ADVANCED BIOSTATISTICS
ABSTAT17
Gibbs Sampling

Lisete Sousa

Department of Statistics and Operations Research, CEAUL
Faculty of Sciences of Lisbon University

IGC, April 10th - 13th, 2017

Contents

- Review
- Markov Chain Monte Carlo
- The Gibbs Sampler

Review

Bayesian Statistics

If $\mathbf{y}$ is a future observation from a model indexed by the same parameter $\boldsymbol{\theta}$, predictions on $\mathbf{y}$ are based on the predictive distribution

$$p(\mathbf{y}|\mathbf{x}) = \int f(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{x}) d\boldsymbol{\theta}$$

where $f(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta})$ is the distribution of $\mathbf{y}$ under the assumed parametric model. Again, summary predictions can be obtained in the form of predictive expectations

$$E[g(\mathbf{y})|\mathbf{x}] = \int g(\mathbf{y}) p(\mathbf{y}|\mathbf{x}) d\mathbf{y}$$

for suitable choices of $g(.)$.

Thus, in the continuous case, the integration operation plays a fundamental role in Bayesian Statistics.

Several numerical approximation strategies have been suggested, such as,

- Laplace Approximation,
- Sampling-importance-resampling (SIR),
- Markov Chain Monte Carlo Methods (MCMC).

Monte Carlo Methods

Consider then the problem of approximating an integral of the form

$$\int g(\boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{x}) d\boldsymbol{\theta} = E[g(\boldsymbol{\theta})|\mathbf{x}] < \infty,$$

where $\boldsymbol{\theta}$ and $\mathbf{x}$ may be vectors and $g(\boldsymbol{\theta})$ can depend on any other values besides $\boldsymbol{\theta}$.

If $p(\boldsymbol{\theta}|\mathbf{x})$ is a density and if we can simulate a random sample $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n$ from $p(.|\mathbf{x})$, then the Monte Carlo method approximates the integral by

$$\frac{1}{n} \sum_{i=1}^{n} g(\boldsymbol{\theta}_i),$$

which by the Law of Large Numbers converges almost surely to $E[g(\boldsymbol{\theta})|\mathbf{x}]$.

The precision of this estimate can be measured by the estimated Monte Carlo standard error given by

$$\frac{1}{\sqrt{n(n-1)}}\left\{\sum_{i=1}^{n}\left[g(\boldsymbol{\theta}_i)-\frac{1}{n}\sum_{i=1}^{n}g(\boldsymbol{\theta}_i)\right]^2\right\}^{1/2}.$$

Hence, provided we can sample from the posterior distribution, using such samples, it is easy to estimate characteristics such as the posterior mean or standard deviation of a function of $\boldsymbol{\theta}$, etc.

Difficulties

However difficulties arise when we want to sample from a complex multivariate distribution. This is the rule, rather than the exception, in the Bayesian context.

Indeed very seldom it is possible to sample directly from the posterior distribution and thus obtain an i.i.d. sample from $p(\boldsymbol{\theta}|\mathbf{x})$ and hence special strategies, such as MCMC were devised.

Markov Chain Monte Carlo (MCMC)

Markov Chain Monte Carlo (MCMC) methods, attempt to simulate direct draws from some complex distribution of interest (Walsh, 2004).

Why Markov Chain and why Monte Carlo?

Markov Chain:

Because one uses sample values to randomly generate the next sample value, generating a Markov chain

The transition probabilities between sample values are only a function of the most recent sample value.

Monte Carlo simulation:

Computer experiment involving random sampling from probability distributions.

Usually, when statisticians talk about simulations, they mean Monte Carlo simulations.

The Origin of MCMC

The original Monte Carlo approach was a method developed by physicists to use random number generation to compute integrals:

$$\int_a^b h(x)dx = \int_a^b f(x)p(x)dx = E_{p(x)}\left[f(X)\right] \approx \frac{1}{n}\sum_{i=1}^{n} f(x_i)$$

by decomposing $h(x)$ into the production of a function $f(x)$ and a probability density function $p(x)$, defined over the interval (a, b).

