International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075, Volume-8 Issue-10, August 2019

An Empirical Software Reliability Growth Model for Identification of True Failures

Jagadeesh Medapati, Anand Chandulal Jasti, Ranajikanth TV

Abstract*: *Software Reliability is a special topic of software engineering that deals with the finding of glitches during the software development. Effective analysis of the reliability helps to understand the quality of the software. It also helps to reveal the number of failures occurred in development phase which facilitates refinement of the failures in the developed software's. If the failures are not minimized the number of reviews in the software development process increases which in turn increase the expenditure to develop the software. Every software organization aims at releasing the software in time and also it becomes a mandate to manage the software such that the time to release the software is optimized. It becomes a mandate for any organization to release software patches so as to minimize the errors after software release and thereby if the number of patches increases, the credibility of the software together with the storage area will be at stake. This article presents a novel case study wherein a procedural layout is presented such that the number of failures can be reduced instantaneously and the failures are identified at the early stage. The development procedure laid in this article helps to formulate a basis for the distinction between true failures and non-failures. The work is presented using benchmark datasets and the results showcase a better recognition rate and failure deduction rate.

Index Terms*: *Gamma Distribution, Quality Metrics, Software Engineering, Software Reliability

I. INTRODUCTION

Software reliability growth modeling is aimed at using mathematical tools to analyze the failures obtained during the software designing process. This process helps to assess the reliability of software grounded on the developed model and where it takes the generated failure into account and formulates a basis for the identification of the reliability process. These studies help to emphasize the current methodology of the software, describing the Mean Time To Failure (MTTF), Mean Absolute Error (MAE) and understand the Mean Square Error (MSE). Nevertheless, no serious efforts were made to initiate software failures during the initial development phase to analyze the system together with a process by reducing the failures so that failure-free software can be released just in time. As the number of failures increases, the current study develops different strategies, models and presented various views with the only objective to identify the software failures and to develop the strategies to refine the failures, which are called as review procedures in software firms with a sole purpose to reduce the software failures. If the number of failures increases, then the number of reviews to minimize failures increase substantially is making it difficult for the software to release just-in-time [27]. In the traditional methods of calculating the reliability, the programmers are solely focusing on the failures generated. However, there is no serious effort made in analyzing the failure generated due to some of the internal errors such as network fault, data transmission failure, and other faults at the end output may be tincted as a failure. Overlooking this basic process of analyzing a true failure and an accidental failure, the present traditional systems are evaluating the efficiency of the developed software. In this article the authors try to full fill the gaps and to meet the above two objectives viz., discrimination of true failures and actual failures and identification of the software failures where previous records are not available. This study also suggests an approach wherein the failure rate can be minimized and the true failure is thereby reflected. This approach is completely established on the derived mathematical model based on Exponential Logarithmic Normal Distribution (ELND). The article is categorized as follows: Section 2, Background of the study discusses in depth about the various researches carried out in the field of software reliability. Section 3 of the article presents an overview of the ELND approach and its necessity. Section 4 covers the datasets considered for the study. Section 5 illustrates the methodology of the article, section 6 deals with the experimentation and the results derived thereof. In section 7 of the article, various performance metrics were considered in order to analyze the efficiency of the developed model. In the concluding section 8, the results obtained were discussed.

II. BACKGROUND RESEARCH

To drive the developed software's towards perfection, all software companies try to adopt the policies of software reliability life cycles to develop reliable software. The testing process of software cleared for implementation is generally called as the review. During these reviews, the probability of the failures can be notified. If the failure probability is high, steps are to be initiated to substantially bring down the failure rate considerably before releasing the software in to the market.

Revised Manuscript Received on August 02, 2019.

Jagadeesh Medapati, Department of Computer Science and Engineering, GITAM University, Visakhapatnam, (Andhra Pradesh), India, E-mail: jagadeesh.medapati@gmail.com

Anand Chandulal Jasti, Department of Computer Science and Engineering, GITAM University, Visakhapatnam, (Andhra Pradesh), India, E-mail: dr.chandulal@yahoo.com

Ranajikanth TV, Department of Computer Science and Engineering, Srinidhi Institute of Technology, Hyderabad, (Telangana), India, E-mail: rajini.tv@gmail.com

Many models are demonstrated in the literature by taking this issue and formulating the objectives like developing user-friendly, fully functional software, enhanced capability and ensured maintainability. With these objectives, the software development should be done to prepare failure-free software meeting the user's requirement. Previously, many models are showcased in the literature that fulfills the objectives of the user requirements. Some of the predominant models in this area of research are coined initially from Hudson [1], by proposing the initial study of software reliability and have published and presented a good number of articles to benefit the potential researchers working in this domain. The research in software testing is further taken into consideration by authors like Jelinski & Moranda model [2], Shooman model [3] and the Littlewood & Verrall model [4]. Jelinski & Moranda model [2] has presented in his article that the errors if at all exeunt are fixed and failure intensity is proportional to the quantity of remaining failures. Shooman model [3] has presented a pictorial view of the failure rates and has thrown an insight to identify that the failure rate may decay during different time intervals. Bayesian method of approach is followed by Littlewood & Verrall model [4] in which the authors have presented a derivation for estimating the effect of failures on the software cost. Yamada and Osaki [5] have executed the works to the next level. According to the authors every failure rate can be shown as a two class discrete time model, where the first model represents the error detection process and the second model is used for estimating the future error. The next level of research work in this direction was initiated by Thoma [6], Brown [7]. In a study carried out by Thoma and Brown estimation of the failure rates was based on measures of dispersions and are restricted to the central limit theorem. Authors have also developed models based on hyper geometric distribution to derive a model which can determine the optimal number of failures from a developed software product. Ohishi et al. [8] have proposed a new method namely Gomptez distribution for estimating the software reliability, this methodology is proven to be a most substantive method for estimation of the failures so far. Research is also carried out not only using the Non-Homogeneous Poisson Processes but other distribution methods like a family of Pareto distribution R. Satya Prasad [9] [10], R.L. Kantham et.al, [11], where the authors have develpoed new methods to estimate the failure rates and identify the mean time to failure. Latest studies also include works based on Raleigh distribution, Generalized Laplacian distribution, Weibull distribution, and Gaussian distribution. These models are also limited to study of reliability basing on the error rates. Despite of tremendous research in this area, most of the studies demonstrated by the previous authors are confined to the study of the impact of failure rate and few studies tried to project the time between the failures. Efforts to reduce the error rate or to discriminate the true error from the actual error were not found in the literature. This article is aimed to fulfil this objective in a novel approach.

