

COMPARING BETWEEN MAXIMUM LIKELIHOOD AND LEAST SQUARE ESTIMATORS FOR GOMPERTZ SOFTWARE RELIABILITY MODEL

Lutfiah Ismail Al turk

Statistics Department, King Abdulaziz University, Kingdom of Saudi Arabia

Abstract

Software reliability models (SRMs) are very important for estimating and predicting software reliability in the testing/debugging phase. The contributions of this paper are as follows. First, a historical review of the Gompertz SRM is given. Based on several software failure data, the parameters of the Gompertz software reliability model are estimated using two estimation methods, the traditional maximum likelihood and the least square. The methods of estimation are evaluated using the MSE and R-squared criteria. The results show that the least square estimation is an attractive method in term of predictive performance and can be used when the maximum likelihood method fails to give good prediction results.

Keywords

Software Reliability Models, Gompertz Growth Curve, Time Data, Count Data, Maximum Likelihood Estimation, Least Square Estimation.

1. INTRODUCTION

Because of the rapid growth of the size, complexity, and diversification of computer systems, A wealth of software reliability models (SRMs) have been developed over the years to assess, predict, and improve of software reliability. Some models use a non-homogeneous Poisson process (NHPP) to model the failure process, it is a traditional and important class of SRMs. The NHPP is characterized by its expected mean value function, $H(t)$. This is the cumulative number of failures expected to occur after the software has executed for time t . $H(t)$ is non-decreasing in time t with a bounded condition, $H(\infty) = a$, where a is the expected number of failures to be encountered if testing time is infinite. Gompertz SRM model is based on an NHPP. This SRM model is one of the simplest S-Shaped software reliability growth models. It takes the number of faults per unit of time as independent Poisson random variables. The maximum likelihood estimation has been frequently considered to estimate the parameters of the SRMs ([1], [2], [3], [4], and [5]). In the literature, some cons on the maximum likelihood estimation have been pointed out such as there is no guarantee to always have a unique ML estimate, more specifically

the non-linearity may cause the failing of the MLE method. The MLE method cannot be easily applied to estimate the parameter in the Gompertz SRM, because of its strong non-linearity. As an alternative to the MLE method, when it fails to give accurate results, the least squares estimation (LSE) method can be applied. The rest of the paper is organized as follows: detailed review of the Gompertz SRM will be provided in Section 2. Section 3 discusses the maximum likelihood and the least square estimation methods for the Gompertz SRM. Section 4 presents some selected data sets. Section 5 describes the criteria for comparing the estimation methods used in this study. In Section 6, numerical example is illustrated and Section 7 presents conclusion.

2. Historical Review of the Gompertz SRM

Deterministic SRMs formulated by Gompertz growth curve has been widely used to estimate the error content [6]. In Japan, some computer manufacturers and software houses have actually applied this model. This curve was originally developed to predict demand trend, economic growth, or future population. One of the SRMs that belongs to the Non Homogeneous Poisson Process (NHPP) is the Gompertz model, it gives good approximation to a cumulative number of software faults observed interesting software. The model was first proposed by Goel and Okumoto [7] and has formed the basis for the models using the observed number of faults per unit time group. Ohba [8] presented a detailed study of few interesting models and indicated the use of several other models, such as Gompertz model and logistic curves as used in software reliability study in Japan. Kececioglu [9] provided methods to estimate initial values for the Gompertz model. Parameter estimates may also be obtained using non-linear regression. This approach fits the curve to the data and estimates the parameters from the best fit to the data, where fit is defined as the difference between the data and the curve function fitting the data. Kececioglu, Jiang, and Vassiliou [10] modified the Gompertz model to include a fourth parameter. This fourth parameter shifts the associated growth curve vertically, thus accommodating for S-shaped growth datasets.

The new method is claimed to be more flexible than its predecessor for fitting data with S-shaped trends. Sk.MD. Rafi and Shaheda Akthar [11] proposed a discrete SRGM with discrete exponential, discrete Gompertz and discrete logistic testing effort function (TEF) curve. At the same time they developed discrete imperfect debugging SRGM with discrete TEF. Also, they developed optimal release policy based on cost, reliability and intensity requirements. Swamydoss D and Dr. G.M. Kadhar Nawaz [12] used the Gompertz model to predict the reliability of the software. It is shown that the proposed model can be derived from the well-known statistical theory of extreme value and has the quite similar sympatric property to the classical Gompertz curve. They applied the Gompertz software reliability model to assess the software reliability and to predict the number of initial fault contents. In their work, they considered web based application with alternative approach for parameter estimation in Gompertz SRM, they showed that the least mean square estimation approach is attractive in terms of goodness of fit tests.

