# UNIVERSIDADE DE SÃO PAULO

## NONHOMOGENEOUS POISSON PROCESSES ASSUMING A INVERSE GAUSSIAN ORDER STATISTICS MODEL FOR SOFTWARE RELIABILITY DATA: A BAYESYAN APPROACH

JORGE ALBERTO ACHCAR
KARIN STORANI

Nº 35

## NOTAS

*Instituto de Ciências Matemáticas de São Carlos*

**Instituto de Ciências Matemáticas de São Carlos**

ISSN - 0103-2577

# NONHOMOGENEOUS POISSON PROCESSES ASSUMING A INVERSE GAUSSIAN ORDER STATISTICS MODEL FOR SOFTWARE RELIABILITY DATA: A BAYESYAN APPROACH

**JORGE ALBERTO ACHCAR**
**KARIN STORANI**

**Nº 35**

NOTAS DO ICMSC
Série Estatística

São Carlos
Dez./1996

# Resumo

Métodos Bayesianos usando processos de Poisson não homogêneo são considerados para modelagem de problemas de confiabilidade de software. Um modelo de estatística de ordem Gaussiana Inversa é considerado para modelar as épocas das falhas do software. Algoritmos computacionais como Metropolis Hastings e amostrador de Gibbs são utilizados para obtenção de sumários a posteriori de interesse. Um método de diagnóstico do modelo utilizando o amostrador de Gibbs é proposto. A metodologia desenvolvida neste artigo é exemplificada com um conjunto de dados introduzido por Jelinski e Moranda (1972).

# Nonhomogeneous Poisson Processes Assuming a Inverse Gaussian Order Statistics Model for Software Reliability Data: a Bayesian Approach

Jorge Alberto Achcar
Karin Storani
Department of Computer Science and Statistics
ICMSC, University of São Paulo
C.P. 668, 13560-970
São Carlos, S.P., Brazil

## Abstract

"We consider a Bayesian approach using nonhomogeneous Poisson processes for software reliability and a inverse Gaussian order statistics model to model epochs of the failures of the software. Metropolis algorithms along with Gibbs steps are used to obtain the posterior summaries of interest. A model diagnostics method using the Gibbs samples is proposed to check the modeling assumption. A numerical illustration is considered using a software reliability data set introduced by Jelinski and Moranda (1972)."

*Key Words*: Inverse Gaussian intensity function, Nonhomogeneous Poisson process, Gibbs Sampling, Metropolis algorithm.

# 1 Introduction

Consider the modelling of the number of failures of a software by a point process to count failures (see for example, Musa, Iannino and Okumoto, 1987). Let M(t) be the cumulative number of failures of the software that are observed during time (0,t] and assume that M(t) is modeled by a nonhomogeneous Poisson process with mean value function,

$$m(t)=\theta F(t), \tag{1}$$

where $\theta$ is the total number of bugs and F(t) is the distribution function of a inverse Gaussian distribution (see for example, Chhikara and Folks, 1989), given by

$$F(t)=\Phi\left[\sqrt{\frac{\lambda}{t}}\left(\frac{t}{\mu}-1\right)\right]+e^{2\lambda/\mu}\Phi\left[-\sqrt{\frac{\lambda}{t}}\left(1+\frac{t}{\mu}\right)\right] \tag{2}$$

where $\Phi(x)$ denotes the distribution function of a standard normal $N(0,1)$ distribution.

From (2), the intensity function $\lambda(t)$ which is the derivative of m(t) is given by

$$\lambda(t)=\frac{\theta\sqrt{\lambda}}{\sqrt{2\pi}}t^{-3/2}\exp\left\{-\frac{\lambda(t-\mu)^2}{2\mu^2 t}\right\} \tag{3}$$

Other choices for F(t) in (1) are considered in the literature: for example, if $F(t)=1-e^{-\beta t}$, we have the Goel and Okumoto (1979) process (an exponential order statistics model); if $F(t)=1-(1+\beta t)e^{-\beta t}$, we have the Ohba - Yamada (1982, 1983) process.

Considering some special choices of F(t) in (1) (exponential, Weibull, Pareto and extreme value order statistics models) and using Gibbs sampling with Metropolis-Hastings algorithms, Kuo and Yang (1996) developed a Bayesian analysis for nonhomogeneous Poisson processes.

