ESTIMATION OF ERROR-RATE AFTER DEBUGGING A SOFTWARE

Dr. Meenaksh M. Sagdeo[1]

Abstract: Debugging software is a very important procedure and is related to estimation of software reliability. Whenever some new software is developed, a test procedure is carried out in order to eliminate the bugs in it. The software is run for a fixed duration of time and then debugging takes place. If no error is detected at the end of this period, the corresponding bug cannot be removed. Hence, it becomes important to estimate the error-rate in the revised software package. In this paper the error-rate is estimated assuming that the errors are caused according to Poisson distribution with parameter λ, where, λ is a random variable following Gamma distribution. It is shown that the expected number of bugs causing exactly one error can be used to estimate the expected residual error-rate. Some simulation results using R-package are also given.

Keywords: Compound Poisson distribution, Simulation, Software debugging, Software reliability, Stopping time

1. INTRODUCTION

Whenever some new software is developed, a test procedure is carried out in order to eliminate the bugs in it. The software is run for a fixed duration of time and then debugging takes place. If no error is detected at the end of this period, the corresponding bug cannot be removed. It contributes to the residual error rate. Hence, it becomes important to estimate the error-rate in the revised software package. In [2] Ross studied two problems. One of the problems was to estimate the error rate of the software at a given time t and the other was to develop a stopping rule for determining when to discontinue the testing and declare that the software is ready for use. In [2] he developed a model as estimation and stopping rule procedure. In [3] he developed an estimator of residual error-rate under Poisson distribution with constant parameter. Pavur and Keeling [1] proposed generalized Ross estimator. Unlike the original Ross model, their proposal did not assume that all the discovered bugs are removed. Their goal was to remove bugs that cause more than k errors and leave the bugs that cause fewer than that number of errors. Many other researchers have studied various models for estimating software reliability. In this paper the residual error-rate is estimated assuming that the errors are caused according to Poisson distribution with parameter λ, where, λ is a random variable following Gamma distribution.

i) Notations and Assumptions:

Let m be the total number of bugs in the package and $N_i(t)$ be the number of errors caused by i^{th} bug in time period t, where $t > 0$ and $i = 1, \ldots, m.$,

For further analysis we make the following assumptions:

(1) The i^{th} bug (i =1,…,m) causes errors to occur according to Poisson distribution with parameter λ_i (i =1,…,m)

(2) Each λ_i (i = 1, …, m) is a random variable following Gamma distribution with parameters (α_i, 1) , i =1,…, m.

(3) The number of errors due to i^{th} bug that occur in any t units of operating time, denoted by $N_i(t)$ has Poisson distribution with mean error rate ($\lambda_i t$).

(4) The Poisson processes $\{N_i (t), t \geq 0\}$ for i = 1, 2, …, m caused by m different bugs are independent.

(5) The package is run for t time units and all the resulting errors are noted down. At the end of t units of time, debugging takes place. The bugs corresponding to the errors, which are detected, are removed during debugging. However, if the bug does not cause an error during t time units, it cannot be detected and remains in the revised package.

Let us define an indicator function as follows:
$\psi_i (t) = 1$, if i^{th} bug has not caused an error by time t.
= 0 , otherwise.

Let $\Lambda(t)$ denote the error rate of the revised package.
$\Lambda(t)$ is given by

$$\Lambda(t) = \sum_{i=1}^{m}(\lambda_i \psi_i (t)) \quad (1)$$

We assume that summation is over i, the bug-number varying from 1 to m.

Let $M_j(t)$ be the number of bugs that produced j errors in time t. ($j \geq 1$).

Hence, $M_1(t)$ is the number of bugs producing exactly 1 error, $M_2(t)$ is the number of bugs producing exactly 2 errors and so on.

Let us define another indicator function $I_i(t)$,(i = 1, 2,…, m) as follows:
$I_i(t) = 1$, if i^{th} bug causes 1 error in time t
= 0, otherwise
Then $M_1(t)=\sum_{i=1}^{m}(I_i(t))$

Our aim is to estimate the expected value of the random variable $\Lambda(t)$.
We now prove the main result.

2. THE MAIN RESULT

Theorem:

The expected value of residual error rate in the software package is given by

$$E[\Lambda(t)] =E[\{ \frac{(1+t)}{t} \} M_1(t)]$$

where, $M_1(t)$ is the number of bugs producing exactly 1 error and t is the time duration for testing.

Proof:

The number of errors due to i^{th} bug that occur in any t units of operating time that is $N_i(t)$, has Poisson distribution with mean error rate ($\lambda_i t$). Here, λ_i is not a constant. $\lambda_i \frown G(\alpha_i, 1)$. Hence, $\lambda_i t \frown G(\alpha_i, t)$, where $G(\alpha, \beta)$ denotes gamma distribution with parameters α and β. It should also be noted that the number of errors due to the i^{th} bug that is $N_i(t)$ has a distribution same as that of the number of failures before a total of α_i successes where probability of success is p = (1/ (1+t)). Thus, $N_i(t)$ has a compound Poisson distribution with parameters α_i and p.

