NTNU – Trondheim
Norwegian University of
Science and Technology

Censored Weibull Distributed Data in Experimental Design

Jeanett Gunneklev Støtvig

Master of Science in Physics and Mathematics
Submission date: February 2014
Supervisor: John Sølve Tyssedal, MATH

Norwegian University of Science and Technology
Department of Mathematical Sciences

Problem description

- Give an introduction to regression and experimental design.
- Study methods for estimating censored data.
- Evaluate and discuss the performances of the methods.

Assignment given: October 2, 2013
Supervisor: John Sølve Tyssedal

Preface

This master thesis concludes my master's degree in Applied Physics and Mathematics, with specialization in Industrial Mathematics, at the Norwegian University of Science and Technology. The work was carried out at the Department of Mathematical Sciences.

I would like to thank my supervisor, Associate Professor John Sølve Tyssedal for excellent guidance and motivation during the work of this thesis. I have enjoyed working with you and greatly appreciate the time you spent on this thesis.

Thanks to Karine Kaspersen Forsmo and Andrea Nyberget for proofreading.

Finally, I would like to thank all my friends and family for your support and encouragement, especially my friends in Trondheim for making my years at NTNU the greatest years of my life.

Jeanett Gunneklev Støtvig
Trondheim, February 2014

Abstract

Censoring is a common issue in experiments and survival analysis. One does not know the exact value of a censored data, the value of a censored data is only partially known. For example one can know if the censored data is greater than or smaller than a predetermined value, or within two certain values. Statistical analysis methods assume complete data. Therefore, in order to be able to use statistical methods, the censored data needs to be estimated to some value. One wishes to obtain a data set where the estimated value of the missing data is as close to the original data as possible. Several methods have been developed for this purpose, four of them are considered and tested in this report; the quick and dirty method, maximum likelihood estimation, single imputation and multiple imputation. The performances of the methods are tested for different types of censoring and censoring limits, and are evaluated by the gross variance, which is the expected mean square error. A low value of the gross variance indicates an accurate method.

Weibull distribution is the most commonly used distribution in survival analysis. Therefore, the four methods are tested for Weibull distributed data. The performances of the methods are evaluated through two different examples and two different experiments. Numerical results are obtained through implementations in the programming language R.

The numerical results show that all methods manage censored data, but the quality of the performances are varying. The quick and dirty method appears to be the most unstable method, while the imputation methods are the most reliable and precise methods. The results indicate that multiple imputation using the maximum likelihood estimator is the most accurate and safe method.

Sammendrag

Sensurering er et vanlig problem i eksperimenter og levetidsanalyse. Den eksakte verdien av en sensurert data er ukjent, men noe informasjon har en likevel om den sensurerte verdien. For eksempel er det mulig å vite om en sensurert data er større eller mindre enn en forutbestemt verdi, eller om den ligger mellom to bestemte verdier. Statistiske analysemetoder forutsetter fullstendige datasett. For å kunne benytte seg av statistiske metoder, må derfor sensurert data bli estimert til en verdi. Det er ønskelig at den estimerte verdien av en manglende data er så lik den originale verdien som overhodet mulig. Flere metoder har blitt utviklet for dette formålet, fire av dem blir vurdert og testet i denne masteroppgaven; "the quick and dirty method", "maximum likelihood estimation", "single imputation" og "multiple imputation". Prestasjonene til metodene blir testet for ulike typer sensurering og flere forskjellige sensureringsgrenser, og deretter blir de vurdert ut i fra verdien av bruttovariansen. Bruttovarians er forventet gjennomsnittlig kvadratfeil. En lav bruttovarians indikerer at metoden er presis.

Weibull fordeling er den mest brukte fordelingen i levetidsanalyse. Derfor blir de fire metodene testet for Weibull fordelte data. Prestasjonene til metodene blir vurdert gjennom to ulike eksempler og to ulike eksperimenter. Numeriske resultater oppnås via implementeringer i programmeringsspråket R.

De numeriske resultatene viser at alle metodene kan håndtere sensurerte data, men kvaliteten på prestasjonene er varierende. "The quick and dirty method" fremstår som den mest ustabile metodene, mens imputeringsmetodene er de mest pålitelige og presise metodene. Resultatene indikerer at "multiple imputation" som benytter "maximum likelihood estimation" til å sette startverdier er den mest presise and tryggeste metoden.

Contents

Chapter 1

Introdution

Censored data is a relevant issue in many different contexts, such as in reliability and survival analysis. Censored data is a special case of missing data problems, but whilst the data is completely unknown in missing data problems, the data is only partially unknown in censored data problems. When a data is censored, one does not know the exact value of the data, but one does have some information about the data. For instance one can know whether a data is greater or lower than a certain value or if it lies within a specified interval. These prespecified limits are called censoring limits. If a value lies above a censoring limit, below a censoring limit or within two censoring limits, we say that the value is censored. A value can be either right censored, left censored or interval censored.

Censoring is commonly used in experiments, such as in investigating the survival time of a patient or the lifetime of a component. Experimental design helps us make the most of an experiment, by allowing us to design the experiment such that it assures that we achieve the desired information. When performing experiments, one does not have unlimited amount of money nor time, such that shortening the project's total lifetime by deciding a censoring limit might be practical. And in cases were censoring is present, it is necessary to know how to handle the censored data. Many techniques and methods have been developed in order to able to manage censored data. In this report, four of these methods; the quick and dirty method, maximum likelihood estimation, single imputation and multiple imputation will be tested. The method's ability to manage experimental design with censored data will be investigated.

As mentioned, censoring is a common factor in survival analysis, and in survival analysis, the data is often Weibull distributed. Weibull distribution is frequently applied in such analysis, because of its properties, among others that the domain of a Weibull distributed variable is ranging from 0 to ∞. For a Weibull distributed variable, Y, the linear model can be expressed as

$$\mathbf{Y} = \ln\mathbf{T} = \mathbf{x}^T\boldsymbol{\beta} + \sigma\boldsymbol{\epsilon},$$

where ϵ is standard extreme value distributed.

The first chapters of this thesis contain theory. In chapter 2, censoring is explained, the two probability distributions applied in the thesis, the Weibull distribution and the generalized extreme value distribution, are presented and the maximum likelihood estimates for the parameters of a Weibull distribution are derived. Chapter 3 introduces linear regression; linear regression models, experimental design and truncation are described. Truncation is applied in the method of multiple imputation.

Chapter 4 presents the four methods to be investigated; the quick and dirty method, maximum likelihood estimation, single imputation and multiple imputation. In addition,

the R functions applied in the implementation of the methods are described. The models considered in this thesis include an error term, which needs to be simulated. This simulation is also described in chapter 4. To evaluate the performances of the four methods, the gross variances of the regression coefficients are computed. The gross variance is introduced in chapter 4.

In chapter 5, the two models considered and the two different experiments are presented. Both models are linear models, but the complexity is different.

The results of the first experiment are given and discussed in chapter 6. In chapter 7, the results of the second experiment are presented and discussed.

The performance of the four methods are discussed in chapter 8, based on the results presented in chapter 6 and 7. Concluding remarks are given in chapter 9.

In appendix A, the implementation of the four methods for example 1 and experiment 1 are presented. All implementation is performed in R.

Chapter 2

Theory

2.1 Censoring

Missing data is a problem in many different contexts, such as in surverys, measurements and studies. In the problem of missing data, some observed values of a variable are unknown. Censoring is a type of missing data problem, where the observations are only partially unknown. Different reasons may cause censoring, such as an object withdrawing from a study, an object being lost to follow up, or a study being ended earlier than intended. When dealing with censoring, one does not know exactly when an event occurs. However, one may have some information about the time of occurrence, either by knowing a specific time at which the event has not yet occurred, a time the event has occurred within or a time interval the event has occurred within. Whether an item i is censored or observed can be denoted by an event indicator, δ_i. If an item is observed, we have $\delta_i = 1$, while if the item is censored we use $\delta_i = 0$. Different types of censoring are presented below.

2.1.1 Types of censoring

Type I censoring

Under Type I censoring, a sample of n units are tested, for which the experiment starts at a fixed time zero, $t = 0$, and ends at a time t_{end}. That is, one can only know the exact failure time of units which fail within t_{end}. Units that fail after the time t_{end}, are not observed. For Type I censoring, the total duration of the study is fixed, while the number of censored units is random.

Type II censoring

Under Type II censoring, a sample of n units are testet, for which the experiment starts at a fixed time zero, $t = 0$, and terminates when a fixed number of units, $r(r < n)$, have failed. Failure times are only observed for the r units, that is units failing after the rth unit has failed, are not observed. The total number of censored units is fixed, while the experimental time is random.

Random censoring

In random censoring, the total period of the experiment is fixed, but the units to be studied enter the study at different times. Each item is assumed to have its own failure time and censoring time, and the failure time and the censoring time are independent of each other.

As a result of the independence, the censoring time gives no information about the failure time, which is why random censoring also is referred to as non-informative censoring.

Right censoring

Right censoring occurs if some units still function at the termination time. The censored time will therefore be smaller than the actual failure time. Right censoring is the most common type of censoring in survival analysis. A study of divorces can illustrate right censoring. Couples who are still married when the study ends or drop out of the study for some reasons other than divorce, are right censored. The unit A in figure 2.1 is right censored, because the only knowledge we have of observation A is that it occurred after the end of study, time 4.

Left censoring

If the failure time is only known to be lower than a certain value, the unit is left censored. The item has failed before we start observing, hence the censored time will be greater than the actual failure time. As an example of left censoring, the time children learn to swim can be used. If a study observes the time a child first learns to swim, and the study starts at a fixed time, some children may have learned to swim already. The children who know how to swim before the first observation time, are left censored. In figure 2.1, the observation C is left censored, as we only know that the observation has occured before time 1.

Interval censoring

When dealing with interval censoring, one only knows that the failure times lie within a time interval of two fixed times. Interval censoring is typical for items that need to be tested. Testing ones sight can be used as an example of interval censoring. It is not possible to know exactly when the sight became poorer, but one can know that it happened between the previous testing time and the current testing time. The observation B in figure 2.1 is interval censored, because we don't know the exact time of occurence, but we do know that it happend somewhere between time 2 and time 3.

In this report, only Type I censoring, right censoring and left censoring will be considered.

More theory on missing data problems can be found in Little & Robin [5] and more theory on censoring can be found in Wu & Hamada [10]

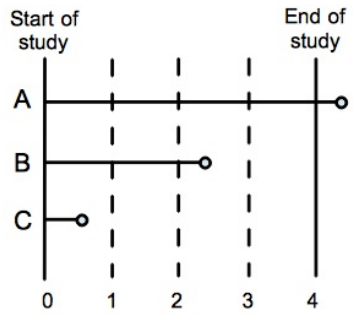


Figure 2.1: Types of censoring.

2.2 Distributions

2.2.1 Weibull distribution

The Weibull distribution is a continuous probability distribution widely applicable in probability theory and statistics. The Weibull distribution was first identified by Fréchet in 1927, and first applied by Rosin & Rammler in 1933, but Waloddi Weibull was the first to study and describe the distribution in detail in 1951, which is where the name of the distribution originates from. While the original purpose of the distribution was to model physical fatigue, the Weibull distribution has many more applications because of its flexibility, and is widely used in problems of reliability and survival analysis.

The probability density function of a Weibull random variable, X, is defined as

$$f(x;\theta,\alpha) = \begin{cases} \frac{\alpha}{\theta}\left(\frac{x}{\theta}\right)^{\alpha-1} e^{-(x/\theta)^{\alpha}}, & x \geq 0, \\ 0, & x < 0, \end{cases} \tag{2.1}$$

where $\alpha > 0$ is the shape parameter and $\theta > 0$ is the scale parameter of the distribution, as defined in Wu & Hamada [10]. The Weibull distribution is related to other probability distributions, such as the exponential distribution for $\alpha = 1$ and the Rayleigh distribution when $\alpha = 2$. If the quantity, X, is "time to failure", the Weibull distribution gives a distribution for which the failure rate is proportional to a power of time, interpreted as follows;

- $\alpha < 1$: The failure rate decreases over time,
- $\alpha = 1$: The failure rate is constant over time,
- $\alpha > 1$: The failure rate increases over time.

The cumulative distribution function for the Weibull distribution is

$$F(x;\theta,\alpha) = \begin{cases} 1 - e^{-(x/\theta)^{\alpha}}, & x \geq 0, \\ 0, & x < 0. \end{cases} \tag{2.2}$$

The mean and variance of the Weibull distribution are expressed as

$$\mathrm{E}(X) = \theta\Gamma\Big(1 + \frac{1}{\alpha}\Big)$$

and

$$\mathrm{var}(X) = \theta^2\left[\Gamma\left(1 + \frac{2}{\alpha}\right) - \left(\Gamma\left(1 + \frac{1}{\alpha}\right)\right)^2\right],$$

where $\Gamma(\cdot)$ is the gamma function [1].

2.2.2 Generalized extreme value distribution

In probability theory and statistics, the generalized extreme value (GEV) distribution is a family of continuous probability distributions developed within extreme value theory to combine the Gumbel, Fréchet and Weibull families, respectively referred to as type I, type II and type III extreme value distributions. The GEV distribution originates from the extreme value theorem (Fisher-Tippett, 1928 and Gnedenko, 1943) and is the limit

[1] The gamma function is defined as $\Gamma(n) = (n-1)!$ if n is a positive integer.

distribution of properly normalized maxima of a sequence of independent and identically distributed random variables. Because of this, the GEV distribution is used as an approximation to model the maxima of long (finite) sequences of random variables.

The probability density function of a GEV(μ, σ, ξ) distribution is

$$f(x;\mu,\sigma,\xi) = \frac{1}{\sigma}\left[1+\xi\left(\frac{x-\mu}{\sigma}\right)\right]^{(-1/\xi)-1}\exp\left\{-\left[1+\xi\left(\frac{x-\mu}{\sigma}\right)\right]^{-1/\xi}\right\},$$

for $1+\xi(x-\mu)/\sigma > 0$, where $\mu \in \mathbb{R}$ is the location parameter, $\sigma > 0$ is the scale parameter and $\xi \in \mathbb{R}$ is the shape parameter, as defined on Wikipedia [11]. The cumulative distribution function is defined as

$$F(x;\mu,\sigma,\xi) = \exp\left\{-1\left[1+\xi\left(\frac{x-\mu}{\sigma}\right)\right]^{-1/\xi}\right\}.$$

The shape parameter, ξ, controls the tail behaviour of the distribution. Different values of ξ refers to the three extreme value distributions, whose cumulative distribution functions are defined as follows

- Gumbel (type I extreme value distribution), $\xi = 0$

$$F(x;\mu,\sigma,0) = e^{-e^{-(x-\mu)/\sigma}} \quad \text{for } x \in \mathbb{R}.$$

- Fréchet (type II extreme value distribution), $\xi = \alpha^{-1} > 0$

$$F(x;\mu,\sigma,\xi) = \begin{cases} 0, & x \le \mu, \\ e^{-((x-\mu)/\sigma)^{\alpha}}, & x > \mu. \end{cases}$$

- Reversed Weibull (type III extreme value distribution), $\xi = -\alpha^{-1} < 0$

$$F(x;\mu,\sigma,\xi) = \begin{cases} e^{-(-(x-\mu)/\sigma)^{\alpha}}, & x < \mu, \\ 1, & x \ge \mu. \end{cases}$$

Type I is the most commonly referred to in discussions of extreme values, and is also the one being considered in this report, when the extreme value distribution is used. The cumulative distribution function and the probability density function of extreme value distribution can be expressed in several ways. The way of representing the probability density function in this report is presented in section 12.2 in Wu & Hamada [10];

$$f(x) = \frac{1}{\sigma}\exp\Big[\frac{x-\mu}{\sigma} - \exp\Big(\frac{x-\mu}{\sigma}\Big)\Big]. \tag{2.3}$$

The corresponding cumulative distribution function is

$$F(x) = \int_{-\infty}^{x} f(t)dt = 1 - \exp\Big[-\exp\Big(\frac{x-\mu}{\sigma}\Big)\Big]. \tag{2.4}$$

The mean and the variance of the Gumbel distribution is given as

$$\begin{aligned} \mathrm{E}(-X) &= -(-\mu+\gamma\sigma) = \mu - \gamma\sigma, \\ \mathrm{var}(-X) &= \frac{\pi^2}{6}\sigma^2, \end{aligned} \tag{2.5}$$

from Wikipedia [12], where γ is the Euler-Mascheroni constant, $\gamma \approx 0.5772$.

The reason for using $-X$ in the expressions of the mean and the variance lies in the definition of the probability distribution. On Wikipedia [12], the random variable X and the location parameter μ are used in the cumulative distribution function for Gumbel distribution [2]. which is rewritten in this report to (2.4) such that the random variable is defined as $-X$ and the location parameter is defined as $-\mu$.

In the method of multiple imputation, the quantile function of the Gumbel distribution is used when computating the imputed values. The quantile function, $Q(p)$, of a probability distribution is the inverse of its cumulative distribution function, $F(x)$. For a given probability in the probability distribution of a random variable, the quantile function specifies the value at which the probability of the random variable will be less than or equal to that probability. Hence, the quantile function for the extreme value distribution, $Q(p) = x = F^{-1}(p)$ is derived as

$$\begin{aligned} F(x) &= 1 - \exp\Big[- \exp\Big(\frac{x-\mu}{\sigma}\Big)\Big] = p \\ \ln(1-p) &= -\exp\Big(\frac{x-\mu}{\sigma}\Big) \\ \ln\big(\ln(1-p)\big) &= \frac{x-\mu}{\sigma} \\ \Rightarrow Q(p) &= x = \mu + \sigma \ln\big(- \ln(1-p)\big). \end{aligned}$$

2.3 The maximum likelihood estimator

Maximum likelihood estimation is a method of estimating unknown parameters of a statistical model, and these parameters are obtained by maximizing the likelihood function of that model. The likelihood function is the probability density function of the joint distribution of the data of a sample or a continuous/ discrete random variable. The likelihood function contains the parameters of a statistical model and is sometimes just referred to as the likelihood. The likelihood of a set of parameter values, θ, given some observed outcomes, t, is equal to the probability of those observed outcomes given the parameter values;

$$\mathcal{L}(\theta|t) = (t|\theta) = \prod_{i=1}^{n} f(t_i, \theta).$$

The logarithm is taken of the likelihood function, which is a practical and valid procedure as the logarithm is a monotonically increasing function. To obtain expressions for the parameters, the partial derivatives of the log likelihood with respect to the parameters are set equal to zero.

More information about the method of maximum likelihood can be found in Warpole, Myers, Myers & Ye [9].

2.3.1 Maximum likelihood estimation for Weibull distribution

The derivation of the maximum likelihood estimates for the Weibull distribution is based on the derivation in Joseph [4].

The density function for the Weibull distribution, as presented in section 2.2.1, is

[2] $F(x; \mu, \sigma) = \exp(-\exp(-(x-\mu)/\sigma))$.

$$f(t;\theta,\alpha) = \Big(\frac{\alpha}{\theta}\Big)\Big(\frac{t}{\theta}\Big)^{\alpha-1} e^{(-t/\theta)^{\alpha}}.$$

The likelihood function of a Weibull distribution is

$$\begin{aligned} L(\theta,\alpha) &= \prod_{i=1}^{n}\Big(\frac{\alpha}{\theta}\Big)\Big(\frac{t_i}{\theta}\Big)^{\alpha-1} e^{(-t_i/\theta)^{\alpha}} \\ &= \Big(\frac{\alpha}{\theta^{\alpha}}\Big)^{n}\prod_{i=1}^{n} t_i^{\alpha-1} e^{(-1/\theta)^{\alpha}}\sum_{i=1}^{n} t_i^{\alpha}. \end{aligned}$$

Applying the natural logarithm we obtain

$$\ln L(\theta,\alpha) = l(\theta,\alpha) = n\ln\alpha - \alpha n\ln\theta + (\alpha-1)\sum_{i=1}^{n}\ln t_i - \frac{1}{\theta^{\alpha}}\sum_{i=1}^{n} t_i^{\alpha}.$$

Calculating the partial derivatives of α and θ and setting them equal to 0, gives us

$$\frac{\partial l(\theta,\alpha)}{\partial\theta} = -\frac{\alpha n}{\theta} + \frac{\alpha}{\theta^{\alpha+1}}\sum_{i=1}^{n} t_i^{\alpha} = 0 \tag{2.6}$$

and

$$\begin{aligned} \frac{\partial l(\theta,\alpha)}{\partial\alpha} &= \frac{n}{\alpha} - n\ln\theta + \sum_{i=1}^{n}\ln t_i - \frac{\partial}{\partial\alpha}\sum_{i=1}^{n}\Big(\frac{t_i}{\theta}\Big)^{\alpha} \\ &= \frac{n}{\alpha} - n\ln\theta + \sum_{i=1}^{n}\ln t_i - \sum_{i=1}^{n}\Big(\frac{t_i}{\theta}\Big)^{\alpha}\ln\Big(\frac{t_i}{\theta}\Big). \end{aligned} \tag{2.7}$$

Equation (2.6) can be expressed as

$$\hat{\theta} = \left(\frac{1}{n}\sum_{i=1}^{n} t_i^{\hat{\alpha}}\right)^{1/\hat{\alpha}},$$

and equation (2.7) can then be solved numerically with $\hat{\theta}$ inserted.

Maximum likelihood estimation for right censored observations

Let the density function be $f(t_i;\theta,\alpha)$, the distribution function be $F(t_i;\theta,\alpha)$ and the survival function be $S(t_i;\theta,\alpha)$. The probability of an item surviving a specific time, t_i, is defined by

$$\begin{aligned} S(t_i;\theta,\alpha) &= P(T > t_i) = \int_{t_i}^{\infty} f(u;\theta,\alpha)\mathrm{du} = F(\infty;\theta,\alpha) - F(t_i;\theta,\alpha) \\ &= 1 - F(t_i;\theta,\alpha). \end{aligned}$$

Suppose n items are put to the test, where r units fail and $n-r$ units do not fail within a time limit, C_r. The lifetime and censoring can be defined as (Y_i,δ_i), where

$$Y_i = \begin{cases} T_i, & \delta_i = 1, \quad \text{for uncensored data,} \\ \min(T_i, C_r), & \delta_i = 0, \quad \text{for right censored data,} \end{cases}$$

where C_r is the time limit, the right censored time. The contribution to the likelihood with right censored observations is the product of the units that failed and the units that did not fail. The contribution of a unit failing at y_i is the density of that time interval; $L_i = f(y_i; \theta, \alpha)$. For a unit still functioning at y_i, meaning the lifetime of the unit exceeds y_i, the contribution to the likelihood is $L_i = S(y_i)$. Thus, the likelihood can be expressed as

$$\begin{aligned} L(\theta, \alpha) &= \prod_{i=1}^{n} L_i(\theta, \alpha) = \prod_{\delta_i=1} f(y_i; \theta, \alpha) \prod_{\delta_i=0} S(y_i; \theta, \alpha) \\ &= \prod_{i=1}^{r} f(t_i; \theta, \alpha) \prod_{i=r+1}^{n} S(t_i; \theta, \alpha). \end{aligned} \tag{2.8}$$

For Weibull distribution, the survival function is defined as $S(t_i) = e^{-(t_i/\theta)^\alpha}$ and the density function is defined in (2.1). Thus, the expression in (2.8) becomes

$$\begin{aligned} L(\theta, \alpha) &= \prod_{i=1}^{r} \frac{\alpha}{\theta}\Big(\frac{t_i}{\theta}\Big)^{\alpha-1} e^{-(t_i/\theta)^\alpha} \prod_{i=r+1}^{n} e^{-(t_i/\theta)^\alpha} \\ &= \Big(\frac{\alpha}{\theta^\alpha}\Big)^r \prod_{i=1}^{r} t_i^{\alpha-1} \exp\Big(-\frac{1}{\theta^\alpha}\sum_{i=1}^{n} t_i^\alpha\Big). \end{aligned}$$

Applying the natural logarithm, we obtain

$$\ln L(\theta, \alpha) = l(\theta, \alpha) = r\ln\alpha - \alpha r \ln\theta + (\alpha - 1)\sum_{i=1}^{r} \ln t_i - \frac{1}{\theta^\alpha}\sum_{i=1}^{n} t_i^\alpha.$$

Then, we calculate the partial derivatives of the log likelihood with respect to θ and α, and set them equal to 0, as follows

$$\frac{\partial l(\theta, \alpha)}{\partial \theta} = -\frac{\alpha r}{\theta} + \frac{\alpha}{\theta^{\alpha+1}}\sum_{i=1}^{n} t_i^\alpha = 0$$

and

$$\frac{\partial l(\theta, \alpha)}{\partial \alpha} = \frac{r}{\alpha} - r\ln\theta + \sum_{i=1}^{r} \ln t_i - \sum_{i=1}^{n} \Big(\frac{t_i}{\theta}\Big)^\alpha \ln\Big(\frac{t_i}{\theta}\Big).$$

The partial derivative with respect to θ can be rearranged to express $\hat{\theta}$ as

$$\hat{\theta} = \left(\frac{1}{r}\sum_{i=1}^{n} t_i^{\hat{\alpha}}\right)^{1/\hat{\alpha}},$$

while the partial derivate with respect to α is to be solved numerically with $\hat{\theta}$ inserted.

