Homework 2

Spencer Matthews

10/20/2021

Problem 1

Ehrenfest model of diffusion

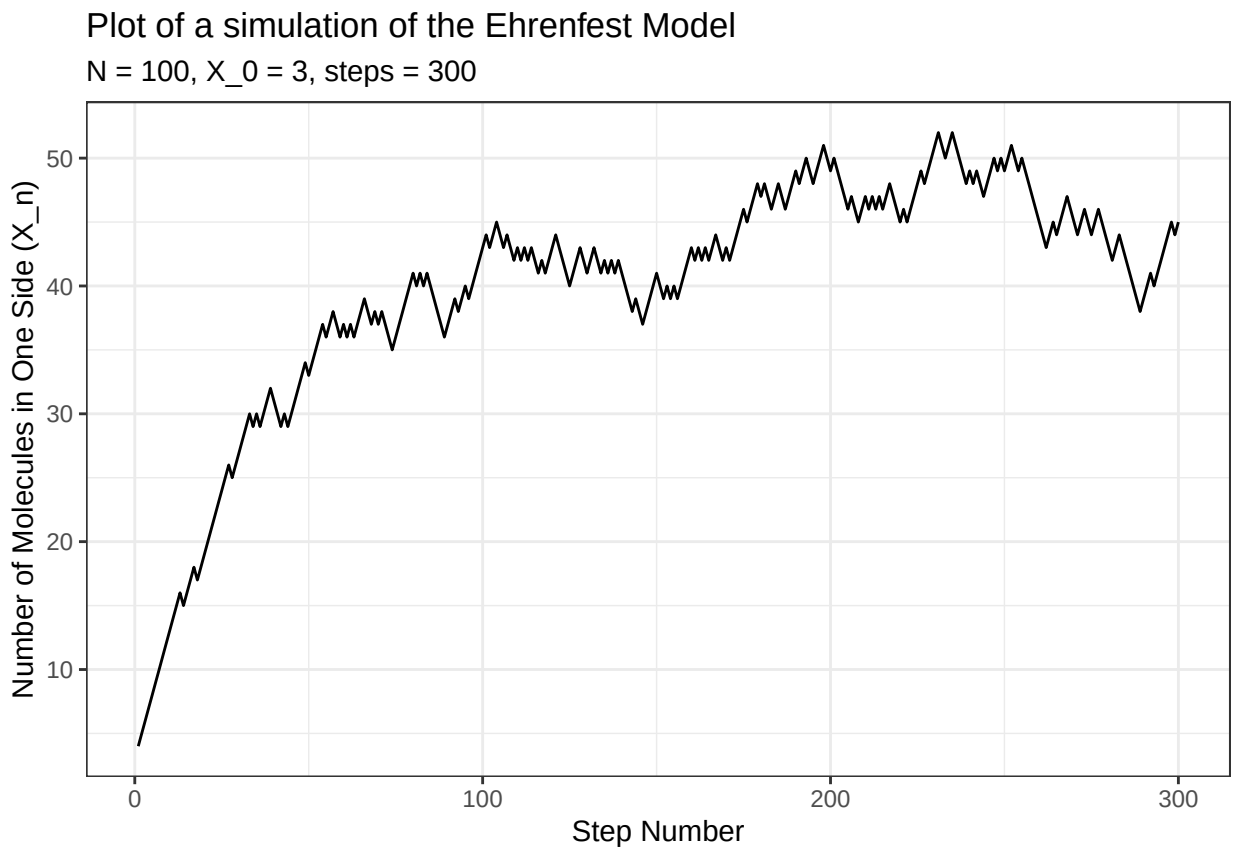
(a) *Implement a simulation of the Ehrenfest diffusion model and plot one realization of the chain with the total number of particles $N = 100$ and 300 time steps.*

