Bayesian Modelling of Outstanding Liabilities in Non-Life Insurance

August 9, 2011

Abstract

This dissertation focuses on the stochastic modelling of outstanding liabilities in non-life insurance, using Bayesian Statistics. Credible intervals and various statistical estimates can then be derived, whereas this is not the case with the numerical methods commonly used in the industry, such as the Chain Ladder, which only give point estimates. We ignore the claim numbers and concentrate only on claim amounts. The modelling requires intensive use of Bayesian methodology and Monte Carlo Markov chain (MCMC) techniques through the WinBUGS package. We also show that the results obtained are quite different from those with the Chain Ladder method, however there is no obvious way to define which method is better as more data would be necessary.

Contents

1 Introduction

In General Insurance, claim reserving is of paramount importance and is still a subject of active research. In certain areas of insurance, such as motor insurance and property insurance, claims are reported quite quickly to the insurer and are settled directly. However, in certain cases, and for certain types of insurance products, the claims occurred may not be reported quickly, and one may even ignore their existence for a long period of time. For example, in Employer Liability Insurance, some workers can be exposed now to some health-threatening substances that are not yet known to be dangerous, as it was the case for asbestosis until recently. However, health problems implied by those substances would be considered to be caused by work conditions and the insurer would have to pay for them. In weather related insurance, the total sum of the damages may be tricky to be evaluated and some time may be needed. Even in a car accident, the liability of the policyholders involved could be hard to determine and a trial could be needed to seal the issue, requiring some delay between the occurrence of the claim and its settlement. Thus, insurance companies have to develop methods to assess the loss they will suffer in the future due to policies originated in previous years. This would enable them to set up reserves now to be able to meet their future liabilities.

For example, a company would like to know an estimate of the amount it will still have to pay in the next *5* years due to policies originating in a specific past year, or the amount it will have to pay in the coming year for all the policies originating in the past. The year in which the policy was in force is known as the origin year, or accident year, and the year in which a claim is settled is called the development year. For example, consider a car accident taking place in 2005. For some reason, the claim is settled in 2008 by the insurance company. The accident year is then 2005, and the development year is 2008, which will be called year *4* (2005 is development year *1*, so 2008 is development year *4*).

Using a matrix representation with l accident years and l development years, we will call y_{ij} the amount "paid" in development year j for all policies in force in accident year i (i.e. paid j-1 years after their originating year). If $i+j > l+1$, then y_{ij} will have to be estimated, which is the core of this dissertation. In the following table, the empty cells repesent these unknown quantities that we want to estimate.

Accident / Development Year	*1*	*2*	*3*	*4*
1	y_{11}	y_{12}	y_{13}	y_{14}
2	y_{21}	y_{22}	y_{23}	
3	y_{31}	y_{32}		
4	y_{41}			

Several methods are used in practice to estimate y_{ij} for i, j such that $i+j>l+1$: the chain ladder, the average cost per claim, and the Bornhuetter-Ferguson ones are the most common [4, 5]. Given the data that we will study, only the claim amounts will be known and we will therefore concentrate on the Chain Ladder method. However, all these recipes are not statistically based. Consequently, no confidence intervals, standard deviations or other statistical tools are available with these methods which only give point estimates. To remedy these problems, we will build statistical models to represent the claim amounts, and estimate the unknown parameters from the data by using Bayesian statistics and Markov chain Monte Carlo (MCMC) algorithms. Several models have already been proposed (see [9, 6] for example), some using Bayesian Statistics (see [19, 20]).

Section 2 will briefly review the concepts of Bayesian Statistics and MCMC methods. It sets the theoretical background needed for the remainder of this dissertation. In Section 3, two different stochastic models of the claim amounts will be presented, with a brief discussion of their respective characteristics. Then, Section 4 will help us select the model that fits the data better, and the goodness of fit of the best one will be checked. Finally, Section 5 explains how to predict future values, and compares the results obtained with those derived using the Chain Ladder method.

2 Theoretical Background

2.1 Bayesian Statistics

In statistics, the use of Bayesian Inference has become more and more common over the last *20* years, as shown by the creation of The International Society for Bayesian Analysis (ISBA) in 1992 “to promote the development and application of Bayesian analysis useful in the solution of theoretical and applied problems in science, industry and government” [13]. Until then, Classical Statistics was the standard approach, but several limitations have been pointed out and one of the most important ones comes from its definition: in classical statistics, the only definition of the probability of an event is its long run frequency in a sequence of independent trials [1]. Thus, the probability that Mr X voted Conservative in the last elections is meaningless. However, one would like to be able to define such a “subjective probability”, which is possible with the use of Bayesian Statistics.

Bayesian Statistics is getting used extensively in Actuarial Science as well, and we therefore want to remind the reader of the basics, which will be used throughout this dissertation.

In Bayesian Statistics, unknown quantities are treated as random variables. Suppose that the distribution of a continuous random variable Y (for example the random loss variable) depends on a parameter θ. We consider θ as a realisation of a random variable Θ, which has a statistical distribution called the prior distribution. Its probability density function will be denoted by $\pi(\theta)$. The conditional density function of Y given θ is denoted $f_Y(y|\theta)$.

Based on the observed data $Y = y$, the distribution of Θ is updated, and we get the distribution of $\Theta|y$, called the posterior distribution. We then calculate an estimate of the random loss by taking the mean of the posterior distribution so as to minimize the quadratic loss function [5].

From a mathematical point of view, we have (see [12] for more details):

$$f_{\Theta Y}(\theta, y) = f_{Y|\Theta}(y|\theta)\pi(\theta)$$

where $f_{\Theta Y}(\theta, y)$ is the joint density function of Θ and Y. By integrating on Θ we get the marginal distribution of Y:

$$f_Y(y) = \int_\theta f_{Y|\Theta}(y|\theta)\pi(\theta)d\theta$$

Moreover,

$$f_{\Theta Y}(\theta, y) = \pi(\theta|y)f_Y(y)$$

so we finally get

$$\pi(\theta|y) = \frac{\pi(\theta)f_Y(y|\theta)}{\int f(y|\theta')\pi(\theta')d\theta'}. \tag{1}$$

The denominator being an integral over the range of θ', it depends only on y and we can then write:

$$\pi(\theta|y) \propto \pi(\theta)f_Y(y|\theta).$$

The denominator is in fact a normalizing constant, so that $\pi(\theta|y)$ is well defined as a distribution. Notice that if the posterior distribution belongs to the same family as the prior distribution, then the latter is called a *conjugate prior distribution*.

If no information is known a priori on θ, one would like to have a prior distribution that represents lack of knowledge. Such a prior distribution is known as *vague* or *non-informative*, however, no prior really reflects complete ignorance. Jeffreys' prior is aimed at providing us with non-informative priors (see [1] for more details), even though any prior sufficiently constant over a large range of values could be considered as well.

Another important point, that will be quite useful throughout this dissertation, is the predictive distribution. Indeed, one of our main concern will be to predict the values taken by another quantity of interest, whose distribution depends on θ as well. This quantity will be called z in this subsection. Once we know $\pi(\theta|y)$, we want to obtain the distribution of $f(z|y)$. We write it as "a *mixture distribution* over the possible values of θ" [1]:

$$f(z|y) = \int f(z|y,\theta)\pi(\theta|y)d\theta = \int f(z|\theta)\pi(\theta|y)d\theta$$

since z and y are conditionally independent given the parameter vector θ.

However, the formal expressions given above are not always analytically tractable and some numerical methods may be needed to evaluate the results.

2.2 MCMC & WinBUGS

MCMC (Markov chain Monte Carlo) methods will enable us to explore posterior distributions by simulation, and are therefore particularly useful for multi-dimensional problems, or problems for which we do not use conjugate prior distributions. These methods have helped Bayesian Statistics develop a lot in recent years. WinBUGS is a software enabling us to perform such an analysis, as we will see throughout this dissertation.

Before explaining the algorithm, let us review quickly the basics of a Markov chain, with details available in [3]. It is a stochastic process $\{\theta^{(1)}, \theta^{(2)}, ..., \theta^{(n)}\}$ such that $\pi(\theta^{(t+1)}|\theta^{(t)}, ..., \theta^{(1)}) = \pi(\theta^{(t+1)}|\theta^{(t)})$ and $\pi(\theta^{(t+1)}|\theta^{(t)})$ is independent of time t. Moreover, for the distribution of $\theta^{(t)}$ to converge to its equilibrium distribution, which is independent of $\theta^{(0)}$, the Markov chain must be irreducible, aperiodic and positive-recurrent.

To sample from $\pi(\theta|y)$ we must construct a Markov chain whose stationary distribution is the posterior distribution $\pi(\theta|y)$. The following algorithm, called the Metropolis-Hastings algorithm, will help us in doing so:

- Set an initial value $\theta^{(0)}$
- Propose a new state θ' from a distribution $q(\theta'|\theta^{(0)})$, known as a proposal distribution
- The proposed state is accepted with probability $p = min(1, \frac{\pi(\theta'|y)q(\theta|\theta')}{\pi(\theta|y)q(\theta'|\theta)})$. Set $\theta^{(1)} = \theta'$ with probability p, otherwise set $\theta^{(1)} = \theta^{(0)}$
- Reiterate the procedure for a number of iterations so that the stationary distribution is reached.

According to Equation (1), we can simplify: $p = min(1, \frac{\pi(\theta')f_Y(y|\theta')q(\theta|\theta')}{\pi(\theta)f_Y(y|\theta)q(\theta'|\theta)})$. Notice that the normalizing constant $f_Y(y)$ does not appear in p, which makes this algorithm particularly suited for Bayesian Statistics. WinBUGS, in addition to the Metropolis-Hastings algorithm, also uses the Gibbs sampler, which can be viewed as a particular case of Metropolis-Hastings ([1], [26]).

An obvious “problem” with this algorithm is convergence: it does not automatically indicate when the stationary distribution is reached up to a certain percentage error. However, several diagnostics are used. For more information, see [10] for the Gelman-Rubin diagnostic and [14] for the Geweke convergence diagnostic.

Consequently, we will follow the method presented in [3] when using WinBUGS:

- Select an initial value $\theta^{(0)}$
- Generate n values until the stationary distribution is reached
- Monitor the convergence of the algorithm through different diagnostics
- Discard the first d observations (burn-in period).