Maximum likelihood estimation for left censored observations

The probability for a left censored observation is given as

$$P(T < t_i) = T(t_i; \theta, \alpha) - F(-\infty; \theta, \alpha) = F(t_i; \theta, \alpha),$$

where the cumulative distribution function is

$$F(t_i;\theta,\alpha) = 1 - S(t_i;\theta,\alpha).$$

When n items are considered, and among them r items are functioning at the censoring limit C_l, while $n-r$ units are not functioning at C_l. The lifetime and censoring can then be expressed as

$$Y_i = \begin{cases} T_i, & \delta_i = 1, \quad \text{for uncensored data,} \\ \max(T_i, C_l), & \delta_i = 0, \quad \text{for left censored data,} \end{cases}$$

where C_l is the left censoring limit. As for the right censored case, the contribution to the maximum likelihood with left censored data is the product of the units that failed and the units that did not fail. If a unit fails at y_i, the contribution to the likelihood function is the cumulative distribution of that time interval, $L_i = F(y_i;\theta,\alpha)$. If the lifetime of a unit do not reach y_i, meaning that the unit is not functioning at y_i, the contribution is the density of that time interval, $L_i = f(y_i)$. The likelihood can then be written as

$$\begin{aligned} L(\theta,\alpha) &= \prod_{i=1}^{n} L_i(\theta,\alpha) = \prod_{\delta_i=1} f(y_i;\theta,\alpha) \prod_{\delta_i=0} F(y_i;\theta,\alpha) \\ &= \prod_{i=1}^{r} f(t_i;\theta,\alpha) \prod_{i=r+1}^{n} F(t_i;\theta,\alpha). \end{aligned} \tag{2.9}$$

The likelihood for Weibull distributed data containing left censored observations given in (2.9), for which the density function is defined in (2.1) and the cumulative distribution function in (2.2), is

$$L(\theta,\alpha) = \prod_{i=1}^{r} \left(\frac{\alpha}{\theta}\right)\left(\frac{t_i}{\theta}\right)^{\alpha-1} e^{-(t_i/\theta)^{\alpha}} \prod_{i=r+1}^{n} \left(1 - e^{-(t_i/\theta)^{\alpha}}\right).$$

Chapter 3

Regression

3.1 Linear regression model

A regression model is a statistical technique for modelling the relationship between a response variable and one or more explanatory variables. The outcome of the response variable depends on the explanatory variables, but not the other way around. Therefore, the response variable is also called the dependent variable, while the regression variable is also known as the independent variable. An important element of regression analysis is the estimation of the regression function, a function that describes how the response variable is related to the explanatory variables.

The regression of a random variable Y on a variable x, is the expectation of Y given the value of x, written as $E(Y|x)$. A linear regression model is given by

$$E[Y|x] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + .. + \beta_k x_k + \epsilon,$$

where

- Y is the response variable,
- $x_1, x_2, ..., x_k$ are the explanatory variables,
- $\beta_0, \beta_1, \beta_2, ..., \beta_k$ are the regression coefficients,
- ϵ is the random error. The errors are normally assumed uncorrelated with equal variance and expectation equal to zero.

β_i determines to what extent each explanatory variable x_i contributes to the response variable Y, given that the others are kept constant.

"The least square method" is the most commonly used method for estimating the unknown β's.

3.2 Experimental design

Experimentation allows an investigator to figure out how the response, also known as the output, responds when the settings of the input variables in a system are intentionally changed. This can be used to understand how the input variables affect the performance of a system, thus provide a basis for choosing the optimal input settings. In addition, the motivation for performing an experiment could be to identify the significant factors.

The principles of experimental design was introduced by R. A. Fisher in the 1930s and have been widely used in various disciplines such as medicine, psychology, business and

science ever since. Experimental design is described in Wu & Hamada [10] as "a body of knowledge and techniques that enables an investigator to conduct better experiments, analyse data efficiently, and make the connections between the conclusions from the analysis and the original objectives of the investigation". Planning an experiment properly is very important in order to be able to answer the research questions of interest as effectively and clearly as possible. One should carefully consider the objective of the experiment when choosing responses, factors, levels and the experimental plan to be conducted.

When performing an experiment the factors are the input and can be considered as the explanatory variables of a regression model, where an appropriate response is also chosen. If one wishes to investigate how the selling price of a house is affected by several factors, the selling price is the chosen response. Factors could be the size of the house, number of bed rooms, year of construction, location etc. The levels describe the extent to which each factor impacts the response and could be denoted as "high" or "low". A factorial experiment considering k factors and p levels is expressed as a p^k factorial design.

Fractional factorial design is preferable when the number of factors to be investigated is large. If for example one considers an experiment of 10 factors, each with two levels, the number of possible single experiments is $2^{10} = 1024$. It is nearly impossible to perform 1024 experiments under the exact same circumstances, and a change in the experimental conditions may lead to wrong estimates of the effects. In addition, it would be time consuming and require a large number of resources to perform that many experiments. Experimental design allows us to decrease the amount of experimental work considerably in relation to the complete factorial experiment by applying blocking or fractional factorial design, which is explained later. Blocking is the arranging of experimental units in groups (blocks) that are similar to one another, hence the variability between units is reduced and more accurate estimates are achieved.

In this report, only 2^3 factorial design will be applied and blocking will not be considered.

3.2.1 Two-level factorial designs

Full factorial two-level experiments are referred to as 2^k designs where k denotes the number of factors to be investigated in the experiment, with two levels. If the two levels are chosen to be "high" and "low", it would be simple and convenient to present "low" by a negative (-) sign or -1, and let a positive (+) sign or 1 denote "high". Then we obtain orthogonal factor columns and the coefficients can easily be computed. The design of a 2^3 factorial experiment is shown in table 3.1.

Table 3.1: A 2^3 level design.

Run no.	Factor A	Factor B	Factor C	Levelcode	Yields (y_i)
1	-	-	-	(1)	y_1
2	+	-	-	a	y_2
3	-	+	-	b	y_3
4	+	+	-	ab	y_4
5	-	-	+	c	y_5
6	+	-	+	ac	y_6
7	-	+	+	bc	y_7
8	+	+	+	abc	y_8

To determine how significant the factors are, several effects are computed. The main effect is the difference between the mean response at the high level and the mean response

at the low level of a factor. When performing an experiment with more than one factor, interactions between two or more factors may be present. Interaction means that the effect of one factor may depend on the level of the other factors. The interaction effect between two factors is defined as the difference between the main effect of one factor when the other factor is at its high level and the main effect of the first factor when the other is at its low level.

The main effect of a factor, using factor A in table 3.1 as an example, is computed as

$$A = \frac{y_2 + y_4 + y_6 + y_8}{4} - \frac{y_1 + y_3 + y_5 + y_7}{4}.$$

The estimation of the interaction effects for the two-factor interactions and three-factor interactions are presented below, with A and B, and A, B and C as examples.

$$\begin{aligned} AB &= \frac{y_1 + y_4 + y_5 + y_8}{4} - \frac{y_2 + y_3 + y_6 + y_7}{4}, \\ ABC &= \frac{y_2 + y_3 + y_5 + y_8}{4} - \frac{y_1 + y_4 + y_6 + y_7}{4}. \end{aligned}$$

3.2.2 Two-level fractional factorial designs

When the number of factors increases, the number of single experiments will consequently increase too, and actions that reduce the number of required experiments should be considered. One way to decrease the number of required runs, is to choose a fraction of the total runs, to be used in the estimation. The selection of runs is preferably chosen such that the main effects and the lower order interactions can be estimated, as the higher order interactions often are assumed to be negligible. Guo and Mettas [1] refers to it as the sparsity of effects principle. Such a selection procedure is called a fractional factorial design. A two-level fractional factorial design is expressed as 2^{k-p}, where k is the number of factors, p represents the number of generators and $2^{-p} = \frac{1}{2^p}$ denotes the fraction. If the signs of one factor are equal to the sign products of some other factors, it is said to be a generator of the design.

For example, a 2^{3-1} design can be generated by constructing a full two-level factorial experiment involving two factors, A and B, and then assign the factor C to the interaction AB. The design is displayed in table 3.2. If the effect of ABC can be ignored due to the sparsity of effects principle, the number of required runs can be reduced by only considering the one half of the full factorial experiment where ABC has the same level (+). A $\frac{1}{2}$ fraction of the 2^3 design expanded with interaction columns is shown in table 3.3. As AB and C have identical signs, a generator for the design is C=AB. Since the signs of the column AB and the column C are exactly the same, there is no possibility of separating these effects from such an experiment. Such effects are called confounded effects or aliased effects, meaning they have the same signs in the factor columns. The aliasing relation is denoted by C=AB, which indicates that the defining relation is I=ABC (if the turns where ABC had level - were chosen, the defining relation would be I=-ABC). In order to identify which effects that are aliased, the effects must be multiplied by the defining relation. The defining relation, I, is just a unity and if multiplied with the effects of this example, the confounding pattern will be

$$\begin{aligned} A &= A(I) = A(ABC) = BC, \\ B &= B(I) = B(ABC) = AC, \\ C &= C(I) = C(ABC) = AB. \end{aligned}$$

Table 3.2: A 2^{3-1} factorial design.

Run no.	Factor A	Factor B	Factor C
1	-	-	+
2	+	-	-
3	-	+	-
4	+	+	+

Table 3.3: A $\frac{1}{2}$ fraction of a 2^3 design expanded with interaction columns.

Run no.	A	B	C	AB	AC	BC	ABC
1	-	-	+	+	-	-	+
2	+	-	-	-	-	+	+
3	-	+	-	-	+	-	+
4	+	+	+	+	+	+	+

3.2.3 Resolution

The resolution of a design is defined as the ability to separate main effects from lower order interactions or as the number of factors in the lowest order effect in the defining relation (the shortest word in the defining relation). In a 2^{k-p} fraction design, resolution R is obtained if no p factor effect is aliased with an effect containing less than $R-p$ factors. In other words, for a resolution R design, the main effects are aliased with $R-1$ factor interactions. The three most important fractional designs are those of resoultion III, IV and V:

Resolution III

The main effects are aliased with two-factor interactions. For a 2^{3-1} design, the defining relation is I=ABC. In other words, the length of the shortest word in the defining relation is three.

Resolution IV

The main effects are aliased with three-factor interactions and two-factor interactions are aliased with other two-factor interactions. For a 2^{4-1} design, the length of the shortest word in the defining relation is four, I=ABCD.

Resolution V

For a resolution of V, the main effects are aliased with four-factor interactions, while two-factor interactions are aliased with three-factor interactions. In a 2^{5-1} design, the defining relation is I=ABCDE. The length of the shortest word in the defining relation is five.

3.3 Truncation

Truncation is described in Sue-Chu [8]. In mathematics, truncation refers to the process of limiting the amount of digits in a number by discarding the least significant ones. Statistical truncation is the term of measurements that have been ended abruptly at some specific value. The knowledge of items that fall outside the specified time interval, is what distinguish truncation from censoring. When values lie ouside the censoring limit, the

values are recorded as censored values, while when values lie outside the truncated limit, the values are not recorded at all. That is, there exists no information about a truncated value.

A truncated distribution is a conditional distribution that results from restricting the domain of a probability distribution. By restricting the domain, the result is a truncated sample, which can be arranged to include the most important data of an analysis. That is, truncation can be applied to any probability distribution, and will lead to a new distribution.

Assume a random variable, $\tilde{T}$, which is distributed with cumulative distribution function, $F_{\tilde{T}}(t)$. To obtain a new random variable T, T is set to have the distribution of $\tilde{T}$ truncated to the restricted domain $(a, b]$. The truncation distribution, $F_T(t)$ of T is then

$$F_T(t) = \begin{cases} 0, & \text{for } t < a, \\ \frac{F_{\tilde{T}}(t) - F_{\tilde{T}}(a)}{F_{\tilde{T}}(b) - F_{\tilde{T}}(a)}, & \text{for } a \leq t \leq b, \\ 1, & \text{for } t > b. \end{cases}$$

The corresponding probability density of T, for the domain $(a, b]$ is

$$f(t|a < T \leq) = \frac{g(t)}{F_{\tilde{T}}(b) - F_{\tilde{T}}(a)},$$

where

$$g(t) = \begin{cases} f_{\tilde{T}}(t), & \text{for all } a < t \leq b, \\ 0, & \text{otherwise.} \end{cases}$$

Then, a truncated distribution with right censoring, which means that the bottom of the distribution has been removed, is defined as

$$f(t|T > C_r) = \frac{g(t)}{1 - F_{\tilde{T}}(C_r)},$$

where $g(t) = f_{\tilde{T}}(t)$ for $C_r < t$ and 0 otherwise. When left censored, the top of the distribution has been removed and the truncated distribution is given as

$$f(t|T \leq C_l) = \frac{g(t)}{F_{\tilde{T}}(C_l)},$$

where $g(t) = f_{\tilde{T}}(t)$ for $C_l \geq t$ and 0 otherwise.

It is possible to generate random variables from a truncated distribution. Consider a non-negative random variable $\tilde{T}$ with the probability distribution function $F_{\tilde{T}}(t) = \int_{t'=0}^{t} f_{\tilde{T}}(t')dt'$, where $f_{\tilde{T}}(t)$ is the density function for $0 \leq t \leq \infty$. Scaled truncation can be used to generate random variables from the truncated distribution $f_T(t)$, for $a \leq t \leq b$. The psuedo code for generating a random variable T, where T represents the quantile function of the extreme value distribution of V, is given as

- Generate $U \sim U[0, 1]$.
- Let $V = F_{\tilde{T}}(a) + [F_{\tilde{T}}(b) - F_{\tilde{T}}(a)] \times U$.
- Return $T = F_{\tilde{T}}^{-1}(V)$.

Scaled truncation is being used later in multiple imputation, to generate random variables from the GEV distribution.

Chapter 4

The methods

4.1 R functions

R is a software programming language and software environment for statistical computing and graphics. R is an implementation of the S programming language, which was created by John Chambers. Ross Ihaka and Robert Gentleman created R in 1993[3] and released the R source code as "free software" in 1995.

The R language is widely used among statisticans and other scientists for developing statistical software and data analysis. R provides a wide variety of statistical and graphical techniques, such as linear and nonlinear modelling, classical statistical tests, time series analysis, classification and so on. In addition, R is highly extensible; users can improve the code of the software or write variations for specific tasks.

In R, a large number of packages and built-in mechanisms are available. Two embedded functions that are commonly used in this report are the *lm* function of the *stats* package and the *survreg* function from the *survival* package. The R code for the four methods can be found in Appendix A.

4.1.1 The *lm* function

The *lm* function is used to fit linear models. The function can be used to perform regression, single stratum analysis of variance and analysis of covariance. The user needs to specify the form of the model to be analysed, typically the form response $\sim$ terms, where response is the response vector and terms is a series of terms which specifies a linear predictor for the response. The *lm* function returns among other the coefficients of the specified model. The simple model, $\mathbf{y} = \beta_0 + \beta_1\mathbf{x}$, can be estimated by the *lm* function as

```
fitlm <- lm(y ~ x)
```

Estimated values of β_0 and β_1 will be returned.

4.1.2 The *survreg* function

The *survreg* function is used to fit parametric survival regression models. The basic input of the *survreg* function is a formula expression, where the response usually is a survival object as returned of the *Surv* function, and the assumed distribution for the response variable. The *Surv* function creates a survival object, which is usually used as a response variable in a model formula, for example in the *survreg* function, as mentioned. The inputs of the *Surv* function are the event times and the types of events (status indicator). The distributions that can be fitted are "weibull", "exponential", "gaussian",

"logistic", "lognormal" and "loglogistic". If one wishes to fit a Weibull model; $\mathbf{y} = \log\mathbf{T} = \beta_0+\beta_1\mathbf{x}_1+\beta_2\mathbf{x}_2+\beta_3\mathbf{x}_3$, based on an example found on page 473 in Warpole, Myers, Myers & Ye [9], the following code can be applied

```
x1 <- c(-1,1,-1,-1,1,1,-1,1)
x2 <- c(-1,-1,1,-1,1,-1,1,1)
x3 <- c(-1,-1,-1,1,-1,1,1,1)

y <- c(7.6,8.4,9.2,10.3,9.8,11.1,10.2,12.6)
status <- c(1,1,1,1,1,0,1,0)

fitsurvreg <- survreg(formula=Surv(y,status)~x1+x2+x3,dist="weibull")
summary(fitsurvreg)
```

The *survreg* function uses the Newton's method (also known as the Newton-Raphson method), and returns among other estimated coefficients (β_0, β_1, β_2 and β_3 in this case) and parameter values (the shape parameter α and the scale parameter θ in this case).

The *survreg* function will be used to estimate Weibull distributed data, and in the case of Weibull distribution it is important to be aware of how *survreg* returns the parameters. The output of the previous fitted model is

```
Call:
survreg(formula = Surv(y, status) ~ x1 + x2 + x3, dist = "weibull")
              Value Std. Error      z        p
(Intercept)  2.3110     0.0141 164.41 0.00e+00
x1           0.0500     0.0142   3.52 4.34e-04
x2           0.0767     0.0149   5.16 2.50e-07
x3           0.1414     0.0155   9.10 9.24e-20
Log(scale)  -3.5060     0.3816  -9.19 4.00e-20

Scale= 0.03

Weibull distribution
Loglik(model)= -4   Loglik(intercept only)= -14.8
Chisq= 21.61 on 3 degrees of freedom, p= 7.9e-05
Number of Newton-Raphson Iterations: 15
n= 8
```

The probability density function of the Weibull distribution is defined to be $f(t) = \frac{\alpha}{\theta}\left(\frac{x}{\theta}\right)^{\alpha-1}e^{-(x/\theta)^{\alpha}}$. Then, the shape parameter α and the scale parameter θ are defined by the output of R as

$$\begin{aligned} \alpha &= 1/\text{scale output} = 1/0.03, \\ \theta &= \exp(\text{Intercept}) = \exp(2.3110). \end{aligned}$$

The column "Value" in the R output shows the regression coefficients. These regression coefficients are half the size of estimated main effects. Main effects were introduced in section 3.2. This is because a main effect measures the change in the expected response when we move from the low level, -1, to the high level, +1, of the factor, while a regression coefficient measures the change in the expected response when the factor changes from 0 to 1. Regression coefficients are used in the numerical experiments in this report.

For some reason, the *survreg* function does not handle left censoring as well as it handles right censoring. The maximum likelihood estimator converges significantly less times for left censoring than for right censoring and in addition the estimated regression coefficients are less precise.

As mentioned earlier, the implementations are performed in R, but at one point, computations are performed in MATLAB through the code in R. In the implementation of single imputation, MATLAB is used to compute the exponential integral, which appears in the computation of the conditional expected value. The two software languages are combined in that R sends the needed input to MATLAB, MATLAB computes the exponential integral and then returns the output of the computation back to R. This operation requires some time, but this is to my knowledge the easiest way to solve the problem.

4.2 Simulation of the error term and Weibull regression

The cumulative distribution function for Weibull distribution is defined in section 2.2.1 as

$$F_T(t) = P(T \leq t) = 1 - e^{(-t/\theta)^\alpha}.$$

A standard linear regression modelling of t is not appropriate for the models considered in this thesis (to be introduced in chapter 5), because Weibull distribution only considers non-negative values of t. Hence, it should be transformed to take values from $-\infty$ to ∞. Using the transformation $Y = \ln T$ we get

$$\begin{aligned} F_Y(y) &= P(Y \leq y) = 1 - e^{(-y/\theta)^\alpha} = P(\ln T \leq y) = P(T \leq e^y) \\ &= 1 - e^{-(e^y/\theta)^\alpha} = 1 - e^{-(e^{y\alpha}/e^{\ln\theta^\alpha})} = 1 - e^{-e^{(y\alpha - \alpha\ln\theta)}} \\ &= 1 - e^{-e^{\alpha(y-\ln\theta)}} = 1 - e^{-e^{((y-\mu)/\sigma)}}, \end{aligned}$$

where $\mu = \ln\theta$ is the location parameter and $\sigma = \frac{1}{\alpha}$ is the scale parameter of the extreme value distribution.

A generalized extreme value distribution, GEV(μ, σ, ξ) is standard extreme value distributed when the location parameter, μ, is equal to 0, the scale parameter, σ, is equal to 1 and the shape parameter, ξ, is equal to 0.

The cumulative distribution for the standard extreme value distribution is defined as

$$F(y) = 1 - e^{-e^y}. \tag{4.1}$$

The linear regression model for the log failure times $y_i = \ln t_i$ on the y data can be expressed as

$$\begin{aligned} y_i &= \ln t_i = \beta_0 + \beta_1 x_{i1} + ... + \beta_k x_{ik} + \sigma\epsilon_i \\ &= \mathbf{x}_i^T \boldsymbol{\beta} + \sigma\epsilon_i, \ i = 1, ..., k, \end{aligned} \tag{4.2}$$

where ϵ is standard extreme value distributed, type I, $\epsilon \sim$ GEV(0,1,0).

To generate values from the standard extreme value distribution we start by generating a value, v, from the uniform distribution on [0,1] and then use that the inverse of $F(y) = v$ is $F^{-1}(v) = y$. $y = F^{-1}(v)$ is standard extreme value distributed. Then we can simulate $\epsilon = y$ as follows

$$\begin{aligned} F(y) &= 1 - e^{-e^y} = v \\ \ln(1-v) &= -e^y \\ \epsilon &= y = \ln(-\ln(1-v)). \end{aligned}$$

4.3 Gross variance

The performances of the methods in this report are compared by examining the gross variances of the regression coefficients. The gross variance is the expected total mean square error. The mean square error of an estimator is a method for quantifying the difference between values implied by an estimator and the true values of the quantity being estimated. The expected mean square error is calculated by computing the square error of the estimated regression coefficients obtained by the methods and then subtracting the true regression coefficients of the model. To obtain a total variance of all n runs, the variances of all n runs are summed, thereafter divided by the total number of runs, and at last divided by the number of regression coefficients, s. In cases where the maximum likelihood estimator does not exist in some of the n runs, the number of acceptable regression coefficients is decreased to r, such that the total variance must be computed for all r runs. The value of r varies for all cases and and is included in the discussion of the results for example 1 in experiement 1, in section 6.1. The gross variance of n runs and s regression coefficients is calculated by

$$\begin{aligned} \text{GV} &= \frac{1}{s}\Big(\frac{1}{n}\sum_{i=1}^{n}(\widehat{\text{coeff}}_{1,i} - \text{coeff}_{1,i})^2 + \frac{1}{n}\sum_{i=1}^{n}(\widehat{\text{coeff}}_{2,i} - \text{coeff}_{2,i})^2 \\ &\quad + ... + \frac{1}{n}\sum_{i=1}^{n}(\widehat{\text{coeff}}_{s,i} - \text{coeff}_{s,i})^2\Big), \end{aligned} \tag{4.3}$$

where $\widehat{\text{coeff}}_{s,i}$ is the estimated regression coefficient of the true regression coefficient $\text{coeff}_{s,i}$. The purpose of computing the gross variance is to distinguish between the methods being investigated. The most accurate method should obtain the smallest gross variance, as a low value of the gross variance implies that the estimated regression coefficients are pretty similar to the actual regression coefficients.

4.4 The quick and dirty method

The concept of the quick and dirty method is to treat the censoring times as actual failure times. This is simply obtained by letting the censoring limit replace all failure times which lie outside the censoring limit. The quick and dirty method is easy to handle and implement, and it is very fast. It is important to keep in mind that the quick and dirty method ignores the censoring information, which may lead to inaccurate results. The quick and dirty method can give inaccurate results especially in cases where the actual failure times differ a lot from the censoring times. On the other hand, if the actual failure times are quite close to the censoring times, the quick and dirty method may perform very well.

The pseudo code for the implementation of the quick and dirty method is stated below.

- Define the expected response vector $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}$ containig k observations, and the vectors of the regression coefficients.

- For n runs:
  - Simulate an error vector, $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, ..., \epsilon_k]^T \sim GEV(0, 1, 0)$.
  - Set $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \sigma\boldsymbol{\epsilon} = [y_1, y_2, ..., y_k]^T$.
  - For right censoring at C_r, replace any $y_i > C_r$ by C_r for $i = 1, 2, ..., k$.
  - For left censoring at C_l, any $y_i < C_l$ is replaced by C_l for $i = 1, 2, ..., k$.
  - Use the lm function to estimate the regression coefficients of $\mathbf{y}$.
  - Save the estimated regression coefficients in a matrix for each iteration.
- Calculate the gross variance from equation (4.3).

4.5 Maximum likelihood estimation

Maximum likelihood estimation is a method of estimating unknown parameters of a statistical model, and these parameter estimates are obtained by maximizing the likelihood function of that model. Although the method has been used earlier by several scientists, it was R. A. Fisher who strongly recommended the use of the method in the time space 1912 - 1922. More details on maximum likelihood and the derivation of the maximum likelihood for Weibull distribution can be found in section 2.3.

The maximum likelihood method is a straightforward method which is easy to implement in R, as the *survreg* function can be used to estimate the regression coefficients of Weibull distributed data. A disadvantage is that the maximum likelihood estimates may not exist. This is the case for instance when the factor's high level are all right censored and all observations at its low level are uncensored.

The pseudo code for the maximum likelihood method is stated as

- Define the expected response vector $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}$ of k observations and the vectors of the regression coefficients.
- For n runs:
  - Simulate an error vector, $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, ..., \epsilon_k]^T \sim GEV(0, 1, 0)$.
  - Set $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \sigma\boldsymbol{\epsilon} = [y_1, y_2, ..., y_k]^T$.
  - State which observations are right censored, left censored or uncensored. Run *survreg* to obtain estimates of the regression coefficients of $\mathbf{y}$ by Newton's method.
  - Save the estimated regression coefficients in a matrix.
- Calculate the gross variance by equation (4.3).

4.5.1 Handling non convergence of the maximum likelihood estimation

The maximum likelihood estimator may not exist, and that can cause problems in the computations of the regression coefficients. When the maximum likelihood estimates do not exist, the estimated regression coefficients and the corresponding standard errors can take on rather obscure or peculiar values. To overcome these problems, a confidence interval is used to assess the validity of the standard errors. If all standard errors for the estimated regression coefficients lie within the confidence interval, the maximum likelihood estimator is assumed to exist and further computations can be accomplished. If one

standard error lies outside the confidence interval, we assume that the maximum likelihood estimator may not exist and the computations are terminated.

Choosing an α of 0.01, a 99 % confidence interval is to be estimated. 10 000 data sets are simulated, giving 10 000 observations of ϵ in 10 000 vectors of eight elements, and then the standard errors from each vector are computed. To obtain a 99 % confidence interval, the 50 biggest and 50 smallest values should be removed, and the new biggest and smallest values are used as limits. Dealing with three coefficients for instance, the lower limit and upper limit can be adjusted as follows:

We now consider a model consisting of three coefficients. Let S_1, S_2 and S_3 be the estimators for the standard deviations of the coefficients and let $V = \min(S_1, S_2, S_3)$. Then

$$\begin{aligned} P(V = \min(S_1, S_2, S_3) \leq C_{lower}) &= 1 - (1 - F_S(C_{lower}))^3 = 0.005, \\ (1 - F_S(C_{lower}))^3 &= 0.995 \\ F_S(C_{lower}) &= 1 - \sqrt[3]{0.995} = 0.0017 \end{aligned}$$

where V is a set of the independent standard errors S_1, S_2 and S_3 of the three coefficients, and C_{lower} is the lower limit of the confidence interval. The computation tells us that 0.17 % of the standard errors should lie below the lower limit of the confidence interval. This means that in the 10 000 estimated data sets, 17 of the estimated errors should lie below the lower limit ($(10000 \cdot 0.17\%)/100\% = 17$). Therefore, the 17 smallest standard errors of the 10 000 data sets are removed, and a new lower limit is obtained.