The p. m. f. of $N_i(t)$ is given by

$$P[N_i(t) = r] = {}^{(r+\alpha_i-1)}C_r\, p^{\alpha_i}\, q^r \qquad \ldots \qquad (2)$$

where, q = 1 - p.

Now consider $E[\Lambda(t)]$. From (1) we get

$$E[\Lambda(t)] = E[\textstyle\sum_{i=1}^{m}(\lambda_i \psi_i(t))] \qquad \ldots \qquad (3)$$

Now, $\lambda_i \psi_i(t)$ is a function that is equal to λ_i when $\psi_i(t) = 1$ that is when no error is detected by t time units. Thus, $\lambda_i \psi_i(t) = \lambda_i$ with probability ${}^{(\alpha_i-1)}C_0\, p^{\alpha_i}\, q^0 = [\frac{1}{(1+t)}]^{\alpha_I}$.

$\lambda_i \psi_i(t)$ is equal to 0 when $\psi_i(t) = 0$. This happens if at least 1 error is detected in t time units and the corresponding bug is removed. Hence,

$$E[\Lambda(t)] = \textstyle\sum_{i=1}^{m}(E[\lambda_i] P[\psi_i(t) = 1])$$

$$= \textstyle\sum_{i=1}^{m}(\alpha_i[\frac{1}{(1+t)}]^{\alpha_I}) \qquad \ldots(4)$$

We now find $E[M_1(t)]$ = Expected number of bugs causing exactly 1 error.

As the Poisson processes caused by different bugs are independent, we have

$$E[M_1(t)] = E[\textstyle\sum_{i=1}^{m}(I_i(t))] = \sum_{i=1}^{n}(E[I_i(t)])$$

$$= \textstyle\sum_{i=1}^{m}([1 . P[\text{ bug i causes 1 error in time t}] + 0]$$
$$= \textstyle\sum_{i=1}^{m}([{}^{(1+\alpha_i-1)}C_1\, p^{\alpha_i}\, q^1]$$

where, $p = \frac{1}{(1+t)}$ and $q = \frac{t}{(1+t)}$

Thus, $E[M_1(t)] = \sum_{i=1}^{m}([{}^{(1+\alpha_i-1)}C_1 [\frac{1}{(1+t)}]^{\alpha_I} [\frac{t}{(1+t)}]^1)$

$$= \textstyle\sum_{i=1}^{m}(\alpha_i [\frac{1}{(1+t)}]^{\alpha_i} [\frac{t}{(1+t)}])$$

ISBN - 978-93-81583-55-5

$$= [\frac{t}{(1+t)}] \sum_{i=1}^{m}((\alpha_i) [\frac{1}{(1+t)}]^{\alpha_I}) \qquad \ldots \qquad (5)$$

Comparing equation (4) with (5), we get
$E [M_1(t)] =[\frac{t}{(1+t)}] E[\Lambda(t)]$
Therefore, $E[\Lambda(t)] = E([\frac{(1+t)}{t}] M_1(t))$.
This proves the theorem.

Thus, $[\{(1 + t)/t\} M_1(t)]$ can be taken as an estimator of the random variable $\Lambda(t)$.

3. VERIFICATION OF THE THEOREM BY SIMULATION:

We give below the R-Code with its output to verify the theorem.

The program simulates 25 bugs, each producing number of errors that follows Poisson distribution with parameter which itself is a random variable following Gamma distribution with parameters (α, β), where α_i takes the values from 1 to 25 and β is taken as 1. The time duration is taken as 30 units of time.

Program in R:

```
t<-30;   # time duration is 30 units of time
resirate<-c(rep(0,500)); # initialization of residual rate
m1<-c(rep(0,500)); # initialization of  total number of bugs
# producing single error

# iter gives the serial number of iteration.
# It ranges from 1 to 500
for(iter in 1:500)
{ lambda<-0;  # lambda is Poisson parameter
             para<-0; # initialization of parametric value

 # j gives the serial number of bug.
# We suppose that there are 25 bugs in all.
           for(j in 1:25)
               { lambda[j]=rgamma(1,j,1);
                 # generation of Gamma variate
                 para[j]=lambda[j]*t;
                 # parameter is t times lambda
               }
         numerr<-0;
 # numerr gives number of errors produced by different
# bugs
           for(i in 1:25)
             { numerr[i]=rpois(1,para[i]);
 # generation of Poisson variate
 # Calculation of Residual Rate in the Package
```

```
                if (numerr[i]==0)
                resirate[iter]=resirate[iter]+para[i];
 # Calculation of number of bugs producing single error
                if (numerr[i]==1)
             m1[iter]=m1[iter]+1;
                }
             }
# resirate is a vector giving the sums of error rates when
 # error is not detected

m<-((1+t)/t)*m1;
# m is a vector giving the estimates of residual error rate
 # based on m1

mean(resirate);
# this gives estimate of average residual error rate based
 # on Poisson parameter

est<-mean(m);
# est gives the estimate of residual rate of errors in the
 # package based on m1
est;

summary(m);
summary(resirate);
mean((resirate-m)^2);
 #  This is measure of error in estimation
var(resirate);
var(m);
```

