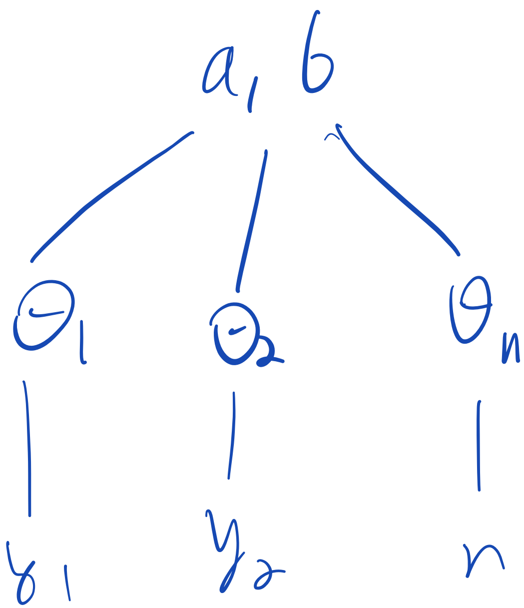
Lecture 5: Monte Carlo Methods

Professor Alexander Franks

Announcements

- Reading: Chapter 10 and 11
- HW 3 posted.



$y_i \sim Bin(n_i, \theta_i)$

$\theta_i \sim Beta(a, b)$

Want: $P(a, b \mid y_1, \dots, y_n) =$

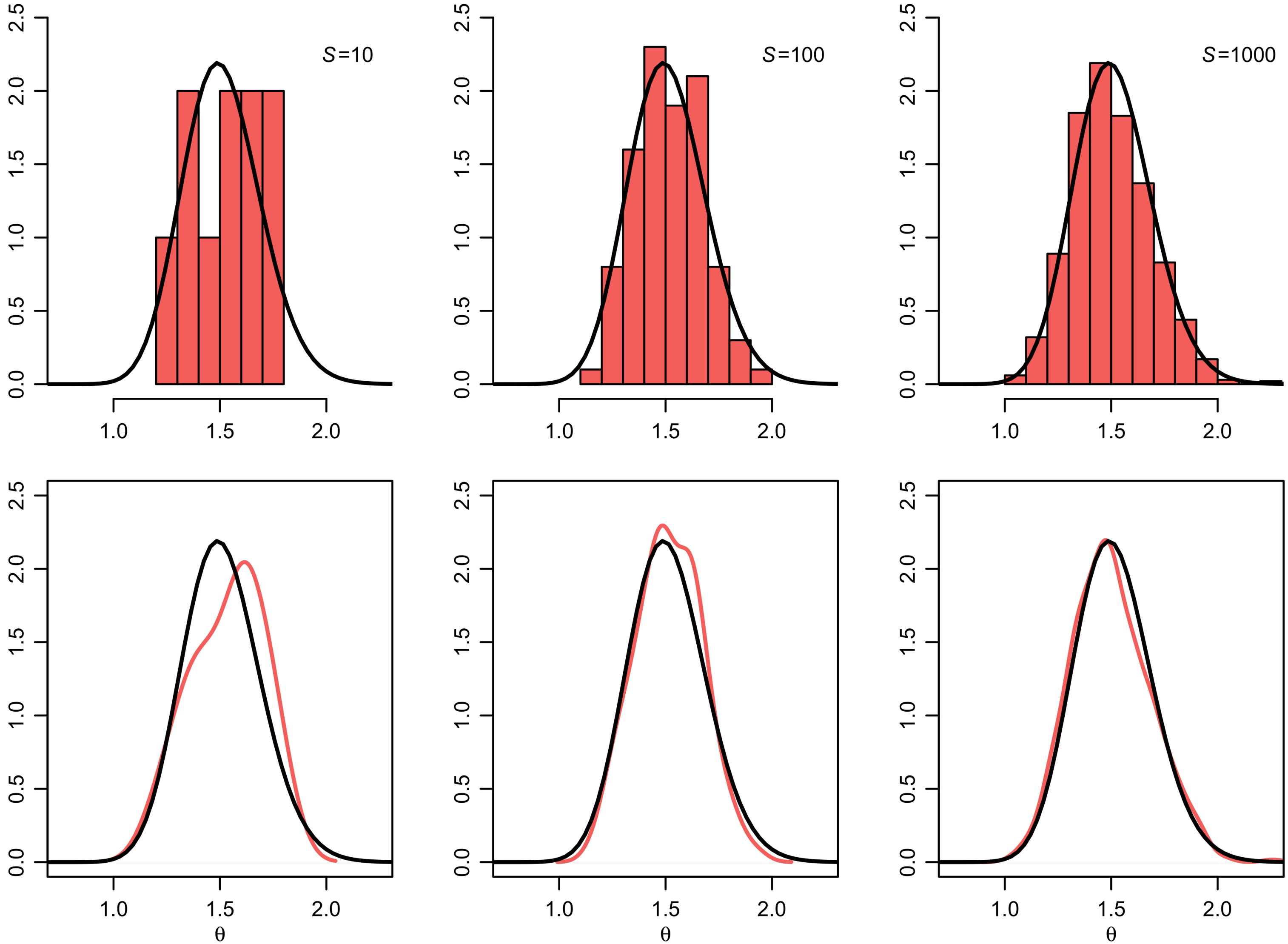
$$\iiint \underbrace{P(a, b, \theta_1, \dots, \theta_n \mid y_1, \dots, y_n)}_{\propto} d\theta$$

$$P(y_1, \dots, y_n \mid \theta_1, \dots, \theta_n, a, b)\, P(a, b, \theta's)$$

$$\left(\prod_{i=1}^{50} P(y_i \mid \theta_i)\right) P(\theta_1, \dots, \theta_{50} \mid a, b)\, P(a, b)$$

Monte Carlo approximations of a distribution

$\frac{1}{S}$



Sampling strategies

- Inversion Sampling (works for univariate distns)
- Grid sampling (works for low dimensional problems)
- Rejection sampling
- Importance sampling
- Markov Chain Monte Carlo

Sampling strategies

Why sampling?

Probability Integral Transform

- Suppose that a random variable, Y has a continuous distribution for with CDF is F_Y.
- Then the random variable $U = F_Y(Y)$ has a uniform distribution
  - This is known as the “probability integral transform PIT”
- By taking the inverse of F_Y we have $F^{-1}(U) = Y$

Inversion Sampling

The inverse transform sampling method works as follows:

1. Generate a random number u from $\mathrm{Unif}[0, 1]$
2. Find the inverse of the desired CDF, e.g. $F_Y^{-1}(u)$.
3. Compute $y = F_Y^{-1}(u)$. y is now a sample from the desired distribution.

