UNIVERSIDADE DE SÃO PAULO

BAYESIAN INFERENCE FOR SOFTWARE RELIABILITY MODELS USING HOMOGENEOUS POISSON PROCESS

JORGE ALBERTO ACHCAR

Nº 29

NOTAS

Instituto de Ciências Matemáticas de São Carlos

Instituto de Ciências Matemáticas de São Carlos

ISSN - 0103-2577

BAYESIAN INFERENCE FOR SOFTWARE RELIABILITY MODELS USING HOMOGENEOUS POISSON PROCESS

JORGE ALBERTO ACHCAR

Nº 29

NOTAS DO ICMSC
Série Estatística

São Carlos
Abr. /1996

INFERENCIA BAYESIANA PARA MODELOS DE CONFIABILIDADE DE SOFTWARE USANDO PROCESSOS DE POISSON HOMOGENEOS

RESUMO

Inferencia Bayesiana usando processos de Poisson homogeneos e considerada para modelar dados de confiabilidade de software.Uma forma generalizada para o modelo de Moranda (1975) e considerada para modelar o numero de falhas que ocorrem em periodos de tempo fixos.Algoritmos de Metropolis-Hastings com etapas de Gibbs sao propostos para desenvolver a inferencia Bayesiana para o modelo de Moranda e para sua forma generalizada.Tambem exploramos alguns criterios de selecao considerando um conjunto de daodos de confiabilidade de software introduzido por Goel (1985).

Palavras-chaves:Modelo de Moranda,Gibbs sampling,algoritmo de Metropolis, selecao de modelos.

Bayesian Inference for Software Reliability Models Using Homogeneous Poisson Process

Jorge Alberto Achcar
University of São Paulo
ICMSC-USP, C. Postal 668
13560-970, São Carlos, S.P., Brazil

Abstract

Bayesian inference using homogeneous Poisson process is considered for modeling software reliability data. A generalized form for the Moranda (1975) model is considered to model the number of failures that occur in fixed time periods. Metropolis-Hastings algorithms along with Gibbs steps are proposed to perform the Bayesian inference for the Moranda's model and for its generalized form. We also explore some Bayesian model selection criteria considering a software reliability data set introduced by Goel (1985).

Key words: Moranda's model, Gibbs sampling, Metropolis algorithm, model selection.

1 Introduction

The number of failures of a software could be modeled by a point process to count failures (see for example, Singpurwalla and Wilson, 1994; or Musa, Iannino and Okumoto, 1987). In this way, the cumulative number of failures $M(t)$ of the software that are observed during time $(0, t]$ could be modeled by a Poisson process with mean value function $m(t)$ and distribution,

$$P(M(t) = n) = \frac{(m(t))^n e^{-m(t)}}{n!} \tag{1}$$

where n =0, 1, 2 ...

Different functions for $m(t)$ (or equivalently for the intensity function $\lambda(t) = dm(t)/\,dt$), specify different models.

If λ is constant (so that $m(t)$ is linear), then $M(t)$ is called a homogeneous Poisson process; otherwise it is called a nonhomogeneous Poisson process.

The literature presents many different models for $m(t)$ (see for example, Goel and Okumoto, 1979; Goel, 1983; Ohba and Yamada, 1982; Musa and Okumoto, 1984).

When the data on software failures are given in terms of the number of failures that occur in fixed time periods, Moranda (1975) assumed that the intensity function should be constant in a particular period, but form a decreasing geometric sequence over them. Thus, the number of failures in the i^{th} time period is a homogeneous Poisson process with intensity function,

$$\lambda_i \;=\; \lambda_a \; k_1^i \tag{2}$$

where $0 < k_1 < 1, \quad \lambda_a > 0$ and $i = 1, 2, \ldots$

By scaling, so that the time periods are of length 1, the distribution of the number of failures m_i in the i^{th} time period is given by

$$P\left\{M_i = m_i\right\} = \frac{e^{-\lambda_i} \lambda_i^{m_i}}{m_i!} \tag{3}$$

where $m_i = 0, 1, 2, \ldots$

A generalized form for the Moranda's model could be given by the introduction of a additional parameter k_2 in the intensity function (1), that is,

$$\lambda_i = \lambda_a \; k_1^{i^{k_2}} \tag{4}$$

Observe that when $k_2 = 1$, we have the Moranda's model (2).

