

# Chapter 10

- 10.1 Numerical integration (overview)
- 10.2 Distributional approximations (overview, more in Chapter 4 and 13)
- 10.3 Direct simulation and rejection sampling (overview)
- 10.4 Importance sampling (used in PSIS-LOO discussed later)
- 10.5 How many simulation draws are needed? (Ex 10.1 and 10.2)
- 10.6 Software (can be skipped)
- 10.7 Debugging (can be skipped)

# Notation

- In this chapter, generic  $p(\theta)$  is used instead of  $p(\theta|y)$
- unnormalized distribution is denoted by  $q(\cdot)$
- proposal distribution is denoted by  $g(\cdot)$

# Numerical accuracy

- Floating point presentation of numbers. e.g. with 64bits
  - closest value to zero is  $\approx 2.2 \cdot 10^{-308}$ 
    - `qr=rnorm(600);prod(dnorm(qr))`  $\rightarrow 0$

# Numerical accuracy

- Floating point presentation of numbers. e.g. with 64bits
  - closest value to zero is  $\approx 2.2 \cdot 10^{-308}$ 
    - `qr=rnorm(600);prod(dnorm(qr))`  $\rightarrow 0$
  - closest value to 1 is  $\approx 1 \pm 2.2 \cdot 10^{-16}$ 
    - `pbeta(0.5,241945,251527)`  $\rightarrow 1$

# Numerical accuracy

- Floating point presentation of numbers. e.g. with 64bits
  - closest value to zero is  $\approx 2.2 \cdot 10^{-308}$ 
    - `qr=rnorm(600);prod(dnorm(qr))`  $\rightarrow 0$
    - `pbeta(0.5,241945,251527, lower.tail=FALSE)`  
 $\approx -1.15 \cdot 10^{-42}$
  - closest value to 1 is  $\approx 1 \pm 2.2 \cdot 10^{-16}$ 
    - `pbeta(0.5,241945,251527)`  $\rightarrow 1$

# Numerical accuracy

- Floating point presentation of numbers. e.g. with 64bits
  - closest value to zero is  $\approx 2.2 \cdot 10^{-308}$ 
    - `qr=rnorm(600);prod(dnorm(qr))`  $\rightarrow 0$
    - `pbeta(0.5,241945,251527, lower.tail=FALSE)`  
 $\approx -1.15 \cdot 10^{-42}$
  - closest value to 1 is  $\approx 1 \pm 2.2 \cdot 10^{-16}$ 
    - `pbeta(0.5,241945,251527)`  $\rightarrow 1$
- Log densities
  - use log densities to avoid over- and underflows in floating point presentation
    - `sum(dnorm(qr,log=TRUE))`  $\rightarrow -847.3$

# Numerical accuracy

- Floating point presentation of numbers. e.g. with 64bits
  - closest value to zero is  $\approx 2.2 \cdot 10^{-308}$ 
    - `qr=rnorm(600);prod(dnorm(qr))`  $\rightarrow 0$
    - `pbeta(0.5,241945,251527, lower.tail=FALSE)`  
 $\approx -1.15 \cdot 10^{-42}$
  - closest value to 1 is  $\approx 1 \pm 2.2 \cdot 10^{-16}$ 
    - `pbeta(0.5,241945,251527)`  $\rightarrow 1$
- Log densities
  - use log densities to avoid over- and underflows in floating point presentation
    - `sum(dnorm(qr,log=TRUE))`  $\rightarrow -847.3$
    - how many observations we can now handle?

# Numerical accuracy

- Floating point presentation of numbers. e.g. with 64bits
  - closest value to zero is  $\approx 2.2 \cdot 10^{-308}$ 
    - `qr=rnorm(600);prod(dnorm(qr))`  $\rightarrow 0$
    - `pbeta(0.5,241945,251527, lower.tail=FALSE)`  
 $\approx -1.15 \cdot 10^{-42}$
  - closest value to 1 is  $\approx 1 \pm 2.2 \cdot 10^{-16}$ 
    - `pbeta(0.5,241945,251527)`  $\rightarrow 1$
- Log densities
  - use log densities to avoid over- and underflows in floating point presentation
    - `sum(dnorm(qr,log=TRUE))`  $\rightarrow -847.3$
    - how many observations we can now handle?
  - compute exp as late as possible
    - e.g. for  $a > b$ , compute  
 $\log(\exp(a) + \exp(b)) = a + \log(1 + \exp(b - a))$   
e.g.  $\log(\exp(800) + \exp(800)) \rightarrow \text{Inf}$ , but  
 $800 + \log(1 + \exp(800 - 800)) \approx 800.69$



# Numerical accuracy

- Floating point presentation of numbers. e.g. with 64bits
  - closest value to zero is  $\approx 2.2 \cdot 10^{-308}$ 
    - `qr=rnorm(600);prod(dnorm(qr))`  $\rightarrow 0$
    - `pbeta(0.5,241945,251527, lower.tail=FALSE)`  
 $\approx -1.15 \cdot 10^{-42}$
  - closest value to 1 is  $\approx 1 \pm 2.2 \cdot 10^{-16}$ 
    - `pbeta(0.5,241945,251527)`  $\rightarrow 1$
- Log densities
  - use log densities to avoid over- and underflows in floating point presentation
    - `sum(dnorm(qr,log=TRUE))`  $\rightarrow -847.3$
    - how many observations we can now handle?
  - compute exp as late as possible
    - e.g. for  $a > b$ , compute  
 $\log(\exp(a) + \exp(b)) = a + \log(1 + \exp(b - a))$   
e.g.  $\log(\exp(800) + \exp(800)) \rightarrow \text{Inf}$ , but  
 $800 + \log(1 + \exp(800 - 800)) \approx 800.69$
    - e.g. in Metropolis-algorithm compute the log of ratio of densities using the identity  
 $\log(a/b) = \log(a) - \log(b)$

# It's all about expectations

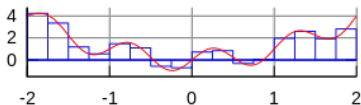
$$E_{\theta}[f(\theta)] = \int f(\theta)p(\theta|y)d\theta$$

- Conjugate priors and analytic solutions
- Grid integration and other quadrature rules
- Independent Monte Carlo, rejection and importance sampling
- Markov Chain Monte Carlo
- Distributional approximations (Laplace, VB, EP)

# Quadrature integration

- The simplest quadrature integration is grid integration
  - Evaluate function in a grid and compute

$$E[-\alpha/\beta] \approx \sum_{t=1}^T w_{\text{cell}}^{(t)} \frac{\alpha^{(t)}}{\beta^{(t)}},$$



where  $w_{\text{cell}}^{(t)}$  is the normalized probability of a grid cell  $t$ , and  $\alpha^{(t)}$  and  $\beta^{(t)}$  are center locations of grid cells

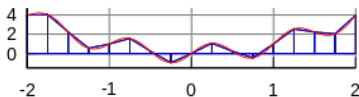
# Quadrature integration

- The simplest quadrature integration is grid integration
  - Evaluate function in a grid and compute

$$E[-\alpha/\beta] \approx \sum_{t=1}^T w_{\text{cell}}^{(t)} \frac{\alpha^{(t)}}{\beta^{(t)}},$$

where  $w_{\text{cell}}^{(t)}$  is the normalized probability of a grid cell  $t$ , and  $\alpha^{(t)}$  and  $\beta^{(t)}$  are center locations of grid cells

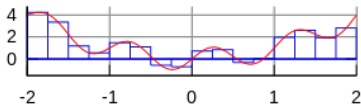
- In 1D further variations, e.g. trapezoid



# Quadrature integration

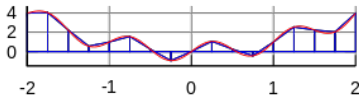
- The simplest quadrature integration is grid integration
  - Evaluate function in a grid and compute

$$E[-\alpha/\beta] \approx \sum_{t=1}^T w_{\text{cell}}^{(t)} \frac{\alpha^{(t)}}{\beta^{(t)}},$$



where  $w_{\text{cell}}^{(t)}$  is the normalized probability of a grid cell  $t$ , and  $\alpha^{(t)}$  and  $\beta^{(t)}$  are center locations of grid cells