The probability of the maximum standard error being smaller than the upper limit of the confidence interval is the probability of all three standard errors being smaller than the upper limit;

$$\begin{aligned} P(V = \max(S_1, S_2, S_3) \leq C_{upper}) &= (F_S(C_{upper}))^3 = 0.995 \\ F_S(C_{upper}) &= \sqrt[3]{0.995} = 0.9983, \end{aligned}$$

where C_{upper} is the upper limit of the confidence interval.

The computation shows that 99.83 % of the standard errors should lie below the upper limit, which means 9983 of the 10 000 estimated standard errors. The 17 greatest standard errors are removed, and the greatest of the remaining standard errors is the new upper limit of the confidence interval.

Both of these limits need to be scaled, because the standard errors estimated by the simulations are the standard errors of ϵ, but we need the standard errors of the regression coefficients. The relation between the standard error of the regression coefficients and the standard error of ϵ is

$$\mathrm{SD}_{\mathrm{coeff}} = \frac{1}{\alpha\sqrt{8}}\mathrm{SD}_{\epsilon},$$

where α is the shape parameter of the Weibull distribution.

The confidence interval needs to be computed for each value of α and σ, as three different values of σ will be considered in the experiments (further explanation in section 6.1.1). For the same values of σ, the same confidence interval is applied. The confidence interval is also included when performing single imputation or multiple imputation initialized by the maximum likelihood estimator.

The necessary code in R is

```
alpha <- 5  # alpha is 5, 3.33 or 2.5

yvec <- vector()
sdvec <- vector()

for (i in 1:10000){
  for (j in 1:8){
    yvec[j] <- rgev(1,0,0,1)
  }
  sdvec[i] <- sd(yvec)
}

nsmallest <- order(sdvec)[1:n]
nlargest <- order(sdvec, decreasing = T)[1:n]

nlower <- sdvec[nsmallest[21]]
nupper <- sdvec[nlargest[21]]

sdlower <- nlower/(sqrt(8)*alpha)
sdupper <- nupper/(sqrt(8)*alpha)
```

The confidence interval obtained for σ was developed with the assumption of no censored data. When dealing with data sets that contain censored data, the uncertainty of the statistics of the data sets will be higher than if the data sets did not contain any censored data. Therefore, we can assume the confidence interval to be a little too strict and can conveniently extend the boundaries to some point. The upper limit is increased by 65 %, while the lower limit is decreased by 35 %. The interval limits for σ are then given as

$$\begin{aligned} &(0.015672, 0.33934) && \text{for original } \sigma = 0.2, \\ &(0.023532, 0.50952) && \text{for original } \sigma = 0.3, \\ &(0.031344, 0.67868) && \text{for original } \sigma = 0.4. \end{aligned}$$

4.6 Single Imputation

Single imputation is a method where one substitutes one value for each missing value, thus obtaining a complete data set, which can be analysed by standard complete data methods of analysis. Each missing value can be imputed from the variable mean of the complete cases or from the mean conditional (conditional expectation) on observed values of other variables, which is the procedure used in this report.

A distinct advantage of single imputation is, as mentioned, that once the values have been filled in, standard complete data methods of analysis can be used on the data set. Another advantage is that the imputations can incorporate the data collector's knowledge. On the other hand, a disadvantage is that simple imputation does not reflect the uncertainty about the predictions of the unknown missing values. Hence, the standard errors of the estimates tend to be too low compared to the standard error of the non-missing values.

Hamada and Wu [2] describe the method of single imputation.

First, we assume a Weibull distributed T with probability density $f(t)$ and cumulative distribution function $F(t)$. As introduced in section 4.2, the transformation $Y = \ln T$ has the extreme value distribution whose probability density, $f(y)$, is given in equation (2.3) and the cumulative distribution, $F(y)$, is given in equation (2.4). Then, the linear regression model can be expressed as in (4.2);

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \sigma\boldsymbol{\epsilon}, \tag{4.4}$$

where $\mathbf{y}$ is the response vector, $\mathbf{X}$ is a matrix of the explanatory variables from the experimental design, $\boldsymbol{\beta}$ is a vector of coefficients, σ is the standard deviation and $\boldsymbol{\epsilon}$ is the error, which is standard extreme value distributed, $\epsilon \sim \text{GEV}(0, 1, 0)$. Then, we must compute the conditional expectation in the case of right censoring and left censoring. The right censoring time is presented as C_r, while the left censoring time is expressed as C_l.

For right censored data at C_r, the truncated generalized extreme value distribution function is

$$\begin{aligned} P(Y \leq y | Y > C_r) &= \frac{P(Y \leq y \cap Y > C_r)}{P(Y > C_r)} = \frac{P(Y \leq y \cap 1 - (Y \leq C_r))}{1 - P(Y < C_r)} \\ &= \frac{P(Y \leq y)}{1 - P(Y \leq C_r)} = \frac{F(y)}{1 - F(C_r)}, \end{aligned}$$

and the truncated density function is defined as

$$f(y|Y > C_r) = \frac{F'(y)}{1 - F(C_r)} = \frac{f(y)}{1 - F(C_r)}.$$

The conditional expectation is

$$E(y|Y > C_r) = \int_{C_r}^{\infty} y \frac{f(y)}{1 - F(C_r)} dy = \frac{1}{1 - F(C_r)} \int_{C_r}^{\infty} y \frac{1}{\sigma} e^{\frac{y-\mu}{\sigma}} e^{-e^{\frac{y-\mu}{\sigma}}} dy. \tag{4.5}$$

The integral in (4.5) can be rewritten using the transformation $z = \frac{y-\mu}{\sigma}$ and $\frac{dz}{dy} = \frac{1}{\sigma}$ as

$$\int_{\frac{C_r-\mu}{\sigma}}^{\infty} (z\sigma + \mu) e^z e^{-e^z} dz,$$

which again can be modified by the transformation $u = e^z$ and $\frac{du}{dz} = e^z = u$;

$$\begin{aligned} \int_{e^{\frac{C_r-\mu}{\sigma}}}^{\infty} (\sigma \ln(u) + \mu) e^{-u} du &= \int_{e^{\frac{C_r-\mu}{\sigma}}}^{\infty} \sigma \ln(u) e^{-u} du + \int_{e^{\frac{C_r-\mu}{\sigma}}}^{\infty} \mu e^{-u} du \\ &= \Big[-\sigma \ln(u) e^{-u} \Big]_{e^{\frac{C_r-\mu}{\sigma}}}^{\infty} + \sigma \int_{e^{\frac{C_r-\mu}{\sigma}}}^{\infty} \frac{1}{u} e^{-u} du \\ &\quad + \int_{e^{\frac{C_r-\mu}{\sigma}}}^{\infty} \mu e^{-u} du \\ &= \sigma \left(\frac{C_r - \mu}{\sigma} e^{-e^{\frac{C_r-\mu}{\sigma}}} + \text{E}_1\left(e^{\frac{C_r-\mu}{\sigma}} \right) \right) \\ &\quad + \mu e^{-e^{\frac{C_r-\mu}{\sigma}}}, \end{aligned}$$

where $\mathrm{E}_1\big(e^{\frac{C_r-\mu}{\sigma}}\big)$ is the exponential integral. [1]
The expression of the expected value, $E(y|Y > C_r)$ in (4.5) becomes

$$E(y|Y > C_r) = \frac{1}{1-F(C_r)}\left(\sigma\left(\frac{C_r-\mu}{\sigma}e^{-e^{\frac{C_r-\mu}{\sigma}}} + \mathrm{E}_1\left(e^{\frac{C_r-\mu}{\sigma}}\right)\right) + \mu e^{-e^{\frac{C_r-\mu}{\sigma}}}\right). \quad (4.6)$$

For left censored data at C_l, the truncated generalized extreme value distribution function is

$$P(Y \le y|Y \le C_l) = \frac{P(Y \le y \cap Y \le C_l)}{P(Y \le C_l)} = \frac{P(Y \le y)}{P(Y \le C_l)} = \frac{F(y)}{F(C_l)}$$

and the truncated density function is defined as

$$f(y|Y > C_r) = \frac{F'(y)}{F(C_l)} = \frac{f(y)}{F(C_l)}.$$

The conditional expected value is defined as

$$\begin{aligned} E(y|Y \le C_l) &= \int_{-\infty}^{C_l} y\frac{f(y)}{F(C_l)}dy = \frac{1}{F(C_l)}\left[\int_{-\infty}^{\infty} yf(y)dy - \int_{C_l}^{\infty} yf(y)dy\right] \\ &= \frac{1}{F(C_l)}\left[\int_{-\infty}^{\infty} y\frac{1}{\sigma}e^{\frac{y-\mu}{\sigma}}e^{-e^{\frac{y-\mu}{\sigma}}}dy - \int_{C_l}^{\infty} y\frac{1}{\sigma}e^{\frac{y-\mu}{\sigma}}e^{-e^{\frac{y-\mu}{\sigma}}}dy\right]. \quad (4.7) \end{aligned}$$

Using that the first interval equals the expected value of the Gumbel distribution as defined in (2.5), and using the same transformations as for right censored data, $z = \frac{y-\mu}{\sigma}$ and $\frac{dz}{dy} = \frac{1}{\sigma}$ for the second integral, the two integrals transform to

$$(\mu - \gamma\sigma) - \int_{\frac{C_l-\mu}{\sigma}}^{\infty}(z\sigma+\mu)e^z e^{-e^z}dz.$$

Another transformation is performed, where $u = e^z$ and $\frac{du}{dz} = e^z = u$ and the expression above becomes

$$\begin{aligned} & (\mu-\gamma\sigma) - \int_{e^{\frac{C_l-\mu}{\sigma}}}^{\infty}(\sigma\ln(u)+\mu)e^{-u}du \\ = \;& (\mu-\gamma\sigma) + \left[\sigma\ln(u)e^{-u}\right]_{e^{\frac{C_l-\mu}{\sigma}}}^{\infty} - \int_{e^{\frac{C_l-\mu}{\sigma}}}^{\infty}\sigma\frac{1}{u}e^{-u}du - \left[\mu e^{-u}\right]_{e^{\frac{C_l-\mu}{\sigma}}}^{\infty} \\ = \;& (\mu-\gamma\sigma) - \sigma\mathrm{E}_1\left(e^{\frac{C_l-\mu}{\sigma}}\right) - \mu e^{-e^{\frac{C_l-\mu}{\sigma}}}. \end{aligned}$$

The expected value, $E(y|Y \le C_l)$ in (4.7) transforms to

$$E(y|Y \le C_l) = \frac{1}{F(C_l)}\left(\mu - \gamma\sigma - \sigma\mathrm{E}_1\left(e^{\frac{C_l-\mu}{\sigma}}\right) - \mu e^{-e^{\frac{C_l-\mu}{\sigma}}}\right). \quad (4.8)$$

[1] The exponential integral is defined as

$$\mathrm{Ei}(x) = -\int_{-x}^{\infty}\frac{e^{-t}}{t}dt.$$

Since $E_1(x) = -\mathrm{Ei}(-x)$, $E_1(x)$ can be written as

$$\mathrm{E}_1(x) = \int_x^{\infty}\frac{e^{-u}}{u}du.$$

Initializing the censored times

When executing a single imputation test, the censored data must be set to some values, to obtain a complete data set. Then, the mean and the variance of the response vector are computed, because the mean and the variance will be used in the computation of the conditional expected value of the originally censored data. The values to be inserted for the censored data could be obtained by among others the quick and dirty method or maximum likelihood estimation. As mentioned earlier, the quick and dirty method manages censored data by setting the censored data equal to the censoring time and then estimating the regression coefficients by linear regression, while the maximum likelihood estimator applies Newton's method to estimate the regression coefficients. The psuedo codes for both approaches are displayed below.

Single imputation using the quick and dirty approach

- Define the expected response vector $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}$ containig k observations, and the vectors of the regression coefficients.
- For n runs:
  - Simulate an error vector, $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, ..., \epsilon_k]^T \sim GEV(0, 1, 0)$.
  - Set $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \sigma\boldsymbol{\epsilon} = [y_1, y_2, ..., y_k]^T$.
  - For right censoring at C_r, replace any $y_i > C_r$ by C_r for $i = 1, 2, ..., k$.
  - For left censoring at C_l, any $y_i < C_l$ is replaced by C_l for $i = 1, 2, ..., k$.
  - Use the *lm* function to estimate the regression coefficients of $\mathbf{y}$.
  - Estimate the mean μ_i for each value of y_i for $i = 1, 2, ..., k$ and the variance σ^2 (and hence the standard error σ).
  - If $y_i > C_r$, set y_i equal to equation (4.6) for $i = 1, 2, ..., k$.
  - If $y_i < C_l$, set y_i equal to equation (4.8) for $i = 1, 2, ..., k$.
  - Apply the *lm* function to estimate the regression coefficients of $\mathbf{y}$.
  - Save the estimated regression coefficients in a matrix.
- Compute the gross variance from equation (4.3).

Single imputation using maximum likelihood estimation

- Define the expected response vector $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}$ of k observations and the vectors of the regression coefficients.
- For n runs:
  - Simulate an error vector, $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, ..., \epsilon_k]^T \sim GEV(0, 1, 0)$.
  - Set $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \sigma\boldsymbol{\epsilon} = [y_1, y_2, ..., y_k]^T$.
  - State which observations are right censored, left censored or uncensored. Run *survreg* to obtain estimates of the regression coefficients of $\mathbf{y}$ by Newton's method.
  - Estimate the mean μ_i for each value of y_i for $i = 1, 2, ..., k$ and the variance σ^2.
  - If $y_i > C_r$, set y_i equal to equation (4.6) for $i = 1, 2, ..., k$.

- If $y_i < C_l$, set y_i equal to equation (4.8) for $i = 1, 2, ..., k$.
- Run the lm function to estimate the regression coefficients of $\mathbf{y}$.
- Save the estimated regression coefficients in a matrix.

- Delete estimated regression coefficients of which the maximum likelihood estimator did not exist.
- Compute the gross variance by equation (4.3).

To avoid trouble with computations of which the maximum likelihood estimator does not exist, a confidence interval as presented in 4.5.1 is applied on the standard errors of the regression coefficients. When the standard errors lie within the interval, the maximum likelihood estimator is assumed to exist and the computation of the regression coefficients is completed. If one or more of the standard errors do not lie within the confidence interval, the maximum likelihood estimator is assumed not to exist, causing unacceptable values for the standard errors and the estimated regression coefficients. In that case, the computation of the regression coefficients is not performed.

4.7 Multiple Imputation

Multiple imputation is a Monte Carlo technique, where the idea is to impute several values, m, for each missing value. The method was first introduced by Rubin in the 1970s and later elaborated in his book published in 1987 [6], and is also discussed in Scafers [7]. The m imputations are ordered such that m complete data sets can be created from the vectors of imputation. Replacing each missing value by the first component in its vector of imputations creates the first completed data set and so on. Each complete data set is then analysed by standard procedures, and the results are later combined to produce estimates and confidence intervals that incorporate missing data uncertainty. The m imputed values are drawn from a truncated distribution, as explanied in section 3.3.

The described procedure suggests that m imputations create m data sets, where each data set is analysed and at the end each parameter of the m data sets is averaged. Since the models considered in this report are linear models, we can change the order of the linear operations, such that the averaging is performed before the analysing. The suggested procedure is therefore; m imputations create m data sets, where the parameters of m values are averaged, and then the data set of averaged values is analysed. This swap saves us for some work and time, as only one data set needs to be analysed, instead of m. It is very important to note that the swapping is only possible when dealing with linear models.

As for simple imputation, two advantages of multiple imputation are the ability to use complete case methods of analysis and the ability to incorporate the data collector's knowledge. A disadvantage of multiple imputation is that the method requires more work in both creating the imputations and analysing the results, although analysing the results only demands performing the same task m times instead of once. In this report, m is chosen to be 5.

Initializing the censored times

As for single imputation, the censored data in multiple imputation must be set to some values, to obtain a complete data set. The mean and the variance of this complete data set are then computed, because the mean and the variance will be used in the computation of the imputation values. The quick and dirty method and maximum likelihod estimation

are used to compute the values to be inserted for the censored data. The pseudo codes for both approaches can be found below.

Multiple imputation using the quick and dirty method

- Define the expected response vector $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}$ consisting of k observations and the vectors of the regression coefficients.
- For n runs:
  - Simulate an error vector, $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, ..., \epsilon_k]^T \sim GEV(0, 1, 0)$.
  - Set $\mathbf{y} = \mathbf{X}^T\boldsymbol{\beta} + \sigma\boldsymbol{\epsilon} = [y_1, y_1, ..., y_k]^T$.
  - For right censoring at C_r, replace any $y_i > C_r$ by C_r for $i = 1, 2, .., k$.
  - For left censoring at C_l, any $y_i < C_l$ is replaced by C_l for $i = 1, 2, ..., k$.
  - Use the *lm* function to estimate the regression coefficients of $\mathbf{y}$.
  - Estimate the mean μ_i for each value of y_i for $i = 1, 2, ..., k$ and the variance σ^2 (and hence the standard error σ).
  - If $\mathbf{y}$ contains right censored data:
    1. Run the truncated pseudo code from section 3.3 to simulate m values for each of the censored values of $\mathbf{y}$.
    2. Replace each censored value by the mean of the m imputed values.
  - If $\mathbf{y}$ contains left censored observations:
    1. Run the truncated pseudo code from section 3.3 to simulate m values for each of the censored values of $\mathbf{y}$.
    2. Replace each censored value by the mean of the m imputed values.
  - Use the *lm* function to estimate the regression coefficients of $\mathbf{y}$.
  - Save the estimated regression coefficients in a matrix.
- Compute the gross variance by equation (4.3).

Multiple imputation using maximum likelihood estimation

- Define the expected response vector $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}$ consisting of k observations and vectors of the regression coefficients.
- For n runs:
  - Simulate an error vector, $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, ..., \epsilon_k]^T \sim GEV(0, 1, 0)$.
  - Set $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \sigma\boldsymbol{\epsilon} = [y_1, y_1, ..., y_k]^T$.
  - State which observations are right censored, left censored or uncensored. Run *survreg* to obtain estimates of the regression coefficients of $\mathbf{y}$ by Newton's method.
  - Estimate the mean μ_i for each value of y_i for $i = 1, 2, ..., k$ and the variance σ^2.
  - If $\mathbf{y}$ contains right censored data:
    1. Run the truncated pseudo code from section 3.3 to simulate m values for each of the censored values of $\mathbf{y}$.
    2. Replace each censored value by the mean of the m imputed values.

- If $\mathbf{y}$ contains left censored observations:
  1. Run the truncated pseudo code from section 3.3 to simulate m values for each of the censored values of $\mathbf{y}$.
  2. Replace each censored value by the mean of the m imputed values.
- Run lm to obtain estimates of the regression coefficients of $\mathbf{y}$.
- Save the estimated regression coefficients in a matrix.

- Compute the gross variance by equation (4.3).

As for the other methods using the maximum likehood estimator, the existence of the maximum likelihood estimator is assessed through the validity of the standard errors of the estimated regression coefficients. As explained in section 4.5.1, computations where all the estimated standard errors of the regression coefficients lie within the confidence interval, are the ones being considered. Computations for which one or more of the standard errors lie outside the confidence interval are not completed and are considered as not valid.

Chapter 5

Examples and experiments

The examples and experiments in this report are the exactly same as the ones in Sue-Chu [8], with the exception of example 2. The intention of using the exact same examples was to obtain an opportunity of comparing the performances of the methods for both Weibull distributed data and normally distributed data, as Sue-Chu examines normally distributed data. Example 2 in Sue-Chu [8] was edited for usage in this report because the original model experiences problems when two values are right censored, due to linear dependency among the coefficient columns. The interaction AB is linear dependent of the factors k, A and B, as AB = - (k + A + B). To avoid these problems, the interaction AB is excluded from the model.

In the end, it turns out that the methods for Weibull distributed data and normal distributed data are implemented differently, such that a comparison is not possible after all.

5.1 Example 1

The first example consists of a simple model. The response, y, is produced by a 2^3 full factorial experiment as shown in table 5.1 . The factors A and C, and the interaction AB have true values equal to 1. The error term, ϵ, is set to be standard extreme value distributed with location 0, scale 1 and shape 0. Thus, the linear regression model of example 1 is defined as

$$y = A + C + AB + \sigma\epsilon, \qquad \epsilon \sim GEV(0, 1, 0).$$

Table 5.1: Example 1, a 2^3 full factorial experiment.

Run no.	A	C	AB	E(y)
1	-1	-1	1	-1
2	1	-1	-1	-1
3	-1	-1	-1	-3
4	1	-1	1	1
5	-1	1	1	1
6	1	1	-1	1
7	-1	1	-1	-1
8	1	1	1	3

A simple example is chosen such that the analysis should be easy to conduct. With real coefficients equal to 1, the computation of the gross variance should be easy and the

performances of the different methods can therefore easily be compared. The maximum likelihood estimator may not exist due to right censored observations being at a high level and no censoring are at a low level.

The regression coefficients of the factors A and C and the interaction AB are estimated n times as respectively $\hat{a}$, $\hat{c}$ and $\hat{ab}$. n is equal to 100. In experiments where the maximum likelihood estimator sometimes does not exist, the number of valid estimates of $\hat{a}$, $\hat{c}$ and $\hat{ab}$ decrease to r, as mentioned in section 4.3. The value of r varies from experiment to experiment. The gross variance, as defined in equation (4.3), for example 1 is given as

$$\mathrm{GV}_{ex1} = \frac{1}{3}\Big(\frac{1}{n}\sum_{i=1}^{n}(\hat{a}_i - 1)^2 + \frac{1}{n}\sum_{i=1}^{n}(\hat{c}_i - 1)^2 + \frac{1}{n}\sum_{i=1}^{n}(\hat{ab}_i - 1)^2\Big).$$

5.2 Example 2

Example 2 is also a 2^3 factorial experiment, but a more complex one than example 1, and it consists of a less symmetric response vector, y. The factors and interaction A, B and AC have respectively 2, 3, -0.5 as true values. In addition, a constant with value 10 is added to the response. The experiment is presented in table 5.2. Again, the error term, ϵ is set to be standard extreme value distributed with location 0, scale 1 and shape 0. The linear regression model of example 2 is expressed as

$$y = 10 + 2A + 3B - 0.5AC + \sigma\epsilon, \qquad \epsilon \sim GEV(0, 1, 0).$$

Table 5.2: Example 2, a 2^3 full factorial experiment.

Run no.	A	B	AC	E(y)
1	-1	-1	1	4.5
2	1	-1	-1	9.5
3	-1	1	1	10.5
4	1	1	-1	15.5
5	-1	-1	-1	5.5
6	1	-1	1	8.5
7	-1	1	-1	11.5
8	1	1	1	14.5

The gross variance is computed as

$$\mathrm{GV}_{ex2} = \frac{1}{3}\Big(\frac{1}{n}\sum_{i=1}^{n}(\hat{a}_i - 2)^2 + \frac{1}{n}\sum_{i=1}^{n}(\hat{b}_i - 3)^2 + \frac{1}{n}\sum_{i=1}^{n}(\hat{ac}_i + 0.5)^2\Big),$$

where $\hat{a}$, $\hat{b}$ and $\hat{ac}$ are the estimates for the regression coefficients A and B, and the interaction AC. As for example 1, the maximum likelihood estimator may not exist due to right censored observations being at a high level and no censoring are at a low level.

5.3 Experiment 1

In experiment 1, the computation of the error vector is performed for each error term in the error vector and also in every of the n runs, such that every error vector is unique for a specific run. Thus, the computations of the n response values are based on different error vectors. As the error vector differs from run to run within a method, it will also differ

from method to method. None of all the single runs would base their estimation on the same errors. The experiment will test the four methods on their ability to give accurate estimates of regression coefficients when dealing with censored values. The changing error will test if the methods are able to perform well for different values.

5.4 Experiment 2

The purpose of experiment 2 is to compare the performances of the methods more directly, by running them with the same error vector. The error terms are still varying within each error vector. The experiment is implemented such that each error vector is estimated first and then applied on all four methods. This way, the error vector applied in the 34th run of the quick and dirty method is exactly the same error vector as the one used in the 34th run of simple imputation. Experiment 2 will test the method's ability to estimate the regression coefficients when censoring is present.

By comparing the results of the computations performed with experiment 1 and experiment 2, one can discuss if the equalness of the error vector is necessary in order to distinguish the performances of the methods.

Chapter 6

Results of experiment 1

6.1 Example 1

The results of the methods for example 1 in experiment 1 is presented in this chapter. The methods are run for nine different combinations of censoring choices, and for three different values of σ. The choices of the censoring limits and σ are explained in the next section.

6.1.1 Choice of C_r, C_l and σ

The censoring in this project is of Type I censoring, where the censoring times are fixed and the number of censored units is random. To test how well the methods perform when censoring is present, the methods are tested for three possible scenarios; right censoring, left censoring and both right and left censoring. The censoring limits are particularly chosen with regard to element 3 and 8 of $\mathbf{y}$ in table 5.1. Whether a value of $\mathbf{y}$ is censored or not, depends on the original value, the censoring limit and the estimated error, defined in 4.2.

Two limits close to the original values were chosen, $C_r = 2.9$ and $C_r = 3.1$ as right censored limits, and $C_l = -2.9$ and $C_l = -3.1$ as left censored limits. In these cases it is very likely that only element 3 and element 8 become censored, but not necessarily. Element 3 and 8 may not be censored at all, and one or more of the other elements could be censored, even though the possibility of that happening is very small. Anyhow, the total number of censored items would be random. In addition, two limits further away, $C_r = 2.0$ and $C_l = -2.0$ are considered. As for the limits $C_r = 2.0$ and $C_l = -2.0$, the chance of the elements 1,2,4,5,6 or 7 being censored is higher than for the other limits. A total of nine experiments are conducted; experiments where data is right censored at 2.0, 2.9 or 3.1, left censored at -2.0, -2.9 or -3.1, and experiments where the data is both left and right censored at 2.0 and -2.0, 2.9 and -2.9, and 3.1 and -3.1. The experiments containing both left and right censored data are chosen with respect to the closeness to the original values, such that the two limits furthest away from the original values are combined and so on.