One problem with applying Monte Carlo integration is in obtaining samples from some complex probability distribution. The attempts to solve this problem are the roots of MCMC methods.

This results in the Metropolis-Hastings algorithm, which constituted an attempt by physicists to compute complex integrals by expressing them as expectations for some distribution. Afterwards, they estimated this expectation by drawing samples from that distribution. (Walsh, 2004)

Another motivation for the development of this methodology was the calculation of the posterior distribution necessary for Bayesian approaches. This often requires the integration of high-dimensional functions, which can be computationally very difficult. In this case, one would have the approximation:

$$\int f(y|x)p(x)dx \approx \frac{1}{n}\sum_{i=1}^{n} f(y|x_i)$$

The Gibbs Sampler

Geman and Geman (1984) introduced the GIBBS sampler as a way of simulating from high-dimensional complex distributions arising in image restoration.

Gelfand and Smith (1990) showed how the algorithm can be used to simulate from posterior distributions, and hence how to be used to solve problems in Bayesian Statistics.

In this situation, the algorithm is based on the fact that, if the joint distribution $p(\boldsymbol{\theta}|\mathbf{x})$ for $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_k)$ is positive over its entire domain, then it is uniquely determined by the k full conditional distributions (Besag, 1974)

$$p(\theta_i|\mathbf{x}, \boldsymbol{\theta}_{(-i)}), i = 1, \ldots, k,$$

where

$$\boldsymbol{\theta}_{(-i)} = (\theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots \theta_k).$$

The algorithm is then a Markovian updating scheme which requires sampling from these full conditional distributions as follows.

Suppose we are given an arbitrary set of initial values

$$\boldsymbol{\theta}^0 = (\theta_1^{(0)}, \ldots, \theta_k^{(0)})$$

then we proceed with the following iterative procedure:

- draw $\theta_1^{(1)}$ from $p(\theta_1 | \mathbf{x}, \theta_2^{(0)}, \ldots, \theta_k^{(0)})$,
- draw $\theta_2^{(1)}$ from $p(\theta_2 | \mathbf{x}, \theta_1^{(1)}, \theta_3^{(0)}, \ldots, \theta_k^{(0)})$,
- ...
- draw $\theta_k^{(1)}$ from $p(\theta_k | \mathbf{x}, \theta_1^{(1)}, \ldots, \theta_{k-1}^{(1)})$.

This completes one iteration of the scheme and a transition from $\boldsymbol{\theta}^0$ to $\boldsymbol{\theta}^1 = (\theta_1^{(1)}, \ldots, \theta_k^{(1)})$.

Iteration of this cycle produces a sequence $\boldsymbol{\theta}^0, \ldots, \boldsymbol{\theta}^t, \ldots$, which is a realization of a Markov chain with transition probabilities given by

$$\pi(\boldsymbol{\theta}^t, \boldsymbol{\theta}^{t+1}) = \prod_{i=1}^{k} p(\theta_i^{(t+1)} | \mathbf{x}, \theta_1^{(t+1)}, \ldots, \theta_{i-1}^{(t+1)}, \theta_{i+1}^{(t)}, \ldots, \theta_k^{(t)})$$

As $t \to \infty$, $\boldsymbol{\theta}^t = (\theta_1^{(t)}, \ldots, \theta_k^{(t)})$ tends in distribution to a random vector whose joint density is $p(\boldsymbol{\theta}|\mathbf{x})$.

