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
- 10.5 How many simulation draws are needed?
- 10.6 Software (can be skipped)
- 10.7 Debugging (can be skipped)

Target distribution: distribuzione a posteriori

- Distribuzione a posteriori di θ dato y

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

- Distribuzione predittiva a posteriori $p(\tilde{y}|y)$

Target distribution: distribuzione a posteriori

- Distribuzione a posteriori di θ dato y

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

- Distribuzione predittiva a posteriori $p(\tilde{y}|y)$
- Finora abbiamo considerato esempi in cui queste possono essere calcolate analiticamente, con simulazioni effettuate direttamente usando routines di distr standard o calcoli numerici su griglie

Target distribution: distribuzione a posteriori

- Distribuzione a posteriori di θ dato y

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

- Distribuzione predittiva a posteriori $p(\tilde{y}|y)$
- Finora abbiamo considerato esempi in cui queste possono essere calcolate analiticamente, con simulazioni effettuate direttamente usando routines di distr standard o calcoli numerici su griglie
- Modelli più complicati richiedono algoritmi più elaborati per approssimare la distribuzione a posteriori

Target distribution: distribuzione a posteriori

- Distribuzione a posteriori di θ dato y

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

- Distribuzione predittiva a posteriori $p(\tilde{y}|y)$
- Finora abbiamo considerato esempi in cui queste possono essere calcolate analiticamente, con simulazioni effettuate direttamente usando routines di distr standard o calcoli numerici su griglie
- Modelli più complicati richiedono algoritmi più elaborati per approssimare la distribuzione a posteriori
- In questa prima parte consideriamo delle procedure per calcolare approssimativamente integrali

Notation

- In this chapter, generic $p(\theta)$ is used instead of $p(\theta|y)$ when there is no ambiguity
- Unnormalized distribution is denoted by $q(\cdot)$
  - $\int q(\theta)d\theta \neq 1$, but finite
  - $q(\cdot) \propto p(\cdot)$
- Proposal distribution is denoted by $g(\cdot)$

Numerical accuracy of computer arithmetic

- Many models use continuous real valued parameters.
- Computers have finite memory and thus the continuous values are also presented with finite number of bits and thus with finite accuracy.
- Most commonly used presentations are floating-point presentations that try to have balanced accuracy over the range of values where it mostly matters.
- As the presentation has finite accuracy there are limitations, eg, with IEC 60559 floating-point (double precision) arithmetic used in current R
  - the smallest positive floating-point number x such that $1 + x \neq 1$ is $2.220446 \cdot 10^{-16}$
  - the smallest non-zero normalized floating-point number is $2.225074 \cdot 10^{-308}$
  - the largest normalized floating-point number $1.797693 \cdot 10^{308}$
  - the largest integer which can be represented is $2^{31} - 1 = 2147483647$
  - see more at https://stat.ethz.ch/R-manual/R-devel/library/base/html/zMachine.html

Numerical accuracy – floating point

- Floating point presentation of numbers. e.g. with 64bits
  - closest value to zero is $\approx 2.2 \cdot 10^{-308}$
    - generate sample of 600 from normal distribution: qr=rnorm(600)
    - calculate joint density given normal: prod(dnorm(qr)) $\rightarrow$ 0 (underflow)

Numerical accuracy – floating point

- Floating point presentation of numbers. e.g. with 64bits
  - closest value to zero is $\approx 2.2 \cdot 10^{-308}$
    - generate sample of 600 from normal distribution: qr=rnorm(600)
    - calculate joint density given normal: prod(dnorm(qr)) $\rightarrow$ 0 (underflow)
    - see log densities in the next slide

Numerical accuracy – floating point

- Floating point presentation of numbers. e.g. with 64bits
  - closest value to zero is $\approx 2.2 \cdot 10^{-308}$
    - generate sample of 600 from normal distribution: qr=rnorm(600)
    - calculate joint density given normal: prod(dnorm(qr)) $\rightarrow$ 0 (underflow)
    - see log densities in the next slide
  - closest value to 1 is $\approx 1 \pm 2.2 \cdot 10^{-16}$
    - Laplace and ratio of girl and boy babies
    - pbeta(0.5, 241945, 251527) $\rightarrow$ 1 (rounding)

Numerical accuracy – floating point

- Floating point presentation of numbers. e.g. with 64bits
  - closest value to zero is $\approx 2.2 \cdot 10^{-308}$
    - generate sample of 600 from normal distribution: qr=rnorm(600)
    - calculate joint density given normal: prod(dnorm(qr)) $\rightarrow$ 0 (underflow)
    - see log densities in the next slide
  - closest value to 1 is $\approx 1 \pm 2.2 \cdot 10^{-16}$
    - Laplace and ratio of girl and boy babies
    - pbeta(0.5, 241945, 251527) $\rightarrow$ 1 (rounding)
    - pbeta(0.5, 241945, 251527, lower.tail=FALSE) $\approx -1.2 \cdot 10^{-42}$ there is more accuracy near 0

Numerical accuracy – floating point

- Floating point presentation of numbers. e.g. with 64bits
  - closest value to zero is $\approx 2.2 \cdot 10^{-308}$
    - generate sample of 600 from normal distribution: qr=rnorm(600)
    - calculate joint density given normal: prod(dnorm(qr)) $\rightarrow$ 0 (underflow)
    - see log densities in the next slide
  - closest value to 1 is $\approx 1 \pm 2.2 \cdot 10^{-16}$
    - Laplace and ratio of girl and boy babies
    - pbeta(0.5, 241945, 251527) $\rightarrow$ 1 (rounding)
    - pbeta(0.5, 241945, 251527, lower.tail=FALSE) $\approx -1.2 \cdot 10^{-42}$ there is more accuracy near 0
  - the largest normalized floating-point number $1.797693 \cdot 10^{308} \rightarrow$ above (in absolute value) Inf (overflow)
  - the largest integer which can be represented is $2^{31} - 1 = 2147483647$

Numerical accuracy – log scale

- Log densities
  - use log densities to avoid over- and underflows in floating point presentation
    - prod(dnorm(qr)) $\rightarrow$ 0 (underflow)
    - sum(dnorm(qr,log=TRUE)) $\rightarrow$ -847.3

Numerical accuracy – log scale

- Log densities
  - use log densities to avoid over- and underflows in floating point presentation
    - prod(dnorm(qr)) $\rightarrow$ 0 (underflow)
    - sum(dnorm(qr,log=TRUE)) $\rightarrow$ -847.3
    - how many observations we can now handle?

Numerical accuracy – log scale

- Log densities
  - use log densities to avoid over- and underflows in floating point presentation
    - prod(dnorm(qr)) $\rightarrow$ 0 (underflow)
    - sum(dnorm(qr,log=TRUE)) $\rightarrow$ -847.3
    - how many observations we can now handle?
  - compute exp as late as possible

Numerical accuracy – log scale

- Log densities
  - use log densities to avoid over- and underflows in floating point presentation
    - prod(dnorm(qr)) $\rightarrow$ 0 (underflow)
    - 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
      $\log(\exp(a) + \exp(b)) = a + \log(1 + \exp(b - a))$

Numerical accuracy – log scale

- Log densities
  - use log densities to avoid over- and underflows in floating point presentation
    - prod(dnorm(qr)) $\rightarrow$ 0 (underflow)
    - 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. $\log(\exp(800) + \exp(800)) \rightarrow$ Inf

Numerical accuracy – log scale

- Log densities
  - use log densities to avoid over- and underflows in floating point presentation
    - prod(dnorm(qr)) $\rightarrow$ 0 (underflow)
    - 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. $\log(\exp(800) + \exp(800)) \rightarrow$ Inf
      but $800 + \log(1 + \exp(800 - 800)) \approx 800.69$

Numerical accuracy – log scale

- Log densities
  - use log densities to avoid over- and underflows in floating point presentation
    - prod(dnorm(qr)) $\rightarrow$ 0 (underflow)
    - 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. $\log(\exp(800) + \exp(800)) \rightarrow$ Inf
      but $800 + \log(1 + \exp(800 - 800)) \approx 800.69$
    - e.g. in Metropolis-algorithm (later) compute the log of ratio of densities using the identity
      $\log(a/b) = \log(a) - \log(b)$

It's all about expectations

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

$$\text{where} \quad p(\theta|y) = \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta}$$

Conversely, we can express any integral over the space of θ as a posterior expectation by defining $f(\theta)$ appropriately

It's all about expectations

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

$$\text{where} \quad p(\theta|y) = \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta}$$

We can easily evaluate $p(y|\theta)p(\theta)$ for any θ, but the integral $\int p(y|\theta)p(\theta)d\theta$ is usually difficult.

It's all about expectations

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

$$\text{where} \quad p(\theta|y) = \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta}$$

We can easily evaluate $p(y|\theta)p(\theta)$ for any θ, but the integral $\int p(y|\theta)p(\theta)d\theta$ is usually difficult.

We can use the unnormalized posterior $q(\theta|y) = p(y|\theta)p(\theta)$, for example, in

It's all about expectations

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

$$\text{where} \quad p(\theta|y) = \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta}$$

We can easily evaluate $p(y|\theta)p(\theta)$ for any θ, but the integral $\int p(y|\theta)p(\theta)d\theta$ is usually difficult.

We can use the unnormalized posterior $q(\theta|y) = p(y|\theta)p(\theta)$, for example, in

- Grid (equal spacing) evaluation with self-normalization

$$E_{p(\theta|y)}[f(\theta)] \approx \frac{\sum_{s=1}^{S}\left[f(\theta^{(s)})q(\theta^{(s)}|y)\right]}{\sum_{s=1}^{S} q(\theta^{(s)}|y)}$$

It's all about expectations

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

$$\text{where} \quad p(\theta|y) = \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta}$$

We can easily evaluate $p(y|\theta)p(\theta)$ for any θ, but the integral $\int p(y|\theta)p(\theta)d\theta$ is usually difficult.

We can use the unnormalized posterior $q(\theta|y) = p(y|\theta)p(\theta)$, for example, in

- Grid (equal spacing) evaluation with self-normalization

$$E_{p(\theta|y)}[f(\theta)] \approx \frac{\sum_{s=1}^{S}\left[f(\theta^{(s)})q(\theta^{(s)}|y)\right]}{\sum_{s=1}^{S} q(\theta^{(s)}|y)}$$

- Monte Carlo methods which can sample from $p(\theta^{(s)}|y)$ using only $q(\theta^{(s)}|y)$

$$E_{p(\theta|y)}[f(\theta)] \approx \frac{1}{S}\sum_{s=1}^{S} f(\theta^{(s)})$$

It's all about expectations

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

- Conjugate priors and analytic solutions (Ch 1-5)
- Grid integration and other quadrature rules (Ch 3, 10)
- Independent Monte Carlo, rejection and importance sampling (Ch 10)
- Markov Chain Monte Carlo (Ch 11-12)
- Distributional approximations (Laplace, VB, EP) (Ch 4, 13)

Numerical integration

Numerical integration refers to methods in which the integral over continuous function is evaluated by computing the value of the function at finite number of points.

- By increasing the number of points where the function is evaluated, desired accuracy can be obtained.
- Numerical integration methods can be divided into
  - Deterministic methods such as many quadrature rule methods (e.g. grid)
    - evaluation points are selected by some deterministic rule
    - good deterministic methods converge faster (need less function evaluations)
  - Simulation (stochastic) methods, such as Monte Carlo
    - evaluation points are selected stochastically (randomly)

Note: Sometimes 'quadrature' is used to refer generically to any numerical integration method (including Monte Carlo), sometimes it is used to refer just to deterministic numerical integration methods.

Metodi (stocastici) di simulazione I

- Metodi Monte Carlo o metodi di simulazione
- Simulate (obtain random) draws from the target distribution
  - these draws can be treated as any observations
  - a collection of draws is a sample
- Use these draws, to estimate the expectation of any function $f(\theta|y)$

$$E[f(\theta)|y] = \int f(\theta)p(\theta|y)d\theta \approx \frac{1}{S}\sum_{s=1}^{S} f(\theta^{(s)})$$

  for example,
  - to compute means, deviations, quantiles
  - to marginalize
  - to draw histograms, etc.
- The estimate is stochastic depending on generated random numbers,

Metodi (stocastici) di simulazione II

- the accuracy can be improved by obtaining more samples.
- Simulation methods can be used for high-dimensional distributions
- Basic Monte Carlo (MC) methods produce independent samples
- Markov chain Monte Carlo (MCMC) methods produce dependent samples but can better adapt to high-dimensional complex distributions
- MCMC methods have been important in making Bayesian inference practical for generic hierarchical models.

Metodi deterministici I

Deterministic numerical integration methods evaluate the integrand $f(\theta)p(\theta|y)$ at selected points $\theta^{(s)}$ and are based on a weighted version of the above MC estimate

$$E[f(\theta)|y] = \int f(\theta)p(\theta|y)d\theta \approx \frac{1}{S}\sum_{s=1}^{S} w_s f(\theta^{(s)})p(\theta^{(s)}|y)$$

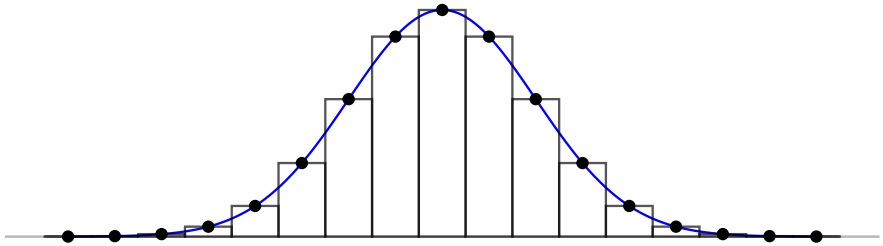
with weight w_s corresponding to the volume of space represented by the point $\theta^{(s)}$.

