A New Methodology for Software Reliability based on Statistical Modeling

Avinash S,Y.Srinivas, P.Annan naidu

Abstract
Reliability is one of the computable quality features of the software. To assess the reliability the software reliability growth models(SRGMS) are used at different test times based on statistical learning models. In all situations, Tradational time-based SRGMS may not be enough, and such models cannot recognize errors in small and medium sized applications.Numerous traditional reliability measures are used to test software errors during application development and testing. In the software testing and maintenance phase, however, new errors are taken into consideration in real time in order to decide the reliability estimate. In this article, we suggest using the Weibull model as a computational approach to eradicate the problem of software reliability modeling. In the suggested model, a new distribution model is suggested to improve the reliability estimation method. We compute the model developed and stabilize its efficiency with other popular software reliability growth models from the research publication. Our assessment results show that the proposed Model is worthier to S-shaped Yamada, Generalized Poisson, NHPP.

Keywords:
Softwarereliability,Software Reliability Growth Models, Testing, Statistical dependencies

1. Introduction

Software-based systems play a important role in this modern world. In the future, the reliance on software systems will increase exponentially. In such circumstances, these software systems should work accurately. The anticipation of a user community of any kind of software system is only one that is nothing more than software that is of very high-caliber. The meaning of high-caliber in this context is: "The software should do what it should do, and it should not do what it should not do." According to this high-caliber meaning, both Dos and Don should be concentrated.

To have better knowledge of software quality, the various factors that inspire software quality should be reviewed. The factors influencing software quality are usability, performance, maintainability, reliability, etc. In order to deliver the software of high-calibre, we need to improve the level of quality factors. In this view, software quality can be increased by improving the reliability of the software. The present work inspects the reliability of software under various software quality factors. The reliability of the software can be stated as the chance that it will work successfully in a certain environment for a certain period of time.

Early Software Reliability Models (SRGM) represents the relation between failure time and cumulative number of faults detect so far. To evaluate the software reliability growth during the test phase and to estimate future failure time of occurrence many of these Srgms have been proposed as parametric [1–14] and nonparametric[15–18] models. Traditional SRGMs area unit on the idea of the principle that the mean depends on the model follows either exponential growth [1, 3], or s-shaped growth [2, 11] or each [4–8]"[23].

The sole purpose of SRGM is to create a software quality at a very low cost in a fair duration with endeared reliability. In software development process reliability is one of the significant capacities. . The reliability of the software is essentially distinct as the probability of a predictable over quantified operation time interval. It would be very significant to know the probability of failure free course of action of a software system for a fixed period of time [10].

To increase the software measure of reliability Several software models are used and it can be classified into two levels namely prediction and estimation models. These two techniques are based on the supervision and assembling of failure data and the determination with statistical hypothesis. At first, using the prediction model, software reliability can be estimated at first in development step and amendments can be done to advance reliability [10]. The other is to estimate the factor values based on measured or assessed data containing random information [10].

Software dependability models are mainly of two kinds: one is with density defect [8] and second one is with improving software reliability [9].First kind of model observe models so as to make an effort to estimate software reliability based on design parameters and use implementation property such as nested loops, number of rows, input / output Estimate the number of software errors present[23].The second types, software reliability growth models, are used on models that try to estimate the reliability of software based on test data[23]. These models seek to show the relationship between fault detection data and well-known mathematical function such as Exponential and logarithmic functions[23]. The relevancy of these models relies on the level of correlation between the tests

Manuscript received September 5, 2023
Manuscript revised September 20, 2023
https://doi.org/10.22937/IJCSNS.2023.23.9.20

In general, to the extent of the software application these models are symbolize on escalating qualitative approach [10]. the reliability of software mainly focused on outcomes like failures and faults ignore the software development process. However, the complexity is decreased and an suitable plan is prepared and new methods are brought in ,which are applied in some classes. So we have to select the correct model that suits our required case[10].In addition, the modeling outcomes can't be acknowledged and applied blindly[10]. In this article, we suggest a reliability model based on software prediction.