In particular, $\theta_i^{(t)}$ tends in distribution to a random quantity whose density is $p(\theta_i|\mathbf{x})$ and

$$\frac{1}{t}\sum_{i=1}^{t} g(\boldsymbol{\theta}^i) \rightarrow E_{\boldsymbol{\theta}|\mathbf{x}}[g(\boldsymbol{\theta})] \ ,$$

for any function $g(.)$, where $E_{\boldsymbol{\theta}|\mathbf{x}}[g(\boldsymbol{\theta})]$ represents the expected value of $g(\boldsymbol{\theta})$ with respect to the posterior density function $p(\boldsymbol{\theta}|\mathbf{x})$.

Checking Convergence

Since convergence usually occurs regardless of our starting point, we can usually pick any feasible (for example, picking starting draws that are in the parameter space) starting point.

However, the time it takes for the chain to converge varies depending on the starting point.

As a matter of practice, most people **throw out a certain number of the first draws**, known as the burn-in. This is to make our draws closer to the stationary distribution and less dependent on the starting point.

Lam, P., MCMC Methods: Gibbs Sampling and the Metropolis-Hastings Algorithm

However, it is unclear how much we should burn-in since our draws are all slightly dependent and we do not know exactly when convergence occurs.

In order to break the dependence between draws in the Markov chain, some have suggested only **keeping every dth draw of the chain**.

This is known as thinning.

Lam, P., MCMC Methods: Gibbs Sampling and the Metropolis-Hastings Algorithm

Example - Univariate Normal

Consider a random vector $\mathbf{Y} = (Y_1, \ldots, Y_n)$, whose components are independent and normally distributed:

$$Y_i \frown N(\mu, \sigma^2),\ \mu \in \mathbb{R} i = 1, \sigma > 0, \ldots, n.$$

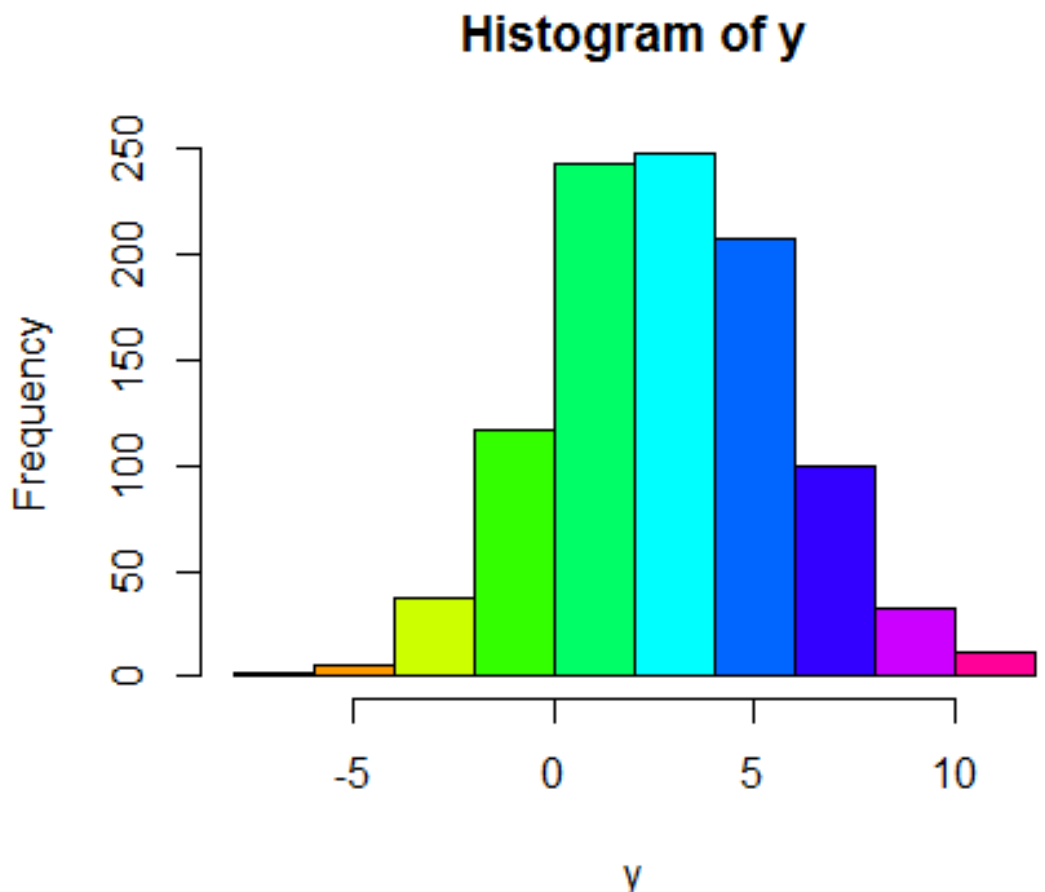
```
> n<-1000
> mu<-3
> sigma<-3
> sigma2<-sigma^2
> y<-rnorm(n,mu,sigma);head(y)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-7.127   1.015   3.086   3.018   5.081  13.260
```

```
> summary(y)
```

```
[1] 4.008252 7.581140 2.693658 3.924530 3.763089 1.599321
```

```
> hist(y,col=rainbow(10))
```



