Jared Yu

Computational Statistics

Problem Set 5

Note: The code for the assignment is at the end in the Appendix.

1. Assume that the instrumental distribution g in importance sampling (with unstandardized weights) is chosen such that

$$f(x) < M \cdot g(x)$$

for all x and a suitable $M \in \mathbb{R}$, where f is the target distribution.

a. Show that $Var_g\left(w^*(X)\right) < M - 1$.

$$Var_g(w^*(X)) = E[w^*(X)^2] - E[w^*(X)]^2$$

$$= \int w^*(x)^2 g(x)dx - \left[\int w^*(x)g(x)dx\right]^2$$

$$= \int \frac{f(x)^2}{g(x)^2} g(x)dx - \left[\int \frac{f(x)}{g(x)} g(x)dx\right]^2$$

$$= \int \frac{f(x)^2}{g(x)} dx - \left[\int f(x)dx\right]^2$$

$$< \int Mf(x)dx - (1)^2$$

$$< M - 1$$

$$\therefore Var_g\left(w^*(X)\right) < M - 1 \blacksquare$$

b. Show that $Var_g(w^*(X) \cdot h(X))$ is finite, if $Var_f(h(x))$ is finite.

$$Var_f(h(x)) = E[h(x)^2] - E[h(x)]^2$$

$$= \int h(x)^2 f(x)dx - \left[\int h(x)f(x)dx\right]^2 < \infty \text{ (given that this is true)}$$

Then,

$$Var_g(w^*(X) \cdot h(X)) = E\{[w^*(X) \cdot h(X)]^2\} - E[w^*(X) \cdot h(X)]^2$$

$$= \int w^*(X)^2 h(x)^2 g(x)dx - \left[\int w^*(x)h(x)g(x)dx\right]^2$$

$$= \int \frac{f(x)^2}{g(x)^2} h(x)^2 g(x) dx - \left[\int \frac{f(x)}{g(x)} h(x) g(x) dx \right]^2$$

$$= \int \frac{f(x)^2}{g(x)} h(x)^2 dx - \left[\int f(x) h(x) dx \right]^2$$

$$< \int M f(x) h(x)^2 dx - \left[\int f(x) h(x) dx \right]^2$$

$$< M \int f(x) h(x)^2 dx - \left[\int f(x) h(x) dx \right]^2 < \infty$$

$$\because \int h(x)^2 f(x) dx - \left[\int h(x) f(x) dx \right]^2 < \infty \text{ (shown before)}$$

$$\therefore Var_g\left(w^*(X) \cdot h(X)\right) < \infty \text{ (i.e., it is finite)} \blacksquare$$

2. Assume that you want to sample from a $N(0,1)$ distribution using a $N(1,2)$ distribution as instrumental distribution. Draw a sample of size 1000 using importance sampling, calculate the weighted mean and weighted variance, and plot a histogram of the weighted sample, i.e., plot a histogram of the draws from $N(1,2)$ times the appropriate weights. How does this histogram compare to a draw directly from $N(0,1)$?

Using RStudio, the mean and variance of the weighted sample are as follows:

$Weighted\ Mean$	$Weighted\ Variance$
0.0256	1.2266

Below (Figure 1) is a histogram of both the weighted sample (left) and target distribution sample (right). The histogram of the weighted sample is heavily skewed towards the left, while maintaining a mean and variance close to the target distribution. In other words, the mean of the weighted sample is close to 0, and the weighted variance is close to 1. The goal of Importance Sampling is to perform an approximation like what is done in Monte Carlo Integration. However, the goal is not necessarily to sample values from an approximated distribution (unlike Rejection Sampling). Therefore, the fact that the two histograms of the samples from Importance Sampling and the $N(1,2)$ distribution look highly dissimilar isn't surprising and is something to be expected.