For the Gompertz growth curve model, the expected cumulative number of errors detected up to testing time t is given by

$$H(t) = G(t) = \eta a^{b^t} ; \quad \eta > 0, 0 < a < 1, 0 < b < 1, \quad (1)$$

where η , a and b are constant parameters to be estimated, b provides a shape parameter to the Gompertz model equation, it models the growth pattern (small values model rapid early reliability growth, and large values model slow reliability growth). The parameter η is the expected initial error content of a software product, the model characteristics are as follows:

The error detection rate is

$$b(t) = \frac{b^t \ln a \ln b}{a^{-b^t} - 1}, \quad (2)$$

while, the intensity function is,

$$\xi(t) = \eta a^{b^t} b^t \ln a \ln b \quad (3)$$

also, the number of remaining errors and the conditional reliability functions are defined respectively by

$$n(t) = \eta (1 - a^{b^t}), \quad (4)$$

$$\begin{aligned} R(x|t) &= \exp\left\{-\left(\eta a^{b^{t+x}} - \eta a^{b^t}\right)\right\} \\ &= \exp\left\{-\eta a^{b^t} \left(a^{b^{t+x}-1} - 1\right)\right\}, \end{aligned} \quad (5)$$

and the mean time between software failures can be obtained by

$$\mu TBF(t) = \frac{a^{-b^t} b^{-t}}{\eta \ln a \ln b}. \quad (6)$$

3. Estimation Procedures

Fitting a proposed model to actual failure data involves estimating the model's parameters from the test data sets. In this section, two methods of estimating the unknown parameters of the Gompertz SRM will be considered, the maximum likelihood and the least square. The maximum likelihood method will be investigated in two cases count and time data, for the illustrative data analysis only the case of time data will be considered.

3.1. The maximum likelihood estimation (MLE) method

MLE method is one of the most popular estimation techniques. The MLE technique estimates parameters by solving a set of simultaneous equations, in the following the MLE method of the NHPP models will be illustrated in the case of time and count data. In time data the time between failures is considered as the random variable, while the interest of the count data is in the number

of faults or failures in specified time intervals. Also the MLE in the case of time and count data will be obtained for the Gompertz SRM.

MLE method of the NHPP SRMs in case of count data

Suppose that the error-detection count data (t_i, y_i) , $i = 1, 2, \dots, n$, are observed during the testing phase. Then the likelihood function for the unknown parameters in an NHPP model with $H(t)$ is given by

$$L(a, \underline{\theta} | \underline{t}, \underline{y}) = P\{J(t_1) = y_1, J(t_2) = y_2, \dots, J(t_n) = y_n\}$$

$$= \prod_{i=1}^n \frac{\{H(t_i) - H(t_{i-1})\}^{y_i - y_{i-1}}}{(y_i - y_{i-1})!} \exp[-\{H(t_i) - H(t_{i-1})\}], \tag{7}$$

where

- $t_0 = 0, y_0 = 0$,
- $\underline{\theta} = (\theta_1, \theta_2, \dots, \theta_n)$ is then the model parameters,
- $\underline{t} = (t_1, t_2, \dots, t_n)$,
- $\underline{S} = (s_1, s_2, \dots, s_n)$,
- $\underline{y} = (y_1, y_2, \dots, y_n)$.

Taking the natural logarithm of both sides of equation (7) and equating its partial derivatives with respect to the unknown parameters to zero, we get

$$\frac{\partial \ln L(a, \underline{\theta} | \underline{t}, \underline{y})}{\partial a} = 0, \tag{8}$$

and

$$\frac{\partial \ln L(a, \underline{\theta} | \underline{t}, \underline{y})}{\partial \theta_i} = 0, \quad i = 1, 2, \dots, n. \tag{9}$$

Then, the $(n+1)$ maximum likelihood estimates \hat{a} and $\hat{\theta}_i$, $i = 1, 2, \dots, n$ can be obtained by solving equations (8) and (9) simultaneously.

