Lab 5: Rejection Sampling - Supplementary Notes

Aihua Li

Jan 31, 2022

Agenda

We can often end up with posterior distributions that we only know up to a normalizing constant. For example, in practice, we may derive

$$p(\theta \mid x) \propto p(x \mid \theta)p(\theta)$$

and find that the normalizing constant $p(x)$ is very difficult to evaluate. Such examples occur when we start building non-conjugate models in Bayesian statistics.

Given such a posterior, how can we appropriate it's density? One way is using rejection sampling. As an example, let's suppose our resulting posterior distribution is

$$f(x) \propto sin^2(\pi x), x \in [0, 1].$$

In order to understand how to approximate the density (normalized) of f, we will investigate the following tasks:

1. Plot the densities of $f(x)$ and the Unif(0,1) on the same plot. According to the rejection sampling approach sample from $f(x)$ using the Unif(0,1) pdf as an enveloping function.
2. Plot a histogram of the points that fall in the acceptance region. Do this for a simulation size of 10^2 and 10^5 and report your acceptance ratio. Compare the ratios and histograms.
3. Repeat Tasks 1 - 3 for Beta(2,2) as an enveloping function.
4. Provide the four histograms from Tasks 2 and 3 using the Uniform(0,1) and the Beta(2,2) enveloping proposals. Provide the acceptance ratios. Provide commentary.
5. (i) Do you recommend the Uniform or the Beta(2,2) as a better enveloping function (or are they about the same)? (ii) If you were to try and find an enveloping function that had a high acceptance ratio, which one would you try and why? For part (ii), either back up your choice using a small paragraph or (ii) empirically illustrate that it's better. (If you wanted to go above and beyond, you could do both, which is highly encouraged and would really prepare you for the exam!)

A quick review of rejection sampling

Goal: generate samples from a complicated pdf $f(x)$.

Suppose

$$f(x) = \frac{\tilde{f}(x)}{\alpha},$$

where $\alpha > 0$ is the normalizing constant, and we may not be able to derive its closed-form. Suppose we can evaluate $\tilde{f}(x)$.

Steps:

1. Choose a proposal distribution q and a ceiling constant $c > 0$, such that $cq(x)$ fully envelops $f(x)$, i.e.

$$cq(x) \geq \tilde{f}(x).$$

2. Sample $X \sim q$. Given the value of X we've sampled, sample $Y \sim Unif(0, cq(X))$.
3. If $Y \leq \tilde{f}(X)$, let $Z = X$. If $Y > \tilde{f}(X)$, reject this X and return to step 2.

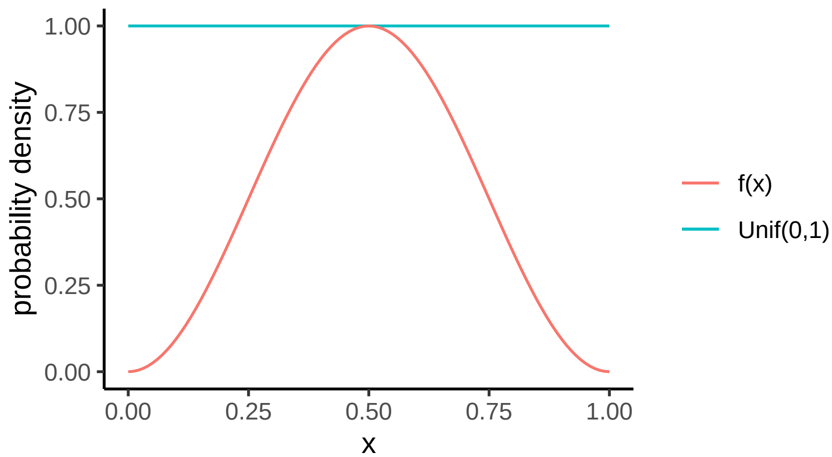
Finally, Z is the desired sample from f. Repeat this procedure until you get the desired number of samples.

Task 1

Plot the densities of $f(x)$ and the $Unif(0,1)$ on the same plot. According to the rejection sampling approach, sample from $f(x)$ using the $Unif(0,1)$ pdf as an enveloping function.

In the first place, compare the densities of the target function and the proposal distribution, and decide on the ceiling constant c.

```
x <- seq(0, 1, 0.01)  # grid on [0, 1]
f_unif <- dunif(x, 0, 1)  # density of unif(0, 1)
f_sin <- function(x) { sin(pi*x)^2 }  # function f(x)=sin(pi*x)^2
ggplot() +
  geom_line(aes(x, f_unif, color = "Unif(0,1)")) +
  geom_line(aes(x, f_sin(x), color = "f(x)")) +
  theme_classic(base_size = 11) + labs(y = "probability density") +
  theme(legend.title = element_blank())
```



Then, write the codes to realize the rejection sampling.