In addition to the censoring limits, the error term, $\sigma\epsilon$ needs to be defined. The simulation of ϵ is described in 4.2, while the value of σ can be fixed by the investigator, by choosing an appropriate value of α, the shape parameter of a Weibull distribution, since $\sigma = \frac{1}{\alpha}$. Three different values of α were chosen; 5, 3.33 and 2.5, such that the values of σ will be 0.2, 0.3 and 0.4.

6.1.2 Testing for $C_r = 2.0$ and $C_l = -\infty$

The results of the experiment of which the data in model 1 is right censored at $C_r = 2.0$ are displayed in table 6.1.

The last unit of $\mathbf{y}$, unit 8 (with value equal to 3), is expected to be right censored in this experiment, although it might not always be the case. In addition, there is also a chance of element 4,5 and 6 (all values equal to 1) being censored. Since the censoring limit is a little far from the original value, the quick and dirty method is expected to perform poorer than the other methods. None of the methods stands out remarkably in neither a positive or a negative way. However, when comparing the gross variances in table 6.1, the quick and dirty method is pretty much giving the poorest result with gross variances of 0.018903, 0.027292 and 0.040402 for $\sigma = 0.2, 0.3$ and 0.4 respectively, even though single imputation using the maximum likelihood estimator gives the highest gross variance for $\sigma = 0.4$, of the value 0.041581. The maximum likelihood estimator gives the best overall result, with the lowest gross variances for $\sigma = 0.3$ and 0.4 as respectively 0.019079 and 0.033097. The lowest gross variance for $\sigma = 0.2$, is obtained as 0.0084327 by the multiple imputation initialized by the maximum likelihood estimator. It is interesting to notice that for $\sigma = 0.2$, both single imputation and multiple imputation initialized by the maximum likehood estimator performs a little better than maximum likelihood estimation itself.

All in all, the results of the four methods are pretty similar, and none of the methods deviates much in either a good or a bad way. The experiment is run 100 times, but it must be mentioned that the maximum likelihood estimator did not exist in every run. For maximum likelihood estimation and both the single imputation and the multiple imputation initialized by the maximum likelihood estimator, the number of runs of which the maximum likelihood estimator existed varies from 93 runs to 100 runs. This means that the number of results to base the analysis on, is a little smaller for methods using the maximum likelihood estimator than for the methods applying the quick and dirty approach.

Graphic presentation of regression coefficients

The gross variance computations are based on the estimated regression coefficients, such that estimated coefficients which lie close to the original value give a low gross variance. To display the estimated regression coefficients obtained by the four methods, a graphic presentation is chosen for one of the experiments. The estimated regression coefficients, $\hat{a}, \hat{b}$ and $\hat{ab}$, of the experiment of which $\sigma = 0.3$ and data is right censored at $C_r = 2.0$, are plotted. The coefficients estimated by the quick and dirty method are displayed in figure 6.1 and the coefficients estimated by maximum likelihood estimation are showed in figure 6.2. Figure 6.3 and figure 6.4 display the estimated regression coefficients obtained by single imputation respectively initialized by the quick and dirty method and the maximum likelihood estimator. At last, the estimated regression coefficients estimated by multiple imputation are displayed in figure 6.5 and figure 6.6 for respectively the quick and dirty approach and the maximum likelihood estimator as initializing method.

The regression coefficients estimated by methods using the quick and dirty approach, displayed in figure 6.1, 6.3 and 6.5, are mostly lower than the original values, which is natural considering that the quick and dirty approach assigns the censoring limit to the censored observations, and for right censoring the censoring limit is lower than the actual values of the censored observations. The regression coefficients estimated by methods using the maximum likelihood estimator, which are showed in figure 6.2, 6.4 and 6.6, are on the other hand more centered around the actual values, but the estimated values are slightly more higher than lower than the actual values.

Table 6.1: Results for $C_r = 2.0$ and $C_l = -\infty$.

σ	QD	MLE	SI using QD	SI using MLE	MI using QD	MI using MLE
	0.017986	0.012186	0.010109	0.0087354	0.012604	0.0092547
0.2	0.019398	0.0082323	0.010790	0.0096961	0.010632	0.0077740
	0.019324	0.010041	0.011689	0.0089031	0.013812	0.0082694
Average	0.018903	0.010153	0.010863	0.0091115	0.012349	0.0084327
	0.027898	0.017867	0.017227	0.024444	0.021109	0.020492
0.3	0.026220	0.018205	0.020401	0.022729	0.019648	0.021884
	0.027757	0.021164	0.020127	0.017453	0.021482	0.021980
Average	0.027292	0.019079	0.019252	0.021542	0.020746	0.021452
	0.038818	0.035997	0.039829	0.041255	0.037019	0.033332
0.4	0.041039	0.030438	0.035930	0.040476	0.030710	0.039402
	0.041349	0.032856	0.032181	0.043013	0.031241	0.041299
Average	0.040402	0.033097	0.035980	0.041581	0.032990	0.038011

6.1.3 Testing for $C_r = 2.9$ and $C_l = -\infty$

The results of right censoring at $C_r = 2.9$ for model 1 can be seen in table 6.2.

Again, item 8 of $\mathbf{y}$ is expected to be censored, but not as often as in the previous experiment. It is possible for the items 4,5 and 6 to be censored, but that is a rare occurrence. All methods should in principle be able to perform as good as the others, and basically, the performances are in fact reasonably similar. Looking at the gross variances in table 6.2, it is hard to identify a definite best or worst method. For $\sigma = 0.2$, maximum likelihood estimation gives the best estimated regression coefficients, with a gross variance equal to 0.0081771, while for $\sigma = 0.3$ and $\sigma = 0.4$, the quick and dirty method gives the best results with gross variances respectively equal to 0.018395 and 0.029122. The greatest gross variance for $\sigma = 0.2$ is given by multiple imputation using the quick and dirty approach, for $\sigma = 0.3$ and 0.4, single imputation using the maximum likelihood estimator gives the poorest results.

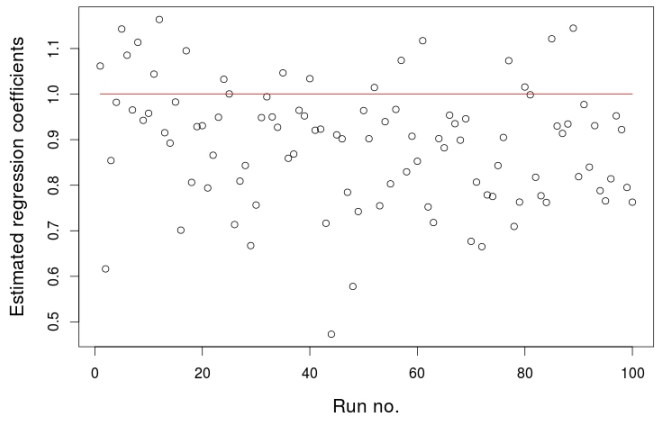
The differences between the gross variances for all methods are very small, it is almost kind of random which method gives the best result. All in all, all four methods perform well for right censoring at 2.9.

The possibility of a non existing maximum likelihood estimator is again present, and in this example the number of acceptable runs varies from 93 to 100 for methods applying the maximum likelihood estimator.

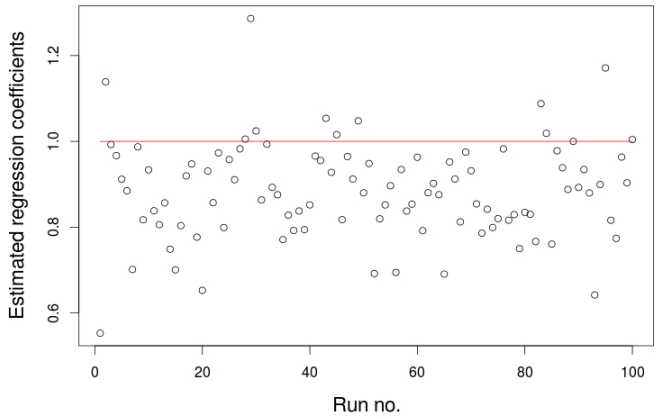
6.1.4 Testing for $C_r = 3.1$ and $C_l = -\infty$

For right censoring at $C_r = 3.1$, the results of the four methods are displayed in table 6.3.

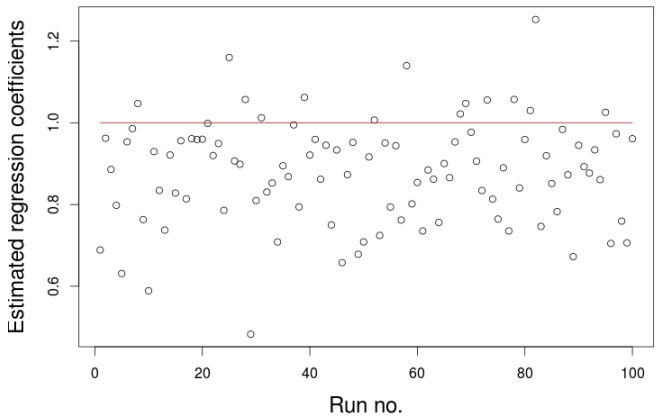
The possibility of censored item is again highest for the last item of $\mathbf{y}$. The chance of the other items being censored is almost negligible small, but we shall not exclude the possibility completely. Again, none of the methods presented itself as a obvious best or worst method. For $\sigma = 0.2$, the best gross variance is given by the method of maximum likelihood, while the worst is given by the quick and dirty method. For $\sigma = 0.3$, multiple imputation initialized by the maximum likelihood estimator gives the most accurate result and multiple imputation using the quick and dirty approach gives the least accurate result. The quick and dirty method gives the best result when $\sigma = 0.4$, and multiple imputation initialized by the quick and dirty approach gives the poorest result.



(a) Estimated regression coefficients of A.

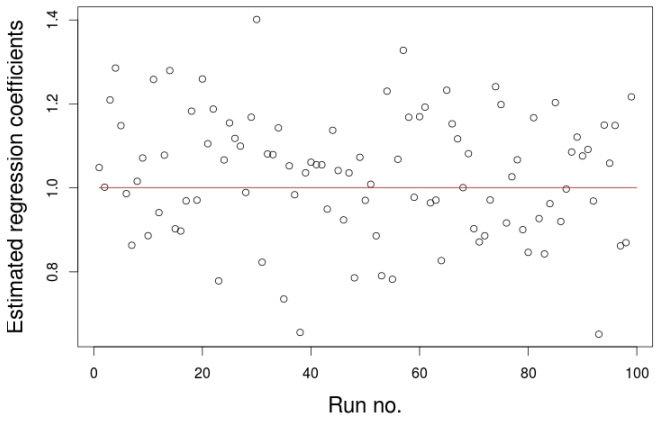


(b) Estimated regression coefficients of C.

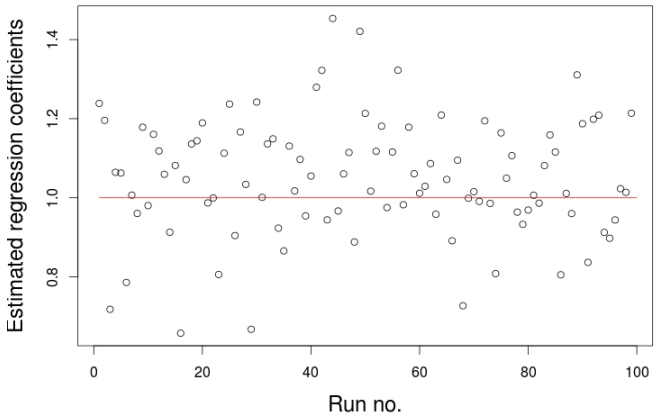


(c) Estimated regression coefficients of AB.

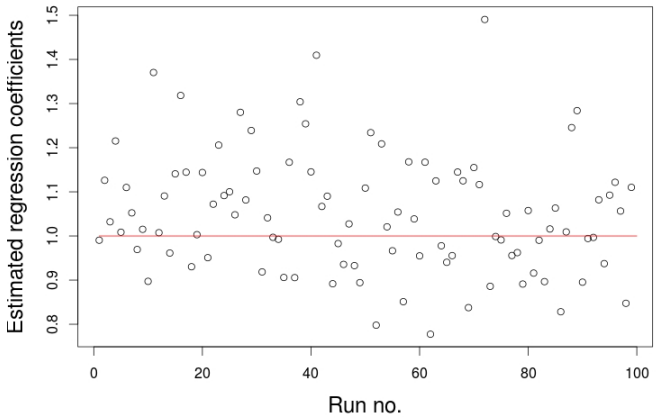
Figure 6.1: The quick and dirty method for $C_r = 2.0$ and $C_l = -\infty$ for $\sigma = 0.3$.



(a) Estimated regression coefficients of A.

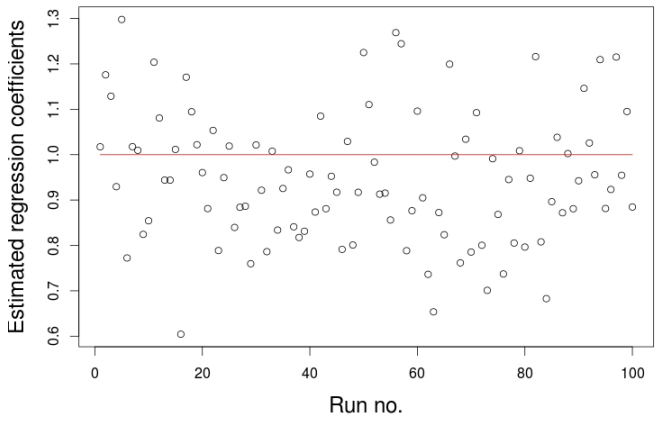


(b) Estimated regression coefficients of C.

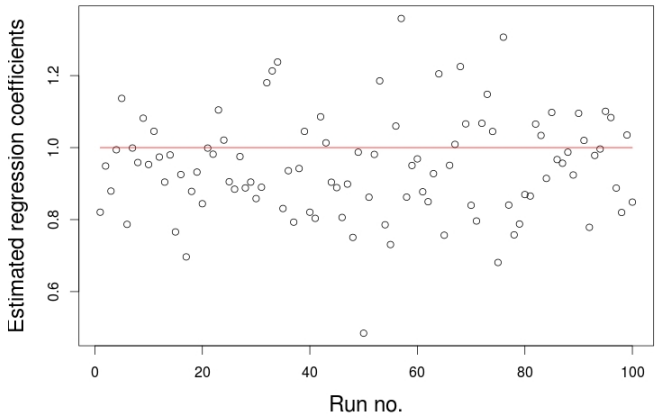


(c) Estimated regression coefficients of AB.

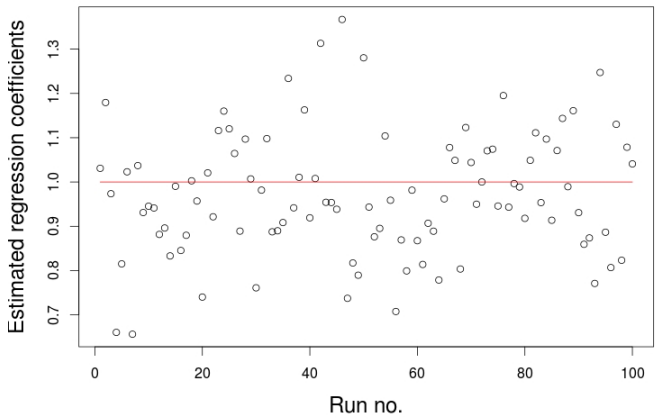
Figure 6.2: Maximum likelihood estimation for $C_r = 2.0$ and $C_l = -\infty$ for $\sigma = 0.3$.



(a) Estimated regression coefficients of A.

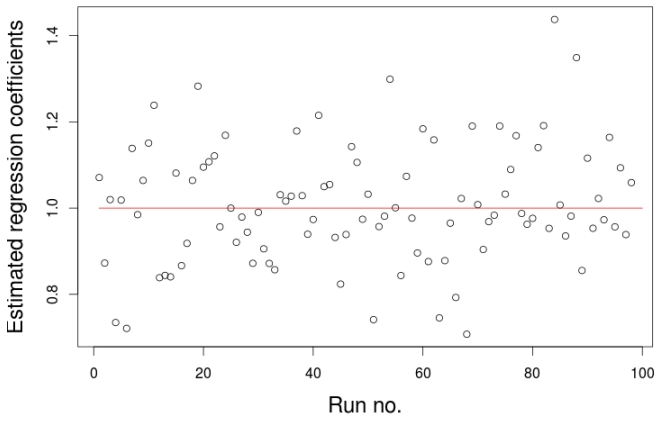


(b) Estimated regression coefficients of C.

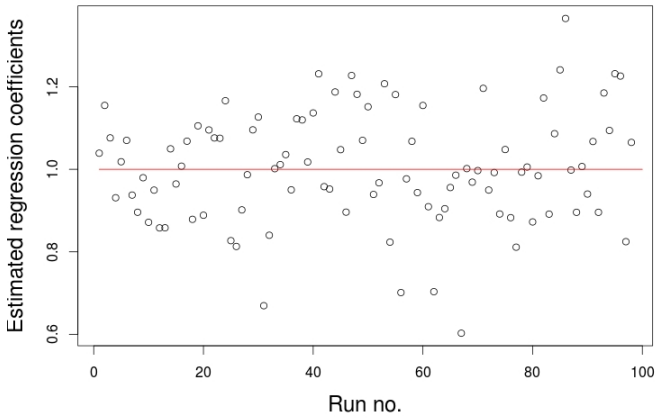


(c) Estimated regression coefficients of AB.

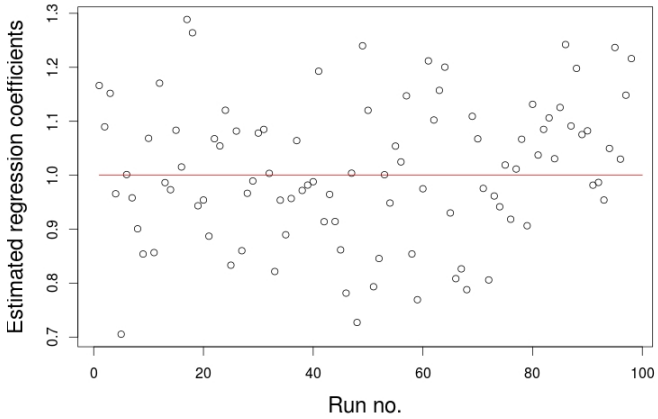
Figure 6.3: Single imputation using QD for $C_r = 2.0$ and $C_l = -\infty$ for $\sigma = 0.3$.



(a) Estimated regression coefficients of A.

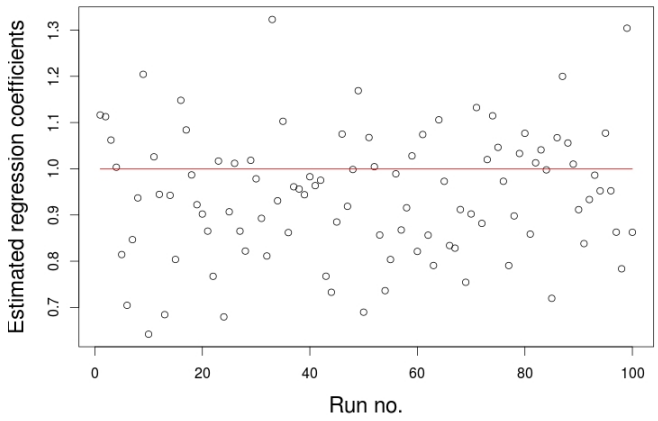


(b) Estimated regression coefficients of C.

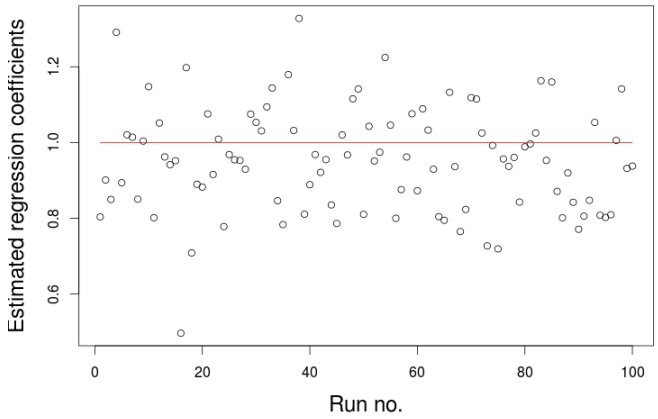


(c) Estimated regression coefficients of AB.

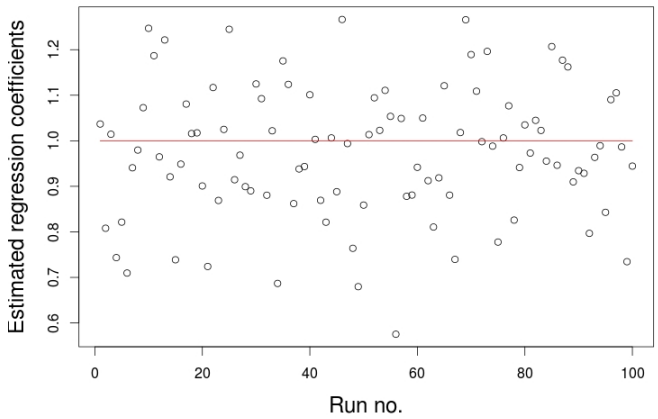
Figure 6.4: Single imputation using MLE for $C_r = 2.0$ and $C_l = -\infty$ for $\sigma = 0.3$.



(a) Estimated regression coefficients of A.

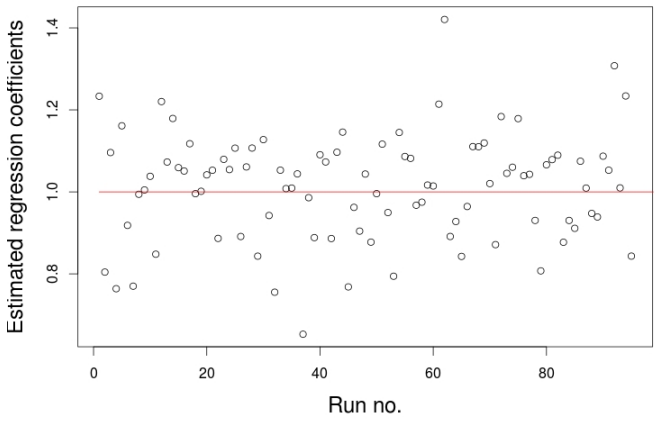


(b) Estimated regression coefficients of C.

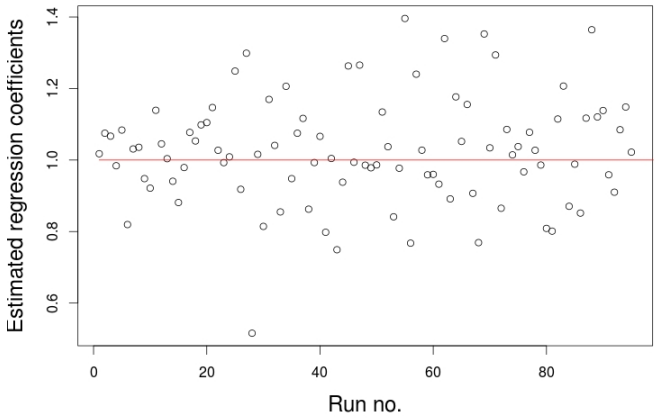


(c) Estimated regression coefficients of AB.

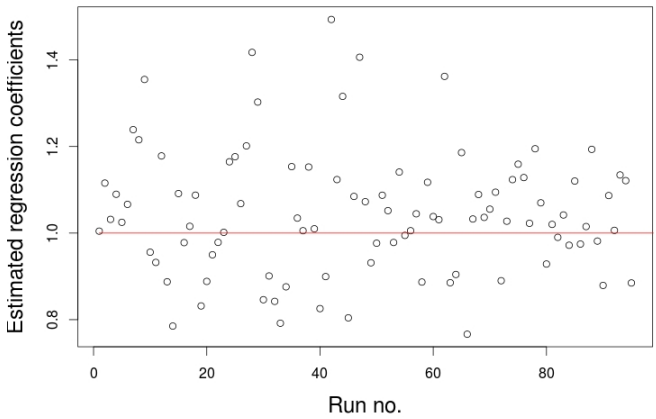
Figure 6.5: Multiple imputation using QD for $C_r = 2.0$ and $C_l = -\infty$ for $\sigma = 0.3$.



(a) Estimated regression coefficients of A.



(b) Estimated regression coefficients of C.



(c) Estimated regression coefficients of AB.

Figure 6.6: Multiple imputation using MLE for $C_r = 2.0$ and $C_l = -\infty$ for $\sigma = 0.3$.

Table 6.2: Results for $C_r = 2.9$ and $C_l = -\infty$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.0088407	0.0076118	0.0069562	0.0096872	0.0091996	0.0097453
0.2	0.0075462	0.0080493	0.010608	0.0073492	0.0092546	0.0077210
	0.0084009	0.0088701	0.0089861	0.0078975	0.0085871	0.0090601
Average	0.0082626	0.0081771	0.0088501	0.0083113	0.0090138	0.0088421
	0.019088	0.021719	0.020997	0.019140	0.021635	0.020530
0.3	0.017992	0.020034	0.024073	0.018956	0.020821	0.020580
	0.018106	0.016592	0.015350	0.025565	0.019483	0.020989
Average	0.018395	0.019445	0.20140	0.021220	0.020646	0.020700
	0.029516	0.030374	0.031802	0.028718	0.035654	0.035631
0.4	0.029465	0.033248	0.032724	0.034547	0.029681	0.036366
	0.028384	0.034998	0.030380	0.037970	0.030050	0.028913
Average	0.029122	0.032873	0.031635	0.033745	0.031795	0.033637

As for right censoring at 2.9, the four methods perform consistently very good for right censoring at 3.1.

The number of runs of which the maximum likelihood estimator do not exist varies between 0 and 9, giving between 91 and 100 acceptable runs.