In this paper, we present Bayesian inference for software reliability models using Metropolis-within-Gibbs algorithms for the Moranda's model with intensity function (2) and for its generalized form with intensity function (4).

We also explore some Bayesian model selection criteria considering a software reliability data set introduced by Goel (1985).

2 Bayesian Inference for the Moranda Model

Considering $m_1, m_2, \ldots, m_n$ the observed number of failures during the first n time periods, and the Moranda (1975) model with intensity function (2), the likelihood function for λ_a and k_1 is given by

$$L(\lambda_a, k_1) \alpha \; \lambda_a^{d_1} \; k_1^{d_2} e^{-\lambda_a \sum_{i=1}^{n} k_1^i} \;, \tag{5}$$

where $d_1 = \sum_{i=1}^{n} m_i$ and $d_2 = \sum_{i=1}^{n} i \, m_i$.

Assuming prior independence, consider the following prior densities for λ_a and k_1:

$$\begin{array}{ll} (i) & \lambda_a \sim \Gamma(b_1, b_2) \\ (ii) & k_1 \sim B(c_1, c_2) \end{array} \tag{6}$$

where $\Gamma(b_1, b_2)$ denotes a gamma distribution $\pi(\lambda_a) \alpha \lambda_a^{b_1 - 1} e^{-b_2 \lambda_a}, \lambda_a > 0$ and $B(c_1, c_2)$ denotes a Beta distribution $\pi(k_1) \alpha k_1^{c_1 - 1}(1 - k_1)^{c_2 - 1}$, $0 < k_1 < 1$. We assume b_1, b_2, c_1 and c_2 Known constants.

The joint posterior density for λ_a and k_1 is given by

$$\pi(\lambda_a, k_1 \mid \mathcal{D}_n) \alpha \lambda_a^{d_1 + b_1 - 1} \; e^{-[b_2 + \sum_{i=1}^{n} k_1^i] \lambda_a} k_1^{d_2 + c_1 - 1} (1 - k_1)^{c_2 - 1}, \tag{7}$$

where $\mathcal{D}_n$ denotes the data set $\{m_1, m_2, \ldots, m_n\}$.

Using Metropolis-within-Gibbs algorithms (see for example, Gelfand and Smith, 1990), we get the joint posterior density (7) approximated from the Gibbs samplers drawn from the following conditional densities:

$$(i) \quad \lambda_a \mid k_1, \mathcal{D}_n \sim \Gamma\left[d_1 + b_1, b_2 + \sum_{i=1}^{n} k_1^i\right]$$

$$(ii) \quad \pi(k_1 \mid \lambda_a, \mathcal{D}_n) \alpha \; k_1^{d_2+c_1-1}(1-k_1)^{c_2-1} exp\left\{-\lambda_a \sum_{i=1}^{n} k_1^i\right\}. \tag{8}$$

A sample of draws from the joint posterior density (7) can now be obtained by successively sampling λ_a from $\pi(\lambda_a \mid k_1, \mathcal{D}_n)$, and given this value of λ_a, simulating k_1 from $\pi(k_1 \mid \lambda_a, \mathcal{D}_n)$. The variable k_1 should be generated using Metropolis-Hastings algorithm (see for example, Chib and Greenberg, 1994).

In this way, observe that the conditional density for k_1 given λ_a and $\mathcal{D}_n$ could be written by,

$$\pi(k_1 \mid \lambda_a, \mathcal{D}_n) \; \alpha k_1^{c_1-1}(1-k_1)^{c_2-1}\psi_1(\lambda_a, k_1), \tag{9}$$

where $\psi_1(\lambda_a, k_1) = exp\left\{d_2 \ell n k_1 - \lambda_a \sum_{i=1}^{n} k_1^i\right\}$.

Thus, the value of k_1 is simulated as: at the s^{th} iteration (given the current value $\lambda_a^{(s)}$, draw a candidate $k_1^{(s)}$ from a Beta density $B(c_1, c_2)$; if is satisfies stationarity, move to this point with probability

$$min\left\{\frac{\psi_1\left(\lambda_a^{(s)}, k_1^{(s)}\right)}{\psi_1\left(\lambda_a^{(s)}, k_1^{(s-1)}\right)}, 1\right\}, \tag{10}$$

and otherwise set $k_1^{(s)} = k_1^{(s-1)}$, where $\psi_1(\lambda_a, k_1)$ is defined in (9).

