Journal of Reliability and Statistical Studies; ISSN (Print): 0974-8024, (Online): 2229-5666
Vol. 11, Issue 2 (2018): 01-07

BAYES ESTIMATORS OF PARAMETERS OF BINOMIAL TYPE RAYLEIGH CLASS SOFTWARE RELIABILITY GROWTH MODEL USING NON-INFORMATIVE PRIORS

Rajesh Singh[1], Pritee Singh[2] and Kailash R. Kale[3]
[1]Department of Statistics, S.G.B. Amravati University, Amravati, India
[2]Department of Statistics, Institute of Science, Nagpur, India
[3]Department of Statistics, R. D. N. College, Mumbai, India
E Mail: [1]rsinghamt@hotmail.com; [2]priteesingh25@gmail.com;
[3]kailashkale10@gmail.com

Received January 12, 2018
Modified October 06, 2018
Accepted October 27, 2018

Abstract

In this paper, an attempt is made to obtain the estimators for the parameters of one parameter Binomial type Rayleigh class Software Reliability Growth Model (SRGM) using Bayesian paradigm. The failure intensity of this model has been characterized by a mathematical function of total number of failures η_0 and scale parameter η_1. The total numbers of failures present in the software initially and the rate at which the failures occurred before testing is unknown to the software tester due to this reason the non-informative priors are proposed for both the parameters. The estimators for the parameters η_0 and η_1 using Bayesian method have been obtained under squared error loss function. The performance of both the proposed Bayes estimators is compared with their related MLEs on the basis of risk efficiencies. The risk efficiencies are obtained by Monte Carlo simulation technique. It is observed that both the proposed Bayes estimators perform well for proper choices of execution time t_e.

Key Words: Binomial Process, Non-Informative Prior, Maximum Likelihood Estimator (MLE), Rayleigh Class, Software Reliability Growth Model (SRGM).

1. Introduction

The software is an essential part of many systems and these systems are dependent on reliable operation of software components. Day by day, the software are updating in features and institutions/industries are trying to make it very portable. Now days, the portability of software in the sense of quality, reliability, size and its cost become the issue of great competition among various companies. It becomes mandatory to produce the software with better quality which can satisfy the user's requirement. Since, the software is manmade, complex and of large magnitude, hence, the chances of occurring failures are high. The failure of software may caused due to programming errors related to memory, language-specific, calling third party libraries, extra compilation, standard library etc. The operational effects of failures are large as well as often vital and may break down the system. Recently, the operation and development cost, magnitude, complexity of software are increased. The software has both harmful and beneficial nature. It is beneficial for failure free, smooth working and fast execution of software however, it is harmful because no one knows when, where and how software will fail [5]. Therefore, the reliability of software is essential to be assessed for better results which satisfy the needs of customer.

The Binomial type of occurrence of software failures which follows one parameter Rayleigh distribution is characterized as failure intensity for software failure data. The failure intensity of this model is assumed to be a function of total number of failures present initially (at time $t = 0$) in the software i.e. η_0 and scale parameter i.e. η_1 involved in Rayleigh distribution. Many researchers like [4], [7], [9]-[11], [13]-[15] have also demonstrated use of Rayleigh Class failure intensity in software reliability. In this paper, the Binomial type Rayleigh Class Software Reliability Growth Model is proposed and their Bayes estimators are obtained. The Bayesian estimators performs better than estimators obtained by classical estimation procedures if the sample size is small and the choice of priors are done properly. Both the parameters are estimated using Bayesian technique as given in [6], [12] and their performance is compared with the corresponding maximum likelihood estimators.

2. Model Evaluation

2.1 Assumptions

i) Failure occurrence process is Binomial.
ii) The fault that causes failure will be removed immediately.
iii) There are finite inherent faults present in software.
iv) Each failure occurs independently and randomly in time according to the constant per fault hazard rate.

2.2 Failures Intensity and Mean Value Function

Let T be the positive random variable representing time to failure with realization t then Rayleigh Class failure intensity $\lambda(t)$ and mean value function is given by

$$\lambda(t) = \eta_0 t \eta_1^{-2} e^{-\frac{1}{2}[t\eta_1^{-1}]^2} \tag{1}$$

$$\mu(t) = \eta_0 \left[1 - e^{-\frac{1}{2}(t\eta_1^{-1})^2}\right] \qquad \text{where } t > 0, \eta_1 > 0, \eta_0 > 0 \tag{2}$$

2.3 Failures experienced up to time t

If $M(t)$ is random variable representing failures experienced in software up to time t then probability of obtaining $M(t) = m$ failures is

$$P[M(t) = m\,] = \binom{\eta_0}{m} \left[1 - e^{-\frac{1}{2}(t\eta_1^{-1})^2}\right]^m \left[e^{-\frac{1}{2}(t\eta_1^{-1})^2}\right]^{(\eta_0 - m)} \tag{3}$$

where, $m = 0,1,\dots,\eta_0$.