First, we will create a function to simulate the Ehrenfest diffusion model.

```
ehrenfest_sim <- function(N, X_0, steps) {
  res <- rep(0, steps)
  X_cur <- X_0

  for (i in 1:steps) {
    p_lose <- X_cur / N
    if (runif(1) <= p_lose) X_cur <- X_cur - 1
    else X_cur <- X_cur + 1
    res[i] <- X_cur
  }
  res
}
```

Now we can run that function to plot one realization.

```
sim_res <- ehrenfest_sim(N = 100, X_0 = 3, steps = 300)

ggplot() +
  aes(x = 1:length(sim_res), y = sim_res) +
  geom_line() +
  theme_bw() +
  xlab("Step Number") +
  ylab("Number of Molecules in One Side (X_n)") +
  ggtitle("Plot of a simulation of the Ehrenfest Model", "N = 100, X_0 = 3, steps = 300")
```



(b) *Use the ergodic theorem to approximate the stationary variance of the number of particles and compare your estimate with the analytical result*

Recall that based on the ergodic theorem, we have that

$$E(i) = \sum i\pi_i = \lim_{N\to\infty} \frac{1}{N} \sum_{k=1}^{N} i$$

and thus

$$E(i^2) = \sum i^2\pi_i = \lim_{N\to\infty} \frac{1}{N} \sum_{k=1}^{N} i^2.$$

Using this information, we can approximate the variance using the shortcut formula $Var(X) = E(X^2) - E(X)^2$. Also note that the N in the above formula is actually equal to the `steps` parameter int he function. In this case, we will take `steps` to be very large, at 10,000. This simulates N (in the equation) approaching infinity.

```
set.seed(16)
new_dat <- ehrenfest_sim(100, 50, 10000)
sum(new_dat^2) / 10000 - mean(new_dat)^2
```

```
## [1] 25.01647
```

Thus the approximation of the variance using the ergodic theorem is 25.02.

We can now compare that to the analytical result. We know that the Ehrenfest model of diffusion has a stationary distribution of a Binomial model with n equal to the number of particles and p equal to 0.5. The variance of this distribution is $np(1-p)$, so we have the analytical variance

$$Var(X) = np(1-p) = 100(0.5)(0.5) = 25$$

Which is extremely close to what we obtained in our estimation using the ergodic theorem.

Problem 2

Recall that we used the following Metropolis-Hastings algorithm to update α and β in the beta-binomial hierarchical model. To propose new values of a positive parameter we multiply its current value by $e^{\lambda(U-0.5)}$ where $U \sim U[0,1]$ and λ is a tuning constant. Prove that the proposal density is

$$q(y_{\text{new}}|y_{\text{cur}}) = \frac{1}{\lambda y_{\text{new}}}$$

Proof

Note that the proposal density is simply the transformation of the uniform random variable, U. Because the main term in the transformation in e to the power of something, the transformation is guaranteed to be monotone increasing. We will let $g(u|y_{\text{cur}}) = e^{\lambda(U-0.5)}$. We see that since $U \sim Unif[0,1]$, $F_U(u) = 1, 0 \leq u \leq 1$. Furthermore, $g^{-1}(y_{\text{new}}|y_{\text{cur}}) = ln(y_{\text{new}})/\lambda$ and

$$\frac{d}{dy_{\text{new}}} g^{-1}(y_{\text{new}}) = \frac{1}{\lambda y_{\text{new}}}$$

This means that the proposal density is

$$q(y_{new}|y_{cur}) = f_U(u|y_{\text{cur}}) \cdot \left| \frac{d}{dy_{\text{new}}} g^{-1}(y_{\text{new}}|y_{\text{cur}}) \right| = 1 \cdot \frac{1}{\lambda y_{\text{new}}} = \frac{1}{\lambda y_{\text{new}}}$$

Hence, the proposal density is

$$q(y_{\text{new}}|y_{\text{cur}}) = \frac{1}{\lambda y_{\text{new}}}$$

Problem 3

Consider a two state continuous-time Markov SIS model, where the disease status X, cycles between the two states: 1 = susceptible, 2 = infected. Let the infection rate be λ_1 and clearance rate by λ_2. Suppose that an individual is susceptible at time 0 ($X_0 = 1$) and infected at time T, ($X_T = 2$). We don't know anything else about the disease status of this individual during the interval [0, T]. If T is small enough, it is reasonable to assume that the individual was infected only once during this time interval. We would like to obtain the distribution of the time of infection, I, conditional on the information we have.