Considering a non-informative prior distribution, according to Jeffrey, we have:

$$p(\mu, \sigma^2) \propto \sigma^{-3}$$

and the posterior distributions for the parameters μ and $\tau = \frac{1}{\sigma^2}$ are:

$$\mu|\tau, \mathbf{y} \frown N\left(\overline{y}, \frac{1}{n\tau}\right)$$

$$\tau|\mathbf{y} \frown Gamma\left(\frac{n}{2}, \frac{\sum(y_i - \overline{y})^2}{2}\right)$$

$$\tau|\mu, \mathbf{y} \frown Gamma\left(\frac{n}{2}, \frac{\sum(y_i - \mu)^2}{2}\right)$$

Now it is possible to sampling from the full conditional distributions and

- draw $\mu^{(1)}$ from $p(\mu|\tau^{(0)}, \mathbf{y})$,
- draw $\tau^{(1)}$ from $p(\tau|\mu^{(1)}, \mathbf{y})$,
- and so on...

In order to simulate $\mu^{(1)}$, we have to initialize τ (represented by $\tau^{(0)}$) and calculate $\overline{y}$:

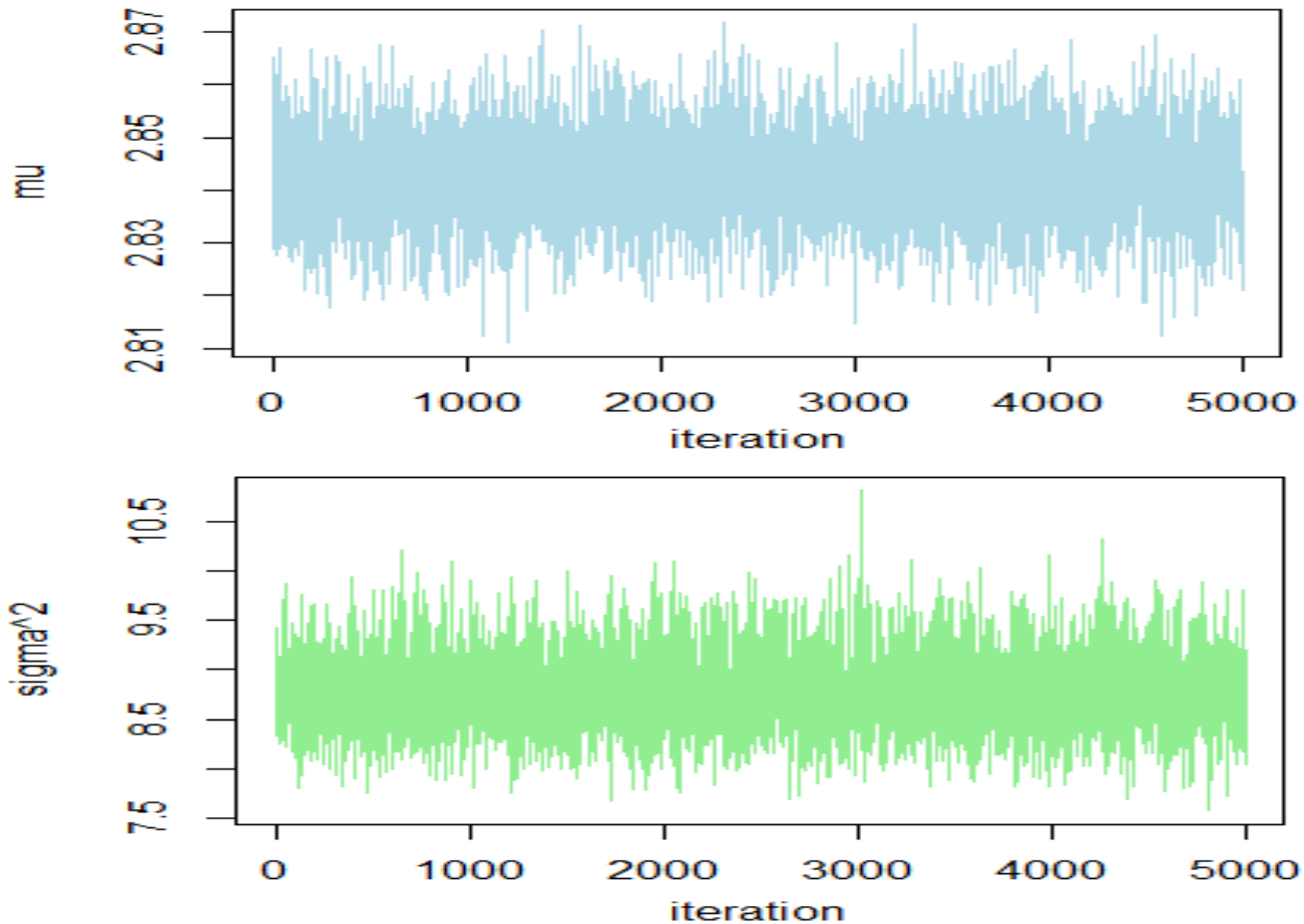
```
> tau<-1
> m<-mean(y)
```

The Gibbs Sampler code, for 5000 simulations, is as follows:

```
> ns<-5000
> Mu<-Sigma2<-rep(0,ns) # store parameters' updates

> for (i in 1:ns) {
