

calhoun-zach-homework5

Zach Calhoun

1/31/2022

Problem 1 (Hoff, 3.12)

Part A

First, we need to solve for the log likelihood.

The log likelihood is given by:

$$\log p(Y|\theta) = \log \binom{n}{y} \theta^y (1-\theta)^{n-y} \quad (1)$$

$$= y \log(\theta) + (n-y) \log(1-\theta) + \log \binom{n}{y} \quad (2)$$

Now that we have this value, we can solve for the first derivative:

$$\frac{\partial \log p(Y|\theta)}{\partial \theta} = \frac{y}{\theta} - \frac{n-y}{1-\theta}$$

And then the second derivative:

$$\frac{\partial^2 \log p(Y|\theta)}{\partial \theta^2} = -\frac{y}{\theta^2} - \frac{n-y}{(1-\theta)^2}$$

Now that we have the second derivative, we can replace y with $\mathbb{E}(y) = n\theta$ to get:

$$-\mathbb{E} \left[\frac{\partial^2 \log p(Y|\theta)}{\partial \theta^2} | \theta \right] = \frac{n\theta}{\theta^2} + \frac{n-n\theta}{(1-\theta)^2} = \frac{n}{\theta} + \frac{n}{1-\theta}$$

This further simplifies:

$$= \frac{n - n\theta + n\theta}{\theta(1-\theta)} = \frac{n}{\theta(1-\theta)}$$

Lastly, we take the square root to see:

$$p_J(\theta) \propto \sqrt{n} \theta^{-1/2} (1-\theta)^{-1/2}$$

Looking at the above, we want to find a distribution for θ that matches this expression with support over $(0, 1)$. The Beta distribution satisfies this criteria, and is proportional to the equation above when $a = 1/2$ and $b = 1/2$, so we can write as follows:

$$p_J(\theta) = \text{beta}(1/2, 1/2) \text{ for the binomial distribution}$$

Part B

First, get the log likelihood:

$$\begin{aligned}\log p(y|\psi) &= \log\left(\binom{n}{y} \exp(\psi y) (1 + \exp(\psi))^{-n}\right) \\ &= \log\left(\binom{n}{y}\right) + \log(\exp(\psi y)) - n \log(1 + \exp(\psi)) \\ &= \log\left(\binom{n}{y}\right) + \psi y - n \log(1 + \exp(\psi))\end{aligned}$$

The first derivative:

$$\frac{\partial \log p(y|\psi)}{\partial \psi} = y - \frac{n \exp(\psi)}{1 + \exp(\psi)}$$

The second derivative:

$$\frac{\partial^2 \log p(y|\psi)}{\partial \psi^2} = -\frac{n \exp(\psi)}{1 + \exp(\psi)} + \frac{n \exp(2\psi)}{(1 + \exp(\psi))^2} = \frac{-n \exp(\psi)}{(1 + \exp(\psi))^2}$$

Since we do not have any Y parameters in this equation, the negative expectation simply becomes:

$$-\mathbb{E}\left[\frac{\partial^2 \log p(Y|\psi)}{\partial \psi^2} | \psi\right] = \frac{n \exp(\psi)}{(1 + \exp(\psi))^2}$$

Given the value above, we can define Jeffrey's prior as being proportional to the squareroot of the expectation above:

$$p_J(\theta) \propto \exp(\psi/2)(1 + \exp(\psi))^{-1}$$

Part C

With $\theta = h(\psi) = \frac{\exp(\psi)}{1 + \exp(\psi)}$, we can solve for $dh/d\psi$:

$$\frac{dh}{d\psi} = \frac{\exp(\psi)}{(1 + \exp(\psi))^2}$$

We can now calculate $p_\psi(\psi)$ as follows:

$$\begin{aligned}p_\psi(\psi) &= \frac{1}{B(1/2, 1/2)} \left(\frac{\exp(\psi)}{1 + \exp(\psi)}\right)^{-0.5} \left(1 - \frac{\exp(\psi)}{1 + \exp(\psi)}\right)^{-0.5} \frac{\exp(\psi)}{(1 + \exp(\psi))^2} \\ &\propto \left(\frac{1 + \exp(\psi)}{\exp(\psi)}\right)^{0.5} \left(1 - \frac{1 + \exp(\psi)}{\exp(\psi)}\right)^{0.5} \frac{\exp(\psi)}{(1 + \exp(\psi))^2} \\ &\propto \left(\frac{1 + \exp(\psi)}{\exp(\psi)}\right)^{0.5} (1 + \exp(\psi))^{0.5} \frac{\exp(\psi)}{(1 + \exp(\psi))^2} \\ &\propto \frac{\exp(\psi/2)}{1 + \exp(\psi)}\end{aligned}$$

This is the same formula arrived at in part B, as expected.