This article is well thought-out as follows: Section II deals with NHPP(Non homogeneous poission process). In section III NASA data set. Section IV Least square estimation ,in section V the software reliability model based on Weibull distribution is highlighted. Conclusion of article in section VI.

2. Non homogeneous poission process(NHPP)

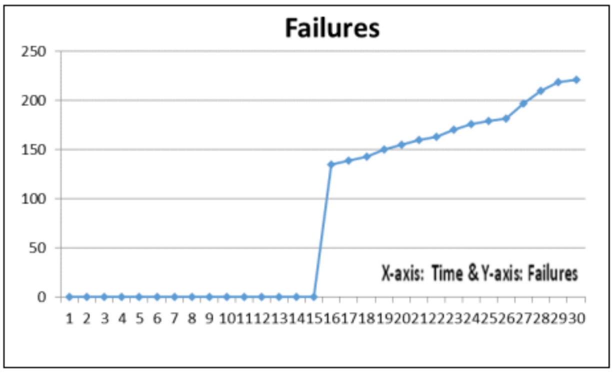
A non-homogeneous stochastic Poisson process is a process with the parameter λ (t) It demonstrates several random phenomena, including predicting forecasts for a certain value.This process is an addition ofthis model where λ (t) can be a stochastic process.The Nonhomogeneous poission process model consists of figuring out an average significance task to point towards the projected number of failures

3. Data set

In this case study, we classify the model on KC2 dataset from NASA and the failures are projected accurately. The failure data acquired is tested for correctness using Specificity, Sensitivity and F-measure. The original dataset of KC2 and its taxonomy process by Gaussian Mixture Model are existing below

Table 1 Data set

Test Time (Weeks)	Estimated Failures
1	75
2	81
3	86
4	90
5	93
6	96
7	98
8	99
9	100
10	100
11	100
12	115
13	120
14	123
15	130
16	135
17	139
18	143
19	150
20	155
21	160
22	163
23	170
24	176
25	179
26	182
27	197
28	210
29	219
30	221



4. Least square estimation method

Least squares parameter estimation involves the finding out of unknown variables during a specified solution.The function of knowledge is very important.The best technique of method of least squares match could also be a quite common approach for decreasing the total of squares of residuals, a residual being the distinction observed value and therefore the appropriate value provided by a model.The linear technique of statistical procedure drawback pans move into multivariate analysis.during this method, statistical inference to failure data obtained during testing or operation is put into practical use.the formula which is given by least square estimation is

Table 2.Least square estimation

Dataset #	a	B
1	31.524466	0.029012
2	23.655141	0.026521

5.Software reliability model based on Weibull distribution and Experimentation

In this method we have created a new statistical method based on weilbul distribution and the formula for the estimate the PDF is presented below.

$$f(x) = \sum_{x=0}^{1} \frac{1}{2\pi} e^{-\left(\frac{x-a}{b}\right)(\mu-\sigma)} \quad (1)$$

The values of a,b are estimated using the least square method and the values of μ and σ are estimated from the dataset.

5.1 Experimentation

Two data sets are considered for the experimentation. Each of the error values are given to the model to estimate the probability density functions.

Determination of fitness rate. to calculate fitness measure A normalized RMSE (Root Mean Square Error) is selected . it is also known as standard error and standard regression error. A Root mean square error measurementnearer to zero means better fit. The equation responsible for calculating the standardized RMSE is

$$\text{NRMSE} = \frac{1}{n}\sqrt{\left[\frac{1}{n}\sum_{i}^{n}(xi-yi)^2\right]} \quad (2)$$

where n: is the no of data points.
x_i: be the i th point of the observed data (original data).
y_i: be the PDF