Kuo, Lee, Choi and Yang (1996) present Bayesian inference for the Ohba - Yamada (1982,1983) process and Achcar, Dey and Niverthi (1996) present a Bayesian analysis for nonhomogeneous Poisson processes assuming a generalized gamma order statistics model.

In this paper, we present Bayesian inference for software reliability models assuming a nonhomogeneous Poisson process with a inverse Gaussian order statistics model given by (1) and (2) and using Metropolis-within-Gibbs algorithms (see for example, Chib and Greenberg, 1995; or Smith and Roberts, 1993). The use of this class of model gives great flexibility for the shape of the intensity function $\lambda(t)$, which implies in better fit for software reliability data.

## 2 Bayesian Inference for The Inverse Gaussian Order Statistics Model

Given the time truncated model testing until time t, the ordered epochs of the observed n failure times are denoted by $x_1, x_2, \ldots, x_n$.

Considering a nonhomogeneous Poisson process with mean value function (1) and a inverse Gaussian order statistics model (2), the likelihood function for $\theta$, $\mu$ and $\lambda$ (see for example, Cox and Lewis, 1966) is given by

$$L(\theta,\mu,\lambda)=\left\{\prod_{i=1}^{n}\lambda(x_i)\right\}\exp\{-m(t)\} \tag{4}$$

That is,

$$L(\theta,\mu,\lambda)=\frac{\theta^{n}\lambda^{n/2}}{(2\pi)^{n/2}}\left\{\prod_{i=1}^{n}x_i^{-3/2}\right\}\exp\left\{-\theta F(x_n)-\frac{\lambda}{2\mu^2}\sum_{i=1}^{n}\frac{(x_i-\mu)^2}{x_i}\right\} \tag{5}$$

where F(t) is given in (2).

For a Bayesian analysis of this model, we consider the use of Metropolis-within-Gibbs algorithms. Considering the introduction of a latent variable N'=N-n which has a Poisson distribution with parameter $\theta\,[1\text{-}F(t)]$ (see Yang, 1994; or Kuo and Yang, 1996), we assume the following prior densities for N', $\theta$, $\mu$ and $\lambda$:

(i) $N' \sim P[\theta\,[1\text{-}F(t)]]$

(ii) $\mu \sim \Gamma(a_1,b_1)$; $a_1, b_1$ known,

(iii) $\lambda \sim \Gamma(a_2,b_2)$; $a_2, b_2$ known, (6)

(iv) $\theta \sim \Gamma(a_3,b_3)$; $a_3, b_3$ known,

where $P(\lambda)$ denotes a Poisson distribution with parameter $\lambda$, $\Gamma(a,b)$ denotes a gamma distribution with mean $a/b$ and variance $a/b^2$. We further assume independence among the parameters $\theta$, $\mu$ and $\lambda$.

The joint posterior density is given by

$$\pi(N',\mu,\lambda,\theta/D)\propto\theta^{N'+n+a_3-1}\lambda^{n/2+a_2-1}\mu^{a_1-1}\frac{\{1-F(x_n)\}^{N'}}{N'!}$$
$$\exp\left\{-\frac{\lambda}{2\mu^2}\sum_{i=1}^{n}\frac{(x_i-\mu)^2}{x_i}-b_1\mu-b_2\lambda-\theta(b_3+1)\right\} \tag{7}$$

where $D=\{n, x_1,\ldots,x_n, t\}$ is the data set and F(t) is given in (2).

For the failure truncated model, similar expressions to (5) and (7) can be applied with t replaced by $x_n$.

The marginal posterior densities for the Gibbs algorithm are given by,

(i) $N'/\mu,\lambda,\theta,D \sim P[\theta(1-F(x_n))]$,

(ii) $\theta/\mu,\lambda,N',D \sim \Gamma[N'+n+a_3, b_3+1]$,

(iii) $\pi(\mu/\lambda,\theta,N',D) \propto \mu^{a_1-1}e^{-b_1\mu}\Psi_1(\mu,\lambda)$,

where $\Psi_1(\mu,\lambda) = \{1-F(x_n)\}^{N'} \exp\left\{-\frac{\lambda}{2\mu^2}\sum_{i=1}^{n}\frac{(x_i-\mu)^2}{x_i}\right\}$, (8)

and

(iv) $\pi(\lambda/\theta,\mu,N',D) \propto \lambda^{a_2-1}e^{-b_2\lambda}\Psi_2(\mu,\lambda)$

where $\Psi_2(\mu,\lambda) = \lambda^{n/2}\{1-F(x_n)\}^{N'} \exp\left\{-\frac{\lambda}{2\mu^2}\sum_{i=1}^{n}\frac{(x_i-\mu)^2}{x_i}\right\}$

The variables $\mu$ and $\lambda$ should be generated using Metropolis-Hastings algorithm.

