Working Paper Series

10/2007

Bayesian analysis and Markov chain Monte Carlo simulation

Medova, E.A.

CAMBRIDGE
Judge Business School

Author contact details are as follows:

Dr E A Medova
Centre for Financial Research
Judge Business School
University of Cambridge
eam28@cam.ac.uk

Please address enquiries about the series to:

Research Support Manager
Judge Business School
Trumpington Street
Cambridge CB2 1AG, UK
Tel: 01223 760546 Fax: 01223 339701
E-mail: research-support@jbs.cam.ac.uk

Bayesian Analysis and Markov Chain Monte Carlo Simulation

Elena Medova

Centre for Financial Research, Judge Business School, University of Cambridge
Trumpington Street, Cambridge CB2 1AG, UK
&
Cambridge Systems Associates Limited
5-7 Portugal Place, Cambridge CB5 8AF, UK

Tel: +44 1223 339593 Fax: +44 1223 339652
Email: eam28@cam.ac.uk Web: www-cfr.jbs.cam.ac.uk

Overview of main concepts

Bayesian analysis offers a way of dealing with information conceptually different from all other statistical methods. It provides a method in which observations are used to update estimates of the unknown parameters of a statistical model. With the Bayesian approach we start with a parametric model that is adequate to describe the phenomenon we wish to analyze. Then we assume a *prior distribution* for the unknown parameters of the model θ which represent our previous knowledge or belief about the phenomenon before observing any data. After observing some data assumed to be generated by our model we update these assumptions or beliefs. This is done by applying *Bayes' theorem* to obtain a *posterior probability density* for the unknown parameters given by

$$p(\theta \mid x) = \frac{p(x \mid \theta) p(\theta)}{\int p(x \mid \theta) p(\theta) d\theta},$$

where θ is the vector of unknown parameters governing our model, $p(\theta)$ is the *prior sampling density* function of θ and x is a sample drawn from the "true" underlying distribution with *sampling density* $p(x \mid \theta)$ that we model. Thus the posterior distribution for θ takes into account both our prior distribution for θ and the observed data x.

A *conjugate prior family* is a class of densities $\{ p(\theta_i) \}$ which has the feature that given the sampling density $p(x|\theta)$ the posterior density $p(\theta_i|x)$ also belongs to the class. The name arises because we say that the prior $p(\theta_i)$ is *conjugate* to the sampling density considered as a *likelihood* function $p(x|\theta)$ for θ given x. The concept of conjugate prior as well as the term was introduced by Raiffa and Schlaifer [14].

After obtaining a posterior distribution for the parameters θ we can compute various quantities of interest such as integrals of the form

$$\int f(y)g(y;\theta)p(\theta|x)dyd\theta\,, \qquad (1)$$

where f is some arbitrary function and g is the probability density function describing a related parametric model. In general, because we are not assuming independence between the each of the individual parameters this integral is difficult to compute, especially if there are many parameters. This is the situation in which *Markov chain Monte Carlo* (MCMC) simulation is most commonly used.

The distinguishing feature of MCMC is that the random samples of the integrand in (1) are *correlated*, whereas in conventional Monte Carlo methods such samples are statistically independent. The goal of MCMC methods is to construct an ergodic Markov chain that converges quickly to its stationary distribution which is the required posterior density or some functional thereof such as (1).

One can broadly categorize the use of MCMC methods as Bayesian or non-Bayesian. *Non-Bayesian* MCMC methods are used to compute quantities that depend on a distribution from a statistical model that is *non-parametric*. In a *Bayesian* application we consider a *parametric* model for the problem of interest. We assume some prior distribution on the parameters and try to compute quantities of interest that involve the posterior distributions. This approach remains suitable if the data is sparse, for example, in extreme value applications [10].

There are many different types of MCMC algorithms. The two most basic and widely used are the *Metropolis-Hastings algorithm* and the *Gibbs sampler* which we will now review.

Metropolis-Hastings algorithm

The Metropolis-Hastings algorithm [11, 8, 4] has been used extensively in physics but was little known to others until Müller [12] and Tierney [19] expounded the value of this algorithm to statisticians. The algorithm is extremely powerful and versatile and has been included in a list of 'top 10 algorithms' [5] and even claimed to be most likely the most powerful algorithm of all time [1].