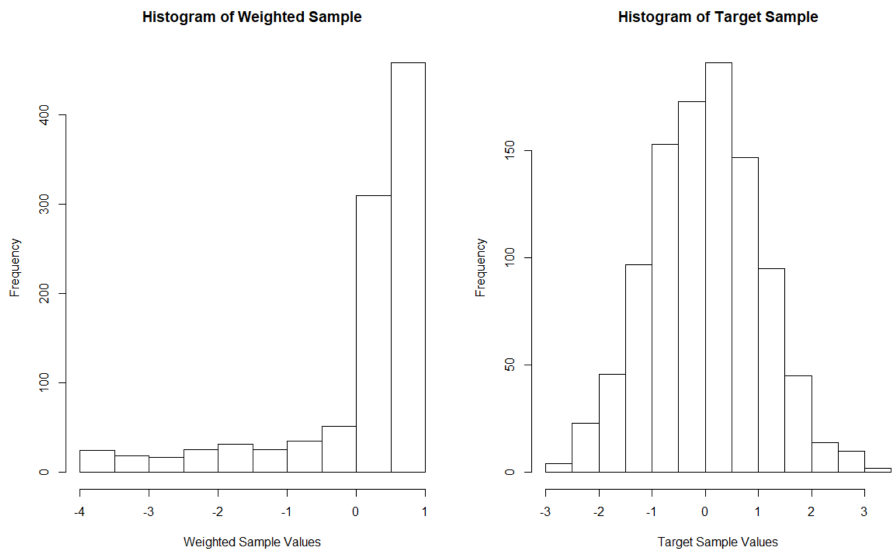

Figure 1

3. Suppose the general population's opinion of the federal government can be classified as "positive," "negative," or "it could be worse." If it is "positive" one day, then it is equally likely to be either "negative" or "it could be worse" the following day. If it is not "positive," then there is one chance in two chance that opinions will hold steady for another day and if it does change, then it is equally likely to become either of the other two opinions.
   a. What is the transition probability matrix for this Markov Chain?

Let $\mathcal{S} = \{0,1,2\}$, where 0 indicates "positive," 1 indicates "negative," and 2 indicates "it could be worse."

Then the following holds:

$$P\left[X^{(t+1)} = 0 | X^{(t)} = 0\right] = 0$$

$$P\left[X^{(t+1)} = 1 | X^{(t)} = 0\right] = P\left[X^{(t+1)} = 2 | X^{(t)} = 0\right] = \frac{1}{2}$$

$$P\left[X^{(t+1)} = 1 | X^{(t)} = 1\right] = P\left[X^{(t+1)} = 2 | X^{(t)} = 2\right] = \frac{1}{2}$$

$$P\left[X^{(t+1)} = 1 | X^{(t)} = 2\right] = P\left[X^{(t+1)} = 0 | X^{(t)} = 2\right] = P\left[X^{(t+1)} = 2 | X^{(t)} = 1\right] = P\left[X^{(t+1)} = 0 | X^{(t)} = 1\right] = \frac{1}{4}$$

Then the transition probability matrix $\boldsymbol{P}$ can be denoted as follows:

$$\boldsymbol{P} = \begin{bmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \end{bmatrix}$$

In the long run

b. How often does the general population hold a non-“negative” opinion of the government? (Hint: Remember that you need to solve the system of equations $\pi P = \pi$ and $\pi e = 1$.)

$$\pi\boldsymbol{P} = [\pi_0\ \pi_1\ \pi_2]\begin{bmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \end{bmatrix}$$

$$= \left[\frac{1}{4}\pi_1 + \frac{1}{4}\pi_2, \frac{1}{2}\pi_0 + \frac{1}{2}\pi_1 + \frac{1}{4}\pi_2, \frac{1}{2}\pi_0 + \frac{1}{4}\pi_1 + \frac{1}{2}\pi_2\right]$$

$$\begin{bmatrix} \pi_0 \\ \pi_1 \\ \pi_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{4}\pi_1 + \frac{1}{4}\pi_2 \\ \frac{1}{2}\pi_0 + \frac{1}{2}\pi_1 + \frac{1}{4}\pi_2 \\ \frac{1}{2}\pi_0 + \frac{1}{4}\pi_1 + \frac{1}{2}\pi_2 \end{bmatrix}$$

