Bayesian Computation

Bayesian inference relies on the ability to compute probabilities and other quantities associated with the posterior distribution.

If $f(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$ is not proportional to a "familiar" density, then approximation of the posterior may be an issue.

Suppose, for example, that we wish to compute $P(\boldsymbol{\theta} \in A|\boldsymbol{y})$, where A is some subset of the parameter space Θ. If the function (of $\boldsymbol{\theta}$) $f(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$ cannot be integrated analytically, then some form of numerical integration is required.

In particular, it will be necessary to approximate

$$m(\boldsymbol{y}) = \int_{\Theta} f(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})\, d\boldsymbol{\theta},$$

or to be able to generate values from the posterior without knowing $m(\boldsymbol{y})$.

Rejection Sampling

Rejection sampling is a method of drawing samples from a density p. Two properties of the method are appealing:

1. The density p need only be known up to a constant of proportionality. Suppose we know only q, where $q(\theta) = Cp(\theta)$ for all θ.

2. We don't need to know how to draw samples from p. Instead, we need to know how to generate observations from a density g for which it is known that

$$\frac{q(\theta)}{g(\theta)} \leq M \quad \text{for all } \theta,$$

where M is a known constant.

Rejection sampling algorithm

1. Draw an observation θ from g.

2. With probability $q(\theta)/[Mg(\theta)]$, accept the observation as a draw from p. Otherwise, return to step 1.

Let X be an observation from step 1. Then the conditional density of X given that X is accepted in step 2 is p.

Proof

Let X and U be independent random variables, where X has density g and U is $U(0,1)$. We will derive the cdf of X conditional on the event A that X is an "accepted" draw.

We have

$$\begin{aligned} P(X \le x | A) &= \frac{P(X \le x \cap A)}{P(A)} \\ &= \frac{P\left(X \le x \cap U \le \frac{q(X)}{Mg(X)}\right)}{P\left(U \le \frac{q(X)}{Mg(X)}\right)}. \end{aligned}$$

Now,

$$P\left(X \le x \cap U \le \frac{q(X)}{Mg(X)}\right) =$$

$$\int_{-\infty}^{x} \int_{0}^{q(\theta)/[Mg(\theta)]} g(\theta)\, du\, d\theta =$$

$$\int_{-\infty}^{x} g(\theta) \int_{0}^{q(\theta)/[Mg(\theta)]} du\, d\theta =$$

$$\int_{-\infty}^{x} g(\theta) \frac{q(\theta)}{Mg(\theta)}\, d\theta =$$

$$\frac{C}{M} \int_{-\infty}^{x} p(\theta)\, d\theta.$$

Likewise, $P(A) = C/M$, and hence

$$P(X \leq x|A) = \int_{-\infty}^{x} p(\theta)\, d\theta \quad \forall\ x.$$

It immediately follows that the conditional pdf of X given A is p. $\square$

An example

Let $Y_1, \ldots, Y_{10}$ be a random sample from a Cauchy distribution with known scale parameter equal to 1 and unknown location θ. Let the prior for θ be $N(0, 25)$.