Once the first d iterations have been discarded, we can assess the accuracy of posterior estimates by calculating their Monte Carlo error, which is an estimate of the difference between the estimated posterior mean and its true value. As a rule of thumb, a Monte Carlo Markov chain should be run until the Monte Carlo error for each parameter is less than 5% its standard deviation (see [25] and [26]). This condition will be achieved for all the simulations we will be doing.

It is also worth noticing that this algorithm does not provide us with an independent sample. As a consequence, the chain might mix poorly and we could obsverve considerable autocorrelation. Thinning, or subsampling, is a way to avoid this: by using only one value every r iterations the autocorrelation will be reduced (see [25] for more information).

3 Models for Outstanding Claim Amounts

The main variable $Y_{i,j}$, representing the stochastic amount of outstanding claim for accident year i and development year j, will be modelled in two different ways: first by using a Lognormal model, then an Exponential one. The data we worked with come from [9]. The main quantities of interest are the total sum of the outstanding liabilities, the sum to be paid at the end of each calendar year, denoted T_i, and the sum to be paid for each accident year, given that past years' amounts are considered as already paid.

3.1 Data: Automatic Facultative business in General Liability (excluding Asbestos & Environmental)

The following values for Y_{ij} have been found in [9] ; they originally come from the Reinsurance Association of America.

Table 1: Values for 10 accident and development years in $1,000. Source: "Historical Loss Development Study", 1991 Edition, published by the Reinsurance Association of America (RAA), p.96.

	Development Year →									
Accident Year ↓	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
1	*5012*	*3257*	*2638*	*898*	*1734*	*2642*	*1828*	*599*	*54*	*172*
2	*106*	*4179*	*1111*	*5270*	*3116*	*1817*	*-103*	*673*	*535*	
3	*3410*	*5582*	*4881*	*2268*	*2594*	*3479*	*649*	*603*		
4	*5655*	*5900*	*4211*	*5500*	*2159*	*2658*	*984*			
5	*1092*	*8473*	*6271*	*6333*	*3786*	*225*				
6	*1513*	*4932*	*5257*	*1233*	*2917*					
7	*557*	*3463*	*6926*	*1368*						
8	*1351*	*5596*	*6165*							
9	*3133*	*2262*								
10	*2063*									

The empty cells represent unknown values. All the amounts are in $*1000* and the accident years go from 1981 (i=*1*) to 1990 (i=*10*). Notice that $y_{27} = -103$ is negative. Several explanations can be considered for such a value: "Typically, these negative values will be the result of salvage recoveries, payments from third parties, total or partial cancellation of outstanding claims, due to initial overestimation of the loss or to possible favorable jury decision in favor of the insurer, rejection by the insurer, or just plain errors" [8]. We will not use this value and consider it as an unknown quantity to be estimated instead. With the lognormal model presented above it is in fact impossible to deal with negative values, which consequently have to be removed or modified. In WinBUGS, this quantity will be entered as *"NA"* instead of *-103.* If we want to keep this negative value, then other models have to be used, such as a lognormal with three parameters (see [8]). This goes beyond the scope of this dissertation.

3.2 Model 1: Lognormal Model

3.2.1 Presentation of the model

The first model considered will be a Lognormal model, similar to the models presented in [7] and [9]. We consider the amounts of outstanding claims to follow a *Lognormal* distribution conditionally on the parameters μ_{ij} and σ^2:

$$Y_{ij}|\mu_{ij}, \sigma^2 \sim LN(\mu_{ij}, \sigma^2)$$

$$\mu_{ij} = m + \alpha_i + \beta_j$$

$$m, \alpha_i, \beta_j \sim N(0, 10^4) \text{ independent}$$

$$\sigma^2 \sim Inv - Gamma(10^{-3}, 10^{-3})$$

for i=*1, ..., l* and j=*1, ..., l*. Here the parameters α_i and β_j denote the effect of year of origin and year of settlement respectively. We will study and interpret their trends in a later section. μ_{ij} is the mean of the logarithm of the data. It is expressed as the sum of m (overall mean), α_i and β_j, to reflect differences in the expected claim amounts for each accident year i and development year j. The parameter σ^2 is modelled using an Inverse Gamma prior distribution as it has to be non-negative. The small values used for the parameters in the *Inv-Gamma* distribution, and the large variance of the *Normal* distribution are intended to provide non-informative prior distributions: these priors are all vague, reflecting our uncertainty and ignorance of the true values taken by $m, \underline{\alpha}, \underline{\beta}$ and σ^2. The large prior variances indicate this uncertainty.

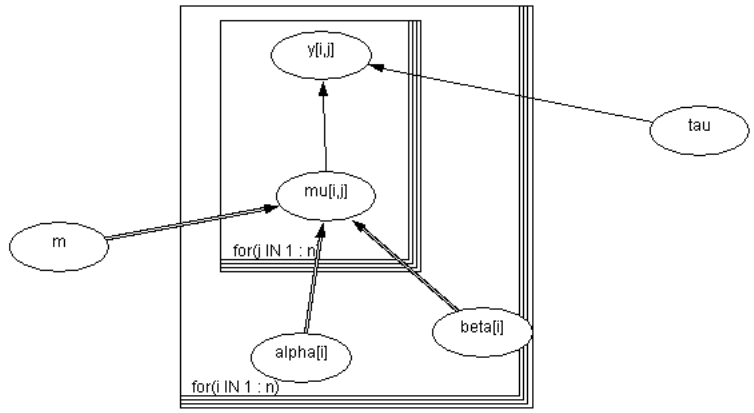


Figure 1: Directed Acyclic Graph representing the structure of the Lognormal Model

The lognormal model is well suited for modelling claim amounts as these ones must be positive (in most cases), and it has fat tails: it is skewed to the right. As mentioned earlier, we will then replace *-103* by a "*NA*" in WinBUGS: it is considered as an unknown value that has to be estimated.

This model can be represented easily in a directed acyclic graph, using the Doodle menu in WinBUGS (see Figure 1).

For now, our aim will be to estimate the values of $\underline{\alpha}$, $\underline{\beta}$, m and σ^2 using the available data. As described in Section 2.1, those parameters are considered as being random variables in Bayesian Statistics. We give them prior distributions as described previously, and will aim to obtain their posterior distributions.

Our analysis is similar to a two-way analysis of variance (ANOVA) with $\underline{\alpha}$ representing the effect of the year of origin, and $\underline{\beta}$, representing the effect of the year of settlement. A restriction is consequently imposed on these effects, for example $\alpha_1 = \beta_1 = 0$ (corner constraint) where α_1 and β_1 are chosen as references for comparison, or $\sum_i \alpha_i = \sum_j \beta_j = 0$ (sum-to-zero constraint) where the respective means of α and β are chosen as references (details in [24]). We will use the sum-to-zero constraint for the remainder of this dissertation.

Under this model and these restrictions, we have to estimate the parameters m, $\alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7, \alpha_8, \alpha_9, \alpha_{10}$, $\beta_2, \beta_3, \beta_4, \beta_5, \beta_6, \beta_7, \beta_8, \beta_9, \beta_{10}$, σ^2, Y_{27}, and Y_{ij} for all i, j such that $i + j > l + 1 = 11$. It is worth noticing that α_1 and β_1 do not need to be estimated, due to the constraint we have set up: $\alpha_1 = -\sum_{i=2}^{10} \alpha_i$ and the corresponding equation holds for β_1.

The next Section shows how to analytically obtain the posterior distributions in this case. The derivation is presented as an example and implies heavy algebraic manipulations. In practice, we will use MCMC methodology (implemented in WinBUGS) to find estimates of the values of our parameters.

3.2.2 Derivation of the posterior distributions

We introduce a new random variable to simplify the following calculations: let $Z_{ij} = log(Y_{ij})$ be the logarithm of the claim amounts. Then, $Z_{ij} \sim N(\mu_{ij}, \sigma^2)$: it has a normal distribution. We consider the parameters $\underline{\alpha}$, $\underline{\beta}$, σ^2 and m, and we split the logarithms of the claim amounts in two categories: known amounts denoted by z^{obs}, and unknown (or missing) amounts denoted by z^{miss}. The aim of this section is to analytically derive the posterior distributions of these quantities (this is in fact what WinBUGS does, in a numerical way). The following algebraic derivations are adapted from [20]. For clarity of the mathematical expressions, we will consider that Y_{27} is known ; the modification to be made in case it is not known is explained at the end of this Section.

Using Bayes Theorem and denoting by f the prior, conditional and marginal densities, we get:

$$f(m, \underline{\alpha}, \underline{\beta}, \sigma^2, z^{mis}|z^{obs}) \propto f(z^{obs}|m, \underline{\alpha}, \underline{\beta}, \sigma^2, z^{mis}) f(m, \underline{\alpha}, \underline{\beta}, \sigma^2, z^{mis})$$

$$f(m, \underline{\alpha}, \underline{\beta}, \sigma^2, z^{mis}|z^{obs}) \propto f(z^{obs}|m, \underline{\alpha}, \underline{\beta}, \sigma^2) f(m) f(\underline{\alpha}) f(\underline{\beta}) f(\sigma^2) f(z^{mis}|m, \underline{\alpha}, \underline{\beta}, \sigma^2)$$

by assuming prior independence. We can sample the missing values directly from the conditional predictive distribution $f(z^{mis}|m, \underline{\alpha}, \underline{\beta}, \sigma^2)$.