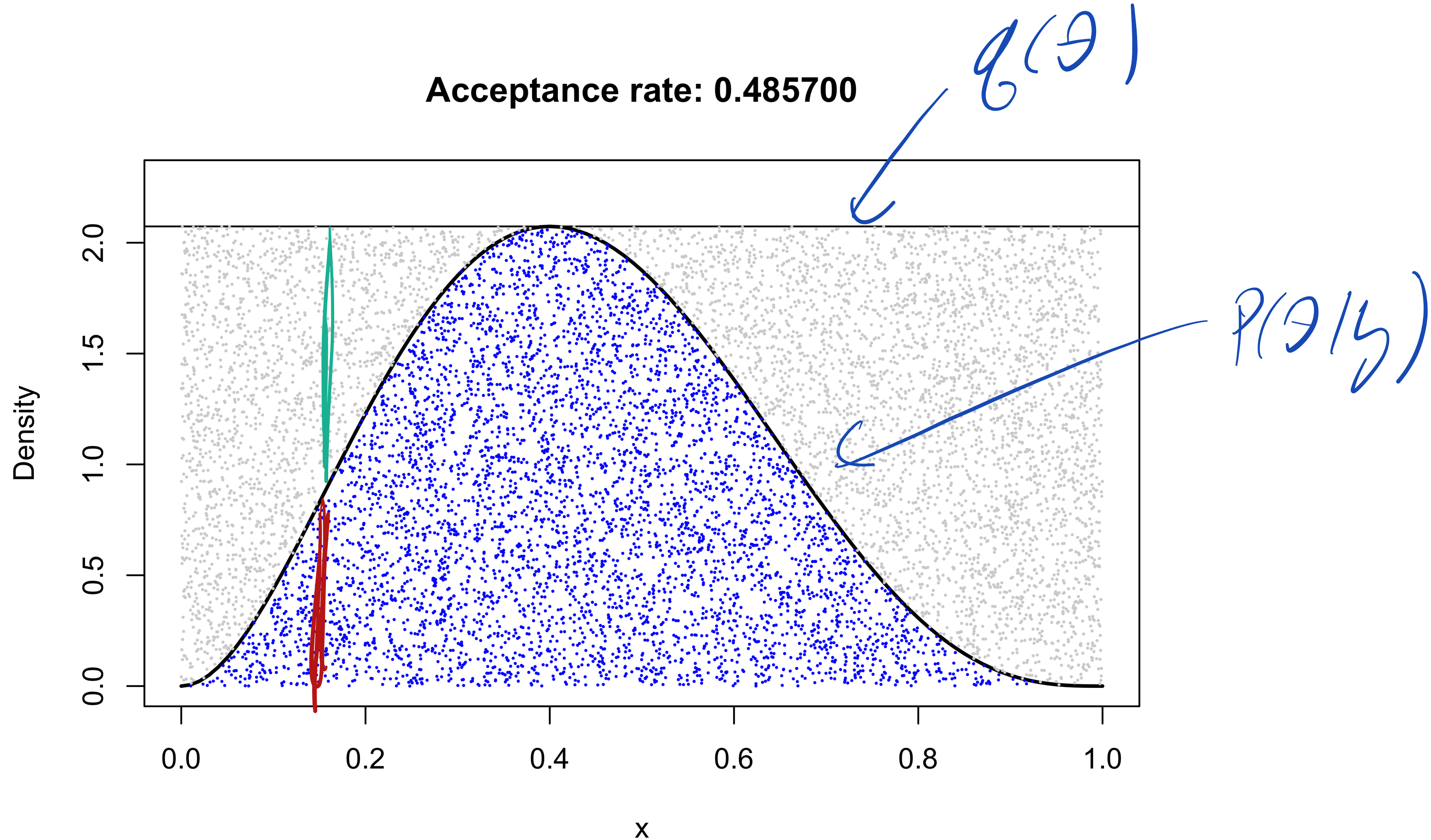
Inversion Sampling

Animation

Inversion Sampling

- Inversion sampling can be a fast and simple way to sample from a distribution
- Only effective if we know the inverse-CDF and can easily compute it
- This is a big challenge in practice. For example, even the normal distribution has a CDF, Φ, which cannot be expressed analytically.
  - Shifts from one hard problem (sampling) to another (computing an integral)
  - Need alternatives!

Rejection Sampling



Rejection Sampling algorithm

1. Choose a proposal density, $q(\theta)$ that we can easily sample from (e.g. uniform or normal) such that:

2.Find $M = \max \frac{p(\theta|y)}{q(\theta)}$

- If $M = \infty$ then q cannot be used as a proposal distribution
- If M is finite, $Mq(\theta)$ “envelopes” $p(\theta \; y)$

3. Draw a sample, $\theta^{(s)}$ from $q(\theta)$

4. Accept $\theta^{(s)}$ as a draw from $p(\theta \mid y)$ with probability $\frac{p(\theta^{(s)}|y)}{Mq(\theta^{(s)})}$

Rejection Sampling

Demo

Rejection Sampling

- What is the expected number of rejections?
- How can we show this algorithm is valid?

Z be a random indicator for accepting θ^* proposed from q targeting p.

$$P(Z=1) = E_q[Z] = E[E[Z \mid \theta^*]]$$

$$= E_q\left[\frac{p(\theta^* \mid y)}{M q(\theta^*)}\right] = \int \frac{p(\theta^* \mid y)}{M \cancel{q(\theta^*)}} \cancel{q(\theta^*)} \, d\theta^*$$

$$= 1/M$$

$$\underbrace{\tilde{p}(\theta \mid y) = L(\theta) p(\theta)}$$

$\int \tilde{p}(\theta \mid y) \, d\theta$

$$\tilde{M} = \max \frac{\tilde{p}}{q} \qquad E_q\left[\frac{\tilde{p}}{\tilde{M} q}\right] = \frac{K}{\tilde{M}} = \frac{1}{M}$$

$\theta^* \sim q$, and consider (Z, θ^*)

$$Z \mid \theta^* \sim \text{Bern}\left(\frac{\tilde{p}(\theta \mid y)}{\tilde{M} q(\theta)}\right)$$

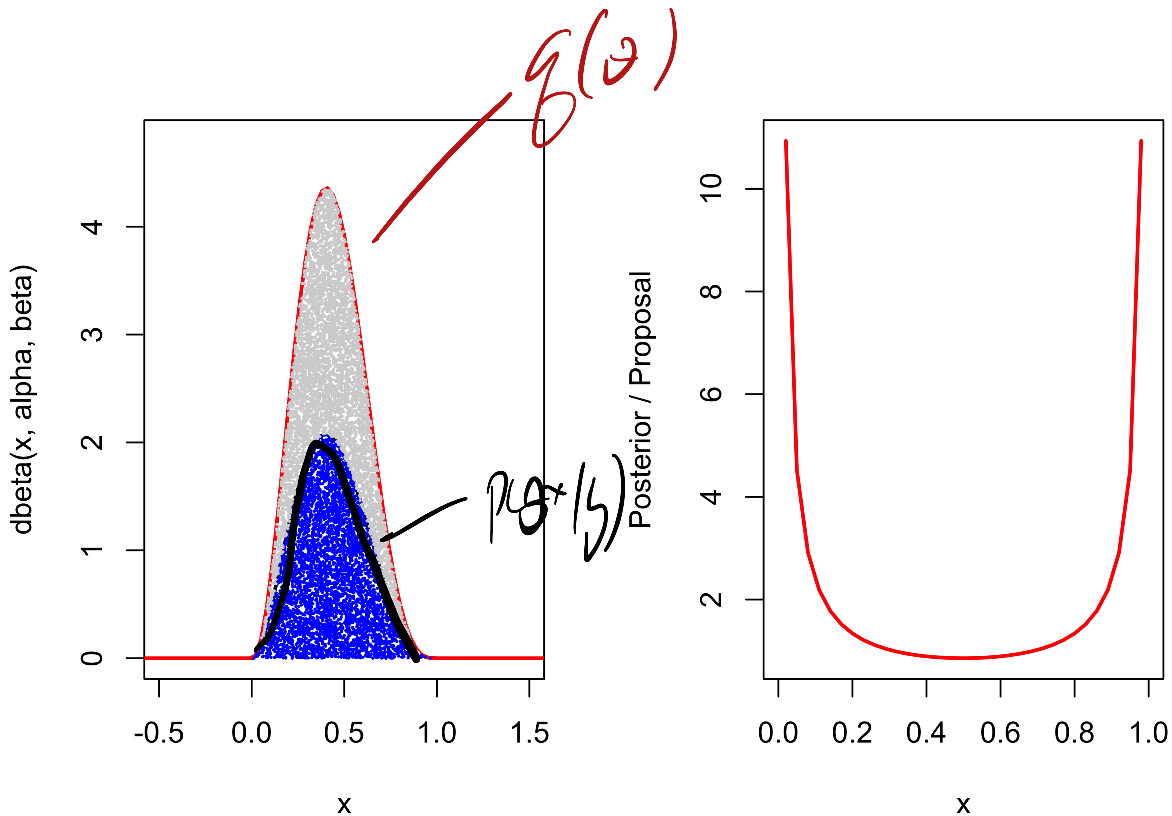
$$P(z,\theta) = q(\theta)\left(\frac{\tilde{p}(\theta|y)}{\tilde{m}\,q(\theta)}\right)^{z}\left(1-\frac{\tilde{p}(\theta|y)}{\tilde{m}\,q(\theta)}\right)^{1-z}$$

