ISSN: 0975-766X
CODEN: IJPTFI
Available Online through Research Article
www.ijptonline.com

BURR TYPE XII SOFTWARE RELIABILITY GROWTH MODEL: AN ORDER STATISTICS APPROACH

Dr.R.Satya Prasad*, M. Anuradha* and Dr. G.Sridevi*
Associate Professor, Dept. of CSE, Acharya Nagarjuna University,Guntur.
Research Scholar, Acharya Nagarjuna University, Guntur
Professor, Dept. of CSE, Malla Reddy Institute of Technology, Hyderabad.
Email: sridevi.gutta2012@gmail.com

Received on: 02-03-2017 *Accepted on: 29-03-2017*

Abstract

In recent years, the size and complexity of the software programs getting expanded and it is particularly pivotal to produce the product which is reliable, since the failures in the product may make an extraordinary misfortune. Several software reliability growth models exist to predict the reliability of software systems. Software reliability has a necessary impact in the product improvement process. Non-homogeneous Poisson process (NHPP) is a probabilistic model for determining software reliability. Use of order statistics is generally a new method for estimating software reliability for time domain data based on NHPP. This paper exhibits Burr Type XII model as a software reliability growth model and determines the expressions for an efficient reliability function using order statistics. Using the maximum likelihood (ML) estimation technique the parameters are estimated. The live datasets are examined and the results are displayed.

Keywords: Burr type XII distribution, Time domain data, NHPP

1. Introduction

The three parameter Burr type XII distribution was first introduced in the literature by Irving W. Burr (1942) and has gained special attention in the last two decades due to the importance of using it in practical situations. It has been applied in the areas of reliability studies and failure time modeling. The Burr XII (BXII) distribution, having logistic and Weibull as special sub models, is a very popular distribution for modeling lifetime data and for modeling phenomenon with monotone failure rates [1].

There are essentially two types of software reliability models-those that attempt to predict software reliability from design parameters and those that attempt to predict software reliability from test data. The first type of models are usually called “defect density” models and use code characteristics such as lines of code, nesting of loops, external references, input/outputs, and so forth to estimate the number of defects in the software. The second type of model is usually called “software reliability growth model”.

The main intention of this paper is to give a systematic procedure to show how to analyze the reliability of a software system using Burr type XII model with order statistics.

2. Ordered Burr Type Xii Model

2.1 Non-Homogenous Poisson Process Model

The main issue in the NHPP model is to determine an appropriate mean value function to denote the expected number of failures experienced up to a certain time. An NHPP is a realistic model for assessing software reliability and has very interesting and useful interpretation in debugging and testing the software. Software reliability probabilistic models can be classified as Markovian models and fault counting models. In Markovian model a Markov process represents the failure process. In fault counting model the failure process is described by stochastic process like Homogeneous Poisson Process (HPP), Non Homogeneous Poisson Process (NHPP) and Compound Poisson Process etc. A majority of failure counting models are based upon NHPP described in the following lines. There are numerous software reliability growth models available for use according to probabilistic assumptions. The Non Homogenous Poisson Process (NHPP) based software reliability growth models are proved to be quite successful in practical software reliability engineering [2]. Model parameters can be estimated by using maximum Likelihood Estimate (MLE). NHPP model formulation is described in the following lines.

A software system is subjected to failures at random times caused by errors present in the system. Let $\{N(t), t \geq 0\}$ be a counting process representing the cumulative number of failures by time ‘t’, where t is the failure intensity function, which is proportional to the residual fault content.

The mean value function and intensity function of Burr Type XII NHPP model are as follows. Let $m(t)$ represent the expected number of software failures by time ‘s’. The mean value function $m(t)$ is finite valued, non-decreasing, non-negative and bounded with the boundary conditions.

$$m(t) = \begin{cases} 0, \; t = 0 \\ a, \; t \to \infty \end{cases}$$

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 parameters $m(t)$ i.e.,

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

Then $N(t)$ is called an NHPP. Thus the stochastic behaviour of software failure phenomena can be described through the $N(t)$ process. Various time domain models have appeared in the literature that describes the stochastic failure process by an NHPP which differ in the mean value function $m(t)$.

2.2 Proposed Model

This paper proposes estimation of software reliability using order statistics approach based on Burr Type XII distribution model. The Burr distribution has a flexible shape and controllable scale and location which makes it appealing to fit to data. It is frequently used to model insurance claim sizes. The mean value function and intensity function of Burr Type XII NHPP model are as follows.

