The Comparative Study of NHPP Half-Logistic Distribution Software Reliability Model using the Perspective of Learning Effects

Hee-Cheul KIM

1. Department of Industrial & Management Engineering, Namseoul University, 91 Daehak-ro Seonghwan-eup Sebuk-gu Cheonan-si Chungcheongnam-do S.KOREA
kim1458@nsu.ac.kr

Abstract

In this study, software products developed in the course of testing, software managers in the process of testing software test and test tools for effective learning effects perspective have been studied using the NHPP software reliability models. The finite failure non-homogeneous Poisson process models presented and want to use half-logistic distribution as the lifetime distribution which used to statistical process. Software error detection techniques known in advance, but influencing factors for considering the autonomous errors-detected factor and learning factors, by prior experience, to find precisely the error factor setting up the testing manager is presented on comparing the problem. The general result, case of the learning factor is greater than automatic error is efficient model, but, when applied to half-Logistic distribution model, case of autonomous errors-detected factor greater learning factor is efficient. This paper, a failure data analysis of applying using time between failures and parameter estimation using maximum likelihood estimation method were made, after the efficiency of the data through trend analysis, model selection was performed using the mean square error and coefficient of determination.

Keywords: Learning Effects, Non-Homogeneous Poisson Process, Half-Logistic Distribution.

1. Introduction

Software failures caused by failure of computer systems in our society can lead to huge losses. Thus, software reliability in the software development process is an important issue. This issue of the user requirements and must meet the cost of testing. Software testing (debugging) in order to reduce costs in terms of changes in the software reliability and testing costs, you need to know in advance is more efficient. Thus, the reliability, cost, and with consideration of release time for software development process are essential. Eventually the software that predicts the contents of a defect in the product development model is needed. Until now, many software reliability models have been proposed. Non-homogenous Poisson Process (NHPP) models rely on an excellent model [1] [2] in terms of the error discovery process, and if a fault occurs, immediately remove the debugging process and the assumption that no new fault has occurred.

In this field, enhanced non-homogenous Poisson Process model presented by Gokhale and Trivedi [1]. Goel and Okumoto [2] proposed an exponential software reliability models. In this model, the total number of defects have S-shaped or exponential-shaped with a mean value function was used. The generalized model relies on these models, delayed S-shaped reliability growth model and inflection S-shaped reliability growth model were proposed by Yamada and Ohba [3]. Zhao [4] proposed a software reliability problems in change point and Shyur [5] using the generalized reliability growth models proposed. Pham and Zhang [6] testing measured coverage, the stability of model, with software stability can be evaluated, presented.

Relatively recently, Huang [7], generalized logistic testing-effort function and the change-point parameter by incorporating efficient techniques to predict software reliability, were present. Chiu, Huang and Lee [8] can explain the learning process that software managers to become familiar with the software and test tools for S-type model.

In this study, characteristics of the applied half-logistic distribution were compared based on finite failure NHPP for influential factors consisting of autonomous errors-detected factor and learning factor.
2. Related research

2.1 Finite NHPP model

This is a class of time domain [1] [3] and [9] software reliability models which assume that software failures display the behavior of a non-homogeneous Poisson process (NHPP). The parameter of the stochastic process, \( \lambda(t) \) which denotes the failure intensity of the software at time \( t \), is time-dependent.

Let \( N(t) \) denote the cumulative number of faults detected by time \( t \) and \( m(t) \) denote its expectation. Then \( m(t) = E[N(t)] \) and the failure intensity \( \lambda(t) \) is related as follows:

\[
m(t) = \int_0^t \lambda(s) \, ds
\]

And,

\[
\frac{dm(t)}{dt} = \lambda(t)
\]

\( N(t) \) was known to have a Poisson PDF (probability density function) with parameter \( m(t) \), that is:

\[
p(N(t) = n) = \frac{[m(t)]^n}{n!} e^{-m(t)}, \quad n = 0, 1, \ldots, \infty
\]

Various time domain models have appeared in the literature which describes the stochastic failure process by NHPP. These models differ in their failure intensity function \( \lambda(t) \) and hence \( m(t) \).

The NHPP models can be further classified into finite failure and infinite failure categories. Finite failure NHPP models assume that the expected number of faults detected given infinite amount of testing time will be finite, whereas the infinite failures models assume that an infinite number of faults would be detected in infinite testing time [10]. Thus, using general order statistics (GOS) has become failure NHPP. On the other hand, using record value statistics (RVS) has become infinite NHPP.