$$P(\theta|z=1) = \frac{P(z=1,\theta)}{P(z=1)} = \frac{\cancel{q(\theta)}\,\dfrac{\tilde{p}(\theta|y)}{\cancel{\tilde{m}}\,\cancel{q(\theta)}}}{K/\cancel{\tilde{m}}}$$

$$= \frac{\tilde{p}(\theta|y)}{K} = p(\theta|y)$$

Proposal most envelope target

Sometimes its not obvious…



Target density in black, proposal density in red

The German Tank Problem

The German Tank Problem

- US and allies wanted to estimate how many tanks were produced by the Germans during WWII
- They used the serial numbers on captured or destroyed tanks to estimate tank production
- Assuming serial numbers, Z_i, are ordered from 1 to n, where n is the number of tanks produced
- How do we estimate n?

The German Tank Problem

- Assume we observe the serial number of only one tank, Z
- What is a plausible sampling model? How do we write the likelihood?
- What is a reasonable prior?

Observe y Discrete Unif$(1, 2, \ldots N)$

The German Tank Problem

- Assume Z is uniformly distributed between 1 and n
- Specify a Poisson prior distribution
  - Assume a priori a rate of 200 tanks / month
- Use Monte Carlo (rejection sampler) to summarize the results. What is a reasonable proposal?

MLE for n?

$$L(n) = \frac{1}{n} I[n > y]$$

$$\hat{n}_{MLE} = y$$

$$\hat{n} \quad = 2$$

$$n \sim \text{Pois}(\lambda)$$

$$P(n|y) \propto \frac{1}{n} I[n \geq y) \frac{e^{-\lambda}\lambda^n}{n!}$$

Propose from $q \sim \text{Pois}(\lambda)$

$$\tilde{M} = \max_n \frac{P(n|z)}{q(n)} = \max_n \frac{\frac{1}{n}\cancel{\frac{e^{-\lambda}\lambda^n}{n!}} I[\,]}{\cancel{\frac{e^{-\lambda}\lambda^n}{n!}}}$$

$$= \max_n \frac{1}{n} I[n \geq y)$$

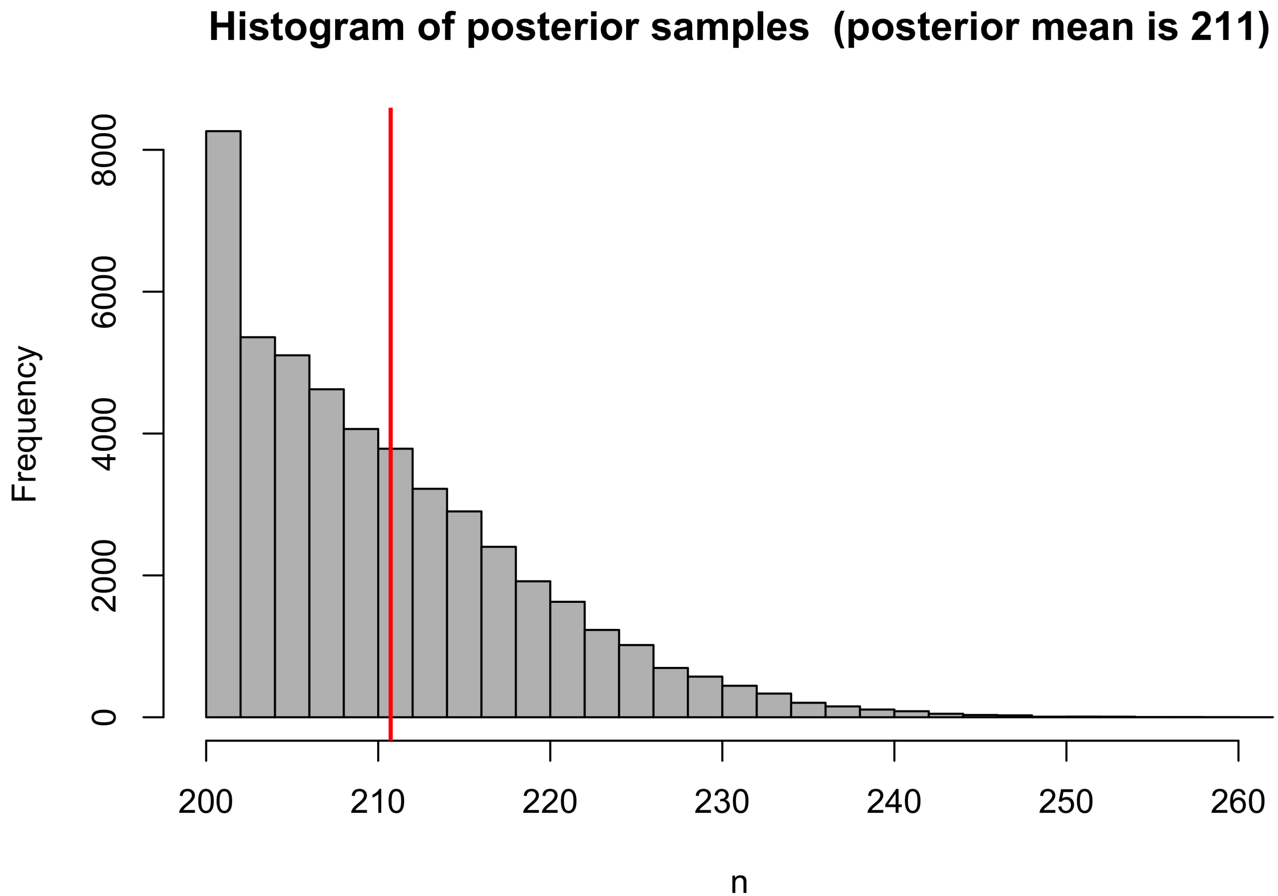
$$= \frac{1}{y}$$

$$\frac{\tilde{p}(n \mid y)}{\frac{1}{y}\, q(n)} = \frac{\frac{1}{n} e^{-\lambda} \cancel{\frac{\lambda^n}{n!}}\, I[\,]\; y}{e^{-\lambda} \cancel{\frac{\lambda^n}{n!}}}$$