- In 1D further variations, e.g. trapezoid



- In 2D and higher
  - nested quadrature
  - product rules

# Monte Carlo - history

- Used already before computers
  - Buffon (18th century; needles)
  - De Forest, Darwin, Galton (19th century)
  - Pearson (19th century; roulette)
  - Gosset (Student, 1908; hat)

# Monte Carlo - history

- Used already before computers
  - Buffon (18th century; needles)
  - De Forest, Darwin, Galton (19th century)
  - Pearson (19th century; roulette)
  - Gosset (Student, 1908; hat)
- "Monte Carlo method" term was proposed by Metropolis, von Neumann or Ulam in the end of 1940s
  - they worked together in atomic bomb project
  - Metropolis and Ulam, "The Monte Carlo Method", 1949

# Monte Carlo - history

- Used already before computers
  - Buffon (18th century; needles)
  - De Forest, Darwin, Galton (19th century)
  - Pearson (19th century; roulette)
  - Gosset (Student, 1908; hat)
- "Monte Carlo method" term was proposed by Metropolis, von Neumann or Ulam in the end of 1940s
  - they worked together in atomic bomb project
  - Metropolis and Ulam, "The Monte Carlo Method", 1949
- Bayesians started to have enough cheap computation time in 1990s
  - BUGS project started 1989 (last OpenBUGS release 2014)
  - Gelfand & Smith, 1990
  - Stan initial release 2012



# Monte Carlo

- Simulate draws from the target distribution
  - these draws can be treated as any observations
  - a collection of draws is sample
- Use these draws, for example,
  - to compute means, deviations, quantiles
  - to draw histograms
  - to marginalize
  - etc.

# Monte Carlo vs. deterministic

- Monte Carlo = simulation methods
  - evaluation points are selected stochastically (randomly)
- Deterministic methods (e.g. grid)
  - evaluation points are selected by some deterministic rule

# How many simulation draws are needed?

- How many draws or how big sample size?
- If draws are independent
  - usual methods to estimate the uncertainty due to a finite number of observations (finite sample size)
- Markov chain Monte Carlo produces dependent draws
  - requires additional work to estimate the **effective sample size**

# How many simulation draws are needed?

- Expectation of unknown quantity

$$E(\theta) \approx \frac{1}{L} \sum_l \theta^{(l)}$$

if  $L$  is big and  $\theta^{(l)}$  are independent, way may assume that the distribution of the expectation approaches normal distribution (see Ch 4) with variance  $\sigma_\theta^2/L$  (asymptotic normality)

- this variance is independent on dimensionality of  $\theta$

# How many simulation draws are needed?

- Expectation of unknown quantity

$$E(\theta) \approx \frac{1}{L} \sum_l \theta^{(l)}$$

if  $L$  is big and  $\theta^{(l)}$  are independent, way may assume that the distribution of the expectation approaches normal distribution (see Ch 4) with variance  $\sigma_\theta^2/L$  (asymptotic normality)

- this variance is independent on dimensionality of  $\theta$
- total variance is sum of the epistemic uncertainty in the posterior and the uncertainty due to using finite number of Monte Carlo draws

$$\sigma_\theta^2 + \sigma_\theta^2/L$$

# How many simulation draws are needed?

- Expectation of unknown quantity

$$E(\theta) \approx \frac{1}{L} \sum_l \theta^{(l)}$$

if  $L$  is big and  $\theta^{(l)}$  are independent, way may assume that the distribution of the expectation approaches normal distribution (see Ch 4) with variance  $\sigma_\theta^2/L$  (asymptotic normality)

- this variance is independent on dimensionality of  $\theta$
- total variance is sum of the epistemic uncertainty in the posterior and the uncertainty due to using finite number of Monte Carlo draws

$$\sigma_\theta^2 + \sigma_\theta^2/L = \sigma_\theta^2(1 + 1/L)$$

# How many simulation draws are needed?

- Expectation of unknown quantity

$$E(\theta) \approx \frac{1}{L} \sum_l \theta^{(l)}$$

if  $L$  is big and  $\theta^{(l)}$  are independent, way may assume that the distribution of the expectation approaches normal distribution (see Ch 4) with variance  $\sigma_\theta^2/L$  (asymptotic normality)

- this variance is independent on dimensionality of  $\theta$
- total variance is sum of the epistemic uncertainty in the posterior and the uncertainty due to using finite number of Monte Carlo draws

$$\sigma_\theta^2 + \sigma_\theta^2/L = \sigma_\theta^2(1 + 1/L)$$

- e.g. if  $L = 100$ , deviation increases by  $\sqrt{1 + 1/L} = 1.005$   
ie. Monte Carlo error is very small (for the expectation)

# How many simulation draws are needed?

- Expectation of unknown quantity

$$E(\theta) \approx \frac{1}{L} \sum_l \theta^{(l)}$$

if  $L$  is big and  $\theta^{(l)}$  are independent, way may assume that the distribution of the expectation approaches normal distribution (see Ch 4) with variance  $\sigma_\theta^2/L$  (asymptotic normality)

- this variance is independent on dimensionality of  $\theta$
- total variance is sum of the epistemic uncertainty in the posterior and the uncertainty due to using finite number of Monte Carlo draws

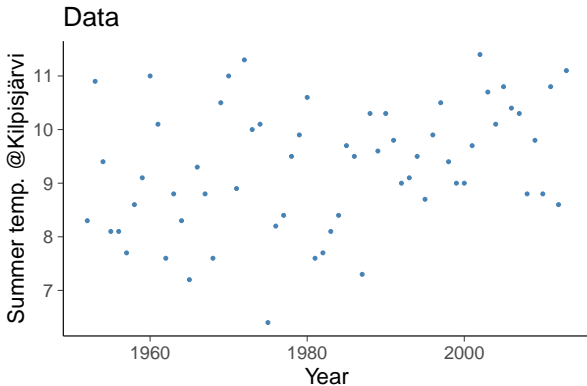
$$\sigma_\theta^2 + \sigma_\theta^2/L = \sigma_\theta^2(1 + 1/L)$$

- e.g. if  $L = 100$ , deviation increases by  $\sqrt{1 + 1/L} = 1.005$   
ie. Monte Carlo error is very small (for the expectation)
- See Ch 4 for counter-examples for asymptotic normality



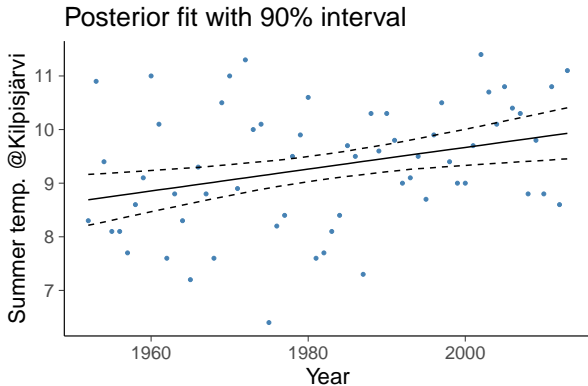
# Example: Kilpisjärvi summer temperature

Average temperature in June, July, and August at Kilpisjärvi, Finland



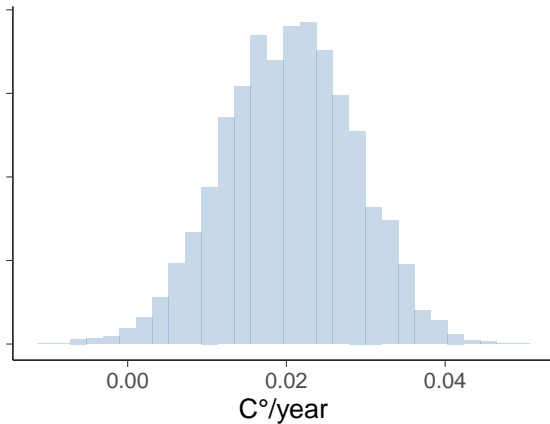
# Example: Kilpisjärvi summer temperature

