Improved approximate Bayesian computation methods for censored and uncensored data

Tatiana Dmitrieva

Follow this and additional works at: https://huskiecommons.lib.niu.edu/allgraduate-thesesdissertations

Recommended Citation
Dmitrieva, Tatiana, "Improved approximate Bayesian computation methods for censored and uncensored data" (2018). Graduate Research Theses & Dissertations. 3512.
https://huskiecommons.lib.niu.edu/allgraduate-thesesdissertations/3512

This Dissertation/Thesis is brought to you for free and open access by the Graduate Research & Artistry at Huskie Commons. It has been accepted for inclusion in Graduate Research Theses & Dissertations by an authorized administrator of Huskie Commons. For more information, please contact jschumacher@niu.edu.
ABSTRACT

IMPROVED APPROXIMATE BAYESIAN COMPUTATION METHODS FOR CENSORED AND UNCENSORED DATA

Tatiana Dmitrieva, Ph.D.
Department of Mathematical Sciences
Northern Illinois University, 2018
Nader Ebrahimi, Director

Approximate Bayesian Computation (ABC) is a method of statistical inference that does not require the exact model to be known. It is generally used for complex models, where the likelihood function is intractable or computationally difficult. ABC has lately become very useful in areas such as genetics, biology, ecology, and epidemiology, due to the complex structures arising in these fields.

Empirical likelihood is a nonparametric estimation approach in statistics that does not require the choice of a known distribution family for the data. Common ABC methods require the choice of summary statistic, which is very difficult to find, distance metric, and a tolerance level. Empirical likelihood allows these choices to be avoided. In this dissertation, two improved methods of the ABC algorithms via empirical likelihood for uncensored data are proposed. In simulations, it is shown that the accuracy of estimation of improved procedures is higher than of existing ABC via empirical likelihood.

Furthermore, for right censored or partially observed data three new algorithms are developed: ABC via empirical likelihood, ABC via comparing survival functions, and ABC via comparing smoothed distribution functions. Consistency, asymptotically unbiasedness, and asymptotic behavior of approximate posterior distribution are established for ABC via comparing survival functions and ABC via comparing smoothed distribution functions. The last two algorithms were implemented on real data.
IMPROVED APPROXIMATE BAYESIAN COMPUTATION METHODS FOR CENSORED AND
UNCENSORED DATA

BY

TATIANA DMITRIEVA
© 2018 Tatiana Dmitrieva

A DISSERTATION SUBMITTED TO THE GRADUATE SCHOOL
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE
DOCTOR OF PHILOSOPHY

DEPARTMENT OF MATHEMATICAL SCIENCES

Dissertation Director:
Nader Ebrahimi
ACKNOWLEDGEMENTS

I wish to express my profound gratitude and a sincere thank you to my research advisor Dr. Nader Ebrahimi, for his valuable guidance, constant mentoring, motivation, and encouragement at all stages of my work.

I am greatly indebted to Dr. Kristin McCullough from Grand View University for valuable suggestions, assistance, and support.

At this moment of my achievement, I would also like to thank the Department of Mathematical Sciences and Division of Statistics, especially Dr. Alan Polansky, Dr. Duchwan Ryu, Dr. Michelle Xia, Dr. Haiming Zhou, Dr. Barbara Gonzalez, and Dr. Sanjib Basu for the inspiration and the effort that was put into teaching me. I would also like to acknowledge the guidance of Michael B. Lagutin, Dr.Sc. Andrey M. Zubkov, Dr. Michael V. Kozlov, Dr. Boris M. Gurevich, Dr.Sc. Andrey M. Raigorodski, and Dr. Galina M. Fateeva from Moscow State University.

In addition, I wish to express gratefulness to Svetlana S. Kuzmina and Dr. Nina A. Raschepkina who taught me mathematics.

Last but not the least, I would like to thank my father Dr.Sc. Valeri Genin, for always believing in me and constantly encouragement me, and the rest of my family members for their prayers and moral support.
DEDICATION

To my parents Valeri Genin and Irina Genina, my husband Peter Dmitriev, and my children Ekaterina,
Andrey, and Eliza.
# TABLE OF CONTENTS

LIST OF TABLES .......................................................... vii
LIST OF FIGURES .......................................................... ix

INTRODUCTION .................................................................. 1

1 BAYESIAN ANALYSIS ...................................................... 3
  1.1 Bayesian Inference ...................................................... 3
  1.2 Prior and Posterior Distributions .................................. 4
  1.3 Bayesian Intractability ................................................. 5

2 ABC METHODS ............................................................... 7
  2.1 ABC Rejection Methods .............................................. 8
  2.2 ABC MCMC Methods ................................................ 10
    2.2.1 MCMC Method ...................................................... 11
    2.2.2 Likelihood-free MCMC Sampler ................................ 16
  2.3 ABC Sequential Methods ............................................ 20
    2.3.1 Importance Sampling ............................................. 20
    2.3.2 Sequential Importance Sampling ............................. 22
    2.3.3 Sequential ABC ..................................................... 27
  2.4 Semi-automatic ABC ................................................ 34

3 EMPIRICAL LIKELIHOOD .................................................. 37
  3.1 Nonparametric Maximum Likelihood ............................ 37
  3.2 Nonparametric Likelihood Ratio ................................... 39
  3.3 Estimating Equations .................................................. 42
  3.4 Computation of Empirical Likelihood ............................ 43
  3.5 Accuracy and Power of Empirical Likelihood .................. 46
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.6 Empirical Likelihood with Inequality Constraint</td>
<td>48</td>
</tr>
<tr>
<td>3.7 Empirical Likelihood for Censored Data</td>
<td>51</td>
</tr>
<tr>
<td>4 ABC WITH EMPIRICAL LIKELIHOOD</td>
<td>57</td>
</tr>
<tr>
<td>4.1 Bayesian Inference and Empirical Likelihood</td>
<td>57</td>
</tr>
<tr>
<td>4.2 ABC via Empirical Likelihood Sampler</td>
<td>59</td>
</tr>
<tr>
<td>4.3 ABC via Empirical Likelihood with Resampling</td>
<td>60</td>
</tr>
<tr>
<td>4.4 ABC via Empirical Likelihood with Testing</td>
<td>61</td>
</tr>
<tr>
<td>4.5 Comparison of ABC - EL algorithms. Examples</td>
<td>63</td>
</tr>
<tr>
<td>4.5.1 Normal Model</td>
<td>63</td>
</tr>
<tr>
<td>4.5.2 Exponential Model</td>
<td>66</td>
</tr>
<tr>
<td>4.5.3 Hydrogen Nanosensor</td>
<td>69</td>
</tr>
<tr>
<td>4.6 Observed Properties of ABC - EL with Testing</td>
<td>75</td>
</tr>
<tr>
<td>4.7 ABC via Empirical Likelihood with Different Prior Distributions</td>
<td>78</td>
</tr>
<tr>
<td>5 COMBINED ABC ALGORITHMS</td>
<td>81</td>
</tr>
<tr>
<td>5.1 Comparison of ABC Algorithms</td>
<td>81</td>
</tr>
<tr>
<td>5.2 Combined ABC Methods</td>
<td>84</td>
</tr>
<tr>
<td>5.3 Comparison of ABC mix and ABC-EL with testing</td>
<td>90</td>
</tr>
<tr>
<td>6 ABC ALGORITHMS FOR CENSORED DATA</td>
<td>93</td>
</tr>
<tr>
<td>6.1 ABC Rejection Algorithm for Right Censored Data</td>
<td>93</td>
</tr>
<tr>
<td>6.1.1 Examples</td>
<td>96</td>
</tr>
<tr>
<td>6.2 ABC-EL for Censored Data</td>
<td>98</td>
</tr>
<tr>
<td>6.2.1 EL weights by Sequential Quadratic Programming</td>
<td>99</td>
</tr>
<tr>
<td>6.2.2 EL weights by EM algorithm</td>
<td>100</td>
</tr>
<tr>
<td>6.2.3 Algorithm</td>
<td>101</td>
</tr>
<tr>
<td>6.2.4 Examples</td>
<td>102</td>
</tr>
<tr>
<td>6.3 ABC Algorithm for Censored Data via Comparing Survival Functions</td>
<td>105</td>
</tr>
<tr>
<td>Chapter</td>
<td>Section</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>6.3.1</td>
<td>The Two-stage Hypothesis Test</td>
</tr>
<tr>
<td>6.3.2</td>
<td>Algorithm</td>
</tr>
<tr>
<td>6.3.3</td>
<td>Asymptotically Unbiased Estimator</td>
</tr>
<tr>
<td>6.3.4</td>
<td>Consistency</td>
</tr>
<tr>
<td>6.3.5</td>
<td>Asymptotic behavior of Approximate Posterior Distribution</td>
</tr>
<tr>
<td>6.3.6</td>
<td>Examples</td>
</tr>
<tr>
<td>6.4</td>
<td>ABC Algorithm for Censored Data via Comparing Smoothed Distribution Functions</td>
</tr>
<tr>
<td>6.4.1</td>
<td>Logspline Smoothing</td>
</tr>
<tr>
<td>6.4.2</td>
<td>Cramer Test</td>
</tr>
<tr>
<td>6.4.3</td>
<td>Algorithm</td>
</tr>
<tr>
<td>6.4.4</td>
<td>Asymptotically Unbiased Estimator</td>
</tr>
<tr>
<td>6.4.5</td>
<td>Consistency</td>
</tr>
<tr>
<td>6.4.6</td>
<td>Asymptotic Behavior of Approximate Posterior Distribution</td>
</tr>
<tr>
<td>6.4.7</td>
<td>Examples</td>
</tr>
<tr>
<td>6.5</td>
<td>Compare ABC Algorithms for Censored Data, Examples</td>
</tr>
<tr>
<td>7</td>
<td>REAL DATA EXAMPLE</td>
</tr>
<tr>
<td>7.1</td>
<td>Simulation program</td>
</tr>
<tr>
<td>7.2</td>
<td>Prior Distributions</td>
</tr>
<tr>
<td>7.3</td>
<td>Concentration Estimation with ABC via Comparing Survival Functions</td>
</tr>
<tr>
<td>7.4</td>
<td>Concentration Estimation with ABC via Comparing Smoothed Distribution Functions</td>
</tr>
<tr>
<td>8</td>
<td>CONCLUSION, DISCUSSION, AND FUTURE WORK</td>
</tr>
<tr>
<td>8.1</td>
<td>Conclusion and Contributions</td>
</tr>
<tr>
<td>8.2</td>
<td>Future Work</td>
</tr>
<tr>
<td></td>
<td>REFERENCES</td>
</tr>
</tbody>
</table>
### LIST OF TABLES

4.1 Summary for three different ABC - EL methods for Normal model example. 66
4.2 Summary for three different ABC - EL methods for exponential model example. 69
4.3 Summary for three different ABC - EL methods for $2 \times 7$ square lattice hydrogen nanosensor model. 74
4.4 Summary for three different ABC - EL methods for $4 \times 4$ square lattice hydrogen nanosensor model. 75
4.5 Mean squared error (MSE), bias for ABC - EL with testing method for three different models. 76
4.6 Mean squared error (MSE), bias for ABC - EL with testing method for different number of iterations. 77
4.7 Mean squared error (MSE), bias for ABC - EL with testing method for different prior distributions. 78
4.8 MSE and bias for normal, exponential, and nanosensor models using ABC - EL, ABC-EL with resampling, ABC-EL with testing methods and different prior distributions. 79

5.1 Summary for normal model example using ABC-REJ, ABC-MCMC, ABC-PMC, ABC-EL with testing methods. 82
5.2 Summary for normal mixture model example using ABC-REJ, ABC-MCMC, ABC-PMC, ABC-EL with testing methods. 82
5.3 Summary for exponential model example using ABC-REJ, ABC-MCMC, ABC-PMC, ABC-EL with testing methods. 83
5.4 Summary for $2 \times 7$ square lattice nanosensor model example using ABC-REJ, ABC-MCMC, ABC-PMC, ABC-EL with testing methods. 84
5.5 Names for combined ABC algorithms. 85
5.6 Summary for normal model using combinations of ABC techniques. 86
5.7 Summary for exponential model using combinations of ABC techniques. 86
5.8 Summary for the $2 \times 7$ square lattice nanosensor using combinations of ABC techniques. 87
<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.9</td>
<td>88</td>
</tr>
<tr>
<td>5.10</td>
<td>90</td>
</tr>
<tr>
<td>5.11</td>
<td>91</td>
</tr>
<tr>
<td>5.12</td>
<td>91</td>
</tr>
<tr>
<td>5.13</td>
<td>92</td>
</tr>
<tr>
<td>6.1</td>
<td>97</td>
</tr>
<tr>
<td>6.2</td>
<td>98</td>
</tr>
<tr>
<td>6.3</td>
<td>103</td>
</tr>
<tr>
<td>6.4</td>
<td>104</td>
</tr>
<tr>
<td>6.5</td>
<td>104</td>
</tr>
<tr>
<td>6.6</td>
<td>105</td>
</tr>
<tr>
<td>6.7</td>
<td>115</td>
</tr>
<tr>
<td>6.8</td>
<td>116</td>
</tr>
<tr>
<td>6.9</td>
<td>126</td>
</tr>
<tr>
<td>6.10</td>
<td>127</td>
</tr>
<tr>
<td>6.11</td>
<td>129</td>
</tr>
<tr>
<td>6.12</td>
<td>130</td>
</tr>
<tr>
<td>6.13</td>
<td>131</td>
</tr>
<tr>
<td>6.14</td>
<td>132</td>
</tr>
<tr>
<td>6.15</td>
<td>132</td>
</tr>
<tr>
<td>6.16</td>
<td>132</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Top-view scanning electron microscopy image of a network of Pd nanowires with a deposition thickness of 7 nanometers.</td>
<td>69</td>
</tr>
<tr>
<td>4.2</td>
<td>$2 \times 5$ square lattice model of the nanosensor</td>
<td>70</td>
</tr>
<tr>
<td>4.3</td>
<td>$4 \times 4$ square lattice model of the nanosensor</td>
<td>71</td>
</tr>
<tr>
<td>4.4</td>
<td>Comparison of MSEs for three ABC-EL methods</td>
<td>75</td>
</tr>
<tr>
<td>4.5</td>
<td>Comparison of MSEs obtained by different ABC via empirical likelihood methods with various priors for the nanosensor model example.</td>
<td>80</td>
</tr>
<tr>
<td>5.1</td>
<td>Comparison of MSEs for ABC-MCMC+EL, ABC-MCMC+EL with resampling, ABC-MCMC+EL with testing on exponential model example.</td>
<td>89</td>
</tr>
<tr>
<td>5.2</td>
<td>Comparison of MSEs for ABC-REJ+EL with testing, ABC-MCMC+EL with testing, ABC-PMC+EL with testing on exponential model example.</td>
<td>89</td>
</tr>
<tr>
<td>5.3</td>
<td>Comparison of MSEs for ABC-PMC, ABC mix, and ABC-EL with testing on exponential model example.</td>
<td>92</td>
</tr>
<tr>
<td>6.1</td>
<td>Comparison of algorithm’s posterior MSEs for the normal model.</td>
<td>130</td>
</tr>
<tr>
<td>6.2</td>
<td>Comparison of algorithm’s posterior MSEs for the exponential model.</td>
<td>131</td>
</tr>
<tr>
<td>7.1</td>
<td>The experimental setup.</td>
<td>133</td>
</tr>
<tr>
<td>7.2</td>
<td>The example of particle trajectories.</td>
<td>134</td>
</tr>
<tr>
<td>7.3</td>
<td>Example of the data.</td>
<td>135</td>
</tr>
</tbody>
</table>
INTRODUCTION

Often, in Bayesian inference and prediction we face problems of not being able to obtain the likelihood function or not knowing it at all. The Approximate Bayesian Computation (ABC) methods have been created to solve these problems. ABC approach is very useful in areas such as genetics, biology, ecology, and epidemiology, due to the complex structures arising in these fields.

Using ABC, we sample from the posterior distribution and do not need to know the likelihood function. That is why ABC methods are also called likelihood-free methods. Unfortunately, it is not possible to calculate the exact posterior distribution and we should use an approximation of it. In this dissertation, different ABC methods have been studied, advantages and disadvantages of each one are described. Also, the theory of empirical likelihood as a nonparametric approximation of the parametric likelihood is introduced. The newly proposed methods of implementation of empirical likelihood theory into the Approximate Bayesian Computations are discussed.

The overall goal of the research is developing improved ABC methods to estimate the true parameter value for the cases when the likelihood function is not known or difficult to obtain, including the situations when the observed sample contains right censored data. This dissertation is organized as follows.

First chapter is the overview of the Bayesian statistics basics. The concepts of the prior and the posterior distributions are reviewed. The classification of intractability according to Wilkinson[53] is provided. The Bayesian principals are considered here to find solutions for cases where either likelihood function is not available or it is complicated.

Second chapter is a brief history of Approximate Bayesian Computations, where basic and advanced ABC algorithms are described. Advantages and disadvantages of each one together with some techniques to improve the methods are considered. Issues with ABC computations and a motivation to develop more advance algorithms are discussed.

Chapter three describes basic principles of empirical likelihood theory. Definitions of nonparametric likelihood function and a profile empirical likelihood ratio function are introduced for censored and uncen-
sored data. The asymptotic property of profile empirical likelihood ratio, ways to construct the hypothesis test and confidence regions are stated. Guidance on computation process of empirical likelihood is given with equality and inequality constraint.

Chapter four is about how the empirical likelihood theory can be used in the Bayesian computations. Known methods are described, and a new improved algorithm is proposed. The idea for a new approach is to incorporate Chi Square test into the selection process of parameter candidates before assigning weights for them via the empirical likelihood calculations. Through simulations, it is shown that the new algorithm allows to improve the accuracy of estimation, and some properties of it are examined.

Chapter five is about the combination of ABC via empirical likelihood with other ABC methods. The best results obtained when some other ABC process implemented first, and then with some modification ABC via empirical likelihood is applied. Simulations have shown that such mix in some cases gives comparable and in others provides a better result of estimation than the one achieved by ABC via empirical likelihood or other ABC algorithms alone.

There are many ABC methods for uncensored data. Recently, ABC rejection algorithm modified for censored data have been proposed in the literature. This approach is described in chapter six along with new ABC techniques which are suitable for right censored data. One of them is ABC via empirical likelihood with a mean constraint. Others are ABC via comparing survival functions and ABC via comparing smoothed distribution functions. The last two algorithms are suitable for uncensored and right censored data. The properties of the last two methods have been presented in chapter six.

In chapter seven, implementation of ABC via comparing survival functions and ABC via comparing smoothed distribution functions on real data is described.

Chapter eight gives conclusion and discussion about future work.
CHAPTER 1
BAYESIAN ANALYSIS

This chapter reveals the basic principles in Bayesian statistic. It shows the weakness of some Bayesian methods and the motivation for the farther work.

1.1 Bayesian Inference

In statistics, we would like to draw conclusions about populations from the observed data. Let us consider the probability space \((\Omega, \mathcal{F}, P)\), where \(\Omega\) is the sample space and represents the population, \(\mathcal{F}\) is a sigma algebra of subsets of \(\Omega\), and \(P\) is the probability measure that satisfies Kolmogorov’s axioms. A random variable \(X\) is a measurable function on \(\Omega\) with values in \(\mathbb{R}\) defined on the probability space \((\Omega, \mathcal{F}, P)\). Sample data is a set of data collected from a procedure determined by a random variable \(X\) through parametric/nonparametric models.

In parametric modeling, we have the sample which comes from a known probability distribution function \(f(x|\theta)\), where \(\theta\) is the unknown parameter. The goal is to find out the information about parameter \(\theta\) based on the sample data (observed data). This is the inversion problem, which is solvable by the Bayes’ theorem.

The Bayes’ theorem tells that

\[
P(A|B) = \frac{P(B|A)P(A)}{P(B)},
\]

where A and B are the events from the probability space, \(P(A|B)\) and \(P(B|A)\) are the conditional probabilities, and assume that \(P(B)\) greater than zero, otherwise the equation is not well defined.

This theorem can be restated for continuous random variables using probability density functions. Suppose, \(X\) and \(Y\) are continuous random variables. \(f(x|y)\) is the conditional probability density function of \(X\)
given that $Y = y$, and $g(y)$ is the marginal probability density function of $Y$. Then the conditional density of $Y$ given that $X = x$ is

$$g(y|x) = \frac{f(x|y)g(y)}{\int f(x|y)g(y)dy}.$$  \hspace{1cm} (1.1)

Notice that the denominator is the marginal density function of $X$. Last equation provides with update on the variable $Y$ after observing the variable $X$.

### 1.2 Prior and Posterior Distributions

The parameter $\theta$ comes from the parameter space $\Theta$. The Bayesian idea is to view the parameter $\theta$ as a random variable with some density function $\pi$ defined on $\Theta$. Then, we can think of $\pi(\theta)$ as prior knowledge of the parameter; $\pi(\theta)$ is called the prior distribution.

Assume, observed data is from a distribution denoted by a density function $f(x|\theta)$. Using $\theta$ instead of $y$ and $\pi(\theta)$ instead of $g(y)$ in the equation 1.1, the Bayes’ theorem can be rewritten

$$\pi(\theta|x) = \frac{f(x|\theta)\pi(\theta)}{\int_\Theta f(x|\theta)\pi(\theta)d\theta}.$$  \hspace{1cm} (1.2)

As have been noticed before the Bayes’ theorem provides the update information on $\theta$ after observing the variable $X$. In other words, the function $\pi(\theta)$ ”improves” after knowing the sample data. $\pi(\theta|x)$ is called the posterior distribution of the parameter $\theta$.

The marginal distribution of the random variable $X$, usually denoted with $m(x)$, is

$$m(x) = \int_\Theta f(x|\theta)\pi(\theta)d\theta,$$  \hspace{1cm} (1.3)

which is exactly the denominator of equation 1.2 and doesn’t depend on $\theta$. Thus, we can rewrite it and say that it is proportional to the numerator - likelihood times prior distribution:

$$\pi(\theta|x) \propto \frac{f(x|\theta)\pi(\theta)}{m(x)} = f(x|\theta)\pi(\theta).$$  \hspace{1cm} (1.4)
The posterior distribution is the most important part in the Bayesian inference of the parameter $\theta$. It has been shown that once we know the posterior distribution, the sample data cannot give us any additional information. All summary statistics can be found directly from the posterior distribution.

1.3 Bayesian Intractability

Posterior distribution is very important for the Bayesian inference. Unfortunately, often it is very difficult or not possible to compute it. Especially, it is hard to evaluate the marginal distribution - the denominator in the equation 1.2. Such problems called intractability problems in Bayesian inference. Let us consider the classification of intractability problems according to Wilkinson [53].

The first category in this classification is the class of tractable posterior distributions. Usually, it is the case of conjugate prior, when the prior distribution belongs to the same family of distributions as the posterior. For example, consider $Y \sim \text{Bin}(n, p)$, prior $\text{Beta}(a, b)$, then the posterior is $\text{Beta}(a + y, b + n - y)$. Both prior and posterior distributions are from the Beta family. The disadvantage of the conjugate prior is that it may not correspond to the known prior information about the parameter.

The second category is the class when the posterior distribution is not analytically tractable. For example, consider $Y_i \sim \text{Bin}(n, p)$, $i = 1, \ldots, n$, and prior $p \sim N(\mu, \sigma^2)$. Then, from the equation 1.2 the posterior distribution is of the form

$$
\pi(p|y) = \frac{1}{m(y)} \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left[ \frac{- (p - \mu)^2}{2\sigma^2} \right] \prod_{i=1}^{n} \binom{n}{y_i} p^{y_i} (1-p)^{n-y_i},
$$

where the factor $m(y)$ is

$$
m(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{\Theta} \exp \left[ \frac{- (p - \mu)^2}{2\sigma^2} \right] \prod_{i=1}^{n} \binom{n}{y_i} p^{y_i} (1-p)^{n-y_i} dp.
$$

Notice, that the last integral is not possible to evaluated analytically. Monte Carlo methods can be used to approximate such integral.

The last category is the class with full intractable posterior distributions, when it is not possible to evaluate the likelihood or it is not known at all. Monte Carlo methods cannot be used in such situations.
This is the case where Approximate Bayesian Computation methods can be helpful, which will be presented in the next chapter.
CHAPTER 2
ABC METHODS

Bayesian Monte Carlo methods are very important and popular in statistical inference for complex models and in different areas of research. However, they are powerless when the likelihood functions are difficult to evaluate or not known. The solution for such cases is ABC inference techniques.

Originally, the ABC algorithm was proposed in 1997 in the work of Simon Tavare [48]. He considered inference from DNA sequence data. In 1999, Jonathan K. Pritchard [43] used ABC methods in his study about human Y chromosome. The name “Approximate Bayesian Computation” appeared in the article about problems in population genetics of Mark Beaumont [5] a little later. After that, ABC started to be used in many other areas besides genetics, such as biology, ecology, epidemiology, and nanotechnology. In 2015, Chkrebtii and his colleagues performed Approximate Bayesian Computations for inference on invasive species models with latent variables of unknown dimension [13]. Recently, ABC method was used to estimate sub-epidemic dynamics by Ibeh and Aris-Brosou [22]. Roding and his collaborators implemented ABC technique to estimate number concentrations of monodisperse nanoparticles in suspension by optical microscopy [45]. Mason in his work applied ABC for the occurrence and size of defects in Advanced Gas-cooled nuclear Reactor boilers calculations [34].

First ABC algorithms were based on the rejection methods. Marjoram [33] suggested the improvement of it based on Markov chain Monte Carlo methods. Sisson in [46] proposed the idea of sequential procedure in ABC, which was corrected in 2009 [47].

Later in this chapter, ABC algorithms will be presented with advantages and disadvantages of each one. Also, some techniques to improve the methods will be considered. Issues with ABC computations and a motivation to develop a more advance algorithm will be discussed.

Notations for this chapter are as follows: (1) $X$ is a random variable, takes values in finite set $A$ with distribution function $f(x|\theta)$, $x$ is a sample data, $\theta$ is the parameter, which comes from a parameter space $\Theta \subset \mathbb{R}^d$, (2) $\pi(\theta)$ is the prior distribution, which determines our prior knowledge of the parameter $\theta$. The goal is to get the sample values from the posterior distribution $\pi(\theta|x)$ or from its approximation when it is not known exactly. Note that the notation of $f(x|\theta)$ is used for likelihood function or for the model, from which the sample data is generated.

## 2.1 ABC Rejection Methods

The first likelihood-free Bayesian computation algorithm based of the rejection sampling can be written as follows.

<table>
<thead>
<tr>
<th>Rejection Sampler (ABC - REJ):</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Generate a parameter candidate $\theta^*$ from the prior $\pi(\theta)$.</td>
</tr>
<tr>
<td>2. Generate a data set $x^* \sim f(x</td>
</tr>
<tr>
<td>3. Accept $\theta^<em>$ if $x^</em> = x$.</td>
</tr>
</tbody>
</table>

Here $f(x|\theta)$ stands for the model, from which the data is simulated. Each accepted $\theta^*$ denoted as $\theta_i^*$ is an independent draw from $\pi(\theta|x)$ because

$$f(\theta_i^*) \propto \sum_{x^* \in \mathcal{F}} f(x^*|\theta_i^*) \pi(\theta_i^*) \mathbb{1}_x(x^*) = f(x|\theta_i^*) \pi(\theta_i^*) \propto \pi(\theta_i^*|x),$$

where $\mathbb{1}_x(x^*)$ is the indicator function and takes values of 1 when $x = x^*$, as shown in [2].

Acceptance rates for this algorithm can be very low. Candidate parameters are generated from the prior, which may deviate a lot from the posterior. Also, the acceptance criterion is very strict. Wilkinson [53]
shows that to get $n$ $\theta'$s the number of simulated data sets on average is
\[ n \frac{1 - P(\mathbb{1}_x(x^*))}{P(\mathbb{1}_x(x^*)))}. \]
This number can be large when $P(\mathbb{1}_x(x^*))$ is small.

ABC-REJ cannot be used for samples coming from continuous distribution due to the zero probability of exact match between observed ($x$) and simulated ($x^*$) data. This was the motivation to improve the algorithm by relaxing the acceptance criterion.

Instead of accepting $\theta$ when the exact match between observed and simulated data happens, accept $\theta$ when the observed and simulated data “close” to each other. To determine “close” the metric $\rho(\cdot, \cdot) : \mathcal{F} \times \mathcal{F} \rightarrow \mathbb{R}^+ = [0, \infty)$ and the tolerance $\varepsilon$ may be used. The algorithm can be written as follows.

### Approximate Rejection Sampler (ABC-REJ 1):

1. Generate a parameter candidate $\theta^*$ from the prior $\pi(\theta)$.
2. Generate a data set $x^* \sim f(x|\theta^*)$.
3. Accept $\theta^*$ if $\rho(x^*, x) < \varepsilon$.

Each accepted $\theta^*$ denoted as $\theta_i^*$ is an independent draw from $f(\theta|\rho(x^*, x) < \varepsilon)$. This is not an exact algorithm, the $\theta_i^*$ values come from approximate posterior distribution, which depends on the tolerance level $\varepsilon$.

\[
f(\theta_i^*) \propto \sum_{x^* \in \mathcal{F}} f(x^*|\theta_i^*) \pi(\theta_i^*) 1_{\rho(x^*, x) < \varepsilon} = \sum_{x^*: \rho(x^*, x) < \varepsilon} f(x^*|\theta_i^*) \pi(\theta_i^*) \approx \pi_{\varepsilon}(\theta_i^*|x),
\]

where $1_{\rho(x^*, x) < \varepsilon}$ is the indicator function and takes values of 1 when $\rho(x, x^* < \varepsilon$ is true.

Notice, that when $\varepsilon$ approaches zero, algorithm ABC-REJ 1 is the same as algorithm ABC-REJ. If $\varepsilon$ gets closer to $\infty$, we accept almost every $\theta$, thus, the distribution of resulting $\theta$’s doesn’t change much from the prior distribution $\pi(\theta)$.

ABC-REJ 1 algorithm is the improvement of the previous one, but still has some problems. The acceptance criterion of this algorithm won’t work as the one works with high-dimensional data. One of the solutions could be to choose the summary statistic $S(x)$. Thus, the algorithm ABC-REJ 1 can be rewritten as follows.
Approximate Rejection Sampler (ABC-REJ 2):

1. Generate a parameter candidate $\theta^*$ from the prior $\pi(\theta)$.
2. Generate a data set $x^* \sim f(x|\theta^*)$.
3. Accept $\theta^*$ if $\rho(S(x^*),S(x)) < \varepsilon$.

The best choice of summary statistic would be the sufficient one. The statistic is the sufficient statistic for $\theta$ if and only if the posterior distribution of $\theta$ given the summary statistic is the same as the posterior distribution $\pi(\theta|S(x)) = \pi(\theta|x)$. Meaning that, if we know $S(x)$, we don’t need to know the full data set to get additional information about the parameter. If the sufficient statistic is known, the algorithm ABC-REJ 2 is the same as ABC-REJ 1. Unfortunately, when the posterior and the likelihood functions are both not available, it is very difficult to determine if the summary statistic is sufficient or not. The choice of the summary statistic will be discussed later.

Wilkinson in [54] argued that using insufficient summary statistics bring more uncertainty to the ABC algorithm and showed that ABC rejection gives exact inference for uniform or measurement error models, provides a distribution for the model error term given a choice of metric and tolerance level.

2.2 ABC MCMC Methods

ABC rejection algorithm may have a low acceptance rate since there is a possibility of using parameter values that are in the tails of the posterior distribution, in the regions of low probability. In 2003, Marjoram [33] suggested the improvement of ABC-REJ by using Markov chain Monte Carlo methods (MCMC), see [52] and [10].
2.2.1 MCMC Method

It is essential for understanding this method to recall Monte Carlo approach. It was created for evaluating complex integrals, for example, \( \int_a^b h(x) \, dx \). The idea is to represent a complex function \( h(x) \) into a product of some function \( g(x) \) and the density function \( f(x) \) defined over the interval \((a, b)\). Then,

\[
\int_a^b h(x) \, dx = \int_a^b g(x) \, f(x) \, dx = E[g(x)],
\]

so, the original integral rewritten as an expectation of \( g(x) \) over the density function \( f(x) \). Consider \( x_1, \ldots, x_n \), a sample from the density \( f(x) \), then

\[
\int_a^b h(x) \, dx = E[g(x)] \approx \frac{1}{n} \sum_{i=1}^{n} g(x_i).
\]

This is Monte Carlo integration and often used in Bayesian statistics for approximation of the posterior distributions.

The next important part for comprehension of MCMC method is to review Markov chains. Consider \( X_n, n = 0, 1, 2, \ldots, \) a discrete stochastic process. Assume, that this process can take finite or countable number of values, which called a state space. The process is a Markov chain if the transition probability \( p_{ij} \) from state \( i \) to state \( j \) is given by a formula:

\[
p_{ij} = Pr(X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, \ldots, X_0 = i_0) = Pr(X_{n+1} = j | X_n = i),
\]

for all states \( i, j, i_{n-1}, \ldots, i_0 \) and \( n = 0, 1, 2, \ldots \). So, the transition probability depends only on the random variable’s value in the current state. Usually, the transition probabilities for a Markov chain are represented as a matrix \( P \), whose \((i, j)^{th}\) entry equals \( p_{ij} \).

Let \( \pi^n_j \) denote the probability

\[
Pr(X_n = j) = \pi^n_j,
\]
then
\[ \pi^n = (\pi^n_0, \pi^n_1, \ldots). \]

If the following limit exist
\[ \lim_{n \to \infty} \pi^n = \pi, \]
then \( \pi \) called the stationary distribution.

Markov chain is called irreducible if all states communicate. State \( i \) and \( j \) communicate if there exist some \( n \geq 0 \) and \( m \geq 0 \) such that \( p^n_{ij} > 0 \) and \( p^m_{ji} > 0 \). State \( i \) is called periodic if \( p^n_{ii} > 0 \) implies that \( n \) is a multiple of some fixed integer greater than one. Markov chain is aperiodic if all the states are aperiodic.

The necessary conditions for existence of the unique stationary distribution are irreducible aperiodic Markov chain with \( \pi_j > 0 \) for all states \( j \). The sufficient condition for existence of the unique stationary distribution is that the
\[ p_{ij}\pi_i = p_{ji}\pi_j, \tag{2.1} \]
which refers to the detailed balance equation.

For using Monte Carlo integration, we need to know how to sample from the complex probability density function \( f(x) \). Metropolis-Hastings algorithm was designed to solve this type of the problem. In Bayesian statistic, it is often used for sampling from the posterior distribution \( \pi(\theta|x) \), where \( \theta \) is the parameter and \( x \) is the observed data. In this case, Metropolis-Hastings algorithm generates a sequence of \( \theta \)'s from \( \pi(\theta|x) \). It starts with some candidate value of \( \theta_1 \), then generates \( \theta^* \) from some proposed density \( q(\theta|\theta_1) \).

The next step is to calculate the ratio of posterior distributions multiplied by the proposed density at the corresponding values of \( \theta \),
\[ \frac{\pi(\theta^*|x)q(\theta_1|\theta^*)}{\pi(\theta_1|x)q(\theta^*|\theta_1)}, \]
where
\[ \pi(\theta|x) = \frac{\pi(\theta)f(x|\theta)}{m(x)}. \]
Then, the ratio can be rewritten

\[
\frac{\pi(\theta^*|x)q(\theta_1|\theta^*)}{\pi(\theta_1|x)q(\theta^*|\theta_1)} = \frac{f(x|\theta^*)\pi(\theta^*)q(\theta_1|\theta^*)}{f(x|\theta_1)\pi(\theta_1)q(\theta^*|\theta_1)}.
\]

The new candidate \( \theta^* \) is accepted with probability

\[
\alpha = \min \left\{ 1, \frac{f(x|\theta^*)\pi(\theta^*)q(\theta_1|\theta^*)}{f(x|\theta_1)\pi(\theta_1)q(\theta^*|\theta_1)} \right\}.
\]

If it is accepted, then \( \theta^* \) becomes \( \theta_2 \), otherwise \( \theta_2 \) will be the same as \( \theta_1 \), then the procedure repeats until the desirable number of parameters is collected. The algorithm can be written as follows.

**Metropolis-Hasting:**

1. Initialize \( \theta_1 \).
2. Set \( i = 1 \).
3. Generate a candidate value \( \theta^* \sim q(\theta|\theta_i) \), where \( q \) is some proposal density.
4. Set \( \theta_{i+1} = \theta^* \) w.p. \( \alpha = \min \left\{ 1, \frac{f(x|\theta^*)\pi(\theta^*)q(\theta_1|\theta^*)}{f(x|\theta_1)\pi(\theta_1)q(\theta^*|\theta_1)} \right\} \). otherwise set \( \theta_{i+1} = \theta_i \).
5. If \( i < N \), increment \( i = i + 1 \) and go to step 3.