Table 3. Estimation Failures

Test Time (Weeks)	Estimated Failures	PDF Values
1	75	0.02186
2	81	0.02124
3	86	0.03264
4	90	0.03787
5	93	0.04882
6	96	0.05267
7	98	0.05989
8	99	0.06176
9	100	0.06878
10	100	0.06776
11	100	0.07251
12	115	0.07682
13	120	0.07987
14	123	0.07876
15	130	0.08962
16	**135**	**0.008182**
17	139	0.04156
18	143	0.05687
19	150	0.09247
20	155	0.09848
21	160	0.098415
22	163	0.09967
23	170	0.010892
24	176	0.010898
25	179	0.035678
26	182	0.05464
27	197	0.07856
28	210	0.06872
29	219	0.07876
30	221	0.09652

The methodology is tested against productivity based on MSE and metrics such as sensivity and specificity.

"We calculate and balance the goodness of fit (GoF) act of the proposed model using NRMSE" [22]. to calculate the square of the distinction among the real and expected valuesNRMSEis used. The smaller NRMSEindicate the smallest adjustment error and better performance.

Table 4.NRMSE for test data

Root Mean Square Error Normalized Model	MSE	
	RMSE	Normalized RMSE
Yamada	22.7066	0.5677
Poisson 23.1365	23.1365	0.5784
NHPP	23.7988	0.5950
Raleigh Distribution	24.2316	0.5897
Proposed method	16.342	0.4123

Sensitivity, Specificity, and F-measure are used to test the productivity of the model. and the values obtained are tabulated below in Table

Table 5.Senstivity,specificity and F-measure

True Postive	False positive	Specificity	Sensitivity	F-measure
0.96	0.21	0.93.7	0.91	0.93
0.80	0.25	0.89	0.87	0.79

6. CONCLUSION

Software reliability is vital .in this article we suggested a software reliability growth model based on Weibull distribution. To estimating and monitoring software reliability the model is mainly used, which is viewed as a calculate of software quality.. Equations to find the maximum likelihood estimates of the parameters based on interval domain data are created. The method is tested against two data sets. We come to conclusion that our method of estimation and the control chart are giving a positive recommendation based on MSE and metrics based on sensivity and Specificity. Thismodel is a simple way for validation and for practitioners of software reliabilityis very suitable. The methodology used in this paper is a lot better than the methodology used by Xie et al [2002]. Therefore, for an early detection of software failures we may come to conclusion that this model is the appropriate choice.

7. References

[1] P. K. Kapur and R. B. Garg, "A software reliability growthmodel for an error removal phenomenon," software EngineeringJournal, vol. 7, no. 4, pp. 291–294, 1992

[2] S. Yamada and S. Osaki, "Software reliability Growth Modeling:Models and Applications," IEEE Transactions on software Engineering, vol. SE-11, no. 12, pp. 1431–1437, 1985.

[3] A. L. Goel and K. Okumoto, "Time-dependent error-detectionrate model for software reliability and other performancemeasures," IEEE Transactions on responsibility, vol. 28, no. 3, pp.206–211

.[4] A. L. Goel, "Software responsibility models: assumptions, limitations,and relevancy," IEEE Transactions on software system Engineering, vol. 11, no. 12, pp. 1411–1423, 1985.

[5] Y.Srinivas,Y.Vamsidhar, "software reliability growth model based on pareto type iii distribution," International Journal of Advanced research in Electrical, electronics and Instrumentation Engineering, Vol. 2, Issue 6, pp 2694-2698, 2013

[6] R. Subburaj, G. Gopal, and P. K. Kapur, "A software system responsibility growth model for important Quality Metrics," South African Journal of business Engineering, vol. 18, no. 2, pp. 93–108, 2007.

[7] S. Ramasamy and G. Govindasamy, "A software system reliabilitygrowth model addressing learning," Journal of Applied Statistics,vol. 35, no. 9-10, pp. 1151–1168, 2008.

[8] R. Subburaj, G. Gopal, and P. K. Kapur, "A software system reliabilitygrowth model for estimating debugging and therefore the learningindices," International Jounal of Performability Engineering, vol.8, no. 5, pp. 539–549, 2012.