We can use the obtained Gibbs samplers 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 intensity function $\lambda_i = \lambda_a k_1^i, \quad i = 1, 2, \ldots$ A Bayes estimator of λ_i with respect to the squared error loss function is given by

$$E(\lambda_i \mid \mathcal{D}_n) = \iint \lambda_a k_1^i \pi(\lambda_a, k_1 \mid \mathcal{D}_n) \, d\lambda_a dk_1. \tag{11}$$

Let $\lambda_a^{(r,s)}$ and $k_1^{(r,s)}$ denote the variates for λ_a and k_1 drawn in the r^{th} replication and s^{th} iteration, where R and S are respectively, the total number of simulations and iterations of the Gibbs sampler.

Then, (11) can be estimated by

$$\hat{\lambda}_i = \frac{2}{RS}\sum_{r=1}^{R}\sum_{s=\frac{S}{2}+1}^{S} \lambda_a^{(r,s)} k_1^{(r,s)^i}. \tag{12}$$

3 Bayesian Inference for the Generalized Moranda Model

Considering now the intensity function (4) in the Poisson distribution (3), the likelihood function for λ_a, k_1 and k_2 is given by,

$$L(\lambda_a, k_1, k_2)\alpha\ \lambda_a^{d_1} k_1^{d_2(k_2)} exp\{-\lambda_a A(k_1.k_2)\} \tag{13}$$

where $d_1 = \sum_{i=1}^{n} m_i,\ d_2(k_2) = \sum_{i=1}^{n} i^{k_2} m_i$ and $A(k_1, k_2) = \sum_{i=1}^{n} k_1^{i^{k_2}}$.

Assuming prior independence among the parameters, consider the following prior densities for λ_a, k_1 and k_2:

$$\begin{array}{ll} (i) & \lambda_a \sim \Gamma(b_1, b_2) \\ (ii) & k_1 \sim B(c_1, c_2) \\ (iii) & k_2 \sim N(\mu_0, \sigma_0^2) \end{array} \tag{14}$$

where: $\Gamma(b_1, b_2)$ and $B(c_1, c_2)$ are defined in (6) and $N(\mu_0, \sigma_0^2)$ denotes a normal distribution with mean μ_0 and variance σ_0^2. We assume $b_1, b_2, c_1, c_2, \mu_0$ and σ_0^2 Known constants.

The joint posterior density for λ_a, k_1 and k_2 is given by,

$$\pi(\lambda_a, k_1, k_2 \mid \mathcal{D}_n)\alpha\lambda_a^{d_1+b_1-1}\ e^{-[b_2+A(k_1,k_2)]\lambda_a}$$
$$k_1^{d_2(k_2)+c_1-1}(1-k_1)^{c_2-1} exp\left\{-\frac{1}{2\sigma_0^2}(k_2-\mu_0)^2\right\} \tag{15}$$

where $\lambda_a > 0,\ 0 < k_1 < 1$ and $-\infty < k_2 < \infty$.

In this case, the marginal posterior densities for the Gibbs algorithm are given by

$$
\begin{array}{ll}
(i) & \lambda_a \mid k_1, k_2, \mathcal{D}_n \sim \Gamma\left[d_1 + b_1, b_2 + A\left(k_1, k_2\right)\right] \\
(ii) & \pi\left(k_1 \mid \lambda_a, k_2, \mathcal{D}_n\right) \alpha k_1^{d_2(k_2)+c_1-1}(1-k_1)^{c_2-1} exp\left\{-\lambda_a A(k_1, k_2)\right\} \\
(iii) & \pi\left(k_2 \mid \lambda_a, k_1, \mathcal{D}_n\right) \alpha k_1^{d_2(k_2)} exp\left\{-\lambda_a A(k_1, k_2)\right\} exp\left\{-\frac{1}{2\sigma_0^2}\left(k_2-\mu_0\right)^2\right\}.
\end{array}
\tag{16}
$$

Observe that, we need to use the Metropolis-Hastings algorithm to generate the variables k_1 and k_2 (see appendix). We could monitor the convergence of the Gibbs samplers using the Gelman and Rubin (1992) method that uses the analysis of variance technique to determine if further iterations are needed.