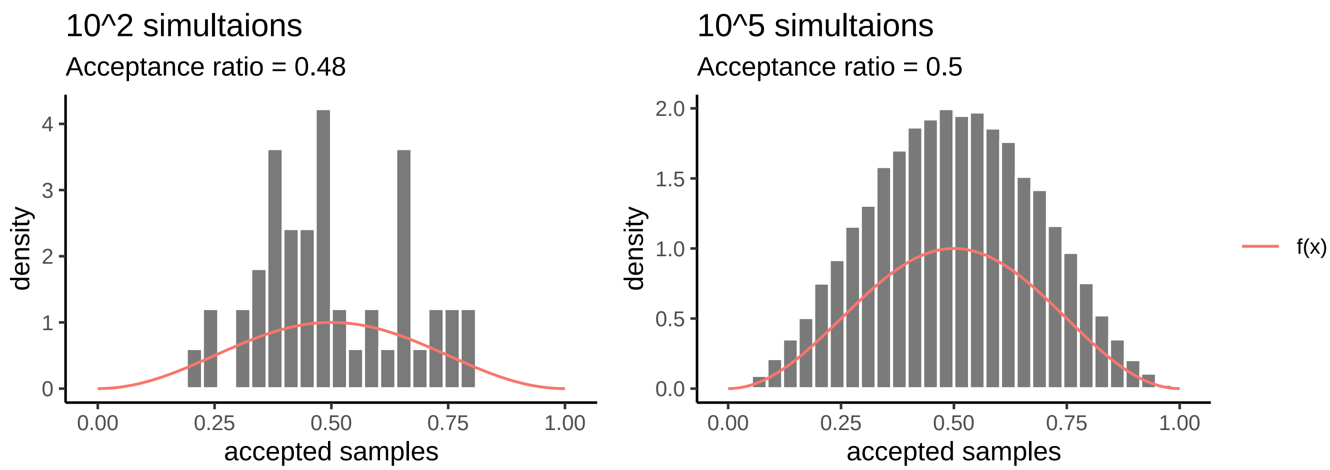
```
rejection_sampling <- function(S) {  # S: total number of simulations
  c <- 1  # constant c
  X <- runif(S, 0, 1)  # generate from the proposal distribution q(x)
  Y <- runif(S, 0, c*dunif(X, 0, 1))

  # get the index of the accepted samples (a vector of TRUE/FALSE and TRUE denotes acceptance)
  accept <- (Y <= f_sin(X))  # accept the sample if Y <= f_sin(X)

  # extract the accepted samples
  Z <- X[accept]

  # compute acceptance ratio
  accept_ratio <- mean(accept)

  return (list(sample = Z, accept_ratio = accept_ratio))
}

# According to task 2, do simulations for S=10^2 and S=10^5
res1 <- rejection_sampling(S = 10^2)
res2 <- rejection_sampling(S = 10^5)
```

Attention: In this example, both the proposal distribution q and the distribution of Y are uniform distribution. Be careful in coding!

Task 2

Plot a histogram of the points that fall in the acceptance region. Do this for a simulation size of 10^2 and 10^5 and report your acceptance ratio. Compare the ratios and histograms.

```
p1 <- ggplot() +
  geom_histogram(aes(res1[["sample"]], ..density..), color = "white", alpha = 0.8) +
  geom_line(aes(x, f_sin(x), color = "f(x)")) + theme_classic(base_size = 10) +
  labs(x = "accepted samples", y = "density", title = "10^2 simultaions",
       subtitle = str_c("Acceptance ratio = ", res1[["accept_ratio"]] %>% round(2))) +
  theme(legend.title = element_blank())
p2 <- ggplot() +
  geom_histogram(aes(res2[["sample"]], ..density..), color = "white", alpha = 0.8) +
  geom_line(aes(x, f_sin(x), color = "f(x)")) + theme_classic(base_size = 10) +
  labs(x = "accepted samples", y = "density", title = "10^5 simultaions",
       subtitle = str_c("Acceptance ratio = ", res2[["accept_ratio"]] %>% round(2))) +
  theme(legend.title = element_blank())
ggarrange(p1, p2, common.legend = T, legend = "right")
```



Observations:

