BAYESIAN PREDICTIVE ANALYSES FOR NON-HOMOGENEOUS POISSON PROCESS IN SOFTWARE RELIABILITY WITH MUSA-OKUMOTO INTENSITY FUNCTION

CHERUIYOT NICKSON

A Thesis Submitted to the Graduate School in Partial Fulfillment of the Requirements for the Award of Master of Science Degree in Statistics of Egerton University

EGERTON UNIVERSITY

APRIL, 2019

DECLARATION AND RECOMMENDATION

DECLARATION

This thesis is my original work and has not been presented to any examination body.

Signature.. Date................................

Cheruiyot Nickson**.**
SM12/23521/14

RECOMMENDATION

This thesis has been submitted for examination with our approval as University supervisors.

Signature: ………………………….. Date: ……………………….

Dr. Orawo Luke Akong’o
Mathematics Department
Egerton University

Signature: ………………………….. Date: ……………………….

Prof. Islam Ali Salim
Mathematics Department
Egerton University

DEDICATION

I dedicate this thesis to my parents Joseah Koech and Janet Koech, my siblings, my wife Jescah Mutaih and my son Dylan. They have been of great encouragement and support both academically, financially and socially to me.

ACKNOWLEDGEMENT

All glory be to God, the Almighty and the most merciful. I am very thankful to Him with the blessing of Whom I achieve success in every aspect of my life. I would like to thank Egerton University for giving me this opportunity to study. Also would like to express my profound gratitude to my supervisors Dr. Luke Akong'o Orawo and Prof. Ali Salim Islam of Egerton University for allowing me to do this research under their supervision. I am sincerely honoured to have been their student. Their positive criticism both typography and technical came in handy when I was writing this thesis. I would like to thank Dr. L.A. Orawo once more whose programming techniques were so helpful during application of my derived methodologies. Words of thanks to Higher Education Loans Board (HELB) for partially financing this study.

I also express my gratitude to my parents, brothers, sisters and my wife for their prayers, financial and moral support throughout this study. I thank Egerton University especially the Department of Mathematics for providing me with the necessary support and facilities towards the success of this study. Special appreciation goes to my classmates Simon Mashigo, Emmanuel Kering, Beth Gitau, Mariam Kamau, Victor Mwenya and Osanya Milton for their positive comments, scholarly support and also their support in social life.

Finally, I wish to sincerely thank many other people who contributed in many small but significant ways, and whose names I may not have mentioned. You all contributed to the success of this work. God bless you all.

ABSTRACT

Due to rapid increase in development of complex computer systems over the past decades, there is need to estimate and predict the reliability of software systems during the testing process. Reliability refers to how well software meets its requirements and the probability of failure free operation for the specified period of time in a specified environment. The high demand and use of software has led to increased quest for more reliable software. For the past few decades several software reliability growth models have been used to describe the behavior of software testing process. Predictive analyses of software reliability model is of great importance for modifying, debugging and determining when to terminate software development testing process. This study performed one-sample Bayesian predictive analyses for Musa – Okumoto software reliability model using informative and non – informative priors. The study mainly focused on four issues on single-sample case that have been outlined in chapter three as issue A, B, C and D that relate to software development testing process. Simulated and secondary data were used to illustrate these issues. For secondary data, Goodness of Fit (GOF) test based on Laplace statistics was performed to check whether the model fit well to the data before it was used to illustrate the derived methodologies and It was found fit well to the data. The study developed explicit solutions to the issues and on issue D coverage probability was computed and found to be a good estimator and thus it will help to solve problems related to reliability of developed software and make a trade-off decision in software industry.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS AND ACRONYMNS

AIAA	America Institute of Aeronautics and Astronautics
CP	Coverage Probability
CPIT	Conditional Probability Integral Transformation
GOF	Goodness of Fit
HPP	Homogeneous Poisson Process
IEEE	Institute of Electrical and Electronics Engineers'
MCMC	Monte Carlo Markov Chain
ML	Maximum Likelihood
MLE	Maximum Likelihood Estimate
MSE	Mean Square Error
NHPP	Non homogeneous Poisson Process
NR	Newton Raphson
PLP	Power Law Process
SRGM	Software Reliability Growth Model
UPL	Upper Prediction Limit

LIST OF SYMBOLS

$\boldsymbol{m}(\boldsymbol{t})$	Mean value function, mean number of failures up to time *t*.
$\boldsymbol{N}(\boldsymbol{t})$	Cumulative number of software failures observed by time *t*.
$\boldsymbol{n}$	Total number of software failures
$\boldsymbol{P}(\boldsymbol{\theta} \mid \boldsymbol{y})$	Bayesian posterior density
$\boldsymbol{P}(\tilde{y} \mid \boldsymbol{y})$	Bayesian posterior predictive distribution of $\tilde{y}$
$\boldsymbol{t_i}$	Time up to the i^{th} failure
$\boldsymbol{\lambda}_{(\boldsymbol{t})}$	Failure intensity function at time *t*
$\boldsymbol{\alpha}$	Scale parameter for the Musa- Okumoto software reliability model
$\boldsymbol{\beta}$	Shape parameter for the Musa- Okumoto software reliability model

CHAPTER ONE
INTRODUCTION

1.1 Background information

Software has become a driver for nearly everything in the 21st century from elementary education to genetic engineering. Thus due to high dependency, the size and complexity of computer systems has grown and this poses a great problem in its reliability as failures are prone to happen during their operations. To avoid the failures and faults, reliability of software needs to be studied during development of software so as to come up with reliable software. Computer software have of late been applied in different fields ranging from automotive mechanical and safety control systems, industrial and quality control processes, hospital healthcare and air traffic control systems among others affecting millions of people. Reliability of software is of a lot of concern to the developers.

Software reliability is defined as the probability of failure free software operations for a specified period of time in a specified environment (Nuria, 2011). With the increasing need of software with zero defects, predicting reliability of software systems is gaining more and more importance (Sonia and Renu, 2014). Software reliability is achieved through testing during the software development stage (Daniel and Hoang, 2001). Software Reliability modeling is done to estimate the form of the curve of the failure rate by statistically estimating the parameters associated with the selected model. In most cases, the reliability development of a complex system often take place by testing a system until it fails, then making repairs and design changes and testing it again. This process continues until a desired level of reliability is achieved (Muralidharan *et al.*, 2008). The purpose of this measure is to estimate the extra execution time during test required to meet a specified reliability objective and to identify the expected reliability of the software when the product is released. During reliability modeling, the software systems are tested in an environment that resembles to the operational environment (Ullah *et al*., 2013).

Developing reliable software is one of the most difficult problems facing the software industry. Schedule pressure, resource limitations, and unrealistic requirements can all negatively impact software reliability. Developing reliable software is especially hard when there is interdependence among the software modules as is the case with much of existing software. It had been found to be a hard problem to know whether or not the software being delivered is reliable due to lack of a well-developed predictive method. After the software is shipped, its reliability is indicated by customer feedback- problem reports, system outages, complaints or compliments, and so forth. However, by then it is too late; software vendors need to know whether their products are reliable before they are shipped to customers. Software reliability growth models attempt to provide that information.

According to Meth (1992), software reliability modeling can provide the basis for planning reliability growth tests, monitoring progress, estimating current reliability, forecasting and predicting future reliability improvements. This is due to the fact that software reliability growth model is a powerful tool for prediction and forecasting. Its predictive analyses are useful for determining when to terminate a development process (Jun-Wu *et a*l., 2007). Usually, upper predictive interval is constructed to indicate the time frame when the *kth (k>0)* future observation will occur with pre-determined confidence level. These predictive intervals are often utilized in software industry in that they provide useful information for software developers to decide the optimal software release time and to refine the quality of software testing tasks.

Over the past decades many software reliability models that can be used for predictive analyses have been developed by different authors; (Jun- Wu *et al*., 2007, Akuno *et al.,* 2014). The Musa – Okumoto reliability model had not been applied to predictive analyses. Musa and Okumoto (1984) developed this model as they were looking for a model with high predictive validity. The Musa – Okumoto software reliability model is one of non-homogeneous Poisson process software model. The model is based on the assumptions that failures are observed during execution time caused by remaining faults in the software; whenever a failure is observed, an instantaneous effort is made to find what caused the failure and the faults are removed prior to future tests and whenever a repair is done it reduces the number of future faults not like other models. The failure intensity function of this model reduces exponentially with time and the expected number of failures has logarithmic function and thus also referred to as Musa- Okumoto logarithmic Poisson model. The model is an infinite failure model. Whenever predictive analyses are done using the above model, the future failures are expected to depict the above features.

The model must remain stable during the entire testing period for any particular testing environment and a reasonably accurate prediction of reliability must be provided by the model. These are the two main aspects of a good reliability model (Kapur *et al*., 2011). The

Musa – Okumoto (1984) model has been used in various testing environment and in many instances, it provides good estimation and prediction of software reliability. Compared to other models when used in testing industrial data set, Musa- Okumoto model is the best performer in terms of fitting and predictive capability to the data (Ullah *et al*., 2013).

Bayesian reliability modeling is one of the best methods in predictive analysis. Development of reliability posterior distribution from which predictive inference is made is the main thing required in Bayesian reliability model. The reliability posterior distribution is usually constructed using prior distribution for the parameters of the software reliability model. The advantage of using Bayesian approach is that it allows prior information such as engineering judgments and test results to be combined with more recent information from test or field data. This is vital since it helps software developers to arrive at a prediction of reliability based upon a combination of all available information. This information includes; the environment under which the software will work, previous tests on the software and even intuition based upon experience (Allan, 2012).

1.2 Statement of problem

Reliable software has been the main goal of any software developer. This is because non-reliable software means that the customers will be dissatisfied with the product thus loss of market shares and significant cost to the supplier. For critical applications such as banking or health monitoring, non reliability can lead to great damage not only to the consumer but also to the developer. Due to the above reasons, there is need to develop reliable software. There are many software reliability growth models that have been used in analyzing software reliability data. Musa- Okumoto is one of the software reliability models which best performed in fitting industrial failure data set. Parameter estimation of Musa-Okumoto (1984) software reliability model has been done using maximum likelihood method. Bayesian predictive analyses for Musa – Okumoto model had not been explored. This study explored Bayesian predictive analyses of Musa-Okumoto reliability model. The research only considered one- sample case with non- informative priors and informative priors.

1.3 Objectives

1.3.1 General objective

To carry out a study on Bayesian predictive analyses for Non-homogeneous Poisson process (NHPP) in software reliability with Musa- Okumoto intensity function.

1.3.2 Specific objectives

(i) To derive one- sample posterior and Bayesian predictive distribution of NHPP in software reliability with Musa – Okumoto intensity function using non – informative and informative priors.

(ii) To derive upper limit credible sets for one – sample and evaluate their coverage probabilities using simulated data on Musa – Okumoto software reliability model.

(iii) To apply one- sample Bayesian predictive analyses for Musa – Okumoto software growth model on real data.

1.4 Assumptions

(i) During testing, the software is executed in a manner similar to the anticipated operational usage.

(ii) Given $0 < t_1 < t_2 < \cdots < t_n < T$ to be software failure times where T is the time truncated, we assume that $t_n = T$.

(iii) Initial operational profile test of the software must have been done.

1.5 Justification

Software is one of the complex intellectual products. During its development, it is inevitable that some errors are made during its formulations as well as during designing, coding and testing the product. Development process of this software includes efforts to discover and correct faults resulting from errors. Dealing with faults cost money as well as impacting on development schedule and system performance. Consequently, there can be too much as well as too little effort spent dealing with faults, thus the system engineer needs the knowledge of software reliability predictive models to understand the current status of the system and make trade – off decision. This is because, if the decision is not made early enough it will affect the stakeholders, managers, developers and end- users. The developers are then required to deliver reliable software with acceptable level of quality within given budget and schedule. For these to happen, there is need for software developers to go for that procedure that minimizes the cost of software development and at the same time guarantees the reliability of the software. This can be achieved through software predictive analysis procedure on the Musa- Okumoto software reliability model, a logarithmic Non- Homogeneous Poisson Process. With this software reliability model, software developers are able to achieve their desire of high quality, cost effective and reliable software to be released to the market early enough.

1.6 Outline of the Thesis

This thesis has five chapters; chapter one, chapter two, three, four and five where chapter one presents background information, objectives and statement of the problem. Chapter two presents literature review, three has materials and methods. In chapter three source and how the data was analyse is presented. Chapter four presents derivation of posterior predictive distributions and real data analyses on the developed methodologies while chapter five has conclusion and recommendation.

CHAPTER TWO
LITERATURE REVIEW

2.1 Software Reliability

Software reliability is defined as the probability of failure free software operations for a specified period of time in a specified environment (Nuria, 2011). Usually software reliability analysis is performed at various stages during the process of software engineering in order to evaluate if the reliability requirements have been achieved. There are two activities that relate to software reliability analysis: Estimation and prediction. In both activities statistical inference techniques and reliability models are applied to failure data obtained from testing or during operation to measure software reliability. According to Ganesh (2002), estimation in most cases is retrospective and it is performed to determine achieved reliability from a point in the past to the present time while prediction activity parameterizes reliability models used for estimation and utilizes the available data to predict the future reliability.

2.2 Categories of Software Reliability models

In literature software reliability have been conducted for the last over 40 years and many models have been proposed for the estimation and prediction of software reliability. There exist some classification systems of software reliability models. Model classifications are helpful for identifying similarity between different models and to provide ideas when selecting an appropriate model. They are classified into the following categories: Markov models and non-homogeneous Poisson process (NHPP) models. Among these models, NHPP models are straightforward to implement in real-world applications.

- Markov models: A model belongs to this class if its probabilistic assumption of the failure process is essentially a Markov process. In these models, each state of the software has a transition probability associated with it that governs the operational criteria of the software.

- Non-homogeneous Poisson process models: A model is in this class if the main assumption is that the failure process is described by a non-homogeneous Poisson process. The main characteristic of this type of model is that there is a mean value function that is defined by the expected number of failures up to a given time. NHPP models are Cox-Lewis (1996) model with intensity $\lambda(t) = e^{\alpha+\beta t}$, the Goel-Okumoto (1979) model with $\lambda(t) = \alpha\beta e^{-\beta t}$, the delayed S-shaped model Yamada *et al*. (1983) with $\lambda(t) = \alpha\beta^2 t e^{-\beta t}$ and Musa-Okumoto (1984) model with $\lambda(t) = \alpha \,/\, (t+\beta)$

Jun – Wu *et al.* (2007). This study focuses on Musa – Okumoto software reliability model.

2.3 Counting Process

A sequenced collection of random variables in a given system is called a stochastic process and when the focus is on counts, the process is called a counting process and is denoted by $N(t)$, $t \geq 0$. Therefore, a counting process is the count of the number of events that occur in any time interval. A stochastic process $\{N\ (t), t \geq 0\}$ is said to be a counting process if $N(t)$ represent the total number of "events" that occurred up to the time t.

A counting process $\{N(t), t > 0\}$ must satisfy:

(i) $N(t) \geq 0$

(ii) $N(t)$ is integer valued

(iii) If $s < t$, then $N(s) \leq N(t)$

(iv) For $s < t\ , N(t) - N(s)$ equal the number of events that have occurred in the interval (s , t).

A counting process is said to possess stationary increment if the number of events which occur in any interval of time depends only on the length of the time interval. In other words, the process has the stationary increment if the number of the events in the intervals $(t_1 + s, t_2 + s)$ $\{N(t_2 + s) - N(t_1 + s)\}$ has the same distribution as the number of events in the interval (t_1,t_2) $\{N(t_2) - N(t_1)\}$ for all $t_2 > t_1$ and $s>0$. Also it is said to possess independent increment if the number of events which occur in disjoint time intervals are independent. If this property of independent increment is achieved then the counting process is a Poisson process (Sheldon, 1997).

A software system receives different types of input each with its own different path through the software; this will lead to creation of a capability of bringing different errors into light (Jelinski and Moranda, 1972). The different input types are viewed as arriving randomly to the software leading to detection of errors in a random way. These results will mean that there is an underlying random process that governs the software failures and thus justifies the use of stochastic methods to model software failures (Singpurwala and Simon, 1994). There are some probabilistic models describing the counting process. These are homogeneous and non-homogeneous Poisson processes. The following definitions are given in terms of software failure as that is the focus of the study.