The resulting sequence of \( \theta \)'s is a Markov chain with stationary distribution \( \pi(\theta|x) \). To show this it is sufficient to prove that the Metropolis-Hasting transition probability satisfy the detailed balance equation 2.1 with \( \pi(\theta|x) \). We sample from \( q(\theta|\theta_i) \) and accept the new value with probability \( \alpha \). Therefore, the transition
The probability is

\[ Pr(\theta_i \to \theta^*) = q(\theta^*|\theta_i) \cdot \alpha(\theta_i, \theta^*) = q(\theta^*|\theta_i) \cdot \min \left\{ \frac{\pi(\theta^*|x)q(\theta_i|\theta^*)}{\pi(\theta_i|x)q(\theta^*|\theta_i)}, 1 \right\}. \]

The detailed balance equation based on these notations is

\[ Pr(\theta_i \to \theta^*) \pi(\theta_i|x) = Pr(\theta^* \to \theta_i) \pi(\theta^*|x) \]

or

\[ q(\theta^*|\theta_i) \alpha(\theta_i, \theta^*) \pi(\theta_i|x) = q(\theta_i|\theta^*) \alpha(\theta^*, \theta_i) \pi(\theta^*|x). \]  
(2.2)

Let us consider three possible cases based on the probability \( \alpha \).

Case 1. \( \alpha = 1 \), which is \( \pi(\theta^*|x)q(\theta_i|\theta^*) = \pi(\theta_i|x)q(\theta^*|\theta_i) \).

Notice, that in this case we get the equation 2.2 right away. Hence, the detailed balance equation is satisfied.

Case 2. \( \pi(\theta^*|x)q(\theta_i|\theta^*) < \pi(\theta_i|x)q(\theta^*|\theta_i) \).

Which means

\[ \alpha(\theta_i, \theta^*) = \frac{\pi(\theta^*|x)q(\theta_i|\theta^*)}{\pi(\theta_i|x)q(\theta^*|\theta_i)} \]

and

\[ \alpha(\theta^*, \theta_i) = 1. \]

Here

\[ q(\theta^*|\theta_i) \alpha(\theta_i, \theta^*) \pi(\theta_i|x) = \]

\[ = q(\theta^*|\theta_i) \frac{\pi(\theta^*|x)q(\theta_i|\theta^*)}{\pi(\theta_i|x)q(\theta^*|\theta_i)} \pi(\theta_i|x) = \]

\[ = \pi(\theta^*|x) q(\theta_i|\theta^*) \]

\[ = q(\theta_i|\theta^*) \alpha(\theta^*, \theta_i) \pi(\theta^*|x). \]
Equation 2.2 is true. Thus, the detailed balance equation is satisfied.

Case 3. $\pi(\theta^*|x)q(\theta_i|\theta^*) > \pi(\theta_i|x)q(\theta^*|\theta_i)$.

Which means

$$\alpha(\theta_i, \theta^*) = 1$$

and

$$\alpha(\theta^*, \theta_i) = \frac{\pi(\theta_i|x)q(\theta^*|\theta_i)}{\pi(\theta^*|x)q(\theta_i|\theta^*)}.$$ 

Here

$$q(\theta_i|\theta^*) \alpha(\theta^*, \theta_i) \pi(\theta^*|x) =$$

$$= q(\theta_i|\theta^*) \frac{\pi(\theta_i|x)q(\theta^*|\theta_i)}{\pi(\theta^*|x)q(\theta_i|\theta^*)} \pi(\theta^*|x) =$$

$$= \pi(\theta_i|x) q(\theta^*|\theta_i)$$

$$= q(\theta^*|\theta_i) \alpha(\theta_i, \theta^*) \pi(\theta_i|x).$$

Equation 2.2 is true. Therefore, the detailed balance equation is satisfied.

There is a series of issues with Metropolis-Hastings algorithm. Some of them are how to choose the starting value, which proposal density to use, after how many runs the chain reaches stationarity, correlation between the members of the chain.

The solutions to these problems are discussed in Walsh [52]. For starting point it is recommended to use an approximate MLE value, near the distribution’s mode. This idea may not work if the distribution is multimodal; it is possible to get stuck near one of the modes. One of the suggestions is to use several starting points and get several different chains. Another approach is to use simulated annealing on one chain. The idea of this approach is initially to accept down-hill moves with some probability, and later to decrease it.

The choice of proposal distribution is also very important. The two general options are random walk and independent chain. Often, the standard deviation of the proposal distribution is used for increasing the acceptance probability of the chain. This need to be done carefully, too large standard deviation may lead
to big moves and low acceptance probability, autocorrelation problem appears. On the other hand, small standard deviation may cause high acceptance probability but the autocorrelation problem arises again.

What to do with correlation between members of the chain, which comes from the Metropolis-Hasting algorithm? There is a helpful result in time series analysis that is if our sample from a stationary and correlated distribution, we are able to get unbiased parameter estimates when the sample is large enough. For example, approximately twenty times as many sample values are needed for the same accuracy as with an uncorrelated chain, when the autocorrelation is 0.90. There is a thinning approach for decreasing an autocorrelation, that is to take every k-th value from the chain after some period.

It is a common practice with Metropolis-Hastings algorithm to get a very long sample (6000 or more), and drop the first values up to 5000 (in some cases may be even more), which called a burn-in period. The reason is we want our sequence to reach stationarity (settle down). The graph such as time series trace, autocorrelation as a function of the time lag, partial autocorrelation as a function of lag are helpful to determine the burn-in period. Also, the Geweke and Raftery-Lewis tests are useful to find out if additional burn-in is required.

2.2.2 Likelihood-free MCMC Sampler

Marjoram [33] showed a modified Metropolis-Hastings algorithm, which doesn’t require knowing the likelihood function and refers to likelihood-free MCMC algorithm. It can be written as follows.

**Likelihood-free MCMC Sampler:**

1. Initialize $\theta_1$.

2. Set $i = 1$. 
3. Generate a candidate value \( \theta^* \sim q(\theta|\theta_i) \), where \( q \) is some proposal density.

4. Generate a data set \( x^* \sim f(x|\theta^*) \).

5. Set \( \theta_{i+1} = \theta^* \) w.p. \( \alpha = \min\{1, \frac{\pi(\theta^*)q(\theta_i|\theta^*)}{\pi(\theta_i)q(\theta^*|\theta_i)} \mathbb{1}(x^* = x)\} \), otherwise set \( \theta_{i+1} = \theta_i \).

6. If \( i < N \), increment \( i = i + 1 \) and go to step 3.

Here \( \mathbb{1}(A) = 1 \), if \( A \) is true, and 0 otherwise.

Marjoram [33] proved that the resulting sequence of \( \theta^* \)'s is a Markov chain with stationary distribution \( \pi(\theta|x) \) using the detailed balance equation 2.1. The proof goes as follows. Assume, we sample \( \theta^* \) from \( q(\theta|\theta_i) \), and accept it with probability

\[
\alpha(\theta_i, \theta^*) = \min\left\{ 1, \frac{\pi(\theta^*)q(\theta_i|\theta^*)}{\pi(\theta_i)q(\theta^*|\theta_i)} \mathbb{1}(x^* = x) \right\}.
\]

Suppose,

\[
\frac{\pi(\theta^*)q(\theta_i|\theta^*)}{\pi(\theta_i)q(\theta^*|\theta_i)} \leq 1.
\]

Then, \( \alpha(\theta^*, \theta_i) = 1 \). Therefore, the transition probability is

\[
Pr(\theta_i \rightarrow \theta^*) = q(\theta^*|\theta_i) \alpha(\theta_i, \theta^*) f(x^*|\theta^*).
\]

Then, the detailed balance equation is

\[
\pi(\theta|x) \cdot Pr(\theta_i \rightarrow \theta^*) = \pi(\theta|x) \cdot q(\theta^*|\theta_i) \alpha(\theta_i, \theta^*) f(x^*|\theta^*).
\]

Plug in the defined \( \alpha \) with assumptions, and using the Bayes’ theorem rewrite posterior distribution,

\[
\pi(\theta|x) \cdot Pr(\theta_i \rightarrow \theta^*) = \frac{\pi(\theta_i)f(x|\theta_i)}{m(x)} \cdot q(\theta^*|\theta_i) \frac{\pi(\theta^*)}{\pi(\theta_i)} \frac{q(\theta_i|\theta^*)}{q(\theta^*|\theta_i)} \mathbb{1}(x^* = x) f(x^*|\theta^*).
\]
Simplifying and regrouping,

\[ \pi(\theta_i|x) \cdot Pr(\theta_i \rightarrow \theta^*) = \frac{\pi(\theta^*)f(x|\theta^*)}{m(x)} f(x|\theta_i) q(\theta_i|\theta^*) \mathbb{1}(x^* = x). \]

Using assumption that \( \alpha(\theta^*, \theta_i) = 1 \),

\[ \pi(\theta_i|x) \cdot Pr(\theta_i \rightarrow \theta^*) = \pi(\theta^*|x) \cdot Pr(\theta^* \rightarrow \theta_i). \]

Simplifying,

\[ \pi(\theta_i|x) \cdot Pr(\theta_i \rightarrow \theta^*) = \pi(\theta^*|x) \cdot Pr(\theta^* \rightarrow \theta_i). \]

This ends the proof.

The special techniques for the MCMC samplers to solve problems such as burn-in periods, autocorrelated output, choosing the starting point, and proposal density can be applied here as well.

Due to strict requirement of the exact match between the observed and simulated data the algorithm is not accepting many proposed parameters. The solution is the same as with the rejection ABC to relax the exact match and change it with

\[ \rho(x^*, x) < \varepsilon. \]

**Approximate MCMC Sampler (ABC - MCMC 1):**

1. Initialize \( \theta_1 \).

2. Set \( i = 1 \).

3. Generate a candidate value \( \theta^* \sim q(\theta|\theta_i) \), where \( q \) is some proposal density.

4. Generate a data set \( x^* \sim f(x|\theta^*) \).
5. Set $\theta_{i+1} = \theta^*$ w.p. $\alpha = \min\{1, \frac{\pi(\theta^*)q(\theta|\theta^*)}{\pi(\theta_i)q(\theta^*|\theta_i)} \mathbb{1}(\rho(x^*, x) < \varepsilon)\}$, otherwise set $\theta_{i+1} = \theta_i$.

6. If $i < N$, increment $i = i + 1$ and go to step 3.

Here $\mathbb{1}(A) = 1$, if $A$ is true, and 0 otherwise.

It is necessary when we work with high dimensional data to include a summary statistic into the calculations of the distance between the observed and simulated data, such as

$$\rho(S(x^*), S(x)) < \varepsilon.$$ 

Approximate MCMC Sampler (ABC - MCMC 2):

1. Initialize $\theta_1$.

2. Set $i = 1$.

3. Generate a candidate value $\theta^* \sim q(\theta|\theta_i)$, where $q$ is some proposal density.

4. Generate a data set $x^* \sim f(x|\theta^*)$.

5. Set $\theta_{i+1} = \theta^*$ w.p. $\alpha = \min\{1, \frac{\pi(\theta^*)q(\theta|\theta^*)}{\pi(\theta_i)q(\theta^*|\theta_i)} \mathbb{1}(\rho(S(x^*), S(x)) < \varepsilon)\}$, otherwise set $\theta_{i+1} = \theta_i$.

6. If $i < N$, increment $i = i + 1$ and go to step 3.

Here $\mathbb{1}(A) = 1$, if $A$ is true, and 0 otherwise.

Notice, that the resulting sequence of the parameters here is not from exact posterior distribution, but only from its approximation. MCMC methods use dependent observations, thus the algorithm spends more
time getting the parameter values from the regions of the posterior distribution with higher probability than
the ABC-REJ. The acceptance probability of the proposed parameters is higher using MCMC method.

2.3 ABC Sequential Methods

Several important disadvantages of MCMC methods such as difficulty in assessing when the Markov
chain reaches a stationary distribution, easiness of the algorithm to get stuck in the areas of local modes lead
us to develop a better method. One of the ideas is to consider a sequential Monte Carlo approach. Instead of
getting just one sample a sequence of samples is taken into an account with slowly decreasing tolerance level
\( \varepsilon \) from some large value \( \varepsilon_1 \) to some reasonably small \( \varepsilon_T \) to ensure that the approximate posterior distribution
is close enough to the exact one. Sequential methods are based on importance sampling techniques.

2.3.1 Importance Sampling

Importance sampling techniques were originally developed for integral approximation. Now it is a
useful sampling approach in Bayesian analysis, and foundation for sequential methods in Approximate
Bayesian Computations. The idea is to generate values from some well known distribution and “close”
to the desirable one compensating by an importance weights. As a result, we get a set of values with the
corresponding weights. The next step will be to resample from such set based on the weights.

Suppose, we have a complicated integral, \( \int_a^b h(x) dx \). The function \( h(x) \) can be represented as a product
of some function \( g(x) \) and the density function \( f(x) \) defined over the interval \( (a, b) \). Then,

\[
\int_a^b h(x) \, dx = \int_a^b g(x) \, f(x) \, dx = E_f[g(x)],
\]

so, the original integral rewritten as an expectation of \( g(x) \) over the density function \( f(x) \). Consider
\( x_1, \ldots, x_n \), a sample from the density \( f(x) \), then

\[
\int_a^b h(x) \, dx \approx \frac{1}{n} \sum_{i=1}^{n} g(x_i).
\]
What if it is difficult to sample from \( f(x) \)? The solution is to consider the density \( p(x) \), which roughly approximates (or “close” to) the density \( f(x) \) and easy to sample from. Then

\[
\int_a^b g(x) f(x) \, dx = \int_a^b g(x) \frac{f(x)}{p(x)} p(x) \, dx = E_p \left[ g(x) \frac{f(x)}{p(x)} \right].
\]

Consider \( x_1, \ldots, x_n \), a sample from the density \( p(x) \), then

\[
\int_a^b g(x) f(x) \, dx \simeq \frac{1}{n} \sum_{i=1}^n g(x_i) \frac{f(x_i)}{p(x_i)}.
\]

The importance sample formulation is

\[
\int_a^b g(x) f(x) \, dx \simeq \frac{\sum_{i=1}^n g(x_i) w_i}{\sum_{i=1}^n w_i},
\]

where

\[
w_i = \frac{f(x_i)}{p(x_i)}.
\]

The advantages of importance sampling are a distribution \( p(x) \), from which it is easy to sample, can be cheap; it may be done for any target distribution \( f(x) \); the good choice of \( p(x) \) can lead to big improvements compared to MCMC sampling; computationally easy; it can be used as a basis in sequential Monte Carlo method.

The disadvantages are that resampling needed; the bad choice of \( p(x) \) can lead to big losses in efficiency and even to infinite variance; doesn’t work well with high dimensions; the method may miss important regions of \( f(x) \).

The general importance ABC algorithm may be constructed as follows.

\[\text{Likelihood-free Importance Sampler (1)}:\]

1. Generate a parameter candidate \( \theta^* \) from the prior \( \pi(\theta) \).
2. Generate a data set \( x^* \sim f(x|\theta^*) \).
3. Compute \( w = g(\theta^*, x^*, x) \), where \( g \) is some function.
4. Take \( \theta^* \) with corresponding weight.

When the data is high dimensional it is better to use summary statistics, then the importance ABC algorithm may be rewritten as follows.

### Likelihood-free Importance Sampler (2):

1. Generate a parameter candidate \( \theta^* \) from the prior \( \pi(\theta) \).
2. Generate a data set \( x^* \sim f(x|\theta^*) \).
3. Compute \( w = g(\theta^*, S(x^*), S(x)) \), where \( g \) is some function.
4. Take \( \theta^* \) with corresponding weight.

Despite the simplicity of the method compare to MCMC approach, there is a serious problem with importance sampling of choosing the distribution “close” to the desirable one. In practice it is a difficult task, especially when we have a nonstandard high dimensional distribution. Importance sampling can be improved by involving it into the sequential methods, which are described in the next section.

### 2.3.2 Sequential Importance Sampling

The improvement of the importance sampling is the sequential importance sampling. Suppose we are interested in sampling from some distribution functions \( f_1, f_2, \ldots, f_n \) sequentially, which means first sample from \( f_1 \), then from \( f_2 \), and so on. Importance sampling technique is used at every sampling. Let \( p_1, p_2, \ldots, p_n \) be importance distributions for \( f_1, f_2, \ldots, f_n \) correspondingly. Then the weights are \( w^{(1)} = f_1/p_1, \ldots, w^{(n)} = f_n/p_n \).
Specifically, as explained by Doucet [16], at time 1 the goal is to estimate $f_1$ by drawing i.i.d. sample $x_j^{(1)}, j = 1, ..., N$ from $p_1(\cdot)$. Then,

$$
\hat{p}_1(x) = \frac{1}{N} \sum_{j=1}^{N} \delta(x_j^{(1)} - x),
$$

and

$$
\hat{f}_1(x) \sim \sum_{j=1}^{N} w_j^{(1)} \delta(x_j^{(1)} - x),
$$

where $\delta(\cdot)$ is Dirac delta function, and $w_j^{(1)} = f_1(x_j^{(1)})/p_1(x_j^{(1)})$.

At time 2, the goal is to approximate $f_2$. Importance distribution is $p_2(\cdot)$. The “sequential” idea is to reuse the sample values from the previous drawing $x_j^{(1)}, j = 1, ..., N$. We have

$$
p_2(x^{(1)}, x^{(2)}) = p_2(x^{(1)}) \cdot p_2(x^{(2)}|x^{(1)}).
$$

Assume, that $f_1(x^{(1)}) \approx f_2(x^{(1)})$ and $p_2(x^{(1)}) = p_1(x^{(1)})$, then

$$
p_2(x^{(1)}, x^{(2)}) = p_1(x^{(1)}) \cdot p_2(x^{(2)}|x^{(1)}).
$$

Now, we need to sample

$$
X_j^{(2)}|X_j^{(1)} \sim p_2(x^{(2)}|x^{(1)}), j = 1, ..., N.
$$

The weights will be calculated in the following fashion:

$$
w^{(2)} = w^{(2)}(x^{(1)}, x^{(2)}) = \frac{f_2(x^{(1)}, x^{(2)})}{p_2(x^{(1)}, x^{(2)})}.
$$

With the expanded denominator,

$$
w^{(2)} = w^{(2)}(x^{(1)}, x^{(2)}) = \frac{f_2(x^{(1)}, x^{(2)})}{p_1(x^{(1)}) p_2(x^{(2)}|x^{(1)})}.
$$

After multiplying the numerator and the denominator by $f_1(x^{(1)})$ and some rearrangements, we get:

$$
w^{(2)} = w^{(2)}(x^{(1)}, x^{(2)}) = \frac{f_1(x^{(1)}) f_2(x^{(1)}, x^{(2)})}{p_1(x^{(1)}) f_1(x^{(1)}) p_2(x^{(2)}|x^{(1)})}.
$$
which is the same as

\[ w^{(2)} = w^{(2)}(x^{(1)}, x^{(2)}) = \frac{f_1(x^{(1)})}{p_1(x^{(1)})} \cdot \frac{f_2(x^{(1)}, x^{(2)})}{f_1(x^{(1)}) p_2(x^{(2)}|x^{(1)})}. \]

It is easy to see that the first fraction is just the previous weight. The second fraction refers to the incremental weight.

In general,

\[ p_i(x^{(1)}, \ldots, x^{(i)}) = p_1(x^{(1)}) \cdot p_2(x^{(2)}|x^{(1)}) \cdots p_i(x^{(i)}|x^{(1)}, \ldots, x^{(i-1)}). \]

There is a need to sample

\[ X_j^{(i)}|X_j^{(i-1)} \sim p_i(x^{(i)}|X_j^{(1)}, \ldots, X_j^{(i-1)}), \quad j = 1, \ldots, N. \]

The weights are

\[ w^{(i)}(x^{(1)}, \ldots, x^{(i)}) = w^{(i-1)}(x^{(1)}, \ldots, x^{(i-1)}) \cdot \frac{f_i(x^{(1)}, \ldots, x^{(i)})}{f_{i-1}(x^{(1)}, \ldots, x^{(i-1)}) p_i(x^{(i)}|x^{(1)}, \ldots, x^{(i-1)})}. \]

At any time i, we get

\[ X_j^{(i)} \sim p_i(x^{(1)}, \ldots, x^{(i)}), \quad w^{(i)}_j = \frac{f_j(x^{(1)}, \ldots, x^{(i)})}{p_i(x^{(1)}, \ldots, x^{(i)})}, \quad j = 1, \ldots, N. \]

This is the way to get the approximation of distributions \( f_1, \ldots, f_n \) by sequential importance sampling (SIS).

Common problem is how to sample from the difficult density function f. We can apply SIS by building a sequence \( f_1, \ldots, f_n \) such that \( f_n = f \), and finding importance functions \( p_1, \ldots, p_n \). Del Moral and coauthors proposed in [38] to move from one sample to another within SIS using a Markov kernel \( K_j : \Omega \times \mathcal{F} \to [0, 1] \),
where \((\Omega, \mathcal{F})\) is a measurable space, and \(K_i(x, x^*)\) is a density function. Then, the marginal distribution of the sample values \(X_j^{(i)}, \ j = 1, \ldots, N\) is

\[
p_i(x^*) = \int_{\Omega} p_{i-1}(x)K_i(x, x^*)dx.
\] (2.3)

In order to use the importance sample estimates of \(f_i\), \(p_i\) should be calculated point-wise.

The choice of the sequence of transitional kernels \(K_i\) as a Markov chain Monte Carlo kernels of invariant distributions \(f_i\) is recommended by Del Moral [38]. This allows important distributions \(p_i\)'s to be reasonably close to the corresponding target distributions. Efficient MCMC algorithms can be developed for getting those \(p_i\) "good enough". There is a big drawback in the SIS that \(p_i\) in most cases is not possible to find using the equation 2.3 due to the complexity in integration. The solution offered by Del Moral [38] is the proposal of artificial backward in time Markov kernels

\[
L_{i-1} : \Omega \times \mathcal{F} \to [0, 1],
\]

where \(L_{i-1}(x^{(i)}, x^{(i-1)})\) is a density function. Then, the importance sampling performed on artificial joint target distribution \(\tilde{f}_i(x^{(1)}, \ldots, x^{(i)})\), which is defined as

\[
\tilde{f}_i(x^{(1)}, \ldots, x^{(i)}) = f_i(x^{(i)}) \prod_{k=1}^{i-1} L_k(x^{(k+1)}, x^{(k)}).
\] (2.4)

Importance joint distributions are \(\{p_i(x^{(1)}, \ldots, x^{(i)})\}, \ i = 1, \ldots, n\). \(\{f_i(x^{(i)})\}, \ i = 1, \ldots, n\) are marginal distributions by construction.

At the time \(i\), the weights are

\[
w^{(i)}(x^{(1)}, \ldots, x^{(i)}) = \frac{\tilde{f}_i(x^{(1)}, \ldots, x^{(i)})}{p_i(x^{(1)}, \ldots, x^{(i)})}.
\]

After applying equation 2.4 and rewriting the denominator, we get

\[
w^{(i)}(x^{(1)}, \ldots, x^{(i)}) = \frac{f_i(x^{(i)}) \prod_{k=1}^{i-1} L_k(x^{(k+1)}, x^{(k)})}{p_i(x^{(1)}, \ldots, x^{(i-1)}) p_i(x^{(i)}|x^{(1)}, \ldots, x^{(i-1)})}.
\]
Using the assumption form SIS that \( p_j(x^{(1)}, \ldots, x^{(i-1)}) = p_{i-1}(x^{(1)}, \ldots, x^{(i-1)}) \) and multiplying numerator and denominator by \( \tilde{f}_{i-1}(x^{(1)}, \ldots, x^{(i-1)}) \), we see

\[
w^{(i)}(x^{(1)}, \ldots, x^{(i)}) = \frac{\tilde{f}_{i-1}(x^{(1)}, \ldots, x^{(i-1)}) f_i(x^{(i)}) \prod_{k=1}^{i-1} L_k(x^{(k+1)}, x^{(k)})}{p_{i-1}(x^{(1)}, \ldots, x^{(i-1)}) p_i(x^{(i)}|x^{(1)}, \ldots, x^{(i-1)})}.
\]

Now, we can split the last fraction into the product of two fractions and apply equation artificial for \( \tilde{f}_{i-1} \):

\[
w^{(i)}(x^{(1)}, \ldots, x^{(i)}) = \frac{\tilde{f}_{i-1}(x^{(1)}, \ldots, x^{(i-1)}) f_i(x^{(i)}) \prod_{k=1}^{i-1} L_k(x^{(k+1)}, x^{(k)})}{p_{i-1}(x^{(1)}, \ldots, x^{(i-1)})} \cdot \frac{\prod_{k=1}^{i-2} L_k(x^{(k+1)}, x^{(k)}) p_i(x^{(i)}|x^{(1)}, \ldots, x^{(i-1)})}{p_i(x^{(i)}|x^{(1)}, \ldots, x^{(i-1)})}.
\]

Notice that the first fraction is the previous weight, simplify the second fraction:

\[
w^{(i)}(x^{(1)}, \ldots, x^{(i)}) = w^{(i-1)}(x^{(1)}, \ldots, x^{(i-1)}) \cdot \frac{f_i(x^{(i)}) L_{i-1}(x^{(i)}, x^{(i-1)})}{\tilde{f}_{i-1}(x^{(i-1)}) p_i(x^{(i)}|x^{(1)}, \ldots, x^{(i-1)})}.
\]

Following the suggestion of Del Moral [38], we use Markov kernel as the transition probability we get

\[
w^{(i)}(x^{(1)}, \ldots, x^{(i)}) = w^{(i-1)}(x^{(1)}, \ldots, x^{(i-1)}) \cdot \frac{f_i(x^{(i)}) L_{i-1}(x^{(i)}, x^{(i-1)})}{\tilde{f}_{i-1}(x^{(i-1)}) K_i(x^{(i-1)}, x^{(i)})}.
\]

At time \( i \), the approximation of the target distribution \( f_i \) is given by

\[
f_i(x) \propto \sum_{j=1}^{N} w^{(i)}_j \delta(x - x^{(j)}_i),
\]

where \( \delta(\cdot) \) is Dirac delta function.

The problem with the sequential importance sampling is that variance of the unnormalized importance weights get to be bigger and bigger with every new point in time, which may lead to a degeneracy of the algorithm. The solution mentioned in [13] is to resample the current values with treating corresponding normalized weights as the probabilities. The new sample should be taken with the equal weights. The condition of when to do resampling is determined by effective sample size (ESS), which can be found by

\[
n_{\text{eff}} = \frac{1}{\sum_{j=1}^{N} \left( w^{(i)}_j \right)^2},
\]
where \( \{W_j^{(i)}\}, j = 1, \ldots, N \) are normalized weights of \( \{w_j^{(i)}\}, j = 1, \ldots, N \) introduced earlier in this section.

If ESS is less than some predetermined value, usually \( N/2 \), then the resampling need to be performed.

### 2.3.3 Sequential ABC

In 2007 Sisson [46] introduced a first sequential Monte Carlo (SMC) algorithm named ABC-PRC (PRC is for partial rejection control). His work is based on the theory described by Del Moral [38] described in the previous section. The algorithm goes as follows.

<table>
<thead>
<tr>
<th>ABC-PRC:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Initialize ( \varepsilon_1, \ldots, \varepsilon_T ), and specify initial sampling distribution ( \mu_1 ). Set ( t = 1 ).</td>
</tr>
<tr>
<td>2. Set particle indicator ( i = 1 ).</td>
</tr>
<tr>
<td>2.1 If ( t = 1 ), sample ( \theta^{**} \sim \mu_1(\theta) ) independently.</td>
</tr>
<tr>
<td>\hspace{1cm} If ( t &gt; 1 ), get ( \theta^* ) from the population ( {\theta_t^{(i)}} ) with weights ( {W_t^{(i)}} )</td>
</tr>
<tr>
<td>\hspace{1cm} ( \theta^{**} \sim K_t(\theta</td>
</tr>
<tr>
<td>\hspace{1cm} Generate ( x^{**} \sim f(x</td>
</tr>
<tr>
<td>2.2 Set</td>
</tr>
<tr>
<td>( \theta_t^{(i)} = \theta^{**}, W_t^{(i)} = \begin{cases} \pi(\theta_t^{(i)})/\mu_1(\theta_t^{(i)}), &amp; \text{if } t = 1, \ \pi(\theta_t^{(i)})L_{t-1}(\theta^*</td>
</tr>
<tr>
<td>\hspace{1cm} If ( i &lt; N ), increment ( i = i + 1 ) and go to 2.1.</td>
</tr>
<tr>
<td>3. Normalize ( W_t^{(i)} ) such that ( \sum_{i=1}^{N} W_t^{(i)} = 1 ).</td>
</tr>
<tr>
<td>\hspace{1cm} If ( ESS = \left[ \sum_{i=1}^{N} (W_t^{(i)})^{-1} \right] &lt; E ), then resample with replacement and assign ( W_t^{(i)} = 1/N ).</td>
</tr>
<tr>
<td>\hspace{1cm} ( E ) is predetermined value.</td>
</tr>
<tr>
<td>4. If ( t &lt; T ), increment ( t = t + 1 ) and go to step 2.</td>
</tr>
</tbody>
</table>

This method is proved to produce a biased posterior sample by Beaumont [4]. The problem was in the calculating of incremental weights. In the paper [6] the ideal case when \( x = x^* \) was considered. We know
that in this case ABC-REJ and ABC-MCMC give samples from exact posterior distribution \( \pi(\theta|x) \). It is expected that ABC-PRC will behave the same way and

\[
E(\theta_t w_t) = \int \theta_t \pi(\theta|x) \, d\theta_t. \tag{2.5}
\]

Without loss of generality suppose that \( \theta_{t-1} \)'s come from exact posterior distribution \( \pi(\theta|x) \). Then the joint density of \((\theta_{t-1}, \theta_t)\) when \( \theta_t \) is accepted is

\[
p(\theta_{t-1}, \theta_t|x) \propto f(x|\theta_t) \cdot \pi(\theta_{t-1}|x) K_t(\theta_t|\theta_{t-1}).
\]

According to Beaumont’s paper [4], where he takes into account weights suggested by Sisson [46], we get

\[
E(\theta_t w_t) \propto \int \int \theta_t \cdot \frac{\pi(\theta_t)}{\pi(\theta_{t-1})} \cdot \frac{L_{t-1}(\theta_t|\theta_{t-1})}{K_t(\theta_t|\theta_{t-1})} \cdot f(x|\theta_t) \cdot \pi(\theta_{t-1}|x) K_t(\theta_t|\theta_{t-1}) \, d\theta_{t-1} \, d\theta_t. \tag{2.6}
\]

Then, applying the fact that posterior distribution is proportional to the likelihood times prior, the equation 2.6 can be rewritten

\[
E(\theta_t w_t) \propto \int \int \theta_t \cdot \frac{\pi(\theta_t)}{\pi(\theta_{t-1})} \cdot \frac{L_{t-1}(\theta_t|\theta_{t-1})}{K_t(\theta_t|\theta_{t-1})} \cdot f(x|\theta_t) \cdot \pi(\theta_{t-1}|x) f(x|\theta_{t-1}) K_t(\theta_t|\theta_{t-1}) \, d\theta_{t-1} \, d\theta_t.
\]

After simplifying,

\[
E(\theta_t w_t) \propto \int \int \theta_t \cdot \frac{\pi(\theta_t) L_{t-1}(\theta_t|\theta_{t-1})}{\pi(\theta_{t-1}) K_t(\theta_t|\theta_{t-1})} \cdot f(x|\theta_t) \cdot \pi(\theta_{t-1}|x) \, d\theta_{t-1} \, d\theta_t
\]

\[
\propto \int \theta_t \cdot \pi(\theta_t|x) \cdot L_{t-1}(\theta_t|\theta_{t-1}) f(x|\theta_{t-1}) \, d\theta_{t-1} \, d\theta_t
\]

\[
\propto \int \theta_t \cdot \pi(\theta_t|x) \cdot \left( \int L_{t-1}(\theta_t|\theta_{t-1}) f(x|\theta_{t-1}) \, d\theta_{t-1} \right) \, d\theta_t.
\]

From the equation 2.5 we expect the integral in the parenthesis of the last equation to be the same constant for every value of \( \theta_t \). This is impossible for the random walk proposal. Thus, as stated in Beaumont [4], the weight in the ABC-PRC algorithm proposed by Sisson [46] are inappropriate.
The correction of ABC-PRC algorithm was obtained, see Beaumont [4] and Sisson [47]. Both papers showed similar way for improvement by calculating the importance weights for accepted $\theta_t$'s using the following formula:

$$w_t^{(i)} = \frac{\pi(\theta_t^{(i)})}{\pi'(\theta_t^{(i)})},$$

where

$$\pi'(\theta_t) = \sum_{j=1}^{N} W_{t-1}(\theta_t^{(j)}) K_t(\theta_t|\theta_t^{(j)}),$$

is the distribution used to get the $\theta_t$'s. To show that the resulting sample is from the unbiased posterior we can evaluate the following expectation:

$$E(\theta_t w_t) \propto \int \int \theta_t \frac{\pi(\theta_t)}{\pi'(\theta_t)} f(x|\theta_t) \pi'(\theta_t) \pi''(\theta_{t-1}) d\theta_{t-1} d\theta_t$$

$$\propto \int \theta_t \pi(\theta_t|x) \left\{ \int \pi''(\theta_{t-1}) d\theta_{t-1} \right\} d\theta_t$$

$$\propto \int \theta_t \pi(\theta_t|x) d\theta_t.$$

This is the proof of unbiasedness of the resulting sample. The following algorithm was proposed by Sisson [47]:

**ABC-PRC Algorithm (corrected):**

1. Initialize $\epsilon_1, ..., \epsilon_T$, and specify initial sampling distribution $\mu_1$. Set $t=1$.
2. Set partical indicator $i=1$.
   2.1 If $t=1$, sample $\theta^{**} \sim \mu_1(\theta)$ independently.
   2.2 If $t > 1$, get $\theta^*$ from the population $\{\theta_t^{(j)}\}$ with weights $\{W_t^{(i)}\}$.
   2.3 Get $\theta^{**} \sim K_t(\theta|\theta^*)$, where $K_t$ is a transition kernel.
Generate \( x^{**} \sim f(x|\theta^{**}) \), repeat 2.1 until \( \rho(S(x^{**}), S(x)) < \varepsilon_t \).

2.2 Set

\[
\theta_t^{(i)} = \theta^{**}, W_t^{(i)} = \begin{cases} 
\pi(\theta_t^{(i)})/\mu_1(\theta_t^{(i)}), & \text{if } t = 1, \\
\pi(\theta_t^{(i)})/\sum_{j=1}^{N} W_{t-1}(\theta_{t-1}^{(j)}) K_t(\theta_t^{(i)}|\theta_t^{(j)}), & \text{if } t > 1,
\end{cases}
\]

If \( i < N \), increment \( i = i + 1 \) and go to 2.1.

3. Normalize \( W_t^{(i)} \).

If \( ESS = \left[ \sum_{i=1}^{N} \left( W_t^{(i)} \right)^2 \right]^{-1} < E \), then resample with replacement and assign \( W_t^{(i)} = 1/N \).

4. If \( t < T \), increment \( t = t + 1 \) and to step 2.

Beaumont in [4] showed the improvement of ABC - PRC using the same idea as Sisson, but called it ABC - PMC (PMC stands for population Monte Carlo). He illustrated the algorithm considering the special case, when the transition kernel function chosen to be an independent random walk component-wise, for every \( i \):

\[
K_t \left( \theta_t^{(i)} | \theta_{t-1}^{(i)} \right) = \frac{1}{\tau^{(i)}} \varphi \left( \frac{\theta_t^{(i)} - \theta_{t-1}^{(i)}}{\tau^{(i)}} \right),
\]

where \( \tau \) is a scale factor. In this case, Beaumont showed the optimal choice of \( \tau \) for each iteration is

\[
\left[ \tau^{(i)} \right]^2 = 2 \text{ var}(\theta_t^{(i)}|x),
\]

under the posterior distribution. The following algorithm proposed by Beaumont [4]:

**ABC-PMC:**

1. Initialize \( \varepsilon_1, ..., \varepsilon_T \). Set \( t = 1 \).
2. Set particle indicator \( i = 1 \).
   2.1 If \( t = 1 \), sample \( \theta^{**} \sim \pi(\theta) \) independently.
      If \( t > 1 \), get \( \theta^* \) from the population \( \{ \theta_{t-1}^{(i)} \} \) with weights \( \{ W_{t-1}^{(i)} \} \)
      \( \theta^{**} \sim N(\theta^*, \sigma_t^2) \), \( \sigma_t^2 \) is twice the empirical variance of the \( \theta_{t-1} \).
Generate $x^* \sim f(x|\theta^*)$, repeat 2.1 until $\rho(S(x^*),S(x)) < \epsilon_t$.

2.2 Set

$$
\theta_t^{(i)} = \theta^*, W_t^{(i)} = \begin{cases} 1, & \text{if } t = 1, \\
\pi(\theta_t^{(i)}) / \sum_{j=1}^{N} W_{t-1}^{(j)} \varphi(\sigma_t^{-1}(\theta^* - \theta_t^{(j)})), & \text{if } t > 1,
\end{cases}
$$

If $i < N$, increment $i=i+1$ and go to 2.1.