It follows that

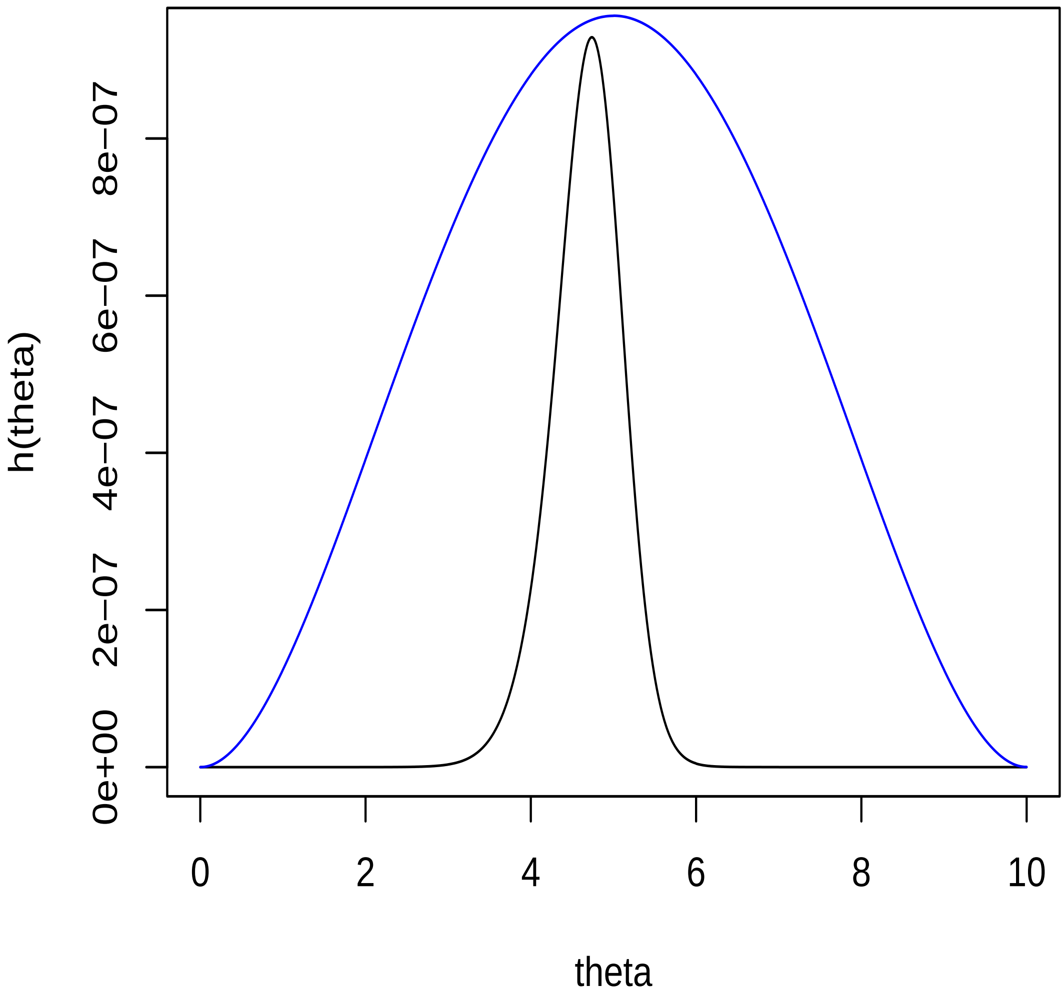
$$\pi(\theta|\boldsymbol{y}) \propto h(\theta) =$$

$$\exp(-\theta^2/50) \prod_{i=1}^{10} \left[1 + (\theta - y_i)^2\right]^{-1}.$$

A random sample was generated from the Cauchy distribution with location and scale parameters equal to 5 and 1, respectively. The order statistics were:

-21.847	-0.718	3.069	3.616	4.462
4.768	4.880	5.218	5.355	5.726

The median for this sample is 4.615.



$h(\theta)$: —— $Cg(\theta)$: ——

Let g be the density of the random variable $10Y$, where Y has the beta$(3,3)$ distribution. The blue curve above is a plot of $Cg(\theta)$, where $C = (5.1)10^{-6}$.

We have

$$\frac{h(\theta)}{Cg(\theta)} \leq 1 \quad \text{for} \quad 0 < \theta < 10,$$

and so we may use the following rejection algorithm to generate a value from the posterior:

1. Generate a value from the beta(3,3) distribution and multiply it by 10. Call the result $\tilde{\theta}$.

2. Generate a value, call it U, from the $U(0,1)$ distribution. Accept $\tilde{\theta}$ as a draw from the posterior if $U \leq h(\tilde{\theta})/[Cg(\tilde{\theta})]$. Otherwise go back to 1.