The Metropolis-Hastings algorithm can draw samples from any *target* probability density π for the uncertain parameters $\boldsymbol{\theta}$ requiring only that this density can be calculated at θ. The algorithm makes use of a *proposal density* $q(\theta^t, \zeta)$ which depends on the current state of the chain θ^t to generate each new proposed parameter sample ζ. The proposal ζ is 'accepted' as the next state of the chain $(\theta^{t+1} := \zeta)$ with *acceptance probability* $\alpha(\theta^t, \zeta)$ and 'rejected' otherwise. It is the specification of this probability α that allows us to generate a Markov chain with the desired target stationary density π. The Metropolis-Hastings algorithm can thus be seen as a generalized form of *acceptance/rejection sampling* with values drawn from approximate distributions which are 'corrected' in order that they behave asymptotically as random observations from the target distribution.

The algorithm in step-by-step form is as follows:

a) Given the current position of our Markov chain θ^t, generate a new value ζ from the proposal density q (see below).

b) Compute the *acceptance* probability

$$\alpha\left(\theta^t, \zeta\right) := \min\left(1, \frac{\pi(\zeta) q\left(\zeta, \theta^t\right)}{\pi\left(\theta^t\right) q\left(\theta^t, \zeta\right)}\right), \qquad (2)$$

where π is the density of the target distribution.

c) With probability $\alpha(\theta^t, \zeta)$, set $\theta^{t+1} := \zeta$, else set $\theta^{t+1} := \theta^t$.

d) Return to step a).

This algorithm generates a discrete time ergodic Markov chain $\left(\boldsymbol{\theta}^t\right)_{t \geq 0}$ with stationary distribution Π corresponding to π, i.e. as $t \to \infty$

$$P\left(\boldsymbol{\theta}^t \in B\right) \to \Pi\left(B\right)$$

for all suitably (Borel) measurable sets $B \in \mathbb{R}^n$.

Some important points to note [4]:

- We need to specify a starting point θ^0, which may be chosen at random (and often is). Preferably θ^0 should coincide with a mode of the density π.

- We should also specify a *burn-in period* to allow the chain to reach equilibrium. By this we mean that we discard the first n values of the chain in order to reduce the possibility of bias caused by the choice of the starting value θ^0 .

- The *proposal distribution* should be a distribution that is easy to sample from. It is also desirable to choose its density q to be 'close' or 'similar' to the target density π, as this will increase the acceptance rate and increase the efficiency of the algorithm.

- We only need to know the *target density function* π up to proportionality — that is, we do not need to know its normalizing constant, since this cancels in the calculation (2) of the *acceptance function* α.

The choice of the burn-in period still remains somewhat of an art, but is currently an active area of research. One can simply use the 'eyeballing technique' which merely involves inspecting visual outputs of the chain to see whether or not it has reached equilibrium.

When the proposal density is *symmetric*, i.e. $q(\theta^t, \zeta) = q(\zeta, \theta^t)$ (the original Metropolis algorithm), the computation of the acceptance function α is significantly faster. In this case from (2) a proposal ζ is accepted with probability $\alpha=\pi(\zeta)/\pi(\theta^t)$, i.e. its likelihood $\pi(\zeta)$ relative to that of $\pi(\theta^t)$ (as originally suggested by Ulam for acceptance/rejection sampling).

Random walk Metropolis

If $q(\theta^t, \zeta) := f(|\theta - \zeta|)$ for some density f and norm $|\cdot|$ then this case is called a *random walk* chain because the proposed states are drawn according to the process following $\boldsymbol{\zeta} = \theta^t + \boldsymbol{\nu}$,

where $\nu \sim F$, the distribution corresponding to *f*. Note that since this proposal density q is symmetric the acceptance function is of the simple Metropolis form described above. Common choices for q are the multivariate normal, multivariate t or the uniform distribution on the unit sphere.

If $q(\theta^t, \zeta) := q(\zeta)$ then the candidate observation is drawn *independently* of the current state of the chain. Note however that the state of the chain θ^{t+1} at $t+1$ does depend on the previous state θ^t because the acceptance function $\alpha(\theta^t, \cdot)$ depends on θ^t.

In the random walk chain we only need to specify the *spread* of q, i.e. a maximum for $|\theta - \zeta|$ at a single step. In the independence sampler we need to specify the spread and the location of q.

Choosing the spread of q is also something of an art. If the spread is large, then many of the candidates will be far from the current value. They will therefore have a low probability of being accepted, and the chain may remain stuck at a particular value for many iterations. This can be especially problematic for multi-modal distributions; some of its modes may then not be explored properly by the chain. On the other hand, if the spread is small the chain will take longer to traverse the support of the density and low probability regions will be under-sampled. The research reported in [13] suggests an optimal acceptance rate of around 0.25 for the random walk chain. In the case of the independence sampler it is important [2] to ensure that the tails of q dominate those of π, otherwise the chain may get stuck in the tails of the target density. This requirement is similar to that in importance sampling.