## 3 Bayesian Inference and Model Checking

We can use the Gibbs samples to get inferences on the parameters of the software reliability model or functions of these parameters. In this case, we could approximate posterior moments of interest. As a special case, consider the mean value function m(t). Considering the inverse gaussian order statistics model, we have m(t)= $\theta$ F(t) with F(t) given by (2) and a Bayes estimator of m(t) with respect to the squared error loss function is given by E[m(t)/D].

Let $\theta^{(r,s)}, \lambda^{(r,s)}$ and $\mu^{(r,s)}$ denote the variates for $\theta$, $\lambda$ and $\mu$ drawn in the $r^{th}$ iteration and the $s^{th}$ replication where R and S are respectively the total number of iterations and simulations of the Gibbs sampler. Then E(m(t)/D) can be estimated by

$$\hat{m}(t) = \frac{2}{RS}\sum_{s=1}^{S}\sum_{r=\frac{R}{2}+1}^{R}\theta^{(r,s)}g(\mu^{(r,s)},\lambda^{(r,s)})$$

$$\text{where } g(\mu^{(r,s)},\lambda^{(r,s)}) = \Phi\left[\sqrt{\frac{\lambda^{(r,s)}}{t}}\left(\frac{t}{\mu^{(r,s)}}-1\right)\right] + e^{2\lambda^{(r,s)}/\mu^{(r,s)}}\Phi\left[-\sqrt{\frac{\lambda^{(r,s)}}{t}}\left(1+\frac{t}{\mu^{(r,s)}}\right)\right] \quad (9)$$

For model checking, observe that, if the model (1) is correct, m(t)/$\theta$ =F(t) has a standard uniform distribution.

Therefore, at each failure time, we could consider empirical Q-Q plots of the Monte Carlo estimates of m(t)/$\theta$ versus a Uniform (0,1) distribution. Departure from a uniform distribution indicates model inadequacy.

# 4 An Example

In table 1, we have a software reliability data set introduced by Jelinski and Moranda (1972). The data consists of the number of days between the 26 failures that ocurred during the production phase of a software ( NTDS data Naval Tactical Data System).

From the data of table 1, we have n=26 and $x_n = x_{26} = 250$.

| i | $t_i$ | $x_i$ | i | $t_i$ | $x_i$ | i | $t_i$ | $x_i$ |
|---|---|---|---|---|---|---|---|---|
| 1 | 9 | 9 | 11 | 1 | 71 | 21 | 11 | 116 |
| 2 | 12 | 21 | 12 | 6 | 77 | 22 | 33 | 149 |
| 3 | 11 | 32 | 13 | 1 | 78 | 23 | 7 | 156 |
| 4 | 4 | 36 | 14 | 9 | 87 | 24 | 91 | 247 |
| 5 | 7 | 43 | 15 | 4 | 91 | 25 | 2 | 249 |
| 6 | 2 | 45 | 16 | 1 | 92 | 26 | 1 | 250 |
| 7 | 5 | 50 | 17 | 3 | 95 | | | |
| 8 | 8 | 58 | 18 | 3 | 98 | | | |
| 9 | 5 | 63 | 19 | 6 | 104 | | | |
| 10 | 7 | 70 | 20 | 1 | 105 | | | |

Table 1. NTDS data ($t_i = x_i - x_{i-1}$ is the interfailure time)

Assuming the inverse gaussian order statistics model (2) and the failure truncated model with t replaced by $x_{26}$=250, we consider (from (6)), the priors $N' \sim P[\theta(1-F(t))]$, $\mu \sim \Gamma[756;2.75]$, $\lambda \sim \Gamma[100;1]$ and $\theta \sim \Gamma[81;2.25]$. From the marginal posterior densities for $N', \mu, \lambda$ and $\theta$ given in (8), we generated 10 separate Gibbs chains each of which ran for 1200 iterations, and we monitored the convergence of the Gibbs samples using the Gelman and Rubin (1992) method that uses the analysis of variance technique to determine if further iterations are needed.

