

Minerva Access is the Institutional Repository of The University of Melbourne

Author/s:

Attanayake, Dona Nayomi Sandarekha

Title:

Risk Analysis and Probabilistic Decision Making for Censored Failure Data

Date:

2019

Persistent Link:

<https://hdl.handle.net/11343/241687>

Terms and Conditions:

Terms and Conditions: Copyright in works deposited in Minerva Access is retained by the copyright owner. The work may not be altered without permission from the copyright owner. Readers may only download, print and save electronic copies of whole works for their own personal non-commercial use. Any use that exceeds these limits requires permission from the copyright owner. Attribution is essential when quoting or paraphrasing from these works.

Risk Analysis and Probabilistic Decision Making for Censored Failure Data

Dona Nayomi Sandarekha Attanayake

The School of Mathematics and Statistics
The University of Melbourne

Submitted in total fulfilment of the requirements of the
degree of Doctor of Philosophy

July 2019

Abstract

Operation and maintenance of a fleet always require a high level of readiness, reduced cost, and improved safety. In order to achieve these goals, it is essential to develop and determine an appropriate maintenance programme for the components in use. A failure analysis involving failure model selection, robust parameter estimation, probabilistic decision making, and assessing the cost-effectiveness of the decisions are the key to the selection of a proper maintenance programme. Two significant challenges faced in failure analysis studies are, minimizing the uncertainty associated with model selection and making strategic decisions based on few observed failures. In this thesis, we try to resolve some of these problems and evaluate the cost-effectiveness of the selections. We focus on choosing the best model from a model space and robust estimation of quantiles leading to the selection of optimal repair and replacement time of units. We first explore the repair and replacement cost of a unit in a system. We design a simulation study to assess the performance of the parameter estimation methods, maximum likelihood estimation (MLE), and median rank regression method (MRR) in estimating quantiles of the Weibull distribution. Then, we compare the models; Weibull, gamma, log-normal, log-logistic, and inverse-Gaussian in failure analysis. With an example, we show that the Weibull and the gamma distributions provide competing fits to the failure data. Next, we demonstrate the use of Bayesian model averaging in accounting for that model uncertainty. We derive an average model for the failure observations with respective posterior model probabilities. Then, we illustrate the cost-effectiveness of the selected model by comparing the distribution of the total replacement and repair cost. In the second part of the thesis, we discuss the prior information. Initially, we assume, the parameters of the Weibull distribution are dependent by a function of the form $\rho = \sigma/\mu$ and re-parameterize the Weibull distribution. Then we propose a new Jeffreys' prior for the parameters μ and ρ . Finally, we designed a simulation study to assess the performance of the new Jeffreys' prior compared to the MLE.

Declaration

This is to certify that:

1. the thesis comprises only my original work towards the PhD;
2. due acknowledgement has been made in the text to all other material used; and
3. the thesis is fewer than 100,000 words in length, exclusive of tables, maps, bibliographies and appendices.

Dona Nayomi Sandarekha Attanayake

Acknowledgements

It is my absolute pleasure to thank many people who make this thesis possible. I would first like to express my sincere gratitude to my supervisor Assoc. Prof. Andrew Robinson and my co-supervisors Dr. Nicholas Armstrong and Prof. Murray Aitkin. Their immense support, wisdom and guidance greatly impacted to make everything possible.

I would also like to thank my friends and colleagues for all the useful discussions and suggestions. Without your constant support and friendliness I would never be able to be apart from my home country for such a long time. Finally, with a heart full of love and joyment I express my love and gratitude to my parents, brother, sister, brother-in-law and my partner, Asanka. Without your love, encouragement and endurance to all my craziness, I would never be able to complete this thesis...

Contents

1	Introduction	1
1.1	Background	2
1.2	What are censored data?	4
1.3	Failure models	6
1.3.1	Log-location-scale distributions	7
1.3.2	Gamma distribution	9
1.3.3	Inverse-Gaussian distribution	9
1.4	Robustness	10
1.5	Survival analysis	11
1.5.1	Empirical Survivor Function	12
1.5.2	Hazard function	13
1.6	Data	14
1.7	Chapter synopsis	14
2	Cost-Effectiveness of the Probabilistic Decisions	17
2.1	Background	17
2.1.1	Periodic preventive maintenance policies	19
2.1.2	Periodic replacement with minimal repair at failure	20
2.2	Optimal replacement time	21
2.3	Optimal replacement time for the Weibull distributed data	23
2.4	Application	24
2.5	Discussion	31
3	Quantile estimation with MLE and MRR	33
3.1	Background	33
3.2	Maximum Likelihood Estimation (MLE)	36
3.3	Median Rank Regression (MRR)	37
3.4	Numerical results	38

3.4.1	Complete data - No censoring	38
3.4.2	75% Fail	45
3.4.3	10% Fail	50
3.5	Discussion	56
4	Bayesian Analysis: Model selection and model averaging?	59
4.1	Background	59
4.2	A single model?	62
4.2.1	Model selection via AIC, BIC and DIC	63
4.2.2	Priors and convergence diagnostics	64
4.2.3	Numerical results	66
4.3	Model averaging	72
4.3.1	BMA and posterior model probabilities	74
4.3.2	Importance sampling	74
4.3.3	Reverse importance sampling	75
4.3.4	Numerical results	76
4.4	Discussion	83
5	Jeffreys' prior for coefficient of variation	85
5.1	Background	85
5.2	Jeffreys' Prior for the Coefficient of Variation	88
5.3	Numerical results	92
5.3.1	Simulation study - Complete data	93
5.3.2	25% Fails	97
5.3.3	Case Study: Application to wind speed data	100
5.4	Discussion	105
6	Conclusions and possible future directions	109
A	Bayesian Model Averaging Appendix	111
A.1	Prior Distributions	111
B	Jeffreys' Prior Appendix	113
B.1	Distribution of the random variable Y	113
B.2	Jeffreys' prior for the parameters μ and ρ^2	114

C R Scripts	117
C.1 R code for cost computation	117
C.2 R code for quantile estimation with MLE and MRR	133
C.3 R code for Bayesian model averaging	149
C.4 R Code for the simulation - Jeffreys' Prior	188
Bibliography	203

List of Figures

1.1	How it all connects.	2
1.2	Steps of a risk analysis.	3
2.1	Weibull probability plot comparing the Weibull and gamma distributions.	24
2.2	Probability distribution function and empirical cumulative distribution function of the expected cost rate per unit hour in \$ for the Weibull distribution. The repair cost of a unit was fixed to \$1. Parameters of the Weibull distribution was estimated with Bayesian analysis for the data.	26
2.3	Distribution of repair and replacement cost rates (in \$) for suboptimal solutions. The repair cost was set to \$1 and the Weibull distribution was fitted to the data. Bayesian estimation was used in estimating the Weibull parameters.	27
2.4	Optimum expected cost rates per unit hour (in \$) for varying cost ratios of replacement to repair and quantiles for the Weibull distribution. The repair cost was set to \$1 when computing the cost rates.	28
2.5	Variation of optimal repair and replacement cost (\$) rates for varying shape parameters of the Weibull distribution	28
2.6	Optimal repair and replacement cost rate given by the Weibull and the gamma distributions in \$. The repair cost was set to \$1.	29
2.7	Optimal repair and replacement cost resulting from the parameter estimation methods; BE, MLE and MRR method. The Weibull distribution was used as the failure model and the repair cost was set to \$1.	30
3.1	Distribution of the Weibull model for different α values.	34
3.2	Absolute bias of the estimated 10^{th} quantile and the 80^{th} quantile using the MLE and the MRR method. Each point is the mean of 1000 simulations.	39
3.3	Mean squared error of the estimated 10^{th} quantile and the 80^{th} quantile using the MLE and the MRR method. Each point is the mean of 1000 simulations.	40
3.4	Distribution of the total expected repair and replacement cost rate estimated at the 10^{th} quantile from MLE and MRR method. The repair cost is set to 1.	41

3.5	Empirical density of the total expected cost rate estimated at the 10^{th} quantile from MLE and MRR method. The repair cost is set to 1.	42
3.6	Distribution of the total expected repair and replacement cost rate estimated at the 80^{th} quantile from MLE and MRR method. The repair cost is set to 1.	43
3.7	Empirical cumulative density of the total expected cost rate estimated at the 80^{th} quantile from MLE and MRR method. The repair cost is set to 1.	44
3.8	Absolute bias of the estimated 10^{th} quantile and the 80^{th} quantile using the MLE and the MRR method. The data are 25% censored. Each point is the mean from 1000 simulations.	45
3.9	Mean squared error of the estimated 10^{th} quantile and the 80^{th} quantile using the MLE and the MRR method. Each point is the mean of 1000 simulations.	46
3.10	Distribution of the total expected repair and replacement cost rate estimated at the 10^{th} quantile for the censored data (25%) from MLE and MRR method. The repair cost is set to 1.	46
3.11	Empirical density of the total expected cost rate estimated at the 10^{th} quantile from MLE and MRR method. The data are censored with 75% fails. The repair cost is set to 1.	47
3.12	Distribution of the total expected repair and replacement cost rate estimated at the 80^{th} quantile from MLE and MRR method with censored data (75% fails). The repair cost is set to 1.	48
3.13	Empirical cumulative density of the total expected cost rate estimated at the 80^{th} quantile from MLE and MRR method for 75% fails. The repair cost is set to 1.	49
3.14	Absolute bias of the estimated 10^{th} quantile and the 80^{th} quantile using the MLE and the MRR method. The data are 90% censored with only 10% fails. Each point is the mean from 1000 simulations.	50
3.15	Mean squared error of the estimated 10^{th} quantile and the 80^{th} quantile using the MLE and the MRR method, 10% fails. Each point is the mean of 1000 simulations.	52
3.16	Distribution of the total expected repair and replacement cost rate estimated at the 10^{th} quantile for the censored data (10% fails) from MLE and MRR method. The repair cost is set to 1.	52
3.17	Empirical density of the total expected cost rate estimated at the 10^{th} quantile from MLE and MRR method. The data are censored with 10% fails. The repair cost is set to 1.	53
3.18	Distribution of the total expected repair and replacement cost rate estimated at the 80^{th} quantile from MLE and MRR method with censored data (10% fails). The repair cost is set to 1.	54
3.19	Empirical cumulative density of the total expected cost rate estimated at the 80^{th} quantile from MLE and MRR method for 10% fails. The repair cost is set to 1.	55

4.1	Bayesian mechanism.	60
4.2	Plot of autocorrelation function of the Weibull distribution after thinning the posterior samples.	66
4.3	Brooks-Gelman-Rubin plot of the Weibull model.	67
4.4	Cumulative distribution of posterior deviances.	69
4.5	Cumulative distribution of posterior deviances from the Jeffreys' prior.	70
4.6	Empirical cumulative distribution of expected repair and replacement cost rate of the single models (Weibull and gamma) and the averaged model. The shown cost rates are for the optimal replacement at 80 th quantile of each distribution and the repair cost was set to \$1 when calculating the cost rates.	78
4.7	Empirical cumulative distribution of expected repair and replacement cost rate of the single models and the averaged model. The shown cost rates are for the optimal replacement at 80 th quantile of each distribution and the repair cost was set to \$1 when calculating the cost rates, 75% fails.	80
4.8	Empirical cumulative distribution of expected repair and replacement cost rate of the single models and the averaged model. The shown cost rates are for the optimal replacement at 80 th quantile of each distribution and the repair cost was set to \$1 when calculating the cost rates, 25% fails.	82
5.1	Relative bias % of the estimate of the relative variation ρ . Each point on the graph is the average of $(\hat{\rho} - \rho)/\rho$ from 1000 simulations. The results are displayed for the small sample sizes 5, 10 and 25. Note that the location of the Weibull distribution is fixed and equal to 1.	95
5.2	MSE of the estimate of the relative variation of the Weibull distribution. Each point on the graph is the mean of the MSE calculated from 1000 simulations. Note that the location parameter is fixed and equal to 1.	96
5.3	Relative bias of the estimates of relative variation of the Weibull distribution. All the points are the average of $(\hat{\rho} - \rho) * 100\% / \rho$ from 1000 simulations. The results are displayed for the large sample sizes 50, 100 and 1000. The location parameter of the distribution is fixed and set to 1.	97
5.4	MSE of estimates of the relative variation of the Weibull distribution. Each point on the graph is the mean of MSE from 1000 simulation. Note that the location parameter is fixed and equal to 1. The results are illustrated for the sample sizes 50, 100 and 1000.	98
5.5	Relative bias % of the estimate of the relative variation ρ . Each point on the graph is the average of $(\hat{\rho} - \rho)/\rho$ from 1000 simulations. The results are displayed for the sample sizes 10, 25 and 50. Note that the location of the Weibull distribution is fixed and equal to 1.	99
5.6	Relative bias % of the estimate of the relative variation ρ . Each point on the graph is the average of $(\hat{\rho} - \rho)/\rho$ from 1000 simulations, 25% fails. The results are displayed for the sample sizes 100, 500 and 1000. Note that the location of the Weibull distribution is fixed and equal to 1.	100

5.7	MSE of estimates of the relative variation of the Weibull distribution. Each point on the graph is the mean of MSE from 1000 simulation, 25% fails. Note that the location parameter is fixed and equal to 1. The results are illustrated for the sample sizes 10, 25 and 50.	101
5.8	MSE of estimates of the relative variation of the Weibull distribution. Each point on the graph is the mean of MSE from 1000 simulation, 25% fails. Note that the location parameter is fixed and equal to 1. The results are illustrated for the sample sizes 100, 500 and 1000. . . .	102
5.9	Empirical distribution of the wind speeds at the two locations.	103
5.10	Posterior distribution of the estimated relative variation at the two locations. Estimates are from thinned MCMC sampling at every 20 th observation.	103

List of Tables

2.1	Parameter estimates with standard errors	25
4.1	Numerical summaries of the model parameters based on the posterior samples for complete data.	67
4.2	AIC, BIC, DIC and estimated evidence values in log deviance scale for the complete data.	68
4.3	Numerical summaries of the model parameters based on the posterior samples for the data censored at 75% of the sample.	70
4.4	AIC, BIC, DIC and estimated evidence values in log deviance scale for the censored data with 75% failures.	71
4.5	Numerical summaries of the model parameters based on the posterior samples censored at 25% of the data sample.	71
4.6	AIC, BIC, DIC and estimated evidence values in log deviance scale for the censored data with 25% failures.	71
4.7	Posterior model probabilities for the complete data.	77
4.8	Posterior model probabilities for the censored data, 75% fails.	78
4.9	Posterior model probabilities for the censored data, 25% fails.	81
5.1	MLE and Bayesian estimates of the relative variation and the location of the Weibull distribution. The estimates are the average of 1000 simulations and the true values of the parameters considered are provided for comparison. Note that the location of the Weibull distribution is fixed (equals to 1) and the results are summarised for the sample sizes 5, 10 and 25.	94
5.2	ML and Bayesian estimates of the location and relative variation of the Weibull distribution when the number of units is 50, 100, and 1000. The estimates are the average of 1000 simulations. Note that the location parameter is set to 1. The true parameter values considered in the simulation are given for the purpose of comparison.	95
5.3	Summary statistics of wind speed data at the locations Col du Grand-St-Bernard and Chasseral, Switzerland. All the measurements are in ms^{-1}	103

5.4	Estimated relative variability at Chasseral and Col du Grand-St-Bernard. ML estimates are presented with 95% confidence intervals and the Bayesian estimates are presented with 95% credible intervals. All the estimates are in ms^{-1}	104
-----	--	-----

1. Introduction

When managing or operating a fleet successfully, we always need to address the core objectives such as high level of readiness of the fleet with maximised efficiency, reduced cost, and operational safety. This can be viewed as a trade-off between unpredictable failures and the certainty of availability of the units to provide the required services. In this context, the primary question that would arise is, when should the components be withdrawn from the fleet for repair or replacement? The components should be withdrawn at an appropriate time in order to minimize the operational disruptions and related life costs of the components and to ensure fleet safety. Thus, the evaluation of the probabilistic risk associated with the components of the fleet and estimation of the related costs are of greater importance.

In this thesis, we will carry out a failure analysis along with a cost computation to aid decision making. The failure data modelling, parameter estimation, and cost computations outlined in this thesis will mainly center on fleet management. However, the methods discussed here can apply for any failure process.

In Chapter 2, we will establish a method to compute the repair and replacement cost of the units in a fleet. The methods outlined in Chapter 2 will utilize in the future chapters to evaluate the cost of either choice of parameter estimation or model selection. We will also look at a sensitivity analysis to find out any impact on the optimal strategy when the other constraints change.

Chapter 3 will compare the parameter estimation methods. In particular, we will consider the maximum likelihood estimation (MLE) and median rank regression method (MRR) in quantile estimation. A simulation study will compare the parameter estimation method followed by the repair and replacement cost computation. In Chapter 4, we will investigate the problem of model uncertainty in failure data

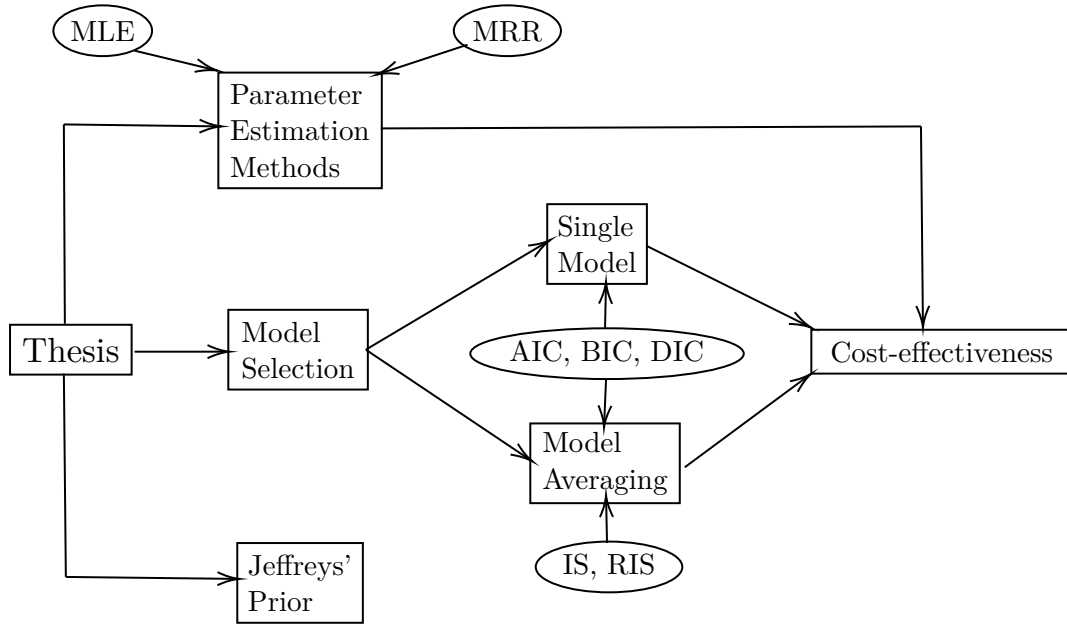


FIGURE 1.1: How it all connects.

analysis. Analysts can either select a single model for their predictions or use an average model to account for any model misspecification. We will examine how the repair and replacement cost varies according to the choice of a model or an average.

Chapter 5 slightly deviates from the failure data modelling. In this chapter, we will derive a new Jeffreys' prior for a function of parameters of the Weibull distribution. The log-location-scale form of the Weibull distribution will use for the derivations. The resulting prior will compare with the MLE using a simulation study. We will then illustrate an application to a real-life data set. This chapter is not connected to the other chapters but uses the same parameter estimation techniques such as MLE and Bayesian estimation and heavily censored data. Figure 1.1 shows how all the chapters in the thesis connects.

The remainder of the introductory chapter will focus on the background to the current research, with a further explanation of the aims and objectives of the thesis. In the conclusions, we will discuss some possible future directions to extend the research.

Section 1.1: Background

Analyzing time-to-failure data is prevalent in many disciplines. Many statisticians and risk analysts are familiar with the concepts of failure analyses. Furthermore,

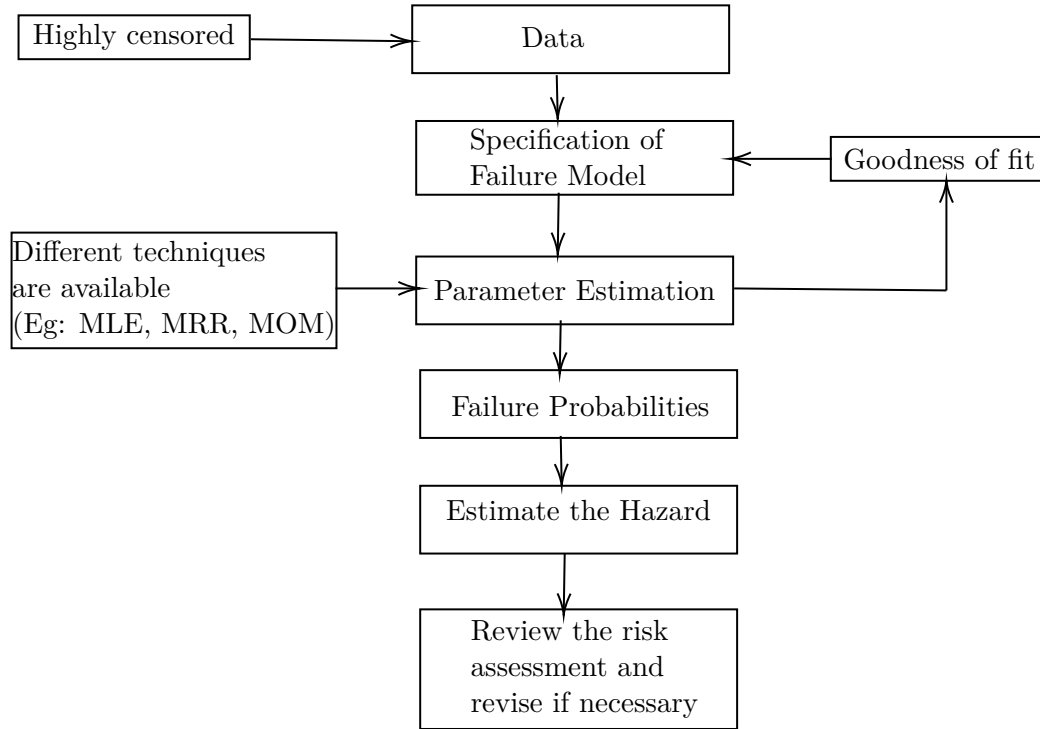


FIGURE 1.2: Steps of a risk analysis.

the methods used in risk analyses are in continuous discussion and are developing. Statistical risk analysis consists of failure data modelling, parameter estimation, and hazard estimation. Figure 1.2 illustrates the steps in a risk analysis.

Identification of the characteristics of the failure distribution of the components is vital in determining a proper maintenance program for the components. Predominantly, the precise specification of the underlying distribution of the life of the fleet and robust estimation of the parameters of the specified distribution is crucial. To arrive at these objectives, the remaining useful life (RUL) of the components could use for analyses and estimations. However, it should note that the evaluation of the statistical properties such as unbiasedness and consistency of the parameters might be distracting if the study purely focuses on risk management. Olteanu and Freeman (2010) stated two aspects in parameter estimation, one considers the quality of the estimated parameters and the other concentrates on the risk management as in this thesis.

In most of the failure analysis problems, only a fewer number of failures are observed with many un-failed or censored data. Thus, the parameters of the failure distribution should estimate with many censored cases.

Section 1.2: What are censored data?

In survival or reliability analysis we are focused on a non-negative random variable; T , which provides the time to an event of interest occurs. The random variable T is known as survival time, lifetime or failure time. Broadly, when the data are partially observed on this random variable, they are called censored data. That is if the survival or failure time of a subject is not observed due to another event, then the resulting data are censored. Meeker and Escobar (1998) state that censoring restricts the capability of exactly determining the failure times. If the subjects have not failed by the end of the considered time or at the inspection time, then the data are called "right censored". If a subject failed before the study or at the time of inspection, those components are "left censored" (Smith, 2002).

When we consider a sample of complete data, the random variables would be $T_1, T_2, T_3, \dots, T_n$ and we assume that the random variables are independent and identically distributed, all following the same failure distribution.

- In most failure tests, the study is terminated at a pre-specified time-point; t_c . Then the subjects that are not failed at the time of termination are called "Type I censored" or "time censored" and t_c is known as censoring time (Meeker and Escobar, 1998; Gijbels, 2010). Here, the censoring time is non-random (or fixed). Instead of the failure times $T_1, T_2, T_3, \dots, T_n$, now we observe $Y_1, Y_2, Y_3, \dots, Y_n$ where,

$$Y_i = \begin{cases} T_i & \text{if } T_i \leq t_c \\ t_c & \text{if } T_i > t_c \end{cases}$$

- Second type of censoring is known as "Type II censoring" or "failure censoring". This occurs when the study is terminated after observing a predetermined number of failures. Failure censored tests are not frequent in practice even though the statistical analysis is simpler compared to analysis of time censored data (Meeker and Escobar, 1998).

Let $T_{(1)}, T_{(2)}, T_{(3)}, \dots, T_{(n)}$ be the ordered failure times and suppose the study terminates after observing r ($< n$) number of failures. Therefore we can only observe the failure times, $T_{(1)}, T_{(2)}, T_{(3)}, \dots, T_{(r)}$. Thus, the failure times of the remaining $n - r$ subjects that are not failed at the time of termination are considered as $T_{(r)}$ as their failure times are only known to be larger than the value of $T_{(r)}$.

- In some studies the exact failure time is unknown. But only the upper and lower bounds of the failure times are available. Thus, we have interval censored failure times.
- Another mostly practiced type of censoring is random censoring. In random censoring a subject is assumed have a lifetime of T_i and a censoring time of C_i . An important assumption regarding the random censorship is that, we assume T_i and C_i to be independent of each other. Random censorship can be either right or left censored. Suppose the a subject is right randomly censored. Then, the practitioner observes the minimum of T_i and C_i which are identified as the survival time or the censored time respectively. Thus, the observations will consist, $(Y_1, \delta_1), (Y_2, \delta_2), \dots, (Y_n, \delta_n)$ where, $Y_i = \min(T_i, C_i)$.

Here, δ_i is an indicator variable having the values;

$$\delta_i = \begin{cases} 1 & \text{if the observation is not censored} \\ 0 & \text{if the observation is censored} \end{cases}$$

If the failure of a subject is known to be occurred before a certain time, then we call the observation is left random censored. Here, the practitioner observes the maximum of the failure time T_i and the censoring time C_i . Then, the observations contains, $(Y_1, \delta_1), (Y_2, \delta_2), \dots, (Y_n, \delta_n)$, where $Y_i = \max(T_i, C_i)$.

Specification of the failure distribution and the underlying assumptions of the failure model can introduce model uncertainty to the problem. This problem should be addressed with care as the decisions are dependent on the specified model. The choice of the failure distribution depends on the goodness of fit of the fitted model, to the data. Moreover, practitioners have different opinions with regards to the parameter

estimation method to be used, such as median rank regression (MRR) and maximum likelihood (ML) method.

Section 1.3: Failure models

Reliability of a system can be estimated with either non-parametric methods or parametric methods. Even though the non-parametric methods are easy to use, it limits the study from extrapolating results beyond the last observed data point (Kalaiselvan and Rao, 2016). In non-parametric methods, failure data are analysed without assuming any underlying distribution. There are some non-parametric methods that can be used to analyse the failure data such as Kaplan-Meier estimate, simple actuarial and standard actuarial methods (Kalaiselvan and Rao, 2016).

Kaplan-Meier method (Kaplan and Meier, 1958) is one of the mostly used non-parametric methods in reliability analysis. Ying and Wei (1994) used the Kaplan-Meier for dependent failure time observations. They showed the consistency and asymptotic normality of the Kaplan-Meier estimates still hold even when the observations are weakly dependent. Chuang et al. (2001) discussed the use of the Kaplan-Meier estimation for correlated survival observations in a dental implant survival analysis. Schaubel and Cai (2004) proposed a non-parametric conditional survival function estimation method through their corresponding cumulative hazard functions. Pepe and Mori (1993) suggested several descriptive methods for summarizing failure data such as marginal probability and conditional probability estimators arguing that the Kaplan-Meier estimate is not always appropriate for failure data analysis. A discussion of Kaplan-Meier estimation versus a parametric method based on the Gompertz distribution was presented in Cantor and Shuster (1992).

There are other methods and models used in the failure data analyses such as parametric and non-parametric Bayesian approaches (see for example Kalbfleisch, 1978; Yang et al., 2012), and regression models when the subjects experience multiple failures (see Shiao and Lin, 1999; Aljahdali, Sheta, and Rine, 2001; Wei, Lin, and Weissfeld, 1989; Prentice, Williams, and Peterson, 1981). However, in this project we mainly use parametric methods in analysing failure data. In a parametric approach we assume

an underlying distribution for the failure data. The fitted failure model to the data will give information about the expected number of failures (or the proportion of failures) of the fleet according to the remaining life of the components. The accurate specification of the failure model is essential for predicting future incidents.

Due to the simplicity and flexibility, the Weibull distribution has frequently used as a model in failure and time-to-event data analysis. Additionally, the Weibull distribution adequately fits for different types of failure data. Nevertheless, other competing failure models such as the log-normal distribution, the gamma distribution, and the log-logistic distribution can provide competing fits. For example, the failure model defined by the UK Defense Standard (DEFSTAN) 00-970; the airworthiness design guidelines followed by the Royal Australian Air Force (RAAF) is the log-normal distribution. Therefore, an elucidative comparison of different candidate distributions in failure data modelling may lead to more robust parameter estimates.

In this thesis, we consider five potential candidate distribution to capture the variations in the failure data.

1.3.1 Log-location-scale distributions

If a random variable x has a log-location-scale distribution with a location parameter $-\infty < \mu < \infty$ and scale parameter $\sigma > 0$, then the probability distribution function is of the form,

$$f(x; \mu, \sigma) = \frac{1}{\sigma x} \phi \left(\frac{\log(x) - \mu}{\sigma} \right); \quad x > 0 \quad (1.1)$$

The cumulative distribution function is given by,

$$F(x; \mu, \sigma) = \Phi \left(\frac{\log(x) - \mu}{\sigma} \right) \quad (1.2)$$

The quantile function of the log-location-scale distributions takes of the form,

$$\log(x_p) = \mu + \sigma \Phi^{-1}(p) \quad (1.3)$$

In Chapter 4 the log-location-scale form of the Weibull distribution will use along with the log-normal distribution and log-logistic distribution from the family.

The cumulative distribution, $F(x)$ (cdf) and the probability distribution, $f(x)$ (pdf) functions of the Weibull distribution can be given by,

$$F(x; \mu, \sigma) = \Phi_{sev} \left(\frac{\log x - \mu}{\sigma} \right) \quad (1.4)$$

$$f(x; \mu, \sigma) = \frac{1}{\sigma x} \phi_{sev} \left(\frac{\log x - \mu}{\sigma} \right), \quad x > 0 \quad (1.5)$$

Here, $\Phi_{sev}(z) = 1 - \exp(-\exp(z))$ and z follows the standard smallest extreme value (SEV) distribution with $\mu = 0$ and $\sigma = 1$ and $\phi_{sev} = \exp[z - \exp(z)]$.

Next, the cdf and pdf of the log-normal distribution take the following forms.

$$F(x; \mu, \sigma) = \Phi_{norm} \left(\frac{\log x - \mu}{\sigma} \right) \quad (1.6)$$

$$f(x; \mu, \sigma) = \frac{1}{\sigma x} \phi_{norm} \left(\frac{\log x - \mu}{\sigma} \right), \quad x > 0 \quad (1.7)$$

Here, $\Phi_{norm}(z)$ and ϕ_{norm} are the cdf and pdf of the standard normal distribution.

Similarly, cdf and pdf of the log-logistic distributions can expressed as,

$$F(x; \mu, \sigma) = \Phi_{logis} \left(\frac{\log x - \mu}{\sigma} \right) \quad (1.8)$$

$$f(x; \mu, \sigma) = \frac{1}{\sigma x} \phi_{logis} \left(\frac{\log x - \mu}{\sigma} \right), \quad x > 0 \quad (1.9)$$

Here, Φ_{logis} and ϕ_{logis} are the cdf and pdf of the logistic distribution.

1.3.2 Gamma distribution

Suppose a random variable (r.v.) x follows a gamma distribution with rate $\alpha > 0$ and shape $\beta > 0$. The probability distribution function of the r.v. is given by,

$$f(x; \alpha, \beta) = \frac{\alpha^\beta x^{\beta-1} e^{-\alpha x}}{\Gamma(\beta)} \quad (1.10)$$

The Weibull distribution is a special case of the generalized gamma distribution. The PDF of the generalized gamma distribution is:

$$f(x; \phi, \beta, k) = \frac{\left(\beta/k^\phi\right) x^{\phi-1} e^{-(x/k)^\beta}}{\Gamma(\phi/\beta)} \quad (1.11)$$

When $\phi = 1$, the distribution becomes the Weibull. This can be shown with the power transformation of the generalized gamma distribution (see Kiche, Ngesa, and Orwa, 2019).

1.3.3 Inverse-Gaussian distribution

If a r.v. x follows an inverse-Gaussian distribution with mean $\mu > 0$ and shape $\lambda > 0$, the probability density function of x is given by,

$$f(x; \mu, \lambda) = \left[\frac{\lambda}{2\pi x^3} \right]^{1/2} \exp \left\{ \frac{-\lambda(x - \mu)^2}{2\mu^2 x} \right\} \quad (1.12)$$

The quantile functions of the gamma and inverse-Gaussian distributions do not have simple closed-form representations which limit the use of MRR method to estimate the model parameters in Chapter 3.

In most of the failure data analyses, only limited data available or the available data are heavily censored. There can be analyses based on no failure data. However, the robust parameter estimation is equally crucial for all the situations. Next section briefly reviews the term robustness.

Section 1.4: Robustness

Commonly, statistical conclusions are derived based on the assumptions made about the statistical model specified by the experimenters. However, these assumptions might not be exactly correct and can be beliefs or judgments of the expertise in the field. Thus, it is expected that any violation in the assumptions regarding the underlying situation should cause only minor impact to the final decisions made or in other words the model is *robust* (Box, 1953). Simply, robustness is defined here as the insensitivity to small deviations from the assumptions made in statistical analysis about the underlying situation (Huber, 2011). For example, Box (1953) assessed the sensitivity to the non-normality of the test statistics in comparing variances. According to his study, the robustness of a statistic is more important than other properties such as power or efficiency. Huber (2011) noted that the robustness of a statistic is obtained by sacrificing some degree of efficiency and power of a test. Thus, the development of parameter estimation methods that are insensitive to slight deviations from the true conditions is crucial.

Most of the studies on robustness are centered around the normality of the assumed distributions. Wilcox (2012) distinguished three types of robustness for a statistic namely, qualitative robustness, quantitative robustness and infinitesimal robustness (see also Huber, 2011; Hampel, 1971; Zähle, 2016). Many projects discuss early developments of robustness of statistics such as mean, variance and confidence intervals. Huber (1992) discussed about an approach to robust estimation of the location parameter of a contaminated normal distribution. Jaeckel (1971) considered both symmetric and asymmetric distributions in robust estimation of location parameter. Bickel (1965) compared several proposed robust estimates of location parameter. A review of robust statistics was given in Huber (1972). Another study of Huber (1968) focused on the robust estimation of a confidence interval when estimating an unknown quantity from a sample.

Apart from the normal distribution, there are discussions focused on the other distributions as well. We briefly reviewed the literature for the Weibull distribution as it is one of the most commonly used failure model and in addition to that, we will

be extensively using the statistics of Weibull distribution for failure rate estimation and cost calculations in the future chapters. Gündüz, Başar, and Aydin (2015) studied the use of Kernel smoothing in some robust estimations of the Weibull distribution where the Kernel function is used to estimate median, interquartile range, quartile, and octile skewness. Goegebeur, Guillou, and Rietsch (2015) developed a robust estimation for tail parameters in the context of extreme value. A robust method for the parameter estimation of the Weibull distribution using the L_2 distance (integrated square distance) has proposed by Yang and Scott (2013). The results show a preference towards the L_2 estimate over the ML estimate for both right-censored and complete data. In their study, the Weibull distribution has augmented with a weight parameter in order to account for the contaminated data. Zhang, Xie, and Tang (2006) suggested M-estimator as a robust regression method compared to ordinary least squares (OLS) method in estimating the shape parameter for both multiple censored (without outliers) and complete data (with one or two outliers) (see also Basak and Balakrishnan, 2003; Seki and Yokoyama, 1993).

Next section touches concepts and few models used in survival analysis.

Section 1.5: Survival analysis

The concept of survival analysis is widely used in many areas in order to inspect the lifetime of a subject (such as machines, animals or human). In survival analysis lifetime or survival time is the period from the start of observation until the event of interest (death / failure) occurs (i.e. in survival analysis we are interested in measuring the time taken to an event). Moreover, in survival analysis the time of death is termed as failure time.

A random variable Y is said to be a survival random variable if its observed outcome lies in the interval $[0, \infty)$. Suppose the random variable X has a probability density function f . Then, its cumulative distribution function can be defined as:

$$F(x) = Pr(X < x) = \int_0^x f(u)du \quad (1.13)$$

Then, we can define the survival function as (for all the values of x):

$$\begin{aligned}
 S(x) &= Pr(X > x) \\
 &= \int_x^\infty f(x)dx \\
 &= 1 - F(x) \\
 F(x) &= 1 - S(x)
 \end{aligned} \tag{1.14}$$

If we differentiate both sides of the survival function (equation 1.14) with respect to x ,

$$f(x) = -\frac{d}{dx}S(x) \tag{1.15}$$

$$S(y) = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{if } x = \infty \end{cases}$$

This suggests that the the survival function $S(x)$ is a monotonically decreasing function over the interval $[0, \infty)$.

1.5.1 Empirical Survivor Function

Empirical survivor function is the simplest way to estimate the survivor function. Let's define an indicator function $I_A(x)$, where A is any event of interest.

$$I_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{otherwise} \end{cases}$$

(1.16)

Suppose $X_1, X_2, X_3 \dots X_n$ are independent and identically distributed (i.i.d) random variables with the same distribution as X . Then, the empirical survivor function $S_n(x)$ is defined for all values of x as,

$$\begin{aligned} S_n(x) &= \frac{\text{number of observations} > x}{n} \\ &= \frac{1}{n} \sum_{i=1}^n I_{x,\infty}(X_i) \end{aligned} \quad (1.17)$$

Above equation indicates that for a fixed value of x , empirical survivor function is the average of n i.i.d. random variables, say $Z_I = I_{(x,\infty)}(X_i)$.

Therefore, $nS_n(x) = \sum_{i=1}^n Z_i$

Each Z_i has the same Bernoulli distribution with $P(Z_i = 1) = S(x)$ and $P(Z_i = 0) = 1 - S(x)$. Consequently, $nS_n(x)$ follows a Binomial distribution: $nS_n(x) \sim \text{Bin}(n, S(x))$.

$$\begin{aligned} \mathbb{E}[nS_n(x)] &= nS(x) \\ \mathbb{E}[S_n(x)] &= S(x) \end{aligned} \quad (1.18)$$

Therefore, $S_n(x)$ is an unbiased estimator for $S(x)$.

$$\begin{aligned} \text{Var}[nS_n(x)] &= nS(x)[1 - S(x)] \\ \text{Var}[S_n(x)] &= \frac{1}{n} S(x)[1 - S(x)] \end{aligned} \quad (1.19)$$

In order to examine the accuracy of the estimate, we can use an approximate confidence interval with two standard errors $(S_n(x) \pm 2\sqrt{\frac{S_n(x)(1-S_n(x))}{n}})$.

1.5.2 Hazard function

Hazard function simply is the instantaneous probability of death. It is defined as the probability of observing the failure of a subject during the time interval $(x, x + \delta x)$

given that the subject has survived until time x or greater. Thus, the hazard function can be used to model any time dependent event of interest.

$$h(y) = \lim_{\delta x \rightarrow 0} \frac{Pr(x \leq X < x + \delta)/S(x)}{\delta x} \quad (1.20)$$

Section 1.6: Data

Throughout the thesis, we use a set of telephone failure data for illustration purposes. The data consists of 88 observation of two-way radio-telephone lifetimes. The failure times are in units of 8 hours (a working day).

Section 1.7: Chapter synopsis

Chapter 2 briefly outlines the existing preventive maintenance policies. Preventive maintenance occurs while a system is operating, to avoid any costs due to unexpected failures. We can use a preventive maintenance method to design an optimal repair and replacement strategy for a fleet. We selected the periodic repair and minimal repair at failure policy to evaluate the optimal expected cost rates of repair and replacement for a given strategy.

Chapter 3 compares two popular parameter estimation methods in quantile estimation. We compare the maximum likelihood estimation method and the median rank regression method in estimating the quantiles of the Weibull distribution. We designed a simulation study with different factors and degrees of censoring. The measures of precisions are the mean squared error and relative bias.

Chapter 4 address the problem of model uncertainty. We fit five failure distributions to the telephone failure data. Using Akaike information criteria (AIC), Bayesian information criteria (BIC), and deviance information criteria (DIC), we select the best model to interpret the data. Then we will demonstrate the use of the Bayesian model averaging (BMA) techniques to minimize the model uncertainty. Using the methods

described in Chapter 2 we compute the repair and replacement costs according to model selections.

Chapter 5 develops a new Jeffreys' prior for a function of parameters of the Weibull distribution. We use the log-location-scale form of the distribution and focus on the coefficient of variation parameter (σ/μ) . We will assess the performance of the new Jeffreys' prior compared to the MLE.

A discussion for each of these chapters is given at the end of the chapter, and an overall conclusion and possible future directions are given in Chapter 6.

2. Cost-Effectiveness of the Probabilistic Decisions

In Chapter 1, we have already discussed the importance of aiding decision making. It is essential to quantify the risk associated with plausible decisions. Another aspect of risk analysis is to find the cost associated with each decision. Motivated by that factor, we try to establish a method to compute the repair and replacement cost of components in a fleet. The methods discussed here will be used in the succeeding chapters. Therefore, this is a supportive chapter connecting others.

We first review the existing maintenance policies. In particular, we focus on periodic preventive maintenance policies. Then, a suitable policy is selected to calculate the periodic replacement cost and repair cost at any failures. The periodic replacement time will set to a specific quantile in the failure distribution, and the maintenance costs will calculate accordingly. A sensitivity analysis will be followed by the cost computation to examine the impact of changing factors on the optimal strategy.

Section 2.1: Background

Continuous inspection and maintenance of a fleet of components throughout its life are essential to avoid possible system failures. Additionally, a proper maintenance programme can assure the quality of the service provided. An unexpected failure of a fleet can be adversely affected in monetary terms. Therefore, components of a fleet need to replace when their operating and maintenance costs are no longer acceptable. However, the repair cost of a component can often be high compared to the replacement cost of the component. Also, the rate at which the repair and

replacement are being carried out needed to be slow down to optimize the operation costs.

Conversely, prolonged use of old components might be riskier. This might also affect the availability and the readiness of the components and increase the operation and maintenance costs. Therefore, it is required to optimise the costs of replacement and overhaul. This might lengthen the lifespan and enhance the reliability of the fleet. Alternatively, the development of an optimal maintenance plan to determine the optimal inspection time or optimal replacement time of the component is of interest.

Maintenance of a system can be either corrective or preventive. Corrective maintenance (CM) occurs when a system fails. A CM scheme contains all the actions required to restore the system to a specified condition after experiencing a failure. Preventive maintenance (PM) occurs while a system is operating (Wang, 2002a). All the actions that are required to keep the system in a specific condition such as a pre-scheduled inspection to minimize possible failures include in a PM programme. Under PM policy units replace before the actual failure to minimize the expected cost associated with a failure. However, the utilization of any remaining useful life is lost here. Different approaches are available in developing an appropriate maintenance policy for a single unit in operation or a fleet of components according to either CM or PM; see Nakagawa, 1981; Wang, 2002a; Wang and Pham, 1999, for example.

In this chapter, we outline a cost calculation method to evaluate the optimal time of replacement or overhauling while minimizing the related costs. Additionally, the sensitivity of the results is examined changing, choice of failure model, cost ratios, true shape parameter values and, parameter estimation methods. Sensitivity analysis helps assess the impact on the optimal action when the other constraints change. A cost computation enables the decision-maker to be informed about the relevant cost factors, the relationships between the cost factors and the prospective decisions. Consequently, the best decision can implement while maximizing the life of the fleet with lower risk and minimum cost. The optimal decision is the one which minimizes the total expected cost over an infinite life span. The approach, periodic replacement with minimal repair at failure is used to find the optimum periodic replacement time. However, it is not intended to compare and contrast the preventive maintenance

policies here.

2.1.1 Periodic preventive maintenance policies

This section summarises the periodic preventive maintenance policies along with an application of periodic replacement with minimal repair at failure.

In a periodic replacement policy, units are preventively maintained at the fixed time intervals kT ($k = 1, 2, 3, \dots$) and the failures in between the periodic replacements are repaired (Wang, 2002a). Here, T is a constant, and the maintenance is independent of the failure history. Two basic periodic PM policies are block replacement policy and periodic replacement with minimal repair at the failure.

As the name implies, a block or a group of components are replaced in the block replacement policy. The components are replaced at a pre-specified sequence of equally spaced time points (kT , where $k = 1, 2, 3, \dots$) or at failure, independent of the failure history (Wang, 2002a; Barlow and Proschan, 1996; Cho and Parlar, 1991). Under the block replacement policy, the units are replaced at the specified times irrespective of the servicing age which may cause replacement of practically new units (Cho and Parlar, 1991). Berg and Epstein (1976) thus have modified the block replacement policy to account for this drawback by defining an age limit for the group replacement. Under the modified block replacement model defined by Berg and Epstein (1976), the failures are replaced with new units instantaneously. The units that are of age b ($0 < b < T$) or less are not replaced, but remain in the system. Tango (1978) proposed an extended method where some failed units are replaced with used ones based on the ages of the units at the time of failure. Similar to the standard block replacement policy, in this extended policy the components are exchanged with new ones at time points kT ($k = 1, 2, 3, \dots$). Another modification to the standard block replacement policy is presented in Nakagawa (1981). In his method, if a unit fails just before the planned replacement, that unit is considered to remain failed without repair or replacement, until the next scheduled replacement kT .

Under Tilquin and Cleroux (1975) modified policy of block replacement, additional to the common cost of replacement at failure (c_1) and preventive replacement cost (c_2), some discrete and continuous costs (such as inspection or monitoring costs,

costs with depreciation and etc.) that increase with the age of the components are considered. They provide with examples of both discrete and continuous adjustment costs. Beichelt (1981) presents a generalized block replacement policy considering two modes of system failures. In mode 1, the failures are removed by minimal repair, and in mode 2, failures are removed by replacement. Maintenance under generalized block replacement is executed depending on the mode of failure, and at scheduled times PM is performed. Hanscom and Cl  roux (1975) have discussed the results of block replacement policy for the truncated normal distribution, the Weibull distribution, and the gamma distribution. An application of the block replacement policy to the copper cable deterioration in a petroleum company in Sultan, Oman is presented in Qatan et al. (2016). The results are then compared with a preventive maintenance policy in order to find the most effective maintenance method.

2.1.2 Periodic replacement with minimal repair at failure

Under the periodic replacement with minimal repair at failure, the units are replaced at pre-determined time steps kT where $k = 0, 1, 2, \dots$ and the failures treated with minimal repair (Wang, 2002b; Barlow and Hunter, 1960). In this model, it is assumed that the failure rate of the system remains undisturbed by repairs between the scheduled replacement times. Over the decades, various extensions to this policy have introduced. A summary of periodic replacement policies with minimal repair at failure is presented in Nakagawa (1981). The definition of minimal repair in terms of the failure rate is provided in Nakagawa and Kowada (1983). They also examine the probabilistic quantities such as failure distributions and the time between the failures with an application to the minimal repair and replacement theory when the unit is replaced at time T or at n^{th} failure.