$$= \frac{y}{n}\, I[n \geq y]$$

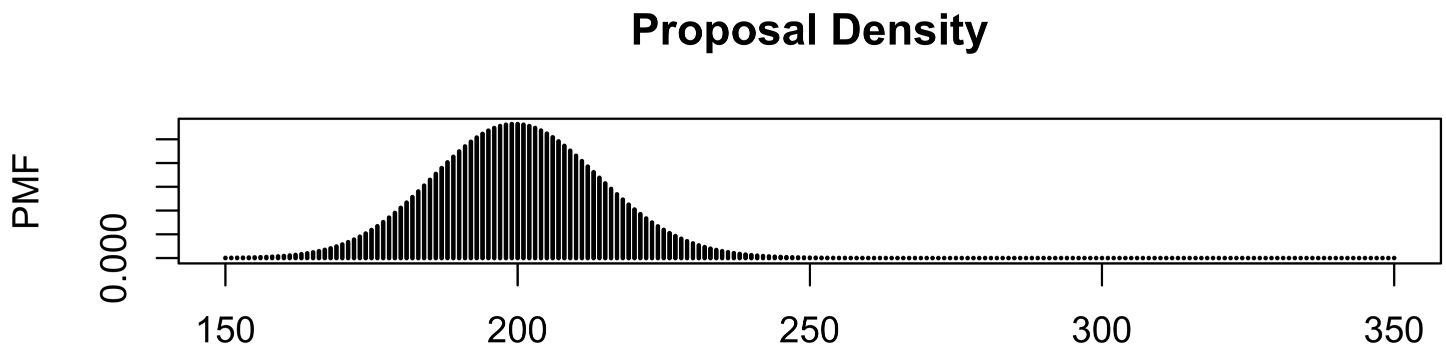
mom $\quad y \quad n$

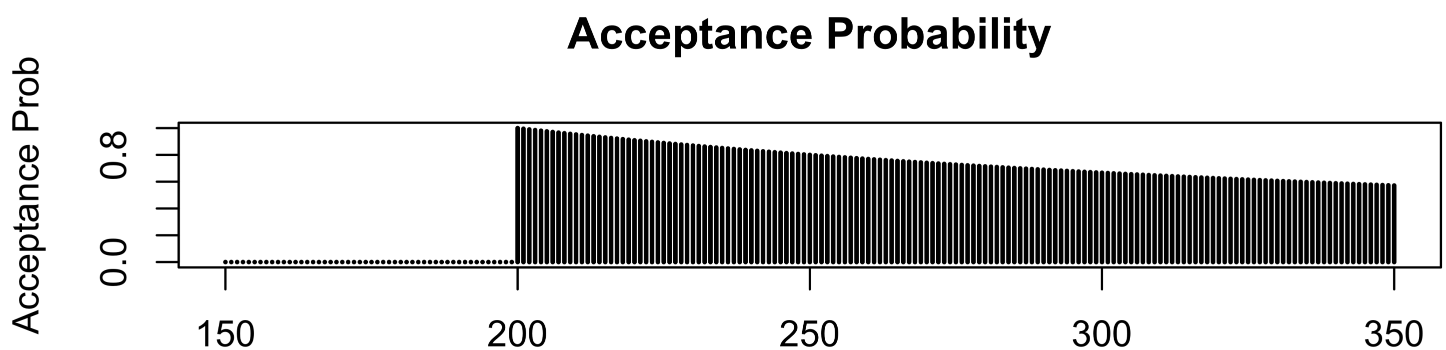
The German Tank Problem, $\cancel{Z}\,y = 200$