The conditional distributions of all the parameters can now be obtained (by Bayes Theorem):

$$f(m|\underline{\alpha}, \underline{\beta}, \sigma^2, z^{obs}) \propto f(z^{obs}|\underline{\alpha}, \underline{\beta}, \sigma^2) f(m) \quad (2)$$

$$f(\underline{\alpha}|m, \underline{\beta}, \sigma^2, z^{obs}) \propto f(z^{obs}|m, \underline{\beta}, \sigma^2) f(\underline{\alpha})$$

$$f(\underline{\beta}|m, \underline{\alpha}, \sigma^2, z^{obs}) \propto f(z^{obs}|m, \underline{\alpha}, \sigma^2) f(\underline{\beta})$$

$$f(\sigma^2|m, \underline{\alpha}, \underline{\beta}, z^{obs}) \propto f(z^{obs}|m, \underline{\alpha}, \underline{\beta}) f(\sigma^2)$$

Here we derive as an example an analytical expression for $f(m|\underline{\alpha}, \underline{\beta}, \sigma^2, z^{obs})$:
First,

$$f(z^{obs}|m, \underline{\alpha}, \underline{\beta}, \sigma^2) = (2\pi\sigma^2)^{-\frac{U}{2}} \exp\{-\frac{1}{2\sigma^2}(\sum_{i=1}^{l} \sum_{j=1}^{l-i+1} (z_{ij} - m - \alpha_i - \beta_j)^2)\} \quad (3)$$

with $l = 10$ in our case, and U is the number of known values (i.e. the number of cells in the upper-triangle), worth $\frac{l(l+1)}{2}$ if all the values in the upper triangle are known.

Considering the fact that m has a normal prior distribution, and by equations (2) and (3), we get (the bounds are omitted in the sums, but it must be clear that they remain unchanged from equation (3)):

$$f(m|\underline{\alpha}, \underline{\beta}, \sigma^2, z^{obs}) \propto \exp\{-\frac{m^2}{2\sigma_m^2} - \frac{1}{2\sigma^2} \sum_{i=1}^{l} \sum_{j=1}^{l-i+1} [m - (z_{ij} - \alpha_i - \beta_j)]^2\}$$

$$f(m|\underline{\alpha}, \underline{\beta}, \sigma^2, z^{obs}) \propto \exp\{-\frac{1}{2}(\frac{m^2}{\sigma_m^2} + \frac{1}{\sigma^2} \sum \sum (m - s_{ij})^2)\}$$

by denoting $s_{ij} = z_{ij} - \alpha_i - \beta_j$. Keeping only the terms involving m and rearranging, we get successively:

$$f(m|\underline{\alpha}, \underline{\beta}, \sigma^2, z^{obs}) \propto \exp(-\frac{1}{2}(\frac{m^2}{\sigma_m^2} + \frac{m^2}{\sigma^2}U - 2\frac{m}{\sigma^2} \sum \sum s_{ij}))$$

$$f(m|\underline{\alpha}, \underline{\beta}, \sigma^2, z^{obs}) \propto \exp(-\frac{1}{2}(\frac{1}{\sigma_m^2} + \frac{U}{\sigma^2})(m^2 - 2m\frac{\sum \sum s_{ij}}{\frac{\sigma^2}{\sigma_m^2} + U}))$$

$$f(m|\underline{\alpha}, \underline{\beta}, \sigma^2, z^{obs}) \propto \exp(-\frac{1}{2\sigma_*^2}(m - m_*)^2)$$

where $\sigma_*^2 = \frac{\sigma^2}{\frac{\sigma^2}{\sigma_m^2} + U}$ and $m_* = \frac{\sum \sum s_{ij}}{\frac{\sigma^2}{\sigma_m^2} + U}$.
Consequently,

$$m|\underline{\alpha}, \underline{\beta}, \sigma^2, z^{obs} \sim N(m_*, \sigma_*^2)$$

By considering that y_{27} is an unknown value, equation (3) would be replaced by:

$$f(z^{obs}|m, \underline{\alpha}, \underline{\beta}, \sigma^2) =$$

$$(2\pi\sigma^2)^{-\frac{U}{2}} \exp\{-\frac{1}{2\sigma^2}((\sum\sum_{j=1}^{l-i+1}(z_{ij}-m-\alpha_i-\beta_j)^2-(z_{27}-m-\alpha_2-\beta_7)^2)\}$$

and the subsequent calculations are modified consequently, with $U = \frac{l(l+1)}{2} - 1$.

3.2.3 Running the model using WinBUGS

Convergence issues

As raised in Section 2.2, one needs to check the convergence of the MCMC algorithm. The first and easiest way to have a rough idea about whether convergence occured is to visually inspect at the trace of several parameters. Figure 2 shows the trace of α_{10}, which does not show any particular pattern and seems to mix well.

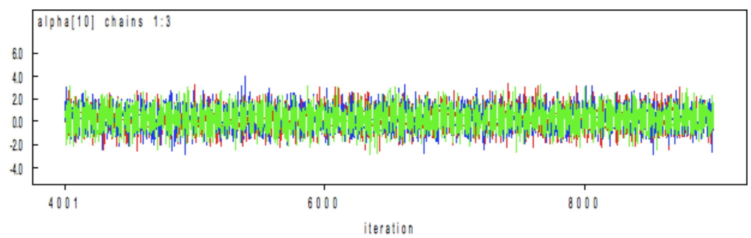


Figure 2: Trace of α_{10}

Then, several diagnostics are performed to have a more precise view on convergence. The Brooks-Gelman-Rubin diagnostic available from the *bgr diag* option in WinBUGS is one of them. We need to generate several chains in parallel, each one starting with different initial values. The diagnostic is performed by comparing the ratio of the between-and-within chain variability (R), and for convergence we should have $R \approx 1$ (see [10] and [16] for more details). R can be estimated by $\widehat{R} = \frac{\hat{V}}{WSS}$ where $\hat{V}$ is the pooled posterior variance estimate and WSS is the mean of the variances within each sample (see [3]). Figure 3 plots the Brooks-Gelman-Rubin diagnostic on α_{10} for three chains of *8,000* iterations. The pooled posterior variance $\hat{V}$ is plotted in green, the average within-sample variance WSS is plotted in blue, and their ratio R is in red. $R \approx 1$ after roughly *4,000* iterations, which indicates that the chain has converged. Consequently, we will consider a burn-in period of *5,000* iterations when running the chains. The initial values used for the three chains are available in Table 14 in the Appendix. Convergence occurred for all parameters.

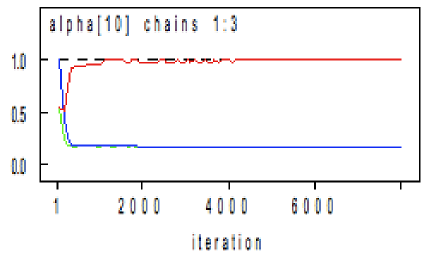
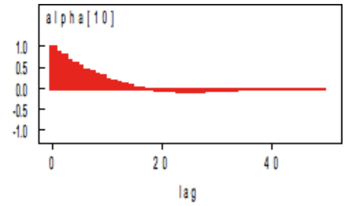


Figure 3: Gelman-Rubin diagnostic

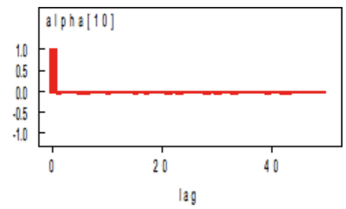
Other tests are available, for example the *CODA* package in the statistical software *R* includes the Geweke (see [14]), the Raftery-Lewis (see [17]) and the Heidelberger-Welch (more details in [18]) diagnostics as well.

The Geweke diagnostic performed on our data, using the *CODA* package in *R*, confirmed the conclusions of the preceding test. This diagnostic splits the sample into two parts, the initial 10% and the last 50%, and applies a Z test to check whether the means of the output are equal. A value $|Z| > 2$ indicates non-convergence. The maximum absolute value observed for all parameters was *1.7614* (for β_5 in Chain *2*), which is between -*2* and *2* and therefore indicates convergence. The Geweke diagnostic confirms the results we had with the Brooks-Gelman-Rubin diagnostic ; convergence occurs for all parameters.

However, the algorithm used does not provide an independent sample, and the chain may, as a consequence, mix poorly. The autocorrelation function of α_{10} is displayed on Figure 4, which strongly confirms that we have a correlated sample. One way to attenuate the correlation within the sample is to thin the chain. This consists of taking only one value every r iterations, instead of every iteration. We have consequently run the same model with a burn-in period of *d=5,000* iterations, a thinning of *r=20* and *8,000* points. In Figure 4, we can see the thinning has heavily reduced the correlation.



(a) Autocorrelation of α_{10} before thinning



(b) Autocorrelation of α_{10} after thinning

Figure 4: Comparison of autocorrelations before and after thinning

Parameter estimates

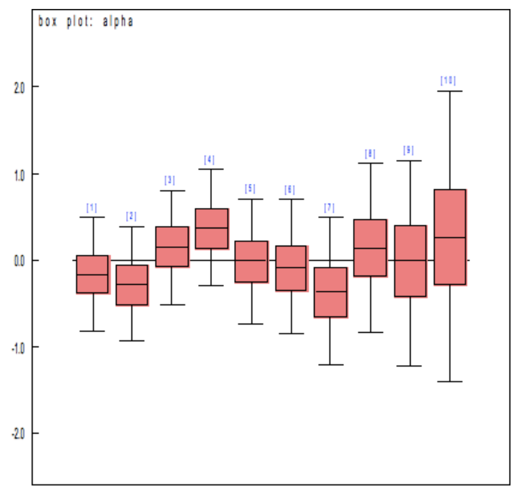
We now want to obtain estimates of $\underline{\alpha}, \underline{\beta}, m$ and σ^2. After discarding the first *5,000* iterations as a burn-in period and running an additional *20,000* iterations, with a thinning of *20*, we obtain estimates for the different parameters. We can compare the different values of α_i with their overall mean in order to see their effect. For example, a negative value of α_i will indicate that the total claim amount to be paid for accident year i is less than the overall mean for this block of business. The estimates of $\underline{\alpha}$ and $\underline{\beta}$, denoting respectively the effect of an accident year and a development year, are available in Table 2, as well as the estimates for m and σ^2.