$$\mathbf{0} = \begin{bmatrix} -\pi_0 + \frac{1}{4}\pi_1 + \frac{1}{4}\pi_2 \\ \frac{1}{2}\pi_0 - \frac{1}{2}\pi_1 + \frac{1}{4}\pi_2 \\ \frac{1}{2}\pi_0 + \frac{1}{4}\pi_1 - \frac{1}{2}\pi_2 \end{bmatrix}$$

$$\mathbf{0} = \begin{bmatrix} -1 & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{4} & -\frac{1}{2} \end{bmatrix}\begin{bmatrix} \pi_0 \\ \pi_1 \\ \pi_2 \end{bmatrix}$$

$$R_2 - R_3 \to R_3 \Rightarrow \mathbf{0} = \begin{bmatrix} -1 & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{4} \\ 0 & -\frac{3}{4} & \frac{3}{4} \end{bmatrix} \begin{bmatrix} \pi_0 \\ \pi_1 \\ \pi_2 \end{bmatrix}$$

$$2R_2 \to 2R_2 \Rightarrow \mathbf{0} = \begin{bmatrix} -1 & \frac{1}{4} & \frac{1}{4} \\ 1 & -1 & \frac{1}{2} \\ 0 & -\frac{3}{4} & \frac{3}{4} \end{bmatrix} \begin{bmatrix} \pi_0 \\ \pi_1 \\ \pi_2 \end{bmatrix}$$

$$R_1 + R_2 \to R_2 \Rightarrow \mathbf{0} = \begin{bmatrix} -1 & \frac{1}{4} & \frac{1}{4} \\ 0 & -\frac{3}{4} & \frac{3}{4} \\ 0 & -\frac{3}{4} & \frac{3}{4} \end{bmatrix} \begin{bmatrix} \pi_0 \\ \pi_1 \\ \pi_2 \end{bmatrix}$$

$$R_2 - R_3 \to R_3 \Rightarrow \mathbf{0} = \begin{bmatrix} -1 & \frac{1}{4} & \frac{1}{4} \\ 0 & -\frac{3}{4} & \frac{3}{4} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \pi_0 \\ \pi_1 \\ \pi_2 \end{bmatrix}$$

$$-\frac{4}{3} R_2 \to R_2 \Rightarrow \mathbf{0} = \begin{bmatrix} -1 & \frac{1}{4} & \frac{1}{4} \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \pi_0 \\ \pi_1 \\ \pi_2 \end{bmatrix}$$

$$R_1 - \frac{1}{4} R_2 \to R_1 \Rightarrow \mathbf{0} = \begin{bmatrix} -1 & 0 & \frac{1}{2} \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \pi_0 \\ \pi_1 \\ \pi_2 \end{bmatrix}$$

$$\Rightarrow \begin{cases} -\pi_0 + \frac{1}{2}\pi_2 = 0 \\ \pi_1 - \pi_2 = 0 \\ \pi_0 + \pi_1 + \pi_2 = 1 \end{cases}$$

$$\Rightarrow \pi_0 = \frac{1}{2}\pi_2, \pi_1 = \pi_2$$

$$\frac{1}{2}\pi_2 + \pi_2 + \pi_2 = 1$$

$$\pi_2 = \frac{2}{5} = \pi_1$$

$$\pi_0 + \pi_2 = \frac{3}{5}$$

Therefore, the public generally holds a non-“negative” opinion $\frac{3}{5}$ of the time.

4. (a) Problem 7.2(a). You do not need to superimpose the true density on your histograms. Also, the **sample path** of a Markov Chain is a plot of the iteration number t versus the realizations of the random variable $X^{(t)}$ for $t = 0,1,2,\cdots$.