Multiple-block updates

When the number of dimensions is large it can be difficult to choose the proposal density q so that the algorithm converges sufficiently rapidly. In such cases it is helpful to break up the space into smaller blocks and to construct a Markov chain for each of these smaller blocks [8]. Suppose that we split θ into two blocks (θ_1, θ_2) and let $q_1\left(\theta_1^t \middle| \theta_2^t, \zeta_1\right)$ and $q_2\left(\theta_2^t \middle| \theta_1^t, \zeta_2\right)$ be the proposal densities for each block. We then break each iteration of the Metropolis-Hastings algorithm into 2 steps and at each step we update the corresponding block. To update block 1 we use the acceptance function given by

$$\alpha\left(\theta_1^t \middle| \theta_2^t, \zeta_1\right) := \min\left(1, \frac{\pi\left(\zeta_1 \middle| \theta_2^t\right) q_1\left(\zeta_1 \middle| \theta_2^t, \theta_1^t\right)}{\pi\left(\theta_1^t \middle| \theta_2^t\right) q_1\left(\theta_1^t \middle| \theta_2^t, \zeta_1\right)}\right) \tag{3}$$

and to update block 2 we use

$$\alpha\left(\theta_2^t \middle| \theta_1^t, \zeta_2\right) := \min\left(1, \frac{\pi\left(\zeta_2 \middle| \theta_1^t\right) q_2\left(\zeta_2 \middle| \theta_1^t, \theta_2^t\right)}{\pi\left(\theta_2^t \middle| \theta_1^t\right) q_2\left(\theta_2^t \middle| \theta_1^t, \zeta_2\right)}\right). \tag{4}$$

If the blocks each consist of just a single variable, then the resulting algorithm is commonly called the *single-update* Metropolis-Hastings algorithm. Suppose in the single-update algorithm it turns out that each of the marginals of the target distribution $\pi\left(\theta_i \middle| \theta_{\sim i}\right)$ can be directly sampled from. Then we would naturally choose $q\left(\theta_i \middle| \theta_{\sim i}\right) := \pi\left(\theta_i \middle| \theta_{\sim i}\right)$ since all candidates ζ will then be accepted with probability 1. This special case is the well-known *Gibbs sampler* [2].

Gibbs sampler

Gibbs sampling is applicable in general when the joint parameter distribution is not known explicitly but the conditional distribution of each parameter given the others is known. Let $P(\theta) = P(\theta_1, ..., \theta_k)$ denote the joint parameter distribution and let $p\left(\theta_i \mid \theta_{\sim i}\right)$ denote the conditional density for the i-th component θ_i given the other $k-1$ components, where $\theta_{\sim i} := \left\{\theta_j : j \neq i\right\}$ for $i = 1, ..., k$. Although we do not know how to sample directly from P we do

know how to sample directly from each $p\left(\theta_i \mid \theta_{\sim i}\right)$. The algorithm begins by picking the arbitrary starting value $\theta^0 = \left(\theta_1^0, ..., \theta_\kappa^0\right)$. It then samples randomly from the conditional densities $p\left(\theta_i \mid \theta_{\sim i}\right)$ for $i = 1, ..., k$ successively as follows:

Sample θ_1^1 from $p\left(\theta_1 \mid \theta_2^0, \theta_3^0, ..., \theta_k^0\right)$

Sample θ_2^1 from $p\left(\theta_2 \mid \theta_1^1, \theta_3^0, ..., \theta_k^0\right)$

…

Sample θ_k^1 from $p\left(\theta_k \mid \theta_1^1, \theta_2^1, ..., \theta_{k-1}^1\right)$.

This completes a transition from θ^0 to θ^1 and eventually generates a sample path $\theta^0, \theta^1, ..., \theta^t, ...$ of a Markov chain whose stationary distribution is *P*.

In many cases we can use the Gibbs sampler which is significantly faster to compute than the more general Metropolis-Hastings algorithm. In order to use Gibbs however we me must know how to directly sample from the conditional posterior distributions for each parameter, i.e. $p\left(\theta_i \middle| \theta_{\sim i}, x\right)$, where *x* represents the data to time *t*.

Use of MCMC in capital allocation for operational risk

Due to lack of reported data on operational losses Bayesian Markov chain Monte Carlo simulation is well suited for the quantification of operational risk and operational risk capital allocation. In [9] a framework for evaluation of *extreme* operational losses has been developed which assumes that market and credit risks may be managed separately but jointly impose a *value at risk* limit u_{VaR} on these risks.

It is assumed that losses beyond the u_{VaR} level belong to the *operational risk* category. In most cases, due to overlapping between risk types a detailed analysis of operational loss data is required to support the assumption that the u_{VaR} level approximately equals the *unexpected loss threshold.* This approach to capital allocation for operational risk takes into account large but rare operational losses, is naturally based on *extreme value theory* (EVT) [6,7] and focusses on

tail events and modelling the worst-case losses as characterized by loss maxima over regular observation periods.