Figure 5a shows the effect of the year of accident. There is no clear pattern, a decrease in α could represent a decline in the activity of the company such as a decreasing number of policyholders, or simply a year where the amount of claims (in value) has decreased, without implying anything else on the health of the company. However, one can notice the increasing pattern of the posterior variance of the estimates. This is due to the fact that we have less data to estimate α_i as i increases: remember we only know the values of the upper-triangle, so for accident year *10* only one claim amount is known whereas *10* claim amounts are known for accident year *1*. This increasing trend of the variance reflects the uncertainty we have about future

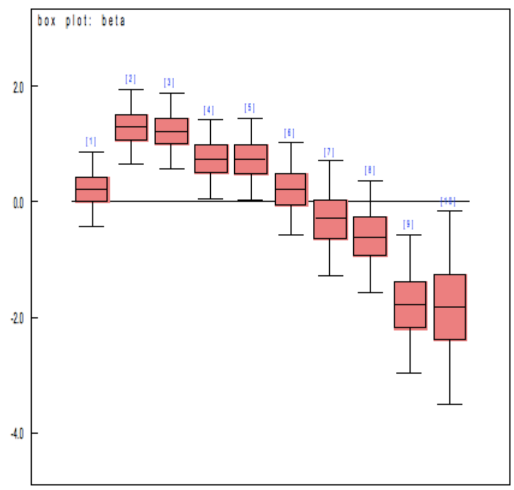
Table 2: Estimated Values of the parameters in Model 1

	Mean	Standard Error	Monte Carlo Error	2.5th percentile	97.5th percentile
α_1	*-0.1717*	*0.327*	*0.0023*	*-0.814*	*0.477*
α_2	*-0.2805*	*0.341*	*0.0029*	*-0.948*	*0.390*
α_3	*0.154*	*0.333*	*0.0026*	*-0.497*	*0.819*
α_4	*0.3724*	*0.350*	*0.0028*	*-0.323*	*1.06*
α_5	*-0.0117*	*0.368*	*0.0028*	*-0.740*	*0.711*
α_6	*-0.0842*	*0.398*	*0.0031*	*-0.871*	*0.696*
α_7	*-0.3755*	*0.434*	*0.0037*	*-1.242*	*0.477*
α_8	*0.1311*	*0.499*	*0.0043*	*-0.851*	*1.107*
α_9	*-0.0087*	*0.597*	*0.0063*	*-1.196*	*1.157*
α_{10}	*0.2749*	*0.842*	*0.0114*	*-1.382*	*1.920*
β_1	*0.2106*	*0.328*	*0.0023*	*-0.428*	*0.855*
β_2	*1.3130*	*0.325*	*0.0026*	*0.677*	*1.953*
β_3	*1.2160*	*0.334*	*0.0027*	*0.558*	*1.878*
β_4	*0.7470*	*0.349*	*0.0028*	*0.056*	*1.432*
β_5	*0.7527*	*0.368*	*0.0030*	*0.012*	*1.467*
β_6	*0.2151*	*0.398*	*0.0032*	*-0.565*	*0.999*
β_7	*-0.3001*	*0.498*	*0.0045*	*-1.282*	*0.685*
β_8	*-0.6040*	*0.498*	*0.0047*	*-1.581*	*0.386*
β_9	*-1.7580*	*0.609*	*0.0062*	*-2.946*	*-0.556*
β_{10}	*-1.7810*	*0.837*	*0.0115*	*-3.424*	*-0.123*
m	*7.1370*	*0.207*	*0.0024*	*6.726*	*7.540*
σ^2	*0.7978*	*0.200*	*0.0016*	*0.495*	*1.271*

payments for recent accident years. Indeed, for past accident years only a few number of claim amounts in subsequent development years are still unknown.



(a) Effect of Year of Accident on claim amounts (α_i). Source: own calculations



(b) Effect of Development Year on claim amounts (β_j). Source: own calculations

Figure 5: Effect of Accident and Devolpment Years on claim amounts for Model 1. Source: own calculations

The effect of development year is shown in Figure 5b. Excluding the payments settled directly at the end of the current accident year, we observe a decreasing trend for β_j: we expect less payments to be made for a given accident year as time increases. Once again, an increasing variance is observed, reflecting the greater uncertainty we have about the future. Figure 6 shows a plot of the standard deviation of α against the accident year, and the standard deviation of β against the development year. They are roughly the same, except that:

- the variance of α_2 is higher than expected: this is due $Y[2, 7]$ being unknown
- the variance of β_7 is also higher than expected (for the same reason).

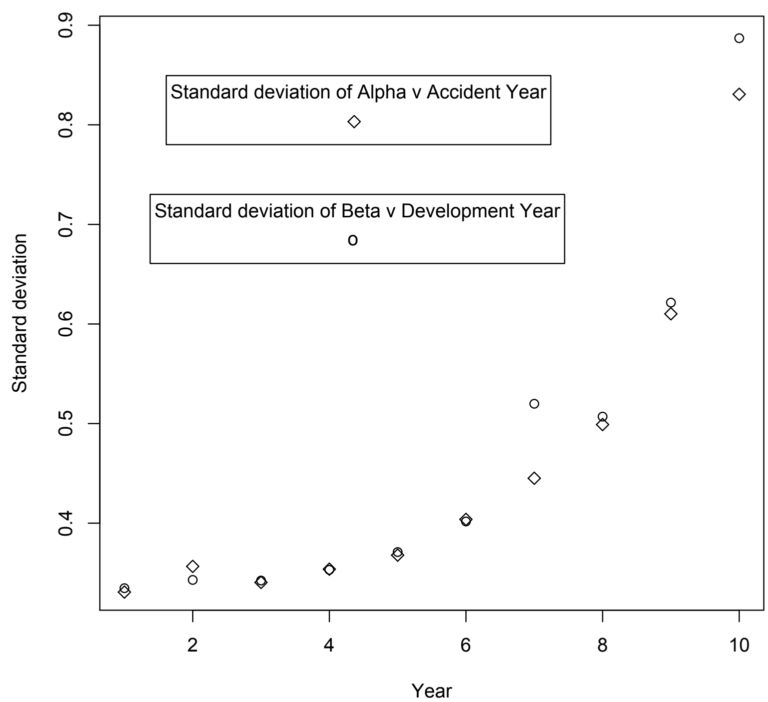


Figure 6: Posterior standard deviations of α_i and β_j against Year. Source: own calculations

3.2.4 Prior Sensitivity

An important criticism about the use of Bayesian Statistics is its subjectivity. Indeed, the posterior distribution may be influenced by the choice of the prior distribution.

Table 3: Comparison of the posterior means of several T_i for Model 1 with different priors (in \$*1,000*).

	T_1	T_3	T_5	T_7	T_8	T_9	*Total Sum*
Model 1	*39345*	*21114*	*9756*	*3612*	*1451*	*1018*	*129081*
Model 1 with Uniform Priors	*40141*	*21427*	*10025*	*3699*	*1427*	*1006*	*131406*

We have consequently assigned different priors to the parameters α, β, m and σ^{-2}: m, and each α_i and β_j are assigned a $Uniform(-5000, 5000)$ distribution, and σ^{-2} is assigned a $Uniform(0, 10000)$ prior distribution (as it has to be positive). All those prior distributions are flat and cover a large range of values.

To help us with comparisons, we introduce now the amount to be paid for each calendar year for this block of business, taking into account all the accident years. Let T_i be the estimator of the amount to be paid in each calendar year. The estimated value will be denoted by T_i as well. For example, T_1 is the estimated sum to be paid next year, considering all accident years: $T_1 = \sum_{k=2}^{10} Y_{k,12-k} = Y_{2,10} + Y_{3,9} + Y_{4,8} + Y_{5,7} + Y_{6,6} + Y_{7,5} + Y_{8,4} + Y_{9,3} + Y_{10,2}$. This is a "diagonal" sum. These quantities will be helpful for the remainder of this dissertation.

Table 12 in the Appendix shows there is little difference between the parameters obtained with the two different sets of priors. Table 3 confirms this analysis, by showing that for each calendar year the posterior means of the claim amounts calculated using the two different sets of priors are very similar (see Section 5.1 for the derivation these results). As a consequence, Model 1 is robust: the choice of prior distributions has little effect on the final results.

3.3 Model 2: Exponential Model

3.3.1 Presentation of the model

Now we consider an exponential model, similar to the one found in [9]. It is defined as follows:

$$Y_{ij}|\lambda_{ij} \sim Exp(\lambda_{ij})$$

$$\lambda_{ij} = exp\{-(\phi + \gamma_i + \delta_j)\}$$

$$\phi, \gamma_i, \delta_j \sim N(0, 10^4)$$

where the sum-to-zero constraint is imposed on the γ_i and δ_j parameters, to give a two-way ANOVA-type analysis. $\underline{\gamma}$ represents the effect of the accident year and $\underline{\delta}$ the effect of the development year.

The expected value of Y_{ij} conditional on λ_{ij} is $\frac{1}{\lambda_{ij}} = exp(\phi + \gamma_i + \delta_j)$ and its variance is $exp\{(\phi + \gamma_i + \delta_j)^2\}$.

The structure of this model can be easily represented using the Doodle menu in WinBUGS (see Figure 7).

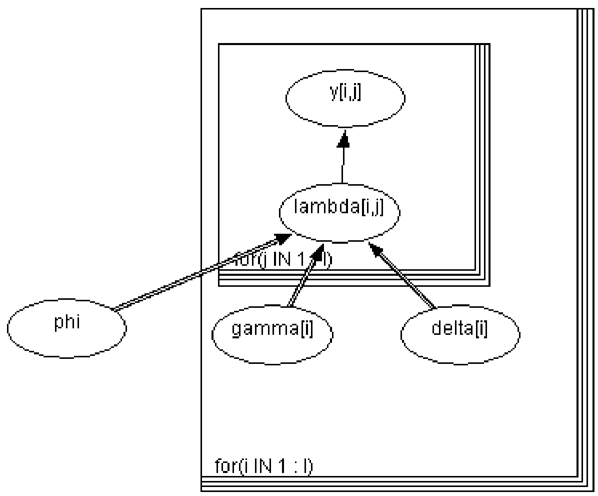
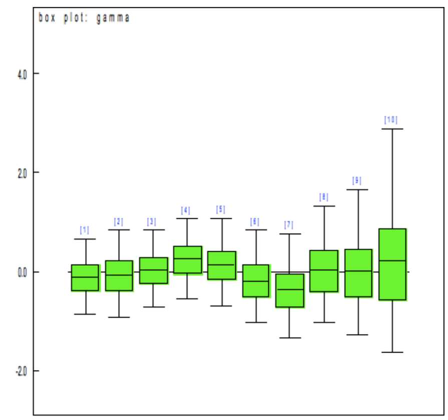


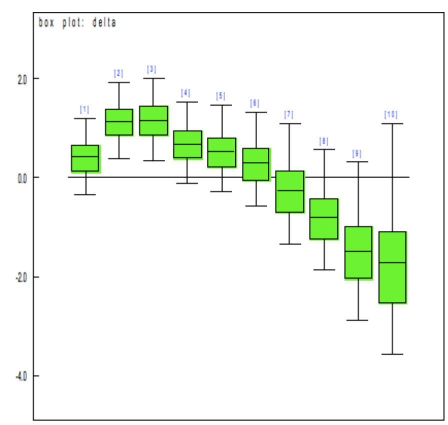
Figure 7: Directed acyclic graph representing the structure of the Exponential Model

3.3.2 Running the model using WinBUGS

Using *20,000* iterations after discarding the first *5,000* iterations, we get estimates for ϕ, γ_i, δ_j and credible intervals (see Table 13 in Appendix). We monitored convergence with the same diagnostics as for Model 1 ; convergence occured for all parameters.