Output of the Program:

Table I:
Mean and Variance of Residual Rate and Estimator Along with the Measure of Error in Estimation (λ is a Gamma Variate)

Sr. No.	Mean			Var	
	[*resirate*]	[*m*]	(*resirate-m*)^2	[*resirate*]	[*m*]
1	.0365	.0413	.0931	.0492	.0411
2	.0348	.0310	.1036	.0705	.0311
3	.0233	.0248	.0615	.0357	.0251
4	.0392	.0475	.1187	.0682	.0469
5	.0308	.0331	.0778	.0428	.0331
6	.0466	.0331	.1216	.0854	.0331
7	.0358	.0310	.0759	.0429	.0311

Observations:

1. Simulation studies have substantiated the above theorem. Please refer to Table 1.

   Although the variates are discrete, output values are not rounded to the nearest integers to show accuracy of the actual estimates, otherwise, most of the times the values will be rounded to zero.

2. To determine whether the proposed estimator is a good estimator of $\Lambda(t)$, we have considered $E [\Lambda(t).- \text{Estimator}]^2$ as a measure of error in estimation. We have evaluated its values by simulation. Please refer to Table 1. The estimates are found to be very accurate, if the number of iterations in the program is increased.

3. If λ is taken as constant then we get the following well known result by Ross.
   $E[\Lambda(t)] = E [M_1(t)/t]$

In Table 2, we verify the same.

Table II:
Mean and Variance of Residual Rate and Estimator Along with the Measure of Error in Estimation (λ = Constant = 4, t = 1)

Sr. No.	Mean			Var	
	[*resirate*]	**[*m*]**	**(*resirate-m*)^2**	**[*resirate*]**	**[*m*]**
1	1.968	1.82	9.74	7.8547	1.7431
2	1.944	1.81	10.006	7.2113	1.9017
3	1.8	1.884	8.404	6.6613	1.6579
4	1.8	1.954	9.85	7.2385	1.8917
5	1.696	1.818	8.866	6.5447	1.6602

4. The values of residual rate and its estimator depend on the Poisson parameter which is dependent on the values of the Gamma parameter α_i (i=1, …,m).

If the error rate is very small and the time duration is also small, then residual error rate seems to have increased as the errors are not detected in the short duration. By considering various durations and the corresponding residual error rate, we can decide the stopping rule.

The following table shows mean and variance values of residual rate and its estimator along with the measure of error in estimation for various values of t.

Here, we take $\alpha_j = (i+1)/100$, for j = 1, …, 25 in the program and number of iterations is 1500.

Table III
Mean and Variance of Residual Rate and Estimator Along with the Measure of Error in Estimation For varying Time Duration t

Sr. No.	t	Mean			Var	
		[*resirate*]	[*m*]	(*resirate-m*)^2	[*resirate*]	[m]
1	10	2.0776	2.2836	4.7220	1.1948	2.2160
2	30	1.9182	1.9716	4.2043	1.9811	1.9996
3	70	1.6625	1.7628	3.6631	1.8370	1.5813
4	100	1.5996	1.6066	3.5070	1.8193	1.5316
5	300	1.3363	1.3953	2.9472	1.4856	1.3519

It is seen that as t increases, the residual rate decreases. We can decide the duration t as per our requirement of accuracy in a particular field of application, e.g. Inventory Management.

5. We have considered Gamma distribution with only one parameter, because any Gamma distribution with two parameters can always be written in terms of Gamma with one parameter by change of variables technique. For example if $X \sim G(\alpha, \beta)$, and $Y = X/\beta$ then $Y \sim G(\alpha, 1)$ distribution.

4. CONCLUSION

When some software is under testing and the bugs cause errors according to Poisson rule where the parameter itself is a random variable, following Gamma distribution, the expected residual error rate can be estimated by just counting the bugs producing exactly single error in a given time duration.

Further we may allow the bug to remain in the software if it produces up to a small finite number of errors. This policy may be applied to more realistic cases in other fields also. For example, depending on the number of unsold cars of a particular model of a car, the manufacturing company may take decision of discontinuing or increasing the production of that particular model. Work in direction is in progress.

Some other variations in the error distributions are also under consideration.

5. REFERENCES

1. Lindsey, Pavur R. and Keeling K. (2008) Estimating the Future Rates of Poisson Events After Removing Certain Variables: Applications to Software. (Source: Internet) *Proceedings of the 2008 Annual Decision Sciences Institute Meeting* (Baltimore, MD:2008)
2. Ross, S. M. (1985). Software Reliability: The Stopping Rule problem. *IEEE Transactions of Software Engineering, 11,* 1472-1476.
3. Ross, S. M. (2002). *Introduction to Probability Models (6th ed.).* New York: Academic press.

****

\--------------

Dr. Meenakshi M. Sagdeo
Associate Professor of Statistics
Govt. of Maharashtra's Ismail Yusuf College, Jogeshwari (East), Mumbai-400060
msagdeo@hotmail.com