2.3.1 Poisson process

A counting process $\{N(t), t > 0\}$ is said to be a Poisson process if;

(i) $N(0) = 0$

(ii) For any time points $t_0 = 0 < t_1 < t_2 < \cdots < t_n$ the random variables $N(t_0, t_1], N(t_1, t_2], \cdots, N(t_{n-1}, t_n]$ are independent random variables. This is called the independent increment property.

(iii) There is a function λ such that $\lambda(t) = \lim_{\Delta t \to 0} \frac{\Pr[N(t+\Delta t) - N(t) \geq 1]}{\Delta t}$

(iv) $\lambda(t) = \lim_{\Delta t \to 0} \frac{\Pr[N(t+\Delta t) - N(t) \geq 2]}{\Delta t} = 0$. This property forestalls the possibility of simultaneous failures.

The above properties (i) to (iv) of Poisson process imply that

$$\Pr[N(t) = n] = \left(\int_0^t \lambda(x)dx\right)^n \exp\left(-\int_0^t \lambda(x)dx\right). \tag{1}$$

2.3.2 Homogeneous Poisson process

As noted by Zhao (2004), that counting process is $N(t), t > 0$ is said to be a homogeneous Poisson process (HPP) if the intensity function $\lambda(t)$ is constant, i.e $\lambda(t) = \lambda, \lambda > 0$ and

(i) $N(0) = 0$. The failure at time zero.

(ii) The process has independent increment and stationary increment. Which is one of the property of a counting process.

(iii) The number of events occurring in any interval of length $t = t_2 - t_1$ has a Poisson distribution with mean λt, that is

$$\Pr[N(t_2) - N(t_1) = n] = \frac{e^{-\lambda t}(\lambda t)^n}{n!} \;;\; 0 \leq t_1 \leq t_2, n = 0,1. \tag{2}$$

A homogeneous Poisson process (HPP) has the following properties (Rigdon and Basu, 2000)

(i) A process is a HPP with constant intensity function λ if and only if times between events are identically and independent distributed exponential random variables with mean $1/\lambda$.

(ii) If $0 < T_1 < T_2 < \cdots < T_n$ are the failure times from a HPP, then the joint probability distribution of $T_1, T_2, \cdots, T_n$ is

$$f(t_1, t_2, \cdots, t_n) = \lambda^n e^{-\lambda t_n} \quad , \quad 0 < t_1 < t_2 < \cdots < t_n \tag{3}$$

(iii) The time to the n^{th} failure from a system modeled by a HPP has a gamma distribution with parameters$\alpha = n, \beta = {}^{1}/_{\lambda}$.

(iv) For a HPP, condition on $N(t) = n$, the failure times $0 < T_1 < T_2 < \cdots < T_n$ are distributed as order statistics from uniform distribution in the interval$(0, t)$.

(v) The probability of a system failure after time t is

$R(t) = \Pr[T > t] = \Pr[N(t) = 0] = e^{-\lambda t}$.

2.3.3 Non - Homogeneous Poisson process

A NHPP is a Poisson process whose intensity function is not constant (Zhao, 2004). A counting process $(N(t) > 0, t > 0)$ has a NHPP if

(i) $N(0) = 0$

(ii) The process has independent increment.

(iii) The number of failures in any interval $(t_1, t_2]$ has Poisson distribution with mean $\int_{t_1}^{t_2} \lambda(t)dt$, that is

$$\Pr[N(t_2) - N(t_1) = k] = \frac{1}{k!}\left(exp\left(-\int_{t_1}^{t_2} \lambda(t)dt\right)\left(\int_{t_1}^{t_2} \lambda(t)dt\right)^k\right) k = 0,1,2,3,... \quad (4)$$

The following are the properties of non-homogeneous Poisson process (Rigdon and Basu, 2000).

(i) The joint pdf of the failure times $T_1, T_2, \cdots, T_n$ from a non-homogeneous Poisson process with intensity function $\lambda(t)$ is given by

$$f(t_1, t_2, \cdots, t_n) = \left(\prod_{i=1}^{n} \lambda(t_i)\right) exp\left(-\int_0^T \lambda(t)dt\right) \quad (5)$$

where T is the stopping time: $T = t_n$ for the failure truncated case while T= t for the time truncated case. Equation (5) is also known as likelihood function.

(ii) If $0 < t_1 < t_2 < \cdots$ are the epoch at which the failure times occurs, then the time between occurrence intervals $T_k = t_k - t_{k-1}, (k = 1,2, \cdots)$ are independent random variables and with densities.

$$f_{t_k}(t_k) = \lambda(t_k)\exp\left(-\int_{t_{k-1}}^{t_k} \lambda(t)dt\right). \quad (6)$$

Again, $(t_1, t_2, \cdots)$ is a Markov sequence with transition density given as

$$\Pr(t_k / t_{k-1}) = \lambda(t_k) exp\left(-\int_{t_{k-1}}^{t_k} \lambda(t)dt\right). \quad (7)$$

(iii) If the failure times of a non-homogeneous Poisson process with intensity function $\lambda(t)$ are $T_1 < T_2 < \cdots < T_n$ conditioned on $T_n = n$ for failure truncated case, then the random variables $T_1 < T_2 < \cdots < T_{n-1}$ are distributed as $n-1$ order statistics or when conditioned on $T_n = t$ for time truncated case, then random variables $T_1 < T_2 < \cdots < T_n$ are distributed as n order statistics from the distribution with density

$$f(t) = \frac{\lambda(t)}{\int_0^{T_0} \lambda(t)dt} \quad , \quad 0 \le t \le T_0 \qquad (8)$$

which reduces to uniform distribution over [0,T_0] when $\lambda(t) = \lambda$.

There are probabilistic models describing software reliability growth, these models are NHPP and one of it is Musa – Okumoto software reliability growth model.

2.4 Musa – Okumoto software reliability growth model

Musa – Okumoto model, also known as logarithmic model introduced by Musa and Okumoto (1984) is one of the extensively applied in software reliability analysis. Infact Musa himself demonstrated that the model is more accurate compared to exponential models Musa *et al*. (1987). The model is a rate function for NHPP, when failure occurs the failure intensity function decreases exponentially (Abdel-Ghaly *et al*., 1986, Lyu, 1996). The possible number of failures over time is a logarithmic function therefore it is called logarithmic Poisson. This reflect the view that as failure is detected and corrected it is expected that the number of failures remaining is less, that is correcting the failure reduces the chance of future failure occurring.

The following are assumptions of Musa – Okumoto software reliability growth model (Musa and Okumoto, 1984, Lyu, 1996).

(i) There is no failure observed at time $t = 0$ i.e $N(0) = 0$ with probability one.

(ii) The cumulative number of failures by time *t*, $N(t)$, follows a Poisson process.

(iii) The software is operated in a similar manner as that in which reliability predictions are to be made.

(iv) Every fault has the same chance of being encountered within a severity class as any other fault in that class.

(v) The failures, when the faults are detected, are independent.

(vi) For small time interval Δt the probabilities of one and more than one failure during $(t, t + \Delta t]$ are $\lambda(t)\Delta t + 0(\Delta t)$ and $0(\Delta t)$, respectively. where $\frac{0(\Delta t)}{\Delta t} \to 0$ as

$\Delta t \rightarrow 0$. It should be noted that the probability of no failure during $(t, t + \Delta t)$ is given by $1 - \lambda(t)\Delta t + 0(\Delta t)$. This implies that the model is Poisson.

(vii) The failure intensity will decrease exponentially with the expected number of failures experienced, that is, $\lambda(t) = \alpha\beta e^{-\frac{m(t)}{\alpha}}$ where $\alpha > 0, \beta > 0$.

From the above assumptions we can show that;

$$\Pr[N(t) = y] = \frac{\{m(t)\}^y e^{-m(t)}}{y!}, \quad y = 1,2,\cdots \tag{9}$$

$$m(t) = \alpha In(1 + \beta t) \tag{10}$$

$$\lambda(t) = \frac{\alpha\beta}{1+\beta t} \tag{11}$$

where $m(t)$ is the expected number of failures observed by time t and $\lambda(t)$ is the failure rate, also known as the intensity function. In this model,α is the expected number of failures to be observed eventually and β is the fault detection rate per fault. In this model, the number of faults to be detected is a random variable whose observed value is dependent on the test and other environmental factor.

Musa-Okumoto model consist of two components; the execution time and calendar time component which are useful for managers and engineers in expressing when a specified reliability goal is expected to be reached. This will help the designers in making a trade-off decision on when to stop the testing process and release the product to the consumers. Execution time is the best time domain for expressing reliability and is the most practical measure of the failure induction stress being placed on a program. The principle objective of a software reliability model is to forecast failure behavior that will be experienced when the program is operational.

There has been a lot of application of Musa- Okumoto software reliability growth model as it is one of the best predictive models, it belongs to the selected models in the America Institute of Aeronautics and Astronautics (AIAA) recommended practice standard on software reliability (Lyu, 1996, Malaiya and Denton, 1997). Musa- Okumoto model have also been used in software cost estimation models with high accuracy (Xia *et al.*, 2008, Nassif *et al.*, 2013, Nassif *et al.*, 2010). A critical review and categorization of software reliability have been done by many researchers (Yadav and Khan, 2009, Sheakh *et al.*, 2012)

Developing a reliable software is a challenging task facing software industry. This therefore calls for a method for checking whether the developed software is reliable or not. To determine when to terminate development process of a software there is need to carry out predictive analyses. Bayesian predictive analyses using various software reliability growth models has attracted a number of researches. For instance predictive analyses for the power law process (PLP) was developed, where most problems that relates to development process of software were solved using Bayesian approach Jun-Wu *et al*. (2007). Akuno et al. (2014) also solved the issues related to software development process by use of Bayesian approach for Goel- Okumoto software reliability growth model. Both models assume failures are finite, a software can be free of errors at a given time when all faults have been removed which might not happen in a real situation. It was interesting to note that predictive analyses for Musa – Okumoto software reliability growth model given that it had not been developed and the model assume failures to be infinite, which is true in real situation. The model assumes that the earlier faults that are removed have great impact than the remaining faults. The study explored the issues related to development process of software taking in consideration Musa – Okumoto software reliability growth model using Bayesian approach.

2.5 Bayesian method

Bayesian method owes its name to the fundamental role of Bayes' theorem. In Bayesian reasoning, uncertainty is attributed not only to data but also to the parameters. Therefore, all parameters are modelled by distributions. Before any data are obtained, the knowledge about the parameters of a problem are expressed in the prior distribution of the parameters. Given actual data, the prior distribution and the data are combined into the posterior distribution of the parameters. The posterior distribution summarizes our knowledge about the parameters after observing the data.

2.5.1 Prior distributions

The most controversial element in Bayesian method is choosing the prior distributions. It has been criticized for introducing subjective information; the use of a prior is purely an educated guess and can vary from one scientist to another. Getting the prior distributions is one of the procedures of developing the main goal of Bayesian statistical analysis to obtain the posterior distributions of the model parameter. The posterior distribution is defined as a weighted average between the knowledge about the parameters before data is observed, which is represented by the prior distribution and the information contained in the data about the unknown parameters which is represented by the likelihood function. Normally before a Bayesian analysis is conducted, the statistician needs to observe the data at hand and formulate or choose a probability model for the data. Once the data model is formulated, a Bayesian analysis requires the assertion of a prior distribution for the unknown parameters of the model. The prior distribution is usually viewed as representing the current state of knowledge or current description of uncertainty, about the model parameters prior to data being observed (Glickman and Van, 2007). Prior distributions are divided into two categories namely, informative and non-informative priors.

When the statistician uses his/her intuitive knowledge about the substantive problem at hand, possibly based on past data along with expert opinion to formulate a prior distribution that properly reflects his/her (and experts') beliefs about the unknown parameters of the model, that is an informative prior and thus this approach has always been criticized as it seems to be subjective and unscientific. However, it can be argued that if prior knowledge or information about the model parameters exists prior to observing data, then it would be unscientific not to include such knowledge or information into data analysis.

The other main approach to choosing a prior distribution is by using non-informative prior. This approach represents ignorance about the model parameters. This approach is also referred to as objective, vague, diffuse and sometimes, reference prior distribution. Choosing a non-informative prior distribution is an attempt towards objectivity as it involves acting as though no prior knowledge about the parameters exists before data is observed. This is achieved through assigning equal probabilities to all values of the parameters. The beauty of this approach is that it directly addresses the criticism of informative prior distributions as being subjectively chosen.

2.5.2 Bayes rule

When θ is a parameter and Y is a random variable, the probability statement about θ given y can be made when we first consider a model providing a joint probability distribution for θ and y (Andrew *et al.*, 1995). The joint probability mass or density function are written as a product of two densities that are often referred to as the prior distribution $\Pr(\theta)$ and the sampling distribution $\Pr(y \mid \theta)$ respectively, that is

$\Pr(\theta, y) = \Pr(\theta)\Pr(y \mid \theta)$.

Conditioning on the known value of y and using the basic conditioning property known as the Bayes rule, we obtain the posterior density as

$$\Pr(\theta / y) = \frac{\Pr(\theta, y)}{\Pr(y)}$$

$$= \frac{\Pr(\theta)\Pr(y / \theta)}{\Pr(y)} \quad (12)$$

where $\Pr(y) = \sum_{\theta} \Pr(\theta) \Pr(y|\theta)$, and the sum is over all possible values of θ and for the case of continuous θ, $\Pr(y) = \int \Pr(\theta)\Pr(y \mid \theta) d\theta$. An equivalent form of the posterior distribution above omits the factor $\Pr(y)$ that is independent of θ and with fixed y which is considered as a constant of proportionality yielding the unnormalized posterior density which is the right side of the equation $\Pr(\theta|y) \propto \Pr(\theta)\Pr(y \mid \theta)$. This expression encloses the technical core of Bayesian inference. The primary task of any specific application is to develop the model $\Pr(\theta, y)$ and perform the necessary computation to summarize $\Pr(\theta \mid y)$ in appropriate ways (Andrew *et al*., 1995).

2.5.3 Bayesian predictive inference

Usually to make inference about an unknown observable, often called predictive inference, it follows the same logic as in the Bayes' rule. Andrew *et al.* (1995) shows that before the data y are considered, the distribution of the unknown but observable y is

$$\Pr(y) = \int \Pr(y, \theta) d\theta = \int \Pr(\theta)\Pr(y/\theta) d\theta \quad (13)$$

where $\Pr(y)$ is the prior predictive distribution which is also known as marginal distribution of the data. This is usually the integral of the likelihood function with respect to the prior distribution and the distribution is not conditional on observed data. After the data y has been observed, we can predict an unknown observable, $\tilde{y}$, from the same process. The distribution of $\tilde{y}$ is called the posterior predictive distribution, as it is the distribution of unobserved observation (prediction) conditional on the observed data. Thus the posterior predictive distribution of $\tilde{y}$ is given as;

$$\Pr(\tilde{y} / y) = \int \Pr(\tilde{y}, \theta / y) d\theta$$

$$= \int \Pr(\tilde{y} / \theta, y) \Pr(\theta / y) d\theta$$

$$= \int \Pr(\tilde{y} / \theta) \Pr(\theta / y) d\theta . \quad (14)$$

The second and third lines display the posterior predictive distribution as an average of conditional predictions over the posterior distribution of θ. The last step follows from the assumed conditional independence of y and $\tilde{y}$ given θ.

Bayesian methods have been applied to many areas for example; used as a basis for computation for the superposition of non-homogeneous Poisson processes (Tae and Lynn, 1999), also been applied on reliability growth models based on the power law process and also in the study of micro arrays (Zhao, 2004), Compodónico and Singpurwalla (1994) applied Bayesian estimation to Musa- Okumoto model. Bayesian predictive analyses on the power law process (PLP) using non – informative priors have also been conducted (Jun – Wu *et al*., 2007). Also Bayesian predictive analyses on software reliability growth model with Goel- Okumoto intensity function has been conducted (Akuno *et al*., 2014). In literature, Bayesian predictive analyses based on Musa – Okumoto software model is conspicuously missing. This therefore means Bayesian predictive analyses based on the model have not been explored and this study perform one-sample Bayesian predictive analyses based on Musa – Okumoto software reliability model.