Nakagawa (1981) discusses three extensions to the periodic replacement with minimal repair if the periodic replacement time is specified and if the failure occurs just before the specified replacement time. According to the developed models, the unit will remain failed until the next scheduled replacement (model I), the failed unit can replace with a spare unit as often as necessary until the next planned replacement (model II) or the failed unit can be replaced by a new unit (model III). Tilquin and

Cleroux (1975) discussed the use of a general cost function ($a(x)$, x is the life of the unit) which increases with the age of the units in addition to the standard cost factors (the minimal repair cost c_1 and the scheduled replacement cost c_2).

An application of the periodic replacement with minimal repair at failure to cumulative damage models was presented in Nakagawa and Kijima (1989). They have evaluated the optimal time (T), the shock (N) and the damage (Z) which minimize the expected costs $C(T)$, $C(N)$ and $C(Z)$ respectively with a numerical example for an exponential case. Liu, Makis, and Jardine (1995) considers the effect of an overhaul in periodic replacement with minimal repair, assuming the overhauling enables to rejuvenate the system. Boland and Proschan (1982) look at the cases; (a). Optimal T , which minimizes the expected costs over a finite life span $(0, t]$, and b). Optimal period T , which minimises the expected total cost per unit time over an infinite life span. They consider a model for the total cost, which an increasing function of the number of previous repairs of the system, in evaluating the optimal time T in the two scenarios. Sheu (1991) views the minimal repair cost of a unit which fails at age y as a function of the age-dependent random element $C(y)$. The deterministic element $c(y)$ depends on the age and the number of minimal repairs to the unit. Thus, the j^{th} minimal repair cost is given by a function $g(C(y), c_j(y))$ where, g is a positive, non-decreasing and continuous function. The evaluation of the optimal interval T for the minimal repair and replacement policy is then discussed for a single component and a multi-unit system. The replacement policy has applied in a variety of fields, including oil and chemical industry (Qatan et al., 2016), transportation, defence, discrete part manufacturing, etc. The applications are summarised in Dekker (1996).

Section 2.2: Optimal replacement time

In order to gain insight into the system, it may be advantageous to make simplifying assumptions even if they are not fully reflective of reality. Therefore, we consider a system with identical units that are operated independently to establish our methods. It is assumed that the failure mode is the same for all the units in the considered system and follows the same failure distribution. The units are replaced at pre-determined

time steps kT where $k = 0, 1, 2, \dots$ and the failures are restored only with minimal repair. Also, the failures between the time period $(0, t]$ are assumed to be identified immediately, and the repair and replacement times are expected to be negligible. Therefore, the failure rate of the system remains undisturbed by the repairs between the failures.

Suppose the failure distribution of the system is $F(t)$ with a continuous density $f(t)$. The hazard rate ($r(t)$) of the distribution is given by,

$$r(t) = \frac{f(t)}{1 - F(t)} \quad (2.1)$$

The minimal repair cost of a unit is denoted by c_1 , and the scheduled replacement cost of a unit is c_2 . Suppose the length of the replacement is T . Then, for an infinite lifespan, the object is to minimize the expected total cost $C(T)$ per unit time to find the optimal replacement time T^* .

The expected cost rate is given by (Barlow and Hunter, 1960),

$$C(T) = \frac{c_1 \int_0^T r(t) dt + c_2}{T} \quad (2.2)$$

The optimal replacement time T^* is found by differentiating the expected cost rate $C(T)$ with respect to T and by equating it to zero.

$$T^* r(T^*) - \int_0^{T^*} r(t) dt = \frac{c_2}{c_1} \quad (2.3)$$

In the case where a solution to equation (2.3) exists and where the hazard rate $r(t)$ is monotonically increasing, then the solution is shown to be unique and the expected total cost can be expressed as,

$$C(T^*) = c_1 r(T^*) \quad (2.4)$$

Section 2.3: Optimal replacement time for the Weibull distributed data

We then derive the optimal replacement time for a Weibull distribution with scale parameter σ and location parameter μ .

Suppose the failure times (t) of a system follows a Weibull distribution with scale parameter σ and location parameter μ . The distribution function, the density function and the hazard function of the Weibull distribution are given by (2.5), (2.6) and (2.7) respectively.

$$F(t, \mu, \sigma) = \frac{1}{\sigma t} \phi_{sev} \left(\frac{\log(t) - \mu}{\sigma} \right) \quad (2.5)$$

$$f(t, \mu, \sigma) = \Phi_{sev} \left(\frac{\log(t) - \mu}{\sigma} \right) \quad (2.6)$$

$$r(t, \mu, \sigma) = \frac{1}{\sigma \exp(\mu)} \left[\frac{t}{\exp(\mu)} \right]^{\frac{1}{\sigma}-1} \quad (2.7)$$

For a system with a single unit in operation, the optimal cost function can be written as,

$$\begin{aligned} \frac{T^*}{\sigma \exp(\mu)} \left[\frac{T^*}{\exp(\mu)} \right]^{\frac{1}{\sigma}-1} - \int_0^{T^*} \frac{1}{\sigma \exp(\mu)} \left[\frac{t}{\exp(\mu)} \right]^{\frac{1}{\sigma}-1} dt &= \frac{c_2}{c_1} \\ \left[\frac{T^*}{\exp(\mu)} \right]^{\frac{1}{\sigma}} \left(\frac{1}{\sigma} - 1 \right) &= \frac{c_2}{c_1} \end{aligned} \quad (2.8)$$

Hence, the optimal replacement time can be found by,

$$T^* = \exp(\mu) \left[\frac{c_2}{c_1 \left(\frac{1}{\sigma} - 1 \right)} \right]^\sigma \quad (2.9)$$

Section 2.4: Application

We applied the method of periodic replacement with minimal repair and replacement to compute the expected cost rate. The data for this example comes from the telephone failure data explained in Chapter 1. We plot the Weibull probability plot for the data to assess the goodness of fit of the Weibull and gamma distributions to the data. Figure 2.1 shows the probability plot for the data comparing the two distributions. According to the Figure 2.1 both distributions showed competing fits to the data. In order to illustrate the use of the cost function, we chose the Weibull distribution. The parameters α and β of the Weibull distribution were estimated using the Bayesian approach, maximum likelihood estimation (MLE), and median rank regression (MRR) method. Non-informative priors were used in the Bayesian analysis (see Appendix A for the full model details used for the Bayesian analysis). Estimates of the parameters are summarised in Table 2.1.

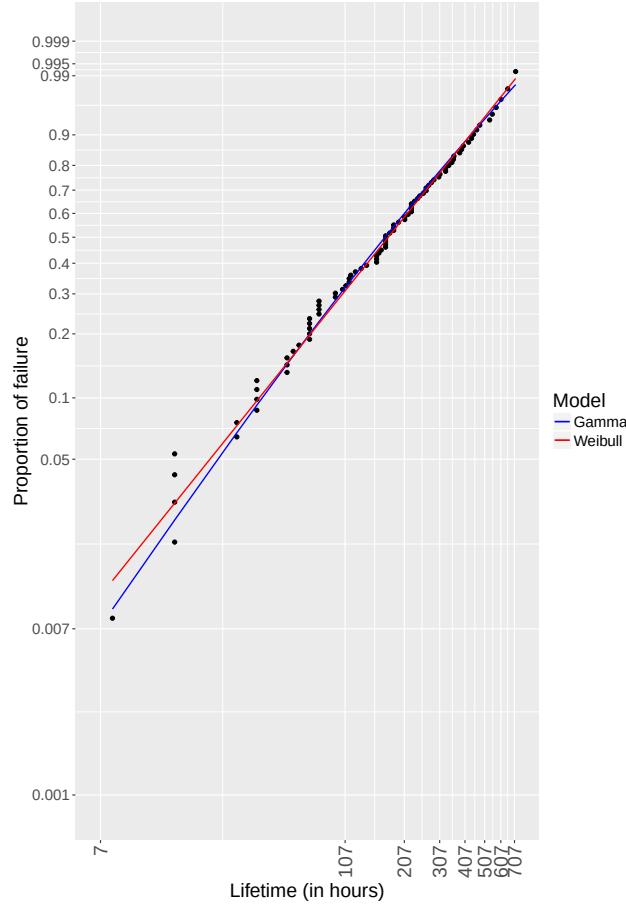


FIGURE 2.1: Weibull probability plot comparing the Weibull and gamma distributions.

TABLE 2.1: Parameter estimates with standard errors

Estimation method	Scale μ	Shape σ
MLE	5.4325 (9.2×10^{-3})	0.7632 (6.8×10^{-3})
MRR	5.4272 (1.1×10^{-2})	0.7680 (8.2×10^{-3})
BE	5.4337 (9.3×10^{-3})	0.7753 (7×10^{-3})

It is our interest to assess the cost effectiveness of the probabilistic decisions made in failure analysis. That is we are interested in assessing the cost of failure modes, uncertainty of failure model selection, monetary impact of parameter estimation of the failure models in decision making. For the purpose of evaluating the cost of each choice of parameter estimation method or selected failure model, we designed an optimal repair and replacement strategy for a fleet of components. Using that optimal repair and replacement time, we computed minimal expected cost rate for a scheduled replacement of a fleet of components with only minimal repairs at failure. However, we encountered the problem of determining the current monetary values of repair and replacement of a unit with the available details. As a remedy, we reversed engineered and approximated the cost ratio for the defined optimal time of replacement and then used it to evaluate the distribution of the minimal expected cost rate of repair and replacement of a unit. In other words, we fixed the optimal replacement time and evaluated the ratio of c_2/c_1 (Equation 2.3). When approximating the cost rate, the repair cost (c_1) was set to \$1. The same cost rate was used to compare the optimal strategies over different conditions.

Figure 2.2 shows the density function and the empirical distribution function (ecdf) of the expected minimal cost rate for an infinite life span of a single unit. The results are obtained from Bayesian estimation using the optimal replacement time. The observed minimal or optimal cost rate was \$0.006308 per unit hour corresponding to the optimal replacement time 330.738 hours. For all other replacement times, the expected cost rates were high. The distribution of the expected repair and replacement cost rate for suboptimal times are shown in Figure 2.3

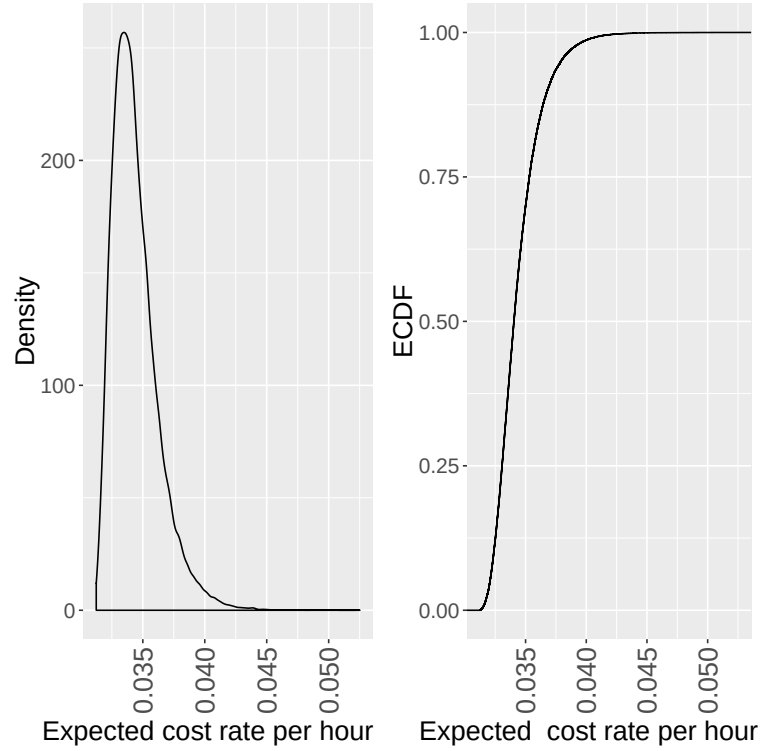


FIGURE 2.2: Probability distribution function and empirical cumulative distribution function of the expected cost rate per unit hour in \$ for the Weibull distribution. The repair cost of a unit was fixed to \$1. Parameters of the Weibull distribution was estimated with Bayesian analysis for the data.

After determining the optimal expected cost rate and looking at the empirical distribution, it is useful to analyse the changes in the cost distribution for any changes of other constraints. We carried out a sensitivity analysis varying the true shape of the Weibull distribution, parameter estimation method, the cost ratio c_2/c_1 , and the failure model. The optimal cost rate of repair and replacement was calculated by changing the above factors. The results are then used to produce several graphs with a series of curves against different choices of optimal replacement times.

Figure 2.4 shows the variation of the expected repair and replacement cost rate per hour for the Weibull distribution, according to various replacement to repair cost ratios. Variations of the curves approximately follow the same pattern. The optimal cost rate increases with increasing cost ratios irrespective of the quantile (or replacement time).

Figure 2.5 provides the optimal repair and replacement cost rates for different values of the shape parameter of the Weibull distribution. For larger quantiles, the repair and replacement cost rate increases with increasing shape parameter. On the other

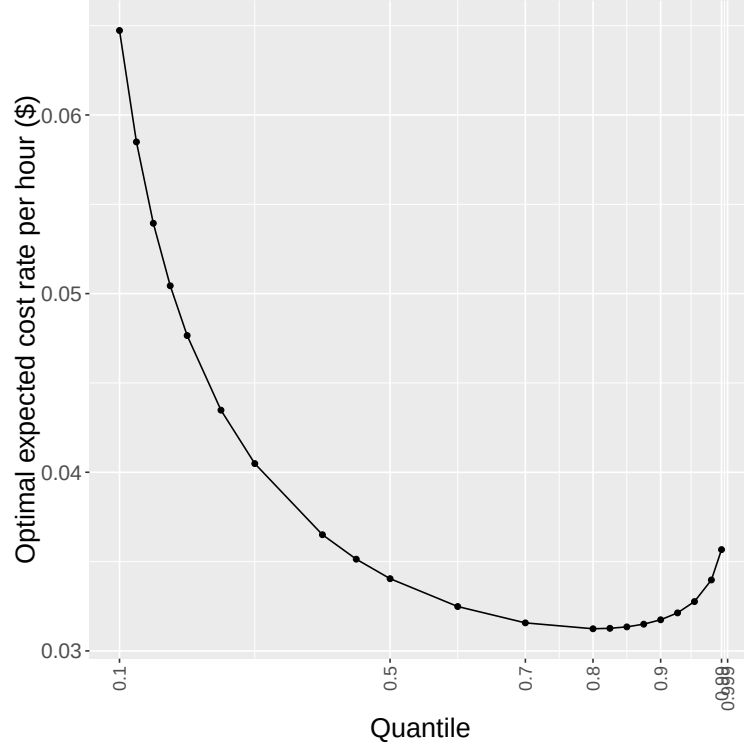


FIGURE 2.3: Distribution of repair and replacement cost rates (in \$) for suboptimal solutions. The repair cost was set to \$1 and the Weibull distribution was fitted to the data. Bayesian estimation was used in estimating the Weibull parameters.

hand, the cost rates decrease with increasing shape parameters for the lower quantiles.

Figure 2.6 presents the optimal repair and replacement cost rates computed for varying quantiles from two different time-to-failure models; Weibull and gamma. Resulting curves follow approximately the same pattern. Approximately after 30th quantile, the gamma model provides relatively higher cost rates than the Weibull distribution.

We observed similar cost curves for all the three-parameter estimation methods; BE, MLE, and MRR method. The results are presented in figure 2.7.

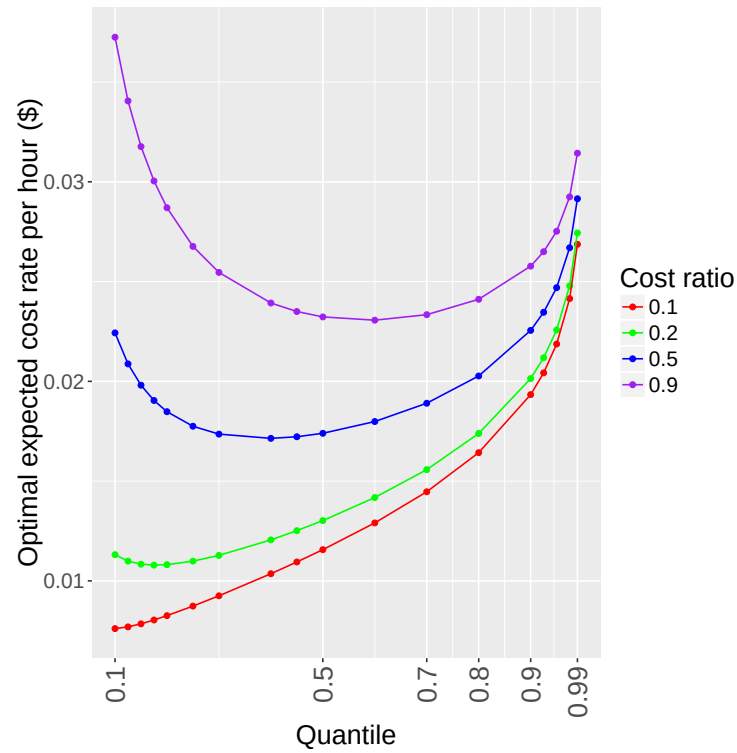


FIGURE 2.4: Optimum expected cost rates per unit hour (in \$) for varying cost ratios of replacement to repair and quantiles for the Weibull distribution. The repair cost was set to \$1 when computing the cost rates.

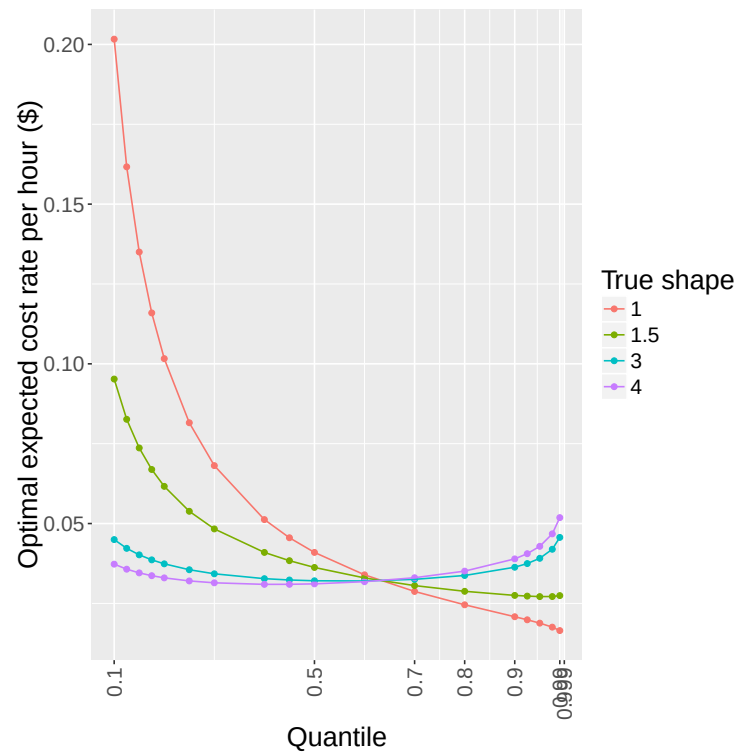


FIGURE 2.5: Variation of optimal repair and replacement cost (\$\$) rates for varying shape parameters of the Weibull distribution

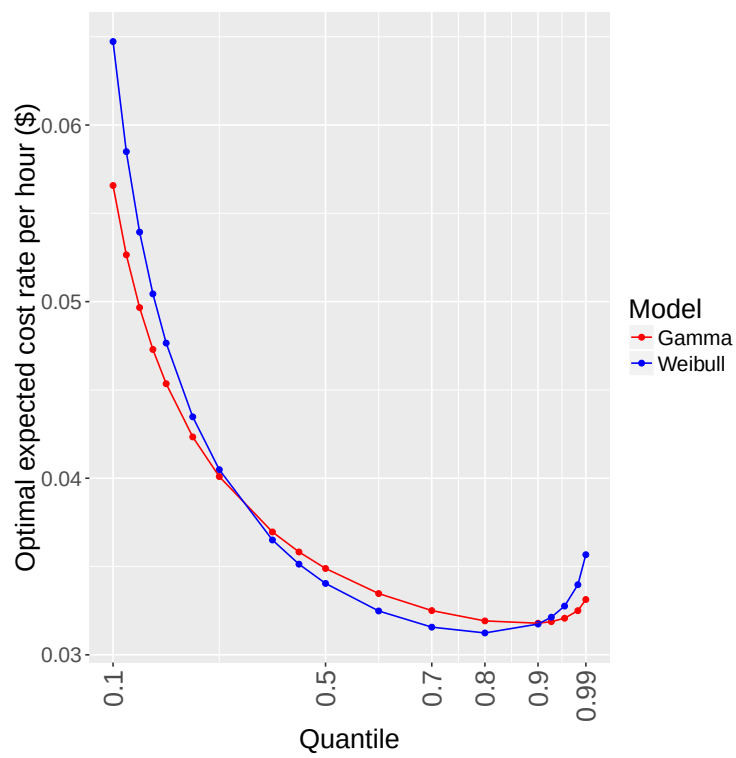


FIGURE 2.6: Optimal repair and replacement cost rate given by the Weibull and the gamma distributions in \$. The repair cost was set to \$1.

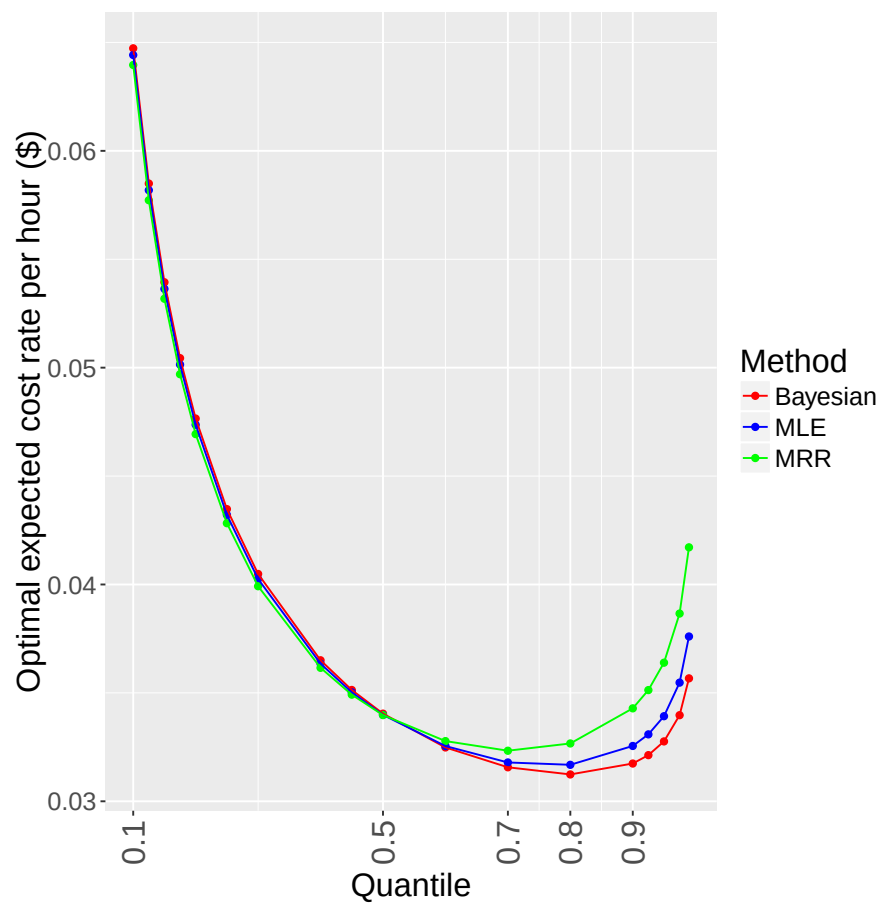


FIGURE 2.7: Optimal repair and replacement cost resulting from the parameter estimation methods; BE, MLE and MRR method. The Weibull distribution was used as the failure model and the repair cost was set to \$1.

Section 2.5: Discussion

In this chapter, we have demonstrated the use of periodic replacement and minimal repair at failure policy with a practical example of telephone failure data. Relatively, the repair costs are higher than the replacement cost of a component. Depending on that fact, it is worthwhile to develop a replacement strategy with minimal repair at failure to minimize the cost of maintenance.

Many pieces of literature are available on the general parameter estimation problems with numerous examples of applications and comparisons. Stepping forward, we were focused on making probabilistic decisions based on the parameter estimations and assessing the cost-effectiveness of such decisions. This chapter acted as a supporter in evaluating and comparing the loss or gain of the probabilistic decisions. In order to achieve our goal, we set the replacement time of the components to a specific quantile of the failure distribution. Then the total repair and replacement cost rate at that quantile was computed with the methods explained in Section 2.2.

The parameter estimates of the fitted Weibull distribution from the three estimation methods; MLE, MRR, and BE showed similar values for both shape and scale. The fit of the Weibull distribution to the data was justified by plotting the data on the Weibull scale. The shape parameter of the Weibull distribution ($\sigma = 1/\beta$) was less than 1, indicating an increasing failure rate of the telephones.

The sensitivity analysis enabled us to compare the optimal expected cost rates for changing values of cost ratios (c_1/c_2), shape parameter of the Weibull distribution, two failure models; Gamma and the Weibull and three different parameter estimation methods. It was noticeable that the two failure models used in the sensitivity analysis resulted in approximately the same minimal cost rates at the chosen quantiles. As a result, we can suspect that the gamma distribution also provides a competing fit to the failure data used in the example and suggest an alternative for the common failure model; the Weibull.

Lack of knowledge of the actual monetary values of the repair and replacement costs was a limitation of our analysis. Nevertheless, all the monetary values are known in any practical situation, and the application remains the same. Moreover, as described

in Nakagawa (1981), the extension of the periodic replacement with minimal repair at failure policy is simple for n number of components.

In summary, a cost analysis of the probabilistic decisions allows the decision-maker to evaluate the economic risk of the choices. Not only it is appropriate to use a cost computation method such as periodic replacement with minimal repair at the failure to minimise the expected costs, but also it helps to make robust probabilistic decisions in monetary terms.

3. Quantile estimation with maximum likelihood estimation (MLE) and median rank regression (MRR) method

Robust estimation of the distribution parameters is vital as we predict future failures based on past failure patterns. Furthermore, accurate failure predictions lead to a high level of readiness and safety. Frequently, we observe heavily censored data in reliability studies. So, it is important to measure the robustness of the statistical methods in estimating the hazard when the data are highly censored. In this chapter, we compare two parameter estimation methods for quantile estimation. We consider the MLE and the median rank regression (MRR) method in estimating the quantiles of the Weibull distribution. Maximum likelihood estimation (MLE) is one of the widely used parameter estimation methods in statistical analyses.

Initially, we explore the literature of the two parameter estimation methods. Then we design a simulation study to compare the performance of the two estimation methods.

Section 3.1: Background

The Weibull distribution is one of the widely used distribution to model failure data. Apart from the log-location-scale form of the Weibull distribution presented in Chapter 1, probability density function (pdf) of the Weibull distribution can be expressed as,

$$f(x|\alpha, \beta) = \frac{\alpha}{\beta} \left(\frac{x}{\beta}\right)^{\alpha-1} \exp\left(-\left(\frac{x}{\beta}\right)^{\alpha}\right), \quad x > 0 \quad (3.1)$$

Here, β is the scale parameter and α is the shape parameter of the distribution.

The shape parameter of the Weibull distribution reflects the failure rate of a system. If the shape parameter $\alpha < 1$, that system has a failure rate decreasing with time. If $\alpha = 1$, the failure rate of a system decreases but at a slower rate. Figure 3.1 shows the shape of the Weibull distribution for different α values.

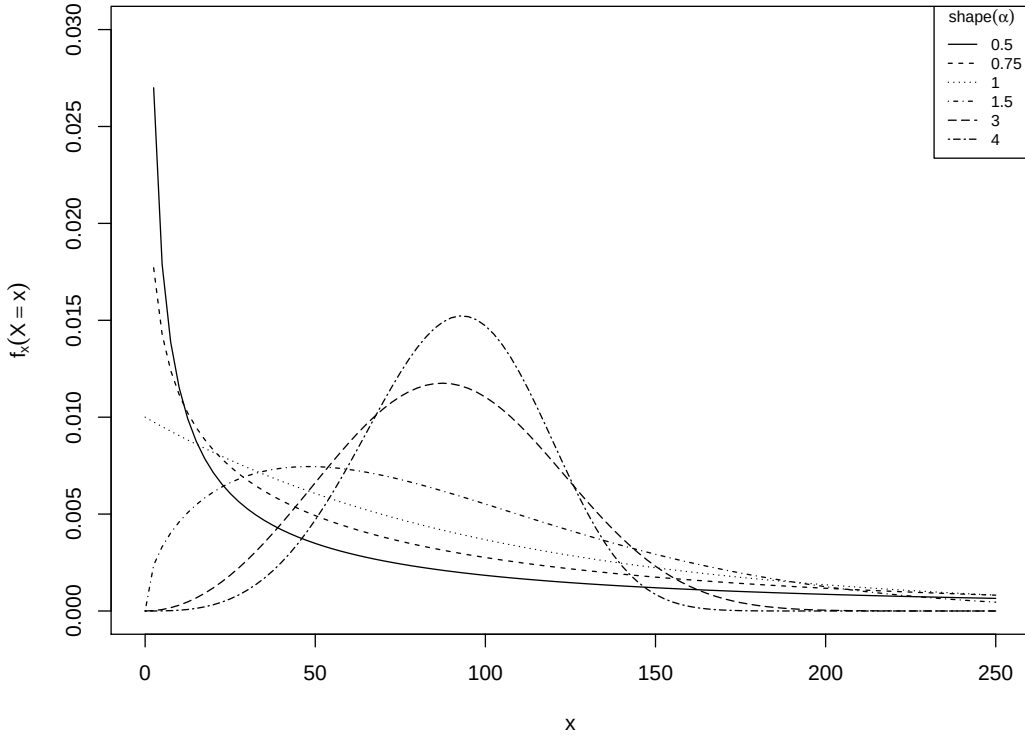


FIGURE 3.1: Distribution of the Weibull model for different α values.

The ML method and MRR method are frequently used parameter estimation methods in reliability analysis. However, the optimality of parameter estimation method can vary according to the type and degree of censoring, sample size, and the evaluation criteria used. In the ML method, first two derivatives of the log-likelihood function must be defined. Additionally, the Fisher information matrix must not be zero and must be a continuous function of the parameter. Additionally, we need to know the form of the distribution function of the random variables. Some of the ML problems may encounter with more than one maximum solution for the likelihood function where some problems may have no ML parameter estimates. For example, when the number of failures is equal to zero, ML estimates do not exist (Jeng and Meeker, 2000).

However, ML method possesses some excellent properties which has expanded the use of the method in parameter estimation. A detailed explanation for the common uses of ML estimation is given by Genschel and Meeker (2010) (see also Meeker and Escobar, 1998; Reiser and Lev, 1979; Cohen and Whitten, 1982; Dümmbgen, Rufibach, and Schuhmacher, 2014; Duerinckx, Ley, and Swan, 2014). On the other hand, when the outliers are present, ML estimations can be unstable (Goegebeur, Guillou, and Rietsch, 2015). Thus, the reliability of the decisions made based on the ML method is questionable, when the data are contaminated.

The MRR method is popular among the practitioners due to its ease of mathematical computation, especially with censored data. Genschel and Meeker (2010) compared ML estimation and MRR methods for estimating Weibull parameters and functions of parameters through a simulation study under different censoring schemes. Their study was motivated by factors such as; the popularity of the Weibull distribution in reliability analysis and, the clear differences of opinions of the usage of the two methods in parameter estimation. The simulation results revealed that MLE outperformed MRR when evaluated under their criteria (i.e. mean squared error, relative efficiency and bias estimates). Also, when the parameters were estimated from data with Type I censoring, the use of ML method was advantageous as it considers the information contained in the location of the censored data whereas MRR does not. With Type II censoring, Genschel and Meeker (2010) found that MLE produces more precise estimators but the differences were smaller than those obtained under Type I censoring. Olteanu and Freeman (2010) evaluated the performance of MLE and MRR method for parameter estimation of the Weibull distribution for small numberd of failures and highly right censored data, particularly censoring greater than 99%. According to their findings, for the highly censored data, none of the methods clearly outperformed the other.

Another comparison of MLE and least squares (LS) methods for censored data for estimating the parameters of the Weibull distribution was presented by Hossain and Zimmer (2003). Their recommendation of a consistent estimator for right censored data was the Herd-Johnson LS method. Somboonsavatdee, Nair, and Sen (2007) graphically examined the use of several LS methods with MLE in estimating some log

location-scale distributions. Based on the relative efficiency values the LS methods performed well for the lognormal distribution, reasonably well for the log-logistic distribution, and poorly for the Weibull distribution. Nevertheless, MRR method can not apply for some competing failure models such as the gamma distribution and the inverse-Gaussian distribution. Because the quantile functions of the two distributions do not have closed forms for their cdfs. Additionally, in MRR method the information related to the position of the censored data after the final failure are not considered.

Section 3.2: Maximum Likelihood Estimation (MLE)

Maximum likelihood estimation is a procedure of determining an estimator $\hat{\theta}$, of θ which maximizes the likelihood of a function. Smith (2002) defined the likelihood $L(\theta)$ of the observed data as a multiple of the joint distribution of the observed data. The maximum likelihood estimator $\hat{\theta}$ of a parameter θ is defined as a function of observed data which maximizes L over values of θ in the parameter space of all possible values of θ .

The likelihood function for a set of independent and identically distributed random variables, $T_1, T_2, \dots, T_n$ is defined as,

$$L(\theta, t) = c \prod_{i=1}^n f(t_i; \theta) \quad (3.2)$$

In maximum likelihood estimation of θ requires a solution to the equation $L'(\theta) = 0$. However, it is often easier to solve the log likelihood rather than the likelihood function [$l(\theta) = \ln(L(\theta))$].

When the data are right-censored, the likelihood of the censored sample can be given by;

$$L(\theta, t) = C \prod_{i=1}^n [f(t_i, \theta)]^{\delta_i} [1 - F(t_i, \theta)]^{1-\delta_i} \quad (3.3)$$

where,

$$\delta_i = \begin{cases} 1 & \text{if component fails at time } t \\ 0 & \text{if component is censored} \end{cases}$$

Section 3.3: Median Rank Regression (MRR)

Median rank regression (MRR) is a parameter estimation procedure in which a least-squares regression line is fitted through the probability plot to estimate the parameters. Application of MRR to Weibull data has discussed in Olteanu and Freeman, 2010. In MRR, it is assumed that the data are linearly related to the quantiles of the distribution. There are different ways of calculating the median rank plotting probabilities. We first rank all the observations; both censored and fails from lowest to highest. Suppose we have n total observations, then the ordered times can be denoted by i where i ranges from 1 to n . The reversed rank only for the observed failure times are given by; $r_k = n - i_k + 1$, $k = 1, \dots, r$, where i_k is the rank of the k^{th} failure among all the ranked observations. We adjusted the ranks for the censored data by computing the reverse rank for the k^{th} failure from the recursive formula;

$$r_k^A = r_{k-1}^A + \frac{n + 1 - r_{k-1}^A}{n + 1 - (k_i) - 1} \quad (3.4)$$

Here, $r_0^A = 0$. Then we used the adjusted ranks to calculate the median rank plotting positions according to the Benard's formula (Benard and Bos-Levenbach, 1955).

$$MR[i] = \frac{r_i - 0.3}{n + 0.4} \quad (3.5)$$

Here, r_i is the i^{th} rank of the data and n is the total number of observations.

As mentioned earlier, the applications of the Weibull distribution as the failure model is frequent. Moreover, the problem of the choice of estimation method to use under different types of censoring and sample sizes has addressed by many authors. The results vary according to the comparison strategies used. However, it should be noted

that most of these studies are restricted to the applications of ML estimation and MRR method.

Section 3.4: Numerical results

This section summarises the results obtained from the simulation study to compare the two parameter estimation methods. We estimated the 10th and the 80th quantiles of the Weibull model. Reason for selecting the 80th was, we realized it is a reasonable replacement time for the system of telephones introduced in Chapter 1. We also estimated a lower quantile to examine the differences according to the degree of censoring and sample sizes. After observing approximately similar patterns for different scale values (the scale parameter only stretches the values across the x-axis), we set the scale parameter to a single value in the simulation.

We first simulated a data set with 100% failures and then introduced censoring. Data were then censored with 75% failures, and 10% failures. After estimating the quantiles, we computed the repair and replacement cost at each quantile. We considered sample sizes 10 and 500 for the complete data in order to compare the results with the heavily censored cases (i.e. 10% fails).

3.4.1 Complete data - No censoring

The simulation study compromised of the following factors for the complete data.

- one scale value (100)
- four different shape values (1.5, 3, 4, 6)
- three different sample sizes (10, 50, 100, 500, 1000)
- two different quantiles (10th, 80th)
- 100% failure
- 1000 replicates

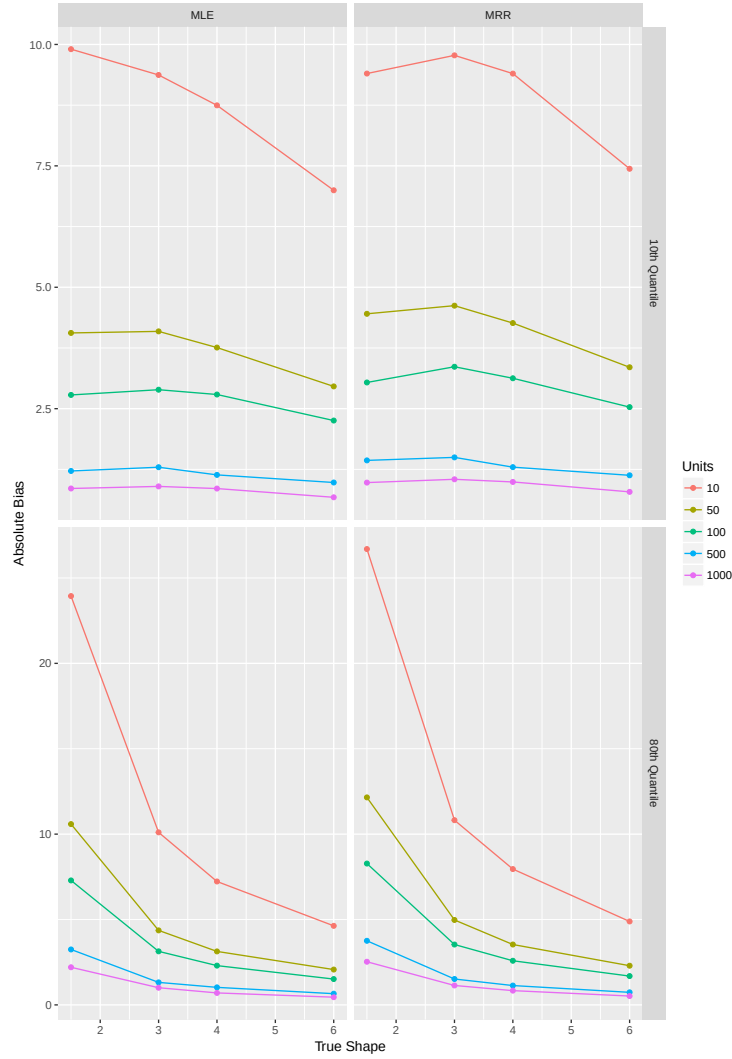


FIGURE 3.2: Absolute bias of the estimated 10^{th} quantile and the 80^{th} quantile using the MLE and the MRR method. Each point is the mean of 1000 simulations.

Figure 3.2 shows the absolute bias of the estimated 10^{th} quantile and the 80^{th} quantile. The absolute bias in estimated quantiles from both MLE and MRR decreases towards zero with increasing sample sizes, and with the increasing true shape parameter. The magnitude of the absolute bias is slightly higher for the estimates from MRR than the MLE, and the pattern is relatively similar for both quantiles.

Figure 3.3 depicts the simulation results for the mean squared error of the estimates of quantiles from MLE and MRR method. The pattern is relatively similar to that of absolute bias. The MSE of the estimated quantiles decreases towards zero with increasing sample sizes. The MSE is comparatively low for the estimates of the 10^{th} quantile than the 80^{th} quantile.

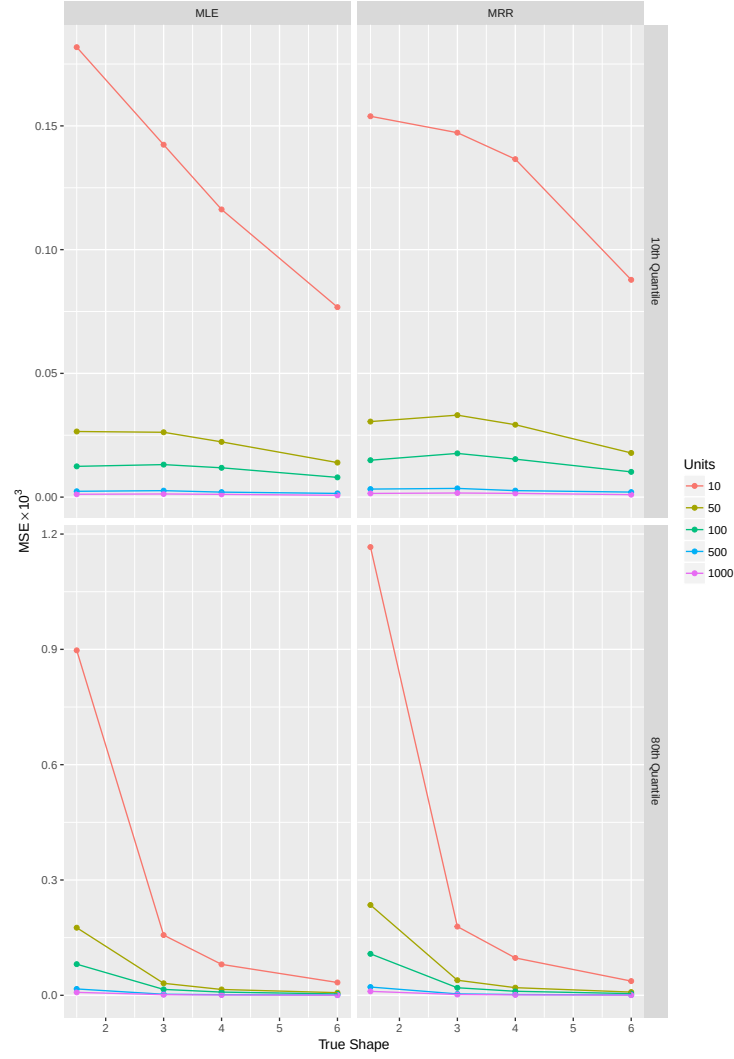


FIGURE 3.3: Mean squared error of the estimated 10^{th} quantile and the 80^{th} quantile using the MLE and the MRR method. Each point is the mean of 1000 simulations.

Figures 3.4 and 3.5 show the empirical distribution and empirical cumulative distribution of the total expected cost rate computed at the 10^{th} quantile from equation 2.3. The repair cost was set to \$1 and the optimal cost ratio c_2/c_1 was evaluated for the true quantile value. Then, we used the optimal cost ratio in the total expected cost rate calculation. For different sample sizes the pattern is relatively constant over the true shape parameter for both MLE and MRR methods. The empirical distribution of the cost rate becomes concentrated with increasing sample size. The empirical distribution from the MLE is slightly concentrated than the distribution from the MRR method for all the factor combinations.

Figure 3.6 represents the distribution of the expected repair and replacement cost rate

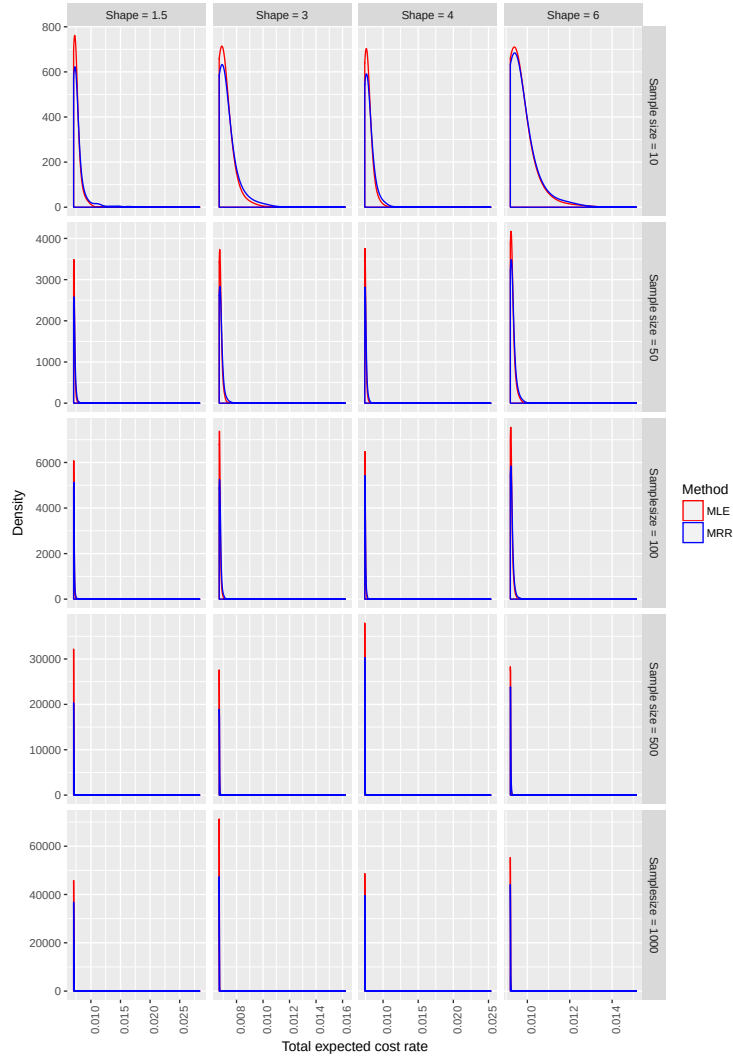


FIGURE 3.4: Distribution of the total expected repair and replacement cost rate estimated at the 10th quantile from MLE and MRR method. The repair cost is set to 1.

estimated at the 80th quantile. Not surprisingly, the expected cost rates are higher at the 80th than the 10th quantile. At the optimum replacement time (at the 80th quantile) we observed the minimum costs. The spread of the distributions increases with increasing shape parameter. For all combination of simulation factors, MLE provides more concentrated values than MRR method. Figure 3.7 shows the empirical cdf of the expected cost rate. The expected cost rates increase with increasing shape parameter values.

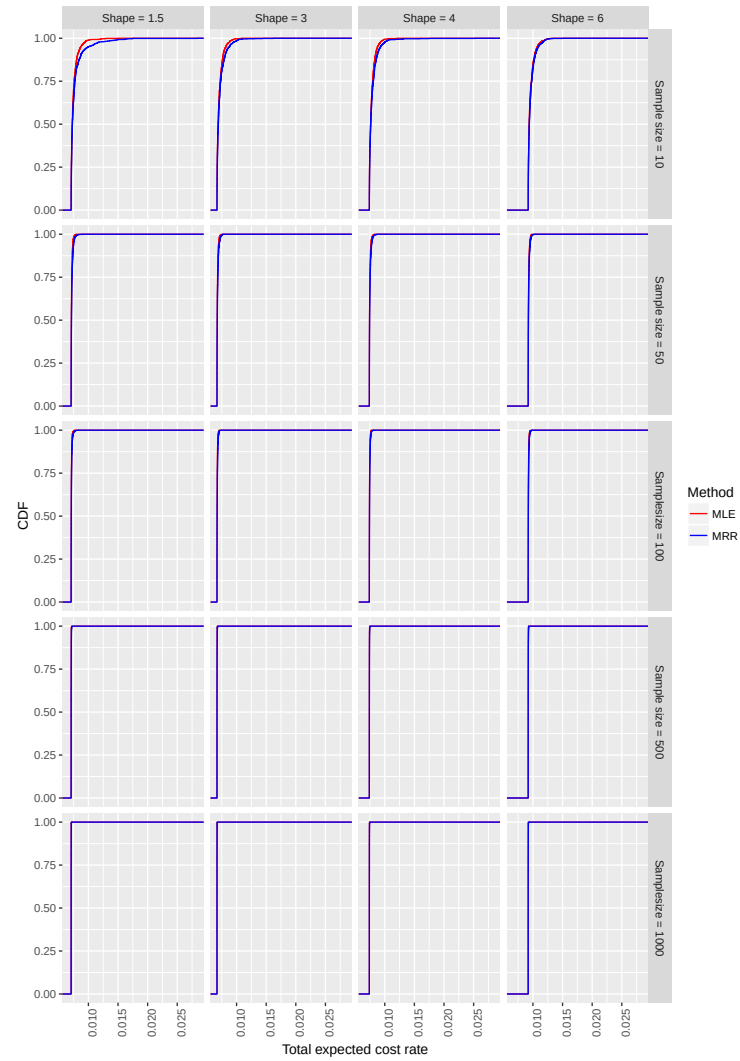


FIGURE 3.5: Empirical density of the total expected cost rate estimated at the 10^{th} quantile from MLE and MRR method. The repair cost is set to 1.

