Degradation Analysis for Heterogeneous Data Using Mixture Model

A thesis presented to

the faculty of

the Russ College of Engineering and Technology of Ohio University

In partial fulfillment

of the requirements for the degree

Master of Science

Yizhen Ji

May 2013

This thesis titled

Degradation Analysis for Heterogeneous Data Using Mixture Model

by

YIZHEN JI

has been approved for

the Department of Industrial and Systems Engineering

and the Russ College of Engineering and Technology by

Tao Yuan

Assistant Professor of Industrial and Systems Engineering

Dennis Irwin

Dean, Russ College of Engineering and Technology

Abstract

JI, YIZHEN, M.S., May 2013, Industrial and Systems Engineering

Degradation Analysis for Heterogeneous Data Using Mixture Model (76 pp.)

Director of Thesis: Tao Yuan

New product testing presents a significant challenge to manufacturers of highly reliable products. For products with high reliability, very few or even no failures might be expected in reliability testing, resulting in limited information about reliability of the products. If a degradation measure that is closely related to failure can be monitored, degradation analysis, as an alternative to the failure time analysis, may lead to improved reliability inference and provide additional information related to the failure mechanisms.

Degradation analysis has attracted considerable attention in recent years. Mixed-effects models have been frequently employed to analyze repeated-measures degradation data of multiple units. In existing studies, the test units are usually assumed to be sampled from a homogeneous population, and the random effects in the degradation models are generally assumed to be normally distributed. However, in practical applications of degradation analysis, excessive variability among the degradation paths of different units may be observed due to different reasons such as quality, degradation mechanism, and external environmental condition, etc. The normal distribution may not be adequate to describe the observed unit-to-unit variability. Reliability analysis for units from a nonhomogeneous population with subgroups has been considered only in failure time analysis.

This thesis considers the degradation analysis for units coming from a nonhomogeneous population. Instead of the normal distribution, this thesis assumes a normal mixture distribution for the random effects. Both maximum likelihood and Bayesian approaches are adopted in this thesis for the inference of the model parameters and for the derivation of the lifetime distribution. Practical example is used to illustrate

the usefulness of the proposed methods. The results show that: (1) both ML method and Bayesian approach give similar estimates for the model parameters; (2) the mixture model has its notable advantages in fitting heterogenous data and providing detailed information regarding different subgroups.

Acknowledgments

In the past two-years graduate study at Ohio University, there are several individuals and institutions giving me unconditional help and support on my research. This thesis would not have been possible to complete without their assistance in direct or indirect way. Therefore, I would like to take this opportunity to thank for their encouragement and support.

First and foremost, I would like to express my utmost gratitude to my advisor, Dr. Tao Yuan for his immense knowledge, patience and steadfast encouragement in all the time of research and writing of this thesis. He not only supports me to overcome all the obstacles in this research work, but also morally encourages me and gives me a great example to be a confident and optimistic person. I cannot imagine having a better advisor and mentor in my master study.

Besides my advisor, I would like to thank the rest of my thesis committee members: Dr. David Koonce, Dr. Diana Schwercha and Dr. Andrew Snow for their guidance, suggestions, questions and insightful comments on my thesis proposal and defense.

In addition, my sincere thanks also express to Mr. Donald Bentley and Mr. Mike Lampl at Ohio Bureaus of Workers' Compensation (OBWC), for offering me an internship as a data analyst at OBWC. The working projects at OBWC gave me hands-on experience on data analysis, which help me apply theoretical knowledge on practical problem solving.

I also would like to thank my boyfriend (Shuluo Ning) and my lab mates (Xi Liu and Nan Cheng) in Ohio University, for their unselfish support in sharing valuable opinions and offering suggestions on modeling and coding for this thesis.

Last but not the least, I would like to thank my family, for their understanding, continuous support and love throughout my life.

Table of Contents

List of Tables

List of Figures

1 INTRODUCTION

This chapter introduces the motivation, objectives and background of this thesis. Topics that are introduced in this chapter include: reliability, degradation analysis, maximum likelihood estimation, Bayesian inference, and Gaussian mixture model.

1.1 Motivation and Objectives

Owing to the quantum leaps in technology, the manufacturing process becomes significantly mature. Meanwhile, the fierce competition among similar products promotes the customer's expectation and satisfaction level of new products. Therefore, products are expected to have high quality and reliability. For this reason, only a few or even no failure occurs even under accelerated test conditions during a long period. In practice, it is expensive and time-consuming to obtain enough failure data for highly reliable products. Under this circumstance, traditional life-time failure data analysis which is based on the collection of failure data loses its effectiveness to perform reliability assessment. In recent decades, since level degradation was widely applied as an alternative expression of failure, the degradation analysis has been developed as a convenient and analytically sound alternative method to the conventional failure time analysis.

In reliability analysis, previous studies usually assume that the units come from a homogeneous population due to similar failure mechanism. Most degradation analyses focus on developing various degradation models and conducting data analysis for the units under the homogeneity assumption. But in practice, because of different quality of the units or different failure mechanism, it is possible that the units come from a heterogeneous population and can be classified into subgroups. Although some pervious studies have observed the heterogeneous failure data, they only suggested using mixture lifetime distributions for reliability analysis. However, degradation analysis considering the units from a heterogeneous population of mixing subgroups has not been studied yet.

The motivation of this thesis stems from the significance of degradation analysis. Since the products become more and more reliable, it is hard to obtain sufficient failure data within an acceptable period to perform conventional failure time analysis. Therefore, degradation analysis can be applied as an effective way to assess reliability of products. Another significant motivation is generated from practical problems. One motivated example is "Percent increase in operating current for GaAs lasers" reported by Meeker and Escober [1]. By plotting degradation curve of this example, it is reasonable to assume that there exists the possibility that the degradation data can be classified into subgroups based on similar degradation rate (slope of the degradation curve). Thus, a model which can describe heterogeneous degradation data is needed. In addition, to analyze heterogeneous degradation data, it is necessary to employ powerful statistical method and corresponding computational algorithm for parameter estimation and failure time distribution derivation.

Driven by above motivations, this thesis has three objectives as follows:

(1) Propose a random-effect degradation model to analyze the degradation data generated from heterogeneous population. A Gaussian mixture model is employed to model the variability among different groups.

(2) Employ both maximum likelihood method and Bayesian approach to estimate the model parameters. For statistical computation, Expectation-Maximization (EM) algorithm and Gibbs sampling are developed corresponding to each method. Moreover, failure time distribution is derived.

(3) Compare and select the most appropriate model to fit the data according to one specific model selection criteria.

1.2 Background

1.2.1 Reliability analysis

Reliability is defined to be the probability that a component or a system will perform a required function adequately for a given period of time when used under stated operating conditions [2]. If the continuous random variable T is defined to be the time to failure of the component or a system, then the reliability function can be expressed as:

$$R(t) = P(T > t) = \int_t^\infty f(u)du \tag{1.1}$$

where $f(u)$ is the probability density function (PDF) of the failure-time distribution.

Based on failure data collected in the life test, the reliability for a component or systems can be assessed by failure time analysis. Traditional failure time analysis records only failure times to predict reliability, which requires both sufficient time and data. In contrast, by interpreting the failure in terms of a specified level of degradation, degradation analysis is a more flexible and practical method to provide information of reliability.

1.2.2 Degradation analysis

For components and systems with high reliability, conventional failure time analysis may not be adequate due to the difficulty of collecting enough failure data within an acceptable period. Although the accelerated testing approach can be used to improve the testing efficiency to some extent, it may still be not good enough. Therefore, degradation analysis and modeling, using a sequence of degradation measures to assess reliability, have attracted considerable attention recently [3]. Compared to the traditional failure data analysis, degradation analysis is a more convenient and analytically sound method of estimating failure-time distributions and assessing reliability [4]. Moreover, degradation analysis provides information related to failure mechanisms as well.

Degradation analysis is usually based on some degradation path models. A general random-effect degradation path model is given by

$$y_{ij} = g(t_{ij}; \boldsymbol{\vartheta}) + \varepsilon_{ij}, i = 1, \ldots, n, j = 1, \ldots, n_i \tag{1.2}$$

where n is the number of units and n_i is the number of measurements for ith unit. y_{ij} is the jth response on ith units and $g(\cdot)$ with unknown parameter vector $\boldsymbol{\vartheta}$ models the expected degradation curve. Generally, λ_i is assumed to be generated from a normal distribution with mean μ and variance σ^2 to account for the unit-to-unit variability. ε_{ij} is the measurement error which is usually assumed to be independent and normally distributed with mean zero and constant variance δ^2.

Degradation-based analysis is shown to perform better than failure-time distribution in estimation precision and efficiency in terms of representative statistics such as: the number of inspections, the amount of measurement error, etc. In addition, degradation analysis shows its biggest advantage in terms of statistical efficiency, especially when it estimates quantiles of failure probabilities is beyond the range of the data [8].

1.2.3 Maximum likelihood estimation

In degradation analysis, maximum likelihood estimation (MLE) is a general and versatile tool for estimating unknown parameters in degradation models. In general, given a fixed set of data and underlying statistical model, the maximum likelihood method chooses values of the model parameters which will maximize the likelihood function. The likelihood can be written as the joint probability of the data $\boldsymbol{t} = (t_1, \ldots, t_n)$, which is collected from the test.

$$L(\lambda) = L(\lambda|\boldsymbol{t}) = \prod_{i=1}^{n} L_i(\lambda|\boldsymbol{t}_i) \tag{1.3}$$

where $L_i(\lambda|\boldsymbol{t}_i)$ is the likelihood contributed by the ith observation, $\boldsymbol{t}_i$ is the ith observation, λ is the vector of parameters to be estimated. To estimate λ from the available data, we find the values of λ that maximize $L(\lambda)$ [4]. For computation convenience, the logarithm

of likelihood function (i.e. *log-likelihood function*) is used instead of the original likelihood function. Mathematically, the estimators of model parameters are obtained by taking the first partial derivatives of the log-likelihood function and setting these partials equal to zero [2].

The likelihood of the random-effect degradation model in (1.2) can be written as

$$L(\mu, \sigma^2, \delta^2) = \prod_{i=1}^{n} \int \left(\prod_{j=1}^{n_i} \frac{1}{\sqrt{2\pi\delta^2}} \exp\left(-\frac{\left(y_{ij} - g(t_{ij}; \lambda_i)\right)^2}{2\delta^2} \right) \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{(\lambda_i - \mu)^2}{2\sigma^2} \right) \right) d\lambda_i \quad (1.4)$$

To compute the MLEs for the model parameters, it may involve difficulty in obtaining the closed form of the integral. Therefore, various approximation algorithms are proposed for numerical computation such as: Pinheiro and Bates' (PB) algorithm, Lindstrom and Bates' (LB) algorithm [8] and Expectation-Maximization (EM) algorithm [6].

1.2.4 Bayesian inference

In addition to the likelihood-based estimation, Bayesian approach also can make statistical inference for degradation models and have its advantages from several aspects. Being free of approximation method, it is simple for Bayesian approach to derive estimators based on previous knowledge and belief on the parameters. Furthermore, with powerful simulation method like Markov Chain Monte Carlo simulation, prediction of unknown parameters is easy to obtain.

Unlike the likelihood-based method, in Bayesian approach, the unknown parameter $\boldsymbol{\lambda}$ is considered to be a random variable with a probability distribution (called prior distribution of $\boldsymbol{\lambda}$). This prior distribution is specified and provides a theoretical description of information about $\boldsymbol{\lambda}$ that was available before any data was obtained [5].

If $f(\boldsymbol{\lambda})$ is assumed to be the joint prior distribution of $\boldsymbol{\lambda}$, the joint posterior distribution can be derived according to the Bayes' formula by using the likelihood of the

data and the prior distribution on $\boldsymbol{\lambda}$,

$$f(\boldsymbol{\lambda}|t) \propto L(\boldsymbol{\lambda}|t) \times f(\boldsymbol{\lambda}) \tag{1.5}$$

where $f(\boldsymbol{\lambda}|t)$ is the posterior distribution of $\boldsymbol{\lambda}$. With the information in the data (i.e. in the likelihood), the prior distribution is updated according to Bayes' rule to obtain the posterior distribution. The posterior distribution, the conditional distribution of $\boldsymbol{\lambda}$ given the sample $t_1, t_2, \ldots, t_n$, summarizes all of the pertinent information about the parameter $\boldsymbol{\lambda}$.

In reliability analysis, Bayesian approach has its particular advantage of making inference on failure data when the sample size is relatively small. Based on previous knowledge and experiences, Bayesian inference is a more flexible and effective method to estimate parameter and to predict reliability, especially when not enough data can be collected or the information obtained directly from the data is limited.