Average temperature in June, July, and August at Kilpisjärvi, Finland



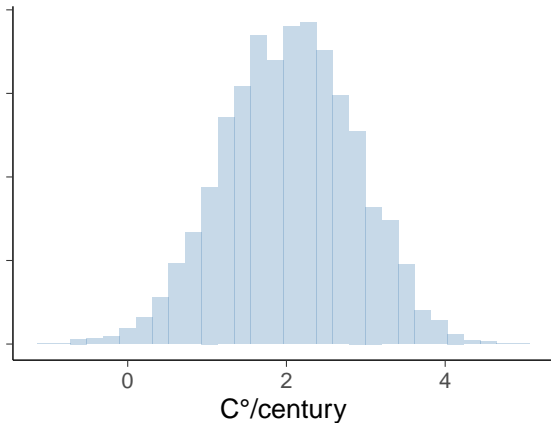
## Example: Kilpisjärvi summer temperature

Posterior of temperature change



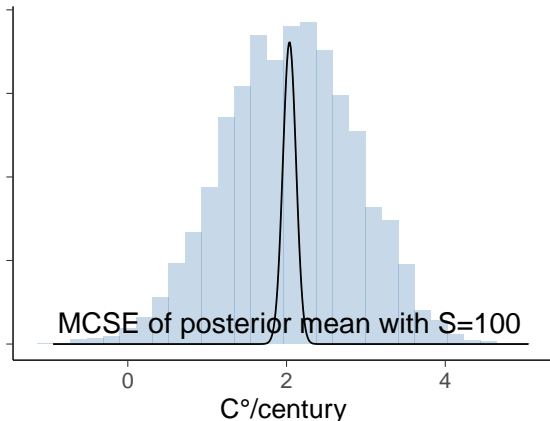
## Example: Kilpisjärvi summer temperature

Posterior of temperature change



## Example: Kilpisjärvi summer temperature

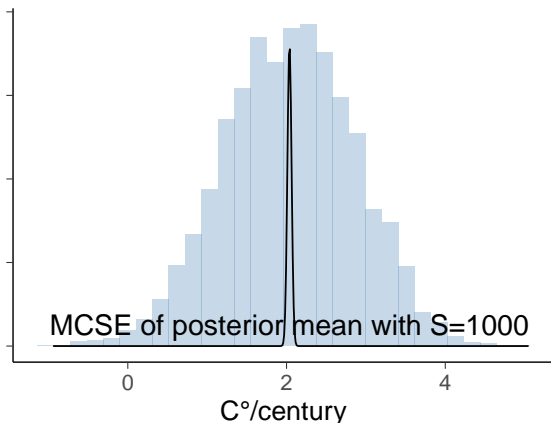
Posterior of temperature change



$\sigma_{\theta} \approx 0.827$ ,  $\text{MCSE} \approx 0.0827$ , total deviation  $\approx 0.831$

## Example: Kilpisjärvi summer temperature

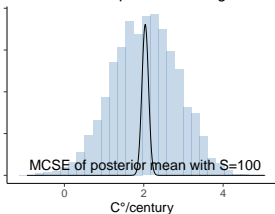
Posterior of temperature change



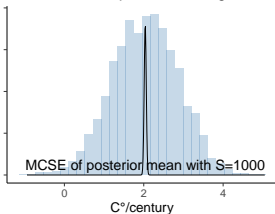
$\sigma_{\theta} \approx 0.827$ , MCSE  $\approx 0.0261$ , total deviation  $\approx 0.827$

# Example: Kilpisjärvi summer temperature

Posterior of temperature change

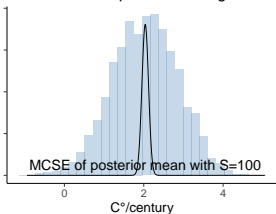


Posterior of temperature change

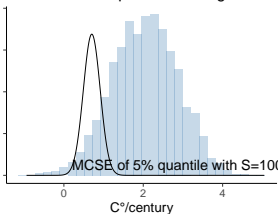


# Example: Kilpisjärvi summer temperature

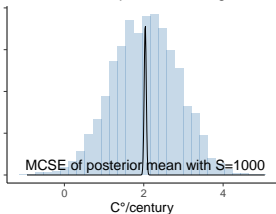
Posterior of temperature change



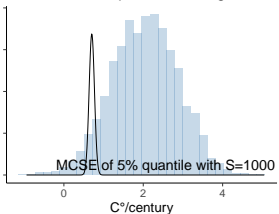
Posterior of temperature change



Posterior of temperature change



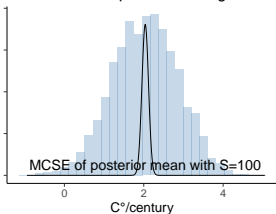
Posterior of temperature change



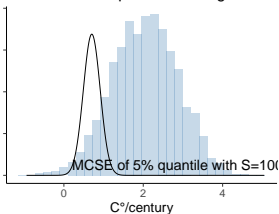


# Example: Kilpisjärvi summer temperature

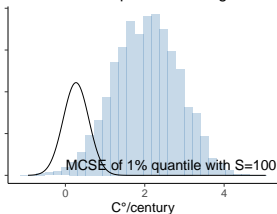
Posterior of temperature change



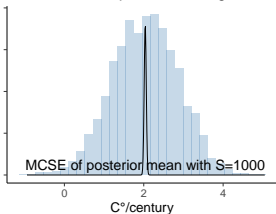
Posterior of temperature change



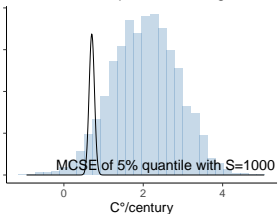
Posterior of temperature change



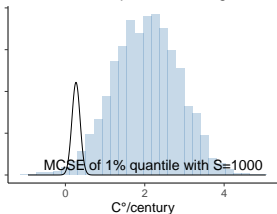
Posterior of temperature change



Posterior of temperature change

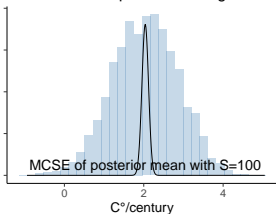


Posterior of temperature change

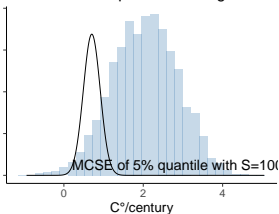


# Example: Kilpisjärvi summer temperature

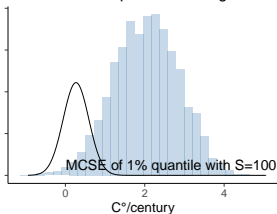
Posterior of temperature change



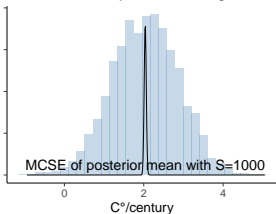
Posterior of temperature change



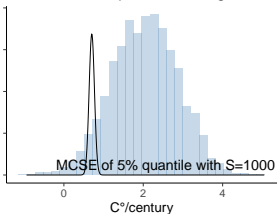
Posterior of temperature change



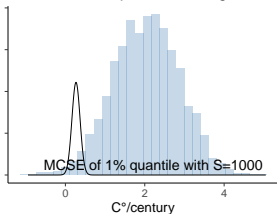
Posterior of temperature change



Posterior of temperature change



Posterior of temperature change



Tail quantiles are more difficult to estimate

# How many simulation draws are needed?

- Posterior probability

$$p(\theta \in A) \approx \frac{1}{L} \sum_l I(\theta^{(l)} \in A)$$

where  $I(\theta^{(l)} \in A) = 1$  if  $\theta^{(l)} \in A$

- $I(\cdot)$  is binomially distributed as  $p(\theta \in A)$ 
  - $\text{var}(I(\cdot)) = p(1 - p)$  (Appendix A, p. 579)
  - standard deviation of  $p$  is  $\sqrt{p(1 - p)/L}$