(a) Effect of Accident Year on claim amounts (γ_i). Source: own calculations



(b) Effect of Development Year on claim amounts (δ_j). Source: own calculations

Figure 8: Effects of Accident and Development Years on claim amounts for model 2. Source: own calculations

The same patterns as in Model 1 are observed when we plot γ and δ, as shown in Figure 8: apart from the payments settled at the end of the current accident year, we observe a decreasing trend for δ_i. We indeed expect less payments to be made for a given accident year as time increases.

Moreover, the variance is once again increasing with time, and the same particularity is observed for years *2* and *7* (see Figure 9).

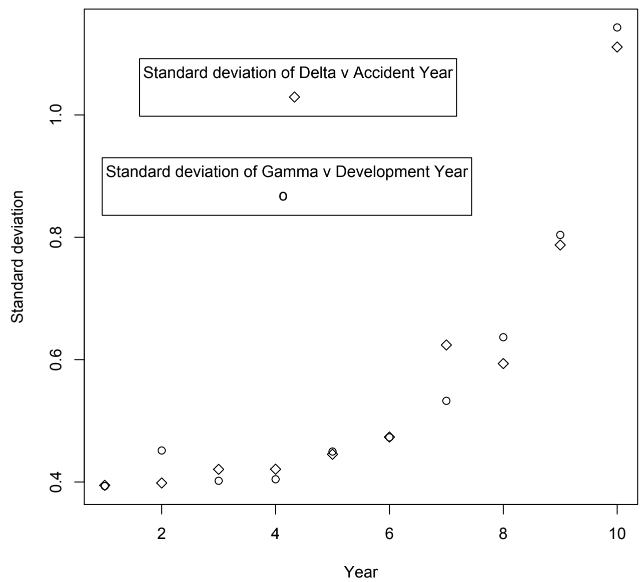


Figure 9: Standard deviations of γ and δ against Year. Source: own calculations

4 Model Selection and Posterior Predictive Checking

4.1 Model comparison (DIC)

In Section 3 we have generated values for the outstanding claims under two different models: a lognormal (Model 1) and an exponential (Model 2). We now want to compare them so that we can decide which one fits the data better.

First, Figure 10 shows a plot of the posterior mean and standard deviation of the predicted claim amounts for each calendar year for models 1 and 2 (the derivation of these predicted claims is treated in Section 5.1). The means of both models exhibit a similar pattern, with the mean of Model 2 being systematically higher. The variance of Model 2 is also higher, and the presence of the unknown value in accident year *2* is of great effect compared to Model 1, which is undesirable. Figure 11 shows the posterior distribution of T_1 for both models ; we can observe that the exponential model has a larger variance.

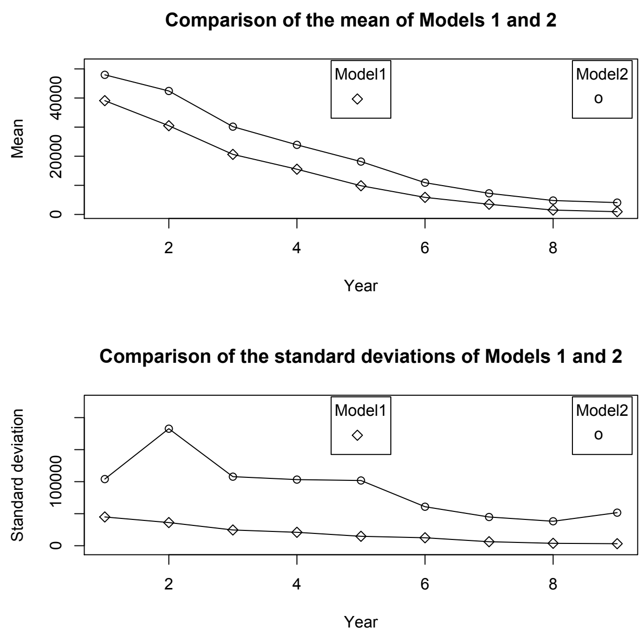
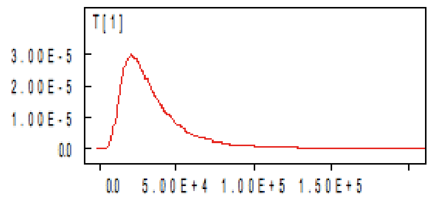
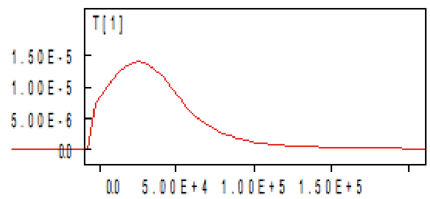


Figure 10: Comparison of the mean and variance of the claim amounts for each calendar year for Models 1 and 2



(a) Lognormal model



(b) Exponential model

Figure 11: Posterior distribution for the first calendar year T_1, using the lognormal and exponential models

However, this does not allow us to discriminate properly between the two models. The Deviance Information Criterion (DIC) will help us do so: a lower value of the DIC indicates a better fit of the model . More information is available in [15]. The DIC is given by the expression:

$$DIC(m) = 2\overline{D(\theta_m, m)} - D(\overline{\theta}_m, m)$$

where $D(\theta_m, m)$ is the deviance measure, $\overline{D(\theta_m, m)}$ is its posterior mean, and $\overline{\theta}_m$ is the posterior mean of the parameters included in the model considered, m.

By denoting by p_D the effective number of parameters, we can write

$$DIC(m) = p_D + \overline{D(\theta_m, m)}$$

since $p_D = \overline{D(\theta_m, m)} - D(\overline{\theta}_m, m)$ (see [28]). Thus, the selection of the model will be influenced by $\overline{D(\theta_m, m)}$, which decreases as the number of parameters increases, and a penalty term p_D, which favours models with small numbers of parameters.

Applying this to our two models within WinBUGS, we observe that Model 1 has a DIC lower than Model 2: the difference between the two equals *5.241*. According

	Dbar	*Dhat*	*pD*	*DIC*
Model 1	*960.837*	*940.017*	*20.821*	*981.658*
Model 2	*969.000*	*951.101*	*17.899*	*986.899*

Table 4: Comparison of the DIC for Model 1 and Model 2

to [15], there is enough evidence to consider Model 1 as better.

4.2 Posterior Predictive Checking & Posterior Predictive P-Value

Having selected Model 1 over Model 2, we now investigate the suitability of this model. The posterior predictive density, denoted as $\pi(y'|y)$ in Section 2.1 can help us doing so, as we can easily generate new values z from this distribution using WinBUGS as we will see later in Section 5.1.

Consider the sum of the known values of each row in Table 1, i.e. only for the upper triangle, and call it $D_i(y,\theta)$. Thus, $D_i(y,\theta) = \sum_{j=1}^{l+1-i} y_{ij}$. We also consider the same sum but with data predicted from Model 1 (again, see Section 5.1 for the derivation), denoted by $D_i(y',\theta)$. Those sums are calculated at each step of the process. If our model correctly fits the data, these sums should not be very different .

Define now a measure of discrepancy between the two sums as:

$$Posterior\ p\text{-}value_i = P(D_i(y',\theta) > D_i(y,\theta)|y)$$

(see [21] for more details). This probability will be approximated using the following approach. At each step k, consider:

$$a_i^{(k)} = \begin{cases} 1 & if D_i^{(k)}(y',\theta) > D_i^{(k)}(y,\theta) \\ 0 & if D_i^{(k)}(y',\theta) < D_i^{(k)}(y,\theta) \end{cases}$$

Then, we have $Posterior\ p\text{-}value_i \approx \frac{1}{n-d+1} \sum_{k=d}^{n} a_i^{(k)}$. The implementation in WinBUGS is available in Section 7.5 of the Appendix. Values near *0.5* indicate that the distributions of *y'* and y are close, whereas values close to *0* or *1* indicate real discrepancies (see [22, 23]).

Table 5 shows that all the *Posterior p-value*s are close to *0.5,* which indicates a good fit of the model, even though it seems to slightly overestimate the true values:

Table 5: *Posterior p-value* for each accident year

Accident Year	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Posterior p-value	*0.665*	*0.602*	*0.713*	*0.726*	*0.426*	*0.637*	*0.477*	*0.568*	*0.620*	*0.496*

most of the *posterior p-values* are above *0.5*. More tests are available, see for example [3] for more details.

We could have considered another test statistic, such that the overall sum of the upper-triangle. By using the same method, we get a Posterior p-value of *0.5923*, which shows adequate goodness of fit as well.

5 Prediction and Comparison with Chain Ladder

5.1 Predictive distribution for future or missing observations using MCMC methods

We want to obtain estimates for future amounts to be paid, as well as for the missing value y_{27} we have in accident year 2, development year 7.