1.2.5 Gaussian mixture model

A Gaussian mixture model is a parametric PDF represented as a weighted sum of Gaussian component densities [6]. In Gaussian mixture model, it is assumed that the observations are generated from a mixture distribution

$$f(t|\theta) = \sum_{k=1}^{G} \pi_k f_k(t|\boldsymbol{\theta_k}) \tag{1.6}$$

where $\boldsymbol{\theta_k} = (\theta_1, \ldots, \theta_G; \sigma_1^2, \ldots, \sigma_G^2; \pi_1, \ldots, \pi_G)$. π_k is the proportion (or weight) satisfying $\pi_k \geq 0, \sum_{k=1}^{G} \pi_k = 1$. G is the number of components in the mixture model. $f_k(\cdot)$ is the PDF of a normal distribution with parameter vector $\boldsymbol{\theta_k}$. π_k and $\boldsymbol{\theta_k}$ can describe the characteristics of the subgroups. The parameters of Gaussian mixture model are usually estimated by EM algorithm. However, only limited studies made statistical inference of Gaussian mixture model under Bayesian framework.

Gaussian mixture model has been successfully applied in many fields, such as: biometric system, psychology, geology, and astrophysics. Clustering and pattern classification are significant applications of Gaussian mixture model in engineering field.

2 Literature Review

This chapter reviews previous research on degradation analysis by using maximum likelihood and Bayesian methods, reliability analysis considering heterogeneous data, and Gaussian mixture model and its applications.

2.1 Statistical inference on degradation analysis

There are numerous studies on applying different statistical methods to analyze the degradation data. Two most frequently used approaches for statistical inferences are the maximum likelihood method and Bayesian approach.

2.1.1 Maximum likelihood method in degradation analysis

Maximum likelihood, as a long-standing method, has been extensively applied on degradation analysis for years. In accelerated degradation analysis, the majority of the literature suggested the maximum likelihood method for predicting model parameters [7]. Maximum likelihood has been used to study linear or log-linear degradation models. For instance, as an extension for general degradation model, Lu et al. [9] developed a random-effect coefficient regression model and applied likelihood-based estimation for linear degradation data. With consideration of the change point, Bae and Kvam [10] proposed two log-linear regression models to characterize the unstablized stage and used maximum likelihood method for related estimation.

Besides linear degradation models, there are many studies on employing maximum likelihood method for parameter estimations of nonlinear degradation models. In study on fatigue crack growth data, Lu and Meeker [11] suggested a nonlinear random-effect model and developed a least-square based two-stage method for deducing the maximum likelihood estimators. Owing to incomplete burn-in (called aging in the industry) in the manufacturing process, the degradation path is not always monotonic. Therefore, to

analyze the degradation behavior of plasma display panels (PDP) and vacuum fluorescent display (VDF), a general nonlinear random-coefficients model was introduced to describe the non-monotonic degradation behavior. Four approximation method based on the likelihood function were introduced for parameter estimation: first-order method, Lindstrom and Bates' (LB) algorithm, adaptive importance sampling, and adaptive Gaussian quadrature [12]. Bae et al. [13] [14] proposed a bi-exponential model (also called "two-compartment model") with random coefficients for nonlinear degradation paths (a longitudinal behavior) in PDP and membrane electrode assemblies (MEAs) in direct methanol fuel cell (DMFC). Model parameters were inferred by means of maximum likelihood approach with LB algorithm. Taking account for the natural ordering of degradation performance, Huang and Dietrich presented a truncated Weibull distribution to fit degradation path and employed maximum likelihood method for statistical inference [15].

2.1.2 Bayesian approach in degradation analysis

Compared to the application of maximum likelihood method in degradation models, Bayesian approach has not been widely applied to analyze degradation data. Robinson and Crowder [16] applied Bayesian approach to analyze nonlinear growth curves with repeated degradation measures and predicted the failure time distribution of fatigue crack growth data. The considerable advantage of the proposed methodology is to improve computation efficiency by avoiding approximations and high dimensional integrals. Gebraeel et al. [17] developed a Bayesian updating procedure for two exponential degradation models to predict stochastic parameters and obtained a closed-form residual-life distribution for the monitored device. By modeling the degradation process as a Wiener process, Pettit and Young made inferences on model parameters and predicted

items reliability by using Bayesian approach. Gibbs sampling was used to obtain posterior distributions and predictive distributions [18].

2.2 Reliability analysis considering heterogeneous data

Statistical methods in reliability analysis developed in the past usually focused on a single population. However, in practice, the data may come from a heterogeneous population with subgroups. In previous studies, there are several discussions concerning reliability analysis for heterogeneous data. By dividing a failure population into subpopulations, Mendenhall and Hard [19] introduced mixed- exponential distribution to represent different types or causes of failure. Estimates of the population parameters were obtained and sampling was censored at a predetermined test termination time. In survival analysis, a new three-parameter family of Gamma and inverse Gaussian distributions on the positive numbers was proposed for the application on failure distributions with heterogeneous population. For this complicated density, a simple saddlepoint approximation was provided for estimates [20]. Bučar et al. [21] proposed that a finite Weibull mixture with positive proportion to assess the reliability of an arbitrary system. EM algorithm was suggested to estimate the mixing proportion, and a m-fold Weibull mixture was derived. In order to classify the aging properties of the lifetimes of multi-component systems, a generalized mixture of Weibull distributions was studied and the result can also extend the cases to exponential or Rayleigh distributions [22]. Erişoğlu [23] discussed a mixture of Exponential-Gamma, Exponential-Weibull and Gamma-Weibull distributions in order to model heterogeneous survival data. Maximum likelihood approach with EM algorithm was employed for parameter estimation.

2.3 Gaussian mixture model

Gaussian mixture models have been proposed and successfully applied in many fields. In general, likelihood-based method with EM algorithm and Bayesian approach are

two major methods to make statistical inference for Gaussian mixture model. In previous studies, the EM algorithm has been extensively used to estimated the parameters of Gaussian mixture model. For instance, Yang and Ahuja [6] applied a finite Gaussian mixture model to detect human skin color and estimate model parameters by EM algorithm. Reynolds [24] applied Gaussian mixture model for speaker identification and verification. To avoid local optimization, Pernkopf and Bouchaffra [25] presented Gaussian mixture models and applied genetic-based EM algorithm (GA-EM) for parameter estimation. However, EM algorithm has its limitations in three aspects: sensitivity to initial value; local convergence by providing biased estimates for model parameters and having difficulty in assessing model uncertainty. Bayesian approach to estimate model parameters for Gaussian mixture models has been developed because of the great advantages of Bayesian computation. Banfield and Raftery [26] introduced model-based Gaussian and non-Gaussian for clustering analysis. In addition, Roberts et al. [27] implemented Bayesian approach to Gaussian mixture modeling in clustering analysis.

To sum up, when a system has a number of components for analysis, the typical approach is to assume homogeneity among all system components in a statistical model. Previous studies on degradation models generally focused on the data from homogeneous population and most of the studies employed maximum likelihood method for parameter estimation. Meanwhile, mixture models were only developed for lifetime analysis, and Gaussian mixture model by using Bayesian inference has not been widely applied on degradation analysis. However, in degradation analysis, if components can be classified into meaningful subgroup, it may be more reasonable to assume heterogeneity among the system components with mixing of subgroups. Therefore, in the next chapter, a degradation model for heterogeneous data will be proposed.

3 Proposed Methodologies

This chapter proposes a mixture model to describe the heterogeneous degradation data. Both two-stage based maximum likelihood method and Bayesian approach are implemented for the parameter estimation. Moreover, model selection criteria is introduced for choosing the best model to fit the data. At the end of this chapter, the failure time distribution is derived by simulation method.

The notations used in this thesis are list as following:

y_{ij}: jth response on the ith unit

t_{ij}: time of the jth measurement for ith unit

n: total number of units

n_i: number of observations for the ith unit, $i = 1, \ldots, n$

λ_i: random coefficient of the degradation path

ε_{ij}: measurement error with constant variance δ^2

π_k: mixing proportion of the mixture model

μ_k: mean of the kth group

σ_k^2: variance of the kth group

$\mathcal{N}$: normal distribution

$\mathcal{D}$: Dirichlet distribution

$\mathcal{IG}$: inverse gamma distribution

3.1 Degradation model

For each unit from a random sample with n units, degradation measurements are available at specified times. If the degradation path is assumed to be an increasing linear curve, the degradation path of the ith unit can be defined as

$$y_{ij} = g(t_{ij}, \lambda) + \varepsilon_{ij} = \lambda_i t_{ij} + \varepsilon_{ij}, i = 1, \ldots, n, j = 1, \ldots, n_i, \tag{3.1}$$

where ε_{ij} is the measurement error and is assumed to follow a normal distribution with constant variance δ^2, $\varepsilon_{ij} \sim \mathcal{N}(0, \delta^2)$. To describe inherent variability among different units, random-effect model or Bayesian hierarchical model can be employed. In previous studies, if the units are sampling from a homogeneous population, unknown parameter λ_i is assumed to be normally distributed in a random-effect model: $\lambda_i \sim \mathcal{N}(\mu, \sigma^2)$. Under Bayesian framework, parameter vectors with a common population distribution (e.g. normal distribution) are applied to model the unit-to-unit variability.

To consider the possibility that the units may come from a heterogeneous population, it is reasonable to assume that λ_i's are generated from a Gaussian mixture distribution with probability density function

$$f(\lambda_i) = \sum_{k=1}^{G} \pi_k \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(\lambda_i - \mu_k)^2}{2\sigma_k^2}\right), i = 1, \ldots, n, \tag{3.2}$$

where G is the number of mixing subgroups, and π_k is the mixing proportions. To separate units from each other based on similar degradation mechanism, latent variables $\mathbf{z}_i = (z_{i1}, \ldots, z_{iG}), i = 1, \ldots, n$, are introduced according to

$$z_{ik} = \begin{cases} 1, & \text{if the } i\text{th unit belongs to the } k\text{th cluster,} \\ 0, & \text{otherwise.} \end{cases} \tag{3.3}$$

for $k = 1, \ldots, G$. The class indicator $\mathbf{z}_i$ hence follows an identical and independent multinomial distribution of a single trial,

$$f(\mathbf{z}_i|\boldsymbol{\pi}) = \prod_{k=1}^{G} \pi_k^{z_{ik}}, i = 1, \ldots, n, \tag{3.4}$$

where $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_G)$

3.2 Two-stage maximum likelihood estimation

Under the heterogeneity assumption, this section proposes a two-stage ML method and EM algorithm to estimate the model parameters for the mixture model, which extends the inference procedure described by Lu and Meeker [8].

3.2.1 Two-stage maximum likelihood method

With the introduction of latent variables, the likelihood function can be rewritten as:

$$L(\Theta) = \prod_{i=1}^{n} \int \left(\prod_{j=1}^{n_i} \frac{1}{\sqrt{2\pi\delta^2}} \exp\left(-\frac{\left(y_{ij} - \lambda_i t_{ij}\right)}{2\delta^2} \right) \sum_{k=1}^{G} \pi_k \frac{1}{\sqrt{2\pi\sigma_k^2}} exp\left(-\frac{(\lambda_i - \mu_k)^2}{2\sigma_k^2} \right) \right) d\lambda_i \qquad (3.5)$$

where $\Theta = (\mu_i, \ldots, \mu_G; \sigma_i^2, \ldots, \sigma_G^2; \pi_i, \ldots, \pi_G)$. Since it is difficult to directly obtain a close-form expression for the likelihood function, approximation algorithms can be applied to obtain the maximum likelihood estimators. Some approximation approaches like PB algorithm, LB algorithm and EM algorithm have been developed in previous research. However, PB and LB algorithms have only been discussed for estimating unknown random-effect coefficients which are generated from a normal distribution, see Bae and Kvam [12]. To estimate random-effect coefficients in the mixture model, a two-stage approach is proposed as following:

Stage 1: For each degradation path, apply the MLE procedure to estimate the model parameters and obtain $\lambda_i, i = 1, \ldots, n$.

Stage 2: For all λ_i obtained from stage 1, treat λ_i's as "pseudo-data" and estimate the parameters in the Gaussian mixture model.

In the second stage, in order to provide a good approximation for random-effect coefficients from the Gaussian mixture distribution, this thesis uses the EM algorithm for estimation which was discussed by Hwang and Kuo [28] for the estimation procedure. The procedure is implemented by repeating the following two steps:

1. E-step

Given $\hat{\pi}^{(m)}$, $\hat{\mu}_k^{(m)}$, and $\hat{\sigma_k^2}^{(m)}$, $\hat{\gamma}_{ik}^{(m+1)}$ can be computed by the following formula:

$$\hat{\gamma}_{ik}^{(m+1)} = \frac{\hat{\pi}_k^{(m)} f_k\left(\lambda_i | \hat{\pi}_k^{(m)}, \hat{\sigma_k^2}^{(m)}\right)}{\sum_{j=1}^{G} \hat{\pi}_j^{(m)} f_j\left(\lambda_j | \hat{\pi}_j^{(m)}, \hat{\sigma_j^2}^{(m)}\right)} \qquad (3.6)$$

where $\hat{\gamma}_{ik}^{(m+1)}$ is the current guess of the probability that sample y_i came from the kth cluster.