# How many simulation draws are needed?

- Posterior probability

$$p(\theta \in A) \approx \frac{1}{L} \sum_l I(\theta^{(l)} \in A)$$

where  $I(\theta^{(l)} \in A) = 1$  if  $\theta^{(l)} \in A$

- $I(\cdot)$  is binomially distributed as  $p(\theta \in A)$ 
  - $\text{var}(I(\cdot)) = p(1 - p)$  (Appendix A, p. 579)
  - standard deviation of  $p$  is  $\sqrt{p(1 - p)/L}$
- if  $L = 100$  and  $p \approx 0.5$ ,  $\sqrt{p(1 - p)/L} = 0.05$   
ie. accuracy is about 5% units

# How many simulation draws are needed?

- Posterior probability

$$p(\theta \in A) \approx \frac{1}{L} \sum_l I(\theta^{(l)} \in A)$$

where  $I(\theta^{(l)} \in A) = 1$  if  $\theta^{(l)} \in A$

- $I(\cdot)$  is binomially distributed as  $p(\theta \in A)$ 
  - $\text{var}(I(\cdot)) = p(1 - p)$  (Appendix A, p. 579)
  - standard deviation of  $p$  is  $\sqrt{p(1 - p)/L}$
- if  $L = 100$  and  $p \approx 0.5$ ,  $\sqrt{p(1 - p)/L} = 0.05$   
ie. accuracy is about 5% units
- $L = 2500$  draws needed for 1% unit accuracy

# How many simulation draws are needed?

- Posterior probability

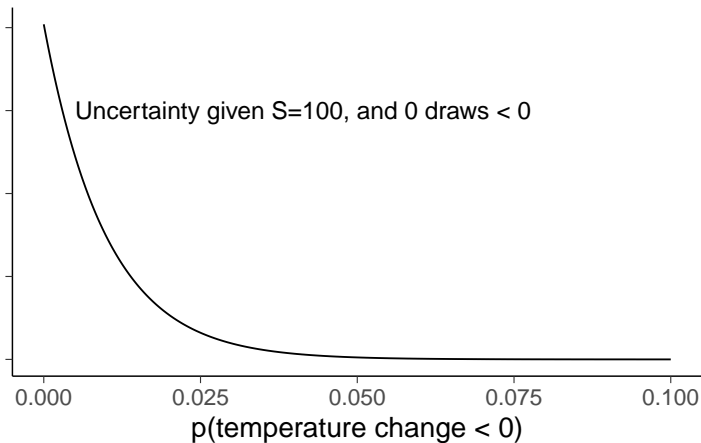
$$p(\theta \in A) \approx \frac{1}{L} \sum_l I(\theta^{(l)} \in A)$$

where  $I(\theta^{(l)} \in A) = 1$  if  $\theta^{(l)} \in A$

- $I(\cdot)$  is binomially distributed as  $p(\theta \in A)$ 
  - $\text{var}(I(\cdot)) = p(1 - p)$  (Appendix A, p. 579)
  - standard deviation of  $p$  is  $\sqrt{p(1 - p)/L}$
- if  $L = 100$  and  $p \approx 0.5$ ,  $\sqrt{p(1 - p)/L} = 0.05$   
ie. accuracy is about 5% units
- $L = 2500$  draws needed for 1% unit accuracy
- To estimate small probabilities, a large number of draws is needed
  - to be able to estimate  $p$ , need to get draws with  $\theta^{(l)} \in A$ ,  
which in expectation requires  $L \gg 1/p$

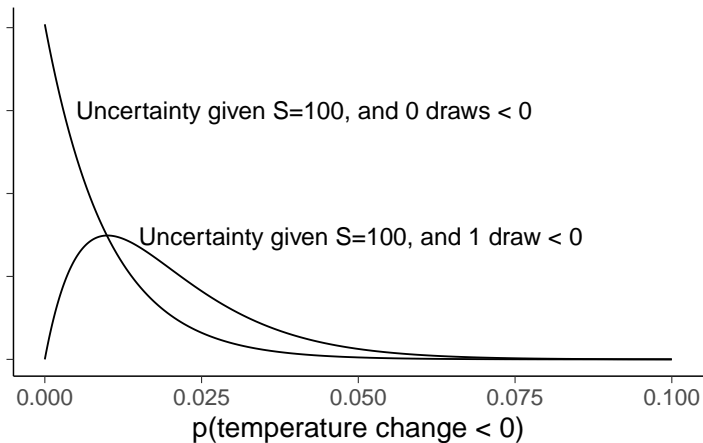
## Example: Kilpisjärvi summer temperature

Posterior uncertainty  $p(\text{temperature change} < 0)$



## Example: Kilpisjärvi summer temperature

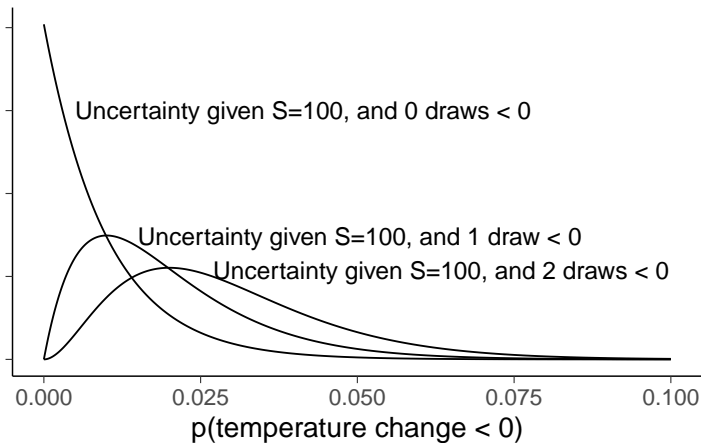
Posterior uncertainty  $p(\text{temperature change} < 0)$





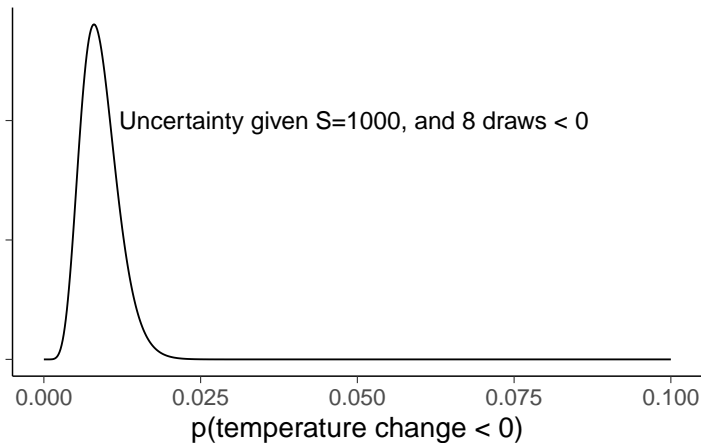
## Example: Kilpisjärvi summer temperature

Posterior uncertainty  $p(\text{temperature change} < 0)$



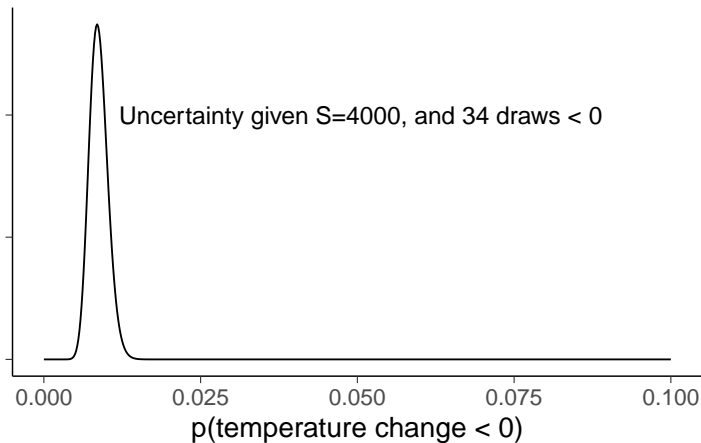
## Example: Kilpisjärvi summer temperature

