I. Introduction

In this thesis, I will provide a detailed explanation of the Markov Chain Monte Carlo method known as the Gibbs sampler and will apply it to the problem of Bayesian graduation of mortality rates in Actuarial Science. This method was proposed by Carlin [5] in 1992. The conclusion of my thesis is that, although this method works in principle, it is a bad one because it is very, very slow for the case where there are many variables. Even then, the method can only be applied effectively when the number of variables is less than or equal to 15.

In Section II, I explain what the process of graduation is. After listing some different methods of graduation, I explain what Bayesian statistics is and focus in on the method of Bayesian graduation. I provide a detailed explanation of life insurance terminology and establish all of the formulas necessary to do a Bayesian graduation of mortality rates. Since it would take an incredible amount of difficult integration to use these formulas directly, we use the Gibbs sampler to do the graduation. In Section III, I explain what the Gibbs sampler is. By applying it to a bivariate example, I illustrate that it works.

In order to apply the Gibbs sampler to Bayesian graduation of mortality rates, we must be able to effectively simulate random observations from the gamma distribution. In Section IV, I show how to simulate them by using a rejection technique proposed by Tadikamalla [15] in 1978. In Section V, I explain what the Metropolis algorithm is and how to use it to simulate from the gamma distribution. Although it is more complicated than Tadikamalla's method, I use it in this thesis because it is more efficient for the purpose of simulating mortality rates for the Gibbs sampler.

In Section VI, I actually apply the Gibbs sampler to Bayesian graduation of mortality rates and provide an explanation of the results that I obtain. I explain how to compute the autocorrelation times for each of the variables representing mortality rates and how to use them in formulas to construct 95% confidence intervals for the mortality rates. As I stated at the beginning of this section, although this method works, it has extremely long autocorrelation times and is very, very slow for many variables.

II. Bayesian Graduation

For insurance, we want estimates of death probability as a function of age. Presumably, this is a smooth, increasing convex function of age, at least beyond 30 years. We want to estimate death probability from statistical data for ages x through x+k, where $x \geq 30$ and k is any integer constant (See [5], p. 67). The statistical data are noisy and not smooth. The process of arriving at a smooth estimate of death probability from noisy data is called graduation. There are various methods of graduation. Some of these are Whittaker graduation, moving-weighted-average graduation, parametric graduation, and Bayesian graduation (see [13]). In this thesis, I will focus only on Bayesian graduation.

In order to understand Bayesian graduation, we must first understand the main idea behind Bayesian statistics (see [11], p. 228), which is as follows: Suppose we have a random variable Y and a random variable θ. Both Y and θ can be multidimensional. The values of Y are observed. The values of θ are not observed. The main problem in statistics in general is to estimate θ using the values of Y. In the Bayesian approach, we assume that the variable θ has pdf $g(\theta)$. This is called the prior pdf. We also assume a conditional probability distribution of Y given θ, which we call $f(y|\theta)$. Suppose we have an actual observation of Y. We can then construct the conditional distribution of θ given Y, which is called the posterior distribution. We can do this as follows: The joint pdf of Y and θ is $p(y,\theta) = f(y|\theta)g(\theta)$. The marginal pdf of Y is $w(y) = \int p(y,\vartheta)d\vartheta$. The conditional pdf of θ given Y is $h(\theta|y) = p(y,\theta) / w(y)$. This is the posterior pdf. The main problem in Bayesian statistics is to sample from the posterior distribution. When we sample, we take the mean of the posterior distribution to be our estimate of θ.

An example of Bayesian estimation is as follows: Suppose we assume the following prior and conditional pdf's:

$$g(\vartheta)=\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{\vartheta^2}{2\sigma^2}\right]$$

$$f(y|\vartheta)=\frac{1}{\sqrt{2\pi}}\exp\left[-\frac{(y-\vartheta)^2}{2}\right]$$

Suppose we are also given n observations of Y, which we denote as y_1, y_2, …, y_n. We want to determine the posteior pdf h(θ|y_1, …, y_n) and its mean and variance. We can proceed as follows:

$$p(y_1,...,y_n,\vartheta)=f(y_1|\vartheta)f(y_2|\vartheta)\cdots f(y_n|\vartheta)g(\vartheta)$$

$$p(y_1,...,y_n,\vartheta)=\left\{\prod_{j=1}^{n}\frac{1}{\sqrt{2\pi}}\exp\left[-\frac{(y_j-\vartheta)^2}{2}\right]\right\}\left\{\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{\vartheta^2}{2\sigma^2}\right]\right\}$$

$$p(y_1,...,y_n,\vartheta)=\frac{1}{\sqrt{2\pi\sigma^2}}\frac{1}{(2\pi)^{n/2}}\exp\left[-\frac{1}{2}\left(\sum_{j=1}^{n}(y_j-\vartheta)^2+\frac{\vartheta^2}{\sigma^2}\right)\right]$$

$$p(y_1,...,y_n,\vartheta)=\frac{1}{\sqrt{2\pi\sigma^2}}\frac{1}{(2\pi)^{n/2}}\exp\left[-\frac{1}{2}\left(\sum_{j=1}^{n}y_j^{\,2}-2\vartheta\sum_{j=1}^{n}y_j+n\vartheta^2+\frac{\vartheta^2}{\sigma^2}\right)\right]$$

$$p(y_1,...,y_n,\vartheta)=\frac{1}{\sqrt{2\pi\sigma^2}}\frac{1}{(2\pi)^{n/2}}\exp\left[-\frac{1}{2}\left[\left(n+\frac{1}{\sigma^2}\right)\left(\vartheta^2-\frac{2n\bar{y}\vartheta}{\left(n+\frac{1}{\sigma^2}\right)}\right)+\sum_{j=1}^{n}y_j^{\,2}\right]\right]$$

$$=\frac{1}{\sqrt{2\pi\sigma^2}}\frac{1}{(2\pi)^{n/2}}\exp\left[-\frac{1}{2}\left[\left(n+\frac{1}{\sigma^2}\right)\left(\vartheta^2-\frac{2n\bar{y}\vartheta}{\left(n+\frac{1}{\sigma^2}\right)}+\frac{n^2\bar{y}^2}{\left(n+\frac{1}{\sigma^2}\right)^2}-\frac{n^2\bar{y}^2}{\left(n+\frac{1}{\sigma^2}\right)^2}\right)+\sum_{j=1}^{n}y_j^{\,2}\right]\right]$$

$$=\frac{1}{\sqrt{2\pi\sigma^2}}\frac{1}{(2\pi)^{n/2}}\exp\left\{-\frac{1}{2}\left[\frac{-n^2\bar{y}^2}{\left(n+\frac{1}{\sigma^2}\right)}+\sum_{j=1}^{n}y_j^{\,2}\right]\right\}*\exp\left\{-\frac{1}{2}\left(n+\frac{1}{\sigma^2}\right)\left(\vartheta-\frac{n\bar{y}}{n+\frac{1}{\sigma^2}}\right)^2\right\}$$

If we hold y_1, …, y_n constant, we obtain the following posterior pdf:

$$h(\vartheta|y_1,...,y_n)=\frac{1}{Z}\exp\left\{-\frac{1}{2}\left(n+\frac{1}{\sigma^2}\right)\left(\vartheta-\frac{n\bar{y}}{n+\frac{1}{\sigma^2}}\right)^2\right\},$$

where Z is a constant that is independent of θ. This is the pdf of a normal distribution. The variance is $\dfrac{1}{n+\dfrac{1}{\sigma^2}}$, and the mean is

$$\frac{n\bar{y}}{n+\dfrac{1}{\sigma^2}}=\left(\frac{1}{1+\dfrac{1}{n\sigma^2}}\right)\bar{y}.$$

We see that as $n\rightarrow\infty$, the mean goes to $\bar{y}$ and the variance goes to 0.

The following comes from [5], unless otherwise stated: When we do Bayesian graduation of mortality data, Y and θ will be multidimensional; we will denote Y by the vector **y** and θ by the vector θ. We want to estimate death probability for ages x through x+k, where x = 35 and k = 20. We are only interested in the unit age intervals [x+j-1, x+j] for j = 1, …, k. The death probability of a living person per unit time at any given time t is known as the force of mortality, denoted by μ(t). We approximate μ(t) by piecewise constants θ_j such that $\mu(t)=\theta_j$ for t ε (x+j-1, x+j) for j = 1, 2, …, k. We define the vector θ to be $\theta=(\theta_1, \ldots, \theta_k)$.

In order to do Bayesian statistics, we must assume a prior distribution for θ. The prior distribution for θ is based on two considerations: Carlin ([5], p. 68) states that Broffitt [3] states that we should start with a distribution where each θ_j has a G(α,β) distribution and the θ_j's are independent. We then add the constraint that the θ_j's are increasing convex. By increasing convex, I mean that they satisfy the following:

$$\theta \; \varepsilon \; S^{INCCON} = \{\theta: \theta_1>0,\ \theta_k<B,\ 0<\theta_2-\theta_1 < \ldots < \theta_k-\theta_{k-1}\},\text{ where}$$

$$B = \text{maximum allowable value for the } \theta_i\text{'s.}$$

We have $g(\vartheta_j)=\dfrac{\vartheta_i^{\alpha-1}e^{-\vartheta_i/\beta}}{\beta^\alpha\Gamma(\alpha)}$. Please note that ordinarily, the pdf of a G(α,β) distribution is defined to be $g(\vartheta_j)=\dfrac{\beta^\alpha\vartheta_i^{\alpha-1}e^{-\beta\vartheta_i}}{\Gamma(\alpha)}$, but throughout this thesis, I have replaced β by 1/β in order to remain consistent with Carlin's definition. From now on, we can omit the normalization constants of pdf's and use the symbol ∝ instead of =. We can rewrite $g(\theta_j)$ as:

$g(\vartheta_i) \propto \vartheta_j^{\alpha-1} \exp\left(\frac{-\vartheta_i}{\beta}\right) * I_{S_i}(\vartheta_i)$, where I is an indicator function and S_i is the subset of S which contains all acceptable θ_i.

I will now define the vector **y**. We have a group of N independent lives. For each unit age interval [x+j-1, x+j], we collect the following two things: (1) d_j, the number of deaths in the interval, and (2) e_j, the exposure for the interval, which is the total number of person-years the lives were under observation in the interval. More specifically, if we define e_{ij} to be the amount of time, measured in years, that life i was under observation between ages x+j-1 and x+j, then we have $e_j = \sum_{i=1}^{N} e_{ij}$. (See [3], p. 118). We define $\mathbf{y} = (d_1, \ldots, d_k, e_1, \ldots, e_k)$. The raw mortality rates are given by $\mathbf{r} = (r_1, \ldots, r_k)$, where $r_j = d_j / e_j$ for j = 1, …, k. The variable e_{ij} can only be equal to a maximum of 1 year. It could be less than a year, if the insured dies or terminates his policy.

Next we must define the conditional pdf $f(\mathbf{y}|\theta)$. The following comes from [3], p. 118: This function is the probability that each of the N lives survived for the whole period that we observed it (given that it survived up till the point when we began the observation) times the probability that the people who actually died did die. Let $x+s_i$ be the age at which observation of life i begins, and let $x+t_i$ be the age at which observation ceases. (The cessation may be caused by death, voluntary withdrawl, or termination of the observation period. Also, since we are only interested in the mortality rates between ages x and x+k, the cessation of observation could be due to the attainment of age x+k). Using this terminology, we have:

$$f(\mathbf{y}|\theta) = \prod_{i=1}^{N} p_i \prod_{i \varepsilon A} q_i$$

where p_i = Pr{Life i will survive from age $x+s_i$ to age $x+t_i$, given that it reaches $x+s_i$}, q_i = Pr{instantaneous death of life i at exactly age $x+t_i$}, and A denotes the subset of subscripts corresponding to those lives that died. In order to express the above mathematically, I must now go into a detailed explanation of life insurance terminology.