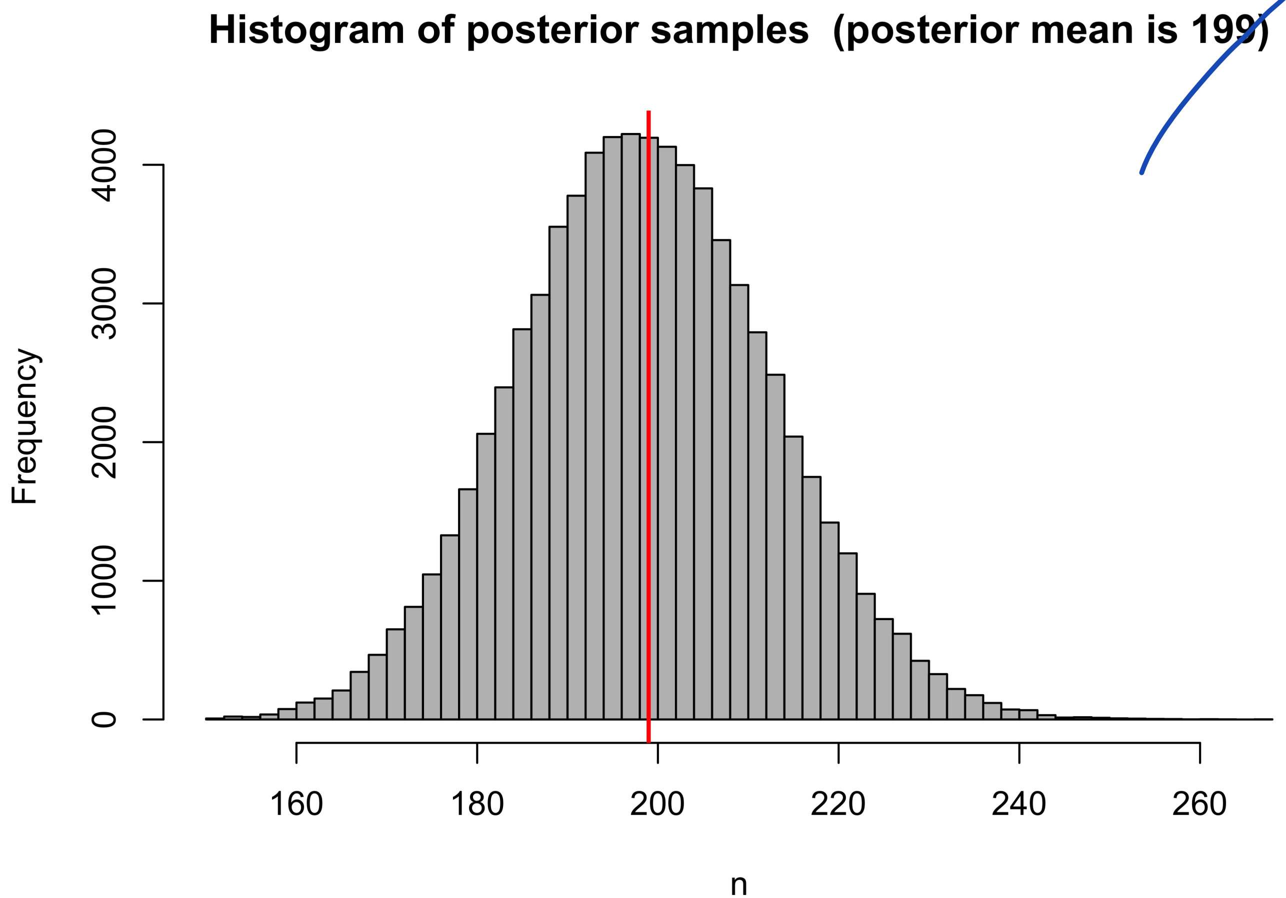
Acceptance rate: 0.48257

The German Tank Problem, Z = 200



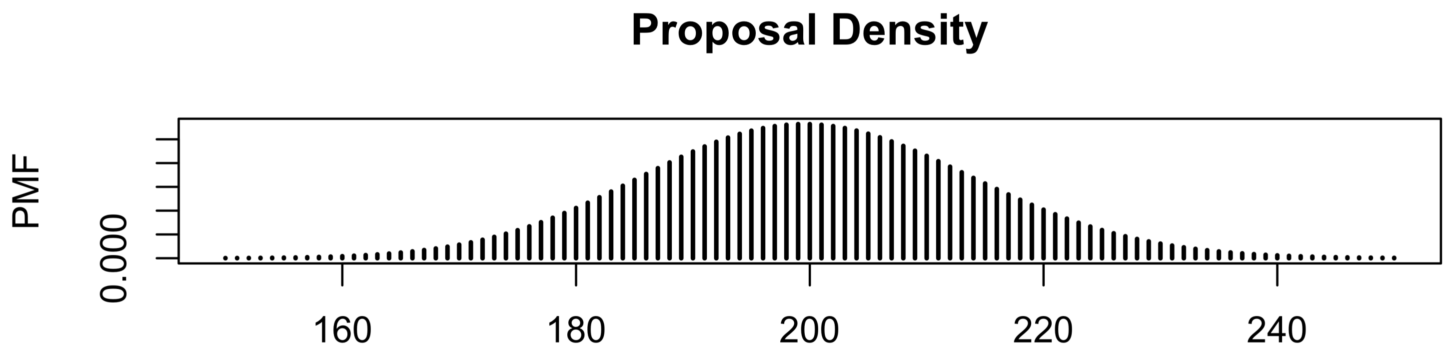


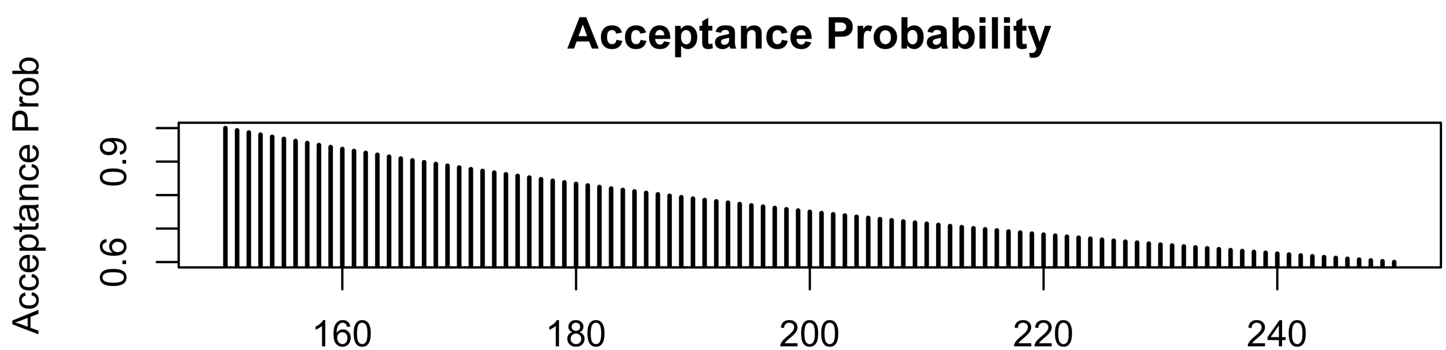
The German Tank Problem, Z = 150



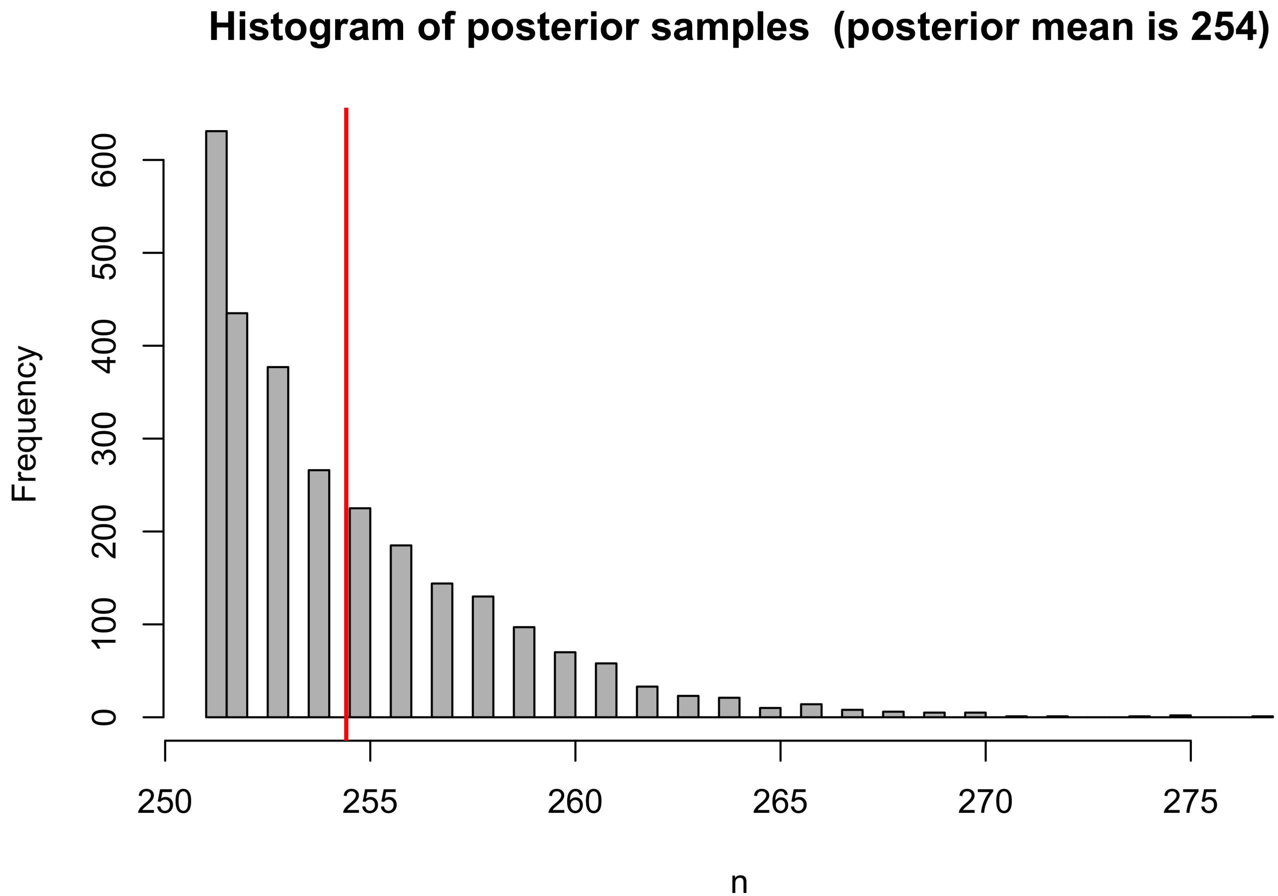
Acceptance rate: 0.75292

German Tank Problem, Z=150



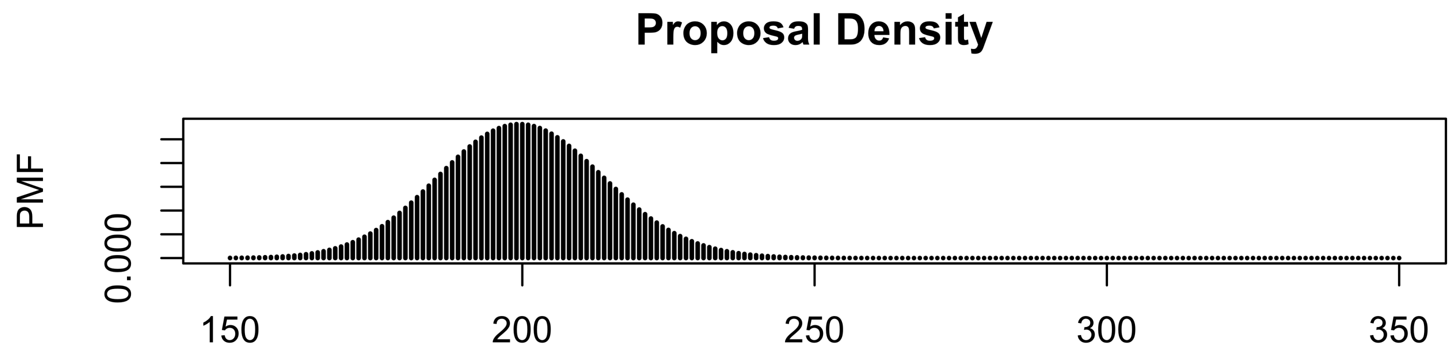


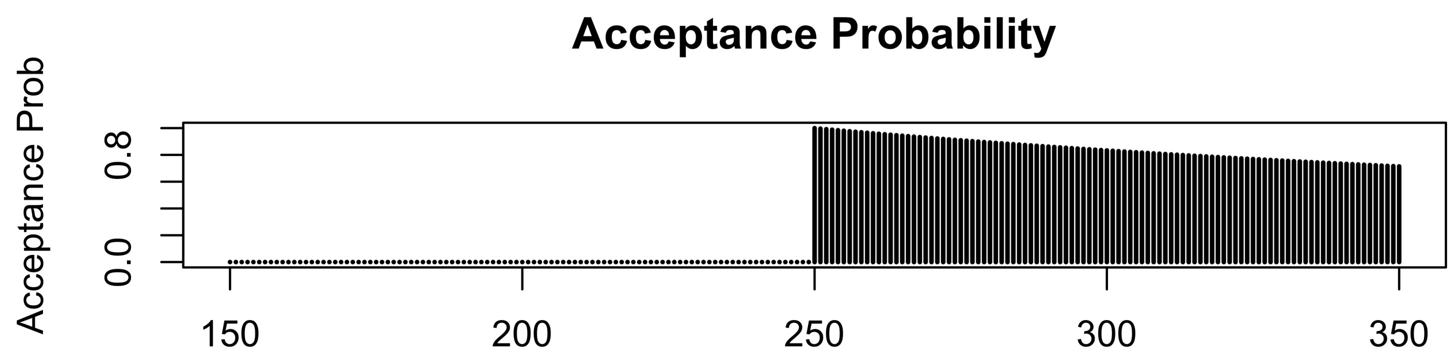
The German Tank Problem, Z = 250



Acceptance rate: 2.749^{-4}

The German Tank Problem, ~~Z~~ y = 250





Importance Sampling

- With rejection sampling we get actual samples from the target density
- Downside: often throw away many samples. Inefficient if rejection rate is high.
  - Main challenge is finding a valid proposal density and one that minimzes the number of rejections