Let y^{mis} be a future data value to be predicted and $\underline{\theta} = (m, \alpha, \beta, \sigma^2)$. Using the conditional independence between y^{mis} and y^{obs} given $\underline{\theta}$, we can write (see[11]):

$$f(y^{mis}|y^{obs}) = \int f(y^{mis}, \underline{\theta}|y^{obs})d\underline{\theta} = \int f(y^{mis}|\underline{\theta})\pi(\underline{\theta}|y^{obs})d\underline{\theta} = E_{\underline{\theta}|y^{obs}}\{f(y^{mis}|\underline{\theta})\}$$

giving the MCMC estimate

$$f(y^{mis}|y^{obs}) \approx \frac{1}{n-d+1}\sum_{k=d}^{n} f(y^{mis}|\underline{\theta}^{(k)})$$

where $f(y^{mis}|\underline{\theta})$ is the sampling density of y^{mis}, n is the total number of iterations (*n=25,000* here), d is the number of discarded iterations (*5,000* here) and $\underline{\theta}^{(k)}$ represents the model parameters calculated at iteration k. This equation implies that we can generate values from the predictive distribution $f(y^{mis}|y^{obs})$ by first generating values from$f(y^{mis}|\underline{\theta})$, which is a known distribution (see Section 3.2.2), and then taking the expectation. This procedure is easily set up in WinBUGS. Table 6 shows the estimates of the outstanding claim amounts to be paid for each accident and development year.

We can also get estimates of the reserves needed for each calendar year (denoted by T_i, see Section 3.2.4) in respect to this block of business and we observe, as expected, that the amounts are decreasing (see Figure 12), since there are less remaining years over time where the company will have to pay, with respect to that block of business. For T_1, there are indeed *8* remaining years, whereas for T_9 there is none.

Notice that the distributions of T_i and their total sum ($\sum T_i$) are positively skewed so high payments could be predicted under this model, which happens in practice (see Figures 13 and 12). The variance is decreasing over time as there is less uncertainty over time (since there are less payments to be made).

Table 6: Estimation of the outstanding claim amounts (in $1,000). Source: own calculations.

Accident Year ↓	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
1										
2							*1309*			*397*
3									*503*	*617*
4								*1841*	*632*	*779*
5							*1730*	*1278*	*442*	*529*
6						*2564*	*1644*	*1194*	*423*	*501*
7					*3280*	*2055*	*1254*	*923*	*317*	*384*
8				*5632*	*5589*	*3403*	*2159*	*1598*	*557*	*668*
9			*8288*	*5226*	*5264*	*3166*	*2006*	*1485*	*511*	*599*
10		*15110*	*13900*	*8778*	*8805*	*5334*	*3413*	*2433*	*852*	*1018*

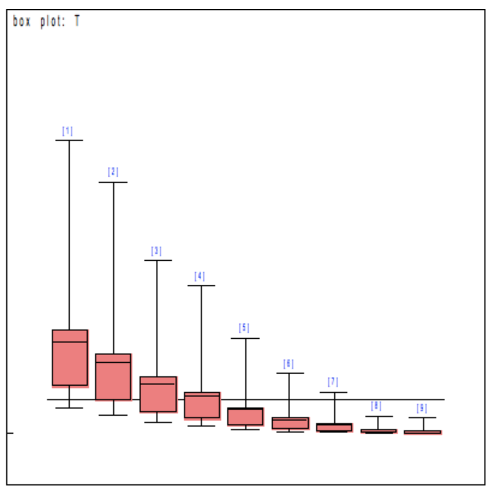


Figure 12: Estimated amounts to be paid per calendar year (in $*1,000*). Source: own calculations.

Table 7: Estimated amounts to be paid per accident year (in $*1,000*). Source: own calculations.

Accident year	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Claim amounts	*known*	*397*	*1119*	*3252*	*3979*	*6326*	*8212*	*19606*	*26545*	*59640*

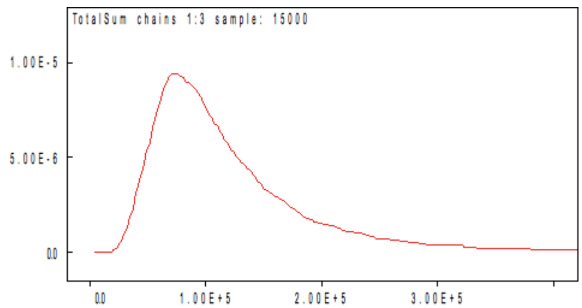


Figure 13: Distribution function of the total sum ($\sum T_i$)

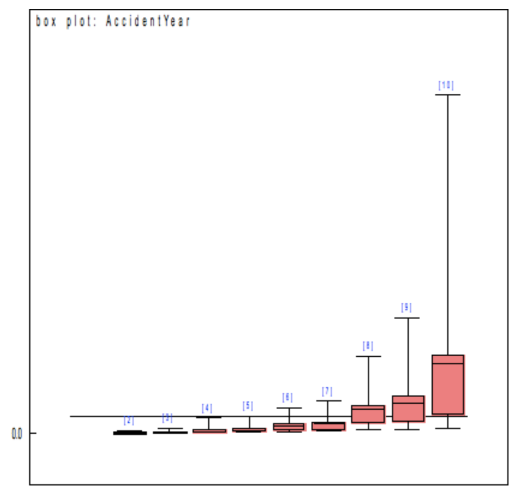


Figure 14: Estimated amounts to be paid per Accident Year

We also looked at the expected amounts to be paid per accident year (see Table 7). They are increasing: for the second accident year only the amount in development year *10* is taken into account, whereas for the last accident year only the payment for the first development year is *not* taken into account. As expected, the variance is increasing due to the bigger number of values to be estimated (see Figure 14).

5.2 Results obtained with the Chain Ladder Method

We now want to implement the ladder method (see [4, 5] for more detailed explanations):

- Consider the data in their cumulative form

Table 8: Estimation of the outstanding claim amounts with the Chain Ladder method (in $1,000). Source: own calculations.

	Devolpment Year →									
Accident Year ↓	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
1										
2										*154*
3									*397*	*220*
4								*900*	*474*	*262*
5							*1098*	*907*	*477*	*264*
6						*1797*	*740*	*612*	*322*	*178*
7					*2114*	*1636*	*674*	*557*	*293*	*162*
8				*3552*	*2861*	*2214*	*912*	*753*	*396*	*219*
9			*3364*	*2373*	*1911*	*1479*	*609*	*503*	*265*	*147*
10		*4125*	*3358*	*2721*	*2192*	*1696*	*698*	*577*	*304*	*168*

- Calculate λ_j, a pooled estimate of the development factor for claims from development year j to j+*1* (independent of the accident year i), with $\lambda_j = \frac{\sum_{i=1}^{l-j+1} y_{i,j}}{\sum_{i=1}^{l-j+1} y_{i,j-1}}$ (see [7] for more details)
- Estimate the remaining (unknown) values in the lower-triangle by using λ_j: for all i in (l+*2*-j, l), $y_{i,j} = \lambda_j y_{i,j-1}$

We keep the negative value we had initially and perform the Chain Ladder method to get estimates of the reserves we need to set up. The use of statistical package R enables us to get the following values for λ, and the estimates of the claim amounts shown in Table 8:

$\lambda = (2.99936, 1.62352, 1.27089, 1.17168, 1.11339, 1.04194, 1.03326, 1.01694, 1.00922).$

The estimated amounts to be paid per calendar year and accident year can be calculated, as we did with the lognormal model. However, the Chain Ladder method only gives point estimates.

5.3 Comparison

Table 9 shows the posterior means of T_i (considering y_{27} as unknown) and the results obtained with the Chain Ladder method (keeping $y_{27} = -106$) . First, note that a company would probably not take the posterior mean to set up reserves. In fact, it would rather set up a reserve that covers its outstanding liabilities with a given probability, such as a *99.5%* probability for example. In this case, a 97.5% probability of meeting the outstanding liability would lead to a reserve of *\$380* million, to be compared to *\$52* million obtained with the Chain Ladder.

The results differ quite a lot between the two methods (see Table 9), as also mentioned in [9]. The main advantage of Model 1 is its ability to give percentage points, as well as several statistical estimates.

Table 9: Expected claim amounts for each calendar year, calculated with the Chain Ladder method and Model 1 (in \$1000). Source: own calculations

	Chain Ladder	*Lognormal model (M1)*	
Assumption:	*Unchanged Data*	y_{27} *unknown*	
		Mean	*97.5 percentile*
T_1	*17501*	*39345*	*126726*
T_2	*13069*	*30941*	*106517*
T_3	*8871*	*21114*	*73307*
T_4	*5725*	*16004*	*61835*
T_5	*3529*	*9756*	*39564*
T_6	*1760*	*5839*	*25604*
T_7	*1061*	*3612*	*16887*
T_8	*451*	*1451*	*7393*
T_9	*168*	*1018*	*6493*
Total Sum $= \sum T_i$	*52135*	*129076*	*379607*

The results exposed above have been obtained by taking the negative value into account for the chain ladder method, but considering it as unknown for the lognormal model. To make more sensible comparisons between the two methods, we now consider a set of assumptions and calculate estimates of the claim amounts with both methods for each assumption. Several assumptions can be made: y_{27} can be left unchanged, it can be replaced by a random positive value, or it can be considered as an unknown value. Moreover, the value of y_{21} seems dubious, as it is very small compared to the other accident years. We could consequently consider it as an unknown value, or change it arbitrarily to a bigger value. Following the suggestions found in [9], we consider the following set of assumptions:

1. $y_{27} = 1$, and the other claim amounts stay unchanged for this accident year

2. $y_{27} = 401$ and $y_{28} = 169$ (which corresponds to setting the value of the cu-

mulative payment made in accident year *2* and development year *7* equal to *16000*)

3. y_{27} is unknown

4. y_{27} and y_{21} are unknown

Table 10 summarizes the results following the set of assumptions. It clearly shows that the lognormal model is much more volatile than the chain ladder method. Considering the different assumptions, the average of the totals for the chain ladder is $*52* million, with a standard deviation of $*541,000* whereas for the lognormal model the total is $*156* million and the standard deviation $*9*3 million, much bigger than for the chain ladder. The lognormal model does not seem to be robust concerning outliers as compared to the chain ladder.

Table 10: Expected claim amounts per calendar year (in $1,000).