MLE Gompertz models' parameters in case of count data

The form of the mean value function is given by equation (1) and the parameters η , a and b would be estimated. The logarithm of the likelihood function ℓ_1 in this case is:

$$\ell_1 = \sum_{i=1}^n (y_i - y_{i-1}) \ln \eta + \sum_{i=1}^n (y_i - y_{i-1}) \ln (a^{b^{t_i}} - a^{b^{t_{i-1}}}) - \sum_{i=1}^n \ln (y_i - y_{i-1}) - \eta \sum_{i=1}^n (a^{b^{t_i}} - a^{b^{t_{i-1}}}) \quad (10)$$

$$\frac{\partial \ell_1}{\partial \eta} = \frac{y_n}{\eta} - a (a^{b^{t_n-1}} - 1) = 0$$

$$\eta = \frac{y_n}{a (a^{b^{t_n-1}} - 1)} \quad ; \quad \ln b > 0. \quad (11)$$

The derivative of ℓ_1 with respect to a is

$$\frac{\partial \ell_1}{\partial a} = \sum_{i=1}^n \left\{ (y_i - y_{i-1}) \left(\frac{b^{t_i} a^{b^{t_i-1}} - b^{t_{i-1}} a^{b^{t_{i-1}-1}}}{a^{b^{t_i}} - a^{b^{t_{i-1}}}} \right) - \frac{y_n}{a (a^{b^{t_n-1}} - 1)} \cdot \sum_{i=1}^n (b^{t_i} a^{b^{t_i-1}} - b^{t_{i-1}} a^{b^{t_{i-1}-1}}) \right\}$$

The summation appearing in the second term of the last equation is equal to

$$b^{t_n-1} a^{b^{t_n}} - 1$$

Then, after equating $\frac{\partial \ell_1}{\partial a}$ to zero, we have

$$\frac{y_n (b^{t_n} a^{b^{t_n-1}} - 1)}{a (a^{b^{t_n-1}} - 1)} = \sum_{i=1}^n \left\{ (y_i - y_{i-1}) \left(\frac{b^{t_i} a^{b^{t_i-1}} - b^{t_{i-1}} a^{b^{t_{i-1}-1}}}{a^{b^{t_i}} - a^{b^{t_{i-1}}}} \right) \right\} \quad (12)$$

Differentiate (10) with respect to b and equate to zero. Thus, the following equation results

$$\frac{\partial \ell_1}{\partial b} = (\ln a) \sum_{i=1}^n \left\{ (y_i - y_{i-1}) \left(\frac{t_i b^{t_i-1} a^{b^{t_i}} - t_{i-1} b^{t_{i-1}-1} a^{b^{t_{i-1}}}}{a^{b^{t_i}} - a^{b^{t_{i-1}}}} \right) - \frac{y_n}{a (a^{b^{t_n-1}} - 1)} t_n a^{b^{t_n}} b^{t_n-1} \ln a \right\} = 0$$

which can be rewritten as

$$\frac{y_n t_n a^{b^{t_n}-1} b^{t_n-1}}{a^{b^{t_n}-1} - 1} = \sum_{i=1}^n \left\{ (y_i - y_{i-1}) \left(t_i b^{t_i-1} a^{b^{t_i}} - t_{i-1} b^{t_{i-1}-1} a^{b^{t_{i-1}}} \right) / \left(a^{b^{t_i}} - a^{b^{t_{i-1}}} \right) \right\} \tag{13}$$

MLE method of the NHPP SRMs in case of time data

Assessment of the reliability of the software at any time during the test process depends on the model assumed for the activation of faulty systems. Most of the models in the literature pertaining to this matter are the so called “time domain” models which consider the time till a fault is activated as a random variable realized according to some stochastic process. This last is generally taken as a non-homogeneous Poisson process. In this situation the collected data are called failure-occurrence time data. In this Section we will introduce the parameter’s estimators by using the MLE method, considering this case.

Suppose that the failure-occurrence time data s_i ($i = 1, 2, \dots, n$) are observed during the testing phase. Then, the joint density function of the observed data, that is, the likelihood function for estimating the unknown parameters in an NHPP model with the mean value function $H(t)$, is given by

$$L(a, \underline{\theta} | \underline{S}) = \exp\{-H(s_n)\} \prod_{i=1}^n h(s_i) \tag{14}$$

where

$$h(s_i) = \left. \frac{dH(t)}{dt} \right|_{t=s_i} \tag{15}$$

and a is the expected number of errors in the system.

$$H(s_n) = H(t) \Big|_{t=s_n} \tag{16}$$

MLE of Gompertz models’ parameters in case of time data

For the Gompertz model we have by equation (1),

$$\left. \begin{aligned} G_p(s_n) &= \eta a^{b^{s_n}} \\ g_p(s_i) &= \eta a^{b^{s_i}} b^{s_i} \ln(a) \ln(b) \end{aligned} \right\} \tag{17}$$

where

$$\eta > 0, 0 < a < 1, 0 < b < 1 .$$

$$L_7(a, b, \eta | \underline{S}) = \exp(-\eta a^{b^{s_n}}) \cdot \eta^n (\ln a)^n (\ln b)^n \prod_{i=1}^n a^{b^{s_i}} b^{s_i} . \quad (18)$$