2.6 Prediction intervals

A prediction interval is a confidence interval for a future observation or a function of some future observations (Jun – Wu *et al.*, 2007). Specifically, a double-sided (bilateral) prediction interval for x_{n+k} a future failure time with confidence level γ is defined by $[X_{n+k,l(\gamma)}, X_{n+k,u(\gamma)}]$ where $X_{n+k,l(\gamma)}$ and $X_{n+k,u(\gamma)}$ are the lower and upper prediction limits respectively such that

$$Pr\{X_{n+k,l(\gamma)} \leq x_{n+k} \leq X_{n+k,u(\gamma)}\} = \gamma.$$

Similarly, a single-sided (unilateral) lower or upper prediction limit for x_{n+k} with level γ is defined by $X_{n+k,L(\gamma)}$ (or $X_{n+k,U(\gamma)}$), which satisfies $Pr\{X_{n+k,L(\gamma)} \leq x_{n+k}\} = \gamma$ or $Pr\{x_{n+k} \leq X_{n+k,U(\gamma)}\} = \gamma$. Both lower and upper prediction limits, $X_{n+k,L(\gamma)}$ and $X_{n+k,U(\gamma)}$ respectively, depend only on a single sample (or a single software) and are called single-sample prediction limits. Prediction limits involving two samples (or two software) can be defined similarly and are called two-sample prediction limits.

2.7 Goodness-of-fit test (GOF) based on Laplace Statistic

Software reliability analysis is an important activity that software developers usually undertake in order to make a trade-off decision for their product. NHPP provides many models for software analysis. For one to get an appropriate NHPP model for a given failure data, GOF test have to be carried out. In literature there are many GOF tests that have been developed for NHPP models. Most of these tests have been carried out on PLP. Other tests such as conditional probability integral transformation (CPIT) require the model to have non-trivial sufficient statistics. GOF test based on the Laplace statistic has been found to be most powerful test in a large number of NHPP models with intensity function of the form $\alpha\lambda(t_i,\beta)$ (Zhao and Wang, 2005).

Suppose a repairable system is observed for τ unit of time and that the number of failures that occurred between the time interval 0 and t is denoted by $N(t)$. The failure times are denoted by $0 < t_1 < t_2 < \cdots < t_n < \tau$ and the data are said to be time-truncated. If the failure times $t_1, t_2 \cdots$ are NHPP with intensity function $\lambda(t)$ from a well-known property of NHPP, the transformed stochastic process $\Lambda(t_1), \Lambda(t_2), \cdots$ is HPP with parameter 1, where $\Lambda(t) = \int_0^t \lambda(u)du$ is cumulative intensity function corresponding to $\lambda(t)$. The problem at hand was to check if the model is suitable for the given data. This is taken as the problem of GOF with the null hypothesis:

H_0 : the failure process is a NHPP with intensity function $\alpha\lambda(t_i,\beta)$.

where α and β are unknown parameters. For this case we choose the famous Laplace test statistic.

$$S_n = \sqrt{\tfrac{12}{n}} \sum_{i=1}^{n} \left(\frac{\Lambda(t_i,\beta_0)}{\Lambda(\tau,\beta_0)} - \frac{1}{2} \right). \tag{15}$$

where β_0 is the true value of parameter β which in our case is not there and therefore must be estimated from the data using NR iterative method. Parameter α disappears in Laplace test statistic and thus it is not necessary to know α. Let $\hat{\beta}_n$ be the MLE of β, if $\hat{\beta}_n$ is used in place of β_0 in equation (15), we get;

$$S_n = \sqrt{\tfrac{12}{n}} \sum_{i=1}^{n} \left(\frac{\Lambda(t_i,\hat{\beta}_n)}{\Lambda(\tau,\hat{\beta}_n)} - \frac{1}{2} \right). \tag{16}$$

The test at $\alpha -$level is: Reject H_0 if

$$|S_n| > z_{\frac{\alpha}{2}} \sqrt{\delta(\hat{\beta}_n)} \ . \tag{17}$$

where $z_{\frac{\alpha}{2}}$ is the upper $\frac{1}{2}\alpha^{th}$ quartile of the standard normal distribution. This is so because, (Zhao and Wang, 2005) showed that

$$S_n \xrightarrow{d} N(0, \delta(\beta)) \tag{18}$$

where $\delta(\beta) = 1 - \dfrac{12}{\Lambda^2(\tau,\beta)} \left(\dfrac{1}{2} \dfrac{\partial \Lambda(\tau,\beta)}{\partial \beta} - E \dfrac{\partial \Lambda(u,\beta)}{\partial \beta} \right)^2 I^{-1}(\beta)$.

2.8 Summary

From literature, it is evident that a lot of work have been done by researchers on Musa – Okumoto model. They are majorly on parameter estimation using MLE and Bayesian approach. However, there is no literature on both classical and Bayesian prediction on Musa –Okumoto model and this study has presented a single-sample Bayesian predictive analyses on the model when non-informative and informative priors are considered.

CHAPTER THREE
MATERIALS AND METHODS

3.1 Introduction

In this chapter we present methodologies and materials that were used in this study. First we present the four issues associated with software development testing then the source of data that will be used in illustration, simulation study and finally how the data was analysed.

3.2 Predictive issues and Bayesian Approach

The study was limited to the development procedures that were used to address four issues in single sample prediction associated with software development testing program. The coverage probability of the credible set was also addressed by the study.

The issues that were addressed in one- sample development program are:

A: What is the probability that at most k software failures will occur in the future time period $(T, \tau]$ with $\tau > T$?

B: Given that the pre-determined target value λ_{tv} for the failure rate of the software undergoing development testing is not achieved at time T, what is the probability that the target value λ_{tv} will be achieved at time $\tau, \tau > T$?

C: Suppose that the target value λ_{tv} for the software failure rate is not achieved at time T, how long will it take so that the software failure rate will be attained at λ_{tv}?

D: What is the upper prediction limit (UPL) of $\lambda_{\tau} = \alpha\beta \,/\, (1 + \beta\tau)$ with level γ. τ being a pre-determined value greater than T?

The study adopted Bayesian approaches based on informative and non-informative priors to develop predictive distribution and derive explicit solutions to the four problems mentioned above. Informative and non-informative prior distributions for both parameters α and β were adopted by the study. For the case of informative prior distributions, the study assumed that the parameters α and β both follow a gamma distribution with parameters a, b and c, d respectively where *a*, *b*, *c* and *d* are known, i.e- $\alpha \sim Gamma\ (a, b)$ and $\beta \sim Gamma\ (c, d)$.

3.3 Source of data

Secondary software failure data was used to illustrate the methodologies that were developed. The study used secondary software failure data in the form of execution times between successive failures from one software system (Xie *et al.*, 2002). The study assumed that the failure times follow the NHPP with intensity function given in Equation (11). Before the data was applied to the model, Goodness of fit test was performed to check whether the model fits well to the data. The study used Laplace test statistics to check goodness of fit for the model.

3.4 Simulation Study

Simulated data from Musa- Okumoto model (1984) was also used to illustrate the methodologies that was developed and to evaluate the coverage probability of the credible sets. The simulated data was assumed to follow NHPP with intensity given in equation (11). The study adopted the thinning method for simulation using the following algorithm:

Step 1: Generate points in the NHPP N(t) with intensity function $\lambda(t)$ in the fixed interval $(0, t_0)$. If the number of points generated, n, is such that $n = 0$, exist. There are no points in the process $N(t)$

Step 2: Denote the ordered points by $T_1, T_2, ..., T_n$. Set $i = 1$ and $k = 0$

Step 3: Generate U_i, uniform distribution between 0 and 1. If $U_i \leq \lambda(T_i) / \lambda^a st(T_i)$, set $k = k + 1$ and $T_k = T_i$

Step 4: Set $i = i + 1$. If $i \leq n^a st$, then go to step 3.

The Programs for R- language (version 3.4.0) were developed to help in simulation of the data.

3.5 Data analysis

The study used both secondary and simulated software failure data. The analysis of both secondary and simulated data was done using a Statistical package called R-language version 3.4.0. For cases where closed forms of predictive distributions and predictive inferences for the single sample case for both informative and non-informative priors were unavailable, MCMC integration algorithm was used to compute predictive estimates. The program codes for obtaining predictive distributions and predictive inferences using simulated and secondary data were developed.

CHAPTER FOUR
RESULTS AND DISCUSION

4.1 Results for predictive issues

In this chapter four issues A, B, C and D as listed in chapter three associated with software development testing program have been presented. The four issues that were addressed are outlined as propositions and their proof given. Predictive distributions were derived using Bayesian method. In this thesis, it is assumed that a reliability growth testing is performed on a computer software system and the number of failures in the time interval $(0,t]$, denoted by $N(t)$ is observed. It is also assumed that $\{N(t), t>0\}$ follows the NHPP with intensity given in equation (11). Let $0<t_1<t_2<\cdots$ be the successive failure times. When testing stops after a pre-determined n number of failures is observed, the failure data is said to be failure-truncated. We denote the n failure times by $Y_{obs}^{f}=\left[t_i\right]_{i=1}^{n}$ where $0<t_1<t_2<\cdots<t_n$ a time-truncated data is when testing is observed for fixed time t. We denote the corresponding observed data by $Y_{obs}^{t}=\{n,t_1,\cdots,t_n;t\}$, where $0<t_1<\cdots<t_n\le t$.

Let $T \square \begin{cases} t_n\, if \quad the \quad observed \quad data \quad are \quad Y_{obs}^{f} \\ t\, if \quad the \quad observed \quad data \quad are \quad Y_{obs}^{t} \end{cases}$

Let Y_{obs} represent Y_{obs}^{f} or Y_{obs}^{t}. The joint density distribution of Y_{obs} is therefore (Crowder *et al.*, 1994):

$$f(Y_{obs}/\alpha,\beta)=(\alpha\beta)^n\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{(-\alpha\ln(1+\beta T))} \qquad \alpha>0,\beta>0 \qquad (19)$$

Case 1: β **, the shape parameter is known**, we adopt the following non – informative prior distribution for α :

$$\pi(\alpha)\propto\frac{1}{\alpha},\alpha>0 \qquad (20)$$

The posterior distribution of α can be obtained from equation (12) as;

$$h(\alpha/Y_{obs})=\frac{f(Y_{obs}/\alpha,\beta)\pi(\alpha)}{\int_0^\infty f(Y_{obs}/\alpha,\beta)\pi(\alpha)d\alpha}.$$

Substituting equation (19) and (20) we have

$$h(\alpha / Y_{obs}) = \frac{\alpha^{n-1}\beta^n \prod_{i=1}^{n}(1+\beta t_i)^{-1} e^{(-\alpha \ln(1+\beta T))}}{\int_0^\infty \alpha^{n-1}\beta^n \prod_{i=1}^{n}(1+\beta t_i)^{-1} e^{(-\alpha \ln(1+\beta T))} d\alpha} \tag{21}$$

Considering the denominator, we get:

$$\int_0^\infty \beta^n \alpha^{n-1} \prod_{i=1}^{n}(1+\beta t_i)^{-1} e^{(-\alpha \ln(1+\beta T))} d\alpha = \beta^n \prod_{i=1}^{n}(1+\beta t_i)^{-1} \int_0^\infty \alpha^{n-1} e^{(-\alpha \ln(1+\beta T))} d\alpha .$$

$$= \frac{\beta^n \prod_{i=1}^{n}(1+\beta t_i)^{-1}\Gamma(n)}{(\ln(1+\beta T))^n} \int_0^\infty \frac{(\ln(1+\beta T))^n}{\Gamma(n)} \alpha^{n-1} e^{(-\alpha \ln(1+\beta T))} d\alpha \tag{22}$$

$$= \frac{\beta^n \prod_{i=1}^{n}(1+\beta t_i)^{-1}\Gamma(n)}{(\ln(1+\beta T))^n} . \tag{23}$$

The integral part of equation (22) integrates to 1 since the integrand is gamma pdf with parameters n and $\ln(1+\beta T)$. Thus equation (22) reduces to equation (23). Equation (21) therefore becomes $h(\alpha / Y_{obs}) = \frac{\alpha^{n-1}\beta^n \prod_{i=1}^{n}(1+\beta t_i)^{-1} e^{(-\alpha \ln(1+\beta T))}}{\beta^n \prod_{i=1}^{n}(1+\beta t_i)^{-1}\Gamma(n) / (\ln(1+\beta T))^n}$ which reduces to;

$$h(\alpha / Y_{obs}) = [\Gamma(n)]^{-1} \alpha^{n-1} (\ln(1+\beta T))^n e^{(-\alpha \ln(1+\beta T))} . \tag{24}$$

Let $\tilde{y}$ be the random variable being predicted. The predictive density of $\tilde{y}$ from equation (13) is;

$$f(\tilde{y} / Y_{obs}) = \int_0^\infty f(\tilde{y} / Y_{obs}) h(\alpha / Y_{obs}) d\alpha . \tag{25}$$

Hence, the Bayesian UPL of $\tilde{y}$ with level γ, denoted as $y_U^{(\beta)}$, must satisfy

$$\gamma = \int_{-\infty}^{y_U^{(\beta)}} f(y / Y_{obs}) dy . \tag{26}$$

Case 2: **The shape parameter β is unknown**; the study considered the following joint prior distribution of α and β where both parameters are assumed to be independent

$$\pi(\alpha,\beta) \propto \frac{1}{\alpha\beta}, \alpha, \beta > 0. \tag{27}$$

Thus the corresponding joint posterior distribution for α and β is given as;

$$h(\alpha,\beta / Y_{obs}) = \frac{\alpha^{n-1}\beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{(-\alpha\ln(1+\beta T))}}{\int_0^\infty\int_0^\infty \alpha^{n-1}\beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{(-\alpha\ln(1+\beta T))}d\alpha d\beta}. \tag{28}$$

Considering the denominator of equation (28) we obtained $\Gamma(n)\int_0^\infty \beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}\left[\ln(1+\beta T)\right]^{-n} d\beta$ and substituting it equation (28) we have;

$$h(\alpha,\beta / Y_{obs}) = \frac{\alpha^{n-1}\beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{(-\alpha\ln(1+\beta T))}}{\Gamma(n)\int_0^\infty \beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}(\ln(1+\beta T))^{-n} d\beta} \tag{29}$$

The integral part of the denominator of equation (29) does not have closed form and thus the study employed the MCMC integration algorithm to obtain its value. The value obtained from the MCMC integration is denoted by a constant value k, hence

$k = \int_0^\infty \beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}(\ln(1+\beta T))^{-n} d\beta$. Therefore equation (29) reduces to;

$$h(\alpha,\beta / Y_{obs}) = [k\Gamma(n)]^{-1}\alpha^{n-1}\beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{(-\alpha\ln(1+\beta T))}. \tag{30}$$

Equation (30) is similar to equation (24), let $\tilde{y}$ be the random variable to be predicted. The predictive density of $\tilde{y}$ from equation (13) is;

$$f(\tilde{y} / Y_{obs}) = \int_0^\infty\int_0^\infty f(\tilde{y} / Y_{obs}, \alpha, \beta) h(\alpha,\beta / Y_{obs}) d\alpha d\beta \tag{31}$$

and the Bayesian UPL denoted by y_U of $\tilde{y}$ with level γ similar to equation (22) is;

$$\gamma = \int_{-\infty}^{y_U} f(\tilde{y} / Y_{obs}) dy. \tag{32}$$

4.2 Main results for prediction using non-informative priors

Preposition 4.2.1.