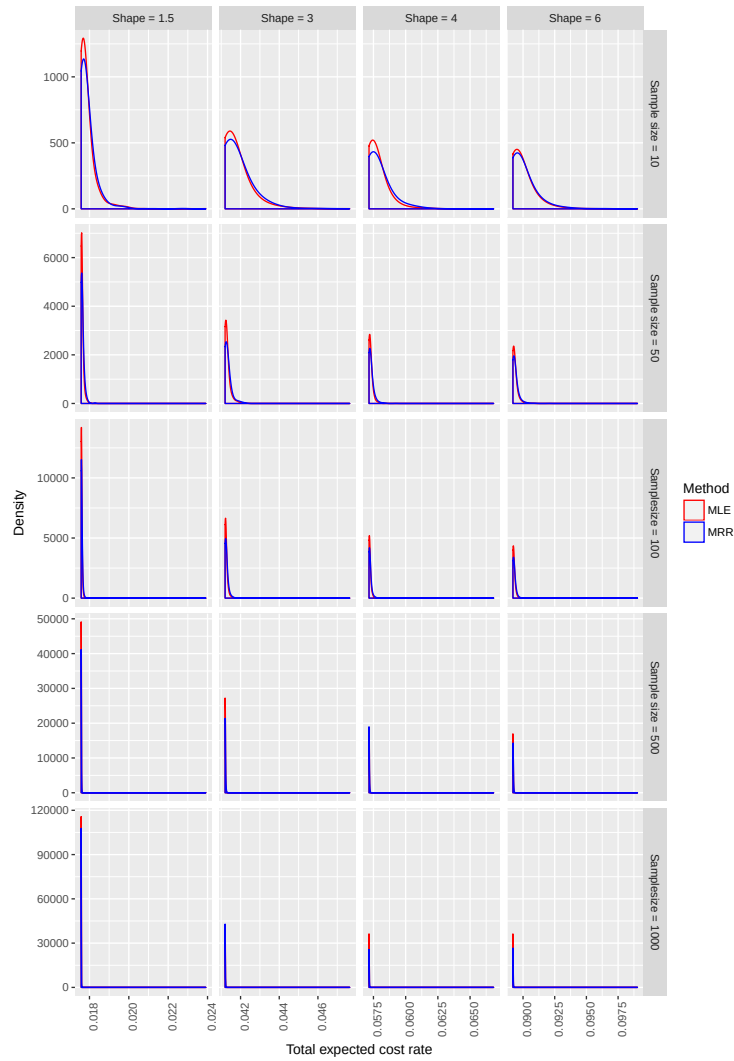


FIGURE 3.6: Distribution of the total expected repair and replacement cost rate estimated at the 80th quantile from MLE and MRR method. The repair cost is set to 1.

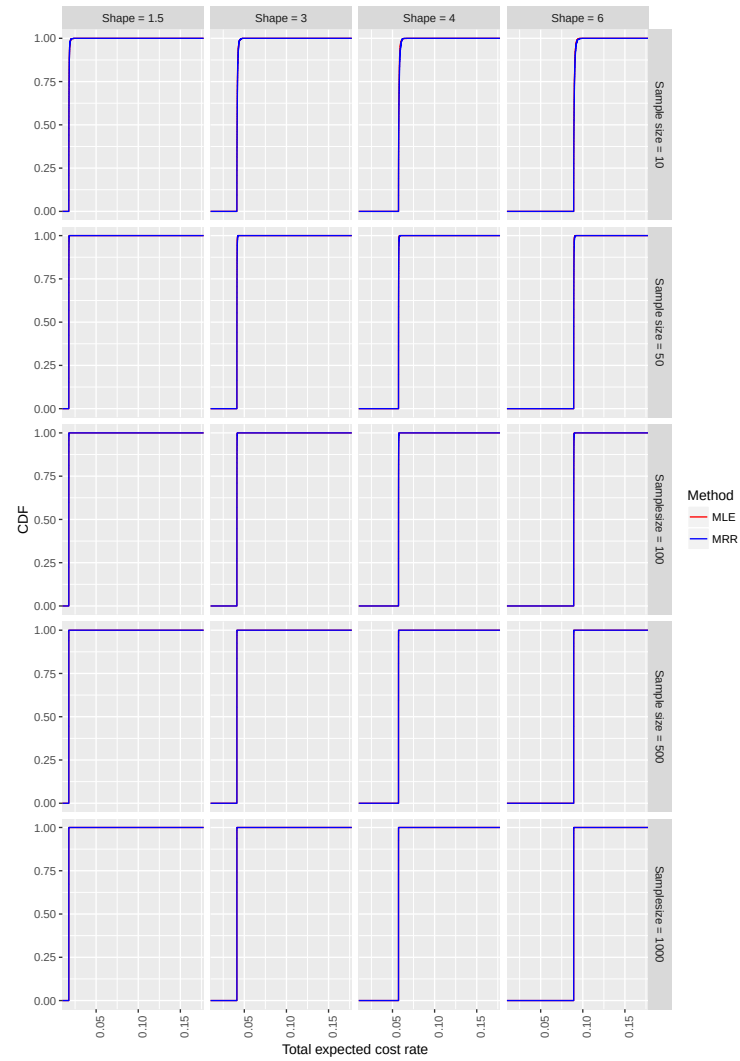


FIGURE 3.7: Empirical cumulative density of the total expected cost rate estimated at the 80th quantile from MLE and MRR method. The repair cost is set to 1.

3.4.2 75% Fail

This section summarises the simulation results obtained from censoring the data by 25%. The specifications used for this set of simulations are outlined below. Sample sizes 10 and 500 are removed from the simulation. The true values of the parameters are used to acquire the actual values of the estimates.

- one scale value (100)
- four different shape values (1.5, 3, 4, 6)
- three different sample sizes (50, 100, 1000)
- two different quantiles (10^{th} , 80^{th})
- 75% failure
- 1000 replicates

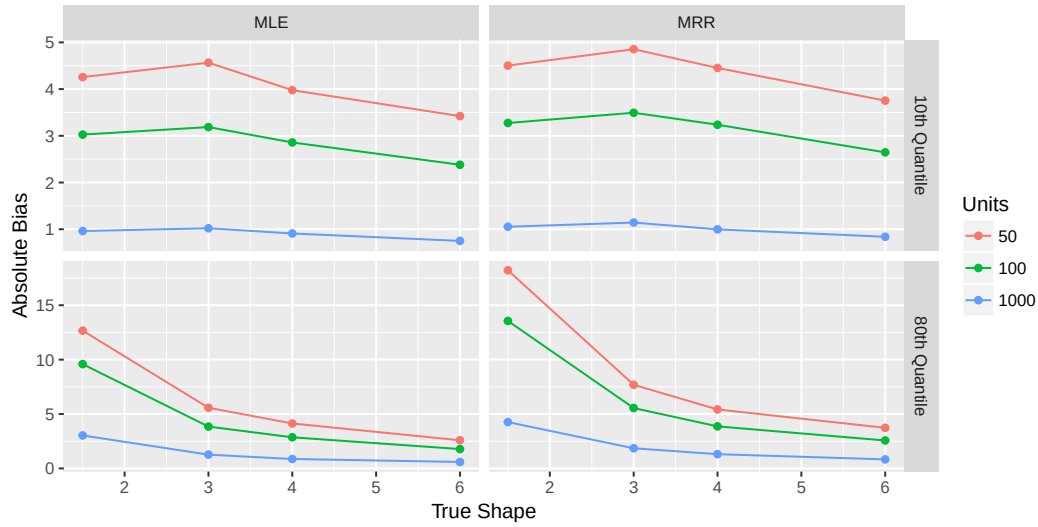


FIGURE 3.8: Absolute bias of the estimated 10^{th} quantile and the 80^{th} quantile using the MLE and the MRR method. The data are 25% censored. Each point is the mean from 1000 simulations.

Figure 3.8 presents the absolute bias of the quantiles 10^{th} and 80^{th} , for both MLE and MRR methods. It appears the absolute bias has slightly increased compared to the complete data. The trends are approximately the same for all the sample sizes. When the sample size and shape parameter, α increase, the absolute bias tends to decrease towards zero.

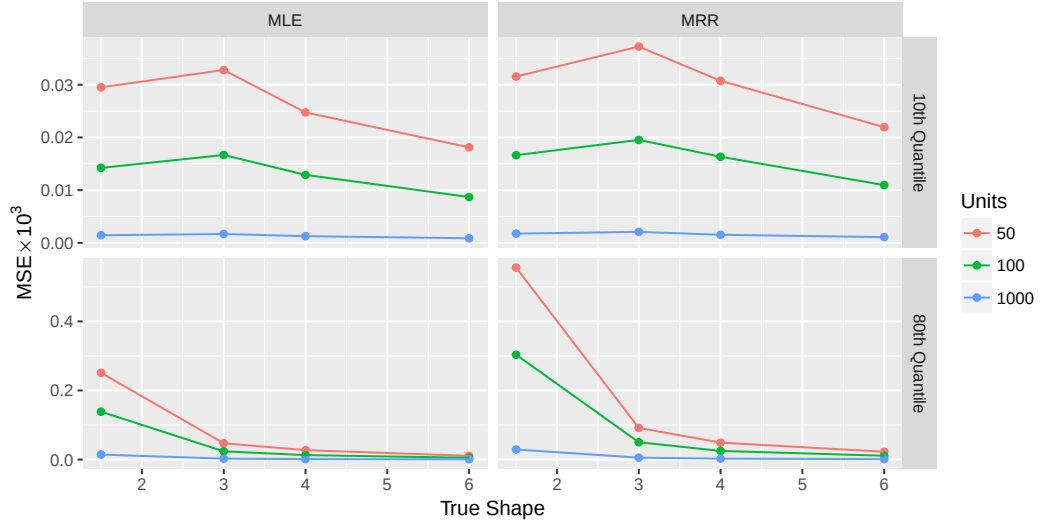


FIGURE 3.9: Mean squared error of the estimated 10th quantile and the 80th quantile using the MLE and the MRR method. Each point is the mean of 1000 simulations.

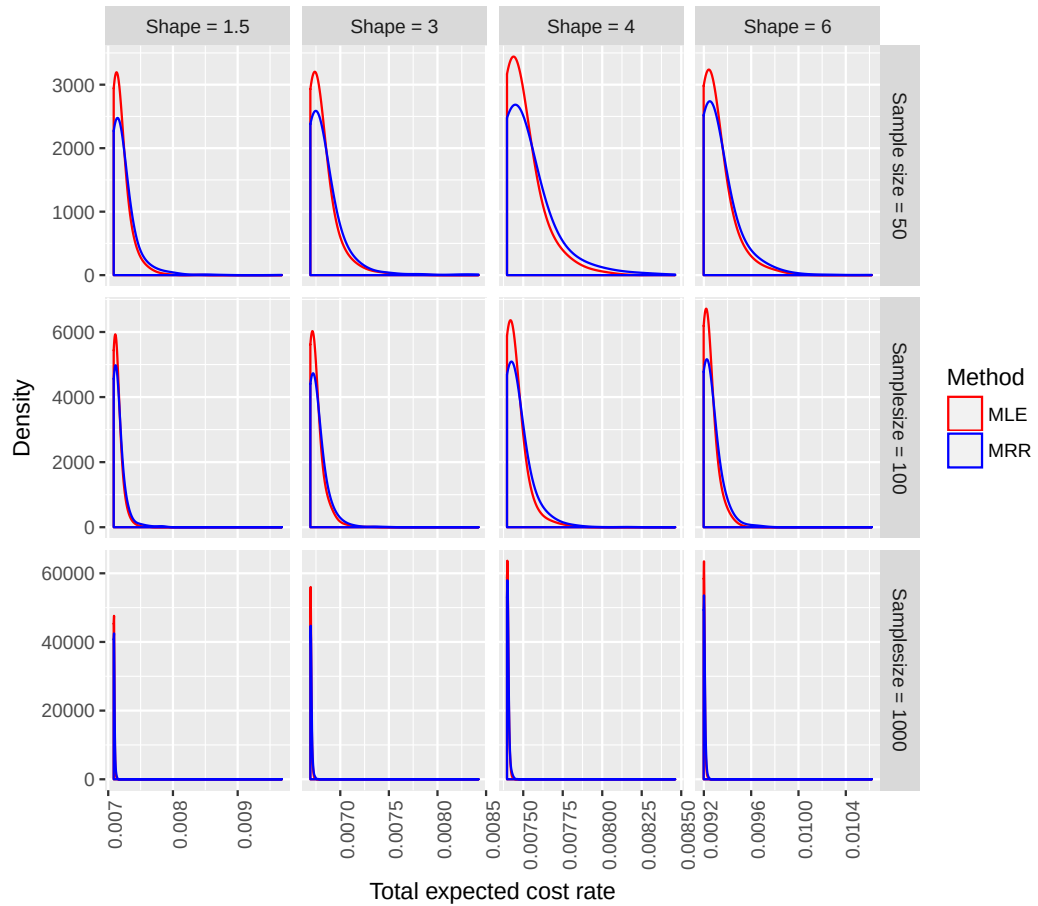


FIGURE 3.10: Distribution of the total expected repair and replacement cost rate estimated at the 10th quantile for the censored data (25%) from MLE and MRR method. The repair cost is set to 1.

Figure 3.9 shows the MSE of the quantile estimates from both MLE and MRR methods. The MSE values decrease towards zero with increasing sample sizes and increasing shape parameter, α . The patterns appear to follow approximately the same for all the sample sizes. The MSE values from 10^{th} quantile show a peak at $\alpha = 3$.

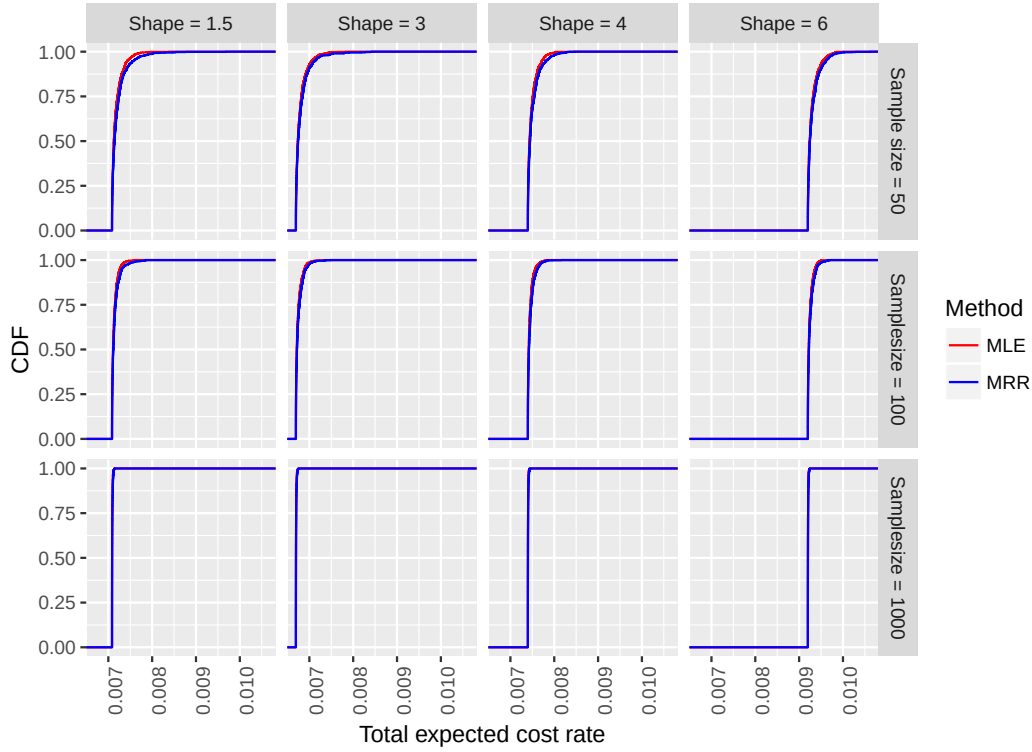


FIGURE 3.11: Empirical density of the total expected cost rate estimated at the 10^{th} quantile from MLE and MRR method. The data are censored with 75% fails. The repair cost is set to 1.

We then computed the repair and replacement costs for each quantile arose from the MLE and the MRR method. As expected, the cost rates are higher for the censored data than for the complete data. When the shape parameter increases, the cost rates increase gradually for both quantiles. Figure 3.10 and 3.12 illustrate the cost rates computed for 10^{th} and 80^{th} quantiles respectively. The empirical cdf shown in figures 3.11 and 3.13 confirms the earlier observations. The distributions have shifted to the right with increasing shape parameter, α .

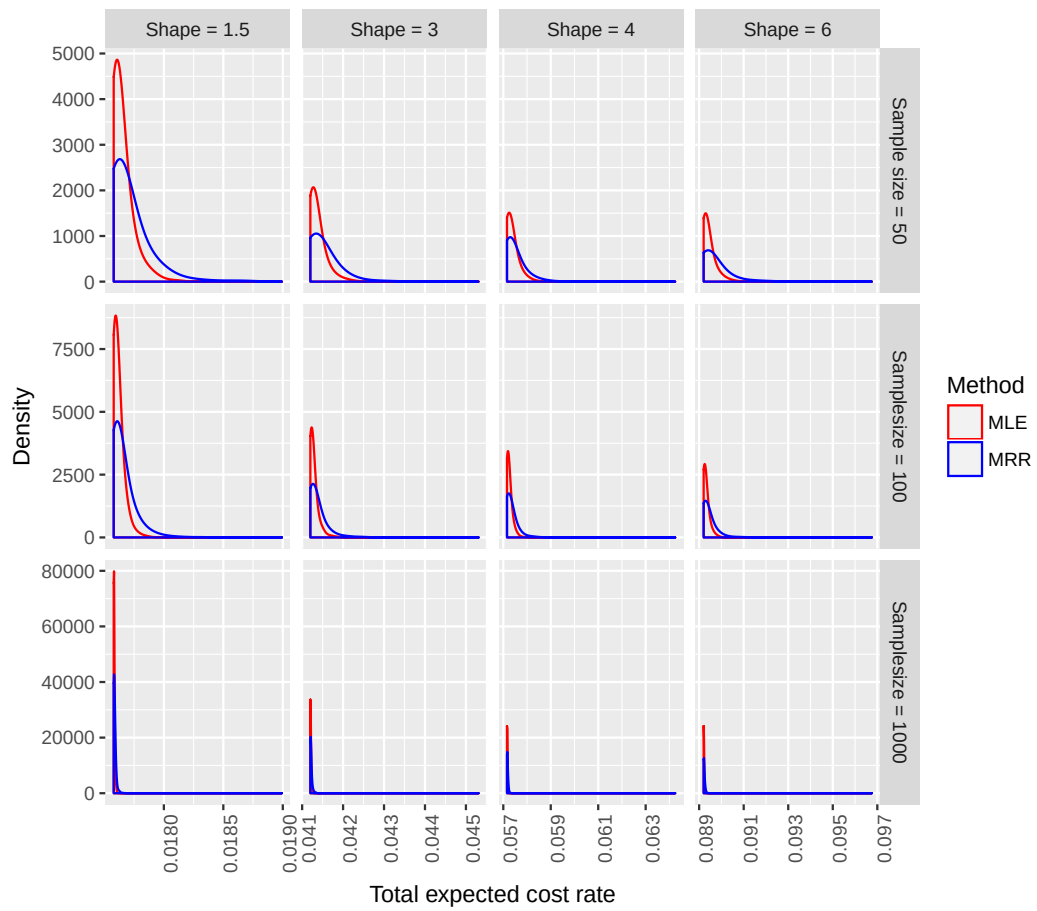


FIGURE 3.12: Distribution of the total expected repair and replacement cost rate estimated at the 80th quantile from MLE and MRR method with censored data (75% fails). The repair cost is set to 1.

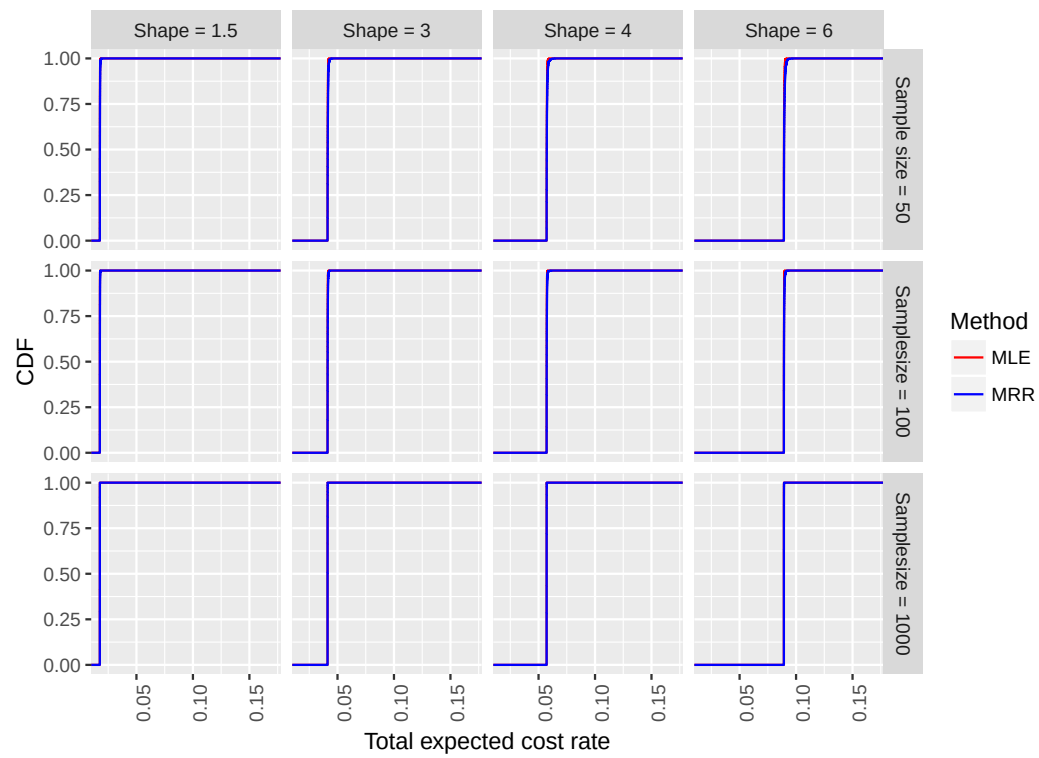


FIGURE 3.13: Empirical cumulative density of the total expected cost rate estimated at the 80^{th} quantile from MLE and MRR method for 75% fails. The repair cost is set to 1.

3.4.3 10% Fail

In this section, we introduce heavy censoring to the simulation. It is our interest to observe the impact of the heavy censoring on the repair and replacement costs and the quantile estimates. We censored the sample by 90%. The specifications used for the simulation are as follows. The true values of the parameters are used to acquire the actual values of the estimates.

- one scale value (100)
- four different shape values (1.5, 3, 4, 6)
- three different sample sizes (50, 100, 1000)
- two different quantiles (10^{th} , 80^{th})
- 10% failure
- 1000 replicates

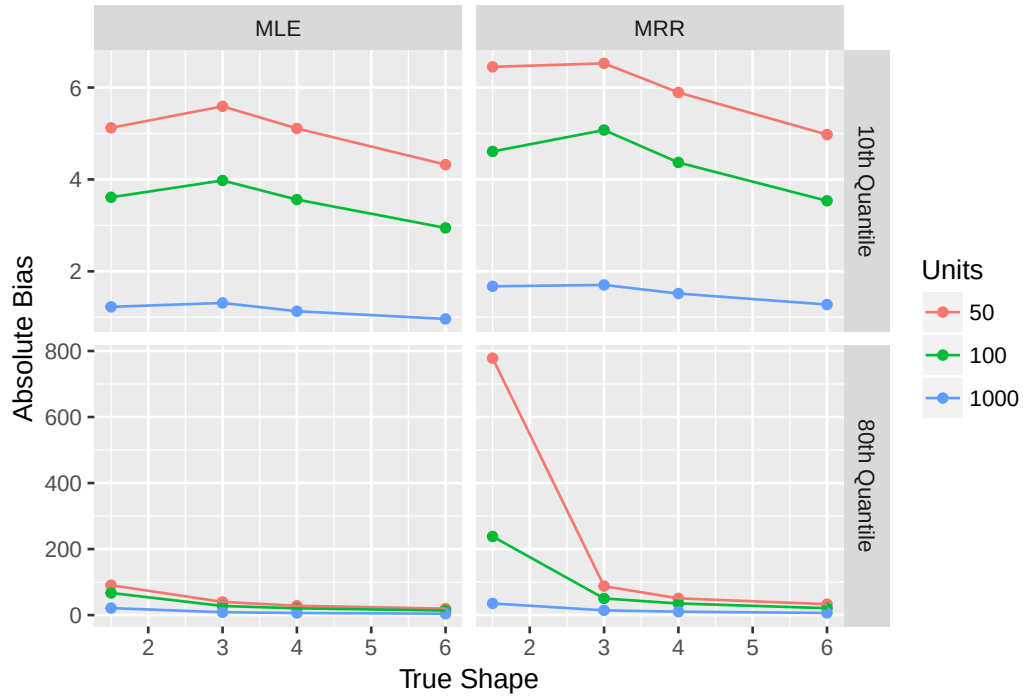


FIGURE 3.14: Absolute bias of the estimated 10^{th} quantile and the 80^{th} quantile using the MLE and the MRR method. The data are 90% censored with only 10% fails. Each point is the mean from 1000 simulations.

Figure 3.14 presents the absolute bias of the quantile estimates from both MLE and MRR methods. It resembles that heavy censoring has dramatically affected the absolute bias of the estimates for both quantiles. The absolute biases of the 80th quantile show noticeably larger values compared to the 10th quantile. The MRR method produces slightly higher bias values than the MLE method. Furthermore, there is a peak at $\alpha = 3$ for the 10th quantile. Absolute biases decrease towards zero with increasing sample sizes and increasing shape parameter. It seems that the reduction of the effective sample size in the censored case has caused this increment of bias. When comparing the "1000" and "100" units lines in Figure 3.14 respectively with "100" and "10" units lines in Figure 3.2, it can be seen that the bias has increased in the former approximately by two times the latter. It is evident the introduction of heavy censoring has increased the bias of the estimates.

Figure 3.15 shows the MSE values for both 10th and 80th quantiles from MRR method and MLE. MSE values have also increased as a result of heavy censoring. At $\alpha = 1.5$, the MRR method has produced considerably higher MSE values. Overall, the MSE values decrease with increasing sample sizes and increasing shape parameter.

Introduction of heavy censoring has increased the repair and replacement cost rates calculated at both quantiles. Figure 3.16 and 3.18 show the distributions of repair and replacement cost rates for 10th and 80th quantiles respectively. Similar to the earlier observations, the cost rates increase with increasing shape parameter. Also, when the sample size increase, we observed concentrated cost rates. Figures 3.17 and 3.19 represent the corresponding empirical cdf for 10th and 80th quantiles respectively.

Figure 3.15 shows the MSE values for both 10th and 80th quantiles from MRR method and MLE. MSE values have also increased as a result of heavy censoring. At $\alpha = 1.5$, the MRR method has produced considerably higher MSE values. Overall, the MSE values decrease with increasing sample sizes and increasing shape parameter.

Introduction of heavy censoring has increased the repair and replacement cost rates calculated at both quantiles. Figure 3.16 and 3.18 show the distributions of repair and replacement cost rates for 10th and 80th quantiles respectively. Similar to the earlier observations, the cost rates increase with increasing shape parameter. Also,

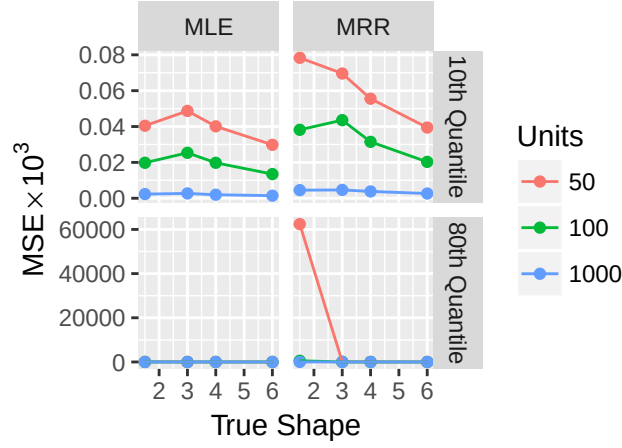


FIGURE 3.15: Mean squared error of the estimated 10th quantile and the 80th quantile using the MLE and the MRR method, 10% fails. Each point is the mean of 1000 simulations.

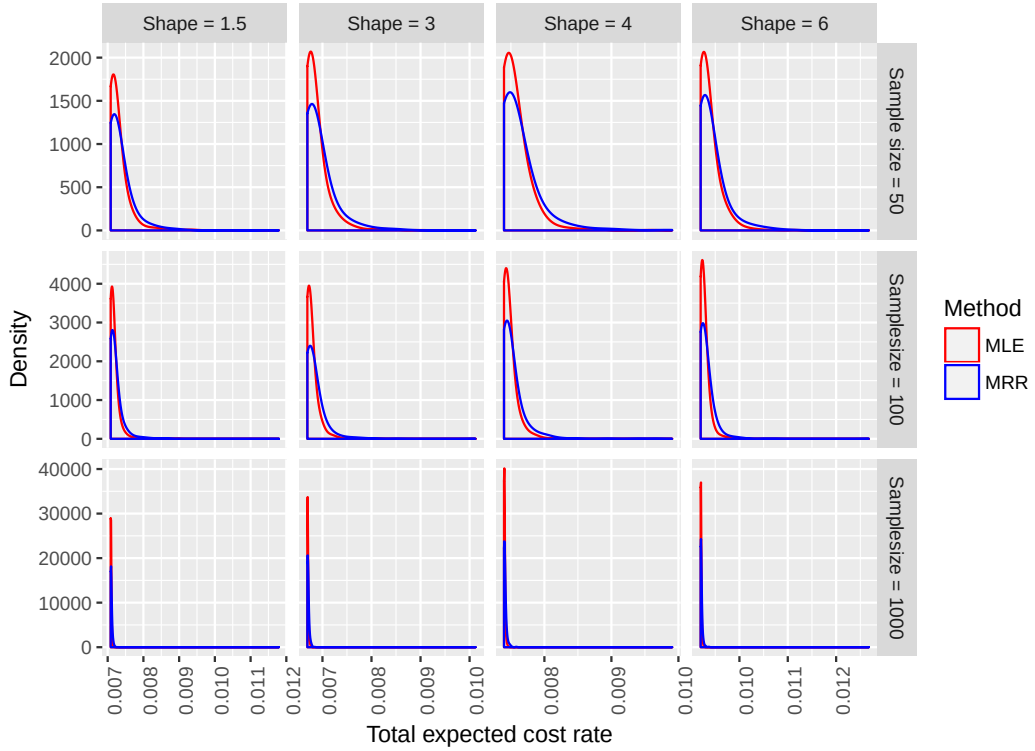


FIGURE 3.16: Distribution of the total expected repair and replacement cost rate estimated at the 10th quantile for the censored data (10% fails) from MLE and MRR method. The repair cost is set to 1.

when the sample size increase, we observed concentrated cost rates. Figures 3.17 and 3.19 represent the corresponding empirical cdf for 10th and 80th quantiles respectively.

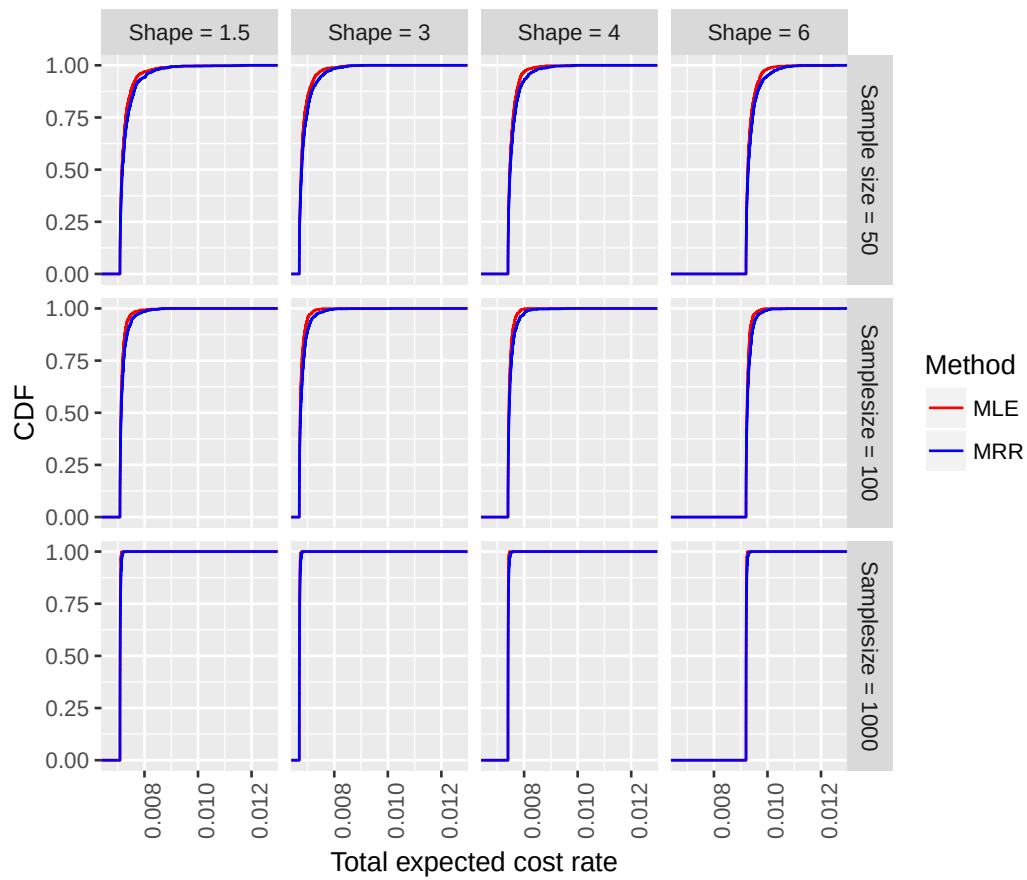


FIGURE 3.17: Empirical density of the total expected cost rate estimated at the 10^{th} quantile from MLE and MRR method. The data are censored with 10% fails. The repair cost is set to 1.

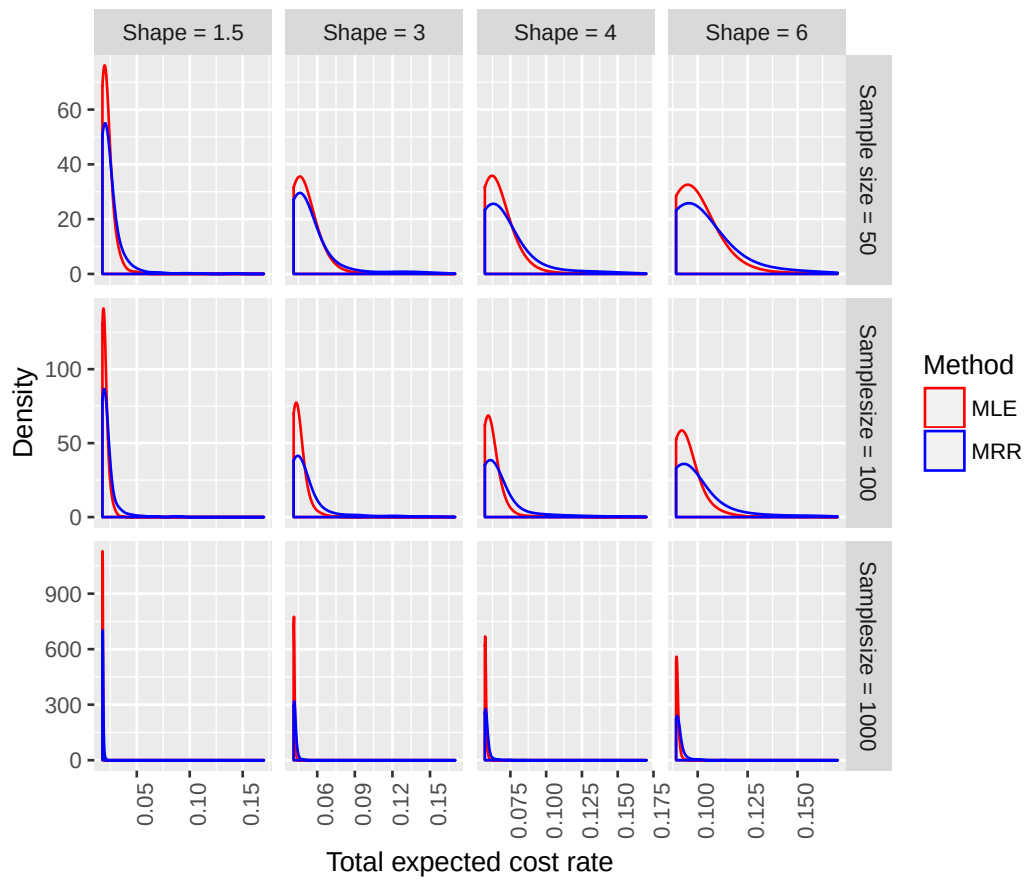


FIGURE 3.18: Distribution of the total expected repair and replacement cost rate estimated at the 80th quantile from MLE and MRR method with censored data (10% fails). The repair cost is set to 1.

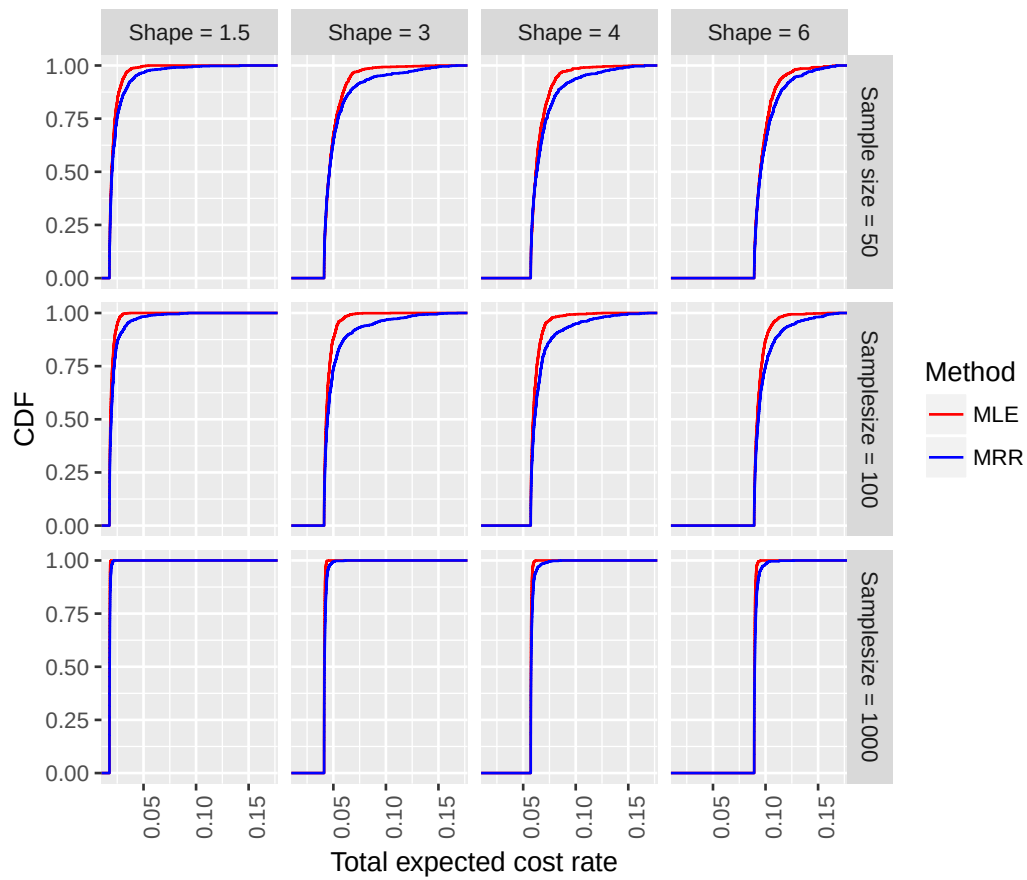


FIGURE 3.19: Empirical cumulative density of the total expected cost rate estimated at the 80th quantile from MLE and MRR method for 10% fails. The repair cost is set to 1.

Section 3.5: Discussion

In this chapter, we have compared the parameter estimation methods maximum likelihood estimation (MLE) and median rank regression (MRR) method in quantile estimation. Furthermore, we computed the repair and replacement cost rates at two quantiles 10^{th} and 80^{th} . We introduced censoring to the data, to observe any impact on the cost estimations. The data were right-censored with Type-II censoring.

The Weibull distribution is used widely in many disciplines to model time-to-fail data. The shape parameter α of the Weibull distribution represents the failure rate of a system. In other words, the value of the shape parameter reflects different failure rates. Specially, the Weibull distribution is commonly used in modelling censored data, due to its well-known ability to explain a variety of lifetime data and its properties (Moala and Penha, 2014).

We used absolute bias and mean squared error (MSE) as the measures of precision in quantiles estimation. Absolute bias and MSE values of the quantile estimates for the complete failures, decrease with increasing sample size. A peak was observed at $\alpha = 3$ for both complete data, and 75% fails. However, with increasing shape parameter, both absolute bias and MSE decreased towards zero. The patterns were broadly the same for the complete data and 75% fails.

On the other hand, when the data are heavy censored, with only 10% fails, both absolute bias and MSE showed comparatively larger values. An unusual higher value of MSE was observed from MRR method at $\alpha = 1.5$. In quantile estimation, MLE outperformed the MRR method, when absolute bias and MSE were considered as measures of precision.

Not surprisingly, the repair and replacement cost rates increase with increasing failure rates. The MLE method produces a more concentrated distribution than the MRR method for all the shape parameter values. We observed higher cost rates for the high degrees of censoring. Therefore, the highly censored data impact on the cost rates adversely.

In summary, MLE method produces more reliable quantile estimates in terms of absolute bias and MSE. Heavy censoring widely affects the robustness of the quantile estimates and repair and replacement cost rates.

4. Bayesian Analysis: Model selection and model averaging?

After a comparison of parameter estimation methods, we turn now to a comparison of failure models that can be used to describe time to failure data. Generally, the Weibull distribution is widespread and commonly used as the failure model in reliability analyses. However, other distributions can provide competitive fits to the experimental data. Alternatively, a combination of models can explain the variations of time to failure data than a single distribution. Considering all these factors, we compare several distributions to select the best fitting model in a given situation. In addition to that, we combine some failure models to derive an average model. We then compare the performance of the average model with the individual models in analysing time-to-failure data.

Firstly, we briefly review the literature of Bayesian analysis and then compare several models via Akaike information criteria (AIC), Bayesian information criteria (BIC) and Deviance information criteria (DIC). Next, the Bayesian model averaging techniques are used to derive an averaged model for both censored data and complete data. We then compute the cost effectiveness of the failure models and compare with the derived average model.

Section 4.1: Background

The Bayesian approach is a technique used to incorporate available prior knowledge or beliefs of the experts from the industry, to the observed data. The Bayesian approach enables the practitioner to introduce additional information to the problem to produce

accurate estimates. By applying Bayesian theory, a posterior distribution is derived for the parameters of the assumed failure model for the observed data. In other words, we infer the model parameters using Bayesian approaches. The development of Markov Chain Monte Carlo (MCMC) methods has made it possible and straightforward to apply the Bayesian theory to many practical problems.

According to the Bayesian point of view, the probabilities are subjective and can be updated with prior knowledge. In Bayesian analysis, parameters are described as densities where frequentists consider parameters as fixed values. Parameters are derived from data and a specific sampling distribution (Murphy, 2012). The prior distribution is used to quantify prior knowledge about the parameter values. Posterior distributions of the parameters are the updated densities of the parameters with data and prior experience. Therefore posterior distribution is a conditional probability distribution of unknown parameters given the observed data. Figure 4.1 illustrates the Bayesian mechanism.

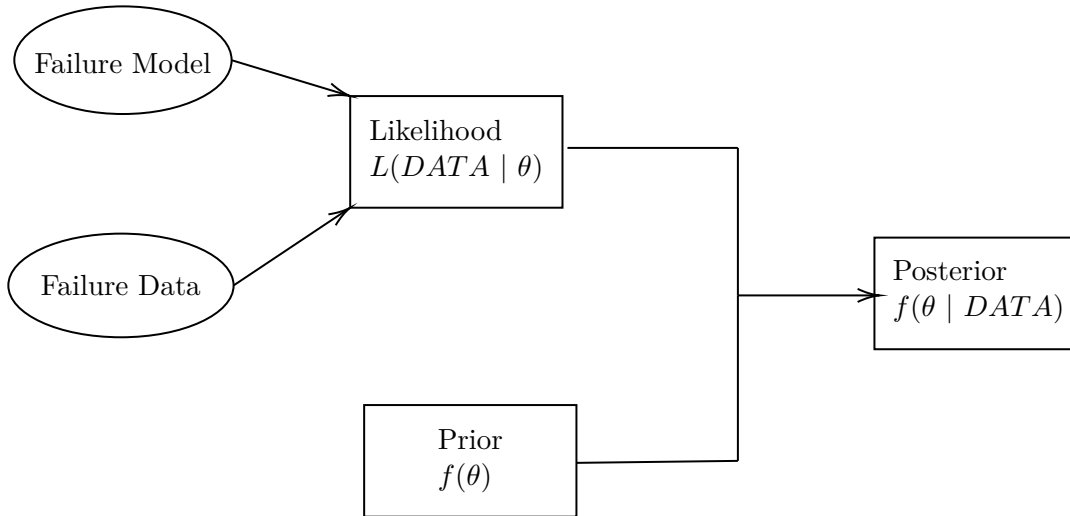


FIGURE 4.1: Bayesian mechanism.

Bayesian inference has been applied in a variety of disciplines, addressing different problems and comparing with the frequentists counterparts. Bayarri and Berger (2004) discussed the Bayesian and frequentists methods of statistical practices along with an explanation of topics in the debate about each paradigm. Achcar, Brookmeyer, and Hunter (1985) provided an illustrative application of Bayesian analysis to some acute leukemia data set. The failure model utilized for the analysis was the log-location scale form of the Weibull distribution. An explanation of likelihood and posterior

distributions from a uniform prior, Jeffreys' prior and conjugate priors were given by Ali, Aslam, and Kazmi (2013). They also discussed the elicitation of hyperparameters for the Lindley distribution and provided a real-life numerical example to support their explanations. Ellison (2004) described an application of the Bayesian paradigm in Ecology.

Hong et al. (2014) discussed Bayesian life test planning for the log-location scale family of distribution. The authors had considered censored data from the Weibull distribution for the illustrative purposes and had quantified the estimation precision of a quantile. Zhang and Meeker (2005) illustrated the Bayesian life test planning for the Weibull distribution with given shape parameter. They had also used type II censored data from the Weibull distribution in estimating any quantile of interest.

Miller (1980) provided Bayesian estimation for two-parameter gamma distribution. The author focused on the conjugate priors and non-informative priors in obtaining the posterior distributions. Antonio Moala, Luiz Ramos, and Alberto Achcar (2013) developed different non-informative priors such as the reference prior, Jeffreys' prior and Tibshirani prior with the copula approach for the two-parameter gamma distribution. A comparison of these non-informative priors had carried out to support the selection of a suitable prior when little is known about the failure mechanism (Antonio Moala, Luiz Ramos, and Alberto Achcar, 2013).

