

# Introduction to Markov Chain Monte Carlo

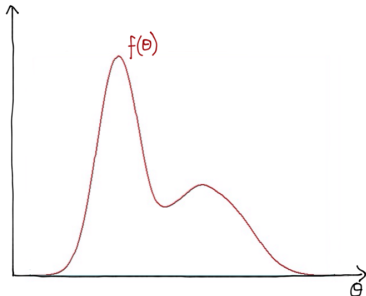
## Recap. on Bayesian inference

Last time we saw that the **posterior distribution** of  $\theta$ , given observed data is

$$p(\theta|\text{data}) \propto p(\text{data}|\theta)p(\theta)$$

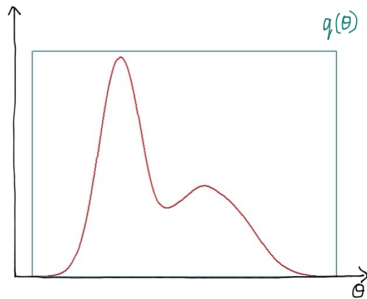
Our aim is to draw samples from this distribution.

# Rejection sampling



- Consider a distribution  $f(\theta)$ , which we can evaluate for any  $\theta$
- How do we draw samples?

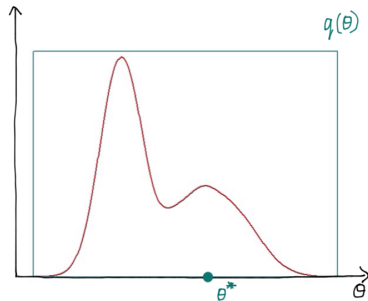
# Rejection sampling



Rejection sampling uses a **proposal distribution  $q(\theta)$**  which:

- is simple to evaluate
- is easy to sample from
- has a greater density than  $f(\theta)$  for all  $\theta$

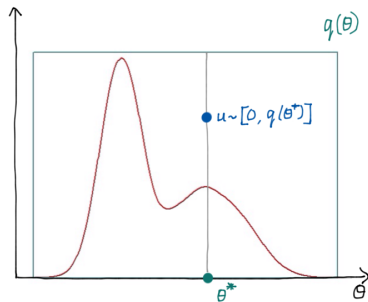
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$

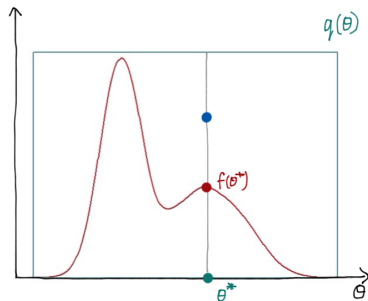
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$

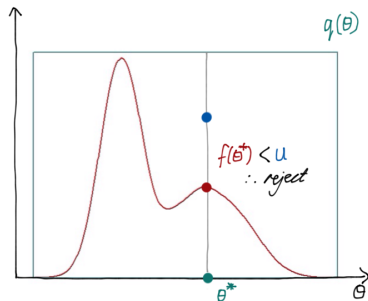
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$

# Rejection sampling

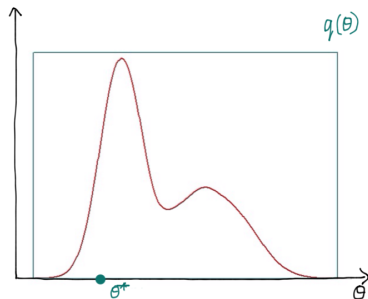


The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject



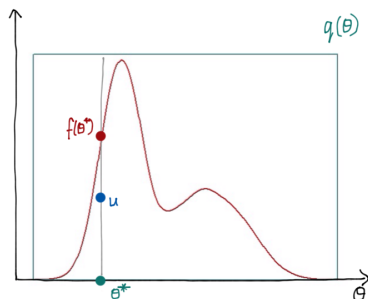
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