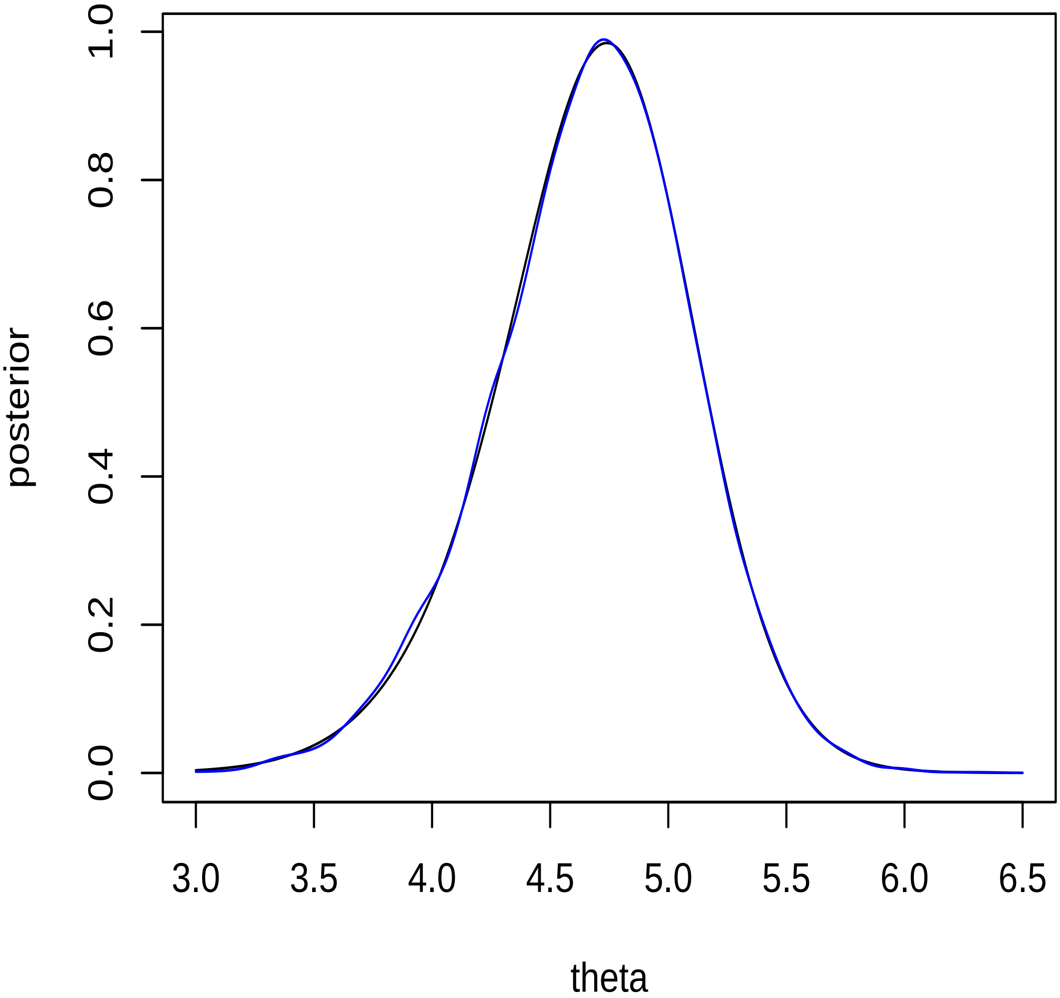
R code for rejection sampling

```
reject = function(y,nreps){
samples=1:nreps
counts=rep(1,len=nreps)
for(i in 1:nreps){
x=10*rbeta(1,3,3)
ratio=posterior(x,y)/
      (3*1.7*dbeta(x/10,3,3)/10^7)
check=sign(runif(1)-ratio)
while(check > 0){
counts[i]=counts[i]+1
x=10*rbeta(1,3,3)
ratio=posterior(x,y)/
      (3*1.7*dbeta(x/10,3,3)/10^7)
check=sign(runif(1)-ratio)
}
samples[i]=x
}
list(counts,samples)
}
```

Ten thousand values were generated from the posterior using the function on the previous page. A total of 53,441 values had to be generated from g in order to get these 10,000.

The median number of generations required from g to get one acceptable value from the posterior was 4.

On the next page is a kernel density estimate of the 10,000 values generated. Superimposed on this estimate is the true posterior. (The constant of proportionality for the true posterior was determined by numerical integration.)



Posterior: ——

Kernel estimate from 10,000 draws: ——