The following comes from [2], pages 45 and 49: In life insurance terminology, we define the survival function s(x) as s(x) = Pr(X>x), where the random variable X is the age at death of a newborn. The pdf of X is denoted f(x). The cdf of X is denoted F(x), and we have s(x) = 1 - F(x). As we know, the force of mortality, μ(x), is the probability of the instantaneous death of a person who is at age x. We can derive a formula for it by using x and Δx as follows: Given that a person has already attained age x, the probability that he or she will die between the ages of x and (x + Δx) is given by:

$$\Pr[x < X \le x + \Delta x | X > x] = \frac{F(x+\Delta x) - F(x)}{1 - F(x)} = \frac{F(x+\Delta x) - F(x)}{\Delta x} * \frac{\Delta x}{1 - F(x)}$$

As Δx approaches 0, it becomes like dx.

Also, as $\Delta x \longrightarrow 0$, $\frac{F(x+\Delta x) - F(x)}{\Delta x} \longrightarrow F'(x) = f(x)$

Therefore, we have

$$\lim_{\Delta x \to 0} \Pr[x < X \le x + \Delta x | X > x] = \frac{f(x) * dx}{1 - F(x)} = \frac{f(x) * dx}{s(x)} = \mu(x)dx \text{ .}$$

Because of this result, the force of mortality, μ(x), is given by:

$$\mu(x) = \frac{f(x)}{s(x)} = \frac{-s'(x)}{s(x)} = -\frac{d}{dx}\ln(s(x)) = -[\ln\{s(x)\}]' \text{ .}$$

Using this result, we can calculate s(x) from μ(x) as follows:

$$-\mu(x) = [\ln(s(x))]'$$

$$-\int_0^x \mu(t)dt = \ln[s(x)]$$

$$s(x) = \exp\left[-\int_0^x \mu(t)dt\right]$$

The following comes from [7], p. 119: We define ${}_tp_x$ to be the probability that a person age x will live t more years. Mathematically, we can express it as follows:

$${}_tp_x = \frac{\Pr(X > x+t)}{\Pr(X > x)} = \frac{s(x+t)}{s(x)} = \frac{\exp\left[-\int_0^{x+t} \mu(t)dt\right]}{\exp\left[-\int_0^{x} \mu(t)dt\right]} = \exp\left[-\int_0^{x+t} \mu(t)dt + \int_0^{x} \mu(t)dt\right] = \exp\left[-\int_x^{x+t} \mu(t)dt\right]$$

Similarly, the probability that life i will survive from age x+s_i to age x+t_i, given that it has already reached age x+s_i, is:

$$\exp\left[-\int_{x+s_i}^{x+t_i}\mu(t)dt\right]$$

The integral contained within this expression is:

$$\int_{x+s_i}^{x+t_i}\mu(t)dt$$

We want to express this integral in terms of the parameters in our vectors **y** and θ. The following comes from [3], p. 118: There are basically 2 things in this integral that we must take into account. By looking at the limits of integration, we see that one of them is the time that life i was under observation. By looking at the integrand, we see that the other thing is the force of mortality. As stated earlier, we have assumed that within each unit age interval [x+j-1, x+j] the force of mortality μ(t) is constant. This means that μ(t) is a step function on the interval [$x+s_i$, $x+t_i$]. We can break this whole interval down into unit age intervals. On each unit age interval [x+j-1, x+j], we take the product of e_{ij}, the amount of time that life i was under observation within the interval, and θ_j, the constant that the force of mortality is equal to within the interval. Summing the products of these two terms on every interval, we obtain:

$$\int_{x+s_i}^{x+t_i}\mu(t)dt = \sum_{j=1}^{k} e_{ij}\vartheta_j$$

This means that we have:

$$\exp\left[-\int_{x+s_i}^{x+t_i}\mu(t)dt\right] = \exp\left[-\sum_{j=1}^{k} e_{ij}\vartheta_j\right]$$

I stated earlier that this expression is the probability that life i will survive from age $x+s_i$ to age $x+t_i$, given that it has already reached age $x+s_i$. This is exactly what we defined p_i to be in the equation

$$f(\mathbf{y}|\theta) = \prod_{i=1}^{N} p_i \prod_{i\varepsilon A} q_i$$

In other words, we have:

$$p_i = \exp\left[-\sum_{j=1}^{k} e_{ij}\vartheta_j\right]$$

$$\prod_{i=1}^{N} p_i = \prod_{i=1}^{N}\exp\left[-\sum_{j=1}^{k} e_{ij}\vartheta_j\right] = \exp\left[-\sum_{i=1}^{N}\sum_{j=1}^{k} e_{ij}\vartheta_j\right]$$

Since i and j are independent, we have:

$$\prod_{i=1}^{N} p_i = \exp\left[-\sum_{j=1}^{k} \vartheta_j \sum_{i=1}^{N} e_{ij}\right] = \exp\left[-\sum_{j=1}^{k} \vartheta_j e_j\right] = \exp\left[-\sum_{j=1}^{k} e_j \vartheta_j\right]$$

As stated earlier, q_i is the probability of the instantaneous death of life i at exactly age $x+t_i$. This means that $q_i = \mu(x+t_i)$. The mortality rate within the interval [x+j-1, x+j] is θ_j. The number of deaths within the interval is d_j. If a life i dies, it dies at time $x+t_i$, since this is the time that the observation of the life ceases. Therefore, we have:

$$\prod_{i\varepsilon A} q_i = \prod_{I\varepsilon A} \mu(x+t_i) = \prod_{j=1}^{k} \vartheta_j^{d_j}$$

Using the above results, we can write f($\mathbf{y}$|θ) as:

$$\mathrm{f}(\mathbf{y}|\theta) = \prod_{i=1}^{N} p_i \prod_{i\varepsilon A} q_i = \left\{\exp\left[-\sum_{j=1}^{k} e_j \vartheta_j\right]\right\} \prod_{j=1}^{k} \vartheta_j^{d_j}$$

$$\mathrm{f}(\mathbf{y}|\theta) = \prod_{j=1}^{k} \vartheta_j^{d_j} \exp(-e_j \vartheta_j)$$

(See [3], p. 119).

The following comes from [3], pp. 68-69: The joint pdf of $\mathbf{y}$ and θ is:

$$\mathrm{p}(\mathbf{y},\theta) = \mathrm{f}(\mathbf{y}|\theta) \prod_{j=1}^{k} g(\vartheta_j) I_{S_j}(\vartheta_j)$$

where I is an indicator function and S_j is the subset of S^{INCCON} which contains all acceptable θ_j, i.e. those that meet the increasing convex criteria.

$$\mathrm{p}(\mathbf{y},\theta) = \left[\prod_{j=1}^{k} \vartheta_j^{d_j} \exp(-e_j \vartheta_j)\right] * \left[\prod_{j=1}^{k} \frac{\vartheta_j^{\alpha-1} \exp\left(\frac{-\vartheta_j}{\beta}\right)}{\beta^{\alpha}\Gamma(\alpha)} I_{S_j}(\vartheta_j)\right],$$

Since p($\mathbf{y}$,θ) depends on α and β, we should really write it as p($\mathbf{y}$,θ|λ), where λ=(α,β). The parameter α is a known constant, and β has an inverse gamma distribution, which we denote IG(a,b). In general, the inverse gamma distribution is defined such that if X has a G(a,b) distribution, then Y = 1/X has an IG(a,b) distribution. (In order to simulate values of Y, I always used Tadikamalla's method [15] to simulate X and then took the reciprocal). Omitting the normalization constant, we can rewrite p($\mathbf{y}$,θ|λ) as:

$$p(\mathbf{y},\theta|\lambda) \propto \left[\prod_{j=1}^{k} \vartheta_j^{\,d_j} \exp(-e_j \vartheta_j)\right] * \left[\prod_{j=1}^{k} \vartheta_j^{\,\alpha-1} \exp\left(\frac{-\vartheta_j}{\beta}\right) I_{S_j}(\vartheta_j)\right]$$

What we would really like to determine is the pdf $h(\theta|\mathbf{y})$. However, it would take an incredible amount of very difficult integration. Therefore, instead, we have to settle for knowing the conditional pdf's $h(\theta_i|\mathbf{y}, \lambda, \theta_{j\neq i})$ for i = 1, 2, …, k. If we hold all variables in $p(\mathbf{y}, \theta|\lambda)$ constant except for any one θ_i, we obtain:

$$h(\theta_i|\mathbf{y}, \lambda, \theta_{j\neq i}) \propto \vartheta_i^{\,d_i} \exp(-e_i \vartheta_i) * \vartheta_i^{\,\alpha-1} \exp\left(\frac{-\vartheta_i}{\beta}\right) * I_{S_i}(\vartheta_i)$$

$$h(\theta_i|\mathbf{y}, \lambda, \theta_{j\neq i}) \propto \vartheta_i^{\,d_i+\alpha-1} \exp\left[\frac{-\vartheta_i}{(\beta^{-1}+e_i)^{-1}}\right] * I_{S_i}(\vartheta_i)$$

We define $\alpha^* = \alpha + d_i$ and $\beta^* = (\beta^{-1}+e_i)^{-1}$ and write:

$$h(\theta_i|\mathbf{y}, \lambda, \theta_{j\neq i}) \propto G(\vartheta_i|\alpha^*,\beta^*) * I_{S_i}(\vartheta_i) \qquad \text{for } i = 1, 2, \ldots, k.$$

Carlin [5] states that the conditional distribution for β is:

$$h(\beta|\mathbf{y}, \theta) = IG\left(a + k\alpha, \left\{b^{-1} + \sum_{i=1}^{k} \vartheta_i\right\}^{-1}\right),$$

for which Carlin uses values of a=3 and b=115. Regarding α as a constant, we now have formulas for $f(\mathbf{y}|\theta)$, $p(\mathbf{y},\theta|\beta)$, $h(\theta_i|\mathbf{y},\beta,\theta_{j\neq i})$, and $h(\beta|\mathbf{y},\theta)$. The vector $(\theta|\mathbf{y},\beta)$ is the set of mortality rates that comes from the posterior distribution. We want to simulate values of this vector and use them to construct confidence intervals for each of the θ variables. To avoid doing an incredible amount of very difficult integration, we can use the Markov Chain Monte Carlo method known as the Gibbs sampler.

III. The Gibbs Sampler

The Gibbs sampler is a Markov Chain Monte Carlo (MCMC) technique that can be used to generate random observations from a joint distribution or marginal distribution, given all of the associated complete conditional distributions. Geman and Geman [8] invented the Gibbs sampler in 1984. In 1989, Goodman and Sokal [9] called it partial resampling.

The following comes from [5], p. 60: We can summarize the method of the Gibbs sampler as follows. Suppose we have a collection of n random variables **U** = (U_1, …, U_n). Suppose that the complete conditional distributions $g_i(U_i|U_{j \neq i})$, i = 1, 2, …, n are available for sampling. Under mild conditions (see [1]), these conditional distributions uniquely determine the full joint distribution, p(U_1, …, U_n), and hence all marginal distributions, p(U_i), i = 1, 2, …, n. Given an arbitrary starting set of values ($U_{1(0)}$, …, $U_{n(0)}$), we draw $U_{1(1)}$ from g_1(U_1|$U_{2(0)}$, …, $U_{n(0)}$), then $U_{2(1)}$ from g_2(U_2|$U_{1(1)}$, $U_{3(0)}$, …, $U_{n(0)}$), and so on up to $U_{n(1)}$ from g_n(U_n|$U_{1(1)}$, …, $U_{n-1(1)}$). So far we have obtained ($U_{1(1)}$, $U_{2(1)}$, …, $U_{n(1)}$). This completes one iteration. After t such iterations, we obtain ($U_{1(t)}$, $U_{2(t)}$, …, $U_{n(t)}$). Under mild conditions (see [8]), as $t \to \infty$, this n-tuple converges to a random observation from the joint density function p(U_1, …, U_n). Thus, if we choose t large enough, we can simply write (U_1, …, U_n) instead of ($U_{1(t)}$, $U_{2(t)}$, …, $U_{n(t)}$).