- Importance sampling: use all samples from proposal to compute expected values.
- Weight the *importance* of each sample by how representative it is of the target distribution

Want $E_{p(\theta|y)}[h(\theta)] \overset{\text{LOTUS}}{=}$

$$= \underbrace{\int h(\theta) \frac{p(\theta|y)}{q(\theta)}}\, q(\theta)\, d\theta$$

$$= E_{q(\theta)}\left[h(\theta) p(\theta) / q(\theta) \right] \qquad p/q = w(\theta)$$

$$\approx \frac{1}{S} \sum^{S} w(\theta_s) h(\theta_s) \qquad \theta_s \overset{iid}{\sim} q(\theta)$$

$$E_q[w(\theta)] = \int \frac{p(\theta|y)}{q(\theta)} q(\theta)\, d\theta = 1$$

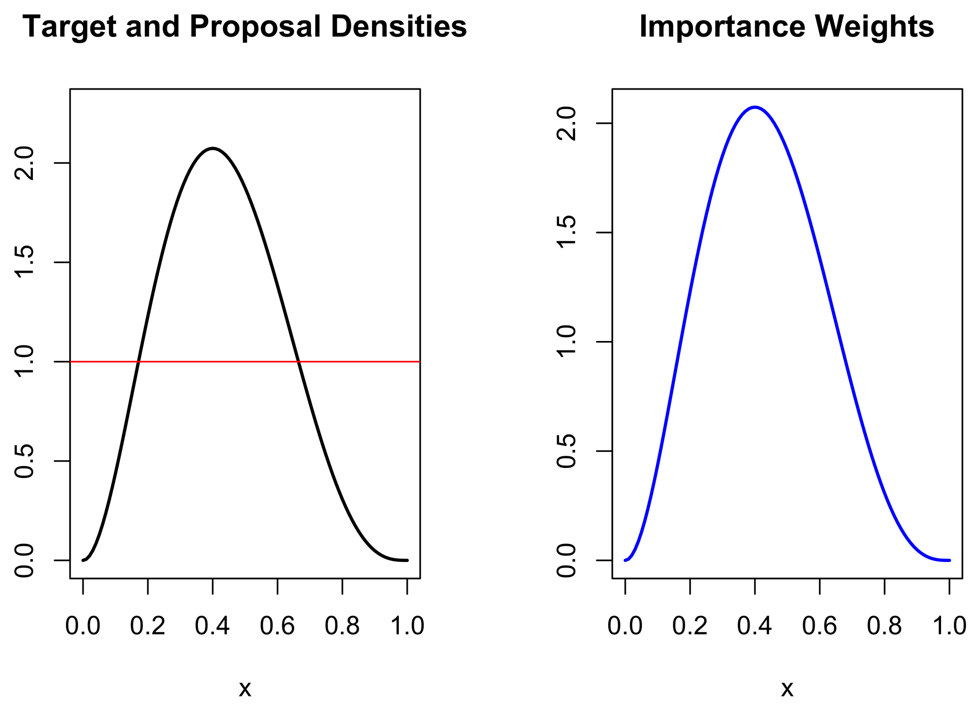
$$\tilde{w}(\theta) = \frac{\tilde{p}(\theta|y)}{q(\theta)} \Big/ \sum \tilde{p}(\theta|y) / q(\theta)$$