III. EXPONENTIAL LOGARITHMIC NORMAL DISTRIBUTION

Understanding the pattern of the failures is essential to estimate the failures. This analysis of the pattern helps to calculate the true failures and the possible non-failures. For this purpose, numerous models have been discussed in the literature by Hudson [1], Jelinski and Moranda [2], Littlewood and Verrall [4], Pham [12], Michael [13], John [14], Shooman (1972), Goel and Okumoto [15], Ohba [18], [19], Kapur et.al., [20], [21], Kuo et.al., [22], Khan et.al., [23], Sobhana et.al., [24], Sultan [25], Somasundaram and Chinnaiyan [26]. However, these models failed to attribute the analysis of the true failure as it is evident that every initial data in the failure data model assumes exponential distribution. Hence, in this article we have considered Exponential Logarithmic Normal Distribution [27]. The Probability Density Function (PDF) for fitting the ELND is given by.

$$\mathrm{f}(p,q) = q(e^{-px}) \text{ If } x > 0; \qquad (1)$$

$$= 0 \text{ otherwise.}$$

Where 'x' represents the failure.

Here the values of p and q are calculated using the lease square method and by using the formulae:

$$\sum \mu_i = np + q\sum t_i \quad \text{and} \qquad (2)$$

$$\sum u_i t_i = p\sum t_i + q\sum t_i^{\,2} \qquad (3)$$

IV. DATASETS CONSIDERED

In order to present the proposed methodology, we have considered two datasets namely, TANDEM presented in Woods (1996) and BROOKS AND MOTELY presented in Brooks and Motley (1980) for highlighting the proposed model. The first dataset of TANDEM consists of failure data executed in four releases, Release 1 to Release 4. Each release consists of failures generated. The second dataset containing a failure data set considered for the experimentation is BROOKS AND MOTELY. The datasets used in the proposed model are given below:

Table I: Original Failures in TANDEM Dataset.

TW	*Release 1*		*Release 2*		*Release 3*		*Release 4*	
	EH	**ND**	**EH**	**ND**	**EH**	**ND**	**EH**	**ND**
1	519	16	384	13	162	6	254	1
2	968	24	1186	18	499	9	788	3
3	1430	27	1471	26	715	13	1054	8
4	1893	33	2236	34	1137	20	1393	9
5	2490	41	2772	40	1799	28	2216	11
6	3058	49	2967	48	2438	40	2880	16
7	3625	54	3812	61	2818	48	3593	19
8	4422	58	4880	75	3574	54	4281	25
9	5218	69	6104	84	4234	57	5180	27
10	5823	75	6634	89	4680	59	6003	29
11	6539	81	7229	95	4955	60	7621	32
12	7083	86	8072	100	5053	61	8783	32
13	7487	90	8484	104	9604	36		
14	7846	93	8847	110	10064	38		
15	8205	96	9253	112	10560	39		
16	8564	98	9712	114	11008	39		
17	8923	99	10083	117	11237	41		
18	9282	100	10174	118	11243	42		
19	9641	100	10272	120	11305	42		
20	10000	100						

Labels in the Table I TW constitutes the Test Weeks, EH constitutes the Execution Hours and ND constitutes the Number of defects. Labels in the Table II TW constitutes the Test Weeks, EH constitutes the Execution Hours and AD constitutes the Number of defects.

Table II. Original Failures in BROOKS AND MOTELY Dataset.

W	EH	AD	W	EH	AD	W	EH	AD
1	7.25	7	13	417.94	479	25	1194.68	1166
2	10.42	29	14	462.69	559	26	1260.01	1184
3	17.5	61	15	505.02	624	27	1327.84	1221
4	24.83	108	16	580.02	681	28	1444.76	1236
5	32.08	134	17	642.85	771	29	1532.84	1244
6	44.66	159	18	716.43	831	30	1610.92	1272
7	64.58	175	19	759.18	888	31	1648.84	1278
8	117.08	223	20	799.85	978	32	1689.92	1283
9	164.26	259	21	896.6	1024	33	1744.42	1286
10	259.36	312	22	985.18	1081	34	1807.42	1289
11	315.11	369	23	1041.93	1110	35	1846.92	1301
12	374.36	408	24	1121.18	1150			

V. METHODOLOGY

The datasets for the experimentation of the proposed model are presented in the above section. Every dataset is considered for the study and for each dataset initial estimates of the proposed Exponential Logarithmic Normal Distribution parameters p and q are calculated. The values obtained using the Least Square Estimation are presented in Table III:

Table III: Estimated values of parameters p and q for the datasets considered.