2.4 Failures remaining by time t

If $Q(t)$ is random variable representing failures remaining in software by time t then probability of obtaining $Q(t) = q$ failures is

$$P[Q(t) = q\,] = \binom{\eta_0}{q} e^{-\frac{q}{2}(t\eta_1^{-1})^2} \left[1 - e^{-\frac{1}{2}(t\eta_1^{-1})^2}\right]^{\eta_0 - q} \tag{4}$$

where $q = 0,1,\dots,\eta_0$.

3. Maximum likelihood estimators

Suppose software is executed up to t_e time and if $t_1, t_2, \dots, t_{m_e} (= \underline{t})$ be failure times of m_e failures experienced upto execution time t_e. The likelihood can be written as

$$L(\eta_0, \eta_1 | \underline{t}) = e^{-\frac{1}{2}[t_e^2(\eta_0 - m_e) + T^*]\eta_1^{-2}} \eta_1^{-2m_e} \eta_0^{\underline{m_e}} \left(\prod_{i=1}^{m_e} t_i\right) \quad (5)$$

where $\eta_0^{\underline{m_e}}$ is Falling Factorials falling up to m_e (cf. [2], [3] and [8] etc.). The MLEs of η_0 and η_1 can be obtained by solving simultaneous equations

$$\sum_{i=1}^{m_e} (\hat{\eta}_{m0} - i + 1)^{-1} = t_e^2 \eta_1^{-2}/2$$

and

$$\hat{\eta}_{m1} = \{[t_e^2(\hat{\eta}_{m0} - m_e) + T^*] m_e^{-1}/2\}^{1/2}$$

where $T^* = \sum_{i=1}^{m_e} t_i^2$.

4. Priors

Generally, the total number of failures present initially at time $t = 0$ in the software is unknown to the tester or he may have very little past experience about it. Similarly, past information about the rate or intensity at which failures are occurs may not be beneficial, if lines of code in the software are different. Similarly, since η_1 is unknown, the experimenter may not have guess about the, how frequently the failures occur i.e. about the scale parameter of Rayleigh density η_1. Hence, in this situation, it can be assumed that no or very little information is available about both the parameters η_0 and η_1. Hence, the non-informative prior for η_0 and η_1 are more suitable and are

$$g(\eta_0) = \begin{cases} \eta_0^{-1} & ; \quad \eta_0 > 0 \\ 0 & ; Otherwise \end{cases}$$

and

$$g(\eta_1) = \begin{cases} \eta_1^{-1} & ; \quad \eta_1 > 0 \\ 0 & ; Otherwise \end{cases}$$

Consider that η_0 and η_1 are independent. Then, the joint prior distribution will be as

$$g(\eta_0, \eta_1) \propto \begin{cases} \eta_0^{-1}\eta_1^{-1} & ; \quad 0 < \eta_0, \eta_1 < \infty \\ 0 & ; \quad \text{Otherwise} \end{cases} \quad (6)$$

5. Bayesian Estimation

5.1 Joint Posterior of η_0 and η_1

The joint posterior of η_0 and η_1 given $\underline{t}$ is

$$\pi(\eta_0, \eta_1 | \underline{t}) \propto e^{-\frac{1}{2}[t_e^2(\eta_0 - m_e) + T^*]\eta_1^{-2}} \eta_1^{-2m_e - 1} \eta_0^{\underline{m_e}} \eta_0^{-1}$$

where, $t_e > 0,\ 0 < \eta_1 < \infty$ and $m_e < \eta_0 < \infty$.

The normalizing constant D is given as

$$D = K_1 \sum_{m=0}^{m_e} S_{m_e}^{(m)} m_e^* \; {}_2F_1(1, m_e - 1;\; m_e - m + 1, T^{**}) \tag{7}$$

where $S_{m_e}^{(m)}$ is Stirling numbers of first kind, ${}_2F_1(a, b;\, c, z)$ is Gauss's Hypergeometric series [1] $T^{**} = 1 - (t_e^2 m_e / T^*)$, $m_e^* = m_e^m (m_e - m)^{-1}$ and $K_1 = 2^{m_e - 1} (T^*)^{-m_e} \Gamma(m_e)$.

5.2 Marginal Posteriors of η_0 and η_1

The marginal posterior of η_0 given $\underline{t}$ is

$$\pi(\eta_0 | \underline{t}) \propto 2^{m_e - 1} \Gamma(m_e) \eta_0^{\overline{m_e}} \eta_0^{-1} [t_e^2 (\eta_0 - m_e) + T^*]^{-m_e} \tag{8}$$

where, $\eta_0 > m_e$.