Considering $\lambda^{(r,s)}, k_1^{(r,s)}$ and $k_2^{(r,s)}$ denoting the variates for λ_a, k_1 and k_2 drawn in the r^{th} replication and s^{th} iteration, where R and S are respectively, the total number of simulations and iterations of the Gibbs Sampler, a Monte Carlo estimate for the intensity function (4) assuming the squared error loss function is given by

$$
\widehat{\lambda}_i = \frac{2}{RS} \sum_{r=1}^{R} \sum_{s=\frac{S}{2}+1}^{S} \lambda_a^{(r,s)} {k_1^{(r,s)}}^{i^{k_2^{(r,s)}}} \tag{17}
$$

4 Some Considerations on Model Selection

For model selection, we could use the predictive density for m_i given $\mathcal{D}_n, i = 1, 2, \ldots, n$.

The predictive density for m_i given $\mathcal{D}_n$ considering the generalized Moranda model with intensity function (4), is given by

$$
f\left(m_i \mid \mathcal{D}_n\right) = \iiint \frac{e^{-\lambda_i} \lambda_i^{m_i}}{m_i!} \pi\left(\lambda_a, k_1, k_2 \mid \mathcal{D}_n\right) d\lambda_a dk_1 dk_2 \tag{18}
$$

where $\lambda_i = \lambda_a k_1^{i^{k_2}}$.

Using the Gibbs samplers, (18) can be approximated by its Monte Carlo estimate,

$$
\widehat{f}(m_i \mid \mathcal{D}_n) = \frac{2}{RS} \sum_{r=1}^{R} \sum_{s=\frac{S}{2}+1}^{S} \frac{e^{-\lambda_a^{(r,s)} {k_1^{(r,s)}}^{i^{k_2^{(r,s)}}}} \left[\lambda_a^{(r,s)} {k_1^{(r,s)}}^{i^{k_2^{(r,s)}}}\right]^{m_i}}{m_i!} \tag{19}
$$

In the same way, we could use the Gibbs samplers generated by (8) to approximate the predictive density for m_i given $\mathcal{D}_n$, $i = 1, 2, \ldots, n$ by its Monte Carlo estimate in the Moranda model (2).

We can use the obtained estimates $c_i = \hat{f}(m_i \mid \mathcal{D}_n)$ in model selection. In this way, we could consider plots of c_i versus i ($i = 1, 2, \ldots, n$) for different models; large values of c_i (in average) indicates the better model.

We also could choose the model such that $c(\ell) = \pi_{i=1}^{n} c_i(\ell)$ is maximum (ℓ indexes models).

5 An Example

In Table 1, we have a data set introduced by Goel (1985) consisting of the observed number of failures per hour of a software tested for 25 hours.

CPU hour of testing	Number of failures per CPU hour	CPU hour of testing	Number of failures per CPU hour
1	27	14	5
2	16	15	5
3	11	16	6
4	10	17	0
5	11	18	5
6	7	19	1
7	2	20	1
8	5	21	2
9	3	22	1
10	1	23	2
11	4	24	1
12	7	25	1
13	2		

Table 1 - Data on software failures during system test

Assuming the Moranda's model with intensity function (2) for the software reliability data of table 1, and considering $b_1 = 16$, $b_2 = 0.8$, $c_1 = 2.4$ and $c_2 = 0.6$ for the prior densities (6) for λ_a and k_1, we generated 10 separate Gibbs chains from the marginal posterior densities (8) each of which ran for 1700 iterations. For each parameter, we consider the $215^{th}, 230^{th}, \ldots, 1700^{th}$ iterations, which for 10 chains yields a sample of size 1000. In table 2, we have the obtained posterior summaries for the parameters

λ_a and k_1, and in figure 1 we have the approximate marginal posterior densities considering the $S = 1000$ Gibbs samples. It is interesting to observe that the maximum likelihood estimators for λ_a and k_1 are given by $\widehat{\lambda}_a = 18.8849$ and $\widehat{K}_1 = 0.88285$.