- Deterministic rules typically have lower variance than simulation methods, but selection of locations gets difficult in high dimensions.
- They are also known as 'grid' or 'quadrature' methods

Quadrature integration

- The simplest quadrature integration is grid integration

$$\mathrm{E}[\theta] \approx \sum_{t=1}^{T} \theta^{(t)} w^{(t)},$$

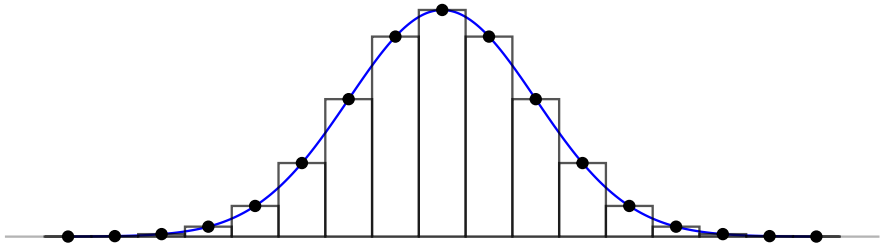


where $w^{(t)}$ is the normalized probability of a grid cell t, and $\theta^{(t)}$ is the center location of the grid cell

Quadrature integration

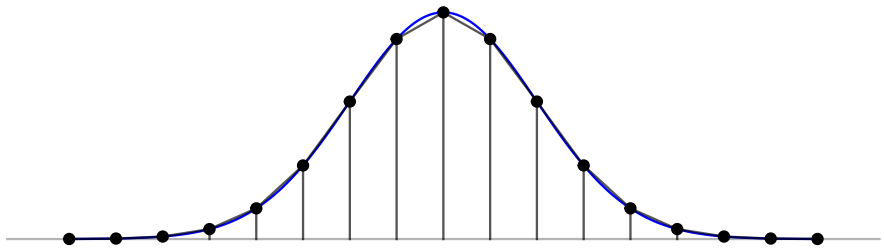
- The simplest quadrature integration is grid integration

$$\mathrm{E}[\theta] \approx \sum_{t=1}^{T} \theta^{(t)} w^{(t)},$$



where $w^{(t)}$ is the normalized probability of a grid cell t, and $\theta^{(t)}$ is the center location of the grid cell

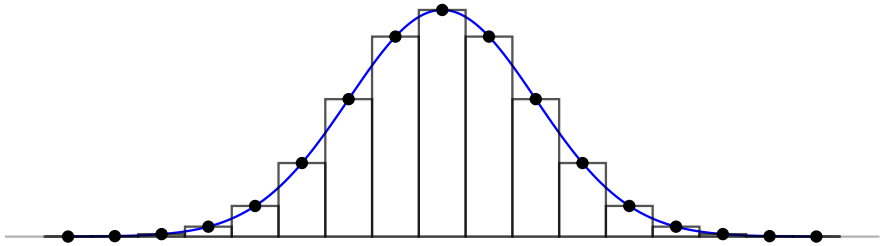
- In 1D further variations with better accuracy, e.g. trapezoid



Quadrature integration

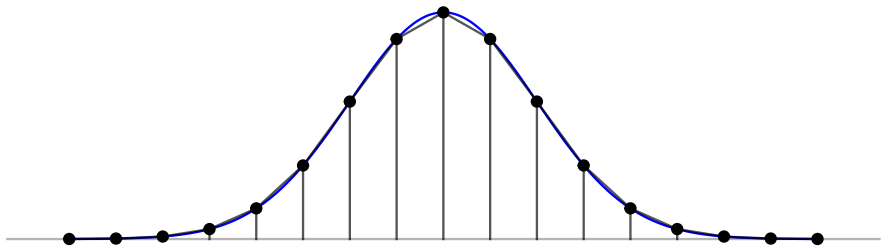
- The simplest quadrature integration is grid integration

$$\mathrm{E}[\theta] \approx \sum_{t=1}^{T} \theta^{(t)} w^{(t)},$$



where $w^{(t)}$ is the normalized probability of a grid cell t, and $\theta^{(t)}$ is the center location of the grid cell

- In 1D further variations with better accuracy, e.g. trapezoid

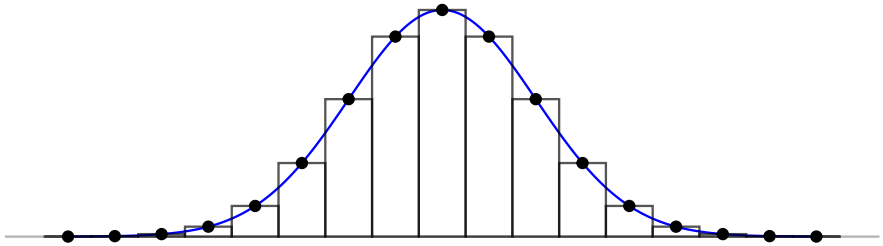


- Adaptive quadrature methods add evaluation points where needed, e.g., R function `integrate()`

Quadrature integration

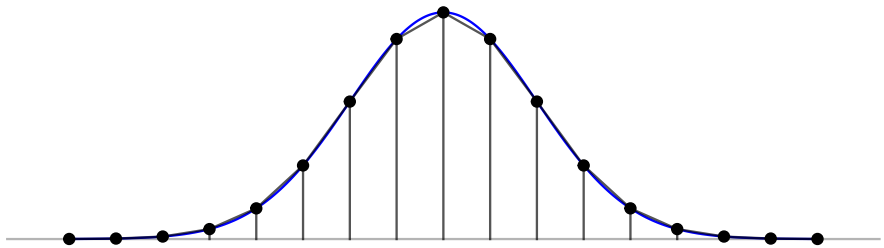
- The simplest quadrature integration is grid integration

$$\mathrm{E}[\theta] \approx \sum_{t=1}^{T} \theta^{(t)} w^{(t)},$$



where $w^{(t)}$ is the normalized probability of a grid cell t, and $\theta^{(t)}$ is the center location of the grid cell

- In 1D further variations with better accuracy, e.g. trapezoid



- Adaptive quadrature methods add evaluation points where needed, e.g., R function `integrate()`
- In 2D and higher
  - nested quadrature
  - product rules

Approssimazioni distributive

(Accenno)

- Distributional (analytic) approximations approximate the posterior with some simpler parametric distribution, from which integrals can be computed directly or by using the approximation as a starting point for simulation-based methods.
- Normal approximation in Chapter 4 and more advanced approximation methods in Ch. 13
- Crude estimation by ignoring some information
  - Before developing approximations or methods for sampling from the target distribution, it is often useful to obtain a rough estimate of the location of the target distribution—that is, a point estimate of the parameters in the model—using some simple technique
  - eg rat tumor ex ((α, β))

Direct simulation

- In simple nonhierarchical Bayesian models, it is often easy to draw from the posterior distribution directly, especially if conjugate prior distributions have been assumed.
- For more complicated problems, it can help to factor the distribution analytically and simulate it in parts, first sampling from the marginal posterior distribution of the hyperparameters, then drawing the other parameters conditional on the data and the simulated hyperparameters.
- It is sometimes possible to perform direct simulations and analytic integrations for parts of the larger problem (eg in HBMs)
- See next slide 'Direct approximation by calculating at a grid of points'

Direct approximation by calculating at a grid of points

- Compute the target density, $q(\theta|y)$, at a set of evenly spaced values $\theta_1, \ldots, \theta_N$ that cover a broad range of the parameter space for θ
- approximate the continuous $q(\theta|y)$ by the discrete density at $\theta_1, \ldots, \theta_N$, with probabilities $q(\theta_i|y)/(\sum_{j=1}^{N} q(\theta_j|y))$
- Note that is equivalent to work with a normalized or unnormalized density
- Once the grid of density values is computed, draw $u \sim U[0, 1]$, then transform by the inverse cdf method to obtain a draw from the discrete approximation

Direct simulation

- Produces independent draws
  - Using analytic transformations of uniform random numbers (e.g. appendix A)
  - factorization
  - numerical inverse-CDF
- Problem: restricted to limited set of models
- The discrete approximation is more difficult to use in higher-dimensional multivariate problems, where computing at every point in a dense multidimensional grid becomes prohibitively expensive

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 $\mathrm{U}(0,1)$, 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 $\mathrm{N}(0,1)$

Direct simulation: Example

- Box-Muller -method:
  If U_1 and U_2 are independent draws from distribution $\mathrm{U}(0,1)$, 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 $\mathrm{N}(0,1)$
  - 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
  - $\rightarrow$ $50^{10} \approx$ 1e17 grid points
  - $\rightarrow$ $1000^{10} \approx$ 1e30 grid points
- R and my current laptop can compute density of normal distribution about 20 million times per second
  - $\rightarrow$ evaluation in 1e17 grid points would take 150 years
  - $\rightarrow$ evaluation in 1e30 grid points would take 1 500 billion years

Indirect sampling

- Rejection sampling
  - draw directly from a proposal distribution, reject some draws, remaining draws are independent draws from the target distribution

Indirect sampling

- Rejection sampling
  - draw directly from a proposal distribution, reject some draws, remaining draws are independent draws from the target distribution
- Importance sampling
  - draw directly from a proposal distribution, weight the draws

Indirect sampling

- Rejection sampling
  - draw directly from a proposal distribution, reject some draws, remaining draws are independent draws from the target distribution
- Importance sampling
  - draw directly from a proposal distribution, weight the draws
- Markov chain Monte Carlo (next)
  - draw directly from a transition distribution forming a Markov chain, draws are dependent draws from the target distribution

Rejection sampling

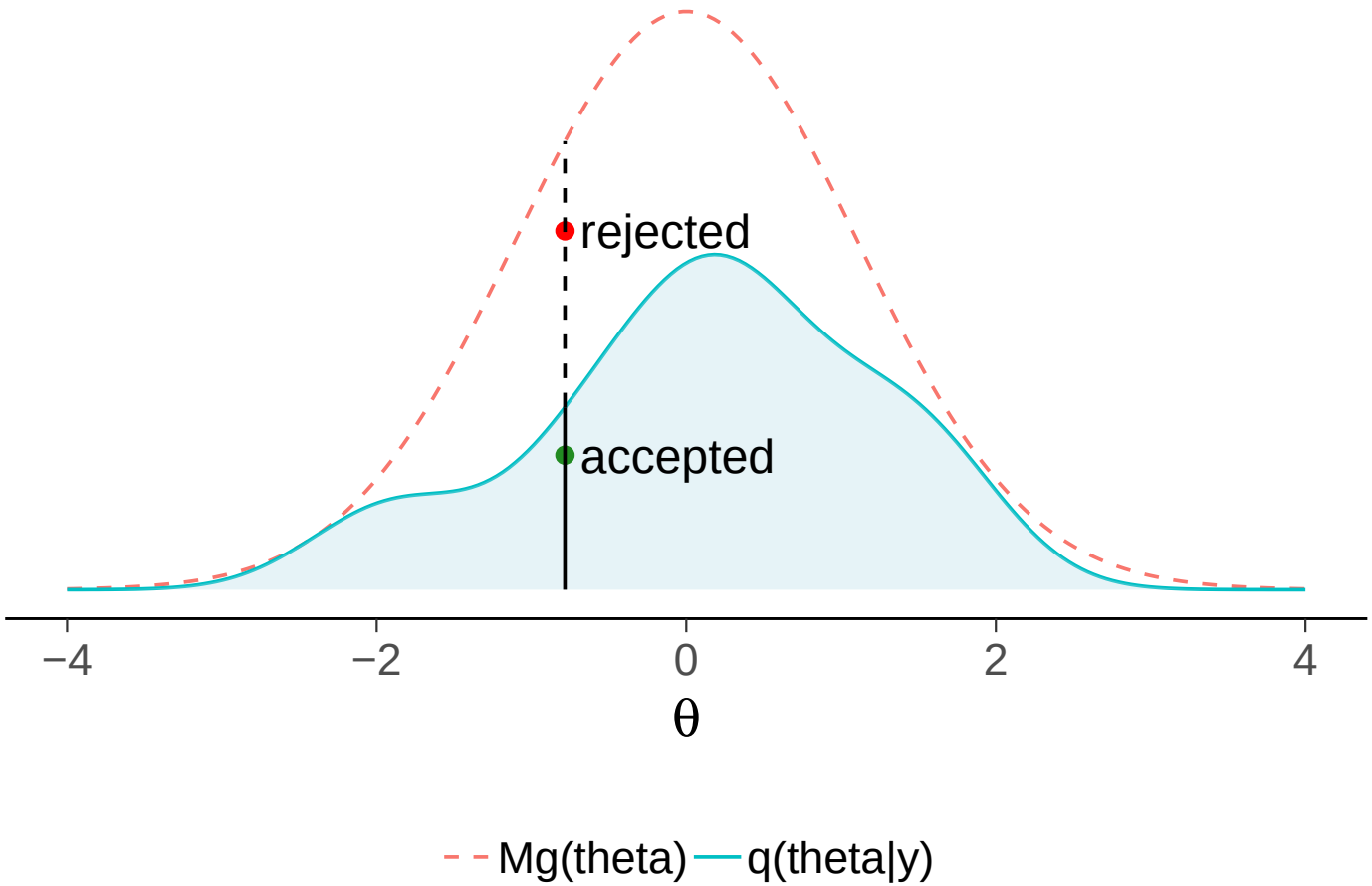
- We can work with the target distribution p as well as with the unnormalized form q instead.
- Let $g(\theta)$ a positive function defined $\forall\theta$ for which $p(\theta|y) > 0$ such that:
  - We can draw from the probability density proportional to g. $g(\theta)$ must have a finite integral (it is not required that integrates to 1)
  - $\exists\, M$: the importance ratio $q(\theta|y)/g(\theta) \leq M\, \forall\theta$
- The rejection sampling proceeds as follows:
  1. Draw $\theta \sim g(\theta)$
  2. Accept with probability $q(\theta|y)/Mg(\theta)$
     a. Draw $u \sim U[0,1]$
     b. If $u < q(\theta|y)/Mg(\theta)$ accept θ otherwise reject it

     If the drawn is rejected, return to step 1.

Nota: The boundedness condition is necessary so that the probability in step 2 is not greater than 1.