(a) *Implement a Metropolis-Hastings sampler to draw realizations from the distribution* $P(I|X_0 = 1, X_t = 2, N_t = 1) \propto P(0 < t < I\colon X_t = 1, I \leq t < T\colon X_t = 2$, *where* N_t *is the number of infections.*

First, we will need to create the density function and the proposal function:

```
density_I <- function(I, lambda1, lambda2, big_t) {
  lambda1 * exp(-lambda1 * I) * exp(-lambda2 * (big_t - I))
}

get_proposal <- function(current_I, delta, big_t) {
  u <- runif(1, min = current_I - delta, current_I + delta)
```

```
  if (u < big_t & u > 0) return(u)
  else if (u >= big_t) return(2 * big_t - u)
  else return(-u)
}
```

Now, we can code up the Metropolis-Hastings algorithm:

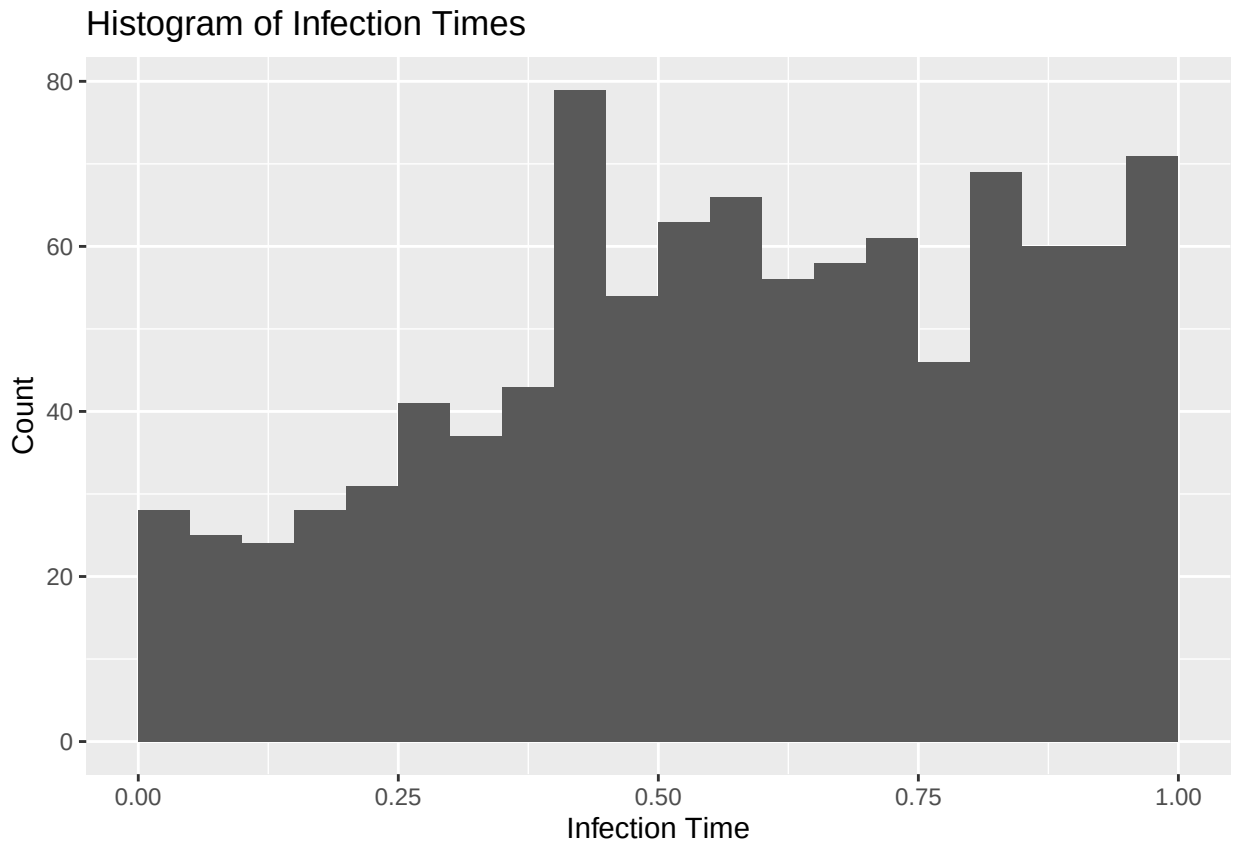
```
mh_sampler <- function(N, I0, lambda1, lambda2, big_t, delta) {
  I_curr <- I0
  res <- numeric(N)
  accepts <- integer(N)

  for (i in 1:N) {
    proposal <- get_proposal(I_curr, delta, big_t)
    acceptance_prob <- min(
      density_I(proposal, lambda1, lambda2, big_t) /
        density_I(I_curr, lambda1, lambda2, big_t),
      1
    )
    cutoff <- runif(1)
    if (acceptance_prob >= cutoff) {
      res[i] <- proposal
      accepts[i] <- 1
    } else {
      res[i] <- I_curr
      accepts[i] <- 0
    }
    I_curr <- res[i]
  }
  print(
    stringr::str_c(
      "Acceptance rate: ",
      round(mean(accepts), 4) * 100,
      "%"
    )
  )
  res
}
```