Table 6.3: Results for $C_r = 3.1$ and $C_l = -\infty$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.0086548	0.0073425	0.0079220	0.0077118	0.0074838	0.0079452
0.2	0.0097494	0.0064637	0.0089186	0.0074871	0.0087324	0.0089398
	0.0078917	0.0071450	0.0076819	0.0093494	0.0088455	0.0089930
Average	0.0087653	0.0069837	0.0081742	0.0081828	0.0082237	0.0086260
	0.016366	0.015631	0.015086	0.018809	0.018961	0.017153
0.3	0.021605	0.019439	0.018724	0.020555	0.021734	0.018194
	0.016161	0.019097	0.020255	0.016844	0.019176	0.018476
Average	0.018044	0.018056	0.018022	0.018736	0.019957	0.017941
	0.027913	0.034563	0.029858	0.036948	0.037368	0.036071
0.4	0.026553	0.035999	0.035749	0.037852	0.037713	0.031684
	0.032711	0.037742	0.033246	0.031840	0.034973	0.038555
Average	0.029059	0.036101	0.032951	0.035547	0.036685	0.035437

6.1.5 Testing for $C_r = \infty$ and $C_l = -2.0$

The results of the experiment with left censoring at -2.0, are shown in table 6.4.

When considering left censoring at -2.0, the third item of the response vector, having a value equal to -3, is expected to be left censored. It is a small chance of items 1,2 and 7 being left censored as well.

The gross variances obtained by the four methods vary more in this experiment with left censored data at -2.0, than they did for right censored at 2.0. Evaluating the gross variances in table 6.4, simple imputation and multiple imputation both initialized by the maximum likelihood estimator give the two best results for all values of σ. The quick and dirty method gives the poorest result for $\sigma = 0.2$, while maximum likelihood estimation performs the worst when $\sigma = 0.3$ and 0.4. The gross variance varies especially for $\sigma = 0.4$,

from 0.032815 for the multiple imputation initialized by the maximum likelihood estimator to 0.11201 for maximum likelihood estimation. It is natural that the gross variance would vary more for a higher value of σ, as a higher value of σ leads to a higher value of the error term.

In this example, the imputation methods, with the exception of single imputation using the quick and dirty approach, give the best results.

The number of acceptable runs for methods using the maximum likelihood estimator decreases substantially when dealing with left censored data instead of right censored data. For right censored data at $C_r = 2.0$, the number of acceptable runs varied from 93 to 100. For left censored data at $C_l = -2.0$ however, the number varies from 39 to 60 acceptable runs.

Table 6.4: Results for $C_r = \infty$ and $C_l = -2.0$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.028639	0.027662	0.011835	0.0083657	0.010674	0.0095753
0.2	0.027121	0.019696	0.020378	0.0097537	0.012327	0.010302
	0.024505	0.013265	0.012539	0.0072168	0.011078	0.0082758
Average	0.026755	0.020208	0.014973	0.0084454	0.011360	0.0093844
	0.036399	0.051422	0.042157	0.016616	0.030273	0.016297
0.3	0.037158	0.045260	0.037179	0.017364	0.023647	0.016011
	0.038476	0.045865	0.032506	0.016652	0.025110	0.018415
Average	0.037344	0.047516	0.037281	0.016877	0.026343	0.016908
	0.046053	0.10609	0.075761	0.044960	0.039004	0.027662
0.4	0.048091	0.12382	0.068749	0.030015	0.040432	0.041420
	0.051616	0.10611	0.080734	0.035952	0.042616	0.029362
Average	0.048587	0.11201	0.075081	0.036976	0.040684	0.032815

6.1.6 Testing for $C_r = \infty$ and $C_l = -2.9$

For left censoring at $C_l = -2.9$, the gross variances obtained by the four methods are displayed in table 6.5.

Again, the third item of $\mathbf{y}$ is expected to be censored, but it will not always be the case. The items 1,2 and 7 could also be left censored, but the possibility is extremely small.

The two methods with the overall lowest gross variances for all values of σ are the quick and dirty method and multiple imputation using the quick and dirty approach. The next best method is multiple imputation initialized by the maximum likelihood estimator. The overall worst method is maximum likelihood estimation, which gives the highest values of the gross variance for all values of σ.

All in all, the performances are good, but maximum likelihood estimation distinguishes itself as the least good method.

The number of acceptable runs for all methods applying the maximum likelihood estimator varies from 52 to 70.

6.1.7 Testing for $C_r = \infty$ and $C_l = -3.1$

Table 6.6 shows the gross variances of the experiment with left censoring at $C_l = -3.1$.

As in the previous example, the third item of the response is expected to be left censored, and in addition there is an extremely small chance of the first, second and

Table 6.5: Results for $C_r = \infty$ and $C_l = -2.9$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.0084592	0.024477	0.0080811	0.0084098	0.0088219	0.0086626
0.2	0.0085777	0.016146	0.0082365	0.0094247	0.0077283	0.0083216
	0.0088554	0.020776	0.0078400	0.012646	0.0077953	0.0092039
Average	0.0086308	0.020466	0.0080525	0.010160	0.0081152	0.0087294
	0.019866	0.039179	0.020316	0.021859	0.018897	0.020116
0.3	0.018440	0.028385	0.021639	0.025338	0.023922	0.020816
	0.019539	0.033378	0.023259	0.027977	0.015779	0.022355
Average	0.019282	0.033647	0.021738	0.025058	0.019533	0.021096
	0.027237	0.10541	0.036467	0.051698	0.033120	0.031262
0.4	0.027851	0.072830	0.042962	0.034893	0.037258	0.028937
	0.031878	0.079606	0.027510	0.029631	0.028419	0.030328
Average	0.028989	0.085949	0.035646	0.038741	0.032932	0.030176

seventh item being censored as well.

The quick and dirty method continues to perform well, not unexpected considering that the censoring limit is close to the original value, and yields the lowest gross variances for $\sigma = 0.3$ and 0.4. For $\sigma = 0.2$, multiple imputation using the quick and dirty approach gives the best gross variance. The greatest gross variances are provided by maximum likelihood estimation for all values of σ. Single imputation and multiple imputation are fairly similar in their performances.

The number of acceptable runs for methods involving maximum likelihood estimation, is slighly higher than for the two other experiments with left censoring, the number varies from 53 to 78 runs.

Table 6.6: Results for $C_r = \infty$ and $C_l = -3.1$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.0072590	0.014601	0.0075967	0.0093927	0.0090571	0.0085983
0.2	0.0082214	0.013629	0.0084026	0.011034	0.0074294	0.0092066
	0.0080732	0.015711	0.0082096	0.011034	0.0066092	0.0066546
Average	0.0078512	0.014647	0.0080696	0.010487	0.0076986	0.0081532
	0.016453	0.039240	0.017937	0.020153	0.019320	0.018735
0.3	0.016336	0.047095	0.017291	0.023234	0.017036	0.018659
	0.017954	0.028645	0.018619	0.019132	0.022410	0.015192
Average	0.016914	0.038327	0.017949	0.020840	0.019589	0.017529
	0.032285	0.055789	0.035045	0.031660	0.040942	0.030530
0.4	0.029676	0.040414	0.043319	0.034971	0.031115	0.032491
	0.027057	0.043470	0.029679	0.036840	0.036404	0.032980
Average	0.029673	0.046558	0.036014	0.034490	0.036154	0.032000

6.1.8 Testing for $C_r = 2.0$ and $C_l = -2.0$

The results of the first example with both right and left censoring are dispayed in table 6.7.

There is a chance of all items of the response vector being censored, but of course the probability of the third and eighth element being censored is higher than for the other

elements.

Single imputation and multiple imputation both initialized by the maximum likelihood estimator gives the lowest values of the gross variances. Not surprisingly, the quick and dirty method gives the highest gross variances, as both censoring limits lie further away from the original values. Multiple imputation initialized by the quick and dirty approach performs a little better than single imputation initialized by the same approach.

For methods applying the maximum likelihood estimator, the number of accpetable runs varies from 38 to 69, although most experiments consist of more than 50 runs.

Table 6.7: Results for $C_r = 2.0$ and $C_l = -2.0$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.068356	0.033060	0.030848	0.010179	0.026441	0.010646
0.2	0.070414	0.016536	0.031695	0.010676	0.023609	0.012145
	0.069667	0.014878	0.030443	0.0094504	0.027131	0.013156
Average	0.069479	0.021491	0.030995	0.010102	0.025727	0.011982
	0.078411	0.052624	0.046099	0.028916	0.032260	0.023070
0.3	0.081861	0.025966	0.048689	0.027239	0.035249	0.024685
	0.078255	0.031666	0.059704	0.031087	0.035696	0.028680
Average	0.079509	0.036752	0.051497	0.029081	0.034402	0.025478
	0.085481	0.078305	0.072992	0.042772	0.052565	0.053458
0.4	0.085630	0.062727	0.074539	0.057021	0.048685	0.036799
	0.092177	0.066544	0.065862	0.039851	0.047353	0.041454
Average	0.087763	0.069192	0.071131	0.046548	0.049534	0.043904

6.1.9 Testing for $C_r = 2.9$ and $C_l = -2.9$

Table 6.8 displays the gross variances obtained by the four methods when data is both right censored at 2.9 and left censored at -2.9.

Again, the third and the eighth element of $\mathbf{y}$ are expected to be censored, but they don't need to be. The chance of other elements being censored is relatively small, but it is present.

For $\sigma = 0.2$, single imputation using the quick and dirty approach gives the lowest gross variance of 0.0073767, while for $\sigma = 0.3$ and 0.4, the quick and dirty methods yields the best gross variances of respectively 0.018106 and 0.032527. The worst gross variances for all values of σ are obtained by maximum likelihood estimation, with the values 0.014371, 0.032795 and 0.063956.

The quick and dirty method shows how well it performs when the actual values are close to the censoring limits, while the imputation methods illustartes the advantage of imputation, as both imputation methods using the maximum likelihood estimator perform better than their initializer, when the initial values are not too good.

The number of acceptable runs for each experiment of every method involving the maximum likelihood estimator varies from 54 to 78.

6.1.10 Testing for $C_r = 3.1$ and $C_l = -3.1$

The gross variances obtained from the four methods for left censoring at -3.1 and right censoring at 3.1, are shown in table 6.9.

As in the two previous examples, the third and eighth element of the response are expected to be censored, but need not to be, and the other elements may be censored as

Table 6.8: Results for $C_r = 2.9$ and $C_l = -2.9$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.0084499	0.014782	0.0074147	0.010361	0.0075742	0.012299
0.2	0.0096322	0.011072	0.0075734	0.010187	0.0075908	0.0078227
	0.0075499	0.017260	0.0071421	0.010464	0.010624	0.0098519
Average	0.0085440	0.014371	0.0073767	0.010337	0.0085963	0.0099912
	0.019101	0.031734	0.016420	0.038286	0.022430	0.019317
0.3	0.016840	0.035035	0.017955	0.027682	0.019452	0.019843
	0.018376	0.031615	0.020559	0.025363	0.020775	0.020643
Average	0.018106	0.032795	0.018311	0.030444	0.020886	0.019928
	0.035369	0.066835	0.047492	0.059473	0.037087	0.038642
0.4	0.030573	0.060673	0.033956	0.040450	0.038037	0.040274
	0.031640	0.064361	0.034110	0.049093	0.031424	0.040356
Average	0.032527	0.063956	0.038519	0.049672	0.035516	0.039757

well, even though the probability of that happening is small.

The results for right censoring at 3.1 and left censoring at -3.1 are pretty similar to the ones for right censoring at 2.9 and left censoring at -2.9. Again, the quick and dirty method gives the best results, while maximum likelihood estimation provides overall the worst results. But overall, the results of the methods are pretty good.

The number of acceptable runs varies from 67 to 82, for all methods using the maximum likelihood estimator.

Table 6.9: Results for $C_r = 3.1$ and $C_l = -3.1$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.0072173	0.024941	0.0079364	0.011783	0.0087936	0.0089530
0.2	0.0071209	0.015771	0.0079451	0.010666	0.010624	0.0062341
	0.0086558	0.018911	0.0088165	0.0090658	0.0085931	0.0093417
Average	0.0076647	0.019874	0.0082327	0.010505	0.0093369	0.0081763
	0.014513	0.034051	0.017909	0.029968	0.019381	0.019974
0.3	0.014324	0.020175	0.018928	0.029428	0.017593	0.018215
	0.017245	0.023594	0.023064	0.020467	0.019210	0.027654
Average	0.015361	0.025940	0.019967	0.026621	0.018728	0.021943
	0.025078	0.050994	0.030453	0.042218	0.030956	0.037581
0.4	0.030406	0.061977	0.049430	0.034521	0.032281	0.035094
	0.029153	0.076476	0.050401	0.036689	0.036760	0.035393
Average	0.028212	0.063149	0.043428	0.037809	0.033332	0.036023

6.1.11 Summary of example 1

Example 1 shows us how the four methods perform for censored data, and in fact the results are quite good.

The quick and dirty method performs well for situations where the actual values are close to the censoring limits, and not so good for the opposite cases. Maximum likelihood estimation gives accurate results for right censored data, but is not able to perform equally well for left censoring. It might have something to do with the implementation of the *survreg* function, but that's just pure conjecture on my part, it's not scientifically

proven. Generally, multiple imputation gives very good results. When the initializing method gives inaccurate results, this is often reflected in the results of multiple imputation, but the results of multiple imputation is still better than the results obtained by the initializing method. Multiple imputation' s ability to perform better than its initializing method can be seen in the results, with the exception in which the actual values are close to the censoring values and the quick and dirty method gives better results than the imputation methods initialized by the quick and dirty method. Single imputation also delivers good results, and also performs better than its initializing methods, but the method is more sensitive to the quality of the performance of the initializing method than multiple imputation.

6.2 Example 2

6.2.1 Choice of C_r, C_l and σ

The complexity of example 2 allows us to vary the censoring limits more than what was possible for example 1. As for example 1, the methods are tested for right censoring, left censoring and both right and left censoring. The censoring limits are chosen with particularly regard to the units 1,4,5 and 8 of $\mathbf{y}$ in table 5.2. If a value of $\mathbf{y}$ is censored or not, depends on the original value, the censoring limit and the estimated error, defined in 4.2.

Two limits close to the original values were chosen, 14 and 15, as right censoring and 5 and 6, as left censoring. In addition, two values further away were chosen, 12 and 8 for respectively right and left censoring. With the less stringent limits, 12 and 8, more values have the opportunity of being censored. Element 4 of $\mathbf{y}$ is expected to be right censored most times, thereafter element 8. Element 1 and secondly element 5 are expected to be left censored most times. There is also a chance of no items being censored at once, since the censoring is based on time and not on amount. A total of nine experiments are conducted; experiments where data is right censored at 12,14 or 15, left censored at 5,6 or 8, and additionally experiments where data is both right and left censored, at 12 and 8, 14 and 6, and 15 and 5. As for example 1, the experiments containing both right and left censored data are chosen with respect to the closeness to the original values.

The error term is estimated by the same procedure as for example 1. The simulation of ϵ is presented in section 4.2, while the value of σ is determined by the prespecified value of α. The values of σ are chosen to be the same as for example 1; 0.2, 0.3 and 0.4.

6.2.2 Testing for $C_r = 12$ and $C_l = -\infty$

Table 6.10 shows the gross variances obtained by the four methods for right censoring at 12.

The fourth and eighth element of $\mathbf{y}$ are expected to be right censored. In addition, the third and seventh element can be censored. On the other hand, there is possible to end up with no censored values.

This example illustates the advantage the maximum likelihood approach has over the quick and dirty approach when the actual values are not close to the censoring limits. The greatest gross variances are all obtained by the quick and dirty method, with 0.35751, 0.35433 and 0.34932 for respectively $\sigma = 0.2, 0.3$ and 0.4. The three lowest gross variances are assigned to methods using the maximum likelihood estimator. The results by single and multiple imputation using the quick and dirty method are better than the results by the quick and dirty method itself.

The quick and dirty method clearly appear as the weakest method, showing off the trouble the method encounters when the censoring limit is far away from the actual values, while the maximum likelihood estimator provides its strength.

Table 6.10: Results for $C_r = 12$ and $C_l = -\infty$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.35264	0.013507	0.15709	0.011170	0.17101	0.010197
0.2	0.35873	0.013299	0.15911	0.012040	0.16552	0.012514
	0.36117	0.014756	0.15848	0.013177	0.16285	0.012739
Average	0.35751	0.013854	0.15823	0.012129	0.16646	0.011817
	0.35497	0.030570	0.15684	0.023417	0.15742	0.031089
0.3	0.35708	0.025221	0.15787	0.034163	0.15738	0.027995
	0.35093	0.028552	0.15015	0.029185	0.16482	0.026164
Average	0.35433	0.028114	0.15495	0.028922	0.15987	0.028416
	0.35182	0.052097	0.15949	0.051441	0.16588	0.063650
0.4	0.34431	0.048662	0.17417	0.054454	0.16994	0.054378
	0.35184	0.051211	0.16381	0.047983	0.17060	0.053638
Average	0.34932	0.050657	0.16582	0.051293	0.16881	0.057222

6.2.3 Testing for $C_r = 14$ and $C_l = -\infty$

The gross variances obtained by the methods for right censoring at 14 are displayed in table 6.11.

Element 4 and 8 of $\mathbf{y}$ are expected to be censored, and maybe element 7 in addition. In other words, the number of censored items can vary from zero to three.

Again, the quick and dirty method presents itself as the weakest method, yielding the lowest gross variances in all cases. The best gross variances for $\sigma = 0.2$ and 0.3 are given by multiple imputation using the maximum likelihood estimator, while single imputation using the quick and dirty approach gives the best result for $\sigma = 0.4$. But, the results obtained by all methods using the maximum likelihood approach are very similar. The imputation methods using the quick and dirty approach perform better than the quick and dirty method itself, showing the improvement imputation can provide.

All in all, the performances are good for all methods, in which the quick and dirty method provides the least good results. None of the methods present itself as a clear winner.

6.2.4 Testing for $C_r = 15$ and $C_l = -\infty$

The results of right censoring at 15 are shown in table 6.12.

The fourth and eighth element of $\mathbf{y}$ are expected to be censored, but they may not be.

The results are pretty good for all the methods, it is hard to distinguish the best methods from the worst. The quick and dirty method provides the greatest gross variance for $\sigma = 0.2$, but at the same time, it gives the lowest gross variance for $\sigma = 0.3$ and 0.4. Maximum likelihood estimation gives the best result for $\sigma = 0.2$, and the worst for $\sigma = 0.4$. The poorest performance for $\sigma = 0.3$ is obtained by single imputation using the maximum likelihood estimator.

The attempt to distinguish the methods in this example does not tell us clearly which method is the most and least accurate, because there is very little difference between the gross variances. If the example was run again, the outcome could be a little different,

Table 6.11: Results for $C_r = 14$ and $C_l = -\infty$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.044466	0.011771	0.020270	0.014153	0.018861	0.013357
0.2	0.044691	0.015004	0.019720	0.011146	0.018580	0.012419
	0.044011	0.011662	0.021012	0.013387	0.018750	0.012068
Average	0.044389	0.012812	0.020334	0.012895	0.018730	0.012615
	0.048952	0.022620	0.030753	0.026373	0.028447	0.031144
0.3	0.050412	0.030334	0.029344	0.030078	0.027177	0.027159
	0.053434	0.030712	0.027192	0.027138	0.026614	0.022794
Average	0.050933	0.027889	0.029096	0.027863	0.027413	0.027032
	0.061838	0.046298	0.042648	0.052123	0.049755	0.050606
0.4	0.056998	0.048589	0.043024	0.046828	0.046996	0.048178
	0.054183	0.045571	0.048136	0.043929	0.043641	0.043704
Average	0.057673	0.046819	0.044603	0.047627	0.046797	0.047496

leaving other methods as the source of the best and worst results. However, the gross variances would still be very similar and hard to distinguish.

Table 6.12: Results for $C_r = 15$ and $C_l = -\infty$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.0098806	0.0086323	0.0088815	0.0090684	0.0080370	0.010005
0.2	0.0089540	0.0076464	0.0088264	0.010617	0.0088691	0.0094464
	0.010259	0.0075843	0.0081612	0.0078642	0.0093051	0.0091101
Average	0.0096979	0.0079543	0.0086230	0.0091832	0.0087371	0.0095205
	0.019030	0.018215	0.018119	0.025337	0.019674	0.019424
0.3	0.017297	0.021793	0.017945	0.022883	0.022873	0.019598
	0.019273	0.019993	0.019792	0.023822	0.015703	0.020360
Average	0.018533	0.020000	0.018619	0.024014	0.019417	0.019794
	0.034322	0.033242	0.038554	0.039780	0.034778	0.036653
0.4	0.034341	0.039266	0.033698	0.035392	0.037416	0.035288
	0.032211	0.040258	0.035160	0.032193	0.037503	0.040630
Average	0.033625	0.037589	0.035804	0.035788	0.036566	0.037524

6.2.5 Testing for $C_r = \infty$ and $C_l = 8$

The gross variances for left censoring at 8, are shown in table 6.13.

Element 1 and 5 of the response vector are clearly expected to be censored. There is also a chance of unit 2 and 6 being censored, and even unit 3.

Single and multiple imputation using the maximum likelihood estimator are the best performing methods, followed be maximum likelihood estimation. The weakest performance is assigned the quick and dirty method, yet the imputation methods using the quick and dirty method perform better than the initializing method itself. When the censoring limit lie further away from the actual values as in this case, we experience large variations in the gross variances. For $\sigma = 0.2$, the lowest gross variance has a value of 0.0074899, while the greastet is 0.41758. The lowest gross variance for $\sigma = 0.3$ is 0.020033, while the greatest is 0.44056. And finally, for $\sigma = 0.4$ the lowest gross variance obtained is 0.032612 and the greatest is 0.46549.

The methods using the maximum likelihood estimator perform well, while the quick and dirty approach reveals its disadvantages.

Table 6.13: Results for $C_r = \infty$ and $C_l = 8$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.41038	0.032005	0.11725	0.0073552	0.089462	0.0084226
0.2	0.42193	0.041002	0.12017	0.0085347	0.093989	0.011623
	0.42043	0.089375	0.11490	0.0065799	0.093755	0.012496
Average	0.41758	0.054127	0.11744	0.0074899	0.092402	0.010847
	0.43925	0.069492	0.15629	0.016890	0.12032	0.022296
0.3	0.44213	0.051367	0.13956	0.018989	0.11876	0.022026
	0.44031	0.075751	0.16549	0.049512	0.10661	0.015778
Average	0.44056	0.065537	0.15378	0.028464	0.11523	0.020033
	0.47451	0.10977	0.18052	0.027705	0.14238	0.035015
0.4	0.46385	0.13785	0.18927	0.036274	0.13850	0.037534
	0.45812	0.087515	0.19056	0.033857	0.14406	0.034041
Average	0.46549	0.11171	0.18678	0.032612	0.14165	0.035530

6.2.6 Testing for $C_r = \infty$ and $C_l = 6$

Table 6.14 shows the gross variances obtained for left censoring at 6.

For left censoring at 6, the first and fifth element of the response vector are expected to be censored, and in addition the sixth item can be censored as well. The items don't necessarily become censored.

Single and multiple imputation using the maximum likelihood estimator give the smallest gross variances, followed by single and multiple imputation using the quick and dirty approach. The worst results are obtained by the quick and dirty method. All in all, the results are good for each method.

Table 6.14: Results for $C_r = \infty$ and $C_l = 6$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.065126	0.024692	0.012499	0.0072587	0.016660	0.010008
0.2	0.060943	0.029710	0.012108	0.0097744	0.017486	0.010597
	0.065734	0.031622	0.012311	0.0091019	0.017061	0.0079775
Average	0.063934	0.028675	0.012306	0.0087117	0.017069	0.0095275
	0.081373	0.068070	0.022212	0.020477	0.026004	0.017928
0.3	0.083377	0.054639	0.022499	0.018046	0.022321	0.021068
	0.077444	0.045367	0.020649	0.017790	0.028935	0.022464
Average	0.080731	0.056025	0.021787	0.018771	0.025753	0.020487
	0.085309	0.069052	0.033333	0.032154	0.040698	0.033055
0.4	0.094217	0.085315	0.029925	0.031340	0.042283	0.042127
	0.090795	0.10203	0.053478	0.031850	0.039001	0.031599
Average	0.090107	0.085466	0.038912	0.031781	0.040661	0.035594

6.2.7 Testing for $C_r = \infty$ and $C_l = 5$

The gross variances obtained from the methods for left censoring at 5, are displayed in table 6.15.

For left censoring at 5, the first and fifth element of $\mathbf{y}$ are expected to be censored, but they might not be. The number of censored values in each run would vary from zero to two.

In this example, maximum likelihood estimation gives the greatest gross variances of 0.025691, 0.061321 and 0.10943 for $\sigma = 0.2, 0.3$ and 0.4 respectively. The lowest gross variances are obtained as 0.0074257 for $\sigma = 0.2$ and 0.018056 for $\sigma = 0.3$, both provided by multiple imputation using the maximum likelihood estimator, and 0.033626 for $\sigma = 0.4$ by single imputation using same approach. The quick and dirty method performs well alongside with the imputation methods.

The results show that maximum likelihood estimation performs weakest for left censoring at 5. As mentioned in section 4.1, the *survreg* function experiences a bit of trouble for left censoring. The better results of the quick and dirty method shows the advantage the method has when the censoring limits are close to original values.

Table 6.15: Results for $C_r = \infty$ and $C_l = 5$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.012762	0.029801	0.012204	0.0093888	0.0099609	0.0083879
0.2	0.012524	0.026963	0.012134	0.0076326	0.0077944	0.0060540
	0.011140	0.020309	0.0096951	0.012104	0.0094736	0.0078352
Average	0.012142	0.025691	0.011344	0.0097085	0.0090763	0.0074257
	0.024262	0.059491	0.022444	0.019773	0.019212	0.016832
0.3	0.024176	0.080820	0.022196	0.019919	0.017135	0.018252
	0.024609	0.043652	0.024474	0.034370	0.020606	0.019084
Average	0.024349	0.061321	0.023038	0.024687	0.018984	0.018056
	0.038358	0.12109	0.038648	0.030854	0.036804	0.035452
0.4	0.043783	0.10006	0.041888	0.027794	0.037658	0.040336
	0.036198	0.10714	0.044599	0.042230	0.036786	0.038838
Average	0.039446	0.10943	0.041712	0.033626	0.037083	0.038209

6.2.8 Testing for $C_r = 12$ and $C_l = 8$

The results of the experiment conducted with right censoring at 12 and left censoring at 8, are shown in table 6.16.