2. M-step

Maximize the likelihood function $L(\mu, \sigma^2, \pi, z|\lambda) = \prod_{i=1}^{n} \prod_{k=1}^{G} \left[\pi_k f_k(\lambda_i|\mu_k, \sigma_k^2)\right]^{z_{ik}}$ with respect to π_k, μ_k, σ_k^2 given $\hat{\gamma}_{ik}^{(m+1)}$, which is the estimate of conditional expectation of $z_{ik}^{(m+1)}$ in the mth iteration. As a result of the maximization, the optimal solution at $(m+1)$th iteration is $\hat{\pi}_k^{(m+1)} = \frac{\sum_{i=1}^{n} \hat{\gamma}_{ik}^{(m)}}{n}$ and $\hat{\mu}_k^{(m+1)}, \hat{\sigma_k^2}^{(m+1)}$ are the optimal solution of the likelihood function

$$L\left(\hat{\mu}_k^{(m+1)}, \hat{\sigma_k^2}^{(m+1)}|\lambda\right) = \prod_{i=1}^{n} \left[\pi_k f_k\left(\lambda_i|\hat{\mu}_k^{(m+1)}, \hat{\sigma_k^2}^{(m+1)}\right)\right]^{z_{ik}^{(m+1)}} \tag{3.7}$$

The two steps are repeated until the convergence criteria are satisfied: the relative improvement of the log likelihood is below a certain threshold ε. For example, in this thesis, the convergence criteria can be written as: $\left|\frac{\ell_k(\Theta)^{(m+1)} - \ell_k(\Theta)^{(m)}}{\ell_k(\Theta)^{(m)}}\right| \leq 0.05$. The initial point $\hat{\gamma}^{(0)}$ is chosen arbitrarily or by certain rules. Since EM algorithm depends on the initial point, the algorithm can be applied with several initial points. The best result can be chosen according to the likelihood [29].

3.2.2 Determination of number of components

Determination of the number of components, G, can be regarded as the selection procedure for different models. Once the number of components is determined, the model which fit data best can be chosen. To measure of the relative goodness of fit of a statistical model, several criteria have been discussed in the previous research like: the Akaike information criterion (AIC) [30], the Bayesian information criterion (BIC) [31], and the Bayes factor [32]. Among these three criteria, the penalty for introducing extra parameters for AIC is weaker than the penalty for BIC. The computational procedure of Bayes factor is very complex. For simplicity, in this thesis, BIC is suggested to determine the number of components. The BIC is approximated as

$$BIC \approx -2\ell_k(\Theta) + d\log(n) \tag{3.8}$$

where $\ell_k(\Theta)$ is the maximized log-likelihood of the model with k components; d is the total number of independent parameters in the model with k components; and n is the sample size. BIC not only rewards goodness of fit, but also includes a penalty that is an increasing function of the number of estimated parameters. However, when this penalty improves the goodness of the fit, it also discourages increasing the number of free parameters in the model. Given a set of candidate models for the data, the preferred model is the one with the minimum BIC value. In order to reduce computational efforts, the first local minimum of the BIC value is chosen. The procedure of model selection based on BIC can be described as following:

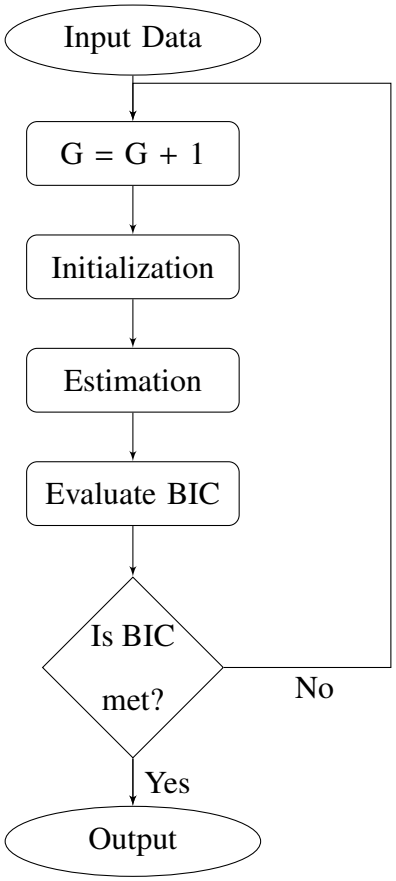

Figure 3.1: Flow chart of parameter estimation and model selection

With the two-stage ML method, model parameters can be estimated and BIC value is calculated given the number of components. Then, the number of components is increased by one and this procedure is repeated. When BIC value reaches the first local minimum, optimal number of components can be decided.

3.2.3 Derivation of failure-time distribution

This section derives the predictive failure-time distribution from the degradation data by using maximum likelihood method. The general nonlinear degradation model given by Equation (3.1) is used to illustrate the procedure.

If y^* is specified as the predetermined degradation level, and the failure time T of one unit can be defined as the first hitting time when the threshold level (i.e. y^*) and the actual degradation path intersects. Hence, the distribution function of T is expressed by

$$F_T(t|\mathbf{\Psi}) = P(T \leq t|\mathbf{\Psi}) = P\left(g(t,\lambda) + \varepsilon \leq y^*|\mathbf{\Psi}\right) = \Phi\left(\frac{y^* - g(t,\lambda)}{\sigma}\right) \tag{3.9}$$

where $\mathbf{\Psi} \equiv (\pi, \mu_k, \sigma_k^2, \delta^2)$ and $\Phi(\cdot)$ denotes the cumulative distribution function (CDF) of the standard normal distribution.

To estimate $F_T(t|\mathbf{\Psi})$ given by equation (3.8), Bae et al. [16] developed a simulation-based algorithm to obtain the estimated value of $F_T(t|\mathbf{\Psi})$ by ML method. In order to estimate $\mathbf{\Psi} = (\pi, \mu_k, \sigma_k^2, \delta^2)$ under ML method, the approximation method can be applied by using PB algorithm [33]. Then $F_T(t|\mathbf{\Psi})$ is estimated by replacing with their estimates $\mathbf{\Psi}$. However, in some circumstances, if $F_T(t|\mathbf{\Psi})$ does not have a closed form, $F_T(t|\mathbf{\Psi})$ can be derived by Markov Chain Monte Carlo simulation(MCMC) described in Lu and Meeker [11].

In MCMC simulation, the estimated parameters $\hat{\mu}_k$ and $\hat{\sigma}_k^2$ can be used to generate the N simulated realizations $\hat{k}$. For N values of $\hat{k}$, by substituting $\hat{k}$ value into degradation model $y^* = g(t,\lambda) + \varepsilon$, the N failure-times $\hat{t}$ can be computed. Then, the failure-time distribution $F_{T|S}(t|\mathbf{\Theta})$ is estimated from the simulated empirical distribution for any time

point t as equation shown below:

$$\hat{F}_T(t) = \frac{number\ of\ (\hat{t} \leq t)}{N}$$

3.3 Bayesian inference

Model parameters of the mixture model can also be estimated under Bayesian framework. In this section, a hierarchical Bayesian approach is proposed to analyze the degradation data.

3.3.1 Prior specification

In Bayesian approach, the first step is to specify the prior distribution for the model parameters. In the proposed model, the prior distribution of mixing proportion π_k follows a Dirichlet distribution: $\pi_k \sim \mathcal{D}(c_1, \ldots, c_G)$.

$$f(\pi_1, \ldots, \pi_k) = \frac{\Gamma(c_1 + \ldots + c_G)}{\Gamma(c_1) \ldots \Gamma(c_G)} \pi_1^{c_1 - 1} \ldots \pi_G^{c_G - 1} \tag{3.10}$$

where $\mathbf{c} = (c_1, \ldots, c_G)$ are pre-specified constants and $\Gamma(\cdot)$ is the Gamma function. The prior distribution of variance δ^2 follows an inverse gamma distribution: $\delta^2 \sim \mathcal{IG}(a, b)$,

$$f(\delta^2) = \frac{b^a}{\Gamma(a)} \left(\delta^2\right)^{-(a+1)} \exp\left(-\frac{b}{\delta^2}\right) \tag{3.11}$$

The prior distribution of mean μ_k follows a normal distribution:
$\mu_k \sim \mathcal{N}(m_k, s_k^2), k = 1, \ldots, G.$

$$f(\mu_k) = \frac{1}{\sqrt{2\pi s_k^2}} \exp\left(-\frac{(\mu_k - m_k)^2}{2s_k^2}\right) \tag{3.12}$$

The prior distribution of variance σ_k^2 follows an inverse gamma distribution:
$\sigma_k^2 \sim \mathcal{IG}(\alpha_k, \beta_k), k = 1, \ldots, G$

$$f(\sigma_k^2) = \frac{\beta_k^{\alpha_k}}{\Gamma(\alpha_k)} \left(\sigma_k^2\right)^{-(\alpha_k+1)} \exp\left(-\frac{\beta_k}{\sigma_k^2}\right) \tag{3.13}$$

3.3.2 Markov Chain Monte Carlo simulation with Gibbs sampling

In the Bayesian framework, inference on each parameter is based on its marginal posterior distribution. For high-dimensional models, marginal posterior distributions of the model parameters need to be obtained by processing multiple levels of integration. In general, the procedure appears not analytically tractable. Under this circumstance, MCMC simulation can be unitized as an efficient way to compute the posterior distributions for Bayesian hierarchical models [34].

In MCMC simulation, Gibbs sampling is one particular MCMC algorithm which is useful for many multidimensional problems. Each iteration of the Gibbs sampling cycles through the unknown parameters, drawing a sample of one parameter conditional on the latest values of all the others. When the number of iteration is large enough, the sample draws on one parameter can be regarded as simulated observation from its marginal distribution. Therefore, the parameters in the model are estimated by simulating from the joint posterior distribution with a Gibbs sampler [34].

According to the Bayesian hierarchical model proposed in 3.3, MCMC simulation at $(n+1)$th iteration consists of the following steps:

(a) Simulate the latent variables $z_i^{(n+1)}$ according to their conditional posterior distributions.

(b) Simulate the mixing proportions $\pi^{(n+1)} = \left(\pi_1^{(n+1)}, \ldots, \pi_G^{(n+1)}\right)$ given $z_i^{(n+1)}$ according to their conditional posterior distributions.

(c) Simulate the mean of mixture normal distributions, $\mu_k^{(n+1)}, k = 1, \ldots, G$ given $z_i^{(n+1)}$ according to their conditional posterior distributions.

(d) Simulate the variance of mixture normal distributions, $\sigma_k^{2(n+1)}, k = 1, \ldots, G$ given $z_i^{(n+1)}$ according to their conditional posterior distributions.

3.3.3 Conditional posterior induction

According to the Bayes' Law, the joint posterior distribution can be derived for all the model parameters conditioning on data as following:

$$\begin{aligned} f(\lambda_1,\ldots,\lambda_i;\mu_1,\sigma_1^2,\ldots,\mu_G,\sigma_G^2;\pi_1,\ldots,\pi_G) \propto & \prod_{i=1}^{n}\prod_{j=1}^{n_i}\left(\delta^2\right)^{-1/2}\exp\left(-\frac{\left(y_{ij}-\lambda_i t_{ij}\right)^2}{2\delta^2}\right) \\ & \times\prod_{i=1}^{n}\prod_{k=1}^{G}\left[\pi_k\left(\sigma_k^2\right)^{-1/2}\exp\left(\frac{(\lambda_i-\mu_k)^2}{2\sigma_k^2}-\right)\right]^{z_{ik}} \\ & \times\prod_{k=1}^{G}\left(s_k^2\right)^{-1/2}\exp\left(-\frac{(\mu_k-m_k)^2}{2s_k^2}\right)\times\prod_{k=1}^{G}\left(\sigma_k^2\right)^{-1/2}\exp\left(-\frac{\beta_k}{\sigma_k^2}\right) \\ & \times\pi_k^{c_k-1}\times\left(\delta^2\right)^{a+1}\exp\left(-\frac{b}{\delta^2}\right) \end{aligned} \tag{3.14}$$

Given the prior distribution, the conditional posterior distribution can be derived.