The probability that at most k failures will occur in the time interval $(T,\tau]$ with $\tau > T$ is

$$\gamma_k = \begin{cases} \dfrac{[\ln(1+\beta T)]^n}{\left[\ln\left(\dfrac{1+\beta\tau}{1+\beta T}\right)\right]^n} \displaystyle\sum_{j=n}^{n+k} \binom{j-n}{n-1} \dfrac{\left[\ln\left(\dfrac{1+\beta\tau}{1+\beta T}\right)\right]^j}{[\ln(1+\beta\tau)]^j} & if\, \beta\, is \quad known \\ \displaystyle\sum_{j=n}^{n+k} \dfrac{\Gamma(j)}{d(j-n)!\Gamma(n)} \int_0^\infty \dfrac{\beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}}{[\ln(1+\beta\tau)]^j} \left[\ln\left(\dfrac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n} d\beta & if\, \beta\, is \quad unknown \end{cases} \tag{33}$$

Proof

We first state the following identity without proof: That is

$$\int_{D(m;a,b)} dF(t_1)\cdots dF(t_m) = \left[F(b)-F(a)\right]^m / m!$$

where m is any positive integer, a and b are two real numbers such that $a<b$, $F(t)$ is an increasing and differentiable function and $D(m;a,b) \equiv \{(t_1,\cdots t_m): a<t_1<\cdots<t_m<b\}$.

The probability that at most k failures will occur in the interval $(T,\tau]$ is $\gamma_k = \Pr\{N(\tau)\le n+k / Y_{obs}\}$. When β is known, we have

$$\gamma_k = \int_0^\infty \Pr\{N(\tau)\le n+k / Y_{obs},\alpha\}.h(\alpha / Y_{obs})d\alpha \tag{34}$$

where $h(\alpha / Y_{obs})$ is given by equation (24) and

$$\Pr[N(\tau)\le n+k / Y_{obs},\alpha] = \sum_{j=n}^{n+k} f(Y_{obs}, N(\tau)=j/\alpha) / f(Y_{obs}/\alpha)\,. \tag{35}$$

From equation (19), we have $f(Y_{obs}/\alpha)=\beta^n\alpha^n\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{[-\alpha\ln(1+\beta T)]}$ and

$$f(Y_{obs},N(\tau)=j/\alpha)=\int_{D(j-n;T,\tau)}f(Y_{obs},x_{n+1},...,x_j,N(\tau)=j)\prod_{\ell=n+1}^{j}dx_\ell$$

$$=\int_{D(j-n,T,\tau)}\alpha^j\beta^j e^{[-\alpha\ln(1+\beta T)]}\prod_{i=1}^{j}(1+\beta t_i)^{-1}\prod_{\ell=n+1}^{j}dt_\ell$$

$$=\alpha^j\beta^j\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{[-\alpha\ln(1+\beta T)]}\int_{D(j-n,T,\tau)}\prod_{\ell=n+1}^{j}(1+\beta t_\ell)^{-1}\prod_{\ell=n+1}^{j}dt_\ell\ . \quad (36)$$

Solving the integral part of equation (36), we proceed as follows: $\int_0^t(1+\beta t)^{-1}dt=\frac{1}{\beta}\ln(1+\beta t)$

substituting the limits T and τ we obtain $\frac{1}{\beta}\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)$ and substituting it into integral part of equation (36) we have

$$\int_{D(j-n,T,\tau)}\prod_{\ell=n+1}^{j}(1+\beta t_\ell)^{-1}\prod_{\ell=n+1}^{j}dt_\ell=\frac{1}{\beta^{j-n}}\frac{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n}}{(j-n)!}\ . \quad (37)$$

Substituting equation (37) into equation (36), we have

$$f(Y_{obs},N(\tau)=j/\alpha)/\,f(Y_{obs}/\alpha)=\alpha^j\beta^j\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{[-\alpha\ln(1+\beta\tau)]}\frac{1}{\beta^{j-n}}\frac{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n}}{(j-n)!}$$

which reduces to

$$f(Y_{obs},N(\tau)=j/\alpha)/\,f(Y_{obs}/\alpha)=\frac{\alpha^{j-n}e^{\left[-\alpha\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]}\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n}}{(j-n)!}\ .$$

Thus equation (35) becomes

$$\Pr[N(\tau)\le n+k/Y_{obs}]=\frac{\sum_{j=n}^{n+k}\alpha^{j-n}e^{\left[-\alpha\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]}\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n}}{(j-n)!}\ . \quad (38)$$

And hence equation (34) becomes

$$\gamma_k = \int_0^\infty \sum_{j=n}^{n+k} \alpha^{j-n} e^{\left[-\alpha \ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]} \frac{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n} \alpha^{n-1}\left[\ln(1+\beta T)\right]^n e^{[-\alpha\ln(1+\beta T)]}}{(j-n)!\Gamma(n)} d\alpha$$

$$= \sum_{j=n}^{n+k} \frac{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n} \left[\ln(1+\beta T)\right]^n}{(j-n)!\Gamma(n)} \frac{\Gamma(j)}{\left[\ln(1+\beta\tau)\right]^j} \int_0^\infty \frac{\left[\ln(1+\beta\tau)\right]^j}{\Gamma(j)} \alpha^{j-n} e^{[-\alpha\ln(1+\beta\tau)]} d\alpha . \quad (39)$$

The integral part of equation (39) integrates to 1 since it is a gamma distribution with parameters j and $\ln(1+\beta\tau)$ and hence equation (39) reduces to

$$\gamma_k = \sum_{j=n}^{n+k} \frac{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n} \left[\ln(1+\beta T)\right]^n \Gamma(j)}{(j-n)!\Gamma(n)\left[\ln(1+\beta\tau)\right]^j} . \quad (40)$$

On re-arranging equation (40) we obtain

$$\gamma_k = \frac{\left[\ln(1+\beta T)\right]^n}{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^n} \sum_{j=n}^{n+k} \binom{j-n}{n-1} \frac{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^j}{\left[\ln(1+\beta\tau)\right]^j} . \quad (41)$$

This is the first formula of equation (33).

When β is unknown, noting that $\Pr\left\{N(\tau) \le n+k / Y_{obs}, \alpha, \beta\right\}$ and $h(\alpha, \beta / Y_{obs})$ are given by equation (38) and (30) respectively, we obtain

$$\gamma_k = \int_0^\infty \int_0^\infty \Pr\left[N(\tau) \le n+k / Y_{obs}, \alpha, \beta\right] . h(\alpha, \beta / Y_{obs}) d\alpha d\beta$$

$$= \sum_{j=n}^{n+k} \frac{1}{(j-n)!\Gamma(n)} \int_0^\infty \int_0^\infty \alpha^{j-1} \beta^{n-1} \prod_{i=1}^{n} (1+\beta t_i)^{-1} \left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n} e^{[-\alpha\ln(1+\beta\tau)]} d\alpha d\beta$$

$$= \sum_{j=n}^{n+k} \frac{\Gamma(j)}{d(j-n)!\Gamma(n)} \int_0^\infty \frac{\beta^{n-1} \prod_{i=1}^{n} (1+\beta t_i)^{-1}}{\left[\ln(1+\beta\tau)\right]^j} \left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n} d\beta . \quad (42)$$

Since the summation of k is from n to $n+k$ and $k's$ are not the same, we substitute letter k with d in equation (42) where $d = k$ as used in equation (30). Equation (42) implies the second formula in equation (33).

Preposition 4.2.2

The probability that the target value λ_{tv} will be achieved at time τ $(\tau > T)$ is

$$\gamma_k = \begin{cases} 1 - \sum_{h=0}^{n-1} \frac{\left[\left(\frac{1+\beta\tau}{\beta}\right)\lambda_\tau \ln(1+\beta T)\right]^h}{h!} e^{-\lambda_\tau\left(\frac{1+\beta\tau}{\beta}\right)\ln(1+\beta T)} \quad \text{if } \beta \text{ is known} \\ 1 - \frac{1}{k}\sum_{h=0}^{n-1} \int_0^\infty \frac{\left[\left(\frac{1+\beta\tau}{\beta}\right)\lambda_{tv} \ln(1+\beta T)\right]^h}{h!} \frac{\beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}}{\left[\ln(1+\beta T)\right]^n} e^{-\lambda_{tv}\left(\frac{1+\beta\tau}{\beta}\right)\ln(1+\beta T)} d\beta \quad \text{if } \beta \text{ is unknown} \end{cases} \tag{43}$$

Proof

Let $f(\lambda_\tau / Y_{obs})$ denote the posterior of $\lambda_\tau = \alpha\beta/(1+\beta\tau)$. Hence, the probability that the target value λ_{tv} will be achieved at time τ is given by

$$\gamma = \Pr\{\lambda_\tau \le \lambda_{tv} / Y_{obs}\} = \int_0^{\lambda_{tv}} f(\lambda_\tau / Y_{obs}) d\lambda_\tau \,. \tag{44}$$

When β is known, making transformation $\lambda_\tau = \alpha\beta/(1+\beta\tau)$, we have $\alpha = \lambda_\tau \frac{(1+\beta\tau)}{\beta}$ and $\frac{d\alpha}{d\lambda_\tau} = \frac{(1+\beta\tau)}{\beta}$. Consequently, the posterior density of λ_τ is

$$f(\lambda_\tau / Y_{obs}) = h(\alpha / Y_{obs})\left|\frac{d\alpha}{d\lambda_\tau}\right| \tag{45}$$

$$f(\lambda_\tau / Y_{obs}) = \frac{1}{\Gamma(n)}\left[\frac{\lambda_\tau(1+\beta\tau)}{\beta}\right]^{n-1}\left[\ln(1+\beta T)\right]^n e^{-\lambda_\tau\left(\frac{1+\beta\tau}{\beta}\right)\ln(1+\beta T)} . \frac{(1+\beta\tau)}{\beta}$$

$$= \frac{1}{\Gamma(n)}\left[\left(\frac{1+\beta\tau}{\beta}\right)\ln(1+\beta T)\right]^n \lambda_\tau^{n-1} e^{-\lambda_\tau\left[\left(\frac{1+\beta\tau}{\beta}\right)\ln(1+\beta T)\right]} . \tag{46}$$

From equation (46), it can easily be noted that λ_τ has gamma distribution with parameters n and $\frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)$. Noting that gamma and Poisson distributions have a relationship defined as

$$\frac{b^a}{\Gamma(a)}\int_0^\lambda x^{a-1}e^{-bx}dx = 1 - \sum_{h=0}^{a-1}\frac{(b\lambda)^h}{h!}e^{-b\lambda} \,. \tag{47}$$

By substituting equation (46)and (47) into equation (44), we obtain

$$\gamma = 1 - \sum_{h=0}^{n-1} \frac{\left[\frac{(1+\beta\tau)}{\beta} \lambda_\tau \ln(1+\beta T) \right]^h}{h!} e^{-\lambda_\tau \left(\frac{1+\beta\tau}{\beta} \right) \ln(1+\beta T)} . \tag{48}$$

Equation (48) implies the first formula of equation (43).

When β is unknown, making transformation $\lambda_\tau = \alpha\beta / (1+\beta\tau)$ and $\beta = \beta$, we obtain

$\alpha = \lambda_\tau \dfrac{(1+\beta\tau)}{\beta}$ and $\beta = \beta$. Note that the Jacobian is $\partial(\alpha,\beta) / \partial(\lambda_\tau,\beta) = (1+\beta\tau)/\beta$. From equation (30),

the joint posterior density of (λ_τ, β) is

$$f(\lambda_\tau, \beta / Y_{obs}) = h(\alpha, \beta / Y_{obs}) \left| \frac{d(\alpha,\beta)}{d(\lambda_\tau,\beta)} \right| . \tag{49}$$

$$f(\lambda_\tau, \beta / Y_{obs}) = \left[\frac{\lambda_\tau (1+\beta\tau)}{\beta} \right]^{n-1} \frac{\beta^{n-1} \prod_{i=1}^{n} (1+\beta t_i)^{-1}}{k\Gamma(n)} e^{-\lambda_\tau \frac{(1+\beta\tau)}{\beta} \ln(1+\beta T)} . \frac{(1+\beta\tau)}{\beta}$$

$$= \frac{\beta^{n-1} \prod_{i=1}^{n} (1+\beta t_i)^{-1}}{k\Gamma(n) \left[\ln(1+\beta T) \right]^n} \left[\frac{(1+\beta\tau)}{\beta} \ln(1+\beta T) \right]^n \lambda_\tau^{\,n-1} e^{-\lambda_\tau \frac{(1+\beta\tau)}{\beta} \ln(1+\beta T)} . \tag{50}$$

By substituting equation (47) and (50) into equation (44), we obtain

$$\gamma = \frac{1}{k} \int_0^\infty \frac{\beta^{n-1} \prod_{i=1}^{n} (1+\beta t_i)^{-1}}{\left[\ln(1+\beta T) \right]^n} \left\{ 1 - \sum_{h=0}^{n-1} \frac{\left[\lambda_\tau \frac{1+\beta\tau}{\beta} \ln(1+\beta T) \right]^h}{h!} e^{\left[-\lambda_\tau \frac{1+\beta\tau}{\beta} \ln(1+\beta T) \right]} \right\}$$

$$= \frac{1}{k} \int_0^\infty \frac{\beta^{n-1} \prod_{i=1}^{n} (1+\beta t_i)^{-1}}{\left[\ln(1+\beta T) \right]^n} d\beta - \frac{1}{k} \sum_{h=0}^{n-1} \int_0^\infty \frac{\left[\lambda_\tau \frac{(1+\beta\tau)}{\beta} \ln(1+\beta T) \right]^h}{h!} e^{-\lambda_\tau \frac{(1+\beta\tau)}{\beta} \ln(1+\beta T)} . \frac{\beta^{n-1} \prod_{i=1}^{n} (1+\beta t_i)^{-1}}{\left[\ln(1+\beta T) \right]^n} d\beta$$

$$=1-\frac{1}{k}\sum_{h=0}^{n-1}\int_0^{\infty}\frac{\left[\lambda_{\tau}\frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)\right]^h}{h!}e^{-\lambda_{\tau}\frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)}.\frac{\beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}}{\left[\ln(1+\beta T)\right]^n}d\beta \quad (51)$$

which is the second formula of equation (43).

Preposition 4.2.3

For a given level γ, the time τ^* required to attain λ_{tv} is

$$\tau^*=\begin{cases}\left[\frac{\chi^2(2n;\gamma)}{2\lambda_{tv}\ln(1+\beta T)}-\frac{1}{\beta}\right]-T & if\,\beta \quad is \quad known\\ \tau-T & if\,\beta \quad is \quad unknown\end{cases} \quad (52)$$

Remark 1: For the second part of equation (52), τ is the solution to the equation

$$\gamma=1-\frac{1}{k}\sum_{h=0}^{n-1}\int_0^{\infty}\frac{\left[\lambda_{\tau}\frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)\right]^h}{h!}e^{-\lambda_{\tau}\frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)}.\frac{\beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}}{\left[\ln(1+\beta T)\right]^n}d\beta\,. \quad (53)$$

Proof

For given level γ, the time required to attain the target value λ_{tv} is $\tau^*=\tau-T$, where τ satisfies equation (44).When β is known, from equation (46), it can easily be seen that

$2\left[\frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)\right]\lambda_{\tau}$ follows a chi-square distribution with $2n$ degrees of freedom. Thus we have

$$2\left[\frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)\right]\lambda_{tv}=\chi^2(2n,\gamma) \quad (54)$$

$$\frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)=\frac{\chi^2(2n;\gamma)}{2\lambda_{tv}}\,. \quad (55)$$

Making τ the subject, we obtain

$$\tau=\frac{\chi^2(2n;\gamma)}{2\lambda_{tv}\ln(1+\beta T)}-\frac{1}{\beta}\,. \quad (56)$$

From equation (56) we can obtain the time required to attain the target λ_{tv} with level γ as

$$\tau^* = \left[\frac{\chi^2(2n;\gamma)}{2\lambda_{tv}\ln(1+\beta T)} - \frac{1}{\beta} \right] - T \,. \tag{57}$$

Equation (57) implies the first formula of equation (52).