Let \( \theta \) denote the expected number of faults that would be detected given finite failure NHPP models. Then, the mean value function of the finite failure NHPP models can also be written as:

\[
m(t) = \theta \, F(t)
\]

Where \( F(t) \) is a CDF (cumulative distribution function). From Equation (4), the (instantaneous) failure intensity \( \lambda(t) \) in case of the finite failure NHPP models is given by:

\[
\lambda(t) = \theta \, F'(t)
\]

This can be re-written as:

\[
\lambda(t) = \left[ \theta - m(t) \right] \frac{F'(t)}{1 - F(t)} = \left[ \theta - m(t) \right] h(t)
\]

Where \( h(t) \) is the failure occurrence rate per fault of the software, or the rate at which the individual faults manifest themselves as failures during testing. The quantity \( \left[ \theta - m(t) \right] \) denotes the expected number of faults remaining in the software at time \( t \). Since \( \left[ \theta - m(t) \right] \) is a monotonically non-increasing function of time (actually \( \left[ \theta - m(t) \right] \) should decrease as more and more faults are detected and removed [3]), the nature of the overall failure intensity, \( \lambda(t) \) is governed by the nature of failure occurrence rate per fault \( h(t) \), from Equation (6). The failure occurrence rate per fault \( h(t) \) can be a constant or increasing, decreasing. In this section, we describe some of the finite failure NHPP models along with their hazard functions.
Let \( \{ t_n, n = 1, 2, \ldots \} \) denote the sequence of times between successive software failures. Then \( t_n \) denote the time between \((n-1)\)th and \(n\)th failure. Let \( x_n \) denote the time to failure \( n \), so that:

\[
x_n = \sum_{i=1}^{n} t_i
\]

The joint density or the likelihood function of \( x_1, x_2, \ldots, x_n \) can be written as \([1, 3 \text{ and } 10]\):

\[
f(x_1, x_2, \ldots, x_n) = e^{-\lambda x_n} \sum_{i=1}^{n} \lambda(x_i)
\]

For a given sequence of software failure times \((x_1, x_2, \ldots, x_n)\), that are realizations of the random variables \((X_1, X_2, \ldots, X_n)\), the parameters of the software reliability growth models are estimated using the maximum likelihood method (MLE).

As a result, the conditional reliability \( \hat{R}(t|x_n) \) is known as follows \([3] \text{ [11]}\):

\[
\hat{R}(t|x_n) = e^{-\int_{x_n}^{t} \lambda(t)\,dt} = \exp[-m(t+x_n)-m(x_n)]
\]

Where \( t \) denote mission time and \( x_n \) is the last failure time.

3. Cumulative and intensity function considering learning factor

Software testing jobs, learning effects are testing by admin can be the same or manipulation of these effects and possible action in any way reflect the software reliability is an important process.

The influential factors that contain autonomous errors-detected factor \( \gamma \) and learning factor \( \eta \) can be considered for the finding software errors.

Therefore , suppose that \( f(t) \) is the intensity function that denotes the fraction of the errors detected at time \( t \), \( F(t) \) is the cumulative distribution function that denotes the fraction of the errors detected within time \((0, t]\), and \( 1-F(t) \) is the fraction of the errors as yet undetected at time \( t \).

Model of considering the influence factors can be expressed as follows \([8]\).

\[
f(t) = (\gamma + \eta F(t))(1-F(t))
\]

Where, \( \gamma > 0, \eta > 0 \).

The autonomous errors-detected factor indicates that the testing staff/software developers spontaneously find software errors which they were unaware. Meanwhile, the learning factor indicates that the testing staff/software developers deliberately set out to find software errors from software system to find software errors from the patterns which were previously detected.

Both factors can improve the efficiency of software debugging. On the other hand, equation (11) from the form of the following hazard function can be changed.

\[
h(t) = (\gamma + \eta F(T))
\]

Where, \( h(t) = \frac{F(t)}{1-F(t)} \)

From equation (12), the cumulative probability and density function can be modified as follows.

\[
F(t) = \left(\frac{h(t)-\gamma}{\eta}\right), \quad f(t) = F'(t) = \left(\frac{h'(t)-\gamma}{\eta}\right)
\]
4. Half-logistic distribution software reliability NHPP model using the perspective of learning effects

In this section, half-logistic distribution model was applied. The mean value function and intensity function of finite failure NHPP model are known as follows [12][13].

\[
m(t) = \theta F(t) = \theta \left( \frac{1-e^{-\beta t}}{1+e^{-\beta t}} \right), \quad \theta > 0, \beta > 0, t \geq 0.
\]

\[
\lambda(t) = \theta F'(t) = \theta \left( \frac{2 \theta \beta e^{-\beta t}}{(1+e^{-\beta t})^2} \right)
\]