For each parameter, we considered the $210^{th}$, $220^{th}$,..., $1200^{th}$ iteration, which for 10 chains yields a sample of size 1000.

In table 2, we have the obtained posterior summaries for the parameters $N', \mu, \lambda$ and $\theta$ and in figure 1, we have the aproximate marginal densities considering S=1000 Gibbs samples.

| | Mean | Median | Variance | 95%Credible Interval |
|---|---|---|---|---|
| $N'$ | 10.4270 | 10 | 11.3821 | [5.0, 18] |
| $\theta$ | 36.2004 | 36.1753 | 11.6491 | [29.8630, 43.4157] |
| $\mu$ | 274.6688 | 274.6903 | 87.7225 | [256.0129, 293.3224] |
| $\lambda$ | 100.1174 | 99.9262 | 84.8426 | [82.9719, 119.4247] |

Table 2. Posterior summaries for the inverse Gaussian order statistics model

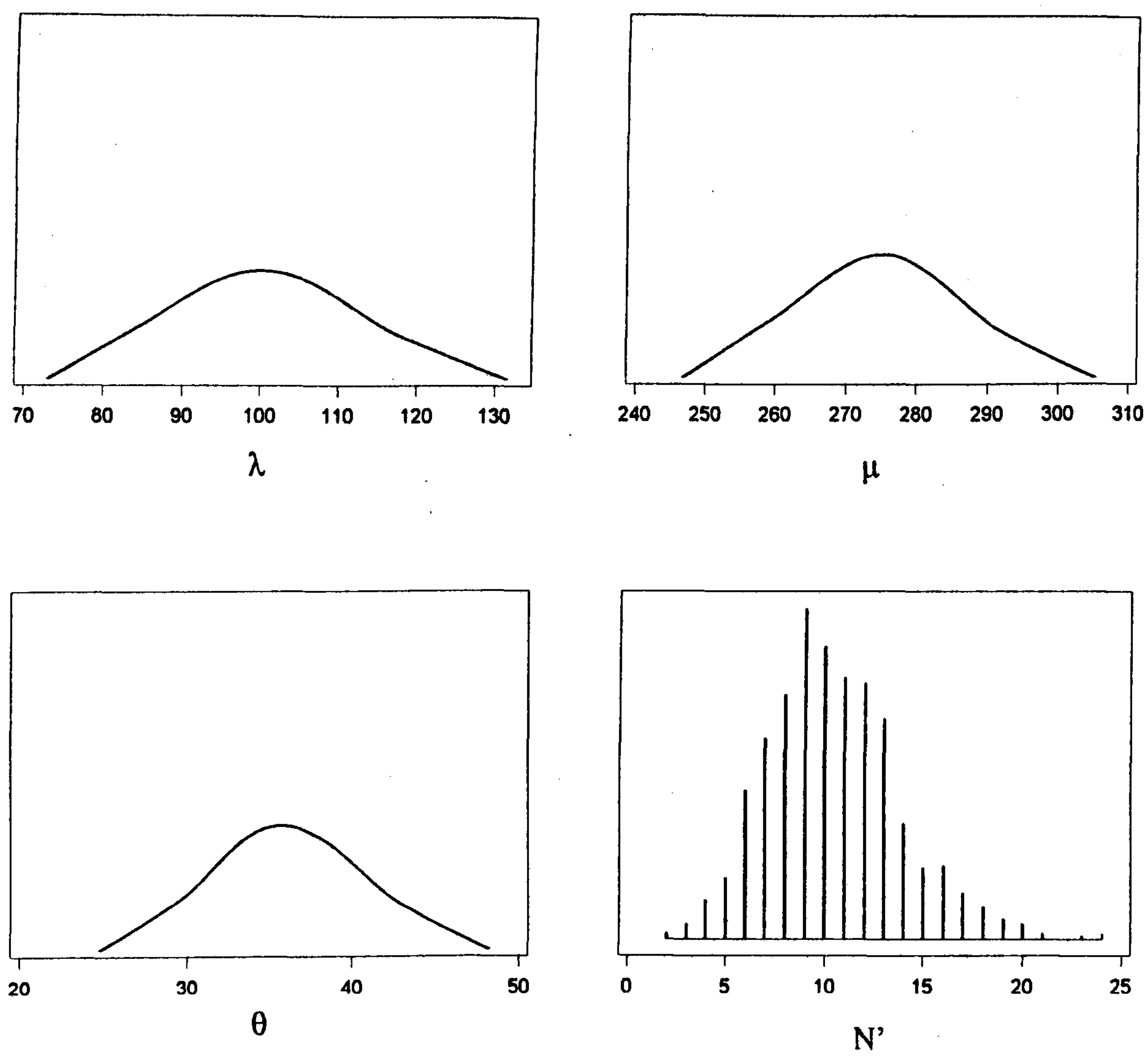


Figure 1. Marginal posterior densities for N', μ, λ and θ.

In table 3, we have approximated Bayes estimates for the mean value function m(t) with respect to the squared error loss function using the Gibbs samples.