The Cumulative distributive function (CDF) is given by

$$m(t) = \int_0^1 \lambda(t)\, dt = a\left[1 - \left(1 + t^c\right)^{-b}\right] \quad (1)$$

$$= a\, F(t)$$

The Probability Density Function (PDF) of Burr XII distribution are given, respectively by

$$\lambda(t) = a\left(cbt^{c-1} \Big/ \left(1 + t^c\right)^{b+1} \right) = a\ f(t)$$

Where t>0, a>0, b>0 and c>0 denote the expected number of faults that would be detached given infinite testing time in case of finite failure NHPP models. In order to have an assessment of the software reliability, a, b and c are unknown parameters and estimated by using Newton Raphson method. Expressions are now delivered for estimating 'a', 'b' and 'c' for the Burr type XII model.

$$p\left\{N(t) = n\right\} = \frac{\left[m(t)\right]^n e^{-m(t)}}{n!}$$

$$\lim_{n \to \infty} p\left\{N(t) = n\right\} = \frac{e^{-a}.a^n}{n!}$$

This is also a Poisson model with mean ‘a’.

Let S_k be the time between $(k-1)^{th}$ and k^{th} failure of the software product. Let X_k be the time up to the k^{th} failure. Let us find out the probability that time between $(k-1)^{th}$ and k^{th} failures, i.e., S_k exceeds a real number ‘s’ given that the total time up to the $(k-1)^{th}$ failure is equal to x.

i.e., $P\left[S_k > \frac{s}{X_{k-1}} = x\right]$

$$R\ S_k/X_{k-1}(s/x) = e^{-[m(x+s)-m(s)]} \quad (2)$$

2.3 Order Statistics

Order Statistics can be used in several applications like data compression, survival analysis, Study of Reliability and many others [3].To improve and understand the logic behind process control methods, it is necessary to give some thought to the behavior of sampling. If the length of a single failure interval is measured, it is clear that occasionally a length will be found which is towards one end of the tails of the process’s normal distribution. This occurrence may lead to the wrong conclusion that the process requires adjustment. If a sample of four or five is taken, it is extremely unlikely that all four or five failure interval lengths will lie towards one extreme end of the distribution. If we take the average or length of four or five failure intervals, we shall have a much more reliable indicator of the state of the process. Any change in the process mean, unless it is extremely large, will be difficult to detect from individual results alone. A large number of individual readings are necessary before such a change was confirmed. A subgroup or a sample is a small set of observations on a process parameter or its output, taken together in time. The size and the frequency of sampling are the two major problems in choosing a subgroup. The smaller the subgroup, there is less opportunity for variation within it. The larger the sample size the narrower the distribution of the means and they become more sensitive to detect change. It is understood that, in any type of process control charting system, careful selection of subgroups is very important. The software failure data is in the form of <failure number, failure time>. By grouping a fixed number of data into one, the noise values may compensate each other for that period and thus the noise inherent in the failure data is reduced to great extent.

2.4 Maximum Likelihood Estimation

Parameter estimation is of primary importance in software reliability prediction. Once the analytical solution for m(t) is known for a given model, the parameters in the solution need to be determined. Parameter estimation is achieved by applying a technique of MLE (Maximum Likelihood Estimation), the most important and widely used estimation technique.

Mathematical derivation of parameter estimation

The mean value function of Burr Type XII distribution representing the number of failures experienced by the time 't' is given by [5]

$$m(t)=\left[a\left(1-\left(1+t^{c}\right)^{-b}\right)\right]$$

In order to group the Time domain data into non overlapping successive sub groups of size r, we need to take m(t) to the power r.

$$m(t)=\left[a\left(1-\left(1+t^{c}\right)^{-b}\right)\right]^{r} \tag{3}$$

$$m'(t)=r\left[a\left(1-\frac{1}{(1+t^{c})^{b}}\right)\right]^{r-1}\cdot\left[\frac{abct^{c-1}}{\left(1+t^{c}\right)^{b+1}}\right]$$

To get the estimates of 'a', 'b' and 'c' for a sample of *n* units, the likelihood function must be obtained first.