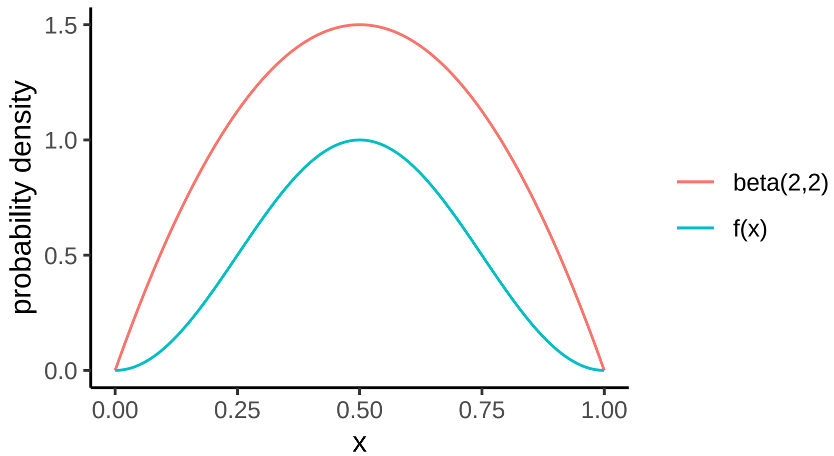
- The acceptance ratio are basically the same for the two cases.
- A larger number of simulations provide more accepted samples, leading to a more desirable histogram.
- The histogram of the accepted samples is different from the red curve of $f(x) = sin^2(\pi x)$. Why? Because $f(x)$ is the unnormalized density as we don't know the normalizing constant, while the histogram represents the normalized density! Indeed, the shape of the histogram and the shape of $f(x)$ are different in a constant.

Task 3

Repeat task 1-3 for $Beta(2, 2)$ as an enveloping function.

First, compare the densities to make sure that $cq(x)$ fully envelops the target function.

```
x <- seq(0, 1, 0.01)  # grid on [0, 1]
f_beta <- dbeta(x, 2, 2)  # density of unif(0, 1)
f_sin <- function(x) { sin(pi*x)^2 }  # function f(x)=sin(pi*x)^2
ggplot() +
  geom_line(aes(x, f_beta, color = "beta(2,2)")) +
  geom_line(aes(x, f_sin(x), color = "f(x)")) +
  theme_classic(base_size = 11) + labs(y = "probability density") +
  theme(legend.title = element_blank())
```