The marginal posterior of η_1 given $\underline{t}$ is

$$\pi(\eta_1 | \underline{t}) \propto e^{-\frac{1}{2}[T^* - t_e^2 m_e] \eta_1^{-2}} \eta_1^{-2m_e - 1} \eta_1^* \tag{9}$$

where, $\eta_1^* = \sum_{m=0}^{m_e} S_{m_e}^{(m)} \left(\frac{t_e^2}{2}\right)^{-m} \eta_1^{2m} \Gamma(m, \eta_1^{-2} t_e^2 m_e / 2)$.

5.3 Bayes Estimators

The Bayes estimator of η_0 is posterior mean i.e.

$$\hat{\eta}_{B0} = K_2 \sum_{m=0}^{m_e} S_{m_e}^{(m)} m_e^{**} \; {}_2F_1(1, m_e + 2;\; m_e - m + 1, T^{**}) \tag{10}$$

where, $K_2 = 2^{m_e + 1} (T^*)^{-m_e - 2} m_e \Gamma(m_e + 2)$ and $m_e^* = m_e^m (m_e - m + 1)^{-1}$.

Similarly, the Bayes estimator of η_1 is posterior mean of (9), is

$$\hat{\eta}_{B1} = K_3 \sum_{m=0}^{m_e} S_{m_e}^{(m)} m_e^{\#} \; {}_2F_1\left(1, m_e - \frac{1}{2};\; m_e - m + \frac{1}{2}, T^{**}\right) \tag{11}$$

where, $K_3 = 2^{m_e - \frac{3}{2}} (T^*)^{-m_e + \frac{1}{2}} m_e \Gamma\left(m_e - \frac{1}{2}\right)$ and $m_e^{\#} = m_e^m \left(m_e - m - \frac{1}{2}\right)^{-1}$.

6. Discussion and Graphical Evaluation

The performance of above proposed Bayes estimators i.e. $\hat{\eta}_{B0}$ and $\hat{\eta}_{B1}$ is compared with corresponding MLEs. The comparison is made on the basis of risk efficiencies under squared error loss calculated by generating m_e failures upto execution time t_e using Monte Carlo simulation technique. To study the performance, the variations over the parameters are considered as $\eta_0 (= 2.0(0.2)3.8)$, $\eta_1 (=$

$20(1)29)$ and for $t_e(= 4.0(0.5)5.0)$. It is noticed that the difference in the risk efficiencies using Monte Carlo simulation technique is very meager between the simulations of generating 1000 and 5000 samples, so to do the fast computation, the study is done for generating 1000 samples and the same is presented here in figures from Fig. 1 to Fig. 3. From these figures, it is seen that the risk efficiencies of proposed Bayes estimator $\hat{\eta}_{B0}$ are increasing as the value of execution time t_e increases. It is also seen that $\hat{\eta}_{B0}$ is performing better than its corresponding MLE for increasing t_e. Further, it observed that the performance of proposed Bayes estimator $\hat{\eta}_{B1}$ remains constant as the value of t_e increases.

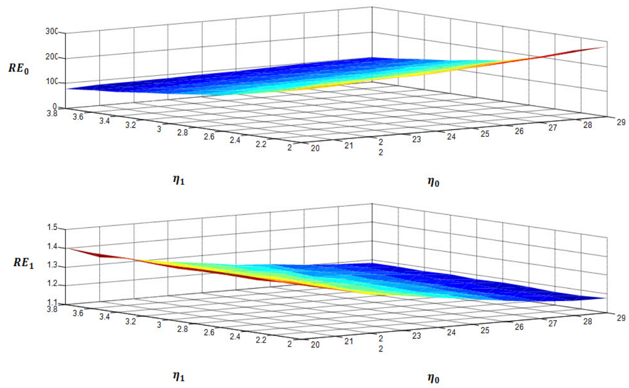

Fig. 1 Risk Efficiencies of $\hat{\eta}_{B0}$ and $\hat{\eta}_{B1}$ for $t_e = 4.0$.

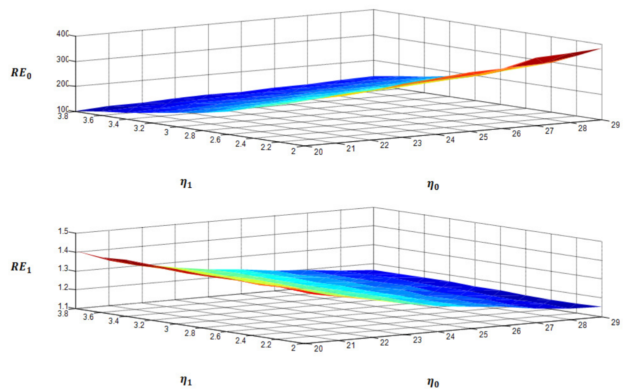

Fig. 2 Risk Efficiencies of $\hat{\eta}_{B0}$ and $\hat{\eta}_{B1}$ for $t_e = 4.5$.