(a) The conditional posterior distribution of the classification indicator z_i is a multinomial distribution

$$z_i|rest,data\sim Multinomial(1,p_k),p_k=\frac{\pi_k\left(\sigma_k^2\right)^{-1}\exp\left(-\frac{(\lambda_i-\mu_k)^2}{2\sigma_k^2}\right)}{\sum_{k=1}^{G}\pi_k\left(\sigma_k^2\right)^{-1}\exp\left(-\frac{(\lambda_i-\mu_k)^2}{2\sigma_k^2}\right)},k=1,\ldots,G \tag{3.15}$$

(b) The conditional posterior distribution of the mixing proportions π_j is a Dirichlet distribution

$$f(\pi|rest,data)\propto\prod_{i=1}^{n}\prod_{k=1}^{G}(\pi_k)^{z_{ik}}\times\prod_{k=1}^{G}\pi_k^{c_k-1} \tag{3.16}$$

$$\pi|rest,data\sim\mathcal{D}\left(c_1+\sum_{i=1}^{n}z_{i1},\ldots,c_G+\sum_{i=1}^{n}z_{iG}\right) \tag{3.17}$$

(c) The conditional posterior distribution of λ_i is a normal distribution

$$f(\lambda|rest, data) \propto \prod_{j=1}^{n_i} \exp\left(-\frac{\left(y_{ij}-\lambda_i t_{ij}\right)^2}{2\delta^2}\right) \times \exp\left(\frac{(\lambda_i-\mu_k)^2}{2\sigma_k^2}\right)$$

$$\propto \exp\left(-\frac{\left(\lambda_i - \frac{\mu_{\gamma_i}\delta^2+\sigma_{\gamma_i}^2\sum_{j=i}^{n_i} y_{ij}t_{ij}}{\sigma_{\gamma_i}^2+\sum_{j=i}^{n_i} t_{ij}^2+\delta^2}\right)^2}{2\frac{\delta^2\sigma_{\gamma_i}^2}{\sigma_{\gamma_i}^2\sum_{j=1}^{n_i} t_{ij}^2+\delta^2}}\right) \tag{3.18}$$

$$\lambda|rest, data \sim \mathcal{N}\left(\frac{\mu_{\gamma_i}\delta^2+\sigma_{\gamma_i}^2\sum_{j=i}^{n_i} y_{ij}t_{ij}}{\sigma_{\gamma_i}^2\sum_{j=1}^{n_i} t_{ij}^2+\delta^2}, \frac{\delta^2\sigma_{\gamma_i}^2}{\sigma_{\gamma_i}^2\sum_{j=1}^{n_i} t_{ij}^2+\delta^2}\right) \tag{3.19}$$

where $\gamma_i = k$ if $z_{ik} = 1$.

(d) The conditional posterior distribution of δ^2 is an inverse gamma distribution

$$f(\delta^2|rest, data) \propto (\delta^2)^{-\frac{-\sum_{i=1}^{n} n_i}{2}} \exp\left(-\frac{\sum_{i=1}^{n}\sum_{j=1}^{n_i}\left(y_{ij}-\lambda_i t_{ij}\right)^2}{2\delta^2}\right)$$

$$\times (\delta^2)^{-(a+1)} \exp\left(-\frac{b}{\delta^2}\right) \tag{3.20}$$

$$\delta^2|rest, data \sim \mathcal{IG}\left(a+\frac{\sum_{i=1}^{n} n_i}{2}, b+\frac{\sum_{i=1}^{n}\sum_{j=1}^{n_i}\left(y_{ij}-\lambda_i t_{ij}\right)^2}{2}\right) \tag{3.21}$$

(e) The conditional posterior distribution of μ_k is a normal distribution

$$f(\mu_k|rest, data) \propto \prod_{k=1}^{G}\left[\exp\left(-\frac{(\lambda_i-\mu_k)^2}{2\sigma_k^2}\right)\right]^{z_{ik}} \times \prod_{k=1}^{G}\exp\left(-\frac{(\mu_{ki}-m_k)^2}{2s_k^2}\right)$$

$$\propto \exp\left(-\frac{\left(\mu_k - \frac{s_k^2\sum_{i:z_{ik}=1}\lambda_i\sigma_k^2 m_k}{s_k^2\sum_{i=1}^{n} z_{ik}+\sigma_k^2}\right)^2}{2\frac{\sigma_k^2 s_k^2}{\sigma_k^2+s_k^2\sum_{i=1}^{n} z_{ik}}}\right) \tag{3.22}$$

$$\mu_k|rest, data \sim \mathcal{N}\left(\frac{s_k^2\sum_{i:z_{ik}=1}\lambda_i\sigma_k^2 m_k}{s_k^2\sum_{i=1}^{n} z_{ik}+\sigma_k^2}, \frac{\sigma_k^2 s_k^2}{\sigma_k^2+s_k^2\sum_{i=1}^{n} z_{ik}}\right) \tag{3.23}$$

(f) The conditional posterior distribution of σ_k^2 is an inverse gamma distribution

$$f(\sigma_k^2|rest, data) \propto (\sigma_k^2)^{-\frac{\sum_{i=1}^{n} z_{ik}}{2}} \times \prod_{k=1}^{G}\left[\exp\left(-\frac{(\lambda_i-\mu_k)^2}{2\sigma_k^2}\right)\right]^{z_{ik}}$$

$$\propto (\sigma_k^2)^{-(\alpha_k+1)} \exp\left(-\frac{\beta_k}{\sigma_k^2}\right) \tag{3.24}$$

$$\sigma_k^2|rest, data \sim \mathcal{IG}\left(\alpha_k + \frac{\sum_{i=1}^{n} z_{ik}}{2}, \beta_k + \frac{\sum_{i:z_{ik}=1}(\lambda_i - \mu_k)^2}{2}\right) \tag{3.25}$$

3.3.4 Determination of number of components

Similarly, BIC criteria is used to determine the number of components under Bayesian framework. Given number of components, model parameters can be estimated by Bayesian approach and BIC value is calculated. The number of components is increased by one and repeat the procedure. When BIC value reaches the first local minimum, optimal number of components is decided.

3.3.5 Derivation of failure-time distribution

This section derives the predictive failure-time distribution from the degradation data under the Bayesian formula. The general degradation model given by Equation (3.1) is used to illustrate the procedure.

By modifying the Bayesian approach discussed by Robinson and Crowder [16], this thesis proposes a simulation method to estimate the failure-time distribution from the degradation data. The Bayesian posterior predictive distribution function of the failure distribution is given by

$$\pi\left(F_T(t)|y\right) = \int_{\mathbf{\Psi}} P(g(t,\lambda) + \varepsilon \leq y^*|\mathbf{\Psi})\pi(\mathbf{\Psi}|y)d\mathbf{\Psi} \tag{3.26}$$

where $\pi(\mathbf{\Psi}|y)$ is the posterior distribution of $\mathbf{\Psi}$. In general, the integration does not have a closed form. However, the Gibbs sampling algorithm provides a straightforward sampling-based solution to such problem. The procedure is described as follows:

If M is assumed as the total number of time points, the failure-time distribution $F_T(t_k), k = 1, \ldots, M$ can be estimated for a pre-specified set of time points t_k. In lth iteration of the Gibbs sampling procedure, the Gibbs sampler simulates values for all the model parameters at lth iteration as $\Theta^{(l)} \equiv (\pi^{(l)}, \mu_k^{(l)}, \sigma_k^{2(l)}, \delta^{2(l)})$. Meanwhile, a simulated value for $F_T(t_k)$ can also be generated , denoted by ${F_T}^{(l)}(t_k)$, for $k = 1, \ldots, M$. To simulate

the value for ${F_T}^{(l)}(t_k)$, n vectors according to $\lambda_i \sim \sum_{k=1}^{G} \pi_k \mathcal{N}(\mu_k, \sigma_k^2)$ for large number of n are randomly generated, denote them as ${\lambda_1}^{(l)}, \ldots, {\lambda_n}^{(l)}$. Consider the jth vector ${\lambda_j}^{(l)}$, the expected response value at time t_k can be calculated, denoted by ${y_i}^{(l)}(t_k) = g\left(t_k, {\lambda_j}^{(l)}\right)$. With formulae ${F_i}^{(l)}(t_k) = \Phi\left(\frac{y^* - {y_i}^{(l)}(t_k)}{\sigma^{(l)}}\right)$, n values for $F^{(l)}(t_k)$ are obtained , denoted by $F_1^{(l)}(t_k), \ldots, F_n^{(l)}(t_k)$. Then, a simulated value for $F(t_k)$ can be obtained by taking average of $F_1^{(l)}(t_k), \ldots, F_n^{(l)}(t_k)$. After N Gibbs sampling iterations, a random sample for $F(t_k)$ from its posterior distribution is obtained, i.e. $F^{(1)}(t_k), \ldots, F^{(N)}(t_k)$. Posterior inference for $F(t_k)$ can be obtained based on the sample statistics.

4 Case Study

In this chapter, a case study is presented to illustrate the methodologies discussed in Chapter 3. A mixture model is proposed for the experimental data. Maximum likelihood method and Bayesian approach can be employed to estimate unknown parameters in the model, respectively. The results for each method are compared in the end. The number of components for the data is determined by BIC criteria, and the best model can be selected as well. MATLAB and WinBUGS are used to perform statistical computing algorithms, and the codes are listed in Appendix.

4.1 Analysis with two-stage maximum likelihood method

The experimental data is collected from "Percent increase in operating current for GaAs lasers test" [1]. Over the life of some laser devices, degradation causes a decrease in light output. Some lasers, however, contain a feedback mechanism that will maintain nearly constant light output by increasing operating current as the lasers degrade. When the operating current gets too high, the device is considered to have failed. The experimental data shows the increase in operating current over time for a sample of GaAs lasers test at 80°C (this temperature, though much higher than the use temperature, was used to accelerate the failure mechanism so that degradation information would be obtained more rapidly). Table 4.1 presents the data obtained from the experiment. In this experiment, 15 unique units were under the test, and each unit was observed over 17 equally spaced time points (from 0 to 4000 hours). At each time point, the percent increase in operating current was obtained. "Failure" of the GaAs lasers is defined to occur when the increase level in operating current reaches 10 percent during observation time.

Table 4.1: Percent increase in operating current for GaAs Lasers tested in 80°C [1]

Times(hours)	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14	#15
0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
250	0.47	0.71	0.71	0.36	0.27	0.36	0.36	0.46	0.51	0.41	0.44	0.39	0.30	0.44	0.51
500	0.93	1.22	1.17	0.62	0.61	1.39	0.92	1.07	0.93	1.49	1.00	0.80	0.74	0.70	0.83
750	2.11	1.90	1.73	1.36	1.11	1.95	1.21	1.42	1.57	2.38	1.57	1.35	1.52	1.05	1.29
1000	2.72	2.30	1.99	1.95	1.77	2.86	1.46	1.77	1.96	3.00	1.96	1.74	1.85	1.35	1.52
1250	3.51	2.87	2.53	2.30	2.06	3.46	1.93	2.11	2.59	3.84	2.51	2.98	2.39	1.80	1.91
1500	4.34	3.75	2.97	2.95	2.58	3.81	2.39	2.40	3.29	4.50	2.84	3.59	2.95	2.55	2.27
1750	4.91	4.42	3.30	3.39	2.99	4.53	2.68	2.78	3.61	5.25	3.47	4.03	3.51	2.83	2.78
2000	5.48	4.99	3.94	3.79	3.38	5.35	2.94	3.02	4.11	6.26	4.01	4.44	3.92	3.39	3.42
2250	5.99	5.51	4.16	4.11	4.05	5.92	3.42	3.29	4.60	7.05	4.51	4.79	5.03	3.72	3.78
2500	6.72	6.07	4.45	4.50	4.63	6.71	4.09	3.75	4.91	7.80	4.80	5.22	5.47	4.09	4.11
2750	7.13	6.64	4.89	4.72	5.24	7.70	4.58	4.16	5.34	8.32	5.20	5.48	5.84	4.83	4.38
3000	8.00	7.16	5.27	4.98	5.62	8.61	4.84	4.76	5.84	8.93	5.66	5.96	6.50	5.41	4.63
3250	8.92	7.78	5.69	5.28	6.04	9.15	5.11	5.16	6.40	9.55	6.20	6.23	6.94	5.76	5.38
3500	9.49	8.42	6.02	5.61	6.32	9.95	5.57	5.46	6.84	10.45	6.54	6.99	7.39	6.14	5.84
3750	9.87	8.91	6.45	5.95	7.10	10.49	6.11	5.81	7.20	11.28	6.96	7.37	7.85	6.51	6.16
4000	10.94	9.28	6.88	6.14	7.59	11.01	7.17	6.24	7.88	12.21	7.42	7.88	8.09	6.88	6.62

Percent increase in operating current is plotted over inspection time points. As indicated in Figure 4.1, it is not difficult to find that the 15 observed units show different degradation rates. The units in group 1 have steep slopes and degrade more rapidly; while the rest units in group 2 have mild slopes and degrade gracefully. Therefore, it is reasonable to assume that the units may be divided into two subgroups (i.e. weak group with steeper slopes and strong group with mild slopes).

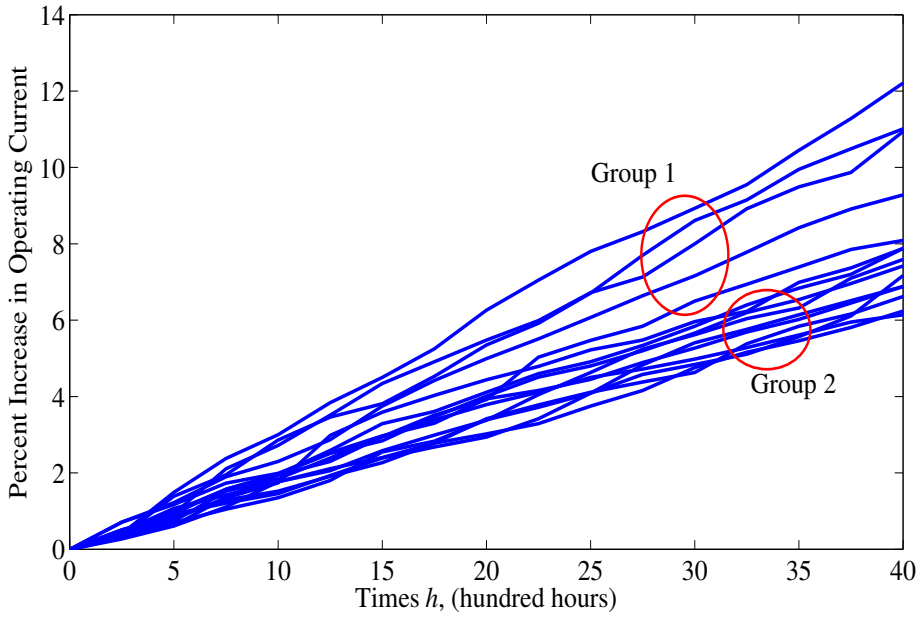

Figure 4.1: Plot of percent increase in operating current for GaAs lasers tested at 80°C

In order to determine number of groups for the data, a mixture model in Chapter 3 is proposed to model the experimental data. Two-stage ML method and EM algorithm are used to estimate the model parameters. The computing procedure is implemented in Matlab. Under ML method, the number of groups, the mean and variance within each group and corresponding BIC value are listed respectively in Table 4.2.