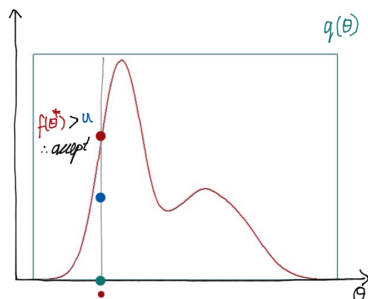
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

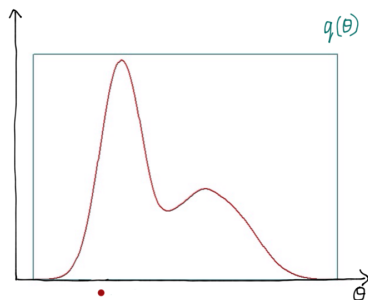
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

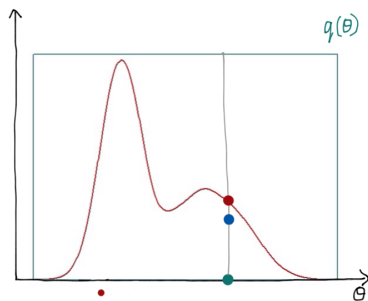
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

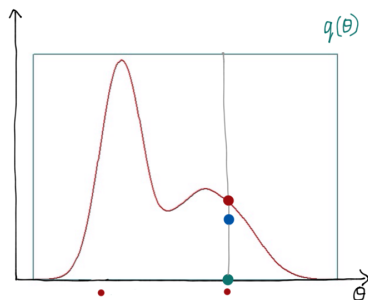
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim Uniform[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

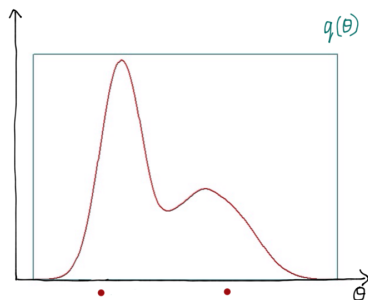
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

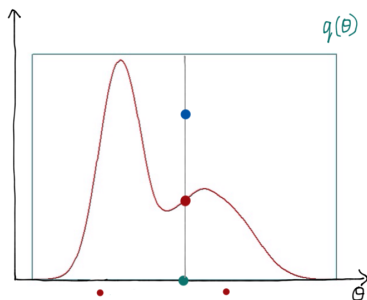
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

# Rejection sampling

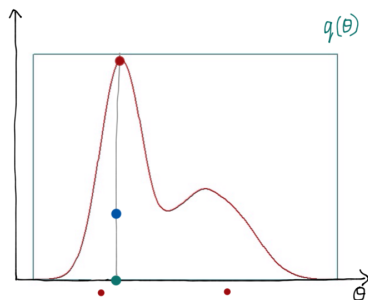


The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4



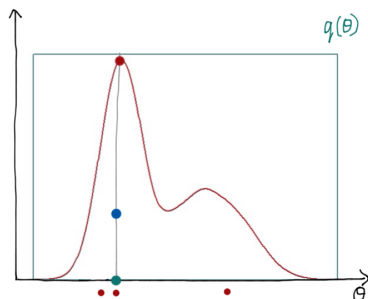
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

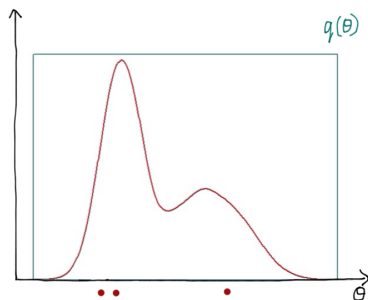
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

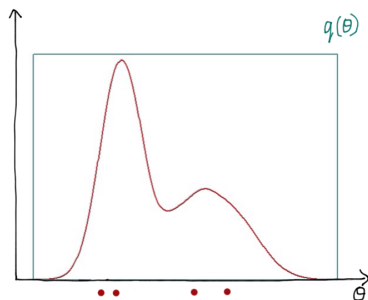
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