(14)

Where \( \theta \) denote the expected number of fault and \( \beta \) is shape parameter. Using (14) equation, the hazard function is derived as follows.

\[
h(t) = \frac{F'(t)}{1-F(t)} = \frac{\beta e^{-\beta t}}{\eta(1+e^{-\beta t})^2}
\]

(15)

Thus, using (15) and (13) equation, the cumulative probability and density probability function can be modified as follows.

\[
F(t) = \frac{\beta}{\eta + e^{-\beta t} - \gamma}, \quad f(t) = \frac{\beta e^{-\beta t}}{\eta(1+e^{-\beta t})^2}
\]

(16)

The mean value function and intensity function of finite failure NHPP model added learning effects, using (4) and (5), and (16) equation, can be expressed as follows.

\[
m(t) = \theta F(t) = \theta F(t) = \theta \left( \frac{\beta}{1+e^{-\beta t} - \gamma} \right), \quad \lambda(t) = \theta F'(t) = \theta \left( \frac{\beta - \gamma(1+e^{-\beta t})}{\eta(1+e^{-\beta t})} \right)
\]

(17)

In this case, the likelihood function, substituting (17) for (9) equation, is as follows.

\[
L_{NHPP}(\beta | D_{n_x}) = \prod_{i=1}^{n} \left[ \theta \left( \frac{\beta e^{-\beta x_i}}{\eta(1+e^{-\beta x_i})^2} \right) \right] \exp \left[ -\theta \left( \frac{\beta - \gamma(1+e^{-\beta x_i})}{\eta(1+e^{-\beta x_i})} \right) \right]
\]

(18)

The maximum likelihood estimation method (MLE) of parameter estimation was used. The log likelihood function, for the maximum likelihood estimation using (18) equation, is derived as follows.

\[
\ln L_{NHPP}(\beta | D_{n_x}) = n \ln \theta + n \ln \beta - n \ln \eta + \beta \sum_{i=1}^{n} x_i - \sum_{i=1}^{n} \ln(1+e^{-\beta x_i}) - \theta \left( \frac{\beta - \gamma(1+e^{-\beta x_i})}{\eta(1+e^{-\beta x_i})} \right)
\]

(19)

Thus, using equation (19), maximum likelihood estimate \( \hat{\beta}_{MLE} \) and \( \hat{\theta}_{MLE} \) could be obtained were as follows:

\[
\frac{\partial \ln L(\alpha, \beta | D_{n_x})}{\partial \beta} = \frac{n}{\theta} \left( \frac{\beta - \gamma(1+e^{-\beta x_i})}{\eta (1+e^{-\beta x_i})} \right) = 0
\]

(20)
\[
\frac{\partial \ln L(a, b | \beta)}{\partial \beta} = a - \frac{\beta}{\beta} \sum_{i=1}^{n} x_i + 2 \sum_{i=1}^{n} x_i e^{-\beta x_i} - \frac{\beta}{\eta} \left( 1 + e^{-\beta x_i} (1 - x_i + \beta x_i - x_i^2) \right) = 0
\]  

(21)

Where, \( x = \{0, x_1, x_2, x_3, ..., x_n\} \).

5. Model comparison with real datasets

In order to investigate the effectiveness of the proposed model, the comparison criteria we used are described as follows [8][14]:

(1) The mean square error (MSE) measures the deviation between the predicted values with the actual observations. It is defined as

\[
MSE = \frac{\sum_{i=1}^{n} \left( m(x_i) - \hat{m}(x_i) \right)^2}{n - k}
\]

(31)

where \( m \) is the total cumulated number of errors observed within time \( (0, t] \), \( \hat{m} \) is the estimated cumulative number of errors at time \( x_i \) obtained from the fitting mean value function, \( n \) is the number of observations and \( k \) is the number of parameters to be estimated.

(2) R square \((R^2)\) can measure how successful the fit is in explaining the variation of the data. It is defined as

\[
R^2 = 1 - \frac{\sum_{i=1}^{n} \left( m(x_i) - \hat{m}(x_i) \right)^2}{\sum_{i=1}^{n} \left( m(x_i) - \frac{\sum_j^n m(x_j)}{n} \right)^2}
\]

(32)

6. Software failure data analysis

In this chapter, software failure data [13][15] is to analyze the characteristics of learning factors. This data set in Table 1 lists and in order to analyze the trust models presented in the first data set should be preceded by a trend test [16].