	Mean	Median	S.D.	95% credible interval
λ_a	18.886	18.749	2.239	(14.7623; 23.6170)
k_1	0.88404	0.88427	0.01199	(0.85884; 0.90664)

Table 2. Posterior summaries for the Moranda's model

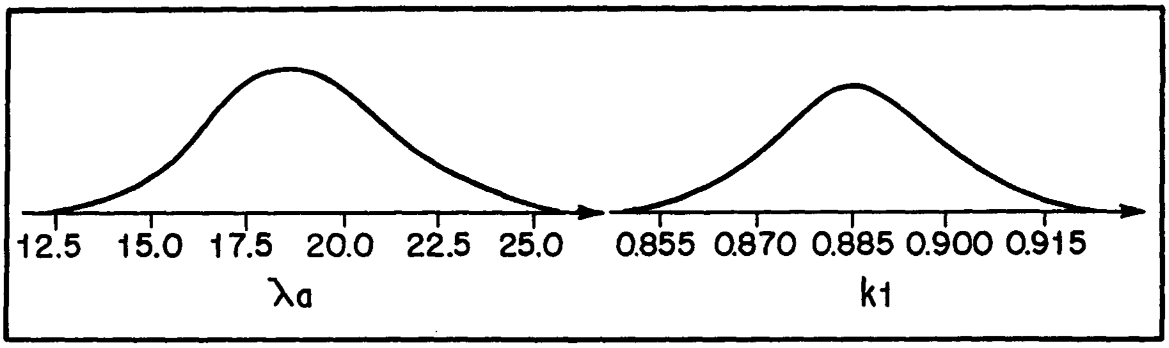

Figure 1. Marginal posterior densities for λ_a and k_1 (Moranda's model).

Assuming the generalized Moranda's model with intensity function (4) for the software reliability data of table 1, and considering $b_1 = 250$, $b_2 = 1$, $c_1 = 1.5$, $c_2 = 13$, $\mu_0 = 0.25$ and $\sigma_0 = 0.04$ for the prior densities (14) for λ_a, k_1 and k_2, we also generated 10 separate Gibbs chains from the marginal posterior densities (16) each of size 1700. Also taking the $215^{th}, 230^{th}, \ldots, 1700^{th}$ iterations, in each chain, we have a sample of λ_a, k_1 and k_2 of size 1000. In table 3, we have the obtained posterior summaries for the parameters λ_a, k_1 and k_2, and in figure 2 we have the approximate marginal posterior densities considering the $S = 1000$ Gibbs samples. In this case the maximum likelihood estimators for λ_a, k_1 and k_2 are given by $\widehat{\lambda}_a = 252.5160$, $\widehat{K}_1 = 0.1000$ and $\widehat{K}_2 = 0.2480$.

	Mean	Median	S.D.	95% credible interval
λ_a	249.87	249.71	14.59	(222.094; 280.003)
k_1	0.10145	0.10129	0.01163	(0.07954; 0.12530)
k_2	0.24816	0.24764	0.02023	(0.20982; 0.28903)