The first and fifth element of $\mathbf{y}$ are expected to be left censored, while the fourth and eighth element are expected to be right censored. The other elements may be censored as well.

The methods using the maximum likelihood estimator perform best, especially the imputation methods, while the methods applying the quick and dirty approach provide the weakest results. The difference in the gross variances is large, especially for $\sigma = 0.2$, where the lowest gross variance is of value 0.067813 while the greatest is 1.5263.

It is easy to consider the method of maximum likelihood as the best approach in this context. In addition, it is worth noticing that the imputation methods perform better than the initializing methods in all cases.

6.2.9 Testing for $C_r = 14$ and $C_l = 6$

The gross variances obtained for the methods with right censoring at 14 and left censoring at 6, are displayed in table 6.17.

Table 6.16: Results for $C_r = 12$ and $C_l = 8$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	1.5266	0.067523	1.1337	0.10323	1.1615	0.095927
0.2	1.5254	0.089822	1.1179	0.054655	1.1648	0.065767
	1.5269	0.067447	1.1239	0.050592	1.1609	0.041746
Average	1.5263	0.074931	1.1252	0.069492	1.1624	0.067813
	1.5322	0.27089	1.0894	0.15199	1.1585	0.12970
0.3	1.5300	0.17571	1.1225	0.12008	1.1654	0.18564
	1.5306	0.12951	1.0994	0.18657	1.1345	0.23360
Average	1.5309	0.19204	1.1038	0.15288	1.1528	0.18298
	1.5393	0.27636	1.1076	0.18788	1.1149	0.16001
0.4	1.5301	0.20697	1.1223	0.23289	1.1113	0.19919
	1.5384	0.23239	1.1182	0.19897	1.1373	0.13215
Average	1.5359	0.23857	1.1160	0.20658	1.1212	0.15850

Unit 5 and 8 of the response are expected to be right censored, while unit 1 and 4 are expected to be left censored. There is also a possibility of other units being censored and a possibility of no values being censored.

For $\sigma = 0.2$, the quick and dirty method gives a clearly greater gross variance than the other methods, while single and multiple imputation using the maximum likelihood approach provide the best results. Then, for $\sigma = 0.3$ and 0.4, imputation methods using the quick and dirty method appear as the best methods. For $\sigma = 0.4$, the methods using the maximum likelihood estimator perform at the same level as the quick and dirty method.

The imputation methods applying the quick and dirty method give the overall best results.

Table 6.17: Results for $C_r = 14$ and $C_l = 6$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.19287	0.086043	0.081578	0.038542	0.091465	0.060430
0.2	0.19254	0.092427	0.074262	0.055900	0.084207	0.062665
	0.19321	0.067865	0.079526	0.062420	0.088792	0.045265
Average	0.19287	0.082112	0.078455	0.052287	0.088155	0.056120
	0.20084	0.18881	0.079927	0.11099	0.087799	0.12951
0.3	0.20327	0.22210	0.083612	0.19249	0.094003	0.13851
	0.19533	0.18962	0.087945	0.10911	0.086723	0.12942
Average	0.19981	0.20018	0.083828	0.13753	0.089508	0.13248
	0.21017	0.23623	0.079225	0.17107	0.097981	0.20229
0.4	0.21268	0.21038	0.088753	0.20861	0.096971	0.20179
	0.21092	0.24691	0.085890	0.19773	0.094794	0.22931
Average	0.21126	0.23117	0.084623	0.19247	0.096582	0.21113

6.2.10 Testing for $C_r = 15$ and $C_l = 5$

Table 6.18 shows the results of the experiment of right censoring at 15 and left censoring at 5.

The fourth and eighth element of the response are expected to be right censored, along

with the first and fifth element as left censored. It is not likely that the other elements will be censored, but it's not completely impossible.

All methods perform better than they did in the previous example, which is natural considering the lower differance between the actual values and the censoring values. All imputation methods give lower gross variances than the initializing methods itself. For $\sigma = 0.2$, the quick and dirty method provides the least accurate result, while for $\sigma = 0.3$ and 0.4, maximum likelihood estimation gives the greatest gross variances. The lowest gross variance for $\sigma = 0.2$ is obtained by multiple imputation using the maximum likelihood estimator, while for $\sigma = 0.3$ and 0.4, multiple imputation initialized by the quick and dirty approach gives the best results.

Overall, each method provides good results.

Table 6.18: Results for $C_r = 15$ and $C_l = 5$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.023273	0.012521	0.012433	0.011979	0.011237	0.011023
0.2	0.021759	0.020291	0.014906	0.012835	0.012438	0.010077
	0.022398	0.017315	0.016485	0.028510	0.011967	0.011078
Average	0.022477	0.016709	0.014608	0.017775	0.011881	0.010726
	0.030678	0.037148	0.025723	0.023259	0.020642	0.032068
0.3	0.032012	0.058817	0.025769	0.033479	0.021766	0.019190
	0.031159	0.034149	0.023655	0.030581	0.016133	0.024160
Average	0.031283	0.043371	0.025049	0.029106	0.019514	0.025139
	0.041357	0.082361	0.039740	0.063951	0.030676	0.042563
0.4	0.047182	0.079931	0.039309	0.050861	0.032467	0.045539
	0.044036	0.080874	0.034914	0.055996	0.034777	0.044809
Average	0.044192	0.081055	0.037988	0.056936	0.032640	0.044304

6.2.11 Summary of example 2

Since the model in experiment 2 is more complex than the model in example 1, the variation of the original values in the response is greater, hence it is easier to vary the range of the censoring limits which leads to greater variations in the estimated regression coefficients, thus greater variations in the gross variances. Example 2 brings out the positive and negative sides of the four methods in a clearer way than example 1 does.

The impression of the performances of the methods are pretty much the same as in example 1. The quick and dirty method performs well when the actual values and censoring values are close. Maximum likelihood estimation also provides good results, mainly better than the quick and dirty method when the censoring values are not so close to the actual values. The imputation methods provide mainly very good results, of which multiple imputation give satisfactory results more frequently than single imputation.

As mentioned, example 2 is more complex than example 1, providing greater variations of the gross variances, as can be seen in the results. It is expected that the results would be less accurate in this example, since the censoring limits generally are less close to the actual values.

Chapter 7

Results of experiment 2

7.1 Example 1

The choices of censoring values and error term are the same as in experiment 1, presented in section 6.1.1.

The purpose of experiment 2 is, as mentioned in chapter 5, to compare the performances of the four methods more directly, by running them for the same error vector. In other words, the error term of each run is exactly the same for all four methods.

7.1.1 Testing for $C_r = 2.0$ and $C_l = -\infty$

The results of example 1 with right censoring at 2.0 are shown in table 7.1.

For every value of σ, the quick and dirty method provides the largest values of the gross variance; 0.019578, 0.025449 and 0.038323. Single imputation initialized by the maximum likelihood estimator yields the lowest gross variance for $\sigma = 0.2$, as 0.0092684. For $\sigma = 0.3$ and 0.4, single imputation initialized by the quick and dirty approach gives the best values, respectively 0.019498 and 0.034537. For $\sigma = 0.2$ and 0.3, all methods except the quick and dirty method give very similar results. Which method is the best is kind of random. The differences increase slightly for $\sigma = 0.4$. Single and multiple imputation using the quick and dirty approach perform well, even if the initializing method itself does not.

All methods perform well, but the quick and dirty method presents itself as the weakest method.

Table 7.1: Results for $C_r = 2.0$ and $C_l = -\infty$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.019561	0.012134	0.011486	0.010619	0.012070	0.010816
0.2	0.018768	0.0096867	0.010643	0.0089163	0.010825	0.0091240
	0.020406	0.0079558	0.011514	0.0082699	0.012064	0.0084464
Average	0.019578	0.0099255	0.011214	0.0092684	0.011653	0.0094621
	0.028642	0.019347	0.022138	0.020263	0.022727	0.020965
0.3	0.024146	0.022241	0.019008	0.021004	0.019458	0.021348
	0.023559	0.017725	0.017348	0.018404	0.017788	0.018560
Average	0.025449	0.019771	0.019498	0.019890	0.019991	0.020291
	0.040020	0.038690	0.036660	0.039789	0.038053	0.039724
0.4	0.037630	0.034823	0.034286	0.038680	0.034546	0.039039
	0.037319	0.031311	0.032664	0.034446	0.033460	0.034535
Average	0.038323	0.034941	0.034537	0.037638	0.035353	0.037766

7.1.2 Testing for $C_r = 2.9$ and $C_l = -\infty$

Table 7.2 shows the results of right censoring at 2.9.

The results are good and very similar for this example. None of the methods deviate in neither a good or bad way. Unlike in the previous example where the quick and dirty method provided the greatest gross variances in all cases, the method yields the best results for all cases in this example. The greatest gross variances are obtained by the imputation methods using the maximum likelihood estimator.

Every method performs very well for right censoring at 2.9. Even though it is difficult to distinguish the methods because the results are so similar, the quick and dirty method is slightly considered as the best method.

Table 7.2: Results for $C_r = 2.9$ and $C_l = -\infty$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.0083832	0.0087976	0.0086596	0.0089309	0.0087304	0.0089124
0.2	0.0064559	0.0082240	0.0070608	0.0077456	0.0071715	0.0077981
	0.0087600	0.0085339	0.0090638	0.0094017	0.0090326	0.0093652
Average	0.0078664	0.0085185	0.0082614	0.0086927	0.0083115	0.0086919
	0.014673	0.018634	0.015861	0.017209	0.015875	0.017096
0.3	0.019424	0.019120	0.020632	0.021869	0.020429	0.022097
	0.016990	0.021087	0.018177	0.019955	0.018220	0.020073
Average	0.017029	0.019614	0.018223	0.019678	0.018175	0.019755
	0.037170	0.035342	0.038821	0.040600	0.039123	0.040290
0.4	0.028284	0.027205	0.029267	0.030790	0.029400	0.031042
	0.028573	0.032640	0.032640	0.032334	0.031250	0.032549
Average	0.031342	0.031729	0.033576	0.034575	0.033258	0.034627

7.1.3 Testing for $C_r = 3.1$ and $C_l = -\infty$

The gross variances obtained by the methods for right censoring at 3.1 are shown in table 7.3.

As for right censoring at 2.9, the results are overall good and very similar. The method of maximum likelihood gives the best results for $\sigma = 0.2$ and 0.4, while the quick and dirty method gives the best result for $\sigma = 0.3$. The greatest gross variances for $\sigma = 0.2$ and 0.3 are provided by multiple imputation initialized by the maximum likelihood estimator, and for $\sigma = 0.4$, the worst result is given by single imputation initialized also by the maximum likelihood estimator.

Again, the differences in the gross variances of all methods are very small, thus all methods perform well for right censoring at 3.1.

7.1.4 Testing for $C_r = \infty$ and $C_l = -2.0$

Table 7.4 presents the results of left censoring at -2.0.

The two best results are obtained by the imputation methods using the maximum likelihood estimator in all cases. The worst results are provided by the quick and dirty method, maximum likelihood estimation and single imputation using the quick and dirty approach. The gross variances are not as similar as in the previous example, varying from 0.0081572 to 0.026819 for $\sigma = 0.2$, from 0.019914 to 0.048146 for $\sigma = 0.3$ and from 0.028508 to 0.10470 for $\sigma = 0.4$. It is intersting to notice that even though maximum

Table 7.3: Results for $C_r = 3.1$ and $C_l = -\infty$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.0084481	0.0088026	0.0088863	0.0091925	0.0088163	0.0092519
0.2	0.0078574	0.0073362	0.0081074	0.0083171	0.0081036	0.0083480
	0.0080800	0.0076329	0.0083112	0.0086726	0.0083106	0.0086500
Average	0.0081285	0.0079325	0.0084350	0.0087274	0.0084102	0.0087500
	0.015183	0.015780	0.015620	0.015940	0.015686	0.016067
0.3	0.017167	0.017719	0.018532	0.019082	0.018334	0.019120
	0.016676	0.018047	0.017516	0.018303	0.017427	0.018268
Average	0.016342	0.017182	0.017223	0.017775	0.017149	0.017818
	0.034761	0.031308	0.036483	0.038169	0.036457	0.038037
0.4	0.030971	0.029985	0.032035	0.032927	0.032297	0.033094
	0.032442	0.030369	0.034192	0.035269	0.034103	0.035133
Average	0.032725	0.030554	0.034237	0.035455	0.034286	0.035421

likelihood estimation provides the worst results in two cases, the imputation methods using this approach yields the best results.

The imputation methods present itself as the best methods in this example, especially the ones initialized by maximum likelihood estimation.

Table 7.4: Results for $C_r = \infty$ and $C_l = -2.0$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.025460	0.021919	0.011383	0.0085308	0.010341	0.0086657
0.2	0.028148	0.018426	0.010469	0.0075369	0.011361	0.0076510
	0.026850	0.019572	0.013225	0.0084040	0.011924	0.0088544
Average	0.026819	0.019972	0.011692	0.0081572	0.011209	0.0083904
	0.041077	0.039126	0.033119	0.018534	0.025290	0.018681
0.3	0.037930	0.049960	0.034737	0.021700	0.022346	0.021745
	0.039365	0.055353	0.035218	0.025530	0.023613	0.019315
Average	0.039457	0.048146	0.034358	0.021921	0.023750	0.019914
	0.047088	0.075107	0.061976	0.034080	0.042872	0.028011
0.4	0.047637	0.13866	0.068497	0.041921	0.037868	0.029560
	0.049759	0.10033	0.064270	0.027044	0.038150	0.027953
Average	0.048161	0.10470	0.064914	0.034348	0.039630	0.028508

7.1.5 Testing for $C_r = \infty$ and $C_l = -2.9$

The gross variances obtained by the four methods for left censoring at -2.9 are shown in table 7.5.

Maximum likelihood estimation provides the worst results with gross variances of 0.018598, 0.035284 and 0.068056 for $\sigma = 0.2, 0.3$ and 0.4 respectively. The best result for $\sigma = 0.2$ is given by single imputation using the quick and dirty method as 0.0075884. For $\sigma = 0.3$ and 0.4, the lowest gross variances are given by the quick and dirty method and has the values 0.018555 and 0.030528. Even though maximum likelihood estimation gives poor results, the imputation methods using this approach give good results, especially multiple imputation.

Overall, the results are good, but the least good results are provided by maximum

likelihood estimation.

Table 7.5: Results for $C_r = \infty$ and $C_l = -2.9$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.0076386	0.018160	0.0070020	0.0096975	0.0076061	0.0080639
0.2	0.0086759	0.018044	0.0082116	0.0091761	0.0087538	0.0086736
	0.0076080	0.019589	0.0075516	0.0087195	0.0082205	0.0084243
Average	0.0079742	0.018598	0.0075884	0.0091977	0.0081935	0.0083873
	0.019566	0.034528	0.018896	0.023091	0.019388	0.020797
0.3	0.019897	0.039307	0.026412	0.036243	0.022362	0.020116
	0.016201	0.032018	0.015544	0.018708	0.016551	0.017332
Average	0.018555	0.035284	0.020284	0.026014	0.019434	0.019415
	0.031308	0.072799	0.050609	0.037100	0.037650	0.032214
0.4	0.026992	0.055069	0.039773	0.028927	0.032201	0.026899
	0.033284	0.076299	0.055046	0.043614	0.041207	0.034817
Average	0.030528	0.068056	0.048476	0.036547	0.037003	0.031310

7.1.6 Testing for $C_r = \infty$ and $C_l = -3.1$

The results of left censoring at -3.1 are presented in table 7.6.

The lowest gross variances in all cases are provided by the quick and dirty method, closely followed by multiple imputation initialized by the maximum likelihood estimator. The greatest gross variances are, as for left censoring at -2.9, given by maximum likelihood estimation. Again, multiple imputation performs well even though the initializing method does not.

The results are all in all pretty good for all methods, but slightly better for the quick and dirty method and slighly worse for maximum likelihood estimation.

Table 7.6: Results for $C_r = \infty$ and $C_l = -3.1$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.0071564	0.014179	0.0084100	0.0080011	0.0086116	0.0079353
0.2	0.0084042	0.012254	0.0088032	0.010512	0.0088187	0.0080137
	0.0084432	0.013352	0.0088418	0.011691	0.0090625	0.0092303
Average	0.0080013	0.013362	0.0086850	0.010068	0.0088309	0.0083931
	0.015901	0.033863	0.017820	0.020019	0.019226	0.017248
0.3	0.018015	0.036192	0.020928	0.020092	0.021190	0.018702
	0.018752	0.032822	0.019933	0.022832	0.020121	0.020312
Average	0.017556	0.034292	0.019560	0.020981	0.020179	0.018754
	0.032494	0.054825	0.044020	0.039168	0.035869	0.030439
0.4	0.033459	0.054105	0.038932	0.036171	0.034777	0.035300
	0.027132	0.052798	0.041165	0.029321	0.033033	0.027877
Average	0.031028	0.053909	0.041372	0.034887	0.034560	0.031205

7.1.7 Testing for $C_r = 2.0$ and $C_l = -2.0$

Table 7.7 shows the results obtained for right censoring at 2.0 and left censoring at -2.0.

The greatest gross variances are given by the quick and dirty method for $\sigma = 0.2$ and 0.3, and by maximum likelihood estimation for $\sigma = 0.4$. For $\sigma = 0.2$ and 0.3, the best

results are obtained by single imputation initialized by the maximum likelihood estimator, while for $\sigma = 0.4$ the best result is obtained by multiple imputation using the quick and dirty approach.

Overall, for right and left censoring not close to the original values, the methods are doing a good job.

Table 7.7: Results for $C_r = 2.0$ and $C_l = -2.0$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.069778	0.021961	0.033289	0.015269	0.028472	0.015307
0.2	0.068867	0.011970	0.026406	0.0096821	0.026780	0.010042
	0.064740	0.014030	0.027266	0.0096899	0.021992	0.0095164
Average	0.067795	0.015987	0.028987	0.011547	0.025748	0.011622
	0.076559	0.032153	0.042944	0.023140	0.032919	0.023285
0.3	0.074564	0.028273	0.040087	0.022179	0.031505	0.022620
	0.076947	0.030476	0.050525	0.019501	0.036111	0.019462
Average	0.076023	0.030301	0.044519	0.021607	0.033512	0.021789
	0.085816	0.083670	0.064380	0.048405	0.045965	0.053104
0.4	0.090038	0.098831	0.077474	0.056835	0.051295	0.060493
	0.097045	0.094925	0.074582	0.058618	0.054235	0.052770
Average	0.090966	0.092475	0.072145	0.054619	0.050498	0.055456

7.1.8 Testing for $C_r = 2.9$ and $C_l = -2.9$

The gross variances given by the four methods for right censoring at 2.9 and left censoring at -2.9, are shown in table 7.8.

For $\sigma = 0.2$ and 0.3, the best results, 0.0076984 and 0.014995 are given by single imputation using the quick and dirty approach, while the worst results, 0.013515 and 0.040353 are provided by maximum likelihood estimation. The lowest gross variance, 0.030607, for $\sigma = 0.4$ is given by the quick and dirty method, while again, the worst result is given by maximum likelihood estimation. The results are mostly better in this example than in the previous example, which is not surprising considering that the censoring limits are closer to the actual values.

The performances of the methods are overall good. None of the methods is a clear winner, but the method of maximum likelihood presents itself as the poorest method.

7.1.9 Testing for $C_r = 3.1$ and $C_l = -3.1$

Table 7.9 displays the results of right censoring at 3.1 and left censoring at -3.1.

The quick and dirty method gives the lowest gross variances in all cases, followed by the imputation methods initialized by the quick and dirty approach. Maximum likelihood estimation yields the worst results, followed by single imputation and then multiple imputation both applying the maximum likelihood estimator.

The results are pretty good for all methods. The quick and dirty method appears to be the best method while the maximum likelihood estimator appears to be the poorest method.

7.1.10 Summary of example 1

Again, the results of example 1 show us that the methods generally perform well for censored data, as it did for example 1 in experiment 1.

Table 7.8: Results for $C_r = 2.9$ and $C_l = -2.9$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.0090737	0.015571	0.0077400	0.017740	0.0082221	0.0092118
0.2	0.0078370	0.012512	0.0076785	0.0088448	0.0084299	0.0087326
	0.0085076	0.012461	0.0076768	0.013641	0.0076913	0.010090
Average	0.0084728	0.013515	0.0076984	0.013409	0.0081144	0.0093448
	0.015551	0.035934	0.015512	0.023506	0.016511	0.018792
0.3	0.016961	0.037039	0.015611	0.021332	0.016665	0.020773
	0.014790	0.048086	0.013861	0.021811	0.015263	0.015904
Average	0.015767	0.040353	0.014995	0.022216	0.016146	0.018490
	0.029447	0.062272	0.048400	0.035390	0.035989	0.032904
0.4	0.030483	0.078613	0.033912	0.040440	0.033614	0.038984
	0.031892	0.071447	0.056902	0.056643	0.038095	0.052256
Average	0.030607	0.070777	0.046405	0.044158	0.035899	0.041381

Table 7.9: Results for $C_r = 3.1$ and $C_l = -3.1$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.0064087	0.019568	0.0069858	0.013249	0.0072975	0.0090056
0.2	0.0083868	0.019404	0.0097689	0.014203	0.0099161	0.010940
	0.0079434	0.013262	0.0087869	0.010469	0.0087956	0.010106
Average	0.0075796	0.017411	0.0085139	0.012640	0.0086697	0.010017
	0.015773	0.033461	0.023203	0.029423	0.018858	0.021492
0.3	0.017221	0.021312	0.017590	0.019913	0.017892	0.019376
	0.015604	0.049898	0.017769	0.042908	0.018722	0.024479
Average	0.016199	0.034890	0.019494	0.030748	0.018491	0.021782
	0.033688	0.070532	0.036722	0.054757	0.037518	0.043324
0.4	0.029841	0.057568	0.039514	0.049807	0.032539	0.039973
	0.031897	0.088731	0.041145	0.070451	0.037356	0.053265
Average	0.031809	0.072277	0.039127	0.058338	0.035804	0.045521

Since the error term is exactly the same for each method in each run, it is easier to examine the correlation between the performances of the imputation methods and the performances of the methods being used as initializers, if there is one. The results of example 1 in experiment 2 shows that the imputation methods usually perform better when the initializing method performs better, and otherwise. The ability of the imputation methods to give good results even though the initializing method does not, can also be seen from the results. The exception is for the quick and dirty approach when the censoring values are very close to the actual values. Then, the initializing method obtains better results than the imputation methods.

The results of the methods are pretty much the same as achieved in experiment 1. The quick and dirty method performs well for cases where the censoring limits are close to the original values and not so good for the opposite cases. Maximum likelihood estimation presents itself as a good method, but experiences problems when left censored data is present. The imputation methods give very good results, and quite often the gross variances of both single and multiple imputation are pretty similar.

7.2 Example 2

The censoring values and the error term are chosen to be the same as in experiment 1, as explained in section 6.2.1.

7.2.1 Testing for $C_r = 12$ and $C_l = -\infty$

Table 7.10 presents the results obtained by the methods for right censoring at 12.

In this example, there is a clear distinction between the results obtained by methods applying the quick and dirty approach and methods using the maximum likelihood estimator, where methods applying the maximum likelihood estimator give the most accurate results. The gross variances of the three methods using the maximum likelihood estimator are pretty similar, while for the methods using the quick and dirty method, the imputation methods give clearly better results than the quick and dirty method itself.

The maximum likelihood estimator shows the ability to perform well for censored data further away from the actual values, whereas the quick and dirty approach reveals its weakness for the same situation.

Table 7.10: Results for $C_r = 12$ and $C_l = -\infty$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.36143	0.012349	0.16071	0.011375	0.16927	0.011686
0.2	0.35556	0.013349	0.15361	0.011678	0.16147	0.011730
	0.35611	0.013368	0.15862	0.013489	0.15705	0.013613
Average	0.35770	0.013022	0.15765	0.012181	0.16260	0.012343
	0.34778	0.025607	0.15546	0.022205	0.15555	0.022681
0.3	0.35834	0.027668	0.16527	0.027046	0.16745	0.027771
	0.34040	0.032239	0.14845	0.030855	0.14685	0.031102
Average	0.34884	0.028505	0.15639	0.026702	0.15662	0.027185
	0.35791	0.047729	0.17494	0.051448	0.17632	0.052340
0.4	0.36795	0.042181	0.18325	0.045148	0.18387	0.046009
	0.35056	0.057595	0.16640	0.049912	0.17021	0.051776
Average	0.35881	0.049168	0.17486	0.048836	0.17680	0.050042

7.2.2 Testing for $C_r = 14$ and $C_l = -\infty$

The gross variances provided by the methods for right censoring at 14 are shown in table 7.11.

The greatest gross variances in all cases are obtained by the quick and dirty method. For $\sigma = 0.2$, the lowest gross variance is given by single imputation using the maximum likelihood estimator, while for $\sigma = 0.3$ and 0.4 the lowest gross variances are provided by single imputation applying the quick and dirty method. Both single and multiple imputation give pretty similar results for the same initializing method.

Overall, the methods perform well for right censoring at 14, but the quick and dirty method is the least accurate method.

7.2.3 Testing for $C_r = 15$ and $C_l = -\infty$

The results obtained for right censoring at 15 are presented in table 7.12.

The gross variances provided by all four methods are pretty good and similar for right censoring at 15. For $\sigma = 0.2$, the lowest and greatest gross variances are 0.0085997 and

Table 7.11: Results for $C_r = 14$ and $C_l = -\infty$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.041198	0.014314	0.017425	0.013392	0.016773	0.013535
0.2	0.046514	0.0094251	0.020264	0.0085167	0.020765	0.0085107
	0.042384	0.013586	0.017690	0.013374	0.018428	0.013661
Average	0.043365	0.012442	0.018460	0.011761	0.018553	0.011902
	0.048683	0.031359	0.029047	0.031192	0.029219	0.031569
0.3	0.046675	0.029384	0.026825	0.028464	0.027131	0.028125
	0.047938	0.025172	0.025185	0.022378	0.025904	0.022702
Average	0.047765	0.028638	0.027019	0.027345	0.027418	0.027465
	0.058439	0.050056	0.039787	0.046379	0.040050	0.046893
0.4	0.059558	0.051805	0.043771	0.051745	0.044097	0.052342
	0.057097	0.058498	0.043337	0.058216	0.043386	0.058632
Average	0.058365	0.053453	0.042298	0.052113	0.042511	0.052622

0.0096976, for $\sigma = 0.3$, the gross variance varies from 0.018089 to 0.020158 and for $\sigma = 0.4$, the lowest gross variance is 0.031537 while the greatest is 0.035849.