According to regulatory requirements [20] operational risk capital calculation requires two distributions – a *severity* distribution of loss values and a *frequency* distribution of loss occurrences. In the approach described here a unified resulting asymptotic model known as the *peaks over threshold* (POT) model [15,16,18] is applied. It is based on an asymptotic theory of extremes and a point process representation of exceedances over a threshold given by the POT model. The following is assumed.

Given an *i.i.d.* sequence of random losses $\boldsymbol{X}_1,\ldots,\ \boldsymbol{X}_n$ drawn from some distribution we are interested in the distribution of the *excess* $\boldsymbol{Y} := \boldsymbol{X} - u$ over the threshold u. The distribution of excesses is given by the conditional distribution function in terms of the tail of the underlying distribution function F as

$$F_u(y) := P(\boldsymbol{X} - u \le y \mid \boldsymbol{X} > u) = \frac{F(u+y) - F(u)}{1 - F(u)} \text{ for } 0 \le y \le \infty . \qquad (5)$$

The limiting distribution $G_{\xi,\beta}(y)$ of excesses as $u \to \infty$ is known as the *generalized Pareto distribution* (GPD) with *shape* parameter ξ and *scale* parameter β given by

$$G_{\xi,\beta}(y) = \begin{cases} 1 - \left(1 + \xi \dfrac{y}{\beta}\right)^{-1/\xi} & \xi \neq 0 \text{ where } y \in [0, \xi] \ \ \xi \ge 0 \text{ or } y \in [0, -\beta/\xi] \ \ \xi < 0 \\ 1 - \exp\left(-\dfrac{y}{\beta}\right) & \xi = 0 . \end{cases} \qquad (6)$$

The identification of an appropriate threshold u is again somewhat of an art and requires a data analysis based on a knowledge of EVT [3, 6, 7].

The *capital provision* for operational risk over the unexpected loss threshold u is given in [10] as

$$\lambda_u E(\boldsymbol{X} - u \mid \boldsymbol{X} > u) = \lambda_u \frac{\beta_u + \xi u}{1 - \xi}, \qquad (7)$$

where $E(\boldsymbol{X}-u \mid \boldsymbol{X}>u)=\frac{\beta_u+\xi u}{1-\xi}$ is the *expectation* of excesses over the threshold u (which is defined for $\xi \leq 1$ and must be replaced by the *median* for ξ>1), $\beta_u := \sigma+\xi(u-\mu)$ and the exceedances form a Poisson point process with *intensity*

$$\lambda_u := \left(1+\xi\frac{(u-\mu)}{\sigma}\right)^{-1/\xi}, \tag{8}$$

usually measured in days per annum.

The accuracy of our model depends on accurate estimates of the ξ, μ, σ and β parameters. To address this, hierarchical Bayesian Markov chain Monte Carlo simulation is used to determine the parameter estimates of interest through intensive computation. The empirical estimation efficiency of this method when back-tested on large data sets is surprisingly good.

Hierarchical Bayesian parameter estimation considers the parameters to be random variables possessing a joint probability density function. The prior density $f_{\theta|\psi}$ of the random parameter vector θ is parametric with a vector of random *hyper-parameters* ψ and is conjugate prior to the sampling density $f_{X|\theta}$ so that the calculated posterior density $f_{\theta|X_1,\ldots,X_n,\psi} := f_{\theta|\psi+}$ is of the same form with the new hyper-parameters $\psi+$ determined by ψ and the observations $X_1,\ldots,X_n$. In the hierarchical Bayesian model the *hyper-hyper parameters* φ are chosen to generate a vague prior due to the lack of a prior distribution for the hyper-parameters before excess loss data is seen. Hence, we can decompose the posterior parameter density $f_{\theta|X,\psi}$ with the observations X and the initial hyper-hyper parameters φ as

$$\begin{aligned} f_{\theta|X,\psi} &\propto f_{X|\theta}(X \mid \theta) f_{\theta|\psi}(\theta \mid \psi) f_{\psi}(\psi \mid \varphi) \\ &\propto f_{X|\theta}(X \mid \theta) f_{\psi|\theta}(\psi \mid \theta, \varphi) \\ &\propto f_{X|\theta}(X \mid \theta) f_{\psi}(\psi \mid \varphi+). \end{aligned}$$

Here the Bayesian update of the prior parameter density $f_\theta \propto f_{\theta|\psi} f_\psi$ is performed in 2 stages. First by updating the hyper-hyper parameters φ to $\varphi+$ conditional on θ then evaluating the corresponding posterior density for this θ given the observations X.