Take $\sigma_{t+1}^2$ as twice the weighted empirical variance of the $\theta_t^{(i)}$'s.

3. Normalize $W_t^{(i)}$.

If $ESS = \left[ \sum_{i=1}^{N} \left( W_t^{(i)} \right)^2 \right]^{-1} < E$, then resample with replacement and assign $W_t^{(i)} = 1/N$.

E is predetermined value.

4. If $t < T$, increment $t = t + 1$ and to step 2.

Well known problem with the ABC rejection and ABC Markov chain Monte Carlo methods considered earlier is the high rejection rates of the parameters. Sequential Monte Carlo techniques do help to decrease the rejection rates. Bonassi in [7] proposed the improvement of sequential idea which allows to increase the efficiency of the sequential algorithms by adding the adaptive weights. Instead of using the transition kernel for the parameters $K_t(\theta|\theta^*)$, Bonassi offered a joint kernel $K_t(x,\theta|x^*,\theta^*)$ for a distribution of accepted pair $(x,\theta)$. In the paper [7] he used a product kernel

$$
K_t(x,\theta|x^*,\theta^*) = K_{t,x}(x|x^*) \cdot K_{t,\theta}(\theta|\theta^*).
$$

Then, the joint approximate distribution for the accepted pair $(\theta,x)$ at time $t$ is

$$
p_t(\theta,x) \propto \sum_{i=1}^{N} W_{t-1}^{(i)} K_{t,x}(x|\theta_t^{(i)}) \cdot K_{t,\theta}(\theta|\theta_t^{(i)}),
$$

and posterior approximate importance distribution is

$$
p_t(\theta|x) \propto \sum_{i=1}^{N} W_{t-1}^{(i)} K_{t,x}(x|\theta_t^{(i)}) \cdot K_{t,\theta}(\theta|\theta_t^{(i)}),
$$

where $x$ is observed data.
Bonassi also proved that at any time $t$ the posterior approximated distribution $\hat{\pi}_t(\theta|x)$, where $x$ is observed data, determines a correct importance sampling with the target $\pi(\theta|x^* : \rho(x^*, x) < \varepsilon_t)$, and the resulting acceptance rates are higher than for general ABC-SMC.

Let event $A$ be the acceptance event at time $t$, such that

$$A = \{x^* : \rho(x^*, x) < \varepsilon_t\},$$

where $x$ is observed data, and $x^*$ is generated from the model. Then, the approximation of the joint density of the accepted pair $(x, \theta)$ is

$$\hat{\pi}_t(x, \theta|A) = \int \int \pi(\theta_{t-1}|A)\pi(x_{t-1}|\theta_{t-1}, A)K_x(x|x_{t-1})K_{\theta}(\theta|\theta_{t-1})dx_{t-1}d\theta_{t-1}.$$

Approximation of the posterior density

$$\hat{\pi}_t(\theta|x) \propto \int \int \pi(\theta_{t-1}|A)\pi(x_{t-1}|\theta_{t-1}, A)K_x(x|x_{t-1})K_{\theta}(\theta|\theta_{t-1})dx_{t-1}d\theta_{t-1},$$

where $x$ is observed data. The distribution where $\theta$’s are generated from at time $t$ is

$$p(\theta) = \sum_{i=1}^{N} v^{(i)}_{t-1}K_{\theta,t}(\theta|\theta^{(i)}_{t-1}),$$

where $v^{(i)}_{t-1} = W^{(i)}_{t-1}K_{t,x}(x|x^{(i)}_{t-1})$. Once $\theta$ is found, the next step is to generate the data set, such that the event $A$ is happened. So, the importance density is

$$p(\theta|A) \propto p(\theta)pr(A|\theta).$$

Now, the weights will be

$$w(\theta) = \frac{\pi(\theta|A)}{p(\theta|A)} \propto \frac{\pi(\theta) \cdot pr(A|\theta)}{p(\theta) \cdot pr(A|\theta)}.$$
Because, $pr(A|\theta) = \int_A \pi(x, \theta) dx$, we can simplify the last expression,

$$w(\theta) \propto \frac{\pi(\theta)}{p(\theta)},$$

which exactly as it appears in the algorithm. Thus, the importance sampling is correct.

In order to show that the acceptance rates are higher for ABC - SMC with adaptive weights (ABC - SMC AW) than for ABC - SMC, Bonassi compared the two probabilities of event A at time $t+1$, $pr_0(A_{t+1})$ for ABC - SMC and $pr_1(A_{t+1})$ for ABC - SMC AW, where $A_{t+1} = \{x^* : \rho(x, x^*) < \epsilon_{t+1}\}$. Let $\pi(\theta)$ be the proposal density for ABC - SMC, $\pi(\theta|x)$ be the proposal density for ABC - SMC AW, $m(x)$ be marginal density, $x$ be observed data. The proof goes as follows.

It is clear that

$$pr_1(A_{t+1}) = \int \pi(\theta|x) \ pr_0(A_{t+1}|\theta) \ d\theta.$$

Using the definition of the conditional probability, we get

$$pr_1(A_{t+1}) = \int \frac{f(x|\theta) \pi(\theta)}{m(x)} \ pr_0(A_{t+1}|\theta) \ d\theta.$$

Then, it can be rewritten,

$$pr_1(A_{t+1}) = \frac{E_\pi [pr_0(A_{t+1}|\Theta) f(x|\Theta)]}{m(x)}.$$

Applying assumption, that $cov(pr_0(A_{t+1}|\Theta) f(x|\Theta)) > 0$,

$$pr_1(A_{t+1}) > \frac{E_\pi [pr_0(A_{t+1}|\Theta)] E_\pi [f(x|\Theta)]}{m(x)}.$$

After simplifying,

$$pr_1(A_{t+1}) > E_\pi [pr_0(A_{t+1}|\Theta)],$$

which is the same as

$$pr_1(A_{t+1}) > \int pr_0(A_{t+1}|\theta) \pi(\theta) \ d\theta = pr_0(A_{t+1}).$$
Thus, the ABC - SMC AW has a higher acceptance rate and it is more efficient. Below is the algorithm proposed by Bonassi [7]:

**ABC-SMC AW:**

1. Initialize $\varepsilon_1, \ldots, \varepsilon_T$, and specify initial sampling distribution $\mu_1$. Set $t=1$.

2. Set particle indicator $i=1$.

   2.1 If $t=1$, sample $\theta^{**} \sim \pi(\theta)$.

   If $t > 1$, compute weights $v^{(i)}_{t-1} = W^{(i)}_{t-1} K_{\theta,t}(x|x^{(i)}_{t-1})$, and normalize them.

   Get $\theta^*$ from the $\{\theta_{t-1}^{(i)}\}$ w.p. $\{v^{(i)}_{t-1}\}$

   Get $\theta^{**} \sim K_{\theta,t}(\theta|\theta^*)$, where $K_{\theta,t}$ is a transition kernel.

   Generate $x^{**} \sim f(x|\theta^{**})$, repeat 2.1 until $\rho(S(x^{**}), S(x)) < \varepsilon_t$.

2.2 Set

   $$\theta_t^{(i)} = \theta^{**}, W_t^{(i)} = \begin{cases} 1, & \text{if } t = 1, \\ \pi(\theta_t^{(i)}) / \sum_{j=1}^{N} v_{t-1}^{(j)} K_{\theta,t}(\theta_t^{(i)}|\theta_{t-1}^{(j)}), & \text{if } t > 1, \end{cases}$$

   If $i < N$, increment $i=i+1$ and go to 2.1.

3. Normalize $W_t^{(i)}$.

4. If $t < T$, increment $t = t + 1$ and go to step 2.

### 2.4 Semi-automatic ABC

As mentioned above, it is beneficial to use summary statistics within ABC algorithm. The question though is which one of summary statistics is the best to use? Fearnhead and Prangle in [19] introduced the semi-automatic technique for constructing appropriate summary statistics, which will increase the accuracy of desirable parameter estimation. In theory, it had been shown that the best choice of summary statistic is the posterior mean of the parameter [19]. Unfortunately, in practice it is not available. Fearnhead and Prangle used simulations to estimate the posterior mean based on the observed data, and then used it as an estimate of summary statistic within ABC algorithm.
First, the ABC algorithm need to be run one time to obtain a region of posterior mass. Fearnhead and Prangle call this region a training subset of parameter space for parameter values simulations. This step is important if the parameter prior distribution is non-informative. In the case of informative prior of the parameter this step better to avoid. Here, the arbitrarily chosen summary statistic is used within ABC algorithm.

Second, the parameters should be simulated from the prior distribution truncated to the training subset, which was determined in the first step. Then, for each value of the parameter the data set is simulated. This procedure is repeated M times. Hence, there are M sets of parameter values with corresponding data sets.

The third step is to estimate summary statistics based on simulated parameter and data sets. For this reason, the linear regression approach is used in [19]. Also, it is mentioned that, in general, more sophisticated methods do not perform better than linear regression. Transformations of the data sets are used as explanatory variables. For, \( f(x) = (x, x^2, x^3) \), that is a vector of length 3n if the data set is of length n, which contains all observations of the data, its second and third powers. The response variables are the parameter values.

Suppose, we have \( \theta^{(1)}_i, \ldots, \theta^{(M)}_i \), the simulated i-th values of the parameter from each of M sets. \( f(x^{(1)}_i), \ldots, f(x^{(M)}_i) \) are the transformations of the simulated observations for i-th values of the parameter from each of M sets. The fitted regression model is

\[
\theta_i = E(\theta_i|x_i) + \varepsilon_i = \beta^{(i)}_0 + \beta^{(i)}_1 f(x_i) + \varepsilon_i,
\]

where \( \varepsilon_i \) is the error with mean of zero. The estimate of \( E(\theta_i|x_i) \) is \( \hat{\beta}^{(i)}_0 + \hat{\beta}^{(i)}_1 f(x_i) \). The i-th summary statistic is \( \hat{\beta}^{(i)}_1 f(x_i) \), since the constant term does not make a difference when we calculate a distance between two summary statistics of simulated and true observations.

Finally, the ABC algorithm can be run with the determined choice of summary statistics. It is important to notice that semi-automatic ABC is more robust to the choice of summary statistic than a regular ABC. The first reason is that the arbitrary chosen summary statistic in the first step is only to make the second step more efficient, and, hence, the final result does not depend much on this choice. The second reason is
that even though the semi-automatic ABC depends on the choice of explanatory variables for constructing summary statistics, a large number of explanatory variables can be chosen.
CHAPTER 3
EMPIRICAL LIKELIHOOD

Efficiency of the common ABC methods hinges on the choice of summary statistic, distance metric, and tolerance level. There are several authors such as Marin [32], Wilkinson [53], and Csillery [14] who have studied the theory behind these choices and how they affect the validity of the algorithms. Mengersen, Pudlo, and Robert [36] proposed incorporating empirical likelihood into one of the basic ABC algorithms, bypassing the problem of determining the best summary statistic, distance metric, and tolerance level. Also, it eliminates the need to have a simulation model. In their paper they show, through simulation studies, that the combination leads to a great time savings.

Empirical likelihood is a nonparametric estimation approach in statistics that does not require the choice of a known distribution family for the data. Empirical likelihood methods show good results when applied to censored and truncated data. Since the constraints and prior knowledge can be integrated easily into empirical likelihood techniques, they can be combined with ABC methods. In this chapter, the theory of empirical likelihood performed based on Owen [39].

3.1 Nonparametric Maximum Likelihood

First, the empirical cumulative distribution function is defined. Then, it is proved to be a nonparametric maximum likelihood estimate.

The following notations are used, $X \in \mathbb{R}$ is a random variable. $F(x) = Pr(X \leq x)$, where $-\infty < x < \infty$, is a cumulative distribution function. Let $F(x-) = Pr(X < x)$, then $Pr(X = x) = F(x) - F(x-)$. $\mathbb{1}_{A(x)}$ is the event indicator function, so if $A(x)$ is true, the function is equal to 1, otherwise it is equal to 0.

**Definition 3.1.1.** Given $X_1, \ldots, X_n \in \mathbb{R}$ are random variables. The empirical cumulative distribution function of $X_1, \ldots, X_n$ is

$$F_n(x) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{X_i \leq x}, \text{ for } -\infty < x < \infty.$$
**Definition 3.1.2.** Given $X_1, \ldots, X_n \in \mathbb{R}$ are independent random variables with common cumulative distribution function (CDF). The nonparametric likelihood of the CDF $F$ is

$$L(F) = \prod_{i=1}^{n} (F(X_i) - F(X_i^-)).$$

This definition reflects the fact that the likelihood is the product of probabilities of getting the observed values from the cumulative distribution function. If we have a continuous distribution, then based on Definition 3.1.2 the likelihood function of it is zero. In order to have positive nonparametric likelihood, we should have all observed values with positive probabilities.

The following theorem shows that the empirical cumulative distribution function maximizes the nonparametric likelihood function.

**Theorem 3.1.1.** Given $X_1, \ldots, X_n \in \mathbb{R}$, assumed independent with common CDF. Let $F_n$ be their empirical CDF and let $F$ be any CDF. If $F \neq F_n$, then $L(F) < L(F_n)$.

**Proof:** Let $z_1, z_2, \ldots, z_m$ be distinct values in $\{X_1, X_2, \ldots, X_n\}$, and $n_j \geq 1$ be the number of $X_i$ that are equal to $z_j$. Denote $p_j = F(z_j) - F(z_j^-)$, and assume that $p_j > 0$. If at least one $p_j = 0$, then $L(F) = 0 < L(F_n)$. Put $\hat{p}_j = n_j / n$, and for at least one $j \hat{p}_j \neq p_j$.

Now, using the definition the likelihood function and empirical cumulative distribution function it can written:

$$\log \left( \frac{L(F)}{L(F_n)} \right) = \log \left( \frac{\prod_{j=1}^{m} p_j^{n_j}}{\prod_{j=1}^{m} \hat{p}_j^{n_j}} \right) = \log \left( \prod_{j=1}^{m} \left( \frac{p_j}{\hat{p}_j} \right)^{n_j} \right).$$

Using properties of the logarithm function,

$$\log \left( \frac{L(F)}{L(F_n)} \right) = \sum_{j=1}^{m} n_j \log \left( \frac{p_j}{\hat{p}_j} \right).$$

Using the notation for $\hat{p}_j$,

$$\log \left( \frac{L(F)}{L(F_n)} \right) = n \sum_{j=1}^{m} \hat{p}_j \log \left( \frac{p_j}{\hat{p}_j} \right).$$
Applying the knowledge that \( \log(x) \leq x - 1 \) for all \( x > 0 \), and \( \log(x) = x - 1 \) if and only if \( x = 1 \). Also, using the assumption that for at least one \( j \hat{p}_j \neq p_j \),

\[
\log \left( \frac{L(F)}{L(F_n)} \right) = n \sum_{j=1}^{m} \hat{p}_j \log \left( \frac{p_j}{\hat{p}_j} \right) < n \sum_{j=1}^{m} \hat{p}_j \left( \frac{p_j}{\hat{p}_j} - 1 \right).
\]

Simplifying, and using the fact that \( \sum_{j=1}^{m} \hat{p}_j = 1 \) due to the notations,

\[
\log \left( \frac{L(F)}{L(F_n)} \right) < n \sum_{j=1}^{m} \hat{p}_j \left( \frac{p_j}{\hat{p}_j} - 1 \right) = n \left( \sum_{j=1}^{m} p_j - 1 \right).
\]

By assumption \( \sum_{j=1}^{m} p_j \leq 1 \), so

\[
\log \left( \frac{L(F)}{L(F_n)} \right) < n \left( \sum_{j=1}^{m} p_j - 1 \right) \leq 0.
\]

Hence, \( L(F) < L(F_n) \).

### 3.2 Nonparametric Likelihood Ratio

In parametric inference, hypothesis tests and confidence intervals can be set up based on likelihood ratio. Usually, \( \theta \) is used as the notation for unknown parameter, which comes from a parameter space \( \Theta \), \( \hat{\theta} \) is the maximum likelihood estimator. If \( L(\theta) \ll L(\hat{\theta}) \), then the hypothesis that \( \theta = \theta_0 \) is rejected. Similar, in nonparametric case we can define the nonparametric likelihood ratio.

**Definition 3.2.1.** For a distribution \( F \), the nonparametric likelihood ratio is

\[
R(F) = \frac{L(F)}{L(F_n)},
\]

where \( L(F) \) is nonparametric likelihood, \( F_n \) is the empirical CDF from definition 3.1.1.

Let us look at the nonparametric likelihood ratio in more details. Suppose, we have \( n \) distinct observations \( \{X_1, \ldots, X_n\} \). They are independent, identically distributed with common distribution \( F \). Let \( p_i \) the
probability that F places on $X_i$. $p_i \geq 0$, then $\sum_{i=1}^n p_i \leq 1$, and $L(F) = \prod_{i=1}^n p_i$. In this case, the definition 3.2.1 can be expanded to

$$R(F) = \frac{L(F)}{L(F_n)} = \frac{\prod_{i=1}^n p_i}{\prod_{j=1}^m \hat{p}_j^{n_j}} = \prod_{j=1}^m \left( \frac{p_j}{\hat{p}_j} \right)^{n_j}.$$ 

Now, assume that not all $n$ observations are distinct. Let $z_1, z_2, \ldots, z_m$ be distinct values in \{X_1, X_2, \ldots, X_n\}, and $n_j \geq 1$ be the number of $X_i$ that are equal to $z_j$. Denote $p_j = F(z_j) - F(z_{j-})$, and put $\hat{p}_j = n_j/n$. Then,

$$R(F) = \frac{L(F)}{L(F_n)} = \frac{\prod_{j=1}^m p_j^{n_j}}{\prod_{j=1}^m \hat{p}_j^{n_j}} = \prod_{j=1}^m \left( \frac{p_j}{\hat{p}_j} \right)^{n_j}.$$ 

Using the notation $\hat{p}_j = n_j/n$, we can say that

$$R(F) = \prod_{j=1}^m \left( \frac{p_j n_j}{n_j} \right)^{n_j}.$$ 

In addition, we can write that $L(F) = \prod_{i=1}^n w_i$, where $w_i$ are the observation specific weights, chosen in such a way that $p_j = \sum_{i \in X_i = z_j} w_i$.

When some of the observed values are the same, the likelihood value is not unique. It is enough to consider just the maximum value. The maximum of the product is obtained when the weights are the same within the $j$-th group, so it is $p_j/n_j$,

$$\max \left( \prod_{i=1}^n w_i \right) = \prod_{j=1}^m \left( \frac{p_j}{n_j} \right)^{n_j}.$$ 

Also, we can write it as,

$$\prod_{j=1}^m \left( \frac{p_j}{n_j} \right)^{n_j} = L(F) \cdot \prod_{j=1}^m \left( \frac{1}{n_j} \right)^{n_j}.$$ 

In the terms of the nonparametric likelihood ratio, 

$$R(F) = \frac{L(F)}{L(F_n)} = \frac{\prod_{j=1}^m \left( \frac{p_j}{n_j} \right)^{n_j}}{\prod_{j=1}^m \left( \frac{n_j}{n} \right)^{n_j}}.$$
After canceling the term $\prod_{j=1}^{m} (n_j)^{n_j}$, we have

$$R(F) = \frac{L(F)}{L(F_n)} = \frac{\prod_{j=1}^{m} \left( \frac{p_j}{n_j} \right)^{n_j}}{\prod_{j=1}^{m} \left( \frac{1}{n} \right)^{n_j}} \geq \frac{\prod_{i=1}^{n} w_i}{\prod_{i=1}^{n} \left( \frac{n}{n} \right) w_i} = \prod_{i=1}^{n} n w_i.$$  

Hence, we can treat the likelihood ratio as: $R(F) = \prod_{i=1}^{n} n w_i$, where $w_i \geq 0, \sum_{i=1}^{n} w_i \leq 1$ despite the fact that some data values may be the same.

Like parametric likelihood method, the interest is in maximizing the likelihood ratio. Let $\theta \in \Theta$ be the parameter of interest, and $\theta = T(F)$, where $T$ is some function of distribution. $F \in \mathcal{F}$, where $\mathcal{F}$ is the set of distributions.

**Definition 3.2.2.** For a given distribution $F \in \mathcal{F}$, and for parameter $\theta = T(F)$ the profile likelihood ratio function is

$$\mathcal{R}(\theta) = \sup \{ R(F) | \theta = T(F), F \in \mathcal{F} \}.$$  

For statistical inference, we need to know the distribution of the profile empirical likelihood ratio. Then, we can set up the hypothesis test as follows: reject $H_0 : \theta_0 = T(F_0)$ if $\mathcal{R}(\theta_0) < r_0$, where $r_0$ is a threshold. The confidence region is $\{ \theta | \mathcal{R}(\theta) \geq r_0 \}$. The threshold $r_0$ may be chosen by empirical likelihood theorem, which tells the asymptotic behavior of profile likelihood. The following theorem is the empirical likelihood theorem for the univariate mean.

**Theorem 3.2.1.** Let $X_1, \ldots, X_n$ be independent random variables with common distribution $F_0$. Let $E(X_i) = \mu_0$, and assume that $0 < \text{Var}(X_i) < \infty$. Then

$$-2 \log(\mathcal{R}(\mu_0)) \xrightarrow{d} X^2_1$$ as $n \rightarrow \infty$.

Proof can be found in Owen[39].

The following example clarifies the above theorem.
Example 3.1. Let us consider the profile empirical likelihood ratio for the mean. Suppose, we have sample data of a bounded random variable \( \{X_1, \ldots, X_n\} \). Let \( w_i \) be the weights that CDF places on observation \( X_i \). Assume that \( \sum_{i=1}^{n} w_i = 1 \). If \( \sum_{i=1}^{n} w_i < 1 \), then \( F \) puts probability \( 1 - \sum_{i=1}^{n} w_i \) on the bounded interval without sample data. In this case the weights need to be reassigned, so \( \sum_{i=1}^{n} w_i = 1 \).

The profile empirical likelihood ratio for the mean can be written as follows,

\[
R(\mu) = \max \left\{ \prod_{i=1}^{n} n w_i \mid \sum_{i=1}^{n} w_i X_i = \mu, \ w_i \geq 0, \ \sum_{i=1}^{n} w_i = 1 \right\}. \tag{3.1}
\]

Confidence region can be constructed using the given threshold \( r_0 \) in a similar to parametric approach way,

\[
\{ \mu \mid R(\mu) \geq r_0 \} = \left\{ \sum_{i=1}^{n} w_i X_i \mid \prod_{i=1}^{n} n w_i \geq r_0, \ w_i \geq 0, \ \sum_{i=1}^{n} w_i = 1 \right\}.
\]

3.3 Estimating Equations

Owen [39] showed the way to incorporate estimating equations into the empirical likelihood theory. Let \( X \in \mathbb{R}^d \) be a random variable, and \( \theta \in \mathbb{R}^p \) be a parameter of interest. A vector valued function \( m(X, \theta) \in \mathbb{R}^s \) is called an estimating function. If \( E[m(X, \theta)] = 0 \), then the equation \( \frac{1}{n} \sum_{i=1}^{n} m(X_i, \hat{\theta}) = 0 \) is called the estimating equation. The true parameter can be estimated by solving estimating equation for \( \hat{\theta} \). For example, estimating function for the mean is \( m(X, \mu) = X - \mu \). Now, the first constraint from equation 3.1 may be modified.

\[
\sum_{i=1}^{n} w_i X_i = \mu \iff \sum_{i=1}^{n} w_i (X_i - \mu) = 0 \iff \sum_{i=1}^{n} w_i m(X_i, \mu) = 0,
\]

where \( \sum_{i=1}^{n} w_i = 1 \). Then, profile empirical likelihood ratio for the mean can be rewritten,

\[
R(\mu) = \max \left\{ \prod_{i=1}^{n} n w_i \mid \sum_{i=1}^{n} w_i m(X_i, \mu) = 0, \ w_i \geq 0, \ \sum_{i=1}^{n} w_i = 1 \right\}.
\]
The profile empirical likelihood ratio for $\theta$ is defined by

$$R(\theta) = \max \left\{ \prod_{i=1}^{n} n w_i | \sum_{i=1}^{n} w_i m(X_i, \theta) = 0, \ w_i \geq 0, \ \sum_{i=1}^{n} w_i = 1 \right\}. \quad (3.2)$$

Like the profile empirical likelihood ratio theorem for the mean (Theorem 3.2.1), the following theorem is about asymptotic property of equation 3.2.

**Theorem 3.3.1.** Let $X_1, \ldots, X_n \in \mathbb{R}^d$ be independent random vectors with common distribution $F_0$. Let $\theta \in \Theta \subseteq \mathbb{R}^p$ be the parameter of interest. For $\theta \in \mathbb{R}^p$ and $X \in \mathbb{R}^d$, let $m(X, \theta) \in \mathbb{R}^s$ be an estimating function. Suppose, $\theta_0 \in \Theta$ such that $\text{Var}(m(X_i, \theta_0)) < \infty$ and has a rank of $q > 0$. If $\theta_0$ satisfies $E(m(X, \theta_0)) = 0$, then

$$-2 \log R(\theta_0) \overset{d}{\to} \chi^2_q \text{ as } n \to \infty.$$ 

According to [53], this theorem is the consequence of the generalized version of Theorem 3.2.1.

### 3.4 Computation of Empirical Likelihood

In this section, we will consider the computation of empirical likelihood for different parameters in order to be able to do parameter estimation, setting hypothesis tests and confidence regions. Suppose, we have the data sample $\{X_1, \ldots, X_n\}$, where $X_i \in \mathbb{R}^d$, $\theta$ is the parameter of interest, for simplicity, let $\theta \in \mathbb{R}^d$. Therefore, the goal is to find $R(\theta)$, which means to maximize

$$\prod_{i} n w_i \quad \text{or} \quad \sum_{i=1}^{n} \log(n w_i)$$

over the convex space of vectors under constraints:

$$\sum_{i=1}^{n} w_i m(X_i, \theta) = 0, \ \sum_{i=1}^{n} w_i = 1, \ \text{and} \ w_i \geq 0. \quad (3.3)$$
In practice, it is easier to work with the logarithm transformation. It is legal since we are maximizing a monotonous function over the convex set \( \{(w_1, \ldots, w_n) | \sum_{i=1}^{n} w_i = 1, \ w_i \geq 0 \} \). In this case, the global maximum exists.

The method of Lagrange Multipliers is used to work out this problem. The Lagrange Multipliers are \( \lambda \in \mathbb{R}^d \) and \( \gamma \). Consider the Lagrangian \( G \):

\[
G = \sum_{i=1}^{n} \log(nw_i) - n\lambda \left( \sum_{i=1}^{n} w_i m(X_i, \theta) \right) + \gamma \left( \sum_{i=1}^{n} w_i - 1 \right).
\]

Here we have \( d+1 \) multipliers. One multiplier \( \gamma \) is for the constraint \( \sum_{i=1}^{n} w_i = 1 \), and \( d \) multipliers of \( \lambda \) for \( d \) constraints from the estimating equation, which is a function with the range from \( \mathbb{R}^d \).

The first step is to find the first partial derivatives of \( G \) with respect to \( w_i \) and set it to zero,

\[
\frac{\partial G}{\partial w_i} = \frac{1}{w_i} - n\lambda \frac{m(X_i, \theta)}{1 + \lambda m(X_i, \theta)} + \gamma = 0.
\] (3.4)

Next, multiply every part by \( w_i \),

\[
w_i \frac{\partial G}{\partial w_i} = 1 - n\lambda \frac{m(X_i, \theta)}{1 + \lambda m(X_i, \theta)} + w_i \gamma = 0.
\]

Then, take the sum over \( i \) and apply the constraints 3.3,

\[
\sum_{i=1}^{n} w_i \frac{\partial G}{\partial w_i} = n + \gamma = 0.
\]

This implies that \( \gamma = -n \). Now we can plug in this result in 3.4 and using the second equality solve for \( w_i \),

\[
w_i = \frac{1}{n} \frac{1}{1 + \lambda m(X_i, \theta)}.
\] (3.5)

\( \lambda \) can be found using the numerical methods by solving \( d \) equations, which we get after substituting the last formula for weights into the main constraint \( \sum_{i=1}^{n} w_i m(X_i, \theta) = 0 \),

\[
\frac{1}{n} \sum_{i=1}^{n} \frac{m(X_i, \theta)}{1 + \lambda m(X_i, \theta)} = 0,
\] (3.6)
subject to \( n \) inequality constraints for \( \lambda \)

\[ 1 + \lambda^\prime m(X_i, \theta) > 0, \quad (3.7) \]

since the weights are nonnegative, \( w_i \geq 0 \).

In computations, it is not always possible to find the vector \( \lambda \). The problem arises when \( 1 + \lambda^\prime m(X_i, \theta) \to 0 \). Owen proposed a solution for this case.

We know,

\[ \log R(F) = \log \left( \prod_{i=1}^{n} n w_i \right) = \sum_{i=1}^{n} \log(n w_i). \]

Using the equation 3.5 for weights, we get

\[ \log R(F) = \sum_{i=1}^{n} \log \left( \frac{1}{1 + \lambda^\prime m(X_i, \theta)} \right). \]

Simplifying,

\[ \log R(F) = - \sum_{i=1}^{n} \log \left( 1 + \lambda^\prime m(X_i, \theta) \right) \equiv \mathbb{L}(\lambda). \]

Now, instead of maximizing the product of the weights, we look for a minimum of the function \( \mathbb{L} \) over \( \lambda \). If we take the first derivative of \( \mathbb{L} \) with the respect to \( \lambda \), we get

\[ - \sum_{i=1}^{n} \frac{m(X_i, \theta)}{1 + \lambda^\prime m(X_i, \theta)} = 0, \]

which is equivalent to the equation 3.6. First, we had a maximization problem over \( n \) variables \( w_i \) subject to \( d \) equality constraints after we eliminated the multiplier \( \gamma \). Now, we have a minimization question over \( d \) variables \( \lambda \) subject to \( n \) inequality constraints 3.7. This is an example of convex duality.

If \( d \) is much less than \( n \), this interchange can decrease the computation time. Unfortunately, the issue of vanishing the denominator in 3.6 remains. Owen’s recommendation is to define the pseudo-logarithm function,

\[ \log^*(z) = \begin{cases} 
\log(z), & \text{if } z \geq 1/n \\
\log(1/n) - 1.5 + 2nz - (nz)^2/2, & \text{if } z \leq 1/n.
\end{cases} \quad (3.8) \]
This allows to redefine $L$ so that it is convex over all of $\mathbb{R}^d$, the value near solution stays the same, and $n$ constraints 3.7 don’t have to be used. The advantage of the equation 3.8 is that when the argument is less than $1/n$ it corresponds to $w_i$ which is greater than 1, so it is not a valid solution.

Define new function $L_*$ by changing log function in $L$ with $\log_*$,

$$L_*(\lambda) = -\sum_{i=1}^{n} \log_*(1 + \lambda' m(X_i, \theta)).$$

Instead of minimizing $L$ over $\lambda$ subject to $n$ inequality constraints, we can minimize $L_*$ over $\lambda$ without any constraints. The value of $L_*$ matches $L$ at any $\lambda$ satisfying the inequality constraints, including the solution if it exists.

Take the first derivative of $L_*$ with respect to $\lambda$ and set it to zero,

$$-\sum_{i=1}^{n} \log_*(1 + \lambda' m(X_i, \theta)) \cdot m(X_i, \theta) = 0,$$

where

$$\log'_*(z) = \begin{cases} 1/z, & \text{if } z \geq 1/n \\ 2n - n^2z, & \text{if } z \leq 1/n, \end{cases}$$

thus, avoiding the problem of a vanishing denominator in (21).

### 3.5 Accuracy and Power of Empirical Likelihood

Theorem 3.2.1 suggests that the value of the mean $\mu_0$ should be rejected at the $\alpha$ level if

$$-2 \log \mathcal{R}(\mu_0) > \chi^2_{(1)}^{2,1-\alpha}.$$

In simulations, for accuracy, $F_{1,n-1}^{1,\alpha}$ cut off is better to use instead of $\chi^2_{(1)}^{2,1-\alpha}$ as have been noticed in [8]. Since $F_{1,n-1}$ distribution is the square of a $t_{n-1}$ distribution and $\chi^2_{(1)}$ distribution is the standard normal distribution $N(0, 1)$ squared. As the sample size $n$ approaches infinity, $t_{n-1}$ distribution gets close to $N(0, 1)$ and $F_{1,n-1}$ distribution reaches $\chi^2_{(1)}$. For large $n$ there is no difference what tolerance level to use, but for small samples
the results are more accurate with $F_{1,n-1}^{1-\alpha}$ threshold.

Empirical likelihood confidence intervals are asymptotic confidence intervals. For example, for the mean it is constructed in such way that all values of the mean $\mu$ such that

$$-2 \log \mathcal{R}(\mu) \leq \chi^2_{(1)}^{2,1-\alpha}$$

are taken into in the interval. We would like the coverage to be $1 - \alpha$ for any sample size $n$, where $\alpha$ is a significance level, but here we have it only for large $n$,

$$Pr(-2 \log \mathcal{R}(\mu) \leq \chi^2_{(1)}^{2,1-\alpha}) \to 1 - \alpha,$$

as $n \to \infty$. The coverage accuracy gets closer to zero as sample size increases. The rate of convergence of it to zero when sample size approaches infinity as stated in [8] is $1/n$. Such rate of convergence is the same as in most parametric cases.

High accuracy and the good power of the test is a trade off problem. We would like to have high power and a confidence interval not too large at the same time. One of the methods to evaluate the power of empirical likelihood is to calculate curvature of the profile likelihood ratio $\mathcal{R}$ at the nonparametric maximum likelihood estimator. For the mean $\mu$ near $\bar{X}$, large $n$, and where $\text{Var}(X_i) = \sigma^2$,

$$-2 \log \mathcal{R}(\mu) = \left( \frac{\mu - \bar{X}}{\sigma/\sqrt{n}} \right)^2.$$

Large values of the curvature corresponds to shorter intervals for a given coverage level, and therefore the high power. It is known that

$$-2 \log \mathcal{R}(\mu_0 + \tau \sigma/\sqrt{n}) \to \chi^2_{(1)}(\tau^2),$$

in distribution, where $\tau^2$ is a noncentrality parameter. This shows that the powers of parametric inferences and empirical likelihood inferences are similar.
3.6 Empirical Likelihood with Inequality Constraint

Empirical likelihood as have been seen in previous sections is a maximization problem with equality constraint. In general, maximization problem can be solved with equality and inequality constraints. For example, Moon and Schorfheide [37] in theirs work considered empirical likelihood calculations with inequality constraints.

Suppose, we need to maximize $f(x)$ subject to constraints:

$$g_1(x) = a_1, \ldots, g_m(x) = a_m,$$

and

$$h_1(x) \leq b_1, \ldots, h_k(x) \leq b_k.$$

The corresponding Lagrangian with coefficients $\lambda^*_1, \ldots, \lambda^*_m, \nu^*_1, \ldots, \nu^*_k$ is

$$L(x, \lambda, \nu) = f(x) + \sum_{i=1}^{m} \lambda_i (a_i - g_i(x)) + \sum_{j=1}^{k} \nu_j (b_j - h_j(x)).$$

The main result from [49] is as follows. Assume $x^* = (x^*_1, \ldots, x^*_n)$ maximizes $f(x)$ subject to the constraints $g_i(x) = a_i$, for $i = 1, \ldots, m$ and $h_j(x) = b_j$, for $j = 1, \ldots, k$. Then either

1) the vectors $\nabla g_1(x^*), \ldots, \nabla g_m(x^*), \nabla h_1(x^*), \ldots, \nabla h_k(x^*)$ are linearly dependent, or

2) there exists vectors $\lambda^* = (\lambda^*_1, \ldots, \lambda^*_m)$ and $\nu^* = (\nu^*_1, \ldots, \nu^*_k)$ such that

$$\nabla f(x^*) - \sum_{i=1}^{m} \lambda^*_i \nabla g_i(x^*) - \sum_{j=1}^{k} \nu^*_j \nabla h_j(x^*) = 0$$

that is

$$\nu^*_j (h_j(x^*) - b_j) = 0,$$ where $\nu^*_j \geq 0.$
Notice, that from the last equation we have either $\nu^*_j = 0$ or $h_j(x^*) - b_j = 0$. These conditions are called Karush-Kuhn-Tucker conditions [6]. In practice, several candidates solutions can be found from them, and if there is an optimal solution, then one of the candidates will be the desirable solution. The following theorem from [2] shows why it works.

**Theorem 3.6.1.** Suppose that $x^* = (x_1^*, ..., x_n^*)$ is a maximum of the Lagrangian for $f(x)$ with non-negative coefficients $\lambda_1^*, ..., \lambda_m^*, \nu_1^*, ..., \nu_k^*$. $x^*$ is admissible for the maximization of $f(x)$ with equality and inequality constraints: $g_1(x) = a_1, ..., g_m(x) = a_m$ and $h_1(x) \leq b_1, ..., h_k(x) \leq b_k$. When the inequality constraint $i$ does not tight at $x^*$, meaning that $h_i(x^*) < b_i$, the corresponding Lagrange coefficient is zero. Then $x^*$ is a solution for the maximization of $f(x)$.