(b) *Run your MCMC for 1000 iterations and plot the histogram of the posterior distribution of the infection time.*

```
set.seed(100)
res <- mh_sampler(
  N = 1000,
  I0 = 0.5,
  lambda1 = 0.1,
  lambda2 = 0.5,
  big_t = 1,
  delta = 0.2
)
```

```
## [1] "Acceptance rate: 97.8%"
```

```
ggplot() +
  aes(x = res) +
  geom_histogram(binwidth = 0.05, boundary = 0) +
  xlab("Infection Time") +
  ylab("Count") +
  ggtitle("Histogram of Infection Times")
```



(c) *Try a couple of sets of values for λ_1 and λ_2 and examine the effect of these changes on the posterior distribution of the infection time. Comment on the effect of the relationship between λ_1 and λ_2 on the shape of the infection time posterior density/histogram.*

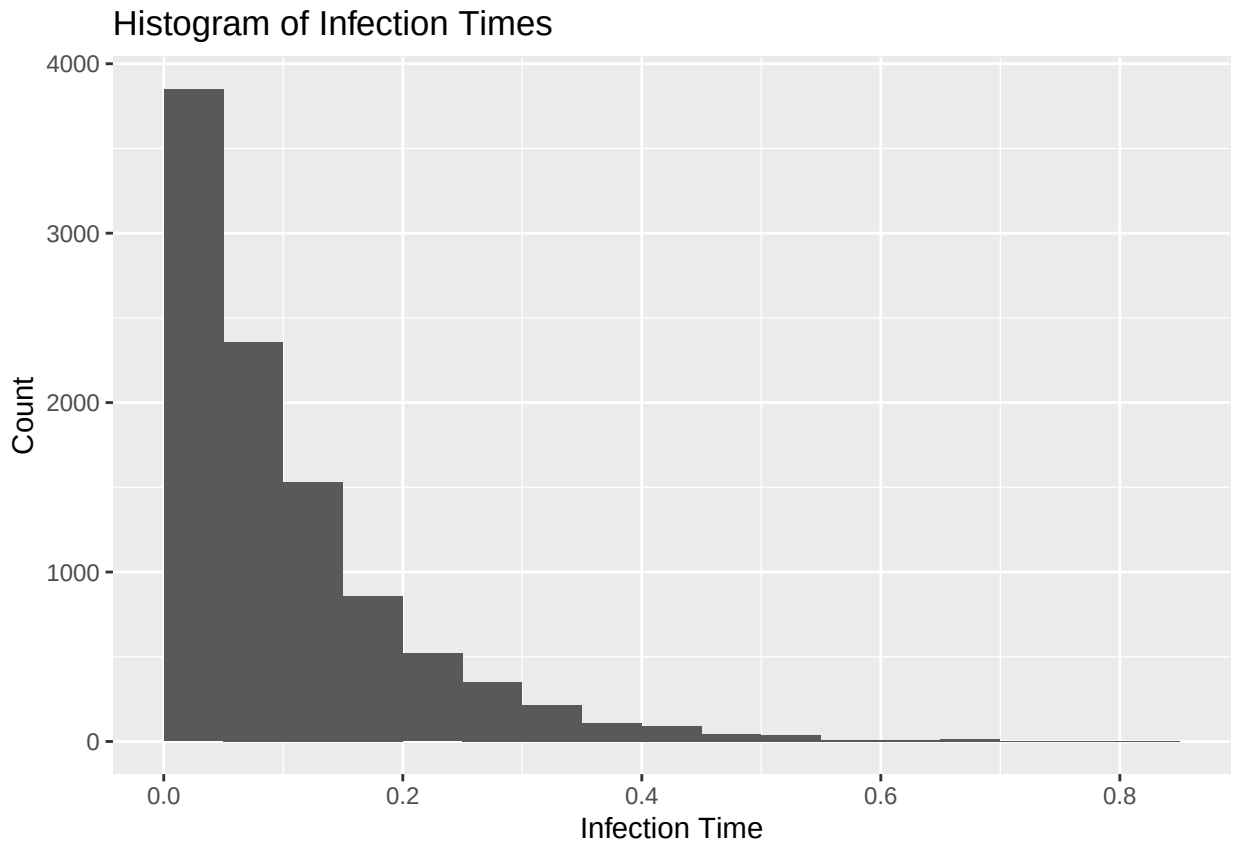
Let's try three different cases, one where λ_1 is much larger than $lambda_2$, one where the opposite is true, and one where they are the same. Here we will use 10,000 samples to get a better idea of the final distribution.

First, $\lambda_1 >> \lambda_2$.

```
set.seed(100)
res_1 <- mh_sampler(
  N = 10000,
  I0 = 0.5,
  lambda1 = 10,
  lambda2 = 0.1,
  big_t = 1,
  delta = 0.2
)
```

```
## [1] "Acceptance rate: 64.71%"
```

```
ggplot() +
  aes(x = res_1) +
  geom_histogram(binwidth = 0.05, boundary = 0) +
  xlab("Infection Time") +
  ylab("Count") +
  ggtitle("Histogram of Infection Times")
```