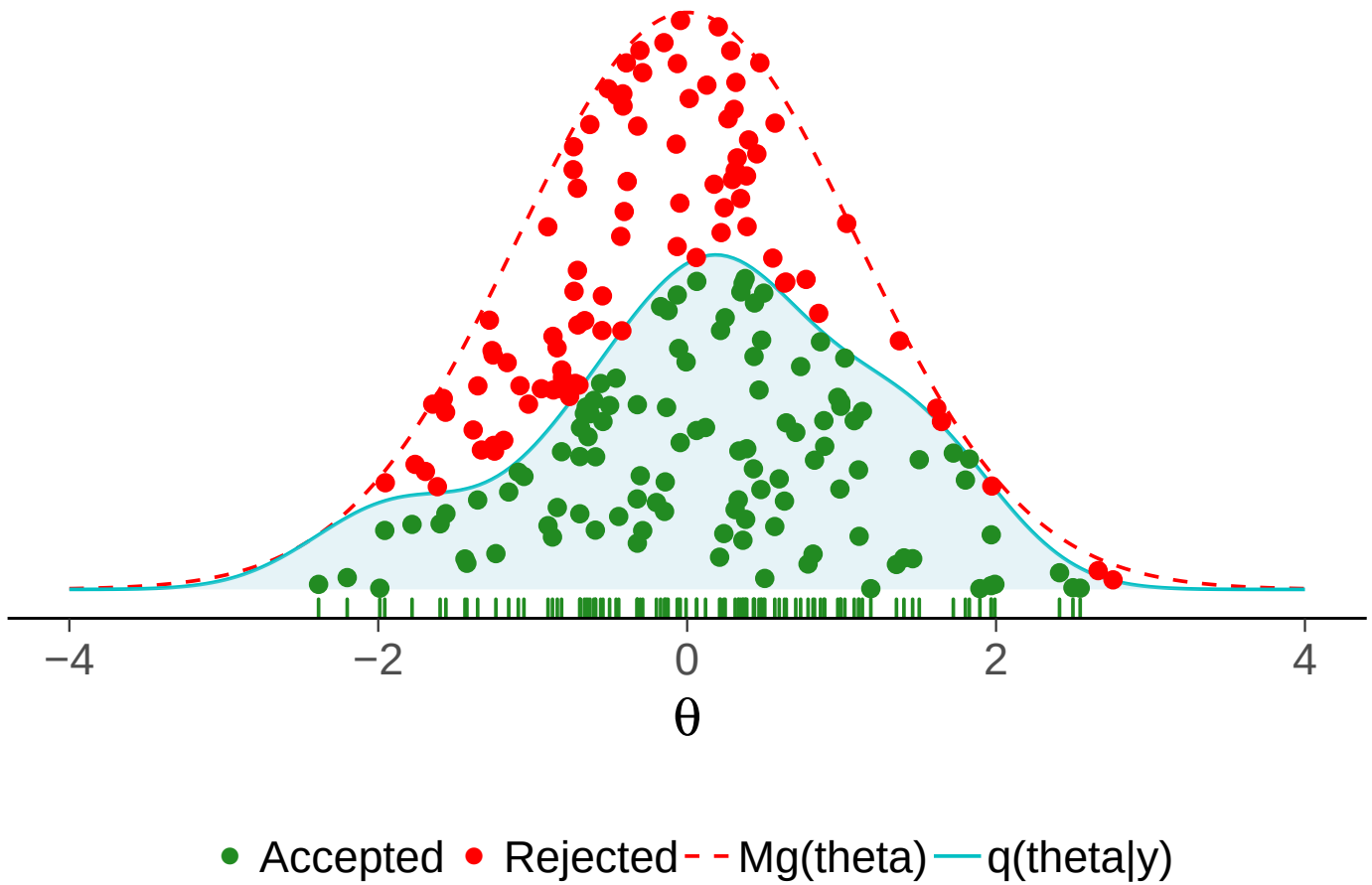
Rejection sampling

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



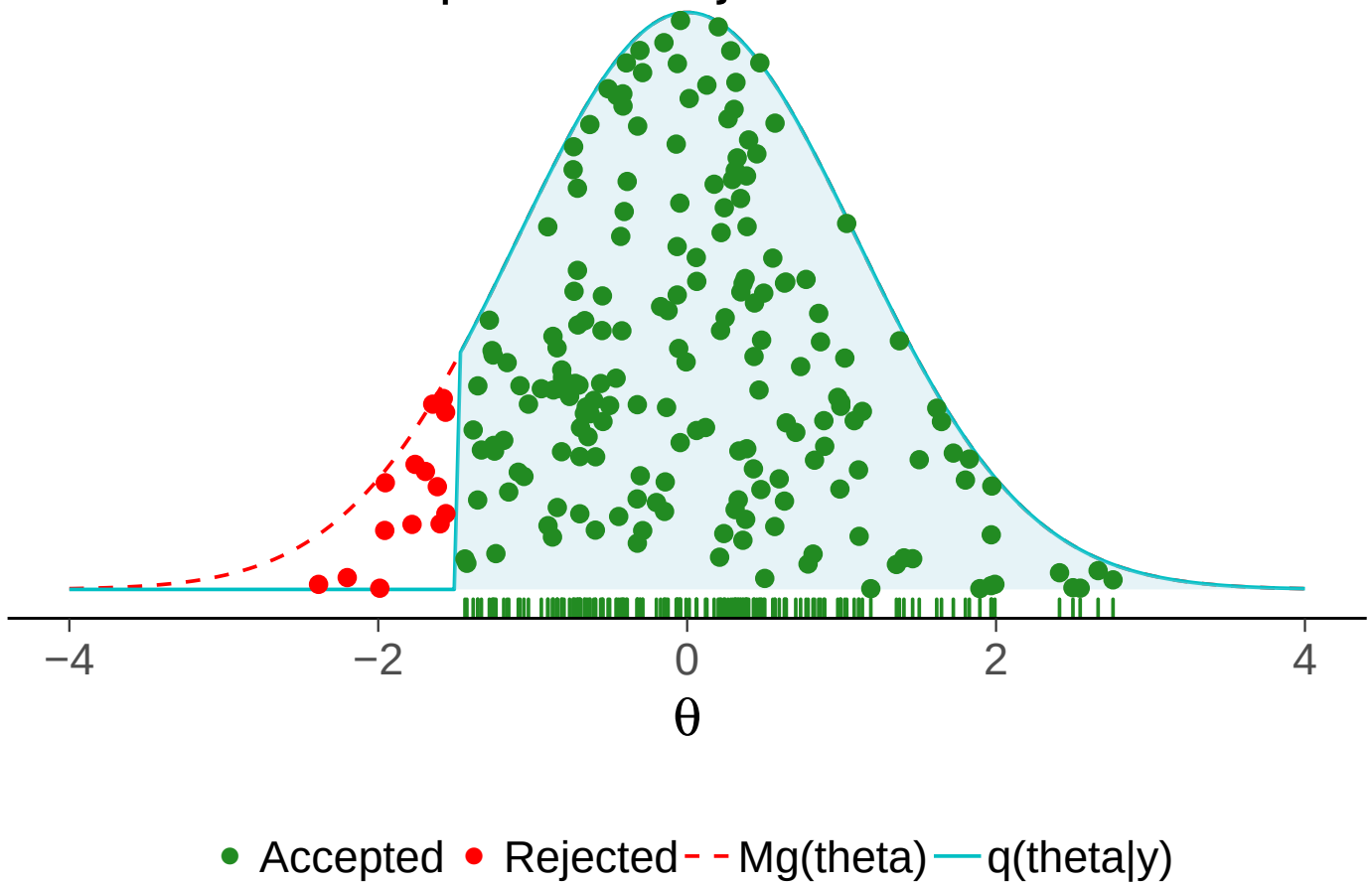
Rejection sampling

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



Rejection sampling

- Proposal forms envelope over the target distribution $q(\theta|y)/Mg(\theta) \leq 1$
- Draw from the proposal and accept with probability $q(\theta|y)/Mg(\theta)$
- Common for truncated distributions, in which case all draws from the truncated part are rejected.



Rejection sampling

- A good approximate density $g(\theta)$ should be roughly proportional to $p(\theta|y)$
  - If $g \propto p$, (with a suitable value of M) we can accept every draw with probability 1
- When g is not nearly proportional to p, M must be set so large that almost all draws will be rejected.
- Rejection sampling is self-monitoring—if the method is not working efficiently, few simulated draws will be accepted.
- 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

Importance sampling (and numerical integration

- Aim: to estimate $E(f(\theta)|y) = \int f(\theta)p(\theta|y)d\theta$,
- Problem: we cannot generate random draws from $p(\theta|y)$ (a closed form is not available)
- Let $g(\theta)$ be a normalized density from which we can generate random draws, then we can write,

$$E(f(\theta)|y) = \int f(\theta)\frac{p(\theta|y)}{g(\theta)}g(\theta)d\theta$$
$$= c^{-1}\int f(\theta)w(\theta)g(\theta)d\theta$$

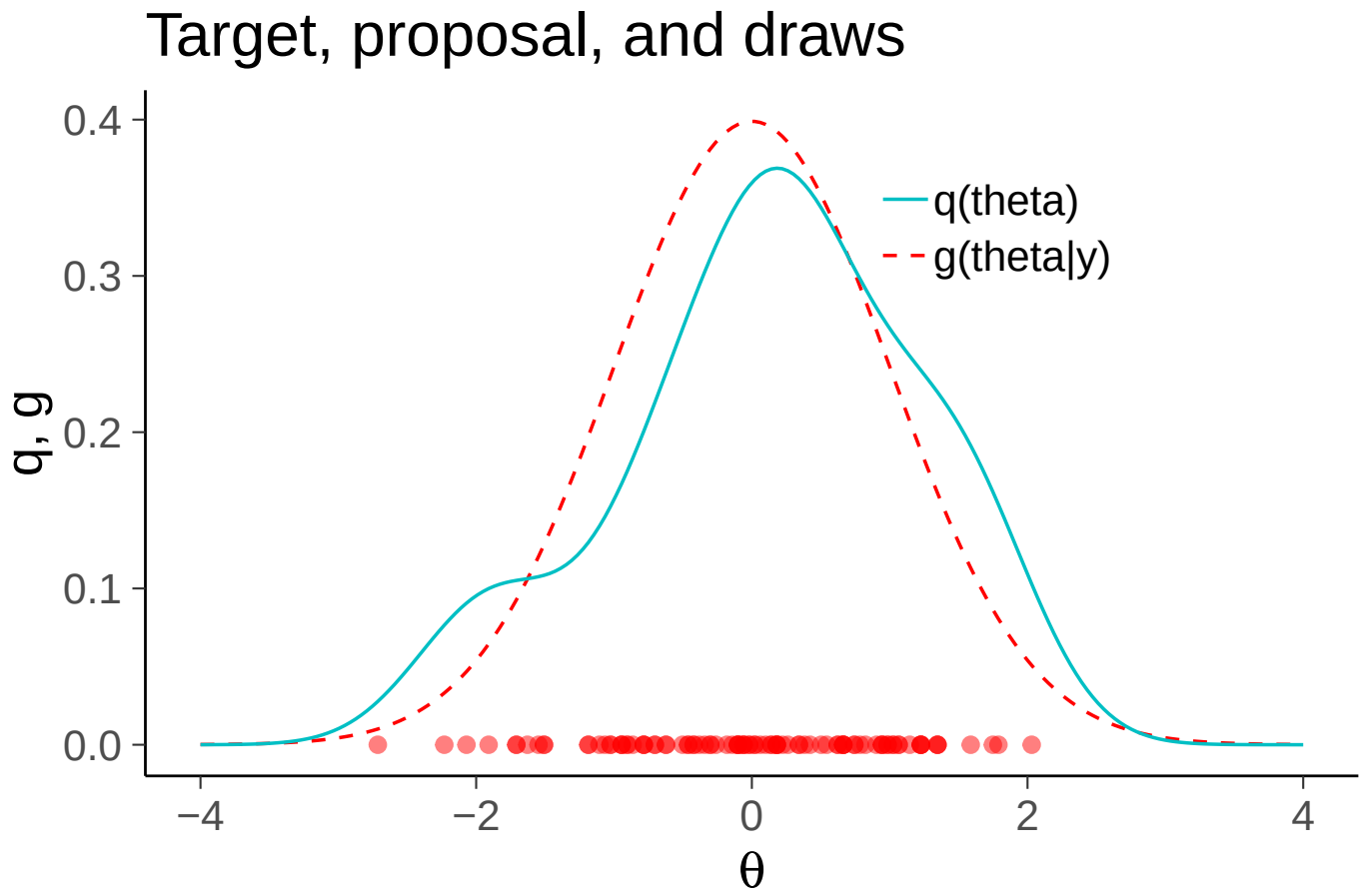
  where $w(\theta) = \frac{q(\theta|y)}{g(\theta)}$ and $c = \int q(\theta|y)d\theta$
- Draw $\theta^1, \ldots, \theta^S$ from $g(\theta)$
- Estimate $E(f(\theta)|y)$ by

$$\frac{\sum_s w_s f(\theta^{(s)})}{\sum_s w_s} \quad \text{where} \quad w_s = \frac{q(\theta^{(s)})}{g(\theta^{(s)})}$$