Posterior uncertainty  $p(\text{temperature change} < 0)$



## Example: Kilpisjärvi summer temperature

Posterior uncertainty  $p(\text{temperature change} < 0)$



# How many simulation draws are needed?

- Less draws needed with
  - deterministic methods
  - marginalization (Rao-Blackwellization)
  - variance reduction methods, such, control variates

## How many simulation draws are needed?

- Number of independent draws needed doesn't depend on the number of dimensions
  - but it may be difficult to obtain independent draws in high dimensional case

# Direct simulation

- Produces independent draws
  - Using analytic transformations of uniform random numbers (eg. appendix A)
  - factorization
  - numerical inverse-cdf
- Problem: restricted to limited set of models

# Random number generators

- Good pseudo random number generators are sufficient for Bayesian inference
  - pseudo random generator uses deterministic algorithm to produce a sequence which is difficult to make difference from truly random sequence
  - modern software used for statistical analysis have good pseudo RNGs

## Direct simulation: Example

- Box-Muller -method:

If  $U_1$  and  $U_2$  are independent draws from distribution  $U(0, 1)$ , and

$$X_1 = \sqrt{-2 \log(U_1)} \cos(2\pi U_2)$$

$$X_2 = \sqrt{-2 \log(U_1)} \sin(2\pi U_2)$$

then  $X_1$  and  $X_2$  are independent draws from the distribution  $N(0, 1)$



## Direct simulation: Example

- Box-Muller -method:

If  $U_1$  and  $U_2$  are independent draws from distribution  $U(0, 1)$ , and

$$X_1 = \sqrt{-2 \log(U_1)} \cos(2\pi U_2)$$

$$X_2 = \sqrt{-2 \log(U_1)} \sin(2\pi U_2)$$

then  $X_1$  and  $X_2$  are independent draws from the distribution  $N(0, 1)$

- not the fastest method due to trigonometric computations
- for normal distribution more than ten different methods
- e.g. R uses inverse-CDF

# Grid sampling and curse of dimensionality

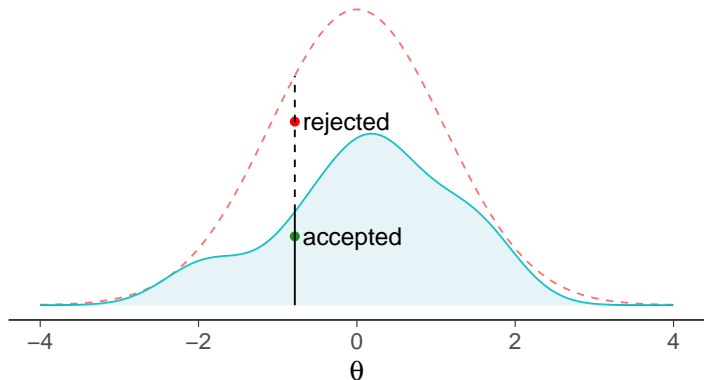
- 10 parameters
- if we don't know beforehand where the posterior mass is
  - need to choose wide box for the grid
  - need to have enough grid points to get some of them where essential mass is
- e.g. 50 or 1000 grid points per dimension
  - $50^{10} \approx 1\text{e}17$  grid points
  - $1000^{10} \approx 1\text{e}30$  grid points
- R and my current laptop can compute density of normal distribution about 20 million times per second
  - evaluation in  $1\text{e}17$  grid points would take 150 years
  - evaluation in  $1\text{e}30$  grid points would take 1 500 billion years

## Indirect sampling

- Rejection sampling
- Importance sampling
- Markov chain Monte Carlo (next week)

# Rejection sampling

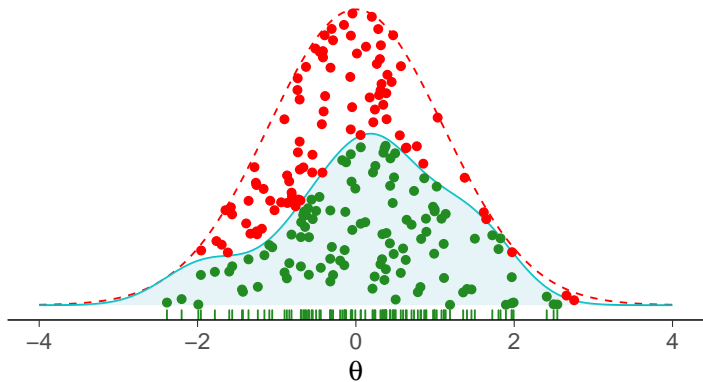
- Proposal forms envelope over the target distribution  
 $q(\theta|y)/Mg(\theta) \leq 1$
- Draw from the proposal and accept with probability  
 $Mg(\theta)/q(\theta|y)$



--  $Mg(\theta)$  —  $q(\theta|y)$

# Rejection sampling

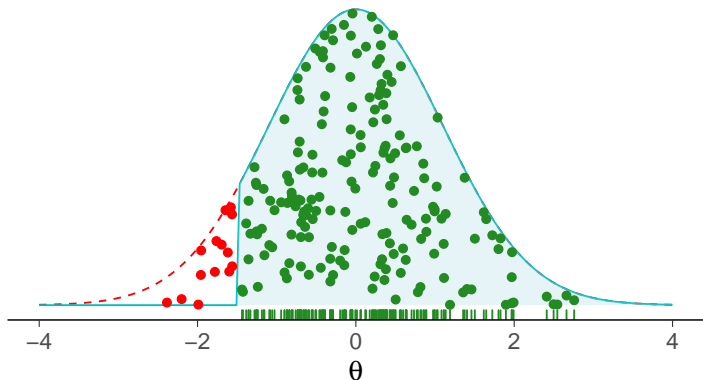
- Proposal forms envelope over the target distribution  
 $q(\theta|y)/Mg(\theta) \leq 1$
- Draw from the proposal and accept with probability  
 $Mg(\theta)/q(\theta|y)$



• Accepted • Rejected - - Mg(theta) — q(theta|y)

# Rejection sampling

- Proposal forms envelope over the target distribution  
 $q(\theta|y)/Mg(\theta) \leq 1$
- Draw from the proposal and accept with probability  
 $Mg(\theta)/q(\theta|y)$
- Common for truncated distributions



• Accepted • Rejected - - Mg(theta) — q(theta|y)

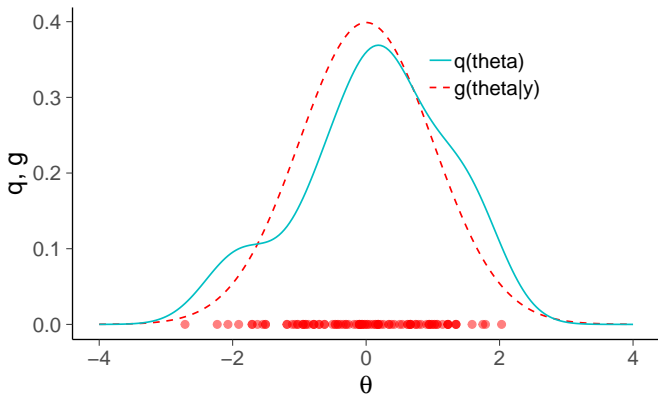
# Rejection sampling

- The number of accepted draws is the effective sample size
  - with bad proposal distribution may require a lot of trials
  - selection of good proposal gets very difficult when the number of dimensions increase
  - reliable diagnostics and thus can be a useful part

