A Methodology for Identification of Failures in Software Development Process

Avinash seekoli, Srinivas.yaramalle

Abstract: ***Software reliability is a branch of software engineering, which in specific deals with the performance of the software. In terms of identification of failures many methodologies that were present in the literature fail to identify the failure rate of a newly developed application. Hence the proposal article aims at understanding the failure rate at an early stage. The methodology proposed based on normal distribution.***

Keywords: software reliability, normal distribution, failure rate, software engineering

I. INTRODUCTION

Software engineering is a specialized branch which most of the methods projected in the area of research, mostly concerns with the identification of failure rate and estimation of failure rate and estimation of failure rate[14]. Most of the methodology present in the literature are mostly coin with the objective of estimation the growth rate[7] and in many practical situations the error rate is estimated based on factors and these models mostly inclined towards the prediction of the failures however software reliability[1] plays a crucial rate in particular during the development of a software. The software that are developed need to undergo several testing methodologies, mostly consider as reduce in the software community. However, most of these works are merely confined in the estimation of the time period during which a failure may occur. During the developer of software, the general practice is to conduct a review, identify the failure, overview the failures and reconduct the review. If the identification of the early failure rate is not assured then the error rate cannot be minimized, and as such needs more no of reviews to overcome this situation, hence it is necessary to develop methodologies that can identify the failures at the earliest stage at thereby no of reviews during the particular time can be minimized. Also, the present-day research work is mostly concerned with the identification of the failure based on supervised cases only. However while developing the novel softwares,these concept may set initial difficulties as most of the cases if, software is developed for a specific application.to overcome this situation, the models that are derived should be changed such that they capable of identify the failure during the development of software projects, which are mostly carried out in a unsupervised manner[14].

Revised Manuscript Received on October 15, 2019
***** Correspondence Author
Avinash Seekoli, Cse Department, Centurion University Technology And Management, Paralakhemundi, India.
Email: avinash.seekoli@cutm.ac.in
Dr. Srinivas.Yarramalle, IT Department, GITAM Institute Of Technology, Vishakapatnam, Country. Email: Srinivas.yarramalle@gitam.edu

The software's are developed in 2 modes in mode-1,software which is developed earlier in particular language is again redeveloped (or) it is mostly a maintained project where new plugins (or) updates will be incorporated.in such cases based on the previous history, the errors can be easily identified however in specific cases i.e while a software company need to developed where such templates are not readily available, identify the failure becomes a challenge task, there is very little work that has been carried out in this area of research work therefore in the present article a methodology is developed for understanding these errors with minimal no of reviews. The main advantage here is that as no of reviews minimize, the cost of the software minimizes with this effort the present article aims at developing a methodology based on gaussian mixture model[5][6]. The advantage beyond the choice is that we can easily estimate the failures basing on the patterns. Therefore, in this methodology the probability will be identified and based on these probabilities[13], the model will be developed and based on least square methodology, the original errors are estimated. The rest of the article is presented as follows.in section 2 the dataset consider is presented, section -3 deals with gaussian mixture model, section 4 the experimentation, section 5 presented. The results derive are evaluated using metrics such as true positive, specificity, sensitivity-measure and false positive the final concluding section -6 summarizes the architect

II. DATA SET

The present data set is about the kc2 NASA data set. The failure rate is to be identified among the failure data. This data contains information about the modules which are actually faire modules and some of which may not be actual failure, but treated as failures. The role of any software developer is to identify these failures beforehand so as to deliver failure free software and release just in-time. With this objective the software failures are to be underlined.in order to achieve this we have used methodologies called enhance least square methodology. The accuracy of the identified rate is measured using specifying, sensitivity and f-measure

Table- I: Original Failures Data (KC2-NASA)

Time (Weeks-n)	Estimated Failures-U_i
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

III. GAUSSIAN MIXTURE MODEL

In this article gaussian mixture model[4] is consider, the main advantage beyond the choice of gaussian mixture model is that in the earlier literature works are presented based on log normal distribution how ever in general statistical variable 'x' is 10-log of the failure n i.e

X=logN----------(1)

N should not consider negative values there fore the lognormal distribution cannot handle these destruction a need to be converted in to normal distribution using the conversion formula

$F(x)=\int_{-\infty}^{x} f(t)\,dt$----------(2)

To solve this equation involves lot of numerical calculation therefore to solve this equation tools like mat lab etc will be consider

In the present article we have consider the gaussian mixture model given by the formula

$F(x)= {}^{1}/_{\sqrt{2\pi\sigma}}\, e^{\frac{-(x-\mu)^2}{\sigma}}$ ----------(3)

Any failure data generally will be having the ranges from -∞ to ∞ therefore to manage this vivid nature of errors, Gaussian mixture model will be well suitated.And hence gaussian mixture model is consider. Another advantage of considering gaussian mixture model is that all the other distribution are the particular cases of gaussian mixture model. When we consider the multivariate gaussian model therefore to have better approximation of the error we have consider gaussian mixture model