Table 3. Posterior summaries for the generalized Moranda's model

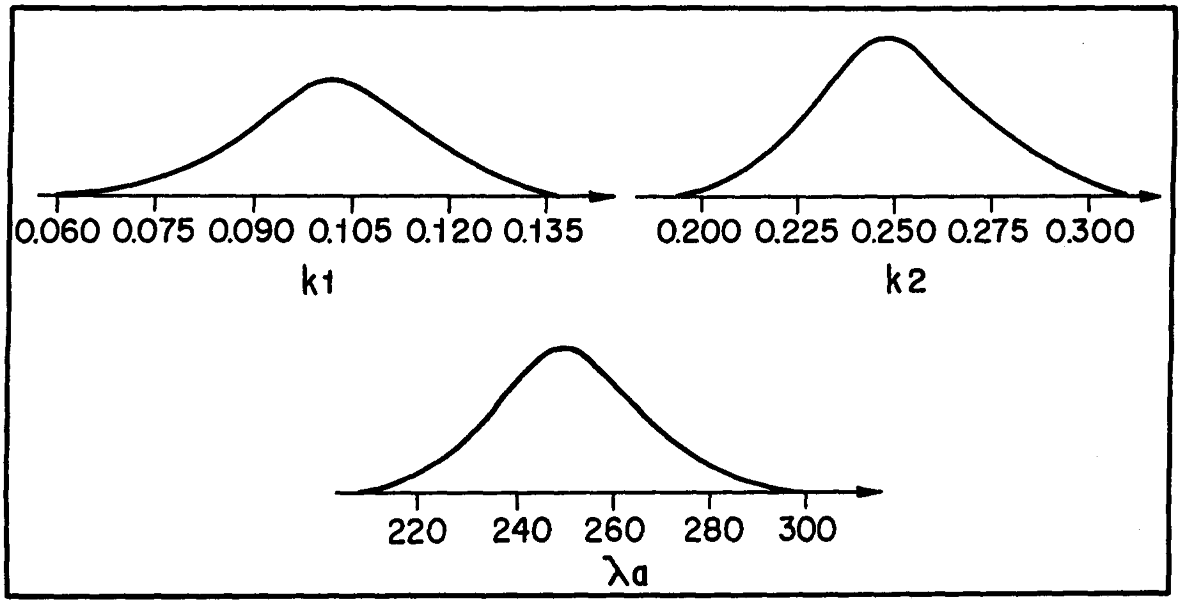

Figure 2. Marginal posterior densities for λ_a, k_1 and k_2
(Generalized Moranda's model)

In Table 4, we have approximate Bayes estimates for the intensity function $\lambda_i, i = 1, 2, \ldots, 25$ with respect to the squared error loss function considering the different models and using the Gibbs samples.

i	m_i	$\widehat{\lambda}_i$ (From(2))	$\widehat{\lambda}_i$ (from (4))
1	27.0000	16.6764	25.2901
2	16.0000	14.7280	16.4425
3	11.0000	13.0097	12.3227
4	10.0000	11.4940	9.8606
5	11.0000	10.1567	8.2027
6	7.0000	8.9767	7.0034
7	2.0000	7.9352	6.0931
8	5.0000	7.0158	5.3776
9	3.0000	6.2041	4.8002
10	1.0000	5.4872	4.3244
11	4.0000	4.8541	3.9256
12	7.0000	4.2948	3.5868
13	2.0000	3.8007	3.2954
14	5.0000	3.3640	3.0423
15	5.0000	2.9780	2.8206
16	6.0000	2.6367	2.6249
17	0.0000	2.3350	2.4510
18	5.0000	2.0682	2.2955
19	1.0000	1.8322	2.1557
20	1.0000	1.6234	2.0294
21	2.0000	1.4387	1.9149
22	1.0000	1.2752	1.8106
23	2.0000	1.1305	1.7153
24	1.0000	1.0023	1.6279
25	1.0000	0.8889	1.5475

Table 4. Bayes estimates for $\lambda_i, i = 1, 2, \ldots, 25$.

In figure 3, we have the graphs of the predictive densities $c_i = f(m_i \mid \mathcal{D}_n), i = 1, 2, \ldots, n$ approximated by its Monte Carlo estimate based on $S = 1000$ Gibbs Samples (see (19)) against i for the Moranda's model (2) and for the generalized Moranda's model (4) (see also table 5).

m_i	i	c_i (from(2))	c_i (from (4))
27	1	0.007579	0.063673
16	2	0.087905	0.091899
11	3	0.098471	0.106843
10	4	0.109904	0.120415
11	5	0.111538	0.076686
7	6	0.116553	0.145141
2	7	0.012495	0.044388
5	8	0.126682	0.169569
3	9	0.082581	0.151770
1	10	0.024635	0.060173
4	11	0.177350	0.191009
7	12	0.073060	0.043070
2	13	0.162694	0.200697
5	14	0.121911	0.102755
5	15	0.098481	0.088398
6	16	0.035311	0.034349
0	17	0.103811	0.090995
5	18	0.041876	0.054615
1	19	0.291472	0.250476
1	20	0.316193	0.266785
2	21	0.236793	0.263804
1	22	0.348197	0.294772
2	23	0.199545	0.257910
1	24	0.356929	0.316817
1	25	0.353628	0.325773
$c(\ell)$		2.244×10^{25}	2.039×10^{23}
		(from (2))	(from (4))

Table 5. Values for $c_i = f(m_i/\mathcal{D}_n)$, $i = 1, 2, \ldots, 25$ for the Moranda's model and the generalized Moranda's model.