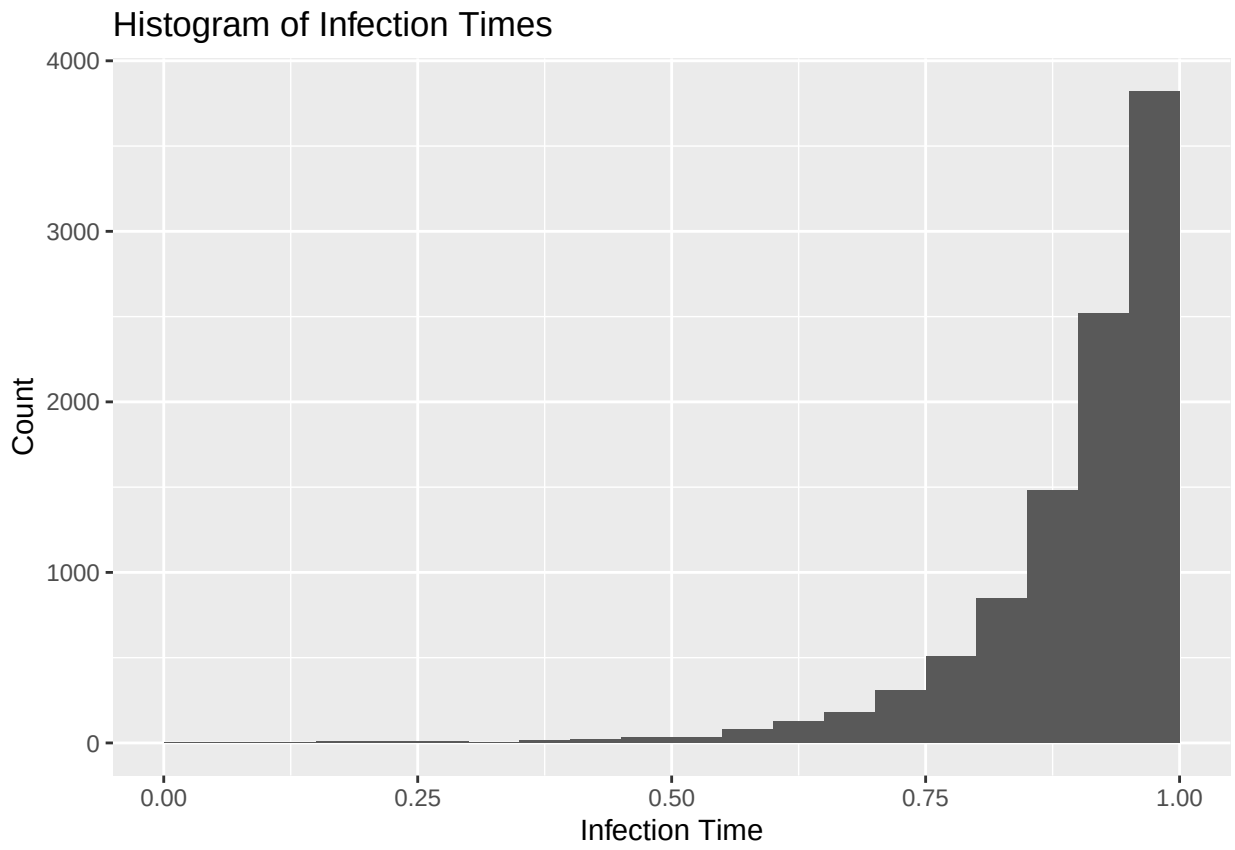
Now, $\lambda_1 << \lambda_2$.

```
set.seed(100)
res_2 <- mh_sampler(
  N = 10000,
  I0 = 0.5,
  lambda1 = 0.1,
  lambda2 = 10,
  big_t = 1,
  delta = 0.2
)
```

```
## [1] "Acceptance rate: 63.99%"
```

```
ggplot() +
  aes(x = res_2) +
  geom_histogram(binwidth = 0.05, boundary = 0) +
  xlab("Infection Time") +
```

```
  ylab("Count") +
  ggtitle("Histogram of Infection Times")
```