Importance Sampling

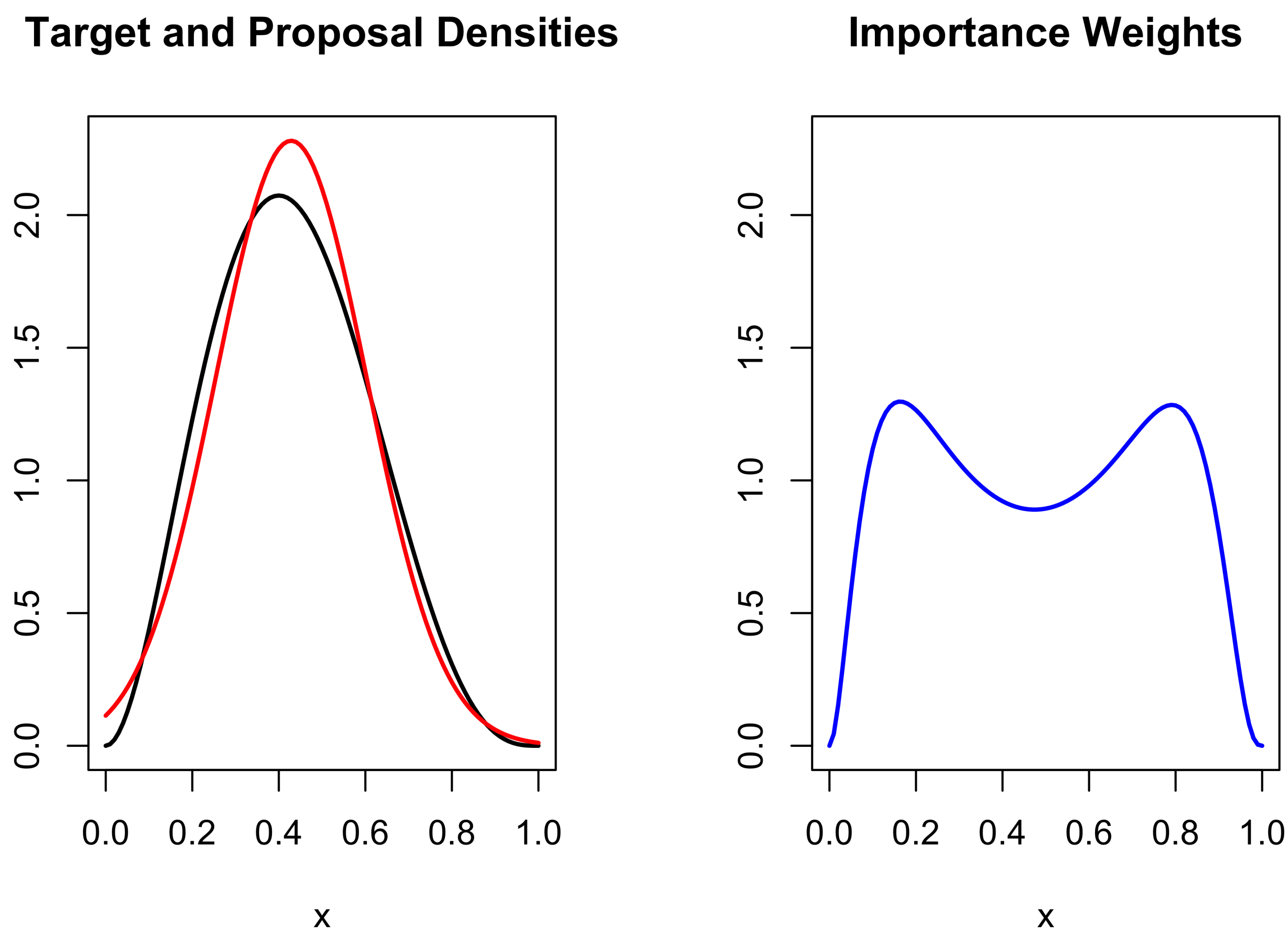
- Examine histogram of log importance weights
- Want the largest weights to be not too large
- Effective sample size: $\frac{1}{\sum_{s=1}^{S} \tilde{w}(\theta^s)^2}$ where $\tilde{w}(\theta^s)$ are the normalized weights

Importance Sampling

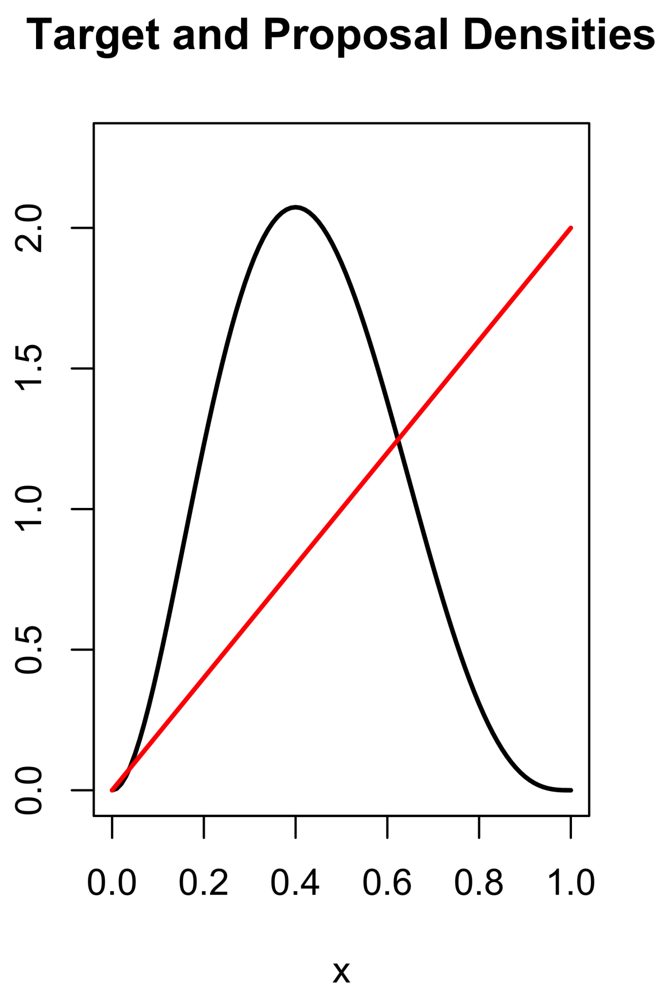
- Target: a Beta(3, 4)
- Proposal: Uniform