Carlin suggests repeating this entire process G times and using only the final set of values ($U_{1(t)}$, $U_{2(t)}$, …, $U_{n(t)}$) each time. He states that if we do this, we obtain independent and identically distributed n-tuples ($U_1^{(g)}$, …, $U_n^{(g)}$), g = 1, 2, …, G, from the joint distribution. (Actually, no matter how large we make t, these G n-tuples will always be correlated to some extent, however weak it may be. The larger we make t, the smaller the correlation will be). These n-tuple observations can be used to obtain information about the marginal distributions and the joint distributions. Generating the G n-tuples in this way requires discarding all of the intermediate values between the starting set and the final set each time. Since it doesn't matter whether we use the same starting values for each parallel sampling chain, a more efficient method is to make t very, very large and to just run the whole process once. We can make use of every set of values that is generated. In order to do so, however, we must examine the autocorrelation time of each of the variables, which is something that Carlin did not even take into account.

We treat each series of values {$U_{j(1)}$, $U_{j(2)}$, …, $U_{j(t)}$}, for j = 1, 2, …, n, as a time series. We compute the autocorrelations for this time series. The autocorrelation of order k is a measure of how correlated the values k values apart are. For a series $\{X_1, \ldots, X_n\}$, it is defined as

$$p_k = \frac{Cov(X_t, X_{t+k})}{Var(X)}$$

The sample autocorrelation function is:

$$r_k = \frac{\sum_{t=1}^{n-k}(x_t - \bar{x})(x_{t+k} - \bar{x})}{\sum_{t=1}^{n}(x_t - \bar{x})^2}$$

(See [4], p. 16). The following comes from [9]: The autocorrelation time is the time required, in terms of the number of values apart, for the autocorrelation to go to zero. More specifically, it is the variable τ, which is defined as follows:

$$\tau = 1 + 2\sum_{t=1}^{\infty} p_t$$

We will use an estimator of τ which is defined as follows:

$$\tau \geq 1 + 2\sum_{t=1}^{3\tau} r_t$$

We define the quantity n_{eff} as follows:

n_{eff} = (total number of samples) / (autocorrelation time)

It is the effective number of samples. We want n_{eff} to be as large as possible. In order to construct a confidence interval for the random variable X, represented by the time series $\{X_1, \ldots, X_n\}$, we need to determine what the error bar is. It is given by:

$$\text{error bar} = \sqrt{\frac{Var(X)}{n_{eff}}},$$

where $Var(X) = \frac{1}{n-1}\sum_{j=1}^{n}\left(x_j - \bar{x}\right)^2$.

A 95% confidence interval for X is $\bar{X} \pm (1.96)$(error bar).

I will now illustrate that the Gibbs sampler works by applying it to the two-variable case. Suppose that our collection of random variables is $\mathbf{U} = (U_1, U_2) = (X, Y)$, and suppose that the conditional distributions $g_1(x|y)$ and $g_2(y|x)$ are available for sampling. We will generate the following "Gibbs sequence" of random variables:

$$Y_0', X_0', Y_1', X_1', Y_2', X_2', \ldots, Y_t', X_t'.$$

We can specify the first value, Y_0'. The rest of the variables in the sequence are obtained iteratively by alternately generating values from:

$$X_j' \sim g_1(x \mid Y_j' = y_j')$$
$$Y_{j+1}' \sim g_2(y \mid X_j' = x_j')$$

As I stated before, under mild conditions, as $t \to \infty$, X_t' converges to X, an actual observation from $f_X(x)$. Similarly, Y_t' converges to Y, an actual observation from $f_Y(y)$. (See [6], p. 168).

Let's look at an example where we know the joint pdf, and hence the marginal pdf's, in advance. We will use the joint pdf to compute the conditional pdf's and then use the conditional pdf's in the Gibbs sampler. The test of whether the Gibbs sampler works will be to see if the observations of X that are generated by it yield accurate information about the actual marginal distribution of X, the pdf of which we know in advance.

Suppose that $f(x, y) = 2\, e^{-x-y}$ if $0 < x < y < \infty$. This pdf is given in [10]. We can compute the marginal pdf of X as follows:

$$f_X(x) = \int_x^\infty 2e^{-x-y}dy = -2e^{-x}\int_x^\infty -e^{-y}dy = (-2e^{-x})[e^{-y}]_x^\infty = (-2e^{-x})(0-e^{-x}) = 2e^{-2x};$$

$$f_X(x) = 2e^{-2x} \text{ ; x>0.}$$

We can compute the marginal pdf of Y as follows:

$$f_Y(y) = \int_0^y 2e^{-x-y}dx = 2e^{-y}\int_0^y e^{-x}dx = -2e^{-y}\int_0^y -e^{-x}dx = (-2e^{-y})[e^{-x}]_0^y$$
$$= (-2e^{-y})(e^{-y}-1) = 2e^{-y}(1-e^{-y})$$
$$f_Y(y) = 2e^{-y}(1-e^{-y}) = 2e^{-y} - 2e^{-2y} \text{ ; y>0.}$$

Now let's compute the conditional pdf's, which we need to know in order to apply the Gibbs sampler. We can do this as follows:

The conditional pdf of Y given X is:

$$f_{Y|X}(y|x) = \frac{f(x,y)}{f_X(x)} = \frac{2e^{-x-y}}{2e^{-2x}} = e^{x-y} \text{ for } 0 < x < y < \infty.$$

Its cdf is:

$$F_{Y|X}(y|x) = \int_x^y f_{Y|X}(y|x) = \int_x^y e^x e^{-v} dv = -e^x [e^{-v}]_x^y = -e^{-x}(e^{-y} - e^{-x}) = e^{-x}(e^{-x} - e^{-y})$$

In order to simulate a random observation from this distribution, we generate a random number U on [0,1] and calculate y as follows:

$$U = e^{-x}(e^{-x} - e^{-y})$$
$$Ue^x = e^{-x} - e^{-y}$$
$$e^{-y} = e^{-x} - Ue^x$$
$$y = -\ln(e^{-x} - Ue^x)$$

The conditional pdf of X given Y is:

$$f_{X|Y}(x|y) = \frac{f(x,y)}{f_Y(y)} = \frac{2e^{-x-y}}{2e^{-y}(1-e^{-y})} = \frac{e^{-x}}{1-e^{-y}} \text{ for } 0 < x < y < \infty.$$

Its cdf is:

$$F_{X|Y}(x|y) = \int_0^x f_{X|Y}(v|y)dv = \frac{1}{1-e^{-y}} \int_0^x e^{-v} dv = \frac{-1}{1-e^{-y}} \int_0^x -e^{-v} dv = \frac{-[e^{-v}]_0^x}{1-e^{-y}} = \frac{1-e^{-x}}{1-e^{-y}}$$

In order to simulate a random observation from this distribution, we generate a random number U on [0,1] and calculate x as follows:

$$U = \frac{1-e^{-x}}{1-e^{-y}}$$
$$1-e^{-x} = U(1-e^{-y})$$
$$e^{-x} = 1 - U(1-e^{-y})$$
$$x = -\ln[1 - U(1-e^{-y})]$$

When simulating from $F_{X|Y}(x|y)$ and $F_{Y|X}(y|x)$, we may have to simulate repeatedly until we get an acceptable value of X or Y, that is, one that satisfies the relationship $0 < X < Y < \infty$. First we must come up with an appropriate starting value for Y_0'. The marginal pdf of X is exponential with mean 0.5. Let's suppose that U = 0.2. Using the formula for $F_{Y|X}(y|x)$, let's define $Y_0' = -\ln\left[e^{-0.5} - 0.2e^{0.5}\right] = 1285$.

I used 100 iterations of the Gibbs sampler to generate each value of X, and I generated 10,000 values of X. The following chart shows a breakdown of the actual and expected results:

Lower Bd.	Upper Bd.	Gibbs	Expected	Percent Deviation
0	0.1	1764	1813	-2.70
0.1	0.2	1596	1484	7.55
0.2	0.3	1178	1215	-3.05
0.3	0.4	960	995	-3.52
0.4	0.5	842	814	3.44
0.5	0.6	660	667	-1.05
0.6	0.7	542	546	-0.73
0.7	0.8	478	447	6.94
0.8	0.9	370	366	1.09
0.9	1	307	300	2.33
1	1.1	240	245	-2.04
1.1	1.2	204	201	1.49
1.2	1.3	170	164	3.66
1.3	1.4	124	135	-8.15
1.4	1.5	111	110	0.91
1.5	1.6	73	90	-18.89
1.6	1.7	69	74	-6.76
1.7	1.8	50	60	-16.67
1.8	1.9	48	50	-4.00
1.9	2	39	41	-4.88
2	2.1	27	33	-18.18
2.1	2.2	20	27	-25.93
2.2	2.3	22	22	0.00
2.3	2.4	16	18	-11.11
2.4	2.5	19	15	26.67
2.5	2.6	9	12	-25.00
2.6	2.7	8	10	-20.00
2.7	2.8	6	8	-25.00
2.8	2.9	16	7	128.57
2.9	3	1	5	-80.00
3	3.1	6	4	50.00
3.1	3.2	3	4	-25.00
3.2	3.3	4	3	33.33
3.3	3.4	3	2	50.00
3.4	3.5	3	2	50.00
3.5	Infinity	12	9	33.33

Since the results obtained from the Gibbs sampler are very similar to the expected results, we can conclude that the Gibbs sampler is a very effective method.

IV. Simulating Random Observations From The Gamma Distribution Using Tadikamalla's Method

We want to generate values of the random variable X, where X has a gamma (α, β) distribution. The pdf of X is given as follows:

$$g(x) = \frac{x^{\alpha-1}e^{-x/\beta}}{\beta^{\alpha}\Gamma(\alpha)}\text{; x>0.}$$

As I stated in Section II of this thesis, the pdf of a gamma (α, β) distribution is ordinarily defined to be:

$$g(x) = \frac{\beta^{\alpha}x^{\alpha-1}e^{-\beta x}}{\Gamma(\alpha)}\text{; x>0.}$$

However, throughout this thesis, I have replaced β by $1/\beta$ in order to remain consistent with Carlin's [5] definition.

To generate a value of X, we can first generate a value of X from a gamma (α, 1) distribution and then perform a transformation. The pdf of the gamma (α, 1) distribution is:

$$f(x) = \frac{x^{\alpha-1}e^{-x}}{\Gamma(\alpha)}\text{; x>0.}$$

Notice that this is simply g(x) with $\beta = 1$.

Let's suppose that α is an integer, and let's define the constant $m = \alpha$. The moment-generating function for a gamma (α, β) distribution, given by g(x), is:

$$M_X(t) = \left(\frac{\frac{1}{\beta}}{\frac{1}{\beta}-t}\right)^{\alpha}\text{; t<1/}\beta.$$

Therefore, the moment-generating function for a gamma (α, 1) distribution, given by f(x), is:

$$M_X(t) = \left(\frac{1}{1-t}\right)^{\alpha}\text{; t<1.}$$

The exponential distribution is a special case of the gamma (α, β) distribution with $\alpha=1$. Therefore, the moment-generating function for an exponential distribution with parameter $1/\beta$ is:

$$M_X(t) = \left(\frac{\frac{1}{\beta}}{\frac{1}{\beta} - t} \right); \text{ t<1/β.}$$

If β=1, then:

$$M_X(t) = \left(\frac{1}{1-t} \right); \text{ t<1.}$$

Suppose we have a collection of m variables $\{X_1, \ldots, X_m\}$ and that each X_i has an exponential distribution with parameter β=1. Let's define $S = X_1 + \ldots + X_m$. The moment-generating function of S is:

$$M_S(t) = M_{X_1}(t) * M_{X_2}(t) * * * * M_{X_m}(t)$$

$$M_S(t) = \left(\frac{1}{1-t} \right) * \left(\frac{1}{1-t} \right) * * * * \left(\frac{1}{1-t} \right) = \left(\frac{1}{1-t} \right)^m$$

This shows that S has a gamma distribution with parameters α=m and β=1. So, if α is an integer, to simulate values from the gamma (α, 1) distribution, represented by f(x), we need to simulate α values of X from an exponential distribution with parameter β=1. The sum of these α values is the desired result.