The time required to attain the target λ_{tv} with level γ when β is unknown is $\tau^* = \tau - T$ where τ is the solution to

$$\gamma = 1 - \frac{1}{k}\sum_{h=0}^{n-1}\int_0^{\infty} \frac{\left[\lambda_\tau \frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)\right]^h}{h!} e^{-\lambda_\tau \frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)} \cdot \frac{\beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}}{\left[\ln(1+\beta T)\right]^n} d\beta \,. \tag{58}$$

Preposition 4.2.4

The Bayesian UPL of $\lambda_\tau = \dfrac{\alpha\beta}{(1+\beta\tau)}$ with level γ is

$$\lambda_U^{(\beta)}(\tau) = \begin{cases} \dfrac{\beta\chi^2(2n;\gamma)}{2(1+\beta\tau)\ln(1+\beta T)} & if\, \beta \quad is \quad known \\ \lambda_{tv} & if\, \beta \quad is \quad unknown \end{cases} \tag{59}$$

Remark 2: For the second part of equation (59), λ_{tv} is the solution to

$$\gamma = 1 - \frac{1}{k}\sum_{h=0}^{n-1}\int_0^{\infty} \frac{\left[\lambda_{tv} \frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)\right]^h}{h!} e^{-\lambda_{tv} \frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)} \cdot \frac{\beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}}{\left[\ln(1+\beta T)\right]^n} d\beta \,. \tag{60}$$

Proof

For a pre-determined $\tau(\tau > T)$, the Bayesian upper prediction limit for $\lambda_\tau = \dfrac{\alpha\beta}{(1+\beta\tau)}$ with level γ is $\lambda_U^{(\beta)}(\tau)$ satisfying $\gamma = \Pr\left\{\lambda_\tau \le \lambda_U^{(\beta)}(\tau) / Y_{obs}\right\}$. From equation (44) and (54), we have

$$\gamma = \int_0^{\lambda_\tau^{(\beta)}(\tau)} \left[\frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)\right]^n \frac{\lambda_\tau^{n-1}}{\Gamma(n)} e^{-\lambda_\tau \frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)} d\lambda_\tau \,. \tag{61}$$

From equation (61) it can easily been seen that

$$2\left[\frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)\right]\lambda_U^{(\beta)}(\tau) = \chi^2(2n;\gamma) \,. \tag{62}$$

Making $\lambda_U^{(\beta)}(\tau)$ the subject in equation (62) we have

$$\lambda_U^{(\beta)}(\tau) = \frac{\beta\chi^2(2n;\gamma)}{2(1+\beta\tau)\ln(1+\beta T)} . \tag{63}$$

Equation (63) implies the first part of equation (59).

When β is unknown, the Bayesian UPL for $\lambda_\tau = \dfrac{\alpha\beta}{(1+\beta\tau)}$ with level γ is λ_{tv} where λ_{tv} is the solution to

$$\gamma = 1 - \frac{1}{k}\sum_{h=0}^{n-1}\int_0^\infty \frac{\left[\lambda_\tau \frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)\right]^h}{h!} e^{-\lambda_\tau \frac{(1+\beta\tau)}{\beta}\ln(1+\beta T)} . \frac{\beta^{n-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}}{\left[\ln(1+\beta T)\right]^n} d\beta . \tag{64}$$

4.3 Main results for prediction using informative priors

The joint density of Y_{obs} as in equation (19) will also be used in this section.

Case 1: When β the shape parameter is known, we adopt the following informative prior for α, that is $\alpha \sim Gamma(a,b)$, where a and b are known

$$\pi(\alpha) \propto \alpha^{a-1}e^{-b\alpha} . \tag{65}$$

The posterior of α is thus obtained from equation (12) as

$$h(\alpha / Y_{obs}) = \frac{f(Y_{obs} / \alpha,\beta)\pi(\alpha)}{\int_0^\infty f(Y_{obs} / \alpha,\beta)\pi(\alpha)d\alpha} .$$

Substituting equation (19) and equation (65) into equation (12) we have

$$h(\alpha / Y_{obs}) = \frac{\alpha^n\beta^n\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{-\alpha\ln(1+\beta T)}.\alpha^{a-1}e^{-b\alpha}}{\int_0^\infty \alpha^n\beta^n\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{-\alpha\ln(1+\beta T)}.\alpha^{a-1}e^{-b\alpha}d\alpha}$$

$$= \frac{\beta^n\prod_{i=1}^{n}(1+\beta t_i)^{-1}\alpha^{n+a-1}e^{-\alpha[\ln(1+\beta T)+b]}}{\int_0^\infty \beta^n\prod_{i=1}^{n}(1+\beta t_i)^{-1}\alpha^{n+a-1}e^{-\alpha[\ln(1+\beta T)+b]}d\alpha} . \tag{66}$$

Considering the denominator of equation (66)

$$\int_0^\infty \beta^n \prod_{i=1}^n (1+\beta t_i)^{-1} \alpha^{n+a-1} e^{-\alpha[\ln(1+\beta T)+b]} d\alpha = \frac{\beta^n \prod_{i=1}^n (1+\beta t_i)\Gamma(n+a)}{[\ln(1+\beta T)+b]^{n+a}}. \tag{67}$$

Thus equation (66) reduces to

$$h(\alpha / Y_{obs}) = [\Gamma(n+a)]^{-1} \alpha^{n+a-1} [\ln(1+\beta T)+b]^{n+a} e^{-\alpha[\ln(1+\beta T)+b]}. \tag{68}$$

Case 2: Shape parameter β is unknown, we assume the informative priors for α and β as $\alpha \sim Gamma(a,b)$ and $\beta \sim Gamma(c,d)$. This implies that $\pi(\alpha) \propto \alpha^{a-1} e^{-b\alpha}$ and $\pi(\beta) \propto \beta^{c-1} e^{-d\beta}$. Since α and β are independent the joint prior density $\pi(\alpha,\beta)$ is given as $\pi(\alpha,\beta) \propto \pi(\alpha).\pi(\beta)$. Implying that

$$\pi(\alpha,\beta) \propto \alpha^{a-1} e^{-b\alpha} \beta^{c-1} e^{-d\beta}. \tag{69}$$

From equation (11) and (54) we obtain the joint posterior density of α and β as

$$h(\alpha,\beta / Y_{obs}) = \frac{\alpha^{a-1} e^{-b\alpha} \beta^{c-1} e^{-d\beta} \alpha^n \beta^n \prod_{i=1}^n (1+\beta t_i)^{-1} e^{-\alpha \ln(1+\beta T)}}{\int_0^\infty \int_0^\infty \alpha^{a-1} e^{-b\alpha} \beta^{c-1} e^{-d\beta} \alpha^n \beta^n \prod_{i=1}^n (1+\beta t_i)^{-1} e^{-\alpha \ln(1+\beta T)} d\alpha d\beta}$$

$$= \frac{\alpha^{n+a-1} \beta^{n+c-1} \prod_{i=1}^n (1+\beta t_i)^{-1} e^{-d\beta} e^{-\alpha[\ln(1+\beta T)+b]}}{\int_0^\infty \beta^{n+c-1} \prod_{i=1}^n (1+\beta t_i)^{-1} e^{-d\beta} \left\{ \int_0^\infty \alpha^{n+a-1} e^{-\alpha[\ln(1+\beta T)+b]} d\alpha \right\} d\beta}$$

$$= \frac{\alpha^{n+a-1} \beta^{n+c-1} \prod_{i=1}^n (1+\beta t_i)^{-1} e^{-d\beta} e^{-\alpha[\ln(1+\beta T)+b]}}{\Gamma(n+a) \int_0^\infty \frac{\beta^{n+c-1} \prod_{i=1}^n (1+\beta t_i)^{-1} e^{-d\beta}}{[\ln(1+\beta T)+b]^{n+a}} d\beta}. \tag{70}$$

Letting $p = \int_0^\infty \frac{\beta^{n+c-1} \prod_{i=1}^n (1+\beta t_i)^{-1} e^{-d\beta}}{[\ln(1+\beta T)+b]^{n+a}} d\beta$ equation (70) reduces to

$$h(\alpha,\beta / Y_{obs}) = [p\Gamma(n+a)]^{-1} \alpha^{n+a-1} \beta^{n+c-1} \prod_{i=1}^n (1+\beta t_i)^{-1} e^{-d\beta} e^{-\alpha[\ln(1+\beta T)+b]}. \tag{71}$$

Preposition 4.3.1

The probability that at most k failures will occur in the interval (T,τ) with $\tau > T$ is

$$\gamma_k = \begin{cases} \left[\dfrac{\ln(1+\beta T)+b}{\ln(1+\beta\tau)+b}\right]^a \dfrac{\left[\ln(1+\beta T)+b\right]^n}{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^n} \displaystyle\sum_{j=n}^{n+k} \binom{j+a-1}{n+a-1} \dfrac{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^j}{\left[\ln(1+\beta\tau)+b\right]^j} & if\,\beta \quad is \quad known \\ \displaystyle\sum_{j=n}^{n+k} \dfrac{\Gamma(j+a)}{(j-n)!\,p\Gamma(n+a)} \int_0^\infty \dfrac{\beta^{n+c-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1} e^{-d\beta}}{\left[\ln(1+\beta\tau)+b\right]^{j+a}} d\beta & if\,\beta \quad is \quad unknown \end{cases} \tag{72}$$

Proof

The probability that at most k failures will occur in the interval (T,τ) is

$\gamma_k = \Pr\left[N(\tau) \le n+k / Y_{obs}\right]$, when β is known, we have

$$\gamma_k = \int_0^\infty \Pr\left[N(\tau) \le n+k / Y_{obs}, \alpha\right] h(\alpha / Y_{obs}) d\alpha \tag{73}$$

where $h(\alpha / Y_{obs})$ is given by equation (68) and

$$\Pr\left[N(\tau) \le n+k / Y_{obs}, \alpha\right] = \sum_{j=n}^{n+k} f(Y_{obs}, N(\tau) = j/\alpha) / f(Y_{obs} / \alpha). \tag{74}$$

From equation (15) $f(Y_{obs} / \alpha) = \alpha^n \beta^n \prod_{i=1}^{n} (1+\beta t_i)^{-1} e^{-\alpha \ln(1+\beta T)}$

and

$$f(Y_{obs}, N(\tau) = j/\alpha) = \int_{D(j-n;T,\tau)} f(Y_{obs}, x_{n+1}, \ldots, x_j, N(\tau) = j) \prod_{\ell=n+1}^{j} dx_\ell$$

$$f(Y_{obs}, N(\tau) = j/\alpha) = \int_{D(j-n;T,\tau)} \alpha^j \beta^j e^{-\alpha \ln(1+\beta T)} \prod_{i=1}^{j} (1+\beta t_i)^{-1} \prod_{\ell=n+1}^{j} dt_\ell$$

$$= \alpha^j \beta^j \prod_{i=1}^{n} (1+\beta t_i)^{-1} e^{-\alpha \ln(1+\beta\tau)} \int_{D(j-n;T,\tau)} \prod_{\ell=n+1}^{j} (1+\beta t_\ell)^{-1} \prod_{\ell=n+1}^{j} dt_\ell\,. \tag{75}$$

Solving the integral part in equation (75), we proceed as follows:

$\int_0^t (1+\beta t)^{-1} dt = \frac{1}{\beta}\ln(1+\beta t)$. Substituting the limits T and τ we have $\frac{1}{\beta}\ln(1+\beta\tau) - \frac{1}{\beta}\ln(1+\beta T)$ which reduces to $\frac{1}{\beta}\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)$. Therefore the integral part of equation (75) becomes

$$\int_{D(j-n;T,\tau)} \prod_{\ell=n+1}^{j} (1+\beta t_\ell)^{-1} \prod_{\ell=n+1}^{j} dt_\ell = \frac{1}{\beta^{j-n}} \frac{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n}}{(j-n)!}. \tag{76}$$

Substituting equation (76) to equation (75) we have

$$f(Y_{obs}, N(\tau) = j/\alpha) = \alpha^j \beta^j \prod_{i=1}^{j} (1+\beta t_i)^{-1} e^{-\alpha \ln(1+\beta\tau)} \frac{1}{\beta^{j-n}} \frac{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n}}{(j-n)!}.$$

From equation (74), we obtain

$$f(Y_{obs}, N(\tau) = j/\alpha) / f(Y_{obs}/\alpha) = \frac{\alpha^j \beta^j \prod_{i=1}^{n} (1+\beta t_i)^{-1} \frac{1}{\beta^{j-n}} \frac{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n}}{(j-n)!}}{\alpha^n \beta^n \prod_{i=1}^{n} (1+\beta t_i)^{-1} e^{-\alpha \ln(1+\beta T)}}$$

$$= \frac{\alpha^{j-n} \left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n} e^{-\alpha \ln\left(\frac{1+\beta\tau}{1+\beta T}\right)}}{(j-n)!}.$$

Thus equation (74) becomes

$$\Pr\left[N(\tau) \le n+k / Y_{obs}, \alpha\right] = \sum_{j=n}^{n+k} \alpha^{j-n} \left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n} e^{-\alpha \ln\left(\frac{1+\beta\tau}{1+\beta T}\right)} \frac{1}{(j-n)!} \tag{77}$$

and equation (73) becomes

$$\gamma_k = \int_0^\infty \sum_{j=n}^{n+k} \alpha^{j-n} \left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n} \frac{e^{-\alpha \ln\left(\frac{1+\beta\tau}{1+\beta T}\right)}}{(j-n)!\Gamma(n+a)} .\alpha^{n+a-1} \left[\ln(1+\beta T)+b\right]^{n+a} e^{-\alpha[\ln(1+\beta T)+b]} d\alpha$$

$$= \sum_{j=n}^{n+k} \left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n} \frac{\left[\ln(1+\beta T)+b\right]^{n+a}}{(j-n)!\Gamma(n+a)} \int_0^\infty \alpha^{j+a-1} e^{-\alpha[\ln(1+\beta T)+b]} d\alpha \tag{78}$$

$$= \sum_{j=n}^{n+k} \frac{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n} \left[\ln(1+\beta T)+b\right]^{n+a}}{(j-n)!\Gamma(n+a)} \frac{\Gamma(j+a)}{\left[\ln(1+\beta\tau)+b\right]^{j+a}}. \tag{79}$$

The integral part of equation (78) integrates to 1 since it is a gamma distribution with parameters $j+a$ and $\ln(1+\beta T)+b$.

On re-arranging equation (79), it becomes

$$\gamma_k = \left[\frac{\ln(1+\beta T)+b}{\ln(1+\beta\tau)+b}\right]^a \frac{\left[\ln(1+\beta T)+b\right]^n}{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^n} \sum_{j=n}^{n+k} \binom{j+a-1}{n+a-1} \frac{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^j}{\left[\ln(1+\beta\tau)+b\right]^j} \,. \tag{80}$$

This implies the first formula of equation (72).

When β is unknown, from equation (71) and equation (77) we have

$$\gamma_k = \int_0^\infty\int_0^\infty \Pr\left[N(\tau)\le n+k / Y_{obs},\alpha,\beta\right] h(\alpha,\beta / Y_{0bs})\, d\alpha\, d\beta$$

$$= \int_0^\infty\int_0^\infty \sum_{j=n}^{n+k} \frac{\alpha^{j-n}\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n}}{(j-n)!} e^{-\alpha\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)} \frac{\alpha^{n+a-1}\beta^{n+c-1}}{p\Gamma(n+a)} \prod_{i=1}^{n}(1+\beta t_i)^{-1} e^{-d\beta} e^{-\alpha\left[\ln(1+\beta T)+b\right]} d\alpha\, d\beta$$

$$= \int_0^\infty \sum_{j=n}^{n+k} \frac{\left[\ln\left(\frac{1+\beta\tau}{1+\beta T}\right)\right]^{j-n} \beta^{n+c-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{-d\beta}}{(j-n)!\,p\Gamma(n+a)} \left\{\int_0^\infty \alpha^{j+a-1} e^{-\alpha\left[\ln(1+\beta T)+b\right]} d\alpha\right\} d\beta$$

$$= \sum_{j=n}^{n+k} \frac{\Gamma(j+a)}{(j-n)!\,p\Gamma(n+a)} \int_0^\infty \frac{\beta^{n+c-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{-d\beta}}{\left[\ln(1+\beta\tau)+b\right]^{j+a}} d\beta \ . \tag{81}$$

Equation (81) implies the second formula of equation (72).