IV. EXPERMENTATION

In this article, the gaussian mixture model is consider each of the failure data is given as input to the pdf of gaussian mixture model. From this pdf values ,the values which are mostly high are consider to be the failures there fore the estimates of the failures are modeled and these values are corelated with that of the values obtained based on least square approximation where basing on the least square, a value is consider to be zero and the value above it are assumed to be negative below it assumed to positive and the values are estimated[8] for fitting of a line using the equation

$a\sum y=na+b\sum x$----------(4)

$b\sum xy=a\sum_1^1 x+b\sum_1^1 x^2$----------(5)

By using the values of above table the values of A=2.123,B=0.0156 By finding the Ati-log to the values of A and B,the values of a and b will be calculated.The values of a=132.73 and b=1.036.by using these two values,the actual failures can be computed by using the log normal distribution method and he values of actual failures are depicted in the table

a=132.7210234

b=1.042046719

Table-II-Computing A and B values for KC2 Data set

SI.NO	Estimated Failures(U_i)	Vv_i=Log(U_i)	Time (t_i)	$v_i t_i$	t_i^2
1	75	1.875061263	-15	-28.1259	255
2	81	1.908485019	-14	-26.7188	196
3	86	1.934498451	-13	-25.1485	169
4	90	1.954242509	-12	-23.4509	144
5	93	1.968482949	-11	-21.6533	121
6	96	1.982271233	-10	-19.8227	100
7	98	1.991226076	-9	-17.921	81
8	99	1.995635195	-8	-15.9651	64
9	100	2	-7	-14	49
10	100	2	-6	-12	36
11	100	2	-5	-10	25
12	115	2.06069784	-4	-8.24279	16
13	120	2.079181246	-3	-6.23754	9
14	123	2.089905111	-2	-4.17981	4
15	130	2.113943352	-1	-2.11394	1
16	135	2.130333768	0	0	0
17	139	2.1430148	1	2.143015	1
18	143	2.155336037	2	4.310672	4
19	150	2.176091259	3	6.528274	9
20	155	2.190331698	4	8.761327	16
21	160	2.204119983	5	11.0206	25
22	163	2.212187604	6	13.27313	36
23	170	2.230448921	7	15.61314	49
24	176	2.245512668	8	17.9641	64
25	179	2.252853031	9	20.27568	81
26	182	2.260071388	10	22.60071	100
27	197	2.294466226	11	25.23913	121
28	210	2.322219295	12	27.86663	144
29	219	2.340444115	13	30.42577	169
30	221	2.344392274	14	32.82149	196
N=30	$\sum U_i$ =4105	$\sum v_i$ = 63.45545331	$\sum t_i$ = -15	$\sum v_i t_i$ = 3.263344	$\sum t_i^2$ =2255

Table-III-Actual Failures for case study

SI.NO	Time t_i	Actual failures $A_i = a + bt_i$	t_i^2	$(A_i)\,t_i$
1	-15	117	255	-1755
2	-14	118	196	-1652
3	-13	119	169	-1547
4	-12	120	144	-1440
5	-11	121	121	-1331
6	-10	122	100	-1220
7	-9	123	81	-1107
8	-8	124	64	-992
9	-7	125	49	-875
10	-6	127	36	-762
11	-5	128	25	-640
12	-4	129	16	-516
13	-3	130	9	-390
14	-2	131	4	-262
15	-1	132	1	-132
16	0	133	0	0
17	1	134	1	134
18	2	135	4	270
19	3	136	9	408
20	4	137	16	548
21	5	138	25	690
22	6	139	36	834
23	7	140	49	980
24	8	141	64	1128
25	9	142	81	1278
26	10	143	100	1430
27	11	144	121	1584
28	12	145	144	1740
29	13	146	169	1898
30	14	147	196	2058
N=30	$\sum t_i = -$	$\sum A_i = 3966$	$\sum t_i^2 = 2255$	$\sum A_i\,t_i = 1755$

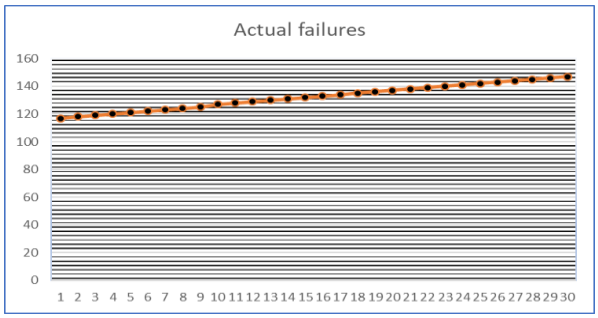

Fig. 1:Graph plotted for Actual failures of Case study one

V.performance evaluation of the proposed model using exponential normal distribution method

In order to evaluate the output we have used the evaluation metrics based on parameters such as true positive,false negative ,sensitivity,specificity and f measre.