Bayesian approached have adopted not only for complete failures but also for data with a variety of censoring schemes. For example, Ahmed (2014) elucidate Bayesian estimation for progressive type-II censoring and compare that with the MLE. Another comparison of Bayes approach to MLE in estimating the survival function and hazard rate of the Weibull distribution was explained by Mohammed A et al. (2012). They had claimed that the MLE produce more efficient estimates of hazard rate and survival function compared to the corresponding Bayesian estimates when MSE considered as the precision estimator. Awwad, Raqab, and Al-Mudahakha (2015) discussed the parameter estimation of the Weibull distribution from a progressive type-II censored data. For other applications of Bayesian reliability analysis with censored data Storlie et al., 2013; Calabria and Pulcini, 1994; Moala and Penha, 2014; Guure et al., 2015; Kaminskiy and Krivtsov, 2005; Huang and Wu, 2012; Danish and Aslam, 2014; Awwad,

Raqab, and Al-Mudahakha, 2015; Lin and Lio, 2012; Yang et al., 2015; Bracale and De Falco, 2015; Chiodo and Mazzanti, 2006; Maffei, 1999; Achcar, Brookmeyer, and Hunter, 1985, see

There are different loss functions considered in Bayesian estimation. Basu and Ebrahimi (1991) developed Bayes estimators for reliability function and the mean lifetime of exponential distribution with asymmetric loss functions. Overestimation of reliability function or mean life is generally considered riskier than underestimation (Basu and Ebrahimi, 1991). Thus, they have shown asymmetric loss functions are more suitable than the squared error loss functions when estimating the reliability function or the average life.

Some analysts prefer frequentists' methods while some others advocate the Bayesian paradigm as superior. Gelman (2008) discussed the objections to Bayesian analysis in non-Bayesian perspective. Goodwin (2015) explained two substitutions to approximating the Bayesian approach and assessed the performance of the alternatives in producing accurate decisions.

Section 4.2: A single model?

The frequent choice of the failure distribution in reliability analysis or failure data analysis is the Weibull distribution. The Weibull distribution possesses the ability to model different failure rates with the shape parameter of the distribution see Calabria and Pulcini, 1994; Fan and Hsu, 2014; Goodwin, 2015, for example. Although the most popular model is the Weibull, in some situations, analysts have to select the best model out of many possible failure distributions for accurate predictions and estimations. Information criteria such as Bayesian information criteria (BIC), Akaike information criteria (AIC), and deviance information criteria (DIC) can be used for this model selection. These information criteria are applied to pick up a model from many candidates that provide evidence of the best fit. At this point, we do not make decisions but rather find the best functional form explaining the variations of the data.

4.2.1 Model selection via AIC, BIC and DIC

Suppose there are two parameters that describes the likelihood function (i.e. $L(\mu, \sigma)$), then the profile likelihood of a single parameter $P(\mu)$ (or $P(\sigma)$) eliminates the nuisance parameter σ (or μ) from the model and the likelihood is expressed in terms of μ . Thus, the profile likelihood can be defined by,

$$P(\mu) = \sup_{\sigma} L(\mu, \sigma) = L(\mu, \hat{\sigma}) \quad (4.1)$$

where $\hat{\sigma}$ is the ML estimate of σ .

Profile likelihood allows to identify the plausible values of the interested parameter while allowing the user to compute likelihood confidence intervals (Meeker and Escobar, 1998). So, the relative profile likelihood for μ is,

$$RP(\mu) = \max_{\sigma} \left[\frac{L(\mu, \sigma)}{L(\hat{\mu}, \hat{\sigma})} \right] \quad (4.2)$$

Nevertheless, the disadvantage of relative profile likelihood is, it assumes that for all the values of μ , σ takes its maximum value (Bar-Lev, 2004). Meeker and Escobar (1998) provides the $100(1 - \alpha)\%$ confidence interval for μ as the interval over $RP(\mu) > \exp[-\chi^2_{(1-\alpha, 1)}/2]$. Consequently, the values of μ with high relative likelihood are more plausible than values of μ with low relative likelihood values.

We use the Akaike information criteria (AIC), the Bayesian information criteria (BIC) and the deviance information criteria (DIC) to select the best model among the fitted distributions. Assume that we fit two models with parameters (μ, σ) and (α, β) .

The model with the smallest AIC is considered to be the best fit among the aforementioned distributions. AIC is defined as follows.

$$AIC = -2\ln[L(\hat{\theta})] + 2K$$

Here K is the number of estimated parameters.

If the sample size is small (as in our study) or the number of estimated parameters is large then (AIC_c) can be used as an alternative to AIC with a greater reliability.

(AIC_c) is an extension of AIC with a larger penalty for extra parameters. (AIC_c) is defined by,

$$AIC_c = -2\ln[L(\hat{\theta})] + 2K \left(\frac{n}{n - K - 1} \right)$$

This can be re-written as,

$$AIC_c = AIC + \frac{2K(K + 1)}{n - K - 1}$$

Another frequently used method of discriminating models is the Bayesian information criteria (BIC). BIC is defined by the formula,

$$BIC = -2\ln[L(\hat{\theta})] + K\ln(n)$$

Here n is the total number of uncensored observations and K is the number of estimated parameters in the model (Volinsky and Raftery, 2000). Similar to the AIC, model with the smallest BIC value is chosen as the best fit to the data.

The deviance information criteria was proposed by Spiegelhalter et al. (2002).

$$DIC = D(\bar{\theta}) + 2p_D \tag{4.3}$$

$$p_D = \overline{D(\theta)} - D(\bar{\theta}) \tag{4.4}$$

Here, p_D is the effective number of parameters, $D(\hat{\theta})$ is the deviance at the posterior mean $\bar{\theta}$ and $\overline{D(\theta)}$ is the mean of the posterior deviances.

4.2.2 Priors and convergence diagnostics

Parameters of the selected probabilistic distributions are estimated using the Bayesian method. When applying the Bayesian method, prior distributions of the parameters are needed to be assigned. All the prior distributions assumed here are proper probability distributions. Hence the resulting posterior distributions are also proper.

A distribution is called proper, if it satisfies the conditions of a probabilistic distribution. Prior distributions assumed for the candidate models are set to be weakly informative as prior information regarding telephone failures are not available. Consequently, they will not have a strong impact on the posterior distributions. Following is the list of prior distributions chosen in this study. For the Weibull distribution $\mu \sim Uniform(0, 20)$, $\sigma \sim Inverse - gamma(0.1, 0.1)$, for the log-normal distribution $\mu \sim Uniform(0, 20)$, $\sigma \sim Inverse - gamma(0.1, 0.1)$, for the log-logistic distribution $\mu \sim Uniform(0, 20)$, $\sigma \sim Inverse - gamma(0.1, 0.1)$, for the gamma distribution $\alpha \sim Gamma(0.1, 0.1)$, $\beta \sim Gamma(0.1, 0.1)$ and for the inverse-Gaussian distribution $\mu \sim Gamma(0.1, 0.1)$, $\lambda \sim Gamma(0.1, 0.1)$ (see Appendix A for the full model).

Since direct sampling from the posterior distribution is difficult, Markov chain Monte Carlo (MCMC) methods are used. Hamiltonian Monte Carlo (HMC) method (also known as Hybrid Monte Carlo method) is used for obtaining random samples from the posterior distribution. The convergence of the Markov chains is observed with the use of trace plots and Brook- Gelman-Rubin plot to ensure the chain converges to the desired distribution. Markov chains are thinned to reduce the autocorrelation between the random samples. As the first steps of the algorithm can be dependent on the starting values, they are discarded from further evaluations. These discarded points are known as "burn-in" values.

Convergence of the Markov chains was observed using the history plot or the trace plot of the posterior samples. The trace plots showed random variations, more like a horizontal band indicating the convergence of the chains. Autocorrelation function of the posterior samples of the Weibull model were calculated and illustrated on the Figure 4.2. Figure 4.2 shows insignificant autocorrelation values after thinning. This ensures the independence of the posterior samples. Similar patterns were observed for the other candidates and thus, not presented here. Alternatively, Brooks-Gelman-Rubin (BGR) statistic was used to examine the convergence of the chain. BGR statistic can be calculated by,

$$\hat{R} = \frac{m+1}{m} \left(\frac{n-1}{n} + \frac{B}{W} \right) - \frac{n-1}{mn} \quad (4.5)$$

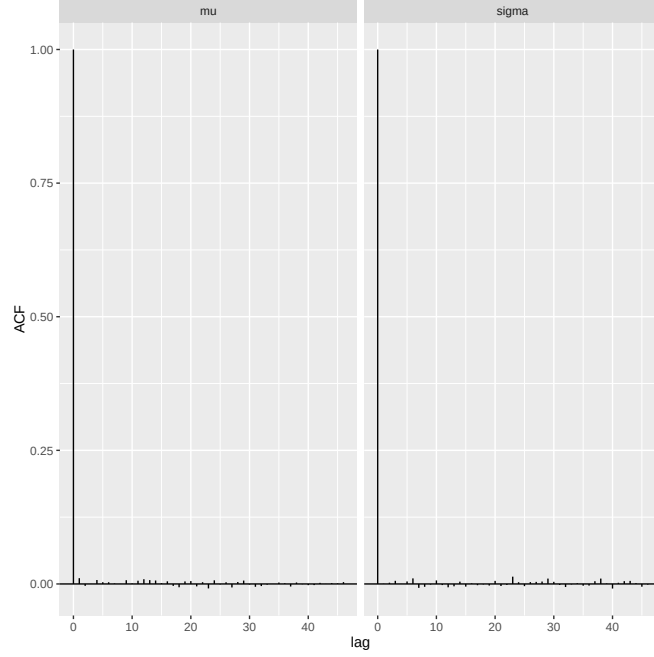


FIGURE 4.2: Plot of autocorrelation function of the Weibull distribution after thinning the posterior samples.

Here W denotes the average within sequence variation of the m number of chains and B represents the average between sequence variance. If the $\hat{R}$ value is closer to 1, it can be concluded that the m chains with n simulations are closer to the target distribution. On the other hand, if the $\hat{R}$ value is large, it suggests that the convergence of the chains can be achieved by running the chains for further number of iterations as the chain has not covered the entire distribution see Brooks and Gelman, 1998; Li and Shi, 2010a.

4.2.3 Numerical results

This section summarises the numerical results from the failure model fitting to the data. We summarise parameter estimates for the complete data and censored data where data are censored at 25% (75% failure) and 75% (25% failure).

Complete Data

Table 4.1 shows the numerical summaries based on the posterior samples for the complete data. Table records the posterior mean, standard deviation (SD), median, first quantile and third quantile of the model parameters.

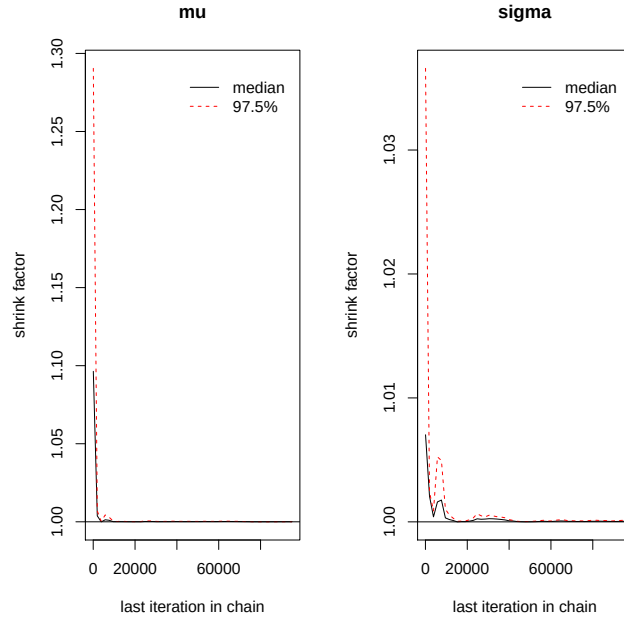


FIGURE 4.3: Brooks-Gelman-Rubin plot of the Weibull model.

TABLE 4.1: Numerical summaries of the model parameters based on the posterior samples for complete data.

Model	Parameter	Mean	SD	Q_1	median	Q_3
Weibull	μ	5.43	0.08	5.37	5.43	5.49
	σ	0.77	0.06	0.72	0.77	0.81
Log-normal	μ	4.99	0.10	4.92	4.99	5.06
	σ	0.96	0.07	0.91	0.96	1.01
Log-logistic	μ	5.06	0.10	4.99	5.06	5.13
	σ	0.54	0.04	0.51	0.54	0.57
Gamma	α	0.01	0.0012	0.01	0.01	0.01
	β	1.52	0.21	1.37	1.51	1.66
Inverse-Gaussian	μ	164.90	16.99	152.98	163.60	175.50
	λ	102.00	15.88	90.84	101.20	112.20

Table 4.2 summarises the estimated goodness of fit values for the candidate distributions for the complete data (Refer Section 4.3.2 and 4.3.3 for explanation of importance sampling and reverse importance sampling). As can be seen in the table, the Weibull distribution has the lowest values indicating the best fit out of the selected. However, the gamma distribution also provides approximate values implying an adequate fit as the Weibull model. This is not surprising as the Weibull distribution is a special case of the generalized gamma distribution (refer to the Section 1.3.2).

It can be observed that the gamma and the Weibull models provide similar fits to the

TABLE 4.2: AIC, BIC, DIC and estimated evidence values in log deviance scale for the complete data.

Model	AIC	BIC	DIC	Evidence (IS)	Evidence (RIS)
Weibull	1112.6	1117.5	1112.6	1126.1	1126.1
Gamma	1113.0	1117.9	1113.0	1125.3	1125.3
Log-normal	1123.7	1128.6	1123.8	1136.9	1136.9
Log-logistic	1123.8	1128.7	1123.8	1136.7	1136.8
Inverse-Gaussian	1136.1	1141.0	1143.5	1205.5	1205.6

telephone failure data. To further support this observation and to find the best fit for the data, the cumulative distributions of the deviance of each candidate model was examined. Suppose we draw K random parameter estimates from the posterior distribution. Then, the random draws were substituted into the likelihood functions to calculate the likelihoods $L_k(\theta_k)$. The deviance for model M_k is $D_k = -2\log L_k(\theta_k)$ see Liu and Aitkin, 2008; Aitkin, Liu, and Chadwick, 2009; Aitkin, Boys, and Chadwick, 2005, for more clarification.

Computed deviance values were sorted to obtain the cumulative distribution of the deviances. Figure 4.4 displays the cumulative distributions of the deviances. Cumulative distributions of the deviances of the gamma and the Weibull distributions bundle together with lower deviances. Inverse-Gaussian distribution shows larger deviance values while the cumulative deviance distributions of the log-normal and the log-logistic distributions also make a cluster. The cumulative deviance distribution selects the Weibull model as the best fit with lower deviance values. However, it is noticeable that the deviances of both the Weibull and the gamma distributions are low compared to the other three models.

Deviance can be sensitive to the selected prior distribution. Therefore, we used Jeffreys' prior to obtain the deviance distributions for the models and compared them to identify any sensitivity toward the selection of the models and BMA. As there are no available prior information, we used the Jeffreys' prior for the comparison. Figure 4.5 shows the results for the Jeffreys' prior. Figure 4.5 illustrates similar patterns, showing lower deviances for the Weibull distribution.

Then we introduced censoring to the problem. We censored the data at 75% fails (25% of the data are censored) and 25% fails (75% of the data are censored) producing

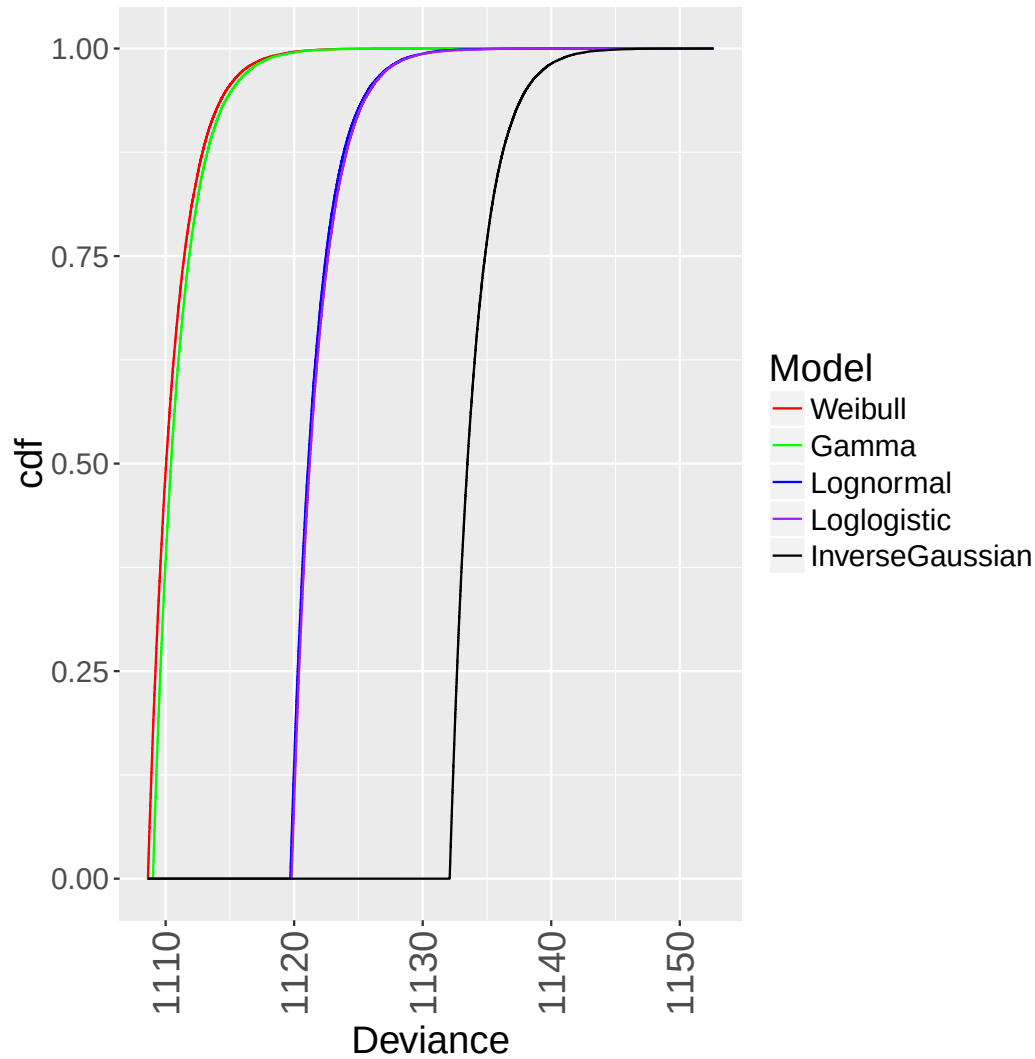


FIGURE 4.4: Cumulative distribution of posterior deviances.

highly censored data for the purpose of comparison. Next two sections summarize the numerical and graphical results for the censored data.

75% Fails

Table 4.3 summarises the parameter estimates with their quantiles for the data, 75% fails.

Similar to the complete data, we compare the candidate failure models for the censored data with AIC, BIC, DIC and the importance sampling. Table 4.4 represents the computed information criteria values for the five models for 25% censored data.

According to the estimated information criteria values presented in Table 4.4, gamma distribution shows the best fit when only 25% of the data are censored. However, it

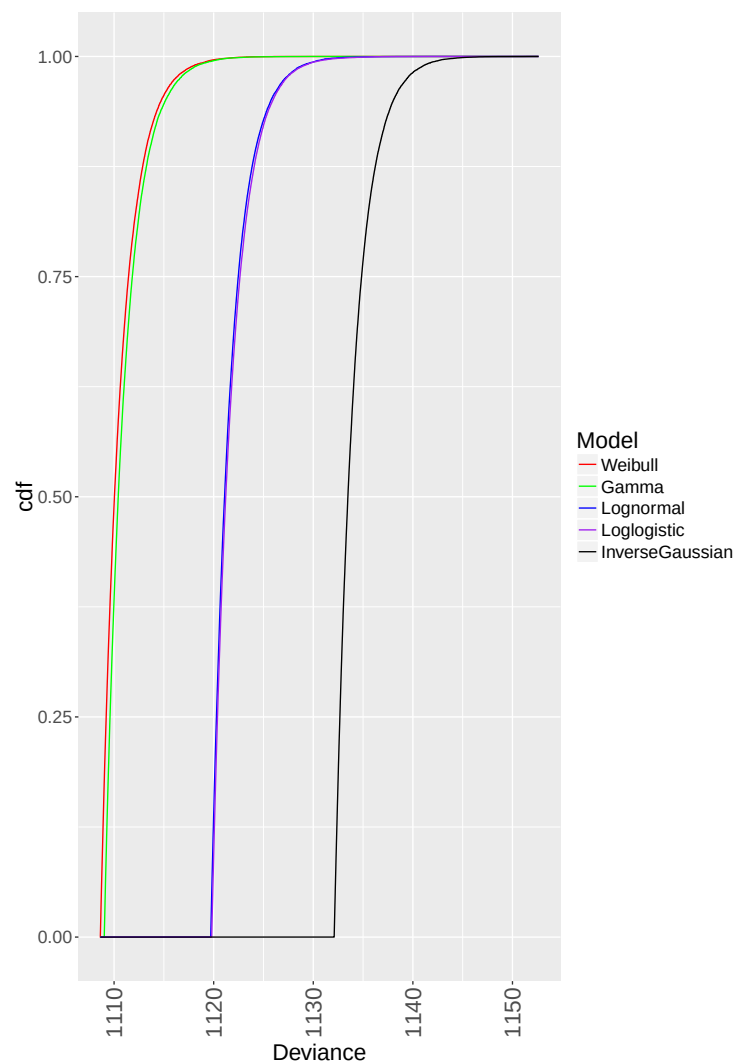


FIGURE 4.5: Cumulative distribution of posterior deviances from the Jeffreys' prior.

TABLE 4.3: Numerical summaries of the model parameters based on the posterior samples for the data censored at 75% of the sample.

Model	Parameter	Mean	SD	Q_1	median	Q_3
Weibull	μ	5.42	0.10	5.36	5.42	5.49
	σ	0.78	0.09	0.72	0.77	0.83
Log-normal	μ	5.06	0.12	4.98	5.06	5.14
	σ	1.08	0.10	1.01	1.07	1.14
Log-logistic	μ	5.09	0.11	5.02	5.09	5.17
	σ	0.61	0.06	0.56	0.60	0.65
Gamma	α	6.9×10^{-3}	0.00	6.1×10^{-3}	6.8×10^{-3}	7.7×10^{-3}
	β	1.48	0.04	1.34	1.47	1.62
Inverse-Gaussian	μ	181.57	144.35	166.36	179.79	194.85
	λ	94.21	66.71	83.51	93.37	104.03

should be noted that the Weibull distribution also shows very closer values. The rest of the models rank as log-logistic, log-normal and Inverse-Gaussian distribution in

TABLE 4.4: AIC, BIC, DIC and estimated evidence values in log deviance scale for the censored data with 75% failures.

Model	AIC	BIC	DIC	Evidence (IS)	Evidence (RIS)
Weibull	847.4	851.8	847.5	860.1	864.6
Gamma	847.4	851.7	847.4	859.2	863.7
Log-normal	850.6	854.9	850.6	863.1	867.6
Log-logistic	849.3	853.6	849.2	861.7	866.2
Inverse-Gaussian	860.4	864.8	870.1	933.6	938.1

descending order of goodness of fit.

25% Fails

Table 4.5 presents the parameter estimates for the censored data with only 25% fails.

TABLE 4.5: Numerical summaries of the model parameters based on the posterior samples censored at 25% of the data sample.

Model	Parameter	Mean	SD	Q_1	median	Q_3
Weibull	μ	5.09	0.20	4.95	5.07	5.21
	σ	0.63	0.13	0.53	0.61	0.70
Log-normal	μ	5.10	0.23	4.93	5.07	5.24
	σ	1.17	0.20	1.02	1.14	1.28
Log-logistic	μ	4.96	0.19	4.83	4.94	5.07
	σ	0.59	0.12	0.50	0.57	0.65
Gamma	α	0.01	0.00	0.01	0.01	0.01
	β	1.84	0.47	1.55	1.84	2.17
Inverse-Gaussian	μ	152.96	22.74	136.78	150.58	166.82
	λ	98.08	16.52	86.45	97.06	108.61

TABLE 4.6: AIC, BIC, DIC and estimated evidence values in log deviance scale for the censored data with 25% failures.

Model	AIC	BIC	DIC	Evidence (IS)	Evidence (RIS)
Weibull	323.8	326.3	323.8	334.8	339.4
Gamma	324.2	326.7	324.3	334.6	339.0
Log-normal	325.5	327.9	325.5	336.4	340.9
Log-logistic	324.5	326.9	324.5	335.4	339.9
Inverse-Gaussian	329.6	331.9	342.2	400.2	404.75

When the data are heavily censored, the goodness of fit values show closer values for the failure models except for the Inverse-Gaussian distribution. According to the *AIC* and *BIC* values, the Weibull distribution provides the best fit. Nevertheless, the other goodness of fit measures support the gamma distribution as the best fit. Apart

from that, the estimated goodness of fit values of both log-logistic and log-normal distributions suggest competing fits when the data are heavily censored.

Nevertheless, there can be situations where a single model does not adequately explain the data or several models provide competitive fits. We experienced this problematic situation for the telephone failure data. Also, when the data are highly censored we experienced approximately similar goodness of fit values for the data for all the models considered, except for the inverse-Gaussian distribution. Bayesian model averaging (BMA) can be used to overcome this problematic situation by accounting for any model uncertainty.

However, many applications of BMA centered on nested model selection in regression analysis, where only a few focuses on non-nested model selection and model averaging. Moreover, the applications are constrained only to the calculation of the posterior model probabilities. The computed model probabilities can be effectively used to arrive at more reliable, precise and robust probabilistic decisions. The BMA allows to account for the model misspecifications (Zhang et al., 2013; Hoeting et al., 1999) and we quote from Hoeting et al. (1999).

"Data analysts typically select a model from some class of models and then proceed as if the selected model had generated the data. This approach ignores the uncertainty in model selection, leading to over-confident inferences and decisions that are riskier than one thinks they are. Bayesian model averaging (BMA) provides a coherent mechanism for accounting for this model uncertainty"

In the following sections, we discuss BMA in probabilistic decision making to address the problems mentioned above considering BMA as a tool for minimizing model uncertainty. Also, the cost-effectiveness of the decisions made discussed with the cost functions introduced in Chapter 2.

Section 4.3: Model averaging

BMA has proposed by many authors. Hoeting et al. (1999) outlined the importance, difficulties, and implementation of BMA with several practical problems. Wasserman (2000) reviewed the Bayesian model selection and model averaging with non-informative

priors. Aitkin, Liu, and Chadwick (2009) discussed the model comparison and model average in small area estimation with an application to lung cancer mortality data. They have used model averaging as a robust method of selecting a model for area effect. Bullock and Boone (2007) have used the BMA tool to overcome the problem of lack of model fitting. They have observed higher goodness-of-fit values from the averaged model over the single candidates through a simulation study for a set of loblolly pine tree diameter data. Zhang et al. (2013) have applied BMA to integrate whole-stand models, distribution models, and individual-tree models in predicting stand basal areas. They have observed more accurate and precise predictions.

An application of BMA in wind speed modelling was discussed by Li and Shi (2010b). The derived average model for wind speed data has applied to several sites in North Dakota. They emphasized the robustness of the results in explaining the long-term wind speed at all the sites over the conventional Weibull model.

Thamrin, McGree, and Mengersen (2013) asserted the robustness of the predictions from BMA with an application to survival data. They also declared that when the sample size was small, an averaged model was suitable when there was no "best" single model. Another application of quasi – Bayesian model averaging for conditional quantile modelling in daily financial returns was conducted by Tsiotas (2015). They have shown that when a single model has considered it created model uncertainty, whereas an averaged model could reduce that. A BMA method for quantile estimation in accelerated life testing has adopted by Yu and Chang (2012), and they evaluate the performance of the s-credibility intervals.

One of the difficulties of applying BMA is the computation of the integral in equation 4.7 (Hoeting et al., 1999). We use importance sampling and reverse importance sampling to evaluate the integral. Next section discusses the importance sampling method and computation of posterior model probabilities.

4.3.1 BMA and posterior model probabilities

Suppose there are $M_1, M_2, M_3, \dots, M_k$ competing models available and the parameter of interest is θ . Then, the posterior probability of the model j can be defined as,

$$pr(M_j|DATA) = \frac{L(DATA|M_j)pr(M_j)}{\sum_{i=1}^k L(DATA|M_i)pr(M_i)} \quad (4.6)$$

where $L(DATA|M_j)$ is the integrated likelihood for a vector of parameters of the model θ_j given by,

$$L(DATA|M_j) = \int pr(DATA|\theta_j, M_j)pr(\theta_j|M_j)d\theta_j \quad (4.7)$$

Therefore, the posterior distribution of the common parameter of interest given the data is,

$$pr(\theta|DATA) = \sum_{j=1}^k pr(\theta|M_j, DATA)pr(M_j|DATA) \quad (4.8)$$

The posterior distribution given in 4.8 can be interpreted as an average of the posterior distributions of the candidate models, weighted by their posterior model probabilities.

Additionally, we transformed the computed AIC, BIC, and DIC values to get the posterior model probabilities for the candidate distributions. The equation 4.9 was used for the transformation.

$$w_i = \frac{\exp\left\{-\frac{1}{2}\Delta_j\right\}}{\sum_{k=1}^K \exp\left\{-\frac{1}{2}\Delta_m\right\}} \quad (4.9)$$

Here, Δ_j is the median deviance, AIC or BIC of the j_{th} model and the lowest median deviance, AIC or BIC obtained for the K candidate models (Liu and Aitkin, 2008).

4.3.2 Importance sampling

Computation of the posterior model probabilities in BMA involves the evaluation of the integrated likelihood. Monte Carlo methods are used to compute the integral in

equation 4.7. The integral in the equation 4.7 is in the form,

$$I = \int_X f(x) dx \quad (4.10)$$

This integral can be expressed in the form,

$$I = \int_X \frac{f(x)}{g(x)} g(x) dx \quad (4.11)$$

Here the function $g(x)$ is normalized;

$$\int_X g(x) dx = 1 \quad (4.12)$$

In the current study, the function $g(x)$ is selected as the multivariate Gaussian distribution. The selection of $g(x)$ as close to as $f(x)$ enhances the rate of convergence of the integral. Gaussian distribution is a good approximation to the function $g(x)$ when the principle of asymptotic normality is considered.

Using the multivariate Gaussian distribution ($g(x)$) N random variates are simulated.

$$x_i \leftarrow g(x)$$

Then the integral is approximated by computing the average,

$$I = \mathbf{E}_g \left(\frac{f(x)}{g(x)} \right) \approx \frac{1}{N} \sum_{i=1}^N \frac{f(x_i)}{g(x_i)} \quad (4.13)$$

This procedure is known as "importance sampling". The effectiveness of the method depends on the accurate selection of the function $g(x)$. The function $g(x)$ should resemble the properties of $f(x)$ for a better approximation of the integral.

4.3.3 Reverse importance sampling

Reverse importance sampling was introduced by Ruanaidh and Fitzgerald (2012). This method is similar to the importance sampling method but the random variates

are drawn from the joint posterior density. These posterior samples are used to approximate the integral by computing the integral of a function of which the normalization constant is known. Consider the integral,

$$I = \int_X g(x) dx \quad (4.14)$$

The function $g(x)$ is chosen in a way that the integral $I = 1$.

Now, consider the evidence,

$$E = \int_X f(x) dx \quad (4.15)$$

And the integral in equation 4.14 can be re-written as,

$$I = \int_X \frac{g(x)}{f(x)} f(x) dx \quad (4.16)$$

Dividing the equation 4.16 by E we get,

$$\frac{I}{E} = \int_X \frac{g(x)}{f(x)} \frac{f(x)}{E} dx \quad (4.17)$$

Since I is known, the inverse of the evidence can be computed by,

$$\frac{I}{E} = \mathbf{E}_{\mathbf{g}} \left(\frac{g(x)}{f(x)} \right) \approx \frac{1}{N} \sum_{i=1}^N \frac{g(x)}{f(x)} \quad (4.18)$$

Again it is important to select $g(x)$ closer to $f(x)$ as possible (Ruanaidh and Fitzgerald, 2012).

4.3.4 Numerical results

Complete data

In this section, we summarize the posterior model probabilities calculated from the importance sampling method and reverse importance sampling method described in the previous section. Also, we report the model weights from the AIC, BIC, and DIC

for the complete data. The telephone failure data defined in chapter 1 are used for the analysis.

Table 4.7 summarizes the posterior model probabilities calculated for the five failure models. The Weibull distribution has the highest posterior probability, which follows by the gamma distribution. Both the Weibull and gamma distributions take about 99% of the posterior model weights. The log-normal distribution and the log-logistic distribution have approximately the same posterior probabilities. The inverse-Gaussian distribution has the lowest model probability.

TABLE 4.7: Posterior model probabilities for the complete data.

Model	AIC	BIC	DIC	IS	RIS
Weibull	0.54	0.54	0.54	0.40	0.40
Gamma	0.44	0.44	0.44	0.58	0.58
Log-normal	0.021	0.021	0.021	0.018	0.018
Log-logistic	0.02	0.02	0.02	0.01	0.01
Inverse-Gaussian	4.3×10^{-6}	4.3×10^{-6}	1.0×10^{-7}	2.2×10^{-18}	2.2×10^{-18}

Then we used the evaluated posterior model probabilities to develop an average model for the complete data. From the developed averaged model, the 80th quantile was computed, and then the repair and replacement cost rate was calculated using the methods established in Chapter 2. The results are illustrated in figure 4.6. All these methods were applied to the data introduced in the Chapter 1.

The Weibull distribution shows the lowest cost rates, and the gamma distribution shows the highest cost rates. Also, the average model shows cost rates slightly higher than the Weibull model. Decision-makers can use the averaged model instead of single models to avoid any overconfident decisions for the complete data.

Next, we evaluate the impact of censoring to the Bayesian model averaging. Similar to the previous analysis for the censored data, we compute the posterior model probabilities for each of the candidate distribution and examine the repair and replacement cost rate at 80th quantile.

75% Fails

Table 4.8 summarises the computed posterior probabilities for each of the model when there are 75% fails.

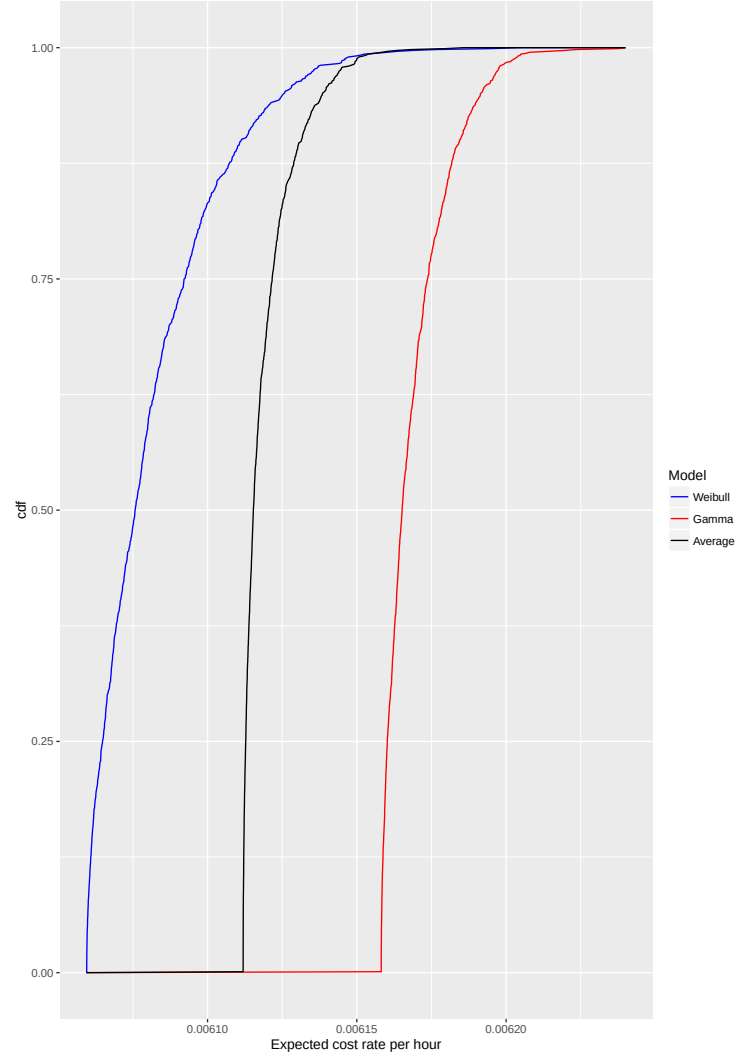


FIGURE 4.6: Empirical cumulative distribution of expected repair and replacement cost rate of the single models (Weibull and gamma) and the averaged model. The shown cost rates are for the optimal replacement at 80th quantile of each distribution and the repair cost was set to \$1 when calculating the cost rates.

TABLE 4.8: Posterior model probabilities for the censored data, 75% fails.

Model	AIC	BIC	DIC	IS	RIS
Weibull	0.37	0.37	0.37	0.30	0.30
Gamma	0.39	0.39	0.39	0.49	0.49
Log-normal	0.08	0.08	0.08	0.07	0.07
Log-logistic	0.15	0.15	0.15	0.14	0.14
Inverse-Gaussian	6×10^{-4}	6×10^{-4}	4.9×10^{-6}	3.4×10^{-17}	3.4×10^{-17}

We observed that for the complete data, the average model was dominated by the Weibull and gamma distributions. But, when the data are censored, both log-normal and log-logistic distributions start playing a noticeable role in the average model

(Table 4.8). This observation further support the use of an average model over a single model when the data are censored.

Then we used the computed posterior model probabilities to derive an average model. Using the methods established in Chapter 2, we computed the expected repair and replacement cost rate for the telephone failure data at the 80th quantile. As the log-normal and log-logistic distributions showed a considerable contribution to the averaged model, we used them in the computation of cost rates. Figure 4.7 depicts the cumulative expected cost rates for the single models (except the inverse-Gaussian distribution) and the averaged model for an optimal replacement at the 80th quantile. We removed the inverse-Gaussian distribution from further analysis as it showed an inconsiderable posterior probability compared to the other models.

Cumulative distributions of log-normal and log-logistic distributions bundle up together giving lower costs. Both the Weibull and gamma distribution showed higher cost rates and similar patterns in the cumulative distribution. As shown on Figure 4.7, if a single model is used in decision making, we might either underestimate or overestimate the expected cost rates in failure analysis.

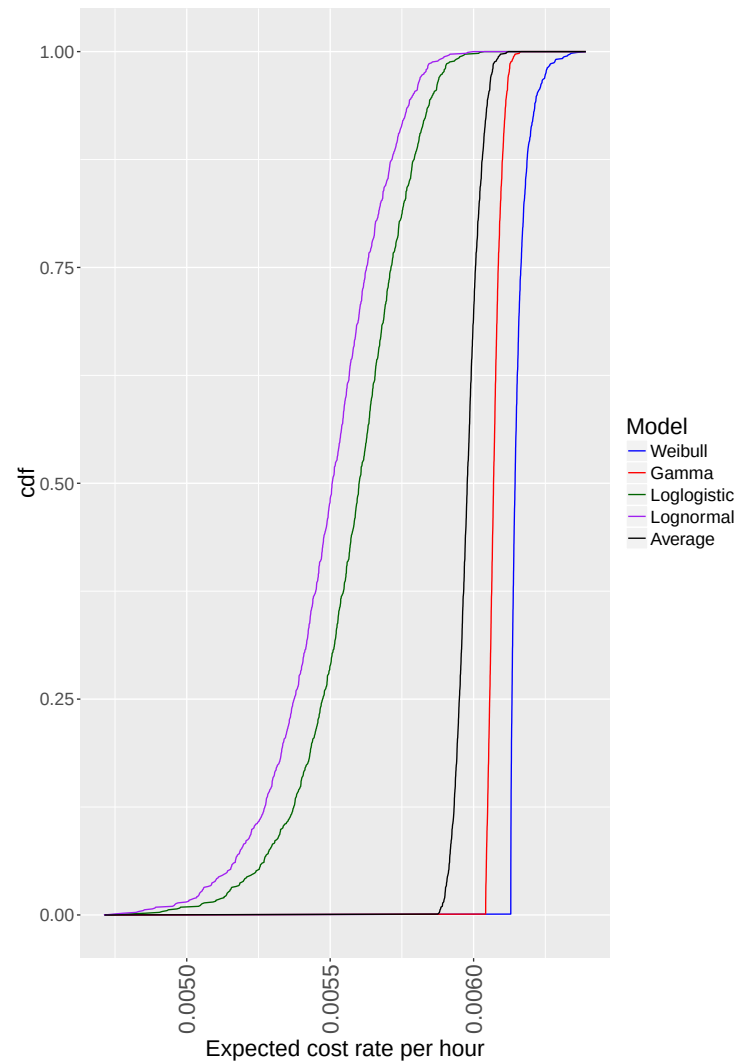


FIGURE 4.7: Empirical cumulative distribution of expected repair and replacement cost rate of the single models and the averaged model. The shown cost rates are for the optimal replacement at 80th quantile of each distribution and the repair cost was set to \$1 when calculating the cost rates, 75% fails.

25% Fails

We introduced heavy censoring to the data by allowing only 25% fails and calculated the posterior model probabilities for the five candidate distributions. Table 4.9 summarises the posterior model weights for the heavily censored data.

TABLE 4.9: Posterior model probabilities for the censored data, 25% fails.

Model	AIC	BIC	DIC	IS	RIS
Weibull	0.33	0.33	0.34	0.29	0.29
Gamma	0.27	0.27	0.28	0.34	0.35
Log-normal	0.14	0.14	0.14	0.13	0.14
Log-logistic	0.24	0.24	0.23	0.22	0.22
Inverse-Gaussian	0.02	0.02	3.5×10^{-5}	1.9×10^{-15}	1.9×10^{-15}

Influence of the log-logistic and log-normal distributions to the average model increases as the data are heavily censored. We used these model weights to in repair and replacement cost calculations and the results are displayed in Figure 4.8.

Cumulative cost distributions of the four models show similar patterns and have closer values. Cumulative distribution of the averaged model lies closer to gamma model. However, all the four single models have considerably and approximately equally contributed the average model except for the log-normal model. Therefore, when the data are heavily censored, uncertainty of the model selection is high and the use of a single model can produce biased results.

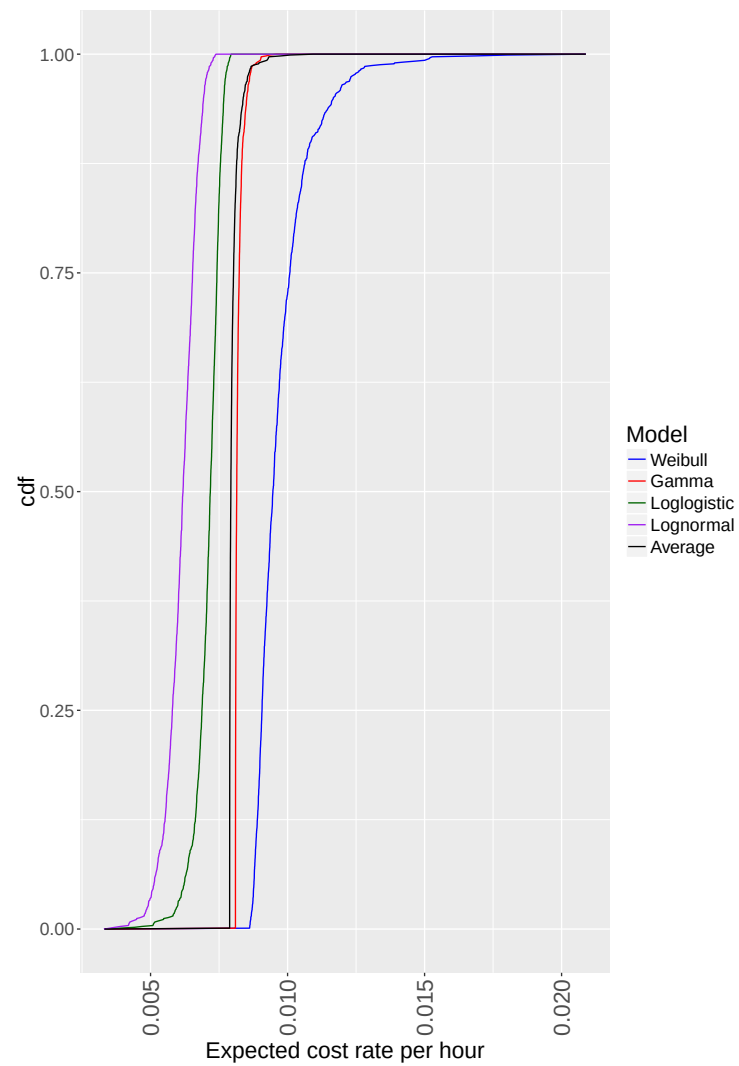


FIGURE 4.8: Empirical cumulative distribution of expected repair and replacement cost rate of the single models and the averaged model. The shown cost rates are for the optimal replacement at 80^{th} quantile of each distribution and the repair cost was set to \$1 when calculating the cost rates, 25% fails.

Section 4.4: Discussion

In Bayesian analysis, we integrate information that arises from empirical studies with the present data (or the likelihood) to obtain more accurate estimates. This approach is vital when we have to solve problems, which need probabilistic decisions, and there is only little experimental data available (Calabria and Pulcini, 1994).

In this chapter, we considered the problem of failure model selection for both complete and censored data. Akaike information criteria (AIC), Bayesian information criteria (BIC), and deviance information criteria (DIC) were used to selecting the best model out of seven candidates. The two statistics, AIC and BIC, support the principle of parsimony. But the assumptions are different for the two methods. BIC assumes the correct model is included in the candidate models, but AIC does not assume that the actual model exactly included in the candidate models. According to all the three criteria, the Weibull model was the best among the candidates. However, the gamma distribution produced goodness of fit values closer to the Weibull model for the complete data. Also, when the data are heavily censored four failure models showed competing fits. Those observation led us to the idea of model averaging to account for any uncertainty in model selection.

Through this chapter, then we discussed the importance of implementing BMA to account for model misspecification when the data are censored. BMA method resolves the problem of selecting a single model for a data set by averaging over candidate models. It elevates the precision and accuracy of estimated quantities and accounts for the model uncertainty. We examined the practical use of the BMA method with a real data set. One argument put forward against the BMA is the complexity of the interpretation of the results. Hoeting et al. (1999) suggested some ways to simplify the explanations. They recommended to either focus on the posterior model effects as we have illustrated in section 4.3.4 or to consider a single model when summarising the results. Moreover, if the practitioners are focused on reducing any imprecision that can be caused by the estimation of the parameters and model misspecification, robust Bayesian approaches and BMA can be employed effectively to achieve the expected levels of reliability when either the data are censored or not. Then the quality of the

estimates increases and hence we can choose the best decisions.

Finally, we calculated the repair and replacement cost according to the average models and the contributing failure models. To avoid any overconfident decisions, the average model can be used instead of the single models especially when the data are heavily censored.

5. Jeffreys' prior for coefficient of variation

This chapter introduces a new Jeffreys' prior to a function of parameters of the Weibull distribution. We consider the log location-scale form of the Weibull distribution and assume a condition where the parameters, location, and scale are dependent by a function of the form *scale/location*. A Bayesian framework developed for a comparison of relative variability can be used to produce more precise decisions when little experimental data are available.

We have already discussed in Chapter 4, the use of Bayesian approaches in integrating prior information with available data. There we showed how to combine several possible failure models to get an average model for both informative and non-informative priors. However, in the previous chapters, we defined priors for only the individual parameters. Motivated by the factor that the functions of the parameters can be of interest, we develop a Jeffreys' prior to obtaining the posterior distribution of the relative variability.

We will first review the literature of the approach of Jeffreys' prior and functional forms of parameters discussed in previous studies. Then we derive the new Jeffreys' prior followed by a simulation study to assess its performance. Finally, we illustrate the use of the prior with a real data set.