Lab Component

Task 4

```
# For reproducibility
set.seed(1)

# Define target function
fx <- function(x) {
  return(sin(pi*x)^2)
}

sim_fun <- function(f, envelope = "unif", par1 = 0, par2 = 1, n = 10^2, plot = TRUE){

  # Store functions for generating a random sample and
  # for calculating the likelihood of that sample
  r_envelope <- match.fun(paste0("r", envelope))
  d_envelope <- match.fun(paste0("d", envelope))
  proposal <- r_envelope(n, par1, par2)

  # Density ratio is used so that a simple runif can be applied
  # against the density ratio (avoiding the use of q(x))
  density_ratio <- f(proposal) / d_envelope(proposal, par1, par2)

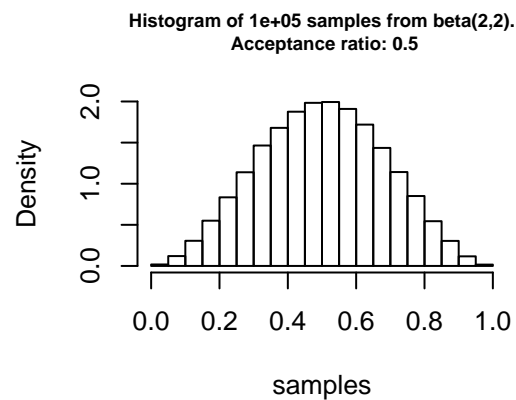
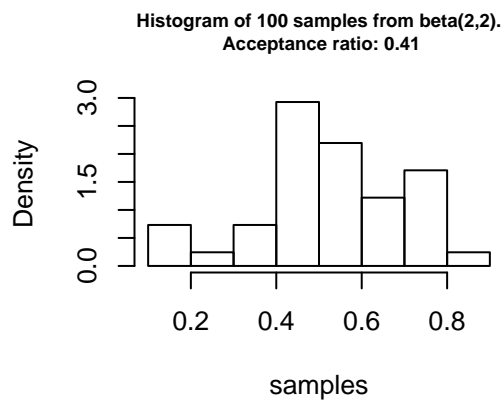
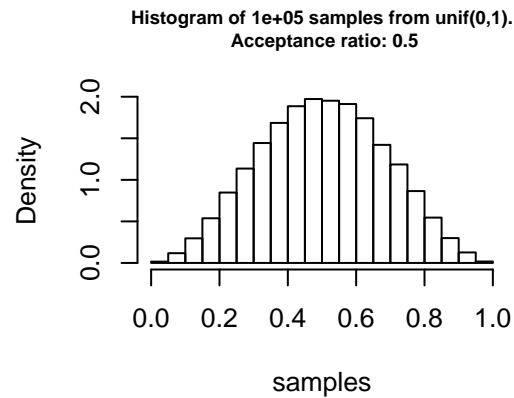
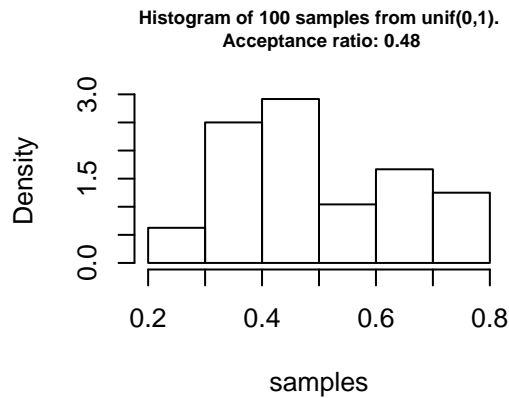
  # Keep sample if the randomly generated value is less
  # than the ratio.
  samples <- proposal[runif(n) < density_ratio]

  # Store the acceptance ratio for plotting.
  acceptance_ratio <- length(samples) / n
  if (plot) {
    hist(samples, probability = TRUE,
          main = paste0("Histogram of ",
                        n, " samples from ",
                        envelope, "(", par1, ",", par2,
                        ").\n Acceptance ratio: ",
                        round(acceptance_ratio, 2)),
          cex.main = 0.75)
  }
  list(x = samples, acceptance_ratio = acceptance_ratio)
}

# Set up the four plots
par(mfrow = c(2, 2), mar = rep(4, 4))

# Plot the uniform.
unif_1 <- sim_fun(fx, envelope = "unif", par1 = 0, par2 = 1, n = 10^2)
unif_2 <- sim_fun(fx, envelope = "unif", par1 = 0, par2 = 1, n = 10^5)

# Plot the betat distributions.
beta_1 <- sim_fun(fx, envelope = "beta", par1 = 2, par2 = 2, n = 10^2)
beta_2 <- sim_fun(fx, envelope = "beta", par1 = 2, par2 = 2, n = 10^5)
```



As shown in the graphs above, the acceptance ratio ends up being the same for both the uniform and the beta distribution. This can be rationalized by the fact that the area between both of these curves and the pdf for which we are estimating appear to be roughly the same, which means that proposed samples will be rejected at the same rate using both proposals. In both cases, the acceptance ratio is less with fewer samples. This is caused by random noise, since the acceptance ratio converges to its “true” value when increasing the number of samples.