# Importance sampling

- Proposal does not need to have a higher value everywhere

Target, proposal, and draws

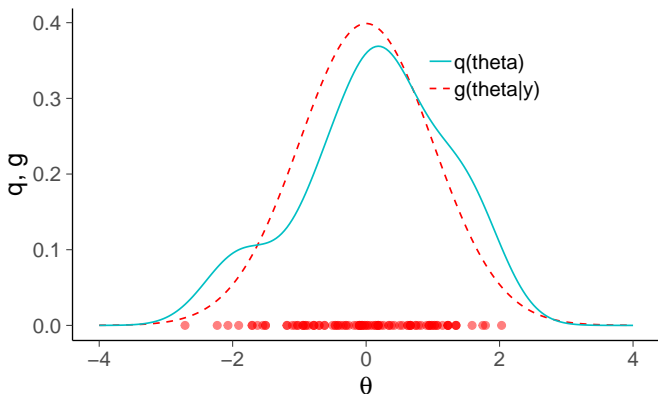




# Importance sampling

- Proposal does not need to have a higher value everywhere

Target, proposal, and draws

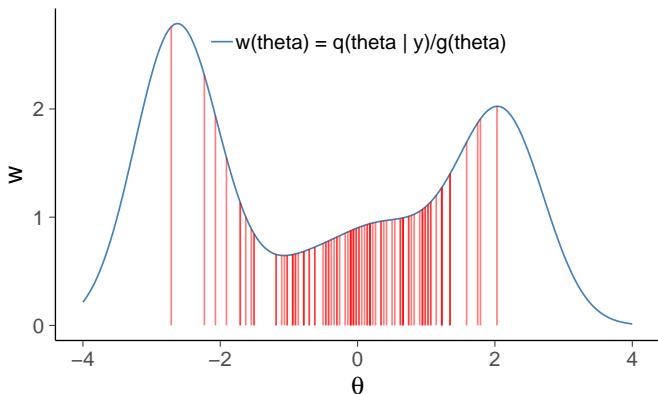


$$E[f(\theta)] \approx \frac{\sum_s w_s f(\theta^{(s)})}{\sum_s w_s}, \quad \text{where} \quad w_s = \frac{q(\theta^{(s)})}{g(\theta^{(s)})}$$

# Importance sampling

- Proposal does not need to have a higher value everywhere

## Draws and importance weights



$$E[f(\theta)] \approx \frac{\sum_s w_s f(\theta^{(s)})}{\sum_s w_s}, \quad \text{where} \quad w_s = \frac{q(\theta^{(s)})}{g(\theta^{(s)})}$$

# Importance sampling

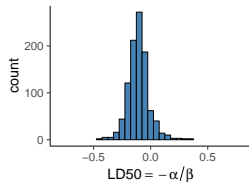
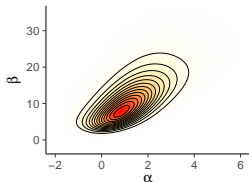
- Resampling using normalized importance weights can be used to pick a smaller number of draws with uniform weights
- Selection of good proposal gets more difficult when the number of dimensions increase
- Often used to correct distributional approximations

# Importance sampling

- Variation of the weights affect the effective sample size
  - if single weight dominates, we have effectively one sample
  - if weights are equal, we have effectively  $S$  draws
- Central limit theorem holds only if variance of the weight distribution is finite
- See Vehtari, Gelman and Gabry (2017). Pareto smoothed importance sampling. arXiv preprint arXiv:1507.02646, <https://arxiv.org/abs/1507.02646> for improved diagnostics and stability.

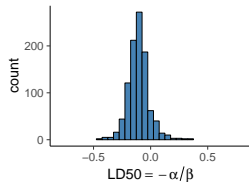
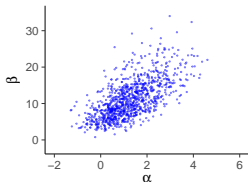
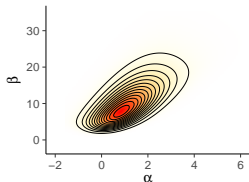
# Exmple: Importance sampling in Bioassay

Grid

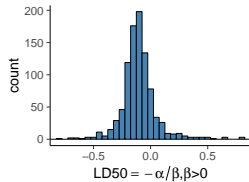
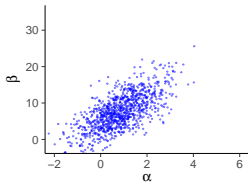
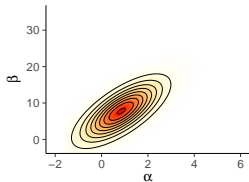


# Exmple: Importance sampling in Bioassay

Grid

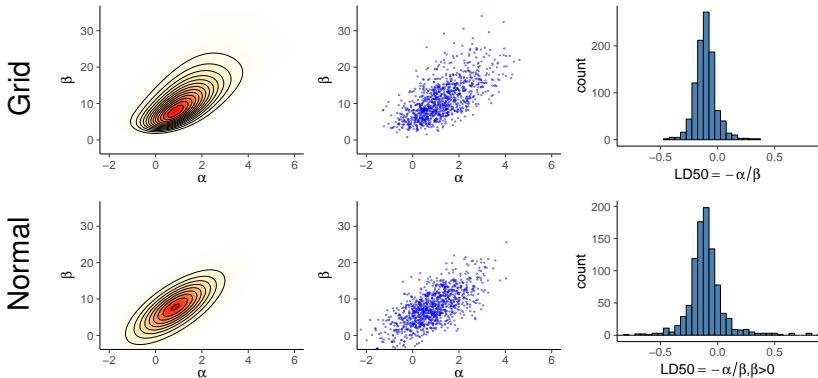


Normal



Normal approximation is discussed more in BDA3 Ch 4

## Exmple: Importance sampling in Bioassay



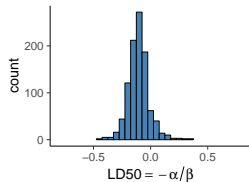
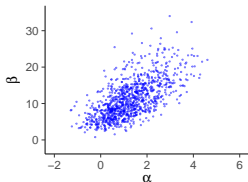
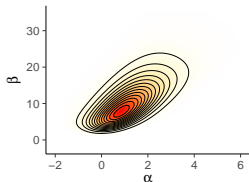
Normal approximation is discussed more in BDA3 Ch 4

But the normal approximation is not that good here:

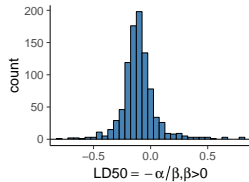
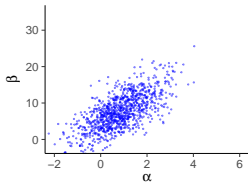
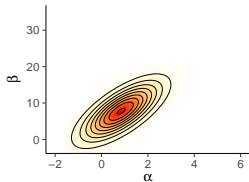
Grid  $sd(LD50) \approx 0.1$ , Normal  $sd(LD50) \approx .75$ !