  are called importance ratios or importance weights

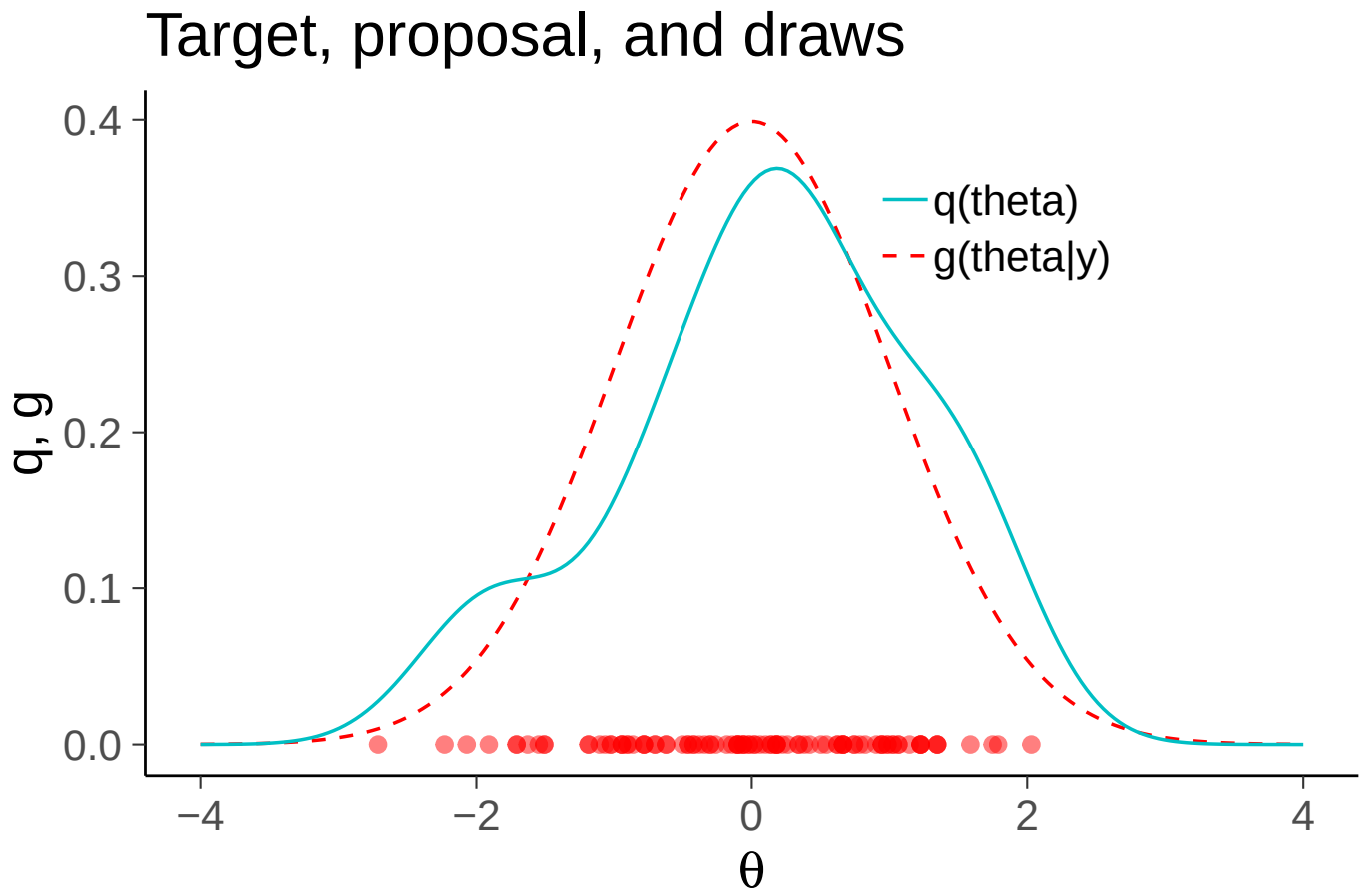
Importance sampling

- Proposal does not need to have a higher value everywhere



Importance sampling

- Proposal does not need to have a higher value everywhere

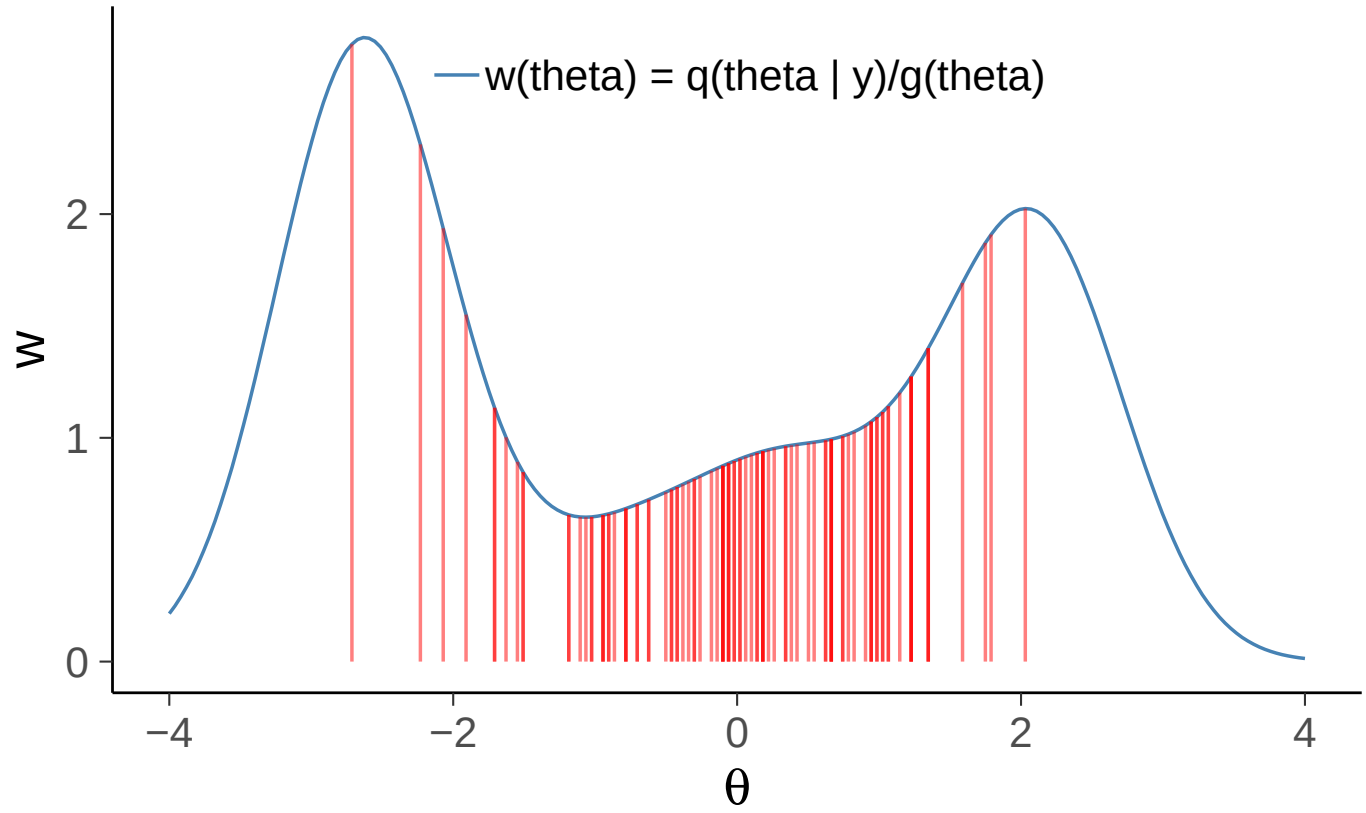


$$\mathrm{E}[f(\theta)] \approx \frac{\sum_s w_s f(\theta^{(s)})}{\sum_s w_s}, \qquad \text{where} \quad w_s = \frac{q(\theta^{(s)})}{g(\theta^{(s)})}$$

Importance sampling

- Proposal does not need to have a higher value everywhere

Draws and importance weights



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

Importance sampling

- Unlike the rejection sampling (RS), the proposal $g(\theta)$ must be normalized
- IS is not a very useful method if the importance ratios vary substantially
- Estimates will be poor if the largest ratios are too large relative to the others
- The worst possible scenario occurs when the importance ratios are small with high probability but with a low probability are huge,
  - eg, if q has wide tails compared to g (It is a bad idea to use a normal for approximating a t_3, viceversa it is a good idea to use a t_3 for approximating a normal
- There is always the possibility that we have missed some extremely large but rare importance weights. (In contrast, we do not have to worry about small importance ratios, because they have little influence)

- The approximating distribution g in importance sampling should cover all the important regions of the target distribution.
- Variation of the weights affect the effective sample size (see eq.10.4 as a rough guide)

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

  that is based on variance of $\tilde{w}(\theta^s)$
  - if single weight dominates, we have effectively one sample
  - if weights are equal, we have effectively S draws
- Resampling using normalized importance weights can be used to pick a smaller number of draws with uniform weights (See importance resampling also called sampling-importance resampling or SIR, p. 266)
- Selection of good proposal gets more difficult when the number of dimensions increase

- Often used to correct distributional approximations (including variational inference in machine learning).

Importance resampling

- Importance weights can be used to obtain a sample that approximates the target distribution by using the SIR method
- $g(\theta)$ can be unnormalized
- If the ratio $q(\theta)/g(\theta)$ is bounded, then we can use rejection sample also
- SIR: Once S draws, $\theta^1, \ldots, \theta^S$, from the approximate distribution g have been sampled, a sample of $k < S$ draws can be simulated as follows.
  1. Sample a value θ from the set $\theta^1, \ldots, \theta^S$, where the probability of sampling each θ^s is proportional to the weight, $w(\theta^s) = q(\theta^s|y)/g(\theta^s)$
  2. Sample a second value using the same procedure, but excluding the already sampled value from the set.
  3. Repeatedly sample without replacement $k - 2$ more times.
- In other words, we sample θ from the discrete didtr over $\theta^1, \ldots, \theta^S$ with probabilities $w(\theta^s)$ (weighted bootstrap)

Nota: We don't recommend 'without replacement' anymore

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

How many simulation draws are needed?

- Bayesian inferences are usually most conveniently summarized by random draws from the posterior distribution of the model parameters.

  Riportiamo usualmente:
  - percentili della distribuzione a posteriori di parametri univariati (quantili 2.5%, 25%, 50%, 75%, e 97.5%, da cui ricaviamo gli intervalli a posteriori 50% and a 95%)
  - scatterplot delle simulazioni, contour plot di funzioni di densità e altri grafici per visualizzare la distribuzione a posteriori in 2D o 3D
- We also use posterior simulations to make inferences about predictive quantities. Given each draw θ^s, we can sample any predictive quantity, $y^s \sim p(\tilde{y}|\theta^s)$ or, for a regression model, $y^s \sim p(\tilde{y}|\tilde{X}, \theta^s)$
- Finally, given each draw θ^s, we can simulate a replicated dataset y^s_{rep}. We can then check the model by comparing the data to these posterior predictive replications.

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?

Our goal in Bayesian computation is to obtain a set of independent draws θ^s, $s = 1, \ldots, S$, from the posterior distribution, with enough draws S so that quantities of interest can be estimated with reasonable accuracy.

How many simulation draws are needed?

- Expectation of unknown quantity θ (with mean μ_θ and sd σ_θ)

$$\mathrm{E}(\theta) \approx \frac{1}{S} \sum_{s=1}^{S} \theta^{(s)}$$

  if S is big and $\theta^{(s)}$ are independent, we may assume that the distribution of the expectation approaches normal distribution (see Ch 4) with variance σ_θ^2/S (asymptotic normality) The posterior mean is then estimated to an accuracy of approximately $\sigma_\theta/\sqrt{S}$.
  - this variance is independent on dimensionality of θ

How many simulation draws are needed?

- Expectation of unknown quantity θ (with mean μ_θ and sd σ_θ)

$$\mathrm{E}(\theta) \approx \frac{1}{S} \sum_{s=1}^{S} \theta^{(s)}$$

if S is big and $\theta^{(s)}$ are independent, we may assume that the distribution of the expectation approaches normal distribution (see Ch 4) with variance σ_θ^2/S (asymptotic normality) The posterior mean is then estimated to an accuracy of approximately $\sigma_\theta/\sqrt{S}$.

  - this variance is independent on dimensionality of θ
  - total variance is sum of the epistemic uncertainty in the posterior and the uncertainty due to using finite number of Monte Carlo draws (Monte Carlo error),

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

How many simulation draws are needed?

- Expectation of unknown quantity θ (with mean μ_θ and sd σ_θ)

$$\mathrm{E}(\theta) \approx \frac{1}{S} \sum_{s=1}^{S} \theta^{(s)}$$

if S is big and $\theta^{(s)}$ are independent, we may assume that the distribution of the expectation approaches normal distribution (see Ch 4) with variance σ_θ^2/S (asymptotic normality) The posterior mean is then estimated to an accuracy of approximately $\sigma_\theta/\sqrt{S}$.

  - this variance is independent on dimensionality of θ
  - total variance is sum of the epistemic uncertainty in the posterior and the uncertainty due to using finite number of Monte Carlo draws (Monte Carlo error),

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

How many simulation draws are needed?

- Expectation of unknown quantity θ (with mean μ_θ and sd σ_θ)

$$\mathrm{E}(\theta) \approx \frac{1}{S} \sum_{s=1}^{S} \theta^{(s)}$$

if S is big and $\theta^{(s)}$ are independent, we may assume that the distribution of the expectation approaches normal distribution (see Ch 4) with variance σ_θ^2/S (asymptotic normality) The posterior mean is then estimated to an accuracy of approximately $\sigma_\theta/\sqrt{S}$.

  - this variance is independent on dimensionality of θ
  - total variance is sum of the epistemic uncertainty in the posterior and the uncertainty due to using finite number of Monte Carlo draws (Monte Carlo error),

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

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

How many simulation draws are needed?

- Expectation of unknown quantity θ (with mean μ_θ and sd σ_θ)

$$\mathrm{E}(\theta) \approx \frac{1}{S}\sum_{s=1}^{S}\theta^{(s)}$$

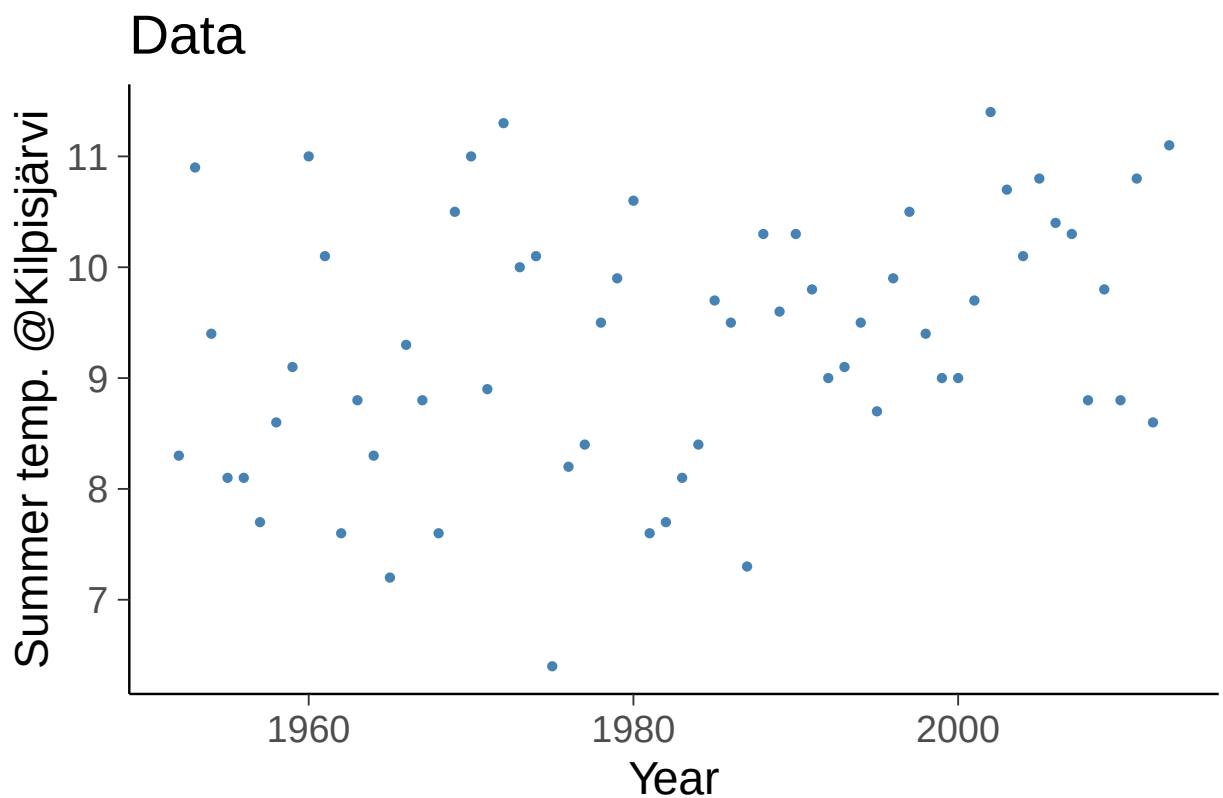
  if S is big and $\theta^{(s)}$ are independent, we may assume that the distribution of the expectation approaches normal distribution (see Ch 4) with variance σ_θ^2/S (asymptotic normality) The posterior mean is then estimated to an accuracy of approximately $\sigma_\theta/\sqrt{S}$.
  - this variance is independent on dimensionality of θ
  - total variance is sum of the epistemic uncertainty in the posterior and the uncertainty due to using finite number of Monte Carlo draws (Monte Carlo error),