Task 5

i: Without changing the constant c applied to both of the proposal enveloping functions, the performance of both is about the same.

ii: However, Beta(2,2) ends up being the better enveloping function when we apply a constant to decrease the area between the curve and the complicated pdf. This can be demonstrated by looking at the graph below:

```
set.seed(1)

fx <- function(x) {
  return(sin(pi*x)^2)
}

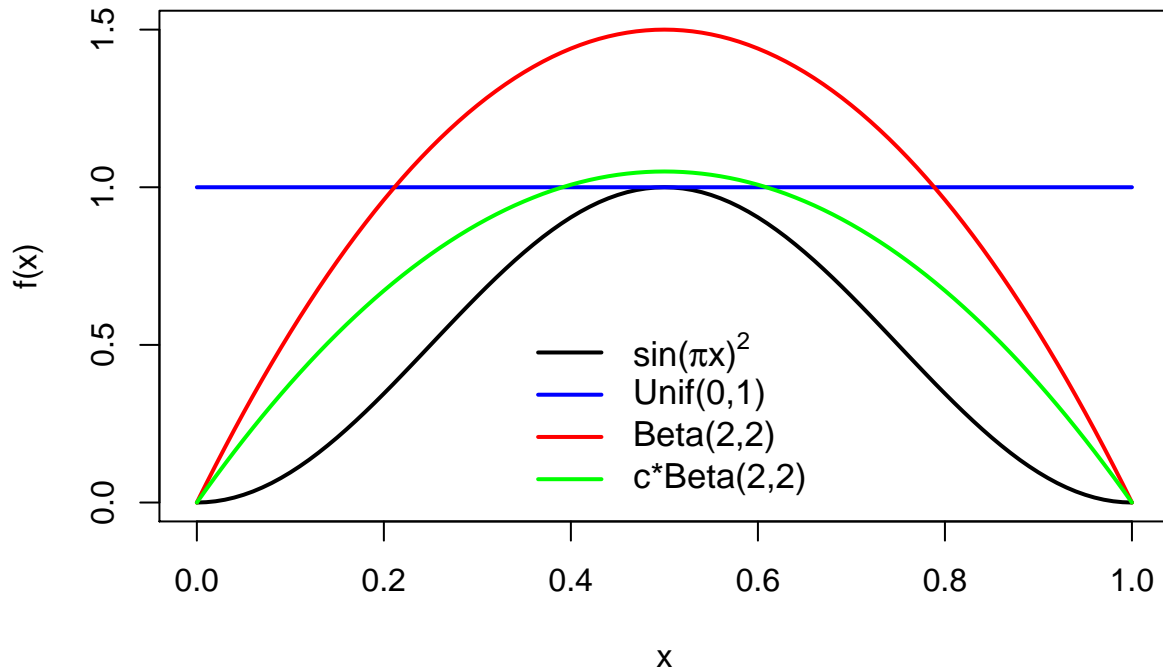
x <- seq(0, 1, by=0.01)

plot(fx, xlim=c(0,1), ylim=c(0,1.5), ylab="f(x)", lwd=2)
curve(dunif, add = TRUE, col= "blue", lwd=2)
curve(dbeta(x,2,2), add=TRUE, col="red", lwd=2)
curve(0.7*dbeta(x,2,2), add=TRUE, col="green", lwd=2)
```

```

legend("bottom", legend=c(expression(paste("sin(",pi,"x)"^2)), "Unif(0,1)","Beta(2,2)",
    "c*Beta(2,2)"), col=c("black","blue","red","green"),
    lty=c(1,1,1,1), bty= "n",cex=1.1, lwd=2)

