PARETO DISTRIBUTION: A SOFTWARE RELIABILITY GROWTH MODEL

R.R.L. KANTAM$^1$ and R. SUBBA RAO$^2$

$^1$ Department of Statistics, Acharya Nagarjuna University.
$^2$ Shri Vishnu Engineering College for Women, Bhimavaram.

ABSTRACT: A counting process representing the number of failures experienced in a given period of time by a system is proposed as a stochastic model for studying the reliability of the developed software. A Non Homogeneous Poisson Process (NHPP) with its mean value function specified by a Pareto model is considered. Its parameters are estimated to assess the reliability of a software system. The results are illustrated for a live software failure data.

Keywords: Non Homogeneous Poisson Process, Mean Value function, ML estimation

1. Introduction

Several studies have been undertaken to investigate the software error occurrence phenomenon. The objective of such studies is to improve software performance. These studies can be placed in one of the two categories. The first category emphasizes empirical analysis of data collected from software projects. The second category deals with the development of models for quantitative assessment of software performance. Software Reliability Engineering is a discipline that ensures failure free operation of software at the user end by employing scientific techniques to remove the maximum number of faults. The quality of the software system has many attributes such as maintainability, portability, usability, security, reliability, availability etc. Software reliability is the most dynamic attribute, which can measure and predict the operational quality of the product. The process of locating the faults in software to remove them is called the debugging process. The chronology of failure occurrence and fault removal can be utilized to provide an estimate of the software reliability and the level of fault content. Theory of probability plays a major role in software reliability model building.

A software system is subject to failures at random times caused by errors present in the system. Let $\{N(t), t > 0\}$ be a counting process representing the cumulative number of failures by time $t$. Since there are no failures at $t=0$, we have

$$N(0)=0.$$  \hfill (1)

It is reasonable to assume that the number of software failures during non overlapping time intervals do not affect each other. In other words, for any finite collection of times $t_1 < t_2 < \ldots < t_n$, the ‘$n$’ random variables $N(t_1), \{N(t_2) - N(t_1)\}, \ldots, \{N(t_n) - N(t_{n-1})\}$ are independent. This implies that the counting process $\{N(t), t > 0\}$ has independent increments.

*Corresponding author’s email: kantam-rrl@rediffmail.com
Let $m(t)$ represent the expected number of software failures by time `$t$'. Since the expected number of errors remaining in the system at any time is finite, $m(t)$ is a bounded, non decreasing function of `$t$' with the following boundary conditions

\[ m(t) = 0, \quad t = 0, \]
\[ = a, \quad t \to \infty \]

where `$a$' is the expected number of software errors to be eventually detected.

Suppose $N(t)$ is known to have a Poisson probability mass function with parameter $m(t)$.

That is

\[ P\{N(t) = n\} = \frac{m(t)^n \cdot e^{-m(t)}}{n!}, \quad n = 0,1,2,\ldots,\infty \]

Then $N(t)$ is called an NHPP. Various time domain models have appeared in the literature which describe the stochastic failure process by an NHPP which differ in the mean value functions $m(t)$. Some of them are due to Goel – Okumoto [1], Little Wood [5], Yamada et al. [11], Musa et al. [6], Hossain and Dahia [2], Pham [7] and the references there in. The mean value function is a non negative non decreasing function of $t$ with a limit as $t\to\infty$. Specifically all these characteristics are those of the cumulative distribution function of a continuous random variable with the additional property as its limit is 1 as $t\to\infty$. Exploring this property a distribution function multiplied by a positive constant is considered as the mean value function by many authors to develop a number of software reliability growth models through NHPP. On these lines, following wood [10] motivated by Little Wood [5], the authors have considered an NHPP with mean value function defined through the cumulative distribution of a Pareto random variable to propose it as an SRGM. The theoretical model is presented in section 2. Estimation is discussed in section 3 along with a comparative study.