>   mu<- Mu[i]<-rnorm(1,m,1/(n*tau))
>   tau<-rgamma(1,n/2,sum((y-mu)^2)/2)
>   Sigma2[i]<-1/tau
> }
```

The convergence of both parameters is checked graphically:



By the trace, we can see that there is no need of burn-in, and the estimates of μ and σ^2 correspond to the mean of the simulated parametrs over the 5000 iterations:

```
> mean(Mu[1:ns])
[1] 2.842592
> sd(Mu[1:ns])
[1] 0.008841489
```

$\hat{\mu} = 2.843 \approx 3$

```
> mean(Sigma2[1:ns])
[1] 8.792755
> sd(Sigma2[1:ns])
[1] 0.3968448
```

$\hat{\sigma}^2 = 8.793 \approx 9$

Acknowledgements:

Antónia Turkman (DEIO – FCUL) and Carina Silva-Fortes (ESTeSL – IPL), for allowing the use of some material produced by us in previous courses.

Bibliography:

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. *IEEE Trans. Pat. Anal. Mach. Intel.* 6, 721–741.

Sorensen, D. and Gianola, D. (2002). *Likelihood, Bayesian, and MCMC Methods in Quantitative Genetics*. New York: Springer.

Walsh, B. (2004). *Lecture Notes for EEB 581*, v. 26th April.