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

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

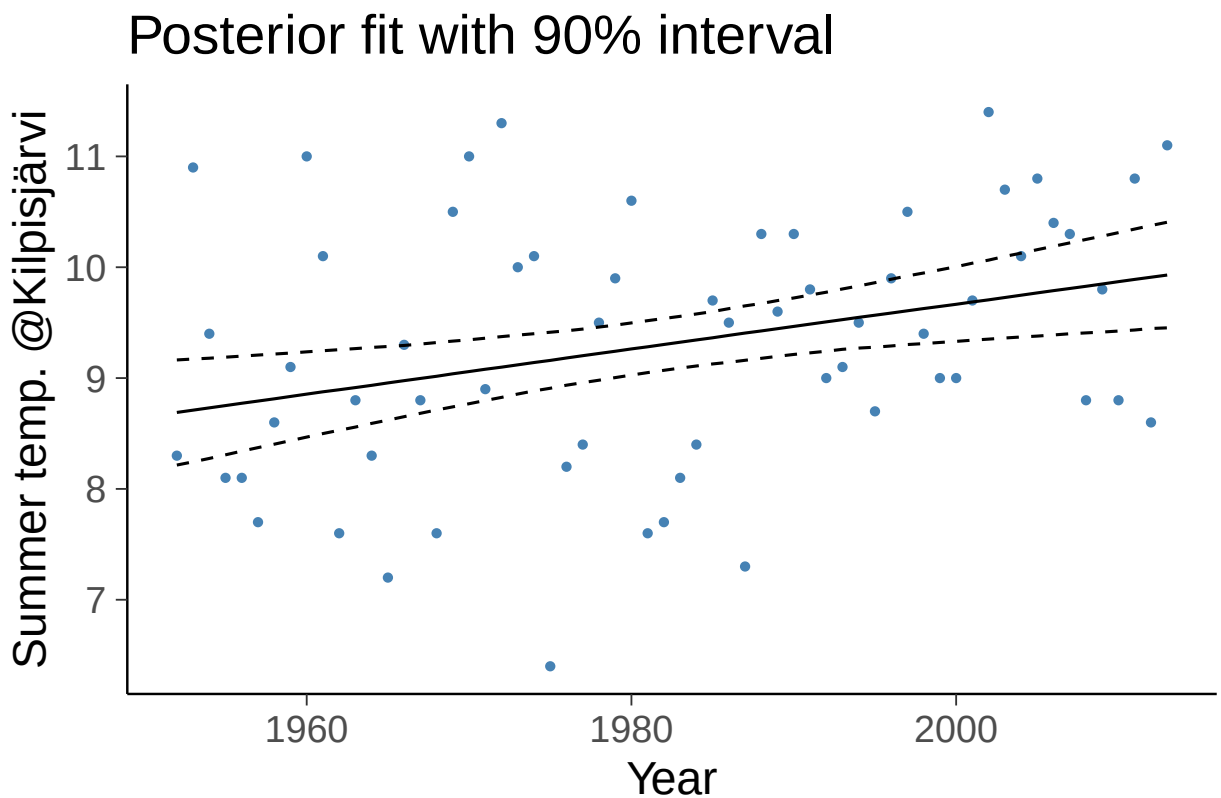
Example: Kilpisjärvi summer temperature

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

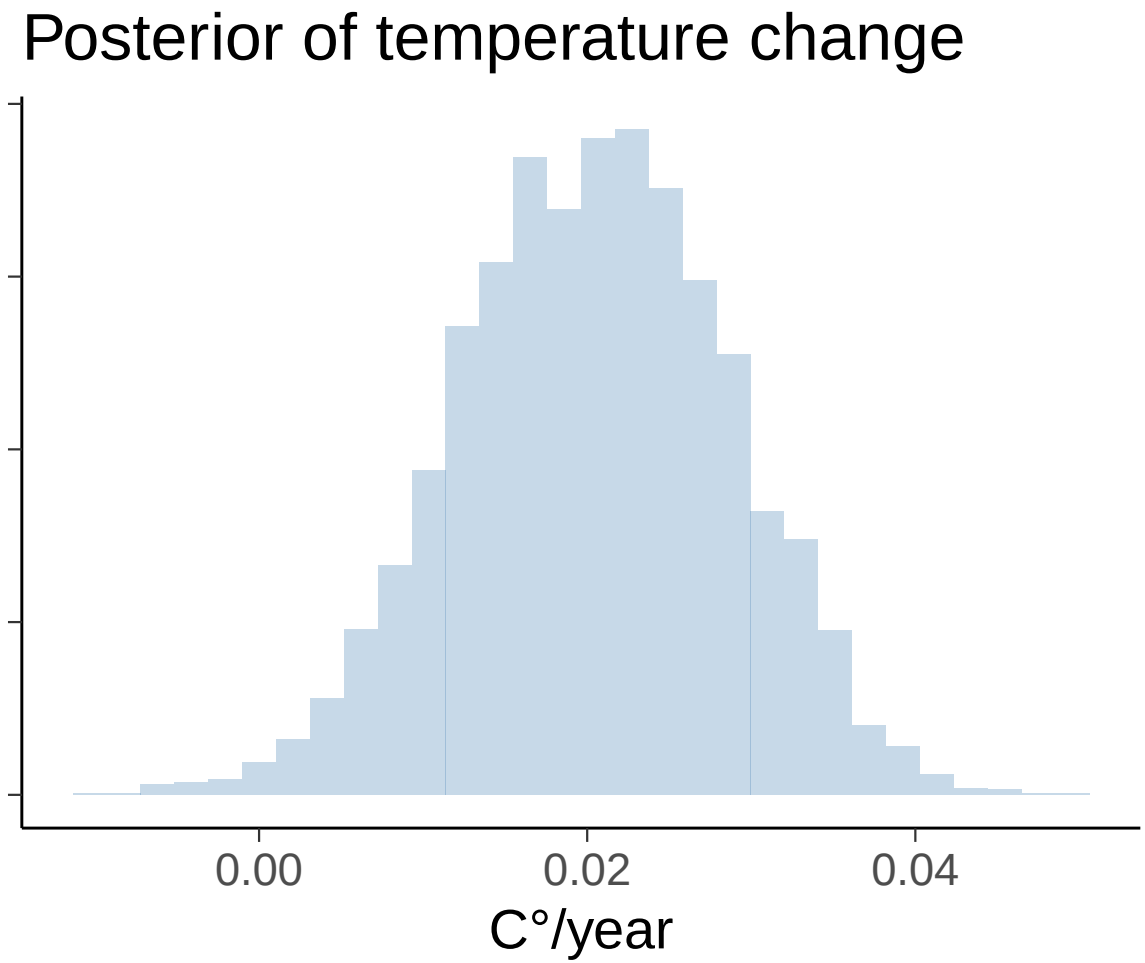


Example: Kilpisjärvi summer temperature

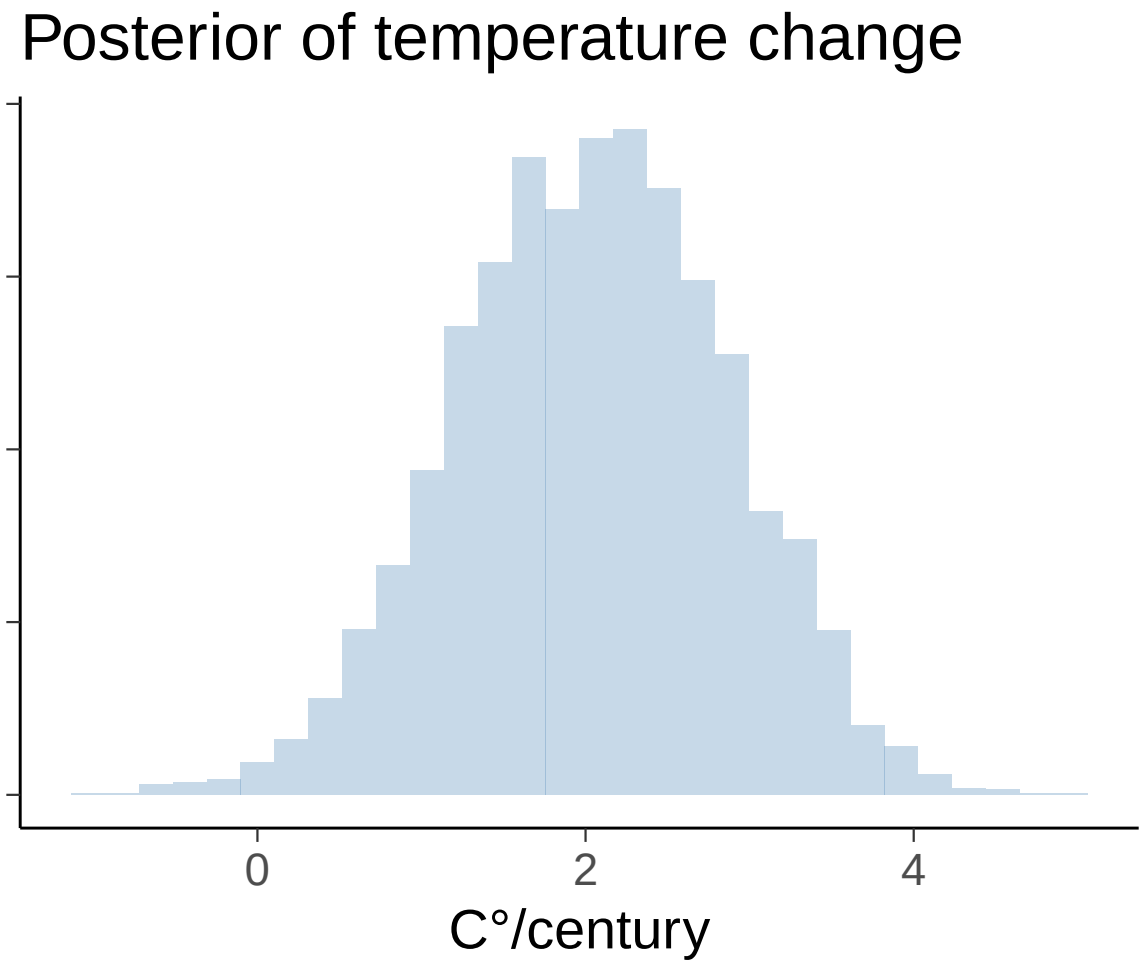
Average temperature in June, July, and August at Kilpisjärvi, Finland