2. The Proposed SRGM

In this section we consider $m(t)$ as given by

\[ m(t) = a\left[1 - \left(1 + \left(\frac{t}{\sigma}\right)^{\alpha}\right)^{-\alpha}\right] \]

(4)

where $[m(t)]/a$ is the cumulative distribution function of Pareto distribution of type IV (Johnson et al., 2004) for the present choice

\[ \lim_{t \to \infty} P\{N(t) = x\} = \lim_{t \to \infty} \frac{m(t)^x \cdot e^{-m(t)}}{x!} = \frac{e^{-a} \cdot a^x}{x!} \]

(5)

which is also a Poisson model with mean `$a$'.

Let $\overline{N}(t)$ be the number of errors remaining in the system at time `$t$'.

$\overline{N}(t) = N(\infty) - N(t)$
Pareto Distribution: A Software Reliability Growth Model

\[ E[N(t)] = E[N(\infty)] - E[N(0)] \]

\[ = a - m(t) = a - a \left[1 - \left(1 + \frac{t}{\sigma}\right)^{-\alpha}\right] = a \left(1 + \frac{t}{\sigma}\right)^{-\alpha} \]  

(6)

Let \( S_k \) be the time between \((k - 1)th\) and \(kth\) failures of the software product. Let \( X_k \) be the time up to the \(kth\) failure. Then the probability that \( S_k \) exceeds a real number \( s \), given that the total time up to the \((k - 1)th\) failure is equal to \( x \), is

\[ P[S_k > s/X_{k-1} = x] = e^{-m(x+s) - m(s)} \]  

(7)

This expression is called software reliability and is denoted by

\[ R_{S_k/X_{k-1}}(s/x) \]  

(8)

3. Estimation of Parameters

Let \( S_1, S_2, \ldots \) be a sequence of times between successive software failures associated with an NHPP \( \{N(t)\} \). Let \( X_k \) be equal to \( \sum_{i=1}^{k} S_i \), \( k = 1, 2, 3, \ldots \) which represents the time to failure \( K \).

Suppose we are given \( 'n' \) software failure times say \( x_1, x_2, \ldots, x_n \); (i.e.) there are \( 'n' \) time instants at which the first, second, third, \ldots \( n^{th} \) failures of a software are observed. This is a special case of a life testing experiment in which only one product is put to test and its successive failures are recorded alternately separated by error detections and debugging.

The likelihood function of such sample data is

\[ L = e^{-m(x)} \prod_{k=1}^{n} m'(x_k) \cdot\]  

where \( m'(t) \) is derivate of \( m(t) \).

\[ L = e^{-a \left\{ 1 - \left[1 + \left(\frac{x}{\sigma}\right)^{-\alpha}\right]\right\} - \prod_{k=1}^{n} a \frac{\alpha}{\sigma} \left[1 + \frac{x_k}{\sigma}\right]^{-\alpha}} \]  

\[ \log L = -a \left[1 - \left(\frac{x_k}{\sigma}\right)^{-\alpha} \right] + n \log a + n \log \alpha - n \log \sigma + \sum_{i=1}^{n} \log \left[1 + \frac{x_i}{\sigma}\right]^{-\alpha+1} \]  

(10)

Then the log likelihood equations to estimate the unknown parameters \( a \) and \( \sigma \) are given by

\[ \frac{\partial \log L}{\partial a} = 0 \Rightarrow \hat{a} = \frac{n}{\left[1 - \left(\frac{x_n}{\sigma}\right)^{-\alpha}\right]} \]  

(11)
The three equations (11), (12) and (13) are to be solved iteratively. When $\alpha$ is assumed to be known only one equation - that of $\sigma$ has to be solved by numerical methods to proceed for further evaluation of reliability measures. In order to overcome the iterative technique for solving $\sigma$ one may go for the two methods of modification for ML procedure as suggested by Srinivasa Rao and Kantam [8] for log-logistic distribution. We consider the method – I of Srinivasa Rao and Kantam [8] and use it in equation (13) to get the modified MLE (MMLE) of $\sigma$ which when used in equations (12) and (11) give the estimates of ‘$\alpha$’ and ‘$a$’ respectively, because method- I of Srinivasa Rao and Kantam [8] is rated as better when adopted to Pareto distribution (Subba Rao [9]). It may be noted that by method - I of Srinivasa Rao and Kantam [8], the modifications in the present case are the following approximations.