Hence the log likelihood function is

$$\ell_7 = -\eta a^{b^{s_n}} + n \ln \eta + n \ln(\ln a) + n \ln(\ln b) + \sum_{i=1}^n b^{s_i} \ln a + \sum_{i=1}^n s_i \ln b . \quad (19)$$

Differentiate with respect to η , a and b and equate to zero to get the estimates of η , a and b respectively. These give

$$\eta = n/a^{b^{s_n}} , \quad (20)$$

$$\frac{nb^{s_n}}{a} = \frac{1}{a} \left(\frac{n}{\ln a} + \sum_{i=1}^n b^{s_i} \right) , \quad (21)$$

and

$$ns_n b^{s_n-1} \ln a = \frac{1}{b} \left(\frac{n}{\ln b} + \sum_{i=1}^n s_i + \sum_{i=1}^n s_i b^{s_i} \ln a \right) . \quad (22)$$

Equations (20), (21) and (22) can be solved numerically to obtain the ML estimates of $\hat{\eta}$, \hat{a} and \hat{b} .

3.2. The least squares estimation (LSE) method

LSE is a popular technique and widely used in many fields for function fit and parameter estimation (Liu, 2011). The LSE may be simple but very useful in estimating model parameters, it finds values of the parameters such that the sum of the squares of the difference between the fitting function and the experimental data is minimized. Mathematically, the LSE method concerns in determining the value of the unknown parameters that minimizes the following quantity:

$$LS(\underline{\theta} | \underline{t}) = \sum_{i=1}^n (i - H(t_i))^2 , \quad (23)$$

where $H(t_i)$ the total cumulated number of errors observed within time is $(0, t_i]$

4. Datasets

The following seven published datasets are chosen for our evaluations:

1. NTDS data: is from Goel and Okumoto [7], which originated from the U.S. Navy Fleet Computer Programming Center. These failure data were collected during the development phase of the software for the real-time multicomputer complex system that is the central part of the Navel Tactical Data System (NTDS).
2. The F11-D program data: were presented by Moranda [14] and pertain to a record of errors which occurred during the debugging of a data reduction program called the F11-D program. This program consists of “approximately 3-4 thousand” FORTRAN statements.
3. DACS data: is from J. Musa’s [15] “Software Reliability Data”, available from DACS, Rome Air Development Center, New York.
4. DS4 data: contains 20 inter-events times which are Weibull distributed with scale parameter 10 and shape parameter 2 [16].
5. DS5 data: represent the time between failures of software product reported by Musa [17], this data consists of 136 failures.
6. DS6 data: were presented by Guo et al. [18], and contain 21 failures of a repairable system.
7. DS7 data: is from Xie et al. [19], this data set contains 30 failures.

5. Model Comparison Criteria

In order to investigate the effectiveness of the selected estimation methods, the comparison criteria we used are described as follows:

- (1) The mean square error (MSE) measures the deviation between the predicted values with the actual observations [20]. Thus, the MSE value is defined as:

$$MSE = \frac{\sum_{i=1}^n (H(t_i) - \hat{H}(t_i))^2}{n-k}. \quad (24)$$

A smaller MSE indicate better performance.

- (2) R-squared (R^2) can measure how successful the fit is in explaining the variation of the data. A value of R^2 close to 1 indicates the best model, this measure can be expressed as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^n (H(t_i) - \hat{H}(t_i))^2}{\sum_{i=1}^n (H(t_i) - \sum_{j=1}^n H(t_j)/n)^2}, \quad (25)$$

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

This section evaluates the performance of two methods of estimation for the Gompertz SRM based on several real data sets. For the estimation of parameters for Gompertz model the maximum likelihood and the least square methods are used. The results of MSE and R^2 have been listed in Table (1) and Figure (1), Table (1) shows that, comparing the two method of estimation on all the seven real data sets, the MSE value for the LSE method is smaller than the MSE value for the MLE method. Also we can see that the R-squared for the LSE is much closer to 1 than the R-squared for the MLE method. Also, Figure 1 shows that the MSE of the MLE and LSE methods for Gompertz SRM based on different 7 data sets. From this figure, it is so clear that the least square method gives better results than the MLE method for the selected data sets.