From table 5 and figure 3, we observe better fit of the generalized Moranda's model for the Goel (1985) data of table 1 (in average there are bigger values of c_i for model with intensity (4); also bigger value for $c(\ell)$).

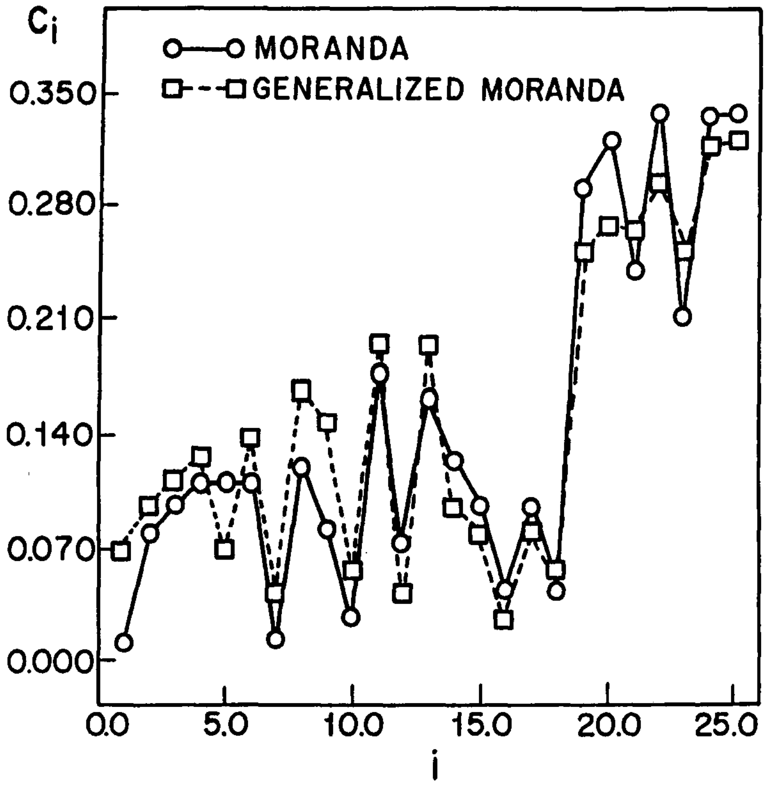

Figure 3. Graphs of c_i versus i for models (2) and (4)

6 Concluding Remarks

The inclusion of an additional parameter k_2 (see (4)) in the Moranda's model could improve the fit of the model for software reliability data. With the use of Metropolis-within-Gibbs algorithms, we can obtain accurate Bayesian inferences for the proposed model and for model selection. We also could get similar inference results considering different prior densities for the parameters of the model.

APPENDIX

Metropolis-Hastings Algorithm to Generate the Variables k_1 and k_2 in the Joint Posterior Density (15)

To generate a sample of draws from the joint posterior density (15), we sucessively sample λ_a from a gamma distribution $\Gamma[d_1 + b_1, b_2 + A(k_1, k_2)]$ and given this value of λ_a, we simulate k_1 from $\pi(k_1 \mid \lambda_a, k_2, \mathcal{D}_n)$ and k_2 from $\pi(k_2 \mid \lambda_a, k_1, \mathcal{D}_n)$.

In this way, observe that the conditional density for k_1 given λ_a, k_2 and $\mathcal{D}_n$ (see (16)) could be written in the form,

$$\pi\left(k_1 \mid \lambda_a, k_2, \mathcal{D}_n\right) \;\alpha k_1^{c_1-1}(1-k_1)^{c_2-1}\psi_2(\lambda_a, k_1, k_2) \qquad (A.1)$$

where $\psi_2(\lambda_a, k_1, k_2) = exp\{d_2(k_2)\ell n k_1 - \lambda_a \; A(k_1, k_2)\}$.

Thus, the value of k_1 is simulated as: at the s^{th} iteration (given the current value $\lambda_a^{(s)}$ and $k_2^{(s-1)}$), draw a candidate $k_1^{(s)}$ from a Beta distribution $B(c_1, c_2)$, if it satisfies stationarity, move to this point with probability

$$min\left\{\frac{\psi_2\left(\lambda_a^{(s)}, k_1^{(s)}, k_2^{(s-1)}\right)}{\psi_2\left(\lambda_a^{(s)}, k_1^{(s-1)}, k_2^{(s-1)}\right)}, 1\right\},$$

and otherwise set $k_1^{(s)} = k_1^{(s-1)}$.