Preposition 4.3.2

The probability that the target value λ_{tv} will be achieved at time τ $\tau > T$ is

$$\gamma = \begin{cases} 1-\displaystyle\sum_{h=0}^{n+a-1} \frac{\left[\lambda_{tv}\frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]\right]^h}{h!} e^{-\lambda_{tv}\frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]} & if\,\beta \quad is \quad known \\ 1-\dfrac{1}{p}\displaystyle\sum_{h=0}^{n+a-1}\int_0^\infty \frac{\left[\lambda_{tv}\frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]\right]^h}{h!} e^{-\lambda_{tv}\frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]} \frac{\beta^{n+c-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{-d\beta}}{\left[\ln(1+\beta T)+b\right]^{n+a}} d\beta & if\,\beta \quad is \quad unknown \end{cases} \tag{82}$$

Proof

Let $f(\lambda_\tau / Y_{obs})$ denote the posterior of $\lambda_\tau = \alpha\beta/(1+\beta\tau)$. Hence, the probability that the target value λ_{tv} will be achieved at time τ is given by equation (44). When β is known, making transformation $\lambda_\tau = {}^{\alpha\beta}\!/\!_{(1+\beta\tau)}$, we have $\alpha = \lambda_\tau \dfrac{(1+\beta\tau)}{\beta}$ and $\dfrac{d\alpha}{d\lambda_\tau} = \dfrac{(1+\beta\tau)}{\beta}$. Consequently, the posterior density of λ_τ is

$$f(\lambda_\tau / Y_{obs}) = h(\alpha / Y_{obs}) \left| \frac{d\alpha}{d\lambda_\tau} \right|$$

$$= \left[\tfrac{\lambda_\tau (1+\beta\tau)}{\beta} \right]^{n+a-1} \frac{1}{\Gamma(n+a)} \left[\ln(1+\beta T) + b \right]^{n+a} e^{-\lambda_\tau \frac{(1+\beta\tau)}{\beta}[\ln(1+\beta T)+b]} \frac{(1+\beta\tau)}{\beta}$$

$$= \frac{\lambda_\tau^{n+a-1}}{\Gamma(n+a)} \left[\tfrac{(1+\beta\tau)}{\beta} \left[\ln(1+\beta T) + b \right] \right]^{n+a} e^{-\lambda_\tau \left[\frac{(1+\beta\tau)}{\beta}[\ln(1+\beta T)+b] \right]} . \quad (83)$$

Equation (83) follows a gamma distribution with parameters $n+a$ and $\left[\tfrac{(1+\beta\tau)}{\beta} \left[\ln(1+\beta T) + b \right] \right]$. From the relationship of gamma and Poisson distribution

$$\frac{b^a}{\Gamma(a)} \int_0^\lambda x^{a-1} e^{-bx} dx = 1 - \sum_{h=0}^{a-1} \frac{(b\lambda)^h}{h!} e^{-b\lambda} . \quad (84)$$

From equation (44), (83) and (84) we have

$$\gamma = 1 - \sum_{h=0}^{n+a-1} \frac{\left[\lambda_{tv} \frac{(1+\beta\tau)}{\beta} \left[\ln(1+\beta T) + b \right] \right]^h}{h!} e^{-\lambda_{tv} \frac{(1+\beta\tau)}{\beta}[\ln(1+\beta T)+b]} . \quad (85)$$

Equation (85) implies the first formula of equation (82).

When β is unknown, making transformation on $\lambda_\tau = {}^{\alpha\beta}\!/\!_{(1+\beta\tau)}$ and $\beta = \beta$, we obtain $\alpha = \lambda_\tau \frac{(1+\beta\tau)}{\beta}$ and $\beta = \beta$. Note that the Jacobian is ${}^{\partial(\alpha,\beta)}\!/\!_{\partial(\lambda_\tau,\beta)} = \dfrac{(1+\beta\tau)}{\beta}$. From equation (71), the joint posterior density of (λ_τ, β) is

$$f(\lambda_\tau, \beta / Y_{obs}) = h(\alpha, \beta / Y_{obs}) \left| \frac{d(\alpha,\beta)}{d(\lambda_\tau,\beta)} \right|$$

$$= \frac{\beta^{n+c-1} \prod_{i=1}^{n} (1+\beta t_i)^{-1}}{P\Gamma(n+a)} \left[\tfrac{\lambda_\tau (1+\beta\tau)}{\beta} \right]^{n+a-1} e^{-d\beta} e^{-\lambda_\tau \frac{(1+\beta\tau)}{\beta}[\ln(1+\beta T)+b]} . \frac{(1+\beta\tau)}{\beta}$$

$$= \frac{\beta^{n+c-1} \prod_{i=1}^{n} (1+\beta t_i)^{-1}}{P\Gamma(n+a)} \left(\tfrac{1+\beta\tau}{\beta} \right)^{n+a} e^{-d\beta} \lambda_\tau^{n+a-1} e^{-\lambda_\tau \left(\frac{1+\beta\tau}{\beta} \right)[\ln(1+\beta T)+b]}$$

$$= \frac{\beta^{n+c-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{-d\beta}}{p\left[\ln(1+\beta T)+b\right]^{n+a}} \frac{\left[\frac{(1+\beta\tau)}{\beta}\left(\ln(1+\beta T)+b\right)\right]^{n+a}}{\Gamma(n+a)} \lambda^{n+a-1} e^{-\lambda_\tau \frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]}. \quad (86)$$

Thus from equation (44), (84) and (86) we obtain,

$$\gamma = \frac{1}{p}\int_0^\infty \frac{\beta^{n+c-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{-d\beta}}{\left[\ln(1+\beta T)+b\right]^{n+a}} \left\{1 - \sum_{h=0}^{n+a-1} \frac{\left[\lambda_\tau \frac{(1+\beta\tau)}{\beta}\left(\ln(1+\beta T)+b\right)\right]^h}{h!} e^{-\lambda_\tau \frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]}\right\} d\beta$$

$$= 1 - \frac{1}{p}\sum_{h=0}^{n+a-1}\int_0^\infty \frac{\left[\lambda_{tv} \frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]\right]^h}{h!} e^{-\lambda_{tv} \frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]} \frac{\beta^{n+c-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{-d\beta}}{\left[\ln(1+\beta T)+b\right]^{n+a}} d\beta . \quad (87)$$

Equation (87) implies the second formula of equation (82).

Preposition 4.3.3

For a given level γ, the time τ^* required to attain λ_{tv}

$$\tau^* = \begin{cases} \left[\frac{\chi^2(2n;\gamma)}{2\lambda_{tv}\left[\ln(1+\beta T)+b\right]} - \frac{1}{\beta}\right] - T & \text{if } \beta \text{ is known} \\ \tau - T & \text{if } \beta \text{ is unknown} \end{cases} \quad (88)$$

Remark 3: For the second part of equation (88), τ is the solution of the equation

$$\gamma = 1 - \frac{1}{p}\sum_{h=0}^{n+a-1}\int_0^\infty \frac{\left[\lambda_{tv} \frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]\right]^h}{h!} e^{-\lambda_{tv} \frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]} \frac{\beta^{n+c-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{-d\beta}}{\left[\ln(1+\beta T)+b\right]^{n+a}} d\beta . \quad (89)$$

Proof

For given level γ, the time required to attain the target value λ_{tv} is $\tau^* = \tau - T$, where τ satisfies equation (44). When β is known, from equation (83), it can easily been seen that $2\left[\frac{(1+\beta\tau)}{\beta}\left\{\ln(1+\beta T)+b\right\}\right]\lambda_\tau$ follows a chi-square distribution with $2n$ degrees of freedom. Thus we have

$$2\left[\frac{(1+\beta T)}{\beta}\left\{\ln(1+\beta T)+b\right\}\right]\lambda_{tv} = \chi^2(2n;\gamma) \quad (90)$$

$$\frac{(1+\beta T)}{\beta}\left[\ln(1+\beta T)+b\right] = \frac{\chi^2(2n;\gamma)}{2\lambda_{tv}}. \quad (91)$$

making τ the subject

$$1+\beta\tau = \frac{\beta\chi^2(2n;\gamma)}{2\lambda_{tv}\left[\ln(1+\beta T)+b\right]}$$

$$\tau = \frac{\chi^2(2n;\gamma)}{2\lambda_{tv}\left[\ln(1+\beta T)+b\right]} - \frac{1}{\beta}. \quad (92)$$

Hence $\tau^* = \tau - T$ is given as

$$\tau^* = \left[\frac{\chi^2(2n;\gamma)}{2\lambda_{tv}\left\{\ln(1+\beta T)+b\right\}} - \frac{1}{\beta}\right] - T. \quad (93)$$

Equation (93) implies the first formula of equation (88).

When β is unknown, the time required to attain the target value λ_{tv} with level γ is $\tau^* = \tau - T$.

Where τ is the solution to

$$\gamma = 1 - \frac{1}{p}\sum_{h=0}^{n+a-1}\int_0^\infty \frac{\left[\lambda_{tv}\frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]\right]^h}{h!} e^{-\lambda_{tv}\frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]} \frac{\beta^{n+c-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{-d\beta}}{\left[\ln(1+\beta T)+b\right]^{n+a}} d\beta. \quad (94)$$

Preposition 4.3.4

The Bayesian UPL of $\lambda_\tau = \frac{\alpha\beta}{(1+\beta\tau)}$ with level γ is

$$\lambda_U^{(\beta)}(\tau) = \begin{cases} \frac{\beta\chi^2(2n;\gamma)}{2(1+\beta\tau)\left[\ln(1+\beta T)+b\right]} & if\, \beta \quad is \quad known \\ \lambda_{tv}^* & if\, \beta \quad is \quad unknown \end{cases} \quad (95)$$

Remark 4: The second part of equation (95) is such that λ_{tv}^* is the solution to

$$\gamma = 1 - \frac{1}{p}\sum_{h=0}^{n+a-1}\int_0^\infty \frac{\left[\lambda_{tv}^*\frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]\right]^h}{h!} e^{-\lambda_{tv}^*\frac{(1+\beta\tau)}{\beta}\left[\ln(1+\beta T)+b\right]} \frac{\beta^{n+c-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{-d\beta}}{\left[\ln(1+\beta T)+b\right]^{n+a}} d\beta. \quad (96)$$

Proof

For a pre-determined $\tau(\tau > T)$, the Bayesian UPL for λ_τ with level γ is $\lambda_U^{(\beta)}(\tau)$ satisfying $\gamma = \Pr(\lambda_\tau \le \lambda_U^{(\beta)} / Y_{obs})$. From equation (44) and (90) we have

$$\gamma = \int_0^{\lambda_U^{(\beta)}(\tau)} \frac{\left[\frac{(1+\beta\tau)}{\beta}\left(\ln(1+\beta T)+b\right)\right]^{n+a}}{\Gamma(n+a)} \lambda_\tau^{n+a-1} e^{-\lambda_\tau\left[\frac{(1+\beta\tau)}{\beta}\left(\ln(1+\beta T)+b\right)\right]}. \quad (97)$$

This implies that

$$2\left[\frac{(1+\beta\tau)}{\beta}\left(\ln(1+\beta T)+b\right)\right]\lambda_U^{(\beta)}(\tau) = \chi^2(2n;\gamma). \quad (98)$$

Making $\lambda_U^{(\beta)}(\tau)$ the subject from equation (98) we obtain

$$\lambda_U^{(\beta)}(\tau) = \frac{\beta\chi^2(2n;\gamma)}{2(1+\beta\tau)[\ln(1+\beta T)+b]}. \tag{99}$$

Equation (99) implies the first formula of equation (95).

When β is unknown, the Bayesian UPL for λ_τ with level γ is λ_{tv}^* where λ_{tv}^* is the solution to

$$\gamma = 1 - \frac{1}{p}\sum_{h=0}^{n+a-1}\int_0^\infty \frac{\left[\lambda_{tv}^*\frac{(1+\beta\tau)}{\beta}\{\ln(1+\beta T)+b\}\right]^h}{h!}e^{-\lambda_{tv}^*\frac{(1+\beta\tau)}{\beta}[\ln(1+\beta T)+b]}\cdot\frac{\beta^{n+c-1}\prod_{i=1}^{n}(1+\beta t_i)^{-1}e^{-d\beta}}{[\ln(1+\beta T)+b]^{n+a}}d\beta. \tag{100}$$

4.4 Real Data Examples for Bayesian Prediction

Table 4.1 is the real data on the time between failures which has been use to illustrate the developed methodologies for the one-sample Bayesian predictive analyses (Xie *et al*., 2002)

Table 4.1: Time between Failures Data.

Failure No	**Time between failures**	**Cumulative time between failures**	**Failure No.**	**Time between failures**	**Cumulative time between failures**
1	30.02	30.02	16	15.53	151.78
2	1.44	31.46	17	25.72	177.50
3	22.47	53.93	18	2.79	180.29
4	1.36	55.29	19	1.92	182.21
5	3.43	58.72	20	4.13	186.34
6	13.2	71.92	21	70.47	256.81
7	5.15	77.07	22	17.07	273.88
8	3.83	80.90	23	3.99	277.83
9	21	101.90	24	176.06	453.93
10	12.97	114.87	25	81.07	535.00
11	0.47	115.34	26	2.27	537.27
12	6.23	121.57	27	15.63	552.90
13	3.39	124.96	28	120.78	673.68
14	9.11	134.07	29	30.81	704.49
15	2.18	136.25	30	34.19	738.68

4.4.1 Parameter estimation using maximum likelihood estimation

Parameter estimation for NHPP model to be used in software reliability analysis is of primary importance. Parameter estimation was obtain by applying a technique of Maximum Likelihood Estimate (MLE) using Musa-Okumoto model. The MLE is consistent and asymptotically normally distributed as the sample size increases (Zhao, 1996). To obtain the

MLE of α and β for a sample of n units, we first take the log-likelihood of equation (19). The Log-likelihood is

$$\ell(\alpha,\beta) = n\log\alpha + n\log\beta - \sum_{i=1}^{n}\log(1+\beta t_i) - \alpha\log(1+\beta T)\,. \tag{101}$$

Differentiating partially equation (101) with respect to α and β and equating the derivatives to zero in order to optimize the values of α and β we get their estimates.

$$\frac{\partial\ell(\alpha,\beta)}{\partial\alpha} = \frac{n}{\alpha} - \ln(1+\beta T) = 0 \tag{102}$$

$$\frac{\partial\ell(\alpha,\beta)}{\partial\beta} = \frac{n}{\beta} - \sum_{i=1}^{n}\left(\frac{t_i}{(1+\beta t_i)}\right) - \frac{\alpha T}{(1+\beta T)} = 0\,. \tag{103}$$

Solving equations (102) and (103) for α and β we obtain;

$$\hat{\alpha} = \frac{n}{\ln(1+\hat{\beta}T)} \tag{104}$$

$$\frac{n}{\hat{\beta}} = \sum_{i=1}^{n}\left(\frac{t_i}{(1+\hat{\beta}t_i)}\right) + \frac{nT}{(1+\hat{\beta}T)\ln(1+\hat{\beta}T)}\,. \tag{105}$$

Equations (104) and (105) do not have closed forms and the values of $\hat{\alpha}$ and $\hat{\beta}$ can only be obtained numerically. There are number of numerical methods but this study used Newton Raphson (NR) as it converges faster than other methods (Lin and Chou, 2012). NR method was applied using R-program version 3.4.0 and by letting equation (103) be equal to $g(b)$,

$$g(b) = \frac{n}{\beta} - \sum_{i=1}^{n}\left(\frac{t_i}{(1+\beta t_i)}\right) - \frac{nT}{(1+\beta T)\ln(1+\beta T)}\,. \tag{106}$$

Taking derivative of $g(b)$ with respect to β we get;

$$g'(b) = -\frac{n}{\beta^2} + \sum_{i=1}^{n}\left(\frac{t_i}{(1+\beta t_i)}\right)^2 + \frac{nT^2\ln(1+\beta T) + nT^2}{(1+\beta T)^2\left[\ln(1+\beta T)\right]^2}\,. \tag{107}$$

The parameter $'b'$ is estimated by iterative Newton Raphson method using $b_{n+1} = b_n - \frac{g(b_n)}{g'(b_n)}$ which is substituted in finding $\hat{\alpha}$. NR iterative method was applied to get the estimates of $\hat{\alpha}$ and $\hat{\beta}$ from the data in Table 4.1. The initial value of β was assumed to $\beta_0 = 0.0001$.The

Newton Raphson method depends on the initial guess being close to the true value. If this requirement is not satisfied the procedure might converge to a minimum instead of a maximum, or just simply diverge and fail to produce any estimates at all. Methods of finding good initial estimates depend very much on the problem at hand and may require some ingenuity. Applying NR iterative method on data on table 4.1 with Musa – Okumoto model we obtained the estimates for $\hat{\beta} = 0.008282448$ and $\hat{\alpha} = 15.285550499$.