**Proof.** The corresponding Lagrangian for this problem is:

$$L(x, \lambda, \nu) = f(x) + \sum_{i=1}^{m} \lambda_i (a_i - g_i(x)) + \sum_{j=1}^{k} \nu_j (b_j - h_j(x)).$$

Lagrange coefficients are non-negative, and for any $x = (x_1, ..., x_n)$, for which the constraints are not violated, we have:

$$f(x) \leq L(x, \lambda, \nu).$$

Suppose, all constraints are satisfied at $x^*$. If the inequality constraint does not tight, then the corresponding coefficient is zero. If the inequality constraint is tight for some $j$, then $\nu_j(b_j - h_j(x^*)) = 0$. Thus, for all $j$ we have,

$$\nu_j(b_j - h_j(x^*)) = 0.$$

Therefore, at $x^*$

$$f(x^*) = L(X^*, \lambda, \nu).$$

Since $x^*$ is the maximum for Lagrangian, then for all admissible $x$

$$L(x, \lambda, \nu) \leq L(X^*, \lambda, \nu).$$
Combining all together,

\[ f(x) \leq L(x, \lambda, \nu) \leq L(X^*, \lambda, \nu) = f(x^*), \]

exactly what need to be shown.

The following example clarifies above discussion.

**Example 3.2.** Suppose we need to maximize \( f(x) = x^3 - 3x \) subject to \( x \leq 2 \). The corresponding Lagrangian is

\[ L(x, \nu) = x^3 - 3x + \nu(2 - x). \]

First derivative, constraint and Karush-Kuhn-Tucker conditions are

\[ \frac{dL}{dx} = 3x^2 - 3 - \nu = 0, \]
\[ x \leq 2, \]
\[ \nu(2 - x) = 0, \]
\[ \nu \geq 0. \]

There are two cases to consider. Case 1: let \( x < 2 \), then \( \nu = 0 \) and

\[ 3x^2 - 3 = 0, \]

which gives us \( x = \pm 1 \). Both solutions satisfy the constraint. Case 2: let \( x = 2 \), then

\[ 3 \cdot 2^2 - 3 - \nu = 0, \]

give us \( \nu = 9 \), which satisfy the assumption that \( \nu \geq 0 \). Now we have,

\[ f(-1) = 2, \quad f(1) = -2, \quad f(2) = 2. \]

Thus, there are two solutions of -1 and 2.
For our maximization problem of profile likelihood ratio for the parameter $\theta$ when we change equality constraint with inequality we have,

$$R(\theta) = \max \left\{ \prod_{i=1}^{n} w_i \mid \sum_{i=1}^{n} w_i m(X_i, \theta) \leq 0, w_i \geq 0, \sum_{i=1}^{n} w_i = 1 \right\}.$$  

If the inequality constraint is not tight, it can be ignored and the corresponding coefficient in the Lagrangian will be zero. Then the Lagrangian reaches maximum when all the weights will be the same. If the inequality constraint is tight, then it can be treated as equality constraint.

### 3.7 Empirical Likelihood for Censored Data

The data is censored when some observations are only partially known. There are different types of censored data: interval censoring, left, and right censoring. The most common form is the right censored data, which occurs when a subject leaves the study before an event occurs, or the study ends before the event has occurred. The right censoring is considered here. The definition, likelihood function, calculations for empirical likelihood ratio, and necessary theorems and definitions for right censoring data described below are taken from Owen’s book [39].

**Definition 3.7.1.** $X_i$ is a real-valued random variable. For each $X_i$ there is a $Y_i \in \mathbb{R}$. $Y_i$ may be random. If $X_i \leq Y_i$, we observe $X_i$, otherwise $X_i$ is censored to $(Y_i, \infty)$. We say that $X_i$ is right censored by $Y_i$.

Suppose, that $X_1, \ldots, X_n$ is a sample from cumulative distribution function $F$, and $Y_1, \ldots, Y_n$ are right censoring times. Assume that $X_1, \ldots, X_n$ are conditionally independent when the censoring times are known. Let $Z_i = \min(X_i, Y_i)$ and $\delta_i = 1_{X_i \leq Y_i}$ indicate an uncensored event when $\delta_i = 1$, $i = 1, \ldots, n$. Notice, that if $X_i = Y_i$, then the $X_i$ is not censored.

Let $\mathcal{X} = (X_1, \ldots, X_n)$ and $\mathcal{Y} = (Y_1, \ldots, Y_n)$. The likelihood for $F$ and $G$ for right-censored data is the product of a marginal distribution and conditional likelihood function,

$$L(F,G; \mathcal{X}, \mathcal{Y}) = L(F,G; \mathcal{Y}) \times L(F,G; \mathcal{X} | \mathcal{Y}),$$
where

\[ L(F, G; \mathcal{Y}) = G(Y_1, \ldots, Y_n) \]

and

\[ L(F, G; \mathcal{X} | \mathcal{Y}) = \prod_{i: \delta_i = 1} F(\{X_i\}) \prod_{i: \delta_i = 0} F((Y_i, \infty)) = \prod_{i=1}^n F(\{Z_i\}) F((Z_i, \infty))^{1-\delta_i}. \]  

(3.9)

Any factor of \(0^0\) is assumed to be 1.

Usually, inferences for \(F\) are based on conditional likelihood function 3.9, which doesn’t depend of \(G\). Such conditional likelihood can be found from the \(Z_i\) and censoring index \(\delta_i\). The knowledge about censoring time \(Y_i\) for uncensored \(X_i\) is not required. There is no loss of information since the marginal distribution of \(Y_i\) doesn’t depend on \(F\). The likelihood ratio for \(F\) will be the same using full likelihood or conditional one assuming that there are no any dependencies between \(F\) and \(G\).

There is a nonparametric maximum likelihood estimator (NPMLE) for right censored data. It is easier to introduce it using the hazard function and the cumulative distribution function.

It is known that the survival function [24] is

\[ S(t) = F([t, \infty]) = 1 - F((-\infty, t)). \]

In some cases, when the interest is in the proportion of survived subjects, it is more natural to use survival function rather than cumulative distribution function. The hazard function, also known as the failure rate, is

\[ \lambda(t) = \lim_{h \to 0^+} \frac{1}{h} Pr(X \leq t + h | X \geq t) = \frac{f(t)}{S(t^+)} = \frac{f(t)}{S(t)}, \]

where \(f(t)\) is a density function. Given that subject survived to at least time \(t\), \(\lambda(t)dt\) is the probability of failure before time \(t + dt\). For continuous data the hazard function may be rewritten

\[ \lambda(t) = \frac{d \log(S(t))}{dt}, \]
and survival function can be written as

\[ S(t) = \exp\left(- \int_0^t \lambda(u)du\right). \]

For discrete data with \( F(\{t_j\}) > 0 \), for a finite or countably infinite number of \( t_j \) according to [23], the hazard function may be rewritten as

\[ \lambda_j = \Pr(X = t_j \mid X \geq t_j) = \frac{F(\{t_j\})}{S(t_j)}, \]

survival function is

\[ S(t) = \prod_{t_j \leq t} (1 - \lambda_j), \]

and distribution function can written as

\[ F(\{t_j\}) = \lambda_j \prod_{t_j < t} (1 - \lambda_j), \]

and

\[ F((-\infty, t]) = 1 - \prod_{t_j \leq t} (1 - \lambda_j). \]

In general, the cumulative hazard is

\[ \Lambda(t) = \int_0^t \frac{dF(u)}{F((-\infty, u))}. \]

For continuous distributions,

\[ \Lambda(t) = \int_0^t \lambda(u)du, \]

and for discrete distributions,

\[ \Lambda(t) = \sum_{t_j \leq t} \lambda_j. \]

Let the observed failure times be \( t_1 < t_2 < \cdots < t_k \). Denote \( \hat{F} \) to be NPMLE for \( F \). Assume that \( t_0 = 0 \) and \( t_1 > t_0, t_{k+} = \infty \). Let \( d_j \geq 1 \) be the number of failures at \( t_j \) and \( m_j \) be the number censored observations
in the interval $[t_j, t_{j+1})$. The NPMLE assigns the probability of 0 for values in the interval $(t_j, t_{j+1})$, where $j < k$. The number $n_j$ is the number of subjects at risk right before time $t_j$,

$$n_j = (d_j + m_j) + \cdots + (d_k + m_k).$$

If $\lambda_j = Pr(X = t_j \mid X \geq t_j)$ are the hazard probabilities of the distribution $F$, then the conditional likelihood 3.9 can by written

$$L_c(F) = \prod_{j=1}^{k} \lambda_j^{d_j} (1 - \lambda_j)^{n_j - d_j},$$

and the NPMLE is

$$\hat{\lambda}_j = \frac{d_j}{n_j}.$$

Then the cumulative distribution function of the NPMLE is

$$F((-\infty, t]) = 1 - \prod_{j: t_j \leq t} \left(1 - \frac{d_j}{n_j}\right),$$

which is well known Kaplan-Meier estimator. The NPMLE is unique if the biggest observed failure time is larger than the biggest observed censored time.

Suppose we are interested in $S(t)$ at a specific time $t$. The profile empirical likelihood function for $S(t)$ is defined by

$$\mathcal{R}(s, t) = \max \left\{ \prod_{j=1}^{k} \frac{\lambda_j^{d_j} (1 - \lambda_j)^{n_j - d_j}}{\hat{\lambda}_j^{d_j} (1 - \hat{\lambda}_j)^{n_j - d_j}} \mid 0 \leq \lambda_j \leq 1, \prod_{t_j \leq t} (1 - \lambda_j) = s \right\}.$$

In order to maximize likelihood ratio, generally Lagrange multipliers technique is applied, and the $\lambda_j$ must be

$$\lambda_j = \frac{d_j}{n_j + \gamma 1_{t_j \leq t}},$$

where $\gamma$ is a Lagrange multiplier and can be found from the following equation

$$\sum_{j: t_j \leq t} \log \left(\frac{n_j - d_j - \gamma}{n_j + \gamma}\right) - \log(s) = 0.$$
The following theorem tells about the asymptotic behavior of the profile empirical likelihood function for $S(t)$ at the specific time.

**Theorem 3.7.1.** For $i = 1, \ldots, n$, let $X_i, Y_i \in \mathbb{R}$ be independent random variables. $X_i$ follows cumulative distribution $F$ and $Y_i$ follows cumulative distribution $G$. Let $Z_i = \min(X_i, Y_i)$ be observed random variable and $\delta_i = 1_{X_i \leq Y_i}$ be censoring index for $i = 1, \ldots, n$. Assume that $G((-\infty, t)) < 1$ and $0 < S(t) < 1$. Then

$$-2\log \mathcal{H}(S(t), t) \xrightarrow{d} \chi^2_1,$$

as $n \to \infty$.

The proof of this theorem may be found in Li [29].

Adimari [1] studied empirical likelihood inferences for the mean of a distribution for the right censoring data under the assumptions that exact failure times are independent of censoring times and the largest observed value is not censored. He proposed to use the following function

$$l_\mu(\mu) = 2n \sum_{i=1}^n \tilde{p}_i \log \left\{ 1 + \lambda (Z_i - \mu) \right\},$$

where $Z_i$ for $i = 1, \ldots, n$ is a right censored data, $\tilde{p}_i$ for $i = 1, \ldots, n$ are probabilities of the observe failure times assigned by Kaplan-Meier distribution estimator, and $\lambda$ can be found from the equation

$$\sum_{i=1}^n \frac{\tilde{p}_i(Z_i - \mu)}{1 + \lambda (Z_i - \mu)} = 0.$$  

Adimari showed that under some conditions and suitable correction $l_\mu$ approaches $\chi^2$ distribution as $n \to \infty$.

Pan and Zhou in [41] showed that the empirical likelihood ratio with mean constraint for right censored data follows $\chi^2$ distribution when the sample size approaches infinity.

**Theorem 3.7.2.** For $i = 1, \ldots, n$, let $X_i, Y_i \in \mathbb{R}$ be independent random variables. $X_i$’s are exact failure times follows continuous cumulative distribution $F$ and $Y_i$’s are censoring times follows cumulative distribution $G$. Let $Z_i = \min(X_i, Y_i)$ be observed random variable and $\delta_i = 1_{X_i \leq Y_i}$ be censoring index for $i = 1, \ldots, n$. Assume that there is a constraint

$$\int g(t) dF(t) = \theta_0,$$
where $\theta_0$ is the true value and $g(t)$ satisfies regularity conditions. Then, the empirical likelihood ratio for $\theta_0$

$$-2\log \mathcal{L}(\theta_0) \xrightarrow{d} \chi^2_{(1)}, \text{ as } n \to \infty.$$
CHAPTER 4

ABC WITH EMPIRICAL LIKELIHOOD

Mengersen, Pudlo, and Robert [36] proposed a new approach in Approximate Bayesian Computations using the empirical likelihood. The idea is to use importance sampling within ABC, but instead of importance weights they employed the empirical likelihood weights. Here, the parameter estimation is done in the same way as with the importance sampling methods. This method does not require any model simulations within an algorithm, thus, it can work faster and enlarge the range of the complex model for which it may be used, especially in population genetics. The disadvantage of this algorithm is the same as with any importance sampling techniques, large variance of the weights, and sensitivity to the prior information about the parameter. In this chapter, existing ABC via empirical likelihood methods are described as well as an improved algorithm is proposed. Improved ABC via empirical likelihood allows to obtain more accurate parameter estimates, and it is less sensitive to the prior knowledge of the parameter. This can be achieved through the introduction of hypothesis testing procedure for more careful selection of parameter candidates before assigning weights.

4.1 Bayesian Inference and Empirical Likelihood

Likelihood function is essential to Bayesian statistics. Lazar in [28] explored different ways of using empirical likelihood for Bayesian inference and researched the validity of them. One of her recommendations is to use profile empirical likelihood ratio function $R(\theta)$ instead of likelihood function in the Bayes’ theorem. Then, the posterior distribution can be written as follows,

$$
\pi_{el}(\theta|x) = \frac{\pi(\theta)R(\theta)}{\int \pi(\theta)R(\theta)d\theta},
$$

where $\theta$ is the parameter of interest, and $x$ is the set of the known observations. Now, the question arises if the inference from such posterior is the valid one. To show it Lazar introduced a definition of validity built
on the coverage properties of posterior sets, and the numerical method to check the validity of likelihoods. From the Bayesian theory, if the prior distribution and the model are chosen correctly, then the sets of particular amounts of posterior probability will have the right probability of the parameter being in this set. The following definitions are taken from Lazar [28]:

**Definition 4.1.1.** A posterior density formed on empirical likelihood $L_{el}$ is valid by coverage for the model $f(x|\theta_{el})$ if and only if $Pr(\theta \in S_\alpha(x)) = \alpha$ for every $S_\alpha(x)$, a posterior coverage set function of level $\alpha$, under the measure $\pi(\theta_{el}) f(x|\theta_{el})$.

**Definition 4.1.2.** The empirical likelihood $L_{el}$ is valid by coverage if and only if the posterior distribution $\pi_{el}(\theta_{el}|x)$ is valid by coverage for every absolutely continuous prior.

The numerical procedure to check validity of empirical likelihood proposed by Lazar for one dimensional case is to consider the function $H(\theta)$,

$$H(\theta) = \int_{-\infty}^{\theta} \pi_{el}(\theta_{el}|x).$$

Then, in this case, the posterior coverage set function is,

$$S_\alpha(x) = (-\infty, \theta_{el}^\alpha],$$

where $\theta_{el}^\alpha$ is the $\alpha$-th percentile of the posterior $\pi_{el}(\theta_{el}|x)$. If the posterior distribution based on empirical likelihood is valid, then $H(\theta)$ should be distributed uniformly over the interval $(0,1)$. In practice, as shown in [20], for a sample size at least 50 with a uniform prior for the mean, the distribution of $H(\theta)$ is very close to the uniform. Thus, empirical likelihood with a uniform prior for sample size at least 50 provides a valid Bayesian inference.

However, Lazar mentioned an example when it is not beneficial to use empirical likelihood instead of the usual likelihood. Thus, the validity of the posterior inference has to be verified for every case separately.
4.2 ABC via Empirical Likelihood Sampler

Natural way to use empirical likelihood is to think about it as it is a true likelihood. The validity of this idea is shown in the previous section based on Lazar’s work [28]. In contrast to ABC rejection algorithm, for ABC via empirical likelihood simulated data set is not needed, and there is no selection process for parameter candidates. Each parameter $\theta$ simulated from the prior distribution is accepted with the weight computed as product of weights which maximize the profile empirical likelihood ratio function,

$$\mathcal{R}(\theta) = \max \prod_{i=1}^{n} nw_i,$$

over mean constraint,

$$\sum_{i=1}^{n} w_i m(X_i, \theta) = 0, \sum_{i=1}^{n} w_i = 1, w_i \geq 0 \text{ for } i = 1, \ldots, n.$$

The basic ABC via empirical likelihood as appears in [14] is

**ABC via Empirical Likelihood Sampler (ABC - EL):**

1. Set $j=1$.
2. Generate a parameter candidate $\theta_j^*$ from the prior $\pi(\theta)$.
3. Compute $w_i$, which come from profile empirical likelihood ratio:

   $$\mathcal{R}(\theta_j^*) = \max \left\{ \prod_{i=1}^{n} nw_i | \sum_{i=1}^{n} w_i m(X_i, \theta_j^*) = 0, w_i \geq 0, \sum_{i=1}^{n} w_i = 1 \right\}.$$

4. Calculate $p_j = \prod_{i=1}^{n} w_i$.
5. If $j < J$, increment $j = j + 1$ and go to step 2.
6. Normalize the weights, $p_j^* = \frac{p_j}{\sum_{j=1}^{J} p_j}$ for $j = 1, \ldots, J$.
7. Take $\theta_j^*$ with corresponding weight $p_j^*$. 
This algorithm is likelihood-free, since it is based on empirical likelihood ratio function. In chapter three it has been shown that empirical likelihood is a well behaving approximation of the parametric likelihood. Therefore, this algorithm can be classified as an ABC algorithm.

4.3 ABC via Empirical Likelihood with Resampling

Panek in [42] introduced an improved ABC algorithm via empirical likelihood by applying importance sample resampling method for decreasing the variance of the weights. Due to this technique, the improved ABC algorithm via empirical likelihood (the author refers to his algorithm as Empirical likelihood weighted resampling sampler) is more efficient and the parameter estimates are more accurate.

The problem with ABC via empirical likelihood algorithm is that large values of weights get a small number of parameter values, and a lot of parameters get very small weight. This is the situation when the variance of weights may be very large. Similar problem arises with the algorithms which contain the importance sample technique. The solution is the resampling.

Suppose we have \( J \) parameters \( \theta_1, \ldots, \theta_J \) and \( J \) corresponding weights \( w_1, \ldots, w_J \). The goal is to concentrate the attention on the parameters with large weights and lessen the number of the parameter values with small weights. It can be done by drawing \( J \) samples from \( (\theta_1, \ldots, \theta_J) \) according to theirs weights. New sample values get the same weights. Panek’s idea is to modify the new sample values by adding a Gaussian white noise with zero mean and some small variance. Then, the corresponding weights as a product of empirical likelihood weights should be calculated. It is not necessary to do the resampling at every iteration of the algorithm, only when the weight’s variance is too large. The common measure to determine when resampling need to be done is the effective sample size,

\[
ESS = \frac{1}{\sum_{j=1}^{J} (p_j^*)^2 },
\]

where \( p_j^* \) is the normalized weight as introduced in the previous section. Effective sample size varies from 1 (degenerated sample, when only one weight equals to one and all other weights are zero) to \( J \) (uniform
sample, all the weights are the same and equal to $1/J$. The common thresholds for effective sample size is $J/2$.

The empirical likelihood weighted resampling sampler proposed by Panek goes as follows,

**ABC via Empirical Likelihood Resampling Sampler:**

1. For $j = 1, \ldots, J$ get $\theta_j^*$ with corresponding weight $p_j^*$ by ABC - EL.
2. Calculate $\text{ESS} = 1/\sum_{j=1}^{J} (p_j^*)^2$.
   While $\text{ESS} < E$, where $E$ is the threshold.
   1. Set $j=1$.
   2. Sample $\theta^{**}$ from $\{\theta_j^*\}_{j=1}^{J}$ with weights $\{p_j^*\}_{j=1}^{J}$.
   3. Set $\theta_j = \theta^{**} + e$, where $e \sim N(0, \sigma^2)$.
   4. Compute $w_i$'s, which come from profile empirical likelihood ratio:

   $R(\theta_j) = \max \left\{ \prod_{i=1}^{n} w_{ij} \left| \sum_{i=1}^{n} w_i m(X_i, \theta_j) = 0, w_j \geq 0, \sum_{i=1}^{n} w_i = 1 \right. \right\}$.

2.5 Calculate $h_j = \prod_{i=1}^{n} w_i$.
2.6 If $j < J$, increment $j = j + 1$ and go to step 2.2.
2.7 Normalize the weights, $h_j^* = \frac{h_j}{\sum_{j=1}^{J} h_j}$ for $j = 1, \ldots, J$.
2.8 Set $\theta_j^* = \theta_j$ and $p_j^* = h_j$ for $j = 1, \ldots, J$.

According to Panek [42], the reason to add the normally distributed noise is to avoid duplicate weighted samples since we sample values out of $J$ with replacement.

### 4.4 ABC via Empirical Likelihood with Testing

In this section a new improvement of ABC via empirical likelihood by adding hypothesis testing procedure rather than resampling is proposed. The test allows to incorporate a selection process for parameter candidates randomly chosen form the prior distribution within ABC via empirical likelihood method. The
null hypothesis for the test is that the parameter candidate generated from the prior is equal to the true value of the parameter. According to Theorem 3 in section 3.3, it is known that under some conditions the asymptotic distribution of the logarithm of profile empirical likelihood ratio is proportional to chi-squared distribution, that is for parameter $\theta$

$$-2 \log \mathcal{P}(\theta) \overset{d}{\rightarrow} \chi^2_q \text{ as } n \rightarrow \infty,$$

where $q$ is the degrees of freedom, see theorem 3 in section 3.3 how to determine $q$. Therefore, when the test statistic is small, the null hypothesis cannot be rejected, and such parameter candidate with the weight calculated via empirical likelihood tools will get accepted into the posterior sample. As a result, the addition of the hypothesis testing procedure allows to obtain more accurate parameter estimates than using original ABC via empirical likelihood or ABC via empirical likelihood with resampling.

A moderate level of significance $\alpha$ for the test is recommended to be chosen, so it is a little higher than usual. The reason for it is that it is beneficial to increase type I error, which is rejecting sometimes suitable parameter candidates, and decrease type II error, which is not rejecting wrong parameter candidates, since type II error is more dangerous here. In simulation, it has been noticed that too high values of $\alpha$ lead the parameter estimate to rely more on sample, and empirical likelihood weight’s effect is weakened; too low values of $\alpha$ make the parameter estimate to depend more on empirical likelihood procedures than on the sample, and the results are similar to ABC via empirical likelihood sampler described in section 4.2. Therefore, for best results, a moderate level of significance should be used. The ABC algorithm via empirical likelihood with testing goes as follows,

**ABC via Empirical Likelihood with Testing Sampler:**

1. Set $j=1$.
2. Find a parameter candidate $\theta_j^*$ by doing the following steps.
   1. Generate a parameter candidate $\theta_j^*$ from the prior $\pi(\theta)$.
   2. Evaluate p-value for $-2 \log \mathcal{P}(\theta_j^*)$ under $\chi^2$ distribution.
   3. If the p-value $< \alpha$, go to step 2.1. $\alpha$ is a predetermined level of significance.
3. Compute $w_i$, which come from profile empirical likelihood ratio:

$$R(\theta^*_j) = \max \left\{ \prod_{i=1}^{n} n w_i \mid \sum_{i=1}^{n} w_i m(X_i, \theta^*_j) = 0, \; w_i \geq 0, \; \sum_{i=1}^{n} w_i = 1 \right\}.$$ 

4. Calculate $p_j = \prod_{i=1}^{n} w_i$.

5. If $j < J$, increment $j = j + 1$ and go to step 2.

6. Normalize the weights, $p_j^* = \frac{p_j}{\sum_{j=1}^{J} p_j}$ for $j = 1, \ldots, J$.

7. Take $\theta_j^*$ with corresponding weight $p_j^*$.

The results using ABC - EL with testing algorithm are more accurate than using ABC - EL or ABC - EL with resampling. The disadvantage of this method is that the chi-squared distribution for the profile empirical likelihood ratio is asymptotic, so may not work well for small samples.

### 4.5 Comparison of ABC - EL algorithms. Examples

In this section implementation of ABC via empirical likelihood methods described in sections 4.2, 4.3, and 4.4 will be illustrated. The proposed methods are created for fully intractable models. The comparison of the methods will be done by analyzing biases and mean squared errors (MSEs). For these calculations the true parameter values are needed. So, the examples for well-known normal and exponential models are presented here. Also, three ABC - EL methods will be applied for the square lattice nanosensor model with tractable and intractable likelihood from [17].

#### 4.5.1 Normal Model

The sample consist of $n$ observations randomly taken from standard Normal distribution. The probability density function is of the form,

$$f(x) = \frac{1}{\sqrt{2\pi}} \exp \left( -\frac{x^2}{2} \right).$$
The likelihood function is

\[ L(x_1, \ldots, x_n) = \frac{1}{(\sqrt{2\pi})^n} \exp\left(-\frac{1}{2} \sum_{i=1}^{n} x_i^2\right). \]

The maximum likelihood estimator (MLE) of the mean is the sample mean,

\[ \hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} x_i. \]

Four samples of 10, 25, 25 and 50 observations were generated from standard normal distribution. Now, suppose that the distribution from which the samples were drawn is unknown and likelihood function is intractable. Let us estimate the mean using three ABC - EL methods with the same prior distribution \( \mathcal{N}(0.5, 0.7^2) \).

For ABC - EL every parameter candidate \( \theta^* \) drawn from the prior distribution \( \mathcal{N}(0.5, 0.7^2) \) is used to determine the constraint for the mean,

\[ \sum_{i=1}^{n} w_i X_i = \theta^*. \]

Then, the weights \( w_i \)'s are found using the following formula,

\[ w_i = \frac{1}{n} \frac{1}{1 + \lambda (X_i - \theta^*)}, \]

where \( i = 1, \ldots, n \), \( n \) is the sample size, which is 10 for the first sample, 25 for the second sample, 25, and 50 for the third and fourth respectively; and \( \lambda \) is determined from the following equation

\[ \frac{1}{n} \sum_{i=1}^{n} \frac{X_i - \theta^*}{1 + \lambda (X_i - \theta^*)} = 0. \]

This equation has only one solution for \( \lambda \) because the derivative of the left part is less or equal to zero,

\[ -\frac{1}{n} \sum_{i=1}^{n} \frac{(X_i - \theta^*)^2}{(1 + \lambda (X_i - \theta^*))^2} \leq 0. \]
Newton method is used to solve for $\lambda$ using the following interval as suggested by Owen in [39],

$$\frac{1 - n^{-1}}{\theta^* - X_{(n)}} < \lambda < \frac{1 - n^{-1}}{\theta^* - X_{(1)}},$$

knowing that $X_{(1)} < \theta^* < X_{(n)}$. If $\theta^*$ is outside of this interval, another candidate $\theta^*$ will be drawn from the prior distribution.

Once for a given $\theta^*$ all $w_i$’s are found, the weight for the $\theta^*$ is calculated as a product of all normalized $w_i$’s. In this fashion, 100 parameter candidates with corresponding weights are found and the weights are normalized. The ABC - EL parameter estimate is the weighted average of 100 parameter candidates.

For ABC - EL with resampling (ABC-EL resample) the same procedure is done as for ABC - EL. Once the sample of parameter candidates with corresponding weights is ready, the effective sample size (ESS) is calculated. The ESS cut off is chosen to be 90% of the parameter candidate sample. The simulations for the examples in this section show that this cut off works better than the usual one, which is half of the posterior sample. Thus, while ESS is less then 90% of the ABC posterior sample, i.e. less than 90, resample 100 values with replacement from the parameter candidate’s sample using corresponding weights as probabilities and add the normal error with mean of zero and standard deviation of 0.001. The ABC - EL with resampling estimate is the weighted average of the last set of 100 resampled values, which have ESS of at least 90.

For ABC - EL with testing (ABC-EL test) each parameter candidate $\theta^*$ drawn from prior distribution $\mathcal{N}(0.5, 0.7^2)$ is tested before the weight is assigned. First, $\theta^*$ is used in the constraint,

$$\sum_{i=1}^{n} w_i X_i = \theta^*,$$

in order to find profile empirical likelihood ratio function $R(\theta^*)$, which is

$$R(\theta^*) = \max \left\{ \prod_{i=1}^{n} n w_i \left| \sum_{i=1}^{n} w_i (X_i - \theta^*) = 0, \ w_i \geq 0, \ \sum_{i=1}^{n} w_i = 1 \right\}.$$ 

Then, the test statistic $-2 \log R(\theta^*)$ is calculated. Since it is known that, under the true mean $\theta$, $-2 \log R(\theta)$ asymptotically follows chi-squared distribution with one degree of freedom (see theorem 3.2), the p-value
is determined. ABC - EL with testing uses the level of significance of 0.25. If the p-value is greater than 0.25, then for $\theta^*$ the weight is found using the same actions as for ABC - EL; otherwise, another parameter candidate is drawn from the prior. ABC - EL with testing parameter estimate is a weighted average of 100 candidates.

The true mean is zero. The results for three algorithms are as follows in the table 4.1 which includes mean squared error (MSE) with bias on 25 runs of the algorithm.

<table>
<thead>
<tr>
<th>S. Size</th>
<th>ABC - EL MSE</th>
<th>Bias</th>
<th>ABC - EL resample MSE</th>
<th>Bias</th>
<th>ABC - EL test MSE</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.079</td>
<td>0.281</td>
<td>0.043</td>
<td>0.202</td>
<td>0.041</td>
<td>0.201</td>
</tr>
<tr>
<td>25</td>
<td>0.002</td>
<td>0.036</td>
<td>0.0006</td>
<td>0.02</td>
<td>0.0001</td>
<td>0.009</td>
</tr>
<tr>
<td>25</td>
<td>0.042</td>
<td>0.203</td>
<td>0.029</td>
<td>0.167</td>
<td>0.029</td>
<td>0.17</td>
</tr>
<tr>
<td>50</td>
<td>0.004</td>
<td>0.059</td>
<td>0.005</td>
<td>0.067</td>
<td>0.004</td>
<td>0.066</td>
</tr>
</tbody>
</table>

Table 4.1: Summary for three different ABC - EL methods for Normal model example.

For the first two samples, it can be concluded that ABC - EL with testing produces the best estimates out of three methods since it has the smallest mean squared errors and biases. For the third sample, it is shown that ABC - EL with testing and ABC-EL resampling are comparable and have more accurate estimates than ABC - EL because their mean squared errors and biases are similar, and are smaller than ABC - EL method has. For the fourth sample with 50 observations, it can be derived based on corresponding MSEs and biases that all three methods behave very similar.

### 4.5.2 Exponential Model

The sample consist of n observations randomly taken from exponential distribution. The probability density function is of the form,

$$f(x; \theta) = \frac{1}{\theta} \exp \left( -\frac{x}{\theta} \right),$$
where $x$ is positive and $\theta \in \Theta = \{ \theta | 0 < \theta < \infty \}$.

The likelihood function is

$$L(\theta; x_1, \ldots, x_n) = \frac{1}{\theta^n} \exp \left( -\frac{1}{\theta} \sum_{i=1}^{n} x_i \right).$$

The maximum likelihood estimator (MLE) of $\theta$ is the sample mean,

$$\hat{\theta} = \frac{1}{n} \sum_{i=1}^{n} x_i.$$

Three samples of 10, 25, and 50 observations were generated from exponential distribution with mean of 5. Now, suppose that the distribution from which the samples were drawn is unknown and likelihood function is intractable. Let us estimate the mean using three ABC - EL methods with the same prior distribution $\mathcal{U}(2, 6)$. For ABC - EL every parameter candidate $\theta^*$ drawn from the prior distribution $\mathcal{U}(2, 6)$ is used to determine the constraint for the mean,

$$\sum_{i=1}^{n} w_i X_i = \theta^*.$$  

Then, the weights $w_i$'s are found using the following formula,

$$w_i = \frac{1}{n} \frac{1}{1 + \lambda (X_i - \theta^*)},$$

where $i = 1, \ldots, n$, $n$ is the sample size, which is 10 for the first sample, 25 for the second sample, and 50 for the third; and $\lambda$ is determined from the following equation

$$\frac{1}{n} \sum_{i=1}^{n} \frac{X_i - \theta^*}{1 + \lambda (X_i - \theta^*)} = 0.$$

This equation has only one solution for $\lambda$ because the derivative of the left part is less or equal to zero,

$$- \frac{1}{n} \sum_{i=1}^{n} \frac{(X_i - \theta^*)^2}{(1 + \lambda (X_i - \theta^*))^2} \leq 0.$$
Newton method is used to solve for $\lambda$ using the following interval as suggested by Owen in [39],

$$\frac{1-n^{-1}}{\theta^*-X_{(n)}} < \lambda < \frac{1-n^{-1}}{\theta^*-X_{(1)}}$$

knowing that $X_{(1)} < \theta^* < X_{(n)}$. If $\theta^*$ is outside of this interval, another candidate $\theta^*$ will be drawn from the prior distribution.

Once for a given $\theta^*$ all $w_i$’s are found, the weight for the $\theta^*$ is calculated as a product of all normalized $w_i$’s. In this fashion, 100 parameter candidates with corresponding weights are found and the weights are normalized. The ABC - EL parameter estimate is the weighted average of 100 parameter candidates.

For ABC - EL with resampling the same procedure is done as for ABC - EL. Once the sample of parameter candidates with corresponding weights is ready, the effective sample size (ESS) is calculated. While ESS is less then 90, resample 100 values with replacement from the parameter candidate’s sample using corresponding weights as probabilities and add the normal error with mean of zero and standard deviation of 0.001. The ABC - EL with resampling estimate is the weighted average of of the last set of 100 resampled values, which have ESS of at least 90.

For ABC - EL with testing each parameter candidate $\theta^*$ drawn from prior distribution $\mathcal{U}(2,6)$ is tested before the weight is assigned. First, $\theta^*$ is used in the constraint,

$$\sum_{i=1}^{n} w_i X_i = \theta^*,$$

in order to find profile empirical likelihood ratio function $R(\theta^*)$, which is

$$R(\theta^*) = \max \left\{ \prod_{i=1}^{n} n w_i | \sum_{i=1}^{n} w_i (X_i - \theta^*) = 0, w_i \geq 0, \sum_{i=1}^{n} w_i = 1 \right\}.$$

Then, the test statistic $-2 \log R(\theta^*)$ is calculated. Since it is known that, under the true mean $\theta$, $-2 \log R(\theta)$ asymptotically follows chi-squared distribution with one degree of freedom (see theorem 3.2), the p-value is determined. ABC - EL with testing uses the level of significance of 0.25. If the p-value is greater than 0.25, then for $\theta^*$ the weight is found using the same actions as for ABC - EL; otherwise, another parameter
candidate is drawn from the prior. ABC - EL with testing parameter estimate is a weighted average of 100 candidates.

The true mean is 5. The results are as follows in table 4.2 which includes mean squared error (MSE) with bias on 25 runs of the algorithm with 100 iterations in it.

<table>
<thead>
<tr>
<th>Method</th>
<th>ABC - EL</th>
<th>ABC - EL resample</th>
<th>ABC - EL test</th>
</tr>
</thead>
<tbody>
<tr>
<td>S.Size</td>
<td>MSE</td>
<td>Bias</td>
<td>MSE</td>
</tr>
<tr>
<td>10</td>
<td>0.204</td>
<td>0.447</td>
<td>0.074</td>
</tr>
<tr>
<td>25</td>
<td>0.274</td>
<td>0.515</td>
<td>0.238</td>
</tr>
<tr>
<td>50</td>
<td>0.019</td>
<td>0.122</td>
<td>0.011</td>
</tr>
</tbody>
</table>

Table 4.2: Summary for three different ABC - EL methods for exponential model example.

For all three samples ABC - EL with testing produces the best estimates out of three methods; it has the smallest mean squared errors and biases.

### 4.5.3 Hydrogen Nanosensor

The square lattice nanosensor used for the example in this section is a special hydrogen gas (H$_2$) nanosensor, which is constructed from a network of ultra-small Palladium (Pd) nanowires, as shown in figure 4.1, which is described in more details in [17]. The nanosensor works based on the increasing resistance of Pd nanowires when they adsorb H$_2$. Electrical contacts are located along the left-hand and right-hand sides of the nanosensor, and the electrical current moves in one direction from one contact to

![Figure 4.1: Top-view scanning electron microscopy image of a network of Pd nanowires with a deposition thickness of 7 nanometers.](image)
another through the network. There are a lot of loosely bound electrons inside the nanowire. When an electrical force is applied on the wires opposite end, the free electrons flow in the direction of the force.