As the results show, there is very little variation in the gross variances, making it very difficult to decide which method performs the best and which performs the poorest.

Table 7.12: Results for $C_r = 15$ and $C_l = -\infty$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.010499	0.0081804	0.0089538	0.0087696	0.0089933	0.0088282
0.2	0.0096770	0.0093236	0.0088618	0.0096939	0.0088336	0.0097978
	0.0089168	0.0083703	0.0079834	0.0089862	0.0081442	0.0090731
Average	0.0096976	0.0086248	0.0085997	0.0091499	0.0086570	0.0092330
	0.018282	0.018359	0.017771	0.018973	0.017595	0.018930
0.3	0.016738	0.018702	0.017054	0.019421	0.016987	0.019265
	0.019247	0.020425	0.019893	0.022079	0.019739	0.022138
Average	0.018089	0.019162	0.018239	0.020158	0.018107	0.020111
	0.032147	0.029903	0.032541	0.033640	0.032650	0.034341
0.4	0.035301	0.040718	0.037693	0.041329	0.037063	0.040520
	0.027163	0.034703	0.029073	0.032577	0.029368	0.032349
Average	0.031537	0.035108	0.033436	0.035849	0.033027	0.035737

7.2.4 Testing for $C_r = \infty$ and $C_l = 8$

Table 7.13 shows the results obtained for left censoring at 8.

The imputation methods using the maximum likelihood estimator give the best results, followed by maximum likelihood estimation itself. The poorest results are obtained by the quick and dirty method, which in fact gives the least accurate results for all values of σ. The results provided by the imputation methods using the quick and dirty method are clearly better than the results obtained from the quick and dirty method itself. It is expected that the quick and dirty method will perform poorer than maximum likelihood estimation as the censoring limit are further away from the actual values.

As expected, the results obtained by imputation methods using the maximum likelihood estimator are good, while the results by the quick and dirty method are the least

accurate.

Table 7.13: Results for $C_r = \infty$ and $C_l = 8$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.41433	0.035881	0.11052	0.010103	0.091914	0.0096848
0.2	0.40881	0.028397	0.10011	0.0065899	0.098535	0.0064882
	0.42577	0.029514	0.11600	0.0066002	0.10773	0.0072348
Average	0.41630	0.031264	0.10888	0.0077644	0.099393	0.0078026
	0.43678	0.065358	0.14955	0.017164	0.11911	0.017130
0.3	0.43153	0.050826	0.13155	0.015890	0.11784	0.016436
	0.42870	0.085293	0.14015	0.023866	0.10960	0.021808
Average	0.43234	0.067159	0.14042	0.018973	0.11552	0.018458
	0.46746	0.12689	0.18282	0.028557	0.14801	0.028527
0.4	0.46766	0.11674	0.16597	0.033087	0.13680	0.031974
	0.46319	0.11974	0.19491	0.068490	0.14028	0.036123
Average	0.46610	0.12112	0.18123	0.043378	0.14170	0.032208

7.2.5 Testing for $C_r = \infty$ and $C_l = 6$

For left censoring at 6, the results are presented in table 7.14.

Again, the results obtained by the imputation methods applying maximum likelihood estimation are pretty good and similar, and it is difficult to distinguish the performances of the two methods from each other, but single imputation is slightly better than multiple imputation. For $\sigma = 0.2$ and 0.3, the quick and dirty method gives the least accurate results, while maximum likelihood estimation provides the least accurate result for $\sigma = 0.4$. In all cases, the imputation methods give better results than the initializing methods.

The results are mainly quite good for all methods, but the imputation methods perform the best, especially the ones being initialized by the maximum likelihood estimator.

Table 7.14: Results for $C_r = \infty$ and $C_l = 6$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.062769	0.027749	0.013165	0.0083120	0.016197	0.0084653
0.2	0.060828	0.028848	0.011543	0.0089611	0.015872	0.0089747
	0.060993	0.038759	0.012403	0.0070840	0.016473	0.0071500
Average	0.061530	0.031785	0.012370	0.0081190	0.016181	0.0081967
	0.075118	0.062418	0.021230	0.018370	0.026273	0.019200
0.3	0.073715	0.067541	0.023421	0.019463	0.027034	0.019937
	0.080975	0.079426	0.024811	0.023387	0.030379	0.023554
Average	0.076603	0.069795	0.023154	0.020407	0.027895	0.020897
	0.10023	0.13875	0.039731	0.041566	0.047151	0.041850
0.4	0.085661	0.078902	0.030172	0.027080	0.037563	0.027193
	0.084281	0.10554	0.030668	0.022993	0.035335	0.024162
Average	0.090057	0.10773	0.033524	0.030546	0.040016	0.031068

7.2.6 Testing for $C_r = \infty$ and $C_l = 5$

The results of left censoring at 5 can be found in table 7.15.

For left censoring at 5, single and multiple imputation initialized by the maximum likelihood estimator and multiple imputation using the quick and dirty approach give good and similar results for all values of σ. Maximum likelihood estimation provides the greatest gross variances for all cases. Not surprisingly, the quick and dirty method performs very well, considering that the censoring limit is close to the actual values. Even though maximum likelihood estimation gives the poorest results, the imputation methods initialized by this method provides results among the best.

All methods perform well for left censoring at 5, the least accurate results are given by the method of maximum likelihood.

Table 7.15: Results for $C_r = \infty$ and $C_l = 5$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.012730	0.021058	0.012283	0.0081045	0.0097628	0.0081212
0.2	0.012632	0.029994	0.013108	0.0090506	0.0093891	0.0084832
	0.014636	0.033708	0.010312	0.010260	0.0096691	0.010404
Average	0.013333	0.028253	0.011901	0.0091384	0.0096070	0.0090028
	0.022415	0.053851	0.019526	0.016714	0.017447	0.016508
0.3	0.023140	0.081960	0.023917	0.019873	0.021632	0.019133
	0.023881	0.055128	0.022823	0.019695	0.020680	0.018953
Average	0.023145	0.063646	0.022089	0.018761	0.019920	0.018198
	0.040734	0.11599	0.043613	0.038579	0.039572	0.037903
0.4	0.036127	0.094122	0.037890	0.036721	0.033923	0.036642
	0.036078	0.073632	0.037164	0.033230	0.033573	0.027708
Average	0.037646	0.094581	0.039556	0.036177	0.035689	0.034084

7.2.7 Testing for $C_r = 12$ and $C_l = 8$

The results obtained by the methods for right censoring at 12 and left censoring at 8 are shown in table 7.16.

In this example, there is a clear distinction between the results obtained by methods using the quick and dirty method and the maximum likelihood estimator, of which the methods using the maximum likelihood estimator perform the best. Multiple imputation using maximum likelihood estimation gives the most accurate results for all cases; 0.052605 for $\sigma = 0.2$, 0.12515 for $\sigma = 0.3$ and 0.21375 for $\sigma = 0.4$, closely followed by single imputation applying the maximum likelihood estimator as well. The quick and dirty method provides the greatest gross variances for all cases; 1.5256, 1.5306 and 1.5335 for $\sigma = 0.2, 0.3$ and 0.4 respectively. The imputation methods using the quick and dirty method give better results than the quick and dirty method itself.

In this example, the distinction between the quick and dirty method's and the method of maximum likelihood's ability to handle censoring limits far away from the actual values is evident, in favor of the maximum likelihood estimator.

7.2.8 Testing for $C_r = 14$ and $C_l = 6$

Table 7.17 presents the results obtained for right censoring at 14 and left censoring at 6.

For right censoring at 14 and left censoring at 5, the quick and dirty method gives the greatest gross variances for $\sigma = 0.2$ and 0.3, while maximum likelihood estimation yields the greatest gross variance for $\sigma = 0.4$. The imputation methods using the maximum likelihood estimator are the best methods for $\sigma = 0.2$, while for $\sigma = 0.3$ and 0.4, the

Table 7.16: Results for $C_r = 12$ and $C_l = 8$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	1.5264	0.060662	1.1305	0.040698	1.1733	0.039588
0.2	1.5260	0.068572	1.1283	0.052725	1.1601	0.051698
	1.5244	0.089436	1.1190	0.066813	1.1586	0.066530
Average	1.5256	0.072890	1.1259	0.053412	1.1640	0.052605
	1.5315	0.16198	1.1159	0.11887	1.1473	0.12051
0.3	1.5297	0.20839	1.1191	0.16397	1.1344	0.15867
	1.5307	0.12941	1.1096	0.097657	1.1325	0.096261
Average	1.5306	0.16659	1.1149	0.12683	1.1381	0.12515
	1.5369	0.28033	1.1024	0.21245	1.1104	0.21240
0.4	1.5324	0.21562	1.0987	0.17252	1.1174	0.16514
	1.5311	0.32441	1.0848	0.26595	1.1100	0.26370
Average	1.5335	0.27345	1.0953	0.21697	1.1126	0.21375

best methods are the imputation methods initialized by the quick and dirty method. This example shows that the maximum likelihood estimator is more affected by the value of the variance, σ. The gross variance of the quick and dirty method does not change much when the value of σ changes, which is the opposite case for the maximum likelihood estimator, where the gross variance increases considerably when the variance increases. Greater uncertainty in the data gives greater uncertainty in the computations and consequently in the computed gross variances.

The imputation methods give the most accurate results.

Table 7.17: Results for $C_r = 14$ and $C_l = 6$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.19314	0.096810	0.081159	0.077336	0.087107	0.081918
0.2	0.19366	0.10043	0.079136	0.073597	0.088334	0.074195
	0.19398	0.086166	0.079690	0.064763	0.088496	0.068120
Average	0.19359	0.094469	0.079995	0.070573	0.087979	0.074744
	0.19920	0.17837	0.077586	0.13710	0.086834	0.13353
0.3	0.19912	0.19379	0.080271	0.14693	0.088369	0.14982
	0.19685	0.15632	0.075927	0.12458	0.082053	0.12368
Average	0.19839	0.17616	0.077928	0.13620	0.085752	0.13568
	0.21246	0.21742	0.088179	0.17116	0.098602	0.16795
0.4	0.21051	0.24452	0.084780	0.19896	0.094677	0.19813
	0.21444	0.22459	0.086875	0.15931	0.096457	0.15913
Average	0.21294	0.22884	0.086611	0.17648	0.097082	0.17507

7.2.9 Testing for $C_r = 15$ and $C_l = 5$

The results of right censoring at 15 and left censoring at 5, are shown in table 7.18.

Multiple imputation applying the quick and dirty method gives the most accurate results in all cases; 0.012031, 0.020622 and 0.035303 for $\sigma = 0.2, 0.3$ and 0.4 respectively. The least accurate results in all cases are obtained by maximum likelihood estimation, yielding gross variances of 0.024945, 0.049329 and 0.084801 for $\sigma = 0.2, 0.3$ and 0.4 respectively. Common to all imputation methods is that the imputation method performs

better than the method used as initializer.

The results are pretty good for all four methods.

Table 7.18: Results for $C_r = 15$ and $C_l = 5$.

σ	QD	MLE	SI(w/QD)	SI(w/MLE)	MI(w/QD)	MI(w/MLE)
	0.022373	0.021200	0.015170	0.013333	0.012925	0.013810
0.2	0.023678	0.023349	0.014404	0.015136	0.012172	0.013937
	0.021765	0.030286	0.014916	0.014355	0.010997	0.014077
Average	0.022605	0.024945	0.014830	0.014275	0.012031	0.013941
	0.029056	0.058110	0.025126	0.027743	0.020844	0.027533
0.3	0.030128	0.048249	0.024731	0.024901	0.020309	0.024550
	0.029707	0.041627	0.025839	0.024737	0.020713	0.022565
Average	0.029630	0.049329	0.025232	0.025794	0.020622	0.024883
	0.044781	0.075961	0.037558	0.051218	0.033274	0.044168
0.4	0.045957	0.093262	0.037816	0.050864	0.034342	0.050428
	0.052803	0.085179	0.042660	0.064016	0.038294	0.060651
Average	0.047847	0.084801	0.039345	0.055366	0.035303	0.051749

7.2.10 Summary of example 2

When all four methods are performed for the same error term, the same units of the response will be censored for all methods. Then, it is easier to compare the performances of the methods to each other more directly, considering that the starting point is the same for all methods. Yet, the results are pretty similar to the results obtained for various error terms as investigated in experiment 1. The quick and dirty method performs well for censoring limits close to the actual values, and the maximum likelihood estimator is very accurate for right censoring and not so good for left censoring. The imputation methods give mainly better results than the initializing methods.

We can see from the results that the performances of the imputation methods and the initializing methods are related. If the initializing method gives a greater gross variance, the imputation method also gives a greater gross variance.

All in all, the performances of the methods are good, but as expected, the gross variances are greater than they were for example 1, since example 2 is a more complex model.

Chapter 8

Discussion

Chapter 6 and 7 showed the performances of the four methods for the two different models presented in chapter 5. Until now, we have discussed the performances of the methods for the nine different censoring cases for each model in each experiment. In this chapter, the performances of each method will be summarized and evaluated separately.

8.1 The quick and dirty method

Throughout all experiments, the quick and dirty method has performed well for all cases where the censoring limits are close to the original values, and not so well for the opposite cases. When the censoring limits are very close to the actual values, as they were for right censoring at 2.9 and 3.1 and left censoring at -2.9 and -3.1 in example 1, the quick and dirty method gives among the most accurate results, and sometimes even the most accurate results. It's not surprising that the results are excellent when the censored values are substituted by values that similar to the actual values. However, since we don't know the actual values of censored data, it is impossible to predict the outcome of computations performed by the quick and dirty method.

The imprecision of the results obtained by the quick and dirty method depends, as mentioned, on the deviation between the actual values and the censoring limits, since the quick and dirty method substitutes censored data by the censoring limits, treating them as actual values. The results presented in chapter 6 and 7 showed that the results were more accurate for example 1, where the differences between the censoring limits and the actual values are smaller than they are in example 2. The fact that example 1 is a simpler model than example 2 should not be forgotten in this discussion either. The least accurate results are obtained when both the right censoring limit and the left censoring limit are far away from the actual values, as for right censoring at 12 and left censoring at 8 in example 2. This significant variation of precision indicates that the quick and dirty method is an unstable method.

Advantages of the quick and dirty method are that the method is easy to use, the implementation is easy to perform, and the method is time efficient. On the other hand, it is not possible to know in advance if the results obtained are satisfactory or not. There is always a possibility of the results being poor, which is a clear disadvantage of the method.

The quick and dirty method is an appropriate method when the number of resources is very limited, but there are as mentioned no guarantees for the precision of the results, they can be very good or they can be very bad.

8.2 Maximum likelihood estimation

Maximum likelihood estimation performs good throughout the experiments, but experiences some problems when the data is left censored. When the data is right censored, the performance of the method is very good. The method is rarely the absolute best performing method, but in several cases the results obtained by maximum likelihood estimation are among the most accurate results. When the data is left censored, the results are poorer than if the data is right censored, even though the difference between the actual values and the censoring limits are similar for both cases of censoring.

The precision of the method of maximum likelihood depends on the difference between the censoring limits and the actual values, but not to the same extent as the quick and dirty method. The results are more accurate when the deviation between censoring limits and actual values are smaller. Since the exact values of censored data are unknown, it is not possible to know in advance how well the performance of the maximum likelihood estimator would be, but generally the results are quite good. The results provided by maximum likelihood estimation in these experiments are never as poor as some of the results obtained by the quick and dirty method.

Maximum likelihood estimation is easy to use and time efficient as the quick and dirty method, but when using maximum likelihood estimation there is always a possibility that the maximum likelihood estimator does not exist, meaning the method does not converge. Whenever the maximum likelihood estimator does not exist, the results are rather obscure and should not be included in the analysis. Since not all runs of an experiment are usable, due to non convergence of the maximum likelihood estimator, less than intended values have been used in the analysis. The issue of the non converging maximum likelihood estimator is particularly relevant when the data is left censored, as mentioned earlier. The reason why the maximum likelihood estimator does not handle left censoring as well as right censoring is not known. The problem may lie in the implementation of the *survreg* function, but that is just a conjecture on my part.

The method of maximum likelihood is an accurate method, which can easily be used in analysis with censored data, but one should be aware of the problems arising when dealing with left censoring.

8.3 Single imputation

Single imputation is a well performing method, as can be seen from the results in chapter 6 and 7, sometimes it even causes the most accurate results. The concept of single imputation is to compute the conditional expectation of the censored data on observed values of the other variables. Using available information about other values than the censored ones, increases the possibility of obtaining accurate results. In most of the censored cases investigated in this report, single imputation presents very good results. Single imputation is initialized by the quick and dirty method and the maximum likelihood estimator. The performance of single imputation is related to the performance of the initializing method; when the initializing method gives inaccurate results, single imputation does too. However, the results provided by single imputation is generally better than the ones obtained by the initializing method. The exceptions are cases where the actual values and the censoring limits are very close when the quick and dirty method is applied and cases with right censoring when the maximum likelihood estimator is used. When single imputation is initialized by the maximum likelihood estimator, the issue of non convergence of the maximum likelihood estimator arises, and the number of acceptable values for the standard errors and the estimated regression coefficients decreases.

Single imputation is a more complex method than both the quick and dirty method and maximum likelihood estimation, and requires a more comprehensive implementation. Especially when dealing with Weibull distributed data, as the exponential integral is included in the expression of the conditional expectation of the censored data. The process of applying MATLAB through R is very time consuming, which makes single imputation the absolute slowest method of the methods considered in this report. The time consumption is a clear disadvantage in this context, as is the underestimation of the standard errors of the estimated regression coefficients.

It is worth noticing that the results provided by single imputation using maximum likelihood estimation are very close to the results obtained by the quick and dirty method, whenever the quick and dirty method is performing well. Taking into account that single imputation is performing among the best for all censoring cases, single imputation using maximum likelihood estimation appears to be both accurate and safe. However, the time consumption is a disadvantage, comparing with multiple imputation.

All in all, single imputation is a reliable and precise method, which should be used when performing complex experiments and resources allow us to.

8.4 Multiple imputation

The performances of multiple imputation are generally pretty good through all experiments evaluated in this report. Considering the fact that the method imputes m values drawn from a truncated distribution and then uses the average value in the computations, it is not surprising that the method yields accurate results. The numerical computations in chapter 6 and 7, show that multiple imputation quite often is the method providing the best results. As for single imputation, multiple imputation is initialized by the quick and dirty method and the maximum likelihood estimator. And again, the performance of multiple imputation is also related to the performance of the initializing method. Whenever the results of the initializing method are good, the results of multiple imputation are also good, and usually better than the initializer's results. The exceptions are cases when the quick and dirty method is applied and the censoring limits are close to the actual values, and when the maximum likelihood estimator is applied for right censoring. When multiple imputation is initialized by the maximum likelihood estimator, the issue of non convergence of the maximum likelihood estimator arises, and the number of acceptable values for the standard errors and the estimated regression coefficients decreases.

Multiple imputation is a more complex method than the quick and dirty method and maximum likelihood estimation. The method requires more implementation, but unlike single imputation, we don't need to include MATLAB in the numerical computations. Thus the computation of the estimated regression coefficients is far less time consuming for multiple imputation than for single imputation.

Although multiple imputation requires more work and time than a simpler method, the good results are worth the extra effort, when the data sets to be analysed are not too big and the value of m is small. Multiple imputation is a precise and reliable method. As for single imputation, it is worth noticing that whenever the quick and dirty method provides the best results, the results obtained by multiple imputation are very close. Since the quick and dirty approach is an unstable method, multiple imputation using maximum likelihood estimation should be considered as the most reliable and accurate method.

Chapter 9

Conclusion

In this report we have investigated how four different methods manage censored data. The methods were tested for two different experiments and two different examples, both a 2^3 factorial experiment, but the complexity of the models were different. Three scenarios of censoring were considered; right censoring, left censoring, and both right and left censoring. Type I censoring was used, meaning the censoring times were fixed, while the number of censored units is random.

The results of the two conducted experiments show us that all four methods have the ability of handling censored data, but the quality of the results is somewhat varying. The performances of the methods depend on the which type of censoring is present and on the difference between the censoring limits and the actual values.

The achievement of the quick and dirty method is very dependent on the difference between the actual values and the censoring limits, since the quick and dirty method treats the censoring limits as actual values for the censored observations. The smaller the devations between the censoring limits and the original values are, the more accurate results the quick and dirty method yields.

Maximum likelihood estimation is not that dependent on the censoring limits, but the results show that maximum likelihood estimation performs slighly better when the censoring limits are closer to the actual values. However, the maximum likelihood estimator is more sensible for the value of the variance, σ, than the quick and dirty method is, yielding more accurate results for the lower values of σ. It is expected that greater variance will yield greater deviation in the estimated regression coefficients. When censoring is present, there is always a possibility that the maximum likelihood estimator would not converge and as the maximum likelihood estimator does not exist in several runs, the number of acceptable values of the estimated regression coefficients decreases. This also applies for the cases in which the maximum likelihood estimator is used as an initializer for the imputation methods.

Common to both imputation methods are their ability to mostly perform better than the initializing methods. Single imputation is able to achieve good results even though the initializer is not, but single imputation is however affected by the performance of the initializing method. When the initializing method gives inaccurate results, the same applies for single imputation, but the results obtained by single imputation are better than the ones provided by the initializer.

As for single imputation, multiple imputation is able to perform better than the initializing methods in most cases. In addition, multiple imputation is also affected by the performance of the initializer, poor results provided by the initializing method yields poor results obtained by multiple imputation, yet the results obtained by multiple imputation are better than the ones provided by the initializer.

These numerical results indicate that it is wise to apply the quick and dirty method when the censoring limits are really close to the actual values, that one should use maximum likelihood estimation when the censoring limits are a little further away from the actual values, and that one should apply the imputation methods when the difference between the censoring limits and the actual values are greater. The only problem here is that there is no way of knowing whether the actual values are close or not close to the censoring limits, since we don't know the exact values of censored data. The evaluating of which method to prefer must be based on other factors, such as the size of the experiment, the resources available and what level of accuracy one wants to achieve. Bearing in mind that the exact value of a censored data is unknown, the quick and dirty method is a risky choice, and should be avoided if a safer option is available, for instance methods using the maximum likelihood estimator. Based on the numerical results, the imputation methods applying the maximum likelihood estimator appear as the safest and most accurate methods. Considering the time consumption and the fact that single imputation underestimates the standard errors of the estimated regression coefficients, multiple imputation is a more appropriate choice than single imputation.

The purpose of experiment 2 was to conduct the methods for the same error vector, to check if similar starting point was necessary for distinguishing the performances of the methods. The similarity in the results for all cases in experiment 1 and experiment 2 show that same error term are not necessary in order to be able to distinguish the methods. However, it was interesting to see how the methods performed for the same starting point.

To obtain a more precise analysis of the methods, several more models and censoring limits should be considered. It may also be necessary to look for alternative ways to compute the maximum likelihood estimator, as the *survreg* function experiences problems for left censoring, either by trying to find another built-in mechanisms which will do the job or by implementing the whole function yourself. Another solution to the issue with the computation of the exponential integral in single imputation should also be sought. To expand the experiment further, the methods could be tested for type II censoring. That is, experiments of which the number of censored units is fixed and the experimental time is random.

It is important to emphazise that it is not possible to draw a final conclusion about which method is by far the best or worst method in general based on this analysis. In order to do so, the number of data sets, models and censoring limits must be incredible higher and the analysis would be extremely more comprehensive. That being said, the analysis do give us an indication of what methods are the most and the least accurate and reliable. All four methods are potential candidates for handling censored data, but multiple imputation using maximum likelihood estimation appears to be the most appropriate one.

Bibliography

[1] Guo, H. and Mettas, A. (2010). *Design of Experiments and Data Analysis*, "2010 Reliability and Maintainability Symposium, San Jose, CA, USA.

[2] Hamada, M. and Wu, C. F. J. (1991). *Analysis of Censored Data From Highly Fractionated Experiments*, Vol. 33, NO.1 , pp. 25-38.

[3] Ihaka, R. (1998). *R : Past and Future History*, Interface98 (Technical report). Statistics Department, The University of Auckland, Auckland, New Zealand.

[4] Joseph, G. (2012). *Experimental Design for Reliability Improvement.* Trondheim: Norwegian University of Science and Technology.

[5] Little, J. A. & Rubin, B. (2002). *Statistical analysis with missing data*, Second edition.

[6] Rubin, D. B. (1987). *Multiple imputation for nonresponse in surveys.*

[7] Schafer, J. L. (1999). *Multiple imputation: a primer*, Department of Statistics, The Pennsylvania State University. Statistical Methods in Medical Research, vol 8. NO.1, pp. 3-15.

[8] Sue-Chu, A. M. (2013). *Methods for Dealing with Censored Data using Experimental Design.* Trondheim: Norwegian University of Science and Technology.

[9] Walpole, R.E., Myers, R.H., Myers, S. L. and Ye, K. (2007). *Probability and Statistics for Engineers and Scientists*, 8th edition.

[10] Wu, C. F. J. and Hamada, M. (2009). *Experiments: Planning, Analysis, and Optimization*, 2nd edition.

[11] http://en.wikipedia.org/wiki/Generalized_extreme_value_distribution. *Generalized extreme value distribution*, [Reading date 20 October, 2013]

[12] http://en.wikipedia.org/wiki/Gumbel_distribution. *Gumbel distribution*, [Reading date 20 October, 2013]

Appendix A

R code

The four methods are implemented in R. The code for example 1 and experiment 1 are included. The R packages needed to perform all computations are included in the code for the quick and dirty method.