# Exmple: Importance sampling in Bioassay

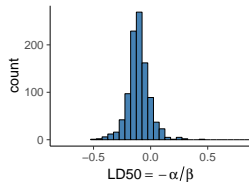
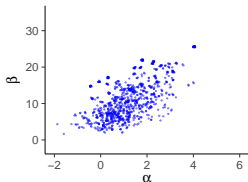
Grid



Normal



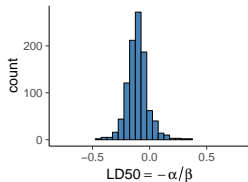
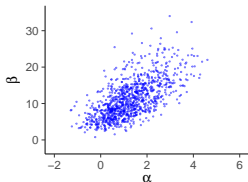
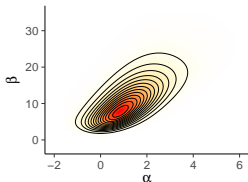
SIR



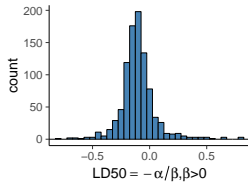
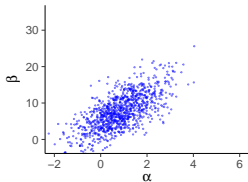
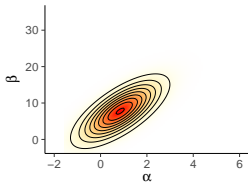


# Exmple: Importance sampling in Bioassay

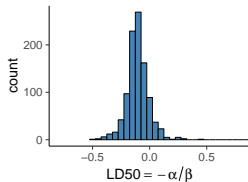
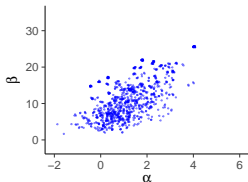
Grid



Normal



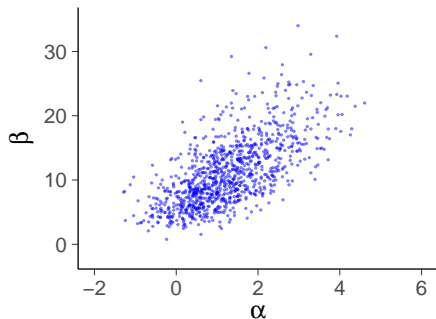
SIR



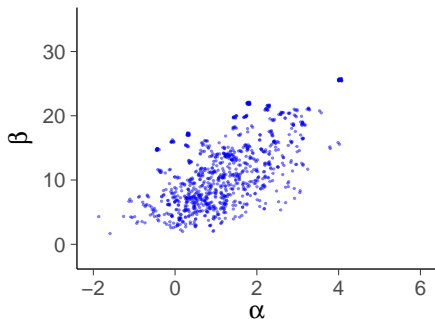
Grid  $\text{sd}(\text{LD50}) \approx 0.1$ , SIR  $\text{sd}(\text{LD50}) \approx 0.1$

## Exmple: Importance sampling in Bioassay

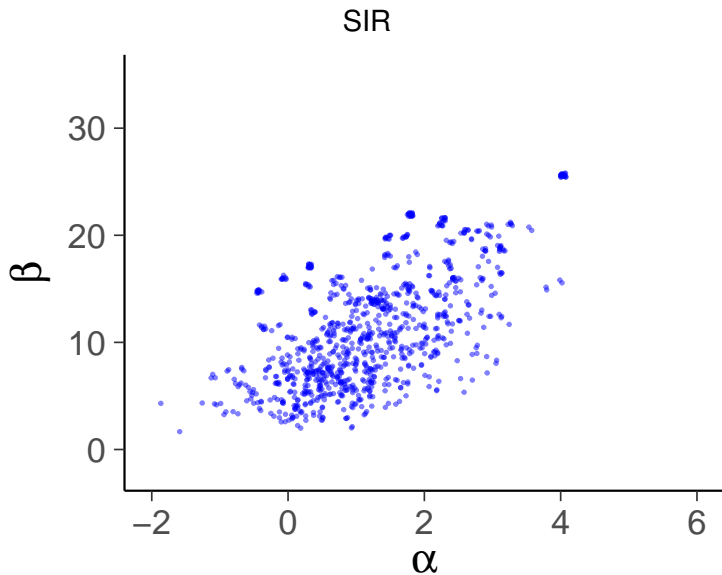
Grid



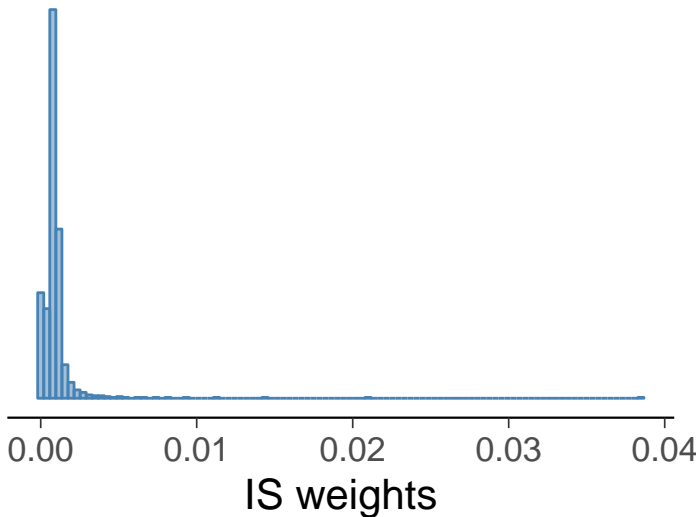
SIR



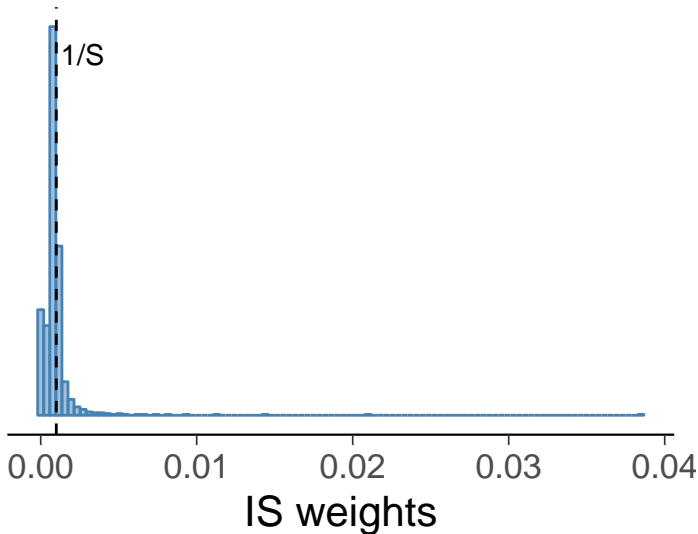
# Exmple: Importance sampling in Bioassay



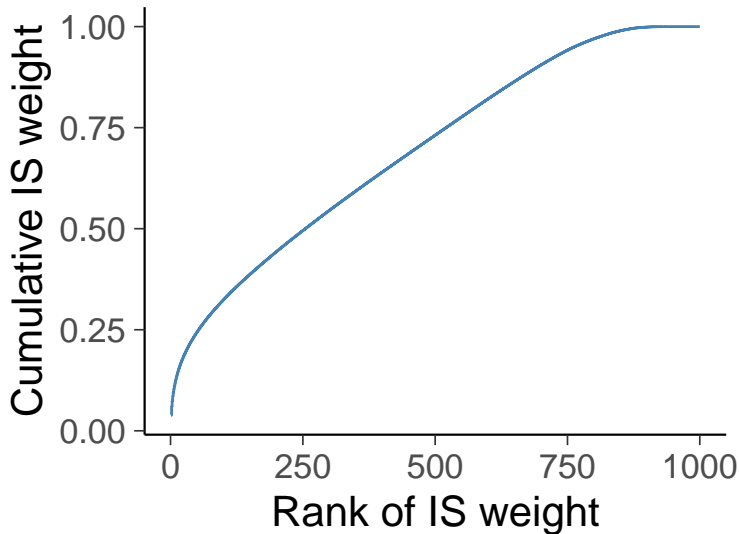
## Exmple: Importance sampling in Bioassay



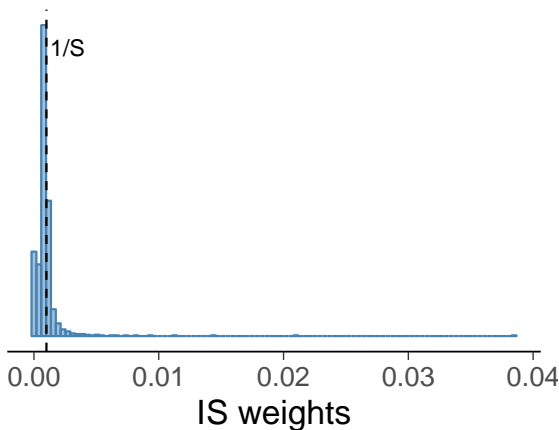
## Exmple: Importance sampling in Bioassay



