

# Group Work 03

```
library(bayesrules) # R package for our textbook
library(tidyverse) # Collection of packages for tidying and plotting data
library(janitor) # Helper functions like tidy and tabyl
```

## Note

No group problems from Friday

## 1 BR Exercise 7.6: Proposing a new location

In each situation below, complete Step 1 of the Metropolis-Hastings algorithm. That is, starting from the given current chain value  $\lambda^{(i)} = \lambda$  and with `set.seed(84735)`, use the given proposal model to draw a  $\lambda'$  proposal value for the next value in the chain  $\lambda^{(i+1)}$

- (a)  $\lambda = 4.6, \lambda' | \lambda \sim N(\lambda, 2^2)$
- (b)  $\lambda = 2.1, \lambda' | \lambda \sim N(\lambda, 7^2)$
- (c)  $\lambda = 8.9, \lambda' | \lambda \sim Unif(\lambda - 2, \lambda + 2)$
- (d)  $\lambda = 1.2, \lambda' | \lambda \sim Unif(\lambda - 0.5, \lambda + 0.5)$
- (e)  $\lambda = 7.7, \lambda' | \lambda \sim Unif(\lambda - 3, \lambda + 3)$
- (f) Discuss why these proposal models “make sense”

## 2 BR Exercise 7.7: Calculate the acceptance probability

Suppose that a Markov chain is currently at  $\lambda^{(i)} = 2$  and the proposal for  $\lambda^{(i+1)}$  is  $\lambda' = 2.1$ . For each pair of unnormalized pdf  $f(\lambda)L(\lambda|y)$  and proposal  $q(\lambda'|\lambda)$ , calculate the acceptance probability  $\alpha$  used in Step 2 of the Metropolis-Hastings algorithm.

- (a)  $f(\lambda)L(\lambda|y) = \lambda^{-1}, \lambda' | \lambda \sim N(\lambda, 2^2)$  with pdf  $q(\lambda'|\lambda)$
- (b)  $f(\lambda)L(\lambda|y) = e^{3\lambda}, \lambda' | \lambda \sim N(\lambda, .5^2)$  with pdf  $q(\lambda'|\lambda)$
- (c)  $f(\lambda)L(\lambda|y) = e^{-1.9\lambda}, \lambda' | \lambda \sim Unif(\lambda - .3, \lambda + .3)$  with pdf  $q(\lambda'|\lambda)$

- (d)  $f(\lambda)L(\lambda|y) = e^{-\lambda^4}$ ,  $\lambda'|\lambda \sim \text{Exp}(\lambda)$  with pdf  $q(\lambda'|\lambda)$
- (e) For which of these scenarios is there a 100% acceptance probability? Explain why we'd certainly want to accept  $\lambda$  in these scenarios

### 3 BR 7.14

Consider a Gamma-Poisson model in which rate  $\lambda$  has a Gamma(1,0.1) prior and you observe one Poisson data point,  $Y = 4$ . In this exercise, you will simulate the posterior model of  $\lambda$  using an independence sampler.

- (a) Which of the following would make the most reasonable proposal model to use in your independence sampler for  $\lambda$ : Normal, Beta, or Exponential? Hint: Recall that  $\lambda > 0$ .
- (b) Using the proposal model that you identified in part a, simulate a tour of 1000  $\lambda$  values. Tune your algorithm until you are satisfied and produce a trace plot of your final results.
- (c) This is a situation where we can derive the exact posterior model of  $\lambda$ . What is it?
- (d) Plot a histogram of your tour and overlay the exact model. How good is your Markov chain approximation?

## 4 Independence Sampler for truncated normal prior

Metropolis-Hastings and Independence Samplers are powerful when we *don't* have a closed-form solution. Suppose we instead use a truncated normal model for  $\lambda$ . This pdf *looks* like a normal distribution, but is truncated at lower and/or upper bounds. The code chunk below loads the {truncnorm} package, which gives you access to `dtruncnorm`, `rtruncnorm`, etc, and plots a truncated normal with parameters  $\mu = 0$ ,  $\sigma = 1$  and  $a = 0$ .

- (a) Test different values of `a`, `mean`, and `sd` to get a sense of what each parameter does. Provide explanations of each.
- (b) Tune the truncated normal prior to cover a similar range as the Gamma distribution above
- (c) Modify your independence sampler from the previous question to use a truncated normal prior distribution. Tune your algorithm until you are satisfied and produce a trace plot of your final results.
- (d) Compare your posterior to the posterior from the last question.

```
library(truncnorm)

ggplot(data = data.frame(x = c(-4, 4)), aes(x = x)) +
  stat_function(fun = dtruncnorm,
               args = list(a = 0, mean = 0, sd = 1))
```

