STAT 7630: Homework 5
(Due: Thursday, 10/24/2024)

Note: Show all your work for the necessary steps to receive full credit.

Please turn in the HW on paper, hand-written and/or typed. For computational problems, return only the relevant parts of the output with comments/annotations. Questions taken from the textbook are marked with BR for "Bayes Rules!". From the code you are using to answer the problems, turn in the relevant output, and the figures (if requested), preferably printed from the output. No need to turn in your code or long lists of generated samples.

Please disclose any us of AI in your solutions. Regardless of you use it or not, make sure you submit your own work, not copy from other source(s). Any suspicion of AI use will result in automatic 0 or substantial point loss in any question.

BR Chapter 6:
Do Exercise 6.3. [Read Section 6.3 for more details about "mixing slowly."]

BR Chapter 7:
Do Exercises 7.6 and 7.7.

Hints:

- In 7.6 use `R` functions like `rnorm` and `runif` to draw the requested value, and (just for the purposes of this exercise!), remember to enter `set.seed(84735)` before random draws.
- In 7.7 use `R` functions like `dnorm`, `dunif`, and `dexp` to calculate the value of the proposal density for the specified current value and proposed value of the parameter. These proposal density values are the q parts that go in the numerator and denominator of the Metropolis-Hastings acceptance ratio probability.

Additional Questions (AQ):
AQ1. The Gibbs sampler can take a long time to converge if the target distribution is multimodal. Suppose we are trying to sample from a density for a parameter θ that is a mixture of three normal densities. Specifically, θ has density:

$$f(\theta) = 0.45\phi_1(\theta|\mu_1, \sigma_1^2) + 0.10\phi_2(\theta|\mu_2, \sigma_2^2) + 0.45\phi_3(\theta|\mu_3, \sigma_3^2),$$

where $\mu_1 = -3$, $\mu_2 = 0$, $\mu_3 = 3$, and $\sigma_1^2 = \sigma_2^2 = \sigma_3^2 = 1/3$, and $\phi_j(\cdot|\mu_j, \sigma_j^2)$ represents the normal density with mean μ_j and variance σ_j^2 for $j = 1, 2, 3$. The marginal distribution of θ is plotted in Figure 1.

Note that we could easily sample from this distribution using ordinary Monte Carlo methods, but consider using Gibbs sampling to sample from it.

(a) If the indicator $\delta \in \{1, 2, 3\}$, argue that the full conditional distribution for θ is:

$$\theta|\delta \sim N(\mu_\delta, \sigma_\delta^2).$$

What are μ_δ and σ_δ^2 here, for $\delta = 1, 2, 3$?

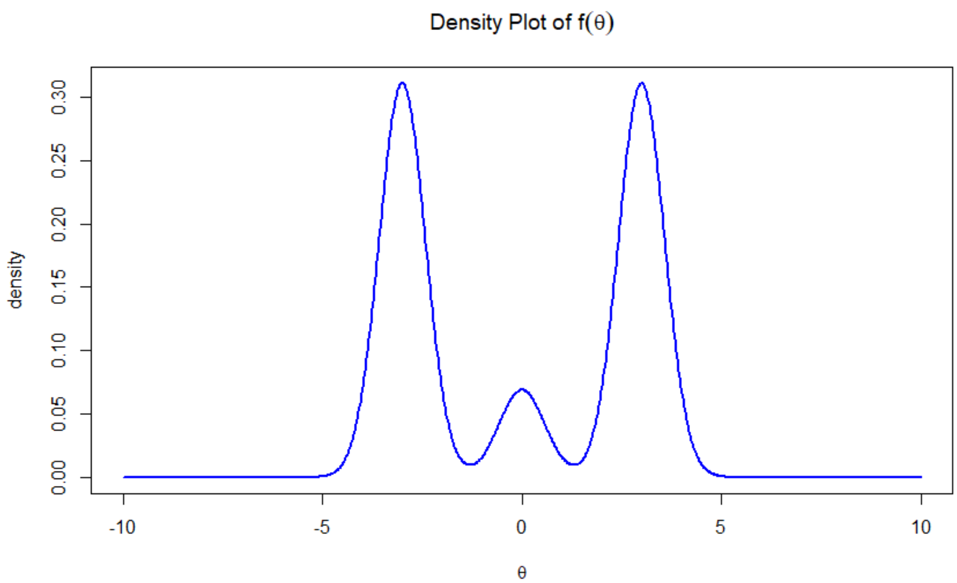

Figure 1: Plot of the multimodal density in AQ1.

(b) Use Bayes' theorem to show that the full conditional for δ is:

$$P(\delta = k|\theta) = \frac{P(\delta = k) \times \phi_k(\theta|\mu_k, \sigma_k^2)}{\sum_{m=1}^{3} P(\delta = m) \times \phi_m(\theta|\mu_m, \sigma_m^2)}, \quad \text{for } k \in \{1, 2, 3\},$$

(c) Write a Gibbs sampling algorithm (you can use the helpful `R` commands given on the course web page) to sample from the joint density of (θ, δ). Begin the chain with the initial values $\delta^{[0]} = 2$ and $\theta^{[0]} = 0$, and generate 1000 values of θ. Provide a plot of a relative frequency histogram of the θ values using a command like: `hist(theta.values, freq=F)` and comment on how it compares to the true marginal density of θ plotted on the last page. (You could try repeating part (c) a few times to get a sense of the variability in the Gibbs sampler results as well.)