In general, the Laplace trend test analysis is used. As a result of this test in this Figure 1, as indicated in the Laplace factor is between 2 and -2, reliability growth shows the properties. Thus, using this data it is possible to estimate the reliability [17][18].

<table>
<thead>
<tr>
<th>Failure Number</th>
<th>Failure Time(hours)</th>
<th>Failure Interval (hours)</th>
<th>Failure Number</th>
<th>Failure Time(hours)</th>
<th>Failure Interval (hours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>30.02</td>
<td>30.02</td>
<td>16</td>
<td>151.78</td>
<td>15.53</td>
</tr>
<tr>
<td>2</td>
<td>31.46</td>
<td>1.44</td>
<td>17</td>
<td>177.5</td>
<td>25.72</td>
</tr>
<tr>
<td>3</td>
<td>53.93</td>
<td>22.47</td>
<td>18</td>
<td>180.29</td>
<td>2.79</td>
</tr>
<tr>
<td>4</td>
<td>55.29</td>
<td>1.36</td>
<td>19</td>
<td>182.21</td>
<td>1.92</td>
</tr>
<tr>
<td>5</td>
<td>58.72</td>
<td>3.43</td>
<td>20</td>
<td>186.34</td>
<td>4.13</td>
</tr>
<tr>
<td>6</td>
<td>71.92</td>
<td>13.2</td>
<td>21</td>
<td>256.81</td>
<td>70.47</td>
</tr>
</tbody>
</table>
Table 2. MLE, MSE and $R^2$ considering Influential factors for each model

<table>
<thead>
<tr>
<th>Influential factors</th>
<th>Half-logistic model</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\eta$</td>
<td>$\gamma$</td>
</tr>
<tr>
<td>0.3</td>
<td>0.7</td>
</tr>
<tr>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>0.7</td>
<td>0.3</td>
</tr>
</tbody>
</table>

Explanatory notes:
- $\eta$: Learning factor, $\gamma$: Autonomous errors-detected factor
- $M\text{LE}$: Maximum likelihood estimation; $M\text{SE}$: Mean square error,
- $R^2$: Coefficient of determination, the smallest and highest are indicated in boldface;

The estimation of parameters for each model used maximum likelihood method. In order to facilitate parameter estimation, the time to failure of the original variables ($\text{Failure time} \times 10^{-2}$) to convert the data was applied. The result of parameter estimation has been summarized in Table 2. These calculations, solving numerically, the initial values given to 0.001 and 3.0 and tolerance value for width of interval given $10^{-3}$ using C-language checking adequate convergent, were performed iteration of 100 times. The result of mean square error (MSE) and $R^2$-square ($R^2$) are has been summarized in Table 2. In Table 2, in term of influential factors, unlike the general thought, half-logistic model is generally having autonomous errors-detected factor greater than a learning factor is efficient, due to the small MSE in each influential factor. But, in terms of the coefficient of determination, learning factors are higher with increasing explanatory power. Thus, the higher the learning factor model is more utility model. In Figure 1, reliability considering influential factors for each is showing a decrease form.
In terms of reliability, the greater the learning factor has shown low reliability. In terms of comparison of reliability, case of reliability for assumed mission time, autonomous errors-detected factor than learning factor model has shown high reliability. Namely, cases of the autonomous errors-detected greater or equal greater than learning factor shows tendency to rise slightly, but, case of lower shows tend to rise gradually have decreasing form as mission time. Eventually, the reliability has been sensitive to the learning factor for the mission time. Therefore, half-logistic model is judged more reliable model in this field.

![Reliability Vs. Mission Time](image)

Explanatory notes: \( \eta \) is and \( \gamma \) is gamma.

**Figure 2.** Comparison reliability considering influential factors for each model

7. **Conclusion**

The characteristics of the half-logistic distribution model, based on finite failure NHPP for Influential factors consisting of autonomous errors-detected factor and learning factor, were studied in this paper.

In this paper, the following conclusions were obtained.

In term of influential factors, the half-logistic distribution is generally having autonomous errors-detected factor greater than learning factor is efficient. Generally, when learning factor is the highest and autonomous errors-detected factor is the lowest, model was effective depressing. Because of the coefficient of determination, learning factors are higher with increasing explanatory power. Thus, the higher the learning factor model is more utility model. In terms of reliability, the greater the learning factor has shown low reliability. Eventually, the reliability has been sensitive to the learning factor for the mission time. Therefore, in this paper, half-logistic distribution characteristics can be used as an alternative reliability model in this field. An alternative study for this area will be valuable research.

8. **Acknowledgment**

Funding for this paper was provided by Namseoul University.

9. **References**