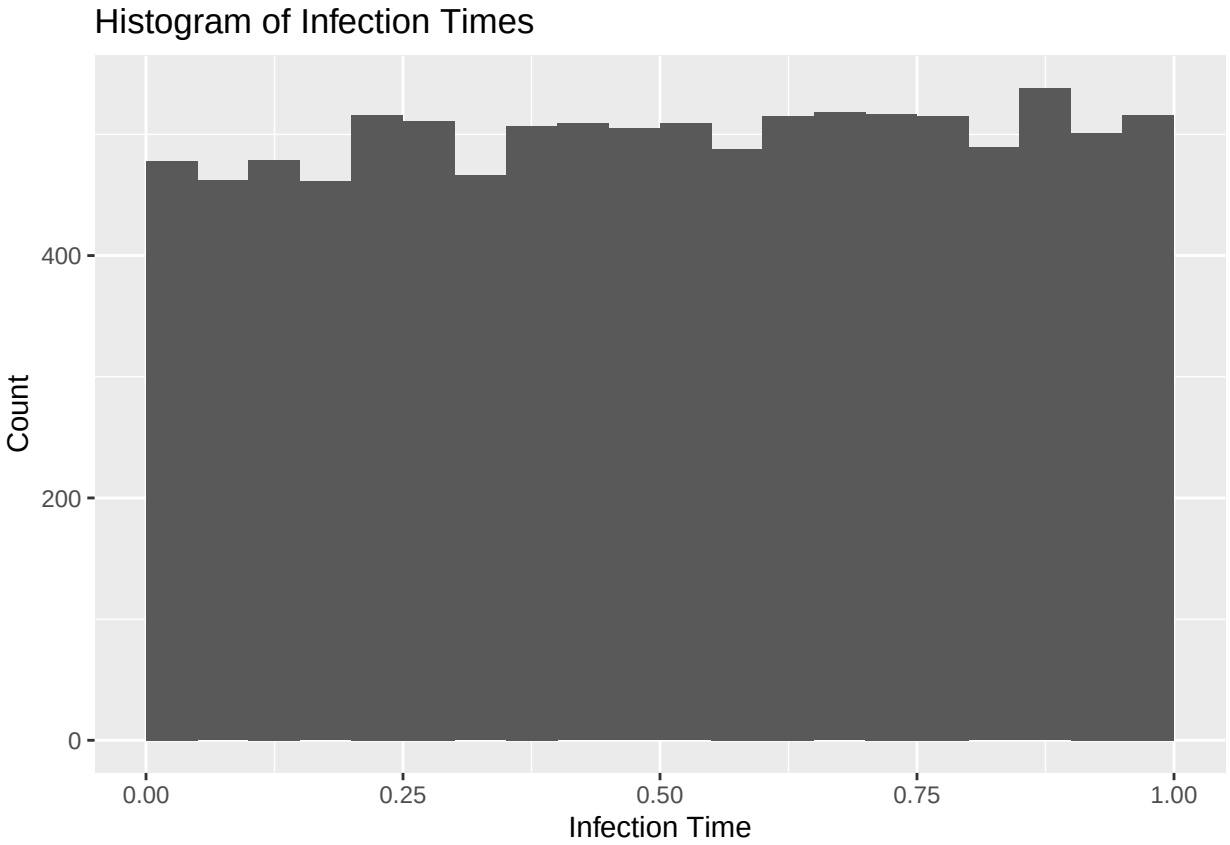


And finally, $\lambda_1 = \lambda_2$.

```
set.seed(100)
res_3 <- mh_sampler(
  N = 10000,
  I0 = 0.5,
  lambda1 = 0.5,
  lambda2 = 0.5,
  big_t = 1,
  delta = 0.2
)
```

```
## [1] "Acceptance rate: 100%"
```

```
ggplot() +
  aes(x = res_3) +
  geom_histogram(binwidth = 0.05, boundary = 0) +
  xlab("Infection Time") +
  ylab("Count") +
  ggtitle("Histogram of Infection Times")
```



We see that when the λ's are the same, the distribution is basically uniform. However, making λ_1 much larger than λ_2 make the distribution skewed right, whereas flipping the relationship makes the distribution skewed left.

Intuitively, this makes sense. If $lambda_1$ is much smaller than λ_2, there is high probability that the person recovered from the infection before time 1 if they got it early on in the period, thus the density is more closely clustered around time 1. For the opposite relationship, the inverse is true. And when the two λ's are equal, there is uniform probability for any infection time in the interval.