When the nanosensor is exposed to hydrogen gas, the Pd starts to adsorb the $H_2$ and increases in volume. Expanded in volume palladium nanowires prevent the electrical current to go through the nanosensor. When the hydrogen gas exits the system, the Pd decreases in volume, the particles tend to come back to the original locations, and the electrical current flows through the nanosensor as before. This process is called a cycle of $H_2$. If not all Pd particles returned to their original positions, there will be a gap. If there are so many gaps that the electrical current cannot go through the nanosensor, it becomes nonconductive, and it is considered to be broken. A high powered microscope is required to determined how many wires break and their location. This procedure is very costly and inefficient, and the network may be destroyed in the process.

The square lattice is a good approximation for the structure of the nanosensor as stated in [18]. That is, we can think of a nanosensor as a square lattice with $n$ columns of vertical wires, and $m$ rows of horizontal wires connected together with $n \cdot m$ nanobonds. There are $n \cdot (m - 1) + m \cdot (n - 1)$ nanowires in this model.

On figure 4.2 there is an example of a $2 \times 5$ square lattice model of the nanosensor with no broken wires. There are 2 columns and 5 rows of nanowires connected with 10 nanobonds, and 13 nanowires all together. Assuming, that there are electrical contacts on the left-hand side and on the right-hand side of the nanonetwork, the electrical current has many paths to go through the nanosensor. On figure 4.3 there

![Figure 4.2: 2 × 5 square lattice model of the nanosensor](image)
is an example of a $4 \times 4$ square lattice model of the nanosensor with some broken wires, which are not shown on the picture. There are 6 nanowires which are broken and 18 nanowires which are still working. The nanosensor is workable since the electrical current still able to find the path to get through this $4 \times 4$ nanosensor.

A Bernoulli random variable is employed to each nanowire to determine if it is functioning or broken. Let $X_{n \times m}$ be the random variable that represents the number of cycles of $H_2$ it withstands before nanowires break and the electricity cannot go through the nanosensor. In [18] and [17] $X_{n \times m}$ is known as the lifetime of the nanosensor, a discrete random variable. Also, the reliability function of the nanosensor is defined as follows

$$R_{n \times m}(x) = Pr(X_{n \times m} > x)$$

for $x = 0, 1, 2, \ldots$, and $R_{n \times m}(0) = 1$. $E(X_{n \times m})$ is the expected lifetime of the nanosensor. Let $p$ be the probability that the nanowire doesn’t break during the cycle and it is the same for each cycle. It is assumed, that nanowires break independently of each other.

Here, two cases of square lattice nanosensor model are considered. First special nanosensor is of size $2 \times m$. Such network consist of $m$ horizontal wires and 2 columns of vertical wires, and looks like a ladder, see figure 4.2 where $m = 5$. The reliability function for $2 \times m$ network as shown in [18] for $m \geq 2$ is

$$R_{2 \times m}(x) = Pr(X_{2 \times m} > x) = 1 - (1 - p^x)^m.$$
The expectation for $2 \times m$ network as shown in [18] for $m \geq 2$ is

$$E(X_{2\times m}) = \sum_{k=1}^{m} (-1)^{k+1} \binom{m}{k} \frac{1}{1-p^k} \approx \frac{1}{2} - \frac{1}{p} \log \frac{1}{\sum_{k=1}^{m} \frac{1}{k}}.$$  

Three samples of 10, 25, and 50 lifetimes were generated from square lattice $2 \times 7$ nanosensor with probability of nanowire not breaking of 0.9. Even though the likelihood function is tractable as shown in [17], assume that it is not and the probability of nanowire not breaking (which is the reliability of a nanowire) is not known. Three ABC - EL methods with the same prior distribution $U(0, 1)$ are used to estimate the reliability parameter. For ABC - EL every parameter candidate $\theta^*$ drawn from the prior distribution $U(0, 1)$ is used to determine the constraint for the mean,

$$\sum_{i=1}^{n} w_i X_i = g(\theta^*),$$

where $g(\theta^*)$ for square lattice $2 \times 7$ nanosensor is

$$g(\theta^*) = E(X_{2\times 7}) \approx \frac{1}{2} - \frac{1}{\log \theta^*} \sum_{i=1}^{7} \frac{1}{i}.$$  

Here $X_{2\times 7}$ is a lifetime of square lattice $2 \times 7$ nanosensor, see [18] for more details. Now, the weights $w_i$'s are found using the following formula,

$$w_i = \frac{1}{n} \frac{1}{1 + \lambda(X_i - g(\theta^*))},$$

where $i = 1, \ldots, n$, $n$ is the sample size, which is 10 for the first sample, 25 for the second sample, and 50 for the third; and $\lambda$ is determined from the following equation

$$\frac{1}{n} \sum_{i=1}^{n} \frac{X_i - g(\theta^*)}{1 + \lambda(X_i - g(\theta^*))} = 0.$$  

This equation has only one solution for $\lambda$ because the derivative of the left part is less than or equal to zero,

$$-\frac{1}{n} \sum_{i=1}^{n} \frac{(X_i - g(\theta^*))^2}{(1 + \lambda(X_i - g(\theta^*)))^2} \leq 0.$$
Newton method is used to solve for $\lambda$ using the following interval as suggested by Owen in [39],

$$\frac{1 - n^{-1}}{g(\theta^*) - X_{(n)}} < \lambda < \frac{1 - n^{-1}}{g(\theta^*) - X_{(1)}},$$

knowing that $X_{(1)} < g(\theta^*) < X_{(n)}$. If $g(\theta^*)$ is outside of this interval, another candidate $\theta^*$ will be drawn from the prior distribution.

Once for a given $\theta^*$ all $w_i$’s are found, the weight for the $\theta^*$ is calculated as a product of all normalized $w_i$’s. In this fashion, 100 parameter candidates with corresponding weights are found and the weights are normalized. The ABC - EL parameter estimate is the weighted average of 100 parameter candidates.

For ABC - EL with resampling the same procedure is done as for ABC - EL. Once the sample of parameter candidates with corresponding weights is ready, the effective sample size (ESS) is calculated. While ESS is less than 90% of the sample, keep resample 100 values with replacement from the parameter candidate’s sample using corresponding weights as probabilities and add the normal error with mean of zero and standard deviation of 0.001. The ABC - EL with resampling estimate is the weighted average of the last set of 100 resampled values, which have ESS of at least 90.

For ABC - EL with testing each parameter candidate $\theta^*$ drawn from prior distribution $U(0, 1)$ is tested before the weight is assigned. First, $\theta^*$ is used in the constraint,

$$\sum_{i=1}^{n} w_i X_i = g(\theta^*),$$

in order to find profile empirical likelihood ratio function $R(\theta^*)$, which is

$$R(\theta^*) = \max \left\{ \prod_{i=1}^{n} n w_i \left| \sum_{i=1}^{n} w_i (X_i - g(\theta^*)) = 0, w_i \geq 0, \sum_{i=1}^{n} w_i = 1 \right\}.$$ 

Then, the test statistic $-2 \log R(g(\theta^*))$ and the p-value are calculated. ABC - EL with testing uses the level of significance of 0.25. If the p-value is greater than 0.25, then for $\theta^*$ the weight is found using the same actions as for ABC - EL; otherwise, another parameter candidate is drawn from the prior. ABC - EL with testing parameter estimate is a weighted average of 100 candidates.
The results are as follows in table 4.3 which includes mean squared error (MSE) with bias on 25 runs of the algorithm with 100 iterations in it.

<table>
<thead>
<tr>
<th>Method</th>
<th>ABC - EL</th>
<th>ABC - EL resample</th>
<th>ABC - EL test</th>
</tr>
</thead>
<tbody>
<tr>
<td>S.Size</td>
<td>MSE</td>
<td>Bias</td>
<td>MSE</td>
</tr>
<tr>
<td>10</td>
<td>0.099</td>
<td>0.31</td>
<td>0.002</td>
</tr>
<tr>
<td>25</td>
<td>0.024</td>
<td>0.14</td>
<td>0.004</td>
</tr>
<tr>
<td>50</td>
<td>0.001</td>
<td>0.021</td>
<td>0.004</td>
</tr>
</tbody>
</table>

Table 4.3: Summary for three different ABC - EL methods for $2 \times 7$ square lattice hydrogen nanosensor model.

For all three samples, it can be concluded that ABC - EL with testing produces the best estimates out of three methods since it has the smallest mean squared errors and biases.

Second case is the nanosensor of size $n \times m$. Such network consist of $m$ rows of horizontal wires and $n$ columns of vertical wires, see figure 4.3 where $n = m = 4$. The lower bound for the reliability function for $n \times m$ network as shown in [18] is

$$R_{n \times m}(x) \geq 1 - (1 - p^{(n-1)x})^m.$$  

The lower bound for the expectation for $n \times m$ network as shown in [18] is

$$E(X_{n \times m}) \geq \sum_{k=1}^{m} (-1)^{k+1} \binom{m}{k} \frac{1}{1 - p^{(n-1)k}} \approx \frac{1}{2} - \frac{1}{\log p^{n-1}} \sum_{k=1}^{m} \frac{1}{k}.$$  

Three samples of 10, 25, and 50 lifetimes were generated from square lattice $4 \times 4$ nanosensor with probability of nanowire not breaking of 0.9. The likelihood function in this case is intractable as shown in [21]. Assume that the probability of nanowire not breaking (which is the reliability of a nanowire) is not known. Three ABC - EL methods with the same prior distribution $\mathcal{U}(0,1)$ are used to estimate the reliability. ABC - EL with testing uses the level of significance of 0.25. The results are as follows in table 4.4 which includes mean squared error (MSE) with bias on 25 runs of the algorithm with 100 iterations in it.

For all three samples, it can be derived that ABC - EL with testing produces the best estimates out of three methods since it has the smallest mean squared errors and biases. Notice, that mean squared errors and
<table>
<thead>
<tr>
<th>Method</th>
<th>ABC - EL</th>
<th>ABC - EL resample</th>
<th>ABC - EL test</th>
</tr>
</thead>
<tbody>
<tr>
<td>S.Size</td>
<td>MSE</td>
<td>Bias</td>
<td>MSE</td>
</tr>
<tr>
<td>10</td>
<td>0.196</td>
<td>0.442</td>
<td>0.201</td>
</tr>
<tr>
<td>25</td>
<td>0.024</td>
<td>0.128</td>
<td>0.004</td>
</tr>
<tr>
<td>50</td>
<td>0.003</td>
<td>0.049</td>
<td>0.005</td>
</tr>
</tbody>
</table>

Table 4.4: Summary for three different ABC - EL methods for $4 \times 4$ square lattice hydrogen nanosensor model.

Biases are much bigger here than for $2 \times 7$ square lattice hydrogen nanosensor. The reason is that for this case ABC - EL implemented with inequality constraint as described in section 3.6, which is much weaker than the equality constraint. In general, for $n \times m$ square lattice hydrogen nanosensor it is possible to determine only the lower bound for the expected lifetime of the nanosensor as shown in [18], which was used as a constraint in all ABC via empirical likelihood algorithms here. The figure 4.4 illustrates the difference in MSE values for these methods.

Figure 4.4: Comparison of MSEs for three ABC-EL methods.

4.6 Observed Properties of ABC - EL with Testing

In simulations for normal, exponential, and hydrogen nanosensor models, it has been shown that ABC - EL with testing gives more accurate parameter estimates than ABC - EL or ABC - EL with resampling. In this section several observed properties via simulations of ABC - EL with testing method are considered.
The first question of interest is how the mean squared error (MSE) and the bias changes when the sample of observed data increases in size. To analyze this let us consider the results from tables 4.1 - 4.4 for three different models: normal $N(0,1)$, exponential $Exp(5)$, square lattice nanosensor with reliability of 0.9. For the normal model, only one of the samples of 25 observations is chosen. Combined information is presented in the table 4.5.

<table>
<thead>
<tr>
<th>S. Size</th>
<th>S. Mean</th>
<th>MSE</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Normal Model</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.134</td>
<td>0.041</td>
<td>0.201</td>
</tr>
<tr>
<td>25</td>
<td>0.146</td>
<td>0.029</td>
<td>0.170</td>
</tr>
<tr>
<td>50</td>
<td>-0.075</td>
<td>0.004</td>
<td>0.066</td>
</tr>
<tr>
<td></td>
<td>Exponential Model</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>4.763</td>
<td>0.056</td>
<td>0.232</td>
</tr>
<tr>
<td>25</td>
<td>4.363</td>
<td>0.168</td>
<td>0.404</td>
</tr>
<tr>
<td>50</td>
<td>4.853</td>
<td>0.008</td>
<td>0.077</td>
</tr>
<tr>
<td></td>
<td>2 × 7 Square Lattice Nanosensor</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>21.4</td>
<td>0.0003</td>
<td>0.018</td>
</tr>
<tr>
<td>25</td>
<td>24.32</td>
<td>0.00001</td>
<td>0.004</td>
</tr>
<tr>
<td>50</td>
<td>26.52</td>
<td>0.00003</td>
<td>0.005</td>
</tr>
<tr>
<td></td>
<td>4 × 4 Square Lattice Nanosensor</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>13.4</td>
<td>0.002</td>
<td>0.047</td>
</tr>
<tr>
<td>25</td>
<td>9.04</td>
<td>0.0005</td>
<td>0.022</td>
</tr>
<tr>
<td>50</td>
<td>8.96</td>
<td>0.0004</td>
<td>0.021</td>
</tr>
</tbody>
</table>

Table 4.5: Mean squared error (MSE), bias for ABC - EL with testing method for three different models.

As expected, there is a noticeable decrease in the mean squared error and the bias as the sample size increases from 10 to 50. For normal model and $4 \times 4$ square lattice nanosensor mean squared error and bias decrease as the sample size increases from 10 to 25 and to 50. For the exponential model as the sample size changes from 10 to 25 MSE and bias increase. This is unusual, but it seems to be a problem in the sample itself, because the sample mean is a lot lower than the true mean, which is 5. For $2 \times 7$ square lattice nanosensor mean squared error and bias increase as sample size changes from 25 to 50, but the difference is negligible.

The second question of interest is if the mean squared error and the bias decrease when the number of iterations, the number of suitable parameter estimates chosen from the prior distribution, within the ABC - EL with testing algorithm increases. Let us consider the same examples as before when sample size is
normal and exponential model to estimate the mean, and square lattice nanosensor and estimate the reliability. The number of iterations of 50, 500, and 5000 has been used. MSE and bias are computed based of 25 runs of the algorithm. The results are as follows in the table 4.6

<table>
<thead>
<tr>
<th># of iterations</th>
<th>S. Mean</th>
<th>MSE</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal Model</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>0.134</td>
<td>0.045</td>
<td>0.211</td>
</tr>
<tr>
<td>500</td>
<td>0.134</td>
<td>0.040</td>
<td>0.200</td>
</tr>
<tr>
<td>5000</td>
<td>0.134</td>
<td>0.041</td>
<td>0.202</td>
</tr>
<tr>
<td>Exponential Model</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>4.763</td>
<td>0.072</td>
<td>0.255</td>
</tr>
<tr>
<td>500</td>
<td>4.763</td>
<td>0.057</td>
<td>0.238</td>
</tr>
<tr>
<td>5000</td>
<td>4.763</td>
<td>0.056</td>
<td>0.237</td>
</tr>
<tr>
<td>2×7 Square Lattice Nanosensor</td>
<td></td>
<td>0.0003</td>
<td>0.018</td>
</tr>
<tr>
<td>50</td>
<td>21.4</td>
<td>0.0003</td>
<td>0.018</td>
</tr>
<tr>
<td>500</td>
<td>21.4</td>
<td>0.0003</td>
<td>0.018</td>
</tr>
<tr>
<td>5000</td>
<td>21.4</td>
<td>0.0003</td>
<td>0.018</td>
</tr>
<tr>
<td>4×4 Square Lattice Nanosensor</td>
<td></td>
<td>0.0022</td>
<td>0.0474</td>
</tr>
<tr>
<td>50</td>
<td>13.4</td>
<td>0.0022</td>
<td>0.047</td>
</tr>
<tr>
<td>500</td>
<td>13.4</td>
<td>0.0022</td>
<td>0.047</td>
</tr>
<tr>
<td>5000</td>
<td>13.4</td>
<td>0.0022</td>
<td>0.047</td>
</tr>
</tbody>
</table>

Table 4.6: Mean squared error (MSE), bias for ABC - EL with testing method for different number of iterations.

As expected, for normal and exponential models when the number of iterations increases from 50 to 500, MSE and bias becomes smaller. Further increase of iterations is not beneficial, MSE and bias stay very similar as the number of iterations increases from 500 to 5000. For the nanosensor model there are no noticeable changes in MSE and bias as the number of iterations within algorithm increases.

The third question of interest is does the prior distribution effect the parameter estimate when ABC - EL with testing is implemented. Consider the same models as before: normal and exponential model to estimate the mean, and square lattice nanosensor and estimate the reliability. This time different prior distributions were used. For normal model the priors are $N(0.5, 0.7^2)$, $N(0.1, 0.7^2)$, and $U(-2, 2)$. For exponential model the priors are $Gamma(1.5, 4)$, $Gamma(2, 3)$, where the first parameter is the shape and the second parameter is the scale; and $U(2, 6)$. For the square lattice nanosensor model the priors are $Beta(4, 1)$, $Beta(5, 1.9)$, and $U(0, 1)$. The results are summarized in the table 4.7
<table>
<thead>
<tr>
<th>Prior</th>
<th>S. Mean</th>
<th>MSE</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal Model</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$N(0.5,0.7^2)$</td>
<td>0.134</td>
<td>0.040</td>
<td>0.200</td>
</tr>
<tr>
<td>$N(0.1,0.7^2)$</td>
<td>0.134</td>
<td>0.030</td>
<td>0.172</td>
</tr>
<tr>
<td>$U(-2,2)$</td>
<td>0.134</td>
<td>0.031</td>
<td>0.175</td>
</tr>
<tr>
<td>Exponential Model</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$Gamma(2,3)$</td>
<td>4.763</td>
<td>0.036</td>
<td>0.175</td>
</tr>
<tr>
<td>$Gamma(1.5,4)$</td>
<td>4.763</td>
<td>0.041</td>
<td>0.191</td>
</tr>
<tr>
<td>$U(2,6)$</td>
<td>4.763</td>
<td>0.052</td>
<td>0.220</td>
</tr>
<tr>
<td>$2 \times 7$ Square Lattice Nanosensor</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$U(0,1)$</td>
<td>21.4</td>
<td>0.0003</td>
<td>0.018</td>
</tr>
<tr>
<td>$Beta(5,1.9)$</td>
<td>21.4</td>
<td>0.0003</td>
<td>0.018</td>
</tr>
<tr>
<td>$Beta(4,1)$</td>
<td>21.4</td>
<td>0.0003</td>
<td>0.018</td>
</tr>
<tr>
<td>$4 \times 4$ Square Lattice Nanosensor</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$U(0,1)$</td>
<td>13.4</td>
<td>0.002</td>
<td>0.047</td>
</tr>
<tr>
<td>$Beta(5,1.9)$</td>
<td>13.4</td>
<td>0.002</td>
<td>0.047</td>
</tr>
<tr>
<td>$Beta(4,1)$</td>
<td>13.4</td>
<td>0.002</td>
<td>0.047</td>
</tr>
</tbody>
</table>

Table 4.7: Mean squared error (MSE), bias for ABC - EL with testing method for different prior distributions.

For normal and exponential models as prior distribution changes, MSE and bias also change. For the nanosensor there is no noticeable difference in MSE and bias’s values when different prior distributions are used.

4.7 ABC via Empirical Likelihood with Different Prior Distributions

In this section three ABC via empirical likelihood approaches are analysed on how there are sensitive to the choice of the prior distribution.

Let us consider the same examples from section 4.5: normal model, exponential model, $2 \times 7$ square lattice nanosensor, and $4 \times 4$ square lattice nanosensor model. For the normal model three different prior distributions $N(0.5,0.7^2)$, $N(0.1,0.7^2)$, and $U(-2,2)$ are used. For the exponential model the priors are $U(2,6)$, $Gamma(2,6)$, and $Gamma(1.5,4)$. For the nanosensor model priors are $U(0,1)$, $Beta(5,1.9)$, and $Beta(4,1)$. The methods used here are ABC-EL, ABC-EL with resampling, and ABC-EL with testing. The results are as follows in table 4.8.
Normal Model

<table>
<thead>
<tr>
<th>Prior</th>
<th>Method</th>
<th>ABC-EL MSE</th>
<th>ABC-EL Bias</th>
<th>ABC-EL resample MSE</th>
<th>ABC-EL resample Bias</th>
<th>ABC-EL test MSE</th>
<th>ABC-EL test Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N(0.5, 0.7^2)$</td>
<td>0.134</td>
<td>0.073</td>
<td>0.268</td>
<td>0.043</td>
<td>0.203</td>
<td>0.040</td>
<td>0.200</td>
</tr>
<tr>
<td>$N(0.1, 0.7^2)$</td>
<td>0.134</td>
<td>0.034</td>
<td>0.180</td>
<td>0.026</td>
<td>0.157</td>
<td>0.030</td>
<td>0.172</td>
</tr>
<tr>
<td>$U(-2, 2)$</td>
<td>0.134</td>
<td>0.013</td>
<td>0.074</td>
<td>0.014</td>
<td>0.110</td>
<td>0.031</td>
<td>0.175</td>
</tr>
</tbody>
</table>

Exponential Model

<table>
<thead>
<tr>
<th>Prior</th>
<th>Method</th>
<th>ABC-EL MSE</th>
<th>ABC-EL Bias</th>
<th>ABC-EL resample MSE</th>
<th>ABC-EL resample Bias</th>
<th>ABC-EL test MSE</th>
<th>ABC-EL test Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gamma(2, 6)</td>
<td>4.763</td>
<td>89</td>
<td>9</td>
<td>116</td>
<td>10</td>
<td>0.012</td>
<td>0.093</td>
</tr>
<tr>
<td>Gamma(1.5, 4)</td>
<td>4.763</td>
<td>1.95</td>
<td>1.19</td>
<td>0.892</td>
<td>0.682</td>
<td>0.041</td>
<td>0.191</td>
</tr>
<tr>
<td>$U(2, 6)$</td>
<td>4.763</td>
<td>0.168</td>
<td>0.406</td>
<td>0.076</td>
<td>0.261</td>
<td>0.052</td>
<td>0.220</td>
</tr>
</tbody>
</table>

$2 \times 7$ Square Lattice Nanosensor

<table>
<thead>
<tr>
<th>Prior</th>
<th>Method</th>
<th>ABC-EL MSE</th>
<th>ABC-EL Bias</th>
<th>ABC-EL resample MSE</th>
<th>ABC-EL resample Bias</th>
<th>ABC-EL test MSE</th>
<th>ABC-EL test Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>$U(0, 1)$</td>
<td>21.4</td>
<td>0.126</td>
<td>0.351</td>
<td>0.001</td>
<td>0.018</td>
<td>0.0003</td>
<td>0.018</td>
</tr>
<tr>
<td>Beta(5, 1.9)</td>
<td>21.4</td>
<td>0.003</td>
<td>0.046</td>
<td>0.0004</td>
<td>0.016</td>
<td>0.0003</td>
<td>0.018</td>
</tr>
<tr>
<td>Beta(4, 1)</td>
<td>21.4</td>
<td>0.0008</td>
<td>0.026</td>
<td>0.002</td>
<td>0.049</td>
<td>0.0003</td>
<td>0.018</td>
</tr>
</tbody>
</table>

Table 4.8: MSE and bias for normal, exponential, and nanosensor models using ABC-EL, ABC-EL with resampling, ABC-EL with testing methods and different prior distributions.

Figure 4.5 illustrates the results shown in table 4.8 for $2 \times 7$ square lattice nanosensor. As prior changes, MSE obtained by ABC-EL or ABC-EL with resampling varies a lot, but MSE attained by ABC-EL with testing stays the same. One of MSE values received by ABC-EL is not seen on the graph since it is very high.

ABC via empirical likelihood algorithms have many advantages when compared to other ABC techniques, such as there is no choice of tolerance level, distance metric, model for data simulation, summary statistic, and huge time improvement. However, these methods are easily affected by prior information. ABC-EL with testing is least sensitive to the choice of prior distribution among them.

Implementing the hypothesis testing procedure within ABC-EL algorithm improved the accuracy of the parameter estimation, decreased the affect of prior information on the result, and the algorithm does not take much more time to run than original ABC-EL. Hence, ABC-EL with testing is recommended to use rather than original ABC-EL or ABC-EL with resampling.
Figure 4.5: Comparison of MSEs obtained by different ABC via empirical likelihood methods with various priors for the nanosensor model example.
CHAPTER 5
COMBINED ABC ALGORITHMS

In this chapter, the research goal is to improve the accuracy of estimates obtained by ABC via empirical likelihood, in particular ABC-EL with testing. As it was mentioned before, ABC-EL with testing method and other ABC via empirical likelihood algorithms are sensitive to the prior information of the parameter of interest. The improvement can be achieved through expanding the prior knowledge of the parameter. ABC methods described in Chapter 2 are suggested to use for this purpose. Different combinations of ABC approaches from Chapter 2 and three ABC via empirical likelihood techniques from Chapter 4 were considered, and the best mix was chosen.

5.1 Comparison of ABC Algorithms

In this section different ABC methods from Chapter 2 (ABC-REJ, ABC-MCMC, ABC-PMC) and ABC-EL with testing from Chapter 4 are compared on the following examples.

Example 5.1. Consider normal model example. Ten observations are generated from standard normal distribution: -0.993, -0.037, 0.335, -1.031, 2.632, 0.469, 0.882, -0.219, -0.536, -0.165. Suppose, the likelihood function is intractable and the true mean need to be estimated. ABC-REJ, ABC-MCMC, ABC-PMC, and ABC-EL with testing are applied to estimate the true mean.

Prior distribution chosen to be $N(0.5, 0.7^2)$. Simulation model is also normal. Posterior sample size is 100. Summary statistic if needed is the mean. Tolerance level is 0.0001. Tolerance function for ABC-PMC is $0.0005/t$, where $t = 1,...,5$. Results: time in seconds, estimate, mean squared error (MSE), and bias for each method are summarized in the table 5.1. Estimate is of the first run of the algorithm. MSE and bias are calculated based on 25 runs of the algorithm.
Table 5.1: Summary for normal model example using ABC-REJ, ABC-MCMC, ABC-PMC, ABC-EL with testing methods.

For the normal model, ABC-EL with testing does not produce the estimate better than other algorithms; MSE and bias are not the lowest. This is a motivation for searching the ways to improve ABC-EL with testing algorithm.

Example 5.2. Consider normal mixture model example. Ten observations are generated from normal mixture $0.5N(0,1) + 0.5N(0,0.01)$: 0.021, -0.321, -0.0227, 0.008, 0.004, 0.003, -0.014, -0.007, 0.454, 0.780. Suppose, the likelihood function is intractable and the true mean need to be estimated. ABC-REJ, ABC-MCMC, ABC-PMC, and ABC-EL with testing are applied to estimate the true mean.

Prior distribution chosen to be $N(0.5,0.7^2)$. Simulation model is also normal. Posterior sample size is 100. Summary statistic if needed is the mean. Tolerance level is 0.0001. Tolerance function for ABC-PMC is $0.0005/t$, where $t = 1,...,5$. Results: time in seconds, estimate, mean squared error (MSE), and bias for each method are summarized in the table 5.2. MSE and bias are calculated based on 25 runs of the algorithm.

Table 5.2: Summary for normal mixture model example using ABC-REJ, ABC-MCMC, ABC-PMC, ABC-EL with testing methods.

For the normal mixture model, although ABC-MCMC estimate shown in the table 5.2 is most accurate, ABC-EL with testing gives the overall most accurate parameter estimates since MSE and bias are the lowest.
Example 5.3. The exponential model example is considered. Ten observations are generated from exponential distribution with the mean of 5: 0.75, 4.19, 4.24, 1.41, 3.44, 7.71, 9.75, 0.94, 1.71, 13.49. Suppose, the likelihood function is intractable and the true mean need to be estimated. ABC-REJ, ABC-MCMC, ABC-PMC, and ABC-EL with testing are applied to estimate the true mean.

Prior distribution chosen to be $U(2,6)$. Simulation model is also exponential. Posterior sample size is 100. Summary statistic if needed is the mean. Tolerance level is 0.0001. Tolerance function for ABC-PMC is $0.0005/t$, where $t = 1, \ldots, 5$. Results: time in seconds, estimate, mean squared error (MSE), and bias for each method are summarized in the table 5.3. MSE and bias are calculated based on 25 runs of the algorithm.

<table>
<thead>
<tr>
<th>Method</th>
<th>Time (sec.)</th>
<th>Estimate</th>
<th>MSE</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC-REJ</td>
<td>128</td>
<td>4.657</td>
<td>0.176</td>
<td>0.412</td>
</tr>
<tr>
<td>ABC-MCMC</td>
<td>106</td>
<td>4.799</td>
<td>0.154</td>
<td>0.373</td>
</tr>
<tr>
<td>ABC-PMC</td>
<td>343</td>
<td>4.660</td>
<td>0.181</td>
<td>0.416</td>
</tr>
<tr>
<td>ABC-EL test</td>
<td>1.3</td>
<td>4.829</td>
<td>0.066</td>
<td>0.249</td>
</tr>
</tbody>
</table>

Table 5.3: Summary for exponential model example using ABC-REJ, ABC-MCMC, ABC-PMC, ABC-EL with testing methods.

For the exponential model ABC-EL with testing gives the most accurate parameter estimate; MSE and bias are the lowest.

Example 5.4. Square lattice nanosensor model is used. Ten nanosensor lifetimes were generated from $2 \times 7$ square lattice nanosensor model with true reliability of 0.9: 26, 16, 23, 7, 27, 37, 23, 24, 16, 15. Suppose, the likelihood function is intractable and the true reliability need to be estimated. ABC-REJ, ABC-MCMC, ABC-PMC, and ABC-EL with testing are applied to estimate the true reliability.

Prior distribution chosen to be uninformative, $U(0,1)$. Simulation model is also nanosensor model. Posterior sample size is 100. Summary statistic if needed is the mean. Tolerance level is 3. Tolerance function for ABC-PMC is $15/t$, where $t = 1, \ldots, 5$. Results: time in seconds, estimate, mean squared error (MSE), and bias for each sample and method are summarized in the table 5.4. MSE and bias are calculated based on 25 runs of the algorithm.
Table 5.4: Summary for $2 \times 7$ square lattice nanosensor model example using ABC-REJ, ABC-MCMC, ABC-PMC, ABC-EL with testing methods.

For the nanosensor model ABC-EL with testing gives the most accurate parameter estimates; MSE and bias are the lowest.

ABC-EL with testing work faster than any other method, but not always produces the best estimates with the lowest mean squared error and the bias.

5.2 Combined ABC Methods

Results from simulations in previous section show that there is a motivation to improve ABC methods introduced in Chapter 2 by combining them with ABC via empirical likelihood algorithms. As have been mentioned before, the ABC via empirical likelihood methods are sensitive to the prior information. Therefore, the combination is proposed to be as follows. First step is to determine an interval of the posterior mass of the parameter using one run of ABC-REJ, ABC-MCMC, or ABC-PMC algorithm. Second step is to run one of the three ABC via empirical likelihood (ABC-EL, ABC-EL res., ABC-EL test) using a prior truncated to the interval of posterior mass found in the first step. The easiest way to do it is to use the uniform prior on the interval with lower bound to be the minimum value of the parameter posterior mass and upper bound to be the maximum value of the parameter posterior mass found in the first step. This process is similar to the semiautomatic ABC, where the pilot run is performed to find a region of non-negligible mass, which is used for truncating the prior. The names for the new algorithms, which are based on the methods used in the first and second steps, can be found in the table 5.5. For instance, if in the first step ABC-REJ and in the second step ABC-EL with resampling were applied, then the name is ABC-REJ+EL res., as can be found in the first row and second column of the table 5.5.
Table 5.5: Names for combined ABC algorithms.

<table>
<thead>
<tr>
<th>1st step</th>
<th>2nd step</th>
<th>ABC-EL</th>
<th>ABC-EL res.</th>
<th>ABC-EL test</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC-REJ</td>
<td>ABC-REJ+EL</td>
<td>ABC-REJ+EL res.</td>
<td>ABC-REJ+EL test</td>
<td></td>
</tr>
<tr>
<td>ABC-MCMC</td>
<td>ABC-MCMC+EL</td>
<td>ABC-MCMC+EL res.</td>
<td>ABC-MCMC+EL test</td>
<td></td>
</tr>
<tr>
<td>ABC-PMC</td>
<td>ABC-PMC+EL</td>
<td>ABC-PMC+EL res.</td>
<td>ABC-PMC+EL test</td>
<td></td>
</tr>
</tbody>
</table>

To make algorithm more time efficient we recommend to use ABC algorithm in the first step with high tolerance for the accepted values of the parameter candidates. It is the same suggestion as for the pilot run in semiautomatic ABC. In this case the parameter posterior mass will not be the best, but satisfactory for the ABC-EL to produce accurate results thereafter. The results will be comparable or better than using just the ABC method described in Chapter 2 with low tolerance.

Example 5.5. For simulations, standard normal model is considered. Three samples of sizes 10, 25, and 50 observations were studied. Suppose, the likelihood function is intractable, the true mean is unknown and needs to be estimated. For the first step ABC-REJ, ABC-MCMC, and ABC-PMC are applied. Prior distribution chosen to be $N(0.5, 0.7^2)$. Simulation model is also normal. Posterior sample size is 100. Summary statistic if needed is the mean. Tolerance level is 0.001. Tolerance function for ABC-PMC is $0.005/t$, where $t = 1, ..., 5$. An interval of posterior mass for the parameter is determined for each sample. For the second step, a uniform prior over the determined interval of posterior mass is used within ABC-EL, ABC-EL with sampling (ABC-EL res.), and ABC-EL with testing (ABC-EL test). Time in seconds, MSE and bias, based on 25 runs of the algorithms, are recorded for each case in the table 5.6. It is possible to calculate MSE and bias since the true value of the parameter is known.

Example 5.6. Let us consider exponential model with mean of 5. Three samples of sizes 10, 25, and 50 observations were studied. Suppose, the likelihood function is intractable, the true mean is unknown and needs to be estimated. For the first step ABC-REJ, ABC-MCMC, and ABC-PMC are applied. Prior distribution chosen to be $U(2, 6)$. Simulation model is also exponential. Posterior sample size is 100. Summary statistic if needed is the mean. Tolerance level is 0.001. Tolerance function for ABC-PMC is $0.005/t$, where $t = 1, ..., 5$. An interval of posterior mass for the parameter is determined for each sample. For the second step, a uniform prior over the determined interval of posterior mass is used within ABC-EL,
Table 5.6: Summary for normal model using combinations of ABC techniques.

ABC-EL with sampling (ABC-EL res.), and ABC-EL with testing (ABC-EL test). Time in seconds, MSE and bias, based on 25 runs of the algorithms, are recorded for each case in the table 5.7. It is possible to calculate MSE and bias since the true value of the parameter is known.

Table 5.7: Summary for exponential model using combinations of ABC techniques.
Example 5.7. Third, $2 \times 7$ square lattice nanosensor model with reliability of 0.9 is examined. Three samples of sizes 10, 25, and 50 observations were studied. Suppose, the likelihood function is intractable, the true reliability is unknown and needs to be estimated. For the first step ABC-REJ, ABC-MCMC, and ABC-PMC are applied. Prior distribution chosen to be $U(0, 1)$. Simulation model is also $2 \times 7$ squared lattice nanosensor. Posterior sample size is 100. Distance metric is euclidean. Summary statistic if needed is the mean. Tolerance level is 14. Tolerance function for ABC-PMC is $70/t$, where $t = 1, \ldots, 5$. An interval of posterior mass for the parameter is determined for each sample. For the second step, a uniform prior over the determined interval of posterior mass is used within ABC-EL, ABC-EL with sampling (ABC-EL res.), and ABC-EL with testing (ABC-EL test). Time in seconds, MSE and bias, based on 25 runs of the algorithms, are recorded for each case in the table 5.8. It is possible to calculate MSE and bias since the true value of the parameter is known.

<table>
<thead>
<tr>
<th>SS</th>
<th>Time</th>
<th>MSE</th>
<th>Bias</th>
<th>Time</th>
<th>MSE</th>
<th>Bias</th>
<th>Time</th>
<th>MSE</th>
<th>Bias</th>
<th>Time</th>
<th>MSE</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>54</td>
<td>0.009</td>
<td>0.095</td>
<td>1</td>
<td>0.0005</td>
<td>0.021</td>
<td>1.8</td>
<td>0.0003</td>
<td>0.18</td>
<td>8</td>
<td>0.0003</td>
<td>0.018</td>
</tr>
<tr>
<td>25</td>
<td>256</td>
<td>0.003</td>
<td>0.051</td>
<td>0.9</td>
<td>2 $\cdot$ 10$^{-3}$</td>
<td>0.004</td>
<td>1.8</td>
<td>1 $\cdot$ 10$^{-3}$</td>
<td>0.004</td>
<td>7</td>
<td>1 $\cdot$ 10$^{-3}$</td>
<td>0.003</td>
</tr>
<tr>
<td>50</td>
<td>1046</td>
<td>0.0009</td>
<td>0.029</td>
<td>1</td>
<td>3 $\cdot$ 10$^{-3}$</td>
<td>0.006</td>
<td>2</td>
<td>2 $\cdot$ 10$^{-3}$</td>
<td>0.005</td>
<td>10</td>
<td>3 $\cdot$ 10$^{-3}$</td>
<td>0.005</td>
</tr>
</tbody>
</table>

Table 5.8: Summary for the $2 \times 7$ square lattice nanosensor using combinations of ABC techniques.