Section 5.1: Background

In Bayesian probability theory, the Jeffreys' prior has used extensively as a non-informative or an objective prior for a parameter space. The prior was named after

Sir Harold Jeffreys (Jeffreys, 1939) and is proportional to the square root of the determinant of the Fisher information matrix;

$$\pi(\theta) \propto \sqrt{\det I(\theta)} \quad (5.1)$$

The Fisher information matrix quantifies the amount of information carried by a random variable; say T about an unknown parameter θ . Mathematically, the Fisher information matrix is the variance of the score or the expected value of the observed information. The Jeffreys' prior is well-known for its invariant property. On the other hand, the Jeffreys' prior can produce improper posterior distributions.

There are many discussions and comparisons available for the Jeffreys' prior to other non-informative priors and to the ML method. Sun (1997) compared the performance of the Jeffreys prior of the Weibull distribution to the reference prior. The reference prior was first developed by Bernardo (1979) and then extended by Berger and Bernardo (1992). When deriving the reference prior, the goal is to develop a prior that maximizes the expected posterior information about the parameters provided by the experimental data relative to the information in the prior. Moala, Achcar, and Tomazella (2012) defined the reference prior as the Kullback-Leibler distance between the posterior distribution and the prior distribution. The reference prior differs from the Jeffreys' by the way that it treats the parameters. If there is a nuisance parameter presence in the problem, the reference prior uses ordered parametrization and single out the parameter of interest. For small sample sizes, his (Sun, 1997) investigation provided evidence supporting the reference prior over the Jeffreys' prior. Kim, Lee, and Kang (2014) looked at the reference prior and the Jeffreys' prior for the inverse Weibull distribution where the shape and the scale were the parameters of interest. With the use of a simulation study, they also showed, the Jeffreys' prior is not a second-order matching prior to either of the parameters using the property of frequentists coverage.

Although most commonly, the priors are assigned to the individual parameters of a distribution, there are some scenarios where functions of the parameters are of interest. For example, estimations of the reliability function or the hazard rate of the Weibull

distribution are of greater interest in failure analysis. Sinha (1986) demonstrated the Bayes estimates of the reliability function and the hazard rate using the Jeffreys' prior and compared with the MLE. Moala and Penha (2014) suggested the use of the Jeffreys prior to estimating the reliability function due to its invariance property under the one-to-one transformation of the parameters. The authors compared the Jeffreys' prior and the reference prior in integrating the prior information of the reliability function for complete data. Some non-informative priors including Jeffreys' prior, reference prior, and Tibshirani's prior in yielding the posterior density of the reliability function of the Weibull distribution was compared by Moala, Rodrigues, and Tomazella (2009).

In this chapter, we consider the log location-scale form of the Weibull distribution with the location parameters μ , and the scale parameter σ . We assume the parameters of the Weibull distribution are dependent by a function of the form;

$$\rho = \frac{\sigma}{\mu} \quad (5.2)$$

The new parameter ρ is the coefficient of variation which can use as an efficient measure to compare the variability. The coefficient of variation is the dispersion relative to the mean. Several disciplines such as sport sciences, wind resource assessment, population studies, quality control, agriculture, etc. have considered the coefficient of variation as a measure of variability instead of the ordinary standard deviation (see Snedecor and Cochran, 1989; Diamandis and Christopoulos, 1996; Simpson, Roe, and Lewontin, 2003). Additionally, Bedeian and Mossholder (2000) reviewed the uses of the coefficient of variation to quantify the diversity in workgroups. Krebs (1989) used the coefficient of variation in Ecology stating that it is the most convenient method of comparing the relative variability. Zyl and Merwe (2017) discussed a Bayesian approach to compute the control limits for the coefficient of variation when the data normally distributed. They derived the reference prior and the probability matching prior for the coefficient of variation for the pooled samples. Moreover, they explained about the usefulness and other unique properties of the coefficient variation through a Bayesian simulation with an application to a set of medical data. A recent application to evaluate the variations of wind energy by Lee, Fields, and Lundquist (2018) discussed the advantages of using

a robust coefficient of variation function over the general variability measures. The methodological and interpretive misuses of the coefficient of variation were discussed by Sørensen (2002) in demography.

The main advantage of using the coefficient of variation (CoV) instead of standard deviation is, CoV has no units. Thus, CoV allows easy comparison of different systems. On the other hand, since $\sigma = 1/\beta$ (β is the shape parameter of the Weibull distribution), the measure can use for risk assessment.

Computations, applications, and interpretation of the CoV are frequent and convenient for the normal distribution. In contrast, the use of the CoV of other log location-scale distributions such as the Weibull or the lognormal is infrequent. It should be noted that a measure of CoV can be useful in areas such as failure analysis and wind speed assessment, as the Weibull distribution is employed commonly to model the data. If the variability of wind speeds from different locations needed to be compared, then a measure of relative variation can provide comprehensive interpretations than a general dispersion measure.

In this chapter, we reparameterize the Weibull distribution and introduce a new Jeffreys' prior to the CoV (ρ) for the Weibull distribution. The results are compared with the MLE and a weakly informative prior via a Monte Carlo simulation study to assess the performance of the new Jeffreys' prior. Here we explore the performance of the Jeffreys' prior in the context of log location-scale models. Also, the usage of the new Jeffreys' prior is illustrated with an application to wind speed data.

Section 5.2: Jeffreys' Prior for the Coefficient of Variation

In this section, Jeffreys prior is derived for the location and CoV of the Weibull distribution.

Suppose X is the lifetime of a component in a fleet following a Weibull distribution. The density function of the Weibull distribution is given by,

$$F(x; \mu, \sigma) = \Phi_{sev} \left(\frac{\log x - \mu}{\sigma} \right) \quad (5.3)$$

Here, $\Phi_{sev}(z) = 1 - \exp(-\exp(z))$ and z follows the standard smallest extreme value (SEV) distribution with $\mu = 0$ and $\sigma = 1$.

The probability density function of the Weibull distribution is given by,

$$f(x; \mu, \sigma) = \frac{1}{\sigma x} \phi_{sev}\left(\frac{\log x - \mu}{\sigma}\right) \quad (5.4)$$

Here, $\phi(z) = \exp(z - \exp(z))$ and z follows the standard SEV distribution with $\mu = 0$ and $\sigma = 1$.

Suppose the location parameter μ and the scale parameter σ of the Weibull distribution are dependent by a function of the form,

$$\rho = \frac{\sigma}{\mu} \quad (5.5)$$

Estimation of the location parameter and the relative variability computed as in equation 5.5 is discussed under Bayesian analysis. Next section discusses the derivation of the Jeffreys prior for the relative variability (ρ).

To derive the Fisher's information matrix for the Weibull data, we transform the parameters (μ, σ) to (θ, ρ) , where $\theta = \mu$ and $\rho = \frac{\sigma}{\mu}$. Thus, $\sigma = \rho\theta$.

The probability density function and the likelihood function in these new parameters are given by equations 5.6 and 5.7, respectively.

$$f(x; \theta, \rho) = \frac{1}{\theta \rho x} \exp \left\{ \left(\frac{\log x - \theta}{\theta \rho} \right) - \exp \left(\frac{-\theta}{\theta \rho} \right) \right\} \quad (5.6)$$

$$L(\theta, \rho | x) = \frac{1}{(\theta \rho)^n} \prod_{i=1}^n \frac{1}{x_i} \exp \left\{ \left(\frac{i - \theta}{\theta \rho} \right) - \exp \left(\frac{i - \theta}{\theta \rho} \right) \right\} \quad (5.7)$$

Here $x > 0$, $\theta > 0$ and $\rho > 0$

In order to develop the Jeffreys prior for the parameters (θ, ρ) , we consider the expected values of the following partial derivatives of the log density function.

$$\Lambda_{ij} = \frac{\partial^i \Lambda}{\partial \theta^i \partial \rho^j} \quad (5.8)$$

Here, $i, j = 1, 2$.

Let,

$$\Lambda = \log f(x; \theta, \rho) = -\log \theta - \log \rho - \exp\left(\frac{1-\theta}{\theta\rho}\right) \quad (5.9)$$

We derived the partial derivatives and hence the Fisher's information matrix as follows (Sun, 1997; Penha, 2014).

$$\Lambda_{10} = -\frac{1}{\theta} - \frac{1}{\theta\rho} - \frac{(1-\theta)}{\theta^2\rho} + \frac{(1-\theta)}{\theta^2\rho} \exp\left(\frac{1-\theta}{\theta\rho}\right) + \frac{1}{\theta\rho} \exp\left(\frac{1-\theta}{\theta\rho}\right) \quad (5.10)$$

$$\Lambda_{01} = -\frac{1}{\rho} - \frac{(1-\theta)}{\theta\rho^2} + \frac{(1-\theta)}{\theta\rho^2} \exp\left(\frac{1-\theta}{\theta\rho}\right) \quad (5.11)$$

$$\begin{aligned} \Lambda_{20} = & \frac{1}{\theta^2} + \frac{2}{\theta^2\rho} + \frac{2}{\theta^2} \left(\frac{1-\theta}{\theta\rho}\right) - \frac{1}{\theta^2\rho} \left(\frac{1-\theta}{\theta\rho}\right) \exp\left(\frac{1-\theta}{\theta\rho}\right) - \frac{1}{\theta^2} \left(\frac{1-\theta}{\theta\rho}\right)^2 \exp\left(\frac{\log x_1 - \theta}{\theta\rho}\right) \\ & - \frac{1}{\theta^2\rho} \exp\left(\frac{1-\theta}{\theta\rho}\right) - \frac{2}{\theta^2} \left(\frac{1-\theta}{\theta\rho}\right) \exp\left(\frac{1-\theta}{\theta\rho}\right) - \frac{1}{\theta^2\rho} \exp\left(\frac{1-\theta}{\theta\rho}\right) \\ & - \frac{1}{\theta^2\rho} \left(\frac{1-\theta}{\theta\rho}\right) \exp\left(\frac{1-\theta}{\theta\rho}\right) - \frac{1}{\theta^2\rho^2} \exp\left(\frac{1-\theta}{\theta\rho}\right) \end{aligned} \quad (5.12)$$

$$\Lambda_{02} = \frac{1}{\rho^2} - \frac{2}{\rho^2} \left(\frac{1-\theta}{\theta\rho}\right) + \frac{1}{\rho^2} \left(\frac{1-\theta}{\theta\rho}\right)^2 \exp\left(\frac{1-\theta}{\theta\rho}\right) - \frac{2}{\rho^2} \left(\frac{1-\theta}{\theta\rho}\right) \exp\left(\frac{1-\theta}{\theta\rho}\right) \quad (5.13)$$

$$\begin{aligned} \Lambda_{11} = & \frac{1}{\theta\rho^2} + \frac{1}{\theta\rho} \left(\frac{1-\theta}{\theta\rho}\right) - \frac{1}{\theta\rho} \left(\frac{1-\theta}{\theta\rho}\right)^2 \exp\left(\frac{1-\theta}{\theta\rho}\right) - \frac{1}{\theta\rho} \left(\frac{1-\theta}{\theta\rho}\right) \exp\left(\frac{1-\theta}{\theta\rho}\right) \\ & - \frac{1}{\theta\rho^2} \left(\frac{1-\theta}{\theta\rho}\right) \exp\left(\frac{1-\theta}{\theta\rho}\right) - \frac{1}{\theta\rho^2} \exp\left(\frac{1-\theta}{\theta\rho}\right) \end{aligned} \quad (5.14)$$

Suppose $Y = \exp\left(\frac{1-\theta}{\theta\rho}\right)$. We define $\gamma_i = \int_0^\infty [\log(y)]^i e^{-y} dy$. Here γ_i is the i^{th} moment of $\log(y)$ and it can be shown that Y follows an exponential distribution with mean 1 (Sun, 1997) (see Appendix B).

Using integration by parts,

$$\begin{aligned}\mathbf{E}\left[\left(\frac{1-\theta}{\theta\rho}\right)\exp\left(\frac{1-\theta}{\theta\rho}\right)\right] &= \int_0^\infty y \log(y) e^{-y} dy \\ &= \int_0^\infty [1 + \log(y)] e^{-y} dy \\ &= 1 + \gamma_1\end{aligned}\tag{5.15}$$

$$\begin{aligned}\mathbf{E}\left[\left(\frac{1-\theta}{\theta\rho}\right)^2 \exp\left(\frac{1-\theta}{\theta\rho}\right)\right] &= \int_0^\infty y [\log(y)^2] \exp^{-y} dy \\ &= \gamma_2 + 2\gamma_1\end{aligned}\tag{5.16}$$

Therefore the expected values of the partial derivatives can be written as,

$$\mathbf{E}[\Lambda_{20}] = -\left[\frac{1}{\theta^2} + \frac{1}{\theta^2\rho^2} + \frac{2(1+\gamma_1)}{\theta^2\rho} + \frac{(\gamma_2+2\gamma_1)}{\theta^2}\right]$$

$$\mathbf{E}[\Lambda_{02}] = \frac{-(1+\gamma_2+2\gamma_1)}{\rho^2}$$

$$E[\Lambda_{11}] = -\left[\frac{1}{\theta\rho} + \frac{(\gamma_2+2\gamma_1)}{\theta\rho} + \frac{(1+\gamma_1)}{\theta\rho^2}\right]$$

Therefore the Fisher's information matrix for the Weibull parameters is,

$$I(\theta, \rho) = \begin{bmatrix} -\left(\frac{1}{\theta^2} + \frac{1}{\theta^2\rho^2} + \frac{2(1+\gamma_1)}{\theta^2\rho} + \frac{(\gamma_2+2\gamma_1)}{\theta^2}\right) & -\left(\frac{1}{\theta\rho} + \frac{(\gamma_2+2\gamma_1)}{\theta\rho} + \frac{(1+\gamma_1)}{\theta\rho^2}\right) \\ -\left(\frac{1}{\theta\rho} + \frac{(\gamma_2+2\gamma_1)}{\theta\rho} + \frac{(1+\gamma_1)}{\theta\rho^2}\right) & \frac{-(1+\gamma_2+2\gamma_1)}{\rho^2} \end{bmatrix}\tag{5.17}$$

It is known that the Jeffreys prior $[\pi(\theta, \rho)]$ is proportional to the square root of the determinant of the Fisher's information matrix; which is,

$$\pi(\theta, \rho) \propto \sqrt{\det I(\theta, \rho)}\tag{5.18}$$

Therefore, the Jeffreys prior for the parameters θ and ρ^2 is (see Appendix B),

$$\pi(\theta, \rho) = \frac{1}{\theta \rho^2} \quad (5.19)$$

Here the prior is given for ρ^2 . The Jeffreys prior for ρ is derived as follows.

Suppose we use the following reparameterization.

$$\theta = p(\phi) \quad (5.20)$$

Then, the prior distribution of θ is given by,

$$p(\theta) = p(\phi) \left[\frac{\partial \phi}{\partial \theta} \right] \quad (5.21)$$

Using this relationship, define $\phi = \mu \rho^2$, $\theta = \rho$ and $p(\phi) = \frac{1}{\mu \rho^2}$.

$$\begin{aligned} p(\rho) &= \frac{1}{\mu \rho^2} \times \left(\frac{\partial \mu \rho^2}{\partial \rho} \right) \\ &\propto \frac{1}{\rho} \end{aligned} \quad (5.22)$$

When the two parameters μ and σ are independent, the Jeffreys prior for the parameters is,

$$\pi(\mu, \sigma) = \frac{1}{\sigma} \quad (5.23)$$

Section 5.3: Numerical results

In this section, we evaluate the performance of the Jeffreys' prior derived in the previous section. A simulation study designed to compare the parameter estimates with the results from the MLE and weak informative prior for the parameters of the Weibull distribution. A case study with an application to a set of real data is followed by the simulation.

5.3.1 Simulation study - Complete data

A single sample tells only a little about the data. In this section, we conducted a simulation study to compare the performance of the new Jeffreys' prior to the MLE and weakly informative prior for complete data. The behavior of the proposed prior was evaluated under different sample sizes and different ρ values. We fixed the location parameter of the Weibull distribution to one, as the results were widely similar for different values of the location. We computed the Bayes estimates using the MCMC sampling and then computed the relative bias and the mean squared error as measures of precision.

Monte Carlo estimates = Average of the N estimates

$$\text{Relative Bias} = \frac{(\hat{\theta} - \theta) * 100\%}{\theta}$$

$$\text{Mean Squared Error (MSE)} = \mathbb{E}[(\hat{\theta} - \theta)^2]$$

The simulation was designed with the following factors.

- Location = 1
- Five different rho (ρ) values (0.25, 0.50, 0.75, 1.00, 1.25)
- Six different sample sizes (5, 10, 25, 50, 100, 1000)
- 1000 replicates
- 100% fails

For all the combination of the factors, we computed the MLE and the BE with the new Jeffreys' prior and weakly informative prior introduced in Chapter 4 for the Weibull distribution (see Appendix B for the prior distributions) using MCMC methods. We report the average Bayes estimates, maximum likelihood estimates, relative bias, and MSE. For the interpretation convenience, the results are presented separately for the small sample sizes (5, 10 and 25) and the large sample sizes (50, 100 and 1000).

Table 5.1 illustrates the ML and Bayesian estimates of the location and the coefficient of variation along with their true values considered in the simulation study. The numbers of units considered in table 5.1 are 5, 10 and 25. Table 5.2 summarizes the

estimated ML and Bayesian quantities with their true values for the larger sample sizes. As expected, the MLE provides closer estimations to the true value than the Bayesian estimates when the sample size increases. Based on the estimated quantities, the measures of precision; MSE, and relative bias were computed.

TABLE 5.1: MLE and Bayesian estimates of the relative variation and the location of the Weibull distribution. The estimates are the average of 1000 simulations and the true values of the parameters considered are provided for comparison. Note that the location of the Weibull distribution is fixed (equals to 1) and the results are summarised for the sample sizes 5, 10 and 25.

Sample size	True Location	True Rho	MLE		BE		Weakly Informative Prior	
			Location	Rho	Location	Rho	Location	Rho
5	1	0.25	0.97	0.22	0.99	0.32	0.98	0.26
10	1	0.25	0.98	0.23	0.99	0.27	0.99	0.26
25	1	0.25	0.99	0.24	0.99	0.26	0.99	0.25
5	1	0.50	0.96	0.51	1.02	0.76	0.99	0.52
10	1	0.50	0.98	0.48	0.99	0.60	0.99	0.51
25	1	0.50	0.98	0.49	0.99	0.53	0.99	0.50
5	1	0.75	0.95	1.17	1.06	1.27	0.99	0.76
10	1	0.75	0.97	0.90	1.01	1.06	0.99	0.75
25	1	0.75	0.99	0.76	0.99	0.84	0.99	0.76
5	1	1.00	0.93	2.90	1.13	1.74	1.06	1.00
10	1	1.00	0.96	1.41	1.02	1.50	0.99	0.99
25	1	1.00	0.98	1.04	0.99	1.18	0.98	1.01
5	1	1.25	0.92	4.71	1.21	2.08	1.11	1.25
10	1	1.25	0.97	2.19	1.07	1.88	1.01	1.24
25	1	1.25	0.98	1.43	1.01	1.61	0.99	1.26

Figure 5.1 represents the simulation results for the bias of the relative variability, ρ of the Weibull distribution for both Bayesian and ML methods. The results are the mean of 1000 simulations. The relative biases of the relative variation for the Jeffreys' prior are positive and follow approximately the same pattern, irrespective of the sample size and the scale. In contrast, the biases for the ML start with a negative value and then change the direction when the relative variability, ρ is about 0.5. The relative biases for ML increase noticeably with increasing ρ when the number of units is 5, where the biases for Bayesian estimates are comparatively low. While the number of components increases, the relative biases decrease for both ML and Jeffreys' prior. Relative biases from weakly informative prior show approximately similar values to the Jeffreys' prior when the ρ values are lesser than 0.5. As expected, when the prior distributions are weakly informative relative biases decrease towards zero with increasing ρ values.

Figure 5.2 shows the MSE of the relative variability, ρ from the simulation for the small sample sizes. Resulting MSE values for the Bayesian method are closer to

TABLE 5.2: ML and Bayesian estimates of the location and relative variation of the Weibull distribution when the number of units is 50, 100, and 1000. The estimates are the average of 1000 simulations. Note that the location parameter is set to 1. The true parameter values considered in the simulation are given for the purpose of comparison.

Sample size	True Location	True Rho	MLE		BE		Weakly Informative Prior	
			Location	Rho	Location	Rho	Location	Rho
50	1	0.25	0.99	0.24	0.99	0.25	0.99	0.25
100	1	0.25	0.99	0.24	0.99	0.25	0.99	0.25
1000	1	0.25	0.99	0.25	0.99	0.25	0.99	0.25
50	1	0.50	0.99	0.50	0.99	0.52	1.00	0.50
100	1	0.50	0.99	0.49	0.99	0.50	0.99	0.49
1000	1	0.50	0.99	0.49	0.99	0.50	0.99	0.49
50	1	0.75	0.99	0.75	0.99	0.78	0.99	0.75
100	1	0.75	0.99	0.75	1.00	0.76	0.99	0.75
1000	1	0.75	1.00	0.74	1.00	0.75	1.00	0.75
50	1	1.00	0.99	1.02	0.99	1.08	0.99	1.00
100	1	1.00	0.99	1.01	0.99	1.04	0.99	0.99
1000	1	1.00	0.99	1.00	1.00	1.00	1.00	1.00
50	1	1.25	0.98	1.31	0.99	1.42	1.00	1.25
100	1	1.25	0.99	1.26	0.99	1.31	1.00	1.25
1000	1	1.25	1.00	1.25	1.00	1.25	1.00	1.24

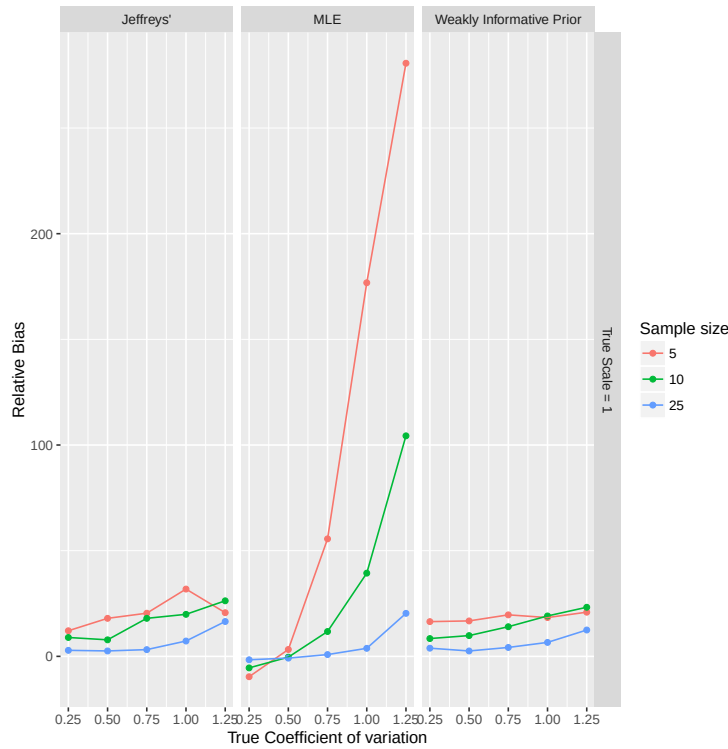


FIGURE 5.1: Relative bias % of the estimate of the relative variation ρ . Each point on the graph is the average of $(\hat{\rho} - \rho)/\rho$ from 1000 simulations. The results are displayed for the small sample sizes 5, 10 and 25. Note that the location of the Weibull distribution is fixed and equal to 1.

zero but increase slightly with increasing ρ values. On the other hand, considerable increments and notably higher values of MSE can be observed with increasing ρ for

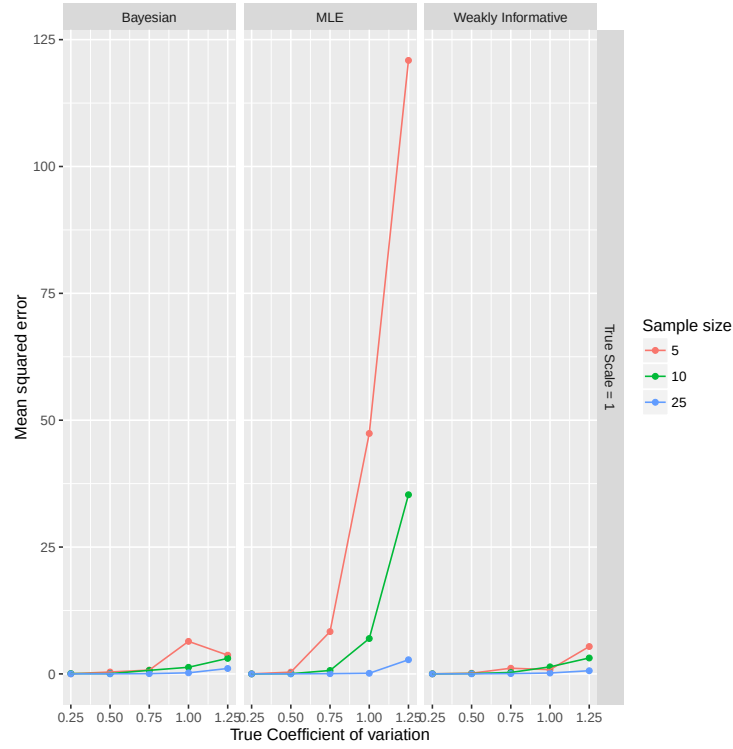


FIGURE 5.2: MSE of the estimate of the relative variation of the Weibull distribution. Each point on the graph is the mean of the MSE calculated from 1000 simulations. Note that the location parameter is fixed and equal to 1.

ML method. When the number of units increases, MSE of the relative variability decreases to zero with decreasing ρ .

The relative biases of the relative variability, ρ from ML and Bayesian methods for the large sample sizes are presented in figure 5.3. The patterns of the biases were relatively similar for various location parameter values and hence only the results when the location equal to one are provided. The relative biases from the Jeffreys' prior are positive and significantly higher than that from the ML method. When the sample size increases, the relative biases from both ML and Jeffreys' decrease close to zero. Nonetheless, the relative biases increase with increasing ρ values. Compared to the other two methods relative biases from weakly informative priors show noticeably low values. However, when the sample size is 1000 all the three methods show approximately similar results.

Figure 5.4 displays the computed MSE for the relative variability, ρ from ML and Bayesian estimation methods. The results are from 1000 simulations for the sample sizes 50, 100 and 1000. Overall, the MSE for relative variability follows the same

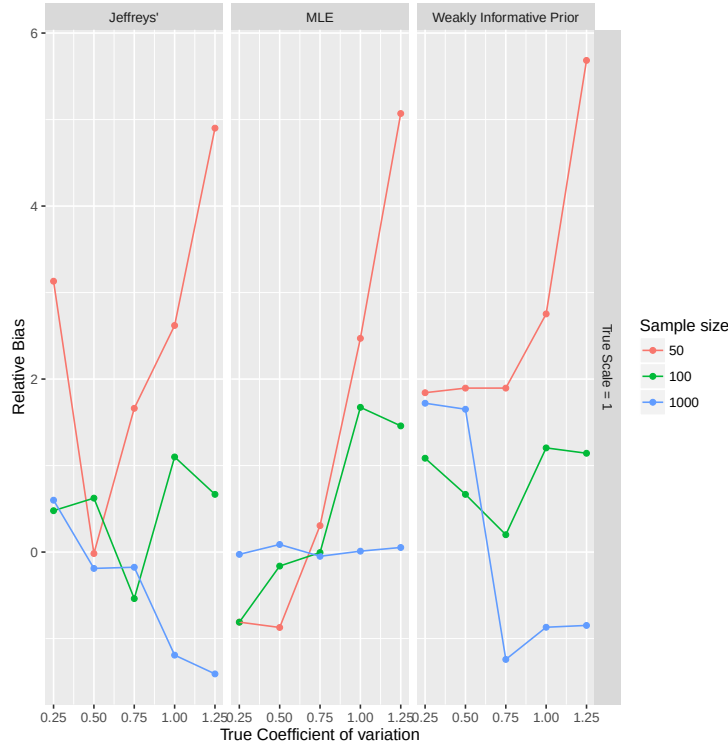


FIGURE 5.3: Relative bias of the estimates of relative variation of the Weibull distribution. All the points are the average of $(\hat{\rho} - \rho) * 100\% / \rho$ from 1000 simulations. The results are displayed for the large sample sizes 50, 100 and 1000. The location parameter of the distribution is fixed and set to 1.

pattern for both ML and Jeffreys' prior. Although MSE values decrease closer to zero with increasing sample sizes, they increase with increasing ρ values. Observed MSE values from both MLE and Jeffreys' prior for the sample size 1000 are approximately the same. However, the MSE values from weakly informative prior are considerably smaller than the other two methods.

5.3.2 25% Fails

We extended the analysis by introducing censoring to the problem. The set of specifications used for the simulation are as follows.

- Location = 1
- Five different rho (ρ) values (0.25, 0.50, 0.75, 1.00, 1.25)
- Six different sample sizes (10, 25, 50, 100, 500, 1000)

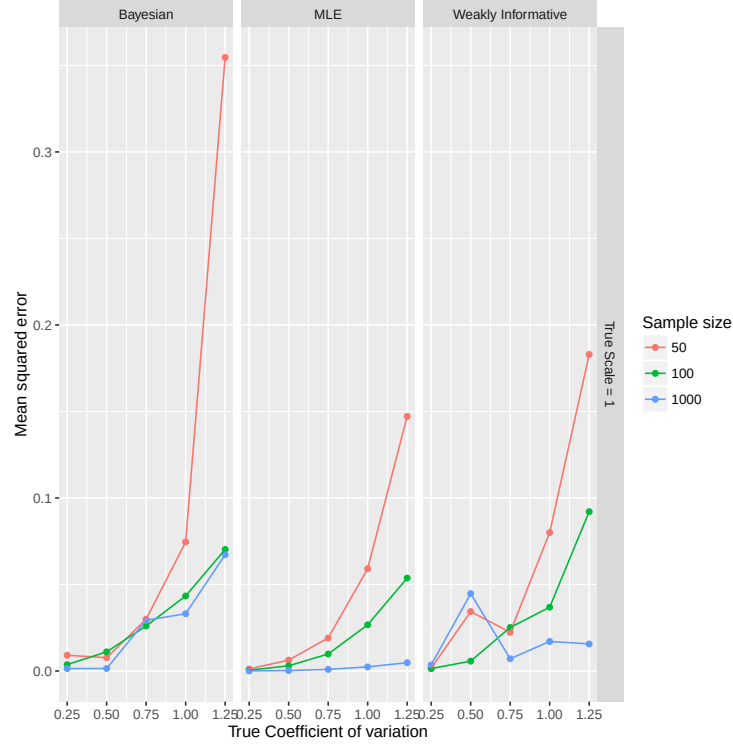


FIGURE 5.4: MSE of estimates of the relative variation of the Weibull distribution. Each point on the graph is the mean of MSE from 1000 simulation. Note that the location parameter is fixed and equal to 1. The results are illustrated for the sample sizes 50, 100 and 1000.

- 1000 replicates
- 25% fails

Location parameter of the Weibull was set to 1 as there was no noticeable difference in the patterns of the parameter estimates from different location values. When summarising the results we separate the sample sizes in to two groups as (10, 25, 50) and (100, 500, 1000) for the purpose of comparison.

Figure 5.5 represents the simulation results for the bias of parameter estimates of ρ , for both BE and MLE with sample sizes 10, 25 and 50. Relative biases from the derived Jeffreys' prior and the weakly informative prior show approximately similar pattern over the true value of ρ and are lower than the biases from the MLE for ρ . Biases from MLE start with negative values and increase with increasing values of ρ with a change of sign. Relative biases from all the parameter estimation methods decrease with increasing sample size. However, biases from BE methods show a reduction after $\rho = 0.75$ for the sample of size 10.

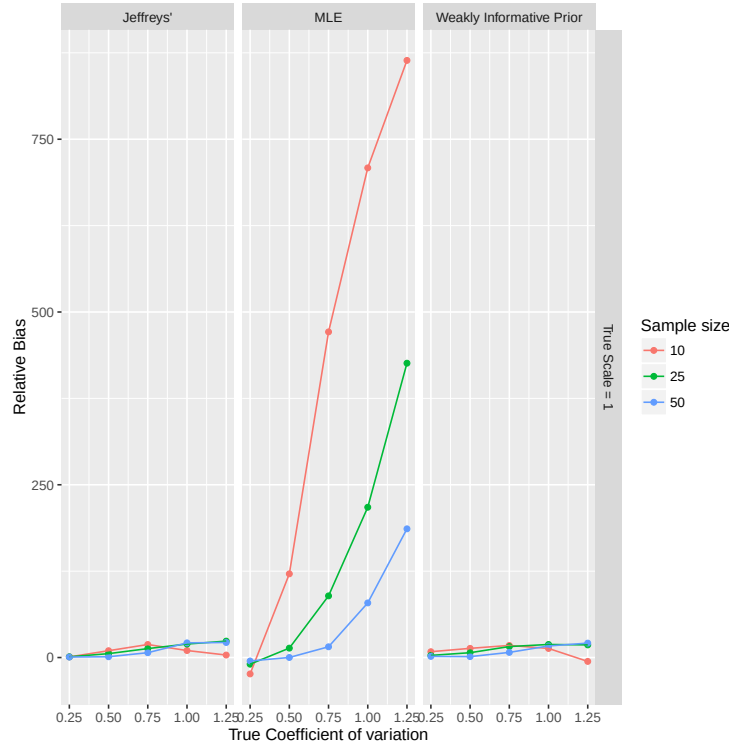


FIGURE 5.5: Relative bias % of the estimate of the relative variation ρ . Each point on the graph is the average of $(\hat{\rho} - \rho)/\rho$ from 1000 simulations. The results are displayed for the sample sizes 10, 25 and 50. Note that the location of the Weibull distribution is fixed and equal to 1.

Figure 5.6 displays the relative biases for the estimates of ρ for both BE and MLE with only 25% fails. Results from the Bayesian estimates show relatively similar patterns across the true values of ρ with approximately same magnitudes. When the sample size increases the relative biases decrease for all the parameter estimation methods. Relative biases for MLE are higher than the results from the Bayesian estimation. When compared to the relative biases from the complete data, it seems introduction of heavy censoring has increased the biases for both BE and MLE.

Figure 5.7 shows the MSE values for the estimated relative variability, ρ for both MLE and BE under heavy censoring. MSE for the MLE are considerably high compared to the Bayesian estimates. Both Jeffreys' prior and weakly informative prior have produced approximately similar results and are showing relatively similar patterns across the true coefficient of variation.

Figure 5.8 illustrates the simulation results for the MSE values for the estimates of the relative variation, ρ for the sample sizes 100, 500 and 1000. As the sample

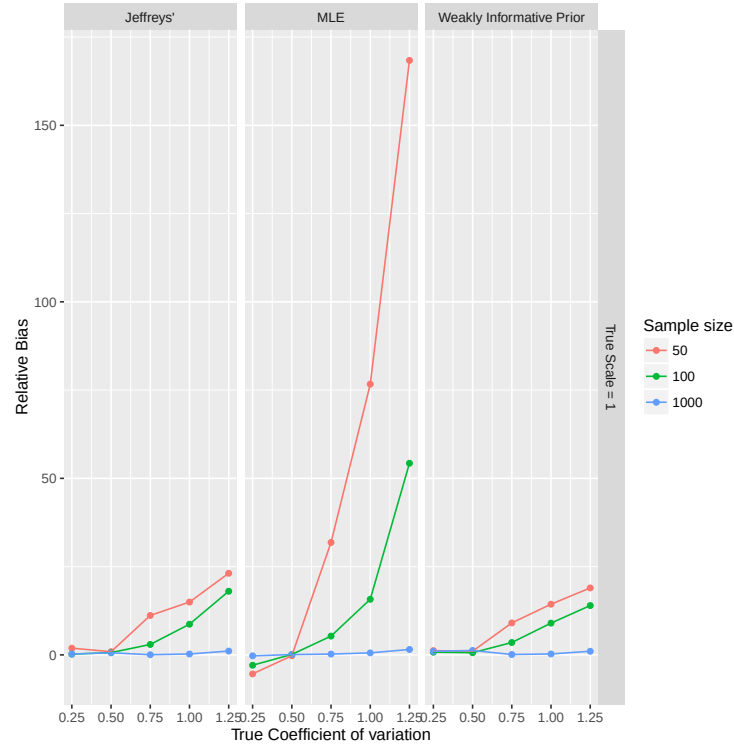


FIGURE 5.6: Relative bias % of the estimate of the relative variation ρ . Each point on the graph is the average of $(\hat{\rho} - \rho)/\rho$ from 1000 simulations, 25% fails. The results are displayed for the sample sizes 100, 500 and 1000. Note that the location of the Weibull distribution is fixed and equal to 1.

size increase, MSE values decrease towards zero for all the estimation methods and the choice of prior. MSE for the estimates of ρ shows a noticeably higher value for MLE for a sample size of 100, when the true value is equal to 1.25. MSE for ρ show approximately equal behaviour for both prior choices for the Bayesian analysis. On another note, introduction of substantial censoring has increased the MSE values for the estimates of the ρ .

5.3.3 Case Study: Application to wind speed data

We now apply the Jeffreys' prior to compare the variability of wind speed at two locations. Two published sets of wind speed data from Col du Grand-St-Bernard and Chasseral, Switzerland are used to compute and compare the relative variability of speed. Each data set has 24 observations, and the wind speeds are measured in ms^{-1} at the height of 10m from the ground level. Table 5.3 provides summary statistics of the data for the locations. Figure 5.9 represents the empirical densities of wind

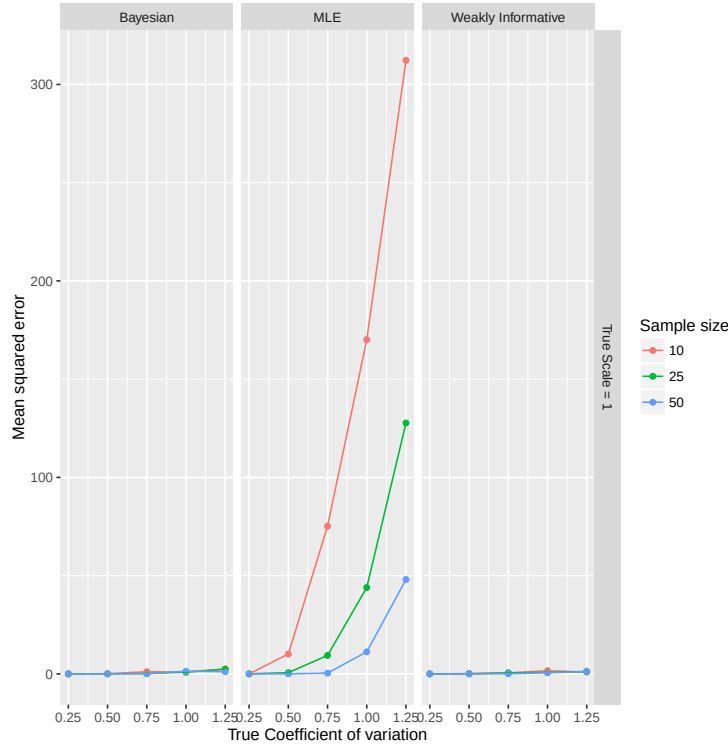


FIGURE 5.7: MSE of estimates of the relative variation of the Weibull distribution. Each point on the graph is the mean of MSE from 1000 simulation, 25% fails. Note that the location parameter is fixed and equal to 1. The results are illustrated for the sample sizes 10, 25 and 50.

speeds. Wind speeds of both locations vary between $4ms^{-1}$ and $12ms^{-1}$. In general, Chasseral has experienced winds with high average speed. Variability of the winds at Chasseral is greater than Col du Grand-St-Bernard.

We then perform a Bayesian analysis with MCMC sampling to calculate the relative variance of the two locations. The new Jeffreys' prior discussed in section 5.2 was used in the analysis. In order to find the posterior distribution of the parameters, 100 000 MCMC samples were drawn, and the first 5000 samples were discarded as burn-in values. Next, every 20th sample was taken to minimize the autocorrelation between the samples. The results of the Bayesian analysis are presented along with the MLE of the parameters in table 5.4.

Figure 5.10 shows the posterior distributions of the estimated relative variation of wind speed at the two locations. The posterior distributions broadly follow the empirical densities of the wind speeds. Variability of the wind speed at Col du Grand-St-Bernard is relatively smaller compared to Chasseral. Location of the posterior distribution

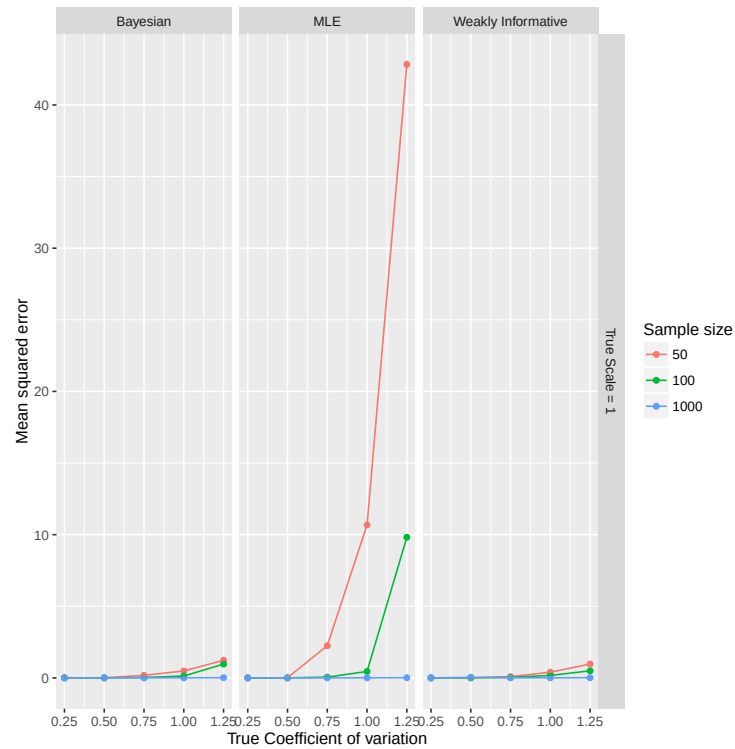


FIGURE 5.8: MSE of estimates of the relative variation of the Weibull distribution. Each point on the graph is the mean of MSE from 1000 simulation, 25% fails. Note that the location parameter is fixed and equal to 1. The results are illustrated for the sample sizes 100, 500 and 1000.

at Chasseral is shifted to the right, indicating a higher relative variability compared to Col du Grand-St-Bernard. Additionally, the posterior distribution of relative variability at Chasseral has a lower peak and slightly wider spread compared to Col du Grand-St-Bernard.

TABLE 5.3: Summary statistics of wind speed data at the locations Col du Grand-St-Bernard and Chasseral, Switzerland. All the measurements are in ms^{-1}

	Chasseral	Col du Grand-St-Bernard
Minimum	5.6000	4.2000
Mean	7.8667	5.3583
Maximum	11.7000	6.7000
Standard deviation	1.7583	0.7529

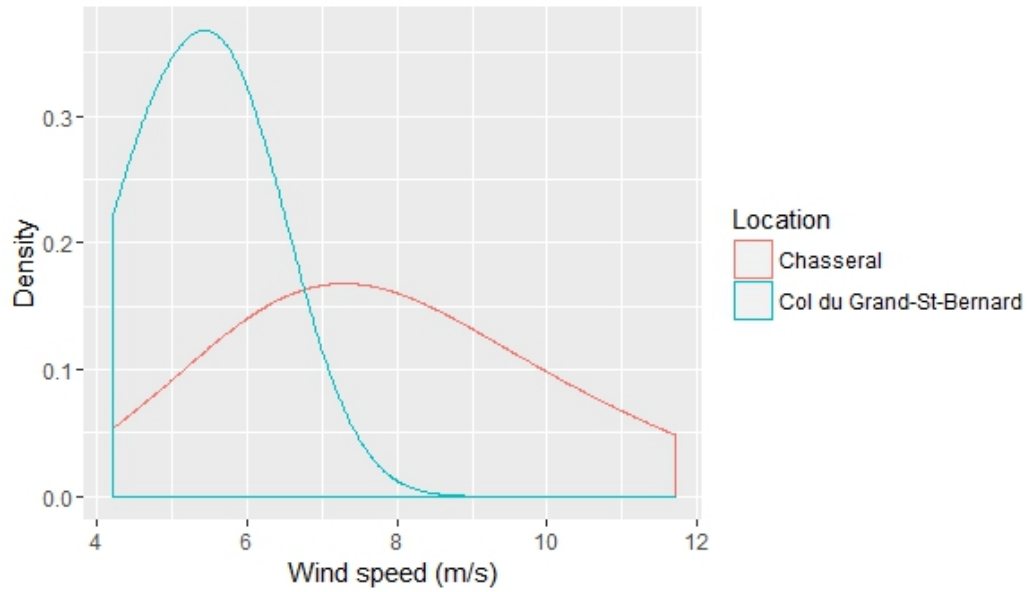


FIGURE 5.9: Empirical distribution of the wind speeds at the two locations.

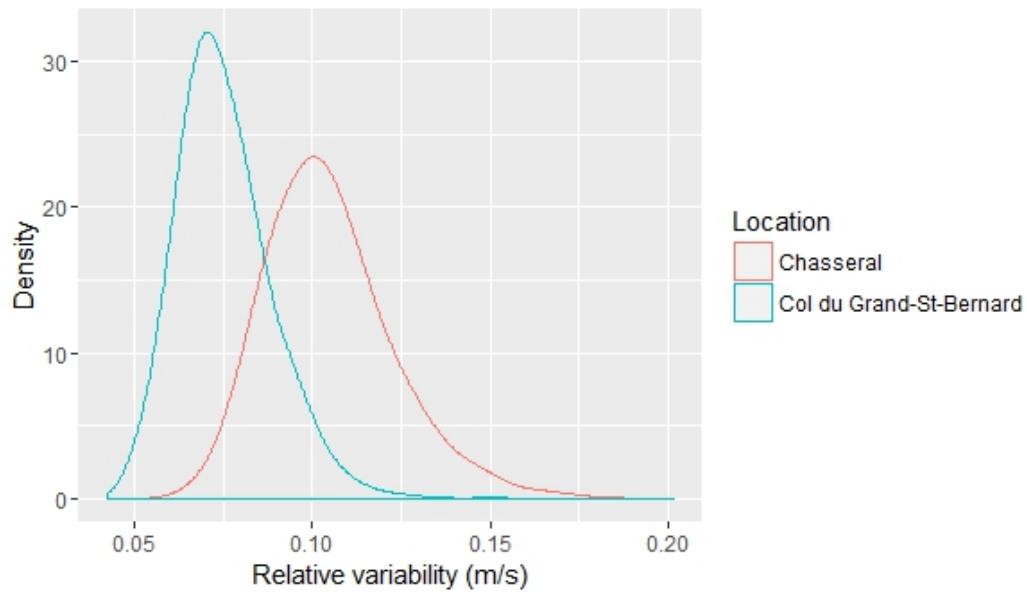


FIGURE 5.10: Posterior distribution of the estimated relative variation at the two locations. Estimates are from thinned MCMC sampling at every 20th observation.

TABLE 5.4: Estimated relative variability at Chasseral and Col du Grand-St-Bernard. ML estimates are presented with 95% confidence intervals and the Bayesian estimates are presented with 95% credible intervals. All the estimates are in ms^{-1}

	Chasseral	Col du Grand-St-Bernard
MLE	0.0989 (0.0681, 0.1297)	0.0705 (0.0480, 0.0931)
BE	0.10240 (0.0918, 0.1145)	0.07338 (0.0662, 0.0828)

Section 5.4: Discussion

In this chapter, we have derived a new Jeffreys' prior for the parameters, relative variation, and location of the Weibull distribution. The performance of the prior distribution was evaluated with a simulation study, and we have illustrated the application of the prior with the use of two real data sets. Under reparameterization, Jeffreys' prior provides invariant and universal priors. Also, it performs well for one-dimensional parameters. However, it should be reminded that there are some complications in the methods of the Jeffreys' prior when the parameters are multi-dimensional (Sun, 1997; Moala and Penha, 2014).