Below in Figure 2 is a grid of the three sample paths for each of the starting values of the Metropolis-Hastings algorithm. Based on the plot on the top-left for $x^{(0)} = 0$, it seems that the chain is having trouble getting towards a more desired area of around 7 and 10 in line with the means of the mixture distributions. Therefore, the pattern is to continuously edge upwards until it reaches that area and can bounce around to attempt to generate from the target distribution. This same pattern is seen in the sample path on the bottom-left for $x^{(0)} = 15$. In this bottom-left sample path, the chain seems to be trying to edge down towards the 7 and 10 range from 15, but is using all the iterations to try and descend towards this area. In the top-right sample path for $x^{(0)} = 7$, it seems that in the first 3,000 or so iterations it’s trying to edge upwards to create some balance in the generated samples. In other words, it’s trying to sample from both mixtures of a mean initially at 7 and then 10. Later, it’s able to bounce back and forth somewhat, but it’s not quite able to sample proportionately from these two regions within the number of iterations that it’s been tested on. In my opinion, I think that these are all examples of bad mixing.

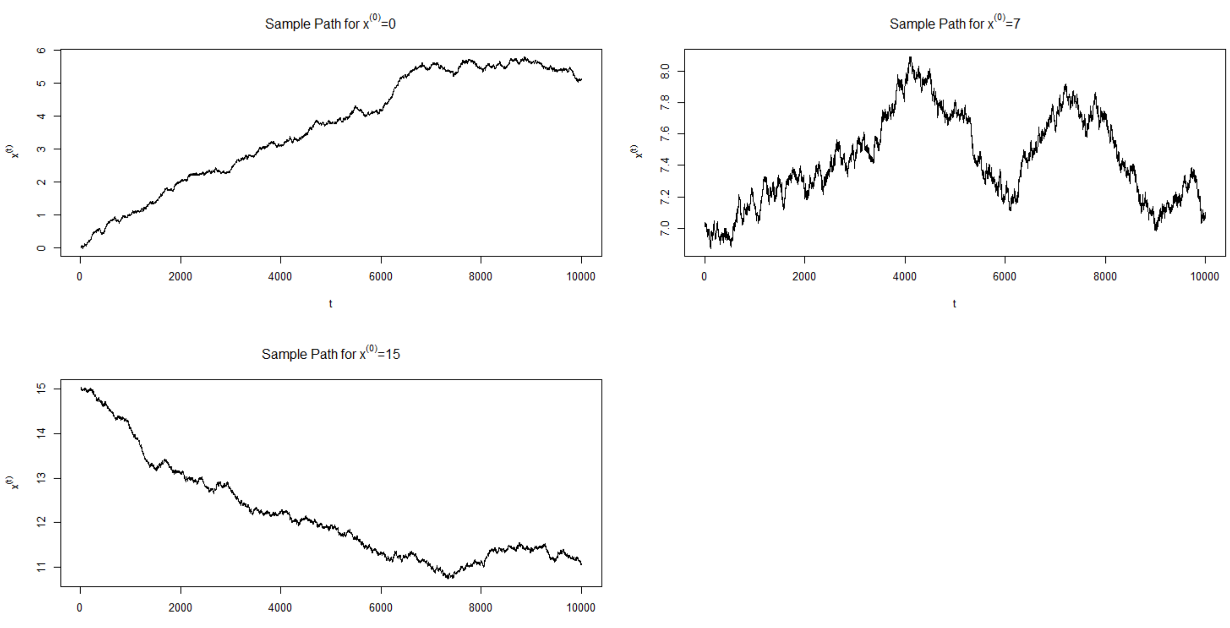

Figure 2

Below in Figure 3 is a grid of histograms for each of the simulations using the Metropolis-Hastings algorithm. In each of the histograms, it seems apparent that the chains are not behaving as intended. The ideal histogram would appear like the image of the histogram on p.205 of the textbook, where there is a bimodal distribution that has a larger peak around 7 and a smaller peak around 10. None of the histograms below resemble such a distribution and so based on these three chains it seems that the behavior of the chain is inconsistent with what is expected. A possible remedy is to try different initializations, run them for longer, and to account for a burn-in period before plotting the data. An alternate possibility is that the proposal distribution is inadequate and therefore a new one should be tried. Ideally, the sample path would fluctuate around the ideal locations of 7 and 10 in proportions like the mixtures of 0.7 and 0.3. The histogram ideally would also have the bimodal distribution with the appropriately proportioned 'humps.'