Example 5.8. Last, normal mixture model $0.5N(0, 1) + 0.5N(0, 0.01)$ is used. Three samples of sizes 10, 25, and 50 observations were studied. Suppose, the likelihood function is intractable, the true mean is unknown and needs to be estimated. For the first step ABC-REJ, ABC-MCMC, and ABC-PMC are applied. Prior distribution chosen to be $N(0.5, 0.7^2)$. Simulation model is also normal. Posterior sample size is 100. Distance metric is euclidean. Summary statistic if needed is the mean. Tolerance level is 0.001. Tolerance
function for ABC-PMC is $0.005/t$, where $t = 1, \ldots, 5$. An interval of posterior mass for the parameter is determined for each sample. For the second step, a uniform prior over the determined interval of posterior mass is used within ABC-EL, ABC-EL with sampling (ABC-EL res.), and ABC-EL with testing (ABC-EL test). Time in seconds, MSE and bias, based on 25 runs of the algorithms, are recorded for each case in the table 5.9. It is possible to calculate MSE and bias since the true value of the parameter is known.

<table>
<thead>
<tr>
<th>Normal mixture model. First step - ABC-REJ</th>
<th>ABC-REJ</th>
<th>ABC-REJ+EL</th>
<th>ABC-REJ+EL res.</th>
<th>ABC-REJ+EL test</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS</td>
<td>Time</td>
<td>MSE</td>
<td>Bias</td>
<td>Time</td>
</tr>
<tr>
<td>10</td>
<td>7</td>
<td>0.017</td>
<td>0.128</td>
<td>0.6</td>
</tr>
<tr>
<td>25</td>
<td>6</td>
<td>0.027</td>
<td>0.165</td>
<td>0.5</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>0.015</td>
<td>0.123</td>
<td>0.6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Normal mixture model. First step - ABC-MCMC</th>
<th>ABC-MCMC</th>
<th>ABC-MCMC+EL</th>
<th>ABC-MCMC+EL res.</th>
<th>ABC-MCMC+EL test</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS</td>
<td>Time</td>
<td>MSE</td>
<td>Bias</td>
<td>Time</td>
</tr>
<tr>
<td>10</td>
<td>15</td>
<td>0.014</td>
<td>0.117</td>
<td>0.7</td>
</tr>
<tr>
<td>25</td>
<td>13</td>
<td>0.026</td>
<td>0.159</td>
<td>0.5</td>
</tr>
<tr>
<td>50</td>
<td>14</td>
<td>0.016</td>
<td>0.127</td>
<td>0.6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Normal mixture model. First step - ABC-PMC</th>
<th>ABC-PMC</th>
<th>ABC-PMC+EL</th>
<th>ABC-PMC+EL res.</th>
<th>ABC-PMC+EL test</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS</td>
<td>Time</td>
<td>MSE</td>
<td>Bias</td>
<td>Time</td>
</tr>
<tr>
<td>10</td>
<td>25</td>
<td>0.017</td>
<td>0.127</td>
<td>0.6</td>
</tr>
<tr>
<td>25</td>
<td>20</td>
<td>0.030</td>
<td>0.173</td>
<td>0.5</td>
</tr>
<tr>
<td>50</td>
<td>13</td>
<td>0.015</td>
<td>0.122</td>
<td>0.4</td>
</tr>
</tbody>
</table>

Table 5.9: Summary for normal mixture model using combinations of ABC techniques.

As have been mentioned before, original ABC-EL and ABC-EL with resampling are more sensitive to the prior information than ABC-EL with testing. Thus, there is a possibility that once we have a more reliable prior, original ABC-EL will produce the lowest MSEs and biases. Results in tables 5.6, 5.7, 5.8, and 5.9 contradict this proposition. Using ABC-EL with testing in the second step gives comparable and often better results, lower MSEs and biases, than the original ABC-EL and ABC-EL with resampling. Especially clear it can be seen on the exponential example when on the first step ABC-MCMC was implemented, table 5.7. The figure 5.1 illustrates this case.

A relevant question is which ABC algorithm is better to choose in the first step of the combination. Since all the samples of the same size in simulations were used exactly the same, last column in the tables 5.6, 5.7, 5.8, and 5.9 reveals the answer. The MSEs and biases for the same observed samples are very similar in all tables except the exponential model example, where the use of ABC-PMC in the first step
Figure 5.1: Comparison of MSEs for ABC-MCMC+EL, ABC-MCMC+EL with resampling, ABC-MCMC+EL with testing on exponential model example.

for the sample of size 10 notably decrease the MSE and bias. The figure 5.2 illustrates this case. It is not surprising since ABC-PMC is the most advanced among all methods recommended here for the first step.

Therefore, the best combination found via simulations is ABC-PMC+EL test, where in the first step ABC-PMC is applied, and in the second step ABC-EL with testing is implemented. Since the mixture provides comparable and sometimes even lower MSE and bias than other combinations, it may improve the accuracy of estimation. Later, ABC-PMC+EL test will be called ABC mix.
Note, that using one of the ABC via empirical likelihood algorithm in the first step and ABC-REJ, or ABC-MCMC, or ABC-PMC in the second step does not improve accuracy of the estimation, because ABC methods without empirical likelihood procedure are not sensitive to the prior distribution. However, it is a good idea to do so for time efficiency, especially when prior is uninformative. In this case, first run ABC-EL with testing, then use all parameter candidates and corresponding weights as probabilities within ABC-REJ, or ABC-MCMC, or ABC-PMC as empirical prior distribution.

5.3 Comparison of ABC mix and ABC-EL with testing

Simulation results in the previous section showed that it is beneficial to use the ABC mix. The major difference between ABC mix and ABC-EL with testing is in the prior distribution. In the ABC mix more accurate prior information is employed. Here, ABC mix is compared with ABC-PMC and ABC-EL with testing on several examples from the previous section.

Example 5.5 (continued). Let us continue with the standard normal model. ABC-PMC, ABC mix, and ABC-EL with testing are employed to estimate the true mean. For ABC-PMC and ABC-EL with testing by themselves the prior distribution is $N(0.5, 0.7^2)$. Posterior sample size is 100. Time in seconds, MSE and bias based on 25 runs of the algorithms are recorded for each case in the table 5.10. It is possible to calculate MSE and bias since the true value of the parameter is known.

<table>
<thead>
<tr>
<th></th>
<th>Normal model</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ABC-PMC</td>
<td>ABC mix</td>
<td>ABC-EL test</td>
</tr>
<tr>
<td>S.Size</td>
<td>Time  MSE Bias</td>
<td>Time  MSE Bias</td>
<td>Time  MSE Bias</td>
</tr>
<tr>
<td>10</td>
<td>25  0.028 0.165</td>
<td>26  0.033 0.181</td>
<td>1.4  0.039 0.196</td>
</tr>
<tr>
<td>25</td>
<td>17  0.027 0.162</td>
<td>18  0.032 0.177</td>
<td>1.4  0.030 0.173</td>
</tr>
<tr>
<td>50</td>
<td>13  0.004 0.061</td>
<td>17  0.006 0.078</td>
<td>4  0.005 0.068</td>
</tr>
</tbody>
</table>

Table 5.10: Summary for standard normal model using ABC-PMC, ABC mix, and ABC-EL with testing.

Example 5.6 (continued). Let us continue with exponential model with true mean of 5. ABC-PMC, ABC mix, and ABC-EL with testing are employed to estimate the true mean. For ABC-PMC and ABC-EL with testing by themselves the prior distribution is $U(2,6)$. Posterior sample size is 100. Time in seconds, MSE
and bias based on 25 runs of the algorithms are recorded for each case in the table 5.11. It is possible to calculate MSE and bias since the true value of the parameter is known.

<table>
<thead>
<tr>
<th>S. Size</th>
<th>Time</th>
<th>MSE</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>178</td>
<td>0.193</td>
<td>0.424</td>
</tr>
<tr>
<td>25</td>
<td>18</td>
<td>0.270</td>
<td>0.513</td>
</tr>
<tr>
<td>50</td>
<td>13</td>
<td>0.019</td>
<td>0.124</td>
</tr>
</tbody>
</table>

Table 5.11: Summary for exponential model using ABC-PMC, ABC mix, and ABC-EL with testing.

Example 5.7 (continued). Let us continue with $2 \times 7$ square lattice nanosensor model with reliability of 0.9. ABC-PMC, ABC mix, and ABC-EL with testing are employed to estimate the true reliability. For ABC-PMC and ABC-EL with testing by themselves the prior distribution is $U(0,1)$. Posterior sample size is 100. Time in seconds, MSE and bias based on 25 runs of the algorithms are recorded for each case in the table 5.12. It is possible to calculate MSE and bias since the true value of the parameter is known.

<table>
<thead>
<tr>
<th>S. Size</th>
<th>Time</th>
<th>MSE</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>92</td>
<td>0.010</td>
<td>0.097</td>
</tr>
<tr>
<td>25</td>
<td>332</td>
<td>0.002</td>
<td>0.048</td>
</tr>
<tr>
<td>50</td>
<td>598</td>
<td>0.0008</td>
<td>0.027</td>
</tr>
</tbody>
</table>

Table 5.12: Summary for nanosensor model using ABC-PMC, ABC mix, and ABC-EL with testing.

Example 5.8 (continued). Let us continue with normal mixture model $0.5N(0,1) + 0.5N(0,0.01)$. ABC-PMC, ABC mix, and ABC-EL with testing are employed to estimate the true mean. For ABC-PMC and ABC-EL with testing by themselves the prior distribution is $N(0.5,0.7^2)$. Posterior sample size is 100. Time in seconds, MSE and bias based on 25 runs of the algorithms are recorded for each case in the table 5.13. It is possible to calculate MSE and bias since the true value of the parameter is known.

The simulation results in the table 5.11 suggest to use ABC mix since for all samples MSEs and biases are smaller, and the obtained estimates will be more accurate. As shown in the tables 5.10, 5.12, and 5.13 results recommend to use ABC-EL with testing since for all samples MSEs and biases are comparable to the ones obtained by ABC mix or ABC-PMC, but ABC-EL with testing takes less time to run.
In conclusion, combining ABC methods from Chapter 2 together with ABC via empirical likelihood techniques can be more beneficial than using just each of those algorithms by themselves. The best mix is the combination of ABC-PMC and ABC-EL with testing (ABC mix). There are situations when ABC mix produces lower MSE and bias than ABC-PMC or ABC-EL with testing, thereby the improvement of the accuracy of estimation is achieved. The figure 5.3 illustrates such case on exponential model example.

![Exponential Model](image)

Figure 5.3: Comparison of MSEs for ABC-PMC, ABC mix, and ABC-EL with testing on exponential model example.

<table>
<thead>
<tr>
<th>S. Size</th>
<th>Time</th>
<th>MSE</th>
<th>Bias</th>
<th>Time</th>
<th>MSE</th>
<th>Bias</th>
<th>Time</th>
<th>MSE</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>25</td>
<td>0.017</td>
<td>0.127</td>
<td>33</td>
<td>0.011</td>
<td>0.102</td>
<td>8</td>
<td>0.011</td>
<td>0.103</td>
</tr>
<tr>
<td>25</td>
<td>20</td>
<td>0.030</td>
<td>0.173</td>
<td>24</td>
<td>0.020</td>
<td>0.140</td>
<td>4</td>
<td>0.022</td>
<td>0.147</td>
</tr>
<tr>
<td>50</td>
<td>16</td>
<td>0.015</td>
<td>0.122</td>
<td>25</td>
<td>0.018</td>
<td>0.134</td>
<td>9</td>
<td>0.017</td>
<td>0.129</td>
</tr>
</tbody>
</table>

Table 5.13: Summary for normal mixture model using ABC-PMC, ABC mix, and ABC-EL with testing.
CHAPTER 6
ABC ALGORITHMS FOR CENSORED DATA

The data is censored when some observations are only partially known. There are different types of censored data: interval censoring, left, and right censoring. The most common form is the right censored data, which occurs when a subject leaves the study before an event occurs, or the study ends before the event has occurred. The right censoring is considered here.

There are many different ABC methods developed, but only few applicable for censored data. The existing ABC algorithm which works with censored data is modified ABC rejection recently proposed by Kristin McCullough and Nader Ebrahimi [35]. This method is described in the following section. The research goal is to improve it for the data with high proportions of censored observations via (i) implementing empirical likelihood weights, (ii) introducing hypothesis testing procedure for comparing survival functions, and (iii) employing testing approach for comparing smoothed distribution functions.

6.1 ABC Rejection Algorithm for Right Censored Data

This algorithm was developed to assess the reliability of nanocomponents. The difficulty of such analysis is in the structure and fabrication processes of nanodevices; also, the failure data consists of high proportion of censored observations since in reliability testing the true failure time may not be observed for all devices. Due to the complex structure of the nanocomponents it is very difficult to get the likelihood function, and if there is no likelihood, the Bayesian analysis for inference of reliability won’t be useful. Therefore, Approximate Bayesian Computations which are applicable for censored data are needed here.

The algorithm is the modification of basic rejection ABC method and suitable for right censored data. The parameter candidate is generated from the prior distribution, the sample is simulated from the model given the parameter candidate. Observed and simulated samples are compared based on two conditions unlike in rejection algorithm. The first condition is the distance between the summary statistics of observed
and simulated non-censored data must be smaller than some chosen in advance threshold. The second condition is that every censored time of the simulated sample must be larger than corresponding censored time from the observed sample.

The notations are as follows. The censored data is \( X = \{ (x_i, \delta_i) : i = 1, \ldots, n \} \), where \( n \) is the total number of observations; \( x_i = \min(y_i, z_i) \), where \( y_i \) is observed failure time and \( z_i \) is the censored failure time. Assume, that \( y_i \) and \( z_i \) are independent. If \( x_i = y_i \), then \( \delta_i = 1 \). If \( x_i = z_i \), then \( \delta_i = 0 \). Denote \( X' \) as the set of exact failure times, i.e. \( X' = \{ x_i : (x_i, \delta_i) \in X \text{ and } \delta_i = 1 \} \). Similar, \( X_s \) and \( X'_s \) are the set of simulated censored data and the set of simulated exact failure times respectively. They are generated from the simulated model \( f(\cdot) \). Let \( \Delta = \{ \delta : (x_i, \delta_i) \in X \} \) and \( q : \Delta \rightarrow \Delta \) be a random permutation of \( \Delta \). The algorithm from [35] goes as follows.

**ABC-REJ for Censored Data:**

1. Generate a parameter candidate \( \theta^* \) from the prior \( \pi(\theta) \).
2. Simulate \( x^*_j \sim f(\cdot|\theta^*), j = 1, \ldots, n \) and generate \( q(\Delta) \) to get

   \[
   X_s = \{ (x^*_j, \delta^*_j) : j = 1, \ldots, n \}.
   \]

3. Reorder the elements of \( X_s \) by mapping \( q(\Delta) \) back to \( \Delta \) so that

   \[
   X'_s = \{ (x^*(i), \delta_i) : i = 1, \ldots, n \},
   \]

   where \( x^*_j = x^*(i) \) when \( q^{-1}(\delta^*_j) = \delta_i \).

4. Accept \( \theta^* \) if the following conditions are met:

   \[
   (a) \sum_{i=1}^{n} \mathbb{1}[ (x^*(i) - x_i)(1 - \delta_i) > 0 ] = n - \sum_{i=1}^{n} \delta_i
   \]

   \[
   (b) \rho(S(X'_s), S(X')) \leq \varepsilon, \text{ where } \varepsilon \text{ is predetermined threshold.}
   \]
The basic rejection procedure can be improved by using semiautomatic idea described in Chapter 2. The goal of the first part is to determine a new prior distribution $\pi^*(\cdot)$, which is an original prior $\pi(\cdot)$ truncated to the region of non-negligible posterior mass found in the first part. In the second part, the best suitable summary statistic is obtained. Finally, the first part is repeated with new prior and new summary statistic. The advanced algorithm from [35] goes as follows.

**Semi-automatic ABC for Censored Data:**

**Part I: Pilot run, determine $\pi^*(\theta)$:**

1. Generate a parameter candidate $\theta^*$ from the prior $\pi(\theta)$.
2. Simulate $x^*_j \sim f(\cdot | \theta^*), \ j = 1, \ldots, n$ and generate $q(\Delta)$ to get
   \[ X_s = \{(x^*_j, \delta^*_j) : \ j = 1, \ldots, n\}. \]
3. Reorder the elements of $X_s$ by mapping $q(\Delta)$ back to $\Delta$ so that
   \[ X_s = \{(x^r(i), \delta_i) : \ i = 1, \ldots, n\}, \]
   where $x^r_j = x^*(i)$ when $q^{-1}(\delta^*_j) = \delta_i$.
4. Accept $\theta^*$ if the following conditions are met:

   (a) $\sum_{i=1}^{n} 1 \left[ (x^r(i) - x_i)(1 - \delta_i) > 0 \right] = n - \sum_{i=1}^{n} \delta_i$

   (b) $\rho(S(X'_s), S(X')) \leq \varepsilon$, where $\varepsilon$ is predetermined threshold.

**Part II: Determine new summary statistic:**

1. Generate a parameter candidate $\theta^*_k$ from the prior $\pi^*(\theta), \ k = 1, 2, \ldots$.
2. For each $\theta^*_k$ simulate $x^*_j \sim f(\cdot | \theta^*_k), \ j = 1, \ldots, n$ and generate $q(\Delta)$ to get
   \[ X_{sk} = \{(x^r(i), \delta_i) : \ i = 1, \ldots, n\}. \]
3. Compute $S(X'_{s_k})$ if the following condition is met

$$\sum_{i=1}^{n} \mathbb{1}[(x^*(i) - x_i)(1 - \delta_i) > 0] = n - \sum_{i=1}^{n} \delta_i.$$ 

4. Perform regression on the accepted $S(X'_{s_k})$ and corresponding $\theta^*_k$ to get $\hat{\beta} S(\cdot)$.

Part III: Repeat part I using

(a) new prior distribution $\pi^*(\theta)$,
(b) new summary statistic $\hat{\beta} S(\cdot)$,
(c) new tolerance level $\tau < \varepsilon$.

In the second part, step 4, regression can be performed on different subsets of summary statistics, for example on mean and variance, or on mean and standard deviation. Then, using standard model comparison techniques to choose the best combination. This additional step shouldn’t increase by much the time it takes the algorithm to run.

This is a new ABC algorithm that works with censored data. The algorithm works well. For the cases, when the likelihood function is known the results are similar to the ones obtained by Bayesian analysis. The accuracy of estimation using this algorithm depends on time given to run it and the possibilities to determine informative summary statistic.

6.1.1 Examples

In this section ABC rejection modified for censored data (ABC-REJ censored) is illustrated on the following examples.

Example 6.1. Consider a normal model. Samples of size of 25 with 4%, 20%, 40%, and 60% of censored observations are drawn from the normal distribution with mean of 10 and standard deviation of 3. Suppose, the likelihood function is intractable and the true mean needs to be estimated.
Parameter candidates are randomly generated from the uniform prior distribution $U(5, 20)$. New data set is simulated from the normal model given the chosen parameter candidate with the same number of censored data values as the observed sample. The censoring times in the simulated data set are larger than the corresponding censoring times of the observed sample. Euclidean distance metric is used to determine the difference between uncensored data of observed and simulated samples. The tolerance level is 200. Posterior sample size is 100. The mean estimate is a mean of accepted parameter candidates. Mean estimate, mode of density of accepted parameter candidates, 95% highest posterior density (HPD) interval, mean squared error (MSE), bias, acceptance rate, and time obtained by ABC rejection algorithm modified for censored data are presented in the table 6.1. MSE and bias are of accepted parameter candidates. Acceptance rate is the proportion of accepted parameter candidates over all parameter candidates generated from the prior distribution.

<table>
<thead>
<tr>
<th>method</th>
<th>ABC-REJ censored</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
</tr>
<tr>
<td>4%</td>
<td>9.553</td>
</tr>
<tr>
<td>20%</td>
<td>10.696</td>
</tr>
<tr>
<td>40%</td>
<td>12.4184</td>
</tr>
<tr>
<td>60%</td>
<td>13.507</td>
</tr>
</tbody>
</table>

Table 6.1: Summary for normal model using ABC rejection for censored data.

**Example 6.2.** Consider an exponential model. Samples of size of 25 with 4%, 20%, 40%, and 60% of censored observations are drawn from exponential distribution with mean of 0.2. Suppose, the likelihood function is intractable and the true mean needs to be estimated.

Parameter candidates are randomly generated from the uniform prior distribution $U(0, 1)$. New data set is simulated from the exponential model given the chosen parameter candidate with the same number of censored data values as the observed sample. The censoring times in the simulated data set are larger than the corresponding censoring times of the observed sample. Euclidean distance metric is used to determine the difference between uncensored data of observed and simulated samples. The tolerance level is 0.5. Posterior sample size is 100. The mean estimate is a mean of accepted parameter candidates. Mean estimate, mode of density of accepted parameter candidates, 95% highest posterior density (HPD) interval, mean squared
error (MSE), bias, acceptance rate, and time obtained by ABC rejection for right censored data algorithm are presented in the table 6.2. MSE and bias are of accepted parameter candidates. Acceptance rate is the proportion of accepted parameter candidates over all parameter candidates generated from the prior distribution.

<table>
<thead>
<tr>
<th>method</th>
<th>ABC-REJ censored</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS</td>
<td>censored</td>
</tr>
<tr>
<td>25</td>
<td>4%</td>
</tr>
<tr>
<td>25</td>
<td>20%</td>
</tr>
<tr>
<td>25</td>
<td>40%</td>
</tr>
<tr>
<td>25</td>
<td>60%</td>
</tr>
</tbody>
</table>

Table 6.2: Summary for exponential model using ABC rejection for censored data.

Results in tables 6.1 and 6.2 show that if proportion of censored observations rises, MSE and bias become larger for fixed sample size, since it is harder to make accurate estimation. For 60% of censored data values the estimation are the least accurate, MSE and bias are the largest. Acceptance rate is very low for all cases.

## 6.2 ABC-EL for Censored Data

In this section ABC via empirical likelihood method for censored data is described. The parameter estimate is the weighted average of all parameter candidates randomly chosen from the prior distribution, where the weights are determined by empirical likelihood procedure. The crucial part of it is the calculations of the weights for each observation given parameter candidate under the mean constraint. Once the mean constraint is added, these computations are not trivial. The sequential quadratic programming (SQP) method by Chen and Zhou [54, 55] and the way with implementing EM algorithm [56] were proposed as a solution.
Suppose, that \(X_1, \ldots, X_n\) is a sample from cumulative distribution function \(F\), and \(Y_1, \ldots, Y_n\) are right censoring times. Assume that \(X_1, \ldots, X_n\) are independent. Let \(Z_i = \min(X_i, Y_i)\) and \(\delta_i = 1_{X_i \leq Y_i}\) indicate an uncensored event when \(\delta_i = 1\), for \(i = 1, \ldots, n\). The log of empirical likelihood function for \((Z_i, \delta_i)\) is

\[
\log L(w) = \sum_{i=1}^{n} \left[ \delta_i \log w_i + (1 - \delta_i) \log \left( \sum_{Z_j > Z_i} w_j \right) \right].
\]

The maximum of this function under the mean constraint needs to be found. The constraints are

\[
\sum_{i=1}^{n} w_i Z_i \delta_i = \mu, \quad \sum_{i=1}^{n} w_i \delta_i = 1, \quad w_i \geq 0,
\]

where \(\mu\) is known. The straight implementation of Lagrange multiplier approach gives the following equations

\[
\frac{\delta_i}{w_i} + \sum_{k=1}^{n} (1 - \delta_k) \frac{1_{Z_k < Z_i}}{\sum_{Z_j > Z_k} w_j} - \lambda Z_i \delta_i - \gamma \delta_i = 0.
\]

There is no straightforward easy solution for \(w_i\)’s, and SQP or EM methods need to be applied with limitation that the largest observation must be uncensored.

### 6.2.1 EL weights by Sequential Quadratic Programming

The goal is to determine the weights which will maximize the empirical likelihood function for the right censored data with the mean constraint. The maximization of the log likelihood problem for the censored case is more complicated then straight use of Lagrange multiplier as in uncensored case. SQP procedure is recommended to implement instead. Once we can maximize the log likelihood with mean constraint for right censored data, we will use it in finding empirical likelihood ratio function.

SQP method as described in [11] and [12] starts with some initial values of weights \(w_i\)’s. It finds the minimum of negative of the target function \(\log L(w)\), but instead of \(\log L(w)\) it first uses some quadratic function with the same first and second derivatives at the initial weights, finds its minimum with respect to the given constraints. After that, the quadratic function is updated and now has the same first and second
derivatives as $-\log L(w)$ at the new weights. The process repeats until some predetermined convergence criterion is satisfied. The solution will be the last updated weights.

Once we are able to find maximum of log likelihood function under constraints, it is easy to calculate the empirical likelihood ratio function $R$. Consider the following test statistic:

\[-2\log R(c) = -2\log \frac{\max_c L(w)}{\max L(w)}\]
\[= 2\left[\log(\max L(w)) - \log(\max_c L(w))\right]\]
\[= 2[\log(L(\tilde{w})) - \log(L(\hat{w}))],\]

where $\tilde{w}$ is the NPMLE of weights without any constraints, $\hat{w}$ is the NPMLE of weights under constraint $c$. Each estimate may be determined by SQP method. The asymptotic distribution of this test statistic is known (see 3.7). Thus, we are able to test if some value of the mean is a suitable for a given sample.

The disadvantage of SQP is that it becomes more computationally intensive for large sample sizes.

### 6.2.2 EL weights by EM algorithm

Expectation maximization algorithm is a statistical method for estimating parameters when the model depends on some unobserved variables. It is an iterative procedure, each iteration consists of two steps: calculations of the expectation of the log-likelihood and then parameters for maximizing the expected log-likelihood. It ends when the predetermined convergence condition is satisfied. Dempster, Laird, and Rubin presented EM algorithm for more general cases and showed its convergence [15]. Turnbull considered EM algorithm for nonparametric estimation of a distribution function when the data is censored [50].

Zhou in [55] introduced EM algorithm for censored data under the mean constraint which can find the NPMLE of cumulative distribution function (CDF). As consequence, it allows to compute empirical likelihood ratio function. For the case with right censored data, it is recommended to start with Kaplan-Meier estimator for initial estimate of CDF $F$. In the expectation step, the weight $p_j$ at location $t_j$ for given $F$
should be computed as

\[ p_j = \sum_{i=1}^{n} E_F[\mathbb{1}_{X_i = t_j} | Z_i, \delta_i] . \]

If the \( i \)-th observation is right censored, then

\[ E_F[\mathbb{1}_{X_i = t_j} | Z_i, \delta_i] = \begin{cases} \Delta F(t_j) / (1 - F(Z_i)) & \text{for } t_j > Z_i, \\ 0 & \text{otherwise}. \end{cases} \]

If the \( i \)-th observation is uncensored, then

\[ E_F[\mathbb{1}_{X_i = t_j} | Z_i, \delta_i] = \begin{cases} 1 & \text{if } t_j = Z_i, \\ 0 & \text{otherwise}. \end{cases} \]

So, the weight is calculated for the locations with a jump point for given \( F \) or for uncensored observation.

In the maximization step, the observations \( X = t_j \)'s with the weights \( p_j \) from the expectation step are counted to be uncensored. The empirical likelihood technique for uncensored data can be applied to find probabilities that maximize log of empirical likelihood function. Then, \( F \) gets to be updated, and the process repeats. The algorithm stops when the predetermined convergence condition is satisfied. The final \( p_j \)'s are the wanted weights.

The advantage of EM algorithm is that it doesn’t require a lot of memory, it is time efficient.

### 6.2.3 Algorithm

Suppose, that \( X_1, \ldots, X_n \) is a sample from cumulative distribution function \( F \), and \( Y_1, \ldots, Y_n \) are right censoring times. Assume that \( X_1, \ldots, X_n \) are independent. Let \( Z_i = \min(X_i, Y_i) \) and \( \delta_i = \mathbb{1}_{X_i \leq Y_i} \) indicate an uncensored event when \( \delta_i = 1 \), for \( i = 1, \ldots, n \). The log of empirical likelihood function for \((Z_i, \delta_i)\) is

\[
\log L(w) = \sum_{i=1}^{n} \left[ \delta_i \log w_i + (1 - \delta_i) \log \left( \sum_{j > Z_i} w_j \right) \right]. \tag{6.1}
\]
The mean constraints are
\[ \sum_{i=1}^{n} w_i Z_i \delta_i = \mu, \quad \sum_{i=1}^{n} w_i \delta_i = 1, \quad w_i \geq 0, \] (6.2)
where the candidate for \( \mu \) is generated from the prior distribution.

**ABC via Empirical Likelihood Sampler for Right Censored Data:**

1. Set \( j = 1 \).

2. Generate a parameter candidate \( \mu^*_j \) from the prior \( \pi(\mu) \).

3. Compute \( w_i \)'s, \( i = 1, n \), which maximize log likelihood (6.1) under constraints (6.2).

4. Calculate \( p_j = \prod_{i=1}^{n} w_i \).

5. If \( j < J \), increment \( j = j + 1 \) and go to step 2.

6. Normalize the weights, \( p^*_j = \frac{p_j}{\sum_{j=1}^{J} p_j} \) for \( j = 1, ..., J \).

7. Take \( \mu^*_j \) with corresponding weight \( p^*_j \).

This algorithm can be used if some other parameter \( \theta \) needs to be estimated. The dependence of the mean from the parameter \( \theta \) is required to be known, such as some function \( g(\cdot) : \mu = g(\theta) \). Thus, this algorithm is recommended to implement if such function \( g(\cdot) \) can be found and the largest observed value is not censored.

### 6.2.4 Examples

In this section ABC-EL for censored data via sequential quadratic programming (SQP) and expectation—maximization (EM) methods are illustrated on the following examples.
Example 6.3. Consider a normal model. Samples with 4%, 20%, 40%, and 60% of censored observations and of the size of 25 are drawn from normal distribution with mean of 10 and standard deviation of 3. Suppose, the likelihood function is intractable and the true mean needs to be estimated.

Parameter candidates are randomly generated from the uniform prior distribution $U(7,11)$. For each of them weight is calculated as a product of empirical likelihood weights found by SQP and EM techniques. Posterior sample size is 100. The mean estimate is a weighted mean of parameter candidates and their normalized weights used as probabilities. Time in seconds, mean squared error (MSE), and bias for ABC-EL via EM and ABC-EL via SQP are presented in the table 6.3. MSE and bias are calculated based on 25 runs of the algorithms.

<table>
<thead>
<tr>
<th>Normal model</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>method</td>
<td>ABC-EL via SQP</td>
<td>ABC-EL via EM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>censored</td>
<td>Time</td>
<td>MSE</td>
<td>Bias</td>
<td>Time</td>
</tr>
<tr>
<td>4%</td>
<td>0.4</td>
<td>0.7895</td>
<td>0.888</td>
<td>0.4</td>
</tr>
<tr>
<td>20%</td>
<td>0.4</td>
<td>0.9939</td>
<td>0.996</td>
<td>0.5</td>
</tr>
<tr>
<td>40%</td>
<td>0.6</td>
<td>1.2358</td>
<td>1.111</td>
<td>0.7</td>
</tr>
<tr>
<td>60%</td>
<td>0.5</td>
<td>1.4008</td>
<td>1.182</td>
<td>0.7</td>
</tr>
</tbody>
</table>

Table 6.3: Summary for normal model using ABC-EL for right censored data via SQP and EM.

Example 6.4. Consider an exponential model. Samples with 4%, 20%, 40%, and 60% of censored observations and of the size of 25 are drawn from exponential distribution with mean of 0.2. Suppose, the likelihood function is intractable and the true mean needs to be estimated.

Parameter candidates are randomly generated from the uniform prior distribution $U(0,0.6)$. For each of them weight is calculated as a product of empirical likelihood weights found by SQP and EM techniques. Posterior sample size is 100. The mean estimate is a weighted mean of parameter candidates and their normalized weights used as probabilities. Time in seconds, mean squared error (MSE), and bias for ABC-EL via EM and ABC-EL via SQP are presented in the table 6.4. MSE and bias are calculated based on 25 runs of the algorithms.

Tables 6.3 and 6.4 in both examples show that ABC-EL via SQP and ABC-EL via EM are fast methods. In most cases, ABC-EL via SQP produces better estimates than ABC-EL via EM since its MSEs and
Example 6.1 (continued). Consider a normal model. Parameter candidates are randomly generated from the uniform prior distribution $U(5, 20)$. Posterior sample size is 100. Time in seconds, mean squared error (MSE), and bias for ABC-EL via SQP and ABC-REJ for censored data are presented in the table 6.5. MSE and bias are of the posterior distributions.

Example 6.2 (continued). Consider an exponential model. Parameter candidates are randomly generated from the uniform prior distribution $U(0, 1)$. Posterior sample size is 100. Time, mean squared error (MSE), and bias for ABC-EL via SQP and ABC-REJ for censored data are presented in the table 6.6. MSE and bias are of the posterior distribution.

Tables 6.5 and 6.6 in both examples show that ABC-EL via SQP takes less time to run than ABC-REJ for censored data. In most cases, ABC-EL via SQP produces better estimates than ABC-REJ for censored data since its MSEs and biases are smaller. As proportion of censored observations increases, MSE and bias become larger for both algorithms, but for ABC-EL via SQP they rise much slower than for ABC-REJ for
Table 6.6: Summary for exponential model using ABC-EL via SQP and ABC-REJ for right censored data.

censored data. Note that ABC-EL via SQP is recommended to use only if the mean constraint is known and the largest observation is not censored.

6.3 ABC Algorithm for Censored Data via Comparing Survival Functions

In this section new ABC algorithm that can be used for censored data is performed via comparing survival functions of observed and simulated samples. This method does not require the choice of summary statistics, distance metric, or simulation process of censored data. Summary statistics are not needed at all, instead all observations are employed. Distance metric is replaced with hypothesis test for the equality of two survival functions. Exact failure times rather than censored observations is sufficient to use for simulated data set. There are two versions of the this algorithm, for discrete and continuous data. Both algorithms are accurate and more time efficient than ABC rejection for censored data described before.

6.3.1 The Two-stage Hypothesis Test

The central part of ABC via comparing survival functions is to be able to test effectively if the two survivals are the same. The most popular test to use for this purpose is the log-rank test due to its high power, see [24]. However, the log-rank test works well only under proportional hazard assumption, the hazard functions must be proportional over time. In practice, this assumption is often violated, then other statistical test should be applied.
Li, et al. in [30] performed comparison of methods for testing if two survival curves are the same under various patterns of crossing survival functions. In the study, different sample sizes, censoring rates were considered. Situations of crossing survivals at early, middle, and late times were looked at in this work. They found that the location of the crossing point affects the discrimination power of the test statistic, and high censoring rates affect the power. It had been determined that the adaptive Neyman’s smooth tests [27] and the two-stage test, introduced by Qiu and Sheng in [44], have higher power and stability, when survival functions cross at early, middle or late times. When the proportional hazard assumption is satisfied, these methods work very well compare to log-rank test. Also, these tests are found to be more stable when the censoring rate is high.

Two-stage test is used within the new ABC algorithm proposed here. Methods based on survival curves are more sensitive in terms of capturing differences [31]. However, these tests are designed to detect crossings, and lose power if the two survival functions are different but do not cross. The two-stage test we have chosen does not suffer from this problem [44]. The null hypothesis is that two survival functions are the same. The alternative hypothesis is that they are not the same, which include different patterns, such as crossing and running parallel. This is the advantages of the two-stage test in contrast with other tests for comparing survival functions. In the first stage the log-rank test is applied. It differentiates two cases, either survival functions are different with or without crossing, either survival functions are identical with or without crossing. If the null hypothesis is rejected after the first stage, then the two-stage test ends, and the conclusion is that the two survival functions are significantly different. Otherwise, the second stage for detecting crossings is applied. The method used for this purpose is based on the weighted-log rank test, but Qiu and Sheng have proposed a new weighting scheme where test statistic asymptotically independent of the test statistic of the log-rank test used in stage one [44].

