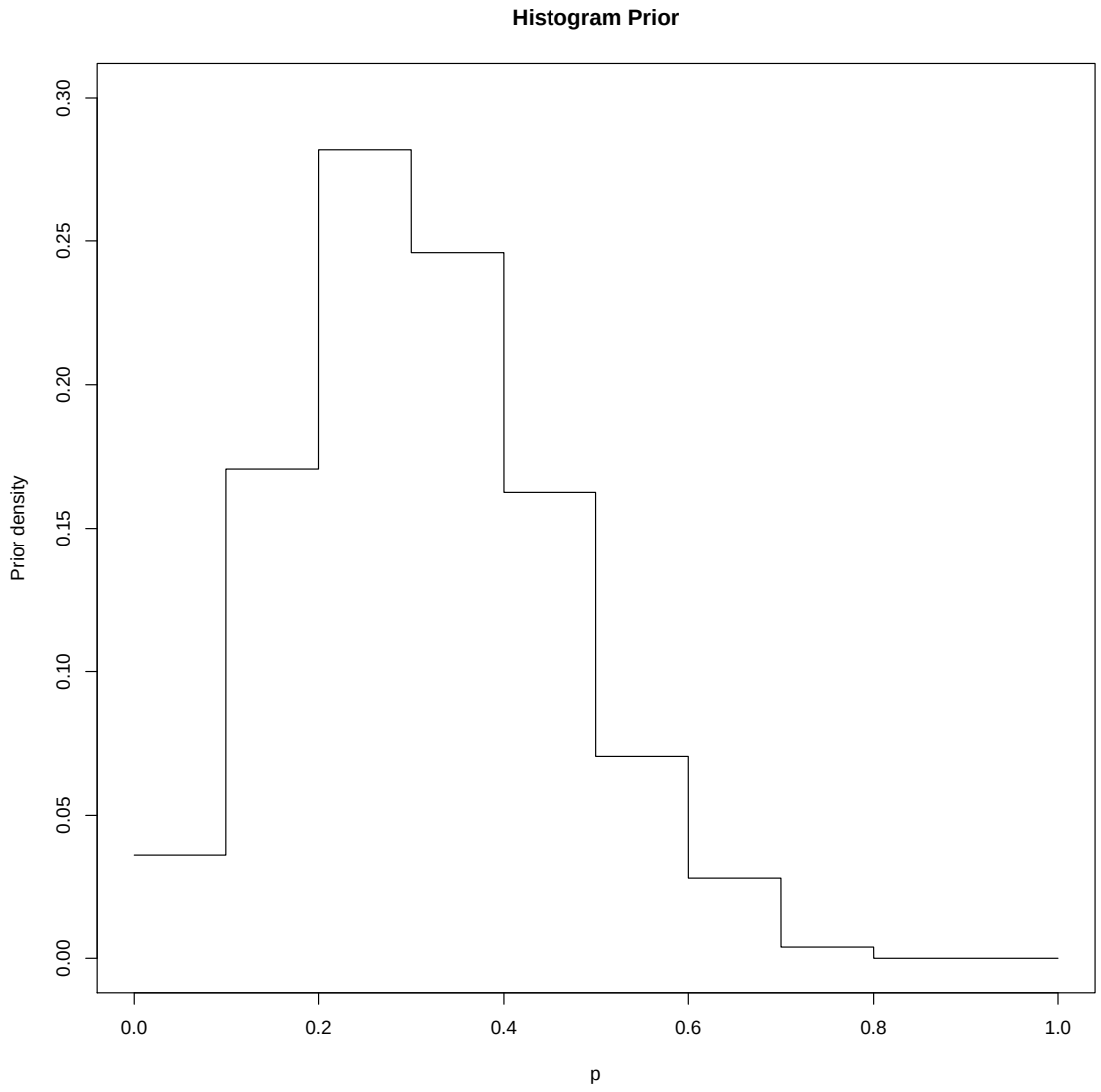
```
#Generate points for interval
interval <- seq(0.1, 1, by = 0.1)

#Prior probability
prior <- c(1, 5.2, 8, 7.2, 4.6, 2.1, 0.7, 0.1, 0, 0)
prior <- prior/sum(prior)

#Create the histogram prior
histprior <- sample(interval, 10000, replace=TRUE, prob = prior)
histprior <- table(histprior)/sum(table(histprior))
histprior <- as.matrix(histprior)
names <- rownames(histprior)
rownames(histprior) <- NULL
histprior <- cbind(as.numeric(names),as.numeric(histprior))
histprior <- rbind(histprior, c(0.9, 0))
histprior <- rbind(histprior, c(1.0, 0))
```

Now let's plot the prior:

```
#Output histogram of prior
curve(get_histprior_value(x, histprior), from=0, to = 1,
      ylim = c(0, 0.3), n = 10000,
      xlab="p", ylab = "Prior density", main = "Histogram Prior")
```



Finally, let's generate and plot the posterior:

```
s <- 11
f <- 16

p <- seq(0, 1, length = 10000)

post <- get_histprior_value(p, histprior) * dbeta(p, s+1, f+1)

post <- post/sum(post)

ps <- sample(p, replace = TRUE, prob = post)

hist(ps, xlab="p", main="Simulated Draws from the Posterior Distribution of p")
```