Additionally, to develop the prior density, it is required the existence of the information matrix. In order to obtain the asymptotic normality of the model, some regulatory conditions need to be satisfied. The Jeffreys' prior is not a second-order matching prior (Sun, 1997), but it is widely used because of its invariance properties, convenient derivations, computations, and understanding.

The results of our simulation are discussed next. For the samples with sizes lesser than 25, the relative biases are lower for the Bayesian estimates than the ML estimates. Relative biases from the Bayesian estimates are positive for both small and large sample sizes. For the ρ values, which are approximately lesser than 0.375, the relative biases are negative. Biases change its direction while ρ increases. When the sample size is small (5 or 10), the MLE method provides estimates with relatively large biases compared to the Bayesian estimations. Also, when the amount of information contributed by the prior distribution increases, biases decrease. According to the MSE values, the Bayesian method provides more precise results than the ML method for small sample sizes. Although the MSE values from the Bayesian method increase slightly when the ρ values increase, they remain closer to zero for the small sample sizes.

In contrast, for sample sizes lesser than 25, the MSE values from the ML method show considerable deviations from zero. The MSE values are only closer to zero when the ρ values are lower than 0.5 for MLE. All the variations of the relative biases and the MSE are significant for both Bayesian and ML as the sample sizes are comparatively

smaller. For all the small sample sizes, mse values resulted from the weakly informative prior remains closer to zero across the different ρ values.

Not surprisingly, when the sample size increases, the ML method shows better results compared to the Jeffreys' prior. Relative biases for large samples start negatively, but closer to zero for the ML method and change the direction gradually with increasing ρ values. All the relative bias values from the Bayesian method are positive and show notably larger values for the sample sizes lower than 100. However, the biases for both MLE and BE of the ρ are closer to zero and follows broadly the same pattern for the sample size 1000. MSE values for both ML and Jeffreys' prior follow approximately the same pattern. However, the MSE values from Jeffreys' prior are slightly higher than that of the MLE. On the other hand, MSE values from weakly informative priors are lower than the other two methods. But, when the sample size is 1000 all the methods show approximately similar variations across the true ρ value.

When heavy censoring was introduced to the data by considering only 25% fails, both the bias and MSE showed substantially higher values specially for MLE. This can be due to the considerable reduction of the effective sample size and hence the amount of information. When examining the simulation results, it seems the use of Bayesian methods can provide better results under heavy censoring even for small sample sizes such as 10 and 25.

From the results of our simulation study, we observed that the new Jeffreys' prior outperforms the MLE in terms of the MSE and the relative bias when the number of observations is smaller for the complete data. However, when the data are heavily censored the newly derived Jeffreys' prior produced better results than MLE even for a sample of size 100. Additionally, when the quantity of information carried in a prior distribution increases (even when we use a weakly informative prior as in the simulation study), the performance of the Bayesian estimates increases for both complete and censored data. This was an expected outcome related to the Bayesian analysis. On the other hand, we noticed the MLE performs better when the number of units increases, specifically, when the number of units is greater than 25 for complete data. It is well known that, in most of the failure data analyses, the number of failures is limited due to many constraints such as time and cost of the studies and the data

are censored. If the observed data are insufficient, the use of the MLE method might be problematic and might produce misleading results. Therefore, the new Jeffreys' prior introduced in this chapter can be effectively used to obtain more precise results when the sample sizes are small and the available prior information is very limited. Moreover, when the variation comparison of two processes or fleets is of interest, rather than the individual variation of the processes, the relative variation $\rho = \sigma/\mu$, can be used for the comparisons. It is shown that this reparameterization is efficient for studies such as wind speed assessments and fleet failure analyses.

6. Conclusions and possible future directions

Failure data analysis is a stimulating area of statistics and the methods of failure data analysis are continuously evolving. In this thesis, we mainly considered a single, time-to-failure data set of telephones to illustrate how probabilistic risk analysis can be carried out to derive the best course of actions in fleet management. The methods discussed in the chapters can be applied to any commercial fleet failure data to evaluate the associated risk and the cost of maintenance and repair.

In chapter 2, we tried to establish a method in evaluating the repair and replacement cost of the components in a fleet. In our work, we considered only a system with one unit. The methods can easily extend to any number of units in operation, allowing ease computation of repair and replacement. Also, we assumed the general norm of higher repair cost compared to the replacement. But, the policies can utilize for alternative conditions. We assumed in our cost calculation that the failure modes were the same for all the units. However, evaluation of the cost of repair and replacement with changing failure modes such as crack, wear-out, or software failures would be of greater interest.

When comparing parameter estimation methods for quantile estimation in chapter 3, we focused only on two parameter estimation methods; the maximum likelihood method and the median rank regression method. An extension of the simulation study comparing other parameter estimation methods such as Bayesian estimation, empirical Bayesian estimation, and method of moments for extreme quantile estimation with different censoring schemes would be of future research interest. In addition to that our simulation study was restricted to the Weibull quantile estimation. If the log-location-scale form of the Weibull distribution is considered in future researches,

then the generalization of the results to the family of distributions might be worthy for comparison purposes. Also, the simulation can be adapted to other failure models.

In chapter 4, we discussed the model selection methods and Bayesian model averaging to account for model uncertainty. There we focused on seven different failure models and derived posterior samples with weakly informative priors. The Bayesian analysis can be effectively extended with informative priors to obtain the posterior samples. But this would require additional information about the failure mechanism. This might involve expert elicitation in deriving appropriate priors and hyperparameter estimation. Nevertheless, it would be of great interest to examine the deviations from the optimal strategy and any possible fluctuations of the repair and replacement costs.

When evaluating the model selection based on the repair and replacement cost, we only compared the costs corresponding to each distribution. However, there can be an opportunity cost involved in selecting one model over another. This claim can be used to extend the cost calculation to a comparison of the opportunity costs of both model selection and parameter estimation.

In chapter 5, we derived a new Jeffreys' prior for a function of parameters of the Weibull distribution. We used the log-location-scale form of the Weibull distribution and derived the Jeffreys' prior for the coefficient of variation. Other non-informative priors can be developed for the same quantity of the Weibull distribution such as the reference prior, Tibshirani's prior and maximal data information prior. It would be interesting to examine the properties of these parameters for the coefficient of variation of the Weibull distribution. Further, an extension to an assessment of the performance of each prior with a generalization of the results to the log-location-scale family would be interesting.

A. Bayesian Model Averaging Appendix

Section A.1: Prior Distributions

In Chapter 4, we used several prior distributions for the Bayesian model averaging. In this section we provide details and functional formats of the selected prior distributions.

Recall, the parameters of the Weibull, log-normal and log-logistic distributions are σ and μ (Note that all the failure models are in log-location and scale form). For the two parameters, we used $\mu \sim Uniform(0, 20)$, $\sigma \sim Inverse - gamma(0.1, 0.1)$ as the prior distributions. Full models of the used prior distributions are:

$$p(\mu) = \frac{1}{b-a} = \frac{1}{20-0} \quad (A.1)$$

$$p(\sigma; \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} \sigma^{(-\alpha-1)} \exp\left(\frac{-\beta}{\sigma}\right) \quad (A.2)$$

Here, $\alpha > 0$ is the shape parameter and $\beta > 0$ is the rate of the Inverse-Gamma distribution and both parameters; α and β are set to 0.1 to make the prior distribution weakly informative.

For the parameters of the gamma model (α - rate and β -shape), we used two gamma distributions as the prior.

$$p(\alpha; a, b) = \frac{a^b \alpha^{b-1} \exp(-a\alpha)}{\Gamma(b)} \quad (A.3)$$

$$p(\beta; a, b) = \frac{a^b \beta^{b-1} \exp(-a\beta)}{\Gamma(b)} \quad (\text{A.4})$$

Here, $a > 0$ is the rate parameter and $b > 0$ is the shape parameter. We set both a and b to 0.1.

For the Inverse-Gaussian model we chose two weakly informative priors as follows.

$$p(\lambda; a, b) = \frac{a^b \lambda^{b-1} \exp(-a\lambda)}{\Gamma(b)} \quad (\text{A.5})$$

$$p(\mu; a, b) = \frac{a^b \mu^{b-1} \exp(-a\mu)}{\Gamma(b)} \quad (\text{A.6})$$

Here again, $a > 0$ is the rate parameter and $b > 0$ is the shape parameter and we set the parameters to 0.1.

B. Jeffreys' Prior Appendix

Section B.1: Distribution of the random variable Y

We provide here the proof of the distribution of the term $Y = \exp\left(\frac{1-\theta}{\theta\rho}\right)$ expressed in the equation 5.14. Recall that $Y = g(x) = \exp\left(\frac{1-\theta}{\theta\rho}\right)$. Then we say Y follows an exponential distribution with mean 1.

Proof: X is an absolutely continuous random variable and X follows a Weibull distribution with parameters μ and ρ . The probability density function of X is given by equation 5.6 over the support $R_X = [0, \infty)$. Also, $g(x)$ is an increasing function over the support R_x . Then the support of Y is $R_y = [0, \infty)$. Therefore, the probability density function of Y is given by,

$$f_Y(y) = f_X(g^{-1}(y)) \frac{dg^{-1}(y)}{dy} \quad \forall y \geq 0 \quad (\text{B.1})$$

Hence,

$$g^{-1}(y) = \exp[\theta\rho \log(y) + \theta]$$

and

$$\begin{aligned} \frac{dg^{-1}(y)}{dy} &= \frac{\theta\rho \exp[\theta\rho \log(y) + \theta]}{y} \\ f_Y(y) &= \frac{1}{\theta\rho \exp[\theta\rho \log(y) + \theta]} \exp[\log(y) - y] \times \frac{\theta\rho \exp[\theta\rho \log(y) + \theta]}{y} \\ &= e^{-y} \end{aligned} \quad (\text{B.2})$$

This is of the form of the probability density function of exponential distribution with mean 1.

Section B.2: Jeffreys' prior for the parameters μ and ρ^2

In this section we provide the mathematical derivation of the Jeffreys' prior for the parameters μ and ρ^2 of the Weibull distribution.

$$\det I(\theta, \rho) = \frac{1}{\theta \rho^2}$$

Proof: Let,

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

Then, the determinant of A, denoted as $\det A$ is,

$$\det A = ad - bc$$

Now the Fisher's information matrix of the parameters θ and ρ is,

$$I(\theta, \rho) = \begin{bmatrix} -\left(\frac{1}{\theta^2} + \frac{1}{\theta^2 \rho^2} + \frac{2(1+\gamma_1)}{\theta^2 \rho} + \frac{(\gamma_2+2\gamma_1)}{\theta^2}\right) & -\left(\frac{1}{\theta \rho} + \frac{(\gamma_2+2\gamma_1)}{\theta \rho} + \frac{(1+\gamma_1)}{\theta \rho^2}\right) \\ -\left(\frac{1}{\theta \rho} + \frac{(\gamma_2+2\gamma_1)}{\theta \rho} + \frac{(1+\gamma_1)}{\theta \rho^2}\right) & \frac{-(1+\gamma_2+2\gamma_1)}{\rho^2} \end{bmatrix}$$

Therefore the determinant of $I(\theta, \rho)$ is,

$$\begin{aligned}
\det[I(\theta, \rho)] &= \left[\frac{-(1 + \gamma_2 + 2\gamma_1)}{\rho^2} \right] \times \left[- \left(\frac{1}{\theta^2} + \frac{1}{\theta^2 \rho^2} + \frac{2(1 + \gamma_1)}{\theta^2 \rho} + \frac{(\gamma_2 + 2\gamma_1)}{\theta^2} \right) \right] \\
&\quad - \left[- \left(\frac{1}{\theta \rho} + \frac{(\gamma_2 + 2\gamma_1)}{\theta \rho} + \frac{(1 + \gamma_1)}{\theta \rho^2} \right) \right]^2 \\
&= \frac{(1 + \gamma_2 + 2\gamma_1)^2}{\theta^2 \rho^2} + \frac{(1 + \gamma_2 + 2\gamma_1)}{\theta^2 \rho^4} + \frac{2(1 + \gamma_1)(1 + \gamma_2 + 2\gamma_1)}{\theta^2 \rho^3} \\
&\quad - \frac{(1 + \gamma_2 + 2\gamma_1)^2}{\theta^2 \rho^2} - \frac{2(1 + \gamma_1)(1 + \gamma_2 + 2\gamma_1)}{\theta^2 \rho^3} \\
&\quad - \frac{(1 + \gamma_1)^2}{\theta^2 \rho^4} \\
&= \frac{(1 + \gamma_2 + 2\gamma_1)}{\theta^2 \rho^4} - \frac{(1 + 2\gamma_1 + \gamma_1^2)}{\theta^2 \rho^4} \\
&= \frac{(\gamma_2 - \gamma_1^2)}{\theta^2 \rho^4}
\end{aligned} \tag{B.3}$$

Here $-\gamma_1$ is the Euler's constant and $\gamma_2 - \gamma_1^2$ is the variance of $\log(Y)$ (Sun, 1997).


```

6         184, 194, 208, 216, 224, 232, 240,
7         246, 256, 264, 272, 280, 288, 304,
8         308, 328, 340, 352, 358, 360, 384,
9         392, 400, 424, 438, 448, 464, 480,
10        536, 552, 576, 608, 656, 716)
11    times = c (1, 4, 2, 4, 3, 1, 1,
12              5, 4, 2, 1, 1, 2, 1, 1,
13              1, 1, 3, 1, 1, 5, 1, 3,
14              1, 2, 1, 4, 1, 1, 1, 1,
15              2, 1, 1, 1, 1, 1, 2, 1,
16              1, 1, 1, 1, 1, 1, 1, 1,
17              1, 1, 1, 1, 1, 1, 1, 1, 1))
18    T.cen <- data.wei[max(2,
19                      ceiling(propFailed*samplesize))]
20    censored <- data.wei > T.cen
21    event <- c (rep(1, length(data.wei[censored == FALSE])),
22              rep(0, length(data.wei[censored == TRUE])))
23    gen.fails <- c (data.wei[censored == FALSE], rep(NA,
24              length(data.wei[censored == TRUE])))
25    rank.fails <- rank (gen.fails, ties.method = "first",
26                      na.last = "keep")
27    gen.fails [is.na(gen.fails)] <- T.cen
28    y.cen <- rep (T.cen, length(data.wei[censored == TRUE]))
29
30    simul.data <- data.frame(sampleddata = data.wei,
31                            Censored = censored,
32                            obs.fails = gen.fails,
33                            ranks = rank.fails)
34
35    sim.data.bay <- list (sampledata = data.wei[censored == FALSE],
36                        obs_fails = gen.fails,
37                        Censoreddata = censored,
38                        Censored = event,
39                        J = length (data.wei[censored == FALSE]),
40                        N_cen = length(data.wei[censored == TRUE]),
41                        y_cen = y.cen,
42                        ranks = rank.fails)
43    return(sim.data.bay)
44 }

```

```
45
46 #Define STAN models for the censored data
47
48 model_weibull <- "
49 data {
50   int<lower=0> J; // number of observations
51   int<lower=0> N_cen; // number of censored observations
52   vector [J]sampledata; // failure in hours
53   vector [N_cen]y_cen; //values of censored data
54 }
55
56 parameters {
57   real mu; // scale
58   real<lower=0> sigma; // shape
59 }
60 model {
61   //prior
62   mu ~ uniform (0, 20);
63   sigma ~ inv_gamma(0.1, 0.1);
64
65   //Likelihood and prior
66   sampledata ~ weibull (1/sigma, exp(mu)); //Uncensored data
67   increment_log_prob(weibull_ccdf_log (y_cen, 1/sigma, exp(mu)));
68   //Censored data
69 }
70 "
71 mod_weibull <- stan_model(model_code = model_weibull)
72
73 #####
74
75 model_gamma <- "data {
76   int<lower=0> J; // number of observations
77   int<lower=0> N_cen; // number of censored observations
78   vector [J]sampledata; // failure in hours
79   vector [N_cen]y_cen; //values of censored data
80 }
81
82 parameters {
83   real<lower=0> alpha; // shape
```

```

84 real<lower=0> beta; // rate
85 }
86
87 model {
88 //define priors of alpha and beta
89 alpha ~ gamma(0.1, 0.1);
90 beta ~ gamma(0.1, 0.1);
91
92 //Likelihood of the data
93 sampledata ~ gamma(beta, alpha);
94 increment_log_prob(gamma_ccdf_log (y_cen, beta, alpha));
95 }"
96
97 mod_gamma <- stan_model (model_code = model_gamma)
98
99 ##### MAXIMUM LIKELIHOOD ESTIMATION #####
100
101 mle_weibull <- function (data, #Data
102                           p.int = c(6, 0.5),
103                           #Initial values of the parameters
104                           samplesize,
105                           censored)
106 {loglik <- function (theta){
107   #theta[1] <- scale -
108   scale parameter of weibull distribution
109   #theta [2] <- shape - shape parameter
110   ll <- sum(ifelse(censored,
111                   suppressWarnings(pweibull(data, 1/theta[2],
112                   exp(theta[1]), lower.tail = FALSE,
113                   log.p = TRUE))),
114             suppressWarnings(dweibull(data, 1/theta[2],
115             exp(theta[1]), log = TRUE))))
116   return (-ll)
117 }
118
119 #Optimise to find the ML estimates
120 estim1 <- optim (p.int, loglik,
121                 method = "BFGS",
122                 control = list(maxit = 10000,

```

```

123         reltol = .Machine$double.eps),
124         hessian = TRUE)
125     #Results
126 results <- estim1
127 return (results)
128 }
129
130 #####
131
132 mle_gamma <- function (data, #Data
133     p.int = c(6, 0.5), #Initial values of the parameters
134     samplesize, censored)
135 {
136     loglik <- function (theta){
137         #theta[1] <- scale -
138         scale parameter of weibull distribution
139         #theta [2] <- shape - shape parameter
140         ll <- sum(ifelse(censored,
141             suppressWarnings(pgammma(data, theta[1],
142             theta[2], lower.tail = FALSE,
143             log.p = TRUE))),
144             suppressWarnings(dgammma(data, theta[1],
145             theta[2], log = TRUE))))
146         return (-ll)
147     }
148
149 #Optimise to find the ML estimates
150 estim1 <- optim (p.int, loglik,
151     method = "BFGS",
152     control = list(maxit = 10000,
153     reltol = .Machine$double.eps),
154     hessian = TRUE)
155 #Results
156 results <- estim1
157 return (results)
158 }
159
160 ##### DEFINE THE MRR FUNCTION#####
161

```

```

162 #Adjusted ranks
163 ri <- function (sim.data) {
164   N <- length (sim.data$obs_fails)
165   r0 <- 0 #Initial rank
166   #Extract the ranks of failure data
167   sj <- sim.data$ranks[sim.data$Censoreddata == FALSE]
168   ri <- NULL
169   ri[1] <- r0 + (N + 1 - r0) / (N + 1 - (sj[1] - 1))
170
171   for (i in 2:length(sj)) {
172     #Adjusted ranks
173     ri[i] <- ri[i - 1] + (N + 1 - ri[i - 1]) / (N + 1 - (sj[i] - 1))
174   }
175   return (ri)
176 }
177
178 #Shape parameter
179 b1 <- function (sim.data) {
180   ti <- sim.data$obs_fails[sim.data$Censoreddata == FALSE]
181   #Failure times
182   yi <- log (ti) #Dependent
183   N <- length (sim.data$obs_fails)
184   n <- length(ti)
185   xi <- log (-log (1 - ((ri(sim.data) - 0.3) / (N + 0.4))))
186   #Explanatory
187   sigma.hat <- (n*sum (xi*yi) - sum(xi)*sum(yi)) / (n*sum(xi^2) -
188     (sum(xi))^2)
189   return (sigma.hat)
190 }
191
192 #Scale parameter
193 b0 <- function (sim.data) {
194   ti <- sim.data$obs_fails[sim.data$Censoreddata == FALSE]
195   #Failure times
196   yi <- log (ti) #Dependent
197   N <- length (sim.data$obs_fails)
198   n <- length(ti)
199   xi <- log (-log (1 - ((ri(sim.data) - 0.3) / (N + 0.4))))
200   #Explanatory

```

```

201
202 b1 <- b1(sim.data)
203 mu.hat <- (1/n)*(sum(yi) - b1*sum(xi))
204
205 #SE of the estimates
206
207 sxx <- sum(xi^2)
208 syy <- sum (yi^2)
209
210 vare <- (n*syy - sum(yi)^2 - ((b1)^2)*(n*sxx - sum(xi)^2))/(n*(n-2))
211 #var of errors
212 varb1 <- n*vare / (n*sxx - (sum(xi))^2)
213 varmu <- varb1*sxx/n
214 return (c(mu.hat, b1, sqrt(vare), sqrt (varb1), sqrt(varmu)))
215 }
216
217 #####
218
219 #### OBTAIN COMPLETE DATA FOR THE ANALYSIS ####
220
221 data.tele <- (weibull.data (samplesize = 88,
222                             propFailed = 1))
223
224 ##### BAYESIAN ESTIMATES #####
225
226 set.seed(12365)
227 weibull.tele <- sampling (mod_weibull, data = data.tele,
228                           chains = 1, iter = 10^6,
229                           warmup = 40000, thin = 20)
230 weibull.tele
231 weib.tele_be <- summary (weibull.tele)
232 se_be.weib <- weib.tele_be$summary[, 3]/sqrt (88)
233
234 set.seed(12569)
235 gamma.tele <- sampling (mod_gamma, data = data.tele,
236                           chains = 1, iter = 10^4,
237                           warmup = 400, thin = 20)
238 gamma.tele
239 summary (gamma.tele)

```

```

240
241 ##### MAXIMUM LIKELIHOOD ESTIMATES #####
242
243 weib_mle.tele <- with (data.tele, mle_weibull(data = obs_fails,
244       p.int = c (1, 6),
245       samplesize = length(obs_fails),
246       censored = Censoreddata))
247 gamma_mle.tele <- with (data.tele, mle_gamma(data =
248       obs_fails,
249       p.int = c(1, 6),
250       samplesize = length(obs_fails),
251       censored = Censoreddata))
252 # Solve the hessian matrix to get the variance covariance matrix
253
254 var_weib.tele <- solve (weib_mle.tele$hessian)
255 se_weib.tele <- sqrt(diag(var_weib.tele))/sqrt(88)
256 var_gamma.tele <- solve (gamma_mle.tele$hessian)
257 se_gamma.tele <- sqrt(diag(var_gamma.tele))/sqrt(88)
258
259 ##### MRR ESTIMATES FOR THE WEIBULL DISTRIBUTION #####
260 weib_mrr.tele <- with (data.tele, b0(sim.data = data.tele))
261 #####
262
263 weibull.coda <- mcmc.list(lapply(1:ncol(weibull.tele),
264       function(x) {mcmc(as.array(weibull.tele)[,x,])}))
265
266 gamma.coda <- mcmc.list(lapply(1:ncol(gamma.tele),
267       function(x) {mcmc(as.array(gamma.tele)[,x,])}))
268
269 posterior_df <- data.frame(weib = weibull.coda[[1]][,1:2],
270       gamma = gamma.coda[[1]][, 1:2])
271
272 #Median of the posterior parameter values
273 median_pos_mu <- median (posterior_df$weib.mu)
274 median_pos_sigma <- median (posterior_df$weib.sigma)
275 median_pos_alpha <- median (posterior_df$gamma.alpha)
276 median_pos_beta <- median (posterior_df$gamma.beta)
277
278 # Assume the optimal replacement time is the

```

```

279 # 80th quantile found with median of posterior parameters
280 #Compute the optimal quantile
281 optimalQ_weib <- qweibull(0.8, shape =
282     1/median_pos_sigma,
283     scale = exp(median_pos_mu))
284 #Compute the cost ratio according to the optimal replacement time
285 costR_weib <- optimalQ_weib*hweibull(optimalQ_weib,
286     shape = 1/median_pos_sigma,
287     scale = exp(median_pos_mu)) -
288     Hweibull(optimalQ_weib,
289     shape = 1/median_pos_sigma,
290     scale = exp(median_pos_mu))
291 #Optimal expected cost rate at the optimal replacement time
292 costrateOpt_weib <- (Hweibull(optimalQ_weib,
293     shape = 1/median_pos_sigma,
294     scale = exp(median_pos_mu))
295     + costR_weib) / optimalQ_weib
296 #80th quantile from posterior values
297 quantile_weib <- qweibull(0.5,
298     shape = 1/posterior_df$weib.sigma,
299     scale = exp(posterior_df$weib.mu))
300 #Total expected cost rate with c1 = 1
301 Totcost_weib <- (Hweibull(quantile_weib,
302     shape = 1/median_pos_sigma,
303     scale = exp(median_pos_mu))
304     + costR_weib) / quantile_weib
305
306 cost <- data.frame(quantile_weib, Totcost_weib)
307
308 # cost_ordered <- cost[order(cost$quantile_weib), ]
309 # plot (quantile_weib, Totcost_weib, type = "l")
310 # print (plot_cost <- qplot (x = quantile_weib,
311 #     y = Totcost_weib,
312 #     data = cost))
313
314 pd <- position_dodge(0.01)
315 print (plot_costdensity <- ggplot (data = cost, aes (Totcost_weib))
316     + geom_density()
317     + xlab ("Expected cost rate per hour ($)"))

```

```

318   + ylab ("Density")
319   #+ xlim (NA, 0.00637)
320   + theme(axis.text.x = element_text(angle = 90,
321   hjust = 1, vjust = 0.5)))
322
323 print (plot_costecdf <- ggplot (data = cost, aes
324   (Totcost_weib))
325   + stat_ecdf ()
326   + xlab ("Expected cost rate per hour ($)")
327   #+ xlim (NA, 0.00637)
328   + theme(axis.text.x = element_text(angle = 90,
329   hjust = 1, vjust = 0.5)))
330 # Arrange the plots
331 grid.arrange (plot_costdensity, plot_costecdf, nrow = 1)
332
333 ##### SUBOPTIMAL SOLUTION #####
334
335 Quantile <- c (0.1, 0.125, 0.15, 0.175, 0.2, 0.25,
336   0.3, 0.4, 0.45, 0.5,0.6, 0.7, 0.8,
337   0.825, 0.85, 0.875, 0.9, 0.925, 0.95,
338   0.975, 0.99)
339
340 optimalQ_weib <- qweibull(Quantile,
341   shape = 1/median_pos_sigma,
342   scale = exp(median_pos_mu))
343 costrateOpt_weib <- (Hweibull(optimalQ_weib,
344   shape = 1/median_pos_sigma,
345   scale = exp(median_pos_mu))
346   + costR_weib) / optimalQ_weib
347 print (plot_suboptimal <- (qplot (Quantile,
348   costrateOpt_weib,
349   #data = shape.df,
350   geom = c("point", "line"))
351   + ylab ("Optimal expected cost
352   rate per hour($)")
353   + scale_x_continuous
354   (name ="Quantile",
355   breaks = c (0.1, 0.5, 0.7,
356   0.8, 0.9,0.99, 0.999),

```

```

357         labels = c (0.1, 0.5, 0.7, 0.8,
358                     0.9, 0.99, 0.999))
359         + theme(axis.text.x =
360                 element_text(angle = 90, hjust = 1,
361                               vjust = 0.5))
362         #+labs (colour = "True shape"))
363
364 #####
365
366 #Sensitivity Analysis
367 #Cost rate
368 #Use 4 different cost rates to see the effect on the Weibull
369     distribution
370
371 Quantile <- c (0.1, 0.125, 0.15, 0.175, 0.2, 0.25,
372               0.3, 0.4, 0.45, 0.5,0.6, 0.7, 0.8,
373               0.9, 0.925, 0.95, 0.975, 0.99)
374 quantiles_weib <- qweibull(Quantile, shape = 1/median_pos_sigma,
375                           scale = exp(median_pos_mu))
376
377 #Total expected cost rate with c1 = 1
378
379 Totcost_0.1 <- (Hweibull(quantiles_weib, shape = 1/median_pos_sigma,
380                         scale = exp(median_pos_mu)) + 0.1) /
381               quantiles_weib
382
383 Totcost_0.2 <- (Hweibull(quantiles_weib, shape = 1/median_pos_sigma,
384                         scale = exp(median_pos_mu)) + 0.2) /
385               quantiles_weib
386
387 Totcost_0.5 <- (Hweibull(quantiles_weib, shape = 1/median_pos_sigma,
388                         scale = exp(median_pos_mu)) + 0.5) /
389               quantiles_weib
390
391 Totcost_0.9 <- (Hweibull(quantiles_weib, shape = 1/median_pos_sigma,
392                         scale = exp(median_pos_mu)) + 0.9) /
393               quantiles_weib
394

```

```

395
396 sensitivity_costR <- data.frame (Quantile,
397                                   quantiles_weib,
398                                   Totcost_0.1,
399                                   Totcost_0.2,
400                                   Totcost_0.5,
401                                   Totcost_0.9)
402
403 sensitivity_costM <- melt (sensitivity_costR,
404                             measure.vars = c ("Totcost_0.1",
405                                                 "Totcost_0.2",
406                                                 "Totcost_0.5",
407                                                 "Totcost_0.9"))
408
409 print (plot_sensitivity_costR <- (ggplot (
410                                   mapping = aes (Quantile,
411                                                   value,
412                                                   group = factor (variable),
413                                                   colour = factor (variable)),
414                                   data = sensitivity_costM)
415   + geom_point()
416   + geom_line()
417   + scale_colour_manual("Cost ratio",
418                         values = c ("red",
419                                     "green",
420                                     "blue",
421                                     "purple"),
422                                   labels = c("0.1", "0.2", "0.5", "0.9"
423
424   ))
425
426   + scale_x_continuous
427   (name = "Quantile",
428     breaks = c (0.1, 0.5, 0.7, 0.8, 0.9,
429                 0.99), labels = c (0.1, 0.5, 0.7,
430                                   0.8, 0.9, 0.99))
431   + ylab ("Optimal expected cost rate
432           per hour ($)")
433   + theme(axis.text.x = element_text(
434           angle = 90, hjust = 1, vjust = 0.5)
435   ))

```



```

472 CostR_weib <- (Hweibull(quantiles_weib, shape = 1/ median_pos_sigma,
473                        scale = exp(median_pos_mu))
474              + costR_weib) / quantiles_weib
475
476 CostR_gamma <- (-pgamma(quantiles_gamma, shape = median_pos_beta,
477                        rate = median_pos_alpha,
478                        lower = FALSE, log = TRUE)
479              + costR_weib)/quantiles_gamma
480
481 sensitivity_model <- data.frame(Quantile, CostR_gamma, CostR_weib)
482
483 sensitivity_ModelM <- melt (sensitivity_model,
484                          measure.vars = c ("CostR_gamma",
485                                           "CostR_weib"))
486
487 print (plot_sensitivity.Model <- (qplot (Quantile,
488                                         value,
489                                         data = sensitivity_ModelM,
490                                         group = factor (variable),
491                                         colour = factor (variable),
492                                         geom = c("point", "line"))
493       + scale_colour_manual("Model",
494                             values = c ("red",
495                                           "blue"),
496                             labels = c ("Gamma", "Weibull"))
497       + scale_x_continuous
498         (name = "Quantile",
499          breaks = c (0.1, 0.5, 0.7, 0.8, 0.9,
500                    0.99),
501          labels = c (0.1, 0.5, 0.7, 0.8, 0.9,
502                    0.99))
503       + theme(axis.text.x = element_text
504               (angle = 90, hjust = 1, vjust = 0.5))
505       +ylab ("Optimal expected cost
506              rate per hour ($)"))
507
508 ##### PARAMETER ESTIMATION METHOD #####
509
510 CostR_BE <- (Hweibull(quantiles_weib, shape = 1/ median_pos_sigma,

```

```

511         scale = exp(median_pos_mu))
512         + costR_weib) / quantiles_weib
513
514 CostR_MLE <- (Hweibull(quantiles_weib, shape = 1/weib_mle.tele$par[2],
515         scale = exp (weib_mle.tele$par[1]))
516         + costR_weib) / quantiles_weib
517
518 CostR_MRR <- (Hweibull(quantiles_weib, shape = 1/weib_mrr.tele[2],
519         scale = exp(weib_mrr.tele[1]))
520         + costR_weib) / quantiles_weib
521
522 sensitivity_estim <- data.frame (Quantile, CostR_BE, CostR_MLE,
523                                 CostR_MRR)
524
525 sensitivity_estimM <- melt (sensitivity_estim,
526                             measure.vars = c("CostR_BE",
527                                                "CostR_MLE",
528                                                "CostR_MRR"))
529
530 print (plot_sensitivity.Estim <- (qplot (Quantile,
531                                         value,
532                                         data = sensitivity_estimM,
533                                         group = factor (variable),
534                                         colour = factor (variable),
535                                         geom = c("point", "line"))
536         + scale_colour_manual("Method",
537                               values = c("red",
538                                           "blue",
539                                           "green"),
540                               labels = c ("Bayesian", "MLE",
541                                           "MRR"))
542         + scale_x_continuous
543           (name = "Quantile",
544            breaks = c (0.1, 0.5, 0.7,
545                        0.8, 0.9)
546            labels = c (0.1, 0.5, 0.7,
547                        0.8, 0.9))
548         + theme(axis.text.x =
549                 element_text(angle = 90,

```

```

550             hjust = 1, vjust = 0.5))
551         + ylab ("Optimal expected cost
552                 rate per hour ($)"))
553
554 ##### WEIBULL PROBABILITY PLOT #####
555
556 #Median Rank Regression
557
558 #Since the data are complete, reverse ranks are not computed
559 #Use Bernard's formula to compute the predictor variable
560
561 B_tele <- with (data.tele, (ranks - 0.3) / (n + 0.4))
562
563 #Compute the quantiles
564 #Weibull quantile
565 pp_wei <- log (-log (1 - B_tele))
566
567 p_gamma <- pgamma(data.tele$obs_fails, shape = gamma_mle.tele$par[1],
568                  rate = gamma_mle.tele$par[2])
569 p_weibull <- pweibull (data.tele$obs_fails,
570                      shape = 1/weib_mle.tele$par[2],
571                      scale = exp(weib_mle.tele$par[1]))
572 q_gamma <- log (-log(1 - p_gamma))
573 q_weibull <- log (-log (1 - p_weibull))
574
575
576 tele_gamma <- data.frame (lifetime = data.tele$obs_fails,
577                          loglife = log (data.tele$obs_fails),
578                          pp_wei = pp_wei,
579                          q_gamma = q_gamma,
580                          q_weibull = q_weibull)
581 tele_gamma_melt <- melt(tele_gamma, id.vars = "loglife",
582                       measure.vars = c ("pp_wei",
583                                         "q_gamma",
584                                         "q_weibull"))
585 #Scale the x-axis
586 lab1 <- seq (7, 720, by = 100)
587 lab2 <- log (lab1)
588 #Scale the y-axis

```

```

589 lab3 <- c (0.001, 0.007, 0.05, 0.1, 0.2,
590           0.3, 0.4, 0.5,0.6, 0.7, 0.8,
591           0.9, 0.99, 0.995, 0.999)
592 lab4 <- log (-log (1 - lab3))
593 lab5 <- qnorm (lab3)
594 lab6 <- qlogis(lab3)
595 print (plot_gamma <- (ggplot (tele_gamma, aes (loglife, pp_wei))
596                       + geom_point ()
597                       + geom_line (aes (loglife, q_gamma, col = "Gamma"),
598                                     size = 0.6)
599                       + geom_line (aes (loglife, q_weibull, col = "Weibull"),
600                                     size = 0.6)
601                       + scale_x_continuous (breaks = lab2, labels = lab1,
602                                             limits = c (1.9, 6.6))
603                       + scale_y_continuous (breaks = lab4, labels = lab3,
604                                             limits = c (-6.99, 1.94))
605                       + xlab ("Lifetime (in hours)")
606                       + ylab ("Proportion of failure")
607                       + theme (axis.text.x = element_text(angle = 90,
608                                                             hjust = 1))
609                       + scale_color_manual( name = "Model",
610                                             values = c ("Gamma" = "blue",
611                                                         "Weibull" = "red"))))

```

Section C.2: R code for quantile estimation with MLE and MRR

Recall that we carried out a simulation study to compare the performance of the MLE and MRR methods in quantile estimation in Chapter 3. Following code can be used to recreate the results illustrated and discussed in Chapter 3. Note that for large sample sizes and number of iterations the programme may take a considerable time to produce the results.

```

1 #DESIGN
2 #Scale (beta: 100)
3 #Five different shapes (alpha: 0.5, 0.75, 1, 1.5, 3)
4 #Five different sample sizes (n: 50, 100, 1000)

```

```

5 #Six different failure rates (r: 1%, 10%, 50%, 70%, 90%, 100%)
6 # Estimates four different quantiles (q: 0.001, 0.1, 0.8, 0.9)
7 #Number of simulations: 1000
8
9 #Use the fitdistrplus package to estimate the parameters with MLE
10
11 #Load required packages
12 require(fitdistrplus)
13 require (ggplot2)
14 require (eha)
15 require (latex2exp)
16
17 #Simulate data
18 weibull.data <- function (samplesize, #sample size
19                             scale, #scale parameter
20                             shape,
21                             propFailed){
22   data.wei <- sort (rweibull (samplesize,
23                               shape = shape,
24                               scale = scale))
25   T.cen <- data.wei[max(2, ceiling(propFailed*samplesize))]
26   censored <- data.wei > T.cen
27   gen.fails <- c (data.wei[censored == FALSE],
28                  rep(NA,length(data.wei[censored == TRUE])))
29   rank.fails <- rank (gen.fails, ties.method = "first",
30                      na.last = "keep")
31   gen.fails [is.na(gen.fails)] <- T.cen
32   simul.data <- data.frame(sampldata = data.wei,
33                            Censored = censored,
34                            obs.fails = gen.fails,
35                            ranks = rank.fails)
36   return(simul.data)
37 }
38
39 #Define the MLE function
40 mle <- function (data, #Data
41                  p.int = c(6, 0.5), #Initial values of the parameters
42                  samplesize,
43                  censored) {

```

```

44 loglik <- function (theta){
45   #theta[1] <- scale - scale parameter of weibull distribution
46   #theta [2] <- shape - shape parameter
47   ll <- sum(ifelse(censored,
48                     suppressWarnings(pweibull(data, theta[2],
49                                                  theta[1],
50                                                  lower.tail = FALSE,
51                                                  log.p = TRUE))),
52             suppressWarnings(dweibull(data, theta[2],
53                                         theta[1],
54                                         log = TRUE))))
55   return (-ll)
56 }
57
58 #Optimise to find the ML estimates
59 estim1 <- optim (p.int, loglik,
60                 method = "BFGS",
61                 control = list(
62                   maxit = 10000,
63                   reltol = .Machine$double.eps),
64                 hessian = TRUE)
65 #Results
66 results <- estim1$par
67 return (results)
68 }
69
70 #Median Rank Regression
71 #Define the MRR function
72
73 #Adjusted ranks
74 ri <- function (sim.data) {
75   N <- nrow (sim.data)
76   r0 <- 0 #Initial rank
77   #Extract the ranks of failure data
78   sj <- sim.data$ranks[sim.data$censored == FALSE]
79   ri <- NULL
80
81   ri[1] <- r0 + (N + 1 - r0) / (N + 1 - (sj[1] - 1))
82

```

```

83   for (i in 2:length(sj)) {
84     #Adjusted ranks
85     ri [i] <- ri [i - 1] + (N + 1 - ri[i - 1]) / (N + 1 - (sj[i] - 1))
86   }
87   return (ri)
88 }
89
90 #Shape parameter
91 b1 <- function (sim.data) {
92
93   ti <- sim.data$obs.fails[sim.data$Censored == FALSE] #Failure times
94   yi <- log (ti) #Dependent
95   N <- nrow (sim.data)
96   n <- length(ti)
97   xi <- log (-log (1 - ((ri(sim.data) - 0.3) / (N + 0.4))))
98   #Explanatory
99   sigma.hat <- (n*sum (xi*yi) - sum(xi)*sum(yi)) /
100                (n*sum(xi^2) - (sum(xi))^2)
101   return (sigma.hat)
102 }
103
104 #Scale parameter
105 b0 <- function (sim.data) {
106
107   ti <- sim.data$obs.fails[sim.data$Censored == FALSE] #Failure times
108   yi <- log (ti) #Dependent
109   N <- nrow (sim.data)
110   n <- length(ti)
111   xi <- log (-log (1 - ((ri(sim.data) - 0.3) / (N + 0.4))))
112   #Explanatory
113
114   b1 <- b1(sim.data)
115   mu.hat <- (1/n)*(sum(yi) - b1*sum(xi))
116
117   return (c(exp (mu.hat), 1/b1))
118
119 }
120
121 test$obs.fails [test$Censored == FALSE]

```

[illegible]


```

200         results.table$scale.mle)
201
202 results.table$Q.MRR <- qweibull (results.table$p,
203                                results.table$shape.mrr,
204                                results.table$scale.mrr)
205
206
207 results.table$cost.MLE <- (Hweibull (results.table$Q.MLE,
208                                     results.table$shape.true,
209                                     results.table$scale.true)
210 + results.table$cost.ratio)/
211 results.table$Q.MLE
212
213 results.table$cost.MRR <- (Hweibull (results.table$Q.MRR,
214                                     results.table$shape.true,
215                                     results.table$scale.true)
216 + results.table$cost.ratio)/
217 results.table$Q.MRR
218
219 results.table$cost.true <- hweibull (results.table$Q.true,
220                                     results.table$shape.true,
221                                     results.table$scale.true)
222
223 results.table$bias.MLE <- abs (results.table$Q.true -
224                               results.table$Q.MLE)
225 results.table$bias.MRR <- abs (results.table$Q.true -
226                               results.table$Q.MRR)
227
228 results.table$mse.mle <- (results.table$Q.true -
229                          results.table$Q.MLE)^2/(10^3)
230 results.table$mse.mrr <- (results.table$Q.true -
231                          results.table$Q.MRR)^2/(10^3)
232
233 #results.table$MRR.new <- -pweibull(results.table$Q.MRR,
234                                   #results.table$shape.true,
235                                   #results.table$scale.true,
236                                   #lower = FALSE, log = TRUE)*100
237
238 #results.table$MLE.new <- -pweibull(results.table$Q.MLE,

```



```
278
279 results.mean <- with (as.data.frame(results.table),
280                        aggregate (
281                            x = list(Scale.MLE = scale.mle ,
282                                Shape.MLE = shape.mle,
283                                Shape.MRR = shape.mrr,
284                                Scale.MRR = scale.mrr,
285                                Q.true = Q.true,
286                                Q.MLE = Q.MLE,
287                                Q.MRR = Q.MRR,
288                                abs.bias.MLE = bias.MLE,
289                                abs.bias.MRR = bias.MRR,
290                                mse.MLE = mse.mle,
291                                mse.MRR = mse.mrr),
292                            by = list(Samplesize = samplesize,
293                                Scale = scale.true,
294                                Shape = shape.true,
295                                p = p),
296                            FUN = mean))
297
298 # results.mean
299 # View (results.mean)
300 # fix (results.mean)
301
302 results.quantile <- with (as.data.frame(results.table),
303                           aggregate (
304                               x = list(Scale.MLE = scale.mle ,
305                                   Shape.MLE = shape.mle,
306                                   Q.true = Q.true,
307                                   Q.MLE = Q.MLE,
308                                   Q.MRR = Q.MRR,
309                                   abs.bias.MLE = bias.MLE,
310                                   abs.bias.MRR = bias.MRR,
311                                   mse.MLE = mse.mle,
312                                   mse.MRR = mse.mrr),
313                               by = list(Samplesize = samplesize,
314                                   Scale = scale.true,
315                                   Shape = shape.true,
316                                   p = p),
```

```

317 FUN = quantile))
318
319 # fix (results.quantile)
320
321 results.sd <- with (as.data.frame(results.table),
322 aggregate (
323 x = list(Scale.MLE.SD = scale.mle ,
324 Shape.MLE.SD = shape.mle ,
325 Q.true.SD = Q.true ,
326 Q.MLE.SD = Q.MLE ,
327 Q.MRR.SD = Q.MRR ,
328 abs.bias.MLE.SD = bias.MLE ,
329 abs.bias.MRR.SD = bias.MRR ,
330 mse.MLE.SD = mse.mle ,
331 mse.MRR.SD = mse.mrr),
332 by = list(Samplesize = samplesize ,
333 Scale = scale.true ,
334 Shape = shape.true ,
335 p = p),
336 FUN = sd))
337
338 # fix (results.sd)
339
340 results <- cbind (results.mean, results.sd[c (8:13)],
341 bias.MLE.low = results.quantile$abs.bias.MLE[, 2],
342 bias.MRR.low = results.quantile$abs.bias.MRR[, 2],
343 bias.MLE.up = results.quantile$abs.bias.MLE[, 4],
344 bias.MRR.up = results.quantile$abs.bias.MRR[, 4],
345 mse.MLE.low = results.quantile$mse.MLE[, 2],
346 mse.MRR.low = results.quantile$mse.MRR[, 2],
347 mse.MLE.up = results.quantile$mse.MLE[, 4],
348 mse.mRR.up = results.quantile$mse.MRR[, 4])
349
350 # results$cost.MLE <- (100*hweibull (results$Q.true,
351 # results$Shape,
352 # results$Scale) -
353 # 100*hweibull (results$Q.MLE,
354 # results$Shape.MLE,
355 # results$Scale.MLE))

```



```

395 # View (results.w)
396
397 # fix(results.w)
398
399 # results.w$truevalue <- results$Q.true
400 # results.w$rel.bias <- (results.w$estimate - results.w$truevalue)*100
401 # results.w$rel.bias <- (results.w$estimate - results.w$truevalue)*100
402 # results.w$rel.bias.SD <- results.w$SD*100/ results.w$truevalue
403 # results.w$ci.bias <- results.w$rel.bias.SD*1.96 /
404 # results.w$ci.bias <- results.w$rel.bias.SD*1.96 /
405 # results.w$ci.mse<- results.w$MSE.SD*1.96 /
406 # results.w$ci.mse<- results.w$MSE.SD*1.96 /
407
408 pd <- position_dodge(0.01)
409
410 print (plot.bias <- (qplot (x = Shape,
411 y = Bias,
412 data = subset(results.w,
413 Parameter%in%c("Q.MLE",
414 "Q.MRR")),
415 group = factor(Samplesize),
416 colour = factor(Samplesize),
417 geom = c ("point", "line"),
418 xlab = "True Shape",
419 ylab = "Absolute Bias")
420 + facet_grid(p~Parameter, scales = "free_y",
421 labeller = labeller (
422 p = c("0.1" = "10th Quantile",
423 "0.8" = "80th Quantile"),
424 Parameter = c ("Q.MLE" = "MLE",
425 "Q.MRR" = "MRR")))
426 +labs(colour = "Units")))
427
428 print (plot.bias.CI <- (qplot (x = Shape,
429 y = Bias,
430 data = subset(results.w,
431 Parameter%in%c("Q.MLE",
432 "Q.MRR")),
433 group = factor(Samplesize),

```

```

434         colour = factor(Samplesize),
435         geom = c("point", "line"),
436         xlab = "True Shape",
437         ylab = "Absolute Bias")
438     + facet_grid(p~Parameter, scales = "free_y",
439                 labeller = labeller (
440         p = c("0.1" = "10th Quantile",
441             "0.8" = "80th Quantile"),
442         Parameter = c("Q.MLE" = "MLE",
443                       "Q.MRR" = "MRR")))
444     + geom_errorbar(aes(x = Shape,
445                         ymin = Bias.CI.Low,
446                         ymax = Bias.CI.Up),
447                     width= 0.01))
448     +labs(colour = "Units"))
449
450 print (plot.mse <- (qplot (x = Shape,
451                             y = MSE,
452                             data = subset(results.w,
453                                             Parameter%in%c("Q.MLE",
454                                                             "Q.MRR")),
455                             group = factor(Samplesize),
456                             colour = factor(Samplesize),
457                             geom = c("point", "line"),
458                             xlab = "True Shape",
459                             ylab = TeX ("$MSE \\times 10^3$"))
460     + facet_grid(p~Parameter,
461                 scales = "free_y",
462                 labeller = labeller (
463         p = c("0.1" = "10th Quantile",
464             "0.8" = "80th Quantile"),
465         Parameter = c("Q.MLE" = "MLE",
466                       "Q.MRR" = "MRR")))
467     + geom_errorbar(aes(x = Shape,
468                         ymin = MSE.CI.Low,
469                         ymax = MSE.CI.Up),
470                     width= 0.1))
471     +labs(colour = "Units"))
472