(d) Repeat part (c), but generate 40,000 values of θ. Again, provide a plot of a relative frequency histogram of the θ values and comment on how it compares to the true marginal density of θ plotted on the last page. (You could try repeating part (d) a few times to get a sense of the variability in these Gibbs sampler results as well.)

(e) Try trace plots for the θ values and plots of the autocorrelation functions for parts (c) and (d). Comment on what these MCMC diagnostics tell you.

AQ2. Two separate test developers have created different IQ tests. Each of these tests is designed so that scores follow a normal distribution with population mean 100 and standard deviation 15. Hence any set of test scores can be standardized such that the standardized scores follow a $N(0, 1)$ distribution. We wish to investigate the coefficient of correlation ρ between test-takers' scores on the two tests. Consider a random sample of n test-takers, who each take both IQ tests. Let $X_1, \ldots, X_n$ be the test-takers' standardized scores on the first test, and let $Y_1, \ldots, Y_n$ be the corresponding standardized scores on the second test. The likelihood is given by:

$$L(\rho|\mathbf{x}, \mathbf{y}) = (2\pi)^{-n}(1-\rho^2)^{-n/2} \exp\left(-\frac{1}{2(1-\rho^2)}\left[\sum x_i^2 - 2\rho \sum x_i y_i + \sum y_i^2\right]\right).$$

(a) Suppose we use the prior $p(\rho) = 1$, $0 < \rho < 1$, for ρ. Show that the posterior for ρ is:

$$p(\rho|\mathbf{x}, \mathbf{y}) \propto (1-\rho^2)^{-n/2} \exp\left(-\frac{1}{2(1-\rho^2)}\left(\sum x_i^2 - 2\rho \sum x_i y_i + \sum y_i^2\right)\right).$$

(b) We gather data on 13 test-takers. Their standardized scores on the two tests are:

X_i : 0.92, 0.42, 3.62, 0.89, −0.69, 0.45, −0.11, −0.14, −0.47, 1.09, −0.34, 0.62, 0.27,

Y_i : 0.26, 1.65, 2.10, 0.62, −1.16, 1.29, −0.82, −0.36, −0.29, 0.86, 0.19, 1.25, 0.33.

Plugging in the necessary summary statistics, write and simplify the posterior (up to a constant of proportionality).

(c) We will use the following proposal density to generate values of ρ:

(i) Given the current value $\rho^{[t]}$, sample $\rho^* \sim \text{Uniform}(\rho^{[t]} - 0.2, \rho^{[t]} + 0.2)$.

(ii) If the sampled $\rho^* < 0$, then set $\rho^* = |\rho^*|$.

(iii) If the sampled $\rho^* > 1$, then set $\rho^* = 2 - \rho^*$.

Argue that this is a symmetric proposal density.

(d) Explain carefully and completely how the provided `R` code below performs the Metropolis-Hastings algorithm.

```
x<-c(0.92,0.42,3.62,0.89,-0.69,0.45,-0.11,-0.14,-0.47,1.09,-0.34,0.62,0.27)
y<-c(0.26,1.65,2.10,0.62,-1.16,1.29,-0.82,-0.36,-0.29,0.86,0.19,1.25,0.33)

sum.xsq <- sum(x^2)
sum.xy <- sum(x*y)
sum.ysq <- sum(y^2)
n <- length(x)

S <- 30000
rho.current <- 0.5 # initial value for M-H algorithm
acs <- 0 # will be to track "acceptance rate"

rho.values <- rep(0,times=S)  # will store sampled values of rho
```

```
for (s in 1:S) {
  rho.proposed <- runif(1,min=rho.current-0.2, max=rho.current+0.2)
  if (rho.proposed < 0) rho.proposed <- abs(rho.proposed)
  if (rho.proposed > 1) rho.proposed <- (2 - rho.proposed)

  log.accept.ratio <-  (-0.5*n*log(1-rho.proposed^2) -
  (1/(2*(1-rho.proposed^2))*(sum.xsq - 2*rho.proposed*sum.xy + sum.ysq))) -
  (-0.5*n*log(1-rho.current^2) -
  (1/(2*(1-rho.current^2))*(sum.xsq - 2*rho.current*sum.xy + sum.ysq)))

  if (log.accept.ratio > log(runif(1)) ) {
    rho.current <- rho.proposed
    acs <- acs + 1
  }

  rho.values[s] <- rho.current
}
```

(e) Using the code above in part (d), sample from the posterior distribution of ρ. Perform diagnostics to check convergence and check autocorrelation, and perform remedial action if needed. Summarize the posterior distribution of ρ, including providing an estimated density plot, point estimate, and 95% interval estimate for ρ.