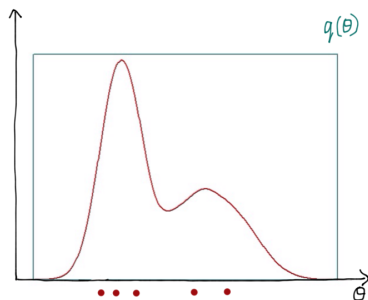
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

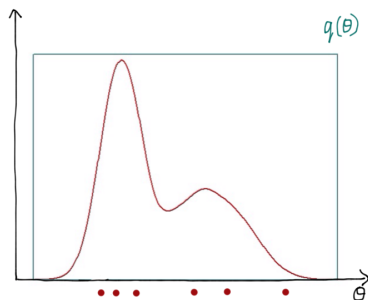
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

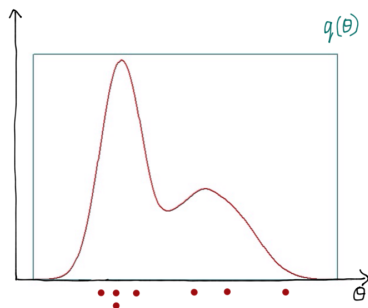
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

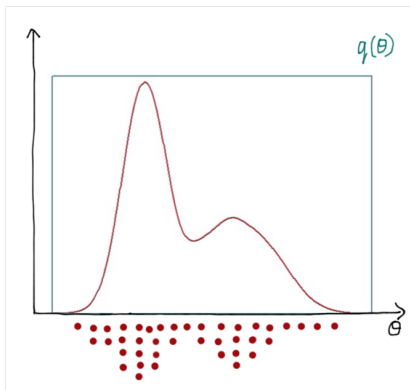
# Rejection sampling



The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4

# Rejection sampling



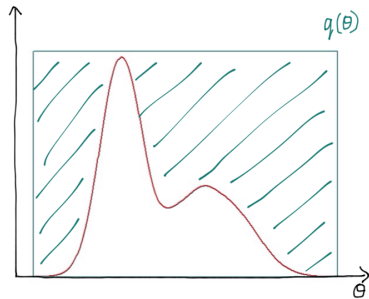
The algorithm proceeds as follows:

1. Sample  $\theta^*$  from  $q(\theta)$
2. Draw  $u \sim \text{Uniform}[0, q(\theta^*)]$
3. Evaluate  $f(\theta^*)$
4. If  $f(\theta^*) > u$  accept, else reject
5. Repeat steps 1-4



# Rejection sampling

- Rejection sampling works best if  $q(\theta) \approx f(\theta)$
- Requiring  $q(\theta) \geq f(\theta)$  for all  $\theta$  can make rejection rate v. high
- Even more limited in high dimensions



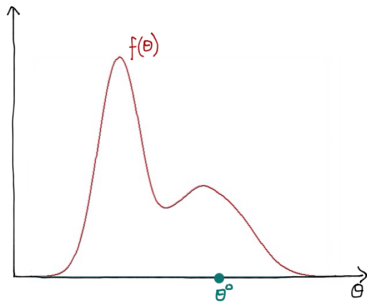
# Markov Chain Monte Carlo

- In Markov Chain Monte Carlo (MCMC) we do not define one proposal density  $q(\theta) \geq f(\theta)$ .
- Rather we build up a **chain** of samples where each proposed  $\theta^*$  depends on the previous one

i.e the proposal density takes the form  $q(\theta^*|\theta)$

- A commonly used MCMC algorithm is **Metropolis-Hastings** (M-H).
- The acceptance rate of M-H is carefully derived to ensure **unbiased samples**.