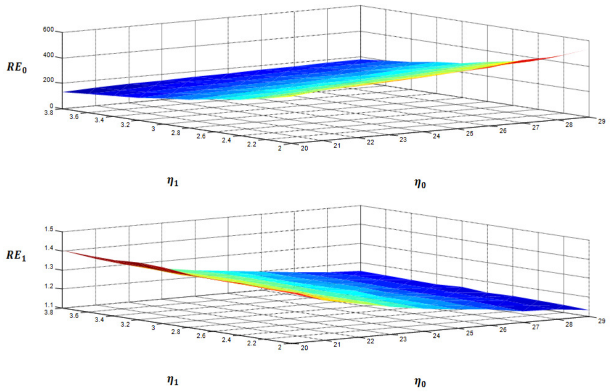

Fig. 3 Risk Efficiencies of $\hat{\eta}_{B0}$ and $\hat{\eta}_{B1}$ for $t_e = 5.0$.

7. Conclusion

On the basis of risk efficiencies, it can be concluded that proposed Bayes estimators for the parameters of Binomial Type Rayleigh Class SRGM are having less risks than their corresponding MLEs when non-informative priors are considered. It can also be suggested that the Bayes estimators i.e. $\hat{\eta}_{B0}$ and $\hat{\eta}_{B1}$ can be preferred over MLEs when proper execution time t_e is chosen.

References

1. Abramowitz, M. and Stegun, I. A. (1965). Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. Dover publications, New York.
2. Gradshteyn, I. S. and Ryzhik, I. M. (1994). Table of Integrals, Series, and Products, Alan Jeffrey (editor) 5th Ed., Academic Press, New York.
3. Graham, R. L. Knuth, D. E. and Patashnik, O. (1994). Concrete Mathematics: A Foundation for Computer Science, 2nd Ed. New York: Addison-Wesley Publishing Co.
4. Kan, S. H. (2002). Metrics and Models in Software Quality Engineering. 2nd Ed., Addison Wesley, New York.
5. Lyu, M. R. (2007). Software Reliability Engineering: A Roadmap, Future of Software Engineering, (FOSE '07), 2007, p. 153-170.
6. Musa, J. D., Iannino, A. and Okumoto, K. (1987). Software Reliability:Measurement, Prediction, Application. McGraw-Hill, New York.
7. Norden, P. V., (1963). Useful Tools for Project Management, Operations Research in Research and Development, B. V. Dean, Ed., John Wiley and Sons, New York.
8. Osgood, B. and Wu, W. (2009). Falling factorials, generating functions and conjoint ranking tables, J. Int. Seq., Vol.12, Article 09.7.8.
9. Putnam L. H. (1978). A general empirical solution to the macro software sizing and estimating problem, IEEE Trans. Soft. Engi., Vol. 4, p. 345–367.
10. Schick G. J. and Wolverton R. W. (1973). Assessment of software reliability, Proc. of Operations Research, Physica-Verlag,Wurzburg-Wien, p. 395-422.

11. Schick G. J. and Wolverton R. W. (1978). An analysis of competing software reliability models, IEEE Trans. Soft. Engi., Vol.SE 4(2), p. 104-120.
12. Singh R. and Andure N. W. (2008). Bayes estimators for the parameters of the Poisson type exponential distribution, IAPQR trans., Vol. 33 (2), p. 121-128.
13. Singh R. and Kale K. R. (2016). Bayes estimates for the parameters of Poisson type length biased exponential class model using non-informative priors, Journal of Reliability and Statistical Studies; Vol. 9, Issue 1, p. 21-28.
14. Vouk M. A. (1992). Using reliability models during testing with non-operational profile, Proc. 2nd Bellcore/Purdue Symp. Issues in Software Reliability Estimation, p. 103–110.
15. Yamada,S. Ohtera, H. and Narihisa, H. (1986). Software reliability growth models with testing-effort, IEEE Trans. Reliability, Vol. 35, p. 19 – 23.

Appendix

$\eta_{\overline{0}}^{m_e}$ is Falling Factorials falling upto m_e. $S_{m_e}^{(m)}$ is Stirling numbers of first kind and can be calculated by using Stirling numbers of second kind. The term ${}_2F_1(a, b;\ c, z)$ is Gauss's Hypergeometric series.