Datasets Considered	*p*	*q*
Tandem Release 1	135.845	0.078
Tandem Release 2	179.573	0.063
Tandem Release 3	49.339	0.237
Tandem Release 4	605.941	0.005
Brooks and Motely	11981.548	0.004

With these values analysis of the proposed model is carried out. Here the first dataset considered TANDEM containing four releases 1 to 4 is presented along with the second failure dataset BROOKS AND MOTELY in the above Tables I and II. Against each of the dataset, the analysis is conducted in a phased manner wherein the first phase- the true failures are calculated and the experimentation is carried out to reduce the failure rate. Against each of the data released, the number of the actual defects highlighted is considered and using these defects the actual failures are predicted and are presented as below: Labels in Table IV to Table VIII, TW constitutes the Test Weeks, ND constitutes the Number of Defects, PD represents Predicted Defect, RES constitutes the Residual and Fault classifies whether the failure is a true failure or not. In this study, the identified residuals where the actual notified errors are subtracted from the predicted errors and the process carried out on the two datasets namely TANDEM and BROOKS AND MOTELY are tabulated in Table IV to Table XVI. The Fault column in every table indicates the outcome of the proposed model on the datasets and it clearly specifies how best the proposed model has identified the true failures and in turn reduce the failure rate when compared to the original dataset.

Table IV: Actual Failures for TANDEM Dataset Release 1; KURTOSIS =-1.225, STD DEV = 29.2529, μ = 69.45 and λ = 100.

Test Week	*Execution Hrs.*	*No. of Defects*	*Predicted Error (PE)*	*PDF*	$\frac{PE - PDF}{PE}$
1	519	16	706.05	2.08E+14	8.676E+22
2	968	24	722.05	2.169E+14	9.021E+22
3	1,430	27	728.05	2.203E+14	9.154E+22
4	1,893	33	740.05	2.272E+14	9.427E+22
5	2,490	41	756.05	2.367E+14	9.804E+22
6	3,058	49	772.05	2.465E+14	1.02E+23
7	3,625	54	782.05	2.528E+14	1.045E+23
8	4,422	58	790.05	2.579E+14	1.066E+23
9	5,218	69	812.05	2.724E+14	1.125E+23
10	5,823	75	824.05	2.805E+14	1.159E+23
11	6,539	81	836.05	2.888E+14	1.193E+23
12	7,083	86	846.05	2.959E+14	1.223E+23
13	7,487	90	854.05	3.016E+14	1.247E+23
14	7,846	93	860.05	3.06E+14	1.266E+23
15	8,205	96	866.05	3.104E+14	1.285E+23
16	8,564	98	870.05	3.134E+14	1.297E+23
17	8,923	99	872.05	3.149E+14	1.304E+23
18	9,282	100	874.05	3.164E+14	1.31E+23
19	9,641	100	874.05	3.164E+14	1.31E+23
20	10,000	100	874.05	3.164E+14	1.31E+23

Table V: Actual Failures for the TANDEM Dataset Release 2; KURTOSIS =-1.280, STD DEV = 37.3, μ = 77.79 and λ = 120.

Test Week	*Execution Hrs.*	*No. of Defects*	*Predicted Error (PE)*	*PDF*	$\frac{PE - PDF}{PE}$
1	384	13	1190.061	1.55E+15	1.70E+24
2	1,186	18	1200.061	1.57E+15	1.72E+24
3	1,471	26	1216.061	1.61E+15	1.76E+24
4	2,236	34	1232.061	1.66E+15	1.81E+24
5	2,772	40	1244.061	1.69E+15	1.84E+24
6	2,967	48	1260.061	1.73E+15	1.88E+24
7	3,812	61	1286.061	1.80E+15	1.96E+24
8	4,880	75	1314.061	1.88E+15	2.05E+24
9	6,104	84	1332.061	1.93E+15	2.10E+24
10	6,634	89	1342.061	1.96E+15	2.13E+24
11	7,229	95	1354.061	2.00E+15	2.17E+24
12	8,072	100	1364.061	2.03E+15	2.21E+24
13	8,484	104	1372.061	2.05E+15	2.23E+24
14	8,847	110	1384.061	2.09E+15	2.27E+24
15	9,253	112	1388.061	2.10E+15	2.29E+24
16	9,712	114	1392.061	2.11E+15	2.30E+24
17	10,083	117	1398.061	2.13E+15	2.32E+24
18	10,174	118	1400.061	2.14E+15	2.33E+24
19	10,272	120	1404.061	2.15E+15	2.34E+24

Table VI: Actual Failures for the TANDEM Dataset Release 3; KURTOSIS =-0.459, STD DEV = 16.883, μ = 38.526 and λ = 42.

Test Week	*Execution Hrs.*	*No. of Defects*	*Predicted Error (PE)*	*PDF*	$\frac{PE - PDF}{PE}$
1	162	6	-51.048	2.94886E+12	3.33699E+21
2	499	9	-47.337	2.21329E+12	2.18616E+21
3	715	13	-42.389	1.45757E+12	1.18239E+21
4	1,137	20	-33.730	6.19326E+11	3.37142E+20
5	1,799	28	-23.834	1.73314E+11	5.28792E+19
6	2,438	40	-8.990	6358282047	5.00248E+17
7	2,818	48	0.906	2686956.251	8.79047E+12
8	3,574	54	8.328	394506986.8	2.24388E+15
9	4,234	57	12.039	2879255046	5.71953E+16
10	4,680	59	14.513	7281596606	2.51723E+17
11	4,955	60	15.750	10797670257	4.69986E+17
12	5,053	61	16.987	15455443389	8.27782E+17
13	9,604	36	-13.938	26462530861	3.60478E+18
14	10,064	38	-11.464	13817749287	1.45285E+18
15	10,560	39	-10.227	9551078809	8.72229E+17
16	11,008	39	-10.227	9551078809	8.72229E+17
17	11,237	41	-7.753	4041980208	2.71818E+17
18	11,243	42	-6.516	2424855358	1.38498E+17
19	11,305	42	-6.516	2424855358	1.38498E+17

Table VII: Actual Failures for the TANDEM Dataset Release 4; KURTOSIS =-1.556, STD DEV = 11.243, μ = 17.666 and λ = 32.