| i | $x_i$ | $n_i$ | $m(x_i)$ | i | $x_i$ | $n_i$ | $m(x_i)$ |
|---|---|---|---|---|---|---|---|
| 1 | 9 | 1 | 0.0442 | 1 | 87 | 14 | 14.3118 |
| 2 | 21 | 2 | 1.4955 | 2 | 91 | 15 | 14.9442 |
| 3 | 32 | 3 | 3.9496 | 3 | 92 | 16 | 14.9737 |
| 4 | 36 | 4 | 4.8902 | 4 | 95 | 17 | 15.3535 |
| 5 | 43 | 5 | 6.4974 | 5 | 98 | 18 | 15.7212 |
| 6 | 45 | 6 | 6.9412 | 6 | 104 | 19 | 16.4220 |
| 7 | 50 | 7 | 8.0144 | 7 | 105 | 20 | 16.5346 |
| 8 | 58 | 8 | 9.6159 | 8 | 116 | 21 | 17.6994 |
| 9 | 63 | 9 | 10.5444 | 9 | 149 | 22 | 20.5412 |
| 10 | 70 | 10 | 11.7552 | 10 | 156 | 23 | 21.0447 |
| 11 | 71 | 11 | 11.9201 | 11 | 247 | 24 | 25.6768 |
| 12 | 77 | 12 | 12.8696 | 12 | 249 | 25 | 25.7507 |
| 13 | 78 | 13 | 13.0214 | 13 | 250 | 26 | 25.7873 |

Table 3. Bayes estimators for $m(x_i)$, i=1,2,3,...,26

In figure 2, we have Q-Q plot of the Monte Carlo estimates of $m(t)/\theta$ versus a Uniform (0,1) distribution. We observe good fit of the inverse gaussian order statistics model (2) to the NTDS data of table 1.