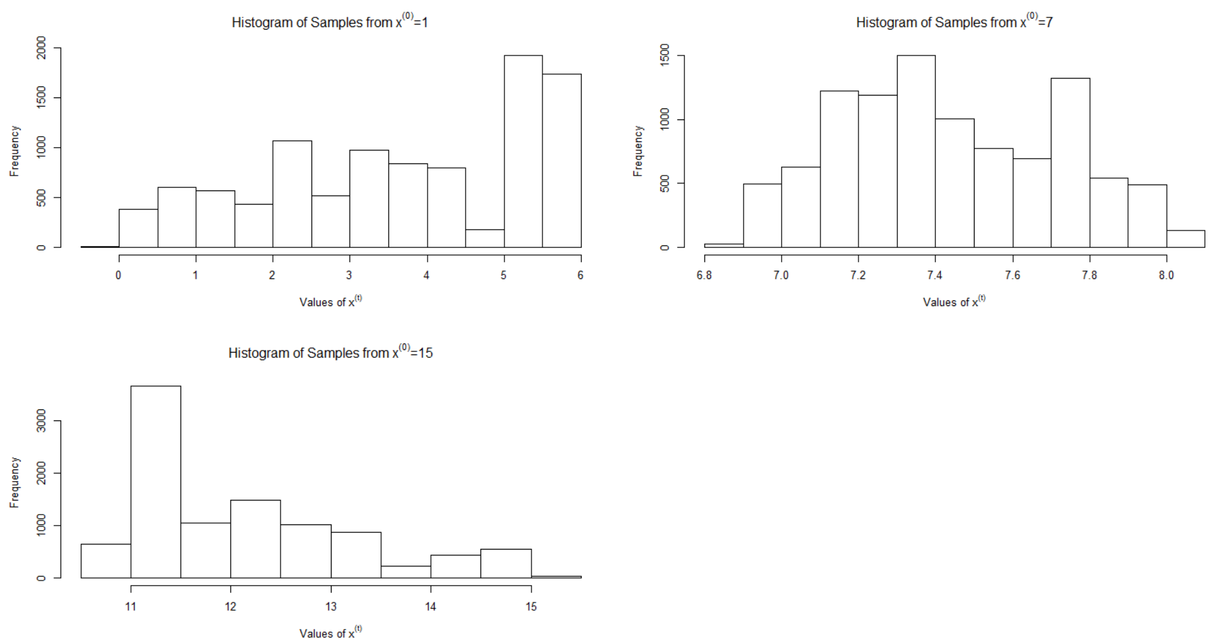

Figure 3

(b) Now change the proposal distribution in Problem 7.2(a) to a Uniform distribution on (0,20) with starting point 7. Does this proposal do any better? Run your code several times and see if you can get a histogram that resembles the expected double hump, then plot the sample path and discuss your results.

Below in Figure 4 is a histogram of the samples on the left-hand side and the sample path on the right-hand side. In this situation, there is clearly the double hump where there is an emphasis around 7 and a lesser emphasis around 10. This is like the histogram on p.205 of the textbook and shows that the chain can resemble the target distribution appropriately. The sample path on the right-hand side this time resembles good mixing, where the values gyrate proportionately between roughly around 7 and 10. This is in accordance with the expected range of values due to the mixture distribution also being focused in these areas. There seems to be an extremely brief burn-in period which is negligible and so it doesn't seem necessary to account for burn-in.