Hierarchical Bayesian MCMC simulation for the parameters is based on the Metropolis-Hasting algorithm described briefly above and in detail in [17]. The idea is that the state of the chain for the parameter vector $\theta := \{\mu_j, \log \sigma_j, \xi_j : j = 1, 2,\ldots, J\}$ converges to a stationary distribution which is the Bayesian posterior parameter distribution $f_{\theta|x,\psi}$ given the loss data x and a vector ψ of *hyperparameters* $\{ m_\mu , s^2_\mu, m_{log\sigma}, s^2_{log\sigma}, m_\xi, s^2_\xi \}$. The hyperparameters are sampled from a conjugate prior *gamma-normal* (GM) distribution and are used to link the parameters $\{\mu_j, \sigma_j, \xi_j : j = 1, 2,\ldots, J\}$ of each individual risk [10].

The aim of the model is to estimate the parameters of interest $\{\mu_j, \sigma_j, \xi_j : j = 1, 2,\ldots, J\}$ conditional on both the data x and the hyperparameters $\{ m_\mu , s^2_\mu, m_{log\sigma}, s^2_{log\sigma}, m_\xi, s^2_\xi \}$. The *posterior* distributions of the parameters are normally distributed:

$$\mu_j \sim \mathrm{N}(m_\mu , s^2_\mu), \log \sigma_j \sim \mathrm{N}(m_{log\sigma}, s^2_{log\sigma}) \text{ and } \xi_j \sim \mathrm{N}(m_\xi, s^2_\xi).$$

A schematic summary of the loss data, parameters and hyperparameters is given in Table 1.

Table 1: *Bayesian hierarchical model*

Data ***x***	*Type 1*	*Type 2*			*Type J*
Business Unit 1	x_{11}	x_{12}	.	.	x_{1J}
Business Unit 2	x_{21}	x_{22}	.	.	x_{2J}
	.	.	.	.	.
	.	.	.	.	.
Business Unit n	$x_{n,1}$	$x_{n,2}$	.	.	$x_{n,J}$

Parameters	θ				*Hyperparameters* ψ
Mean (μ)	μ_1	μ_2	. . .	μ_J	Mean – m_μ Variance – $s^2{}_\mu$
Scale ($\log\sigma$)	Log σ_1	log σ_2	. . .	log σ_J	Mean – $m_{log\sigma}$ Variance – $s^2{}_{log\sigma}$
Shape (ξ)	ξ_1	ξ_2	. . .	ξ_J	Mean – m_ξ Variance – $s^2{}_\xi$

Illustrative example

The data is assumed to represent the operational losses of a bank attributable to three different business units. The data starts on 03.01.1980 and ends on 31.12.1990. The time span is calculated in years hence the parameters will also be measured on a yearly basis. The data has been generated from the Danish insurance claims data [3] by two independent random multiplicative factors to obtain the three sets of loss data summarized in Table 2.

A typical analysis of such data includes time series plots, log histogram plots, sample mean excess plots, QQ plots for extreme value analysis against the GPD, Hill estimate plots of the shape parameter and plots of the empirical distribution functions. All these tests have been performed for the three data sets to conclude that data are heavy tailed and that the POT model is valid.

Table 2: *Summary statistics for data*

Data	X_1 (Danish)	X_2	X_3
Min:	1.000	0.8	1.200
1^{st} Qu.	1.321	1.057	1.585
Mean	3.385	2.708	4.062
Median	1.778	1.423	2.134
3rd Qu	2.967	2.374	3.560
Max	263.250	210.600	315.900
N	2167	2167	2167
StdDev	8.507	6.806	10.209