Table 4.2: MLEs of the parameters in GaAs Laser model

Group#	Subgroup	μ	σ^2	BIC
1	No	0.205	0.200×10^{-6}	-45.253
2	G1	0.272	0.050×10^{-6}	-46.773
	G2	0.180	0.030×10^{-6}	
3	G1	0.272	0.031×10^{-6}	-41.641
	G2	0.192	5.280×10^{-6}	
	G3	0.164	1.470×10^{-6}	

By comparing the BIC values, the first local-minimum BIC was achieved when the units are classified into two subgroups. Moreover, the two-stage ML method can also give the information that each unit belongs to which subgroup and the proportion of each subgroup.

Table 4.3: Unit allocation by ML method in GaAs laser model

Subgroup	Proportion	Unit#
1	0.268	1,2,6,10
2	0.732	3,4,5,7,8,9,11,12,13,14,15

4.2 Analysis with Bayesian approach

The model parameters are also estimated by Bayesian approach. In order to compute the posterior mean of model parameters, all all the noninformative priors should be assigned.

If there is only one group in the data, the prior distribution can be assigned as: $\mu \sim \mathcal{N}(0, 10^{-6})$, $\sigma^2 \sim \mathcal{IG}(1, 0.0001)$ and $\delta^2 \sim \mathcal{IG}(1, 0.0001)$. If the data are coming from more than one group, for instance, the data is consisted of two subgroups, the prior distribution can be specified as: $\mu_1 \sim \mathcal{N}(0, 10^{-6})$, $\mu_2 \sim \mathcal{N}(0, 10^{-6})$, $\sigma_1^2 \sim \mathcal{IG}(1, 0.0001)$, $\sigma_2^2 \sim \mathcal{IG}(1, 0.0001)$ and $\pi \sim \mathcal{D}(1, 1)$.

Once the prior distributions are assigned, Bayesian posterior statistics can be inferred by Gibbs sampling algorithm, which was implemented by both Matlab and WinBUGS. With posterior mean and variance, BIC is calculated and compared for single-component model and two-components model. The posterior mean and variance of model parameters and BIC value are summarized in Table 4.4.

Table 4.4: Bayesian estimates for the parameters in GaAs laser model

Group#	Subgroup	μ	σ^2	BIC
1	No	0.205	0.215×10^{-6}	-45.280
2	G1	0.266	0.103×10^{-6}	-45.913
	G2	0.180	0.039×10^{-6}	

When the data is divided into 3 subgroups, the result failed to converge. After making comparison between the estimates obtained by ML method and Bayesian approach, it indicates that the two methods provide very similar results. Then consistency of results from the two estimation methods validates the proposed degradation model. Moreover, identical with ML method's conclusion, results from Bayesian approach also indicates that two-component model is more appropriate for the experimental data. The information regarding group proportion and subgroup items can be concluded in Table 4.5.

Table 4.5: Unit allocation by Bayesian approach in GaAs laser model

Subgroup	Proportion	Unit#
1	0.324	1,2,6,10
2	0.676	3,4,5,7,8,9,11,12,13,14,15

Given all the estimated value for the model parameters, the degradation paths of the units can be re-plotted based on similar degradation patterns. In Figure 4.2, it is more rigorous to classify the degradation path into two subgroups compared to the initial degradation plot in Figure 4.1.

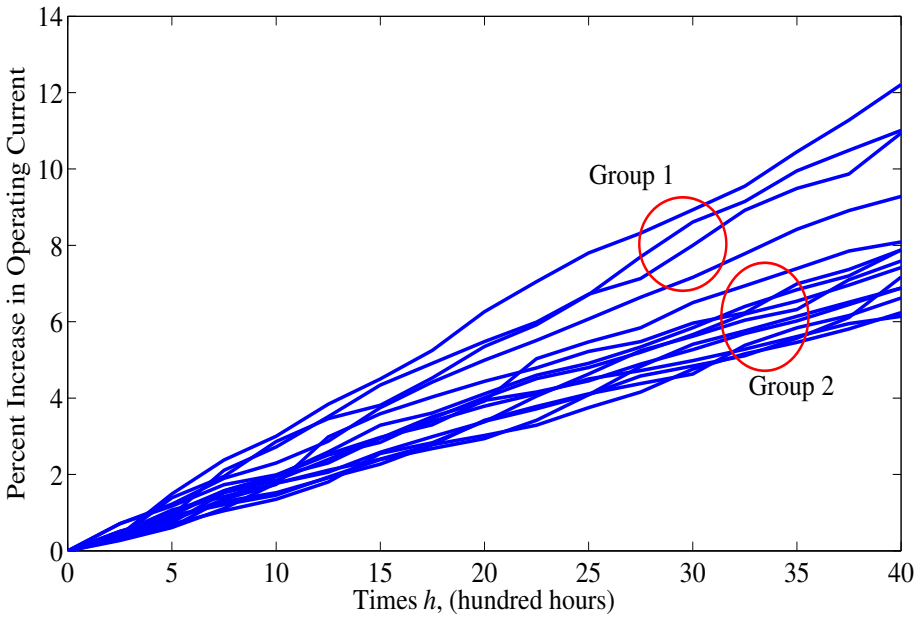

Figure 4.2: Plot of percent increase in operating current for GaAs lasers tested at 80°C with 2 subgroups

4.3 Failure time distribution

By applying the simulation methods introduced in Chapter 3, failure time distributions are derived and plotted by both ML method and Bayesian approach.

If there is only one group in the experimental data, with MLEs and Bayesian posterior statistics, the failure time distribution can be plotted respectively in Figure 4.3. The figure indicates that these two methods give very close failure time distribution. Similarly, if the experimental data can be classified into two subgroups, the failure time distribution plotted by two proposed methods appear to be similar not only for each individual group (Group 1, Group2, respectively), but also for the mixed group (Group 1 and Group 2 together).

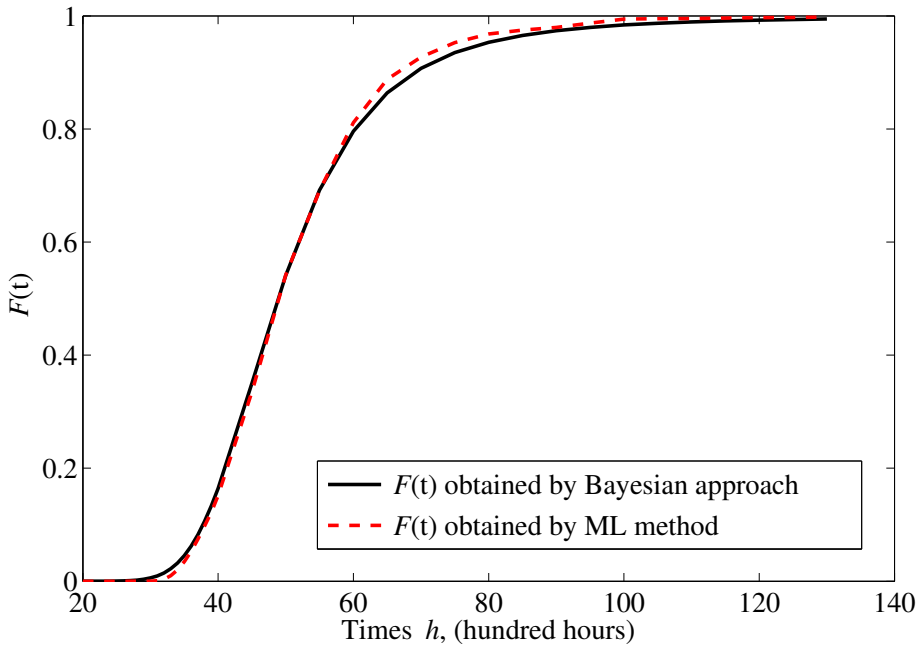

Figure 4.3: $F(t)$ comparison with single component in GaAs laser model

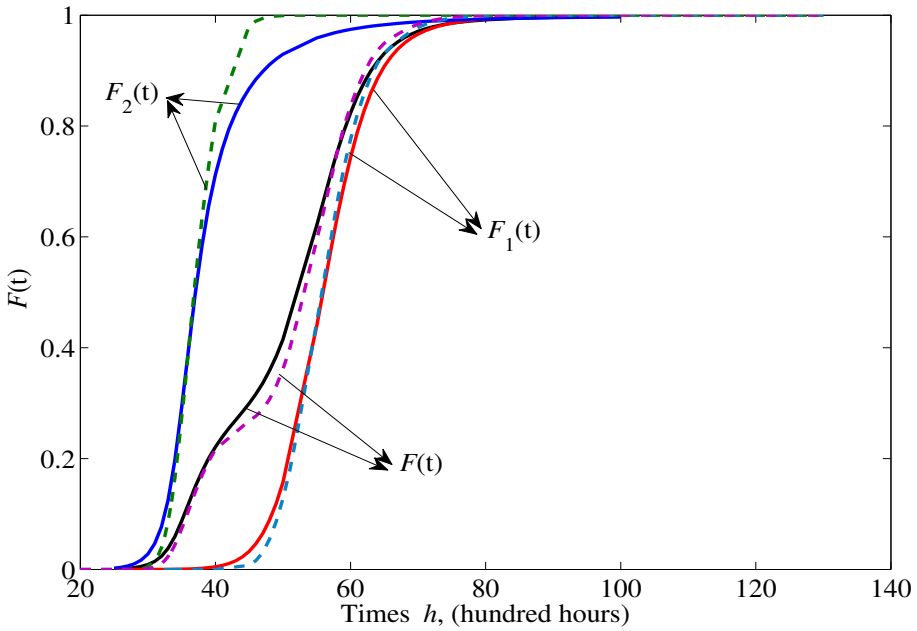

Figure 4.4: F(t) comparison with two components in GaAs laser model

Under Bayesian framework, when sampling the failure-time distribution in Gibbs sampling procedures, the number of simulated model parameters is set to 2000. If tested units are assumed to be sampled from one group, the failure time distribution and 95% point wise confidence interval are plotted in Figure 4.5. If experimental units are classified into 2 subgroups, mean of each individual's failure time distribution are plotted by the dash and dash-dot curves. The solid curve indicates the mean of failure time distribution for the mixed group.