The pdf of the exponential distribution with parameter 1/β is

f(x) = $\frac{1}{\beta} \exp\left(-\frac{x}{\beta} \right)$; x>0.

The cdf is F(x) = 1 - exp(-x/β).

Suppose U is a random number between 0 and 1. Then we have:

$$U = 1 - \exp(-x/\beta)$$
$$\exp(-x/\beta) = 1-U$$
$$-x/\beta = \ln(1-U)$$
$$x = -\beta \ln(1-U)$$

Since (1-U) is also a random number between 0 and 1, we can rewrite this as:

$$x = -\beta \ln(U)$$

If β=1, then x = -ln(U), where x is a random observation from an exponential distribution with parameter β=1.

If we have $U_1, U_2, \ldots, U_m$ and $X_i = -\ln(U_i)$ for i = 1, 2, …, m, then

$$S = X_1 + X_2 + \ldots + X_m = -\ln(U_1) - \ln(U_2) - \ldots - \ln(U_m)$$

$$S = -\ln(U_1 * U_2 **** U_m) = -\ln \prod_{i=1}^{m} U_i$$

S is a random observation from a gamma distribution with parameters $\alpha = m$ and $\beta = 1$. As I stated at the beginning of this section, the pdf for S is given by:

$$f(x) = \frac{x^{\alpha-1}e^{-x}}{\Gamma(\alpha)};\ x>0.$$

Let's consider the transformation $Y = \beta X$. Then we have:

$$X = Y/\beta$$

$$dx = dy/\beta$$

$$f(x) = f(y/\beta) = \frac{\left(\frac{y}{\beta}\right)^{\alpha-1} \exp\left(-\frac{y}{\beta}\right) * \frac{1}{\beta}}{\Gamma(\alpha)} = \frac{y^{\alpha-1}\exp\left(-\frac{y}{\beta}\right)}{\beta^{\alpha}\Gamma(\alpha)} = g(y)$$

We see that the result we get, g(y), is the pdf of a gamma (α, β) distribution. So, to generate Y from a gamma (α, β) distribution, we first generate X from a gamma (α, 1) distribution and then make the transformation $Y = \beta X$.

All along, we have assumed that α is an integer. We have already established that if α is an integer, then $X = -\ln \prod_{i=1}^{m} U_i$ is a random observation from a gamma (α, 1) distribution. Now let's suppose that α is not an integer, in which case we must use a version of the rejection algorithm. The following was taken from [15], p. 420: We define $m = [\alpha]$ = the integer portion of α, such that $\alpha \geq m$, and we define $p = \alpha - m$ as the fractional portion of α. Since we would expect a larger X if α is larger, it is natural to consider taking $X = \left(-\ln \prod_{i=1}^{m} U_i\right)(\alpha / m)$. Tadikamalla [15] proposed the following version of the rejection algorithm:

1. Generate m independent random numbers $U_1, U_2, \ldots, U_m$ and compute $X = \left(-\ln \prod_{i=1}^{m} U_i\right)(\alpha / m)$.
2. Generate another random number r.

3. Compute T(X) = $\left(\frac{X}{\alpha}\right)^p \exp\left(-p\left[\frac{X}{\alpha}-1\right]\right)$

4. If r $\leq$ T(X), return X as the desired value. Otherwise, go to Step 1.

I have used this algorithm at times to generate gamma variates in this thesis.

V. Simulating From The Gamma Distribution Using The Metropolis Algorithm

The following was taken from [12], pp. 73-81, unless otherwise stated: The Metropolis algorithm is a very powerful rejection technique that can be used to sample almost any density function in any number of dimensions. It was invented in 1953 by Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller, which is why it is sometimes called the $M(RT)^2$ algorithm. This method was motivated by an analogy with the behavior of systems in statistical mechanics that approach an equilibrium whose statistical properties are independent of the kinetics of the system. By a system, I mean a point x in a space Ω, typically R^m, that may be thought of as a description of a physical problem. By kinetics, I mean a stochastic transition that governs the evolution of the system. This can be represented by a probability density function K(X|Y) which gives the probability that the evolution of a system known to be at Y will next bring it near X. One condition that a system evolve toward equilibrium and stay there is that the system be on the average as likely to move from a neighborhood of Y into a neighborhood of X as to move exactly in the reverse direction. If the probability density function for observing the system near X in equilibrium is f(X), then the kinetics must satisfy the following equation:

$$K(X|Y)\, f(Y) = K(Y|X)\, f(X)$$

This relation is called detailed balance. The left side of the equation, K(X|Y) f(Y), is the probability of moving from Y to X. The right side of the equation, K(Y|X) f(X), is the probability of moving from X to Y.

To begin with, we are given f(X). In order to use the Metropolis Algorithm, we have to come up with a convenient and correct kinetics, given by the function K, that will equilibrate the system so that the given f(X) turns out to be the chance of observing the system near X. We can use any distribution T(X'|Y) to yield transitions from Y to X'. Then on comparing f(X') with f(Y) and taking into account T as well, the system is either moved to X' (move accepted) or returned to Y (move rejected). The move from Y to X' is accepted with probability A(X'|Y), which must be calculated so as to satisfy detailed balance.

We calculate K as K(X|Y) = A(X|Y) T(X|Y). We can rewrite the detailed balance condition as:

$$A(X|Y)\ T(X|Y)\ f(Y) = A(Y|X)\ T(Y|X)\ f(X)$$

From the detailed balance condition, we obtain:

$$\frac{T(Y|X)f(X)}{T(X|Y)f(Y)} = \frac{A(X|Y)}{A(Y|X)}$$

We assume that it is possible to move from X to Y if one can move from Y to X and vice versa. We define a quantity q(X|Y) as:

$$q(X|Y) = \frac{T(Y|X)f(X)}{T(X|Y)f(Y)} \geq 0$$

Using q(X|Y), we must define A(X|Y) in such a way as to satisfy detailed balance. One frequently used definition is:

$$A(X|Y) = \min\ (1, q(X|Y))$$

I will now show that this choice of A(X|Y) does satisfy detailed balance. We have:

$$q(X|Y) = \frac{T(Y|X)f(X)}{T(X|Y)f(Y)} = \frac{A(X|Y)}{A(Y|X)} = \frac{\min(1, q(X|Y))}{\min(1, q(Y|X))}$$

We see from the definition of q(X|Y) that q(Y|X) = 1 / q(X|Y).
If q(X|Y) > 1, we have q(Y|X) < 1. Thus, we obtain:

$$q(X|Y) = \frac{\min(1, q(X|Y))}{\min(1, q(Y|X))} = \frac{1}{q(Y|X)} = q(X|Y)$$

If q(X|Y) < 1, we have q(Y|X) > 1. Thus, we obtain:

$$q(X|Y) = \frac{\min(1, q(X|Y))}{\min(1, q(Y|X))} = \frac{q(X|Y)}{1} = q(X|Y)$$

If q(X|Y) = 1, then we have:

$$q(X|Y) = \frac{\min(1, q(X|Y))}{\min(1, q(Y|X))} = \frac{1}{1} = 1 = q(X|Y)$$

So we see that our choice of A(X|Y) does satisfy detailed balance. Note that our definition of q(X|Y) is very convenient because we never need to know the normalization constant in f(X) – it cancels out in the fraction.

The Metropolis Algorithm generates a random walk. Suppose that X_1, X_2, X_3, …, X_n are the steps in the random walk. Each X_j is a random variable and has an associated probability $\Phi_j(X)$. The $\Phi_n(X)$ have the property that asymptotically

$$\lim_{n \longrightarrow \infty} \Phi_n(X) = f(X)$$

We can now describe the Metropolis Algorithm concretely as follows. At step n of the random walk, the value of X is X_n. A possible next value for X, X_{n+1}', is sampled from $T(X_{n+1}'|X_n)$, and the probability of accepting X_{n+1}', $A(X_{n+1}'|X_n)$, is computed. If $q(X_{n+1}'|X_n) > 1$, then $A(X_{n+1}'|X_n) = 1$, which means that we accept X_{n+1}', i.e. we set $X_{n+1} = X_{n+1}'$. If $q(X_{n+1}'|X_n) < 1$, then $A(X_{n+1}'|X_n) = q(X_{n+1}'|X_n)$. We generate a random number U on the interval [0,1]. If $U < q(X_{n+1}'|X_n)$, then we accepet X_{n+1}', i.e. we set $X_{n+1} = X_{n+1}'$. If $U \geq q(X_{n+1}'|X_n)$, then we reject X_{n+1}' and use the old value, i.e. $X_{n+1} = X_n$. Notice that if X_{n+1}' is not accepted, we use the previous value rather than sample a new value.

By using the Metropolis Algorithm, we are guaranteed to sample f(X), but only asymptotically. Therefore, we must throw away L steps of the random walk before we actually get a sample from f(X). The number L is usually very difficult to estimate in advance. Usually quite a bit of trial and error is used to estimate an appropriate value.

The rest of this section was not taken from [12]: In order to make q(X|Y) simpler to deal with, it is convenient to define T(X|Y) to be the pdf of a uniform distribution. In this thesis, whenever I have applied the Metropolis algorithm, I have done this in the following way: Suppose that the random variable U has a u[0,1] distribution. Let's define $t = U - \frac{1}{2}$. Then t has a $u\left[-\frac{1}{2}, \frac{1}{2}\right]$ distribution. Let's define v = st, where s is a constant that we call the step size. Then v has a $u\left[\frac{-s}{2}, \frac{s}{2}\right]$ distribution. Given a fixed value of Y, let's generate a value of X by the equation X = Y + v. This means that T(X|Y) is the pdf of a uniform distribution on the interval $\left[Y - \frac{s}{2}, Y + \frac{s}{2}\right]$. So T(X|Y) is given by:

$$T(X|Y) = \frac{1}{\left(Y + \frac{s}{2}\right) - \left(Y - \frac{s}{2}\right)} = \frac{1}{s}$$

Similarly, T(Y|X) is the pdf of a uniform distribution on the interval $\left[X - \frac{s}{2}, X + \frac{s}{2}\right]$. It is given by:

$$T(Y|X) = \frac{1}{\left(X + \frac{s}{2}\right) - \left(X - \frac{s}{2}\right)} = \frac{1}{s}$$