4.4.2 Goodness of Fit test for Musa – Okumoto model

For the case of Musa – Okumoto since $\Lambda(t) = \int_0^t \lambda(u)du$, we have:

$$\Lambda(u,\beta) = \ln(1+\beta u). \tag{108}$$

Therefore substituting in equation (16), we obtain the Laplace test for Musa - Okumoto;

$$S_n = \sqrt{\tfrac{12}{n}} \sum_{i=1}^{n} \left(\frac{\ln(1+\hat{\beta}_n u_i)}{\ln(1+\hat{\beta}_n \tau)} - \frac{1}{2} \right). \tag{109}$$

In order to determine the rejection region for the test statistic we need to derive the asymptotic properties of S_n. To derive the asymptotic properties of S_n, we first need to give the asymptotic properties of maximum likelihood estimator (MLE) $\hat{\beta}_n$ through a Lemma without proof.

Lemma 4.1 *Assume that the usual regularity condition for MLE apply to the function,* $f(u,\beta) = \dfrac{\lambda(u,\beta)}{\Lambda(\tau,\beta)}, 0 < u < \tau$. *Then under* H_0 *and conditionally on* $N(\tau) = n$, *it holds with probability 1 that the likelihood equation (105) admits a sequence of solution* $\{\hat{\beta}_n\}$ *satisfying*

$$\hat{\beta}_n \xrightarrow{a.s} \beta_0 \tag{110}$$

and

$$\sqrt{n}(\hat{\beta}_n - \beta_0) \xrightarrow{d} N(0, I^{-1}(\beta_0)) \tag{111}$$

where $I(\beta) = -E\left(\dfrac{\partial^2 \log f(u,\beta)}{\partial \beta^2} \right)$ is the Fisher information.

The theorem below gives the asymptotic distribution of S_n.

Theorem 4.1 *With the same assumption as in Lemma 4.1, and* $\frac{\partial \Lambda(t,\beta)}{\partial \beta} \to \frac{\partial \Lambda(t,\beta_0)}{\partial \beta}$ *uniformly for* t*, as* $\beta \to \beta_0$*. Then,*

$$S_n \xrightarrow{d} N(0, \delta(\beta_0)) . \tag{112}$$

where $\delta(\beta) = 1 - \frac{6\left(2\beta\tau - (2+\beta\tau)\ln(1+\beta\tau)\right)^2}{\left[\ln(1+\beta\tau)\right]^2 \left[(5\beta^2\tau^2 + 2\beta\tau)\ln(1+\beta\tau) - 2\beta^2\tau^2\right]}$ for Musa – Okumoto model. In this thesis we only derived $\delta(\beta)$ for Musa – Okumoto model as the theorem was proved by Zhao and Wang, (2005). From Lemma 4.1 and equation (108), it can easily be seen that

$$f(u,\beta) = \frac{\beta}{(1+\beta u)\ln(1+\beta\tau)} . \tag{113}$$

Taking logarithms on both sides we obtain;

$$\log f(u,\beta) = \ln\beta - \ln(1+\beta u) - \ln(\ln(1+\beta\tau)) . \tag{114}$$

The first and second derivative of equation (114) we have;

$$\frac{\partial \log f(u,\beta)}{\partial \beta} = \frac{1}{\beta} - \frac{u}{(1+\beta u)} - \frac{(1+\beta\tau)}{\tau\ln(1+\beta\tau)}$$

$$\frac{\partial^2 \log f(u,\beta)}{\partial \beta^2} = -\frac{1}{\beta^2} + \frac{u^2}{(1+\beta u)^2} - \frac{\ln(1+\beta\tau) - 1}{\left[\ln(1+\beta\tau)\right]^2} . \tag{115}$$

For $I(\beta) = -E\left\{\frac{\partial^2 \log f(u,\beta)}{\partial \beta^2}\right\}$ and substituting equation (115) we have;

$$I(\beta) = -E\left[-\frac{1}{\beta^2} + \frac{u^2}{(1+\beta u)^2} - \frac{\ln(1+\beta\tau)}{\left[\ln(1+\beta\tau)\right]^2}\right]$$

$$= \frac{1}{\beta^2} - E\left(\frac{u^2}{(1+\beta u)^2}\right) + \frac{\ln(1+\beta\tau)}{\left[\ln(1+\beta\tau)\right]^2} \tag{116}$$

where $E\left(\frac{u^2}{(1+\beta u)^2}\right) = \int_0^\tau \frac{u^2}{(1+\beta u)^2} f(u,\beta)du$ which we obtain;

$$E\left(\frac{u^2}{(1+\beta u)^2}\right) = \frac{\beta}{\ln(1+\beta\tau)} \int_0^\tau \frac{u^2}{(1+\beta u)^3} du$$

$$= \frac{-\beta\tau^2 - 2\tau(1+\beta\tau) + 2(1+\beta\tau)^2 \ln(1+\beta\tau)}{2\beta(1+\beta\tau)^2 \ln(1+\beta\tau)} . \tag{117}$$

Hence

$$I(\beta) = \frac{1}{\beta^2} + \frac{\tau^2 \ln(1+\beta\tau) - \tau^2}{(1+\beta\tau)^2[\ln(1+\beta\tau)]^2} - \frac{-\beta\tau^2 - 2\tau(1+\beta\tau) + 2(1+\beta\tau)^2 \ln(1+\beta\tau)}{2\beta(1+\beta\tau)^2 \ln(1+\beta\tau)}$$

$$= \frac{(5\beta^2\tau^2 + 2\beta\tau)\ln(1+\beta\tau) + 2\beta^2\tau^2}{2\beta^2(1+\beta\tau)^2[\ln(1+\beta\tau)]^2} . \tag{118}$$

Since $\Lambda(\tau,\beta) = \ln(1+\beta\tau)$, thus

$$\frac{\partial\Lambda(u,\beta)}{\partial\beta} = \frac{u}{1+\beta u} . \tag{119}$$

Expectation of equation (119) is;

$$E\frac{\partial\Lambda(u,\beta)}{\partial\beta} = \int_0^\tau \frac{u}{1+\beta u} f(u,\beta)du$$

$$= \frac{(1+\beta\tau)\ln(1+\beta\tau) - \beta\tau}{\beta(1+\beta\tau)\ln(1+\beta\tau)} . \tag{120}$$

$$\frac{1}{2}\frac{\partial\Lambda(\tau,\beta)}{\partial\beta} - E\frac{\partial\Lambda(u,\beta)}{\partial\beta} = \frac{\tau}{2(1+\beta\tau)} - \frac{(1+\beta\tau)\ln(1+\beta\tau) - \beta\tau}{\beta(1+\beta\tau)\ln(1+\beta\tau)}$$

$$= \frac{2\beta\tau - (2+\beta\tau)\ln(1+\beta\tau)}{2\beta(1+\beta\tau)\ln(1+\beta\tau)} . \tag{121}$$

From equation (118) and (120) $\delta(\beta)$ can be obtained by substituting them in equation (107). Thus we obtain;

$$\delta(\beta)=1-\frac{6\left[2\beta\tau-(2+\beta\tau)\ln(1+\beta\tau)\right]^2}{\left[\ln(1+\beta\tau)\right]^2\left[(5\beta^2\tau^2+2\beta\tau)\ln(1+\beta\tau)-2\beta^2\tau^2\right]}. \quad (122)$$

To test the real data from Xie et al. (2002) on Table 1 for Musa – Okumoto model. It consist of $n=30$ and interfailure times up to the time $\tau=738.68$. To test Musa - Okumoto model, the MLE $\hat{\beta}_n=0.008282448$ and $\delta(\hat{\beta}_n)=0.9327387$. The value of the test statistic is $S_n=$ -0.3446299. If we choose significance level $\alpha=0.05$ $|S_n|=0.3446299<Z_{\frac{\alpha}{2}}\sqrt{\delta(\hat{\beta}_n)}=1.892902$ and the p-value is 0.639394. So we do not reject Musa – Okumoto model. Since the Musa – Okumoto model fit well to the data, it was used to check on the methodologies derived. From Zhao and Wang (2005), $\delta(\beta)$ for Goel – Okumoto was derived and to test the model on the same real data in table 1, MLE $\hat{\beta}_n=0.003969$ and $\delta(\hat{\beta}_n)=0.1046887$. The value of the test statistic is $S_n=0.5760702$. Similarly, we choose the significance level $\alpha=0.05$, then $|S_n|=0.5760702<Z_{\frac{\alpha}{2}}\sqrt{\delta(\hat{\beta}_n)}=0.6341587$ and the p – value is 0.03750259. So we reject Goel – Okumoto model.

4.4.3 Using Non – Informative Priors

(A) Suppose we are interested in the probability γ_k that at most k failures will occur in a future time period $(T,\tau]=(180,250]$. (i) When β is known, ($\beta=0.008282448$), using the first formula in equation (33), we have

$\gamma_0=0.00204337$, $\gamma_1=0.01347748$, $\gamma_2=0.04653484$, $\gamma_3=0.11230530$, $\gamma_4=0.21351423$, $\gamma_5=0.34188371$, $\gamma_6=0.48155675$, $\gamma_7=0.61554018$, $\gamma_8=0.73112395$, $\gamma_9=0.82215131$, $\gamma_{10}=0.88836847$, $\gamma_{11}=0.93328146$, $\gamma_{12}=0.96190403$, $\gamma_{13}=0.97915241$, $\gamma_{14}=0.98903392$, $\gamma_{15}=0.994444044$

(ii) For the case when β is unknown, from the second formula of equation (33), we obtain

γ_0=0.002423218, γ_1=0.015348190, γ_2=0.052195351, γ_3=0.122789747, γ_4=0.229662151, γ_5=0.362653362, γ_6=0.504636623, γ_7=0.639743372, γ_8=0.750133795, γ_9=0.836483617, γ_{10}=0.897655409, γ_{11}=0.941055333, γ_{12}=0.963420189, γ_{13}=0.981411270, γ_{14}=0.989850595, γ_{15}=0.991582449

Figure 4.1 shows the graph of desired probability of k when β is known and when it is unknown for non-informative prior.

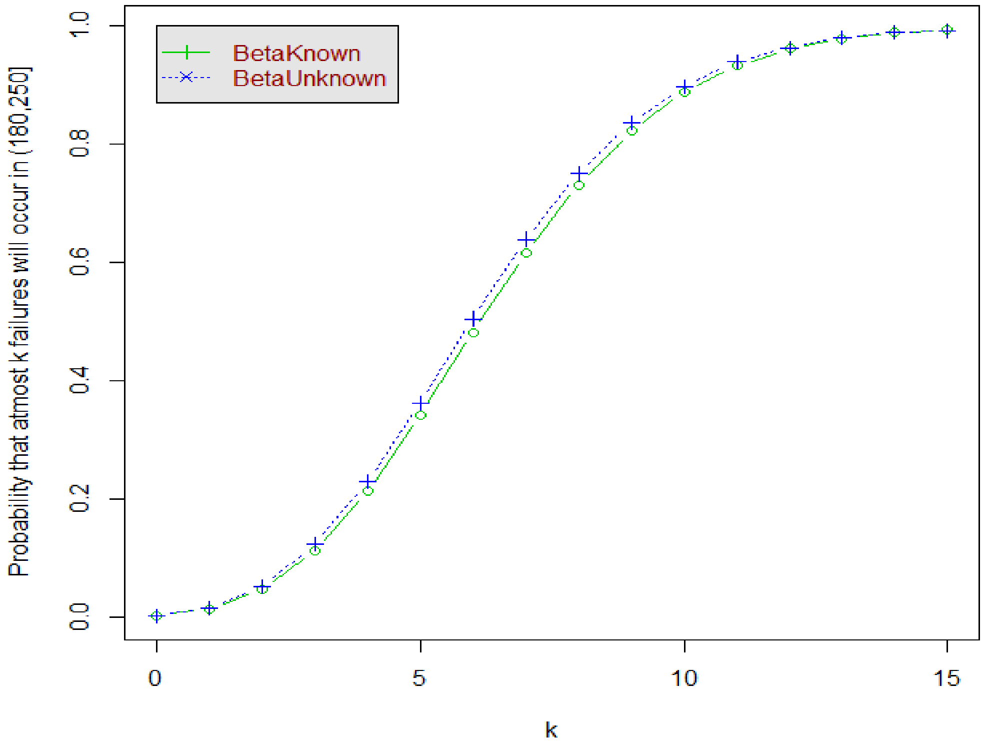

Figure 4.1: The graph of the probabilities γ_k that at most k failures will occur in the time interval (180, 250] for the cases of β known and unknown for non-informative prior.

It can be noted that the probability of failure occurring depend on the length of the time interval. From the graph it can be seen that there is high probability that at most 15 failures will occur during that time interval when β is unknown as compared to when it is known.

(B) Suppose the target value is given by $\lambda_{tv} = 0.03$ chosen arbitrarily. At the time $T = 182.21$, the MLE of the achieved failure rate for this software is

$\hat{\lambda}(182.21) = \hat{\alpha}\hat{\beta} / (1+182.21\hat{\beta}) = 0.05045615$, which is greater than λ_{tv} thus it cannot be achieved at time $T = 182.21$ and development testing will continue. Suppose we want to find the probability that the target value λ_{tv} will be achieved at the time $\tau = 277.83h$. (i) When β is known (say, $\beta = 0.008282448$), from the first formula in equation (43), we obtain $\gamma =$ 1.687506e-06. Hence, the target value will not be achieved. (ii) when β is unknown, from the second formula in equation (43) we have $\gamma = 0.193896$ based on the Monte Carlo $L = 1000$.

(C) Since the target value λ_{tv} was not achieved at $T = 182.21$, we want to know how long it will take for the target value to be achieved. (i) when β is known (say, $\beta = 0.008282448$), let $\gamma = 0.90$, from the first formula in equation (52) we obtain $\tau^* = 538.7523h$. This means that, it will take another 538.7523h in order to achieve the desired failure rate. (ii) when β is unknown, from second formula in equation (52) and Remark1, we obtain $\tau^* = 414h$. Thus, it takes another 414 hours in order to achieve the desired failure rate when β is unknown. This shows a high reduction in time as compared to when β is known. (D) Given $\tau = 900h$, from first formula in equation (59) the Bayesian UPL of $\lambda_\tau = \dfrac{\alpha\beta}{1+\beta\tau}$ with level $\gamma = 0.90$ is given by $\lambda_U^{(\beta)}(\tau) = 0.02473799$.

4.4.4 Using Informative priors

In this section we have used gamma priors for both parameters. The values of parameters of informative priors $Gamma(a,b)$ and $Gamma(c,d)$ are chosen arbitrarily as $a = 2, b = 1/2, c = 2$ and $d = 1/2$.

(A) Suppose we are interested in the probability γ_k that at most k failures will occur in a future time period $(T,\tau] = (180, 250]$. Considering the case when β is known (i.e, $\beta = 0.008282448$), using the first formula in equation (72) we have

γ_0=0.01202933, γ_1=0.06169460, γ_2=0.16742455, γ_3=0.32202724, γ_4=0.49656381, γ_5=0.65870012, γ_6=0.78770083, γ_7=0.87805309, γ_8=0.93488273, γ_9=0.96747044, γ_{10}=0.98470892, γ_{11}=0.99320106, γ_{12}=0.99712719, γ_{13}=0.99884169, γ_{14}=0.99955271, γ_{15}=0.99983403

Figure 4.2 shows the graph of probabilities that at most k failures will occur in the time interval $(180, 250]$ for β known for both informative and non-informative priors. From the graph it can be seen that the probabilities for informative prior is high as compared to that of non-informative. This is more seen at issue C, where there is high reduction of time required to achieve a predetermined target value in informative prior.