The quick and dirty method

The code for the quick and dirty method for right censoring at 2.0 and $\sigma = 0.3$.

```
library(splines)
library(survival)
# library(CensReg)
library(MASS)
library(FrF2)
install.packages('fExtremes')
library(norm)
library(fExtremes)
install.packages('reliaR')
library(reliaR)

install.packages('R.oo')
install.packages('R.matlab')
install.packages('R.utils')

library(R.oo)
library(R.matlab)
library(R.utils)

# Starting MATLAB-server, assigning to MATLAB,
Matlab$startServer()
matlab <- Matlab()
# testing if connected or not connected
isOpen <- open(matlab)
print(matlab)

#########################################################

A  <- c(-1, 1,-1, 1,-1, 1,-1, 1)
C  <- c(-1,-1,-1,-1, 1, 1, 1, 1)
AB <- c( 1,-1,-1, 1, 1,-1,-1, 1)

y <- c(-1,-1,-3,1,1,1,-1,3)
n <- 100
trueCoeffs <- c(1,1,1)                # True coefficients
```

```
lengthY <- length(y)

right <-  2.0                          # Right censored limit
left  <- -Inf                          # Left censored limit

estimatedCoeffs <- matrix(0,3,n)

#########################################################

 for(j in 1:n){

########################

yVector <- vector()
errorvec <- vector()
eps <- vector()

for (i in 1:lengthY){

  v <- runif(1)

  alpha <- 3.33                        # To obtain sd.
  eps[i] <- log(-log(1-v))             # Epsilon
  sd <- 1/alpha                        # Standard error

  errorvec[i] <- eps[i] * sd           # The error vector

  yVector[i] <- y[i] + errorvec[i]
}

#########################################################

# If censored, assign censoring limit as actual value

for (i in 1:lengthY){
  if (yVector[i] > right){
    yVector[i] <- right
  }
  if(yVector[i] < left){
    yVector[i] <- left
  }
}

#########################################################

lmQD <- lm(yVector~A+C+AB)                  # Fitting linear model
coeffs <- lmQD$coef

a <- coeffs[2]; c <- coeffs[3]; ab <- coeffs[4]

values <- c(a,c,ab)
sigma <- summary(lmQD)$sigma

estimatedCoeffs[,j] <- t(values)  # Saving the estimated coefficients
}
```

```
################   GROSS VARIANCE   ###################

# Computing variance of A

aValues <- estimatedCoeffs[1,]
aTotal <- 0

for (i in 1:n){
  error <- (aValues[i]-trueCoeffs[1])^2
  aTotal <- aTotal + error
}

# Computing variance of C

cValues <- estimatedCoeffs[2,]
cTotal <- 0

for (i in 1:n){
  error <- (cValues[i]-trueCoeffs[2])^2
  cTotal <- cTotal + error
}

# Computing variance of AB

abValues <- estimatedCoeffs[3,]
abTotal <- 0

for (i in 1:n){
  error <- (abValues[i]-trueCoeffs[3])^2
  abTotal <- abTotal + error
}

# Computing the gross variance

GV <- (1/3)*((1/n)*aTotal + (1/n)*cTotal + (1/n)*abTotal)
print(GV)
```

Maximum likelihood estimation

The R code for maximum likelihood estimation for right censoring at 2.0 and $\sigma = 0.3$.

```
A  <- c(-1, 1,-1, 1,-1, 1,-1, 1)
C  <- c(-1,-1,-1,-1, 1, 1, 1, 1)
AB <- c( 1,-1,-1, 1, 1,-1,-1, 1)

y <- c(-1,-1,-3,1,1,1,-1,3)
n <- 100
trueCoeffs <- c(1,1,1)

lengthY <- length(y)

right <- 2.0                          # Right censoring value
left <- -Inf                          # Left censoring value
sdupper <- 0.5095169
sdlower <- 0.02353152
```

```
estimatedCoeffs <- matrix(0,3,n)

for (j in 1:n){                        # For loop for 100 runs

yVector <- vector()
errorvec <- vector()
eps <- vector()

#########################################################
for (i in 1:lengthY){

  v <- runif(1)

  alpha <- 3.33                        # To obtain sd
  eps[i] <- log(-log(1-v))             # Epsilon
  sd <- 1/alpha                        # Standard error

  errorvec[i] <- eps[i] * sd           # The error vector

  yVector[i] <- y[i] + errorvec[i]

}             # For loop ends

#########################################################

# Checking if censored and assigning values
# Right censored = 0, event=1, left censored=2, interval censored=3

censored <- vector()

for (i in 1:lengthY){
  if( yVector[i] > right ){
    censored[i] <- 0
  }
  else if (yVector[i] < left){
    censored[i] <- 2
  }
  else{
    censored[i] <- 1
  }
}

#########################################################

x <- exp(yVector)                # survreg using log(y) when computing

m1 <- lm(yVector~A+C+AB)

ysurv <- survreg(formula=Surv(x,censored)~A+C+AB,
         dist="weibull", init=coef(m1))

coeffs <- ysurv$coef
a <- coeffs[2]; c <- coeffs[3]; ab <- coeffs[4]

alpha1 <- ( 1/ysurv$scale)          # Shape parameter
theta1 <- (exp(coeffs[1]))          # Scale parameter
```

```
sigma <- 1/alpha1

sigmavalues[j] <- sigma

if(all(is.na(coeffs)==FALSE)){

   stderror <- summary(ysurv)$table[,2]

   if(stderror[2] < sdupper && stderror[2] > sdlower &&
      stderror[3] < sdupper && stderror[3] > sdlower &&
      stderror[4] < sdupper && stderror[4] > sdlower &&
      is.na(stderror[2:4])==FALSE){

   values <- c(a,c,ab)

   estimatedCoeffs[,j] <- t(values)   # Saving the estimated coefficients
  }
 }
}

############################################################

# Deleting unacceptable values

estimatedCoeffs2 <- estimatedCoeffs[,colSums(estimatedCoeffs==0)==0]

r <- ncol(estimatedCoeffs2)     # New number of acceptable runs

################   GROSS VARIANCE   ###################

# Computing variance of A

aValues <- estimatedCoeffs2[1,]
aTotal <- 0

for (i in 1:r){
  error <- (aValues[i]-trueCoeffs[1])^2
  aTotal <- aTotal + error
}

# Computing variance of C

cValues <- estimatedCoeffs2[2,]
cTotal <- 0

for (i in 1:r){
  error <- (cValues[i]-trueCoeffs[2])^2
  cTotal <- cTotal + error
}

# Computing variance of AB

abValues <- estimatedCoeffs2[3,]
abTotal <- 0

for (i in 1:r){
```

```
  error <- (abValues[i]-trueCoeffs[3])^2
  abTotal <- abTotal + error
}

# Computing the gross variance

GV <- (1/3)*((1/r)*aTotal + (1/r)*cTotal + (1/r) * abTotal)
print(GV)
```

Single imputation, using the quick and dirty method

The R code for single imputation using the quick and dirty method for right censoring at 2.0 and $\sigma = 0.3$.

```
A  <- c(-1, 1,-1, 1,-1, 1,-1, 1)
C  <- c(-1,-1,-1,-1, 1, 1, 1, 1)
AB <- c( 1,-1,-1, 1, 1,-1,-1, 1)

y <- c(-1,-1,-3,1,1,1,-1,3)
n <- 100
trueCoeffs <- c(1,1,1)

lengthY <- length(y)

right <-  2.0                    # Right censoring limit
left  <- -Inf                    # Left censoring limit
emc <- 0.5772                    # The Euler-Mascheroni constant

estimatedCoeffs <- matrix(0,3,n)

################################################

for (j in 1:n){

###############################################

yVector <- vector()
errorvec <- vector()
eps <- vector()

for (i in 1:lengthY){

  v <- runif(1)

  alpha <- 3.33                          # To obtain sd
  eps[i] <- log(-log(1-v))               # Epsilon
  sd <- 1/alpha                          # Standard error

  errorvec[i] <- eps[i] * sd             # The error vector

  yVector[i] <- y[i] + errorvec[i]

}

#################################################
```

```
# Checking if censored and assigning values
# Right censored = 0, event=1, left censored=2, interval censored=3

censored <-vector()

for (i in 1:lengthY){
  if (yVector[i] > right){
    yVector[i] <- right
    censored[i] <- 0
  }
  else if(yVector[i] < left){
    yVector[i] <- left
    censored[i] <- 2
  }
  else
    censored[i] <- 1
}

#######################################################

lmQD <- lm(yVector~A+C+AB)               # Fitting linear model

coeffs <- lmQD$coef
a <- coeffs[2]; c <- coeffs[3]; ab <- coeffs[4]

mu <- coeffs[1] +  A*a + C*c + AB*ab

sigma <- summary(lmQD)$sigma

#########################################################

# Matlab computing E1right and E1left

expintRight <- 0
expintLeft <- 0
setVariable(matlab, mu=mu, sigma=sigma, right=right, left=left,
            expintRight=expintRight, expintLeft = expintLeft)

expintR <- evaluate(matlab,"expintRight=expint(exp((right-mu)/sigma));")
dataR <- getVariable(matlab, "expintRight")
E1right<- dataR$expintRight

expintL <- evaluate(matlab,"expintLeft=expint(exp((left-mu)/sigma));")
dataL <- getVariable(matlab, "expintLeft")
E1left <- dataL$expintLeft

#########################################################

# Right censored

zr <- (right - mu)/sigma
expectedRight <- (1/(1-(1-pgumbel(-right,-mu,sigma))))
                 *(sigma*((zr*exp(-exp(zr))) + E1right)
                 + mu*exp(-exp(zr)))

# Left censored
```

```
zl <- (left-mu)/sigma
expectedLeft <- (1/(1-pgumbel(-left,-mu,sigma)))
                *(mu - emc*sigma - sigma*E1left
                - mu*exp(-exp(zl)))

# Changing value of y if censored

for (i in 1:lengthY){
  if(censored[i]==0){
    yVector[i] <- expectedRight[i]
  }
  else if(censored[i]==2){
    yVector[i] <- expectedLeft[i]
  }
}

# Computing parameters again, with new values for y

lmQD2 <- lm(yVector~A+C+AB)               # Fitting linear model

coeffs2 <- lmQD2$coef
a <- coeffs2[2]; c <- coeffs2[3]; ab <- coeffs2[4]

values <- c(a,c,ab)

estimatedCoeffs[,j] <- t(values)   # Saving the estimated coefficients
}

##################   GROSS VARIANCE   ###################

# Computing variance of A

aValues <- estimatedCoeffs[1,]
aTotal <- 0

for (i in 1:n){
  error <- (aValues[i]-trueCoeffs[1])^2
  aTotal <- aTotal + error
}

# Computing variance of C

cValues <- estimatedCoeffs[2,]
cTotal <- 0

for (i in 1:n){
  error <- (cValues[i]-trueCoeffs[2])^2
  cTotal <- cTotal + error
}

# Computing variance of AB

abValues <- estimatedCoeffs[3,]
abTotal <- 0
```

```
for (i in 1:n){
  error <- (abValues[i]-trueCoeffs[3])^2
  abTotal <- abTotal + error
}

# Computing the gross variance

GV <- (1/3)*((1/n)*aTotal + (1/n)*cTotal + (1/n) * abTotal)
print(GV)
```

Single imputation, using maximum likelihood estimation

The R code for single imputation using maximum likelihood estimation for right censoring at 2.0 and $\sigma = 0.3$.

```
A  <- c(-1, 1,-1, 1,-1, 1,-1, 1)
C  <- c(-1,-1,-1,-1, 1, 1, 1, 1)
AB <- c( 1,-1,-1, 1, 1,-1,-1, 1)

y <- c(-1,-1,-3,1,1,1,-1,3)
n <- 100
trueCoeffs <- c(1,1,1)

lengthY <- length(y)

right <- 2.0                    # Right censoring limit
left  <- -Inf                   # Left censoring limit
emc <- 0.5772                   # The Euler-Mascheroni constant
sdupper <- 0.5095169
sdlower <- 0.02353152

estimatedCoeffs <- matrix(0,3,n)

##########################################################

for (j in 1:100){

##########################################################

yVector <- vector()
errorvec <- vector()
eps <- vector()

for (i in 1:lengthY){

  v <- runif(1)

  alpha <- 3.33                             # To obtain sd
  eps[i] <- log(-log(1-v))                  # Epsilon
  sd <- 1/alpha                             # Standard error

  errorvec[i] <- eps[i] * sd                # The error vector

  yVector[i] <- y[i] + errorvec[i]
}
```

```
#####################################################

# Checking if censored and assigning values
# Right censored = 0, event=1, left censored=2, interval censored=3

censored <- vector()

for (i in 1:lengthY){
  if( yVector[i] > right ){
    censored[i] <- 0
  }
  else if (yVector[i] < left){
    censored[i] <- 2
  }
  else{
    censored[i] <- 1
  }
}

#########################################################

x <- exp(yVector)              # survreg using log(y) when computing

m1 <- lm(yVector~A+C+AB)

ysurv <- survreg(formula=Surv(x,censored)~A+C+AB,
         dist="weibull", init=coef(m1))

coeffs <- ysurv$coef
a <- coeffs[2]; c <- coeffs[3]; ab <- coeffs[4]

mu <- coeffs[1] + A*a + C*c + AB*ab

alpha1 <- ( 1/ysurv$scale)                  # Shape parameter
theta1 <- (exp(coeffs[1]))                  # Scale parameter

sigma <- 1/alpha1

if(all(is.na(coeffs)==FALSE)){
  stderror <- summary(ysurv)$table[,2]

  if(stderror[2] < sdupper && stderror[2] > sdlower &&
     stderror[3] < sdupper && stderror[3] > sdlower &&
     stderror[4] < sdupper && stderror[4] > sdlower &&
     is.na(stderror[2:4])==FALSE){

#########################################################

# Matlab computing E1(right) and E1(left)

expintRight <- 0
expintLeft <- 0
setVariable(matlab, mu=mu, sigma=sigma, right=right, left=left,
            expintRight=expintRight, expintLeft = expintLeft)

expintR <- evaluate(matlab,"expintRight=expint(exp((right-mu)/sigma));")
```

```
dataR <- getVariable(matlab, "expintRight")
E1right<- dataR$expintRight

expintL <- evaluate(matlab,"expintLeft=expint(exp((left-mu)/sigma));")
dataL <- getVariable(matlab, "expintLeft")
E1left <- dataL$expintLeft

#########################################################

  # Right censored

  zr <- (right - mu)/sigma
  expectedRight <- (1/(1-(1-pgumbel(-right,-mu,sigma))))
      *(sigma*((zr*exp(-exp(zr))) + E1right)
      + mu*exp(-exp(zr)))

# Left censored

  zl <- (left-mu)/sigma
  expectedLeft <- (1/(1-pgumbel(-left,-mu,sigma)))
      *(mu - emc*sigma - sigma * E1left
      - mu*exp(-exp(zl)))

  # Imputation if value of y is censored.

  for (i in 1:lengthY){
    yVector[i] <- ifelse(yVector[i] < left,
    expectedLeft[i], yVector[i])
    yVector[i] <- ifelse(yVector[i] > right,
    expectedRight[i], yVector[i])
  }

  if(any(is.na(yVector))==FALSE &&
      any(is.infinite(yVector))==FALSE){

#########################################################

 lmSIml <- lm(yVector~A+C+AB)    # Fitting the model
 coeffs2 <- lmSIml$coef

 a <- coeffs2[2]; c <- coeffs2[3]; ab <- coeffs2[4]

 values <- c(a,c,ab)

 # Saving the estimated coefficients
 estimatedCoeffs[,j] <- t(values)

   }
  }
 }
}

#########################################################

# Deleting all elements not suitable
```

```
estimatedCoeffs2 <- estimatedCoeffs[,colSums(estimatedCoeffs ==0)==0]

r <- ncol(estimatedCoeffs2)     # New number of acceptable runs

###############  GROSS VARIANCE   ###################

# Computing variance of A

aValues <- estimatedCoeffs2[1,]
aTotal <- 0

for (i in 1:r){
  error <- (aValues[i]-trueCoeffs[1])^2
  aTotal <- aTotal + error
}

# Computing variance of C

cValues <- estimatedCoeffs2[2,]
cTotal <- 0

for (i in 1:r){
  error <- (cValues[i]-trueCoeffs[2])^2
  cTotal <- cTotal + error
}

# Computing variance of AB

abValues <- estimatedCoeffs2[3,]
abTotal <- 0

for (i in 1:r){
  error <- (abValues[i]-trueCoeffs[3])^2
  abTotal <- abTotal + error
}

# Computing the gross variance

GV <- (1/3)*((1/r)*aTotal + (1/r)*cTotal + (1/r) * abTotal)
print(GV)
```

Multiple imputation using the quick and dirty method

The R code for multiple imputation using the quick and dirty method for right censoring at 2.0 and $\sigma = 0.3$.

```
A  <- c(-1, 1,-1, 1,-1, 1,-1, 1)
C  <- c(-1,-1,-1,-1, 1, 1, 1, 1)
AB <- c( 1,-1,-1, 1, 1,-1,-1, 1)

y <- c(-1,-1,-3,1,1,1,-1,3)
n <- 100
m <- 5
trueCoeffs <- c(1,1,1)

lengthY <- length(y)
```

```
right <-   2.0                          # Right censoring value
left  <- - Inf                          # Left censoring value

estimatedCoeffs <- matrix(0,3,n)
estimatedRight <- matrix(0,1,m)
estimatedLeft <- matrix(0,1,m)

##########################################################

for (j in 1:n){

##########################################################

yVector <- vector()
errorvec <- vector()
eps <- vector()

for (i in 1:lengthY){

  v <- runif(1)

  alpha <- 3.33                             # To obtain sd
  eps[i] <- log(-log(1-v))                  # Epsilon
  sd <- 1/alpha                             # Standard error

  errorvec[i] <- eps[i] * sd                # The error vector

  yVector[i] <- y[i] + errorvec[i]

}

#########################################################

# Checking if censored and assigning values
# Right censored = 0, event=1, left censored=2, interval censored=3

censored <- vector()

for (i in 1:lengthY){
  if (yVector[i] > right){
    yVector[i] <- right
    censored[i] <- 0
  }
   else if(yVector[i] < left){
    yVector[i] <- left
    censored[i] <- 2
  }
  else
    censored[i] <- 1
}

#########################################################

lmQD <- lm(yVector~A+C+AB)              # Fitting linear model
```

```
coeffs <- lmQD$coef
a <- coeffs[2]; c <- coeffs[3]; ab <- coeffs[4]

mu <- coeffs[1] + A*a + C*c + AB*ab

sigma <- summary(lmQD)$sigma

############   Imputation starts   ##################

# Right censored

distR <- vector()
distInf <- vector()

for (i in 1:lengthY){
  distR[i] <- 1 - pgumbel(-right,-mu[i], sigma)
  distInf[i] <- 1 - pgumbel(-Inf, -mu[i],sigma)
}

for (i in 1:lengthY){
  if(censored[i]==0){
    for (s in 1:m){
      Uright <- runif(1)
      Vright <- distR[i] + (distInf[i] - distR[i])*Uright
      Yright <- mu[i] + sigma * log(-log(1-Vright))
      estimatedRight[,s] <- t(Yright)
    }
    if(censored[i] == 0){
      yVector[i] <- mean(estimatedRight)    # Imputing if censored
    }
  }

# Left censored

distL <- vector()
distminusInf <- vector()

for (i in 1:lengthY){
  distL[i] <- 1 - pgumbel(-left,-mu[i], sigma)
  distminusInf[i] <- 1- pgumbel(Inf, -mu[i], sigma)
}

for ( i in 1:lengthY){
  if (censored[i]==2){
    for (s in 1:m){
      Uleft <- runif(1)
      Vleft <- distminusInf[i] + (distL[i]- distminusInf[i])*Uleft
      Yleft <- mu[i] + sigma * log(-log(1-Vleft))
      estimatedLeft[,s] <- t(Yleft)
    }
    if(censored[i] == 2){
      yVector[i] <- mean(estimatedLeft)
    }
  }
 }
}
```

```
#####################################################

# Computing regression coefficients with multiple imputation
# replacing censored values.

lmQD2 <- lm(yVector~A+C+AB)           # Fitting the model
coeffs2 <- lmQD2$coef
a <- coeffs2[2]; c <- coeffs2[3]; ab <- coeffs2[4]

values <- c(a,c,ab)

# Saving the estimated coefficients
estimatedCoeffs[,j] <- t(values)

}         # j loop ends

###############   GROSS VARIANCE   ####################

# Computing variance of A

aValues <- estimatedCoeffs[1,]
aTotal <- 0

for (i in 1:n){
  error <- (aValues[i]-trueCoeffs[1])^2
  aTotal <- aTotal + error
}

# Computing variance of C

cValues <- estimatedCoeffs[2,]
cTotal <- 0

for (i in 1:n){
  error <- (cValues[i]-trueCoeffs[2])^2
  cTotal <- cTotal + error
}

# Computing variance of AB

abValues <- estimatedCoeffs[3,]
abTotal <- 0

for (i in 1:n){
  error <- (abValues[i]-trueCoeffs[3])^2
  abTotal <- abTotal + error
}

# Computing the gross variance

GV <- (1/3)*((1/n)*aTotal + (1/n)*cTotal + (1/n) * abTotal)
print(GV)
```

Multiple imputation using maximum likelihood estimation

The R code for multiple imputation using maximum likelihood estimation for right censoring at 2.0 and $\sigma = 0.3$.

```
A  <- c(-1, 1,-1, 1,-1, 1,-1, 1)
C  <- c(-1,-1,-1,-1, 1, 1, 1, 1)
AB <- c( 1,-1,-1, 1, 1,-1,-1, 1)

y <- c(-1,-1,-3,1,1,1,-1,3)
n <- 100
m <- 5
trueCoeffs <- c(1,1,1)

lengthY <- length(y)

right <-  2.0                          # Right censoring value
left  <- - Inf                         # Left censoring value
sdupper <- 0.5095169
sdlower <- 0.02353152

estimatedCoeffs <- matrix(0,3,n)
estimatedRight <- matrix(0,1,m)
estimatedLeft <- matrix(0,1,m)

#########################################################

for (j in 1:n){

#########################################################

yVector <- vector()
errorvec <- vector()
eps <- vector()

for (i in 1:lengthY){

  v <- runif(1)

  alpha <- 3.33                                # To obtain sd
  eps[i] <- log(-log(1-v))                     # Epsilon
  sd <- 1/alpha                                # Standard error

  errorvec[i] <- eps[i] * sd                   # The error vector

  yVector[i] <- y[i] + errorvec[i]

}

#########################################################

# Checking if censored and assigning values
# Right censored = 0, event=1, left censored=2, interval censored=3

censored <- vector()

for (i in 1:lengthY){
```

```
  if( yVector[i] > right ){
    censored[i] <- 0
  }
  else if (yVector[i] < left){
    censored[i] <- 2
  }
  else{
    censored[i] <- 1
  }
}

############################################################

x <- exp(yVector)                  # survreg using log(y) when computing.

m1 <- lm(yVector~A+C+AB)

ysurv <- survreg(formula=Surv(x,censored)~A+C+AB,
         dist="weibull", init=coef(m1))

coeffs <- ysurv$coef
a <- coeffs[2]; c <- coeffs[3]; ab <- coeffs[4]

mu <- coeffs[1] + A*a + C*c + AB*ab

alpha1 <- ( 1/ysurv$scale)              # Shape parameter
theta1 <- (exp(coeffs[1]))              # Scale parameter
sigma <- 1/alpha1

if(all(is.na(coeffs)==FALSE)){
  stderror <- summary(ysurv)$table[,2]

  if(stderror[2] < sdupper && stderror[2] > sdlower &&
     stderror[3] < sdupper && stderror[3] > sdlower &&
     stderror[4] < sdupper && stderror[4] > sdlower &&
     is.na(stderror[2:4])==FALSE){

##############   Imputation starts   ###############

# Right censored

distR <- vector()
distInf <- vector()

for (i in 1:lengthY){
  distR[i] <- 1 - pgumbel(-right,-mu[i],sigma)
  distInf[i] <- 1-pgumbel(-Inf, -mu[i], sigma)
}

for (i in 1:lengthY){
  if(censored[i]==0){
    for (s in 1:m){
      Uright <- runif(1)
      Vright <- distR[i] + (distInf[i] -distR[i])*Uright
      Yright <- mu[i] + sigma * log(-log(1-Vright))
      estimatedRight[,s] <- t(Yright)
```

```
    }
    if(censored[i] == 0){
      yVector[i] <- mean(estimatedRight)
    }
  }
}
# Left censored

distL <- vector()
distminusInf <- vector()

for (i in 1:lengthY){
  distL[i] <- 1 - pgumbel(-left,-mu[i],sigma)
  distminusInf[i] <- 1- pgumbel(Inf, -mu[i], sigma)
}

for ( i in 1:lengthY){
  if (censored[i]==2){
    for (s in 1:m){
      Uleft <- runif(1)
      Vleft <- distminusInf[i] + (distL[i] - distminusInf[i])*Uleft
      Yleft <- mu[i] + sigma * log(-log(1-Vleft))
      estimatedLeft[,s] <- t(Yleft)
    }
    if(censored[i] == 2){
      yVector[i] <- mean(estimatedLeft)
    }
  }
}

# Only repeating for suitable values of y
  if(any(is.na(yVector))==FALSE && any(is.infinite(yVector))==FALSE){

########################################################

# Computing regression coefficients with multiple imputation
# replacing censored values.

lmMIml <- lm(yVector~A+C+AB)          # Fitting the linear model

coeffs2 <- lmMIml$coef
a <- coeffs2[2]; c <- coeffs2[3]; ab <- coeffs2[4]

values <- c(a,c,ab)

estimatedCoeffs[,j] <- t(values)    # Saving the estimated effects

   }
  }
 }
}

########################################################

# Deleting all elements not suitable
```

```
estimatedCoeffs2 <- estimatedCoeffs[,colSums(estimatedCoeffs ==0)==0]

r <- ncol(estimatedCoeffs2)       # New number of acceptable runs

###############   GROSS VARIANCE    ##################

# Computing variance of A

aValues <- estimatedCoeffs2[1,]
aTotal <- 0

for (i in 1:r){
  error <- (aValues[i]-trueCoeffs[1])^2
  aTotal <- aTotal + error
}

# Computing variance of C

cValues <- estimatedCoeffs2[2,]
cTotal <- 0

for (i in 1:r){
  error <- (cValues[i]-trueCoeffs[2])^2
  cTotal <- cTotal + error
}

# Computing variance of AB

abValues <- estimatedCoeffs2[3,]
abTotal <- 0

for (i in 1:r){
  error <- (abValues[i]-trueCoeffs[3])^2
  abTotal <- abTotal + error
}

# Computing the gross variance

GV <- (1/3)*((1/r)*aTotal + (1/r)*cTotal + (1/r) * abTotal)
print(GV)
```