$$L=e^{-m(t)}\prod_{i=1}^{n}m'(t_i) \tag{4}$$

$$L=e^{-a\left[1-(1+t^{c})^{-b}\right]^{r}}\prod_{i=1}^{n}r\left[a-\frac{a}{\left(1+t^{c}\right)^{b}}\right]^{r-1}\frac{abct^{c-1}}{\left(1+t^{c}\right)^{b+1}}$$

$$LogL=-a^{r}\left[1-\frac{1}{\left(1+t^{c}\right)^{b}}\right]^{r}+\sum_{i=1}^{n}\log r+\sum_{i=1}^{n}(r-1)\log\left(a-\frac{a}{\left(1+t_i^{c}\right)^{b}}\right)+ \tag{5}$$
$$\sum_{i=1}^{n}\left(\log a+\log b+\log c+(c-1)\log t_i-(b+1)\log(t_i^{c}+1)\right)$$

Differentiating Log L with respect to 'a', and equating to 0 (i.e., $\frac{\partial LogL}{\partial a}=0$) we get

$$a^{r}=n\left[\frac{\left(t^{c}+1\right)^{b}}{\left(t^{c}+1\right)^{b}-1}\right]^{r} \tag{6}$$

Differentiating Log L with respect to 'b', and equating to 0 (i.e., $\frac{\partial LogL}{\partial b} = 0$) we get

$$g(b) = \left[\left(\frac{nr}{(t+1)^b - 1}\right)\log\left(\frac{1}{1+t}\right)\right] - \sum_{i=1}^{n}\left[\frac{r-1}{\left(1+t_i\right)^b - 1}\log\left(\frac{1}{1+t_i}\right)\right] + \frac{n}{b} - \sum_{i=1}^{n}\log(t_i + 1) \quad (7)$$

Again differentiating $g(b)$ with respect to 'b' and equating to 0 $\left(i.e., \frac{\partial^2 LogL}{\partial b^2} = 0\right)$

$$g'(b) = -nr\log\left(\frac{1}{1+t}\right)\frac{\left(t+1\right)^b \log\left(t+1\right)}{\left[\left(t+1\right)^b - 1\right]^2} + \sum_{i=1}^{n}\log\left(\frac{1}{1+t_i}\right)\frac{\left(t_i+1\right)^b \log\left(1+t_i\right)}{\left[\left(t_i+1\right)^b - 1\right]^2} - \frac{n}{b^2} \quad (8)$$

Differentiating Log L with respect to 'c', and equating to 0 (i.e., $\frac{\partial LogL}{\partial c} = 0$) we get

$$g(c) = -nr\frac{\log t}{\left(1+t^c\right)} + \sum_{i=1}^{n}\left(r-1\right)\left(\frac{\log t_i}{1+t_i^c}\right) + \frac{n}{c} + \sum_{i=1}^{n}\left[\log t_i - \left(\frac{2}{1+t_i^c}\right)t_i^c.\log t_i\right] \quad (9)$$

Again differentiating $g(c)$ with respect to 'b' and equating to 0 $\left(i.e., \frac{\partial^2 LogL}{\partial c^2} = 0\right)$

$$g'(c) = nr\left[2\log t.\frac{t^c}{(1+t^c)^2}\right] - \sum_{i=1}^{n}\left[\left(r-1\right)2\log t_i.\frac{t_i^c}{\left(1+t_i^c\right)^2}\right] - \frac{n}{c^2} + \sum_{i=1}^{n}\left[\log t_i - 2\log t_i\left(t_i^c \log t_i \frac{1}{\left(1+t_i^c\right)^2}\right)\right] \quad (10)$$

The parameters 'b' and 'c' are estimated by iterative Newton- Raphson using

$$b_{n+1} = b_n - \frac{g(b_n)}{g'(b_n)} \quad (11)$$

$$c_{n+1} = c_n - \frac{g(c_n)}{g'(c_n)} \quad (12)$$

Which are substituted in (6) to determine 'a'.

3. Data Analysis:

CSR3 Data set [12] has been taken into consideration to analyze the software reliability.

Table 1: CSR3 Data set.