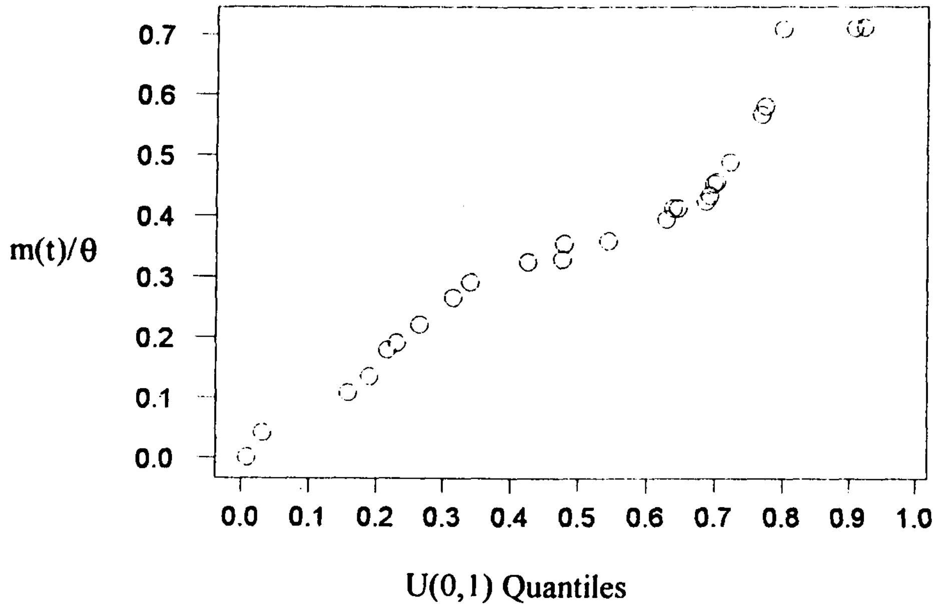


Figure 2. Empirical Q-Q plot for $m(t)/\theta$ versus Uniform(0,1)

# APPENDIX

## The Gibbs Sampling and Metropolis-Hastings Algorithms

The Gibbs sampler is a Markov chain Monte Carlo (MCMC) technique for generating random variables from a distribution without calculating the density itself (see for example, Gelfand and Smith, 1990).

Given a collection of *k* random variables $\underset{\sim}{\theta} = (\theta_1, \theta_2, \ldots, \theta_k)$, we want to generate a random sample from their joint distribution $p(\theta_1, \theta_2, \ldots, \theta_k)$ or $p(\underset{\sim}{\theta})$. By suppressing the dependence on the data, we need the complete conditional distributions $p(\theta_s \backslash \underset{\sim}{\theta}_{(s)})$, s=1,2,...,k where $\underset{\sim}{\theta}_{(s)}$ denotes the random vector of *k*-1 random variables with the $s^{th}$ random variable being deleted. Given the initial values of $\theta_1^{(0)}, \theta_2^{0}, \ldots, \theta_3^{0}$, the Gibbs sampling algorithm proceeds as follows: generate a value $\theta_1^{(1)}$ from the conditional density $p(\theta_1 \backslash \theta_2^{(0)}, \theta_3^{(0)}, \ldots, \theta_k^{(0)})$. Similarly, generate a value $\theta_2^{(1)}$ from the conditional density $p(\theta_2 \backslash \theta_1^{(1)}, \theta_2^{(2)}, \ldots, \theta_k^{(0)})$ and continue up to the have $\theta_k^{(1)}$ from the conditional density $p(\theta_k \backslash \theta_1^{(1)}, \theta_2^{(1)}, \ldots, \theta_{k-1}^{(1)})$.

Then, replace the initial values with the new realization $\underset{\sim}{\theta}^{(1)}$ of $\underset{\sim}{\theta}$, and iterate the above process *t* times, producing $\underset{\sim}{\theta}^{(t)}$. Geman and Geman (1984) showed that the *k*-triple produced at the $t^{th}$ iteration of the sampling scheme, $(\theta_1^{(t)}, \theta_2^{(t)}, \ldots, \theta_k^{(t)})$, converges in distribution to a random variate from $p(\theta_1, \theta_2, \ldots, \theta_k)$ if *t* is sufficiently large. Further, $\theta_i^{(k)}$ can be regarded as a simulated observation from $p(\theta_i)$, the marginal distribution of $\theta_i$.

Replicating the above process *B* times, we obtain *B* many *k*-tuples $\left\{\theta_{1g}^{(t)}, \theta_{2g}^{(t)}, \ldots, \theta_{kg}^{(t)}, g = 1, 2, \ldots, B\right\}$. Upon convergence of the Gibbs sampler, any characteristic of the marginal density $p(\theta_i)$ can be obtained. In particular, if $p(\theta_s \backslash \underset{\sim}{\theta}_{(s)})$ is available in closed form, then

$$\hat{p}(\theta_s) = \frac{1}{B} \sum_{g=1}^{B} p(\theta_s \backslash \underset{\sim}{\theta}_{(s)g}),$$

$s = 1, 2, \ldots, k$.

The Gibbs sampler involves drawing random samples form all full conditional densities of $p(\underset{\sim}{\theta})$. When the conditional densities are not easily identified, such as in cases without conjugate priors, we can employ the Metropolis-Hastings algorithm.