Test Week	*Execution Hrs.*	*No. of Defects*	*Predicted Error (PE)*	*PDF*	$\frac{PE - PDF}{PE}$
1	254	1	-605.024	2.68E+25	1.95983E+45
2	788	3	-603.014	2.64E+25	1.92134E+45
3	1,054	8	-597.989	2.56E+25	1.82785E+45
4	1,393	9	-596.984	2.54E+25	1.80961E+45
5	2,216	11	-594.974	2.51E+25	1.7736E+45
6	2,880	16	-589.949	2.42E+25	1.68616E+45
7	3,593	19	-586.934	2.37E+25	1.63545E+45
8	4,281	25	-580.904	2.28E+25	1.53783E+45
9	5,180	27	-578.894	2.25E+25	1.50638E+45
10	6,003	29	-576.884	2.22E+25	1.47548E+45
11	7,621	32	-573.869	2.17E+25	1.43012E+45
12	8,783	32	-573.869	2.17E+25	1.43012E+45

Table VIII: Actual Failures for the BROOKS AND MOTLEY Dataset. KURTOSIS =-1.577, STD DEV = 468.221, μ = 748 and λ = 1301.

Test Week	*Execution Hrs.*	*No. of Defects*	*Predicted Error (PE)*	*PDF*	$\frac{PE - PDF}{PE}$
1	7.25	7	-11978	1.25E+31	1.1E+54
2	10.42	29	-11955	1.25E+31	1.09E+54
3	17.5	61	-11923	1.23E+31	1.07E+54
4	24.83	108	-11876	1.21E+31	1.04E+54
5	32.08	134	-11850	1.20E+31	1.03E+54
6	44.66	159	-11825	1.19E+31	1.02E+54
7	64.58	175	-11809	1.19E+31	1.01E+54
8	117.08	223	-11761	1.17E+31	9.84E+53
9	164.26	259	-11725	1.15E+31	9.66E+53
10	259.36	312	-11671	1.13E+31	9.4E+53
11	315.11	369	-11614	1.11E+31	9.13E+53
12	374.36	408	-11575	1.10E+31	8.95E+53
13	417.94	479	-11504	1.07E+31	8.63E+53
14	462.69	559	-11423	1.04E+31	8.27E+53
15	505.02	624	-11358	1.02E+31	8E+53
16	580.02	681	-11301	9.95E+30	7.76E+53
17	642.85	771	-11210	9.64E+30	7.39E+53
18	716.43	831	-11150	9.44E+30	7.16E+53
19	759.18	888	-11093	9.24E+30	6.94E+53
20	799.85	978	-11003	8.95E+30	6.61E+53
21	896.6	1024	-10956	8.80E+30	6.45E+53
22	985.18	1081	-10899	8.62E+30	6.25E+53
23	1041.93	1110	-10870	8.53E+30	6.15E+53

24	1121.18	1150	-10830	8.40E+30	6.02E+53
25	1194.68	1166	-10814	8.35E+30	5.96E+53
26	1260.01	1184	-10796	8.30E+30	5.91E+53
27	1327.84	1221	-10759	8.18E+30	5.79E+53
28	1444.76	1236	-10744	8.14E+30	5.74E+53
29	1532.84	1244	-10736	8.11E+30	5.71E+53
30	1610.92	1272	-10707	8.03E+30	5.62E+53
31	1648.84	1278	-10701	8.01E+30	5.6E+53
32	1689.92	1283	-10696	8.00E+30	5.59E+53
33	1744.42	1286	-10693	7.99E+30	5.58E+53
34	1807.42	1289	-10690	7.98E+30	5.57E+53
35	1846.92	1301	-10678	7.94E+30	5.53E+53

Table IX: Goodness of fit statistics for the TANDEM Datasets and BROOKS AND MOTELY Dataset.

Datasets Considered	*Statistic*	*Independent*	*Full*
TANDEM R1	-2 Log(Likelihood)	42.04619255	-1.270866512
	AIC	46.04619255	4.729133488
	SBC	48.0376571	7.716330309
	AICC	46.75207491	6.229133488
TANDEM R2	-2 Log(Likelihood)	43.80049811	3.261328331
	AIC	47.80049811	9.261328331
	SBC	49.68937607	12.09464527
	AICC	48.55049811	10.86132833
TANDEM R3	-2 Log(Likelihood)	39.65728411	-2.536318299
	AIC	43.65728411	3.463681701
	SBC	45.54616206	6.296998639
	AICC	44.40728411	5.063681701
TANDEM R4	-2 Log(Likelihood)	34.03443602	21.5696264
	AIC	38.03443602	27.5696264
	SBC	39.00424932	29.02434635
	AICC	39.36776935	30.5696264
BROOKS AND MOTELY	-2 Log(Likelihood)	101.0392227	75.08236094
	AIC	105.0392227	81.08236094
	SBC	108.1499189	85.74840513
	AICC	105.4142227	81.85655449

Table X: Test of the null hypothesis H0: beta=0 for TANDEM Datasets and BROOKS AND MOTELY Dataset.

Datasets Considered	*Statistic*	*DF*	*Chi-square*
TANDEM R1	-2 Log(Likelihood)	1	43.31705906
	Score	1	10782.16663
	Wald	1	315.2263742
TANDEM R2	-2 Log(Likelihood)	1	40.53916978
	Score	1	6753.258705
	Wald	1	210.4971408
TANDEM R3	-2 Log(Likelihood)	1	42.1936024
	Score	1	8298.533324
	Wald	1	596.7414912
TANDEM R4	-2 Log(Likelihood)	1	12.46480962
	Score	1	450.2293414
	Wald	1	29.1319165
BROOKS AND MOTELY	-2 Log(Likelihood)	1	25.95686181
	Score	1	12010.51314
	Wald	1	48.3723997

Table XI: Regression coefficients for TANDEM Datasets and BROOKS AND MOTELY Dataset.