## Exmple: Importance sampling in Bioassay

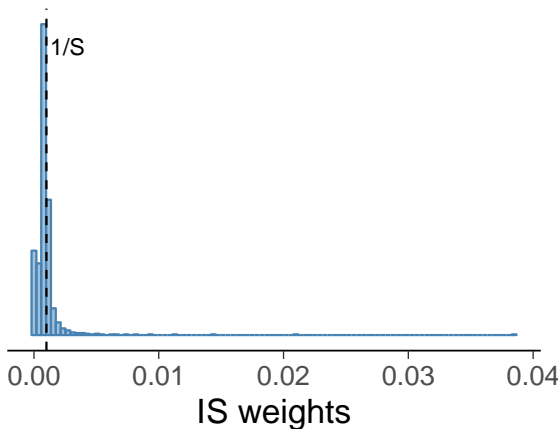


## Exmple: Importance sampling in Bioassay



$$S_{\text{eff}} = \frac{1}{\sum_{s=1}^S (\tilde{w}(\theta^s))^2}, \quad \text{where } \tilde{w}(\theta^s) = w(\theta^s) / \sum_{s'=1}^S w(\theta^{s'})$$

## Exmple: Importance sampling in Bioassay

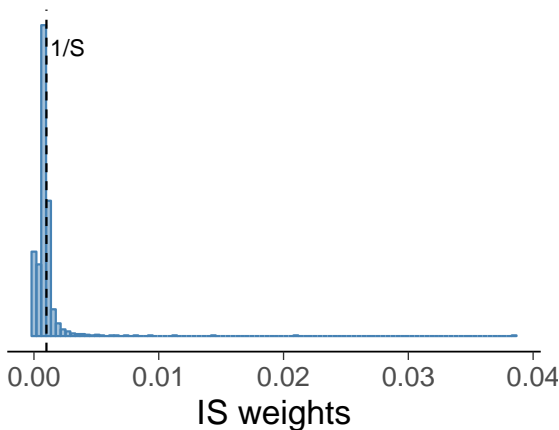


$$S_{\text{eff}} = \frac{1}{\sum_{s=1}^S (\tilde{w}(\theta^s))^2}, \quad \text{where } \tilde{w}(\theta^s) = w(\theta^s) / \sum_{s'=1}^S w(\theta^{s'})$$

BDA3 1st (2013) and 2nd (2014) printing have an error for  $\tilde{w}(\theta^s)$ .  
The normalized weights equation should not have the multiplier S  
(the normalized weights should sum to one). Errata for the book  
[http://www.stat.columbia.edu/~gelman/book/errata\\_bda3.txt](http://www.stat.columbia.edu/~gelman/book/errata_bda3.txt)

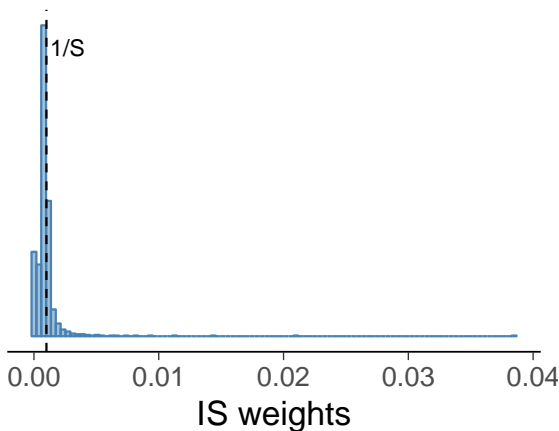


## Exmple: Importance sampling in Bioassay



$$S_{\text{eff}} = \frac{1}{\sum_{s=1}^S (\tilde{w}(\theta^s))^2}, \quad \text{where } \tilde{w}(\theta^s) = w(\theta^s) / \sum_{s'=1}^S w(\theta^{s'})$$
$$S_{\text{eff}} \approx 270$$

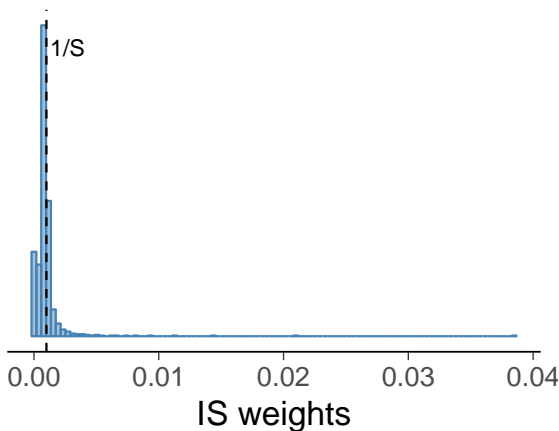
## Exmple: Importance sampling in Bioassay



$$S_{\text{eff}} = \frac{1}{\sum_{s=1}^S (\tilde{w}(\theta^s))^2}, \quad \text{is based on variance of } \tilde{w}(\theta^s)$$

$$S_{\text{eff}} \approx 270$$

## Exmple: Importance sampling in Bioassay

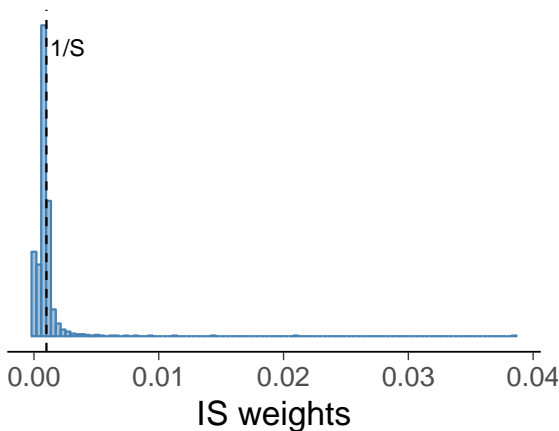


$$S_{\text{eff}} = \frac{1}{\sum_{s=1}^S (\tilde{w}(\theta^s))^2}, \quad \text{is based on variance of } \tilde{w}(\theta^s)$$

$$S_{\text{eff}} \approx 270$$

Pareto- $k$  diagnostic preferably  $< 0.7$ :

## Exmple: Importance sampling in Bioassay



$$S_{\text{eff}} = \frac{1}{\sum_{s=1}^S (\tilde{w}(\theta^s))^2}, \quad \text{is based on variance of } \tilde{w}(\theta^s)$$

$$S_{\text{eff}} \approx 270$$

Pareto- $k$  diagnostic preferably  $< 0.7$ :  $\hat{k} \approx 0.76$

## Pareto smoothed importance sampling

- Pareto- $k$  diagnostic estimate the number of existing moments ( $\lfloor 1/k \rfloor$ )
- Finite variance and central limit theorem for  $k < 1/2$
- Finite mean and generalized central limit theorem for  $k < 1$ , but pre-asymptotic constant grows impractically large for  $k > 0.7$
- See Vehtari, Gelman and Gabry (2017). Pareto smoothed importance sampling. arXiv preprint arXiv:1507.02646, <https://arxiv.org/abs/1507.02646> for improved diagnostics and stability.

# Importance sampling leave-one-out cross-validation

- Later in the course you will learn how  $p(\theta|y)$  can be used as a proposal distribution for  $p(\theta|y_{-i})$ 
  - which allows fast computation of leave-one-out cross-validation

$$p(y_i|y_{-i}) = \int p(y_i|\theta)p(\theta|y_{-i})d\theta$$

# Curse of dimensionality

- Number of grid points increases exponentially
- Concentration of the measure, ie, where is the most of the mass?

# Markov chain Monte Carlo (MCMC)

- Pros
  - Markov chain goes where most of the posterior mass is
  - Certain MCMC methods scale well to high dimensions
- Cons
  - Draws are dependent (affects how many draws are needed)
  - Convergence in practical time is not guaranteed
- MCMC methods in this course
  - Gibbs: “iterative conditional sampling”
  - Metropolis: “random walk in joint distribution”
  - Dynamic Hamiltonian Monte Carlo: “state-of-the-art” used in Stan