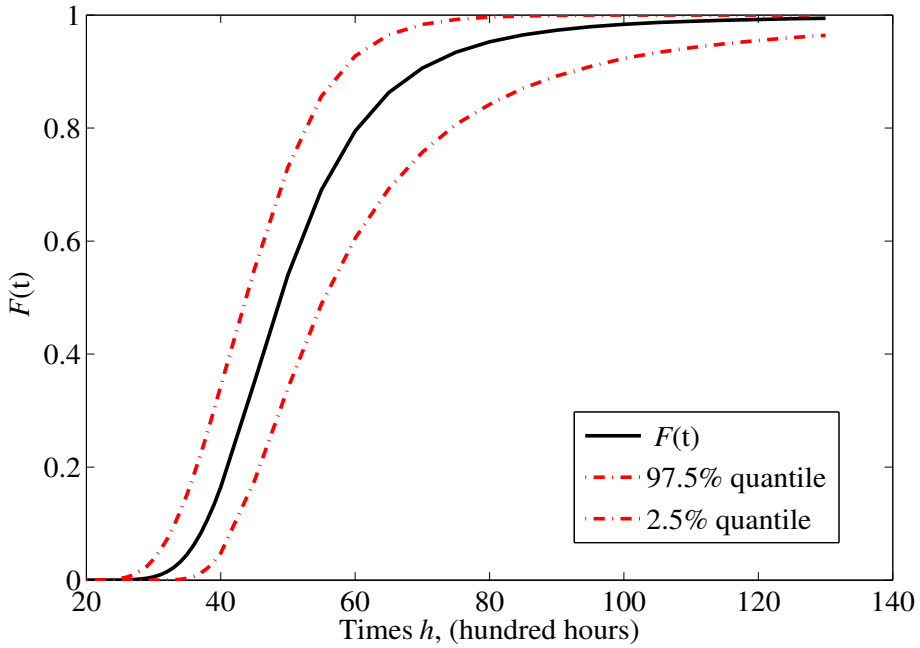

Figure 4.5: $F(t)$ and 95% C.I. with single component

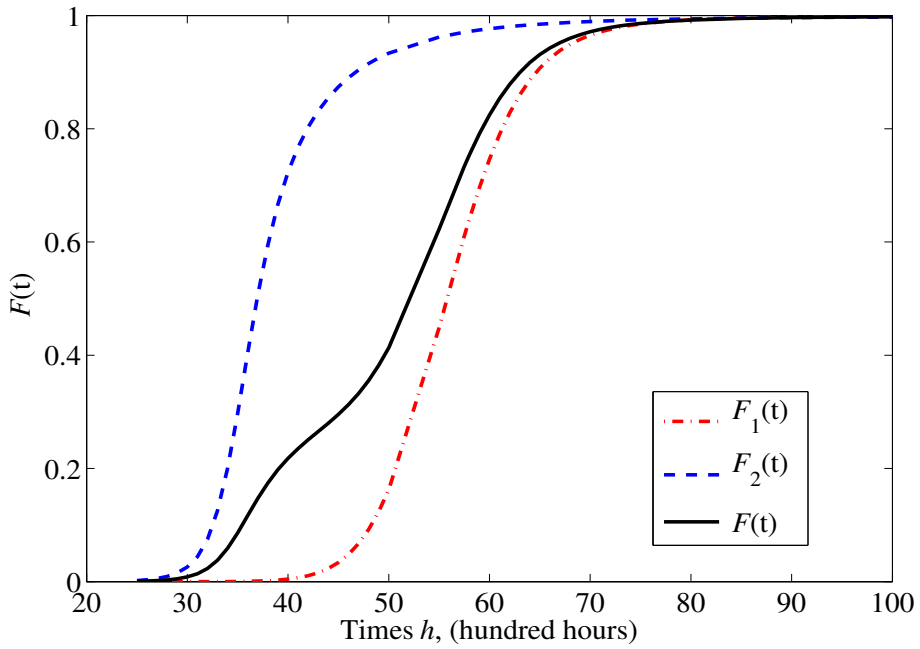

Figure 4.6: $F(t)$ with two components

In Figure 4.7, for each failure time curve above, the 95% point wise confidence interval are plotted as well. As shown, all the solid curves stand for the mean value of the failure time distribution(F(t),F_1(t) and F_2(t), respectively), while the dash-dot curves indicate corresponding 95% point wise confidence interval for each F(t).

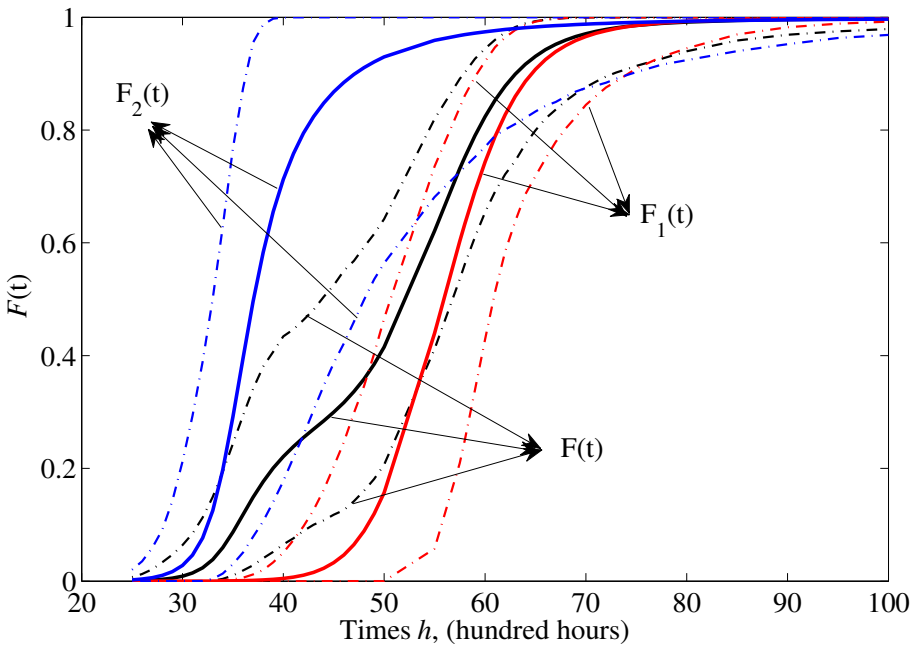

Figure 4.7: F(t) and 95% C. I. with two components

In addition, in order to decide best candidate model, another effective way is to compare probability plot with the failure-time distribution of single-component, two-components. In the figure, if the probability plot is more closer to one degradation path, the degradation model will fit the data better than the other models. In order to construct the probability plot, each failure time t_i is plotted against an estimated value of the probability of failing for the corresponding time $F(t_i)$. Because of skewness and convenience of use, the estimate of $F(t_i)$ by using the approximation to the median

plotting position $\hat{F}(t_i) = \frac{i-0.3}{n+0.4}$, where i is the number of failure unit observed at time t_i and n is the number of units under test.

In this problem, $y^* = 10\%$ increase in current is the specified failure level. All the t_i can be obtained by $t = \frac{y^*}{\hat{\lambda}_i}$, where $\hat{\lambda}_i$ is the estimated value of model parameter λ_i. Therefore, if $y^* = 10\%$ increase in current occurs before 4000 hours, the failure times t_i can be read directly from the figure. However, if $y^* = 10\%$ increase in current occurs after 4000 hours, the pseudo failure times t_i is obtained by prolonging the degradation path until it intersects with the threshold line. Figure 4.8 presents failure time distribution for single-component, two-components and a comparison among single-component, two-components and median plotting position. As shown in the figure, it is obvious to conclude that two-components failure-time distribution capture the curvature of median plotting position. Therefore, the probability plot method also indicates that that two-components degradation model fits data better than single-component model.

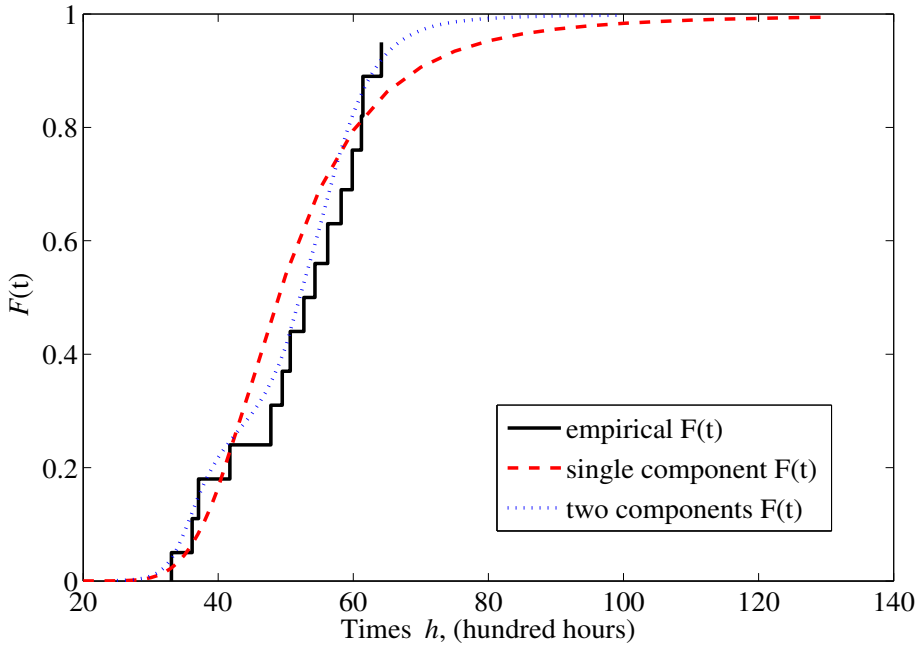

Figure 4.8: Comparison among probability plot, F(t) with single component and two components in GaAs laser model

5 Conclusion and Future Research

This chapter concludes the methodologies and case study proposed in this thesis. In addition, future research is discussed in the end.

5.1 Conclusions

For units with high reliability, degradation analysis which uses the level degradation as an alternative way to express failure, is important and convenient in conducting reliability analysis. In existing studies, most of the degradation models have homogeneity assumption, and the random effects in the degradation models are generally assumed to be normally distributed. However, in practical problems, it is possible that the degradation behavior among different units may vary a lot due to different quality and degradation mechanism. Under this circumstance, the homogeneity assumption is ineffective and normal distribution is not adequate to model the degradation data. Although reliability analysis for units from a heterogeneous population with subgroups has been considered, it was only discussed in failure time analysis.

This thesis proposes a degradation model to consider the units sampled from a heterogeneous population. Instead of using a normal distribution, this thesis proposes a mixture model to describe the random effects. Both two-stage ML method and Bayesian approaches are developed in this thesis for statistical inferences for model parameters, and for the derivation of the failure-time distributions. In ML method, EM algorithm is used for model parameter estimation. Under Bayesian framework, Gibbs sampling algorithm is employed to obtain the estimates. Software Matlab and WinBUGS are employed to for simulation and computation. Meanwhile, in order to determine the number of components, BIC is use as model selection criteria to choose best model to fit data.

To illustrate the proposed methodologies, a practical example is used: the degradation data of 15 GaAs lasers are get by measuring percentage increase input current

as time passes by. Proposed degradation model and algorithms are used to analyze the degradation pattern of the lasers. The estimates obtained from ML method and Bayesian posterior statistics show that two methods provide very similar results in mean, variance and group promotion for each subgroup. As a result, the test lasers can be classified into two groups. 4 out of 15 lasers are in the weak group since they degrade more rapidly, and the remaining 11 units are in the strong group with a slower degradation rate. According to BIC, model with two-components returns the first local minimum BIC value, which means the number of subgroups should be two.

In summary, the main contribution of this thesis can be concluded from three aspects: (1) propose a mixture model to fit the degradation data with subgroups, and find out that the mixture model can fit heterogeneous data better than the general degradation model; (2) developed both ML method and Bayesian approach to estimate the model parameters, and the results show these two method give similar estimates of parameters and failure-time distributions; (3) given the grouping information, the proposed methodology can provide information regarding different failure mechanism or different quality of the products, which may help the manufacturers to improve the products' quality and reliability.

5.2 Future research

This thesis proposed a degradation model with a linearity assumption for the data. However, for some practical problems, it is possible that the non-linear degradation models may be needed to fit the data. Therefore, in the future research, the linear model in this thesis can be extended to non-linear degradation models.

Another future research topic is residual-life distribution and maintenance optimization. Since residual-life distribution can provide estimates of failure time, it can be used for maintenance management for a system. With accurate assessment of system

failure time information, precise evaluation the remaining life of specific degrading components can be estimated, which will effectively reduce the cost of maintenance activities [36].

In addition, condition monitoring, as one significant application of degradation analysis, can be used to evaluate the performance of a system by capturing the current condition of the system and predict the future state of the system [7]. Hence, unnecessary system maintenance can be effectively eliminated and failures become more predictable, which bring the benefits such as reduce cost and enhance reliability of the system. In the future research, condition monitoring can be used for maintenance optimization.

REFERENCES

[1] W. Q. Meeker and L. A. Escobar, *Statistical Methods for Reliability Data*, New York: Wiley, 1998.

[2] C. E. Ebeling, *An Introduction to Reliability and Maintainability Engineering*, Long Grove, IL: WaveLand Press, Inc, 1997.

[3] I. B. Gertsbackh and K. B. Kordonskiy, *Models of Failure*, New York: Springer-Verlag, 1969.

[4] Q. Feng, H. Peng, and D. W. Coit, "A degradation-based model for joint optimization of burn-in, quality inspection, and maintenance: a light display device application," *International Advanced Journal of Manufacturing Technology*, vol. 50, pp. 801-808, 2010.

[5] D. D. Wackerly, W. Mendenhall, and R. L. Scheaffer, *Mathematical Statistics with Applications*, 7th ed, Belmont: Thomson Brooks, 2008.

[6] M. H. Yang and N. Ahuja, "Gaussian mixture model for human skin color and its applications in image and video databases," *Proceedings of International society for optics and photonics*, vol. 3656, pp. 458-466.

[7] W. B. Nelson, *Accelerated Testing: Statistical Models, Test Plans and Data Analysis*, New York: Wiley, 1990.

[8] C. J. Lu, W. Q. Meeker, and L.A. Escobar, "A comparison of degradation and failure-time analysis methods for estimating a time-to-failure distribution," *Statistica Sinica*, vol. 6, pp. 531-546, 1996.

[9] J. C. Lu, J. Park, and Q. Yang, "Statistical inference of a time-to-failure distribution derived from linear degradation data," *Technometrics*, vol. 39, pp. 391-400, 1997.

[10] S. J. Bae and P. H. Kvam, "A change-point analysis for modeling incomplete burn-in for light displays," *IIE Transactions*, vol. 38, no. 6, pp. 489-498, 2006.

[11] C. J. Lu and W. Q. Meeker, "Using degradation measures to estimate of time-to-failure distribution," *Technometrics*, vol. 35, pp. 161-176, 1993.