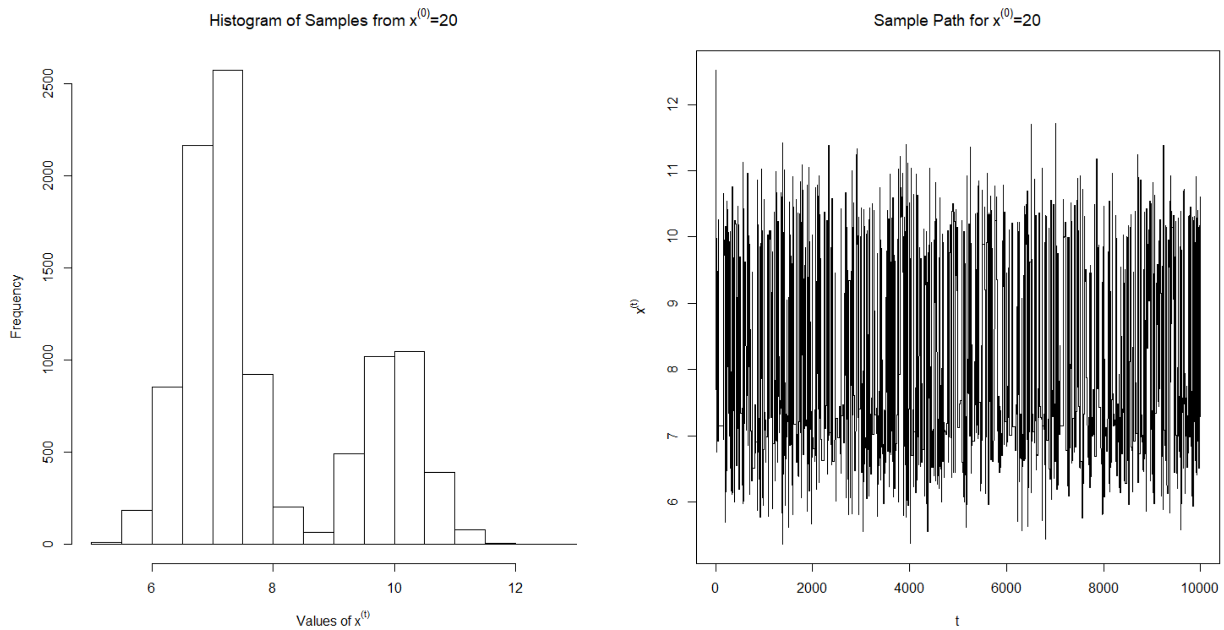

Figure 4

Appendix

```r
library(latex2exp)
### Problem 2
importance_weights <- function(x) {
  dnorm(x = x, mean = 0, sd = 1) /
    dnorm(x = x, mean = 1, sd = sqrt(2))
}

sample_size <- 1e3
set.seed(664)
target_draw <- rnorm(sample_size, # draw from target distribution
                     mean = 0, sd = 1)
instrumental_draw <- rnorm(sample_size, # draw from instrumental distribution
                           mean = 1, sd = sqrt(2))
weighted_sample <- instrumental_draw * # create weighted sample
  importance_weights(instrumental_draw)

mean(weighted_sample); var(weighted_sample) # 0.02561563, 1.226633

par(mfrow = c(1,2)) # plot
hist(weighted_sample, main = 'Histogram of Weighted Sample',
     xlab = 'Weighted Sample Values')
hist(target_draw, main = 'Histogram of Target Sample',
     xlab = 'Target Sample Values')
dev.off()
```

```r
### Problem 4
target_distribution <- function(x, delta = 0.7) { # target distribution
  delta * dnorm(x = x, mean = 7, sd = 0.5) +
    (1 - delta) * dnorm(x = x, mean = 10, sd = 0.5)
}

proposal_distribution <- function(input, given) { # proposal distribution
  dnorm(x = input, mean = given, sd = 1e-2)
}

proposal_sample <- function(x_t) { # sample from proposal
  rnorm(n = 1, mean = x_t, sd = 1e-2)
}

mh_ratio <- function(x_t, x_star) { # M-H ratio
  (target_distribution(x = x_star) * # numerator
     proposal_distribution(input = x_t, given = x_star)) /
    (target_distribution(x = x_t) * # denominator
       proposal_distribution(input = x_star, given = x_t))
}

x_0s <- c(0, 7, 15, 20) # possible x^(0)'s

mh <- function(x_init = x_0s[1], num_iterations = 1e4) {
  x_t <- x_init # Initialize variables
  sampling_matrix <- matrix(NA, nrow = num_iterations)

  for (index in 1:num_iterations) {
    x_star <- proposal_sample(x_t = x_t)
    ratio_r <- mh_ratio(x_t = x_t, x_star = x_star)

    if (ratio_r >= 1) { # first check
      sampling_matrix[index] <- x_star
      x_t <- x_star
    } else {
      u <- runif(n = 1)
      if (u < ratio_r) { # second check
        sampling_matrix[index] <- x_star
        x_t <- x_star
      } else {
        sampling_matrix[index] <- x_t
      }
    }
  }
  return(sampling_matrix)
}