Therefore, in the fraction for q, the T functions cancel out and we get:

$$q(X|Y) = \frac{T(Y|X)f(X)}{T(X|Y)f(Y)} = \frac{f(X)}{f(Y)}$$

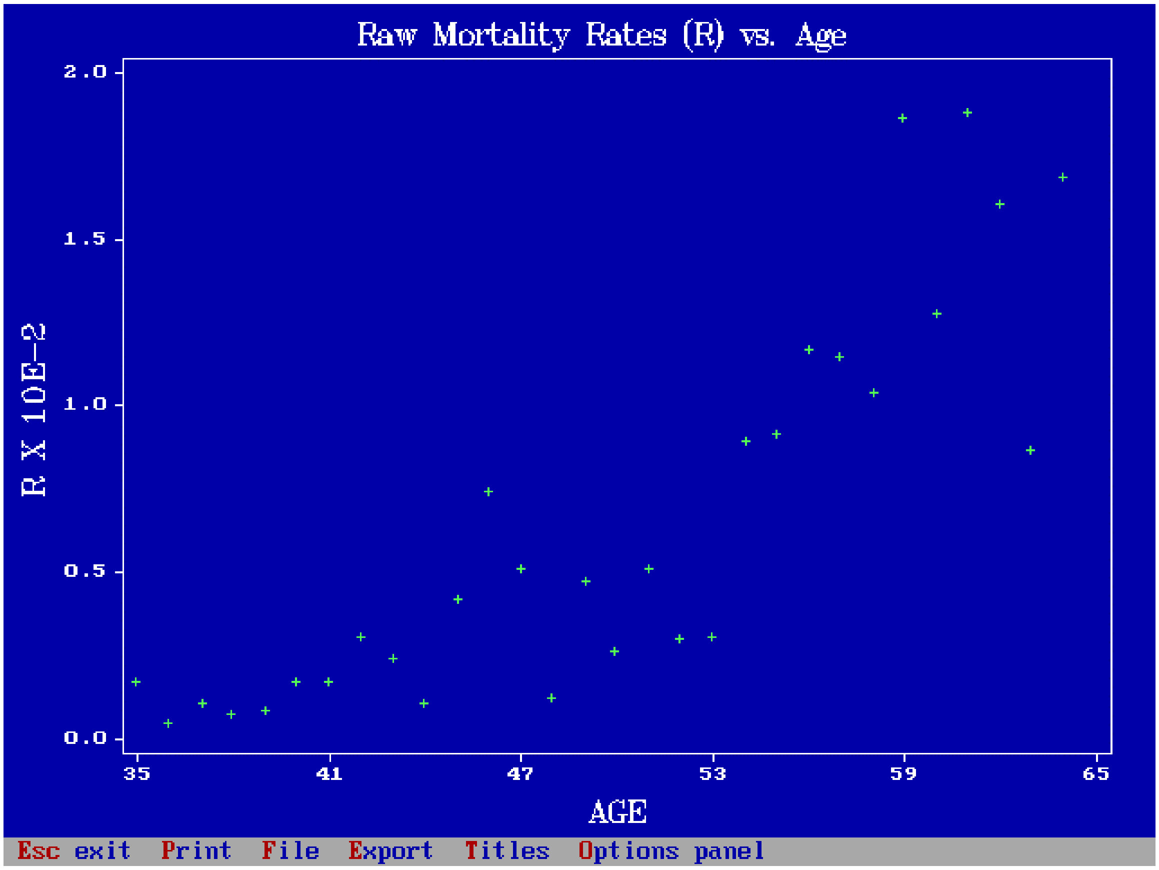
VI. Application Of The Gibbs Sampler To Bayesian Graduation Of Mortality Rates

In this thesis, I have explained how to apply the Gibbs Sampler to a general case in which we have a collection of n random variables $\mathbf{U} = (U_1, U_2, \ldots, U_n)$. We want to apply the Gibbs sampler to the case where we have a collection of k+1 variables $\mathbf{U} = (\theta_1, \theta_2, \ldots, \theta_k, \beta)$. Although we use β in the Gibbs sampler, we are really only interested in the k θ-variables. First, we must come up with starting values for the k+1 variables. The raw mortality rates are given by the vector $\mathbf{r} = (r_1, \ldots, r_k)$. Based on a quadratic regression of r_i on i, Carlin [5] suggests using the values $\theta_i = 0.0000222\ i^2$. We start our Bayesian graduation with a prior assumption that each θ_i has a $G(\alpha,\beta)$ distribution. Carlin uses a value of 1.49 for α and suggests using an initial value of 0.00435 for β. These are the values that I used. When I defined what is meant by increasing convex, I stated that the constant B is the maximum allowable value for the θ_i's. When the number of variables was 30, I defined B to be 0.025, as Carlin did. When I reduced the number of variables to k, where k<30, I defined B to be the initial estimate of θ_{k+1}, in other words $0.0000222\ (k+1)^2$.

Throughout this thesis, the subscripts 1, …, 30 correspond to the ages 35, …, 64. The set of raw mortality rates that was analyzed by Carlin is as follows:

Subscript	Age	d	e	r
1	35	3	1771.5	1.693E-03
2	36	1	2126.5	4.703E-04
3	37	3	2743.5	1.093E-03
4	38	2	2766	7.231E-04
5	39	2	2463	8.120E-04
6	40	4	2368	1.689E-03
7	41	4	2310	1.732E-03
8	42	7	2306.5	3.035E-03
9	43	5	2059.5	2.428E-03
10	44	2	1917	1.043E-03
11	45	8	1931	4.143E-03
12	46	13	1746.5	7.443E-03
13	47	8	1580	5.063E-03
14	48	2	1580	1.266E-03
15	49	7	1467.5	4.770E-03
16	50	4	1516	2.639E-03
17	51	7	1371.5	5.104E-03
18	52	4	1343	2.978E-03
19	53	4	1304	3.067E-03
20	54	11	1232.5	8.925E-03
21	55	11	1204.5	9.132E-03
22	56	13	1113.5	1.167E-02
23	57	12	1048	1.145E-02
24	58	12	1155	1.039E-02
25	59	19	1018.5	1.865E-02
26	60	12	945	1.270E-02
27	61	16	853	1.876E-02
28	62	12	750	1.600E-02
29	63	6	693	8.658E-03
30	64	10	594	1.684E-02

This is a dataset of male ultimate (duration ≥ 16) experience that was originally analyzed by Broffitt [3]. The term ultimate, with duration ≥ 16, means that for each life in the table, the number of years between the issue age (i.e. the age at which the life was underwritten) and the point in time that the life is being considered in the dataset is greater than or equal to 16.



In Section V of this thesis, I have explained what the Metropolis algorithm is. In order to use it, we must come up with a step size, which we call s, for each variable. In order to do this, I followed the following procedure: I ran my program 200 times, each time using a different step size but using the same step size for all 30 variables. I let the variable J go from 100 through 20,000 in multiples of 100 and used 1/J as the step size each time. Each time, for each of the 30 variables, I recorded the average number of rejections (maximum = 10) required before obtaining an acceptable value of the θ variable. This yielded a table with 200 rows and 30 columns. I went down each of the columns and found the average number of rejections closest to 5, i.e. I looked for approximately a 50% acceptance fraction. I selected the step size corresponding to this row as the step size for the θ variable corresponding to the column. The final step sizes that I selected by this process, along with the average number of rejections that they generated, are given in the following table:

Subscript	StepSize	AvgRejections
1	9.00990E-05	5.00000
2	6.66667E-05	4.99000
3	6.89655E-05	5.01500
4	9.25926E-05	4.98000
5	8.92857E-05	4.94000
6	9.34579E-05	4.96500
7	9.61538E-05	5.09500
8	8.62069E-05	4.98500
9	9.34579E-05	4.97500
10	7.87402E-05	4.98000
11	7.29927E-05	4.99500
12	8.00000E-05	4.98500
13	8.13008E-05	4.96500
14	8.13008E-05	5.02000
15	8.84956E-05	5.01000
16	8.62069E-05	4.96500
17	9.25926E-05	4.98500
18	1.02041E-04	4.96000
19	7.09220E-05	4.97500
20	6.25000E-05	5.04000
21	6.49351E-05	5.06000
22	6.49351E-05	5.11000
23	6.75676E-05	5.07500
24	6.89655E-05	5.01500
25	8.77193E-05	5.00000
26	9.70874E-05	5.05000
27	3.84615E-04	4.95000
28	5.88235E-04	5.03000
29	1.25000E-03	5.22500
30	5.00000E-03	5.12000

I used the Metropolis algorithm to simulate all of the values for each θ_i for i = 1, 2, …, k. Every time I applied the Metropolis algorithm, I ran a maximum of 10 steps of it. At the beginning of each step, I tested to see if the new value being simulated fit the increasing convex condition. If it didn't, I rejected it even before carrying out the rest of the algorithm for that step. After 10 steps, if all 10 values were rejected, I simply kept the old value of θ_i.

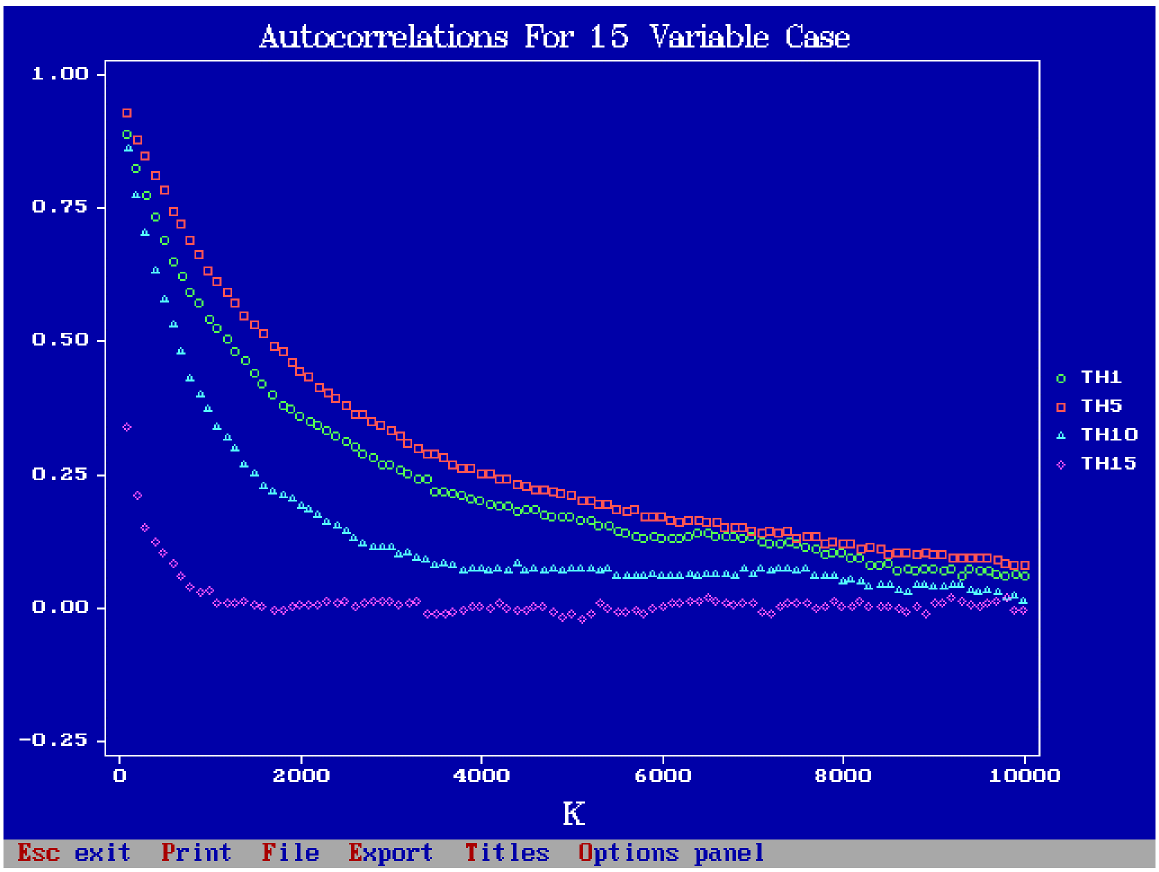
However, I wanted to compare the efficiency of Tadikamalla's method with that of the Metropolis algorithm. A good measure of the efficiency is the average number of rejections required before a new value of θ_j is generated. In order to measure this, I allowed the number of rejections to be unlimited. I ran 100 iterations of the Gibbs sampler for 30 variables with the requirement that each value of θ_j must be different than its preceding value. I did this twice, the first time simulating the values of all the θ variables using Tadikamalla's method and the second time using the Metropolis algorithm with no limit on the number of steps. The average number of rejections for the two methods are as follows:

AVERAGE NUMBER OF REJECTIONS

Subscript	Tadikamalla's method	Metropolis algorithm
1	10049.940	1.540
2	134.380	1.570
3	614.680	1.010
4	123.440	1.600
5	93.740	1.810
6	231.600	2.100
7	148.620	2.150
8	662.580	1.030
9	218.520	0.900
10	237.080	0.500
11	431.120	1.090
12	8163.500	1.450
13	291.940	2.200
14	1412.540	2.920
15	177.320	2.730
16	1354.360	4.980
17	412.800	3.410
18	3219.340	2.840
19	7498.880	1.420
20	634.180	1.670
21	378.980	1.080
22	300.860	0.610
23	437.760	0.820
24	793.200	0.450
25	472.000	0.390
26	410.160	0.340
27	159.580	1.640
28	167.980	1.820
29	4147.440	2.020
30	72.720	3.940

Since the average number of rejections required at each step for Tadikamalla's method is much greater than that for the Metropolis algorithm, we see that the Metropolis algorithm is much more efficient.