Failure No	Time Between Failures (hrs)	Failure No	Time Between Failures (hrs)	Failure No	Time Between Failures (hrs)
1	33	36	5	71	55
2	9	37	66	72	409
3	4	38	289	73	36
4	66	39	3	74	15
5	0.5	40	9	75	573
6	18	41	12	76	583
7	149	42	18	77	60
8	14	43	9	78	19
9	15	44	75	79	20
10	50	45	15	80	79
11	81	46	291	81	24
12	34	47	212	82	540
13	85	48	4	83	52
14	54	49	5	84	1596
15	3	50	308	85	314
16	15	51	269	86	1
17	6	52	276	87	763
18	8	53	1	88	10
19	130	54	400	89	20
20	19	55	294	90	144
21	19	56	227	91	28
22	112	57	118	92	56
23	15	58	13	93	476
24	16	59	47	94	65
25	154	60	89	95	98
26	50	61	242	96	884
27	10	62	99	97	212
28	2	63	609	98	287
29	22	64	83	99	53
30	53	65	2	100	3
31	19	66	26	101	831

32	58	67	586	102	43
33	20	68	708	103	55
34	3	69	6	104	109
35	92	70	4		

Table 2: CSR3 Data set (4th Order) (Michael R.Lyu., 1996a).

Failure No	4th Order Time Between Failures S_k days	4th Order cumulative Time Between Failures $X_n = \sum S_k days$
1	112	112
2	181.5	293.5
3	180	473.5
4	157	630.5
5	163	793.5
6	162	955.5
7	216	1171.5
8	152	1323.5
9	120	1443.5
10	367	1810.5
11	114	1924.5
12	522	2446.5
13	858	3304.5
14	922	4226.5
15	267	4493.5
16	1031	5524.5
17	1322	6846.5
18	474	7320.5
19	1207	8527.5
20	178	8705.5
21	2212	10917.5
22	1088	12005.5
23	248	12253.5
24	1523	13776.5
25	555	14331.5
26	1038	15369.5

The CSR3 data set consists of 26 failures for 4th order statistics in 15369.5 days. By solving the equations using Newton-Raphson method, the MLE's of a, b and c for CSR3 data set are

$a = 6.500000$

$b = 0.911819$

$c = 1.001298$

The estimator of the reliability function at any time x beyond 15369.5 days is give by

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

$$= e^{-[m(955.5+15369.5)-m(15369.5)]}$$

$$= 0.9999475$$

Table 3: CSR3 Data set (5th Order) (Michael R.Lyu., 1996a).

Failure No	5th Order Time Between Failures S_k days	5th Order cumulative Time Between Failures $X_n = \sum S_k days$
1	112.5	112.5
2	246	358.5
3	257	615.5
4	178	793.5
5	316	1109.5
6	137	1246.5
7	192	1438.5
8	372	1810.5
9	129	1939.5
10	820	2759.5
11	1240	3999.5
12	494	4493.5
13	1033	5526.5
14	1330	6856.5
15	1088	7944.5
16	761	8705.5
17	2526	11231.5
18	938	12169.5
19	723	12892.5
20	1439	14331.5

The CSR3 data set consists of 20 failures for 5th order statistics in 14331.5 days. By solving the equations using Newton-Raphson method, the MLE's of a, b and c for CSR3 data set are

$a = 4.00000$

$b = 0.930006$

$c = 1.001078$

The estimator of the reliability function at any time x beyond 14331.5 days is give by

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

$$= e^{-[m(1109.5+14331.5)-m(14331.5)]}$$

$$= 0.9999637$$

4. Conclusion

In this paper Burr type XII distribution model with order statistics has been proposed and tested with live data sets. It is observed that the reliability is high for 5th order statistic than 4th order statistics. Finally it can be concluded that the model has produced very good results and it is highly suitable for computing reliability.

References

1. G Sridevi**,** C.M.Sheela Rani, "Comparison of Software reliability analysis for Burr distribution**",** Journal of Theoretical and Applied Information Technology, 10th November 2015. Vol.81. No.1, ISSN: 1992-8645, E-ISSN: 1817-3195.
2. R.Satya Prasad, K.V.Murali Mohan and G Sridevi, "Burr Type XII Software Reliability Growth Model", International Journal of Computer Applications (IJCA), Vol.108, No.16, December 2014, pp[16-20], Published by Foundation of Computer Science (0975-8887).
3. Arak M.Mathai, "Order Statistics from a Logistic Distribution and Applications to Survival and Reliability Analysis", IEEE Transactions on Reliabiity, vol. 52, No. 2, 2003.
4. Goel. A.L and Okumoto. K., "A Time-dependent error-detection rate model for software and other performance measures," IEEE Trans. Reliability, vol R-28, Aug, pp. 206 – 211, 1979.
5. W. Burr, "Cumulative frequency functions," Annals of Mathematical Statistics, vol. 13, pp. 215–232, 1942.