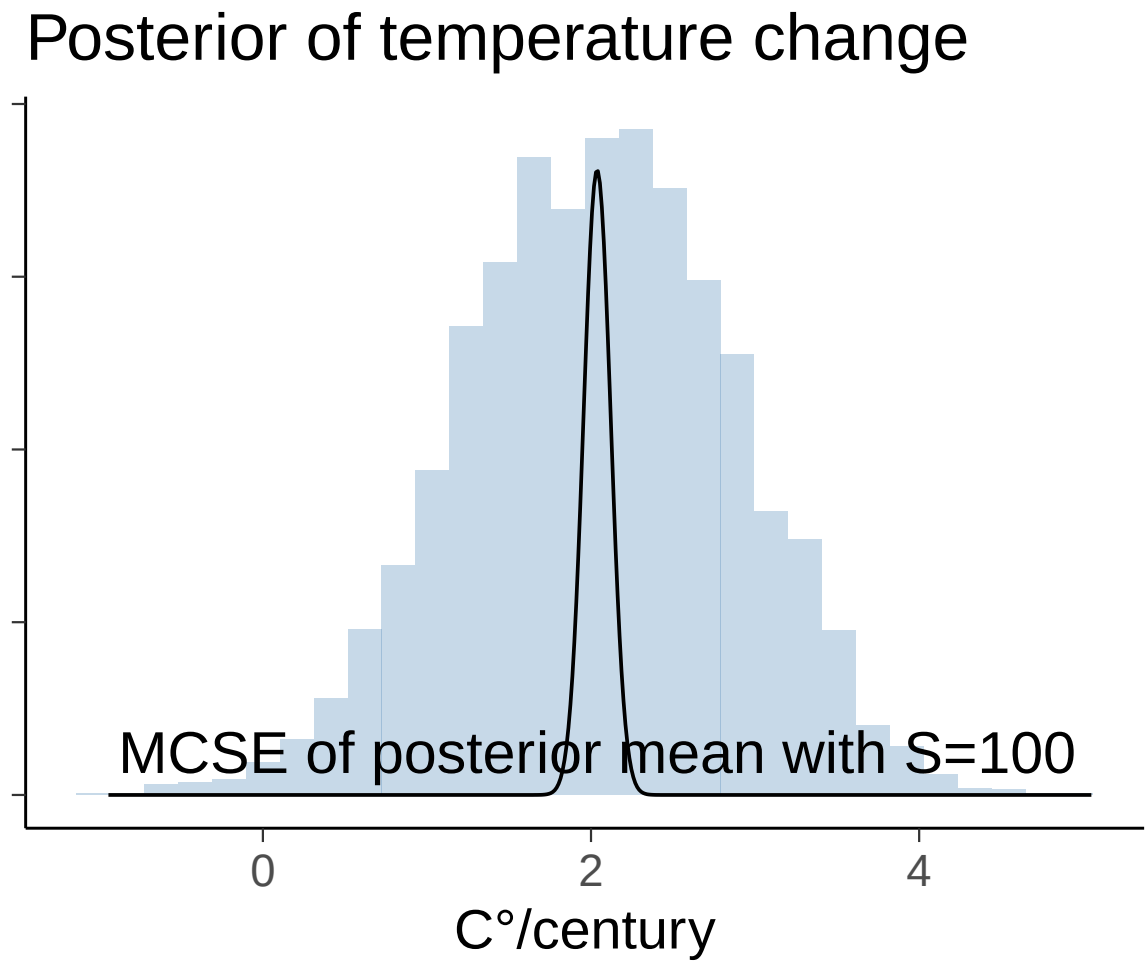
Example: Kilpisjärvi summer temperature



Example: Kilpisjärvi summer temperature



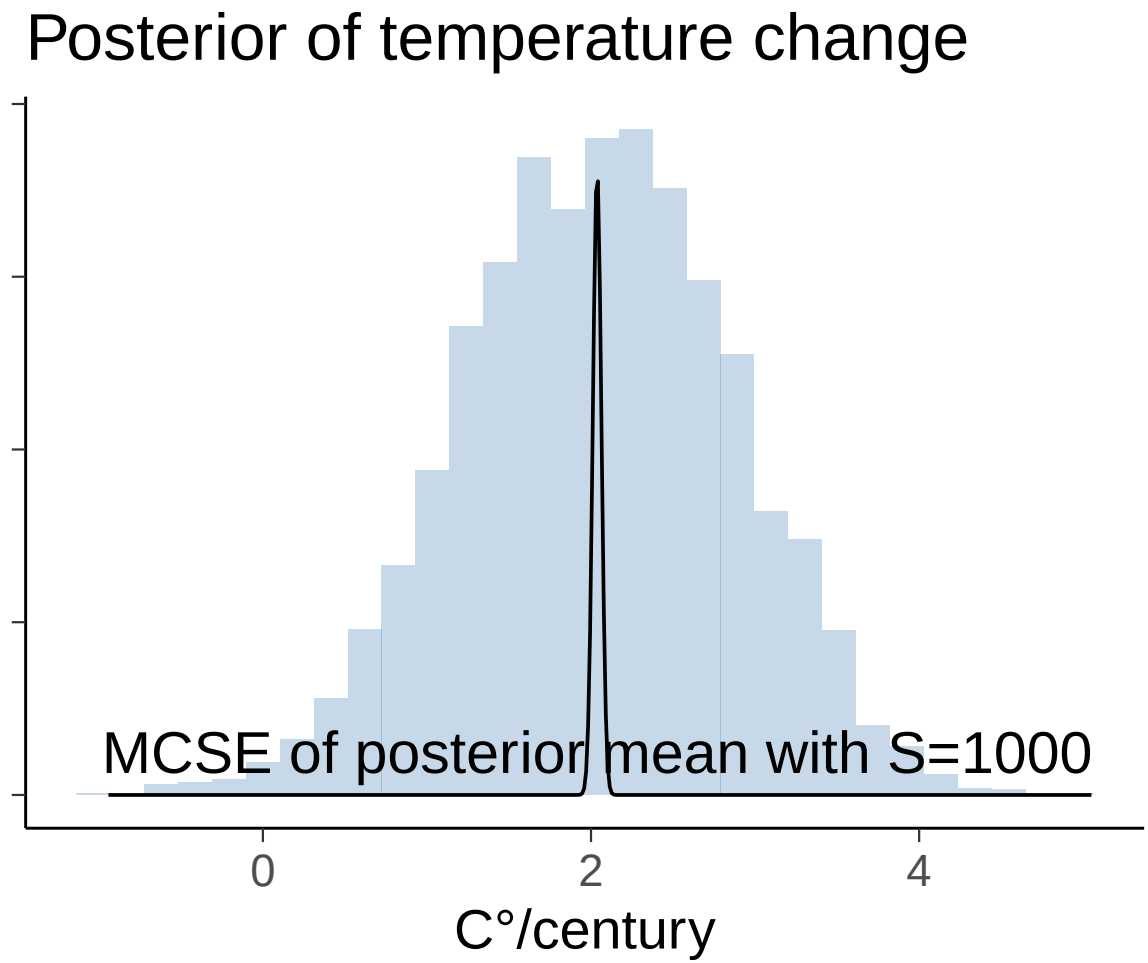
Example: Kilpisjärvi summer temperature



$\sigma_\theta \approx 0.827$, MCSE ≈ 0.0827, total deviation ≈ 0.831

$$\text{total deviation}^2 = \sigma_\theta^2 + \text{MCSE}^2$$

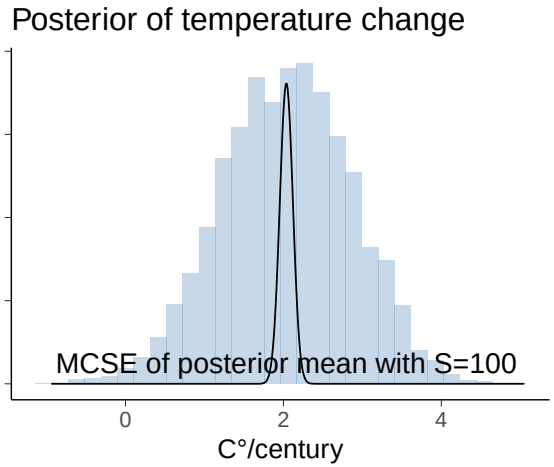
Example: Kilpisjärvi summer temperature

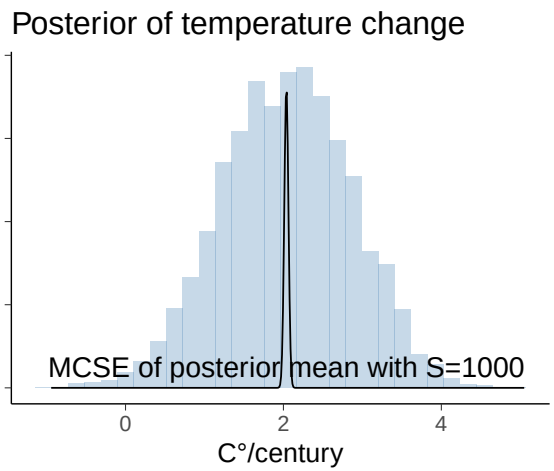


$\sigma_\theta \approx 0.827$, MCSE ≈ 0.0261, total deviation ≈ 0.827

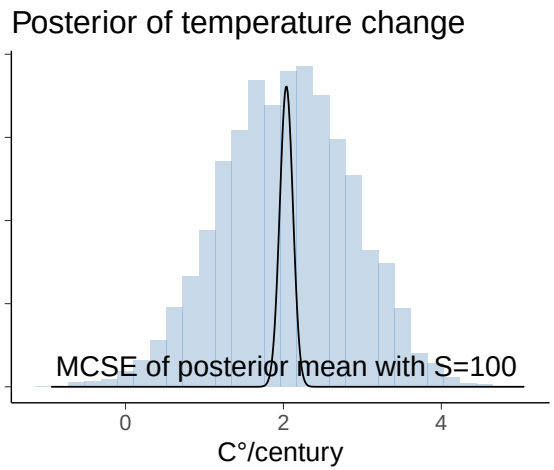
$$\text{total deviation}^2 = \sigma_\theta^2 + \text{MCSE}^2$$

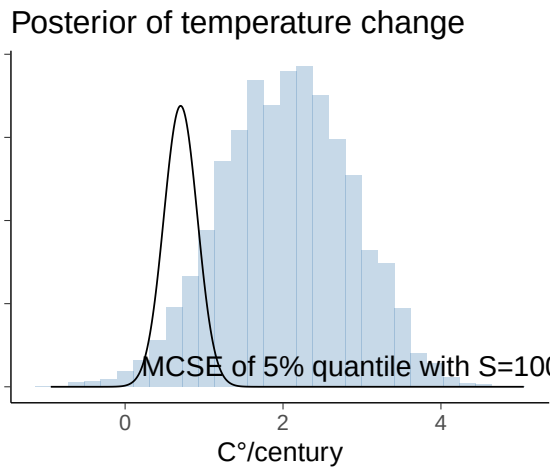
Example: Kilpisjärvi summer temperature

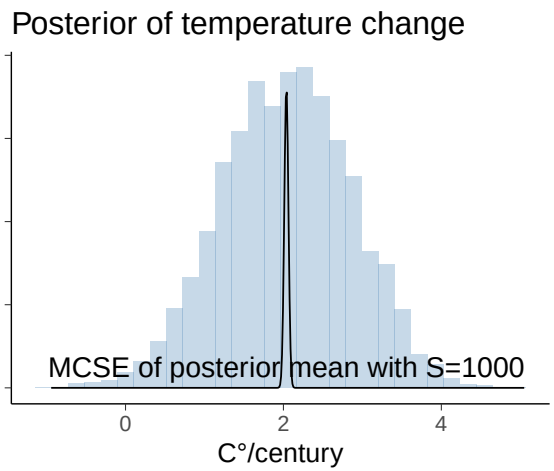


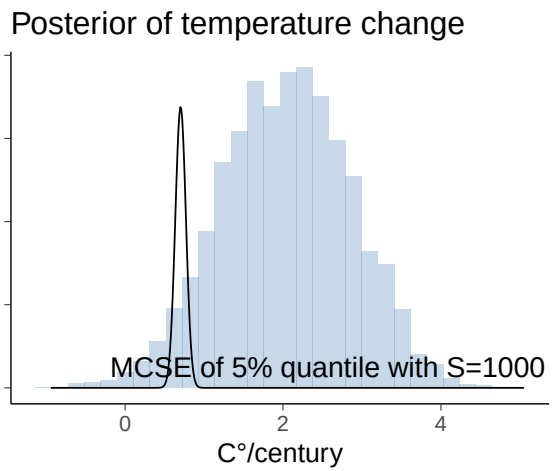


Example: Kilpisjärvi summer temperature

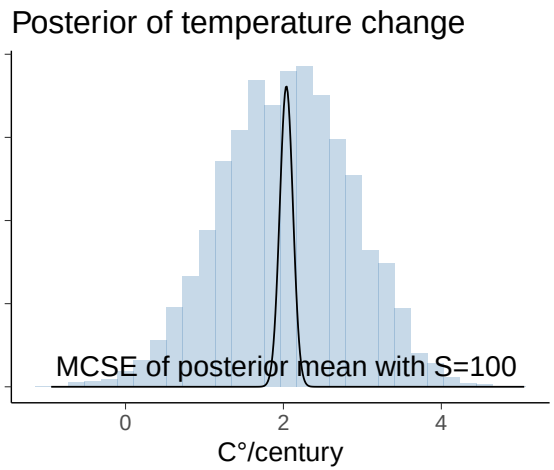


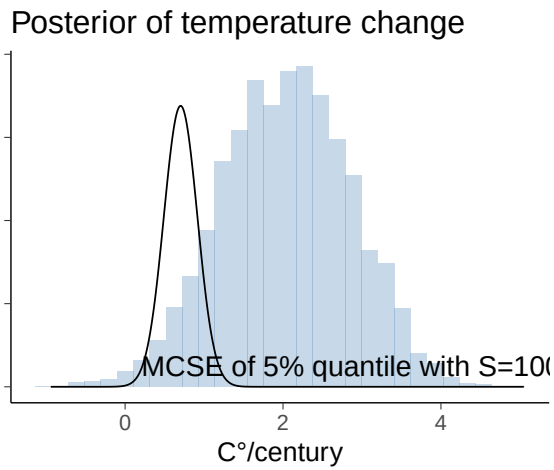


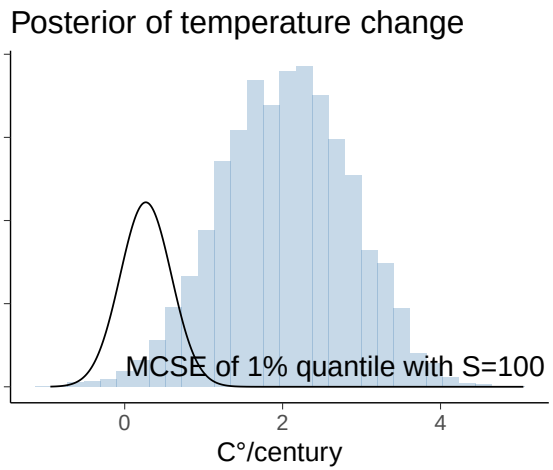


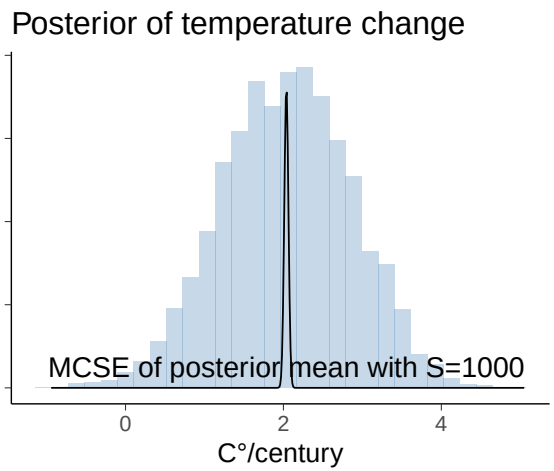


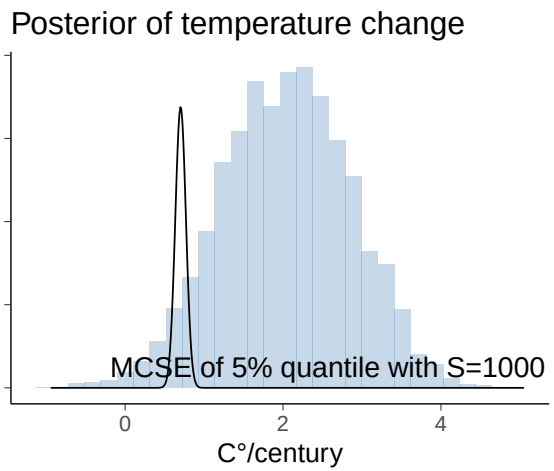
Example: Kilpisjärvi summer temperature

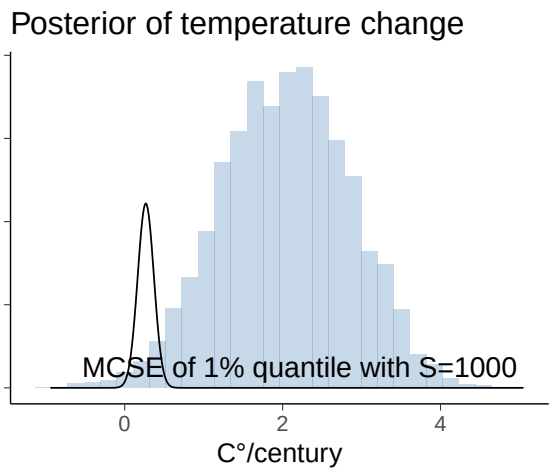




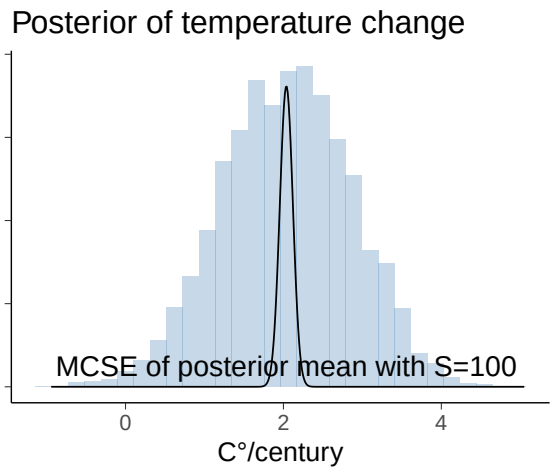


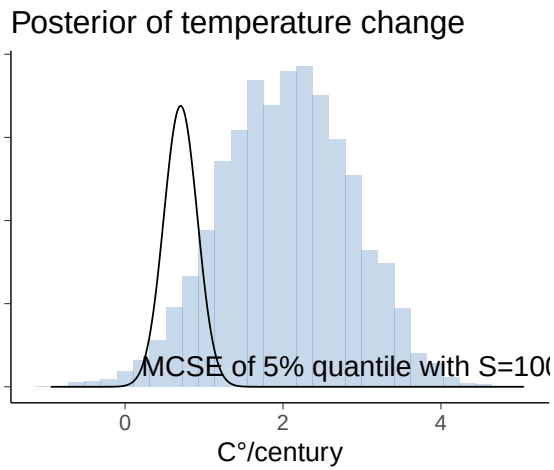


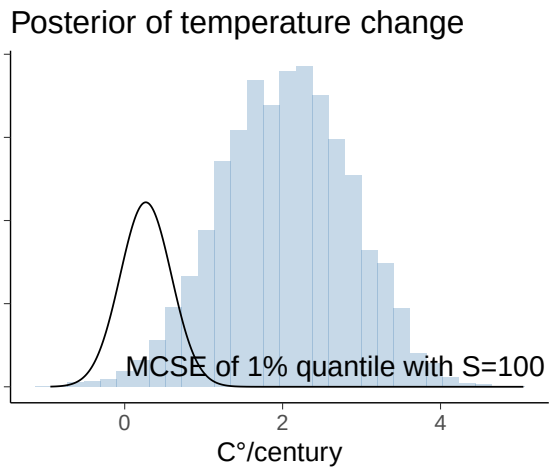


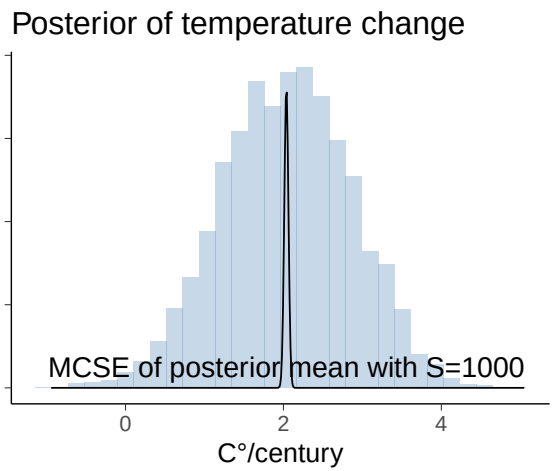


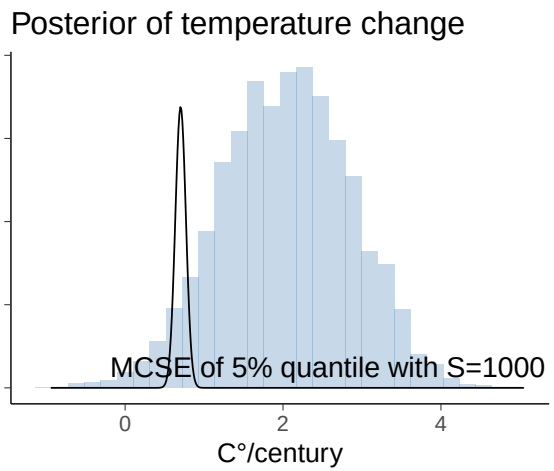
Example: Kilpisjärvi summer temperature

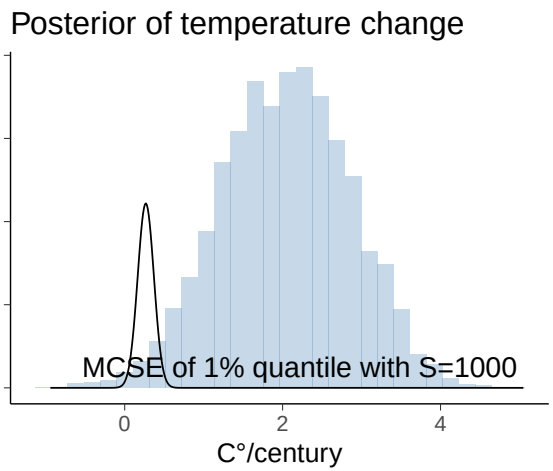












Tail quantiles are more difficult to estimate

How many simulation draws are needed?