Assumption:	*Unchanged Data*	*1*	*2*	*3*	*4*
T_1	*17501*	*17534*	*17418*	*17705*	*17328*
T_2	*13069*	*13090*	*12930*	*13178*	*12943*
T_3	*8871*	*8890*	*8803*	*8997*	*8832*
T_4	*5726*	*5753*	*5709*	*5925*	*5792*
T_5	*3529*	*3548*	*3415*	*3624*	*3521*
T_6	*1759*	*1781*	*1727*	*1907*	*1852*
T_7	*1061*	*1061*	*904*	*1004*	*972*
T_8	*451*	*450*	*449*	*454*	*435*
T_9	*168*	*168*	*168*	*169*	*159*
Total	*52135*	*52275*	*51523*	*52963*	*51834*

(a) Chain Ladder method

Assumption:	*Unchanged Data*	*1*	*2*	*3*	*4*
T_1	-	*88417*	*38689*	*39343*	*25902*
T_2	-	*68315*	*30055*	*30941*	*19614*
T_3	-	*49221*	*20505*	*21112*	*13093*
T_4	-	*35971*	*15306*	*16004*	*9424*
T_5	-	*20941*	*8810*	*9756*	*5632*
T_6	-	*8325*	*4718*	*5839*	*3225*
T_7	-	*10889*	*2709*	*3612*	*1870*
T_8	-	*5005*	*1455*	*1451*	*720*
T_9	-	*3861*	*974*	*1018*	*504*
Total	-	*290945*	*123221*	*129076*	*79984*

(b) Lognormal Model

6 Conclusion

In this dissertation, we have shown how one can use Bayesian Statistics and MCMC techniques to model the outstanding liabilities of an insurance company. We have considered two stochastic models, an exponential and a lognormal, and have been able to select the one fitting the data better through the use of the Deviance Information Criterion. The effect of accident year and development year have been highlighted. We have then focused our attention on estimating the claim amounts for each accident and calendar year.

One of the main advantages of this method is that we now have estimates for the mean and variance of the outstanding claim amounts, as well as percentile points. It can prove very useful if one wants to find the VaR or any other risk measure for a block of business. Nevertheless, it is much more volatile than the Chain Ladder in case of missing values.

We have also shown that the results obtained are quite different from the ones obtained with the Chain Ladder method, which confirms what had been already observed in [9]. The expected total claim amount is indeed *$52* millions with the Chain Ladder, as compared to *$129* million in our model (more than twice as much). However, the comparison between the two methods is not straightforward here, due to a lack of future data: we are not able to assess which method predicts the outstanding liabilities better.

Moreover, the model developed here does not use any information from the claim numbers. If these are available, a different model can be used to take into account their variation as well. The same remark can be made concerning the premiums received.

Finally, the presence of negative values in the upper-triangle can be handled in different ways, we refer the interested reader to [8] and [27] for more information.

7 Appendix

7.1 Validation of methodology through some theoretical considerations

In this Section, we give a theoretical validation of the methodology used in Section 5.1 where we calculated estimates for the outstanding claim amounts to be paid for each accident year.

Let $E(Y_{i.}|\theta)$ be the expected value of the claim amounts to be paid for accident year i (i.e. we take into consideration only the values located in the lower triangle), for $2 \leq i \leq l$.

We have:

$$E(Y_{i.}|\underline{\theta}) = E(\sum_{j=l+2-i}^{l} Y_{ij}|\underline{\theta}) = \sum_{j=l+2-i}^{l} E(Y_{ij}|\underline{\theta}) = \sum_{j=l+2-i}^{l} \exp(m + \alpha_i + \beta_j + \frac{\sigma^2}{2})$$

$$E(Y_{i.}|\underline{\theta}) = \exp(m + \alpha_i + \frac{\sigma^2}{2}) \sum_{j=l+2-i}^{l} \exp(\beta_j).$$

Using WinBUGS, we monitor the random variable $\exp(m+\alpha_i+\frac{\sigma^2}{2}) \sum_{j=l+2-i}^{l} \exp(\beta_j)$ and get its posterior mean. Table 11 shows:

- the expected claim amounts for each accident year, $E(Y_{i.}|\underline{\theta})$, coming from theoretical considerations,
- the calculated claim amounts for each accident year, coming from numerical calculations only (WinBUGS).

These results are very close and confirm the validity of our methodology. This method can also be used to easily obtain the values of the amounts to be paid per accident year.

Table 11: "Expected" and calculated claim amounts for each accident year (in \$1,000). Source: own calculations

Accident Year	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Calculated claim amounts	*known*	*397*	*1119*	*3252*	*3979*	*6326*	*8212*	*19606*	26545	*59640*
"Expected" claim amounts	*known*	*396*	*1120*	*3260*	*3982*	*6347*	*8105*	*19680*	*26650*	59680

7.2 Estimated values of the parameters in Model 1

Table 12: Comparison of the estimated values of the parameters in Model 1 for two different sets of priors. Source: own calculations

	Mean		*Monte Carlo Error*	
	Model 1 initial	*Model 1 with Uniform priors*	*Model 1 initial*	*Model 1 with Uniform priors*
α_1	*-0.1717*	*-0.1668*	*0.0023*	*0.0025*
α_2	*-0.2805*	*-0.2825*	*0.0029*	*0.0029*
α_3	*0.154*	*0.1483*	*0.0026*	*0.0028*
α_4	*0.3724*	*0.3728*	*0.0028*	*0.0027*
α_5	*-0.0117*	*-0.0125*	*0.0028*	*0.0030*
α_6	*-0.0842*	*-0.0861*	*0.0031*	*0.0032*
α_7	*-0.3755*	*-0.3745*	*0.0037*	*0.0034*
α_8	*0.1311*	*0.1343*	*0.0043*	*0.0048*
α_9	*-0.0087*	*-0.0068*	*0.0063*	*0.0068*
α_{10}	*0.2749*	*0.2737*	*0.0114*	*0.0118*
β_1	*0.2106*	*0.2152*	*0.0023*	*0.0025*
β_2	*1.3130*	*1.3160*	*0.0026*	*0.0030*
β_3	*1.2160*	*1.2230*	*0.0027*	*0.0027*
β_4	*0.7470*	*0.7540*	*0.0028*	*0.0026*
β_5	*0.7527*	*0.7473*	*0.0030*	*0.0030*
β_6	*0.2151*	*0.2262*	*0.0032*	*0.0031*
β_7	*-0.3001*	*-0.2937*	*0.0045*	*0.0045*
β_8	*-0.6040*	*-0.5933*	*0.0047*	*0.0044*
β_9	*-1.7580*	*-1.7700*	*0.0062*	*0.0064*
β_{10}	*-1.7810*	*-1.8250*	*0.0115*	*0.0115*
m	*7.1370*	*7.1340*	*0.0024*	*0.0025*
σ^2	*0.7978*	*0.8015*	*0.0016*	*0.0016*

7.3 Estimated Values of the parameters in Model 2

Table 13: Estimated values of the parameters in Model 2. Source: own calculations

	Mean	*Monte Carlo Error*
γ_1	*-0.1169*	*0.0049*
γ_2	*-0.0647*	*0.0069*
γ_3	*0.0362*	*0.0054*
γ_4	*0.2487*	*0.0065*
γ_5	*0.1377*	*0.0065*
γ_6	*-0.1745*	*0.0068*
γ_7	*-0.3514*	*0.0073*
γ_8	*0.0399*	*0.0083*
γ_9	*0.0137*	*0.0106*
γ_{10}	*0.2313*	*0.0175*
δ_1	*0.4157*	*0.0054*
δ_2	*1.134*	*0.0052*
δ_3	*1.159*	*0.0063*
δ_4	*0.6881*	*0.0060*
δ_5	*0.5359*	*0.0058*
δ_6	*0.2969*	*0.0071*
δ_7	*-0.2499*	*0.0072*
δ_8	*-0.7938*	*0.0089*
δ_9	*-1.474*	*0.0097*
δ_{10}	*-1.712*	*0.0178*
ϕ	*7.516*	*0.0104*

7.4 Initial Values for the Brooks-Gelman-Rubin diagnostic

Table 14: Initial values for the Brooks-Gelman-Rubin diagnostic

Chain 1

	α	*NA*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*
	β	*NA*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*
	m	*1*								
	τ	*1*								

Chain 2

	α	*NA*	*-1*	*-1*	*-1*	*-1*	*-1*	*-1*	*-1*	*-1*
	β	*NA*	*1*	*1*	*1*	*1*	*1*	*1*	*1*	*1*
	m	5								
	τ	5								

Chain 3

	α	*NA*	*1*	*1*	*1*	*1*	*1*	*1*	*1*	*1*
	β	*NA*	*1*	*1*	*1*	*1*	*1*	*1*	*1*	*1*
	m	5								
	τ	5								