7. Concluding Remarks

In this paper, we have provided a theoretical review of the Gompertz SRM, several mathematical formulas of the model's characteristics are obtained. The general methods to estimate the parameters of SRMs are LSE and MLE. For this model, those two methods of estimation are considered and evaluated using two different criteria. From the obtained results, we can conclude that the LSE is more useful method for estimating the parameters of the Gompertz SRM. The LSE approach can be adopted as a good alternative for the Gompertz SRM when the MLE method cannot be applied because of the model's strong non-linearity, the least squares estimation (LSE) may be simple but very useful in estimating model parameters. Hence, the LSE methods should be considered when studying software reliability models.

Table 1. MSE and R^2 Results for Gompertz SRM using MLE and LSE

Data set	MSE (MLE)	MSE (LSE)	R-squared(MLE)	R-squared(LSE)
NTDS	229.6723	0.6625226	-2.61194	0.9895808
F11-Dprogram	89.67286	0.8626611	-2.843123	0.9630288
DACS	413.2834	5.757572	-2.510515	0.951094
DS4	160.3058	0.5282724	-3.098042	0.9864953
Musa	6090.256	48.01285	-2.864339	0.9695353
DS6	175.192	42.77778	-3.095397	-1.1036e-13
DS7	332.7993	1.683625	-2.998034	0.979774

Explanatory notes:

MSE: Mean square error, the lowest is the better.

R^2 : R-squared, the closer to 1 is the better.

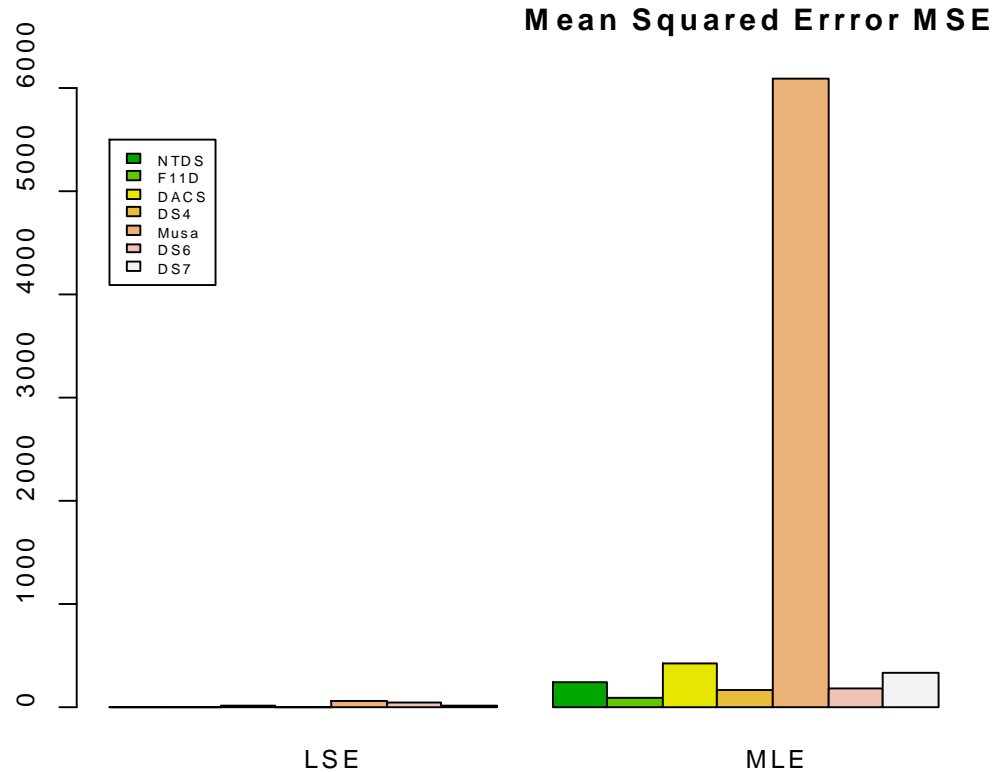


Figure 1. Comparing between MSE values for MLE and LSE methods.