Datasets Considered	*Variable*	*Value*	*Standard Error*	*Wald Chi-Square*
TANDEM R1	Intercept	3.171	0.058	3015.252
	Test Week	0.112	0.007	274.739
	Scale	0.179	0.051	12.460
TANDEM R2	***Intercept***	***3.035***	***-Inf***	***-Inf***
	Test Week	0.145	-Inf	-Inf
	Scale	0.262	-Inf	-Inf
TANDEM R3	Intercept	1.802	0.145	153.863
	Test Week	0.264	0.023	127.997
	Scale	0.213	0.047	20.176
TANDEM R4	Intercept	0.785	0.329	5.689
	Test Week	0.268	0.050	29.132
	Scale	1.044	-Inf	-Inf
BROOKS AND MOTELY	Intercept	4.324	0.277	244.392
	Test Week	0.104	0.015	48.372
	Scale	1.479	-Inf	-Inf

Table XII: Predictions and Residuals for the TANDEM Dataset Release 1.

No. of Defects	*Residuals*	*Cox-Snell Residuals*	*Cumulative Distributions*	*Hazard Function*	*Survival Distribution Function*	*Failure (YES/NO)*
16	-0.511	0.600	0.451	0.329	0.549	Y
24	-0.218	0.804	0.553	0.360	0.447	Y
27	-0.213	0.809	0.554	0.360	0.446	Y
33	-0.124	0.883	0.587	0.365	0.413	Y
41	-0.019	0.981	0.625	0.368	0.375	Y
49	0.047	1.048	0.649	0.367	0.351	N
54	0.031	1.032	0.644	0.368	0.356	N
58	-0.009	0.991	0.629	0.368	0.371	Y
69	0.052	1.053	0.651	0.367	0.349	N
75	0.023	1.023	0.641	0.368	0.359	N
81	-0.012	0.988	0.628	0.368	0.372	Y
86	-0.065	0.937	0.608	0.367	0.392	Y
90	-0.132	0.877	0.584	0.365	0.416	Y

93	-0.211	0.810	0.555	0.360	0.445	Y
96	-0.292	0.747	0.526	0.354	0.474	Y
98	-0.383	0.682	0.494	0.345	0.506	Y
99	-0.485	0.615	0.460	0.333	0.540	Y
100	-0.588	0.556	0.426	0.319	0.574	Y
100	-0.700	0.497	0.391	0.302	0.609	Y

Table XIII: Predictions and Residuals for the TANDEM Dataset Release 2.

No. of Defects	*Residuals*	*Cox-Snell Residuals*	*Cumulative Distributions*	*Hazard Function*	*Survival Distribution Function*	*Failure (YES/NO)*
13	-0.614	0.541	0.418	0.315	0.582	Y
18	-0.434	0.648	0.477	0.339	0.523	Y
26	-0.211	0.810	0.555	0.360	0.445	Y
34	-0.088	0.916	0.600	0.367	0.400	Y
40	-0.070	0.932	0.606	0.367	0.394	Y
48	-0.033	0.968	0.620	0.368	0.380	Y
61	0.062	1.064	0.655	0.367	0.345	N
75	0.124	1.132	0.678	0.365	0.322	N
84	0.092	1.097	0.666	0.366	0.334	N
89	0.005	1.005	0.634	0.368	0.366	N
95	-0.074	0.929	0.605	0.367	0.395	Y
100	-0.168	0.846	0.571	0.363	0.429	Y
104	-0.273	0.761	0.533	0.356	0.467	Y
110	-0.362	0.696	0.502	0.347	0.498	Y
112	-0.489	0.613	0.458	0.332	0.542	Y
114	-0.616	0.540	0.417	0.315	0.583	Y
117	-0.735	0.479	0.381	0.297	0.619	Y
118	-0.871	0.418	0.342	0.275	0.658	Y
120	-0.999	0.368	0.308	0.255	0.692	Y

Table XIV: Predictions and Residuals for the TANDEM Dataset Release 3.

No. of Defects	*Residuals*	*Cox-Snell Residuals*	*Cumulative Distributions*	*Hazard Function*	*Survival Distribution Function*	*Failure (YES/NO)*
6	-0.274	0.760	0.532	0.355	0.468	Y
9	-0.133	0.875	0.583	0.365	0.417	Y
13	-0.030	0.971	0.621	0.368	0.379	Y
20	0.137	1.147	0.682	0.364	0.318	N
28	0.209	1.233	0.708	0.359	0.292	N
36	-1.654	0.191	0.174	0.158	0.826	Y
38	-1.864	0.155	0.144	0.133	0.856	Y
39	-2.103	0.122	0.115	0.108	0.885	Y
39	-2.367	0.094	0.090	0.085	0.910	Y
40	0.301	1.352	0.741	0.350	0.259	N
41	-2.581	0.076	0.073	0.070	0.927	Y
42	-2.821	0.060	0.058	0.056	0.942	Y
42	-3.086	0.046	0.045	0.044	0.955	Y
48	0.219	1.245	0.712	0.358	0.288	N
54	0.073	1.076	0.659	0.367	0.341	N
57	-0.137	0.872	0.582	0.365	0.418	Y
59	-0.367	0.693	0.500	0.347	0.500	Y
60	-0.615	0.541	0.418	0.315	0.582	Y
61	-0.862	0.422	0.344	0.277	0.656	Y

Table XV: Predictions and Residuals for the TANDEM Dataset Release 4.