The formulas for calculating each of the values is given below

True positive =TP+TP/FN

False postive=FP/FP+TN

Specifity=TN/TN+FP

Sensitivity=TP/TP+FN

F-measure=2*TP/(2*TP)+FP+FN

Based on the above metricsthe values derived are presented below

Table-III:Result analysis table

TP	FP	Specifity	Sensitivity	F-measure
0.830	0.235	0.817	0.765	0.754
0.765	0.116	0.743	0.711	0.811

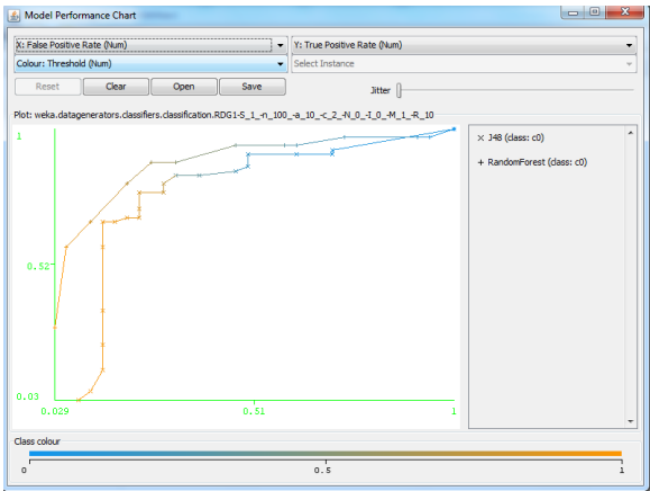

Fig. 2: ROC curve to test the accuracy of the model

VI. CONCLUSION

In this article a novel methodology is presented to highlight the failure rate of the software which the novel in nature.methodology proposed helps toidentify the errors at the most effective manner.

REFERENCES

1. Harminder pal singh dhami(2016),improving software reliability productivity and quality using software metrics,international journal of computer science & technology vol7 issue:4(1),issn 0976-849,issn 2229-4333
2. Jukic(d)(2011),"total least squares fitting bass diffusion model",mathematical and computer modelling.53(2011),pp.1756-1770,doi:10.1016/j.mcm. 2010.12.054.
3. Deva preetha,subbaraj rama Sammy,(2018),"study on software reliability combination models",jour of adv research in dynamical & control systems,vol.10,07-special issue,
4. Mohand said allili,Nizar Bouguila.(2007),"finite generalized gaussian mixture modeling and applications to image or video fore ground segmentation",conference on computer and robot vision(CRV'07),IEEE
5. Y.Baraud,c.giraud and s.huet (2009),Gaussian model selection with an unknown vairaince the annals of statistics,37(2):630-672
6. Jin,y.Zhang et al(,2016),local maxima in the likelihood of gaussain mixture models:structural results and algorithmic consequences in advances in nueral information processing systems 29
7. R.Subburaj,and A.M.J.Muthukumaran (2016),"Dynamically weighted combination of fault based software reliability growth models",Indian journal of science and technology,vol 9
8. Taehyoun kim,kwangkyu Lee,and jongmoonBaik(2015),"An effective approach to estimating thew parameters of software reliability growth models using a real-valued generic algorithm",journal of systems and software ,vol 102,pp.154-144,
9. K.Sharma,R.Garg,C.K.Nagpal,(2010),"selection of optimal software reliability growth model using a distance based approach",IEEETransaction on reliability,vol 59,pp.266-276
10. Gayathri G,Selvi RT(2015).Classification of software reliability to improve the reliability of software,Indian journal of science technology. 8(29):1-5.DOI:10.17485/ijst/2015/v8i29/85287
11. Subburaj r et al(2007),A software reliability growth model for vital quality metrics, south African journal of industrial engineering 18(2),pp 93-108

12. M.R.Lyu and A.Nikora,(1992)"Applying reliability models more effectively,IEEE Software vol.9,pp.43-52
13. B.Klebhanov et al(2012),on a class of distributions stable under random summation,"journal of applied probability,vol 49,pp-303-318
14. Jie zhang et al(2016),Nhpp based software reliability model consedering testing effort and multivariate fault detection rate,journal of systems engineering and electronics vol 27,pp 260-270
15. Q.Y.LI et.al(2011),software reliability growth model with s-shaped testing effort function,journal of Beijing university of aeronautics and astronautics,37(2):149-155

AUTHORS PROFILE

Avinash Seekoli M.Tech computer science technology, research work on software reliability.

R. Srinivas. Yarramalle, Ph. D,imageprocessing, member in ijigsp editorial board , Best researcher awarded by GITAM Deemed to be university