[9] S. Ramasamy and A. M. J. Muthu Kumaran, "Dynamically weighted combination of fault - based mostly software system ReliabilityGrowth Models," Indian Journal of Science and Technology, vol.9, no. 22, Article ID 93967, 2016.

[10] Y. Srinivas, Y.Vamsidhar,"Software responsibility enhancement By using Effective NHPP Models", International Journal of applied science And Technology, vol 4(4), pp 312-314, 2013.

[11] S. Yamada, J. Hishitani, and S. Osaki, "Software-reliabilitygrowth with a Weibull test-effort: a model and application,"IEEE Transactions on responsibility, vol. 42, no. 1, pp. 100–105, 1993.

[12] C.-Y. Huang, S.-Y. Kuo, and M. R. Lyu, "An assessment oftesting-effort dependent software system responsibility growth models,"IEEE Transactions on responsibility, vol. 56, no. 2, pp. 198–211, 2007.

[13] P. K. Kapur, D. N. Goswami, A. Bardhan, andO. Singh, "Flexiblesoftware responsibility growth model with testing effort dependentlearning method," Applied Mathematical Modelling, vol. 32, no.7, pp. 1298–1307, 2008.

[14] J. Xu and S. Yao, "Software responsibility growth model withpartial equation for numerous debugging processes,"Mathematical issues in Engineering, Article ID2476584,Art.ID 2476584, 13 pages, 2016.

[15] P. Roy, G. S. Mahapatra, P. Rani, S. K. Pandey, and K. N.Dey, "Robust feed forward and continual neural network baseddynamic weighted combination models for software system reliabilityprediction," Applied Soft Computing, vol. 22, pp. 629–637, 2014.

[16] Y.-S. Su and C.-Y. Huang, "Neural-network-based approachesfor software system responsibility estimation exploitation dynamic weightedcombinationalmodels," Journal of Systems and software system, vol. 80,no. 4, pp. 606–615, 2007. [17] I. Lakshmanan and S. Ramasamy, "An Artificial Neural-Network Approach to software system responsibility Growth Modeling,"in Proceedings of the third*International Conference on Recent Trends in Computing, ICRTC 2015*, pp. 695–702, March 2015.

[18] S. Ramasamy and I. Lakshmanan, "Application of artificial neural network for software reliability growth modeling with testing effort," *Indian Journal of Science and Technology*, vol. 9,no. 29, Article ID 90093, 2016.

[19] S. Yamada, "*Software Reliability Modeling: Fundamentals and Applications*," Springer-Verlag, Tokyo, Japan, 2014.

[20] M. Xie, "*Software Reliability Modeling*, World Scientific," Singapore Asia, 1991.

[21]A.Wood,"Predicting software reliability," *Computer*, vol. 29,no.11, pp. 69–77, 1996.

[22] R. Subburaj,"*Software Reliability Engineering*, McGraw-Hill Professional," New Delhi, India, 2015.

[23] Zainab ALRahamneh, Mohammad Reyalat, Alaa F. Sheta, Sulieman Bani-Ahmad, Saleh Al- Oqeili. "A New Software Reliability Growth Model: Genetic-Programming-Based Approach" , Journal of Software Engineering and Applications, 2011

[24] M. Xie and M. Zhao, "On some reliability growth models withsimple graphical interpretations," *Microelectronics Reliability*,vol. 33, no. 2, pp. 149–167, 1993.

Avinash S ,pursuing PHD in centurion university in the stream of computer science and engineering ,completed M.tech in 2013,certified in software testing,and 6 years of teaching experience

Dr.Y.Srinivas is currently working as a professor in the department of computer science and engineering ,GITAM university.Research interests include data analytics,software engineering,software reliability,data analytics.

Dr.P.Annan naidu is currently working as a assistant professorin the department of computer science and engineering ,centurion university,odisha.Research interests include Big data,machine learning and software engineering