To simulate k_2, observe that the conditional density for k_2 given λ_a, k_1 and $\mathcal{D}_n$ (see (16)) could be written by

$$\pi\left(k_2 \mid \lambda_a, k_1, \mathcal{D}_n\right) \alpha exp\left\{-\frac{1}{2\sigma_0^2}\left(k_2 - \mu_0\right)^2\right\} \psi_2\left(\lambda_a, k_1, k_2\right) \qquad (A.2)$$

where $\psi_2(\lambda_a, k_1, k_2)$ is defined in (A.1).

Thus, the value of k_2 is simulated as: at the s^{th} iteration (given the current value $\lambda_a^{(s)}$ and $k_1^{(s)}$), draw a candidate $k_2^{(s)}$ from a normal distribution $N(\mu_0, \sigma_0^2)$; if it satisfies stationarity, move to this point with probability,

$$min\left\{\frac{\psi_2\left(\lambda_a^{(s)}, k_1^{(s)}, k_2^{(s)}\right)}{\psi_2\left(\lambda_a^{(s)}, k_1^{(s)}, k_2^{(s-1)}\right)}, 1\right\},$$

and otherwise, set $k_2^{(s)} = k_2^{(s-1)}$.

7 References

Chib, S. and Greenberg, E. (1994). Understanding the Metropolis-Hasting algorithm. Manuscript.

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

Gelman, A.E. and Rubin, D. (1992). Inference from iterative simulation using multiple sequences. Statistical Science, 7, 457-472.

Goel, A.L. (1983). A guidebook for software reliability assessment. Technical Report RADC-TR-83-176.

Goel, A.L. (1985). Software reliability models: assumptions, limitations, and applicability, IEEE Transactions on software engineering, SE-11, 1411-1423.

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

Moranda, P.B. (1975). Prediction of software reliability and its applications, in proceedings of the annual reliability and maintainability symposium, 327-332, Washington, D.C.

Musa, J.D. and Okumoto, K. (1984). A logarithmic Poisson execution time model for software reliability measurement, in proceedings of seventh international conference on software engineering, Orlando, F.L., 230-238.

Musa, J.D., Iannino, A. and Okumoto, K. (1987). Software reliability: measurement, prediction, application. New York: Wiley.

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

Singpurwalla, N. and Wilson, S.P. (1994). Software reliability modeling. International Statistical Review, 62, 289-317.

NOTAS DO ICMSC

SÉRIE ESTATÍSTICA

028/96 RODRIGUES, J.; LEITE, J.G. - Inference for the software reliability using imperfect recapture debbugging model.

027/96 ACHCAR, J.A.; DEY, D.K.; NIVERTHI, M. - A bayesian approach using nonhomogeneous Poisson process for software reliability models.

026/96 ANDRADE, M.G.; VAL, J.B.R. do - Um método numérico baseado na solução do valor médio para a equação de Helmholtz parte II: Rede triangular.

025/96 ANDRADE, M.G.; VAL, J.B.R. do - Um método numérico baseado na solução do valor médio para a equação de Helmholtz parte I: malha quadrada.

024/96 MAZUCHELLI, J.; ACHCAR, J.A. - Análise bayesiana para modelos de crescimento.

023/96 ANDRADE, M.G.; SOARES, S.; CRUZ Jr., G.; VINHAL, C.D.N - Uma abordagem estocástica para o planejamento a longo prazo da operação de sistema hidrotérmicos.

022/95 ACHCAR, J. - A generalized Moranda software reliability model: a bayesian approach.

021/95 RODRIGUES, J. - Inference for the software reliability using asymmetric loss functions: A hierarchical Bayes approach.

020/95 RODRIGUES, J.; BOLFARINE, H.; CORDEIRO, G.M. - Nonlinear quasi-bayesian theory and the inverse linear regression.

019/95 LEANDO, R.A.; ACHCAR, J.A.; - Generation of bivariate lifetime data assuming the Block & Basu exponential