7.5 Code for the Lognormal model (Model 1) in WinBUGS

```
model {
################################################################################
#Model's likelihood
for (i in 1:n) {
for (j in 1:n){
y[i,j] ~ dlnorm(mu[i,j], tau)
y.prime[i,j] ~ dlnorm(mu[i,j], tau) #Useful for goodness of fit
mu[i,j] <- m + alpha[i] + beta[j]
     }  }
################################################################################
#CONSTRAINTS
#CR Constraints
#alpha[1] <- 0
#beta[1] <- 0
#STZ constraint
alpha[1] <- -sum(alpha[2:n])
beta[1] <- -sum(beta[2:n])
################################################################################
#Priors
m ~ dnorm(0,0.01)
for (i in 2:n) {
alpha[i] ~ dnorm(0, 0.01)
beta[i] ~ dnorm(0,0.01)  }
tau ~ dgamma(0.001, 0.001)
sigmasq <- 1/tau
#################################################################################
#sum T to be paid per year (sum diagonal)
for ( i in (n+1):(2*n-1) ) {
for ( a in 1:(i-n) ) {
  Y.T[i-n,a] <- 0
}
for ( a in (i-n+1):n ) {
Y.T[i-n,a] <- y[a, i+1-a]
}
T[i-n] <- sum( Y.T[i-n,1:n])
}
TotalSum <- sum(T[])
#################################################################################
#P-VALUE ETC
```

```
SumLine.prime[1] <- sum( y.prime[1,1:n ] )
SumLine[1] <- sum( y[1, 1:n ] )
PPPValue[1] <- step( (SumLine.prime[1] - SumLine[1]) )
#y[2,7] negative so we do not take this value into account
SumLine.prime[2] <- sum( y.prime[2,1:6 ] )+sum( y.prime[2,8:9] )
SumLine[2] <- sum( y[2,1:6 ] )+sum( y[2,8:9] )
PPPValue[2] <- step( (SumLine.prime[2] - SumLine[2]) )
for ( i in 3:n ) {
SumLine.prime[i] <- sum( y.prime[i,1: (n+1-i) ] )
SumLine[i] <- sum( y[i, 1:(n+1-i) ] )
PPPValue[i] <- step( (SumLine.prime[i] - SumLine[i]) )  }
PPPValueTotal <- sum(PPPValue[])/10

##############################################################################
#ACCIDENT YEAR
AccidentYear[1] <- 0
for (i in 2:n){
  AccidentYear[i] <- sum(y[i,(n+2-i):n])
}
#We can check if the sum is equal to TotalSum (and this is the case!)
#Total <- sum(AccidentYear[])
}
###############################################################################
#DATA
list(n=10)
y[,1]   y[,2]   y[,3]   y[,4]  y[,5]    y[,6]    y[,7]   y[,8]   y[,9]   y[,10]
5012     3257   2638    898    1734      2642     1828    599     54     172
106      4179   1111    5270   3116      1817     NA      673     535    NA
3410     5582   4881    2268   2594      3479     649     603     NA     NA
5655     5900   4211    5500   2159      2658     984     NA      NA     NA
1092     8473   6271    6333   3786      225      NA      NA      NA     NA
1513     4932   5257    1233   2917      NA       NA      NA      NA     NA
557      3463   6926    1368   NA        NA       NA      NA      NA     NA
1351     5596   6165    NA     NA        NA       NA      NA      NA     NA
 3133    2262   NA      NA     NA        NA       NA      NA      NA     NA
2063     NA     NA      NA     NA        NA       NA      NA      NA     NA
END
###############################################################################
#INITS
list(alpha=c(NA,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1),
beta=c(NA,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5), m=5, tau=1)
```

```
list(alpha=c(NA,2,2,2,2,2,2,2,2,2),
beta=c(NA,0,0,0,0,0,0,0,0,0), m=10, tau=10)
```

7.6 Code for the Exponential model (Model 2) in WinBUGS

```
model {
##############################################################################
#Model's likelihood
for (i in 1:n) {
for (j in 1:n){
y[i,j] ~ dexp(lambda[i,j])
            y.prime[i,j] ~ dexp(lambda[i,j])
            lambda[i,j] <- exp(-(phi + gamma[i] + delta[j]))
     }
}
##############################################################################
#CONSTRAINTS
#CR Constraints
       #gamma[1] <- 0
       #delta[1] <- 0
#STZ constraint
       gamma[1] <- -sum(gamma[2:n])
       delta[1] <- -sum(delta[2:n])
##############################################################################
#Priors
phi ~ dnorm(0,0.01)
    for (i in 2:10) {
       gamma[i] ~ dnorm(0, 0.01)
       delta[i] ~ dnorm(0,0.01)  }

##############################################################################
#sum T to be paid per year (sum diagonal)
   for ( i in (n+1):(2*n-1) ) {
   for ( a in 1:(i-n) ) {
       Y.T[i-n,a] <- 0  }
      for ( a in (i-n+1):n ) {
           Y.T[i-n,a] <- y[a, i+1-a]  }
T[i-n] <- sum( Y.T[i-n,1:n])  }
    TotalSum <- sum(T[])
##############################################################################
#P-VALUE ETC
SumLine.prime[1] <- sum( y.prime[1,1:n ] )
        SumLine[1] <- sum( y[1, 1:n ] )
        PPPValue[1] <- step( (SumLine.prime[1] - SumLine[1]) )
```

```
      #y[2,7] negative so we do not take this value into account
SumLine.prime[2] <- sum( y.prime[2,1:6 ] )+sum( y.prime[2,8:9] )
SumLine[2] <- sum( y[2,1:6 ] )+sum( y[2,8:9] )
PPPValue[2] <- step( (SumLine.prime[2] - SumLine[2]) )
for ( i in 3:n ) {  SumLine.prime[i] <- sum( y.prime[i,1:(n+1-i) ] )
SumLine[i] <- sum( y[i, 1:(n+1-i) ] )
PPPValue[i] <- step( (SumLine.prime[i] - SumLine[i]) )  }
############################################################################
#ACCIDENT YEAR
AccidentYear[1] <- 0
for (i in 2:n){
  AccidentYear[i] <- sum(y[i,(n+2-i):n])  }
#We can check if the sum is equal to TotalSum (and this is the case!)
#Total <- sum(AccidentYear[])
}
############################################################################
#Data
list(n=10)
y[,1]   y[,2]   y[,3]   y[,4]   y[,5]   y[,6]  y[,7]  y[,8]   y[,9]  y[,10]
5012    3257    2638    898     1734    2642   1828   599     54     172
106     4179    1111    5270    3116    1817   NA     673     535    NA
3410    5582    4881    2268    2594    3479   649    603     NA     NA
5655    5900    4211    5500    2159    2658   984    NA      NA     NA
1092    8473    6271    6333    3786    225    NA     NA      NA     NA
1513    4932    5257    1233    2917    NA     NA     NA      NA     NA
557     3463    6926    1368    NA      NA     NA     NA      NA     NA
1351    5596    6165    NA      NA      NA     NA     NA      NA     NA
3133    2262    NA      NA      NA      NA     NA     NA      NA     NA
2063    NA      NA      NA      NA      NA     NA     NA      NA     NA
END
############################################################################
#Inits
list(gamma=c(NA,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1),
delta=c(NA,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5), phi=1)
```

References

[1] Personal lecture notes taken during MSc course, 2009/10.

[2] Scollnik, D.P.M (2002), “Actuarial Modeling with MCMC and BUGS”, *North American Actuarial Journal* **5**, No. 2., 96-124

[3] Ntzoufras, I. (2009). *Bayesian Modeling Using WinBUGS*. New York: John Wiley & Sons, Inc.

[4] Boland, P. J. (2007). *Statistical and probabilistic methods in Actuarial Science*. London: Chapman & Hall/CRC.

[5] Personal lecture notes taken during MSc course, 2009/10.

[6] Verrall, R. J. (1991), “On the estimation of reserves from loglinear models”, *Insurance: Mathematics and Economics* **10** (1), 75-80.

[7] Verrall, R. J. (1990), “Bayes and Empirical Bayes Estimation for the Chain Ladder Model”, *ASTlN Bulletin* **20**, 217-243.

[8] de Alba, E. and Ramírez Corzo, M.A. (2005), “Bayesian Claims Reserving When There Are Negative Values in the Runoff Triangle”, 40 th. Actuarial Research Conference ITAM, Mexico August 11-13, 2005.

[9] Mack, T. (1994), “Which stochastic model is underlying the chain ladder method?”, *Insurance: mathematics and economics* **15**, 133–138.

[10] Rubin, D.B. (1992), ”Inference from Iterative Simulation Using Multiple Sequences Andrew Gelman”, *Statistical Science* **7** (4), 457-472.

[11] Personal lecture notes taken during MSc course, 2009/10.

[12] Tse, Y.-K. (2009). *Nonlife Actuarial Models: Theory, Methods and Evaluation*, International Series on Actuarial Science, Cambridge, Cambridge University Press.

[13] International Society for Bayesian Analysis (ISBA), http://www.bayesian.org/

[14] Geweke, J. (1992), “Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments”. In J.M. Bernardo, J.O. Berger, A.P. Dawid and A.F.M. Smith, eds., *Bayesian Statistics* **4**, Oxford: Oxford University Press.

[15] Spiegelhalter, D.J., Best, N.G., Carlin, B. R. and van der Linde, A. (2002), "Bayesian measures of model complexity and fit", *Proceedings of the Royal Society of London Series B - Biological Sciences* **64** (4), 583–639

[16] Brooks S. P., and Gelman, A., (1998), "General Methods for Monitoring Convergence of Iterative Simulations", *Journal of Computational and Graphical Statistics* **7** (4), 434-455

[17] Raftery, A. and Lewis, S. (1992), "How many iterations in the Gibbs sampler?". In J. Bernardo, J. Berger, A. Dawid, and A. Smith, eds., *Bayesian Statistics* **4**, 763-774. Oxford: Claredon Press.

[18] Heidelberger, P. and Welch, P. (1992), "Simulation run length control in the presence of an initial transient", *Operation Research* **31**, 1109-1144

[19] Scollnik, D. (2002), "Implementation of four models for outstanding liabilities in WinBUGS: A discussion of a paper by Ntzoufras and Dellaportas", *North American Actuarial Journal* **6**, 128-136

[20] Ntzoufras, I. and Dellaportas, P. (2002), "Bayesian modelling of outstanding liabilities incorporating claim count uncertainty (with discussion)", *North American Actuarial Journal* **6**, 113-128

[21] Meng, X.-L. (1994), "Posterior predictive p-values", *Annals of Statistics* **22**, 1142-1160

[22] Gelman, A., Meng, X.-L. and Stern, H. (1996), "Posterior predictive assessment of model fitness via realized discrepancies", *Statistica Sinica* **6**, 733-807

[23] Gelman, A. and Meng, X.-L. (1996), "Model checking and model improvement". In W. Gilks, S. Richardson, and D. Spiegelhalter, eds., *Markov Chain Monte Carlo in Practice*, 189-201. London: Chapman & Hall.

[24] Kremer, E. (1982), "lBNR-Claims and the Two-Way Model of ANOVA", *Scand. Act. J.* **1**, 47-55.

[25] Geyer, C. J. (1991), "Practical Markov Chain Monte Carlo", *Statistical Science* **7** (4), 473-483.

[26] Gilks, W.R., Richardson, S. and Spiegelhalter, D.J. (1996). *Markov Chain Monte Carlo in Practice.* London: Chapman and Hall.

[27] Kunkler, M. (2006), "Modelling negatives in stochastic reserving models", *Insurance: Mathematics and Economics* **38** (3), 540-555.

[28] Spiegelhalter, D., Thomas, A., Best, N., Lunn, D. (2003). *WinBUGS User Manual, Version 1.4*. http://www.mrc-bsu.cam.ac.uk/bugs.