Figure 1: *Time series of log 'danish' data X_1*

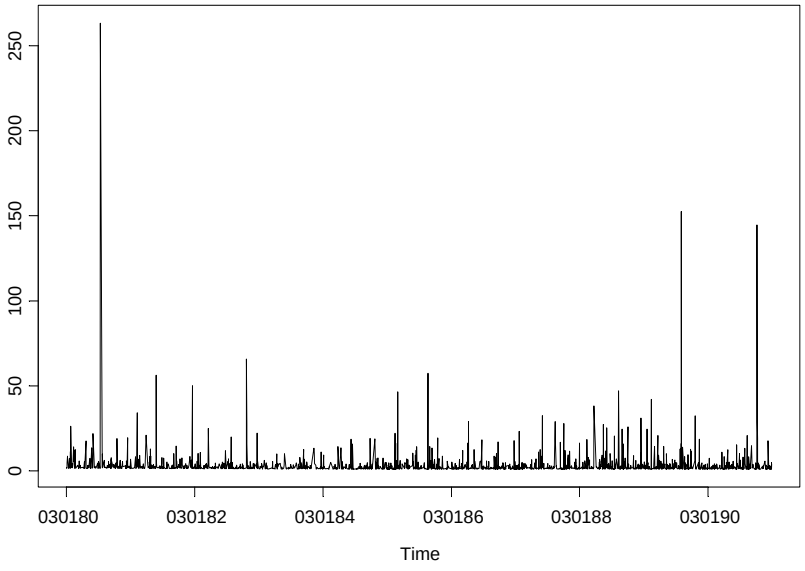
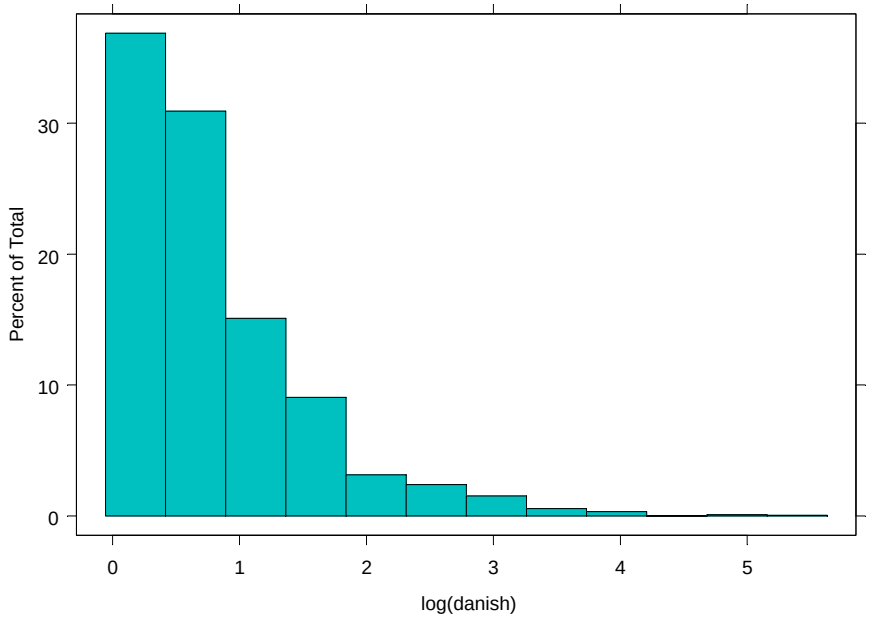


Figure 2: *Histogram of log 'danish' data*



Inputs for the MCMC model:

Threshold $u = 30$

Initial parameters:

μ_1	μ_2	μ_3	$\log\sigma_1$	$\log\sigma_2$	$\log\sigma_3$	ξ_1	ξ_2	ξ_3
20	21	22	3	3.2	2.8	0.5	0.4	0.7

The tables below are a summary of the posterior mean estimates of the parameter values β_i and λ_i based on the MCMC posterior distribution mean parameter values.

For j = 1 (Unit 1)

Code	Mean(μ_1)	Mean($\log\sigma_1$)	Mean(ξ_1)	β_1	λ_1	Expected Excess
1000 loops	37.34	3.14	0.77	18.46	1.41	180.70
2000 loops	36.89	3.13	0.80	18.35	1.39	**211.75**
The number of exceedances above threshold is 15.						

For j = 2 (Unit 2)

Code	Mean (μ_2)	Mean($\log\sigma_2$)	Mean(ξ_2)	β_2	λ_2	Expected Excess
1000 loops	36.41	3.16	0.77	19.04	1.34	186.22
2000 loops	35.76	3.13	0.8	18.5	1.30	**218.40**
The number of exceedances above threshold is 11.						

For j = 3 (Unit 3)

Code	Mean (μ_3)	Mean($\log\sigma_3$)	Mean(ξ_3)	β_3	λ_3	Expected Excess
1000 loops	39.55	3.05	0.79	14.21	1.71	180.52
2000 loops	39.23	3.03	0.82	13.83	1.70	**213.50**
The number of exceedances above threshold is 24.						

The plots in Figure 3 below for the results of 2000 simulation loops show that convergence has been reached for the marginal posterior distributions of all parameters for Unit 1 and that the estimates of these parameters are distributed approximately normally. (Those of σ are thus approximately lognormal.) Similar results hold for the other two units.