```

```

473 print (plot.mse <- (qplot (x = Shape,
474                             y = MSE,
475                             data = subset(results.w,
476                                           Parameter%in%c("Q.MLE",
477                                                         "Q.MRR")),
478                             group = factor(Samplesize),
479                             colour = factor(Samplesize),
480                             geom = c ("point", "line"),
481                             xlab = "True Shape",
482                             ylab = TeX ("$MSE \\times 10^3$"))
483 + facet_grid(p~Parameter,
484              scales = "free_y",
485              labeller = labeller (
486                p = c("0.1" = "10th Quantile",
487                      "0.8" = "80th Quantile"),
488                Parameter = c ("Q.MLE" = "MLE",
489                               "Q.MRR" = "MRR")))
490 + labs(colour = "Units"))
491
492 print (plot_cost_0.1 <- ggplot (data = subset (results.l,
493                                               p%in%c("0.1")),
494                                   aes (Cost, colour = factor(Method),
495                                       group = factor(Method)))
496   #+ geom_vline (data = subset(results,
497                               p%in%c("0.1")),
498                 aes (xintercept = Cost.true,
499                     group = factor(Shape)))
500 + geom_density(stat = "density", adjust = 5)
501 + facet_grid(samplesize~shape.true,
502              scales = "free",
503              labeller = labeller(
504                samplesize =
505                  c ("10" = "Sample size = 10",
506                    "50" = "Sample size = 50",
507                    "100" = "Samplesize = 100",
508                    "500" = "Sample size = 500",
509                    "1000" = "Samplesize = 1000"),
510                shape.true =
511                  c ("1.5" = "Shape = 1.5",

```

```

512         "3" = "Shape = 3",
513         "4" = "Shape = 4",
514         "6" = "Shape = 6"))))
515     + labs (colour = "Method")
516     + xlab ("Total expected cost rate")
517     + ylab ("Density")
518     + scale_color_manual
519       (labels = c ("MLE", "MRR"),
520        values = c ("red", "blue"))
521     + theme(axis.text.x =
522              element_text(angle = 90, hjust = 1)))
523
524 print (plot_cost_0.8 <- ggplot (data = subset (results.l,
525                                               p%in%c("0.8")),
526                                   aes (Cost,
527                                       colour = factor(Method),
528                                       group = factor(Method)))
529       + geom_density(stat = "density", adjust = 5)
530       + facet_grid(samplesize~shape.true,
531                   scales = "free",
532                   labeller = labeller(
533                     samplesize =
534                       c ("10" = "Sample size = 10",
535                         "50" = "Sample size = 50",
536                         "100" = "Samplesize = 100",
537                         "500" = "Sample size = 500",
538                         "1000" = "Samplesize = 1000"),
539                     shape.true =
540                       c ("1.5" = "Shape = 1.5",
541                         "3" = "Shape = 3",
542                         "4" = "Shape = 4",
543                         "6" = "Shape = 6"))))
544     + labs (colour = "Method")
545     + xlab ("Total expected cost rate")
546     + ylab ("Density")
547     + scale_color_manual (labels
548                           = c ("MLE", "MRR"),
549                           values = c ("red",
550                                       "blue"))

```

```

551 + theme(axis.text.x
552         = element_text(angle = 90,
553         hjust = 1)))
554
555
556 print (plot_ecdf_0.1 <- ggplot (data = subset (results.1,
557                                             p%in%c("0.1")),
558         aes (Cost, colour = factor(Method),
559         group = factor(Method)))
560 + stat_ecdf(geom = "step")
561 + facet_grid(samplesize~shape.true,
562             scales = "free_y",
563             labeller = labeller(
564             samplesize =
565             c ("10" = "Sample size = 10",
566             "50" = "Sample size = 50",
567             "100" = "Samplesize = 100",
568             "500" = "Sample size = 500",
569             "1000" = "Samplesize = 1000"),
570             shape.true =
571             c ("1.5" = "Shape = 1.5",
572             "3" = "Shape = 3",
573             "4" = "Shape = 4",
574             "6" = "Shape = 6")))
575
576 + labs (colour = "Method")
577 + xlab ("Total expected cost rate")
578 + ylab ("CDF")
579 + scale_color_manual
580     (labels = c ("MLE", "MRR"),
581     values = c ("red", "blue"))
582 + theme(axis.text.x =
583         element_text(angle = 90,
584         hjust = 1)))
585
586
587 print (plot_ecdf_0.8 <- ggplot (data = subset (results.1,
588                                             p%in%c("0.8")),
589         aes (Cost,
590         colour = factor(Method),
591         group = factor(Method)))

```

```

590       + stat_ecdf (geom = "step")
591       + facet_grid(samplesize~shape.true,
592                   labeller = labeller(
593                       samplesize =
594                         c ("10" = "Sample size = 10",
595                           "50" = "Sample size = 50",
596                           "100" = "Samplesize = 100",
597                           "500" = "Sample size = 500",
598                           "1000" = "Samplesize = 1000"),
599                       shape.true =
600                         c ("1.5" = "Shape = 1.5",
601                           "3" = "Shape = 3",
602                           "4" = "Shape = 4",
603                           "6" = "Shape = 6"))))
604       + labs (colour = "Method")
605       + xlab ("Total expected cost rate")
606       + ylab ("CDF")
607       + scale_color_manual
608         (labels = c ("MLE", "MRR"),
609          values = c ("red", "blue"))
610       + xlim (NA, 0.17)
611       + theme(axis.text.x =
612               element_text(angle = 90,
613                             hjust = 1)))

```

Section C.3: R code for Bayesian model averaging

Following code can be used to reproduce the results explained in Chapter 4.

```

1  require (rstan)
2  require (coda)
3  require (ggplot2)
4  require (reshape2)
5  require (FAdist)
6  require (statmod)
7  require (plyr)
8  require (mvtnorm)
9  #require (invgamma)
10 require (LaplacesDemon)

```

```

11 #####
12 weibull.data <- function (samplesize, #sample size
13                               propFailed){
14                               data.wei <- rep (c (8, 16, 32, 40, 56, 60,
15                                                    64, 72, 80, 96, 104,
16                                                    108, 112, 114,120,128,
17                                                    136, 152, 156, 160, 168,
18                                                    176, 184, 194, 208, 216,
19                                                    224, 232, 240, 246, 256,
20                                                    264, 272, 280, 288, 304,
21                                                    308, 328, 340,352, 358,
22                                                    360, 384, 392, 400, 424,
23                                                    438, 448, 464, 480, 536,
24                                                    552, 576, 608, 656, 716),
25                               times = c (1, 4, 2, 4, 3, 1, 1,
26                                           5, 4, 2, 1, 1, 2, 1,
27                                           1, 1, 1, 3, 1, 1, 5,
28                                           1, 3, 1, 2, 1, 4, 1,
29                                           1, 1, 1, 2, 1, 1, 1,
30                                           1, 1, 2, 1, 1, 1, 1,
31                                           1, 1, 1, 1, 1, 1, 1,
32                                           1, 1, 1, 1, 1, 1, 1,
33                                           1, 1, 1, 1, 1, 1, 1))
34   T.cen <- data.wei[max(2, ceiling(propFailed*samplesize))]
35   censored <- data.wei > T.cen
36   event <- c (rep(1, length(data.wei[censored == FALSE])),
37              rep(0, length(data.wei[censored == TRUE])))
38   gen.fails <- c (data.wei[censored == FALSE], rep(NA,
39              length(data.wei[censored == TRUE])))
40   rank.fails <- rank (gen.fails, ties.method = "first",
41                      na.last = "keep")
42   gen.fails [is.na(gen.fails)] <- T.cen
43   y.cen <- rep (T.cen, length(data.wei[censored == TRUE]))
44
45   simul.data <- data.frame(sampledata = data.wei,
46                             Censored = censored,
47                             obs.fails = gen.fails,
48                             ranks = rank.fails)
49

```

```

50   sim.data.bay <- list (sampledata = data.wei[censored == FALSE],
51                        obs_fails = gen.fails,
52                        Censoreddata = censored,
53                        Censored = event,
54                        J = length (data.wei[censored == FALSE]),
55                        N_cen = length(data.wei[censored == TRUE]),
56                        y_cen = y.cen)
57
58   return(sim.data.bay)
59 }
60
61 #####
62 #TEST THE DATA SIMULATION FUNCTION
63 (test<- (weibull.data (samplesize = 88,
64                       propFailed = 0.8)))
65 #####
66
67
68 #####
69 #GENERATE DATA WITH 75% FAILURES
70
71 data_75 <- (weibull.data(samplesize = 88,
72                          propFailed = 0.75))
73
74 #GENERATE DATA WITH 25% FAILURES
75
76 data_25 <- (weibull.data(samplesize = 88,
77                          propFailed = 0.25))
78
79
80 #####
81
82 #Define STAN models for the censored data
83
84 model_weibull <- "
85
86 data {
87   int<lower=0> J; // number of observations
88   int<lower=0> N_cen; // number of censored observations

```

```

89 vector [J]sampledata; // failure in hours
90 vector [N_cen]y_cen; //values of censored data
91 }
92
93 parameters {
94   real mu; // scale
95   real<lower=0> sigma; // shape
96 }
97
98 model {
99   //prior
100  mu ~ uniform (0, 20);
101  sigma ~ inv_gamma(0.1, 0.1);
102
103  //Likelihood and prior
104  sampledata ~ weibull (1/sigma, exp(mu)); //Uncensored data
105  increment_log_prob(weibull_ccdf_log (y_cen, 1/sigma, exp(mu)));
106                      //Censored data
107 }
108 "
109 mod_weibull <- stan_model(model_code = model_weibull)
110
111 #####
112
113 model_gamma <- "data {
114   int<lower=0> J; // number of observations
115   int<lower=0> N_cen; // number of censored observations
116   vector [J]sampledata; // failure in hours
117   vector [N_cen]y_cen; //values of censored data
118
119 }
120
121 parameters {
122   real<lower=0> alpha; // shape
123   real<lower=0> beta; // rate
124 }
125
126 model {
127   //define priors of alpha and beta
128   alpha ~ gamma(0.1, 0.1);

```

```

128 beta ~ gamma(0.1, 0.1);
129
130 //Likelihood of the data
131 sampledata ~ gamma(beta, alpha);
132 increment_log_prob(gamma_ccdf_log (y_cen, beta, alpha));
133 }"
134
135 mod_gamma <- stan_model (model_code = model_gamma)
136
137 #####
138
139 model_IG <- "functions {
140
141   real IG_log (vector x, real mu, real lambda){
142     vector [num_elements (x)] prob;
143     real lprob;
144     for (i in 1:num_elements(x)) {
145       prob[i] <- (lambda/(2*pi()*(x[i]^3)))^0.5*
146         exp(-lambda*(x[i] - mu)^2/(2*mu^2*x[i]));
147     }
148     lprob <- sum (log(prob));
149     return lprob;
150   }
151
152   real myIG_ccdf(vector x, real mu, real lambda){
153     vector [num_elements (x)] cprob;
154     real lp;
155     for (i in 1:num_elements(x)) {
156       vector[num_elements(x)] a;
157       vector[num_elements(x)] b;
158       a[i] <- ((x[i]/mu - 1)*sqrt(lambda/x[i]));
159       b[i] <- -sqrt(lambda/x[i])*(x[i]/mu + 1);
160       cprob[i] <- normal_cdf(a[i] , 0, 1) +
161         exp(2*lambda/mu)*normal_cdf(b[i], 0, 1);
162     }
163     lp <- sum(log(1 - cprob));
164     return lp;
165   }
166 }

```

```

167
168 data {
169   int<lower=0> J; // number of observations
170   int<lower=0> N_cen; // number of censored observations
171   vector [J]sampledata; // failure in hours
172   vector [N_cen]y_cen; //values of censored data
173 }
174
175 parameters {
176   real<lower=0> mu; // mean
177   real<lower=0> lambda; // shape
178
179 }
180
181 model {
182   //define priors of mu and lambda
183   mu ~ gamma (0.1, 0.1);
184   lambda ~ gamma (0.1, 0.1);
185
186   //Likelihood of the data
187   sampledata ~ IG(mu, lambda);
188   increment_log_prob(myIG_ccdf(y_cen, mu, lambda));
189 }
190 "
191 mod_IG <- stan_model(model_code = model_IG)
192
193 #####
194 model_llogistic <- "
195 data {
196   int<lower=0> J; // number of observations
197   int<lower=0> N_cen; // number of censored observations
198   vector [J]sampledata; // failure in hours
199   vector [N_cen]y_cen; //values of censored data
200 }
201
202 parameters {
203   real<lower=0> sigma; // scale
204   real<lower=0> mu; // location
205 }

```

```

206
207 model {
208 //define priors of alpha and beta
209 mu ~ uniform(0, 20);
210 sigma ~ inv_gamma(0.1, 0.1);
211
212 //Likelihood of the data
213 log(sampleddata) ~ logistic(mu, sigma);
214 increment_log_prob(logistic_ccdf_log (log (y_cen), mu, sigma));
215 }"
216
217 mod_llogistic <- stan_model(model_code = model_llogistic)
218
219 #####
220
221 model_lognormal <- "
222 data {
223 int<lower=0> J; // number of observations
224 int<lower=0> N_cen; // number of censored observations
225 vector [J]sampledata; // failure in hours
226 vector [N_cen]y_cen; //values of censored data
227 }
228
229 parameters {
230 real<lower=0> sigma; // scale
231 real<lower=0> mu; // location
232 }
233
234 model {
235 //define priors of alpha and beta
236 mu ~ uniform(0, 20);
237 sigma ~ inv_gamma(0.1, 0.1);
238
239 //Likelihood of the data
240 log(sampleddata) ~ normal(mu, sigma);
241 increment_log_prob(normal_ccdf_log (log (y_cen), mu, sigma));
242 }"
243
244 mod_lognormal <- stan_model(model_code = model_lognormal)

```

```
245
246 #####
247 #Run the models to get the posterior distributions
248
249 #Censored data - 75% Fail
250
251 #Weibull distribution
252 set.seed(12365)
253 weibull_75 <- sampling (mod_weibull, data = data_75,
254                        chains = 1, iter = 10^6,
255                        warmup = 40000, thin = 20)
256 weibull_75
257 summary (weibull_75)
258
259 set.seed(12365)
260 weibull_25 <- sampling (mod_weibull, data = data_25,
261                        chains = 1, iter = 10^6,
262                        warmup = 40000, thin = 20)
263 weibull_25
264
265 #####
266 #Inverse-Gaussian distribution
267 set.seed(12456)
268 IG_75 <- sampling (mod_IG, data = data_75, chains = 1,
269                  iter = 10^6, warmup = 40000, thin = 20)
270 IG_75
271
272 set.seed(12456)
273 IG_25 <- sampling (mod_IG, data = data_25, chains = 1,
274                  iter = 10^6, warmup = 40000, thin = 20)
275 IG_25
276
277 #####
278
279 #Gamma distribution
280 set.seed(12569)
281 gamma_75 <- sampling (mod_gamma, data = data_75,
282                     chains = 1, iter = 10^4,
283                     warmup = 400, thin = 20)
```

[illegible]

```

323                                     (weibull_75)[,x,]))))
324 weibullcoda_25 <- mcmc.list(lapply(1:ncol(weibull_75),
325                                     function(x) {mcmc(as.array
326                                     (weibull_25)[,x,]))))
327
328 gammacoda_75 <- gamma_75$trace[1:48000, ]
329 gammacoda_25 <- mcmc.list(lapply(1:ncol(gamma_25),
330                                     function(x){mcmc(as.array
331                                     (gamma_25)[,x,]))))
332
333 IGcoda_75 <- mcmc.list(lapply(1:ncol(IG_75),
334                                     function(x) {mcmc(as.array
335                                     (IG_75)[,x,]))))
336 IGcoda_25 <- mcmc.list(lapply(1:ncol(IG_25),
337                                     function(x) {mcmc(as.array
338                                     (IG_25)[,x,]))))
339
340 llogcoda_75 <- mcmc.list(lapply(1:ncol(llogistic_75),
341                                     function(x) {mcmc(as.array
342                                     (llogistic_75)[,x,]))))
343 llogcoda_25 <- mcmc.list(lapply(1:ncol(llogistic_25),
344                                     function(x) {mcmc(as.array
345                                     (llogistic_25)[,x,]))))
346
347 lnormcoda_75 <- mcmc.list(lapply(1:ncol(lognormal_75),
348                                     function(x) {mcmc(as.array
349                                     (lognormal_75)[,x,]))))
350 lnormcoda_25 <- mcmc.list(lapply(1:ncol(lognormal_25),
351                                     function(x) {mcmc(as.array
352                                     (lognormal_25)[,x,]))))
353
354 #####
355
356 #Maximum Likelihood Estimation
357
358 mle_weibull <- function (data, #Data
359                             p.int = c(6, 0.5), #Initial values of the parameters
360                             samplesize,
361                             censored) {

```

```

362     loglik <- function (theta){
363       #theta[1] <- scale -
364       scale parameter of weibull distribution
365       #theta [2] <- shape -
366       shape parameter
367       ll <- sum(ifelse(censored,
368         suppressWarnings(pweibull(data, 1/theta[2],
369           exp(theta[1]),
370             lower.tail = FALSE,
371             log.p = TRUE)),
372         suppressWarnings(dweibull(data, 1/theta[2],
373           exp(theta[1]),
374             log = TRUE))))
375       return (-ll)
376     }
377
378   #Optimise to find the ML estimates
379   estim1 <- optim (p.int, loglik,
380     method = "BFGS",
381     control = list(
382       maxit = 10000,
383       reltol = .Machine$double.eps),
384     hessian = TRUE)
385   #Results
386   results <- estim1
387   return (results)
388 }
389
390 #####
391 mle_gamma <- function (data, #Data
392   p.int = c(6, 0.5),
393   #Initial values of the parameters
394   samplesize,
395   censored) {
396   loglik <- function (theta){
397     #theta[1] <- scale -
398     scale parameter of weibull distribution
399     #theta [2] <- shape - shape parameter
400     ll <- sum(ifelse(censored,

```



```

440     return (-ll)
441   }
442
443   #Optimise to find the ML estimates
444   estim1 <- optim (p.int, loglik,
445                   method = "BFGS",
446                   control = list(
447                     maxit = 10000,
448                     reltol = .Machine$double.eps),
449                   hessian = TRUE)
450
451   #Results
452   results <- estim1
453   return (results)
454 }
455 #####
456 mle_llog <- function (data, #Data
457                       p.int = c(6, 0.5),
458                       #Initial values of the parameters
459                       samplesize,
460                       censored) {
461   loglik <- function (theta){
462     #theta[1] <- scale -
463     scale parameter of weibull distribution
464     #theta [2] <- shape - shape parameter
465     ll <- sum(ifelse(censored,
466                     suppressWarnings(pllog(data, theta[1],
467                                             theta[2],
468                                             lower.tail = FALSE,
469                                             log.p = TRUE))),
470             suppressWarnings(dllog(data, theta[1],
471                                     theta[2],
472                                     log = TRUE))))
473     return (-ll)
474   }
475
476   #Optimise to find the ML estimates
477   estim1 <- optim (p.int, loglik,
478                   method = "BFGS",

```

```

479         control = list(
480             maxit = 10000,
481             reltol = .Machine$double.eps),
482         hessian = TRUE)
483 #Results
484 results <- estim1
485 return (results)
486 }
487
488 #####
489
490 mle_lnorm <- function (data, #Data
491                         p.int = c(6, 0.5),
492                         #Initial values of the parameters
493                         samplesize,
494                         censored) {
495     loglik <- function (theta){
496         #theta[1] <- scale -
497         scale parameter of weibull distribution
498         #theta [2] <- shape - shape parameter
499         ll <- sum(ifelse(censored,
500                         suppressWarnings(plnorm(data, theta[1],
501                                                  theta[2],
502                                                  lower.tail = FALSE,
503                                                  log.p = TRUE))),
504                 suppressWarnings(dlnorm(data,
505                                         theta[1], theta[2],
506                                         log = TRUE))))
507         return (-ll)
508     }
509
510 #Optimise to find the ML estimates
511 estim1 <- optim (p.int, loglik,
512                 method = "BFGS",
513                 control = list(
514                     maxit = 10000,
515                     reltol = .Machine$double.eps),
516                 hessian = TRUE)
517 #Results

```

[illegible]

```

557         censored = Censoreddata))
558
559 mle_gamma_25 <- with (data_25,
560         mle_gamma(data = obs_fails,
561         p.int = c(1, 0.5),
562         samplesize = length (obs_fails),
563         censored = Censoreddata))
564
565 mle_invg_25 <- with (data_25,
566         mle_invgaussian(data = obs_fails,
567         p.int = c(1, 0.5),
568         samplesize = length (obs_fails),
569         censored = Censoreddata))
570
571 mle_llog_25 <- with (data_25,
572         mle_llog(data = obs_fails,
573         p.int = c(1, 0.5),
574         samplesize = length (obs_fails),
575         censored = Censoreddata))
576
577 mle_lnorm_25 <- with (data_25,
578         mle_lnorm(data = obs_fails,
579         p.int = c(1, 0.5),
580         samplesize = length (obs_fails),
581         censored = Censoreddata))
582
583 #####
584
585 ##### BAYESIAN MODEL AVERAGING #####
586
587 #CREATE A VECTOR WITH THE LIKELIHOOD VALUES AT THE ESTIMATED PARAMETER
    VALUES
588
589 likelihood_75 <- c (mle_weib_75$value,
590         mle_gamma_75$value,
591         mle_invg_75$value,
592         mle_llog_75$value,
593         mle_lnorm_75$value)
594

```

```

595 likelihood_25 <- c (mle_weib_25$value,
596                     mle_gamma_25$value,
597                     mle_invg_25$value,
598                     mle_llog_25$value,
599                     mle_lnorm_25$value)
600
601 #COMPUTE AIC VALUES
602
603 k <- 2 #Number of estimated parameters
604
605 aic_75 <- 2*likelihood_75 + 2*k
606
607 aic_25 <- 2*likelihood_25 + 2*k
608
609 #####
610 #COMPUTE BIC VALUES
611
612 n_75 <- with (data_75, length (sampledata))
613 n_25 <- with (data_25, length (sampledata))
614
615 bic_75 <- 2*likelihood_75 + k*log (n_75)
616
617 bic_25 <- 2*likelihood_25 + k*log (n_25)
618
619 #####
620 #COMPUTE DIC VALUES
621
622 posterior_df_75 <- data.frame (mu.weib = c(weibullcoda_75[[1]][, 1]),
623                                sigma.weib = c(weibullcoda_75[[1]][, 2]),
624                                alpha = c(gammacoda_75[, 2]),
625                                beta = c(gammacoda_75[, 1]),
626                                mu.IG = c(IGcoda_75[[1]][, 1]),
627                                lambda.IG = c(IGcoda_75[[1]][, 2]),
628                                mu.llog = c(llogcoda_75[[1]][, 2]),
629                                sigma.llog = c(llogcoda_75[[1]][, 1]),
630                                mu.lnorm = c(lnormcoda_75[[1]][, 2]),
631                                sigma.lnorm = c(lnormcoda_75[[1]][, 1]))
632
633 posterior_df_25 <- data.frame (mu.weib = c(weibullcoda_25[[1]][, 1]),

```

```

634         sigma.weib = c(weibullcoda_25[[1]][,2]),
635         alpha = c(gammacoda_25[[1]][, 1]),
636         beta = c(gammacoda_25[[1]][, 2]),
637         mu.IG = c(IGcoda_25[[1]][, 1]),
638         lambda.IG = c(IGcoda_25[[1]][, 2]),
639         mu.llog = c(llogcoda_25[[1]][, 2]),
640         sigma.llog = c(llogcoda_25[[1]][, 1]),
641         mu.lnorm = c(lnormcoda_25[[1]][, 2]),
642         sigma.lnorm = c(lnormcoda_25[[1]][, 1]))
643
644 mean_posterior_75 <- apply (posterior_df_75, 2, mean)
645 mean_posterior_25 <- apply (posterior_df_25, 2, mean)
646
647 #Compute the deviances
648
649 dev_wei_75 <- NULL
650 dev_gamma_75 <- NULL
651 dev_lnorm_75 <- NULL
652 dev_llogis_75 <- NULL
653 dev_invga_75 <- NULL
654
655 for (i in 1:nrow (posterior_df_75)) {
656   dev_wei_75[i] <- -2*with(data_75,
657     sum(ifelse(Censoreddata,
658       suppressWarnings(pweibull(obs_fails,
659         1/posterior_df_75$sigma.weib[i],
660         exp(posterior_df_75$mu.weib[i]),
661         lower.tail = FALSE, log.p = TRUE)),
662       suppressWarnings(dweibull(obs_fails,
663         1/posterior_df_75$sigma.weib[i],
664         exp(posterior_df_75$mu.weib[i]),
665         log = TRUE))))))
666
667   dev_gamma_75[i] <- -2*with(data_75,
668     sum(ifelse(Censoreddata,
669       suppressWarnings(pgamma(obs_fails,
670         posterior_df_75$beta[i],
671         posterior_df_75$alpha[i],
672         lower.tail = FALSE,

```

```

673         log.p = TRUE)),
674         suppressWarnings(dgamma(obs_fails,
675         posterior_df_75$beta[i],
676         posterior_df_75$alpha[i],
677         log = TRUE))))))
678
679 dev_lnorm_75[i] <- -2*with(data_75,
680         sum(ifelse(Censoreddata,
681         suppressWarnings(plnorm(obs_fails,
682         posterior_df_75$mu.lnorm[i],
683         posterior_df_75$sigma.lnorm[i],
684         lower.tail = FALSE,
685         log.p = TRUE)),
686         suppressWarnings(dlnorm(obs_fails,
687         posterior_df_75$mu.lnorm[i],
688         posterior_df_75$sigma.lnorm[i],
689         log = TRUE))))))
690
691 dev_llogis_75[i] <- -2*with(data_75,
692         sum(ifelse(Censoreddata,
693         suppressWarnings(pllog(obs_fails,
694         posterior_df_75$sigma.llog[i],
695         posterior_df_75$mu.llog[i],
696         lower.tail = FALSE,
697         log.p = TRUE)),
698         suppressWarnings(dllog(obs_fails,
699         posterior_df_75$sigma.llog[i],
700         posterior_df_75$mu.llog[i],
701         log = TRUE))))))
702
703 dev_invgauss_75[i] <- -2*with(data_75,
704         sum(ifelse(Censoreddata,
705         suppressWarnings(pinvgauss(obs_fails,
706         posterior_df_75$mu.IG[i],
707         posterior_df_75$lambda.IG[i],
708         lower.tail = FALSE,
709         log.p = TRUE)),
710         suppressWarnings(dinvgauss(obs_fails,
711         posterior_df_75$mu.IG[i],

```



```

751     posterior_df_25$sigma.lnorm[i],
752     lower.tail = FALSE,
753     log.p = TRUE)),
754     suppressWarnings(dlnorm(obs_fails,
755     posterior_df_25$mu.lnorm[i],
756     posterior_df_25$sigma.lnorm[i],
757     log = TRUE))))))
758
759     dev_llogis_25[i] <- -2*with(data_25,
760     sum(ifelse(Censoreddata,
761     suppressWarnings(pllog(obs_fails,
762     posterior_df_25$sigma.llog[i],
763     posterior_df_25$mu.llog[i],
764     lower.tail = FALSE,
765     log.p = TRUE)),
766     suppressWarnings(dllog(obs_fails,
767     posterior_df_25$sigma.llog[i],
768     posterior_df_25$mu.llog[i],
769     log = TRUE))))))
770
771     dev_invg_25[i] <- -2*with(data_25,
772     sum(ifelse(Censoreddata,
773     suppressWarnings(pinvgauss(obs_fails,
774     posterior_df_25$mu.IG[i],
775     posterior_df_25$lambda.IG[i],
776     lower.tail = FALSE,
777     log.p = TRUE)),
778     suppressWarnings(dinvgauss(obs_fails,
779     posterior_df_25$mu.IG[i],
780     posterior_df_25$lambda.IG[i],
781     log = TRUE))))))
782 }
783
784 #Mean of the deviances
785
786 df_deviance_75 <- data.frame (weib = sort (dev_wei_75),
787     gamma = sort (dev_gamma_75),
788     IG = sort (dev_invg_75),
789     llog = sort (dev_llogis_75),

```

```

790         lnorm = sort (dev_lnorm_75))
791
792 df_deviance_25 <- data.frame (weib = sort (dev_wei_25),
793                               gamma = sort (dev_gamma_25),
794                               IG = sort (dev_invg_25),
795                               llog = sort (dev_llogis_25),
796                               lnorm = sort (dev_lnorm_25))
797
798 mean_dev_75 <- apply (df_deviance_75, 2, mean)
799 mean_dev_25 <- apply (df_deviance_25, 2, mean)
800
801 #Deviance of the posterior mean
802
803 mdev_wei_75 <- -2*with(data_75,
804                        sum(ifelse(Censoreddata,
805                                  suppressWarnings(pweibull(obs_fails,
806                                                            1/mean_posterior_75[2],
807                                                            exp(mean_posterior_75[1]),
808                                                            lower.tail = FALSE,
809                                                            log.p = TRUE))),
810                              suppressWarnings(dweibull(obs_fails,
811                                                            1/mean_posterior_75[2],
812                                                            exp(mean_posterior_75[1]),
813                                                            log = TRUE)))))
814
815 mdev_gamma_75 <- -2*with(data_75,
816                          sum(ifelse(Censoreddata,
817                                    suppressWarnings(pgamma(obs_fails,
818                                                              mean_posterior_75[4],
819                                                              mean_posterior_75[3],
820                                                              lower.tail = FALSE,
821                                                              log.p = TRUE))),
822                              suppressWarnings(dgamma(obs_fails,
823                                                         mean_posterior_75[4],
824                                                         mean_posterior_75[3],
825                                                         log = TRUE)))))
826
827 mdev_lnorm_75 <- -2*with(data_75,
828                          sum(ifelse(Censoreddata,

```

```

829         suppressWarnings(plnorm(obs_fails,
830             mean_posterior_75[9],
831             mean_posterior_75[10],
832             lower.tail = FALSE,
833             log.p = TRUE)),
834         suppressWarnings(dlnorm(obs_fails,
835             mean_posterior_75[9],
836             mean_posterior_75[10],
837             log = TRUE))))))
838
839 mdev_llogis_75 <- -2*with(data_75,
840     sum(ifelse(Censoreddata,
841         suppressWarnings(pllog(obs_fails,
842             mean_posterior_75[8],
843             mean_posterior_75[7],
844             lower.tail = FALSE,
845             log.p = TRUE)),
846         suppressWarnings(dllog(obs_fails,
847             mean_posterior_75[8],
848             mean_posterior_75[7],
849             log = TRUE))))))
850
851 mdev_invg_75 <- -2*with(data_75,
852     sum(ifelse(Censoreddata,
853         suppressWarnings(pinvgauss(obs_fails,
854             mean_posterior_75[5],
855             mean_posterior_75[6],
856             lower.tail = FALSE,
857             log.p = TRUE)),
858         suppressWarnings(dinvgauss(obs_fails,
859             mean_posterior_75[5],
860             mean_posterior_75[6],
861             log = TRUE))))))
862
863 mdev_wei_25 <- -2*with(data_25,
864     sum(ifelse(Censoreddata,
865         suppressWarnings(pweibull(obs_fails,
866             1/mean_posterior_25[2],
867             exp(mean_posterior_25[1]),

```

```

868         lower.tail = FALSE,
869         log.p = TRUE)),
870         suppressWarnings(dweibull(obs_fails,
871         1/mean_posterior_25[2],
872         exp(mean_posterior_25[1]),
873         log = TRUE))))))
874
875 mdev_gamma_25 <- -2*with(data_25,
876         sum(ifelse(Censoreddata,
877         suppressWarnings(pgamma(obs_fails,
878         mean_posterior_25[4],
879         mean_posterior_25[3],
880         lower.tail = FALSE,
881         log.p = TRUE)),
882         suppressWarnings(dgamma(obs_fails,
883         mean_posterior_25[4],
884         mean_posterior_25[3],
885         log = TRUE))))))
886
887 mdev_lnorm_25 <- -2*with(data_25,
888         sum(ifelse(Censoreddata,
889         suppressWarnings(plnorm(obs_fails,
890         mean_posterior_25[9],
891         mean_posterior_25[10],
892         lower.tail = FALSE,
893         log.p = TRUE)),
894         suppressWarnings(dlnorm(obs_fails,
895         mean_posterior_25[9],
896         mean_posterior_25[10],
897         log = TRUE))))))
898
899 mdev_llogis_25 <- -2*with(data_25,
900         sum(ifelse(Censoreddata,
901         suppressWarnings(pllog(obs_fails,
902         mean_posterior_25[8],
903         mean_posterior_25[7],
904         lower.tail = FALSE,
905         log.p = TRUE)),
906         suppressWarnings(dllog(obs_fails,

```

```

907         mean_posterior_25[8],
908         mean_posterior_25[7],
909         log = TRUE))))))
910
911 mdev_invg_25 <- -2*with(data_25,
912         sum(ifelse(Censoreddata,
913         suppressWarnings(pinvgauss(obs_fails,
914         mean_posterior_25[5],
915         mean_posterior_25[6],
916         lower.tail = FALSE,
917         log.p = TRUE))),
918         suppressWarnings(dinvgauss(obs_fails,
919         mean_posterior_25[5],
920         mean_posterior_25[6],
921         log = TRUE))))))
922
923
924 #Deviance at the mean
925 deviance_mean_75 <- c (mdev_wei_75, mdev_gamma_75,
926         mdev_invg_75, mdev_llogis_75,
927         mdev_lnorm_75)
928
929 deviance_mean_25 <- c (mdev_wei_25, mdev_gamma_25,
930         mdev_invg_25, mdev_llogis_25,
931         mdev_lnorm_25)
932
933 #Calculate P_D for the censored data
934
935 pd_75 <- mean_dev_75 - deviance_mean_75
936
937 pd_25 <- mean_dev_25 - deviance_mean_2
938
939 #DEVIANCE INFORMATION CRITERIA
940
941 dic_75 <- deviance_mean_75 + 2*pd_75
942
943 dic_25 <- deviance_mean_25 + 2*pd_25
944
945 #####

```

```

947 ##### IMPORTANCE SAMPLING #####
948
949 ##Weibull distribution
950 posterior_wei <- function (mu, sigma) {
951   mu.min = 0
952   mu.max = 20
953   alpha = 0.1
954   beta = 0.1
955
956   prior_mu <- 1 / (mu.max - mu.min)
957   prior_sigma <- dinvgamma(sigma, shape = alpha, scale = beta)
958   joint_prior <- log (prior_mu) + log (prior_sigma)
959   llikelihood <- with(data_75, sum(ifelse(Censoreddata,
960     suppressWarnings(pweibull(obs_fails,
961       1/sigma, exp(mu),
962       lower.tail = FALSE,
963       log.p = TRUE)),
964     suppressWarnings(dweibull(obs_fails,
965       1/sigma, exp(mu),
966       log = TRUE))))))
967
968   posterior <- exp(llikelihood + joint_prior)
969 }
970
971 #Gamma distribution
972 posterior_gamma <- function (alpha, beta) {
973   #alpha - rate
974   #beta - shape
975   a = 0.1 #shape of hyper distribution
976   b = 0.1 #rate of hyper distribution
977
978   prior_shape <- dgamma (beta, shape = a, rate = b)
979   prior_rate <- dgamma (alpha, shape = a, rate = b)
980   joint_prior <- log (prior_shape) + log (prior_rate)
981   llikelihood <- with(data_75, sum(ifelse(Censoreddata,
982     suppressWarnings(pgamma(obs_fails,
983       beta, alpha,
984       lower.tail = FALSE

```

```

985         log.p = TRUE)),
986         suppressWarnings(dgamma(obs_fails,
987         beta, alpha,
988         log = TRUE))))))
989 posterior <- exp(llikelihood + joint_prior)
990
991 }
992
993 #Log-normal distribution
994 posterior_lnorm <- function (mu, sigma) {
995     mu.min = 0
996     mu.max = 20
997     alpha = 0.1
998     beta = 0.1
999
1000    prior_mu <- 1 / (mu.max - mu.min)
1001    prior_sigma <- dinvgamma(sigma, shape = alpha, scale = beta)
1002    joint_prior <- log (prior_mu*prior_sigma)
1003    llikelihood <- with(data_75, sum(ifelse(Censoreddata,
1004        suppressWarnings(plnorm(obs_fails,
1005        mu, sigma,
1006        lower.tail = FALSE,
1007        log.p = TRUE)),
1008        suppressWarnings(dlnorm(obs_fails,
1009        mu, sigma,
1010        log = TRUE))))))
1011
1012    posterior <- exp(llikelihood + joint_prior)
1013 }
1014
1015
1016 #Log-logistic distribution
1017 posterior_llogis <- function (mu, sigma) {
1018     mu.min = 0
1019     mu.max = 20
1020     alpha = 0.1
1021     beta = 0.1
1022
1023    prior_mu <- 1 / (mu.max - mu.min)

```

```

1024 prior_sigma <- dinvgamma(sigma, shape = alpha, scale = beta)
1025 joint_prior <- log (prior_mu*prior_sigma)
1026 llikelihood <- with(data_75, sum(ifelse(Censoreddata,
1027                                     suppressWarnings(pllog(obs_fails,
1028                                     sigma, mu,
1029                                     lower.tail = FALSE,
1030                                     log.p = TRUE))),
1031                                     suppressWarnings(dllog(obs_fails,
1032                                     sigma, mu,
1033                                     log = TRUE)))))
1034
1035 posterior <- exp (llikelihood + joint_prior)
1036 }
1037
1038 #Inverse-Gaussian distribution
1039
1040 posterior_invg <- function (mu, lambda){
1041   alpha <- 0.1
1042   beta <- 0.1
1043   prior_mu <- dgamma (mu, alpha, beta)
1044   prior_lambda <- dgamma(lambda, alpha, beta)
1045   joint_prior <- log(prior_mu*prior_lambda)
1046   llikelihood <- with(data_75, sum(ifelse(Censoreddata,
1047                                     suppressWarnings(pinvgauss(obs_fails,
1048                                     mean = mu,
1049                                     shape = lambda,
1050                                     lower.tail = FALSE,
1051                                     log.p = TRUE))),
1052                                     suppressWarnings(dinvgauss(obs_fails,
1053                                     mean = mu,
1054                                     shape = lambda,
1055                                     log = TRUE)))))
1056   posterior <- exp (llikelihood + joint_prior)
1057 }
1058
1059 #Draw random variates from multivariate Gaussian density
1060 #Weibull distribution
1061 #Mean vector
1062 mean_wei_75 <- c (mean(posterior_df_75$mu.weib),

```



```

1102
1103 #Log-normal distribution
1104 #Mean vector
1105 mean_lnorm_75 <- c (mean(posterior_df_75$mu.lnorm),
1106                    mean (posterior_df_75$sigma.lnorm))
1107 mean_lnorm_25 <- c (mean(posterior_df_25$mu.lnorm),
1108                    mean (posterior_df_25$sigma.lnorm))
1109 #Covariance matrix
1110 cvar_lnorm_75 <- cov (matrix (c(posterior_df_75$mu.lnorm,
1111                                posterior_df_75$sigma.lnorm),
1112                                nrow = length(posterior_df_75$mu.lnorm)
1113                                ,
1114                                byrow = FALSE))
1115 cvar_lnorm_25 <- cov (matrix (c(posterior_df_25$mu.lnorm,
1116                                posterior_df_25$sigma.lnorm),
1117                                nrow = length(posterior_df_25$mu.lnorm)
1118                                ,
1119                                byrow = FALSE))
1120
1121 set.seed(4562)
1122 mvnorm_lnorm_75 <- rmvnorm (n, mean = mean_lnorm_75,
1123                             sigma = cvar_lnorm_75)
1124 mvnorm_lnorm_25 <- rmvnorm (n, mean = mean_lnorm_25,
1125                             sigma = cvar_lnorm_25)
1126
1127 #Log-logistic distribution
1128 #Mean vector
1129 mean_llogis_75 <- c (mean(posterior_df_75$mu.llog),
1130                    mean(posterior_df_75$sigma.llog))
1131 mean_llogis_25 <- c (mean(posterior_df_25$mu.llog),
1132                    mean(posterior_df_25$sigma.llog))
1133 #Covariance matrix
1134 cvar_llogis_75 <- cov (matrix (c (posterior_df_75$mu.llog,
1135                                posterior_df_75$sigma.llog),
1136                                nrow=length(posterior_df_75$mu.llog),
1137                                byrow = FALSE))
1138 cvar_llogis_25 <- cov (matrix (c (posterior_df_25$mu.llog,
1139                                posterior_df_25$sigma.llog),
1140                                nrow=length(posterior_df_25$mu.llog),

```

```

1139         byrow = FALSE))
1140
1141 set.seed(7895)
1142 mvnnorm_llogis_75 <- rmvnorm (n, mean = mean_llogis_75,
1143                               sigma = cvar_llogis_75)
1144 mvnnorm_llogis_25 <- rmvnorm (n, mean = mean_llogis_25,
1145                               sigma = cvar_llogis_25)
1146
1147
1148 #Inverse-Gaussian Distribution
1149 #Mean vector
1150 mean_invg_75 <- c (mean(posterior_df_75$mu.IG),
1151                   mean(posterior_df_75$lambda.IG))
1152 mean_invg_25 <- c (mean(posterior_df_25$mu.IG),
1153                   mean(posterior_df_25$lambda.IG))
1154 #Covariance matrix
1155 cvar_invg_75 <- cov (matrix (c (posterior_df_75$mu.IG,
1156                                posterior_df_75$lambda.IG),
1157                              nrow = length(posterior_df_75$mu.IG),
1158                              byrow = FALSE))
1159 cvar_invg_25 <- cov (matrix (c (posterior_df_25$mu.IG,
1160                                posterior_df_25$lambda.IG),
1161                              nrow = length (posterior_df_25$mu.IG),
1162                              byrow = FALSE))
1163
1164 set.seed(1259)
1165 mvnnorm_invg_75 <- rmvnorm(n, mean = mean_invg_75,
1166                             sigma = cvar_invg_75)
1167 mvnnorm_invg_25 <- rmvnorm(n, mean = mean_invg_25,
1168                             sigma = cvar_invg_25)
1169
1170
1171 f_wei_75 <- NULL
1172 f_gamma_75 <- NULL
1173 f_lnorm_75 <- NULL
1174 f_llogis_75 <- NULL
1175 f_invg_75 <- NULL
1176 g_wei_75 <- NULL
1177 g_gamma_75 <- NULL

```



```

1217
1218 is_IL_wei <- sum (f_wei_75/g_wei_75)/n
1219 is_IL_gamma <- sum (f_gamma_75/g_gamma_75)/n
1220 is_IL_lnorm <- sum (f_lnorm_75/g_lnorm_75)/n
1221 is_IL_llogis <- sum (f_llogis_75/g_llogis_75)/n
1222 is_IL_invg <- sum (f_invg_75/g_invg_75)/n
1223
1224 evidence_IS <- c (is_IL_wei, is_IL_gamma, is_IL_lnorm,
1225                  is_IL_llogis, is_IL_invg)
1226
1227 is_sum_IL <- sum (is_IL_wei, is_IL_gamma, is_IL_lnorm,
1228                  is_IL_llogis, is_IL_invg)
1229
1230 is_IL_wei/is_sum_IL
1231 is_IL_gamma/is_sum_IL
1232 is_IL_lnorm/is_sum_IL
1233 is_IL_llogis/is_sum_IL
1234 is_IL_invg/ is_sum_IL
1235
1236 sum(is_IL_wei/is_sum_IL,is_IL_gamma/is_sum_IL,
1237     is_IL_lnorm/is_sum_IL,is_IL_llogis/is_sum_IL)
1238
1239 #Express the evidence in -2*log scale for comparison purposes
1240
1241 evi_IS_log <- -2*log (evidence_IS)
1242
1243 f_wei_25 <- NULL
1244 f_gamma_25 <- NULL
1245 f_lnorm_25 <- NULL
1246 f_llogis_25 <- NULL
1247 f_invg_25 <- NULL
1248 g_wei_25 <- NULL
1249 g_gamma_25 <- NULL
1250 g_lnorm_25 <- NULL
1251 g_llogis_25 <- NULL
1252 g_invg_25 <- NULL
1253 for (i in 1:n) {
1254   f_wei_25[i] <- posterior_wei (mvnorm_wei_25[i, 1],
1255                                mvnorm_wei_25[i, 2])

```

```

1256 g_wei_25[i] <- dmnorm(x = c (mvnorm_wei_25[i, 1],
1257                               mvnorm_wei_25[i, 2]),
1258                       mean = mean_wei_25,
1259                       sigma = cvar_wei_25)
1260
1261 f_gamma_25[i] <- posterior_gamma( mvnorm_gamma_25[i, 2],
1262                                   mvnorm_gamma_25[i, 1])
1263 g_gamma_25[i] <- dmnorm(x = c (mvnorm_gamma_25[i, 1],
1264                               mvnorm_gamma_25[i, 2]),
1265                           mean = mean_gamma_25,
1266                           sigma = cvar_gamma_25)
1267
1268 f_lnorm_25[i] <- posterior_lnorm (mvnorm_lnorm_25[i, 1],
1269                                   mvnorm_lnorm_25[i, 2])
1270 g_lnorm_25[i] <- dmnorm (x = c (mvnorm_lnorm_25[i, 1],
1271                                   mvnorm_lnorm_25[i, 2]),
1272                           mean = mean_lnorm_25,
1273                           sigma = cvar_lnorm_25)
1274
1275 f_llogis_25[i] <- posterior_llogis(mvnorm_llogis_25[i, 1],
1276                                    mvnorm_llogis_25[i, 2])
1277 g_llogis_25[i] <- dmnorm (x = c (mvnorm_llogis_25[i, 1],
1278                                    mvnorm_llogis_25[i, 2]),
1279                             mean = mean_llogis_25,
1280                             sigma = cvar_llogis_25)
1281
1282 f_invg_25[i] <- posterior_invg (mvnorm_invg_25[i, 1],
1283                                 mvnorm_invg_25[i, 2])
1284 g_invg_25[i] <- dmnorm(x = c (mvnorm_invg_25[i, 1],
1285                                 mvnorm_invg_25[i, 2]),
1286                           mean = mean_invg_25,
1287                           sigma = cvar_invg_25)
1288 }
1289
1290 is_IL_wei_25 <- sum (f_wei_25/g_wei_25)/n
1291 is_IL_gamma_25 <- sum (f_gamma_25/g_gamma_25)/n
1292 is_IL_lnorm_25 <- sum (f_lnorm_25/g_lnorm_25)/n
1293 is_IL_llogis_25 <- sum (f_llogis_25/g_llogis_25)/n
1294 is_IL_invg_25 <- sum (f_invg_25/g_invg_25)/n

```

```

1295
1296 evidence_IS_25 <- c (is_IL_wei_25, is_IL_gamma_25,
1297                      is_IL_lnorm_25, is_IL_llogis_25,
1298                      is_IL_invg_25)
1299
1300 is_sum_IL_25 <- sum (is_IL_wei_25, is_IL_gamma_25,
1301                     is_IL_lnorm_25, is_IL_llogis_25,
1302                     is_IL_invg_25)
1303
1304 evidence_25 <- evidence_IS_25/is_sum_IL_25
1305 evi_IS_log_25 <- -2*log (evidence_IS_25)
1306
1307 #####REVERSE IMPORTANCE SAMPLING#####
1308
1309 wei_g <- NULL
1310 gamma_g <- NULL
1311 lnorm_g <- NULL
1312 llogis_g <- NULL
1313 invg_g <- NULL
1314 wei_f <- NULL
1315 gamma_f <- NULL
1316 lnorm_f <- NULL
1317 llogis_f <- NULL
1318 invg_f <- NULL
1319
1320 for (i in 1:nrow(posterior_df_75)) {
1321
1322   wei_f[i] <- posterior_wei (posterior_df_75$mu.weib[i],
1323                             posterior_df_75$sigma.wei[i])
1324   wei_g[i] <- dmvnorm(x = c (posterior_df_75$mu.weib[i],
1325                             posterior_df_75$sigma.wei[i]),
1326                       mean = mean_wei_75,
1327                       sigma = cvar_wei_75)
1328
1329   gamma_f[i] <- posterior_gamma(posterior_df_75$alpha[i],
1330                                 posterior_df_75$beta[i])
1331   gamma_g[i] <- dmvnorm(x = c (posterior_df_75$beta[i],
1332                                 posterior_df_75$alpha[i]),
1333                         mean = mean_gamma_75,