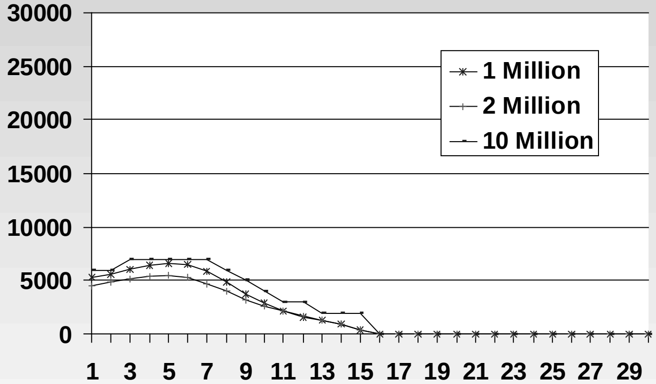
Carlin [5] applied the Gibbs sampler to the case where there are 30 variables. In addition to doing this, I applied it to the cases where there are 15, 20, and 25 variables. For each of the variables, we want to compute the autocorrelation time, which is the time required, in terms of the number of values apart, for the autocorrelation to go to zero. First, let's look at a graph of autocorrelation for some variables for the case where there are 15 variables:



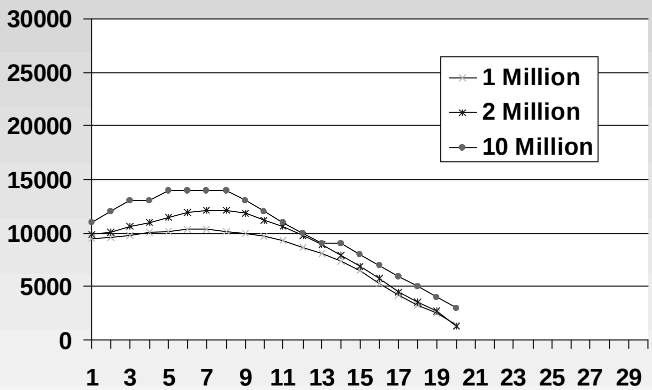
This graph shows the autocorrelations for θ_1, θ_5, θ_{10}, and θ_{15}, which are denoted TH1, TH5, TH10, and TH15, respectively. We see that the autocorrelation steadily decreases over time. The same pattern occurs for all variables in all cases.

For cases where there are 15, 20, 25, and 30 variables, I tried applying the Gibbs sampler in 3 different ways: (1) having 1 million iterations and using every hundredth one, (2) having 2 million iterations and using every hundredth one, and (3) having 10 million iterations and using every thousandth one. For all these different cases, I plotted autocorrelation time and $\sqrt{n_{eff}}$. When I used every hundredth iteration, I multiplied the autocorrelation time by 100. When I used every thousandth iteration, I multiplied the autocorrelation time by 1000. The graphs are as follows:

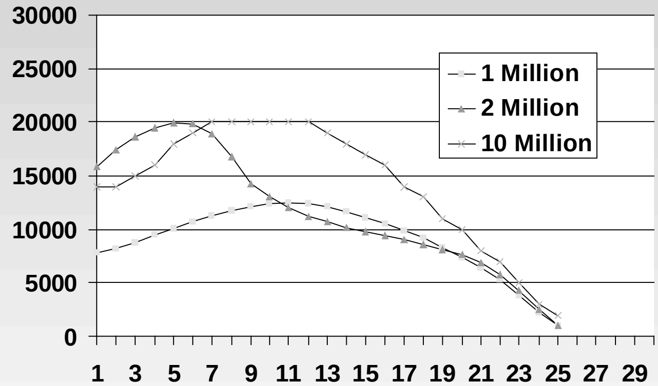
Autocorrelation Times For 15 Variables



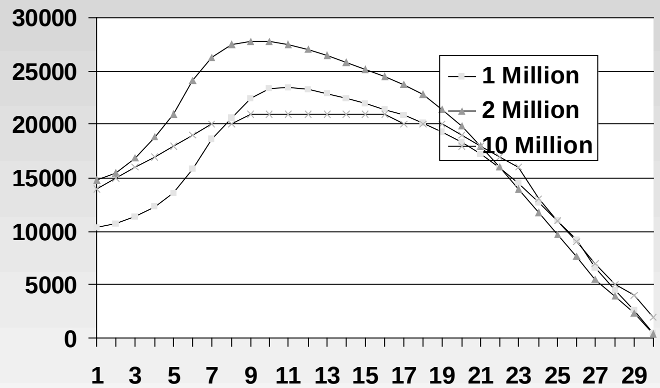
Autocorrelation Times For 20 Variables



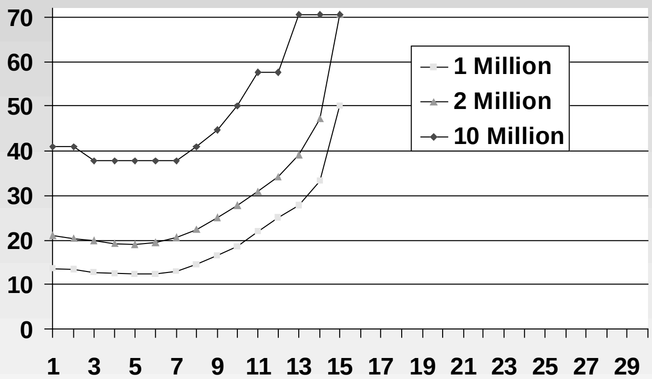
Autocorrelation Times For 25 Variables



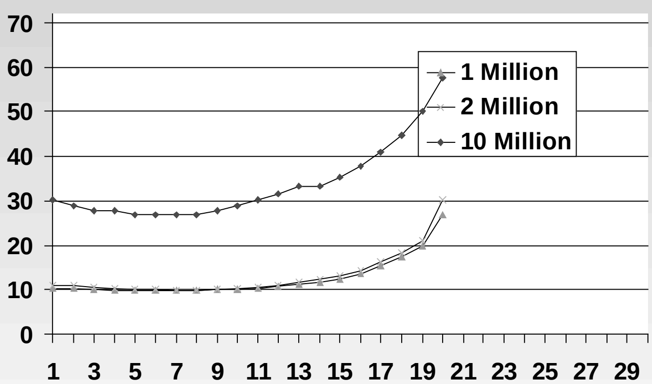
Autocorrelation Times For 30 Variables



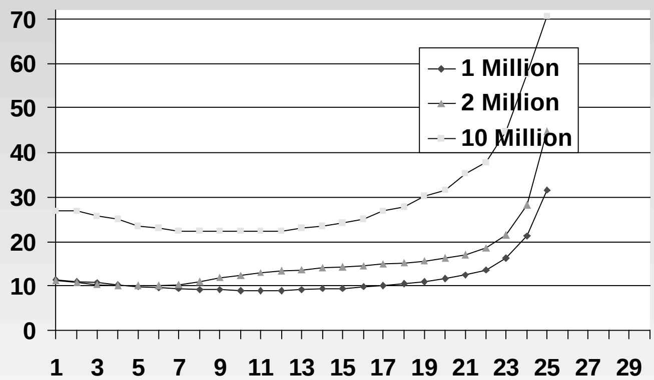
Square Root Of n_{eff} For 15 Variables



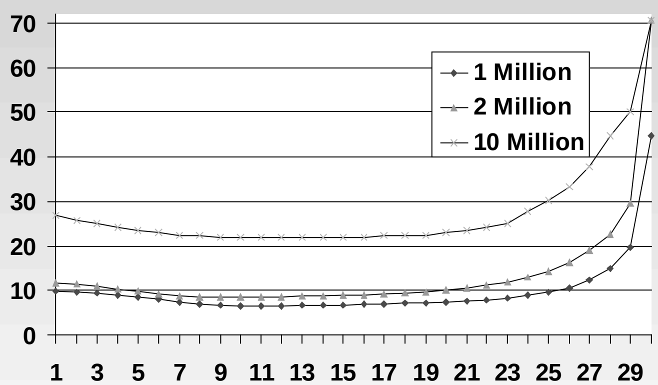
Square Root Of n_{eff} For 20 Variables



Square Root Of n_{eff} For 25 Variables



Square Root Of n_{eff} For 30 Variables



We would expect that, as long as we're using a very large number of iterations with the same number of variables, the autocorrelation time should remain the same. Comparing the autocorrelation time for 1, 2, and 10 million iterations, we see that it only remains approximately the same for the case where there are 15 variables. When we have 20, 25, and 30 variables, the autocorrelation time for 1, 2, and 10 million iterations varies considerably. This suggests that we are seeing statistical noise here and that in order to get a true representation of the autocorrelation time, we have to run even more iterations. Since it would be impractical to run more than 10 million iterations, we conclude that the method described in this thesis can only be effectively applied when the number of variables is less than or equal to 15.

Earlier in this thesis, we defined n_{eff}, the effective number of samples, as:

$$n_{eff} = (\text{total number of samples}) / (\text{autocorrelation time})$$

This means that

$$\sqrt{n_{eff}} = (\text{total number of samples})^{1/2} / (\text{autocorrelation time})^{1/2}$$

Since we expect the autocorrelation time to be the same regardless of the number of iterations used, if we increase the total number of samples by a factor k, we expect $\sqrt{n_{eff}}$ to increase by a factor of $\sqrt{k}$. This means that $\sqrt{n_{eff}}$ for 2 million iterations should be $\sqrt{2}$, or about 1.4, times greater than $\sqrt{n_{eff}}$ for 1 million and that $\sqrt{n_{eff}}$ for 10 million should be $\sqrt{10}$, or about 3.2, times greater than $\sqrt{n_{eff}}$ for 1 million. Let's examine the graphs of $\sqrt{n_{eff}}$ for the case where there are 15 variables. We see that $\sqrt{n_{eff}}$ for 10 million remains constant for variables 13, 14, and 15. Aside from this minor detail, the graphs for 1 million, 2 million, and 10 million approximately fit the pattern that I just described involving the factors of 1.4 and 3.2. The graphs for 20, 25, and 30 variables fall short of this pattern. Let's examine the graphs of $\sqrt{n_{eff}}$ for the case where there are 20 variables. Let's compare the graphs for 1 million and 2 million iterations. The graphs are almost identical, whereas one should be about 1.4 times the other. The reason why they are nearly identical is that the autocorrelation time for

2 million is greater than the autocorrelation time for 1 million. A greater autocorrelation time yields a lower $\sqrt{n_{eff}}$. Let's compare the graphs for 1 million and 10 million iterations. The graph for 1 million remains nearly 10 for variables 1 through 10. Therefore, we would expect the graph for 10 million to be consistently above 30 for these variables. However, it dips significantly below 30, to as low as 26 or 27. Let's examine the graphs of $\sqrt{n_{eff}}$ for the case where there are 25 variables. Let's compare the graphs for 1 million and 2 million iterations. The graphs remain almost identical for variables 1 through 7, whereas one should be about 1.4 times the other. We see the same exact thing happening with the graphs for 1 million and 2 million iterations for the case where there are 30 variables. Thus, we see again that the method described in this thesis can only be effectively applied when the number of variables is less than or equal to 15.

It is interesting to examine autocorrelation time as a function of age, i.e. the variable subscripts. Regardless of the number of variables, the autocorrelation time always starts out low, gradually increases to a point, and then gradually decreases. The middle ages have the highest autocorrelation time, and the early and late ages have the lowest autocorrelation time. I stated earlier in this thesis that n_{eff}, the effective number of samples, is defined as follows:

$$n_{eff} = \text{(total number of samples) / (autocorrelation time)}$$

In other words, n_{eff} is proportional to the reciprocal of the autocorrelation time. Because of this, the middle ages have the lowest n_{eff}, and the early and late ages have the highest n_{eff}. For the late ages, n_{eff} is much higher than for the early ages.