```



In this case, I applied $c = 0.7$ to the Beta function to get the green curve, which is much closer to the complicated pdf we are trying to estimate, while still being greater than the sin function for all X . Because the area is decreased, we should expect that the acceptance ratio will be higher, as demonstrated below:

```

# This function is the exact same as above, except now there is a
# "c" parameter that allows the constant to be changed for the proposal
# function. The changed line of code is commented.
sim_fun <- function(f, c=1, envelope = "unif", par1 = 0, par2 = 1, n = 10^2, plot = TRUE){

  r_envelope <- match.fun(paste0("r", envelope))
  d_envelope <- match.fun(paste0("d", envelope))
  proposal <- r_envelope(n, par1, par2)
  density_ratio <- f(proposal) / d_envelope(proposal, par1, par2)

  # The line of code below is the only part that is changed!
  # Now, we sample on the uniform between 0 and c
  samples <- proposal[runif(n,0,c) < density_ratio]
  acceptance_ratio <- length(samples) / n
  if (plot) {
    hist(samples, probability = TRUE,
          main = paste0("Histogram of ",
                        n, " samples from ",

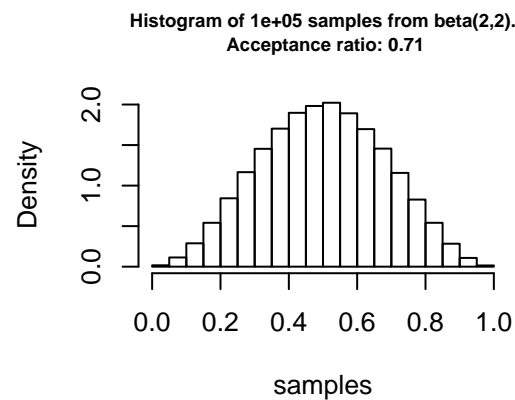
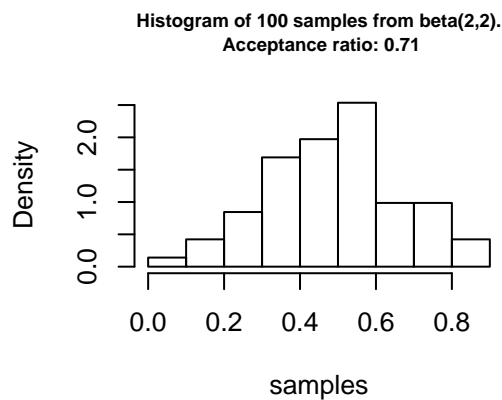
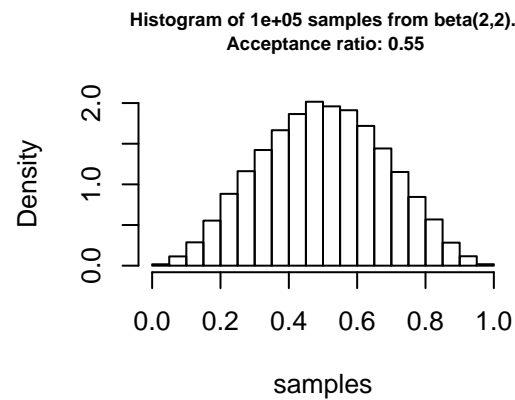
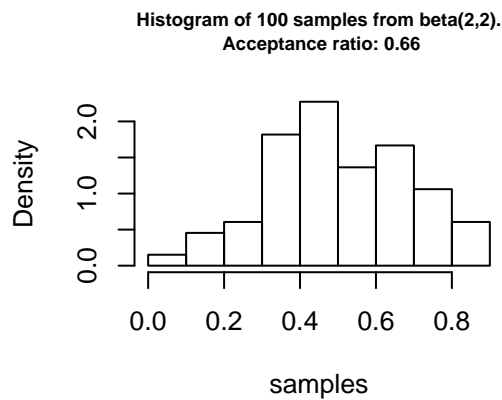
```

```

        envelope, "(", par1, ",", par2,
        ")\n Acceptance ratio: ",
        round(acceptance_ratio,2)),
        cex.main = 0.75)
    }
    list(x = samples, acceptance_ratio = acceptance_ratio)
}

# Let's compare two values of C to see how the acceptance ratio
# changes.
par(mfrow = c(2,2), mar = rep(4, 4))
beta_1 <- sim_fun(fx, c=0.9, envelope = "beta", par1 = 2, par2 = 2, n = 10^2)
beta_2 <- sim_fun(fx, c=0.9, envelope = "beta", par1 = 2, par2 = 2, n = 10^5)
# ATTN: You will need to add in the Beta(2,2) densities on your own to finish task 4.
beta_3 <- sim_fun(fx, c=0.7, envelope = "beta", par1 = 2, par2 = 2, n = 10^2)
beta_4 <- sim_fun(fx, c=0.7, envelope = "beta", par1 = 2, par2 = 2, n = 10^5)

```



The two rows above show using a value of $c = 0.9$ and $c = 0.7$, respectively. We can see that the acceptance ratio is increased to 0.72 when using $c = 0.7$, so we have a better performing sampler using that value of c .