```

```

1334         sigma = cvar_gamma_75)
1335
1336     lnorm_f[i] <- posterior_lnorm (posterior_df_75$mu.lnorm[i],
1337                                   posterior_df_75$sigma.lnorm[i])
1338     lnorm_g[i] <- dmvnorm (x = c (posterior_df_75$mu.lnorm[i],
1339                                   posterior_df_75$sigma.lnorm[i]),
1340                             mean = mean_lnorm_75,
1341                             sigma = cvar_lnorm_75)
1342
1343     llogis_f[i] <- posterior_llogis(posterior_df_75$mu.llog[i],
1344                                    posterior_df_75$sigma.llog[i])
1345     llogis_g[i] <- dmvnorm (x = c (posterior_df_75$mu.llog[i],
1346                                    posterior_df_75$sigma.llog[i]),
1347                             mean = mean_llogis_75,
1348                             sigma = cvar_llogis_75)
1349
1350     invg_f[i] <- posterior_invg (posterior_df_75$mu.IG[i],
1351                                 posterior_df_75$lambda.IG[i])
1352     invg_g[i] <- dmvnorm(x = c (posterior_df_75$mu.IG[i],
1353                                 posterior_df_75$lambda.IG[i]),
1354                             mean = mean_invg_75,
1355                             sigma = cvar_invg_75)
1356 }
1357
1358 ris_IL_wei <- sum (wei_g/wei_f) / n
1359 ris_IL_gam <- sum (gamma_g/gamma_f) / n
1360 ris_IL_lnorm <- sum (lnorm_g/lnorm_f) / n
1361 ris_IL_llogis <- sum (llogis_g/llogis_f) / n
1362 ris_IL_invg <- sum (invg_g/invg_f) / n
1363 evidence_RIS <- 1 / c (ris_IL_wei, ris_IL_gam,
1364                       ris_IL_lnorm, ris_IL_llogis,
1365                       ris_IL_invg)
1366
1367 ris_sum <- sum (evidence_RIS) #Get the total
1368 evidence_RIS/ris_sum #Calculate the weight
1369 -2*log (evidence_RIS) #Expresses in log(deviance)
1370 #1/evidence_RIS
1371 #evidence_IS/sum(evidence_IS)
1372

```

```

1373
1374 #####
1375 wei_g <- NULL
1376 gamma_g <- NULL
1377 lnorm_g <- NULL
1378 llogis_g <- NULL
1379 invg_g <- NULL
1380 wei_f <- NULL
1381 gamma_f <- NULL
1382 lnorm_f <- NULL
1383 llogis_f <- NULL
1384 invg_f <- NULL
1385
1386 for (i in 1:nrow(posterior_df_25)) {
1387
1388   wei_f[i] <- posterior_wei (posterior_df_25$mu.weib[i],
1389                             posterior_df_25$sigma.weib[i])
1390   wei_g[i] <- abs(dmvnorm(x = c (posterior_df_25$mu.weib[i],
1391                                 posterior_df_25$sigma.weib[i]),
1392                               mean = mean_wei_25,
1393                               sigma = cvar_wei_25))
1394
1395   gamma_f[i] <- posterior_gamma(posterior_df_25$alpha[i],
1396                                 posterior_df_25$beta[i])
1397   gamma_g[i] <- dmvnorm(x = c (posterior_df_25$beta[i],
1398                                 posterior_df_25$alpha[i]),
1399                           mean = mean_gamma_25,
1400                           sigma = cvar_gamma_25)
1401
1402   lnorm_f[i] <- posterior_lnorm (posterior_df_25$mu.lnorm[i],
1403                                 posterior_df_25$sigma.lnorm[i])
1404   lnorm_g[i] <- dmvnorm (x = c (posterior_df_25$mu.lnorm[i],
1405                                 posterior_df_25$sigma.lnorm[i]),
1406                           mean = mean_lnorm_25,
1407                           sigma = cvar_lnorm_25)
1408
1409   llogis_f[i] <- posterior_llogis(posterior_df_25$mu.llog[i],
1410                                   posterior_df_25$sigma.llog[i])
1411   llogis_g[i] <- dmvnorm (x = c (posterior_df_25$mu.llog[i],

```

```

1412         posterior_df_25$sigma.llog[i]),
1413         mean = mean_llogis_25,
1414         sigma = cvar_llogis_25)
1415
1416     invg_f[i] <- posterior_invg (posterior_df_25$mu.IG[i],
1417                                posterior_df_25$lambda.IG[i])
1418     invg_g[i] <- dmvnorm(x = c (posterior_df_25$mu.IG[i],
1419                                posterior_df_25$lambda.IG[i]),
1420                           mean = mean_invg_25,
1421                           sigma = cvar_invg_25)
1422 }
1423
1424 wei_f[wei_f == 0] <- max(wei_f)
1425 prop_wei <- (wei_g/wei_f)
1426 prop_wei[prop_wei == Inf] <- 0
1427 sort(prop_wei, decreasing = TRUE)
1428 mean(prop_wei)*length(prop_wei)
1429
1430 ris_IL_wei <- sum (wei_g/wei_f) / n
1431 ris_IL_gam <- sum (gamma_g/gamma_f) / n
1432 ris_IL_lnorm <- sum (lnorm_g/lnorm_f) / n
1433 ris_IL_llogis <- sum (llogis_g/llogis_f) / n
1434 ris_IL_invg <- sum (invg_g/invg_f) / n
1435 evidence_RIS <- 1 / c ( ris_IL_wei, ris_IL_gam,
1436                        ris_IL_lnorm, ris_IL_llogis,
1437                        ris_IL_invg)
1438 -2*log(evidence_RIS)#Express in log(deviance)
1439 ris_sum <- sum (evidence_RIS)
1440 evidence_RIS/ris_sum
1441
1442 #####
1443
1444 ##### MODEL WEIGHTS #####
1445
1446 #Posterior model probabilities
1447
1448 #Akaike information criteria
1449 #Minimum AIC
1450 aic_min_75 <- min(aic_75)

```

```
1451 #Difference of AIC values
1452 delta_aic_75 <- aic_75 - aic_min_75
1453 #Model weights - AIC
1454 aic_w_75 <- exp(-0.5*delta_aic_75) / sum(exp(-0.5*delta_aic_75))
1455
1456 sum(aic_w_75) #Check the total weight
1457
1458 aic_min_25 <- min(aic_25)
1459 #Difference of AIC values
1460 delta_aic_25 <- aic_25 - aic_min_25
1461 #Model weights - AIC
1462 aic_w_25 <- exp(-0.5*delta_aic_25) / sum(exp(-0.5*delta_aic_25))
1463
1464 sum(aic_w_25)#Check the total weight
1465
1466 #Bayesian information criteria
1467 #Minimum BIC
1468 bic_min_75 <- min(bic_75)
1469 #Difference of BIC values
1470 delta_bic_75 <- bic_75 - bic_min_75
1471 #Model weights - BIC
1472 bic_w_75 <- exp(-0.5*delta_bic_75) / sum(exp(-0.5*delta_bic_75))
1473
1474 sum(bic_w_75)#Check the total weight
1475
1476 #Bayesian information criteria
1477 #Minimum BIC
1478 bic_min_25 <- min(bic_25)
1479 #Difference of BIC values
1480 delta_bic_25 <- bic_25 - bic_min_25
1481 #Model weights - BIC
1482 bic_w_25 <- exp(-0.5*delta_bic_25) / sum(exp(-0.5*delta_bic_25))
1483
1484 sum(bic_w_25)#Check the total weight
1485
1486 median_dev_75 <- apply (df_deviance_75, 2, median)
1487
1488 #Minimum median deviance
1489 dev_min_75 <- min(median_dev_75)
```

```

1490
1491 #Deviance difference over the model with minimum deviance
1492 delta_dic_75 <- median_dev_75 - dev_min_75
1493
1494 #Model weights - DIC
1495 dic_w_75 <- exp(-0.5*delta_dic_75) / sum(exp(-0.5*delta_dic_75))
1496
1497 sum(dic_w_75)#Check the total weight
1498
1499 median_dev_25 <- apply (df_deviance_25, 2, mean)
1500
1501 #Minimum median deviance
1502 dev_min_25 <- min(median_dev_25)
1503
1504 #Deviance difference over the model with minimum deviance
1505 delta_dic_25 <- median_dev_25 - dev_min_25
1506
1507 #Model weights - DIC
1508 dic_w_25 <- exp(-0.5*delta_dic_25) / sum(exp(-0.5*delta_dic_25))
1509
1510 sum(dic_w_25)

```

Section C.4: R Code for the simulation - Jeffreys' Prior

This section provides the code for the replication of the simulation results described in Chapter 5. In order to run these code we require R and RStudio. The following packages need be installed on the machine to run the code.

```

1 require (rstan)
2 require (ggplot2)
3 require (coda)
4 require (latex2exp)

```

To run the simulation for the comparison of the Bayesian estimates with the MLE, with different sample sizes and ρ values the following code may used. Note that for large number of repetitions this may take some time.

```

1 #Simulate data

```

```

2 weibull.data <- function (samplesize, #sample size
3                               scale, #scale parameter
4                               rho,
5                               propFailed){
6   data.wei <- sort (rweibull (samplesize, shape = 1/(scale*rho),
7                               scale = exp(scale)))
8   T.cen <- data.wei[max(2, ceiling(propFailed*samplesize))]
9   censored <- data.wei > T.cen
10  event <- c (rep(1, length(data.wei[censored == FALSE])),
11             rep(0,length(data.wei[censored == TRUE])))
12  gen.fails <- c (data.wei[censored == FALSE],
13                rep(NA,length(data.wei[censored == TRUE])))
14  rank.fails <- rank (gen.fails, ties.method = "first",
15                    na.last = "keep")
16  gen.fails [is.na(gen.fails)] <- T.cen
17
18  simul.data <- data.frame(sampleddata = data.wei,
19                          Censored = censored,
20                          obs.fails = gen.fails,
21                          ranks = rank.fails)
22
23  sim.data.bay <- list (sampledata = data.wei,
24                      Censored = event,
25                      J = length (data.wei))
26
27  return(sim.data.bay)
28 }
29
30
31 #Stan codes for the posterior distribution
32
33 #Jeffreys' prior
34
35 model_Jeffreys <- "
36 functions {
37
38   real newweibull_log (vector x, real theta, real rho) {
39     vector [num_elements (x)] prob;
40     real lprob;

```

```

41 for (i in 1:num_elements(x)) {
42   prob[i] <- (1/(theta*rho*x[i]))*exp ((log (x[i]) - theta)/
43     (theta*rho) - exp((log (x[i]) - theta)/
44       (theta*rho)));
45 }
46 lprob <- sum (log (prob));
47 return lprob;
48 }
49 }
50
51 data {
52   int<lower=0> J; // number of observations
53   vector [J]sampledata; // failure in hours
54
55 }
56 parameters {
57   real theta; // scale
58   real<lower=0> rho; // cv
59 }
60
61 model {
62   //prior
63   //theta ~ uniform (0, 20);
64   //Likelihood and prior
65   sampledata ~ newweibull(theta, rho);
66   increment_log_prob(-log(rho));
67 }
68 "
69 mod_Jeffreys <- stan_model(model_code = model_Jeffreys)
70
71
72 #Weak informative priors
73 #Uniform prior for location and inverese-gamma for scale
74 #Find the posterior estimates and then compute the CV
75 #Since the location parameter is 1 sigma and rho is the same in
76   numerical value
77 #Therefore we use the same notation rho instead of sigma for ease
78 #Coding could be done in the same loop in that way

```

```

79 model_weakprior <- "
80 functions {
81
82   real newweibull_log (vector x, real theta, real rho) {
83     vector [num_elements (x)] prob;
84     real lprob;
85     for (i in 1:num_elements(x)) {
86       prob[i] <- 1/(rho*x[i])*exp(((log (x[i]) - theta)/rho) -
87         exp (((log (x[i]) - theta)/rho)));
88     }
89     lprob <- sum (log (prob));
90     return lprob;
91   }
92 }
93
94 data {
95   int<lower=0> J; // number of observations
96   vector [J]sampledata; // failure in hours
97 }
98
99 parameters {
100   real theta; // location
101   real<lower=0> rho; // scale (actually sigma)
102 }
103
104 model {
105   rho ~ inv_gamma(0.1, 0.1);
106   theta ~ uniform (0, 20);
107   //Likelihood and prior
108   sampledata ~ newweibull(theta, rho);
109   //increment_log_prob(-log(rho));
110 }
111 "
112 mod_weakprior <- stan_model(model_code = model_weakprior)
113
114 #Example of Jeffreys' prior
115
116 curve (dinvgamma(x, 0.1, 0.1))
117

```

```

118 set.seed(1589)
119 test <- (weibull.data(samplesize = 100,
120                       scale = 2,
121                       rho = 0.2,
122                       propFailed = 1))
123
124 mcmc.weibull <- sampling (mod_Jeffreys, data = test,
125                          chains = 1, iter = 10^5,
126                          warmup = 5000, thin = 1)
127
128 summary (mcmc.weibull)
129
130 weibull.coda <- mcmc.list(lapply(1:ncol(mcmc.weibull),
131                                  function(x)
132                                  {mcmc(as.array(mcmc.weibull)[,x,])}))
133
134 #Example of the weak informative priors
135
136 set.seed(1589)
137 test <- (weibull.data(samplesize = 100,
138                       scale = 2,
139                       rho = 0.2,
140                       propFailed = 1))
141
142 mcmc.weibull <- sampling (mod_weakprior, data = test,
143                          chains = 1, iter = 10^5,
144                          warmup = 5000, thin = 1)
145
146 summary (mcmc.weibull)
147
148 weibull.coda <- mcmc.list(lapply(1:ncol(mcmc.weibull),
149                                  function(x)
150                                  {mcmc(as.array(mcmc.weibull)[,x,])}))
151
152 #Define maximum likelihood function
153 mle <- function (data, #Data
154                  p.int = c(6, 0.5), #Initial values of the parameters
155                  samplesize) {
156   loglik <- function (theta){

```

```

157   #theta[1] <- mu - scale parameter of weibull distribution
158   #theta [2] <- rho - coefficient of variation
159   #Define new variable z
160   z <- suppressWarnings((log(data) - theta[1]) / (theta[1]*theta[2]))
161   #loglikelihood function of the reparameterized Weibull model
162   ll <- suppressWarnings(-samplesize*(log(theta[1]) + log(theta[2]))
163                         - sum(log(data) - z + exp(z)))
164   return (-ll)
165 }
166
167 #Optimise to find the ML estimates
168 estim1 <- optim (p.int, loglik,
169                 method = "BFGS",
170                 control = list(
171                     maxit = 10000,
172                     reltol = .Machine$double.eps))
173
174 #Results
175 results <- list (estim1)
176 return (results)
177 }
178
179 #Example of MLE function
180
181 with (test, mle (data = sampledata,
182                 p.int = c (1, 0.5),
183                 samplesize = J))
184
185 num.sim <- 1000
186 results.table <- expand.grid (num.sim = 1:num.sim,
187                             samplesize = c (50, 100, 1000),
188                             scale.true = c (1),
189                             rho.true = c (0.2, 0.25, 0.5,
190                                           0.75, 1, 1.25),
191                             propFailed = c(1))
192
193 simulation.Jeffreys <-
194   function(i) {
195     seed = 1000

```

```

196  set.seed(i * 125 + seed)
197  sim.data <- with(results.table[i,],
198                  weibull.data(samplesize = samplesize,
199                               propFailed = propFailed,
200                               rho = rho.true,
201                               scale = scale.true))
202
203  mle.estim <- with (sim.data, mle (data = sampledata,
204                                   p.int =
205                                   c(results.table$scale.true[i],
206                                   results.table$rho.true[i]),
207                                   samplesize =
208                                   results.table$samplesize[i]))
209
210  mcmc.Jeffreys <- sampling (mod_Jeffreys, data = sim.data,
211                            chains = 1, iter = 10^4,
212                            warmup = 100, thin = 5)
213
214  mcmc.weakprior <- sampling (mod_weakprior, data = sim.data,
215                             chains = 1, iter = 10^4,
216                             warmup = 100, thin = 5)
217
218  weibull.coda <- mcmc.list(lapply(1:ncol(mcmc.Jeffreys),
219                                   function(x)
220                                   {mcmc(as.array(mcmc.Jeffreys)[,x,]
221                                   )}))
222
223  weakprior.coda <- mcmc.list(lapply(1:ncol(mcmc.weakprior),
224                                   function(x)
225                                   {mcmc(as.array(mcmc.weakprior)[,x
226                                   ,]
227                                   )}))
228
229  posterior.samples <- data.frame (weibull.coda[[1]][, 1:2])
230
231  posterior.weakprior <- data.frame (weakprior.coda[[1]][, 1:2])
232
233  estim.bay <- apply (posterior.samples, 2, median)
234  estim.weakprior <- apply (posterior.weakprior, 2, median)

```


[illegible]

```

312         MSE_scale.weak = mse_scale.weak,
313         MSE_rho.MLE = mse_rho.MLE,
314         MSE_rho_Jeff = mse_rho.Jeff,
315         MSE_rho_weak = mse_rho.weak),
316     by = list(Samplesize = samplesize,
317             Scale = scale.true,
318             Rho = rho.true),
319     FUN = mean))
320
321
322 #fix(results.mean)
323
324 results.quantile <- with (as.data.frame(results.table),
325     aggregate (
326         x = list(Scale.MLE = scale.mle ,
327             Scale.Jeff = scale.bay,
328             Scale.weak = location.weak,
329             Rho.MLE = rho.mle,
330             Rho.Jeff = rho.bay,
331             Rho.weak = rho.weak,
332             Bias_scale.MLE = absbias_scale.MLE,
333             Bias_scale_Jeff = absbias_scale.Jeff,
334             Bias_scale_weak = absbias_scale.weak,
335             Bias_rho.MLE = absbias_rho.MLE,
336             Bias_rho.Jeff = absbias_scale.Jeff,
337             Bias_rho_weak = absbias_rho.weak,
338             MSE_scale.MLE= mse_scale.MLE,
339             MSE_scale.Jeff = mse_scale.Jeff,
340             MSE_scale.weak = mse_scale.weak,
341             MSE_rho.MLE = mse_rho.MLE,
342             MSE_rho_Jeff = mse_rho.Jeff,
343             MSE_rho_weak = mse_rho.weak),
344         by = list(Samplesize = samplesize,
345             Scale = scale.true,
346             Rho = rho.true),
347         FUN = quantile))
348
349
350

```

```

351
352 results.sd <- with (as.data.frame(results.table),
353                     aggregate (
354                         x = list(Scale.SD.MLE = scale.mle ,
355                                Scale.SD.Jeff = scale.bay,
356                                Scale.SD.weak = location.weak,
357                                Rho.SD.MLE = rho.mle,
358                                Rho.SD.Jeff = rho.bay,
359                                Rho.SD.weak = rho.weak,
360                                Bias_scale.SD.MLE=absbias_scale.MLE,
361                                Bias.scale.SD_Jeff=absbias_scale.Jeff,
362                                Bias.scale.SD_weak=absbias_scale.weak,
363                                Bias_rho.SD.MLE = absbias_rho.MLE,
364                                Bias_rho.SD.Jeff=absbias_scale.Jeff,
365                                Bias_rho.SD.weak = absbias_rho.weak,
366                                MSE_scale.SD.MLE= mse_scale.MLE,
367                                MSE_scale.SD.Jeff = mse_scale.Jeff,
368                                MSE_scale.SD.weak = mse_scale.weak,
369                                MSE_rho.SD.MLE = mse_rho.MLE,
370                                MSE_rho_SD.Jeff = mse_rho.Jeff,
371                                MSE_rho_SD.weak = mse_rho.weak),
372                         by = list(Samplesize = samplesize,
373                                Scale = scale.true,
374                                Rho = rho.true),
375                         FUN = sd))
376
377 results <- cbind (results.mean, results.sd[c (4:21)],
378                  scale.MLE.low = results.quantile$Scale.MLE[, 2],
379                  scale.MLE.up = results.quantile$Scale.MLE[, 4],
380                  rho.MLE.low = results.quantile$Rho.MLE[, 2],
381                  rho.MLE.up = results.quantile$Rho.MLE[, 4],
382                  scale.Jeff.low = results.quantile$Scale.Jeff[, 2],
383                  scale.Jeff.up = results.quantile$Scale.Jeff[, 4],
384                  rho.Jeff.low = results.quantile$Rho.Jeff[, 2],
385                  rho.Jeff.up = results.quantile$Rho.Jeff[, 4],
386                  scale.weak.low = results.quantile$Scale.weak[, 2],
387                  scale.weak.up = results.quantile$Scale.weak[, 4],
388                  rho.weak.low = results.quantile$Rho.weak[, 2],
389                  rho.weak.up = results.quantile$Rho.weak[, 4],

```



```

29 print (plot.Relbias_Rho <- (qplot (x = Rho,
30                                   y = Relbias,
31                                   data = subset(results.Rho,
32                                                 Parameter%in%c("Rho.MLE",
33                                                           "Rho.Jeff", "Rho.weak")),
34                                   group = factor(Samplesize),
35                                   colour = factor(Samplesize),
36                                   geom = c ("point", "line"),
37                                   xlab = "True Shape",
38                                   ylab = "Relative Bias")
39 + facet_grid(Scale~Parameter,
40              scales = "free_y",
41              labeller = labeller(
42                Scale = c("1" = "True Scale = 1"),
43                Parameter = c ("Rho.Jeff" = "Jeffreys'",
44                              "Rho.weak" =
45                                "Weakly Informative Prior",
46                                "Rho.MLE" = "MLE")
47              ))
48 # + geom_errorbar(aes(x = Rho,
49                       ymin = Bias.CI.Low,
50                       # ymax = Bias.CI.Up), width= 0.01)
51 )
52 +labs(colour = "Sample size"))
53
54
55
56
57 print (plot.mse_Rho <- (qplot (x = Rho,
58                               y = MSE,
59                               data = subset(results.Rho,
60                                             Parameter%in%c("Rho.MLE",
61                                                           "Rho.Jeff", "Rho.weak")),
62                               group = factor(Samplesize),
63                               colour = factor(Samplesize),
64                               geom = c ("point", "line"),
65                               xlab = "True Coefficient of variation",
66                               ylab = "Mean squared error")
67 + facet_grid(Scale~Parameter,

```

```

68         scales = "free_y", labeller =
69         labeller(
70         Scale = c("1" = "True Scale = 1")
71         ,
72         Parameter =
73         c ("Rho.Jeff"="Bayesian",
74         "Rho.weak" = "Weakly Informative"
75         ,
76         "Rho.MLE" = "MLE")
77         ))
78         # + geom_errorbar(aes(x = Rho, ymin = Bias.CI.
79         Low,
80         #   ymax = Bias.CI.Up), width= 0.01)
81         )
82         +labs(colour = "Sample size"))

```

Bibliography

- Abbas, Kamran and Yincui Tang (2016). “Objective Bayesian analysis for log-logistic distribution”. In: *Communications in Statistics-Simulation and Computation* 45.8, pp. 2782–2791.
- Achcar, Jorge A, Ron Brookmeyer, and William G Hunter (1985). “An application of bayesian analysis to medical follow up data”. In: *Statistics in medicine* 4.4, pp. 509–520.
- Ahmed, Essam A (2014). “Bayesian estimation based on progressive Type-II censoring from two-parameter bathtub-shaped lifetime model: an Markov chain Monte Carlo approach”. In: *Journal of Applied Statistics* 41.4, pp. 752–768.
- Aitkin, Murray, Richard J Boys, and Tom Chadwick (2005). “Bayesian point null hypothesis testing via the posterior likelihood ratio”. In: *Statistics and Computing* 15.3, pp. 217–230.
- Aitkin, Murray, Charles C Liu, and Tom Chadwick (2009). “Bayesian model comparison and model averaging for small-area estimation”. In: *The Annals of Applied Statistics*, pp. 199–221.
- Akpınar, E Kavak and S Akpınar (2005). “A statistical analysis of wind speed data used in installation of wind energy conversion systems”. In: *Energy conversion and management* 46.4, pp. 515–532.
- Ali, Sajid, Muhammad Aslam, and Syed Mohsin Ali Kazmi (2013). “A study of the effect of the loss function on Bayes Estimate, posterior risk and hazard function for Lindley distribution”. In: *Applied Mathematical Modelling* 37.8, pp. 6068–6078.
- Aljahdali, Sultan H, Alaa Sheta, and David Rine (2001). “Prediction of software reliability: A comparison between regression and neural network non-parametric models”.

- In: *Proceedings ACS/IEEE International Conference on Computer Systems and Applications*. IEEE, pp. 470–473.
- Antonio Moala, Fernando, Pedro Luiz Ramos, and Jorge Alberto Achcar (2013). “Bayesian inference for two-parameter gamma distribution assuming different noninformative priors”. In: *Revista Colombiana de Estadística* 36.2, pp. 319–336.
- Arago, Olivia, Stefan Bretschneider, and Stephan Staudacher (2007). “A Unit Cost Comparison Methodology for Turbofan Engines”. In: *ASME Turbo Expo 2007: Power for Land, Sea, and Air*. American Society of Mechanical Engineers, pp. 81–86.
- Awwad, Raed R Abu, Mohammad Z Raqab, and Intesar M Al-Mudahakha (2015). “Statistical inference based on progressively type II censored data from Weibull model”. In: *Communications in Statistics-Simulation and Computation* 44.10, pp. 2654–2670.
- Bar-Lev, Shaul K (2004). “Likelihood-based inference for the shape parameter of a two parameter Weibull distribution”. In: *Lifetime data analysis* 10.3, pp. 293–308.
- Barlow, Richard and Larry Hunter (1960). “Optimum preventive maintenance policies”. In: *Operations research* 8.1, pp. 90–100.
- Barlow, Richard E and Frank Proschan (1996). *Mathematical theory of reliability*. Vol. 17. Siam.
- Basak, Indrani and N Balakrishnan (2003). “Robust estimation under progressive censoring”. In: *Computational statistics & data analysis* 44.1, pp. 349–376.
- Basu, AP and Nader Ebrahimi (1991). “Bayesian approach to life testing and reliability estimation using asymmetric loss function”. In: *Journal of statistical planning and inference* 29.1-2, pp. 21–31.
- Bayarri, M Jesús and James O Berger (2004). “The interplay of Bayesian and frequentist analysis”. In: *Statistical Science*, pp. 58–80.
- Bedeian, Arthur G and Kevin W Mossholder (2000). “On the use of the coefficient of variation as a measure of diversity”. In: *Organizational Research Methods* 3.3, pp. 285–297.
- Beichelt, Frank (1981). “A generalized block-replacement policy”. In: *IEEE Transactions on Reliability* 30.2, pp. 171–172.

- Benard, A and EC Bos-Levenbach (1955). *The Plotting of Observations on Probability-paper*. Stichting Mathematisch Centrum. Statistische Afdeling.
- Berg, Menachem and Benjamin Epstein (1976). "A modified block replacement policy". In: *Naval Research Logistics Quarterly* 23.1, pp. 15–24.
- Berger, James O and José M Bernardo (1992). "On the development of the reference prior method". In: *Bayesian statistics* 4.4, pp. 35–60.
- Bernardo, Jose M (1979). "Reference posterior distributions for Bayesian inference". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 41.2, pp. 113–128.
- Bickel, Peter J (1965). "On some robust estimates of location". In: *The Annals of Mathematical Statistics* 36.3, pp. 847–858.
- Boland, Philip J and Frank Proschan (1982). "Periodic replacement with increasing minimal repair costs at failure". In: *Operations Research* 30.6, pp. 1183–1189.
- Box, G. E. P. (1953). "Non-Normality and Tests on Variances". In: *Biometrika* 40.3/4, pp. 318–335.
- Bracale, Antonio and Pasquale De Falco (2015). "An Advanced Bayesian Method for Short-Term Probabilistic Forecasting of the Generation of Wind Power". In: *Energies* 8.9, pp. 10293–10314.
- Brooks, Stephen P and Andrew Gelman (1998). "General methods for monitoring convergence of iterative simulations". In: *Journal of computational and graphical statistics* 7.4, pp. 434–455.
- Bullock, Bronson P and Edward L Boone (2007). "Deriving tree diameter distributions using Bayesian model averaging". In: *Forest ecology and management* 242.2, pp. 127–132.
- Calabria, R and Gs Pulcini (1994). "An engineering approach to Bayes estimation for the Weibull distribution". In: *Microelectronics Reliability* 34.5, pp. 789–802.
- Cantor, Alan B. and Jonathan J. Shuster (1992). "Parametric versus non-parametric methods for estimating cure rates based on censored survival data". In: *Statistics in Medicine* 11.7, pp. 931–937. DOI: [10.1002/sim.4780110710](https://doi.org/10.1002/sim.4780110710). eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/sim.4780110710>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/sim.4780110710>.

- Celik, Ali Naci (2004). “A statistical analysis of wind power density based on the Weibull and Rayleigh models at the southern region of Turkey”. In: *Renewable energy* 29.4, pp. 593–604.
- Chiodo, Elio and G Mazzanti (2006). “Bayesian reliability estimation based on a Weibull stress-strength model for aged power system components subjected to voltage surges”. In: *Dielectrics and Electrical Insulation, IEEE Transactions on* 13.1, pp. 146–159.
- Cho, Danny I and Mahmut Parlar (1991). “A survey of maintenance models for multi-unit systems”. In: *European Journal of Operational Research* 51.1, pp. 1–23.
- Chuang, SK et al. (2001). “Kaplan-Meier analysis of dental implant survival: a strategy for estimating survival with clustered observations”. In: *Journal of Dental Research* 80.11, pp. 2016–2020.
- Cohen, Clifford A and Betty Whitten (1982). “Modified maximum likelihood and modified moment estimators for the three-parameter Weibull distribution”. In: *Communications in Statistics-Theory and Methods* 11.23, pp. 2631–2656.
- Danish, Muhammad Yameen and Muhammad Aslam (2014). “Bayesian inference for the randomly censored Weibull distribution”. In: *Journal of Statistical Computation and Simulation* 84.1, pp. 215–230.
- Dekker, Rommert (1996). “Applications of maintenance optimization models: a review and analysis”. In: *Reliability engineering & system safety* 51.3, pp. 229–240.
- Diamandis, Eleftherios P and Theodore K Christopoulos (1996). *Immunoassay*. Academic Press.
- Duerinckx, Mitia, Christophe Ley, and Yvik Swan (2014). “Maximum likelihood characterization of distributions”. In: *Bernoulli* 20.2, pp. 775–802.
- Dümbgen, Lutz, Kaspar Rufibach, and Dominic Schuhmacher (2014). “Maximum-likelihood estimation of a log-concave density based on censored data”. In: *Electronic Journal of Statistics* 8.1, pp. 1405–1437.
- Ellison, A. M. (2004). “Bayesian inference in ecology”. In: *Ecology Letters* 7.6, pp. 509–520.
- Elsner, James B and Brian H Bossak (2001). “Bayesian analysis of US hurricane climate”. In: *Journal of Climate* 14.23, pp. 4341–4350.

- Fan, Tsai-hung and Tsung-ming Hsu (2014). “Constant stress accelerated life test on a multiple-component series system under Weibull lifetime distributions”. In: *Communications in Statistics-Theory and Methods* 43.10-12, pp. 2370–2383.
- Gelman, Andrew (2008). “Objections to Bayesian statistics”. In: *Bayesian Analysis* 3.3, pp. 445–449.
- Genschel, U. and W. Q. Meeker (2010). “A Comparison of Maximum Likelihood and Median-Rank Regression for Weibull Estimation”. In: *Quality Engineering* 22.4, pp. 236–255.
- Gijbels, I. (2010). “Censored data”. In: *Wiley Interdisciplinary Reviews: Computational Statistics* 2.2, pp. 178–188.
- Goegebeur, Yuri, Armelle Guillou, and Théo Rietsch (2015). “Robust conditional Weibull-type estimation”. In: *Annals of the Institute of Statistical Mathematics* 67.3, pp. 479–514.
- Goodwin, Paul (2015). “When simple alternatives to Bayes formula work well: Reducing the cognitive load when updating probability forecasts”. In: *Journal of Business Research* 68.8, pp. 1686–1691.
- Greenfield, Victoria A and David M Persselin (2002). *An economic framework for evaluating military aircraft replacement*. Tech. rep. DTIC Document.
- Guure, Chris Bambe et al. (2015). “Bayesian statistical inference of the loglogistic model with interval-censored lifetime data”. In: *Journal of Statistical Computation and Simulation* 85.8, pp. 1567–1583.
- Gündüz, N, E Başar, and C Aydin (2015). “Performance of Some Robust Estimators for the Weibull Distribution”. In: *Acta Physica Polonica, A*. 128.
- Hampel, Frank R (1971). “A general qualitative definition of robustness”. In: *The Annals of Mathematical Statistics*, pp. 1887–1896.
- Hanscom, MA and Robert Cléroutx (1975). “The block replacement problem”. In: *Journal of statistical computation and simulation* 3.3, pp. 233–248.
- Hoeting, Jennifer A et al. (1999). “Bayesian model averaging: a tutorial”. In: *Statistical science*, pp. 382–401.
- Hong, Yili et al. (2014). “Bayesian Life Test Planning for the Log-Location-Scale Family of Distributions”. In:

- Hossain, Anwar and William Zimmer (2003). “Comparison of estimation methods for Weibull parameters: complete and censored samples”. In: *Journal of statistical computation and simulation* 73.2, pp. 145–153.
- Huang, Syuan-Rong and Shuo-Jye Wu (2012). “Bayesian estimation and prediction for Weibull model with progressive censoring”. In: *Journal of Statistical Computation and Simulation* 82.11, pp. 1607–1620.
- Huber, Peter J (1968). “Robust confidence limits”. In: *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete* 10.4, pp. 269–278.
- (1972). “The 1972 wald lecture robust statistics: A review”. In: *The Annals of Mathematical Statistics*, pp. 1041–1067.
- (1992). “Robust estimation of a location parameter”. In: *Breakthroughs in statistics*. Springer, pp. 492–518.
- (2011). *Robust statistics*. Springer.
- Jaekel, Louis A (1971). “Robust estimates of location: Symmetry and asymmetric contamination”. In: *The Annals of Mathematical Statistics*, pp. 1020–1034.
- Jeffreys Harold, Sir (1939). *Theory of probability*. The International series of monographs on physics. Oxford : Clarendon Press, 1939.
- Jeng, Shuen-Lin and William Q Meeker (2000). “Comparisons of approximate confidence interval procedures for type I censored data”. In: *Technometrics* 42.2, pp. 135–148.
- Kalaiselvan, C. and L. Bhaskara Rao (2016). “Comparison of reliability techniques of parametric and non-parametric method”. In: *Engineering Science and Technology, an International Journal* 19.2, pp. 691 –699. ISSN: 2215-0986. DOI: <https://doi.org/10.1016/j.jestch.2015.11.002>. URL: <http://www.sciencedirect.com/science/article/pii/S2215098615001640>.
- Kalbfleisch, John D (1978). “Non-parametric Bayesian analysis of survival time data”. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 40.2, pp. 214–221.
- Kaminskiy, Mark P and Vasiliy V Krivtsov (2005). “A simple procedure for Bayesian estimation of the Weibull distribution”. In: *Reliability, IEEE Transactions on* 54.4, pp. 612–616.

- Kaplan, Edward L and Paul Meier (1958). “Nonparametric estimation from incomplete observations”. In: *Journal of the American statistical association* 53.282, pp. 457–481.
- Khan, Akbar Ali et al. (2015). “Comparison of loss functions for estimating the scale parameter of log-normal distribution using non-informative priors”. In: *Haceteppe Journal of Mathematics and Statistics*.
- Kiche, J, Oscar Ngesa, George Orwa, et al. (2019). “On Generalized Gamma Distribution and Its Application to Survival Data”. In: *International Journal of Statistics and Probability* 8.5, pp. 85–102.
- Kim, Dal Ho, Woo Dong Lee, and Sang Gil Kang (2014). “Non-informative priors for the inverse Weibull distribution”. In: *Journal of Statistical Computation and Simulation* 84.5, pp. 1039–1054.
- Krebs, Charles J et al. (1989). *Ecological methodology*. Tech. rep. Harper & Row New York.
- Lee, Joseph Cheuk Yi, Michael Jason Fields, and Julie K Lundquist (2018). “Assessing variability of wind speed: comparison and validation of 27 methodologies”. In: *Wind Energy Science (Online)* 3.NREL/JA-5000-72768.
- Li, Gong and Jing Shi (2010a). “Application of Bayesian model averaging in modeling long-term wind speed distributions”. In: *Renewable Energy* 35.6, pp. 1192–1202.
- (2010b). “Application of Bayesian model averaging in modeling long-term wind speed distributions”. In: *Renewable Energy* 35.6, pp. 1192–1202.
- Lin, Yu-Jau and YL Lio (2012). “Bayesian inference under progressive type-I interval censoring”. In: *Journal of Applied Statistics* 39.8, pp. 1811–1824.
- Liu, Charles C and Murray Aitkin (2008). “Bayes factors: Prior sensitivity and model generalizability”. In: *Journal of Mathematical Psychology* 52.6, pp. 362–375.
- Liu, Xiao-Gao, Viliam Makis, and Andrew KS Jardine (1995). “A replacement model with overhauls and repairs”. In: *Naval Research Logistics (NRL)* 42.7, pp. 1063–1079.
- Maffei, G (1999). “Prediction of carbon monoxide acute air pollution episodes. Model formulation and first application in Lombardy”. In: *Atmospheric Environment* 33.23, pp. 3859–3872.

- Meeker, William Q. and Luis A. Escobar (1998). *Statistical methods for reliability data*. Wiley series in probability and statistics. Applied probability and statistics section. New York : Wiley, c1998.
- Miller, Robert B (1980). “Bayesian analysis of the two-parameter gamma distribution”. In: *Technometrics* 22.1, pp. 65–69.
- Moala, Fernando Antonio, Jorge Alberto Achcar, and Vera Lúcia Damasceno Tomazella (2012). “Bayesian estimation of generalized exponential distribution under noninformative priors”. In: *AIP Conference Proceedings*. Vol. 1490. 1. American Institute of Physics, pp. 230–242.
- Moala, Fernando Antonio and Débora Luzia Penha (2014). “Bayesian Estimation of Reliability of the Electronic Components Using Censored Data from Weibull Distribution: Different Prior Distributions”. In: *International Journal of Engineering Science (2008-4870)* 25.4.
- Moala, Fernando Antonio, Josemar Rodrigues, and Vera Lucia D Tomazella (2009). “A note on the prior distributions of Weibull parameters for the reliability function”. In: *Communications in Statistics—Theory and Methods* 38.7, pp. 1041–1054.
- Mohammed A, Al Omari et al. (2012). “Bayesian survival and hazard estimate for Weibull censored time distribution”. In: *Journal of Applied Sciences* 12, pp. 1313–1317.
- Murphy, Kevin P (2012). *Machine learning: a probabilistic perspective*. MIT press.
- Nakagawa, Toshio (1981). “A summary of periodic replacement with minimal repair at failure”. In: *Journal of the Operations Research Society of Japan* 24.3, pp. 213–228.
- Nakagawa, Toshio and Masaaki Kijima (1989). “Replacement policies for a cumulative damage model with minimal repair at failure”. In: *IEEE Transactions on Reliability* 38.5, pp. 581–584.
- Nakagawa, Toshio and Masashi Kowada (1983). “Analysis of a system with minimal repair and its application to replacement policy”. In: *European Journal of Operational Research* 12.2, pp. 176–182.
- Nurock, D and C Porteous (2008). “Methodology to determine the optimal replacement age of mobile mining machines”. In: *Third International Platinum Conference ‘Platinum Transformation*, pp. 297–306.

- Olteanu, Denisa and Laura Freeman (2010). “The evaluation of median-rank regression and maximum likelihood estimation techniques for a two-parameter Weibull distribution”. In: *Quality Engineering* 22.4, pp. 256–272.
- Penha, Fernando Antonio Moala Débora Luzia (2014). “Bayesian Estimation of Reliability of the Electronic Components Using Censored Data from Weibull Distribution: Different Prior Distributions”. In: *International Journal of Industrial Engineering* 25.4.
- Pepe, Margaret Sullivan and Motomi Mori (1993). “Kaplan—meier, marginal or conditional probability curves in summarizing competing risks failure time data?” In: *Statistics in Medicine* 12.8, pp. 737–751. DOI: [10.1002/sim.4780120803](https://doi.org/10.1002/sim.4780120803). eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/sim.4780120803>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/sim.4780120803>.
- Pradhan, Biswabrata and Debasis Kundu (2011). “Bayes estimation and prediction of the two-parameter gamma distribution”. In: *Journal of Statistical Computation and Simulation* 81.9, pp. 1187–1198.
- Prentice, Ross L, Benjamin J Williams, and Arthur V Peterson (1981). “On the regression analysis of multivariate failure time data”. In: *Biometrika* 68.2, pp. 373–379.
- Qatan, M et al. (2016). “Block replacement models for cables subject to deterioration”. In: *Condition Monitoring and Diagnosis (CMD), 2016 International Conference on*. IEEE, pp. 349–352.
- Reiser, Benjamin and Shaul Bar Lev (1979). “Likelihood Inference for Life Test Data”. In: *Reliability, IEEE Transactions on* R-28.1, pp. 38–43.
- Ruanaidh, Joseph JK O and William J Fitzgerald (2012). *Numerical Bayesian methods applied to signal processing*. Springer Science & Business Media.
- Schaubel, Douglas E and Jianwen Cai (2004). “Non-parametric estimation of gap time survival functions for ordered multivariate failure time data”. In: *Statistics in Medicine* 23.12, pp. 1885–1900.
- Seki, Tetsurou and Shinichiro Yokoyama (1993). “Simple and robust estimation of the Weibull parameters”. In: *Microelectronics Reliability* 33.1, pp. 45–52.

- Sheu, Shey-Huei (1991). "Periodic replacement with minimal repair at failure and general random repair cost for a multi-unit system". In: *Microelectronics Reliability* 31.5, pp. 1019–1025.
- Shiau, Jyh-Jeu Horng and Hsin-Hua Lin (1999). "Analyzing accelerated degradation data by nonparametric regression". In: *IEEE Transactions on reliability* 48.2, pp. 149–158.
- Simpson, George Gaylord, Anne Roe, and Richard C Lewontin (2003). *Quantitative zoology*. Courier Corporation.
- Sinha, Sanjoy K (1986). "Bayes estimation of the reliability function and hazard rate of a weibull failure time distribution". In: *Trabajos de Estadística* 1.2, pp. 47–56.
- Smith, Peter J. (2002). *Analysis of failure and survival data*. Chapman & Hall/CRC texts in statistical science series. Boca Raton : Chapman & Hall/CRC, c2002.
- Snedecor, George W and Witiiaam G Cochran (1989). "Statistical methods, 8thEdn". In: *Ames: Iowa State Univ. Press Iowa*.
- Somboonsavatdee, Anupap, Vijayan N Nair, and Ananda Sen (2007). "Graphical estimators from probability plots with right-censored data". In: *Technometrics* 49.4, pp. 420–429.
- Son, Young Sook and Mira Oh (2006). "Bayesian estimation of the two-parameter Gamma distribution". In: *Communications in Statistics—Simulation and Computation* 35.2, pp. 285–293.
- Sørensen, Jesper B (2002). "The use and misuse of the coefficient of variation in organizational demography research". In: *Sociological methods & research* 30.4, pp. 475–491.
- Spiegelhalter, David J et al. (2002). "Bayesian measures of model complexity and fit". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 64.4, pp. 583–639.
- Storlie, Curtis B et al. (2013). "A Bayesian reliability analysis of neutron-induced errors in high performance computing hardware". In: *Journal of the American Statistical Association* 108.502, pp. 429–440.
- Sun, Dongchu (1997). "A note on noninformative priors for Weibull distributions". In: *Journal of Statistical Planning and Inference* 61.2, pp. 319–338.

- Tango, Toshiro (1978). “Extended block replacement policy with used items”. In: *Journal of applied Probability* 15.3, pp. 560–572.
- Thamrin, Sri Astuti, James M McGree, and Kerrie L Mengersen (2013). “Modelling survival data to account for model uncertainty: a single model or model averaging?”. In: *SpringerPlus* 2.1, p. 665.
- Tilquin, C and R Cleroux (1975). “Block replacement policies with general cost structures”. In: *Technometrics* 17.3, pp. 291–298.
- Tsiotas, Georgios (2015). “A quasi-Bayesian model averaging approach for conditional quantile models”. In: *Journal of Statistical Computation and Simulation* 85.10, pp. 1963–1986.
- Volinsky, Chris T and Adrian E Raftery (2000). “Bayesian information criterion for censored survival models”. In: *Biometrics* 56.1, pp. 256–262.
- Wang, Hongzhou (2002a). “A survey of maintenance policies of deteriorating systems”. In: *European journal of operational research* 139.3, pp. 469–489.
- (2002b). “A survey of maintenance policies of deteriorating systems”. In: *European journal of operational research* 139.3, pp. 469–489.
- Wang, Hongzhou and Hoang Pham (1999). “Some maintenance models and availability withimperfect maintenance in production systems”. In: *Annals of Operations Research* 91, pp. 305–318.
- Wasserman, L. (2000). “Bayesian model selection and model averaging”. In: *Journal of Mathematical Psychology* 44.1, pp. 92–107.
- Wei, Lee-Jen, Danyu Y Lin, and Lisa Weissfeld (1989). “Regression analysis of multivariate incomplete failure time data by modeling marginal distributions”. In: *Journal of the American statistical association* 84.408, pp. 1065–1073.
- Wilcox, Rand R (2012). *Introduction to robust estimation and hypothesis testing*. Academic Press.
- Yang, Chia-Han et al. (2012). “Non-parametric Bayesian modeling of hazard rate with a change point for nanoelectronic devices”. In: *IIE Transactions* 44.7, pp. 496–506.
- Yang, Jingjing and David W Scott (2013). “Robust fitting of a Weibull model with optional censoring”. In: *Computational statistics & data analysis* 67, pp. 149–161.

- Yang, Zhaojun et al. (2015). “Bayesian reliability modeling and assessment solution for NC machine tools under small-sample data”. In: *Chinese Journal of Mechanical Engineering* 28.6, pp. 1229–1239.
- Ying, Z and LJ Wei (1994). “The Kaplan-Meier estimate for dependent failure time observations”. In: *Journal of Multivariate Analysis* 50.1, pp. 17–29.
- Yu, I-Tang and Che-Lun Chang (2012). “Applying Bayesian model averaging for quantile estimation in accelerated life tests”. In: *IEEE Transactions on Reliability* 61.1, pp. 74–83.
- Zhang, L-F, M Xie, and L-C Tang (2006). “Robust regression using probability plots for estimating the Weibull shape parameter”. In: *Quality and Reliability Engineering International* 22.8, pp. 905–917.
- Zhang, Xiongqing et al. (2013). “The application of Bayesian model averaging in compatibility of stand basal area for even-aged plantations in southern china”. In: *Forest Science* 60.4, pp. 645–651.
- Zhang, Yao and William Q Meeker (2005). “Bayesian life test planning for the Weibull distribution with given shape parameter”. In: *Metrika* 61.3, pp. 237–249.
- Zyl, R van and AJ van der Merwe (2017). “A Bayesian control chart for a common coefficient of variation”. In: *Communications in Statistics-Theory and Methods* 46.12, pp. 5795–5811.
- Zähle, Henryk (2016). “A definition of qualitative robustness for general point estimators, and examples”. In: *Journal of Multivariate Analysis* 143, pp. 12–31.