The capital provision for operational losses is calculated using expression (7). The probability of such losses is given by the choice of threshold u for extreme operational losses. This threshold must be obtained from an analysis of the historical operational loss data and should agree or exceed the threshold level u_{VaR} of unexpected losses due to market and credit risk. The probability of crossing the combined market and credit risk threshold u_{VaR} is chosen according to the usual *value at risk* (VAR) risk management procedures. The level of losses u due to operational risks is exceeded with probability $\rho \leq \pi$, so that $u \geq u_{VaR}$. The probability of exceeding u depends on the shape of the tail of the loss distribution but is in general very much smaller than π.

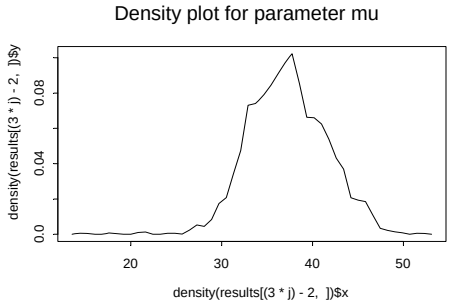
Assuming that three types of losses are the bank business unit losses from operational risk over a period of 11 years the bank should hedge its operational risk for these units by putting aside 944.60 units of capital (1.39 x 211.75 + 1.30 x 218.40 + 1.70 x 213.50) for any *one year period*.

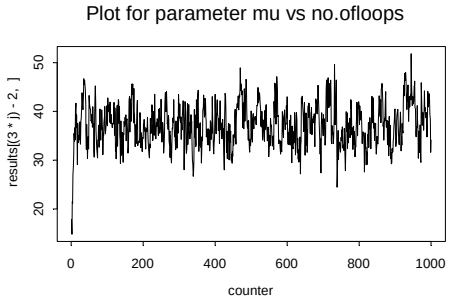
Although in this illustrative example unexpected losses above the combined VaR level (30 units of capital) occur with probability 2.5% per annum, unexpected operational risk losses will exceed this capital sum with probability less than 0.5%. In practice lower tail probabilities might be chosen, but similar or higher probability ratios would obtain.

Note that in this example the loss data for each business unit was generated as independent and the total capital figure takes the resulting diversification effect into account. On actual loss data the dependencies in the realized data are taken into account by the method and the diversification effect of the result can be analyzed by estimating each unit separately and adding the individual capital figures (which conservatively treats losses across units as perfectly correlated) [10]. Although the results presented here are based on very large (2167) original sample sizes, the simulation experiments on actual banking data reported in [10] verify the high accuracy of MCMC Bayesian hierarchical methods for exceedance sample sizes as low as 10 and 25, as in this example.

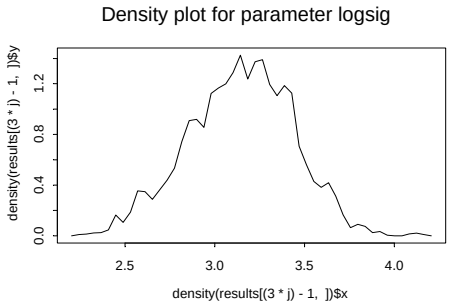
Figure 3: *Simulation results of MCMC for 200k iterations*