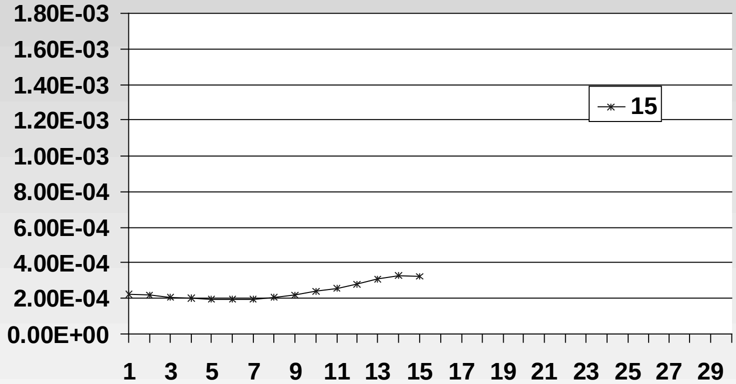
Although we can only feel confident about applying the method described in this thesis when the number of variables is less than or equal to 15, I applied the method anyway for the cases where there are 20, 25, and 30 variables, since it will give us some estimate of the confidence intervals for the mortality rates. When constructing confidence intervals, I have limited myself to the cases where there are only 1 million iterations.

In order to construct confidence intervals, we need to calculate error bars. For each case for each variable, we have a time series of values. As stated earlier in this thesis, the error bar is equal to the sample standard deviation of the series of values divided by $\sqrt{n_{eff}}$. We have already seen the graphs of $\sqrt{n_{eff}}$. The graphs of the sample standard deviations are as follows:

15 Variables

Standard Deviation

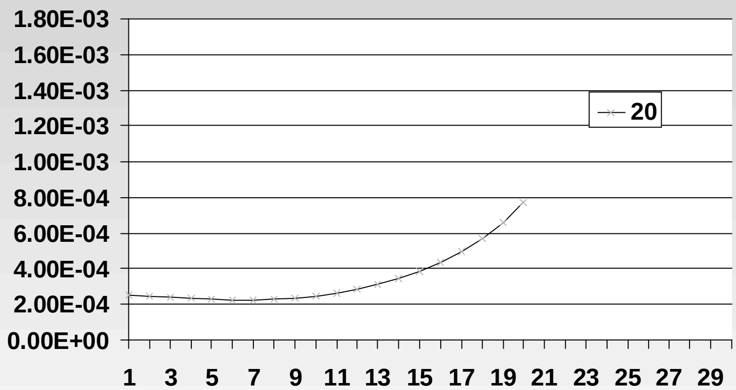
For 1,000,000 Iterations

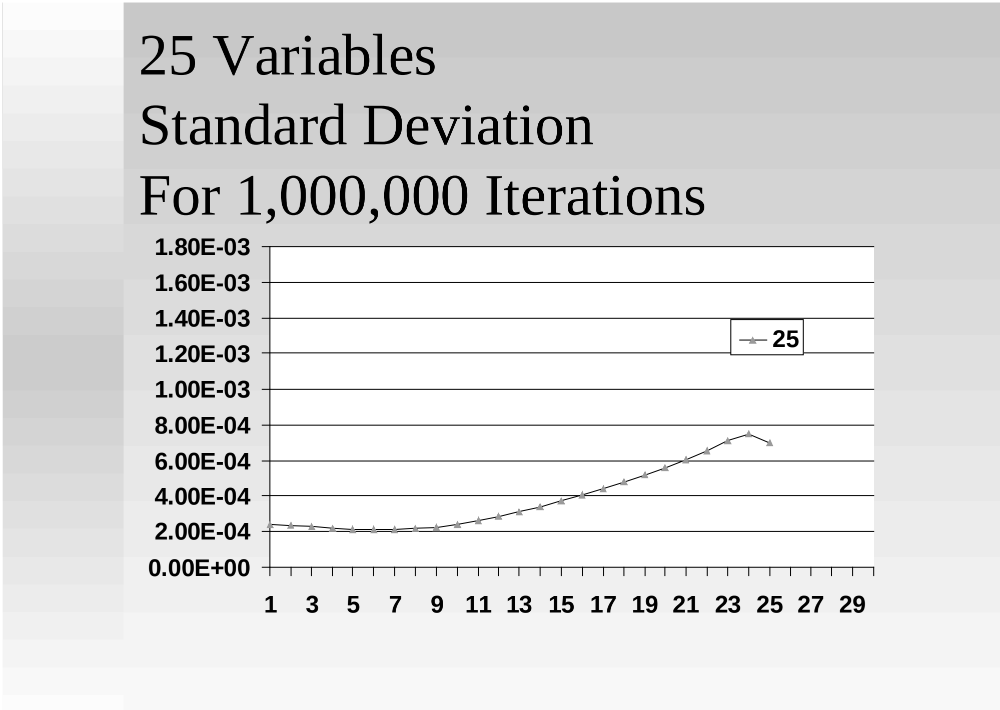


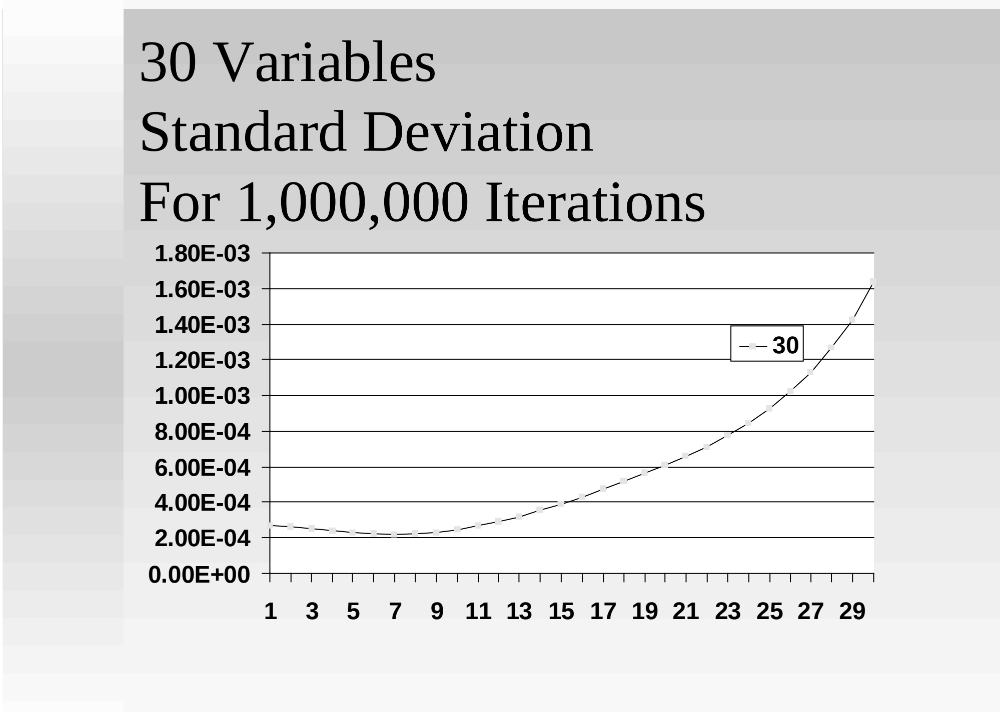
20 Variables

Standard Deviation

For 1,000,000 Iterations

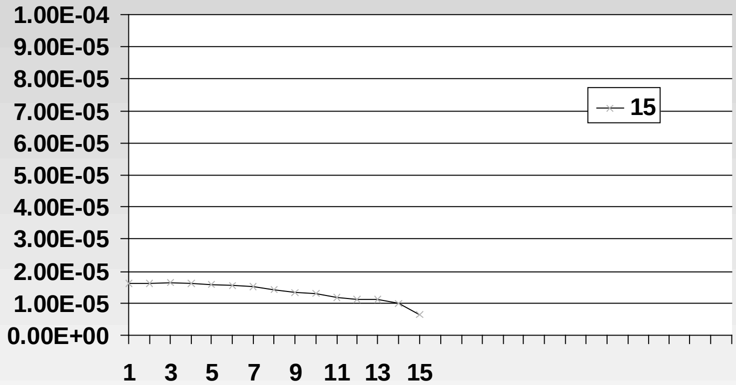




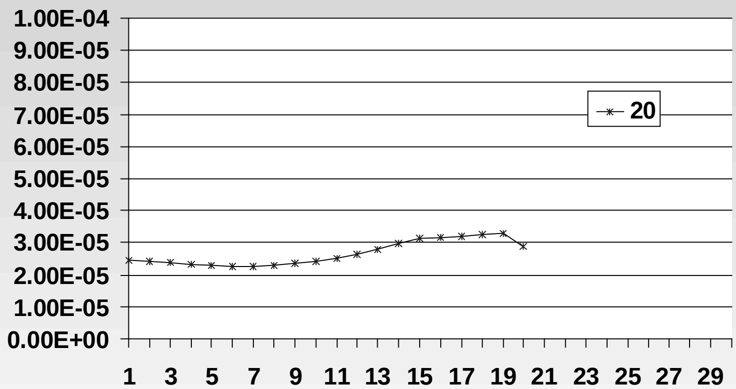


These graphs seem to be basically increasing convex. When age increases, the standard deviation of the simulated mortality rates increases as well. The graphs of the error bars are as follows:

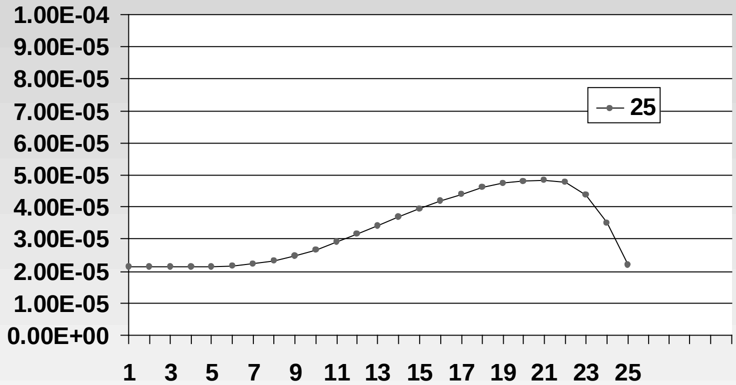
15 Variables
Error Bar
For 1,000,000 Iterations



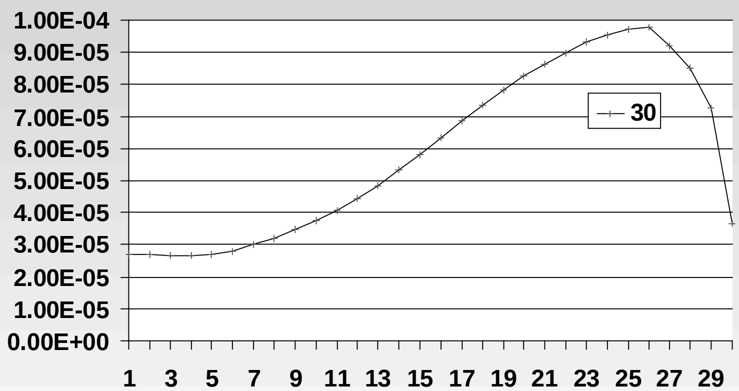
20 Variables
Error Bar
For 1,000,000 Iterations



25 Variables
Error Bar
For 1,000,000 Iterations



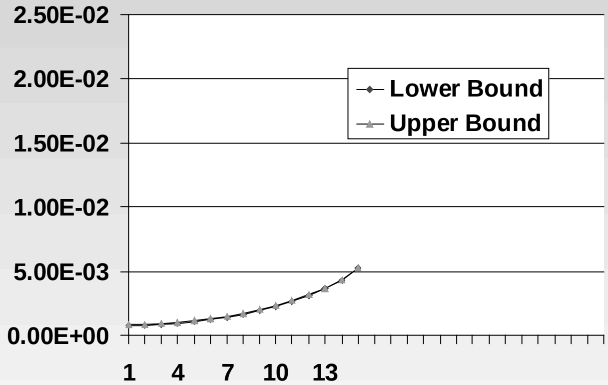
30 Variables
Error Bar
For 1,000,000 Iterations



The graph for 15 variables starts off flat and then gradually decreases. The graph for 20 variables starts off flat, gradually increases and then slightly decreases. The graphs for 25 and 30 variables start off flat and then increase and decrease, forming a parabola shape. On the graphs for 25 and 30 variables, the reason for the sharp decrease in the error bar for the last several variables is the sharp increase in $\sqrt{n_{eff}}$ for those variables. However, the reason why the error bar for the first variables is so small is not because n_{eff} is large but that the sample standard deviation is very small.