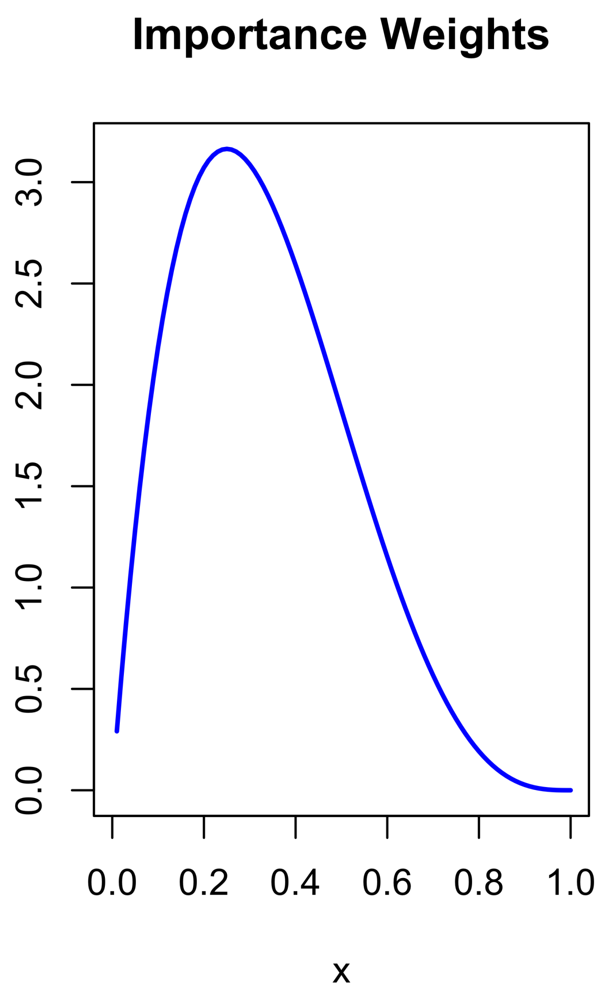


Importance Sampling



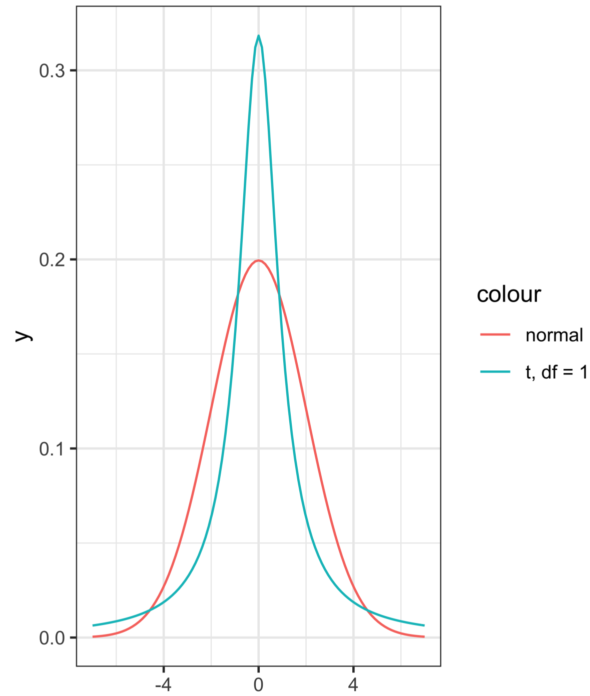
Importance Sampling

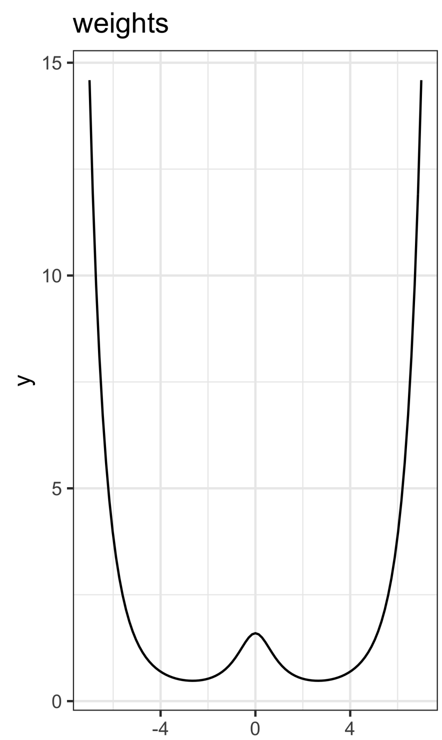




Importance Sampling

- Target: Cauchy(0, 1)
- Proposal: Normal(0, 3)





Importance Sampling

```
df <- 1

proposal_draws <- rnorm(1e5, sd=2)
proposal_draws <- sort(proposal_draws)

w <- dt(proposal_draws, df=df) / dnorm(proposal_draws, sd=2)
w <- w/sum(w)

q <- 0.25
idx <- match(FALSE, cumsum(w) < q)
qt(q, df=df)
```

```
[1] -1
```

```
proposal_draws[idx]
```

```
[1] -0.860783
```

```
q <- 0.05
idx <- match(FALSE, cumsum(w) < q)
qt(q, df=df)
```

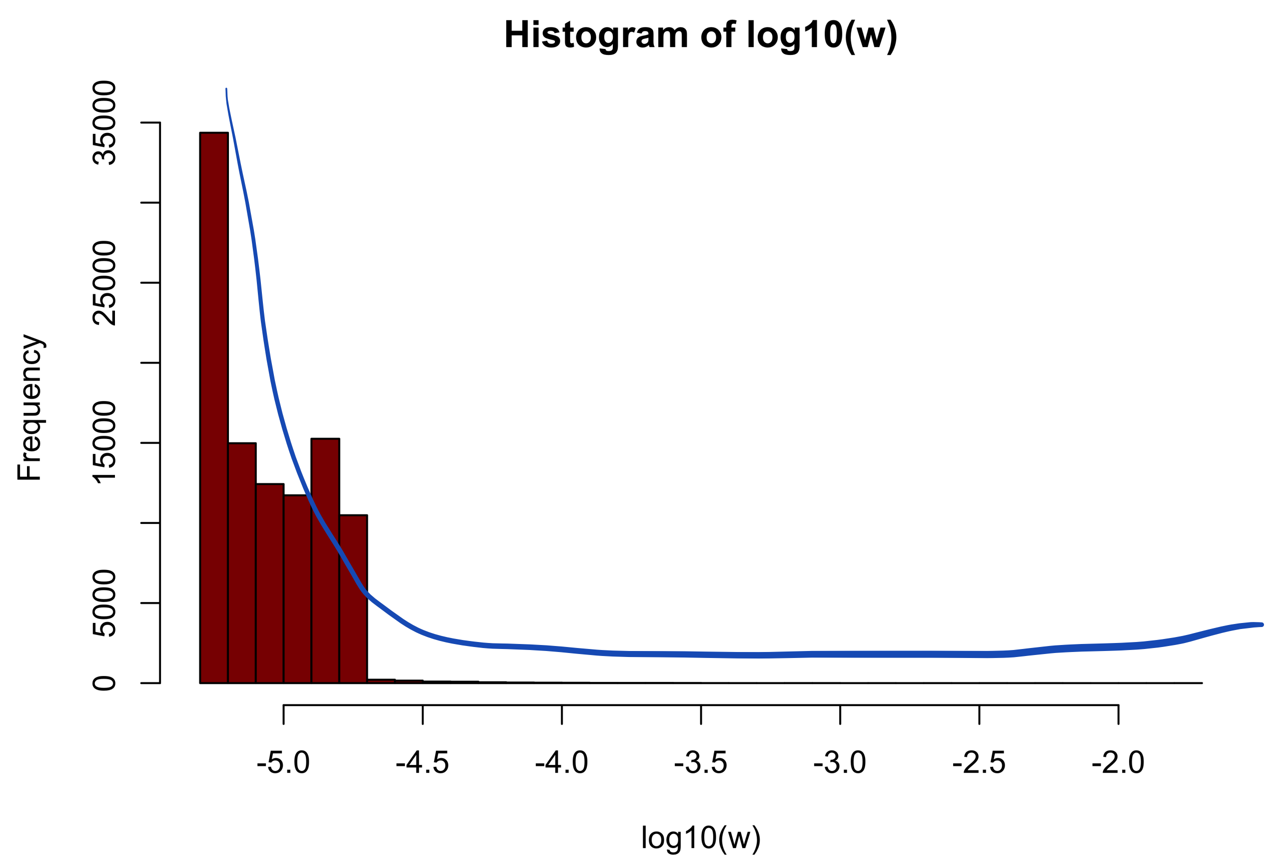
```
[1] -6.313752
```

```
proposal_draws[idx]
```

```
[1] -3.738269
```

Importance Sampling

```
[1] "99% quantile of log10(w) is -4.773680"
[1] "max of log10(w) is -1.769665"
```



A Gamma Puzzle

- Goal: report the exact mean of a Gamma(a, b) using only one random draw
- Allowed to generate a random draw from a proposal distribution of your choosing
- You can evaluate the gamma density and the proposal density at the generated value

$$\int x p(x) dx = \int_0^\infty x \frac{b^a e^{-bx} x^{a-1}}{\Gamma(a)} dx$$

Propose $\text{Gam}(a+1, b)$

$$= \int \frac{\cancel{x}\,\dfrac{\cancel{b^{a}}\,\cancel{e^{-bx}}\,\cancel{x^{a-1}}}{\Gamma(a)}}{\dfrac{b^{\cancel{a+1}}\,\cancel{e^{-bx}}\,\cancel{x^{a}}}{\Gamma(a+1)}} \frac{b^{a+1} e^{-bx} x^{a}}{\Gamma(a+1)}\, dx$$

No x

$$= \int \bigcirc\; \frac{b^{a+1} e^{-bx} x^{a}}{\Gamma(a+1)}\, dx$$

$$\int h \frac{p}{q}$$

A Gamma Puzzle

```
## Generate 10 draws from a gamma 5, 2
proposal_draws <- rgamma(10, shape = 5, rate=2)

## Targate a gamma 4, 2
w <-
  dgamma(proposal_draws, shape=4, rate=2) /
  dgamma(proposal_draws, shape=5, rate=2)

w*proposal_draws
```

```
[1] 2 2 2 2 2 2 2 2 2 2
```