For some posterior inferences, more simulation draws are needed to obtain desired precisions.

- Posterior probability

$$p(\theta \in A) \approx \frac{1}{S} \sum_{I} I(\theta^{(s)} \in A)$$

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

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

How many simulation draws are needed?

For some posterior inferences, more simulation draws are needed to obtain desired precisions.

- Posterior probability

$$p(\theta \in A) \approx \frac{1}{S} \sum_{I} I(\theta^{(s)} \in A)$$

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

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

How many simulation draws are needed?

For some posterior inferences, more simulation draws are needed to obtain desired precisions.

- Posterior probability

$$p(\theta \in A) \approx \frac{1}{S} \sum_{I} I(\theta^{(s)} \in A)$$

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

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

How many simulation draws are needed?

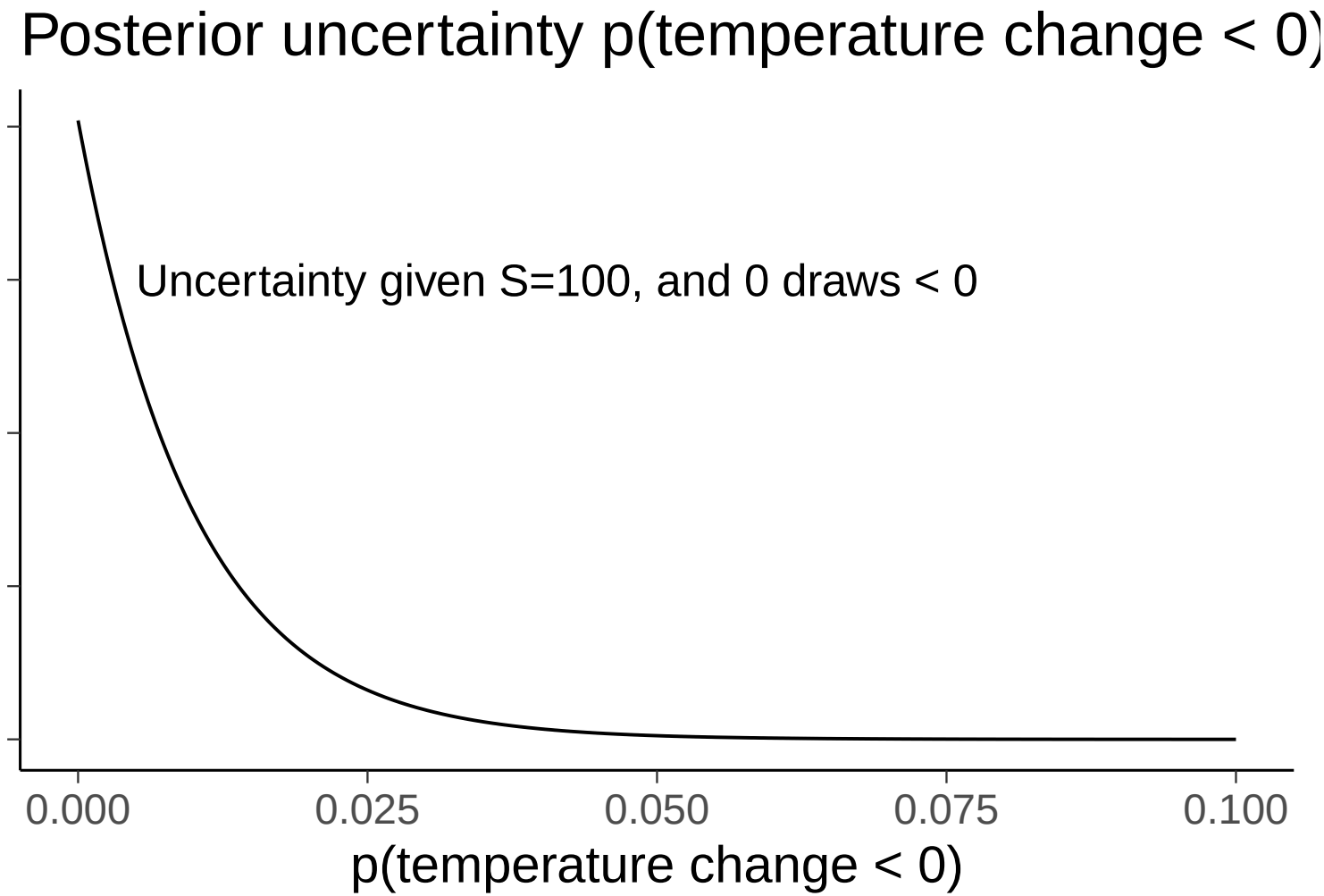
For some posterior inferences, more simulation draws are needed to obtain desired precisions.

- Posterior probability

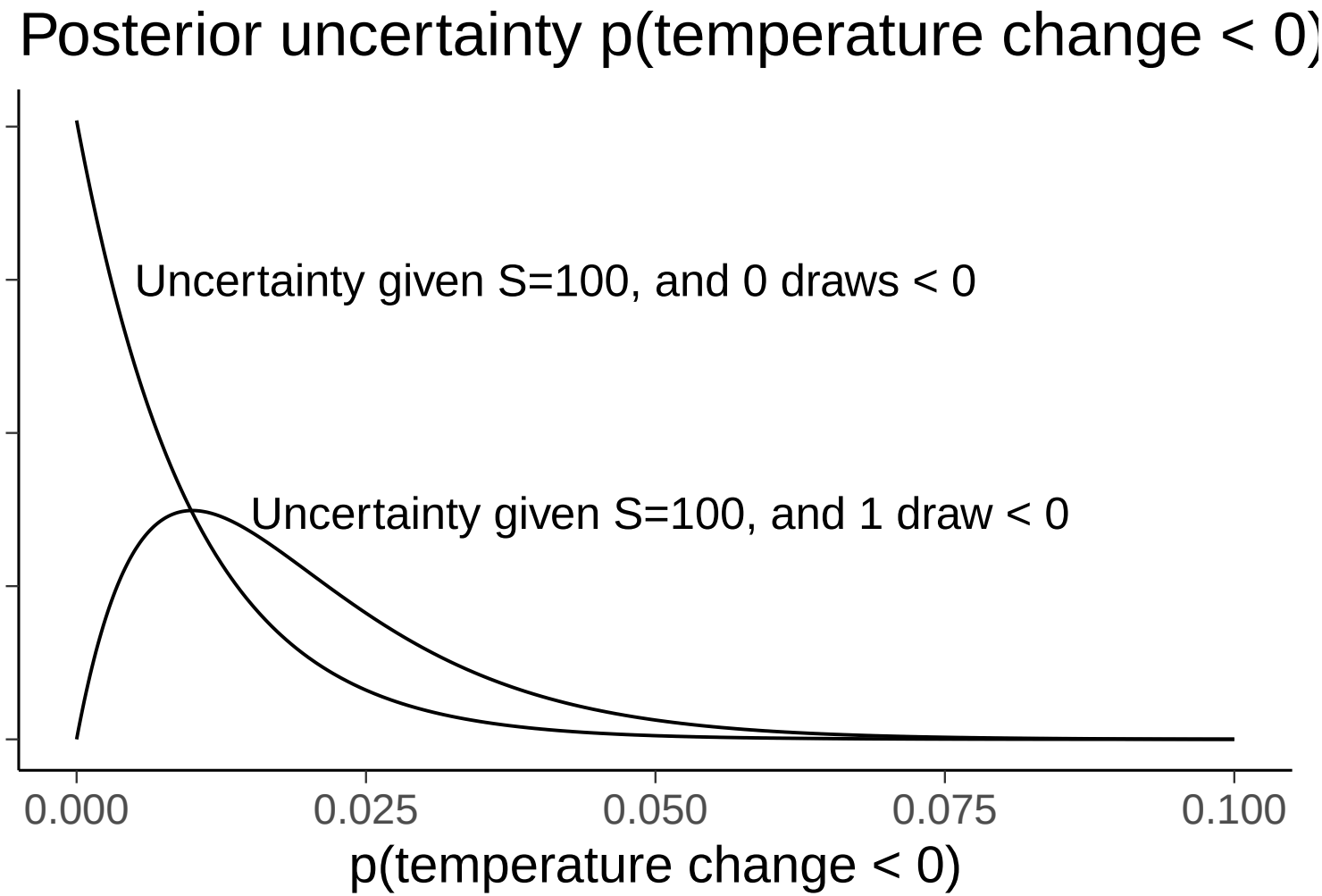
$$p(\theta \in A) \approx \frac{1}{S} \sum_{I} I(\theta^{(s)} \in A)$$

  where $I(\theta^{(s)} \in A) = 1$ if $\theta^{(s)} \in A$
  - $I(\cdot)$ is binomially distributed as $p(\theta \in A)$
    - $\rightarrow$ $\text{var}(I(\cdot)) = p(1-p)$ (Appendix A, p. 579)
    - $\rightarrow$ standard deviation of p is $\sqrt{p(1-p)/S}$
  - if $S = 100$ and $p \approx 0.5$, $\sqrt{p(1-p)/S} = 0.05$
    i.e. accuracy is about 5% units
  - $S = 2500$ 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 $\theta^{(I)} \in A$, which in expectation requires $S \gg 1/p$