sample_x0_1 <- mh(x_init = x_0s[1]) # generate samples
sample_x0_7 <- mh(x_init = x_0s[2])
sample_x0_15 <- mh(x_init = x_0s[3])

par(mfrow = c(2,2)) # plot sample paths and histograms
plot(1:1e4, sample_x0_1, type = 'l', main = TeX('Sample Path for $x^{(0)}=0$'),
     ylab = TeX('$x^{(t)}$'), xlab = TeX('$t$'))
plot(1:1e4, sample_x0_7, type = 'l', main = TeX('Sample Path for $x^{(0)}=7$'),
     ylab = TeX('$x^{(t)}$'), xlab = TeX('$t$'))
plot(1:1e4, sample_x0_15, type = 'l', main = TeX('Sample Path for $x^{(0)}=15$'),
     ylab = TeX('$x^{(t)}$'), xlab = TeX('$t$'))
dev.off()

par(mfrow = c(2,2))
```

```r
hist(sample_x0_1, main = TeX('Histogram of Samples from $x^{(0)}=1$'),
     xlab = TeX('Values of x^{(t)}'))
hist(sample_x0_7, main = TeX('Histogram of Samples from $x^{(0)}=7$'),
     xlab = TeX('Values of x^{(t)}'))
hist(sample_x0_15, main = TeX('Histogram of Samples from $x^{(0)}=15$'),
     xlab = TeX('Values of x^{(t)}'))
dev.off()


# part (b)
proposal_distribution2 <- function(input) {
  dunif(x = input, min = 0, max = 20)
}

proposal_sample2 <- function() {
  runif(n = 1, min = 0, max = 20)
}

mh_ratio2 <- function(x_t, x_star) {
  (target_distribution(x = x_star) * # numerator
     proposal_distribution2(input = x_t)) /
    (target_distribution(x = x_t) * # denominator
       proposal_distribution2(input = x_star))
}

mh2 <- function(x_init = x_0s[4], num_iterations = 1e4) {
  sampling_matrix <- matrix(NA, nrow = num_iterations)
  x_t <- x_init

  for (index in 1:num_iterations) {
    x_star <- proposal_sample2()
    ratio_r <- mh_ratio2(x_t = x_t, x_star = x_star)

    if (ratio_r >= 1) {
      sampling_matrix[index] <- x_star
      x_t <- x_star
    } else {
      u <- runif(n = 1)
      if (u < ratio_r) {
        sampling_matrix[index] <- x_star
        x_t <- x_star
      } else {
        sampling_matrix[index] <- x_t
      }
    }
  }
  return(sampling_matrix)
}

set.seed(664)
sample_x0_20 <- mh2(x_init = x_0s[4])
par(mfrow = c(1,2))
hist(sample_x0_20, main = TeX('Histogram of Samples from $x^{(0)}=20$'),
     xlab = TeX('Values of x^{(t)}'))
plot(1:1e4, sample_x0_20, type = 'l', main = TeX('Sample Path for $x^{(0)}=20$'),
     ylab = TeX('$x^{(t)}$'), xlab = TeX('$t$'))
dev.off()
```