(B) Suppose the target value is given by $\lambda_{tv} = 0.03$. At the time $T = 182.21$, the MLE of the achieved failure rate for this software is $\hat{\lambda}(182.21) = \hat{\alpha}\hat{\beta} / (1+182.21\hat{\beta}) = 0.05045615$, which is greater than λ_{tv} thus it cannot be achieved at time $T = 182.21$ and development testing will continue. Suppose we want to find the probability that the target value λ_{tv} will be achieved at the time $\tau = 277.83h$. (i) When β is known (say, $\beta = 0.008282448$), from the first formula in equation (82), we obtain $\gamma = 0.0007319763$. In this case also as that of non-informative prior, the target value is unlikely to be achieved. (ii) When β is unknown, from the second formula in equation (82), we obtain $\gamma = 0.08730647$ where the Monte Carlo $L = 1000$.

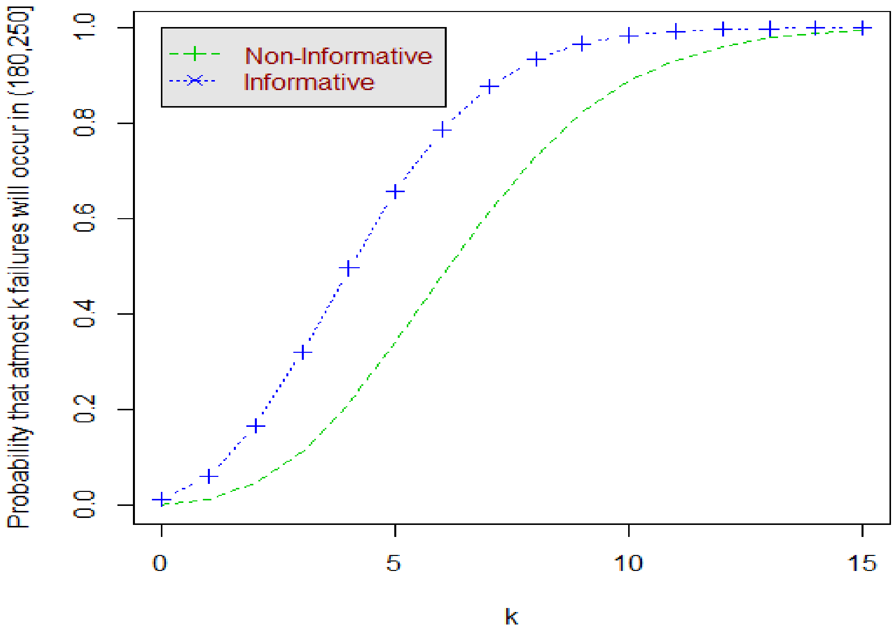

Figure 4.2: The graph of the probabilities γ_k that at most k failures will occur in the time interval (180, 250] for the cases of β known for informative and non-informative prior.

(C) Since the target value $\lambda_{tv} = 0.03$ is not achieved at time $T = 182.21$. It is interesting now to know how long it will take in order to achieve the desired target value. (i) When β is known (say, $\beta = 0.008282448$), using the first formula in equation (88) and letting $\gamma = 0.90$, we obtain $\tau^* =$ 242.3671h. Thus, it will take another 242.3671 hours in order to achieve the target value which is a significant reduction from the value obtained for the case of non-informative prior. (ii) When β is unknown, from the second formula in equation (88) we have $\tau^* = 147$h. It will take another 147 hours for the desired target value to be achieved when β is unknown. . (D) Given $\tau = 900h$, when β is known (i.e, $\beta = 0.008282448$), from first formula in equation (95) the Bayesian UPL of $\lambda_\tau = \dfrac{\alpha\beta}{1+\beta\tau}$ with level $\gamma = 0.90$ is given by $\lambda_U^{(\beta)}(\tau) = 0.01602707$.

4.4.5 Coverage Probability for Bayesian Upper prediction limit (UPL)

Coverage probability (CP) of random interval $[L(X), U(X)]$ is the probability that the interval covers the true parameter θ, that is $P_\theta(\theta \in [L(X), U(X)]) = P_\theta(L(X) \leq \theta, U(X) \geq \theta)$. It should be noted that the interval is random not θ. In this thesis, coverage probability of Bayesian UPL is presented, this was calculated by repeated sampling through simulation. In simulation case, we simulated L = 1000 data sets. In each data set, parameters were estimated and 95% Bayesian UPL were computed. In addition coverage probabilities for the UPL were calculated.

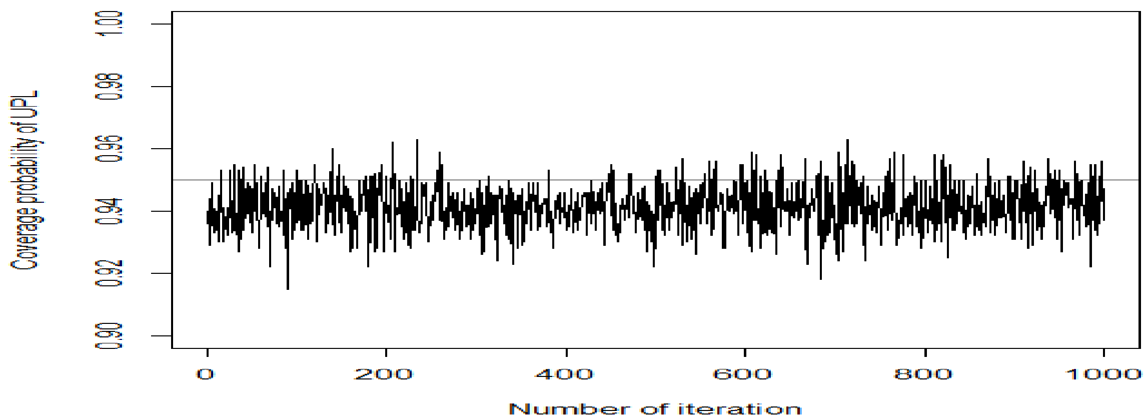

Figure 4.3: Graph of Coverage probability of 95% Bayesian UPL against the number of iterations L=1000.

From the Figure 4.3, all the coverage probabilities are close to the nominal 95% and this means that our Bayesian UPL is a good estimator. Similar observation is also made from the summary of coverage probability as shown in Table 4.2,

Table 4.2: Summary of Coverage probabilities of 95% Bayesian Upper prediction limit.

Min.	**1st Quartile**	**Median**	**Mean**	**3rd Quartile**	**Max.**
0.9150	0.9360	0.9410	0.9414	0.9460	0.9630

From the Table 2, it can be seen that 0.9150 and 0.9630 are the lowest and highest coverage probabilities with the mean of 0.9414. From the results above of CP, Bayesian UPL is a good estimator and this can even be extended to real data.

CHAPTER FIVE
SUMMARY, CONCLUSION AND RECOMMENDATION

5.1 Introduction

This chapter covers the summary of thesis, conclusion from the results and the recommendation for future work.

5.2 Summary

The study has presented results on Bayesian prediction analyses on software reliability. It mainly focuses on deriving Bayesian predictive distribution on single-sample case on four issues that relate to software development testing process. The derived methodologies has been presented on chapter four as preposition and were illustrated using real data on time between failures. Coverage probability of Bayesian UPL was computed using simulated data.

5.3 Conclusion

In software development, predictive analysis is very important as it helps the software developer to make a trade-off decision at the right time. The study used both non-informative and informative priors to derive posterior and predictive distributions for issues that relates to software development testing process. The derivations have been given as preposition in this thesis and their proof given .In all the cases when the shape parameter was known, solutions to posterior and predictive distributions had closed forms while when it is unknown, solutions had no closed forms and the study used Markov Chain Monte Carlo (MCMC). The methodologies developed were illustrated using real data and had explicit solutions. Coverage probability of Bayesian UPL was also computed and found to be a good estimator. These solutions are helpful to software developers in many instances such as resource allocation, when to terminate the testing process, modification needed in the software before termination. Bayesian approach was used as it is advantageous over classical approach as it is available for small sample sizes and allows the input of prior information about reliability growth process and provides full posterior and predictive distributions (Jun-Wu *et al*., 2007).

5.4 Recommendation

The analysis discussed in this thesis is quite useful tool for the software developers to plan and work out the details of the requirement for testing resources so that the product is ready for the release by its due date. It helps the organization to beat the intense competition by launching the reliable and good quality product well on time and thus retaining the present clientele and building future ones.

This study has considered and derived both posterior and predictive distributions for one – sample software on Musa – Okumoto (1984) model on the selected issues that may arise during software development. However, it will be interesting to look at two-sample prediction for Musa – Okumoto (1984) model considering procedures that Jun-Wu *et al* (2007) used. The procedures presented in this thesis can also be extended to other NHPP models such as Cox- Lewis process and the delayed S-shaped process. This is left open for future research.

REFERENCES

Abdel-Ghaly, Chan, P.Y., and Littlewood, B. (1986). *Evaluation of competing software reliability predictions*. *IEEE Trans. Softw. Engineering*, **12**, 950–967.

Akuno, A.O., Orawo, L.A. and Islam, A.S. (2014) One-Sample Bayesian Predictive Analyses for an Exponential Non-Homogeneous Poisson Process in Software Reliability. *Open Journal of Statistics*, **4**, 402-411.

Allan, T. M. (2012). *Bayesian Statistics applied to Reliability Analysis and Prediction*. Raython Missile Systems, Tucson, AZ.

Andrew, G., John, B. C., Hal, S. S. and Donald, B. R. (1995). *Bayesian Data Analysis*. Texts in Statistical Science, Chapman & Hall.

Anjum, M., Haque, A.M and Nesar, A. (2013). Analysis and Ranking of Software Reliability Models Based on weighted Criteria Value. *I.J Information Technology and computers Science.*

Compodónico, S. and Singpurwalla, N.D. (1994). *A Bayesian Analysis of the Logarithmic-Poisson Execution Time Model Based on Expert Opinion and Failure Data*. IEEE ,Transactions on Software Engineering, **20**. NO. 9.

Crowder, M. J., Kimber, A., Sweeting, T., and Smith, R. (1994). *Statistical analysis of reliability data*, **27**. CRC Press.

Daniel, R. J. and Hoang P. (2001). On the Maximum Likelihood Estimates for the Goel-Okumoto Software Reliability Model. *The American Statistician*, **55**: 219-222.

Ganesh, P. (2002). A survey of software reliability models. Deparment of ECE, University of Virginia.

Garg, R., Kapil, S., Kumar, R and Garg, R.K. (2010). Performance Analysis of Software Reliability Models using Matrix Method‖, World Academy of Science, *Engineering and Technology.*

Glickman, M. E. and Van D. D. A. (2007). Basic Bayesian Methods. *In: Methods in Molecular Biology: Topics in Biostatistics* (WT Ambrosius, ed.). Humana Press, Totwa, NJ, USA: **104**: 319-338.

Jelinski, Z., and Moranda, P. (1972). *Software Reliability Research. Statistical Computer Performance Evaluation*, Ed. W. Freiberger. New York: Academic.

Jun-Wu, Y., Guo-Liang, T. and Man-Lai, T. (2007). Predictive analyses for non-homogeneous Poisson processes with power law using Bayesian approach. *Computational Statistics & Data Analysis*, **51**: 4254-4268.

Kapur, P. K., Pham, H., Gupta, A. and Jha, P. C. (2011). *Software Reliability Assessment with OR Applications.* Springer Series in Reliability Engineering, Springer-Verlag London Limited 2011. **58**.

Lin, C. T., and Chou, C. C. (2012). Statistical inference for a lognormal step-stress model with type-I censoring. *IEEE Transactions on Reliability*, ***61***(2), 361-377.

Lyu, M. (1996). *Handbook of Software Reliability Engineering*. New York: McGraw-Hill.

Malaiya, Y. K. and Denton, J. (1997). *What Do the Software Reliability Growth Model Parameters Represent?* Technical Report CS-97-115, Department of Computer Science, Colorado State University.

Meth, M. (1992). *Reliability growth myths and methodologies: a critical view*. In: Proceedings of the Annual Reliability and Maintainability Symposium. IEEE, NewYork, 230–238.

Muralidharan K., Rupal, S. and Deepak H. D. (2008). Future Reliability Estimation Based on Predictive Distribution in Power Law Process. *Quality Technology & Quantitative Management*, **5**: 193-201.

Musa, J. D. and Okumoto, K. (1984). A Logarithmic Poisson Execution Time Model for Software Reliability Measurement. *In Proceedings of the Seventh International Conference on Software Engineering*, pp. 230- 238.

Musa, J.D., Iannino, A. and Okumoto,K. (1987*). Software Reliability Measurement, Prediction, Application*. : McGraw Hill

Nassif, A.B., Ho, D. and Capretz, L.F. (2013). *Towards an Early Software Estimation Using Log-Linear Regression and a Multilayer Perceptron Model*. Journal of Systems and Software, **86**, pp. 144-160.

Nassif, A.B., Capretz, L.F. and Ho, D. (2010). *Software Estimation in the Early Stages of the Software Life Cycle*. International Conference on Emerging Trends in Computer Science, Communication and Information Technology, Nanded, India, 5-13, 2010.

Néstor, R.B. (2010). *Compound and Non Homogeneous Poisson Software Reliability Models*. 461.

Nuria, T. R. (2011). *Stochastic Comparisons and Bayesian Inference in Software Reliability*. Ph.D. Thesis, Universidad Carlos III de Madrid.

Peng,R., Hu,Q.P. and Ng, S.H. (2008) ,"Incorporating Fault Dependency and Debugging Delay in Software Reliability Analysis," *Proceedings of the IEEE ICMIT*, 2008, pp.641-645.

Razeef, M. and Mohsin, N. (2012). Software Reliability Growth Models: Overview and Applications. *Journal of Emerging Trends in Computing and Information Sciences*, **3**: 1309-1320.

Rigdon, S.E. and Basu, A.P. (2000). The Power Law Process: a Model for the Reliability of Repairable systems. *Journal of Quality Technology*, **21**: 251-260.

Sheakh, T.H., Quadri, S.M.K. and Singh, V. (2012). Critical Review of Software Reliability Model. *International Journal of Emerging Technology and Advance Engineering*, **2,** 2250-2459**.**

Sheldon, M.R. (1997). *Introduction to Probability Models*. Academic Press Limited. USA

Singpurwalla, N. D and Simon, P. W. (1994). Software Reliability Modeling *International* 59 *Statistical Review/ Revue Internationale de Statistique*, **62**: 289-317.

Sonia, D., and Renu, D. (2014). *A Study of Various Reliability Growth Models*. International Journal of Advanced Research in Computer Science and Software Engineering, vol4

Tae, Y. Y. and Lynn, K. (1999). Bayesian Computation for the Superposition of Non-homogeneous Poisson Processes. *The Canadian Journal of Statistics,* **27**: 547-556.

Ullah, N., Morisio, M. and Vetro, A. (2013). *A Comparative Analysis of Software Reliability Growth Models using defects data of Closed and Open Source Software*. In: 35TH Annual IEEE Software Engineering Workshop, Heraclion, Crete, Greece, 12-13 October 2012. 187-192.

Xia, W., Capretz, L.F. and Ho, D. (2008). *A Neuro-Fuzzy Model for Function Point Calibration*. WSEAS Transactions on Information Science and Applications, **5**, 22-30.

Xie, M., Goh, T. N., and Ranjan, P. (2002). Some effective control chart procedures for reliability monitoring. *Reliability Engineering & System Safety*, ***77***(2), 143-150.

Yadav, A. and Khan, R.A. (2009) .Critical Review on Software Reliability Models. *International Journal of Recent Trends in Engineering*, **2**.

Yamada, S., Ohba, M. and Osaki, S. (1983). S-shaped reliability growth modeling for software error detection. *IEEE Trans. Reliability*, **32**: 475–484.

Zhao, J., and Wang, J. (2005). A new goodness-of-fit test based on the Laplace statistic for a large class of NHPP models. *Communications in Statistics—Simulation and Computation®*, ***34***(3), 725-736.

Zhao, C., (2004). Bayesian and Empirical Bayes approaches to power law process and microarray analysis. PHD Thesis, University of South Florida. USA.

Zhao, M., and Xie, M. (1996). On maximum likelihood estimation for a general non-homogeneous Poisson process. *Scandinavian journal of statistics*, 597-607.