Suppose we desire to sample a variate from a nonregular density $p(\theta_i \backslash \underset{\sim}{\theta}_{(i)})$. Observe that $p(\theta_i \backslash \underset{\sim}{\theta}_{(i)})$ only needs to be known up to a constant and we denote $p(\theta)$ as the target density, suppressing the subscript, and the conditional variables for brevity. Let us define a transition Kernel q($\theta$ ,$X$) which maps $\theta$ to $X$. If $\theta$ is a real variable wich ranges in $(-\infty, \infty)$, we can construct q such that $X \leftarrow \theta + \sigma Z$, with Z being the standard normal random variable and $\sigma^2$ reflecting the conditional variance of $\theta$ in $p(\theta)$. If $\theta$ is bounded with range (a,b), we can use a transformation to map (a,b) to $(-\infty, \infty)$, then use the transition kernel $q$ and apply the Metropolis algorithm to the density of the transformed variable. Then the Metropolis algorithm proceeds as follows:

(i) start with any point $\theta^{(0)}$, and stage indicator $j$=0;

(ii) generate a point $X$ according to the transition kernel q($\theta^{(j)}$,$X$);

(iii) update $\theta^{(j)}$ to $\theta^{(j+1)}=X$ with probability p=min{1,p($X$)/p($\theta^{(j)}$)}; stay at $\theta^{(j)}$ with probability *1-p*;

(iv) repeat (ii) and (iii) by increasing the stage indicator untilt he process reaches a stationary distribution.

# References

[1] Achcar, J.A; Dey, D.K.; Niverthi, M. (1996). "A Bayesian approach using nonhomogeneos Poisson process for software reliability models", technical report 96-07, department of statistics, University of Connecticut, Storrs, U.S.A.

[2] Chhikara, R.S.; Folks J.L. (1989) The inverse Gaussian distribution, New York: Marcel Dekker, Inc.

[3] Chib, S.; Greenberg, E.(1995). "Understanding the Metropolis-Hastings algorithm", American Statistician, 49, 4, 327-335.

[4] Cox, D.R.; Lewis, P.A. (1996) Statistical Analysis of Series of events. London: Methuen.

[5] Gelfand, A.E.; Smith, A.F.M. (1990)."Sampling-based approaches to calculating marginal densities", Journal of the American Statistical Association, 85, 398-409.

[6] Gelman, A.E.; Rubin, D.(1992). "Inference from iterative simulation using multiple sequences", Statistical Science, 7, 457-472.

[7] Geman, S.; Geman, D. (1984). "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images", IEEE Transactions on Pattern Analysis and Machine Intelligence, 6, 721-741.

[8] Goel, A.L. (1983) "A guidebook for software reliability assessment", technical report RDAC-TR-83-176.

[9]Goel, A.L; Okumoto, K. (1979) "Time dependent error detection rate model for software reliability and other performance measures",IEEE Transactions on Reliability, 28,206-211..

[10] Jelinski, Z.; Moranda, P.B. (1972). "Software reliability research", in Statistical Computer Performance Evaluation, ed. W. Freiberger, New York: Academic Press, pp.465-497.

[11] Kuo, L.; Yang, T.Y. (1996). "Bayesian computation for nonhomogeneous Poisson process in software reliability", Journal of the American Statistical Association, 91, 763-773.

[12] Kuo, L.; Lee, J.; Choi, K.; Yang, T.Y. (1996). "Bayesian inference for S-shaped software reliability growth models", technical report 96-05, department of statistics, University of Connecticut, Storrs, U.S.A.

[13] Musa, J.D.; Iannino, A.; Okumoto, K. (1987). Software Reliability: Measurement, Prediction, Application. New York: Wiley.

[14] Ohba, M.; Yamada, S.; Takeda, K.; Osaki, S. (1982). "S-shaped software reliability growth curve: how good is it ? COMPSAC'82, 38-44.

[15] Yamada, S.; Ohba, M.; Osaki, S. (1983). "S-Shaped reliability growth modeling for software error detection", IEEE Transaction on Reliability, 32, 475-478.

[16] Smith, A.F.M.; Roberts, G.O. (1993). "Bayesian Computations via the Gibbs sampler and related Markov Chain Monte Carlo methods (with discussion)", Journal of the Royal Statistical Society, B, 55,3-23.

[17] Yang, T.Y. (1994). "Computational approaches to Bayesian inference for software reliability", PhD thesis, department of statistics, University of Connecticut, Storrs, U.S.A.