Let \( \alpha \) represent the overall significance level of the two-stage test, and let \( \alpha_1 \) and \( \alpha_2 \) represent the significance level for the stage one and stage two tests, respectively. Since the two stages are asymptotically independent, \( \alpha_1 + \alpha_2 (1 - \alpha_1) = \alpha \). Let \( p_1 \) and \( p_2 \) be the p-values computed during stage one and stage two, respectively. Then the overall p-value for the two-stage test is given below. Note that \( \alpha_1 + p_2 (1 - \alpha_1) \leq \alpha \) is equivalent to \( p_2 \leq \alpha_2 \).
p-value = \begin{cases} 
    p_1, & \text{if } p_1 \leq \alpha_1, \\
    \alpha_1 + p_2(1 - \alpha_1), & \text{otherwise.} 
\end{cases}

The two stage test is for continuous data and may not be suitable for discrete data. While it is common for continuous-time models to be used for discrete data, we provide a separate algorithm for each case.

### 6.3.2 Algorithm

New ABC algorithm for continuous censored data via comparing survival curves goes as follows. First, the parameter candidate is generated from the prior distribution. Then, the sample is simulated from the model given the parameter candidate. This sample doesn’t need to be censored due to the specificity of the two-stage test implemented within algorithm. Next, the two-stage hypothesis test is performed for comparing survival functions of observed and simulated data sets. The parameter candidate is accepted if the null hypothesis that these survival functions are the same cannot be rejected. Below the algorithm is described for continuous data.

**ABC via comparing survivals: continuous case**

1. Draw $\theta^*$ from the prior distribution $\pi(\theta)$
2. Simulate $X^* \sim f(\cdot | \theta^*)$
3. Perform two-stage hypothesis test:
   (a) For observed set $X$ and $X^*$, perform stage one to obtain $p_1$
   (b) If $p_1 > \alpha_1$, then perform stage two to obtain $p_2$
4. Accept $\theta^*$ if $p_2 > \alpha_2$

Unfortunately, two-stage hypothesis test is applicable only for the continuous data. To be able to use ABC via comparing survival functions algorithm for the discrete case, we must make an adjustment to the data
by adding some continuous random variable to both observed and simulated data sets. The following two lemmas justify the validity of the procedure.

**Lemma 1.** Suppose $X_1, X_2$ and $U$ are independent random variables. Let $Y = \min(X_1, X_2)$. Then $Y + U$ is stochastically equivalent to $\min(X_1 + U, X_2 + U)$, i.e. $Y + U \stackrel{st}{=} \min(X_1 + U, X_2 + U)$.

**Proof.** It is clear that

$$P(Y > y) = P(X_1 > y)P(X_2 > y).$$

Then,

$$P(Y + U) = \int P(Y > y - u)f_U(u)du = \int P(X_1 > y - u)P(X_2 > y - u)f_U(u)du.$$

Also,

$$P(\min(X_1 + U, X_2 + U) > y) = \int P(\min(X_1, X_2) > y - u)f_U(u)du$$

$$= \int P(X_1 > y - u)P(X_2 > y - u)f_U(u)du.$$

Thus,

$$P(Y + U > y) = P(\min(X_1 + U, X_2 + U) > y)).$$

\[\square\]

**Lemma 2.** Suppose, $X_1$ and $X_2$ are two discrete random variables, $U_1$ and $U_2$ are continuous random variables with common cumulative distribution function independent of $X_1$ and $X_2$ respectively. Define $Y_1 = X_1 + U_1$ and $Y_2 = X_2 + U_2$. Then $Y_1 \stackrel{st}{=} Y_2$ iff $X_1 \stackrel{st}{=} X_2$.

**Proof.** The moment generating function of $Y_i$ is

$$M_{Y_i}(t) = M_{X_i}(t)M_{U_i}(t) \text{ for } i = 1, 2.$$ 

If $Y_1 \stackrel{st}{=} Y_2$, then

$$M_{Y_1}(t) = M_{X_1}(t)M_{U_1}(t) = M_{X_2}(t)M_{U_2}(t) = M_{Y_2}(t).$$

This implies

$$M_{X_1}(t) = M_{X_2}(t) \text{ and thus } X_1 \stackrel{st}{=} X_2.$$
If $X_1 \overset{d}{=} X_2$, then

$$M_{X_1}(t) = M_{X_2}(t),$$

and thus $Y_1 \overset{d}{=} Y_2$.

New ABC algorithm for discrete censored data via comparing survival curves goes as follows. The observed sample values need to be modified by adding some continuous random variable first. Second, the parameter candidate is generated from the prior distribution. Then, the sample is simulated from the model given the parameter candidate. This sample doesn’t need to be censored due to the specificity of the two-stage test implemented within algorithm. After, the sample values should be modified in the similar fashion as the observed sample. Next, the two-stage hypothesis test is performed for comparing survival functions of modified observed and simulated data sets. The parameter candidate is accepted if the null hypothesis that these survival functions are the same cannot be rejected.

Let $X_{\text{mod}} = X + U_1$, where $X$ is discrete and $U_1$ is a continuous random variable that is independent of $Y$ and $Z$. Applying Lemmas 1 and 2, $X_{\text{mod}}$ can be treated as a continuous random variable and we use it as our observed data set. We choose a $U_1$ such that the adjustment to $X$ is small. We make the same type of adjustment to the simulated data $X^*$ and obtain $X^*_{\text{mod}} = X^* + U_2$. Below the algorithm is described for discrete data.

**ABC via comparing survivals: discrete case**

1. Compute $X_{\text{mod}} = X + U_1$

2. Draw $\theta^*$ from the prior $\pi(\theta)$

3. Simulate $X^* \sim f(\cdot|\theta^*)$, then compute $X^*_{\text{mod}} = X^* + U_2$

4. Two-stage hypothesis test:
   
   (a) Using $X_{\text{mod}}$ and $X^*_{\text{mod}}$, perform stage one to obtain $p_1$
   
   (b) If $p_1 > \alpha_1$, then perform stage two to obtain $p_2$

5. Accept $\theta^*$, if $p_2 > \alpha_2$
6.3.3 Asymptotically Unbiased Estimator

In this section it is shown that the estimator obtained by ABC via comparing survival functions method is asymptotically unbiased.

Suppose we are testing $H_0$ against $H_a$ and we have come up with a test statistic where for large $n$, the power goes to one. Given that $H_0$ is true, it is known that $\hat{\theta}$ is an asymptotically unbiased estimator of $\theta$. We show that $\hat{\theta}$ is still an asymptotically unbiased estimator if given that $H_0$ is accepted, rather than known to be true.

Let $A$ represent the event that $H_0$ is accepted. Let $T$ represent the event that $H_0$ is true, while $F$ represents $H_0$ is false, i.e. the complement of $T$. Note that the power of a test is $1 - \beta$, where $\beta = P(A|F)$. Let us look at the bias first.

\[
E(\hat{\theta}|A) = E(\hat{\theta}|A \cap T)P(T|A) + E(\hat{\theta}|A \cap F)P(F|A)
\]

\[
= E(\hat{\theta}|A \cap T)\left[1 - P(F|A)\right] + E(\hat{\theta}|A \cap F)P(F|A)
\]

\[
= E(\hat{\theta}|A \cap T)\left[1 - \frac{P(A|F)P(F)}{P(A)}\right] + E(\hat{\theta}|A \cap F)\frac{P(A|F)P(F)}{P(A)}
\]

\[
= E(\hat{\theta}|A \cap T)\left[1 - \frac{\beta P(F)}{P(A)}\right] + E(\hat{\theta}|A \cap F)\frac{\beta P(F)}{P(A)}.
\]

Since $\hat{\theta}$ is an asymptotically unbiased estimator of $\theta$ when $H_0$ is true, $E(\hat{\theta}|A \cap T) \to \theta$ as $n \to \infty$. The power of the test goes to one, so $\beta \to 0$ as $n \to \infty$.

Therefore,

\[
E(\hat{\theta}|A) \to \theta \text{ as } n \to \infty.
\]

Hence, the $\hat{\theta}$ given that the null hypothesis is accepted is asymptotically unbiased.
Let $S_X(x)$ and $S_{X^*}(x)$ represent the survival functions for the observed and simulated data, respectively, where $X^*$ is generated from the model given $\theta^*$. The two-stage test is used in our ABC algorithm as the acceptance criterion. It is known that the power of the test approaches one as sample sizes $n_1$ and $n_2$ approach infinity. (Without loss of generality assume $n_1 = n_2 = n$.) The null hypothesis is:

$$H_0 : S_X(x) = S_{X^*}(x).$$

Let $N = 1$. Then the output of our ABC algorithm would be $\theta^*_1$. This candidate value of $\theta$ was only accepted, because the null hypothesis of the two-stage test was not rejected. By the above result, we know that $\theta^*_1$ is an asymptotically unbiased estimator for $\theta$.

Let $N > 1$ and $A_i$ represent the event that $\theta^*_i$ was accepted and therefore $H_0$ was accepted for the corresponding two-stage test. Then, the output of the ABC algorithm would be the sequence $\{\theta^*_i|A_i\}_{i=1}^N$. The ABC estimator for $\theta$ will be the mean of the values in a sequence:

$$\bar{\theta}^* = \frac{1}{N} \sum_{i=1}^N \theta^*_i | A_i.$$

Since all values in a sequence are asymptotically unbiased estimators of $\theta$, the ABC estimator also will be asymptotically unbiased estimator of $\theta$.

**6.3.4 Consistency**

Now we will show that the estimator obtained by ABC via comparing survival functions method is consistent.

Suppose we are testing $H_0$ against $H_a$ and we have come up with a test statistic where for large $n$, the power goes to one. Given that $H_0$ is true, it is known that $\hat{\theta}$ is a consistent and asymptotically unbiased estimator of $\theta$. We show that $\bar{\theta}$ is still a consistent estimator if given that $H_0$ is accepted, rather than known to be true.
Let $A$ represent the event that $H_0$ is accepted. Let $T$ represent the event that $H_0$ is true, while $F$ represents $H_0$ is false, i.e. the complement of $T$. Note that the power of a test is $1 - \beta$, where $\beta = P(A|F)$. Let us look at the variance using the total conditional variance formula:

$$V(\hat{\theta}|A) = E\left[V(\hat{\theta}|A) \bigg| A\right] + V\left[E(\hat{\theta}|A) \bigg| A\right].$$ \hfill (6.3)

Consider the second term in 3.9. Since $\hat{\theta}|A$ is asymptotically unbiased for $\theta$ as shown in the previous section, we have

$$V\left[E(\hat{\theta}|A) \bigg| A\right] \to V[\theta|A] = 0 \text{ as } n \to \infty.$$

Now consider first term in (27).

$$E\left[V(\hat{\theta}|A) \bigg| A\right] = V(\hat{\theta}|A \cap T)P(T|A) + V(\hat{\theta}|A \cap F)P(F|A)$$

$$= V(\hat{\theta}|A \cap T)\left[1 - P(F|A)\right] + V(\hat{\theta}|A \cap F)P(F|A)$$

$$= V(\hat{\theta}|A \cap T)\left[1 - \frac{P(A|F)P(F)}{P(A)}\right] + V(\hat{\theta}|A \cap F)\frac{P(A|F)P(F)}{P(A)}$$

$$= V(\hat{\theta}|A \cap T)\left[1 - \frac{\beta P(F)}{P(A)}\right] + V(\hat{\theta}|A \cap F)\frac{\beta P(F)}{P(A)}.$$

We know that $V(\hat{\theta}|A \cap T) \to 0$ as $n \to \infty$, because $\hat{\theta}$ is a consistent estimator when $H_0$ is true. The power of the test goes to one, so $\beta \to 0$ as $n \to \infty$. Therefore,

$$V(\hat{\theta}|A) \to 0 \text{ as } n \to \infty.$$

Our estimator asymptotically unbiased and variance of it approaches zero as sample size increases to infinity. Thus, $\hat{\theta}$ is a consistent estimator if $H_0$ is accepted by a sufficient condition.
The output of our ABC algorithm would be the sequence of $\theta^*$’s. Each $\theta^*$, the candidate value of $\theta$, was only accepted because the null hypothesis of the two-stage test was not rejected. By the above result, we know that each $\theta^*$ is a consistent estimator for $\theta$.

Let $A_i$ for $i = 1, \ldots, n$ represent the event that $\theta_i^*$ was accepted by the two-stage test. Then, the output of the ABC algorithm would be the sequence $\{\theta_i^* | A_i\}_{i=1}^N$. The ABC estimator for $\theta$ will be the mean of the values in a sequence:

$$\bar{\theta}^* = \frac{1}{N} \sum_{i=1}^N \theta_i^* | A_i.$$  

Since all values in a sequence are consistent estimators of $\theta$, the ABC estimator also will be a consistent estimator of $\theta$.

### 6.3.5 Asymptotic behavior of Approximate Posterior Distribution

Here we will show what is the algorithm’s resulting distribution of the parameter. Let $\{\theta_i^*\}_{i=1}^N$ be the resulting sequence of the algorithm. Each $\theta_i^*$ is independent draw from $f(\theta | H_0 \text{ is accepted})$. Then,

$$f(\theta_i^*) \propto \sum_{x^* \in \mathcal{F}} f(x^*) | \theta_i^* \pi(\theta_i^*)^\mathbb{1}_{H_0 \text{ accepted}} \sum_{x^*: H_0 \text{ accepted}} f(x^* | \theta_i^*) \pi(\theta_i^*) \propto \pi_{H_0}(\theta_i^* | x),$$

where $\mathcal{F}$ is $\sigma$-algebra on some given set $\Omega$, $x^*$ is simulated data, $f(x^* | \cdot)$ is the model for simulated data, and $\pi(\cdot)$ is prior distribution. The resulting approximate posterior distribution $\pi_{H_0}(\theta_i^* | x)$ depends on the test which helps to make a decision to reject the null hypothesis or not.

As sample size of the observed data set approaches infinity, the sample size of the simulated data set approaches infinity. Then the power of the tests (Log rank or two stage test) approaches one,

$$1 - \beta = \Pr(H_1 \text{ is accepted} \mid H_0 \text{ is false}) \to 1.$$  

Meaning that, when, in reality, null hypothesis is false, the alternative hypothesis will be accepted and the corresponding parameter value will not be taken into the resulting sequence of parameter values. Thus, the
wrong parameter values will not be taken into the resulting sequence of parameter values at all. Hence, the resulting posterior distribution will be a true posterior,

\[ \pi_{H_0}(\theta^*_i|x) \rightarrow \pi(\theta^*_i|x), \]

as sample sizes of observed and simulated data sets approach infinity.

### 6.3.6 Examples

In this section ABC via comparing survival functions (ABC via survivals) is illustrated on the following examples.

**Example 6.1** (continued). Consider a normal model. This time the samples of different sizes such as 25, 50, and 100 are considered. Parameter candidates are randomly generated from the uniform prior distribution \( U(5,20) \). New data set is simulated from the normal model based on the chosen parameter candidate. Observed and simulated sample’s survival functions are compared by two-stage test. The tolerance level for the first stage of the test is chosen to be 0.2 and for the second stage is 0.6. Posterior sample size is 100. The mean estimate is a mean of accepted parameter candidates. Mean estimate, mode of density of accepted parameter candidates, 95% highest posterior density (HPD) interval, mean squared error (MSE), bias, acceptance rate, and time obtained by ABC via survivals algorithm are presented in the table 6.7. MSE and bias are of accepted parameter candidates. Acceptance rate is the proportion of accepted parameter candidates over all parameter candidates generated from the prior distribution.

**Example 6.2** (continued). Consider an exponential model. This time the samples of different sizes such as 25, 50, and 100 are considered. Parameter candidates are randomly generated from the uniform prior distribution \( U(0,1) \). New data set is simulated from the exponential model based on the chosen parameter candidate. Observed and simulated sample’s survival functions are compared by two-stage test. The tolerance level for the first stage of the test is chosen to be 0.2 and for the second stage is 0.6. Posterior sample size is 100. The mean estimate is a mean of accepted parameter candidates. Mean estimate, mode of density of accepted parameter candidates, 95% highest posterior density (HPD) interval, mean squared
Table 6.7: Summary for normal model using ABC via survivals algorithm.

<table>
<thead>
<tr>
<th>SS</th>
<th>Censored</th>
<th>Method</th>
<th>Mean</th>
<th>Mode</th>
<th>95% HPD Interval</th>
<th>MSE</th>
<th>Bias</th>
<th>Accept. Rate</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>4%</td>
<td>ABC</td>
<td>9.056</td>
<td>9.347</td>
<td>(6.257,10.750)</td>
<td>2.1146</td>
<td>1.070</td>
<td>0.0195</td>
<td>24 min</td>
</tr>
<tr>
<td>25</td>
<td>20%</td>
<td>ABC</td>
<td>9.356</td>
<td>9.632</td>
<td>(5.442,11.379)</td>
<td>2.3440</td>
<td>1.070</td>
<td>0.0870</td>
<td>5 min</td>
</tr>
<tr>
<td>25</td>
<td>40%</td>
<td>ABC</td>
<td>11.008</td>
<td>11.223</td>
<td>(5.986,12.929)</td>
<td>3.5154</td>
<td>1.573</td>
<td>0.0683</td>
<td>6 min</td>
</tr>
<tr>
<td>25</td>
<td>60%</td>
<td>ABC</td>
<td>12.136</td>
<td>12.465</td>
<td>(9.443,14.159)</td>
<td>6.8127</td>
<td>2.484</td>
<td>0.0591</td>
<td>6 min</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SS</th>
<th>Censored</th>
<th>Method</th>
<th>Mean</th>
<th>Mode</th>
<th>95% HPD Interval</th>
<th>MSE</th>
<th>Bias</th>
<th>Accept. Rate</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>4%</td>
<td>ABC</td>
<td>9.222</td>
<td>9.715</td>
<td>(6.459,10.731)</td>
<td>1.8725</td>
<td>0.886</td>
<td>0.0111</td>
<td>3.5 hr</td>
</tr>
<tr>
<td>50</td>
<td>20%</td>
<td>ABC</td>
<td>9.502</td>
<td>9.641</td>
<td>(5.959,11.299)</td>
<td>1.9240</td>
<td>0.851</td>
<td>0.0562</td>
<td>25 min</td>
</tr>
<tr>
<td>50</td>
<td>40%</td>
<td>ABC</td>
<td>10.438</td>
<td>10.947</td>
<td>(5.630,12.070)</td>
<td>2.7143</td>
<td>1.273</td>
<td>0.0522</td>
<td>24 min</td>
</tr>
<tr>
<td>50</td>
<td>60%</td>
<td>ABC</td>
<td>10.970</td>
<td>11.435</td>
<td>(7.292,12.688)</td>
<td>2.8867</td>
<td>1.438</td>
<td>0.0487</td>
<td>31 min</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SS</th>
<th>Censored</th>
<th>Method</th>
<th>Mean</th>
<th>Mode</th>
<th>95% HPD Interval</th>
<th>MSE</th>
<th>Bias</th>
<th>Accept. Rate</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>4%</td>
<td>ABC</td>
<td>9.728</td>
<td>9.654</td>
<td>(9.019,10.583)</td>
<td>0.2583</td>
<td>0.419</td>
<td>0.0290</td>
<td>5 hr</td>
</tr>
<tr>
<td>100</td>
<td>20%</td>
<td>ABC</td>
<td>10.342</td>
<td>10.417</td>
<td>(9.401,11.092)</td>
<td>0.3053</td>
<td>0.446</td>
<td>0.0376</td>
<td>5 hr</td>
</tr>
<tr>
<td>100</td>
<td>40%</td>
<td>ABC</td>
<td>10.917</td>
<td>10.967</td>
<td>(10.281,11.715)</td>
<td>0.9750</td>
<td>0.920</td>
<td>0.0338</td>
<td>5 hr</td>
</tr>
<tr>
<td>100</td>
<td>60%</td>
<td>ABC</td>
<td>10.919</td>
<td>9.414</td>
<td>(10.114,11.616)</td>
<td>0.9918</td>
<td>0.920</td>
<td>0.0333</td>
<td>6 hr</td>
</tr>
</tbody>
</table>

Results in tables 6.7 and 6.8 show that when sample size increases, MSE and bias get smaller for a fixed percentage of censored observations as expected. If proportion of censored observations rises, MSE and bias become larger for fixed sample size, since it is harder to make accurate estimation.

### 6.4 ABC Algorithm for Censored Data via Comparing Smoothed Distribution Functions

In this section new ABC algorithm that can be used for censored data is performed via comparing smoothed distribution functions of observed and simulated samples. This method also does not require the choice of summary statistics, distance metric, or simulation process of censored data. Summary statistics are not needed at all, instead all observations are employed. Distance metric is replaced with hypothesis error (MSE), bias, acceptance rate, and time obtained by ABC via survivals algorithm are presented in the table 6.8. MSE and bias are of accepted parameter candidates. Acceptance rate is the proportion of accepted parameter candidates over all parameter candidates generated from the prior distribution.
<table>
<thead>
<tr>
<th>SS</th>
<th>censored</th>
<th>Mean</th>
<th>Mode</th>
<th>95% HPD interval</th>
<th>MSE</th>
<th>Bias</th>
<th>Accept. Rate</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>4%</td>
<td>0.255</td>
<td>0.239</td>
<td>(0.122,0.407)</td>
<td>0.0090</td>
<td>0.072</td>
<td>0.0742</td>
<td>11 min</td>
</tr>
<tr>
<td>25</td>
<td>20%</td>
<td>0.330</td>
<td>0.276</td>
<td>(0.179,0.528)</td>
<td>0.0273</td>
<td>0.132</td>
<td>0.125</td>
<td>5.5 min</td>
</tr>
<tr>
<td>25</td>
<td>40%</td>
<td>0.343</td>
<td>0.259</td>
<td>(0.157,0.697)</td>
<td>0.0403</td>
<td>0.148</td>
<td>0.0485</td>
<td>12 min</td>
</tr>
<tr>
<td>25</td>
<td>60%</td>
<td>0.350</td>
<td>0.245</td>
<td>(0.138,0.657)</td>
<td>0.0461</td>
<td>0.157</td>
<td>0.2123</td>
<td>2 min</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SS</th>
<th>censored</th>
<th>Mean</th>
<th>Mode</th>
<th>95% HPD interval</th>
<th>MSE</th>
<th>Bias</th>
<th>Accept. Rate</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>4%</td>
<td>0.242</td>
<td>0.267</td>
<td>(0.147,0.346)</td>
<td>0.0049</td>
<td>0.056</td>
<td>0.043</td>
<td>1 hr</td>
</tr>
<tr>
<td>50</td>
<td>20%</td>
<td>0.298</td>
<td>0.276</td>
<td>(0.158,0.425)</td>
<td>0.0147</td>
<td>0.100</td>
<td>0.0517</td>
<td>47 min</td>
</tr>
<tr>
<td>50</td>
<td>40%</td>
<td>0.321</td>
<td>0.293</td>
<td>(0.176,0.515)</td>
<td>0.0229</td>
<td>0.123</td>
<td>0.1013</td>
<td>10 min</td>
</tr>
<tr>
<td>50</td>
<td>60%</td>
<td>0.334</td>
<td>0.301</td>
<td>(0.158,0.566)</td>
<td>0.0309</td>
<td>0.137</td>
<td>0.1297</td>
<td>13 min</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SS</th>
<th>censored</th>
<th>Mean</th>
<th>Mode</th>
<th>95% HPD interval</th>
<th>MSE</th>
<th>Bias</th>
<th>Accept. Rate</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>4%</td>
<td>0.225</td>
<td>0.211</td>
<td>(0.171,0.291)</td>
<td>0.0017</td>
<td>0.032</td>
<td>0.0357</td>
<td>5 hr</td>
</tr>
<tr>
<td>100</td>
<td>20%</td>
<td>0.165</td>
<td>0.162</td>
<td>(0.128,0.221)</td>
<td>0.0018</td>
<td>0.037</td>
<td>0.0317</td>
<td>5 hr</td>
</tr>
<tr>
<td>100</td>
<td>40%</td>
<td>0.257</td>
<td>0.223</td>
<td>(0.180,0.330)</td>
<td>0.0051</td>
<td>0.058</td>
<td>0.0560</td>
<td>2.3 hr</td>
</tr>
<tr>
<td>100</td>
<td>60%</td>
<td>0.264</td>
<td>0.270</td>
<td>(0.166,0.385)</td>
<td>0.0080</td>
<td>0.068</td>
<td>0.0625</td>
<td>2 hr</td>
</tr>
</tbody>
</table>

Table 6.8: Summary for exponential model using ABC via survivals algorithm.

test for the equality of two smoothed distribution functions. Simulated exact failure times rather than cen-
sored observations is sufficient to use. Two versions of this algorithm are proposed here, for discrete and
continuous data. Both algorithms are accurate and more time efficient than ABC rejection for censored data
described in the previous section 6.1.

### 6.4.1 Logspline Smoothing

Logspline smoothing or density estimation is suitable for any type of censored data assuming that the
observed random sample is from unknown continuous density function. This procedure had been studied
by Kooperberg and Stone in [25] and [26]. The logarithm of the unknown density function is estimated
by a spline. Then, the estimated density is the exponential function of a spline. In general, spline is a
piece-wise polynomial. Here, cubic splines with predetermined finite number of knots (location where
two neighboring polynomials must have at least the same value) and linear in the tails are used. As with
all splines, the most important is to determine which and how many knots to include. Logspline density
estimation procedure automatically finds the best combination of knots by applying stepwise knot deletion and AIC or BIC to select the ultimate model. The coefficients of the piece-wise polynomial are estimated by maximum likelihood techniques. Logspline density estimation is accurate and effective when the proportion of censored observations is high.

Let us consider cubic splines. According to [8] and [21], a cubic spline is a twice continuously differentiable, piecewise cubic polynomial. Suppose, given the integer $K \geq 3$, the numbers $L$ and $U$ such that $-\infty \leq L < U \leq \infty$, and the sequence $t_1, \ldots, t_K$ such that $L < t_1 < \cdots < t_K < U$. To find a cubic spline means to determine four coefficients for cubic polynomial on every of $K + 1$ subintervals $(t_{i-1}, t_i)$, where $i = 1, \ldots, K + 1$, $t_0 = L$ and $t_{K+1} = U$. Meaning that $4K + 4$ parameters need to be estimated. According to the definition of the cubic spline, every cubic polynomial must be twice continuously differentiable with its neighboring polynomials. Which makes three equations at each knot. Therefore, a space of cubic splines with $K$ knots is $K + 4$-dimensional. Logspline smoothing procedure uses natural cubic splines, which are cubic splines with linear to the left of the first knot and to the right of the last knot. To find a natural cubic spline with $K$ knots means to determine four less coefficients than for the cubic spline, which is $4K$. Three equations at each knot are still available. Thus, the dimension of natural cubic splines is $K$. Let $p = K - 1$, then the basis for natural cubic spline space will be $1, B_1, \ldots, B_p$. For example, the basis can be as follows,

$$B_1(x) = x,$$

$$B_{i+1}(x) = d_i(x) - d_{K-1}(x), \ i = 1, \ldots, K - 2$$

where $d_i(x) = \frac{(x - t_i)^3_+ - (x - t_K)^3_+}{t_K - t_i}$.

Consider $\theta = (\theta_1, \ldots, \theta_p)^T \in \Theta \subset \mathbb{R}^p$ such that

$$\int_L^U \exp(\theta_1 B_1(x) + \cdots + \theta_p B_p(x)) \, dx < \infty.$$

Then, a positive density function on $(L, U)$ for $\theta \in \Theta$ is

$$f(x, \theta) = \exp(\theta_1 B_1(x) + \cdots + \theta_p B_p(x) - C(\theta)), \ L < x < U,$$
where $C(\theta)$ is necessary constant for $f(x, \theta)$ to be a proper density function,

$$C(\theta) = \log \left( \int_L^U \exp \left( \theta_1 B_1(x) + \cdots + \theta_p B_p(x) \right) dx \right).$$

The corresponding distribution function is given by

$$F(x, \theta) = \int_L^x f(y, \theta) dy, \quad L < x < U.$$ 

The corresponding survival function is given by

$$S(x, \theta) = \int_x^U f(y, \theta) dy, \quad L < x < U.$$ 

Let $X_1, \ldots, X_n$ be a random sample of size $n$ from a distribution on $(L, U)$ having density function $f$, distribution function $F$ and survival function $S$. If $X_i$ is uncensored, then we observe the survival time. If $X_i$ is right censored, then we observe censoring time $T_i$ which is less than $X_i$. In this case we should consider the interval of a positive length $(T_i, U)$. Assume that the random sample independent of the censoring procedure. Then, for $\theta \in \Theta$ the likelihood function is

$$L(\theta) = \prod_{i=1}^n \left\{ \begin{array}{ll} f(x_i, \theta) = \exp \left[ \sum_{j=1}^p \theta_j B_j(x_i) - C(\theta) \right], & \text{if } X_i \text{ is uncensored,} \\
\int_{T_i}^U f(y, \theta) dy = \int_{T_i}^U \exp \left[ \sum_{j=1}^p \theta_j B_j(y) - C(\theta) \right] dy, & \text{if } X_i \text{ is right censored.} \end{array} \right.$$ 

The parameter $\theta = (\theta_1, \ldots, \theta_p)^T$ is estimated by maximum likelihood method,

$$\hat{\theta} = \arg \max_{\theta \in \Theta} L(\theta).$$ 

The corresponding maximum likelihood estimates of $f$, $F$, and $S$ for $L < x < U$ are

$$\hat{f}(x) = f(x, \hat{\theta}), \quad \hat{F}(x) = F(x, \hat{\theta}), \quad \hat{S}(x) = S(x, \hat{\theta}).$$
In order to get a good fit for a logspline, it is important to choose the right number of knots and the correct placement for them. Kooperberg and Stone [26] are offering two ways to determine it, with and without knot deletion procedure. Suppose we have right censored random sample \(X_1, \ldots, X_n\), \(n_u\) is the number of uncensored observed values and \(n_r\) is the number of right censored observed values. Then, the total number of observations is \(n = n_u + n_r\). Let \(n_c = n_u + 0.5n_r\) and \(N\) be the distinct number of observations. It is recommended to start with \(K\) knots,

\[
K = \min(4n_c^{0.2}, n_c/4, N, 25),
\]

when the knot deletion procedure is applied and

\[
K = \min(2.5n_c^{0.2}, n_c/4, N, 25),
\]

if the knot deletion procedure is not used. If \(K\) is not an integer, then it should be rounded up to the next integer. Special knot placement technique is proposed by Kooperberg and Stone [26] with the fixed number of knots used to get the logspline. In general, it is preferred to implement stepwise knot deletion method. The process starts with \(K\) knots, then it removes the least statistically significant knots except the two permanent knots based on absolute values of its Wald statistic. Now we have several models and the best one will be with corresponding minimal Akaike Information Criterion (AIC). It is recommended to use \(\log(n)\) as penalty parameter. The R package “logspline” is available for density, distribution, quantiles estimation via logsplines.

### 6.4.2 Cramer Test

Essential part of our new ABC method for censored data is implementing instead of distance metric a hypothesis test to determine if the observed and simulated samples are close enough. First, for both sets of observed and simulated data the density will be estimated by logspline. Then, two new samples will be generated from two estimated density functions. Next, the test will be applied to determine if these two samples came from the same continuous distribution function or not. For this reason, the most popular
goodness-of-fit tests are the Kolmogorov-Smirnov test, or Cramer-von Mises test. We will use Cramer test which is introduced in [3] by Baringhaus and Franz. Suppose, there are two random samples of \(d\)-vectors \(X_1, \ldots, X_m\) and \(Y_1, \ldots, Y_n\). The random vectors of the first sample are identically distributed with the distribution function \(F\) and the random vectors of the second sample are identically distributed with the distribution function \(G\). Assume, that \(E||X_i|| = \int ||x|| \, dF(x) < \infty\) and \(E||Y_i|| = \int ||x|| \, dG(x) < \infty\). The null and alternative hypotheses are

\[
H_0 : F = G \quad H_a : F \neq G.
\]

The test statistic is

\[
T_{m,n} = \frac{mn}{m+n} \left[ \frac{1}{mn} \sum_{j=1}^{m} \sum_{k=1}^{n} ||X_j - Y_k|| - \frac{1}{m^2} \sum_{j=1}^{m} \sum_{k=1}^{m} ||X_j - X_k|| - \frac{1}{n^2} \sum_{j=1}^{n} \sum_{k=1}^{n} ||Y_j - Y_k|| \right].
\]

For one-dimensional case the test statistic is

\[
T_{m,n} = \frac{mn}{m+n} \int_{-\infty}^{\infty} \left[ F_m(t) - G_n(t) \right]^2 dt,
\]

where \(F_m\) and \(G_n\) are the empirical distribution functions of two samples. This test statistic is similar to the Cramer-von Mises statistic. In general, the statistic is not distribution-free. The bootstrap method is proposed for getting critical values.

Cramer test produces satisfactory power when compared to its parametric competitors. In simulations in [3], it is shown that Cramer test outperforms its non-parametric competitors Cramer-von Mises and Kolmogorov-Smirnov tests.

### 6.4.3 Algorithm

New ABC algorithm for continuous censored data via comparing smoothed distribution curves goes as follows. First, the parameter candidate is generated from the prior distribution. Second, the sample is
simulated from the model given the parameter candidate. This sample doesn’t need to be censored. Then, for both observed and simulated data the log spline procedure is used to estimate the distribution functions. Next, two sample are generated from the estimated distribution functions. After that, the Cramer hypothesis test is performed for comparing distribution functions of obtained samples. The parameter candidate is accepted if the null hypothesis that two samples came from the same distribution function cannot be rejected. Below the algorithm is described for continuous data.

<table>
<thead>
<tr>
<th>ABC via comparing smoothed distributions: continuous case</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Draw $\theta^*$ from the prior distribution $\pi(\theta)$</td>
</tr>
<tr>
<td>2. Simulate $X^* \sim f(X</td>
</tr>
<tr>
<td>3. Obtain distribution estimators $\hat{F}$ and $\hat{G}$ for observed set $X$ and $X^*$</td>
</tr>
<tr>
<td>4. Draw samples from $\hat{F}$ and $\hat{G}$</td>
</tr>
<tr>
<td>5. Perform Cramer hypothesis test</td>
</tr>
<tr>
<td>6. Accept $\theta^*$ if $p-value &gt; \alpha$</td>
</tr>
</tbody>
</table>

Significance level $\alpha$ is recommended to choose higher than usual 0.05 in order to decrease type II error, do not reject null hypothesis when alternative is true. We want possibly to avoid the mistakes when the samples come from the different distribution but think from the same and accept the wrong parameter candidate. The parameter estimate of ABC via comparing smoothed distributions is the mean of all accepted parameter candidates.

To be able to use ABC via comparing distribution functions for the discrete case, we must make an adjustment to the data in the same fashion as in section 6.3 via two lemmas.

New ABC algorithm for discrete censored data via comparing distribution functions goes as follows. The observed sample values need to be modified by adding some continuous random variable first. Second, the parameter candidate is generated from the prior distribution. Then, the sample is simulated from the model given the parameter candidate. This sample doesn’t need to be censored. After, the sample values should be modified in the similar fashion as the observed sample. For both modified observed and simulated
data the logspline procedure is used to estimate the distribution functions. Next, two sample are generated from the estimated distribution functions. After that, the Cramer hypothesis test is performed for comparing distribution functions of obtained samples. The parameter candidate is accepted if the null hypothesis that two samples came from the same distribution function cannot be rejected.

The notations are as follows. The censored data is \( X = \{(x_i, \delta_i) : i = 1, \ldots, n\} \), where \( n \) is the total number of observations; \( x_i = \min(y_i, z_i) \), where \( y_i \) is observed failure time and \( z_i \) is the censored failure time. Assume, that \( y_i \) and \( z_i \) are independent. If \( x_i = y_i \), then \( \delta_i = 1 \). If \( x_i = z_i \), then \( \delta_i = 0 \). Let \( X_{\text{mod}} = X + U \), where \( X \) is discrete and \( U \) is a continuous random variable that is independent of \( Y \) and \( Z \). Applying Lemmas 1 and 2 from section 6.3, \( X_{\text{mod}} \) can be treated as a continuous random variable and we use it as our observed data set. We choose a \( U \) such that the adjustment to \( X \) is small. We make the same type of adjustment to the simulated data \( X^\ast \) and obtain \( X_{\text{mod}}^\ast = X^\ast + U \). Below the algorithm is described for discrete data.

**ABC via comparing smoothed distributions: discrete case**

1. Compute \( X_{\text{mod}} = X + U \)
2. Draw \( \theta^\ast \) from the prior \( \pi(\theta) \)
3. Simulate \( X^\ast \sim f(\cdot | \theta^\ast) \), then compute \( X_{\text{mod}}^\ast = X^\ast + U \)
4. Obtain distribution estimators \( \hat{F} \) and \( \hat{G} \) for \( X_{\text{mod}} \) and \( X_{\text{mod}}^\ast \)
5. Draw samples from \( \hat{F} \) and \( \hat{G} \)
6. Perform Cramer hypothesis test
7. Accept \( \theta^\ast \) if \( p - \text{value} > \alpha \)

Significance level \( \alpha \) is recommended to choose higher than usual 0.05 in order to decrease type II error, do not reject null hypothesis when alternative is true. We want possibly to avoid the mistakes when the samples come from the different distribution but think from the same and accept the wrong parameter candidate. The parameter estimate of ABC via comparing smoothed distributions is the mean of all accepted parameter candidates.
6.4.4 Asymptotically Unbiased Estimator

Now we will show that the estimator obtained by ABC via comparing smoothed distributions method is asymptotically unbiased.