References

- [1] Jeske, D. R. and Pham, H., 2001. On The Maximum Likelihood Estimates for The Goel-Okumoto Software Reliability Model. *The American Statistician*, 55(3), 219-222.
- [2] Okamura, H., Watanabe, T. and Dohi, T., 2003. An Iterative Scheme for Maximum Likelihood Estimation in Software Reliability Modeling. *Proceeding of 14th International Symposium on Software Reliability Engineering*, 14, 479-490.
- [3] Kundu, S., Nayak, T. N. and Bose, S. 2008. Are Nonhomogeneous Poisson Process Models Preferable to General-Order Statistics Models for Software Reliability Estimation? *Statistical Models and Methods for Biomedical and Technical Systems* (F. Vonta, M. Nikulin, N. Limnios and C. Huber-Carol, eds.), 133-154, Birkhauser, Boston.
- [4] Nayak, T. K., Bose, S., and Kundu, S., 2008. On Inconsistency of Estimators of Parameters of Non-Homogeneous Poisson Process Models for Software Reliability. *Statistics and Probability Letters*, 78, 2217-2221.
- [5] Ohishi, K., Okamura, H., and Dohi, T., 2009. Gompertz Software Reliability Model: Estimation Algorithm and Empirical Validation. *Journal of Systems and Software*, 82, 535-543.
- [6] Yamada, S., Ohba, M., and Osaki, S., 1983. S-shaped reliability growth modeling for software error detection. *IEEE Transactions on Reliability* 32(5): 475-478.

- [7] Goel, A.L. and Okumoto, K., 1979. Time-Dependent Error-Detection Rate Model for Software Reliability and other Performance Measures, IEEE Trans. Reliability, R-28, 3, pp. 206-211.
- [8] Ohba, M., 1984. Software Reliability Analysis Model, IBM. J. Res. Dev, 28, pp. 428-443.
- [9] Kececioglu, D., 1991. Reliability Engineering Handbook, Vol. 2, Englewood Cliffs, NJ: Prentice-Hall.
- [10] Kececioglu, D., Jiang, S., and Vassiliou, P. 1994. A Modified Gompertz Reliability - Growth Model", IEEE Proceedings of the Annual Reliability and Maintainability Symposium, pp. 160 -165.
- [11] Sk.MD. Rafi and Shaheda Akthar, 2011. Discrete Software Reliability Growth Models with Discrete Test Effort Functions. International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.4.
- [12] Swamydoss D, Dr. G.M.Kadhar Nawaz, 2013. An Enhanced Method of Lms Parameter Estimation for Software Reliability Model, International Journal of Innovative Research in Science, Engineering and Technology, Vol. 2, Issue 7, July.
- [13] Liu, J., 2011. Function based Nonlinear Least Squares and application to Jelinski-Moranda Software Reliability Model", stat. ME, 25th.
- [14] Moranda, P. B., 1975. Prediction of Software Reliability during Debugging, in Proceedings of the Annual Reliability and Maintainability Symposium, Washington, DC, IEEE Reliability Society, pp. 327-333.
- [15] Musa, J. D., 1979. Software Reliability Data. Report Available from Data & Analysis Center for software, Rome Air Development Center, Rome, New York, USA.
- [16] Klefsjo, B. and Kumar, U., 1992. Goodness-of-fit Test for the Power-Law Process based on the TTT-Plot, IEEE Trans. Reliability, Vol. 41, No. 4, pp. 593-598.
- [17] Musa, J. D., 1975. A Theory of Software Reliability and its Application. IEEE Trans. On Software Eng., vol. SH-1, pp. 312-327.
- [18] Guo, H., Mettas, A., Sarakakis, G., and Niu, P., 2010. Piecewise NHPP Models with Maximum Likelihood Estimation for Repairable Systems," Reliability and Maintainability Symposium (RAMS), 25-28 Jan. 2010 Proceedings, San Jose, CA.
- [19] Xie M., Goh, T. N., and Rajan P., 2002. Some Effective Control Chart Procedures for Reliability Monitoring. Elsevier Science Ltd, Reliability Engineering and System Safety 77, 143- 150.
- [20] Huang C. Y., Kuo, S. Y, and, Lyu M. R., 2007. An Assesment of Testing Effort Dependent Software Reliability Growth Model". IEEE transactions on Reliability, Vol 56, No: 2.

Author profile

Lutfiah Ismail Al turk is presently Assistant Professor of Mathematical Statistics in Statistics Department at Faculty of Sciences, King AbdulAziz University, Saudi Arabia. Lutfiah Ismail Al turk obtained her B.Sc degree in Statistics and Computer Science from Faculty of Sciences, King AbdulAziz University in 1993 and M.Sc (Mathematical statistics) degree from Statistics Department, Faculty of Sciences, King AbdulAziz University in 1999. She received her Ph.D in Mathematical Statistics from university of Surrey, UK in 2007.