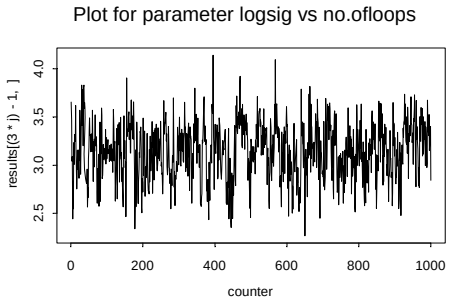
μ_1



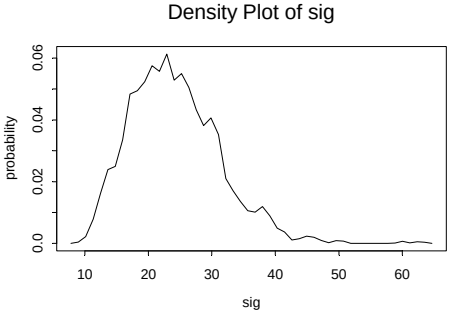


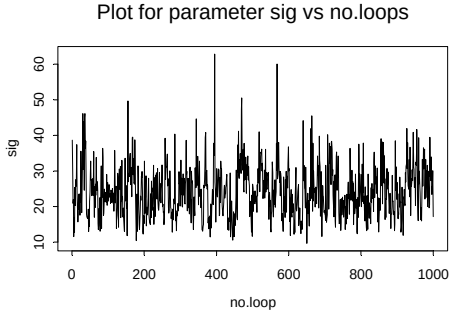
$\log\sigma_1$



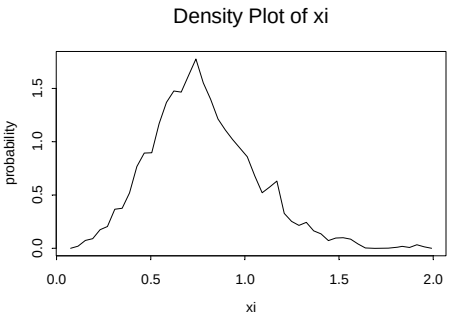


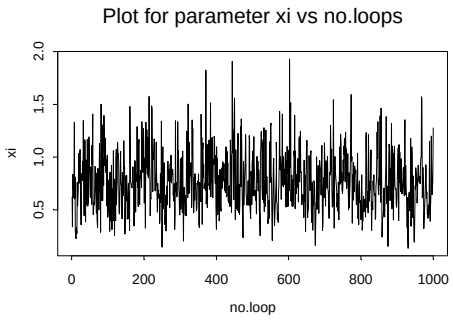
σ_1





ξ_1





Conclusion

In this chapter we have introduced Markov chain Monte Carlo (MCMC) concepts and techniques and shown how to apply them to the estimation of a Bayesian hierarchical model of interdependent extreme operational risks. This model employs the peaks over threshold (POT) model of extreme value theory (EVT) to generate both *frequency* and *severity* statistics for the extreme operational losses of interdependent business units which are of interest at the board level of a financial institution. These are obtained respectively in terms of Poisson exceedences of an unexpected loss level for other risks and the generalized Pareto distribution (GPD).

The model leads to annual business unit capital allocations for unexpected extreme risks which take account of the statistical interdependencies of individual business unit losses.

The concepts discussed in this chapter are illustrated by an artificially created example involving three business units but actual banking studies are described in [10] and in forthcoming work relating to internally collected operational loss data.

References

[1] Beichl I., Sullivan F. (2000). The Metropolis algorithm, *Computing in Science & Engineering* **2**(1): 65–69.

[2] Casella G., Edward I.G. (1992). Explaining the Gibbs sampler, *The American Statistician* **46**: 167-17.

[3] Castillo E. (1988). *Extreme Value Theory in Engineering*. Academic Press, Orlando.

[4] Chib S., Greenberg E. (1995). Understanding the Metropolis-Hastings algorithm, *The American Statistician* **49**(4): 327–335.

[5] Dongarra J., Sullivan F. (2000). The top 10 algorithms, *Computing in Science and Engineering* **2**(1): 22–23.

[6] Embrechts P., Kluppelberg C. & Mikosch T. (1997). *Modelling Extremal Events*, Springer, Berlin.

[7] Galambos J. (1978). *The Asymptotic Theory of Extreme Order Statistics*, Wiley, New York.

[8] Hastings W. (1970). Monte Carlo sampling methods using Markov chains and their applications, *Biometrika* **57**(1): 97–109.

[9] Medova E.A. (2001). Operational risk capital allocation and integration of risks, in *Advances in Operational Risk: Firmwide issues for financial institutions*. Risk Books pp.115-127.

[10] Medova E.A., Kyriacou M.N. (2002). Extremes in operational risk measurement, in *Risk Management: Value At Risk And Beyond*, M.A.H. Dempster, ed., Cambridge University Press, pp. 247-274.

[11] Metropolis N., Rosenbluth A., Rosenbluth M., Teller A., Teller E. (1953). Equations of state calculations by fast computing machines, *Journal of Chemical Physics* **21**(1): 1087–1092.

[12] Muller P. (1993). A generic approach to posterior integration and Gibbs sampling, Technical Report, Purdue University.

[13] Roberts G., Gelman A., Gilks W. (1994). Weak convergence and optimal scaling of random walk Metropolis algorithms, Technical Report, University of Cambridge.

[14] Raiffa H., Schlaifer R. (1961). *Applied Statistical Decision Theory*, Harvard University Press.

[15] Leadbetter M., LindGreen G., Rootzen H. (1983). *Extremes and Related Properties of Random Sequences and Processes*, Springer, Berlin.

[16] Leadbetter M. (1991). On a basis for 'Peaks over Threshold' modeling, *Statistics and Probability Letters* **12**: 357-362.

[17] Smith A, Roberts G. (1993). Bayesian computation via the Gibbs sampler and related Markov chain Monte Carlo methods, *J. Royal Statistical Society*, **B55**: 3-23.

[18] Smith R. (2001). Measuring risk with extreme value theory, Chapter 8 in *Risk Management: Value At Risk And Beyond*, M.A.H. Dempster, ed., Cambridge University Press.

[19] Tierney L. (1994). Markov chains for exploring posterior distributions, *Annals of Statistics* **22**:1701–1762.

[20] *The New Basel Capital Accord* (2001). Bank of International Settlements Press Release.