Suppose we are testing \(H_0\) against \(H_a\) and we have come up with a test statistic where for large \(n\), the power goes to one. Given that \(H_0\) is true, it is known that \(\hat{\theta}\) is an asymptotically unbiased estimator of \(\theta\). It is shown in 6.3.3 that \(\hat{\theta}\) is still an asymptotically unbiased estimator if given that \(H_0\) is accepted, rather than known to be true.

Let \(F(x)\) and \(\hat{G}(x)\) represent the estimated distribution functions for the observed and simulated data, respectively, where simulated sample is generated from the model given \(\theta^*\). The Cramer test is used in our ABC algorithm as the acceptance criterion. It is known that the power of the test approaches one as sample sizes \(n_1\) and \(n_2\) approach infinity. (Without loss of generality assume \(n_1 = n_2 = n\).) The null hypothesis is:

\[
H_0 : F(x) = G(x).
\]

Let \(N = 1\). Then the output of our ABC algorithm would be \(\theta^*_1\). This candidate value of \(\theta\) was only accepted, because the null hypothesis of the Cramer test was not rejected. By the result in 6.3.3, we know that \(\theta^*_1\) is an asymptotically unbiased estimator for \(\theta\).

Let \(N > 1\). Let \(A_i\) represent the event that \(\theta^*_i\) was accepted and therefore \(H_0\) was accepted for the corresponding Cramer test. Then, the output of the ABC algorithm would be the sequence \(\{\theta^*_i|A_i\}_{i=1}^N\). The ABC estimator for \(\theta\) will be the mean of the values in a sequence:

\[
\overline{\theta}^* = \frac{1}{N} \sum_{i=1}^{N} \theta^*_i|A_i.
\]

Since all values in a sequence are asymptotically unbiased estimators of \(\theta\), the ABC estimator also will be asymptotically unbiased estimator of \(\theta\).
6.4.5 Consistency

Now we will show that the estimator obtained by ABC via comparing smoothed distribution functions method is consistent.

Suppose we are testing $H_0$ against $H_a$ and we have come up with a test statistic where for large $n$, the power goes to one. Given that $H_0$ is true, it is known that $\hat{\theta}$ is a consistent and asymptotically unbiased estimator of $\theta$. It has been shown in 6.3.4. that $\hat{\theta}$ is still a consistent estimator if given that $H_0$ is accepted, rather than known to be true.

The output of our ABC algorithm would be a sequence of $\theta^*$ s. Each $\theta^*$, the candidate value of $\theta$, was only accepted because the null hypothesis of the Cramer test was not rejected. By the above result, we know that each $\theta^*$ is a consistent estimator for $\theta$.

Let $A_i$ represent the event that $\theta^*_i$ was accepted and therefore $H_0$ was accepted for the corresponding Cramer test. Then, the output of the ABC algorithm would be the sequence $\{\theta^*_i|A_i\}_{i=1}^N$. The ABC estimator for $\theta$ will be the mean of the values in a sequence:

$$\tilde{\theta}^* = \frac{1}{N} \sum_{i=1}^N \theta^*_i |A_i.$$ 

Since all values in a sequence are consistent estimators of $\theta$, the ABC estimator also will be a consistent estimator of $\theta$.

6.4.6 Asymptotic Behavior of Approximate Posterior Distribution

Here we will show what is the algorithm’s resulting distribution of the parameter. Let $\{\theta^*_i\}_{i=1}^N$ be the resulting sequence of the algorithm. Each $\theta^*_i$ is independent draw from $f(\theta|H_0$ is accepted). Then,

$$f(\theta^*_i) \propto \sum_{x^* \in \mathcal{F}} f(x^*|\theta^*_i) \pi(\theta^*_i) \mathbb{1}_{H_0 \, \text{accepted}} \propto \sum_{x^*: H_0 \, \text{accepted}} f(x^*|\theta^*_i) \pi(\theta^*_i) \propto \pi_{H_0}(\theta^*_i|x),$$
where $\mathcal{F}$ is $\sigma$-algebra on some given set $\Omega$, $x^*$ is simulated data, $f(x^*|\cdot)$ is the model for simulated data, and $\pi(\cdot)$ is prior distribution. The resulting approximate posterior distribution $\pi_{H_0}(\theta^*_i|x)$ depends on the test which helps to make a decision to reject the null hypothesis or not.

As sample size of the observed data set approaches infinity, the sample size of the simulated data set approaches infinity. Then the power of the Cramer test approaches one,

$$1 - \beta = Pr(H_1 \text{ is accepted } | H_0 \text{ is false}) \to 1.$$  

Meaning that, when, in reality, null hypothesis is false, the alternative hypothesis will be accepted and the corresponding parameter value will not be taken into the resulting sequence of parameter values. Thus, the wrong parameter values will not be taken into the resulting sequence of parameter values at all. Hence, the resulting posterior distribution will be a true posterior,

$$\pi_{H_0}(\theta^*_i|x) \to \pi(\theta^*_i|x),$$

as sample sizes of observed and simulated data sets approach infinity.

### 6.4.7 Examples

In this section ABC via comparing smoothed distribution functions (ABC via smoothing) is illustrated on the following examples.

**Example 6.1 (continued).** Consider a normal model. This time the samples of different sizes such as 25, 50, and 100 are considered. Parameter candidates are randomly generated from the uniform prior distribution $U(5,20)$. New data set is simulated from the normal model based on the chosen parameter candidate. Observed and simulated sample’s distribution functions are compared by Cramer test. The tolerance level is 0.7 except for 60% censored observations for the sample size of 50 and 100 where the tolerance level is 0.3 since it takes too long for the algorithm to run. Posterior sample size is 100. The mean estimate is a
mean of accepted parameter candidates. Mean estimate, mode of density of accepted parameter candidates, 95% highest posterior density (HPD) interval, mean squared error (MSE), bias, acceptance rate, and time obtained by ABC via smoothing algorithm are presented in the table 6.9. MSE and bias are of accepted parameter candidates. Acceptance rate is the proportion of accepted parameter candidates over all parameter candidates generated from the prior distribution.

<table>
<thead>
<tr>
<th>Normal Model</th>
<th>ABC via smoothing</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS</td>
<td>censored</td>
</tr>
<tr>
<td>25</td>
<td>4%</td>
</tr>
<tr>
<td>25</td>
<td>20%</td>
</tr>
<tr>
<td>25</td>
<td>40%</td>
</tr>
<tr>
<td>25</td>
<td>60%</td>
</tr>
</tbody>
</table>

| SS | censored | Mean | Mode | 95% HPD interval | MSE | Bias | Accept. Rate | Time |
| 50 | 4% | 9.518 | 9.239 | (8.091,10.770) | 0.7182 | 0.695 | 0.0206 | 13 min |
| 50 | 20% | 9.813 | 9.648 | (8.451,11.615) | 0.7426 | 0.707 | 0.0210 | 12 min |
| 50 | 40% | 10.739 | 10.631 | (9.245,12.550) | 1.2189 | 0.907 | 0.0130 | 19 min |
| 50 | 60% | 12.328 | 12.300 | (10.019,14.252) | 6.7260 | 2.328 | 0.0016 | 7 hr |

| SS | censored | Mean | Mode | 95% HPD interval | MSE | Bias | Accept. Rate | Time |
| 100 | 4% | 9.707 | 9.706 | (8.702,10.735) | 0.4235 | 0.517 | 0.0143 | 1 hr |
| 100 | 20% | 10.374 | 10.329 | (9.440,11.633) | 0.4249 | 0.528 | 0.0174 | 1 hr |
| 100 | 40% | 10.918 | 10.712 | (9.705,11.679) | 1.1512 | 0.940 | 0.0052 | 4 hr |
| 100 | 60% | 11.097 | 11.147 | (9.710,12.482) | 1.6799 | 1.112 | 0.0002 | 78 hr |

Table 6.9: Summary for normal model using ABC via smoothing algorithm.

**Example 6.2** (continued). Consider an exponential model. This time the samples of different sizes such as 25, 50, and 100 are considered. Parameter candidates are randomly generated from the uniform prior distribution $U(0,1)$. New data set is simulated from the exponential model based on the chosen parameter candidate. Observed and simulated sample’s distribution functions are compared by Cramer test. The tolerance level is 0.7. Posterior sample size is 100. The mean estimate is a mean of accepted parameter candidates. Mean estimate, mode of density of accepted parameter candidates, 95% highest posterior density (HPD) interval, mean squared error (MSE), bias, acceptance rate, and time obtained by ABC via smoothing algorithm are presented in the table 6.10. MSE and bias are of accepted parameter candidates. Acceptance
rate is the proportion of accepted parameter candidates over all parameter candidates generated from the prior distribution.

<table>
<thead>
<tr>
<th>SS</th>
<th>censored</th>
<th>Mean</th>
<th>Mode</th>
<th>95% HPD interval</th>
<th>MSE</th>
<th>Bias</th>
<th>Accept. Rate</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>4%</td>
<td>0.284</td>
<td>0.245</td>
<td>(0.124,0.542)</td>
<td>0.0186</td>
<td>0.101</td>
<td>0.0261</td>
<td>6 min</td>
</tr>
<tr>
<td>25</td>
<td>20%</td>
<td>0.314</td>
<td>0.274</td>
<td>(0.120,0.511)</td>
<td>0.0245</td>
<td>0.121</td>
<td>0.0329</td>
<td>5 min</td>
</tr>
<tr>
<td>25</td>
<td>40%</td>
<td>0.321</td>
<td>0.324</td>
<td>(0.113,0.513)</td>
<td>0.0277</td>
<td>0.132</td>
<td>0.0109</td>
<td>15 min</td>
</tr>
<tr>
<td>25</td>
<td>60%</td>
<td>0.424</td>
<td>0.440</td>
<td>(0.195,0.777)</td>
<td>0.0754</td>
<td>0.226</td>
<td>0.0359</td>
<td>7 min</td>
</tr>
</tbody>
</table>

Table 6.10: Summary for exponential model using ABC via smoothing algorithm.

Results in tables 6.9 and 6.10 show that when sample size increases, MSE and bias get smaller for a fixed percentage of censored observations as expected. If proportion of censored observations rises, MSE and bias become larger for fixed sample size, since it is harder to make accurate estimation.

6.5 Compare ABC Algorithms for Censored Data. Examples

In this section three new ABC algorithms for right censored data: ABC-EL for censored data, ABC via comparing survivals, and ABC via comparing smoothed distributions are proposed.

ABC-EL for censored data is developed by implementing empirical likelihood weights calculations. It does not require the choice of summary statistic or distance metric, and there is no need to set up a tolerance level. This algorithm works very fast and parameter estimation can be improved. But it can be used only when the mean constraint is known and the largest observed value is not censored.
ABC via comparing survivals and ABC via comparing smoothed distributions are both methods proposed by introducing more advanced metric for accepting parameter candidates, a testing procedure. As a result, for high proportion of censored observation they both produce more accurate parameter estimates than ABC-REJ for censored data. For low proportion of censored observations the estimates obtained by all three methods are comparable. As sample size gets larger it takes more time for ABC-REJ for censored data to achieve accurate parameter estimates even for low percentages of censored values in the observed sample.

ABC algorithms suitable for right censored data and described in this chapter are compared on the following examples.

**Example 6.1** (continued). Normal model example. ABC rejection for censored data, ABC via survivals, ABC via smoothing are used to estimate a true mean, assuming that standard deviation is known. Mean squared error (MSE), bias, and acceptance rate (Acc.R.) for each case are presented in the table 6.11. MSE and bias are of accepted parameter candidates. Acceptance rate is the proportion of accepted parameter candidates over all parameter candidates generated from the prior distribution. ABC rejection for censored data before was used for sample size of 25 with tolerance of 200. Here, ABC rejection for censored data is implemented for sample size of 50 with tolerance of 600, and for sample size of 100 with much higher tolerance of 1500 for 4% and 60% of censored values, 1900 for 20% and 1700 for 40% of censored observations. Lower tolerance levels for sample size of 50 or 100 improve the accuracy of estimation a little, but it makes the acceptance rate incredibly low and it takes more than 24 hours for the algorithm to run one time on a standard personal computer. The increase of tolerance level for ABC via survivals and ABC via smoothing produces a better estimates as well. In this case acceptance rate decreases and it takes more time for the algorithm to run, but the increase in time is reasonable.

**Example 6.2** (continued). Exponential model example. ABC rejection for censored data, ABC via survivals, ABC via smoothing are used to estimate a true mean. Mean squared error (MSE), bias, and acceptance rate (Acc.R.) for each case are presented in the table 6.12. MSE and bias are of accepted parameter candidates. Acceptance rate is the proportion of accepted parameter candidates over all parameter candidates generated from the prior distribution. ABC rejection for censored data before was used for sample size of 25 with tolerance of 0.5. Here, ABC rejection for censored data is implemented for sample size of 50 with tolerance
Table 6.11: Summary for normal model using ABC-REJ for censored data, ABC via survivals, and ABC via smoothing algorithms.

of 3, and for sample size of 100 with much higher tolerance of 3 for 4% of censored values, 15 for 20%, 40 for 40%, and 25 for 60% of censored data. Lower tolerance levels for sample size of 50 or 100 improves the accuracy of estimation, but it makes the acceptance rate incredibly low. Increase of tolerance level for ABC via survivals and ABC via smoothing gives a better estimates.

In tables 6.11 and 6.12, MSEs and biases rapidly increase for ABC-REJ censored as the proportion of censored observations grows. MSEs and biases expand slow for ABC via survivals and ABC via smoothing as the proportion of censored observations grows. It can be concluded that for small sample sizes and small percentage of censored observation ABC rejection works the best since time, MSE, and bias are the smallest. However, for small samples and very high proportions of censored data it is dangerous to use ABC rejection due to the nature of the algorithm, which is based on comparing only uncensored parts of observed and simulated sets. For large sample sizes and high proportions of censored observations it is better to use ABC via comparing survivals or ABC via comparing smoothed distribution functions. For large sample sizes and low proportions of censored data, all methods perform similar. The graphs 6.1 and 6.2 illustrate the tables 6.11 and 6.12 for the sample size of 100.
Table 6.12: Summary for exponential model using ABC-REJ for censored data, ABC via survivals, and ABC via smoothing algorithms.

<table>
<thead>
<tr>
<th>Method</th>
<th>ABC-REJ censored</th>
<th>ABC via survivals</th>
<th>ABC via smoothing</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MSE</td>
<td>Bias</td>
<td>Acc. R.</td>
</tr>
<tr>
<td>censored</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sample size 25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4%</td>
<td>0.0047</td>
<td>0.048</td>
<td>1.7 \cdot 10^{-5}</td>
</tr>
<tr>
<td>20%</td>
<td>0.0149</td>
<td>0.098</td>
<td>2.3 \cdot 10^{-6}</td>
</tr>
<tr>
<td>40%</td>
<td>0.0163</td>
<td>0.096</td>
<td>9.7 \cdot 10^{-5}</td>
</tr>
<tr>
<td>60%</td>
<td>0.0972</td>
<td>0.269</td>
<td>8 \cdot 10^{-5}</td>
</tr>
<tr>
<td>sample size 50</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4%</td>
<td>0.0019</td>
<td>0.033</td>
<td>4.5 \cdot 10^{-6}</td>
</tr>
<tr>
<td>20%</td>
<td>0.0363</td>
<td>0.177</td>
<td>2.5 \cdot 10^{-7}</td>
</tr>
<tr>
<td>40%</td>
<td>0.0860</td>
<td>0.275</td>
<td>4 \cdot 10^{-8}</td>
</tr>
<tr>
<td>60%</td>
<td>0.1630</td>
<td>0.381</td>
<td>2.3 \cdot 10^{-6}</td>
</tr>
<tr>
<td>sample size 100</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4%</td>
<td>0.0012</td>
<td>0.028</td>
<td>3.3 \cdot 10^{-7}</td>
</tr>
<tr>
<td>20%</td>
<td>0.0495</td>
<td>0.217</td>
<td>6.1 \cdot 10^{-8}</td>
</tr>
<tr>
<td>40%</td>
<td>0.3481</td>
<td>0.582</td>
<td>6.7 \cdot 10^{-8}</td>
</tr>
<tr>
<td>60%</td>
<td>0.3643</td>
<td>0.596</td>
<td>3.2 \cdot 10^{-7}</td>
</tr>
</tbody>
</table>

Figure 6.1: Comparison of algorithm’s posterior MSEs for the normal model.

**Example 6.5.** Consider a normal model. A sample of size of 50 with 16% of censored observations is drawn from normal distribution with mean of 10 and standard deviation of 3. Suppose, the likelihood function is intractable and the true mean needs to be estimated.
ABC-REJ censored, ABC via survivals, and ABC via smoothing are implemented to estimate the mean. Posterior sample size is 100. The tolerance for each algorithm was partially chosen based on run-times. No single run was allowed to take more than 1 hour to complete. Prior distribution is $U(5, 20)$. Each algorithm was run 25 times. The minimum and maximum of estimates of the true mean, mean squared error (MSE), and bias obtained by each method are presented in the table 6.13.

<table>
<thead>
<tr>
<th>Method</th>
<th>Tolerance</th>
<th>Minimum</th>
<th>Maximum</th>
<th>MSE</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC-REJ censored</td>
<td>2.65</td>
<td>11.42</td>
<td>11.70</td>
<td>2.448 202</td>
<td>1.563 14</td>
</tr>
<tr>
<td>ABC via survivals</td>
<td>0.2 and 0.6</td>
<td>9.946 10.215</td>
<td>0.006 610</td>
<td>0.054 49</td>
<td></td>
</tr>
<tr>
<td>ABC via smoothing</td>
<td>0.70</td>
<td>10.09</td>
<td>10.35</td>
<td>0.062 968</td>
<td>0.238 72</td>
</tr>
</tbody>
</table>

Table 6.13: Normal model example. Comparison of ABC algorithms for right censored data.

The MSE and bias of 25 runs of ABC via comparing survival functions procedure is the lowest as shown in the table 6.13. Thus, for this case ABC via survivals performs better than other methods.

Example 6.6. Consider an exponential model. A sample of size of 25 with 24% of censored observations is drawn from exponential distribution with mean of 15. Suppose, the likelihood function is intractable and the true rate = 1/mean needs to be estimated.

ABC-REJ censored, ABC via survivals, and ABC via smoothing are implemented to estimate the rate. Posterior sample size is 100. The tolerance for each algorithm was partially chosen based on run-times. No single run was allowed to take more than 1 hour to complete. Prior distribution is $U(0, 1)$. Each algorithm
was run 25 times. The minimum and maximum of estimates of the true rate, mean squared error (MSE), and bias obtained by each method are presented in the table 6.14.

<table>
<thead>
<tr>
<th>Method</th>
<th>Tolerance</th>
<th>Minimum</th>
<th>Maximum</th>
<th>MSE</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC-REJ censored</td>
<td>6.5</td>
<td>0.0394</td>
<td>0.0442</td>
<td>0.000594</td>
<td>0.024339</td>
</tr>
<tr>
<td>ABC via survivals</td>
<td>0.2 and 0.6</td>
<td>0.0695</td>
<td>0.0776</td>
<td>0.00061</td>
<td>0.007411</td>
</tr>
<tr>
<td>ABC via smoothing</td>
<td>0.50</td>
<td>0.0679</td>
<td>0.0816</td>
<td>0.00069</td>
<td>0.007497</td>
</tr>
</tbody>
</table>

Table 6.14: Exponential model example. Comparison of ABC algorithms for right censored data.

The MSE and bias of 25 runs of ABC via comparing survival functions procedure is the lowest as shown in the table 6.14. Thus, for this case ABC via survivals performs better than other methods.

**Example 6.6** (continued). ABC via survivals is applied to estimate the rate. Different tolerance levels were used. Algorithm was run 25 times. The time it take the algorithm to run, minimum and maximum of estimates of the true rate, mean squared error (MSE), and bias obtained are presented in the table 6.15.

<table>
<thead>
<tr>
<th>Time</th>
<th>Tolerance</th>
<th>Minimum</th>
<th>Maximum</th>
<th>MSE</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>12 min</td>
<td>0.2 and 0.6</td>
<td>0.0695</td>
<td>0.0776</td>
<td>0.00061</td>
<td>0.007411</td>
</tr>
<tr>
<td>30 min</td>
<td>0.4 and 0.8</td>
<td>0.06617</td>
<td>0.07438</td>
<td>0.00021</td>
<td>0.004172</td>
</tr>
<tr>
<td>60 min</td>
<td>0.5 and 0.9</td>
<td>0.06576</td>
<td>0.07260</td>
<td>0.00009</td>
<td>0.002422</td>
</tr>
</tbody>
</table>

Table 6.15: Exponential model example. Comparison of tolerance levels for ABC via survivals.

The MSE and bias of 25 runs of ABC via comparing survival functions procedure get smaller as the tolerance increases, see table 6.15.

**Example 6.6** (continued). ABC via smoothing is applied to estimate the rate. Different tolerance levels were used. Algorithm was run 25 times. The time it take the algorithm to run, minimum and maximum of estimates of the true rate, mean squared error (MSE), and bias obtained are presented in the table 6.16.

<table>
<thead>
<tr>
<th>Time</th>
<th>Tolerance</th>
<th>Minimum</th>
<th>Maximum</th>
<th>MSE</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 min</td>
<td>0.50</td>
<td>0.0679</td>
<td>0.0816</td>
<td>0.00069</td>
<td>0.007497</td>
</tr>
<tr>
<td>15 min</td>
<td>0.70</td>
<td>0.0656</td>
<td>0.0786</td>
<td>0.00046</td>
<td>0.006015</td>
</tr>
<tr>
<td>70 min</td>
<td>0.90</td>
<td>0.0644</td>
<td>0.0732</td>
<td>0.00009</td>
<td>0.001852</td>
</tr>
</tbody>
</table>

Table 6.16: Exponential model example. Comparison of tolerance levels for ABC via smoothing.

The MSE and bias of 25 runs of ABC via comparing smoothed distribution functions procedure get smaller as the tolerance increases, see table 6.16.
CHAPTER 7
REAL DATA EXAMPLE

Interesting paper about applying ABC rejection method for estimating number concentrations of nanoparticles in suspension recently have been published in 2016 in Physical Review [45]. A particle-tracking experiment in a liquid suspension $\Omega = [-A/2, A/2]^3 \subset \mathbb{R}^3$ is considered in the paper. The assumptions are that there is a diffusion equilibrium, the number of particles in the suspension is fixed and uniformly distributed at any time. A fluorescence microscope is used to keep track of particles in two dimensions inside the detection region $w = [-a_x/2, a_x/2] \times [-a_y/2, a_y/2] \times [-a_z/2, a_z/2] \subset \Omega$. $a_x$ and $a_y$ values of the detection region are possible to determine prior to the experiment. But the $a_z$ is not known, since it depends on many different factors of the experiment and image analysis settings. The shape of the liquid suspension is not known, a shape of a cube is chosen with the side of $A = 100 \, \mu m$ for simplicity. However, it is known that the detection region is far from the boundaries of a liquid suspension $|\Omega| >> |w|$. The experimental setup illustration as in [45] can be seen on the figure 7.1.

![Figure 7.1: The experimental setup.](image)
Particle detection and tracking is performed through fluorescence single particle tracking technique. This method allows to record the flow of each fluorescently labeled nanoparticle, see [9] for detailed description of the technique. The example of particle trajectories on a still video frame as in [45] is shown on Figure 7.2.

![Figure 7.2: The example of particle trajectories.](image)

Assume that the particles move randomly in and out of the detection region. If the particle is noticed to be in the detection region for two or less consecutive video frames, such particle will not be taken into account, since it could be just the noise in the images. If some particle was in the detection region, moved out, and came back in, then it is considered to be two different particles. Similar particle behavior is repeated in the simulation process.

The authors graciously agreed to give us the data, which consist of two-dimensional particle coordinates with displacement and some labels. When coordinates are zeros in the data, it means that the particle is outside of the detection region. An example of the data provided by authors of [45] is shown on the Figure 7.3.
In [45] ABC rejection method was implemented to estimate the concentration of the particles in a liquid suspension associated with nanomedicine. There are liposomes, which serve as a nanocarrier of the medicine to the desired destination, some genes. The genes are in the form of plasmid DNA molecules. Liposome encloses the wanted DNA particles, and such structure will be called LPX. For proper dosage, it is very important to know how many DNA molecules on average can be enveloped by a single LPX. It is beneficial to create LPX with higher number of DNAs in order to decrease the toxicity of the organism. First, the concentration of DNA molecules $c_{DNA}$ in a suspension is estimated. Next, the DNA suspension is mixed together with a liposome suspension in equal proportions and LPXs are assembled. Then, the concentration of LPX particles $c_{LPX}$ is estimated. The average number of DNAs inside a single LPX will be the following ratio

$$Q = \frac{1}{2} \frac{c_{DNA}}{c_{LPX}}.$$

We would like to demonstrate new ABC via comparing survival functions and ABC via comparing smoothed distributions methods proposed in 6.3 and 6.4 to estimate the concentration of DNAs, LPXs, and then the ratio. Dr. Roding one of the authors of the paper [45] provided us with data of 25 videos of particle
performance. Each video is 5 seconds long and consist of 96 frames. From each video we will use the total number of particles recorded as a set of observed data. Simulated data set is obtained by a simulation program.

7.1 Simulation program

For the simulation program the following assumptions were used from [45]. First, suppose that in the beginning noninteracting particles uniformly distributed in a cube liquid suspension. Second, the particle perform a 3D Gaussian random walk. Let $\Delta x$, $\Delta y$, and $\Delta z$ be displacements along the three axes, which follow the same normal distribution with the mean of zero and variance of $2D\Delta t$, where $D$ is a diffusion coefficient and $\Delta t$ is time between video frames within a movie,

$$\Delta x, \Delta y, \Delta z \sim \mathcal{N}(0, 2D\Delta t).$$

Since in our case we have a movie of 5 seconds long and 96 frames in a movie, time between frames is approximately 0.053 of a second.

This simulation program depends on three parameters: number of particles $n$, diffusion coefficient $D$, and the height of the detection region $a_z$. There is a liquid suspension of a size $100 \times 100 \times 100 \mu m^3$ and detection region of a size $61 \times 61 \times a_z \mu m^3$ for estimating the number of LPXs. There is a liquid suspension of a size $80 \times 80 \times 80 \mu m^3$ and detection region of a size $48.8 \times 48.8 \times a_z \mu m^3$ for estimating the number of DNA particles. The lengths of the detection regions in both cases are proportional to the lengths of the liquid suspensions. The difference in sizes of liquid suspensions is due to the fact that the number of those particles were estimated in a different volumes of solutions. The volume of the solution with DNAs is twice less than the volume of the solution with LPXs according to [45]. Let us introduce a 3D coordinate system with the origin in center of the cube liquid suspension. Then, the dimensions of the suspension will be $[-50, 50]$ along each of the axes for estimating the number of LPXs and $[-40, 40]$ along each of the axes for estimating the number of DNAs. The dimensions of the detection region for estimating the number of LPXs will be $[-30.5, 30.5]$ along x-axis and y-axis, and $[-a_z/2, a_z/2]$ along the z-axis. The dimensions of
the detection region for estimating the number of DNAs will be $[-24.4, 24.4]$ along $x$-axis and $y$-axis, and $[-a_z/2, a_z/2]$ along the $z$-axis.

For each of $n$ particles 96 coordinates are generated and a Gaussian random walk is formed which starts at random within the liquid suspension. For $i$-th particle, $i = 1, \ldots, n$, let $l_i$ be the counts of times the consecutive coordinates of a Gaussian random walk all are in the detection region. For example, a particle has some coordinates outside of the detection region, then some inside, then some outside, and then inside again, in this case $l_i = 2$, two times consecutive coordinates of a Gaussian random walk of this particle were inside the detection region. Then the simulated total number of particles noticeable through the microscope in one movie is

$$s = \sum_{i=1}^{n} l_i.$$ 

In the same fashion the process is repeated 25 times.

### 7.2 Prior Distributions

There are three parameters we need the prior distributions for: number of particles $n$, diffusion coefficient $D$, and the height of the detection region $a_z$. These parameters are assumed to be mutually independent.

For the concentration $c$ an improper prior distribution is chosen $U(c_{\text{min}}, c_{\text{max}})$. Careful studying of observed data sets and simulations helped to determine $c_{\text{min}} = 2000$ and $c_{\text{max}} = 4500$ particles in a liquid suspension $\Omega$ for estimating number of LPX particles; $c_{\text{min}} = 7000$ and $c_{\text{max}} = 9500$ particles in a liquid suspension $\Omega$ for estimating number of DNA molecules. Then, the number of particles $n$ for the simulation program follow Poisson distribution with parameter $c$.

For the height of the detection region $a_z$ an improper prior distribution is elected $U(a_{z,\text{min}}, a_{z,\text{max}})$. In [45] it is noticed that the height of the detection region is many times smaller than its length or width, and is a matter of couple micrometers. Hence, $a_{z,\text{min}}$ is decided to be 0.4 $\mu$m and $a_{z,\text{max}}$ to be 2 $\mu$m.

For diffusion coefficient $D$ the prior used is inverse gamma which is derived in [45],

$$D \sim \Gamma^{-1}(\alpha = m - 1, \beta = m r^2 / 4 \Delta t),$$
where $\Delta t$ is time between frame, in our case $\Delta t = 0.053$, $r^2$ is the average of estimated mean-squared displacement for each of 25 videos, and $m$ is the average of the number of observed particle displacements for each of 25 videos.

$$m = \frac{1}{25} \sum_{j=1}^{25} m_j,$$

where $m_j$ is the total number of observed particle displacements for video $j$, see Figure 7.3.

$$r^2 = \frac{1}{25} \sum_{j=1}^{25} r_j^2,$$

where

$$r_j^2 = \frac{1}{m_j} \sum_{i=1}^{m_j} (\Delta x_j^2 + \Delta y_j^2),$$

where $\Delta x_j^2 + \Delta y_j^2$ is particle displacement, square root of which can be found in every 3-rd column of the data set, see Figure 7.3.

### 7.3 Concentration Estimation with ABC via Comparing Survival Functions

The data $X$ we chose to work with is the total number of particles noticed by microscope in each of 25 videos. Since it is a discrete data set, it needs to be adjusted

$$X_{\text{mod}} = X + U_1,$$

where $U_1$ is required to be continuous random variable. Here we chose $U_1$ to be exponentially distributed with the rate of 2. Three parameter candidate $n$, $a_z$, $D$ are drawn from the prior distribution independently. Then, $X^*$ is generated by a simulation program based on parameter candidates. Once again it is a discrete data set and needs to be modified

$$X^*_{\text{mod}} = X^* + U_2,$$

where $U_2 \sim \text{exp}(2)$. The next step is to run a two-stage hypothesis test. The two significance levels are accepted to be 0.2 for the first one, and 0.6 for the second one. They are larger than usual 0.05, since we
want to avoid cases of type II errors when two survival functions are different but we fail to acknowledge it. The posterior sample size is only 100 accepted values due to a limited computer power available.

The LPX concentration estimate is $2.359 \cdot 10^9$ particles per milliliter. The DNA concentration estimate is $9.04 \cdot 10^9$ particles per milliliter. The ratio of the number of DNAs per one LPX is 1.916.

### 7.4 Concentration Estimation with ABC via Comparing Smoothed Distribution Functions

Here, the same settings as in the previous section are used. Three parameter candidate $n$, $a_z$, $D$ are drawn from the prior distribution independently. Then, $X^*$ is generated by a simulation program based on the parameter candidates, and gets adjusted after. Then, for the observed and simulated data the distribution estimators are obtained by log-spline smoothing method and samples are drawn from both estimations. The next step is to run a Cramer hypothesis test. The significance level is chosen to be 0.6. It is larger than usual 0.05, since we want to avoid the case of type II error when two distribution functions are different but we fail to acknowledge it. The posterior sample size is only 100 accepted values due to a limited computer power available.

The LPX concentration estimate is $2.441 \cdot 10^9$ particles per milliliter. The DNA concentration estimate is $9.13 \cdot 10^9$ particles per milliliter. The ratio of the number of DNAs per one LPX is 1.87. By ABC rejection method with tolerance level of 0.5, using the same summary statistic as with ABC via comparing survivals and with ABC via comparing smoothed distributions, the LPX concentration estimate is $2.509 \cdot 10^9$ particles per milliliter, the DNA concentration estimate is $9.088 \cdot 10^9$ particles per milliliter. The ratio of the number of DNAs per one LPX in this case is 1.811. In [45] this ratio is determined by ABC rejection method using a different summary statistic, a vector with $i^{th}$ component being the number of particles with trajectory durations consisting of $i$ positions in the detection region, and equal to 1.583.
CHAPTER 8
CONCLUSION, DISCUSSION, AND FUTURE WORK

This chapter summarizes the dissertation thesis, discusses its contributions, and outlines the future research.

8.1 Conclusion and Contributions

The contributions of this dissertation are two new algorithms for uncensored data, ABC-EL test and ABC-EL mix; three new algorithms for right censored data, ABC via empirical likelihood, ABC via comparing survival functions, and ABC via comparing smoothed distribution functions; establishing properties for some of the new methods; implementation of ABC via comparing survival functions and ABC via comparing smoothed distribution functions on real data.

It is known that empirical likelihood technique can be used within the ABC framework, see Mengersen [36]. The method works fast, but desire to be more accurate. Panek in his thesis [42] introduced an improvement of ABC via empirical likelihood. In this dissertation a better algorithm ABC via empirical likelihood with testing (ABC-EL test) is developed via adding the hypothesis testing procedure within original ABC-EL algorithm for parameter candidates drawn from the prior distribution. It also works fast, the results of estimation are a lot more accurate, and it is less sensitive to the prior distribution than ABC-EL by Mengersen[36] or Panek [42].

All ABC methods via empirical likelihood are more or less sensitive to the prior information which is crucial in the Bayesian analysis. The combination of ABC via empirical likelihood with testing and ABC-PMC methods is proposed in this work. The ABC-PMC algorithm is targeted on obtaining reliable prior information, which is to be used later within ABC-EL with testing. In simulations, it has been shown that such mix in some cases gives comparable and in others provides a better result of estimation than could be achieved by ABC via empirical likelihood with testing or ABC-PMC algorithm alone. The combination of
ABC-EL for attaining the reliable prior, which is to be used after within some other ABC also have been considered, but the improvement in estimation have not been found.

There are many ABC methods for uncensored data but only one has been recently proposed for censored data. In this thesis three methods suitable for right censored data are proposed. One of them is ABC via empirical likelihood with a mean constraint. The advantage of which is that this algorithm works very fast and more accurate than ABC rejection modified for censored data. It is suggested to be used only when the mean constraint can be determined, which is sometimes difficult to get in practice, and the largest observed value is not censored. Other algorithms are ABC via comparing survival functions and ABC via comparing smoothed distribution functions. In contrast to ABC rejection for censored data, here a more advance way of accepting suitable parameter candidates such as hypothesis testing procedure is applied. As a result, these two techniques provide more accurate parameter estimates when the proportion of censored data is high. For cases with low percentage of censored observations, the outcomes are comparable to the ones provided by ABC rejection. It is more time efficient since the acceptances rates are higher than for ABC rejection modified for censored data. It has been shown that the parameter estimates obtained by these two approaches are asymptotically unbiased and consistent; the resulting posterior distribution gets closer to the true posterior when the sample sizes of observed and simulated data sets get closer to infinity.

ABC via comparing survival functions and ABC via comparing smoothed distribution functions have been applied to a real data set to estimate the particle concentration in a liquid suspension. The results from both methods are very similar.

8.2 Future Work

The future work can be focused on improving ABC via empirical likelihood for right censored data. One of the approaches is to look for the way to improve the prior information by employing other ABC algorithms for right censored data. Another approach is to test the parameter candidate before accepting it and calculating the empirical likelihood weight.

Another direction of future work is developing the ABC algorithms for other type of censored or truncated data such as left censored and interval censored data. The following definitions are taken from Owen
[39]. The random variable $X_i$ is said to be left censored by $Z_i$ to $(−\infty, Z_i)$, where $Z_i$ may be random, if $X_i < Z_i$. $X_i$ is observed if $Z_i \leq X_i$. The random variable $X_i \in \mathbb{R}$ is censored to the interval $(Z_{i,k}, Z_{i,k+1}]$ for $Z_{i,1} < Z_{i,1} < \cdots < Z_{i,K_i}$. ABC via comparing smoothed distribution functions proposed in this thesis is suitable for right censored data. It can be modified for interval censored data if a good estimation of the distribution function for interval censored data is available. The article of Pan on empirical study of estimation of the survival for interval censored data [40] will be helpful.

Interesting area is studying ABC approaches for multivariate data. Recently, Grazian and Liseo [20] presented Approximate Bayesian Computation for copula estimation.
REFERENCES