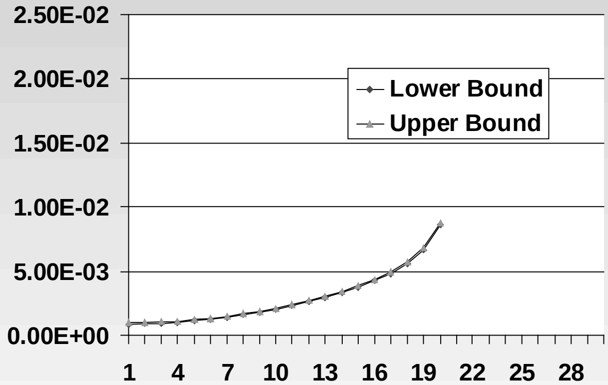
As stated earlier in this thesis, a 95% confidence interval for a time series $\{X_1, \ldots, X_n\}$ is $\overline{X} \pm$ (1.96)(error bar). I have graphed the confidence intervals for the cases where there are 15, 20, 25, and 30 variables. On each graph, the lower bounds for the variables are represented by an increasing convex curve, and the upper bounds for the variables are represented by another increasing convex curve. For each case, the curves overlap and are hard to distinguish from one another. This means that the confidence intervals are very narrow. This is a positive quality—the narrower, the better. Because the intervals are so narrow, I have included a table for each graph. The graphs and tables are as follows:

15 Variables
Confidence Intervals
For 1,000,000 Iterations



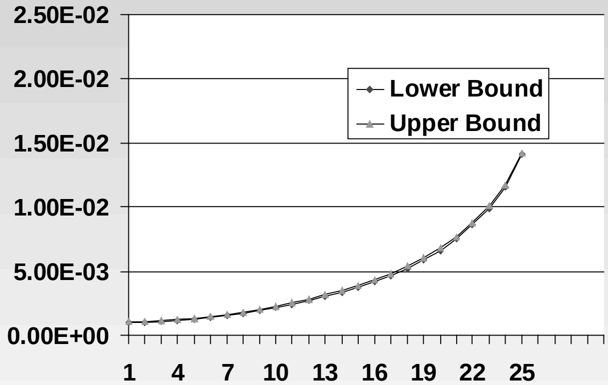
Subscript	Lower Bound	Upper Bound
1	7.819E-04	8.453E-04
2	8.086E-04	8.721E-04
3	8.640E-04	9.279E-04
4	9.508E-04	1.014E-03
5	1.071E-03	1.134E-03
6	1.228E-03	1.289E-03
7	1.422E-03	1.481E-03
8	1.657E-03	1.713E-03
9	1.935E-03	1.988E-03
10	2.262E-03	2.313E-03
11	2.650E-03	2.696E-03
12	3.105E-03	3.149E-03
13	3.639E-03	3.682E-03
14	4.302E-03	4.341E-03
15	5.290E-03	5.315E-03

20 Variables
Confidence Intervals
For 1,000,000 Iterations



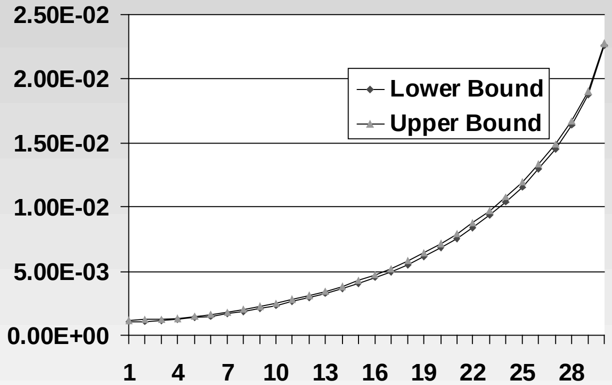
Subscript	Lower Bound	Upper Bound
1	8.830E-04	9.790E-04
2	9.071E-04	1.002E-03
3	9.564E-04	1.050E-03
4	1.031E-03	1.123E-03
5	1.133E-03	1.223E-03
6	1.261E-03	1.350E-03
7	1.415E-03	1.504E-03
8	1.596E-03	1.686E-03
9	1.804E-03	1.896E-03
10	2.039E-03	2.134E-03
11	2.305E-03	2.404E-03
12	2.602E-03	2.706E-03
13	2.934E-03	3.044E-03
14	3.306E-03	3.423E-03
15	3.734E-03	3.856E-03
16	4.234E-03	4.359E-03
17	4.840E-03	4.965E-03
18	5.601E-03	5.730E-03
19	6.674E-03	6.803E-03
20	8.636E-03	8.749E-03

25 Variables
Confidence Intervals
For 1,000,000 Iterations



Subscript	Lower Bound	Upper Bound
1	9.896E-04	1.073E-03
2	1.014E-03	1.098E-03
3	1.065E-03	1.149E-03
4	1.142E-03	1.227E-03
5	1.247E-03	1.331E-03
6	1.378E-03	1.463E-03
7	1.535E-03	1.623E-03
8	1.717E-03	1.808E-03
9	1.924E-03	2.021E-03
10	2.156E-03	2.261E-03
11	2.415E-03	2.529E-03
12	2.701E-03	2.825E-03
13	3.015E-03	3.149E-03
14	3.361E-03	3.505E-03
15	3.746E-03	3.901E-03
16	4.177E-03	4.341E-03
17	4.665E-03	4.838E-03
18	5.223E-03	5.404E-03
19	5.869E-03	6.055E-03
20	6.629E-03	6.818E-03
21	7.524E-03	7.713E-03
22	8.589E-03	8.776E-03
23	9.889E-03	1.006E-02
24	1.157E-02	1.171E-02
25	1.413E-02	1.421E-02

30 Variables
Confidence Intervals
For 1,000,000 Iterations



Subscript	Lower Bound	Upper Bound
1	1.083E-03	1.189E-03
2	1.111E-03	1.216E-03
3	1.166E-03	1.272E-03
4	1.252E-03	1.356E-03
5	1.366E-03	1.471E-03
6	1.508E-03	1.618E-03
7	1.677E-03	1.795E-03
8	1.873E-03	1.999E-03
9	2.095E-03	2.231E-03
10	2.344E-03	2.492E-03
11	2.622E-03	2.782E-03
12	2.927E-03	3.102E-03
13	3.261E-03	3.451E-03
14	3.625E-03	3.834E-03
15	4.026E-03	4.254E-03
16	4.468E-03	4.716E-03
17	4.958E-03	5.228E-03
18	5.506E-03	5.794E-03
19	6.117E-03	6.424E-03
20	6.802E-03	7.126E-03
21	7.565E-03	7.903E-03
22	8.415E-03	8.767E-03
23	9.360E-03	9.725E-03
24	1.042E-02	1.079E-02
25	1.160E-02	1.198E-02

26	1.295E-02	1.333E-02
27	1.451E-02	1.487E-02
28	1.637E-02	1.670E-02
29	1.875E-02	1.904E-02
30	2.257E-02	2.272E-02

We have seen what the confidence intervals are. As discussed earlier in this thesis, when we are using Bayesian statistics, as we are doing, we take the mean of the posterior distribution as a single point estimate of the variables we are interested in, in this case the vector θ. The mean of θ for each case is given in the following table:

Subscript	15 Vars	20 Vars	25 Vars	30 Vars
1	8.136E-04	9.310E-04	1.031E-03	1.136E-03
2	8.403E-04	9.545E-04	1.056E-03	1.163E-03
3	8.960E-04	1.003E-03	1.107E-03	1.219E-03
4	9.823E-04	1.077E-03	1.184E-03	1.304E-03
5	1.102E-03	1.178E-03	1.289E-03	1.419E-03
6	1.259E-03	1.306E-03	1.420E-03	1.563E-03
7	1.452E-03	1.460E-03	1.579E-03	1.736E-03
8	1.685E-03	1.641E-03	1.763E-03	1.936E-03
9	1.962E-03	1.850E-03	1.972E-03	2.163E-03
10	2.287E-03	2.086E-03	2.208E-03	2.418E-03
11	2.673E-03	2.354E-03	2.472E-03	2.702E-03
12	3.127E-03	2.654E-03	2.763E-03	3.015E-03
13	3.661E-03	2.989E-03	3.082E-03	3.356E-03
14	4.321E-03	3.365E-03	3.433E-03	3.729E-03
15	5.302E-03	3.795E-03	3.823E-03	4.140E-03
16		4.297E-03	4.259E-03	4.592E-03
17		4.903E-03	4.752E-03	5.093E-03
18		5.666E-03	5.313E-03	5.650E-03
19		6.738E-03	5.962E-03	6.271E-03
20		8.693E-03	6.723E-03	6.964E-03
21			7.619E-03	7.734E-03
22			8.682E-03	8.591E-03
23			9.975E-03	9.542E-03
24			1.164E-02	1.060E-02
25			1.417E-02	1.179E-02
26				1.314E-02
27				1.469E-02
28				1.654E-02
29				1.889E-02
30				2.265E-02

VII. BIBLIOGRAPHY:

[1] Besag, J. "Spatial Interaction and the Statistical Analysis of Lattice Systems", Journal of the Royal Statistical Society, Series B, 36 (1974): 192-236.

[2] Bowers, et al. Actuarial Mathematics. Itasca, IL: The Society of Actuaries, 1986.

[3] Broffitt, J. "Increasing and Increasing Convex Bayesian Graduation", Transactions, Vol. XL (1988): 115-48.

[4] Broverman, Samuel. ACTEX Study Manual For The Course 120 Examination Of The Society Of Actuaries. Winsted, CT: ACTEX Publications, Inc., 1992.

[5] Carlin, Bradley P. "A Simple Monte Carlo Approach To Bayesian Graduation", Transactions, Vol. XLIV. Itasca, IL: The Society Of Actuaries, 1992, pp. 55-76.

[6] Casella, G. & George, E. "Explaining the Gibbs Sampler", The American Statistician, August 1992, Vol. 46, No. 3, pp. 167-174.

[7] Crofts, Gauger, & London. ACTEX Study Manual For The Course 150 Examination Of The Society Of Actuaries. Vol. 1. Winsted, CT: ACTEX Publications, Inc., 1993.

[8] Geman, S. & Geman, D. "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 6 (1984), pp. 721-741.

[9] Goodman, J. & Sokal, A. "Multigrid Monte Carlo method. Conceptual foundations", Physical Review D, Vol. 40 (1989), No. 6, pp. 2035-2071.

[10] Grimmett, G. & Welsh, D. Probability: An Introduction. Oxford: Clarendon Press, 1986.

[11] Hogg, R. & Craig, A. Introduction to Mathematical Statistics: Fourth Edition. New York: Macmillan Publishing Co., Inc., 1978.

[12] Kalos, M. & Whitlock, P. Monte Carlo Methods: Volume 1: Basics. New York: John Wiley & Sons, Inc., 1986.

[13] London, Dick. Graduation: The Revision of Estimates. Winsted, CT: ACTEX Publications, 1985.

[14] Metropolis, Rosenbluth, Rosenbluth, Teller and Teller. "Equation of State Calculations by Fast Computing Machines", The Journal of Chemical Physics, Vol. 21 (1953), No. 6, pp. 1087-1092.

[15] Tadikamalla, Pandu R. "Computer Generation of Gamma Random Variables", Communications of the ACM, Vol. 21, pp. 419-422.