$$\frac{Z_i}{1 + Z_i} \approx \gamma_i + \delta_i Z_i, \quad i=1,2,3, \ldots n-1$$

$$\frac{Z_i}{(1 + Z_i)^{\alpha-1}} \approx \gamma_i + \delta_n Z_n$$

where $Z_i = \frac{x_i^2}{\sigma}$, $i=1,2,3, \ldots n$

The log likelihood equation (13) for $\sigma$ can be written as

$$\frac{1}{\alpha} \left\{ \frac{a \alpha Z_n}{(1 + Z_n)^{\alpha-1}} - n + \alpha (\alpha + 1) \sum_{i=1}^{n-1} \frac{Z_i}{1 + Z_i} \right\} = 0$$

With the suggested approximation it becomes

$$\frac{\partial \log L}{\partial \alpha} \equiv \frac{1}{\sigma} \left\{ a \alpha \left( \gamma_n + \delta_n Z_n \right) - n + \left( \alpha + 1 \right) \sum_{i=1}^{n-1} \left( \gamma_i + \delta_n Z_n \right) \right\} = 0$$

(14)

We can solve equation (14) for $\sigma$ to get MMLE. The corresponding estimate of ‘$\alpha$’ can be obtained by the equation (11). An estimator of software reliability can also be obtained from equation (7).

4. Illustration

The procedures narrated and derived in Sections 2 and 3 are explained by software failure data taken from Jelinski and Moranda [3]. The data are originally from the U.S. Navy Fleet computer programming centre, and consist of the errors in the development of software for the real-time, multi computer complex which forms the core of the Naval Tactical Data System (NTDS). The NTDS software consisted of some 38 different
modules. Each module is supposed to follow three stages; the production (development) phase, the test phase, and the user phase. The data are based on the trouble reports or ‘software anomaly reports’ for one of the larger modules, denoted as A-module. The times (days) between software failures and additional information for this module are summarized in the following table.

Table 1 : NTDS Data

<table>
<thead>
<tr>
<th>Error Number N</th>
<th>Time between Errors $S_k$ days</th>
<th>Cumulative Time $x_n = \sum S_k$ days</th>
<th>Error Number N</th>
<th>Production (Check out) Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Production (Check out) Phase</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>9</td>
<td>9</td>
<td>19</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>21</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>11</td>
<td>32</td>
<td>21</td>
<td>11</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>36</td>
<td>22</td>
<td>33</td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>43</td>
<td>23</td>
<td>7</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>45</td>
<td>24</td>
<td>91</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>50</td>
<td>25</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>58</td>
<td>26</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>63</td>
<td>Test Phase</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>7</td>
<td>70</td>
<td>27</td>
<td>87</td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>71</td>
<td>28</td>
<td>47</td>
</tr>
<tr>
<td>12</td>
<td>6</td>
<td>77</td>
<td>29</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td>78</td>
<td>30</td>
<td>9</td>
</tr>
<tr>
<td>14</td>
<td>9</td>
<td>87</td>
<td>31</td>
<td>135</td>
</tr>
<tr>
<td>15</td>
<td>4</td>
<td>91</td>
<td>User Phase</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td>92</td>
<td>32</td>
<td>258</td>
</tr>
<tr>
<td>17</td>
<td>3</td>
<td>95</td>
<td>Test Phase</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>3</td>
<td>98</td>
<td>33</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>34</td>
<td>35</td>
</tr>
</tbody>
</table>

Actually, 26 software errors are found during production phase and five additional errors during the test phase. One error is observed during the user phase and two more errors are noticed in a subsequent test phase indicating that a network of the module has taken place after the user error is found. The 26 data points are considered as an ordered random sample of size 26, supposed to have come from a Pareto distribution with parameters $\alpha$, $\sigma$. This supposition is verified using the q–q plot–correlation coefficient method at $\alpha = 2, 3, 4$. It is found that Pareto distribution fits reasonably well to the
present data. Hence the log likelihood equation of ‘a’ and σ are solved simultaneously by Newton Raphson method in succession, at α = 2, 3, 4. The ML estimates so obtained are