No. of Defects	*Residuals*	*Cox-Snell Residuals*	*Cumulative Distributions*	*Hazard Function*	*Survival Distribution Function*	*Failure (YES/NO)*
1	-1.053	0.349	0.295	0.246	0.705	Y
3	-0.222	0.801	0.551	0.360	0.449	Y
8	0.491	1.634	0.805	0.319	0.195	N
9	0.341	1.407	0.755	0.345	0.245	N
11	0.274	1.315	0.732	0.353	0.268	N
16	0.381	1.464	0.769	0.339	0.231	N
19	0.285	1.330	0.736	0.352	0.264	N
25	0.292	1.339	0.738	0.351	0.262	N
27	0.101	1.107	0.669	0.366	0.331	N
29	-0.095	0.909	0.597	0.366	0.403	Y
32	-0.264	0.768	0.536	0.356	0.464	Y
32	-0.532	0.587	0.444	0.326	0.556	Y

Table XVI: Predictions and Residuals for Brooks and Motley Dataset.

No. of Defects	*Residuals*	*Cox-Snell Residuals*	*Cumulative Distributions*	*Hazard Function*	*Survival Distribution Function*	*Failure (YES/NO)*
7	-2.482	0.084	0.080	0.077	0.920	Y
29	-1.164	0.312	0.268	0.228	0.732	Y
61	-0.524	0.592	0.447	0.327	0.553	Y
108	-0.057	0.945	0.611	0.367	0.389	Y
134	0.055	1.056	0.652	0.367	0.348	N
159	0.122	1.130	0.677	0.365	0.323	N
175	0.114	1.121	0.674	0.365	0.326	N
223	0.252	1.287	0.724	0.355	0.276	N
259	0.298	1.347	0.740	0.350	0.260	N
312	0.380	1.463	0.768	0.339	0.232	N
369	0.444	1.559	0.790	0.328	0.210	N
408	0.441	1.554	0.789	0.329	0.211	N
479	0.497	1.644	0.807	0.318	0.193	N
559	0.548	1.730	0.823	0.307	0.177	N
624	0.554	1.740	0.825	0.305	0.175	N
681	0.537	1.712	0.819	0.309	0.181	N
771	0.558	1.747	0.826	0.305	0.174	N

831	0.529	1.697	0.817	0.311	0.183	N
888	0.491	1.634	0.805	0.319	0.195	N
978	0.484	1.622	0.803	0.320	0.197	N
1024	0.426	1.531	0.784	0.331	0.216	N
1081	0.376	1.457	0.767	0.339	0.233	N
1110	0.299	1.348	0.740	0.350	0.260	N
1150	0.230	1.259	0.716	0.357	0.284	N
1166	0.140	1.150	0.683	0.364	0.317	N
1184	0.051	1.053	0.651	0.367	0.349	N
1221	-0.022	0.979	0.624	0.368	0.376	Y
1236	-0.113	0.893	0.590	0.366	0.410	Y
1244	-0.211	0.810	0.555	0.360	0.445	Y
1272	-0.293	0.746	0.526	0.354	0.474	Y
1278	-0.392	0.676	0.491	0.344	0.509	Y
1283	-0.492	0.612	0.458	0.332	0.542	Y
1286	-0.593	0.552	0.424	0.318	0.576	Y
1289	-0.695	0.499	0.393	0.303	0.607	Y
1301	-0.790	0.454	0.365	0.288	0.635	Y

VI. PERFORMANCE EVALUATION METRICS

To assess the outputs derived from the proposed model, we have considered the metrics such as Mean Squared Error (MSE), R2, Sum of Squares Error (SSE) and Root Mean Squared Error (RMSE). The formulas for the calculation of the above metrics are appended below:

Mean Squared Error

$$MSE = \frac{\Sigma(|Actual\ Failure_i - Estimated\ Failure_i|)^2}{n-1} \quad (4)$$

Mean Absolute Percent Error

$$MAPE = \frac{\Sigma \frac{|Actual\ Failure_i - Estimated\ Failure_i|}{Actual\ Failure_i} X100}{n} \quad (5)$$

Error of Sum of Squares

$$SSE = \sum_{i=1}^{n}(x_i - \bar{x})^2 \quad (6)$$

Coefficient of Determination

$$R^2 = 1 - \frac{SSE_{res}}{SSE_{tot}} \quad (7)$$

Root Mean Square Error

$$RMSE = \sqrt{\frac{\sum(|Actual\ Failure_i - Estimated\ Failure_i|)^2}{n-1}} \quad (8)$$

The results of the performance evaluation metrics are presented in the following Table XVII.

Table XVII: Comparison of Performance Evaluation Metrics.

Dataset Considered	*MSE*	*R^2*	*SSE*	*RMSE*
Tandem R1	11.317	0.988	192.388	3.364
Tandem R2	26.088	0.982	417.408	5.108
Tandem R3	116.630	0.664	1866.081	10.800
Tandem R4	3.861	0.980	34.749	1.965
Brooks and Motely	9659.621	0.966	309107.871	98.283

Table XVII clearly shows that the MSE is less for the Release 1 and the R^2 is almost coming to 1, which indicates that the model performs better. The SSE and RMSE metrics also showcase significant measures. This demonstrates that the proposed methodology is delivering an outstanding performance in predicting the failures.

VII. EXPERIMENTATION

The experimentation conducted across the two datasets viz., TANDEM and BROOKS AND MOTELY was represented graphically as shown in Figures 1 to Figure 5.