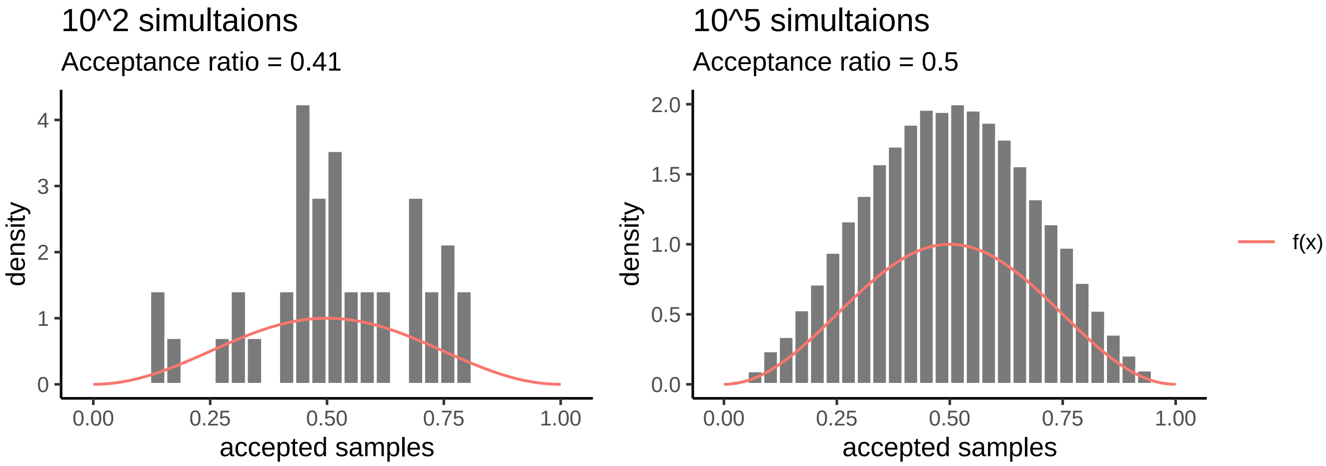


Then, code the rejection sampling and make the required histograms.

```
rejection_sampling <- function(S) {  # S: total number of simulations
  c <- 1  # constant c
  X <- rbeta(S, 2, 2)  # generate from the proposal distribution q(x)
  Y <- runif(S, 0, c*dbeta(X, 2, 2))

  # get the index of the accepted samples (a vector of TRUE/FALSE and TRUE denotes acceptance)
  accept <- (Y <= f_sin(X))  # accept the sample if Y <= f_sin(X)

  # extract the accepted samples
  Z <- X[accept]

  # compute acceptance ratio
  accept_ratio <- mean(accept)

  return (list(sample = Z, accept_ratio = accept_ratio))
}

# According to task 2, do simulations for S=10^2 and S=10^5
res1 <- rejection_sampling(S = 10^2)
res2 <- rejection_sampling(S = 10^5)
```

```
p1 <- ggplot() +
  geom_histogram(aes(res1[["sample"]], ..density..), color = "white", alpha = 0.8) +
  geom_line(aes(x, f_sin(x), color = "f(x)")) + theme_classic(base_size = 10) +
  labs(x = "accepted samples", y = "density", title = "10^2 simultaions",
       subtitle = str_c("Acceptance ratio = ", res1[["accept_ratio"]] %>% round(2))) +
  theme(legend.title = element_blank())
p2 <- ggplot() +
```

```
  geom_histogram(aes(res2[["sample"]], ..density..), color = "white", alpha = 0.8) +
  geom_line(aes(x, f_sin(x), color = "f(x)")) + theme_classic(base_size = 10) +
  labs(x = "accepted samples", y = "density", title = "10^5 simultaions",
       subtitle = str_c("Acceptance ratio = ", res2[["accept_ratio"]] %>% round(2))) +
  theme(legend.title = element_blank())
ggarrange(p1, p2, common.legend = T, legend = "right")
```



Observation: The acceptance ratio is similar to the uniform case. Also, in the larger simulations, there are more accepted samples and thus the histogram resembles the shape of the target function.

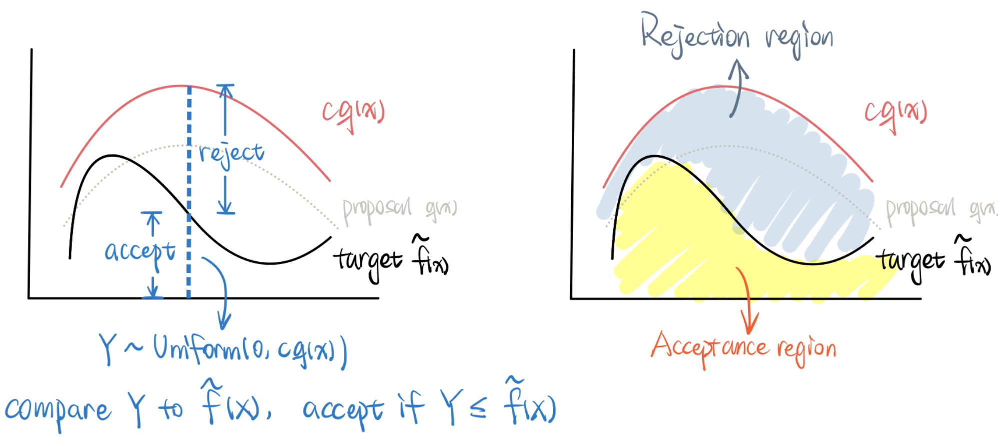
Task 4 (to be completed in homework)

Provide the four histograms from Tasks 2 and 3 using the Uniform(0,1) and the Beta(2,2) enveloping proposals. Provide the acceptance ratios. Provide commentary.

Task 5 (to be completed in homework)

(i) Do you recommend the Uniform or the Beta(2,2) as a better enveloping function (or are they about the same)? (ii) If you were to try and find an enveloping function that had a high acceptance ratio, which one would you try and why? For part (ii), either back up your choice using a small paragraph or (ii) empirically illustrate that it's better. (If you wanted to go above and beyond, you could do both, which is highly encouraged and would really prepare you for the exam!)

Intuition for rejection sampling:



The closer the proposal distribution is to the target distribution, the smaller the c, the higher the acceptance ratio and thus the more efficient our sampler is.

To have a higher acceptance ratio and a more efficient rejection sampling, we want the rejection region to be as small as possible. To achieve this:

- Choose a proposal distribution q, such that its shape is quite similar to the target function.
- Let the constant c be as small as possible.