<table>
<thead>
<tr>
<th>α</th>
<th>a</th>
<th>α</th>
<th>σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>49.653079</td>
<td>1.337628</td>
<td>269.013288</td>
</tr>
<tr>
<td>3</td>
<td>52.695625</td>
<td>1.052718</td>
<td>371.136944</td>
</tr>
<tr>
<td>4</td>
<td>123.032906</td>
<td>1.321060</td>
<td>497.887758</td>
</tr>
</tbody>
</table>

The value of L – the likelihood function is maximum, at the triplet

\[ \hat{a} = 49.653079, \quad \hat{\alpha} = 1.337628, \quad \hat{\sigma} = 269.013288 \]

Hence we may accept the three values as ML estimates of a, α, σ. The estimator of the reliability function from equation (8) at any time x beyond 250 days is given by

\[ R(s) = e^{-\left[ m(x+s) - m(s) \right]} \]

\[ R(250) = e^{-\left[ m(250 + 250) - m(250) \right]} = 0.091961 \]

5. Comparison

For the NTDS data Goel and Okumoto [1] have obtained ML estimates of a, σ as \[ \hat{a} = 33.99 \] and \[ \hat{\sigma} = 172.71157 \]. For the same data with our SRGM (Pareto approach) \[ \hat{a} = 49.653079, \quad \hat{\sigma} = 269.013288 \]. A choice between these two models for the data under consideration can be made with reference to a number of criteria. One of them is that model which gives maximum joint probability with the data and the corresponding estimates. In other words that model which gives more value for the likelihood function can be decided as a better fit. We evaluated the values of likelihood functions under Goel and Okomoto [1] NHPP and its estimates, and that under Pareto structure and its estimates. The natural log of Goel and Okumoto [1] structure at the sample and the associated estimates is

\[ L(a, \frac{1}{\sigma}) = n \ln a + n \ln b - a \left( 1 - e^{-bs} \right) - b \sum_{k=1}^{n} S_k = -142.6973007 \]

For the same structure the authors’ value is -72.295455. This indicates that the model presented is a better fit than that offered by [1].

6. Conclusion

The authors proposed an SRGM through NHPP based on Pareto type mean value function and suggested a modified M.L. estimator of its parameters to assess software reliability for a given software failure data. The value of the likelihood function of the present model at a live data is found to be more than that of a popular model suggested by Goel and Okomoto [1] for the same data. Therefore, authors claim their model to be a better fit.

References


**R.R.L. Kantam** is a professor of statistics at Acharya Nagarjuna University, Guntur-India. He has 31 years of teaching experience in statistics at Under Graduate and Post Graduate programs. As researcher in Statistics, he has successfully guided 8 students for M.Phil. in statistics and 5 students for Ph.D. in statistics. He has authored 51 research publications appeared various statistics journals published in India and other countries like US, UK, Germany, Pakistan, Sri Lanka and Bangladesh. He has been a referee for Journal of Applied Statistics (U.K), METRON (Italy), Pakistan Journal of Statistics (Pakistan), IAPQR – Transactions (India), Assam Statistical Review (India) and Gujarat Statistical Review (India). He has been a special speaker in technical sessions of a number of Seminars and Conferences. His areas of research interest are Statistical Inference, Reliability Studies, Quality Control Methods and Actuarial Statistics. As a teacher his present teaching areas are Probability Theory, Reliability, and Actuarial Statistics. His earlier teaching topics include Statistical Inference, Mathematical Analysis, Operations Research, Econometrics, Statistical Quality Control, Measure theory.

**R. Subba Rao** is an associate professor of mathematics at Shri Vishnu Engineering College for Women, Bhimavaram, India. He has 17 years of teaching experience both in Under Graduate and Post Graduate Programs. He has published several papers in different National and International Journals. His areas of research interest are Statistical Inference, Reliability Studies, Quality Control methods and Software Engineering. As a teacher his present teaching areas are Probability Theory, Bio Statistics and Mathematical Methods. His earlier teaching topics include Quantitative Techniques, Research Methodology, Optimization Techniques, Statistical Inference, Simulation Modeling and Discrete Structures.