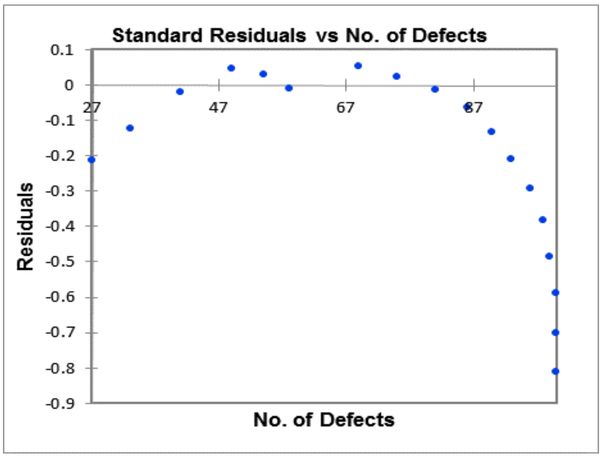

Fig. 1: Standard Residuals versus Number of Defects for the TANDEM Dataset Release 1.

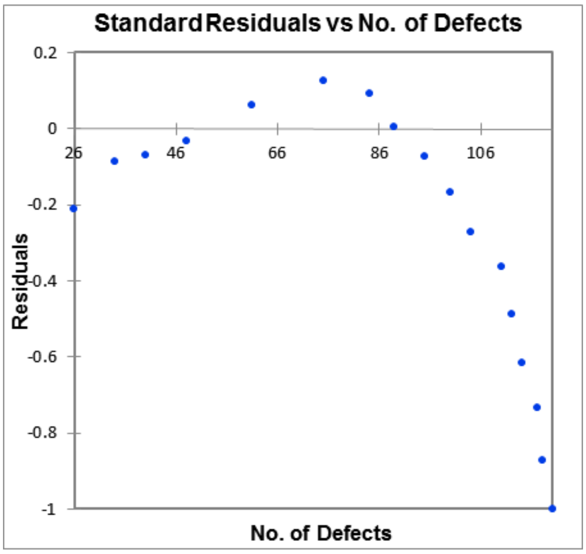

Fig. 2: Standard Residuals versus Number of Defects for the TANDEM Dataset Release 2.

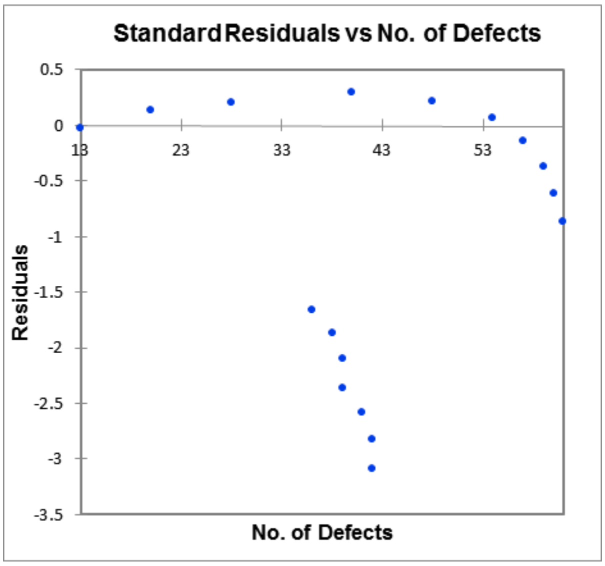

Fig. 3: Standard Residuals versus Number of Defects for the TANDEM Dataset Release 3.

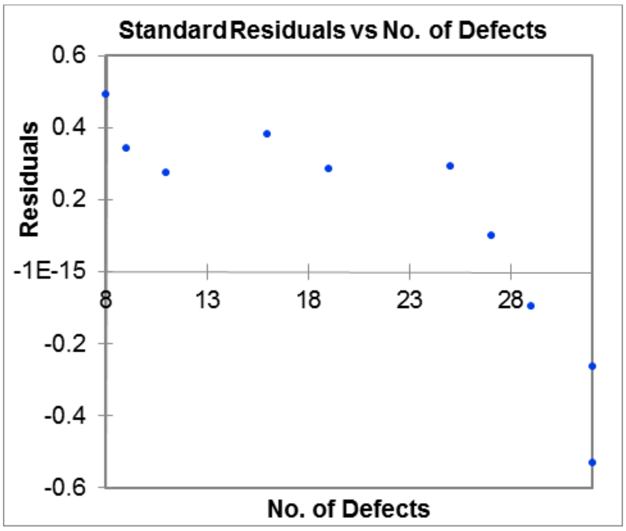

Fig. 4: Standard Residuals versus Number of Defects for the TANDEM Dataset Release 4.

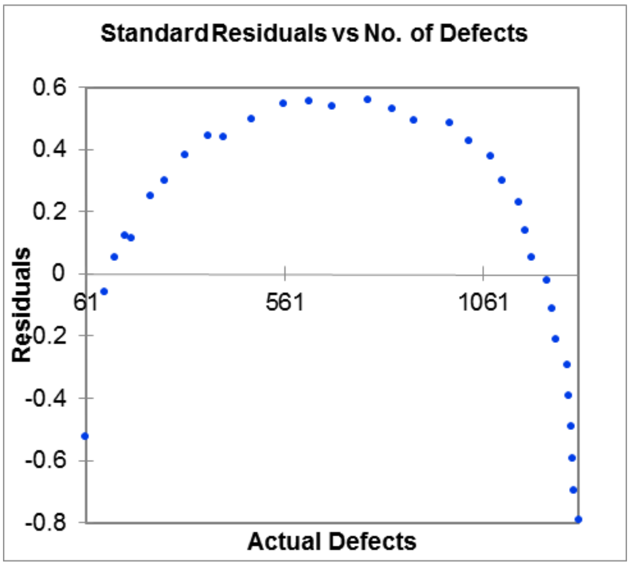

Fig. 5: Standard Residuals versus Number of Defects for the Brooks and Motley Dataset.