[12] S. J. Bae and P. H. Kvam, "A nonlinear random-coefficients model for degradation testing," *Technometric*, vol. 46, no. 3, pp.460-469, 2004.

[13] S. J. Bae, S. J. Kim, M. S. Kim, B. J. Lee, and C. W. Kang, "Degradation analysis of nano-contamination in plasma display panels," *IEEE Transactions on Reliability*, vol. 57, no. 2, pp. 222-229, 2008.

[14] S. J. Bae, S.-J. Kim, S.-K. Um, J.-Y. Park, J.-H. Lee, and H. Cho, "A prediction model of degradation rate for membrane electrode assemblies in direct methanol fuel cells," *International Journal of Hydrogen Energy*, vol. 34, pp. 5749-5758, 2009.

[15] W. Huang and D. L. Dietrich, "An alternative degradation reliability modeling approach using maximum likelihood estimation," *IEEE Transactions on Reliability*, vol. 54, pp. 310-317, 2005.

[16] M. E. Robinson and M. J. Crowder, "Bayesian methods for a growth-curve degradation model with repeated measures," *Lifetime Data Analysis*, vol. 6, pp. 357-374, 2000.

[17] N. Gebraeel, M. Lawley, R. Li, and J. K. Ryan, "Life distribution from component degradation signals: a Bayesian approach," *IIE Transactions*, vol. 37, no. 6, pp. 1425-1436, 2005.

[18] L. I. Pettit and K. D. S. Young, "Bayesian analysis for inverse gaussian lifetime data with measures of degradation," *Journal of Statistical Computation and Simulation*, vol. 63, pp. 217-234, 1999.

[19] W. Mendenhall and R. J. Harder, "Estimation of parameters of mixed exponentially distributed failure time distributions from censored life test data," *Biometrika*, vol. 45, pp. 504-520, 1958.

[20] P. Hougaard, "Survival models for heterogeneous populations derived from stable distributions," *Biometrika*, vol. 73, pp. 387-396, 1986.

[21] T. Bučar, M. Nagode and M. Fajdiga, "Reliability approximation using finite Weibull mixture distributions," *Reliability Engineering and System Safety*, vol. 84, pp. 241-251, 2004.

[22] M. Franco, J. M. Vivo, and N. Balakrishnan, "Reliability properties of generalized mixtures of Weibull distributions with a common shape parameter," *Journal of Statistical Planning and Inference*, vol. 141, pp. 2600-2613, 2011.

[23] U. Erişoğlu, M. Erişoğlu, and H. Erol "A mixture model of two different distributions approach to the analysis of heterogeneous survival data," *World Academy of Science, Engineering and Technology*, vol. 78, pp. 41-45, 2011.

[24] D. A. Reynolds, "Speaker identification and verification using Gaussian mixture speaker models," *Speech Communication*, vol. 17, pp. 91-108, 1995.

[25] F. Pernkopf and D. Bouchaffra, "Genetic-based EM algorithm for learning Gaussian mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 1344-1348, 2005.

[26] J. D., Banfield and A. E. Raftery, "Model-based Gaussian and non-Gaussian clustering," *Biometrics*, vol. 49, pp. 803-821, 1993.

[27] S.J. Roberts, D. Husmeier, I. Rezek and W. Penny, "Bayesian approaches to Gaussian mixture modeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 1133-1142, 1998.

[28] J. Y. Hwang and W. Kuo "Model-based clustering for integrated circuit yield enhancement," *European Journal of Operational Research*, vol. 178, pp. 143-153, 2007.

[29] G. Celeux and G. Govaert, "A classification EM algorithm for clustering and two stochastic versions," *Computational Statistics and Data Analysis*, vol. 14, pp. 315-332, 1992.

[30] H. Akaike, "Information theory and an extension of the maximum likelihood principle," *Second International Symposium on Information Theory*, vol. 1, pp. 267-281, 1973.

[31] G. Schwarz, "Estimating the dimension of a model," *The Annuals of Statistics*, vol. 6, pp. 461-464, 1978.

[32] H. Bensmail, G. Celeux, A. Raftery, and C. P. Robert, "Inference in model-based cluster analysis," *Statistics and Computing*, vol. 7, pp. 1-10, 1997.

[33] J. C. Pinheiro and D. M. Bates, "Approximations to the log-likelihood function in the nonlinear mixed-effects model," *Journal of Computational and Graphical Statistics*, vol. 4, pp. 12-35, 1995.

[34] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, *Bayesian Data Analysis, 2nd edition*, London: Chapman and Hall, 2004.

[35] M. A. Freitas, M. G. Toledo, E. A. Colosimo and M. C. Pires, "Using degradation data to assess reliability: A case study on train wheel degradation," *Quality and Reliability Engineering International*, vol. 25, pp. 607-629, 2009

[36] N. Gebraeel, "Sensory-updated residual life distributions for components with exponential degradation patterns," *IEEE Transactions on Automation Science and Engineering*, vol. 3, no. 4, pp. 382-393, 2006.

APPENDIX: APPENDIX

This chapter includes MATLAB codes for computation procedure in this thesis.

A.1 Two stage ML method for parameter estimation and failure time distribution derivation

```
%% Input data
t = load('tC17.csv')/100;
y = load('yC17.csv');
n  = 15;
ni = 17*ones(1,n);
ni_max = max(ni);
yt = zeros(1, n);
t2 = zeros(1, n);
for i = 1 : n
    yt(i) = t(i,:) * y(i,:)';
    t2(i) = t(i,:) * t(i,:)';
end
%% Stage 1:
lambda   = zeros(n, 1);
delta2   = zeros(n, 1);
epsilon2 = zeros(n, ni_max);
var_delta2_lambda = zeros(n, 1);
for i = 1 : n
    lambda(i) = yt(i) / t2(i);
end
```

```
for i = 1 : n
    for j = 1 : ni(i)
        epsilon2(i,j) = ( y(i,j) - lambda(i)*t(i,j) )^2;
    end
end

delta2_common = sum(sum(epsilon2)) / sum(ni);
disp(delta2_common);

for i = 1 : n
   delta2(i) = sum( epsilon2(i,1:ni(i)) ) / ( ni(i)-1 );
   var_delta2_lambda(i) = delta2(i) / t2(i);
end

disp(lambda);

%% Stage 2:
% Single Component
options = statset('Display','final');
result1 = gmdistribution.fit(lambda,1,'Options',options);
disp(result1.mu);
disp(result1.Sigma);
disp(result1.BIC);
disp(result1.AIC);

% Two components
```

```
result2 = gmdistribution.fit(lambda,2,'Options',options);
disp(result2.mu);
disp(result2.Sigma);
disp(result2.BIC);
disp(result2.AIC);
[idx,nlogl,P] = cluster(result2, lambda);
disp(result2)

% Three components
result3 = gmdistribution.fit(lambda,3,'Options',options);
% disp(result2.mu);
% disp(result2.Sigma);
disp(result3.BIC);
disp(result3.AIC);
disp(result3)
```

A.2 ML method for deriving failure time distribution for single component

```
L      = 1000;
tm     = [2000:100:4000, 4500:500:13000];
tm     = tm/100;
ntm    = size(tm,2);
F      = zeros(ntm,1);
ystar  = 10;
```

```
mu     = 0.204660249554367;   % results from C17_2stage.m
sigma2 = 0.001997607400332;
sigma  = sqrt(sigma2);
delta2 = 0.037445188214323;
delta  = sqrt(delta2);

Fsim = zeros(ntm, L);
for j = 1 : L
   contd = 1;
   while contd
       lam_sim = normrnd(mu, sigma);
       if lam_sim > 0
           contd = 0;
       end
   end
   for k = 1 : ntm
       Ey  = lam_sim * tm(k);
       tmp = (ystar - Ey)/delta;
       Fsim(k,j) = 1-normcdf(tmp,0,1);
   end
end
F(:) = mean(Fsim,2);

R = zeros(ntm, 2);
for k = 1 : ntm
    R(k,:) = [tm(k), F(k)];
```

```
end
plot(R(:,1), R(:,2),'g-');
```

A.3 ML method for deriving failure time distribution for multiple components

```
L      = 1000;
tm     = [2000:50:4000, 4500:100:13000];
tm     = tm/100;
ntm    = size(tm,2);
F1     = zeros(ntm,1);
F2     = zeros(ntm,1);
F      = zeros(ntm,1);
ystar  = 10;

mu     = [0.271650796281094,0.180150631318982];
sigma2 = [5.333759594876598e-004,2.906869882085853e-004];
sigma  = sqrt(sigma2);
delta2 = 0.037445188214323;
delta  = sqrt(delta2);
pi     = [0.267864, 0.732136];

Fsim1 = zeros(ntm, L);
Fsim2 = zeros(ntm, L);

for j = 1 : L
   contd = 1;
```

```
    while contd
        lam_sim = normrnd(mu(1), sigma(1));
        if lam_sim > 0
            contd = 0;
        end
    end
    for k = 1 : ntm
        Ey  = lam_sim * tm(k);
        tmp = (ystar - Ey)/delta;
        Fsim1(k,j) = 1-normcdf(tmp,0,1);
    end

    contd = 1;
    while contd
        lam_sim = normrnd(mu(2), sigma(2));
        if lam_sim > 0
            contd = 0;
        end
    end
    for k = 1 : ntm
        Ey  = lam_sim * tm(k);
        tmp = (ystar - Ey)/delta;
        Fsim2(k,j) = 1-normcdf(tmp,0,1);
    end
end
F1(:) = mean(Fsim1,2);
```

```
F2(:) = mean(Fsim2,2);
F(:)  = pi(1)*F1(:) + pi(2)*F2(:);

R = zeros(ntm, 4);
for k = 1 : ntm
    R(k,:) = [tm(k), F1(k), F2(k), F(k)];
end
plot(R(:,1), R(:,2),'r--',R(:,1), R(:,3),'b-.',R(:,1), R(:,4),'k-');
```

A.4 Bayesian approach for parameter estimation and failure time distribution plot for single component

```
%% Input data
t = load('tC17.csv')/100;
y = load('yC17.csv');

n  = 15;
ni = 17*ones(1,n);

ni_max = max(ni);

yt = zeros(1, n);
t2 = zeros(1, n);
for i = 1 : n
    yt(i) = t(i,:) * y(i,:)';
    t2(i) = t(i,:) * t(i,:)';
end
```

```
%% Setup data matrics and vectors
epsilon2 = zeros(n, ni_max);

nite = 20000;
istart = nite / 2 + 1;    % discard the burn-in draws.

lambda = zeros(n, nite);
delta2 = zeros(1, nite);
mu     = zeros(1, nite);
sigma2 = zeros(1, nite);

L      = 1000;
tm     = [2000:100:4000, 4500:500:13000];
tm     = tm/100;
ntm    = size(tm,2);
F      = zeros(ntm,nite);
ystar  = 10;

%% Prior distributions
% mu ~ N(m, s2)
m     = 0;
s2    = 1000000;
% sigma2 ~ IG(alpha, beta)
alpha = 1;
beta  = 0.0001;
```

```
% delta2 ~ IG(a, b)
a     = 1;
b     = 0.0001;

%% Initialization
lambda(:,1) = 1;
delta2(1)   = 1;
mu(1)       = 1;
sigam2(1)   = 1;

%% Gibbs sampling
lambda_curr = lambda(:,1);
mu_curr     = mu(1);
sigma2_curr = sigma2(1);
delta2_curr = delta2(1);

for ite = 2 : nite
   if mod(ite, 100)==0
       disp(ite);
   end
   % update lambda
   for i = 1 : n
     mean_tmp  = (mu_curr*delta2_curr+sigma2_curr*yt(i))
     / (sigma2_curr*t2(i)+delta2_curr);
     sd_tmp = sqrt( delta2_curr*sigma2_curr
     / (sigma2_curr*t2(i)+delta2_curr) );
```

```
  lambda_curr(i) = normrnd(mean_tmp, sd_tmp);
end
lambda(:,ite)    = lambda_curr;

% update delta2
for i = 1 : n
    for j = 1 : ni(i)
        epsilon2(i,j) = (y(i,j) - lambda_curr(i)*t(i,j))^2;
    end
end

a_tmp           = a + sum(ni)/2;
b_tmp           = b + sum(sum(epsilon2))/2;
inv_delta2_curr = gamrnd(a_tmp, 1.0/b_tmp);
delta2_curr     = 1.0/inv_delta2_curr;
delta2(ite)     = delta2_curr;

% updata mu
mean_tmp = (sigma2_curr*m+s2*sum(lambda_curr))
 / (sigma2_curr + n*s2);
sd_tmp   = sqrt( sigma2_curr*s2 / (sigma2_curr + n*s2) );
mu_curr  = normrnd(mean_tmp, sd_tmp);
mu(ite)  = mu_curr;

% update sigma2
alpha_tmp       = alpha + n/2;
```

```
   beta_tmp        = beta + sum( (lambda_curr-mu_curr).^2 )/2;
   inv_sigma2_curr = gamrnd(alpha_tmp, 1.0/beta_tmp);
   sigma2_curr     = 1.0 / inv_sigma2_curr;
   sigma2(ite)     = sigma2_curr;

%    % update F
   if ite >= istart
       Fsim = zeros(ntm, L);
       for j = 1 : L
          contd = 1;
          while contd
             lam_sim = normrnd(mu_curr, sqrt(sigma2_curr));
             if lam_sim > 0
                contd = 0;
             end
          end
          for k = 1 : ntm
             Ey  = lam_sim * tm(k);
             tmp = (ystar - Ey)/sqrt(delta2_curr);
             Fsim(k,j) = 1-normcdf(tmp,0,1);
          end
       end
       F(:,ite) = mean(Fsim,2);
   end
end
```

```
%% Computing posterior statistics


lambda_posterior = zeros(n, 4);
for i = 1 : n
   lambda_posterior(i,:) = [mean(lambda(i,istart:nite)),
   prctile(lambda(i,istart:nite),50),...
   prctile(lambda(i,istart:nite),2.5),
   prctile(lambda(i,istart:nite),97.5)];
end
disp(lambda_posterior);
mu_posterior = [mean(mu(istart:nite)),
prctile(mu(istart:nite),50),...
prctile(mu(istart:nite),2.5),
prctile(mu(istart:nite),97.5)];
disp(mu_posterior);
sigma2_posterior = [mean(sigma2(istart:nite)),
prctile(sigma2(istart:nite),50),...
 prctile(sigma2(istart:nite),2.5),
 prctile(sigma2(istart:nite),97.5)];
disp(sigma2_posterior);
delta2_posterior = [mean(delta2(istart:nite)),
prctile(delta2(istart:nite),50),...
prctile(delta2(istart:nite),2.5),
prctile(delta2(istart:nite),97.5)];
disp(delta2_posterior);
```

```
R = zeros(ntm, 5);
for k = 1 : ntm
    R(k,:) = [tm(k)*100, mean(F(k,istart:nite)),
    prctile(F(k,istart:nite),50), ...
     prctile(F(k,istart:nite),2.5),prctile(F(k,istart:nite),97.5)];
end
plot(R(:,1)/100, R(:,2),'k-', R(:,1)/100, R(:,3),'b--', ...
     R(:,1)/100, R(:,4),'r-.', R(:,1)/100, R(:,5),'r-.');
```