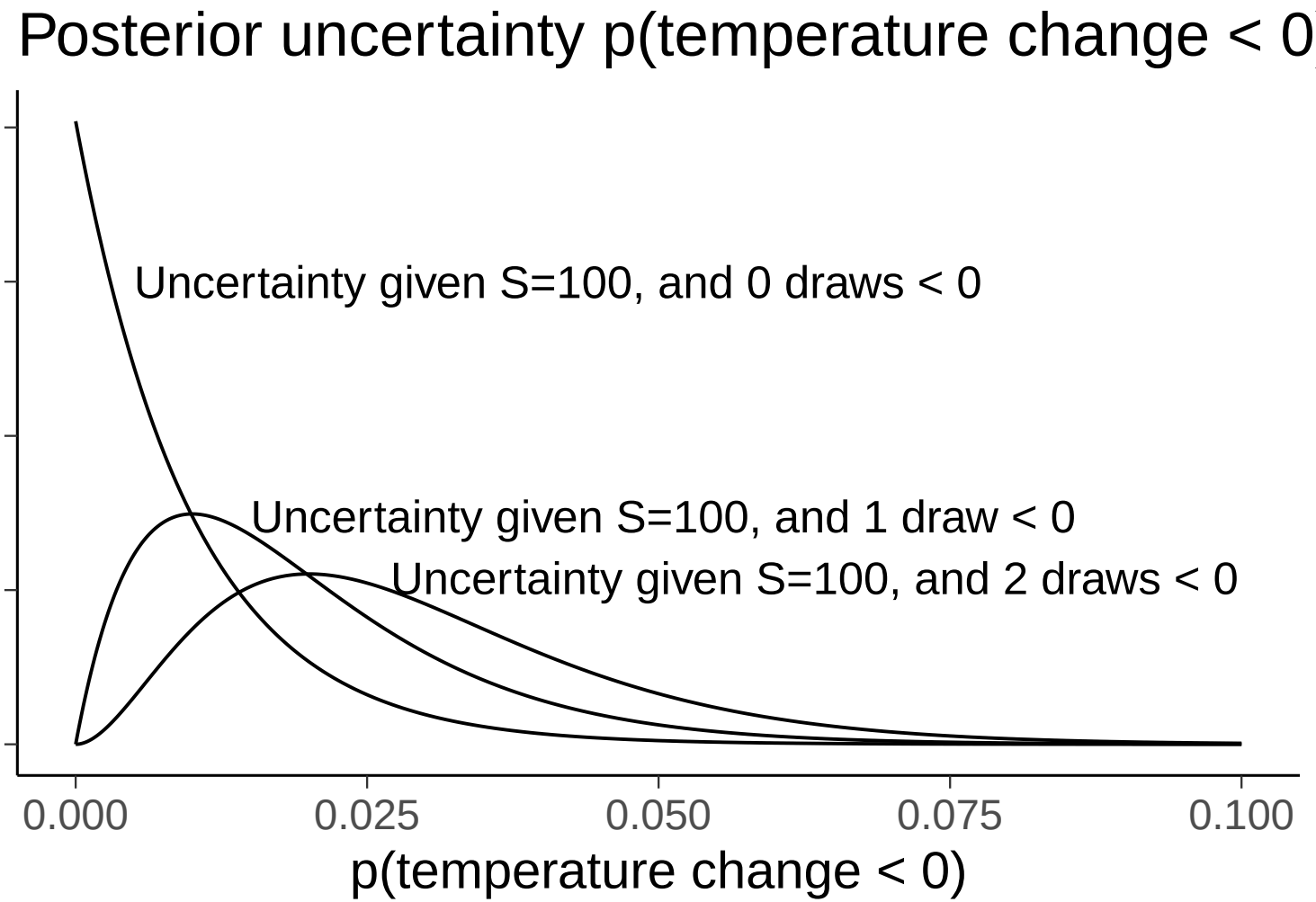
Example: Kilpisjärvi summer temperature



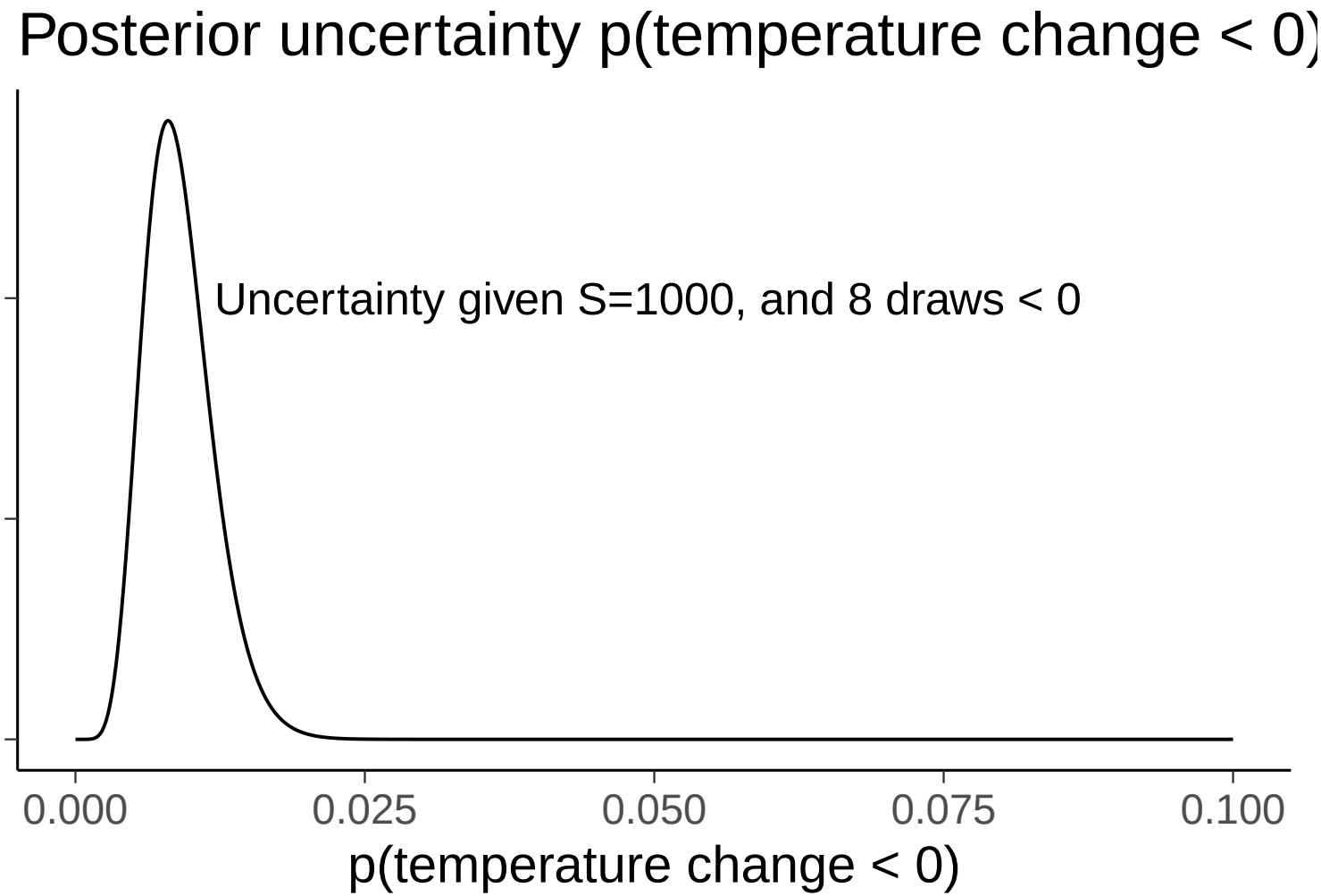
Example: Kilpisjärvi summer temperature



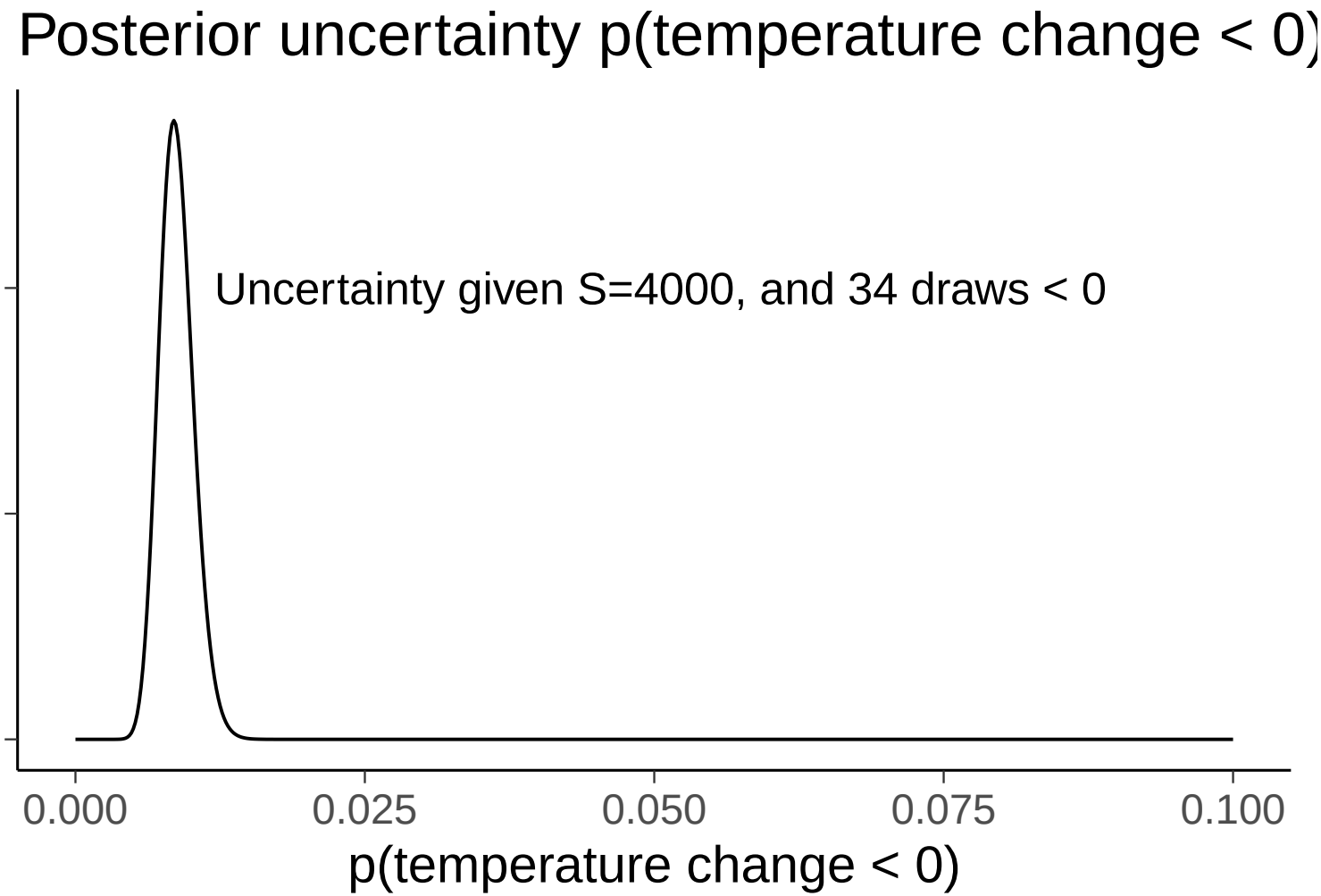
Example: Kilpisjärvi summer temperature



Example: Kilpisjärvi summer temperature



Example: Kilpisjärvi summer temperature



How many simulation draws are needed?

In general, fewer simulations are needed to estimate

- posterior medians of parameters, probabilities near 0.5, and low-dimensional summaries than
- extreme quantiles, posterior means, probabilities of rare events, and higher-dimensional summaries.

In most of the examples in BDA, a moderate number of simulation draws (typically 100 to 2000) is used emphasizing that applied inferences do not typically require a high level of simulation accuracy.

How many digits to show in reports?

- Too many digits make reading of the results slower and give false impression of the accuracy

How many digits to show in reports?

- Too many digits make reading of the results slower and give false impression of the accuracy
- Don't show digits which are just random noise
  - use Monte Carlo standard error estimates to check how many digits are likely to stay the same if the sampling would be continued.

How many digits to show in reports?

- Too many digits make reading of the results slower and give false impression of the accuracy
- Don't show digits which are just random noise
  - use Monte Carlo standard error estimates to check how many digits are likely to stay the same if the sampling would be continued.
- Show meaningful digits given the posterior uncertainty
  - You can compare posterior standard error or posterior intervals to the mean value. Posterior interval length can be used to determine also how many digits to show for the interval endpoints.

How many digits to show in reports?

- Too many digits make reading of the results slower and give false impression of the accuracy
- Don't show digits which are just random noise
  - use Monte Carlo standard error estimates to check how many digits are likely to stay the same if the sampling would be continued.
- Show meaningful digits given the posterior uncertainty
  - You can compare posterior standard error or posterior intervals to the mean value. Posterior interval length can be used to determine also how many digits to show for the interval endpoints.
- Example: The mean and 90% central posterior interval for temperature increase C°/century based on posterior draws

How many digits to show in reports?

- Too many digits make reading of the results slower and give false impression of the accuracy
- Don't show digits which are just random noise
  - use Monte Carlo standard error estimates to check how many digits are likely to stay the same if the sampling would be continued.
- Show meaningful digits given the posterior uncertainty
  - You can compare posterior standard error or posterior intervals to the mean value. Posterior interval length can be used to determine also how many digits to show for the interval endpoints.
- Example: The mean and 90% central posterior interval for temperature increase C°/century based on posterior draws
  - 2.050774 and [0.7472868 3.3017524] (NO! Too many digits)

How many digits to show in reports?

- Too many digits make reading of the results slower and give false impression of the accuracy
- Don't show digits which are just random noise
  - use Monte Carlo standard error estimates to check how many digits are likely to stay the same if the sampling would be continued.
- Show meaningful digits given the posterior uncertainty
  - You can compare posterior standard error or posterior intervals to the mean value. Posterior interval length can be used to determine also how many digits to show for the interval endpoints.
- Example: The mean and 90% central posterior interval for temperature increase C°/century based on posterior draws
  - 2.050774 and [0.7472868 3.3017524] (NO! Too many digits)
  - 2.1 and [0.7 3.3] (Good compared to the interval length)

How many digits to show in reports?

- Too many digits make reading of the results slower and give false impression of the accuracy
- Don't show digits which are just random noise
  - use Monte Carlo standard error estimates to check how many digits are likely to stay the same if the sampling would be continued.
- Show meaningful digits given the posterior uncertainty
  - You can compare posterior standard error or posterior intervals to the mean value. Posterior interval length can be used to determine also how many digits to show for the interval endpoints.
- Example: The mean and 90% central posterior interval for temperature increase C°/century based on posterior draws
  - 2.050774 and [0.7472868 3.3017524] (NO! Too many digits)
  - 2.1 and [0.7 3.3] (Good compared to the interval length)
  - 2 and [1 3] (depends on the context)

How many digits to show in reports?

- Too many digits make reading of the results slower and give false impression of the accuracy
- Don't show digits which are just random noise
  - use Monte Carlo standard error estimates to check how many digits are likely to stay the same if the sampling would be continued.
- Show meaningful digits given the posterior uncertainty
  - You can compare posterior standard error or posterior intervals to the mean value. Posterior interval length can be used to determine also how many digits to show for the interval endpoints.
- Example: The probability that temp increase is positive

How many digits to show in reports?

- Too many digits make reading of the results slower and give false impression of the accuracy
- Don't show digits which are just random noise
  - use Monte Carlo standard error estimates to check how many digits are likely to stay the same if the sampling would be continued.
- Show meaningful digits given the posterior uncertainty
  - You can compare posterior standard error or posterior intervals to the mean value. Posterior interval length can be used to determine also how many digits to show for the interval endpoints.
- Example: The probability that temp increase is positive
  - 0.9960000 (NO!)

How many digits to show in reports?

- Too many digits make reading of the results slower and give false impression of the accuracy
- Don't show digits which are just random noise
  - use Monte Carlo standard error estimates to check how many digits are likely to stay the same if the sampling would be continued.
- Show meaningful digits given the posterior uncertainty
  - You can compare posterior standard error or posterior intervals to the mean value. Posterior interval length can be used to determine also how many digits to show for the interval endpoints.
- Example: The probability that temp increase is positive
  - 0.9960000 (NO!)
  - 1.00 (depends on the context, 1.00 hints it's not exactly 1, but larger than 0.99)

How many digits to show in reports?

- Too many digits make reading of the results slower and give false impression of the accuracy
- Don't show digits which are just random noise
  - use Monte Carlo standard error estimates to check how many digits are likely to stay the same if the sampling would be continued.
- Show meaningful digits given the posterior uncertainty
  - You can compare posterior standard error or posterior intervals to the mean value. Posterior interval length can be used to determine also how many digits to show for the interval endpoints.
- Example: The probability that temp increase is positive
  - 0.9960000 (NO!)
  - 1.00 (depends on the context, 1.00 hints it's not exactly 1, but larger than 0.99)
  - With 4000 draws MCSE $\approx$ 0.002. We could report that probability is very likely larger than 0.99, or sample more to justify reporting three digits

How many digits to show in reports?

- Too many digits make reading of the results slower and give false impression of the accuracy
- Don't show digits which are just random noise
  - use Monte Carlo standard error estimates to check how many digits are likely to stay the same if the sampling would be continued.
- Show meaningful digits given the posterior uncertainty
  - You can compare posterior standard error or posterior intervals to the mean value. Posterior interval length can be used to determine also how many digits to show for the interval endpoints.
- Example: The probability that temp increase is positive
  - 0.9960000 (NO!)
  - 1.00 (depends on the context, 1.00 hints it's not exactly 1, but larger than 0.99)
  - With 4000 draws MCSE $\approx$ 0.002. We could report that probability is very likely larger than 0.99, or sample more to justify reporting three digits
  - For probabilities close to 0 or 1, consider also when the model assumption justify certain accuracy

How many simulation draws are needed?

- Less draws needed with
  - deterministic methods
  - marginalization (Rao-Blackwellization)
  - variance reduction methods, such, control variates
- Grid sampling and curse of dimensionality
  - Number of grid points increases exponentially
  - Concentration of the measure, i.e., where is the most of the mass?

How many simulation draws are needed?

- Less draws needed with
  - deterministic methods
  - marginalization (Rao-Blackwellization)
  - variance reduction methods, such, control variates
- Grid sampling and curse of dimensionality
  - Number of grid points increases exponentially
  - Concentration of the measure, i.e., where is the most of the mass?
- 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

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 (the most used)
  - Gibbs: “iterative conditional sampling”
  - Metropolis: “random walk in joint distribution”
  - Dynamic Hamiltonian Monte Carlo: “state-of-the-art” used in Stan