VIII. CONCLUSION

In this article a novel methodology has been presented to minimize the number of failures and also for helping the software developers to understand the actual failures that are derived from the project due to technical flaws and also underlined the predicted failures, which are not the failures but reported as failures due to the technical issues or human failures. The results presented in this article on two benchmark datasets helps to understand the potentiality of the model. The results also attribute the significance of the model which can be applied in a software firm to reduce the review times and also to release the software just in time along with increased the profits.

REFERENCES

1. G.R. Hudson, Programming errors as a birth-death process, Technical Report SP-3011, System Development Corp, 1967.
2. Z. Jelinski, P.B. Moranda, "Software reliability research", in *Statistical Computer Performance Evaluation*, 1st edition, W. Freiberger, Ed. New York: Academic Press, 1972, pp. 465-484.
3. M.L. Shooman, "*Statistical Computer Performance Evaluation*" 1st Edition, W. Freiberger, Ed. New York: Academic Press, 1972.
4. B. Littlewood, J.L. Verrall, "A Bayesian reliability growth model for computer software", *Journal of the Royal Statistical Society Series C*, 1973, Vol.22, No.3, pp.332-346.
5. S. Yamada, M. Ohba, "S-shaped software reliability growth models and comparisons", *App. Stock. Mod. Data. Ana.*, 1985, Vol.11, pp.1431-1437.
6. Y. Tohma, R. Jacoby, Y. Murata, M. Yamamoto, "Hyper-geometric distribution model to estimate the number of residual software faults", *Proceedings of the Thirteenth Annual International Computer Software & Applications Conference*, 1989, USA, pp. 610-617.
7. D.B. Brown, "Correction to a cost model for determining the optimal number of software test cases", *IEEE Transactions on Software Engineering*, 1989, Vol.15, No.2, pp.218-221.
8. K. Ohishi, H. Okamura, T. Dohi, "Gompertz software reliability model: estimation algorithm and empirical validation", *Journal of Systems and Software*, 2009, Vol.82, pp.535-543.
9. R. SatyaPrasad, O. Naga Raju, R.R.L. Kantam, "SRGM with imperfect debugging by genetic algorithms", *International Journal of Software Engineering & Applications*, 2010, Vol.1, No.2, pp.66-79.
10. R. Satya Prasad, K.R.H. Rao and R.R.L. Kantam, "Software reliability with SPC", *International Journal of Computer Science and Emerging Technologies*, 2011, Vol.2, No.2, pp.233-237.
11. R.R.L Kantam, M.S. Ravikumar, "Limited failure censored life test sampling plan in Burr type X distribution", *Journal of Modern Applied Statistical Methods*, 2016, Vol.15, No.2, pp.428-454.
12. H. Pham, *Software Reliability*. Springer-Verlag, London, 2000.
13. M.R. Lyu, *Handbook of Software Reliability and System Reliability*. McGraw Hill Inc., Hightstown, NJ, USA, 1996.
14. J.D. Musa, *Software Reliability Engineering: More Reliable Software Faster and Cheaper*. 2nd Edition, AuthorHouse, 2004.
15. A.L. Goel, K. Okumoto, "Time-dependent error detection rate model for software reliability and other performance measures", *IEEE Transactions on Reliability*, 1979, Vol.R.28, No.3, pp.206-211.
16. A. Wood, "Predicting software reliability", *IEEE Computer*, 1996, Vol.29, No.11, pp.69-77.
17. W.D. Brooks, R.W. Motley, *Analysis of discrete software reliability models – technical report*, New York: Rome Air Development Centre, 1980.
18. M. Ohba, "Software reliability analysis models", *IBM Journal of Research and Development*, 1984, Vol.28, No.4, pp.428-443.
19. M. Ohba, "Inflection S-shaped software reliability growth model", *Stochastic Models in Reliability Theory*, 1984, Vol.235, pp.144-162.
20. P.K. Kapur, A. Gupta, P.C. Jha, "On modeling failure phenomenon of multiple releases of a software in operational phase for product and project type software-a theoretical framework",

in *Quality Reliability and Infocom Technology*, P.K. Kapur, A.K. Verma, Eds. MacMillan India Ltd., 2007, pp.155-165.

21. P.K. Kapur, A. Kumar, V.B. Singh, F.M. Nailama, "On modelling software reliability growth model for errors of different severity", *Proceeding of National Conference on Computing for Nation Development*, 2007, pp.279-284.
22. Y. Kuo, C.Y. Hung, M.R. Lyu, “Framework for modeling software reliability, using various testing-efforts and fault-detection rates”, *IEEE Transactions on Reliability*, 2001, Vol.50, Iss.3, pp.310-320.
23. M.G.M. Khan, N. Ahmad, L.S. Rafi, “Modeling and analysis of software reliability with Burr type X testing-effort and release determination”, *Proc of the CASON-2008, IEEE Comp Soc*, 2008, China, pp.759-762.
24. K. Sobhana, R. Satya Prasad, Ch. S. Chowdary “Software reliability growth model on Burr type III-an order statistics approach”, *International Journal of Software Engineering*, 2015, Vol.8, No.1, pp.57-68.
25. S.H. Aljahdali, "Multi-objective optimization for ensemble of models in software reliability prediction", *International Journal of Software Engineering*, 2009, Vol.2 No.1, pp.29-16.
26. S. Somasundaram, R. Chinnaiyan, "Reliability evaluation of object oriented software systems using communication variables – a review", *International Journal of Software Engineering*, 2009, Vol.2, No.2, pp.1-19.
27. Jagadeesh Medapati, Prof Anand Chandulal J and Prof Rajinikanth T V, “A Novel Assessment to Achieve Maximum Efficiency in Optimizing Software Failures” International Journal of Advanced Computer Science and Applications(IJACSA), 10(3), 2019.