A.5 Bayesian approach for parameter estimation and failure time distribution plot for multiple components

```
%% Input data
scale = 100;

t = load('tC17.csv')/scale;
y = load('yC17.csv');

n  = 15;
ni = 17*ones(1,n);

ni_max = max(ni);

yt = zeros(1, n);
t2 = zeros(1, n);
for i = 1 : n
```

```
    yt(i) = t(i,:) * y(i,:)';
    t2(i) = t(i,:) * t(i,:)';
end

%% Number of clusters
K = 2;

%% Setup data matrics and vectors
epsilon2 = zeros(n, ni_max);

nite   = 4000;
istart = nite / 2 + 1;    % discard the burn-in draws.

lambda = zeros(n, nite);
delta2 = zeros(1, nite);
mu     = zeros(K, nite);
sigma2 = zeros(K, nite);
pi     = zeros(K, nite);

z      = zeros(n, K);
p      = zeros(1, K);
gamma  = zeros(n, nite);

L      = 1000;
tm     = [2500:100:5000,5500,5750,5800:100:7500,
7750,8000:500:10000];
```

```
tm     = tm/scale;
ntm    = size(tm,2);
F1     = zeros(ntm,nite);
F2     = zeros(ntm,nite);
F      = zeros(ntm,nite);
ystar  = 10;

%% Prior distributions
% mu_k ~ N(m_k, s2_k)
m     = zeros(1, K);
s2    = 1000000*ones(1, K);
% sigma2_k ~ IG(alpha_k, beta_k)
alpha = 1*ones(1, K);
beta  = 0.0001*ones(1, K);
% delta2 ~ IG(a, b)
a     = 1;
b     = 0.0001;
% pi ~ D(xi)
xi    = ones(K,1);

%% Initialization
lambda(:,1) = scale*[0.0027,0.0024,0.0018,0.0017,
0.0018,0.0027,0.0016,0.0016,...
0.0020,0.0030,0.0019,0.0020,0.0021,0.0017,0.0016]';
delta2(1)   = 0.04;
mu(:,1)     = [0.0018;0.0027]*scale;
```

```
%sigma2(:,1) = [2.9e-08; 5.3e-08];
sigma2(:,1) = [1; 1];
gamma(:,1)  = [2,2,1,1,1,2,1,1,1,2,1,1,1,1,1]';
pi(:,1)     = [0.73;0.27];

%% Gibbs sampling
lambda_curr = lambda(:,1);
mu_curr     = mu(:,1);
sigma2_curr = sigma2(:,1);
delta2_curr = delta2(1);
pi_curr     = pi(:,1);
gamma_curr  = gamma(:,1);

for ite = 2 : nite
   if mod(ite, 100)==0
       disp(ite);
   end

   % update the latent variables, z and gamma
   for i = 1 : n
     for k = 1 : K
       p(k) = pi_curr(k)*normpdf( lambda_curr(i),
       mu_curr(k), sqrt(sigma2_curr(k)) );
     end
     p = p / sum(p);
     z(i,:) = mnrnd(1,p);
```

```
  gamma_curr(i) = find(z(i,:),1,'first');
end
%force to have two clusters
gamma_curr(15) = 1;
z(15,:)=[1 0];
gamma_curr(10) = 2;
z(10,:)=[0 1];
gamma(:,ite) = gamma_curr;

% update lambda
for i = 1 : n
  mean_tmp    = (mu_curr(gamma_curr(i))*delta2_curr
  +sigma2_curr(gamma_curr(i))*yt(i)) ...
      / (sigma2_curr(gamma_curr(i))*t2(i)+delta2_curr);
  sd_tmp      = sqrt( delta2_curr*sigma2_curr(gamma_curr(i))...
      / (sigma2_curr(gamma_curr(i))*t2(i)+delta2_curr) );
  lambda_curr(i) = normrnd(mean_tmp, sd_tmp);
end
lambda(:,ite) = lambda_curr;

% update delta2
for i = 1 : n
    for j = 1 : ni(i)
        epsilon2(i,j) = (y(i,j) - lambda_curr(i)*t(i,j))^2;
    end
end
```

```
a_tmp           = a + sum(ni)/2;
b_tmp           = b + sum(sum(epsilon2))/2;
inv_delta2_curr = gamrnd(a_tmp, 1.0/b_tmp);
delta2_curr     = 1.0/inv_delta2_curr;
delta2(ite)     = delta2_curr;

% update pi
c = xi;
for k = 1 : K
    c(k) = c(k) + sum(z(:,k));
end
pi_curr = Dirichletrnd(c);
pi(:,ite) = pi_curr;

% updata mu and sigma
for k = 1 : K
   idx = find(z(:,k)==1);
   nk  = sum(z(:,k));
   if nk>=1
      mean_tmp   = (sigma2_curr(k)*m(k)+
      s2(k)*sum(lambda_curr(idx))) ...
          / (sigma2_curr(k) + nk*s2(k));
      sd_tmp     = sqrt( sigma2_curr(k)*s2(k) /
      (sigma2_curr(k) + nk*s2(k)) );
      mu_curr(k) = normrnd(mean_tmp, sd_tmp);
```

```
      alpha_tmp       = alpha(k) + nk/2;
      beta_tmp        = beta(k) + sum( (lambda_curr(idx)
      -mu_curr(k)).^2 )/2;
      inv_sigma2_curr = gamrnd(alpha_tmp, 1.0/beta_tmp);
      sigma2_curr(k)  = 1.0 / inv_sigma2_curr;
   end

end

mu(:,ite)     = mu_curr;
sigma2(:,ite) = sigma2_curr;

% update F
if ite >= istart
    Fsim1 = zeros(ntm, L);
    for j = 1 : L
       contd = 1;
       while contd
          lam_sim1 = normrnd(mu_curr(1), sqrt(sigma2_curr(1)));
          if lam_sim1 > 0
             contd = 0;
          end
       end
       for k = 1 : ntm
          Ey  = lam_sim1 * tm(k);
```

```
            tmp = (ystar - Ey)/sqrt(delta2_curr);
            Fsim1(k,j) = 1-normcdf(tmp,0,1);
         end
      end
      F1(:,ite) = mean(Fsim1,2);

      Fsim2 = zeros(ntm, L);
      for j = 1 : L
         contd = 1;
         while contd
            lam_sim2 = normrnd(mu_curr(2), sqrt(sigma2_curr(2)));
            if lam_sim2 > 0
               contd = 0;
            end
         end
         for k = 1 : ntm
            Ey  = lam_sim2 * tm(k);
            tmp = (ystar - Ey)/sqrt(delta2_curr);
            Fsim2(k,j) = 1-normcdf(tmp,0,1);
         end
      end
      F2(:,ite) = mean(Fsim2,2);

      F(:, ite) = pi_curr(1)*F1(:,ite) + pi_curr(2)*F2(:,ite);
   end
end
```

```
%% Computing posterior statistics
istart = nite/2;
lambda_posterior = zeros(n, 4);
for i = 1 : n
   lambda_posterior(i,:) = [mean(lambda(i,istart:nite)),
   prctile(lambda(i,istart:nite),50),...
       prctile(lambda(i,istart:nite),2.5),
       prctile(lambda(i,istart:nite),97.5)];
end
disp(lambda_posterior);

mu_posterior = zeros(K, 4);
for k = 1 : K
    mu_posterior(k,:) = [mean(mu(k,istart:nite)),
    prctile(mu(k,istart:nite),50),...
       prctile(mu(k,istart:nite),2.5),
       prctile(mu(k,istart:nite),97.5)];
end
disp(mu_posterior);

sigma2_posterior = zeros(K, 4);
for k = 1 : K
    sigma2_posterior(k,:) = [mean(sigma2(k,istart:nite)),
    prctile(sigma2(k,istart:nite),50),...
       prctile(sigma2(k,istart:nite),2.5),
```

```
       prctile(sigma2(k,istart:nite),97.5)];
end
disp(sigma2_posterior);

pi_posterior = zeros(K, 4);
for k = 1 : K
    pi_posterior(k,:) = [mean(pi(k,istart:10:nite)),
    prctile(pi(k,istart:10:nite),50),...
       prctile(pi(k,istart:10:nite),2.5),
       prctile(pi(k,istart:10:nite),97.5)];
end
disp(pi_posterior);

delta2_posterior = [mean(delta2(istart:nite)),
prctile(delta2(istart:nite),50),...
       prctile(delta2(istart:nite),2.5),
       prctile(delta2(istart:nite),97.5)];
disp(delta2_posterior);



R1 = zeros(ntm, 5);
for k = 1 : ntm
    R1(k,:) = [tm(k)*scale, mean(F1(k,istart:nite)),
    prctile(F1(k,istart:nite),50), ...
        prctile(F1(k,istart:nite),2.5),
        prctile(F1(k,istart:nite),97.5)];
```

```
end
R2 = zeros(ntm, 5);
for k = 1 : ntm
    R2(k,:) = [tm(k)*scale, mean(F2(k,istart:nite)),
    prctile(F2(k,istart:nite),50), ...
        prctile(F2(k,istart:nite),2.5),
        prctile(F2(k,istart:nite),97.5)];
end
R = zeros(ntm, 5);
for k = 1 : ntm
    R(k,:) = [tm(k)*scale, mean(F(k,istart:nite)),
     prctile(F(k,istart:nite),50), ...
        prctile(F(k,istart:nite),2.5),
        prctile(F(k,istart:nite),97.5)];
end
plot(R1(:,1), R1(:,2),'k-', R2(:,1), R2(:,2),'b--', ...
     R(:,1), R(:,2),'r:');
```

A.6 Comparison among probability plot, failure time plot for single component and multiple components

```
%% Input data
t = load('tC17.csv');
y = load('yC17.csv');

n  = 15;
```

```
ni = 17*ones(1,n);

ni_max = max(ni);

yt = zeros(1, n);
t2 = zeros(1, n);
for i = 1 : n
    yt(i) = t(i,:) * y(i,:)';
    t2(i) = t(i,:) * t(i,:)';
end

tf     = zeros(n, 1);
lambda = zeros(n, 1);
F      = zeros(n, 1);
ystar  = 10;

for i  = 1 : n
    lambda(i) = yt(i) / t2(i);
    tf(i)     = ystar / lambda(i);
end

tf = sort(tf);
disp(tf);

for i = 1 : n
    F(i) = (i-0.3)/(n+0.4);
```

```
end
R = [tf,F];

F1 = load('FT_C17_single_Bayes.txt');
F2 = load('FT_C17_multiple_Bayes.txt');

%plot(tf,F,'ok', F1(:,1),F1(:,2),'r-', F2(:,1),F2(:,2),'b--');

save FT_C17.txt R -ASCII

R=load('FT_C17_Approximate.txt');
plot(R(:,1)/100,R(:,2),'k-', F1(:,1)/100,
F1(:,2),'r--', F2(:,1)/100,F2(:,2),'b:');
```

Thesis and Dissertation Services