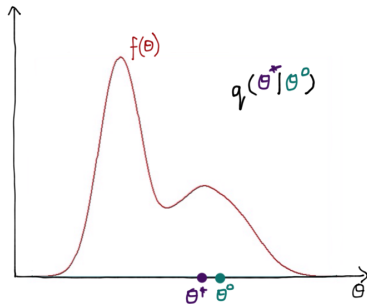
# Metropolis-Hastings



The algorithm proceeds as follows:

1. Initialise  $\theta^0$ , set  $\theta = \theta^0$

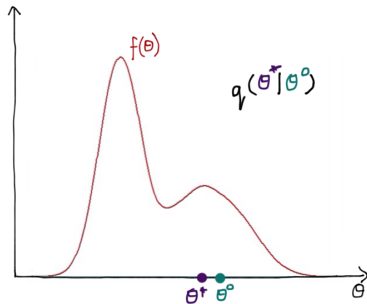
# Metropolis-Hastings



The algorithm proceeds as follows:

1. Initialise  $\theta^0$ , set  $\theta = \theta^0$
2. Sample  $\theta^* \sim q(\theta^* | \theta)$

# Metropolis-Hastings



The algorithm proceeds as follows:

1. Initialise  $\theta^0$ , set  $\theta = \theta^0$
2. Sample  $\theta^* \sim q(\theta^*|\theta)$
3. Compute acceptance probability,  $r$

# Metropolis-Hastings

## Acceptance

- If  $q(\theta^*|\theta)$  symmetric, then

$$r = \min \left( 1, \frac{f(\theta^*)}{f(\theta)} \right)$$

The algorithm proceeds as follows:

1. Initialise  $\theta^0$ , set  $\theta = \theta^0$
2. Sample  $\theta^* \sim q(\theta^*|\theta)$
3. Compute acceptance probability,  $r$

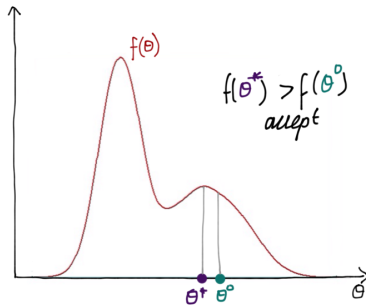
# Metropolis-Hastings

## Acceptance

- If  $q(\theta^*|\theta)$  symmetric, then

$$r = \min \left( 1, \frac{f(\theta^*)}{f(\theta)} \right)$$

- Definitely move to  $\theta^*$  if more probable than  $\theta$



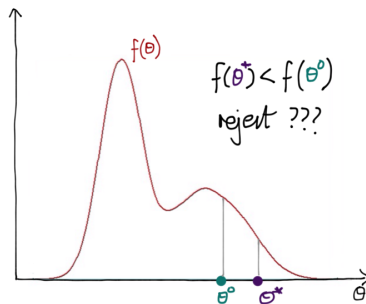
# Metropolis-Hastings

## Acceptance

- If  $q(\theta^*|\theta)$  symmetric, then

$$r = \min \left( 1, \frac{f(\theta^*)}{f(\theta)} \right)$$

- Definitely move to  $\theta^*$  if more probable than  $\theta$
- May move if  $\theta^*$  less probable





# Metropolis-Hastings

## Acceptance

- If  $q(\theta^*|\theta)$  symmetric, then

$$r = \min \left( 1, \frac{f(\theta^*)}{f(\theta)} \right)$$

- Definitely move to  $\theta^*$  if more probable than  $\theta$
- May move if  $\theta^*$  less probable

- If  $q(\theta^*|\theta)$  asymmetric, then

$$r = \min \left( 1, \frac{f(\theta^*)q(\theta|\theta^*)}{f(\theta)q(\theta^*|\theta)} \right)$$



# Metropolis-Hastings

## Acceptance

- If  $q(\theta^*|\theta)$  symmetric, then

$$r = \min \left( 1, \frac{f(\theta^*)}{f(\theta)} \right)$$

- Definitely move to  $\theta^*$  if more probable than  $\theta$
- May move if  $\theta^*$  less probable

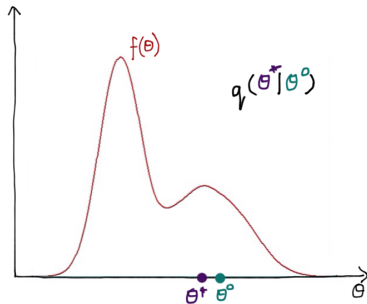
- If  $q(\theta^*|\theta)$  asymmetric, then

$$r = \min \left( 1, \frac{f(\theta^*)q(\theta|\theta^*)}{f(\theta)q(\theta^*|\theta)} \right)$$

The algorithm proceeds as follows:

1. Initialise  $\theta^0$ , set  $\theta = \theta^0$
2. Sample  $\theta^* \sim q(\theta^*|\theta)$
3. Compute acceptance probability,  $r$

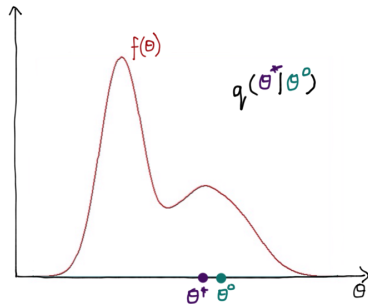
# Metropolis-Hastings



The algorithm proceeds as follows:

1. Initialise  $\theta^0$ , set  $\theta = \theta^0$
2. Sample  $\theta^* \sim q(\theta^*|\theta)$
3. Compute acceptance probability,  $r$

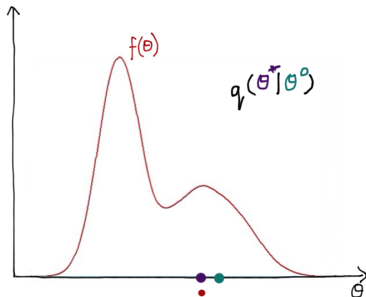
# Metropolis-Hastings



The algorithm proceeds as follows:

1. Initialise  $\theta^0$ , set  $\theta = \theta^0$
2. Sample  $\theta^* \sim q(\theta^* | \theta)$
3. Compute acceptance probability,  $r$
4. Draw  $u \sim \text{Uniform}[0, 1]$

# Metropolis-Hastings

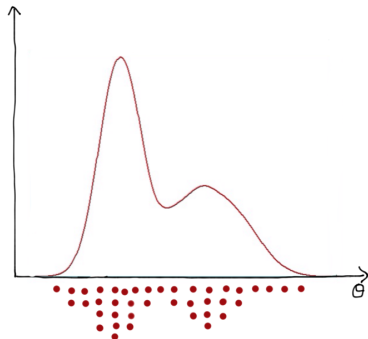


The algorithm proceeds as follows:

1. Initialise  $\theta^0$ , set  $\theta = \theta^0$
2. Sample  $\theta^* \sim q(\theta^*|\theta)$
3. Compute acceptance probability,  $r$
4. Draw  $u \sim \text{Uniform}[0, 1]$
5. Set new sample to

$$\theta^{(s+1)} = \begin{cases} \theta^*, & \text{if } u < r \\ \theta^{(s)}, & \text{if } u \geq r \end{cases}$$

# Metropolis-Hastings



The algorithm proceeds as follows:

1. Initialise  $\theta^0$ , set  $\theta = \theta^0$
2. Sample  $\theta^* \sim q(\theta^*|\theta)$
3. Compute acceptance probability,  $r$
4. Draw  $u \sim \text{Uniform}[0, 1]$
5. Set new sample to

$$\theta^{(s+1)} = \begin{cases} \theta^*, & \text{if } u < r \\ \theta^{(s)}, & \text{if } u \geq r \end{cases}$$

6. Repeat steps 2-5